Models and Methods for Reflecting and Improving Analytical Tools and Practice: Using Flow, FlowSpaces, $m^3$ [em-cubed], CDG+ and TextWorlds+ for mindful multi-level text analysis

by

Aaron R. N. Smith

M.A., Simon Fraser University, 2007
B.A. (Honours First Class), Simon Fraser University, 2003

Dissertation submitted in partial fulfilment of the Requirements for the Degree of Doctor of Philosophy

in the
School of Interactive Arts + Technology
Faculty of Communication, Art and Technology

© Aaron R. N. Smith 2016

SIMON FRASER UNIVERSITY

Spring 2016

All rights reserved. However, in accordance with the Copyright Act of Canada, this work may be reproduced, without authorization, under the conditions for Fair Dealing. Therefore, limited reproduction of this work for the purposes of private study, research, education, satire, parody, criticism, review and news reporting is likely to be in accordance with the law, particularly if cited appropriately.
Approval

Name: Aaron R. N. Smith
Degree: Doctor of Philosophy
Title: Models and Methods for Reflecting and Improving Analytical Tools and Practice: Using Flow, FlowSpaces, m3[em-cubed], CDG+ and TextWorlds+ for mindful multi-level analysis

Examining Committee: Chair: Jim Bizzochi
Professor

Steve DiPaola
Senior Supervisor
Professor

Barbara Dancygier
Supervisor
Professor
Department of English
University of British Columbia

Mark Blair
Internal Examiner
Associate Professor
Department of Psychology

Kenny Chow
External Examiner
Associate Professor
School of Design
Hong Kong Polytechnic University

Date Defended/Approved: April 1, 2016
Abstract

This Dissertation offers a transdisciplinary integration of several strands of research from human cognition and language understanding across to analytical practice and analytical tool design. This work aims to enable reflective practice for the individual analyst by developing theoretical models, applied methods, and prototype systems that are focused on overcoming core challenges related to deficits in the cognitive and functional capacities of human analysts and their analytical tools. Five integrally related research projects are first grounded by an elaboration of the problem space and a thorough inventory of the concepts, theories, and methods drawn from disciplines that study human cognition and enact analytical practice. The research focus proceeds from a broad view on analysis, through to the often overlooked analytical task of shallow evaluative judgment in the preparatory phase of analysis, and finally to the deeper and under-supported task of analytical reading in the execution phase of analysis. The collective goal of these projects is to understand, capture, and enable the processes of general analysis, shallow evaluation and deep comprehension, by creating models, methods, and systems to interactively capture and reflect these cognitive and functional processes.

At the broadest level, both the Flow Model and the FlowSpaces System aim to enable process-focused reflection on focal activities by theorizing a general analysis process, and by capturing and analyzing workflow focus data. The m³ prototype system and method captures shallow analytical judgments in the preparatory phase of the analysis process as bundles, tags, and highlights. The capture of these metadata objects aims to enable file-focused reflection, on a record of procedural states of files as they are processed through workflows and more generally upon the preparatory process of feature identification and boundary judgment. Finally, to better understand the execution of deeper textual analysis, extended Cognitive Discourse Grammar offers models of discourse context, discourse content, and the dynamics of discourse processing, as well as several novel methods and interfaces for meta-contextual classification of text. These models are the basis for the applied method and proposed system of TextWorlds+ discourse analysis, which captures interactive TextWorld models as external representations of the internal mental models of conceptual structure formed while reading. Interaction with these models aims to enable meta-cognitive reflection upon the internal cognitive processes of language comprehension.
**Keywords:** General Analysis process model; multi-focal multi-tool multi-tasking workflow; reflective practice; discourse processing models; Cognitive Discourse Grammar; Text World Theory; Text Worlds+ Discourse Analysis method
Dedication

I would like to dedicate this work to my family. To my wife MRB and my sons GRS, RWS, and CXS. To my uncle, HP.
Acknowledgements

My gratitude and earnest thanks to Steve D. & Barbara D.

Thanks to Tim H. Thanks to the SCIENCE, NLP, & iViz lab members, whose openness and conversation have been invaluable.

Thanks to others.
# Table of Contents

Approval ......................................................................................................................... ii
Abstract .......................................................................................................................... iii
Dedication ....................................................................................................................... v
Acknowledgements ....................................................................................................... vi
Table of Contents ......................................................................................................... vii
List of Figures ................................................................................................................ xix
Preface .......................................................................................................................... xxxv
Executive Summary ...................................................................................................... xxxvi
Chapter 1 - Introduction .............................................................................................. 1
  1.1 - Challenges .......................................................................................................... 3
    1.1.1 - The Analyst .................................................................................................. 3
    1.1.2 - The Analytical Tool Builder ........................................................................ 4
    1.1.3 - The Sense-Making Model .......................................................................... 4
    1.1.4 - The Functionality of Analytical Tools ....................................................... 5
    1.1.5 - The Functions of Computational Approaches ........................................... 5
    1.1.6 - The Functions of Discourse Methods ....................................................... 5
    1.1.7 - The Functions of Comprehension Models ............................................... 6
    1.1.8 - The Paradox of Transdisciplinary Research ............................................. 6
    1.1.9 - Synopsis of Challenges ............................................................................. 7
  1.2 - Motivation ........................................................................................................... 8
    1.2.1 - Define the Problem Situation at a Human Scale ....................................... 9
    1.2.2 - Capture analyst sense-making datasets for each Level ......................... 10
    1.2.3 - Focus on ‘Rough Dark’ Patches ............................................................... 11
    1.2.4 - Specify Research Production Goals, through Twin Lenses .................. 11
  1.3 - Research Questions ............................................................................................ 13
    1.3.1 - Flow - the Cognitive-Functional Flow Model of General Analysis . 13
1.3.2 - FlowSpaces ................................................................. 14
1.3.3 - m³ - (multi-modal-meta) analytical media management ........ 14
1.3.4 - CDG+ - Extended Cognitive Discourse Grammar ............. 14
1.3.5 - TextWorlds+ ............................................................. 14
1.4 - Research Methodology ................................................... 14
  1.4.1 - Research Products ..................................................... 14
  1.4.2 - Research Goals ....................................................... 15
  1.4.3 - Goals by Problem Context ......................................... 16
  1.4.4 - Research Products and Goals by Project ....................... 17
  1.4.5 - Theory-Based Design goals ....................................... 19
1.5 - Contributions .................................................................. 22
  1.5.1 - Discussion of Primary Contributions ........................... 22
  1.5.2 - Contributions by Chapter ......................................... 25
1.6 - Overview ....................................................................... 28
  Chapter 02. The Problem Situation ......................................... 28
  Chapter 03. The Cognition Literature ................................. 30
  Chapter 04. The Analysis Literature ..................................... 30
  Chapter 05. Flow - The Cognitive-Functional Flow Model of Analysis ....32
  Chapter 06. FlowSpaces - multi-focal multi-application multi-tasking ......33
  Chapter 07. m³ (multi-modal meta) analytical media management ....34
  Chapter 08. CDG+ Extended Cognitive Discourse Grammar .......... 36
  Chapter 09. TextWorlds+ ..................................................... 40
  Chapter 10. Conclusion ....................................................... 43
Part I: Foundations .................................................................. 45
  Chapter 2 - The Problem Situation ......................................... 47
    2.1 - Discursive Challenges of Transdisciplinary Research ............. 49
    2.2 - Cognitive Limits of Human Analysts ............................. 50
      2.2.1 - Cognitive Limits vis. Procedural Complexity .................. 51
      2.2.2 - Cognitive Limits vis. Metacognitive Awareness ............... 51
2.2.3 - Summary for Cognitive Limits of Human Analysts.......................... 52
2.3 - Functional Limits of Analytical Tools........................................... 52
  2.3.1 - Tool Limits vis. Toolmakers’ Expertise.................................... 52
  2.3.2 - Tool Limits vis. Data Scale................................................... 53
  2.3.3 - Tool Limits vis. Data Modality ............................................ 53
  2.3.4 - Tool Limits vis. Data Complexity ........................................ 54
  2.3.5 - Tool Limits vis. Data Types................................................ 55
  2.3.6 - Tool Limits vis. Functional Coverage ..................................... 55
  2.3.7 - Tool Limits vis. Task Model .............................................. 56
  2.3.8 - Tool Limits vis. Analysis Model .......................................... 56
  2.3.9 - Tool Limits vis. Cognitive Externalization............................... 57
  2.3.10 - Tool Limits vis. Application of Cognitive Findings.................... 57
  2.3.11 - Tool Limits vis. Toolmakers’ Knowledge of Real-world Usage .... 57
  2.3.12 - Summary for Functional Limits of Analytical Tools ................. 58

Chapter 3 - The Cognition Literature...................................................... 59
  3.1 - Language Comprehension......................................................... 60
    3.1.1 - Compositional Comprehension........................................... 61
    3.1.2 - Constructivist Comprehension ........................................... 62
    3.1.3 - Discourse Processing Models.......................................... 63
  3.2 - Semantics............................................................................... 65
    3.2.1 - Formal Semantics ............................................................. 66
    3.2.2 - Model-Theoretic Semantics .............................................. 66
    3.2.3 - Cognitive Semantics ....................................................... 67
    3.2.4 - Empirical Frame Semantics ............................................. 67
  3.3 - Knowledge Representation...................................................... 68
    3.3.1 - Logical Representations .................................................. 69
    3.3.2 - Network Representations ............................................... 70
    3.3.3 - Procedural Representations ............................................. 71
    3.3.4 - Frame Representations ................................................... 72
3.3.5 - Propositional Representations ................................................................. 74
3.4 - Reasoning ........................................................................................................ 76
  3.4.1 - Formal Rules for Reasoning ...................................................................... 76
  3.4.2 - Content Rules for Reasoning .................................................................... 77
  3.4.3 - Mental Models for Reasoning .................................................................... 77
  3.4.4 - Discourse Models for Reasoning ............................................................... 79
  3.4.5 - Rich Worlds for Reasoning ........................................................................ 80
3.5 - Decision-making ............................................................................................. 82
  3.5.1 - Classic Decision Making .......................................................................... 82
  3.5.2 - Naturalistic Decision Making .................................................................... 83
  3.5.3 - Recognition Primed Decision Making ....................................................... 83
  3.5.4 - Prototype Recall in Decision Making ....................................................... 84
3.6 - Cognition Synopsis .......................................................................................... 85
Chapter 4 - The Analysis Literature ...................................................................... 88
  4.1 - Cognitive Linguistics ..................................................................................... 90
  4.2 - Discourse Analysis ....................................................................................... 91
  4.3 - Computational Text Analysis ........................................................................ 93
    4.3.1 - Vector space model ................................................................................ 94
      4.3.1.1 - Document indexing ........................................................................ 94
      4.3.1.2 - Term weighting ............................................................................. 94
      4.3.1.3 - Computing Document Similarity Coefficients ............................... 95
      4.3.1.4 - Vector Space Model Summary .................................................. 97
    4.3.2 - Dimensionality Reduction (Projection) Techniques ............................ 97
      4.3.2.1 - Topology Preserving Algorithms ................................................. 97
      4.3.2.2 - Non-linear (Minimization) Projection Techniques ...................... 100
      4.3.2.3 - Hybrid (Linear/Minimization) Projection Techniques ............... 102
    4.3.3 - Summary ............................................................................................. 105
  4.4 - Visual Analytics ............................................................................................ 105
    4.4.1 - Challenges ........................................................................................... 106
5.4 - Recommendations for Analytical Tool-Builders ........................................ 148
  5.4.1 - Recommendation 1: Dedicated Preparatory Analytical Tools............. 148
  5.4.2 - Recommendation 2: Dedicated Analytical Reading Tools ............... 149

Chapter 6 - FlowSpaces ...................................................................................... 152
  6.1 - Overview .................................................................................................... 156
  6.2 - Previous Work ............................................................................................ 157
  6.3 - The Concepts ............................................................................................. 159
    6.3.1 - Tasks ..................................................................................................... 159
    6.3.2 - Activities ............................................................................................... 159
    6.3.3 - Tools ...................................................................................................... 160
    6.3.4 - Tool Usage Cycles ............................................................................... 160
    6.3.5 - Activity Spaces .................................................................................... 160
    6.3.6 - Activity Space Usage Cycles ............................................................... 161
    6.3.7 - Activity, Tool, & Task Focus ............................................................... 161
  6.4 - The Problem ................................................................................................ 161
    6.4.1 - Social & Epistemological Levels ....................................................... 162
    6.4.2 - Technological levels ............................................................................. 162
    6.4.3 - Pragmatic & Grounded Theoretical Levels ....................................... 163
    6.4.4 - Usage Example ..................................................................................... 164
      6.4.4.1 - Apple's Spaces .............................................................................. 165
      6.4.4.2 - HyperSpaces ................................................................................. 174
      6.4.4.3 - DockSpaces .................................................................................... 174
      6.4.4.4 - Workspace Arrangement ............................................................... 176
      6.4.4.5 - Application-Space Assignment ..................................................... 177
    6.4.5 - Integration Example .............................................................................. 177
      6.4.5.1 - Slife ................................................................................................. 178
      6.4.5.2 - Backtrack ....................................................................................... 182
      6.4.5.3 - Growl ............................................................................................. 184
      6.4.5.4 - Keyboard Maestro ......................................................................... 184
8.3.4 - CDG/+ Common Ground ................................................................. 274
8.3.6 - CDG Synopsis ........................................................................ 275
8.4 - The Problems ........................................................................... 276
  8.4.1 - Coverage of Linguistic Phenomena in Generative Linguistics .................................................................................................................. 277
  8.4.2 - Text Worlds Monograph .......................................................... 278
  8.4.3 - Text World Theory .................................................................. 278
  8.4.4 - Static Text World Diagrams ...................................................... 279
  8.4.5 - The Underspecified Method of Text World Analysis .......... 279
  8.4.6 - The High Cost of Text World Analysis .................................... 280
  8.4.7 - Toward a Deeper Understanding of Discourse Processing Tasks. 281
  8.4.8 - World Placement ................................................................... 282
  8.4.9 - How to help where help is needed ........................................ 284
  8.4.10 - The CDG+ Problem Synopsis ............................................. 286
8.5 - The Solutions ............................................................................ 287
  8.5.1 - CDG+ Context Model .............................................................. 288
  8.5.2 - CDG+ Discourse Processing Model ........................................ 296
  8.5.3 - CDG+ Discourse Model .......................................................... 300
  8.5.4 - CDG+ Deictic Matrix ............................................................... 306
  8.5.5 - CDG+ Meta-Contextual Frame ............................................... 310
    8.5.5.1 - Meta-Contextual Frame Design ........................................ 312
    8.5.5.2 - Remote Compression & World Perspective Adoption ........ 338
  8.5.6 - CDG+ Manifest Semantic Analysis ......................................... 343
  8.5.7 - CDG+ Personal Ground .......................................................... 349
  8.5.8 - CDG+ Task Model for Analytical Reading .......................... 350
8.6 - CDG+ Synopsis ......................................................................... 371
Chapter 9 - TextWorlds+, Extended TextWorlds .................................... 377
  9.1 - Overview ................................................................................. 380
  9.2 - The Concepts .......................................................................... 383
  9.3 - The Problem ............................................................................ 384
List of Figures

Figure 0.1. Visual Overview of the Goals, Target Processes, and Products of this Dissertation research. At the broadest level, each research project in this dissertation seeks to understand, capture, and enable specific aspects of the processes of analysis, judgement, and comprehension, by working towards the production of cognitive and functional process models that are useful as methods and implemented as systems.

Figure 1.1. A. Word Clouds or Tag Clouds offer a view on the frequency of terms in a selected text. Cognitive Discourse Grammar explains why unfiltered Tag Clouds emphasize deictic (e.g., the, this, that, a), connective (e.g., of, and), and stative (e.g., is, are, have) terms. Such terms are either determiners or functional predicates that frequently connect arguments within propositions. The skewed frequency distribution of such terms is explained by the nearly one-to-one relation between determiners and entities, and by the ratio of the number of predicates to arguments (e.g., often 2:1), considering the total number of tokens in these deictic, connective, stative, and other relational classes compared to the pool of available entities. Thus, with only a small number of tokens in each deictic, connective & stative class and given linguistic conventions which specify nearly one deictic per entity and two entities for each connective or stative term, the ratio of ‘stop words’ to topical content words is highly skewed. These ‘stop words’ A) are usually filtered out to produce B) a more clearly topical or thematic view.

Figure 1.2. Chapter 5 develops the Function-Argument-Scope Model, which becomes a methodological template for the general approach taken in each subsequent chapter. Functions are of several types, primarily Tasks and Features, which are used to connect two types of Arguments, Resources & Products. These objects, tasks and properties are classified according to their scope as broad, shallow, or deep. Thus, the Cognitive Functional Flow Model of General Analysis identifies and classifies tasks required in the process of general analysis to produce a model of that
process which is further segmented into shallow and deep phases of analysis.

Figure 1.3. Chapter 6 is described in terms of the Function-Argument-Scope Model. This chapter broadly observes and analyzes a user’s focused Tool usage to produce Workflow data and visualizations and to analyze that data to identify and classify transitional patterns.

Figure 1.4. Chapter 7 is described in terms of the Function-Argument-Scope Model. This chapter offers recommendations and prototypes for broadly enabling the management of analytical data, focusing on facilitating the capture of shallow analytical judgments of files as metadata. The background color of this figure draws upon the semantic mapping between Goal/Process and Color in Figure 0.1. Accordingly, this color indicates that the primary goal of Chapter 7 is to better understand the preparatory analytical process by prototyping the m³ system and method to capture shallow analytical judgments as bundles, tags, and highlights. Capturing these data objects is calculated to enabling file-focused reflection on both procedural states of files as they are used in workflows, and specifically upon the process of feature & boundary identification and evaluation.

Figure 1.5. Chapter 8 is described in terms of the Function-Argument-Scope Model. This chapter broadly integrates cognitive, linguistic, and discourse analytical theories and methods, identifying and addressing problems with the original framework and extending the models and methods of Cognitive Discourse Grammar. Per the semantic Goal/Process/Color mapping established in Refer to Figure 0.1, the goal of Ch 08 is to understand the comprehension process through the production of CDG+ models.

Figure 2.1. A Word Cloud view of Chapter 2. A. Frequent Deictic and relational terms. B Frequent conceptual terms.

Figure 3.1. A Word Cloud view of Chapter 3 - The Cognition Literature. A. Frequent Deictic and relational terms, with Author names highlighted. B Frequent conceptual terms.
Figure 4.1. A Word Cloud view of Chapter 4 - The Analysis Literature. A. Frequent Deictic and relational terms. B Frequent conceptual terms.

Figure 5.1. A Word Cloud view of Chapter 5 - Flow. A. This Chapter is highly connective, specifying extensive (e.g., and) and possessive connections and properties (e.g., of), then referential (e.g., the) and purposive relations (e.g., to), containment (e.g., in), and alternation relations (e.g., or) and definitions (e.g., is). B. This Chapter focuses on handling of analytical media in analysis.

Figure 5.2.1. The top level Cognitive-Functional Flow Model of Analysis, an extension of the Sense-Making model. Original Sense-Making model elements are in bold. Primarily cognitive tasks in parentheses are extracted from the articles describing the model (Pirolli and Card 2005) and from the literature on cognition and analysis.

Figure 5.2.2. The major tasks in Cognitive-Functional Flow Model of Analysis, expanded to view their sub-group network of overlapping related tasks. Cognitive-Functional Task Models that specify a required sub-task network and its dynamics must be identified for each major and minor task in the Cognitive-Functional Flow Model of Analysis. Chapters 8 and 9 will thus consider the merits of the Task Model for analytical reading derived from the Cognitive Linguistic discourse processing model, Cognitive Discourse Grammar.

Figure 5.2.2.1. Objects upon which Cognitive Work is executed in the Cognitive Flow Model of Analysis, an extension of the (Pirolli and Card 2005) Sense-Making model. Process elements are action predicates and thus take the arguments listed. This diagram shows the arguments typically taken by process predicates.

Figure 5.3.2.3.1. Analytic Reading: preparatory and executive analytic cognitive functions of analyst readers. (This diagram is read top to bottom and clockwise from the center node.) Figure 5.3.2.3.2. Task Model for analytical reading: Discursive Tasks by discourse participant and Model Mechanism, extrapolated from the CDG+ Discourse Processing Model.
(This diagram is read top to bottom and clockwise from the center node.) These tasks are elaborated fully in Chapter 8, CDG+.

Figure 6.1. A Word Cloud view of Chapter 6 - FlowSpaces. The most frequent terms in this chapter A. without and B. with stop-word removal. The most frequent terms excluded by stop-list filtering are relations that indicate deictic attachment (e.g., the, this, that), ownership (e.g., of), inclusion (e.g., and), alternation (e.g., or), state (e.g., with), location (e.g., in, at, to, on), purpose (e.g., for), conditionality (e.g., if), consequence (e.g., then) and means (e.g., by).

Figure 6.4.4.1. The default OS X Spaces interface offers up to 16 contiguous, unlabeled, undifferentiated, numerically indexed desktop spaces.

Figure 6.4.4.1.1. CMD+Tab Application Switcher

Figure 6.4.4.2. Column A - Launching Exposé shrinks open windows into a labeled matrix. Column B - Selecting from the Exposé matrix zooms back in, with the selected window on the top of the ‘pile’ of open windows.

Figure 6.4.4.3. Column A: Keyboard Shortcut zooms out to Overview Mode. Column B: While in Overview Mode, mouse selection of another Space zooms in to new Space.

Figure 6.4.4.4. When application windows are visible in Spaces overview mode, users may drag and drop single or multiple (using Option-click) application windows between Spaces.

Figure 6.4.4.5. A) Overlapping windows, B) sorted by Exposé while in Spaces.

Figure 6.4.4.6 - Heads up Display (HUD) Style matrix animates transition

Figure 6.4.4.7 - Default interface for assigning Applications to a specific Space

Figure 6.4.4.8. The first menubar icon indicates the user is in Space 4.

Figure 6.4.4.9. The first menubar icon indicates that Spaces is off.

Figure 6.4.4.10. Search, collect, sort, tag; Manage; Read, note; Write; Converse; Convert; Design, develop; Emulate & analyze; Admin; Display.

Figure 6.4.5.1.1. Slife Cocoa interface to add and set targets for Activities.

Figure 6.4.5.1.2. Manual assignment of Applications to Activities by first creating and selecting and Activity, then checking a list.
Figure 6.4.5.1.3. Slife Visualization of Application Usage by Month. Icon indicates App and Color indicates Activity.

Figure 6.4.5.4. Slife Visualization of Application Usage by Day. Icon indicates App and Color indicates Activity.

Figure 6.4.5.3.1. Growl Notification displayed on screen

Figure 6.4.5.5.1. Splunk> integrates semistructured system log data.

Figure 6.4.5.5.2. Proportional Events logged across Activity Spaces

Figure 6.4.5.5.3. Total Events logged across Activity Spaces, (highlighting Analysis in Yellow).

Figure 6.4.5.5.4. Overview of Analytical Views generated in Splunk>.

Figure 6.5.1. The FlowSpaces system-of-systems augments the default OS X Spaces configuration and functionality using HyperSpaces, DockSpaces, Keyboard Maestro & Growl to log digital Application Usage, by Prototypical Task and by current Activity Space Focus. Thus, FlowSpaces is an event-log generating, visually distinct, conceptually labelled, task-specific, tool-collection, and Activity-focusing virtual Desktop workspace system.

Figure 6.5.2. Dragging app windows between task-spaces reclassifies those windows with the users’ current real-time task focus.

Figure 6.7.1. Floating Average of tool usage by Activity Space, Tool, and Task.

Figure 6.8.1.1. Stacked Timeline View of Tool Usage over time. The minutes of the day are on the horizontal axis, while each day in a four-year period is on the vertical axis.

Figure 6.8.1.2. Redundant logging system verified the basic usage patterns over time, though different temporal granularities in the records create different density patterns (the rightmost column does not bin data below a given duration threshold, while the leftmost column does).

Figure 6.8.1.3. Parreto Distribution of aggregated Tool Usage shows proportional contribution of each Application to the overall tool usage.

Figure 6.8.1.4. Parreto Analysis by Activity Space shows the relative contribution of each application to the total App Usage logged in each Activity Space.
Figure 6.8.2.1. Raw App Usage Event Durations by Activity have an extreme range. Density patterns suggest a lower average. In 2012, the collection procedure was extended to capture the activation of Screen Savers. This notably reduced the variance in event durations.

Figure 6.8.3.2. Floating Average of App Usage Event Durations by Last Application and Activity Space. While a linear pattern is evident for continuously and frequently used App Transition Pairs, the extreme variation in referring Applications that are used neither continuously nor frequently produce a background noisy distribution which cannot be resolved into a linear pattern using a floating average.

Figure 6.8.3.2. App Usage Event Durations by Transition Type. First column is Within App (Window) Transitions. Second column is Between App transitions. Third is between Activity Space transitions.

Figure 6.8.2.4. Moving Average duration by Referring App of the Writing app, Scrivener.

Figure 6.8.3.5. Aggregated Activity Spaces Emulate & Analyze, Seek, and Read, Note & Write. Each Aggregated Activity Space exhibits a unique usage pattern.

Figure 6.8.3.6. Floating Average of the most-used Search application. Note its use in both Writing and Analysis & Emulation contexts.

Figure 6.8.3.7. Average Hourly Durations and counts reveal highly transitional Workflows (Application Sequence Pairs) by Activity Space.

Figure 6.10.1.1. Linear model of word count by Application duration

Figure 6.10. Gephi Network analysis of first two years of data as a transition network.

Figure 7.1. A Word Cloud view of Chapter 7 - m3 [em-cubed]. The most frequent terms in this chapter A. without and B. with stop-word removal. The most frequent terms excluded by stop-list filtering are relations that indicate deictic attachment (e.g., the, this, that), ownership (e.g., of), inclusion (e.g., and), alternation (e.g., or), state (e.g., with), location (e.g., in, at, to, on), purpose (e.g., for), conditionality (e.g., if), consequence (e.g., then) and means (e.g., by).
Figure 8.1.A. These most common words, or stop-words, are usually removed from vector space representations of document word frequency. However, many of these words serve an important role of indicating the deictic attachment of a term (determiners) or the semantic connections between terms. B. The top 300 words used in this Chapter.

Figure 8.4.8.1 The Standard CDG Tense and Remote System relationships in Text Worlds.

Figure 8.5.1.1. A collapsed view of the four related types of Situations modelled by CDG+ Context Model. These main situations are also represented within the CDG+ Discourse Model that is produced by the CDG+ Discourse Processing Model.

Figure 8.5.1.2. The CDG+ Context Model expanded to its first level reveals the top-level components of each Situation in the Model.

Figure 8.5.1.3. The CDG+ Context Model expanded to its second level. This reveals the major components of each Situation in the Context Model.

Figure 8.5.1.4. The CDG+ Context Model expanded to its third level.

Figure 8.5.2.1 Conceptual Outline of components in the original CDG model.

Figure 8.5.3.1. An expanded view of the CDG+ Discourse Model in its position relative to the wider context of the CDG+ Context Model. The CDG Discourse Processing Model encompasses the dynamics of the interaction situation that draws exhaustively upon the verbal situation, selectively from the knowledge situation, and only occasionally from the immediate situation.

Figure 8.5.3.2. A proposition asserts a basic situation. It is comprised of a semantic predicate connected to at least one argument.

Figure 8.5.3.3. A Text World and Sub World from the Cognitive Discourse Grammar model of Discourse Processing.

Figure 8.5.3.4. The Common Ground from the Cognitive Discourse Grammar model of Discourse Processing.

Figure 8.5.4.1. The revised CDG+ Deictic Matrix reduces labels and repositions the standard TextWorlds Tense and Remote Deictic System relationships.
Figure 8.5.4.2 The TextWorlds deictic matrix should be viewed as two rotatable rings with adjustable segment bounds. This rotation and adjustment ensures that each Time Zone may overlap with each Epistemic Mode.

Figure 8.5.5.1.1. The CDG+ contextual frame is both a container and a Euclidean metric space that employs Cartesian co-ordinates. Both the interior surface of the ball’s shell and the plane defining its great circle are important surfaces.

Figure 8.5.5.1.2. The CDG+ contextual frame is an extension and formalization of the temporal, proximal (e.g., remote) and modal deictic systems found in standard TextWorlds.

Figure 8.5.5.1.3. In standard TextWorlds, the speakers’ contextual position is called the Origo. However, in standard TextWorlds, the temporal, modal, and proximal deictic systems are not conceived as intersecting to produce both the Origo and an extensible multi-dimensional classification space.

Figure 8.5.5.1.4. While standard TextWorlds does not consider further deictic or contextual dimensions, the extended CDG+ contextual frame accommodates both the major deictic system (e.g., Temporal, Proximal, Modal) and the full range of experiential dimensions of context.

Figure 8.5.5.1.5. The group of mental requisite dimensions of context. Entities, Actions, and Properties classified in this region require a mind.

Figure 8.5.5.1.6. The group of mental non-requisite dimensions of context. Entities, Actions, and Properties classified in this region do not require a mind.

Figure 8.5.5.1.7. The CDG+ contextual frame centers upon prototypical core values of the three major deictic dimensions and extends through to their polar extremities.

Figure 8.5.5.1.8. Now contributes the temporal dimension to the Origo. While actually a single point, this point is expanded to an area for the purposes of illustration.

Figure 8.5.5.1.9. The Present Time Zone occupies the first radial band from the center and thus encompasses and extends beyond the Origo Now point. It is also defined along the transition point between the Past and Future Time Zones at the line demarking the deictic system of Modality.
Figure 8.5.5.1.10. The Past Time Zone occupies the backward half of the second hemiplanar radial band.

Figure 8.5.5.1.11. The Future Time Zone occupies the forward half of the second hemiplanar radial band.

Figure 8.5.5.1.12. The Temporal dimension of deixis with reference to the speaker’s Origo.

Figure 8.5.5.1.13. The Proximal dimension of deixis begins here within the spatial and physical locus of the speakers’ Origo and extends outward through the relatively proximal towards the distal.

Figure 8.5.5.1.14. The Perspectival Proximal Plane, which describes perspectival distance. The major deictic dimension of Modality coincides with the contextual dimension of Perspective. Thus, Perspective, or any other contextual dimension, can be selected in conjunction with the major deictic dimension of Proximity, to specify the kind of proximal/remoteness cline intended for any particular designation of remoteness.

Figure 8.5.5.1.15. The core existential and epistemic modal values situate the speakers’ Origo prototypically with reference to Certain knowledge of Actual events.

Figure 8.5.5.1.16. Probability is the mid range of the Certainty scale.

Figure 8.5.5.1.17. Possibility covers a wide area that includes Actuality, Probability and Improbability.

Figure 8.5.5.1.18. The existential and epistemic Modal dimension of deixis The core values of the modal system extended to their polar extremities.

Figure 8.5.5.1.19. Improbability marks the lower end of the Certainty scale.

Figure 8.5.5.1.20. Impossible events have an extremely low probability, whereas non probabilistic events have no assigned probability.

Figure 8.5.5.1.21. For each core value, opponent values are captured in the second radial hemiplanar band. Lexical opposites may be noted in the third radial band. Negations are stored in the outermost band that intersects with the hemi-planar horizon.
Figure 8.5.5.1.22. The existential-epistemic Modal dimension of deixis prototypically situates discourse contents, with reference to the speakers’ Origo, as Certain knowledge of Actual events. However, many other Modal dimensions exist, such as Truth, Belief, Realization, Verification. In the Meta Contextual Frame, each Modality is attracted to its prototypical contextual dimension. Thus, most modals cluster towards the mental and perspectival dimensions. Some, like doing and realization include a component of making which can be accomplished through physical, corporeal action. Others, like verification, have a strong social component.

Figure 8.5.5.1.23. The core deictic network includes the Origo-defining core values (bold) and their opponent terms. For each of the three major deictic systems of Temporality, Proximity and Modality, the major dimension is defined by both its core and polar extremities, as opponent-core-opponent values. Negations are only included in the outermost radial band as required.

Figure 8.5.5.1.24. The radial bands of the hemi-plane have dynamic labels that take their slot label based on the type of term being classified. Actions are classified into layered temporal-modal featural slots whose labels are based primarily on their temporal values.

Figure 8.5.5.1.25. The Meta-Contextual Frame uses concentric radial bands to classify entities, properties, and relations. States are point samples of underlying conditions, past and present.

Figure 8.5.5.1.26. Manners are ongoing states of action whose span matches that of the action.

Figure 8.5.5.1.27. The CDG+ contextual frame affords position-based classification in multiple dimensions simultaneously. Each point on the hemi-plane denotes a unique 5-ary value that specifies a temporal, existential\epistemic, experiential\contextual and proximal value assignment.

Figure 8.5.5.2.1. The user swipes upward from the Origo to compress the hemi-plane into a remote parallel plane.
Figure 8.5.5.2.2. Dragging upward moves the workspace against a virtual spring that provides resistance and settles at a 70% of the original hemiplane, occluding only the rear quadrant.

Figure 8.5.5.2.3. Contextual ground for internal ‘remote’ sub-world.

Figure 8.5.1 Interface for the Manifest Semantic Analysis prototype. Each term of each definition is classified by part of speech and assigned a colour-coded dimensional class.

Figure 8.5.9.1. The Discursive Tasks of Discourse Participants in The CDG+ Context Model.

Figure 8.5.9.2. The Discursive Tasks of Discourse Participants by Participant and Mechanism in the CDG+ Discourse Processing Model.

Figure 8.5.9.3. The Task Model of Analytical Reading. Speakers begin the Discourse.

Figure 8.5.9.4. The Task Model of Analytical Reading. Readers accept the topic.

Figure 8.5.9.5. The Task Model of Analytical Reading. Readers identify propositions.

Figure 8.5.9.6. The Task Model of Analytical Reading. To identify propositions, Readers segment utterances.

Figure 8.5.9.7. The Task Model of Analytical Reading. To identify propositions, Readers evoke relevant information; by activating Frames, decomposing propositional elements, and obtaining resemblance relations.

Figure 8.5.9.8. The Task Model of Analytical Reading. To evoke relevant information, textual cues activate Frames, from which Readers select relevant Frames or slot values (or update slot values).

Figure 8.5.9.9. The Task Model of Analytical Reading. To evoke propositions, Readers decompose propositional elements, which are used to Search CDG Common Ground to establish propositional coherence, and to search the Activated Frames to obtain resemblance relations.

Figure 8.5.9.10. The Task Model of Analytical Reading. To evoke relevant information, Readers obtain resemblance relations by collecting
situational features and comparing these against frame situational features.

Figure 8.5.9.11. The Task Model of Analytical Reading. To evoke relevant information from activated Frames, Readers accept matching resemblance relations between current and Frame situation features. Accepted matches permit the selection of relevant frames and slot values.

Figure 8.5.9.12. The Task Model of Analytical Reading. To identify propositions, Readers infer unstated propositions, which create SubWorlds.

Figure 8.5.9.13. The Task Model of Analytical Reading. To manage propositions, Readers integrate, accommodate, evaluate, negotiate, identify, and add propositions.

Figure 8.5.9.14. The Task Model of Analytical Reading. To manage propositions, Readers integrate propositions regardless of their source. Readers accommodate embedded information in matrix propositions. Readers evaluate propositional coherence and truth. Readers negotiate propositional meaning. Readers identify propositions belonging to Figure and Ground. And Readers add propositions to appropriate World levels.

Figure 8.5.9.15. The Task Model of Analytical Reading. To manage propositions, Readers integrate propositions regardless of their source in the text, evoked from Frames, or inferred by the reader.

Figure 8.5.9.16. The Task Model of Analytical Reading. To manage propositions, Readers accommodate embedded information in matrix propositions by marking that information as given, and incrementing it into the CDG Common Ground.

Figure 8.5.9.17. The Task Model of Analytical Reading. To manage propositions, Readers evaluate propositional coherence and truth.

Figure 8.5.9.18. The Task Model of Analytical Reading. To evaluate proposition coherence, Readers compare current propositions against those in the CDG Common Ground by evaluating functional relations between them.
Figure 8.5.9.19. The Task Model of Analytical Reading. To evaluate propositional coherence, Readers mark new information using the emphasis placement mechanism.

Figure 8.5.9.20. The Task Model of Analytical Reading. To evaluate propositional coherence, Readers update coherent information reference chains, deictic counterparts, and TextWorld predications.

Figure 8.5.9.21. The Task Model of Analytical Reading. To evaluate propositional coherence, Readers track reference, by searching the entity register of the CDG Common Ground and the World Building Elements of each Text World to identify deictic shifts that require the creation of new Text Worlds, and actions or description chains that require updating.

Figure 8.5.9.22. The Task Model of Analytical Reading. To evaluate propositional truth, Readers assess each terms’ truth probabilistically and assess speaker’s authority.

Figure 8.5.9.23. The Task Model of Analytical Reading. To negotiate propositional meaning, Hearers may request clarification or more information from Speakers.

Figure 8.5.9.24. The Task Model of Analytical Reading. To negotiate propositional meaning, Hearers search for implications of further information provided by the Speaker.

Figure 8.5.9.25. The Task Model of Analytical Reading. To negotiate propositional meaning, Readers accept propositional meaning and increment accepted propositions into the CDG Common Ground.

Figure 8.5.9.26. The Task Model of Analytical Reading. Readers identify propositions belonging to Figure and Ground.

Figure 8.5.9.27. The Task Model of Analytical Reading. To identify propositions belonging to Ground, Readers classify propositions as belonging to the class of world-building elements, which includes times, locations, enactors, entities, and modal expressions.

Figure 8.5.9.28. The Task Model of Analytical Reading. To identify propositions belonging to Figure, Readers classify propositions as belonging to the class of function-advancing propositions, which includes actions,
processes, modifications, descriptions, and RST-style functional relations between propositions (Mann and Sandra Thompson 1988; Taboada 2006).

Figure 8.5.9.29. The Task Model of Analytical Reading. To manage propositions, Readers add propositions to their appropriate world level.

Figure 8.5.9.30. The Task Model of Analytical Reading. Readers create representations of conceptual structure, or mental models, in their minds.

Figure 8.5.9.31. The Task Model of Analytical Reading. To specify representations as Worlds, Readers create TextWorlds for actual events and SubWorlds for epistemically divergent events.

Figure 8.5.9.32. The Task Model of Analytical Reading. To create TextWorlds, Readers add propositions to the appropriate zone, or update world-building elements and function-advancing propositions, or shift TextWorlds as deictic disjunctures are detected.

Figure 8.5.9.33. The Task Model of Analytical Reading. To create SubWorlds, Readers add propositions to the appropriate world zone, or update world-building elements and function-advancing propositions.

Figure 8.5.9.34. The Task Model of Analytical Reading. An overview of the entire CDG+ Task Model for Analytical Reading.

Figure 9.1.A. These most common words, or stop-words, are usually removed from vector space representations of document word frequency. However, many of these words serve an important role of indicating the deictic attachment of a term (determiners) or the semantic connections between terms. B. The top 300 words used in this Chapter.

Table 9.5.1.1.1 The determination of basic Time Zone and specific verbal Aspect is a stepwise comparison between Speech time, Reference time and Event time.

Figure 9.6.1 Text World 1. Inferences positions TW1 as an actual current report. Figure 9.6.2 Sentence 1. Propositions are chunked and their relations are labeled.
Figure 9.6.3 Text World 2. The deictic matrix shifts to accommodate the unfolding presentation of past actions. The CASTLE frame is evoked by a single term ‘treasury’.

Figure 9.6.4 Common Ground as of Sentence 1. Entities figuring in multiple actions or descriptions become containers for any subsequent actions and descriptions involving that entity and its coreferences.

Figure 9.6.5 Sentence 2. Propositions are chunked.

Figure 9.6.6 Text World 2, S2. Inferred alternation and internal aspect shift are indicated by discourse marker badges connected to soft-bounded containers.

Figure 9.6.7 Sentence 3. Chunked propositions.

Figure 9.6.8 Text World 2, Sentence 3. Inferences based on a negated graded comparison Sub World attach to ‘the man’ in TW2. The ANGER frame underscores and explains the destructive attack of S1 and momentary appearance state in S2.

Figure 9.6.9 Sentence 4. Chunked propositions.

Figure 9.6.10 Text World 2 S4. Not all propositional relations are clearly marked with discourse markers for each argument. Inference is often required to label each argument and propositional relation.

Figure 9.6.11 Text World 2 Sentence Five (S5). The entire TextWorld description is collapsed into the node ‘everything.’ In CDG+, the integration of RST inter-propositional analysis adds a degree of complexity not found in the original CDG approach.

Figure 9.7.1.1 Overview of preprocessing workflow design in Orange.

Table 9.7.1.1 The TextWorlds method includes eight tasks that include fifteen steps.

Figure 10.4.1. Stacked Timeline View of Raw Tool Usage Events over time. The minutes of the day are on the horizontal axis, while each day in a four-year period is on the vertical axis.

Figure 10.4.2. Space 1 Seek, Space 2 Collect, Space 3 Sort, Space 4 Tag, Space 5 Manage, Space 6 Read, Space 7 Note, Space 8 Write, Space 9 Converse, Space 10 Convert, Space 11 Design, Space 12
Develop, Space 13 Emulate, Space 14 Analyze, Space 15 Admin, Space 16 Display.

Figure 10.4.3. All App Usage Events on a single timeline aggregated with a 21-day floating average.

Figure 10.4.4. All App Usage Events by continuous usage, multi-app usage, and task-transitional usage, and by frequency-weighted grouping of Activity Spaces.

Figure 10.4.4. App Usage Events filtered for continuous single-app usage in a focused single-tasking Activity Space. Most continuously used Spaces are Search, Manage, Converse, Analyze, and Write.

Figure 10.4.5. App Usage Events filtered for transitional multi-app usage in a focused single-tasking Activity Space. By sum of Event Counts, the most used Spaces are Search, Manage, Collect, Converse, Write, Analyze, Note, Emulate, Admin, Display.

Figure 10.4.6. App Usage Events filtered for transitional multi-app multi-tasking usage across multi-tasking Activity Spaces. This pattern shadows that of focused multi-app usage.
Preface

The genre of the Dissertation supplies its own range of scholarly expectations, including those specific to its ideal form and format, which are reasonable given the traditional specialization of focus within a given discipline. In the current context of trans-disciplinary scholarship presented in this Dissertation, certain of these expectations require amendment in order to accommodate the necessity of a broader scope, and multiple focal areas that are intrinsic to transdisciplinary approach. The reader will note a considered response to the challenge of trans-disciplinary scholarly writing, in Chapter 2, the Problem Situation, and will find that the author has concluded that it is better to provide foundational detail in the body of the Dissertation than in an Appendix. Thus, the first four Chapters establish the theoretical, conceptual, and methodological foundation for the research projects described in Chapters 5 through 9.

Readers with scholarly expertise in the domains of cognition or analysis are encouraged to skim these foundational sections, which are included for the benefit of those readers whose expertise hails from different quarters. Beyond this extensive effort to actively establish, rather than presume, a trans-disciplinary common ground for this integrative research, the presentation of this research includes extensive visual modelling. External visual representations and models of processes, situations, and data are central to the reflective purpose of this work. Over 125 full-page and half-page thumbnail images are included in this text, with the original higher-resolution (Ultra-HD 4K) versions available on-line (Permanent Archival URL: http://ivizlab.sfu.ca/thesis/smith). Thus, my commitment to establishing foundational detail and providing visual detail accounts for nearly half of the volume of this Dissertation by page count.

A note about stylistic format choices. Bold is applied to concepts to foreground them for analysis with current VA tools. Italics is often applied to descriptive elements, and also on second mention of an important concept. This is a done as a means of tagging entities.
Executive Summary

This dissertation is a compendium of integrally related research projects that proceeds from a broad overview of analysis, through to the shallow and often overlooked analytical task of evaluative judgment in the preparatory phase of analysis, and finally to the deeper and under-supported task of analytical reading in the execution phase of analysis. Each research project is described in its own Chapter.

Chapter 1, Introduction, introduces the problem space, summarizes its Challenges, Motivations and Contributions, and thus provides an Overview of this Dissertation Research Programme. It introduces Research Questions for both general Research Programme and specific Research Project and briefly outlines the methodological and design objectives of each Research Project.

Part I, Foundations, provides the conceptual background as the common foundation to ground the Research Projects reported on in Parts II through IV.

Chapter 2, The Problem Situation, elaborates upon Challenges that serve to Motivate each Research Project that was briefly introduced in Chapter 1, Introduction.

Chapter 3, The Cognition Literature, offers unfamiliar analysts and analytical tool-builders a selective discussion of literatures from the research domains of Cognition, in order to clarify the role of certain domains of higher-level cognition in the analysis process in general and in the analytical task of analytical reading in particular.

Chapter 4, The Analysis Literature, offers unfamiliar analysts and analytical tool-builders a curated discussion of literatures from the pragmatic domain of Analysis, to identify the tasks involved in the analysis process in general and in text analysis in particular.

Part II, Modelling Cognitive-Functional Analytical Process, focuses at the highest level on the whole analysis process, identifying gaps in standard models of this process, repairing those gaps, and producing a novel method for examining digital functional tool-usage flow dynamics in the context of activity and task focus.

Chapter 6, FlowSpaces, presents a novel method and system for capturing real-time digital tool-usage and user-interface-usage data, converting semi-structured system-log-based textual data into multi-tool multi-tasking workflow patterns, and reflecting workflow patterns back to the tool-builders and tool-users, toward building a predictive model for workflow and window states.

Part III, Supporting Ubiquitous Shallow Analytical Judgment, moves focus from the highest level to the intermediate level, where many focused and cross-cutting preparatory analytical tasks are under-supported in current analytical tools.

Chapter 7, m³ [em-cubed] (multi-modal-meta) analytical media manager, introduces a prototype system of systems is designed to facilitate, capture and visualize the process of conceptualization and judgment that occurs initially while preparing digital content for reading and eventually also while actually engaging in shallow reading of digital analytical media.

Part IV, Modelling Conceptual Structures in Deep Analytical Reading, moves from the intermediate level to the deepest level of focus, where a single challenging, under-supported analytical task, analytical reading, is selected to build towards improved, comprehensive support of this task and each of its requisite and related sub-tasks.

Chapter 8, CDG+, reviews the origins, rationale, defence and repair of the CDG approach to discourse processing and modelling. This Chapter presents a comprehensive and detailed view of each component and dynamic of the extended Cognitive Discourse Grammar Discourse Processing Model.

Chapter 9, TextWorlds+, lays out the Goals, Features, Method, and Exemplar of analysis using the TextWorlds analytic method to produce TextWorlds as models of the users’ discourse comprehension process.

Part V, Discussion, presents the deeper arguments behind the overall Research Programme and offers further elaboration upon each of the common Challenges, Motivations, Research Questions and Contributions that situate each Research
Project that were briefly introduced in Chapter 1, Introduction. This discussion highlights the Findings, Recommendations and proposed Future Work toward improved analytical tool designs from each Chapter.

Chapter 10, Conclusion, reviews the deeper arguments supporting this Research Programme, elaborating with further Discussion of the findings and recommendations produced by each Chapter, and concludes summarizing the original Contributions of this dissertation.

While each research project is described in its own Chapter, each project shares a common framework with a core set of Goals, objects of study (e.g., Procedural Phases), and methods of producing results (e.g., Products) (See Figure 0.1).

Figure 0.1. Visual Overview of the Goals, Target Processes, and Products of this Dissertation research. At the broadest level, each research project in this dissertation seeks to understand, capture, and enable specific aspects of the processes of analysis, judgement, and comprehension, by working towards the production of cognitive and functional process models that are useful as methods and implemented as systems.
Each project is thus distinguished by a unique combination of Goals and Target Phases, and has consequently produced related but unique Research Products. The collective goal of this research is to understand, capture, and enable the related processes of general analysis, shallow evaluation, and deep comprehension.

To better understand the Analysis process, focusing on General All-Source Data Analysis in particular, Chapter 5 develops the theoretical Cognitive-Functional Flow Model as refinement and extension to the standard Sense-Making Model of Analysis. Continuing this focus on the overall Analysis process, Chapter 6 designs and implements a method and system for nearly transparent observation of multi-tool multi-focus multi-tasking, captures these functional phenomena as a longitudinal workflow dataset. Chapter 6 also analyses these data and produces recommendations for future research and implementation. Both the Flow Model and the FlowSpaces system aim to enable process-focused reflection, to facilitate and improve meta-cognitive reflection on the user’s tool usage process and optimize workflow sequences.

The primary goal of Chapter 7 is to capture the preparatory phase of the analysis process. Thus, the m³ prototype system and method proposes the capture of shallow analytical judgments as bundles, tags, and highlights. The capture of these data objects is calculated to enable file-focused reflection, first on an easy record of procedural states of files as they are processed through workflows and then, more specifically, upon the process of feature identification and boundary judgment (e.g., classification) in the preparatory phase of analysis.

To better understand the execution phase of deeper textual analysis, Chapter 8 introduces the extended version of Cognitive Discourse Grammar (CDG+) which includes models of discourse context, discourse content, and the dynamics of discourse processing, as well as several novel methods and interfaces for meta-contextual classification of text. These models are the basis for the method and system of TextWorlds+ discourse analysis which is described and exemplified in Chapter 9. TextWorlds models are interactive external representations of the mental models of conceptual structure formed while reading that are externalized and captured using the TextWorlds+ method. The primary goal of enabling meta-cognitive reflection upon internal cognitive processes is made possible by constructing and interacting with TextWord models to capture sense-making.
Chapter 1 - Introduction

This Dissertation offers a transdisciplinary integration of several strands of research across human cognition and language understanding, analytical practice and analytical tool design. The methods and theory developed here strive to establish and maintain focus on the fact that, at the center of each of these research domains, and at the core of all these integrated perspectives is the pragmatic and cognitive human being; a language user and digital tool user, who is an active reader, sense-maker, comprehender, thinker, and analyzer.

Traditionally, the role that best describes this human at the center is the analyst. The analyst is typically a disciplinary specialist who knowledgeably selects from a pool of data-appropriate processing methodologies to produce practical, strategic insights. Across select domains, this dissertation identifies domain independent features of this analysis process, noting that general analysis typically iterates among various cognitive-functional steps along paired continuums of depth and completion, that range from the shallow analysis required for preparation through the deeper analysis involved in the execution of traditional analytic activities. Thus, in this definition, general analysis is simultaneously a goal-focused adjudication and application of relevant methods to multi-modal data; and a context-relevant packaging of analytical products.

In this context, analytical expertise is supported by both background 3.3.4 - Frame Representations, personal experiential knowledge (Glenberg 1997) and procedural knowledge (Kang and Stasko 2014; Hutchins and others 2007; Pirolli and Card 2005; Gevamay and Wildavsky 1997; Russell and others 1993; Wildavsky 1987). Knowledge of goals includes both objectives and metrics for success (Gevamay and Wildavsky 1997; Russell and others 1993). Knowledge of data includes both its features and context. Knowledge of methods includes both preparation and execution of data processing tasks. Consequently, any analyst’s ability to productively engage in the process of analysis can be limited by insufficient knowledge of relevant goals, data, or methods.
Figure 1.1. A. Word Clouds or Tag Clouds offer a view on the frequency of terms in a selected text. Cognitive Discourse Grammar explains why unfiltered Tag Clouds emphasize deictic (e.g., the, this, that, a), connective (e.g., of, and), and stative (e.g., is, are, have) terms. Such terms are either determiners or functional predicates that frequently connect arguments within propositions. The skewed frequency distribution of such terms is explained by the nearly one-to-one relation between determiners and entities, and by the ratio of the number of predicates to arguments (e.g., often 2:1), considering the total number of tokens in these deictic, connective, stative, and other relational classes compared to the pool of available entities. Thus, with only a small number of tokens in each deictic, connective & stative class and given linguistic conventions which specify nearly one deictic per entity and two entities for each connective or stative term, the ratio of ‘stop words’ to topical content words is highly skewed. These ‘stop words’ A) are usually filtered out to produce B) a more clearly topical or thematic view.
1.1 - Challenges

This Dissertation research addresses a number of specific challenges at various levels, for a range of interested parties belonging to several audiences. In their most simplified state, these specific Challenges can be generalized as Cognitive Limits of human analysts and Functional Limits of analytical tools. Both are discussed in detail in Chapter 2, The Problem Situation. This Section begins by briefly highlighting Challenges that are specific to core individuals, frameworks, and methods in this problem space. The following sections thus discuss Challenges relevant to individuals, including The Analyst and the Analytical Tool Builder; Challenges posed by frameworks like the Sense-Making model of analysis; and Challenges specific to methods and procedures including the Functionality of Analytical Tools, the Functions of Computational Approaches to Text, the Functions of Discourse Methods, and The Functions of Comprehension Models. The section concludes with a brief consideration of the Paradox of Transdisciplinary Research. Each of these Challenges is discussed briefly in turn below, and are further elaborated in Chapter 2, The Problem Situation.

1.1.1 - The Analyst

The first and primary challenges concern the analyst. The analyst’s core challenges relate to various difficult tasks in the analysis process. Analysis is not easy and analytical tools cannot always help. This is a problem, because the job still has to be done. Beyond the challenges of doing the work, the analyst is further challenged whenever they are asked to describe their work retrospectively. In such cases, they typically describe the simplest aspect of their work, because the real complexity of the work is not easy to relate, nor even to keep in mind beyond the moments in which the flow of tasks is accomplished. Analysts forget as they move from one context to the next (Radvansky and others 2011; 2010; Radvansky and Copeland 2006), and even forget important discoveries (Czerwinski and Horvitz 2002) and where they were in a process (Salvucci 2010; Mark and others 2005). Moreover, doing analytical work while meta-cognitively reflecting upon that work and articulating that reflection makes hard work that much more difficult (Hoffman and others 1995). And on the rare occasion that the analyst can be convinced to
briefly make their work harder for the sake of science, the talk-aloud protocol and meta-cognitive reflection process produces very specific results (e.g., in this situation, for these goals, with this data I chose this method, encountered these problems, and resolved them this way, to come to this result, and deduce this conclusion) that don’t necessarily generalize.

1.1.2 - The Analytical Tool Builder

Next there is the analytical tool builder. Their primary challenges relate first to specific knowledge deficits regarding The Analyst and their process. Specifically, analytical tool builders do not know what analysts do, how analysts actually use tools to complete their work, which tools they actually use, and what functions analysts would use tools to accomplish if those tools existed, were better, or easier to use. When tool builders don’t know a domain, they typically try to ask the expert. There are methods for such knowledge elicitation processes, though not all analytical tool builders are qualified for this specialized techno-social inquiry. Thus, when the target analyst is not available for comment, when The Analyst cannot articulate their process, or when the tool builder cannot conduct original research themselves, analytical tool builders usually turn to a standard procedural schematic, the notional Sense-Making model of Intelligence Analysis, as a specification for the analysis process.

1.1.3 - The Sense-Making Model

So next, we consider Challenges raised by the Sense-Making model. Its primary challenge relates to the comprehensiveness and the focus of its coverage of analytical tasks in the analysis process. Though no procedural schema can ever offer perfect coverage, such process models should aim to be comprehensive. Since the Sense-Making Model does not reflect all the tasks that The Analyst must do for analysis, and moreover since it does not highlight or rank specifically challenging tasks in the process, when analytical tool builders use this model as a specification for the analysis process, they build analytical tools that neither offer adequate support across all analytical tasks, nor provide focused support the most challenging analytical tasks.
1.1.4 - The Functionality of Analytical Tools

Next, there is the set of functions offered by analytical tools, and particularly by Visual Analytical tools. The purpose of analytical tools is to facilitate analytical procedures for analysts. Ideally, analytical tools should help analysts across all of the tasks required to accomplish the process of analysis. In this case, the individual functions or tasks accomplished by digital tools should ideally match the tasks identified in the Sense-Making model. However, not only are some tasks from the Sense-making model not well supported by current analytical tools. In some tools, certain tasks are not supported at all. So not only do analytical tools not offer support across the entire Sense-Making Model of the Analysis Process, but they don’t support analytical tasks that are absent in that model. Ideally analytical tools should help with difficult tasks — work that the analyst finds most difficult (Thomas and others 2007; Wong and others 2006) — where that difficulty is supported by the known strengths of digital tools (Pirolli and Card 2005).

1.1.5 - The Functions of Computational Approaches

Then, there is the set of functions offered by computational analytical approaches in various domains. The challenge here is knowing, from among the great diversity of algorithmic, computational, and manual approaches to various data types available across many domains, which will be most suited to the problem data and task. For example, not everyone knows the strengths of computational approaches to language. Only certain people, like computational linguists, know that computers are now able to do certain language classification tasks well and inexpensively using Natural Language Processing algorithms and techniques. But not all analytical tool-builders know what is possible (or cheap) to accomplish using NLP tools.

1.1.6 - The Functions of Discourse Methods

Next, there is the set of functions offered by discourse analytical methods. Not everyone knows the strengths of discourse approaches to human language. Only certain experts, like discourse analysts, know what to look for and how to identify
the cues, features, parts and relations of discourse. And not all analytical tool-builders know what is required to produce a Discourse Analysis.

1.1.7 - The Functions of Comprehension Models

And then there is the set of functions offered by discourse processing models. Not everyone knows how to evaluate the strengths of discourse comprehension models of human language. Only certain people, like discourse psychologists, know what cognitive and functional steps humans take in the process of coming to understand human language in the real world. And not all analytical tool-builders know what is required to produce and update discourse comprehension.

1.1.8 - The Paradox of Transdisciplinary Research

And finally, there is the author of this research and the reader. Transdisciplinary scholarship often presents a real challenge for both readers and writers. (This challenge is detailed in Section 2.1, Discursive Challenges of Transdisciplinary Research.) In sum, the paradox of transdisciplinary scholarship is that its texts are simultaneously written for multiple audiences, yet processed by a single reader. Consequently, transdisciplinary writers are constrained to first making specific rhetorical choices in order to establish a broad foundation (Part I, Foundations) based on the (unknowable) quantum state of reader topic knowledge, and only then may they proceed to build an argument on that quantum foundation. Each functional shift (e.g., from a didactic to an expository mode, and from thence to argumentation) is a necessary gamble taken to improve the writer’s odds of potentially altering the knowledge state of the novice reader while reinforcing a shared foundation with the knowledge state of the expert reader, and is necessary to establish the conceptual connections required to follow the arguments.

Evaluation of transdisciplinary writing would appear to be almost as challenging as its reading. Since few readers are experts in each domain, evaluation of transdisciplinary texts must begin with a direct and continuous assessment of their goals. Ultimately, the reader must also consider the quality of the argumentation provided and the originality of the contributions. Thus, in the case of this dissertation, readers must determine for themselves how well this work extends cognitive theory of analysis; how it reveals cognitive-functional analytical practice;
how it supports analytical expertise, and how well it exemplifies analytical creativity. To facilitate ongoing reader evaluation, Research Questions and goals are discussed extensively in the introduction and are reviewed periodically throughout the text.

Practically speaking, all of this means is that, regardless of the reader’s initial knowledge state, the act of reading amounts to agree to accompany the author on a journey, first to explain the process through which the author came to discover the problem situation, and then to describe the ways forward that are proposed here. Thus, the price of entry into this guided tour of the author’s dissertation path is a bit of learning (or review) at regular intervals until the identifiable domain-specific terms are sufficiently fleshed out for standard discourse assumptions of a shared common ground to resume. The path forward begins here.

1.1.9 - Synopsis of Challenges

To review, analytical tool-builders are not necessarily analytical experts, computational linguists, discourse analysts, or discourse psychologists. And yet they need these particular domains of expertise to know what is possible (or cheap) to accomplish using NLP tools, what is required to produce a Discourse Analysis, what is necessary to produce discourse comprehension. And tool builders require an integrative perspective to see how all of this can be useful to their task of building analytical tools. This Dissertation offers both a background in these various domains and provides the sort of integrative perspective that is required to put all this together to build something new, and to recommend how other tool builders may do likewise.

The ability to address the broad range of challenges specified above clearly require either an interdisciplinary team of experts, or an individual using a transdisciplinary approach, drawing upon a wide knowledge base, and employing a broad, multi-focal skill-set. The Dissertation Research Program presents five related Dissertation Research Projects that take the latter approach.

The challenges discussed in this section (beyond those inhering transdisciplinary discourse) can be summarized as the obvious facts that analysts face difficult tasks in the analysis process (e.g., externalizing the results of classification and reading), which are not supported by clear gaps and omissions in
published understanding of the analysis process, and shortcomings in the match between tasks in the analysis process and tasks supported by analytical tools. Thus, it should now be clear that two core problems define two levels in my research. These include Cognitive Limits of human analysts, and Functional Limits of analytical tools and the questions of how best to make each address and support the other.

For a deeper discussion of these two core problems, the reader is referred first to Chapter 2, Section 2.2, Cognitive Limits of human analysts in relation to the Procedural Complexity of the analysis process, and to the Meta-cognitive Awareness of internal knowledge states and digital procedural states. Next, the reader is referred to Section 2.3, Functional Limits of analytical tools, which offers an extensive consideration of specific Challenges classified generally in relation to 1) Gaps in Toolmakers’ Expertise & Knowledge, 2) Challenging Data Features, and 3) Poor Functional Coverage of Tasks in the Standard Analysis Model. Section 2.3 will discuss specifically, for current analytical tools, how limitations in these general classes relate to 1) the shortfalls in Analytical Toolmakers’ Expertise in Eliciting Tool Requirements and in their Implementation of Relevant Findings from Cognitive Science, and the shortfalls in Analytical Toolmakers’ Knowledge of Real-World Analysis Process, and of Real-World Tool Usage and of; 2) Difficulties with Data Scale, Data Data Modality, Data Complexity, and Data Type, and 3) Overall Gaps in the Functional Coverage of Task Models, of Analysis Models, and particularly a lack to analytical tools focused on the challenging task of Cognitive Externalization.

In summary, this Dissertation research provides the integrative perspective required to show how all of this knowledge of cognitive and analytical process can be made useful, and how, in fact, it turns out to be essential to building analytical tools that support under-supported tasks in real-world general analysis of text.

1.2 - Motivation

In response to these detailed challenges, the clear motivation of these Research Projects is to build toward improved theory and tools that overcome unacknowledged cognitive and functional task-gaps in analytical tools.
1.2.1 - Define the Problem Situation at a Human Scale

At the most general level, the ‘human scale’ focus of this research underscores a desire to make a contribution toward understanding the cognitive and functional tasks required at each step of the analysis process. In this research, this contribution is initially met by contributing to the knowledge of digital tool usage workflows; specifically, of which tools to use with which data, what procedures are required to prepare data for each tool, and how to both prepare and execute the required analytical work. This general level of research focus is provided by Chapter 6, FlowSpaces.

At an intermediate level, the human scale focus means understanding that the preparatory phase of analysis involves basic cognitive analytical tasks (e.g., shallow judgments) which are not sufficiently supported in current Visual Analytics tools. This level of contribution is therefore met by building means and mechanisms to more effectively capture and reuse this shallow preparatory sense-making, in Chapter 7, m³ [em-cubed].

At the deepest level, this human scale focus means understanding, supporting, and modelling the language comprehension process that unfolds as the analyst reads the text. This level of research focus is provided by the Chapter 8, CDG+, and Chapter 9, TextWorlds+.

Thus, taking a human scale view of this large problem space naturally takes a multi-level focus that moves from understanding the whole, to its widest parts, and then to its deepest parts. Accordingly, the report on the overarching Dissertation Research Programme begins in Part I by establishing the Foundations and introducing the background domains of Cognition and Analysis. Then, in Part II, Modelling Cognitive-Functional Analytical Process, the research focus upon analysis at the highest, general level of cognitive functional procedural models and the collection of a task-tool usage workflow dataset. In Part III, Supporting Ubiquitous Shallow Analytical Judgment, the research moves to a focus on analysis in intermediate, preparatory level, which includes discovering ways to support cross-cutting cognitive applications of ubiquitous shallow judgments. Finally, in Part IV, Modelling Conceptual Structures in Deep Analytical Reading, the focus moves to the deepest level of language comprehension during analytical reading.
1.2.2 - Capture analyst sense-making datasets for each ‘Human Scale’ Level

The common thread in my research across each of these levels of Research Focus is an effort to identify and refine methods and technologies to record and model analyst sense-making activities; generally, while working with digital tools in both the preparation and execution of analysis, and specifically while working with digital texts at shallow preparatory and deeper analytical levels. This common thread works at each level to addresses Research Question 2, How can transparent, low-cost observational methods be designed to produce empirical datasets that capture functional and cognitive processes?

Consequently, most research and development time has involved designing and automating methods and technologies for the transparent collection of three types of personal dataset. Each dataset joins a deep individual focus with broad, multi-year event coverage to produce and analyze deep data on detailed user activities. These datasets are personalized collections of digital tool workflow, document and metadata annotation, and personal language comprehension.

1) Personal workflow data is generated through a combination of manual and computational pattern recognition and event-extraction from computer-generated log files that record user activity while using applications across specific activity classes. This system is described in Chapter 6, FlowSpaces.

2) A prototype system for curating personal multi-modal meta + analytical media data collections (e.g., annotated personal document corpora data) integrates each multi-modal document selected for collection over time with its objective external metadata and shallow personal annotation data. This prototype is described in Chapter 7, m³ [em-cubed].

3) Eventually, the revised and extended Cognitive Discourse Grammar+ Model of Discourse Processing, which is elaborated in Chapter 8, CDG+, will be implemented using natural language processing and machine learning to accomplish computationally tractable steps of the manual method for incrementally building TextWorlds as conceptual models of reader comprehension. This method is introduced and exemplified in Chapter 9, TextWorlds+. The computational implementation of this method should eventually enable users to build and interact with dynamic, personalized sense-making models. Until then, manually generated models of the
conceptual structures generated while reading provide personalized data on the detailed comprehension process.

Each dataset joins a *deep individual focus* with *broad, multi-year event coverage* to produce and analyze *deep data* on *detailed user activities*. These datasets are personalized collections of *digital tool workflow, document* and *metadata annotation*, and *individual language comprehension*.

### 1.2.3 - Focus on ‘Rough Dark’ Patches

The editors and contributors of the *Visual Analytics* research agenda, *Illuminating the Path*, would certainly agree that it is not sufficient, having designed a street light, to ask passersby how well it illuminates the ground beneath, and upon a receiving a nod, to conclude that the path has been illuminated. Instead, tool-makers must understand the paths that users travel in their entirety and pay special attention to those *rough dark patches* for which no illumination has yet been designed. This dissertation research, and particularly the *FlowSpaces* project, can help tool makers see the wider context of tool usage, and may even be used to help identify these ‘rough, dark patches’ in analytical tool usage and functionality. This focus on tools and methods for identifying ‘rough, dark patches’ discussed at length in the Chapter 6, *FlowSpaces*.

This dissertation has leveraged the authors professional experience with the class of analytical tools produced in the field of *Visual Analytics* to identify *analytical reading* as a particularly rough and dark patch, essential unsupported in this class of tools. This research therefore focuses on identifying and supporting the requisite functional and cognitive tasks that are necessary steps in the preparation and analysis of the difficult modality of text. This focus is discussed at length in Chapter 7, *m³ [em-cubed]*, Chapter 8, *CDG+*, and Chapter 9, *TextWorlds+*.

### 1.2.4 - Specify Research Production Goals, through Twin Cognitive and Functional Lenses

Overall, at the three levels of research focus identified above, this dissertation focuses on understanding and identifying the analysis process at the level of *cognitive tasks* (e.g., internal cognitively executed tasks) and *functional tasks* (e.g.,
general classes of both digital and analog actions undertaken in the process of analysis).

At the broadest cognitive level, the focus is on understanding and identifying the analysis process at the level of cognitive and procedural functions identified in the literature on cognition and analysis (Chapter 5, Flow). At the broadest functional level, the focus is on understanding and identifying the analysis process at the level of digital task functions as revealed by digital tool usage across dedicated Task Spaces (Chapter 6, FlowSpaces).

At the intermediate cognitive level, the focus is on understanding and identifying the cross-cutting shallow analytical tasks at the level of specific tasks required to accomplish shallow analytical reading in the preparatory phases (Section 5.3.1, Preparation Phase). At the intermediate functional level, the focus is on each preparatory step in the analytical reading process, particularly at the level of identifying and prototyping tools required to capture shallow analytical judgment during shallow reading in the preparatory phase (Chapter 7, m3 [em-cubed]).

At the deepest cognitive level, the focus is on understanding and identifying the analytical reading process at the level of specific tasks required to accomplish analytical reading in the execution phase of analysis (Chapter 8, CDG+). At the deepest functional level, the focus is on operationalizing the analytical reading process. This level involves many tasks. The first requirement is identifying computational means of accomplishing tasks required to do analytical reading in the execution phase of analysis. The second requirement is to design a method or protocol to execute and externalize this sequence of requisite reading tasks as a conceptual model of textual contents. Finally, in future work, the research will build a computational reading system for interacting with the reading model through interactive Situation Indexed Event Indexed TextWorlds. (Chapter 9, TextWorlds+)

Thus, this research proceeds on the premise that the helping with currently-under-supported, difficult analytical tasks like analytical reading requires a new, transdisciplinary approach that integrates knowledge of this challenging cognitive task from domains in Cognitive Science (Chapter 3) and particularly from Discourse Processing (Section 3.1.3) into the design of analytical tools.
Likewise, the broader objective of assisting analysts in the externalization of their cognitive and functional processes, will almost certainly require a new transdisciplinary knowledge-integrating approach to designing analytical tools capable of capturing and reflecting this cognitive and functional procedural data back to the user. Thus, this research proceeds on the premise that the generation of useful, detailed descriptions of these linear, real-world process chains (e.g., analytical reading and shallow analytical judgment) requires a new approach to capturing and reflecting this data back to the user. This research is thus best described as novel, integrative theory-based design.

1.3 - Research Questions

This Dissertation reports on a broad, multi-level Research Programme that encompasses five related Research Projects. Each project approaches the problem situation - understanding and supporting the process of general text-focused analysis - from a successively closer perspective, through progressively closer lens. Each Research Project in my Dissertation Research Programme - Cognitive-Functional Flow Model of General Analysis, FlowSpaces, m³ (multi-modal-meta) analytical media management, contributions to the original Cognitive Discourse Grammar as CDG+, including the CDG+ Task Model for Analytical Reading produced by mining the CDG+ Model of Discourse Processing, and the extended manual and computational TextWorlds+ Discourse Analysis methods) are grounded in the following two transdisciplinary questions:

1) How can cognitive theory and analytical practice inform the design of analytical tools?

2) How can transparent, low-cost observational methods be designed to produce empirical datasets that capture functional and cognitive processes?

Beyond these two General Research Questions, each Research Project Chapter may also have a major and minor focus questions related to that specific level of research:

Flow - the Cognitive-Functional Flow Model of General Analysis

In what cognitive, functional, and procedural areas can the Sense-Making model be extended to better characterize the process of general analysis?
• How can the Sense-Making model be extended into a general model of analysis process?

FlowSpaces

In what functional areas does the general analyst use digital tools?
• What does general analysis look like over time?
• What tool usage patterns can be identified over time?

m³ [em-cubed] - (multi-modal-meta) analytical media management

In what functional areas can analytical tool-builders improve support of under-supported phases of analysis?

CDG+ - Extended Cognitive Discourse Grammar

What is Cognitive Discourse Grammar, what are its shortcomings, and how can it be improved for implementation in analytical tools to support analytical reading?

TextWorlds+

How can models of human discourse processing leverage existing advances in computational approaches to language analysis to build interactive models of human conceptualization?

1.4 - Research Methodology

This research method followed here focuses on producing Research Products to meet specific Research Goals which address Core Problems in Specified Contexts. I argue that the production of these Research Products to the following specifications will establish a way of addressing the Core Problems in the following specific Contexts, and thus answer both general and focus Research Questions for each Research Project.

1.4.1 - Research Products

Overall, this dissertation focuses on identifying and supporting the functional and cognitive tasks that are requisite to the preparation and analysis of the difficult modality of text. To this end, this research aims towards producing Research
Products in three basic classes: 1) cognitive and conceptual models; 2) observational and analytical methods; and 5) observational and analytical prototype tools.

1.4.2 - Research Goals

My basic Research Products are further specified with requirement that the production of models and tools must provide certain missing and preferred features. These requirements thus provide focused constraints on the production of Research Products. These requirements are offered below as a list of targeted Research Goals.

The first Research Goal is to extend a cognitive model to include:

1. The cognitive and functional tasks and cognitive-functional flow dynamics involved in the general analysis process that are missing from the extending the Sense-Making Model of Intelligence Analysis.

2. The cognitive and functional tasks and dynamics involved in the analytical reading process that are missing from the cognitive-functional Flow Model of General Analysis.

The second research goal is to build a conceptual model for representing the mechanisms and dynamics of:


2. The CDG+ Model of Discourse Processing from the original Text Worlds approach.

3. The CDG+ Task Model for Analytical Reading from the CDG+ Model of Discourse Processing.

The third research goal is to build prototypes and build toward analytical tools for:

1. Supporting each analytical task across both preparation and execution phases of the analysis process, especially challenging and currently under-supported analytical tasks.

2. Supporting the process of externalizing internal cognitive processes and models.
3. Supporting the process of understanding ongoing shallow evaluative judgments and of capturing these shallow evaluative judgments as annotations.

4. Supporting the process of understanding ongoing language comprehension processes and capturing the dynamics of deeper semantic integration of reader knowledge with text contents.

5. Supporting the process of understanding ongoing procedural states and annotating digital workflow states.

6. Capturing, analyzing, and visualizing the process of using digital tools in multi-tool multi-tasking sessions as nested within functional space-usage sessions.

1.4.3 - Goals by Problem Context

The first core problem regarding the Cognitive Limits of human analysts that exists in relation to the Procedural Complexity of the analysis process is addressed by the goal of building toward analytical tools that help with the entire preparation and execution process of (currently un-supported and) challenging tasks in the analysis process. This will help with externalizing the products shallow analytical judgements and deeper analytical reading.

In particular, in relation to the analysts’ Meta-Cognitive Awareness of their own internal knowledge states, this first core problem is addressed by the goal of building toward analytical tools that help with the process of externalizing internal cognitive processes. This will help facilitate meta-cognitive reflection on the process of building Mental Models & Schemas and making analytical judgments during analysis.

Also, in relation to the analysts’ Meta-Cognitive Awareness of their ongoing digital workflow procedural states, this first core problem is addressed by the goal of building toward analytical tools that help with the process of understanding ongoing procedural states and annotating digital procedural states. This will help with way-finding in the ongoing analysis process, and by identifying workflows to facilitate reuse in the future.

The second core problem of Functional Limits of analytical tools exists in relation to Analytical Toolmakers’ Knowledge of Real-World Analysis Process, and is addressed by the goal of extending the Sense-Making model of intelligence
analysis to include **cognitive-functional flow dynamics** and **missing cognitive and functional tasks**. This will help by identifying gaps and friction points in the coverage of tasks within the analysis process to use as leverage points for the improved design of analytical tools.

Likewise, relative to **Functional Coverage of the Analysis Model**, the second core problem is addressed by the goal of **building analytical tools** that help with under-supported **real-world cognitive and functional analysis processes**. This will help by focusing on supporting analytical judgement and reading tasks that are not currently (well) supported in analytical tools.

Next, in relation to **Analytical Toolmakers’ Knowledge of Real-World Analytical Reading**, the second core problem is addressed by the goal of extending the **Cognitive-Functional Flow Model of Analysis** to include the **CDG+ Task Model for Analytical Reading** based upon the **CDG+ Model of Discourse Processing**. This helps by specifying and automating the specific cognitive and functional tasks and dynamics involved in the analytical reading process.

Finally, in relation to **Analytical Toolmakers’ Knowledge of Real-World Tool Usage** the second core problem is addressed by the goal of **building an analytical tool to capture, analyze, and visualize all tool-usage sessions as nested within functional space-usage sessions** to help with characterizing the functional flow of tools between tasks both within and beyond analysis.

### 1.4.4 - Research Products and Goals by Project

This section lists specific products of this research, by research project.

**Flow** and the **Cognitive-Functional Flow Model of General Analysis**,

- Designs a revised cognitive-functional flow model of the analysis to address gaps in tool-builder knowledge.
- Designs a dynamic cognitive-functional flow model of the reading process at shallow and deep levels to address gaps in tool-builder knowledge and tool-user meta-cognitive self-knowledge.

**FlowSpaces,**

- Builds a novel task logging system for capturing task/tool usage to record events as text-based log data.
- Builds a novel text analysis system for converting semi-structured text to extract events from event-log data.
- Conducts a longitudinal observational case study of multi-tool multi-tasking to collect data on tool/activity workflows.
- Conducts a visual analytic case study in exploratory data-analysis to identify functional transitions among tool usage events.

m³ [em-cubed] (multi-modal meta) analytical media management,

- Prototypes a system-of-systems for preparatory document analysis to facilitate shallow analysis of texts.

CDG+ or Extended Cognitive Discourse Grammar,

- Catalogues a detailed inventory of terminological, conceptual, and functional shortcomings of the original CDG approach to focus the production of novel contributions to CDG theory.
- Designs a pragmatic CDG+ Contextual Model for the CDG+ Discourse Processing Model to situate each model mechanism theoretically in levels.
- Designs an itemized model component visualization of the CDG+ Discourse Model to index the model features exhaustively.
- Designs a discursive CDG+ Task Model of the dynamics of discourse participants’ actions to visualize the model dynamics.
- Revises a cognitive discourse processing model of deep analytical reading to present the theoretical viability of mechanisms.
- Designs an analytical reading task model to specify reading sub-tasks to update the Cognitive-Functional Flow Model of Analysis.

In TextWorlds+ or Extended Text World Analysis,

- Designs both manual and computational methods of representing conceptual structures toward implementing interactive models of reading.
- Executes a manual TextWorlds case study of conceptual model generation to exemplify the TextWorlds process of analysis.

Thus, to summarize the specific Research Products listed above, this Dissertation Research Programme has conducted a series of related Research Projects that have produced 1) a theoretical model of general analysis (e.g., cognitive-functional Flow), 2) a data collection system-of-systems, 3) a hand-coded data fusion methodology, and 4) a novel exploratory analysis of a unique
longitudinal case study workflow dataset that has produced 5) recommendations for improving tool usage and design (e.g., FlowSpaces). This dissertation also presents two complimentary approaches to addressing two particularly ‘rough dark patches’ that Visual Analytics has yet to illuminate; specifically in the ‘shallow’ 6) preparatory phase of collection, classification, & resource management (e.g., m3 [em-cubed]) and in the ‘deep’ 7) execution phase of analytical reading (e.g., CDG+ and TextWorlds+).

1.4.5 - Theory-Based Design goals

These novel research methodologies produced by this Dissertation Research Programme are a product of the tension between facts, requirements, and potentials. These existential modalities², of actuality, necessity, and possibility are determined by discovering what is done, what must be done, and what could be done.

What is done is reflected as daily routines, which are defined as short series of tasks that accomplish some goal. Once a series of tasks has been repeated, it becomes a routine. Thus, repetition of tasks is explained as productivity. Routines are not always optimal, but they are productive.

What must be done is reflected at either ends the scale of necessity as optimized subroutines and methodological descriptions. At the smaller end of the scale, not only are tasks productive, but they are arranged so as to be efficient. On a larger scale, given a series of goals, a series of optimized subroutines can be designed to meet overarching practical and theoretical goals.

What could be done is reflected as projections into a system’s design so as to implement a methodological description that interrupts daily routines and replaces them with optimized subroutines that collectively achieve overarching practical and theoretical goals. Practical goals here include reducing the cost-structure of routines, like making a task easier or take less time. Theoretical goals include building specified structural or procedural features into the system design that enable achievement of previously unobtainable goals.

On the most pragmatic level, the goal of this Research Programme has been to design and prototype a system of systems (SoS) (van de Laar and others 2013) that
supports and automates the analytic process along the continuum from shallow to deep analysis of multi-modal data including text. Thus, the first three chapters focus on establishing and refining the theoretical and pragmatic basis for the human-information Discourse Analysis that occurs as analysts make sense of multi-modal analytical information, and particularly as analysts read.

To this end, the Literature Chapters assemble & articulate the cognitive & linguistic theory for modelling interaction between analysts and information during moments of analyst-driven human-information discourse. Chapters on m⁴ [em-cubed] and TextWorlds+ describe my progress towards designing & developing applications that provide technological scaffolding to model, facilitate, and visualize the dynamic process of analytical reading.

The envisioned SoS integrates two major kinds of functionality in several related areas; 1) digital resource organization as applies to the management of i) analytical media and ii) metadata, and 2) digital resource annotation, in the forms of iii) ubiquitous file tagging, iv) shallow color-based semantic role highlighting and v) deeper content analysis and annotation.

Ultimately, this research concludes that what analytic readers urgently require is a digital analytic environment³ that prioritizes reading and annotation and affords analyst readers with novel interactions to enable multi-level annotation and layered textual analysis. In the ideal system of systems both deeper reader analysis and surface or shallow analytical judgments like classification will unfold supported by a systematic, coherent, and theory-based toolset.

Multi-level annotation encompasses several complimentary analytical and annotation methods over the course of a multi-modal data analysis workflow. At the highest level of object granularity and the lowest level of analysis, surface analysis involves the determination of features of media containers and generalized typifications of its contents. These annotations are applied to the whole document as metadata. Typical document-level metadata includes Bibliographic information, Library of Congress & Dewey classifications, publisher or vender data, and social media metadata. This metadata is typically collected from online sources and managed in a meta-database but can also be stored in simple tags. Thus, the first level of all metadata has the original analog or digital document container (e.g., The paperback or PDF file) as its object.
Shallow analysis at finer levels of granularity range in depth from identifying and annotating major document-structure components (e.g. Headers, Chapters, Sections, subsections); or its language-structure components (e.g. Parts of speech, verb tense & aspect.); to the deeper rhetorical-structure functional relations that hold between contiguous phrases; and finally, the deepest conceptual level of the situations and interconnections of the things, features, and actions that occur within each scene or situation described in a text.

As yet, none of these levels of analytic annotation are unproblematic for either manual or computational analysis. And while the scope of this work is broad, many of the key components are currently in place — some exist in a working prototype system⁴ for shallow analysis and the remainder are synthesized from the literatures discussed in Chapters 3 and 4, and again in more depth in Chapter 8, CDG+.

The long-term goal of this Research Programme is the methodological design and computational development of a procedural algorithm to implement and extend the Cognitive Discourse Grammar model and methods of text analysis first described by Paul Werth (1999). Building this analytic discourse comprehension model into a usable visual discourse analytic application not only requires a deep familiarity with the model and method but also with basic text pattern recognition and language processing methods, while implementation using standard frameworks that leverage established natural language processing algorithms requires the capacity to code in Java.

In the first of several productive parallels between the broad and fine levels of my research, the capture and conversion of semi-structured system-log-based textual data into multi-tool multi-tasking workflow patterns in FlowSpaces has also required extensive writing and testing of countless regular expressions in java-like code.

Ultimately, the effort required to build and refine a system for processing four years and hundreds of gigabytes of log files — billions of events of raw data — into a usable, analyzable user workflow dataset has proven to be a dissertation-sized project in itself. This dissertation thus reports on progress
1.5 - Contributions

Consequent to the production of these Research Products, I argue that the Research Programme reported here provides three primary Contributions.

1. This research identifies unacknowledged gaps (e.g., missing tasks) in the standard Notional Model of Analyst Sense-Making (a.k.a The Sense-Making Model of Analysis or the Think-Loop Model of Analysis) and reconceptualizes the analysis process as the cognitive-functional Flow Model of General Analysis, and produces a unique dataset on the on transitions between apps and tasks in digital tool workflows by function and Activity focus.

2. This research identifies one broad unacknowledged gap in the coverage of analytical tools, namely, the support and capture of shallow analytical judgments. Visual Analytics tools typically ignore the richly productive, cross-cutting, shallow analytical judgments made throughout the preparatory phase of analysis. Since this shallow-cross-cutting analytical sense-making could have value at subsequent points in the process, this research prototypes m³ [em-cubed] (multi-modal meta) Tagging tools and proposes Analytical Media Management tools to capture shallow analytical judgments, classifications, and evaluations as tags and other multi-modal meta-data as annotations. This prototype tagging system is designed to work with any analytical tools, and provides a common storage point for simple Tag annotations on all data files of any type using any analytical tool, at any point in the analytical workflow.

3. Finally, this research identifies one deeper unacknowledged gap, and so focuses on understanding, modeling, and supporting the difficult task of analytical reading. This research supplies a cognitive task model that can specify required sub-tasks for the task of analytical reading and uses this CDG+ Task Model of Analytical Reading to produce recommendations for building an analytical reading tool that will actually support the required and related sub-tasks of analytical reading.

1.5.1 - Discussion of Primary Contributions

Since tool makers are not necessarily analysts; since analysts are not always accessible and since analysts are not always able to explain the internal cognitive and digital functional aspects of their process, the tool-building discipline of Visual Analytics, generally employs the approach of using a model, like Pirolli & Card's
Sense-Making model of analysis, as a specification for building analytical tools. In Part II: Modelling Cognitive-Functional Analytical Process, I have drawn on my experience as a working analyst using various analytical tools to work through this model. I have reflected deeply on this model of analysis and identified problems and improvements.

Most professional analyst would agree that analysis includes task that are not explicitly included in the sense-making model of analysis. Nevertheless, analyst must still accomplish such tasks by whatever means necessary. Thus, to identify missing analytical tasks I have returned to the literature on the analysis process and to specific areas in Cognitive Science and even to the original published text of the Sense-Making model. First, however, reflecting on my experience as an analyst, I offer a reconceptualization of the analysis process in both cognitive and functional terms, wherein each analytical task is conceptualized as a composite bundle of related and requisite cognitive and function sub-tasks. Moreover, both the overall analysis process and individual analytical tasks have both preparatory and execution phases. Thus, in Chapter 5, Flow, I use these deep reflections and gap observations to produce the Cognitive-Functional Flow Model of General Analysis as a revision and reconceptualization of the Sense-Making Model of Intelligence Analysis. I then go on to characterize digital tool usage broadly in functional terms in Chapter 6, FlowSpaces.

Considering only analytical tasks from these unacknowledged gaps, I selected one broad yet shallow unsupported task upon which to focus in Part III, Supporting Ubiquitous Shallow Analytical Judgment. I thus consider the simple cross-cutting task of analytical judgment both conceptually and functionally, in order to identify how better support for this type of classification and evaluation task could be built into analytical tools. For this broad and shallow case at the intermediate level, I built a prototype system-of-systems by integrating, extending, and customizing tools for applying tags & other annotations. The goal at this level is to facilitate capture of shallow analytical judgments in a manner & form that could potentially benefit the ongoing work of analysts in practical ways (Chapter 7, m3 [em-cubed]).

Next, also considering only unsupported analytical tasks from the unacknowledged gaps, in Part IV, Modelling Conceptual Structures in Deep Analytical Reading, I selected the deep, difficult task of analytical reading to
consider conceptually and functionally, in order to identify how improved support for this pivotal analytical task could be built into analytical tools. For this case, I spent a long time considering theories and models of this task from various disciplines beyond Visual Analytics. VA offers little support and no theoretical insights into this task. Accordingly, I have constructed one cognitive and functional Task Model of analytical reading, by extending a model of Online Discourse Processing (e.g., language comprehension) that offers both an explicitly cognitive-functional approach, a focus on visualizing the models produced by readers, and a well-developed set of mechanisms for visually representing the dynamics of reader comprehension within the model. This model is called Cognitive Discourse Grammar (CDG). CDG’s original Text Worlds approach details how readers produce rich mental models of text contents while reading.

Analysts and tool builders in the field of Visual Analytics will not be familiar with the CDG model of human discourse processing, though the original approach of CDG is a recognized and widely published as an analytical approach in the literary analysis fields of stylistics analysis and poetics. Even linguists and discourse analysts are not likely to be universally aware of its strengths, features, utility, and possible applications beyond the field of stylistics. For these reasons, this dissertation devotes a significant amount of time and space to the description, extension, and application of Cognitive Discourse Grammar. I have identified shortcomings in the original CDG approach, and provide a revision and extension to the original CDG approach, which I call CDG+.

In the research projects reported here, not only do I introduce the benefits and mechanics of CDG and CDG+ to several new audiences which would seem to have a clear need for it, I also offer several contributions to the development CDG as a theoretical research paradigm. I note and repair specified shortcomings. I position CDG as a fully fledged Discourse Processing Model. I produce a range of illuminating visual conceptual models of the context surrounding the CDG+ Discourse Processing Model, its components, and emergent dynamics. I create a new interactive classification rubric for deictic & multi-dimensional contextual Meta-Classification called the CDG+ Meta-Contextual Frame to situate TextWorlds and TextWorld contents along Major Deictic Dimensions and minor contextual dimensions. I also extend the original approach into a manual method for
externalizing interactive TextWorld+ Models of conceptual contents of text that are produced as mental models in the reader’s mind while reading.

For the purposes of transdisciplinary application of CDG+ in Visual Analytics, I specify a Task Model for analytical reading, based on the CDG+ Discourse Processing Model, and report on progress made toward building this Task Model into a computational model and method for pre-processing texts using natural-language processing.

The main contributions at this deep level thus include introducing CDG+ to the field of Visual Analytics, defending the selection of CDG+ as a plausible model of language comprehension, identifying and improving upon its shortcomings, providing an applied exemplar of its function, and reflecting on how CDG+ can be implemented to provide robust support for analytical reading within analytical tools.

Future work will prototype tools for automating production of TextWorld Models and facilitating interaction with conceptual models formed while reading, to facilitate capture of deep semantic and conceptual structure in a manner & form that could potentially benefit the ongoing work of analysts in practical ways (Chapter 9, TextWorlds+).

1.5.2 - Contributions by Chapter

This section summarizes the discussion of the Contributions, indexed by Chapter. In order to address deficits in the Analytical Toolmakers’ Knowledge of Real-World Analysis Process, I have, in Chapter 5, Flow, extended the Sense-Making model of intelligence analysis to include cognitive and functional tasks from the literature on analysis and cognition, as the Cognitive-Functional Flow Model of General Analysis.

Next, in order to address deficits in both 1) the Meta-cognitive Awareness of Analysts’ ongoing digital procedural states, and 2) the Analytical Toolmakers’ Knowledge of Real-World Tool Usage, I have, in Chapter 6, FlowSpaces, developed a novel data-capture & processing methodology and implemented a novel system-of-systems for capturing real-time digital tool-usage and user-interface-usage data as a necessary precursor to reflecting workflow patterns back
to the analyst in a **visual analytical system**, and back to the toolmaker as an exemplar of functional tool evaluation.

Then, in order to capture evaluations that may help the analysts’ mindful or *Meta-cognitive Awareness* of both 1) *their own internal knowledge states*, and 2) *their progress in ongoing digital procedural states*, I have, in Chapter 7, m3 [em-cubed], prototyped and implemented a novel system-of-systems for capturing **shallow judgments** of file features, and for annotating digital procedural states to help with way-finding in the ongoing analysis process, both using the simple mechanism of **tags**.

Also, in order to capture evaluations that may help the analysts’ *Meta-cognitive Awareness of their own internal knowledge states* in their progress through shallow reading of file contents, I have (in Chapter 7, m3 [em-cubed]) built a novel colour-based taxonomy for capturing deeper judgments of file contents, using the simple mechanism of highlights. Also, I have designed for future implementation a novel dynamic colour-picker, wherein drag&drop into hierarchical coding categories automatically-assigns new concepts an incrementally unique colour, based the reader’s assignment of the new concept to one of 10 core dimensional categories. Hue, Saturation, and Brightness are each further tied to specific model dimensions, such that a precise colour assignment is equivalent to the attribution of a specific multi-dimensional feature set. Also designs are proposed for future implementation, include a highlight-linker, to enable the specification of relations between highlighted ‘phrases’ in a text, a file binder to manage duplicates, versions, and near-duplicate analytical media, and a metadata leveller to create bidirectional updates between metadata management applications.

Then, in order to address deficits in the Analytical Toolmakers’ Knowledge of Real-World Analytical Reading, I have, in Chapter 8, **CDG+**, extended the Cognitive-Functional Flow Model of Analysis to include a Task Model for analytical reading based upon the **CDG+ Model of Discourse Comprehension**. This will help with specifying and in future work automating the specific cognitive and functional tasks and dynamics involved in the analytical reading process. This current specification also produces the manual Method TextWorlds Analysis, which can be used as a means of manually capturing, modelling, and visualizing the dynamic
construction of personal conceptual structures (e.g., mental models & schemas) in manually created, interactive, digital visualizations called TextWorlds.

Finally, in order to address the analyst’s Meta-cognitive Awareness of their own internal knowledge states during analytical reading, I have (in Chapter 9, TextWorlds+, and made progress towards the future implementation of a unique method and system of capturing real-time online reading comprehension data, as a means of capturing, modelling and visualizing the dynamic construction of personal conceptual structures (e.g., mental models & schemas) in an interactive visual analytical system. Specifically, I have designed the Situation Model, the manual interactive digital TextWorld visualizations (e.g., interactive diagrams), and the Meta-Contextual Frame for deictic and contextual meta-analysis, and exemplified their usage.

So, in these various ways, this work makes scholarly contributions to the empirical understanding of analysis, to the pragmatic availability of technology, to the specification of analytical methodology, and to the extension of discourse processing and discourse analytical theory. This work contributes to scholarly understanding of real world analysis. It contributes the design and prototypes of a system of systems for analytical preparation, and points towards the future implementation of an interactive analytical reading system that can be interactively personalized to each analysts’ requirements. Finally, it contributes to methodology, extending and refining methods of both deep and shallow reader-driven analysis of text. If analysts actually do read analytically, then, for as long as analytical reading remains a core process for digital readers and analysts, this research should remain relevant to reading analysts by offering a practical method of externalizing internal mental models. These methods can be found nowhere else. Future work will solidify the relevance of the methods refined and extended here by reducing the cost of using this analytical method.

This section has outlined my objectives for making academic contributions to each domain discussed in the following chapters. Chapter 2, The Problem Situation establishes how the disciplinary foundations of my research draw upon particular subfields in the general study of Cognition and Analysis; particularly from the domains of Discourse Comprehension, Reasoning, Decision-Making, and Knowledge Representation, Visual Analytics, Intelligence Analysis, Discourse
Analysis, Semantics, and Cognitive Linguistics Analysis. This section has introduced the most significant contributions made to these domains by my dissertation research.

Specifically, practitioners in domains of analysis discussed in Chapter 4, The Analysis Literature, will benefit from externalizing and revising mental models generated while processing and analyzing text. The practitioners of text world theory approach to Discourse Analysis and those in the broader field of Cognitive Linguistics Analysis more generally should appreciate both theoretical and methodological improvements to pragmatic tools for engaging in these methods of text-focused analysis.

1.6 - Overview

The final section of this Chapter 1, Introduction, provides a Chapter by Chapter preview, as a detailed roadmap of the overall research journey. Thus this section can be used as a rough outline of the related Research Projects undertaken for this Dissertation.

Chapter 02. The Problem Situation

Chapter Two, The Problem Situation, provides the extended consideration of two dimensions of the problem situation. First, there is challenge inherent in all research that spans disciplinary boundaries. Next, there are the classes of problems addressed and resolved by the research through the development of prototype tools, data collections, and empirically-based recommendations for analytical tool design.

Thus, Chapter 2 first elaborates upon the Paradox of Transdisciplinarity, in order to reframe the reader’s view of the genre, change expectations, and suggest selective reading or skimming for familiar material. Next, Chapter 2 elaborates on the two Core Problems in the analytical process problem space addressed by this research, namely, the Cognitive Limits of human analysts, and Functional Limits of analytical tools.

Chapter Two begins by elaborating upon the unique discourse situation, first introduced in Chapter 1, Introduction, in which both writers and readers of
transdisciplinary research find themselves. In transdisciplinary writing, because the writer is focused on several audiences, they cannot make standard assumptions about the reader. Thus, extensive introduction and review of basic concepts is necessary prior to elaboration of arguments. Advanced focused reading strategies are offered to facilitate reading perseverance for domain experts.

Next, this Chapter highlights the challenges facing modern analysts and points out established problems in the field of Visual Analytics. These challenges primarily regard difficult tasks in the analysis process (e.g., like externalizing the results of classification and reading); gaps and omissions in published understanding of the analysis process; and shortcomings in the match between tasks in the analysis process and tasks supported by analytical tools.

In Chapter Two, each Core Problem is discussed in an eponymous Section. The first, Section 2.2, Cognitive Limits of Human Analysts, first considers cognitive limits that can be attributed to the Procedural Complexity of the analysis process. The next relevant subset of human cognitive limits are viewed as a consequence of the variation over time in the analyst’s Meta-Cognitive Awareness of their own internal knowledge states and their current procedural states in digital analytical workflows. Thus, low-cost methods of reducing the complexity of the analysis process (e.g., by combining tool functionality or by facilitating export trajectories between analytical tools) are proposed. Additionally, low-cost methods of capturing the analyst’s internal knowledge states and current procedural states as tags, are also proposed (e.g., by indexing current procedural states).

Next, Section 2.3, Functional Limits of Analytical Tools, offers an extensive consideration of specific shortcomings in analytical tools. These shortcomings are attributed to 1) Deficits in Toolmakers’ Expertise & Knowledge, 2) Incommensurable Data Features, and 3) Gaps in Functional Coverage of Tasks in the Analysis Model.

Specifically, for current analytical tools, limitations in these general classes relate to 1a) the Analytical Toolmakers’ Expertise in Eliciting Tool Requirements and 1b) in their capacity as regards Implementation of Relevant Findings from Cognitive Science, and 1c) the Analytical Toolmakers’ Knowledge of Real-World Analysis Process, and 1d) of their knowledge of Real-World Tool. Functional Limits of analytical tools can also be attributed to 2) technical challenges dealing with
Data Scale, Modality, Complexity, Type, and 3) Functional Coverage of the general Analysis Model, and of specific Task Models, and particularly of tasks to facilitate Cognitive Externalization.

Chapter 03. The Cognition Literature

The purpose of this Chapter is to quickly move the reader through a small selection of the broad literature of Cognitive Science. So this Chapter offers a selective discussion of the specific primarily research-focused domains in Cognition, in order to clarify the role of selected domains of higher-level cognition in the general process of analysis and analytical reading. This Chapter is also an obvious and convenient location to define domain-specific terms used that will be used throughout the discussion. Thus, this Chapter 3, The Cognition Literature, selects from the broad areas of Cognitive Science, Artificial Intelligence, Semantics, and Discourse Psychology, with particularly attention for literatures that contribute to theories of possible worlds, schemas, scripts, and mental models. The research domains in focus here will be Semantics and Language Comprehension, including Discourse Processing, and Reasoning, primarily from the perspective of mental models.

This Chapter focuses briefly on these five research areas, to consider some of their major dynamics and mechanisms, and to begin to identify how these mechanisms contribute to the transdisciplinary research questions addressed in this research. To review, these are Q1 How can cognitive theory and analytical practice inform the design of analytical tools?, and Q2 How can transparent, low-cost observational methods be designed to produce empirical datasets that capture functional and cognitive processes?

Chapter 04. The Analysis Literature

The purpose of this Chapter 04, Analysis, is to quickly move the reader through a curated selection of the varied literatures on Analysis. So this Chapter offers a selective discussion of the part-pragmatic, part-theoretical domains of Analysis. These selections are made in order to introduce analysts from different disciplines and analytical tool-builders to various important approaches, and in particular to
identify the procedural steps involved in the general process of analysis, with an explicit focus on the analysis of text.

Thus, this Chapter 4, Analysis, draws upon the author’s professional experience as an analyst who has simultaneously developed expertise in applied research and applied analysis using Visual Analytics tools; to focus on the analytic domains of Cognitive Linguistics Analysis, Discourse Analysis, Computational Text Analysis, Visual Analytics, and Intelligence Analysis. The discussion offered here will focus on how a better understanding of the specified findings, models, and methods from these domains can be used to improve the design of analytical tools in order to improve support for each phase and step in the analysis process.

Patterned after the foregoing Chapter, this Chapter 4, The Analysis Literature, dips in to focus on a selection of main research areas and to consider some of their major models and methods. It will begin to identify how these concepts from these applied domains can contribute a response to the transdisciplinary research questions of this dissertation, by identifying how analytical practice can inform the design of analytical tools, and particularly in the design of transparent, low-cost ways of capturing the products of functional and cognitive processes.

Finally, this Chapter is similarly an obvious and convenient location to define domain-specific terms used in this discussion. For example, this Chapter concludes with a consideration of the Notional Model of Analyst Sense-Making (e.g., the so-called Sense-Making model of Analysis Process) described by Pirolli & Card (Pirolli and Card 2005), noting some limitations as opportunities for refinement. These necessary extensions are articulated as the Cognitive-Functional Flow Model of General Analysis described in Chapter 5. The Flow Model describes general analysis as both a process and a product that is produced over phases of preparation and execution, across internal cognitive and external functional tasks or steps, and is based upon the literature on the analysis process summarized in both this Chapter and the previous one.
Chapter 05. Flow - The Cognitive-Functional Flow Model of Analysis

Figure 1.2. Chapter 5 develops the Function-Argument-Scope Model, which becomes a methodological template for the general approach taken in each subsequent chapter. Functions are of several types, primarily Tasks and Features, which are used to connect two types of Arguments, Resources & Products. These objects, tasks and properties are classified according to their scope as broad, shallow, or deep. Thus, the Cognitive Functional Flow Model of General Analysis identifies and classifies tasks required in the process of general analysis to produce a model of that process which is further segmented into shallow and deep phases of analysis.

This Chapter begins noting that analysis is a challenging cognitive task with fairly well identified functional steps. Though a precise record of the underlying cognitive tasks is not a feasible object for this study, literatures on both cognition and analysis (Hutchins and others 2007; Piorilli and Card 2005; Russell and others 1993) do specify certain cognitive tasks as important parts of the analysis process, particularly in activating Schemas (Judd 2015; Ziem 2014; Emmott and Alexander 2014; Walker and others 2013; Cruz and Henriques 2010; Gómez-Rodríguez 2010; Langacker 2009; Bier and others 2008; Kemp and others 2007; Wolpers and others 2007; Geeraerts 2006; Hard and others 2006; Tuggy 2006; Lea and others 2005; Wünsche 2004; Zacks and others 2001; Glenberg and Robertson 1999; Barbara Tversky and Lee 1998; Gobet 1998; Britton and Gülgöz 1991; Rand J. Spiro and others 1988; Stevens S Smith and Kihlstrom 1987; Hintzman 1986; Talmy 1983), in content Evaluation, and in classification judgments. Other cognitive sub-tasks have been identified in these literatures as being involved in specific functional steps (e.g., search, evaluation, model building, reading) in analysis. Thus, even without employing a neuroimaging or cognitive task analysis study, we can extend the standard analysis model that is most widely cited in
Visual Analytics, the sense making model of intelligence analysis, to include missing cognitive and functional steps identified from these literatures.

Insofar as gaps in analytic tool coverage informally seem to correlate with gaps in the conceptual model of analysis used by analytical tool builders, we conclude that the lack of adequate support across all phases and steps of the analysis process is in some part attributable to gaps in published analytical tool-builders’ knowledge of analytical process goals, data, and methods. Also, both data fusion tasks and certain widely acknowledged scale and modality issues just make some tasks in text analysis very hard. Thus, by focusing efforts where gaps in knowledge have plausibly contributed to gaps in analytical tool coverage, I argue that producing improved Cognitive-Functional Flow Model of Analysis should lead, in time, to improved knowledge of analytical process among analytical tool-builders, and should also eventually contribute to better coverage of currently under-supported tasks among analytical tools. This level of my research is thus focused on improving the process model of analysis to include important cognitive tasks and missing functional tasks.

Chapter 06. FlowSpaces - Capturing, analyzing, and visualizing a longitudinal dataset on multi-focal multi-application multi-tasking

Figure 1.3. Chapter 6 is described in terms of the Function-Argument-Scope Model. This chapter broadly observes and analyzes a user’s focused Tool usage to produce Workflow data and visualizations and to analyze that data to identify and classify transitional patterns.

Digital analytical tool usage is complex multi-functional domain with very poorly identified functional steps and transitions. Here, neither prescriptive account of the ideal functional tasks or task-sequences, nor an isolated empirical sampling of functional transitions within a single tool would tell the whole story. Neither the
specific literatures on multi-tasking nor workflow nor that of User Interaction more generally has published longitudinal studies or data on multi-year, multi-tool, multi-tasking. Since no prior systems existed for logging hierarchical event data for core-periphery digital tool usage in functional task-space, I argue that the design, construction, and customization of a reusable system-of-systems for this purpose is a practical and usable research success that should also contribute to improved knowledge of digital tool usage sessions among both analytical tool-builders and analytical tool-users. I trust that in time this knowledge will similarly contribute not only to better coverage of unsupported tasks in analytical tools, but also to meta-cognitively improved tool usage sessions by self-reflective tool users.

Thus, the initial phase at his general level of my research focused on building a system-of-systems ‘meta-tool’ to capture, convert, and produce a longitudinal dataset of multi-tool task focus. Moreover, by examining this dataset on “functional usage patterns of digital tool usage sessions,” we can exemplify the exploratory data analysis method supported by visual analytic tools, and we can identify new empirical patterns for future statistical analyses. For example, exploratory analysis has revealed three distinct transitional patterns in the transition data. We can now see where prototypically non-analytical tools take a cyclical role in analysis; we can see long, contiguous app-usage patterns, that may indicate mature within-app toolsets that succeed in supporting analytical focus and optimal flow experience (Csikszentmihalyi 1991); and finally we can identify which tools and usage sessions presage a functional shift to a new, distinct Activity focus cycle in the digital tool usage workflow. These insights were not hypothesized a-priori, but rather, using a data-grounded methodology, were discovered using exploratory data analysis method. Validity of the patterns derives from the coverage of the data, which is nearly population data coverage (all events for 1439 of 1460 Days) for app-usage events. These findings are consonant with the literature on challenges of complex workflow design (Kohlhase and others 2013; Chinthaka and others 2011).

Chapter 07. m³ [em-cubed] (multi-modal meta) analytical media management
Chapter 7 as described in terms of the Function-Argument-Scope Model. This chapter offers recommendations and prototypes for broadly enabling the management of analytical data, focusing on facilitating the capture of shallow analytical judgments as metadata. The background color of this figure draws upon the semantic mapping between Goal/Process and Color in Figure 0.1. Accordingly, this color indicates that the primary goal of Chapter 7 is to better understand the preparatory analytical process by prototyping the m^3 system and method to capture shallow analytical judgments as bundles, tags, and highlights. Capturing these data objects is calculated to enabling file-focused reflection on both procedural states of files as they are used in workflows, and specifically upon the process of feature & boundary identification and evaluation.

Prototype methods and systems-of-systems are devised and discussed (Chapter 6, FlowSpaces & Chapter 7, m^3 [em-cubed]) to first identify and then to address specific functional gaps in analytical tool coverage. Chapter 7 draws upon Chapter 5, the Flow Model to guide the selection a single unsupported analytical task, analytical reading, and to specify the segmentation of this task into preparatory and execution phases. Since scanning file names and evaluating search results also require reading, this kind of scanning over textually described features of the document is considered shallow reading. This theory-driven segmentation of analytical reading by phase identifies two under-supported target areas for practical analytical tool development. Namely, this Chapter describes a prototype system-of-systems for capturing shallow analytical tasks in preparation for reading and while undertaking shallow reading. Subsequently, Chapter 8, CDG+, describes progress towards the implementation of system to support the deeper execution of analytical reading.

Thus, this Chapter focuses on shallow judgments made in the preparatory phase of analysis process, that Chapter 5, Flow, tells us includes, search, collection, evaluation, and annotation of documents containers and contents. Thus, Chapter 7 introduces and proposes prototypes tools for the preparatory phase of reading, particularly for document collection, management, and conversion. Prototype tools
provide low-cost ways of capturing shallow analytical judgments involved in the preparation tasks leading up to deeper reading, using tags to capture evaluative judgments and workflow process states, and using highlights to capture evaluation of user-specified chunks of text as highlights. These tagging and highlighting prototypes are briefly introduced, as are projections from theory into new designs for improved tools.

**Chapter 08. CDG+ Extended Cognitive Discourse Grammar**

*Figure 1.5. Chapter 8 is described in terms of the Function-Argument-Scope Model. This chapter broadly integrates cognitive, linguistic, and discourse analytical theories and methods, identifying and addressing problems with the original framework and extending the models and methods of Cognitive Discourse Grammar. Per the semantic Goal/Process/Color mapping established in Refer to Figure 0.1, the goal of Ch 08 is to understand the comprehension process through the production of CDG+ models.*

The original CDG approach focused on extending Lewis’ minimalist Possible Worlds and Modal Logics into Rich Worlds, so as to be full of detailed, fully specified objects in relations explicitly patterned after those in the Real World. Rich Worlds can be seen as philosophical correlate of the Mental Simulations postulated in embodied cognition approaches to comprehension (Zwaan 2015; Jones and others 2014; Chow and others 2014; Zwaan 2008) and as basically equivalent to the Mental Spaces of (Fauconnier and Sweetser 1996; Fauconnier 1985). Implications of these relations with other established approaches are discussed further.

Beyond the Rich Worlds Model, CDG also offers a thorough and unflinching integration of insights from across cognitive, discourse, and standard approaches to language - all within a single framework. CDG/+ thus integrates and extends Grice’s Cooperative Principles of Informativeness, Truthfulness, Relevance, and Clarity, Lyon’s Deixis and Anaphor, Givón’s Coherence, Sperber & Wilson’s Relevance, Føllesdal’s Opacity, Lewis’ Accessibility, each with significant

Even after twenty years, this novel framework of individually recognizable components remains a unique system-of-systems that focuses on specifying the formation and externalization of mental representations during online reading comprehension. Nevertheless, when considering how the original CDG approach could be improved upon, it is readily apparent, given the integration of so many concepts and approaches in language, that complexity in the emergent framework is a marked challenge.

The original CDG approach made no specific claims representing itself as a discourse processing model. Instead, it focused its efforts on developing a **model of the product** (Text Worlds) of this online discourse comprehension process. However, given its focus, method, and the emergent dynamics of the mechanisms that the original CDG approach integrates, in this Chapter I will argue that extended version of CDG, which I have called CDG+, should be considered a plausible (Forsythe and Xavier 2006; Forsythe and others 2006) **Discourse Processing Model**.

Thus, at its core, the original CDG provides the analyst with the means to systematically approach the various levels of **context in discourse** and identify the contribution of each level to their comprehension of the described **scene**. Chapter 8, **CDG+**, will therefore identify and visually model each component of the contextual **situation** that CDG describes as contributing to the internal mental representation of the **Discourse Model** produced through the dynamics of a unique **Discourse Processing Model**. This conceptual overview of contextual elements should help the reader to see and understand the position of the individual mechanisms of both the **Discourse Processing Model** and the **Discourse Model** which it produces. This **Context Model** should prove especially useful for the reader who is unfamiliar with this method or domain.
Thus, CDG+ offers a Context Model, which first specifies situational dynamics in the CDG+ Discourse Processing Model that condition the production of the CDG+ Discourse Model; and second, specifies the relations between the situations that are represented as levels in the mental Discourse Model. (See Section 8.5.1, CDG+ Context Model.) The CDG+ Discourse Processing Model is thus positioned as a plausible (Forsythe and Xavier 2006; Forsythe and others 2006) model of the cognitive processes used to build an internal, representational Discourse Model during discourse processing for language comprehension.

Critically, the CDG+ Discourse Processing Model relies on the activation of Knowledge Structures from memory (as van Dijk and Kinsch 1977 specified). CDG/+ specifies these Knowledge Structures as Fillmore’s Cultural Knowledge Frames, and thus integrates Fillmore’s Frame Semantics into a Rich World Model, that is in turn integrated with Johnson-Laird’s Mental Models and van Dijk’s Situation Models (Fillmore 2006; Fillmore and Baker 2001; Baker and others 1998; Fillmore and Atkins 1998; John B Lowe and others 1997; David Kellogg Lewis 1986; 1973; David K Lewis 1968; Johnson-Laird and Girotto 1998; Johnson-Laird 1983; Van Dijk 2006; Staczek and others 1985; Van Dijk and Kintsch 1983; Kintsch and Van Dijk 1978; Van Dijk 1977).

Equally importantly, the CDG+ Discourse Processing Model also relies on the complex dynamic processing of an AI-managed CDG Common Ground, which transacts a host of indexing and matching tasks in its role as a specialized type of socially shared, textually cued, short-term memory. CDG/+ specifies these indexing tasks as anaphor resolution and reference-chain matching, deictic counterpart resolution, identification of new information, and the placement of emphasis (on repeated old information) and accent (on new information).

Moreover, the original CDG approach integrates into CDG Common Ground its own Selection Rules for selecting warranted, inferred information from relevant activated Cultural Knowledge Frames. Then, CDG information Evaluation rules determine coherence and result in the placement of Emphasis on old information and Accent on new information. Finally, Incrementation rules govern the insertion of new information, from the text and from inference, into the CDG Common Ground. Exceptions for non-standard introduction of new information are handled using the principle of Accommodation (Werth 1993; 1984; Werth International
CDG presents a novel **Coherence Constraint** which governs *Incrementation* of new information both *from the text and from inference* into the *CDG Common Ground* (Werth 1999c; 1999i). In CDG, evaluating coherence requires active comparison of each current proposition with those already in the *CDG Common Ground*. First, the *CDG Common Ground* evaluates for functional relations (synonymy, antonymy, meronymy, metonymy) between its N most recently activated propositions and the current proposition. The implication of a matched functional relation is that the current proposition is old, so it is marked to place *Emphasis*. Old information is linked to its antecedent references in the *CDG Common Ground* by **Reference Chaining** (Werth 1999k). Conversely, non-matching propositions are evaluated as new, are marked with *Accent* and are *Incremented* into the CDG Common Ground.

The purpose of the original CDG framework is to create a visual proxy for the mental model created internally while reading. In CDG/+ this is the *Discourse Model*, which specifies three levels in the internal mental representation that is produced while making sense of language (e.g., equivalent to van Dijk & Kintsch’s three level situation model).

Since I could not introduce the entire CDG approach in the available space, I have summarized and specified relations between the *CDG+ Discourse Processing Model* and its product, the *CDG+ Discourse Model*, in the visual *CDG+ Context Model*, which specifies the four Situations that impact both CDG+ Process and Product Models.

Beyond these process and product models, extended CDG+ also addresses various shortcomings in the original approach. Thus, CDG+ offers the *Deictic Matrix* as an aid to analysts when starting to place their first TextWorld, to specify its starting temporal, epistemic, and proximal and accessibility status. Analysts simply select a point within the Matrix that encapsulates their judgment of the TextWorld’s status in four deictic systems. The CDG+ Deictic Matrix addresses the problem of the unlabeled and presumed temporal origin in standard Text World Diagrams. The Deictic Matrix is an incremental improvement to the Standard CDG
Tense and Remote System relationship specification that has reduced label redundancies and made boundaries within the matrix dynamic.

CDG+ also offers a Meta-Contextual Frame and the CDG+ Manifest Semantic Analysis method as an improvement upon the Deictic Matrix. The Meta-Contextual Frame spins the somewhat cluttered multi-dimensional Deictic Matrix around, popping it out of the two-dimensional flat matrix representational space. The result is a bounded three dimensional representational space with a novel frame container. Since in CDG, the focus is on representing the aggregate of reader judgments, there is no explicit focus on supporting the capture of specific reader judgments while reading. CDG+ Manifest Semantic Analysis explores the possibility of using dimensional content evaluations captured in the CDG+ Meta-Contextual Frame to automatically validate automated word-sense selections. Moreover, it provides an aggregation method to weight spot-classifications of single terms by the combined frequencies of the classified terms in their various definitions, to work out differentially weighted dimensional signatures for each sense of a potentially ambiguous term. Thus, both the CDG+ Meta-Contextual Frame and the Manifest Semantic Analysis method are offered to support classification of both individual terms and aggregate TextWorld situations along multiple deictic and contextual dimensions.

Since the original CDG approach does not specify what happens to accepted propositions in the Common Ground once they lose their activation focus, I have specified the longer-term collection of deactivated terms into the CDG+ Personal Ground. Finally, since the Flow Model of Analysis can only be complete with the specification of requisite analytical reading sub-task network, I have specified the CDG+ Task Model for Analytical Reading, which is extrapolated from the entire description of the CDG/+ discussion using a manual analysis of predicate argument structure to identify both tasks and the objects upon which the action is undertaken.

**Chapter 09. TextWorlds+**

In this Chapter, I offer improvements to the manual method of Discourse Analysis from Cognitive Discourse Grammar known as the original Text Worlds approach. The long-term objective of these improvements is to build the extended
CDG+ method into a digital tool for representing the conceptual structure of discourse. This tool for visualizing the analytical process of reasoning over discourse will support analysts as they engage in meaning construction, model construction, reasoning with models, and decision-making with interactive models and existing knowledge bases (e.g., Cultural Knowledge Frames and CDG+ Personal Ground and CDG Common Ground).

The extended TextWorlds+ Analysis method proposes a technology-supported process whereby analysts can interactively build and revise domain specific schemas of text contents over time. Its implementation as an interactive inference-logging analytical reading platform would benefit analysts by externalizing and visualizing expert domain specific schemas of propositional content, which are key in expert performance (Ericsson and Lehmann 1996).

For the past thirty years, well-positioned observers (Heuer 1999) have noted that analysts urgently need tools that help them structure information, clarify assumptions, identify inferential chains, and specify uncertainty about their conclusions. Only recently has research in the emerging field of Visual Analytics begun to take these requirements seriously.

Given how much of the analytic process happens internally while reading (leaving very few traces or manipulable artefacts), and considering the analysts’ requirement to externalize, present, and defend their reasoning process and its conclusions, it is surprising to note that tools to help scaffold, organize, or visualize these implicit analytical tasks are not more widely available, particularly for analysts who must work closely with text.

While various piecemeal computational tools and visual aides have long been used to complement many analytic procedures (Card and others 1999), very few tools exist to assist the analyst specifically while reading. To my knowledge, none exist to help the analyst by formalizing and automating the meta-observational process of exhaustively noting the formation and evolution of their own conceptualization of a text as it unfolds while reading.

The proposed implementation discussed in Chapter 9, TextWorlds+, will leverage established methods from Natural Language Processing, integrating their output into an interactive frame-based knowledge representation schema for
**discourse.** An interactive system capable of producing basic, interactive TextWorld+ models would circumvent the requirement for a high level of linguistic expertise required to undertake deep manual discourse analyses, and effectively resolve the biggest single critique of the CDG/+ method. The proposed system should provide readers with novel interaction with text that focus on identifying connections from their own knowledge bases, and evaluations of textual contents.

Even simply as a manually executed Discourse Analysis methodology; given a text and a reader, the extended TextWorlds+ Analysis method can model a start-to-finish account of the conceptual spaces a reader constructs. Johnson-Laird, Werth, Fauconnier, Lakoff, and Langacker have found that readers create mental and conceptual models, which help them to better understand and reason about the world. It is clear that, regardless of the theory used to explain language and comprehension, readers are in fact somehow able to process and resolve the ambiguities and challenges of reading.

However, mental models are not easily articulated. Moreover, once they are, they become unwieldy to self-consciously interact with while working. Given the limitations of human working memory, working simultaneously with multiple mental models and describing their state, contents, and functional dynamics is just not a reasonable expectation of a human analyst, while they are analytically reading, especially while working as an analyst. Thus, since it is not efficient to do the work twice (once to accomplish it and once more to think about what was done), since we cannot reasonably expect hold all this in our heads simultaneously, and given how quickly we accomplish on-line reading comprehension, in the view presented here, the best solution is externalization via computational implementation. The implemented model must thus provide the interactive mechanisms described in Cognitive Discourse Grammar - the Discourse Model, including TextWorlds, Frames, Common Ground - including all the properties and slots necessary for various classification and reasoning steps. It would pre-process the text to identify each linguistic features that computational linguistics is currently equipped to provide.

In this Chapter, I argue that implementing a discourse model of reasoning should be a top priority for analytical tool builders. Doing so should ultimately enable analysts to interact with this comprehension and reasoning process directly,
to predict and project baseline comprehension networks for an analyst to use as starting point in model-building and refinement. Moreover, doing so should reduce the cognitive load of reasoning with rich discourse models that are presently constructed by standard, manual means.

**Chapter 10. Conclusion**

This Chapter begins by discussing the method proposed and employed for 10.1 - Transdisciplinary Gap Analysis.

Then in the following Section 10.4, Research Questions Answered, I will specify more fully how the two Research Questions established the theoretical grounds for each Research Project’s specific Research Goals. This section provides my methodology for just how, in the research presented here, *theory and practice can inform tool design* generally and specifically how *empirical datasets can be cheaply and transparently produced to model invisible functional and cognitive processes*. Further answers to project-specific Focus Question are also provided for each Research Project, in short form. Thus, Section 10.4, Research Questions Answered, elaborates upon the brief introduction in Section 1.3 Research Questions.

Finally, in the subsequent Section 1.5, Contributions, I will specify the Contributions of this research more fully, by Research Project and Purpose, and also in relation to the Core Problem it resolves. In general, the two goals of this dissertation were identified as extending and building cognitive models and building (toward) analytical tools. Each model and tool produced in this research has a specific purpose, that serves to simultaneously address a Core Problem, a specific challenge, a General Research Question, and to project-specific focus questions. Some also have minor project-specific focus questions. Finally, Contributions are also tallied by type, including Cognitive Models, Prototype Systems, Case Studies, Visual Conceptual Models, Gap/Trouble Indices, Extended Methods for TextWorlds Analysis, and Exemplars.

The oft-repeated mantra, taking a ‘human scale’ view of the problem space, permits measured shifts in focus from the widest, general level of functional workflow, to the intermediate, preparatory level of analytical reading that includes shallow analytical judgments, to the deepest level of language comprehension
situation model formation during analytical reading. This dissertation has worked towards collecting datasets to record and model analyst sense-making activities across each of these levels; generally, while working with digital tools in both the preparation and execution of analysis, and specifically while working with digital texts at shallow preparatory and deeper analytical levels.

Consequently, most research and development time has involved designing and automating methods and technologies for the transparent collection of three types of personal dataset. Each dataset joins a deep individual focus with broad, multi-year event coverage to produce and analyze deep data on detailed user activities. These datasets are personalized collections of digital tool workflow, document and metadata annotation, and personal language comprehension.

The first motivation is to design invisible technologies to capture invisible processes, like judgment or workflow. The rationale is minimal user interaction, and the mantra is Ask Once, But only if you must. The second motive is to render invisible process into visible products, for example, as interactive mental models, or multi-app multi-tasking workflow models. The rationale is that externalization of complexity aids reasoning, and the mantra is See!

This dissertation has identified gaps in both the theoretical models of analysis widely used in Visual Analytics, and in the support offered by VA tools to several important phases and tasks in general, text-focused multi-modal data analysis.
Part I: Foundations

Chapter One, Introduction has introduced the 1.1 Challenges, 1.2 Motivations, Research Questions, 1.4 Research Methodology, and 1.5 Contributions that ground this Dissertation Research Programme.

Part I, Foundations, establishes the conceptual background and as a common basis for the research projects reported in this Dissertation.

In Part I, Chapter 2, The Problem Situation, provides the extended consideration of two dimensions of the problem situation. First, there is a particular challenge inherent in all research that spans disciplinary boundaries. Next, there are the classes of problems addressed and resolved by this research, through the development of prototype tools, data collections, and empirically-based recommendations for analytical tool design.

Thus, Chapter 2 first elaborates upon both the Paradox of Transdisciplinary Research, to explain why this dissertation includes so much definitional and foundational content. Next, Chapter 2 elaborates on the two Core Problems in the analytical process problem space addressed by this research, namely, the Cognitive Limits of human analysts, and Functional Limits of analytical tools.

In Chapter 3, The Cognition Literature, this Part offers a selective discussion of literatures from specific research domains of Human Cognition, in order to clarify the role of specific domains in higher-level cognition in general analysis, and particularly what happens in terms of cognitive tasks during analytical reading.

In Chapter 4, The Analysis Literature, this Part offers a selective discussion of literatures from specific pragmatic domains of Analysis, in order to specify both the higher-level cognitive tasks and the procedural functional tasks involved in analysis.

Taking a human scale view of the analysis process problem space naturally affords a multi-level focus that provides the structure for the remaining Chapters, that moves from understanding it as a whole, to its widest parts, and then to its deepest parts, keeping the human analyst reader in focus throughout. Part I, Foundations, thus begins with a broad introduction to the problem space and to the domains which these research projects draw upon in order to contribute to their solution.
In Part II, *Modelling Cognitive-Functional Analytical Process*, this breadth of focus continues, viewing analysis from the highest, most general level. The research in these Chapters contributes to building cognitive functional procedural models (Chapter 5, *Flow*) and to collecting, analyzing and visualizing digital workflow task-tool usage datasets (Chapter 6, *FlowSpaces*).

In Part III, *Supporting Ubiquitous Shallow Analytical Judgment*, the research moves to a focus on analysis at the intermediate, preparatory level, which includes discovering and prototyping ways to support cross-cutting cognitive tasks like ubiquitous shallow analytical judgments.

In Part IV, *Modelling Conceptual Structures in Deep Analytical Reading*, the focus moves to the deepest level of language comprehension processes active during analytical reading. The broadest contributions at this deepest level include the refocusing of discourse analytical methods to precisely mirror the dynamics of discourse processing models in order to externalize the mental ‘situation models’ created internally while reading, and ultimately to produce interactive visualizations of the conceptual structures of text, in order to fix the gaps in visual analytic tools’ support for analytical reading.

In Part V, *Discussion*, the focus moves to the review of Recommendations and Directions for Future Work.

Thus, the focal energy of this dissertation is in continuous motion, moving in each Part toward a focus on identifying and supporting the functional and cognitive tasks that are requisite to the preparation and analysis of the difficult modality of text.
Chapter 2 - The Problem Situation

This Chapter 2, The Problem Situation, provides an extended consideration of three important dimensions of the problem situation. First, there is challenge inherent in all research that spans disciplinary boundaries. Next, there are the two major classes of problems addressed by the research through the development of prototype tools, data collections, and empirically and experientially-based recommendations for analytical tool design.

Thus, this Chapter first elaborates upon both the Paradox of Transdisciplinary Research, to reframe reader expectations and provide a rationale and strategy for dealing with a large amount of review material. Next, this Chapter elaborates on the two Core Problems in the analytical process problem space addressed by this research, namely, the Cognitive Limits of human analysts, and Functional Limits of analytical tools.

So this Chapter begins by elaborating upon the unique discourse situation, first introduced in Chapter 1, Introduction, in which both writers and readers of transdisciplinary research find themselves. Because the writer is focused on several audiences, they cannot make standard assumptions about the reader. Thus, extensive introduction and review of basic concepts is necessary prior to elaboration of arguments. Advanced focused reading strategies are offered to facilitate reading for domain experts.

Next, this Chapter highlights the challenges facing modern analysts and points out established problems in the field of Visual Analytics. These challenges mostly regard difficult tasks in the analysis process (e.g., like externalizing the results of classification and reading); gaps and omissions in published understanding of the analysis process; and shortcomings in the match between tasks in the analysis process and tasks supported by analytical tools.

Each Core Problem is discussed in an eponymous Section. The first, Section 2.2, Cognitive Limits of human analysts, first attributes cognitive limits to the Procedural Complexity of the analysis process. Next cognitive limits are viewed as a consequence of the variation over time in the analyst’s Meta-Cognitive Awareness of their own internal knowledge states and their current procedural states in digital analytical workflows. Thus, low-cost methods of reducing the complexity of the
analysis process (e.g., combining tool functionality, facilitating export trajectories between analytical tools) are proposed. Additionally, low-cost methods of capturing the analyst’s internal knowledge states and current procedural states as tags, are also proposed.

Next, Section 2.3, Functional Limits of analytical tools, offers an extensive consideration of specific shortcomings in analytical tools. These shortcomings are attributed to 1) Toolmakers’ Expertise & Knowledge, 2) Data Features, and 3) Functional Coverage of Tasks in the Analysis Model.

Specifically, for current analytical tools, limitations in these general classes relate to

1) the Analytical Toolmakers’ Expertise in Eliciting Tool Requirements and to their capacity as regards Implementation of Relevant Findings from Cognitive Science, and also to the Analytical Toolmakers’ Knowledge of Real-World Analysis Process, and Real-World Tool Usage.

2) technical challenges dealing with Data Scale, Modality, Complexity, Type, and

3) Functional Coverage of the general Analysis Model, of specific Task Models, and of tasks to facilitate Cognitive Externalization.


2.1 - Discursive Challenges of Transdisciplinary Research

As suggested in the Introduction, transdisciplinary scholarship often presents a real challenge for both the readers and for writers. Readers may be highly familiar with a few topics and concepts that are derived from areas related to their own expertise, yet understandably unaware of other terms and methods. So transdisciplinary writers must suspend the standard discursive assumption (Grice 1957) that they share a sufficient basis of domain knowledge with their readers to simply begin the discussion.

Instead, the transdisciplinary writer must continuously first adopt a didactic, teaching stance to define terms in a domain then shift to an expository stance to elaborate upon relevant conceptual structures of that domain. To avoid confusion, each definition and explanation should include the domain and the history of the particular conceptualization adopted. Then, after initial introduction and often some reiteration of terminology and definitions, the writer must then also assume other rhetorical stances, like argumentation and interpretation, in order to sufficiently develop conceptual structures to a level that allows expert readers to recognize the authors’ expertise in this area and engage with any claimed contributions to the domain. Thus, this Dissertation presents Chapters 1 through 4 as establishing the required conceptual foundation for the Contributions made in Chapters 5 through 9.

The paradox of transdisciplinary scholarship is that its texts are simultaneously written for multiple audiences, yet for a single reader. If the relevant properties of ‘the audience’ are specified as expertise by domain(s), the reader’s audience membership is a dynamic function of reader’s expertise in relation to the current topic and its relevant domain at each point in the text. At no point along the way will the writer ever know which audience the reader is presently in, but only that audience membership varies continuously. Effective transdisciplinary writing should cause membership to vary towards expertise as a function of progress through the text. For this reason, at each point of introducing new terms, the functional purpose of the text switches back to didactic and expository and then on to argumentation.
So transdisciplinary researchers must write for Schrödinger’s cat\(^6\). Consequently, transdisciplinary writers first make rhetorical choices to establish a foundation\(^7\) based on the quantum state of reader topic knowledge, and only then may they proceed to build an argument on that quantum foundation. Each speech-act functional shift (didactic to expository to argumentation) is a necessary gamble taken to improve the writer’s odds of potentially altering the knowledge state of the novice reader and from there to establish the conceptual connections required to follow the arguments.

Now, then, the evaluation of transdisciplinary scholarship would appear to be almost as challenging as its reading: it begins with a direct and continuous assessment of goals. Ultimately, the reader must also consider the quality of the argumentation provided and the originality of the Contributions. Readers must determine for themselves how well this work extends cognitive analytical theory, reveals cognitive-functional analytical practice, supports analytical expertise, and exemplifies analytical creativity. To facilitate ongoing evaluation, Research Questions and goals are discussed extensively in the introduction, and are reviewed periodically throughout.

Practically speaking, all of this means that the reader must agree to accompany the author on a journey, first to explain the process through which the problem situation was discovered, and then to describe the way forward proposed here. The author cannot assume that reader is knowledgeable in each of the domains covered in this journey. Thus, for the reader, the price of participation in this quantum dissertation journey, is a bit of learning/review at regular intervals until the identifiable domain-specific terms are sufficiently established for standard discourse assumptions to resume. The path for this dissertation journey begins here.

### 2.2 - Cognitive Limits of Human Analysts

To begin, this section offers some general definitions to ground discussion. General Analysis will be described, in Chapter 5, Flow, as iterating among a defined set of cognitive-functional steps along paired continuums of depth and completion, that range from the shallow analysis required for preparation through the deeper analysis involved in the execution of traditional analytical tasks. The
Cognitive-Functional Flow Model of Analysis describes both traditional manual analysis and multi-modal digital data analysis and also provides the grounds for selecting a single poorly supported analytical task, namely goal-driven analytical reading, for focus and targeted prototype development.

**Cognitive Limits vis. Procedural Complexity**

In both traditional/analog and modern/digital contexts, when viewed at the scale of the human analyst, general analysis and analytical reading are both stepwise, serial processes that proceed in linear time. In both general analysis and analytical reading workflows, the analyst must keep in mind both procedural knowledge (e.g., meaning where they currently are in the process, what they intend to accomplish and how best to do so), and propositional knowledge (e.g., meaning what they actually have discovered and its relation to what is already known). This is hard to do with existing tools.

This is, in part, because analysts and readers must both recall many specific recent steps and general procedural protocols for arriving at their conclusions: the how-to procedure, the thus-far progress, and the to-what-end purpose. They must also find a way to organize, manipulate, and evaluate the propositional-knowledge claims contained within the texts and any other data they are analyzing. These include descriptions of the why and how of the what-is; and actions that provides the what-happens and the who-does-it, in addition to all the contextual factors which ground action and description: the who, the what, the where, and the when.

**Cognitive Limits vis. Metacognitive Awareness**

But not even the analyst can always be perfectly aware of each twist and turn in their own analytical trajectory, let alone of the unique rationales for each choice, and their own many discoveries and their myriad implications. Maintaining perfect situational awareness of procedural and propositional facts and relations is no easy task(Zahra and George 2002; van de Laar and others 2013). Some theorists suggest that it is actually impossible except when employing some expressive externalization of these slippery mental processes (Heuer 1999). To be sure, tracking moment-by-moment progress in real time would not be possible using standard knowledge elicitation methods like a think-aloud protocol.
Militello 2009; Klein 2008; Mosier and Ute 2007). After all, at the end of the day, analysts have a big, difficult job which includes, at some point, reading in order to meet their objectives and produce their analysis.

**Summary for Cognitive Limits of Human Analysts**

This section has described **Cognitive Limits of human analysts**, which are discussed in relation to the *Procedural Complexity* of the analysis process, and to the *Meta-Cognitive Awareness* of internal knowledge states and digital procedural states. The following section will add to the Challenges already addressed in this dissertation by discussing the **Functional Limits of analytical tools**.

### 2.3 - Functional Limits of Analytical Tools

The previous section has described **Cognitive Limits of human analysts**, in terms of the *Procedural Complexity* of the analysis process and also in terms of the analyst’s *Meta-cognitive Awareness* of their own internal knowledge states and digital procedural states. This section will contribute to this discussion of Challenges addressed by this research by describing the **Functional Limits of analytical tools**. These include shortcomings in analytical tools classified generally in relation to 1) **Toolmakers’ Expertise**, 2) **Data Features**, and 3) **Functional Coverage of Tasks in the Analysis Model**. These tool limits in these general classes specifically relate to 1) **Analytical Toolmakers’ Expertise, Knowledge of Real-World Tool Usage, and Implementation of Relevant Findings from Cognitive Science**; 2) **Data Scale, Modality, Complexity, Type**, and 3) **Functional Coverage, Task Model, Analysis Model, Cognitive Externalization**.

To begin, I offer a historical perspective on the Challenge. **Visual Analytics (VA)** is a young discipline that has only recently (2005) emerged at the intersection of *information visualization, software systems design, and intelligence analysis*. VA was initially envisioned as a transdisciplinary solution to the complex problem space defined by the intersection of these domains, that aimed to support analytical reasoning with visual interactive technologies.

**Tool Limits vis. Toolmakers’ Expertise**
Thus, VA began with the knowledge that, in the alternative status quo, the modern analyst must execute unassisted manual analysis upon encountering relevant yet un-tooled data modalities\(^9\) and each time they require a level of focus not provided within available digital tools. So initial publications in VA seemed aware that where tool-functional gaps exist (e.g., analytical tasks unsupported by software systems) analytical tool builders must identify sources for both the analytical expertise and analytical creativity that are essential for the design of new tools (Dougherty and Keller 1982) and methods.

**Tool Limits vis. Data Scale**

Thus, in order for analysts of big multi-modal data, like text, to maximize the benefits of working with digital tools, advances in raw computational power must be matched by advances in analytical tool functionality across all relevant data modalities and to all relevant focal lengths. Analytical tool builders should know that they should consider not only the entire spectrum of data types but also the universe of procedures involved in analysis at various levels of detail.

**Tool Limits vis. Data Modality**

While the advent of personal computing has broadened access to robust multi-purpose digital analytical tools (e.g., Microsoft Excel, SPSS, R) for some data types (e.g., structured numerical or categorical data), certain other data modalities are not well supported by digital tools. For example, all text is multi-modal\(^{10}\) (e.g., it has contextual metadata fields containing both embedded and inferred values, beyond its surface textual contents) (Gernsbacher 1997). So in addition to the innate complexity of natural language structures, both the extraction of embedded metadata values and the inference of relational values make computational analysis of text difficult.

While computational methods for purely quantitative data modalities have scaled with data growth, computational methods of analyzing and visualizing certain other, tougher data modalities have not kept pace with data growth, scale, and complexity. Accordingly, the computational analysis and visualization of multiple individual observations, like instrument readings or still images, and even measured changes of individuals, like trajectories through geographic or
conceptual space (Gennady L Andrienko and Natalia V Andrienko 2008; Gennady L Andrienko and others 2009; 2012; 2013b; 2013a), are widely practiced, robustly critiqued, and continuously refined.

**Tool Limits vis. Data Complexity**

However, sequential event structure (Dou and others 2013) and semantic relational structure in certain other encapsulated data modalities, like text, video, or time-series data (Allison 1984; Steele 2006; Malik and others 2015; Shneiderman 2014), traditionally impose a linear processing constraint with high real-time processing costs. These common data modalities are tougher to analyze computationally.

Historically, for computational analysis of textual data, the standard shortcut around this linear constraint ignores sequential structure altogether, opting instead to use mathematical procedures like the Vector Space Model (Salton and others 1975) to identify salient features of text data by rendering it into feature vectors (Salton and others 1975) using the Vector Space Model and Principal Competent Analysis, which characterizes document features using mathematical computations on the frequency of term co-occurrence (Perfetti 1998) within and across document collections. One limitation of the classic Vector Space Model is that it does not distinguish between variance due to measurement noise and variance due to genuine underlying signal variations (Bailey 2012). Another is that, because these statistical methods collapse the innate structure of language into a feature vector, they typically cannot detect variations in word meaning (e.g., for polysemous terms, or between incompatible senses of a given term) in what is known as the "vocabulary mismatch problem" (Börner and others 2003). For example, these methods typically confound the contexts of leadership, gambling, and individual surnames when treating the term “King” as a text feature. While PCA vector space models rank and relate text contents empirically and effectively at the level of keywords or themes, they do not reflect or model the human cognitive process of analyzing textual data modalities.

Over time, the vector space model has developed many variants, which address specific shortcomings raised in ongoing research. For example, Latent Semantic Analysis (Landauer and others 2013; Landauer and Dumais 1997; Dumais and
others 1988; Furnas and others 1988) was invented to resolve the ‘vocabulary-
mismatch’ problem of PCA (Börner and others 2003). LSA approximates semantic
structure using mathematical computations on the frequency of term co-occurrence
(Perfetti 1998) for each term within and across document collections. Accordingly,
LSA handles synonymy (e.g., variability in human word choice) and polysemy (e.g.,
variability in the meanings a single word can take) (Landauer and others 2004) by
considering the context of co-occurring words (Börner and others 2003). However,
Greenberg et. als., (Glenberg and Mehta 2008; de Vega and others 2008a) and
other embodied cognition researchers question how much of meaning is described
in the co-occurrence of LSA, and show empirically that co-occurrence is not
sufficient to derive meaning from abstract, amodal symbols (Glenberg and
Robertson 2000).

**Tool Limits vis. Data Types**

Ten years of using and evaluating (Harger and Crossno 2012) the best available
Visual Analytics tools (e.g., Jigsaw, InSPIRE, Starlight, GeoTime, Tableau, Gephi) has consistently reaffirmed the fact that the difficult data modality of text is not
handled to the level of detail required by analyst readers today. Most VA tools are
good for structured (e.g., numerical and categorical) data analysis, and some are
helpful in filtering through ad-hoc collections of documents using keyword
searches to select and prioritize target documents for analysis.

**Tool Limits vis. Functional Coverage**

But, practically speaking, the analysis process begins long before it is possible to
filter data with analytical tools. It begins with the activation of both procedural and
domain knowledge schemas (Thagard 2005; Hoffman and Militello 2009; Russell
and others 1993; Pirolli and Card 2005). It then proceeds to the selection of search
terms, the evaluation and filtering of results, and the collection of relevant
documents. So, several basic forms of shallow assessment and evaluative judgment
are requisites for the collection and further analysis of the textual data contained
within digital documents. Analysts report that this preparatory process of
evaluating, collecting and preparing textual data for analysis seems to take up most
of their time (Pirolli and Card 2005). The literature warns that analytical preparation
can consume time resources exponentially if left running unchecked (Gevamay and Wildavsky 1997; Heuer 1999; Hampson and Cowley 2005; Kang and Stasko 2014). Few VA tools provide end-to-end support for this extensive preparatory analysis.

**Tool Limits vis. Task Model**

Nevertheless, at some point every analyst of textual data must still actually read (Görg and others 2014). Thus, eventually, once prepared, textual data is made available within an analytical tool, the analyst may identify a subset of documents for analysis (Wise 1999). Then, at the individual document level, the analysis of textual data requires active analytical reading to various depths. Fundamentally, both basic shallow and deeper text analysis involve procedures that are studied in the field of language comprehension. In both shallow and deep phases of the analytical reading task, and at many crucial earlier points in the preparatory analysis process, even the best VA tools fail to support some very basic, very important analytical functions (e.g., classification, collection, reading) and their component tasks (e.g., judgment, exclusion, selection).

Two facts offer reasonable explanations for this deficit in VA coverage of core analytical functions. First, computational semantic approaches to textual data (the alternative to feature vector approaches) are difficult. While corpus statistics are helpful in providing probabilistic decision-rules for word sense disambiguation, the meaning of a text is ultimately an emergent product of a reader engaging with the textual content of a document. So even the best computational semantic approaches do better with a human-in-the-loop (Cranor 2008; Wright and others 2006).

**Tool Limits vis. Analysis Model**

Most importantly, however, analytical tool builders lack an empirical model of the general analysis process that integrates digital procedural functions and internal cognitive tasks and recognizes that important cognitive work transpires in both preparatory and executive phases of analysis. Consequently, it may be argued that analytical tool-builders have not built adequate supports for the collection and shallow evaluation of multi-modal analytical media in general and of textual data
in particular. Moreover, since efficient and effective analysis of a document’s
textual data can primarily occur through analytical reading and since even the best
VA tools leave reading entirely up to the analyst, it seems clear that whatever task
model of analysis is actually used by VA tool builders does not adequately draw
upon theories of language understanding to support reading in the analysis process.

**Tool Limits vis. Cognitive Externalization**

The various analytical literatures surveyed here in Chapter 2, Literature agree
that one of the most difficult aspects of analytical work is consistent and coherent
externalization of the internal cognitive tasks of evaluation, classification,
inference, and schematization (Heuer 1999). It is difficult, even for experts, to
externalize internal cognitive task sequences, and to recall digital functional task
sequences (Gilhooly 1986) (Crandall and others 2006). Analytical tools could be
better at helping with these difficult tasks.

**Tool Limits vis. Application of Cognitive Findings from Cognitive
Science**

Of course, many domains in the field of cognitive science study specific
cognitive functions (e.g., language comprehension, semantics, reasoning &
inference, classification, knowledge representation) relevant to general analysis and
thus to development of analytical tools. However, in the VA literature, findings on
human cognition are not sufficiently explored and their implications are not
universally applied in tool design. Therefore this dissertation draws from the
literatures on human cognition and analysis to extend the Sense-Making model of
intelligence analysis to include both relevant digital functional activities and
internal cognitive tasks which contribute in both preparation and execution of
analysis. This cognitive-functional model building discussion takes place in Chapter
5, **Flow**.

**Tool Limits vis. Toolmakers’ Knowledge of Real-world Tool Usage**

The literature on expertise highlights the methodological challenges for
knowledge elicitation that most analytical tool-builders are just not equipped to
undertake (Hoffman and Militello 2009; Rende 2000; Rand J. Spiro and others 1988). Consequently, it is highly improbable that tool-builders would themselves be able to identify the challenges and bottle-necks in real-world digital analytical workflows, especially when tool-builders are understandably if narrowly focused on what their tool can do. Thus, you cannot simply ask experts, because even they find it difficult to recall task sequences. And since tool-builders may not have the social science research skills to uncover what real workflows look like, what analytical tool-builders in all fields need is an empirically based, pragmatic method for gathering useful, detailed user data on tool usage that takes the wider context of multi-tool-usage into account. This multi-tool multi-tasking workflow data gathering system and analysis methodology is described in Chapter 6, FlowSpaces.

**Summary for Functional Limits of Analytical Tools**

While the handful of modern VA tools that analyze textual data do help to address the larger end of the data scale problem, where analysts must sift through large textual collections to find documents worth reading, at the other end of the scale VA tools offer very few analytical affordances. What is missing in today’s focus on scalability and big data, is a detailed, multi-level focus on the human end of that scale. By the human end of the scale, I mean the experience and practice of using digital tools to engage in analyst-driven analysis of difficult multi-modal analytical data sources like text.

This section has contributed to the characterization of the Challenges addressed by in this dissertation research, by discussing the Functional Limits of analytical tools. The section has offered an extensive consideration of shortcomings in analytical tools classified generally in relation to 1) **Toolmakers’ Expertise**, 2) **Data Features**, and 3) **Functional Coverage of Tasks in the Analysis Model**. These tool limits in these general classes specifically relate to 1) **Analytical Toolmakers’ Expertise, Knowledge of Real-World Tool Usage, and Implementation of Relevant Findings from Cognitive Science**; 2) **Data Scale, Modality, Complexity, Type**, and 3) **Functional Coverage, Task Model, Analysis Model, Cognitive Externalization**.
Chapter 3 - The Cognition Literature

Thus, Chapter 3, The Cognition Literature, selects from the broad domain of Cognitive Science, including Artificial Intelligence, Semantics, and Discourse Linguistics, with particularly attention for literatures which theorize the main theoretical components of Cognitive Discourse Grammar (Chapter 8, CDG+), such as possible worlds, schemas, scripts, and mental models. Thus, from the many fields that study higher-level cognition and cognitive skills, the research domains in focus here will be Semantics and Language Comprehension, including Discourse Processing; Knowledge Representation; Decision-making and Reasoning, primarily from the perspective of mental models.

The rational behind this Chapter is simple. Current research in high-level cognition from the cognitive sciences can clarify and specify the cognitive tasks involved in the analysis process, just as generalized specification of how analysts use tools holds the potential to contribute to the design and improvement of analytical support tools (Thomas and Cook 2005); but only insofar as analytical tool-builders actively discuss how findings from these relevant domains in cognitive science relate at each stage of the analysis process (Ribarsky and others 2009). This Chapter opens that discussion.

Thus, this Chapter introduces the domains of Semantics, Language Comprehension, Reasoning, and Knowledge Representation. These domains study the cognitive processes employed in discourse processing, reasoning using models, and knowledge representation. High level cognitive processes and concepts selected from these domains for discussion here include inference, mental models, long-term memory schemas, and specific models of discourse and reasoning. The concepts introduced here are central to the approach taken to answer research question One (How can cognitive theory and analytical practice inform the design of analytical tools?)
3.1 - Language Comprehension

The goal of research in language comprehension generally and discourse processing in particular is to understand how individuals build meaning from the extended linguistic events in the course of naturalistic communication (Sparks and Rapp 2010). This includes studying the activation of information from memory, the influence those activations have on moment-by-moment comprehension, the ways in which comprehenders make connections within and across linguistic contexts, and the contributions of factors beyond the linguistic stimuli itself on everyday discourse experiences (Sparks and Rapp 2010).

Discourse processing draws from the deep literature on the cognitive processes involved with written language comprehension. This literature identifies individual differences that impact discourse processing, including prior knowledge, language ability, and reader goals (Sparks and Rapp 2010). Of these central concerns in discourse processing, prior knowledge is a key component in TextWorlds, which provides novel mechanisms that are able to model precisely how the activation of relevant prior knowledge from memory affect moment by moment comprehension. Sparks' review of recent discourse processing research in cognitive science thus supports the main assumptions made by regarding prior knowledge and activation of knowledge from memory (Sparks and Rapp 2010). The mechanisms of Cognitive
*Discourse Grammar* describe precisely how the activation of relevant prior knowledge from memory affects moment by moment comprehension.

Within the domain of *discourse processing*, Coulson (Coulson 2001) contrasts *compositional* and *constructivist* accounts of *language comprehension*. Background knowledge and *contextual information* play an important role in both accounts.

### 3.1.1 - Compositional Comprehension

Proponents of *compositional approaches*, like *generative linguists*, explain speakers’ *productivity* (i.e., the exponential combinatorial possibilities using a limited number of words) using mechanisms of *lexical access* and *parsing* in *sentence integration* (Coulson 2001). However, in light of the high *computational demands* of sentence integration even for simple sentences, and given that the *underspecified output* of the *compositional parser* that executes *post-lexical* processing, Coulson argues that the *compositional process* provides an *incomplete account* of the *interpretive process* and is therefore redundant (Coulson 2001).

Other problems for *compositional approaches* include *ambiguity*, which is encountered whenever selection between multiple sense options is required (Tuggy 2006); and *semantic indeterminacy*, in which the *lexicon* offers no set of meanings (Coulson 2001). These problem situations force language users to create an interpretation that is consistent with both the *semantics of the term* and the *contextually available information*. Such terms display systematicity but not truth-functional compositionality (Semantic Leaps: Frame-Shifting and Conceptual Blending in Meaning Construction 2006a).

Clark (Clark 1983) describes such cases and their requirement for using background knowledge in a creative manner as *contextuality*. Clark argues that non-standard interpretations or *nonce senses* result in two problems for parsers that incorporate *sense-selection* (Clark 1983). The *non-parsing problem* occurs when the lexical senses cannot compute the intended meaning, since the lexicon does not have the contextually relevant meaning. The *misparsing problem* is when the compositional meaning is congruent but not intended by the speaker. This happens when existing lexical entries are selected but are not the intended sense. (Clark 1983)
3.1.2 - Constructivist Comprehension

Alternatively, **constructivist accounts** view language as cuing the construction of **frame-based representations of sentence meaning** that involves **search** through **long-term memory** for a **frame** that will **anchor** or situate the **sentence representation** (Coulson 2001). All such **frames** have **slots** which structure the speaker's **interpretation** and **expectations** by permitting participants to explicitly understand the relationships between **elements** by **binding** them to **slots** in the **frame**; and by permitting the **inference** of **unstated elements** (Fillmore 2006).

So readers form **message-level representations** by accessing **frames** from **long-term memory** to assist in the **construction** of **cognitive models** in a **mental space lattice** (Clark 1983). The reader begins activating a **frame** by reading a **word**. **Words** activate **knowledge structures** pertinent to the current **context** (Fillmore 2006). Beyond words, though, many aspects of language function as **cues**: **morphosyntactic marking**, **word order** and other **syntactic language features** prompt retrieval of **abstract frames** which focus attention on aspects of evoked knowledge structures (Langacker 1987). All **information** from the discourse is filled into **slots** in this **frame** (Fillmore 2006). **Frame shifting** involves activating a new frame and integrating it with the existing structure (Coulson 2001).

Coulson explains that **semantic indeterminacy** is not a problem for a **constructivist account**, since words do not activate **lexical entries** but **abstract structures** and **processes** for integrating with contextually available information (Coulson 2001). Consequently, on this view, **meaning construction** is best construed as occurring on a **continuum** that stretches from **prototype frames** with **default values** at one end, to **nonce senses** at the other. **Creative meaning constructions** achieved through **blends** and frame-shifting are closest to the **nonce sense** end of the spectrum (Clark 1983). The constructivist account is even robust enough to explain **apparently compositional** phenomenon in cases of no or little contextual information. In such situations language users evoke **blank frames** with only **default values**. In these cases, the most frequent conceptualization is likely to obtain (Coulson 2001).

On the constructivist account of language comprehension, analytical tools that aim to help analysts with the meaning construction phase of reading in analysis,
must take frames into account. Efforts to scaffold or record analyst sense-making will consider how existing lexical resources like PropBank (Eleni and Aravind 2004), FrameNet (John B Lowe and others 1997; Baker and others 1998; Feldman and Narayanan 2004) and VerbNet (Enhong Chen and Wu 2005), in addition to other lexical resources like WordNet (Gangemi and others 2002; Tonelli and Pighin 2009) might be dynamically accessed to provide contextually cued and empirically grounded resources to scaffold analytical meaning construction while reading.

3.1.3 - Discourse Processing Models

However, by examining the literature on situation models in discourse processing (Morrow and others 1987; Fletcher and Chrysler 1990; Zwaan and van Oostendorp 1993; Zwaan and others 1995b; Zwaan 1996; Radvansky and others 1998; Zwaan 1999; Gernsbacher and Foertsch 1999; Fincher-Kiefer 2001; Radvansky and Copeland 2006; Therriault and Raney 2007; Magliano and Schleich 2010; Radvansky and others 2010; Zwaan 2015), we see that almost the entire literature is based on the work of van Dijk & Kintsch (Kintsch and Van Dijk 1978; Van Dijk 1977) and still cites early work in this area extensively (Van Dijk and Kintsch 1983).

van Dijk & Kintsch (1983) proposed that the mental representation formed by readers have three levels, including the surface code, the textbase, and the situation model (Van Dijk and Kintsch 1983; Kintsch and Van Dijk 1978; Van Dijk 1977; Bransford and others 1972). Thus, within the basic representational models, the surface representation captures lexical and syntactic of the text verbatim, as an exact mental representation of the text that does not remain in working memory for long and is not transferred to long-term memory. The propositional or semantic representation captures the meaning of a text at both a local and a global level and is accessible in long term memory as gist (e.g., the general meaning). Finally, the situational representation represents those aspects of prior knowledge that are triggered by the reading of the text; it is thus a representation based on schematic knowledge that is produced by combining textbase elements with elements from the reader’s general knowledge.

(Graesser and others 2003a) notes that most discourse psychologists adopt van Dijk & Kintsch’s (1983) distinctions of a three-level model (Kintsch 1998; Bower
and Morrow 1990; Glenberg and others 1987b). In Graesser & Zwaan’s Event-Indexing model, events are the focal points of situations expressed by language and are connected in memory along five (and possibly more) dimensions of time, space, causality and intentionality. Readers mentally index story events along these five independent dimensions, and each of the three levels of representation model made reliable contributions to explaining variance in judgments of verb relatedness (Zwaan and others 1995b).

While a wide section of the most recent contributions literature seems to follow the event-indexing model (Zwaan 2015; 2009; 2008; Fischer and Zwaan 2008; Yaxley and Zwaan 2007; Therriault and others 2006; Zwaan 2003; Graesser and others 2003a; Zwaan 1996; Magliano and others 2001; Zwaan 1999; Radvansky and others 1998; Zwaan and others 1995a; Zwaan 1994; Zwaan and van Oostendorp 1993), the alternative is the construction-integration model (Kintsch 1988).

The Construction-Integration model takes a mostly (Kintsch 1992) bottom-up approach, wherein constructions are initially formed acontextually. Thus, meanings are activated, propositions are formed, and inferences and elaborations are produced — in short the representation is constructed — without regard to the discourse context. However, this initial step takes only milliseconds, and creates network of interrelated items, which can be integrated into a coherent contextual structure through a process of spreading activation. Immediately following in the second stage, the integration phase takes place, wherein the range of possible candidates are evaluated for selection, using discourse context. The construction-integration model thus contrasts an expectation-based, predictive views of discourse comprehension (Landauer and others 2013; Kintsch 1998; McNamara and Kintsch 1996; Kintsch 1988; Staczk and others 1985; Kintsch 1985; Guindon and Kintsch 1984; Van Dijk and Kintsch 1983; Kintsch and Van Dijk 1978).

Other notable discourse comprehension models include the structure-building framework of Gernsbacher (Sanders and Gernsbacher 2004; Gernsbacher and others 2004; Graesser and others 2003b; Gernsbacher and Foertsch 1999; Gernsbacher 1997; Gernsbacher and others 1995; Gernsbacher 1991) and Garrod & Sanford’s Memory Focus Model focuses on referential coherence and anaphoric
resolution (Garrod and others 1994; Morrow and others 1990; 1989; Garrod and Sanford 1982; 1977).

Thus, certain general features that are fundamental to discourse processing are captured within all of these models. As text is processed, new information is integrated with representations of information derived from previous text at the level of the sentence and the discourse. Without this integration, reading has no coherence. During this process, some parts of a text may be maintained in short-term memory, while other parts may ‘decay’ in short-term memory and be lost, or incorporated into a new level of long-term memory representation. The reader's prior knowledge will influence the types of representations that are formed and the types of processing that is undertaken. These models imply or directly stipulate a role for both working memory and long-term memory in this process of integration; previously processed information must be maintained in some form, and is available for further manipulation once new text is encountered. (Ledoux and Gordon 2006)

The features, widely nominated for discourse-processing situation models, are a very good match for those of the CDG+ Discourse Processing Model, which propose an integrative framework reference, coreference, coherence, perspective taking, memory, problem solving, updating knowledge, and learning. Werth never actually calls his approach a discourse processing model. In fact, in a monograph of over 150k words, the term only appears in the title of two of Teun van Dijk’s papers, and is otherwise never mentioned directly. Werth does note the similarity between van Dijk & Kintsch’s situation model and the text world, in an endnote (pp 63). Thus, despite the fact that Werth never positioned his work as a Discourse Processing Model, since he did recognize that Text Worlds are a sort of situation model (and vice-versa), and since Text Worlds and situation models match upon many features, I will argue that CDG+ is a fully fledged Discourse Processing Model.

3.2 - Semantics

Philosophers and linguists have been captivated by different things about language. The philosopher’s problem of intentionality, or how reference is possible
using arbitrary symbols led to a focus on truth and reference (Coulson 2001). The linguists problem of how a finite system can generate infinite meanings led to a focus on compositionality (Coulson 2001), wherein a language is compositional if the meaning of a complex expression is systematically related to the meanings of its constituents (Frege 1980).

### 3.2.1 - Formal Semantics

The complimentary interests of philosophers and linguist meet in formal semantics (Coulson 2001), which provides algorithms that seek to establish that the truth of a complex expression depends on the truth of its components, as in propositional logic. Formal semantic theory attempts to address both compositionality and intentionality by accounting for the relationship between the meaning of a sentence and facts about the world that support the truth of propositions of a sentence. Its chief concern is proving truth conditionality for sentences (Coulson 2001). However, formal semantics is unable to resolve certain references and finds it impossible to identify truth-conditions for a great number of common language uses.

And of course, as analysts are reading, or skimming in order to quickly judge the relevance and meaning of texts, they are not likely be attempting to resolve a term’s logical referent, adjudicate the domain of quantifiers, or determine a proposition’s truth conditionality to understand what the text means. Moreover, since online meaning construction during deeper analytic reading seems to be doing something quite different, we need to look elsewhere for a more satisfying account of semantic meaning construction.

### 3.2.2 - Model-Theoretic Semantics

In the model theoretic semantics of modal logic, a model is something like a strictly defined possible world. This strict definition is both a strength and a weakness. You may know what is in it based on the definition of the model. But there is a crucial difference between logical models and cognitive or psychological models (Werth 1999f).
Logic is concerned mainly with possible relationships between logical functions (Werth 1999f). It is less concerned with the arguments of (or whatever is joined by) these functions. The semantics of such logic models is studied as model-theoretic semantics (‘Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness’ 1983b), which claims simply to be able to study the logical relationships that obtain given a tightly defined model. Thus, model-theoretic semantics does not consider the assessment of actual truth in the given model, but only calls the logically permissible relations (logical) possible truth.

3.2.3 - Cognitive Semantics

Since these basic model-theoretic logic models are so underspecified, scholars with an interest in actual as opposed to possible truth may find them trivial. Werth suggests improvements to these basic possible worlds models, proposing that language users actually use a more rich cognitive model that is capable of providing people both affordances and methods to work through linguistically described states of affairs that approach the richness and complexity that humans are used to in the real world beyond language (Werth 1999f).

Thus, given the shortcomings of logical model-theoretical and formal truth-functional approaches to semantic meaning, the cognitive semanticist takes a different approach toward meaning in natural language. Lakoff and Fillmore, and indeed all researchers in Cognitive Linguistics Analysis, point to established commonalities in the interpretation of language and general cognitive processes involved in the comprehension of objects, actions, and events in the world around us (Johnson and Lakoff 1980; Lakoff 1968; 1987; Petruck and others 2004; Feldman and Narayanan 2004; Fillmore and Sato 2002; Fillmore and others 2003; 2002a; 2002b; Fillmore and Baker 2001; Fillmore and Atkins 1998; John B Lowe and others 1997; Fillmore and others 2001; Fillmore 2006; Baker and Sato 2003; Baker and others 1998). In language comprehension and meaning construction, linguistic data suggest people access frames (See Sections 3.3.4 and 3.5.4 of this Chapter) from long-term memory to construct meaning for utterances (Gavins 2007).

3.2.4 - Empirical Frame Semantics
Fillmore outlines a research program in empirical semantics that is distinct from formal semantics, that he calls frame semantics (Fillmore 2006). He presents a view on frames similar to their originator, Minsky C, though he fleshes the concept out and highlight some of the pitfalls a robust theory would have to address.

Fillmore points out that the term as used in cognitive psychology has different origins (See Section 3.3.4, Frame Representations). A frame, in Fillmore’s current usage for describing linguistic meanings, is a system of categories structured in accordance with some motivating context (Fillmore 2006). In Fillmore’s view, words represent categorizations of experience. These categories are motivated by an underlying situation that occurs against a background of knowledge and experience. Fillmore’s methodology looks for the reasons language communities may have found for creating the category created by the word, and explains the word’s meaning by describing that reason. (Fillmore 2006)

3.3 - Knowledge Representation

According to Brachman (Brachman 1992), while the discipline of Artificial Intelligence (AI) has always rested on the formal representation of knowledge, Knowledge Representation (KR) only really emerged as a subfield in its own right in the eighties (Brachman 1992). As a subfield of AI, KR is chiefly concerned with developing a “sufficiently precise notation for representing knowledge (Mylopoulos 1981).” This notation is referred to as a knowledge representation scheme and is typically used to specify a knowledge base that contains facts. This knowledge base is a highly reductive domain model that is intended to be representative of a specific slice of reality.

Mylopoulos draws a distinction between KR-type AI approaches that embed expert knowledge and methods of manipulating this knowledge, and earlier “general purpose heuristic search techniques (Mylopoulos 1981).” Winograd (Winograd 1980) offers a taxonomy of knowledge representation schemes that draws distinction chiefly between declarative and procedural approaches. Mylopoulos further subdivides declarative schemes into logical and (semantic) network types (Mylopoulos 1981). Brachman notes that by the early nineties the KR community had moved away from procedural representations “towards a radically
declarative worldview” (Brachman 1992) and has changed its focus to the contents of the representation (what they say) and less about how to control procedures that process them.

The following subsections offers some details on each approach. Note that most KR systems do not fit neatly into one approach (e.g., procedural, declarative logical, or semantic network) and generally have certain aspects of several. Frames are an excellent example of a plural approach (See Section 3.3.4, Frame Representations).

Accordingly, this section provides background reviews the frames approach to knowledge representation first elaborated by Minksy in 1974 (Minksy 1973). Critiques, modifications, and widespread adoption of this approach across disciplinary boundaries are discussed14.

3.3.1 - Logical Representations

Logical representation schemes “employ the notions of constant, variable, function, predicate, logical connective and quantifier to represent facts as logical formulas in some logic (Mylopoulos 1981).” In this view, a knowledge base is a collection of logical formulas which provides a partial description of a world. Logical formulas are the atomic units for knowledge base manipulation in such schemes, so introducing or deleting logical formulas is what modifies the knowledge base.

Mylopoulos cites four main advantages, two disadvantages, and one example of logical representation schemes. Their first advantage is the “availability of inference rules in terms of which one can define proof procedures (Mylopoulos 1981).” Second, for logical schemes that are close to First Order Logic, clean, well-understood and well accepted formal semantics are available. However, attempts to represent real-world (not purely logical) things are not well handled in such formal semantics. Third, the simplicity of the notation employed ostensibly leads to knowledge base descriptions that are understandable. Finally, fourth, is the conceptual economy of the system, wherein each fact is represented once, independently of its different uses during the course of its presence in the knowledge base. On the other hand, the lack of organizational principles for the
facts constituting a knowledge base can make navigating large systems onerous. Also, representing procedural and heuristic knowledge is also very difficult using logical systems. The language PROLOG is an example of the logical scheme.

3.3.2 - Network Representations

Network representation schemes describe a world in terms of nodes, which represent individuals or objects, and labelled edges or binary associations in the world being modelled. According to this view, a knowledge base is a collection of objects and associations. It thus forms a directed, labelled and possibly acyclic graph. Modifications to the knowledge base occur through the “insertion or deletion of objects and the manipulation of associations (Mylopoulos 1981).”

Whereas in early network representation schemes, representation of new types of knowledge often resulted in an explosion of association types (edge labels), “criticism has triggered a trend towards network schemes with a fixed number of primitive association types which have well-defined semantics and are descriptively adequate in that they can be used to represent any fact expressible in a logical scheme. (Mylopoulos 1981)”

Three common primitive binary association types are classification, wherein the association indicates class membership in one direction and class identity in the other; aggregation, which indicates part-hood in one direction and wholeness in the other; and generalization, which indicates subclass in one direction and superclass in the other. Thus, classification relates an object to its generic type; aggregation relates an object to its components; while generalization relates an object type to a hierarchical parent type. Generalization is particularly useful when there are many categories or object types, as it organizes categories into a hierarchy according to their generality/specificity. Another method of organizing network knowledge bases involves grouping objects and associations into partitions, or categories, which are organized hierarchically. Such partitions have been found useful in representing hypothetical worlds and belief spaces, in model-theoretic semantics.

Once more, Mylopoulos offers three advantages and one disadvantage of network schemes. To their advantage, they directly address issues of information retrieval, since associations can be used to define access paths for traversing a
network knowledge base. Primitive association types also hold great organizational potential within a knowledge base. Finally, the obvious graphical representation of network knowledge bases can make them more understandable. The major drawback of network schemes, which is at least partly due to the fact that semantic networks have been used as representational tools in very different ways, has been the lack of a formal semantics and a standard terminology for the network visualizations (Mylopoulos 1981). They have been used to represent logical formulas, to organize knowledge, to represent the meaning of natural language sentences and to represent linguistic expressions. Semantic networks are closely related to object-oriented data models (Mylopoulos 1981).

3.3.3 - Procedural Representations

Procedural representational schemes view a knowledge base as a collection of procedures expressed in some language. Most procedural schemes have been influenced quite heavily by LISP, which is purely symbolic and offers its users a dynamic run-time environment. Beyond LISP, procedural schemes are classified first on the basis of the activation mechanism they offer, and then on the control structures they exert upon procedures.

Both pattern directed procedure invocations and production systems view a knowledge base as a global database, yet each adds something a unique. For pattern directed procedure invocations, this global database is of assertions paired with a collection of theorems (or demons) that watch over the database and are activated whenever it is modified or searched. Each theorem has an associated pattern which, upon the theorem’s activation, is matched against the data about to be inserted, removed or retrieved from the database. If the match succeeds, the theorem is executed. Thus with theorems, procedures are called whenever a condition is satisfied.

In production systems, a knowledge base is a collection of production rules plus a global database. Production rules, like theorems, consist of a pattern and a body involving one or more actions. The database begins in some initial state and rules are tried out in some pre-specified order until one is found whose pattern matches the database. The body of that rule is then executed and matching of other rules continues.
There are two major differences between the activation mechanism of *theorem patterns* and *production system rules*. Order does not matter to theorem patterns, yet to standard production systems, like Markov algorithms, a fixed ordering of rules determines when will each rule be matched against the database. Also, while theorems can directly call other theorems, productions can only do so indirectly by placing appropriate information on the database.

### 3.3.4 - Frame Representations

This section discusses the early development of the **frame theory of knowledge representation**, detailing its initial proposal, and briefly mentioning some early critiques and adoptions of the approach.

In 1972-73, Terry Winograd participated in discussions about problems in Natural Reasoning at the MIT AI Lab (Winograd 1980). In Winograd’s view, one of the outcomes of those discussions was that in 1974, Marvin Minsky sketched the outlines of a theory of *knowledge representation* for human intelligence that set out to address problems with prevailing theories from *artificial intelligence* and *psychology* (Winograd 1980). In Minsky’s view, the conceptualizations of these approaches were “too minute, local, and unstructured to account — either practically or phenomenologically — for the effectiveness of common-sense thought (Minsky 1973).”

He saw that in order to account for the speed and versatility of mental activities, the *units of reasoning, language, memory, and perception* required **larger, more deeply connected structures**.

Minsky boils down his theory to this: When facing a new situation or experiencing a change in perspective, people select *structures* called *frames* from *memory*. A *frame* “is a remembered framework to be adapted to fit reality by changing details as necessary (Minsky 1973).” It is a *data structure* that stores *stereotyped situations* as a *network* of *nodes & relations*. Its *upper level nodes* are *fixed* and its *lower levels* are *dynamic*, in that they have *terminals* or *slots* that must be filled in with *specific instances* or data from the *situation* that evoked the frame.

Yet *frames* are never stored empty. All *frame terminals* start out (when first evoked) filled with **default values** that are loosely attached and thus easily
displaced by better fitting data. Of course, the situations evoking a frame may not necessarily warrant all the suppositions of that frame. Yet Minsky claims these extra details can be useful for bypassing formulaic logic and yet still producing useful generalizations. He suggests that the theory’s phenomenological power derives from its explicit inclusion of expectations and other such presumptions; which are what populate the default values of a frame.

Once evoked, the assignment or matching process attempts to fit values from the situation to each frame’s terminals, based partly on the information contained in the frame, and partly on knowledge of the system’s goal states. Terminals can specify conditions for their assignment of situational details. For example, a simple condition might require terminal assignments to have certain properties, whereas more complex conditions can specify relations among things assigned to several terminals. So terminals can exercise some control over their changing assignments.

The frame-system is a collection of related frames that is connected by an information retrieval network. Different frames in a frame-system can share the same terminals. This enables the coordination of information gathered from different viewpoints. Whenever the situation data cannot be made to fit the terminal requirements of the evoked frame, another frame can be retrieved; and the fitting process is attempted once more. If no relevant frame is found, the current situation is marked for the resource-intensive creation of a new frame.

Minsky draws deeply upon the problem of vision in artificial intelligence. Not surprisingly, his metaphors and intuitions are drawn from a thoughtful consideration of human experience of vision. In visual scene analysis, the different frames of a system reflect different viewpoints. Physically moving from place to place causes transitions between frames. For non-visual frames, it is actions and causal relations that create a shifting conceptual viewpoint; yet the metaphor of physical movement between conceptual viewpoints retains explanatory power.

The frames theory of knowledge representation has not been immune to critique. The original proponents of the approach were criticized as having difficulty communicating the concept and identifying its distinction from previous ideas (Kuipers 1975). Early evidence suggesting that people use three-dimensional models and integrate several views into a single predominating visual model spurred Feldman to reject the frames theory based on the published supposition
that frames “store a large number of separate views in purely symbolic form” (Feldman 1975). Nevertheless, the utility of this conceptual approach was taken up in linguistics by van Dijk (Van Dijk 1977) in his discussion the cognitive foundations of **pragmatics, or the study of the context of discourse**. Van Dijk described a simple process of comprehension for both **knowledge frames** and for **speech acts** (Austin 1962).

Werth (Werth 1994; 1995a; 1995b; 1999n) developed a **Discourse Analysis** theory that is based directly on the work of Minsky, Fillmore, and Van Dijk. **Cognitive Discourse Grammar** models human **language comprehension** using conceptual structures described as **experientially grounded frames**. This model of **online discourse processing** is based on notions of **mental representation** found in **cognitive psychology and artificial intelligence** that shares the experientialist principles of **Cognitive Linguistics Analysis**.

Berkeley’s FrameNet project in corpus-based computational lexicography began in 1996 (Baker:1998tf; Lowe:1997wc; Fillmore:2003ux; Choudhury and others 2000). FrameNet is distinguished by a commitment to identifying corpus evidence for semantic and syntactic generalizations, and to representing the valences of its target words (mostly nouns, adjectives, and verbs) in which the semantic portion makes use of frame semantics. “The **FrameNet database** contains (a) **descriptions** of the **semantic frames** underlying the **meanings** of the words described, and (b) the **valence representation** (semantic and syntactic) of several thousand **words** and **phrases**, each accompanied by (c) a representative collection of **annotated corpus attestations**, which jointly exemplify the observed linkings between **frame elements** and their **syntactic realizations** (e.g. grammatical function, phrase type, and other syntactic traits) (Baker and others 1998).” More recently, Fillmore (Fillmore 2006) has outlined a research program in **empirical semantics** that is distinct from **formal semantics**, that he calls frame semantics.

### 3.3.5 - Propositional Representations

Since the 1960s, various computational approaches to meaning in cognitive science have developed **propositional theories** of meaning. In their original formulations, such approaches manipulated abstract, context-less, amodal symbols that were arbitrarily related to their referents within a **predicate calculus** (Pylyshyn
and Feldmar 2014; Fodor and Pylyshyn 2014; Pylyshyn 2009; Fodor and Pylyshyn 1988; Pylyshyn 1975). While these approaches may have been well suited to such formal domains as logic, mathematics, or computer science, not even the modern discourse psychologists who actually support a symbolic approach would argue for the psychological plausibility of classical symbolic models like ‘mental language’ that are detached from referential content (de Vega and others 2008a).

More modern symbolically oriented psychological accounts of language understanding define *proposition* as something like a *mental representation of an assertion of an expression in propositional calculus*. To modern discourse psychologists, *propositions* consist of a *relational term* and its *arguments*, which could be unary or n-ary. The representation of propositions commonly look like nodes (e.g., arguments) connected by links (e.g., relational terms) as small networks. Derived from *sentences* or *utterances*, in CDG/+ (e.g., both original CDG and extended CDG+), propositions are considered smallest linguistic units deemed to have truth values.

(Glenberg and others 1999) notes three empirical arguments for propositional theories of *language understanding*. First, recall of text often seems to reflect propositional units and gist (Kintsch 1974). Second, priming seems to follow propositional relations rather than contiguity (McKoon and Ratcliff 1980). Finally, processing negated sentences is more difficult (MacDonald and Just 1989). Glenberg ultimately presents an empirical challenge to propositional theories by establishing that difficulty processing negated sentences is due to their presentation out of *context*. When presented with a supporting pragmatic *context*, Glenberg finds that the difference in sentence-processing times disappears.

While Glenberg hoped to topple the dominant propositional theory of meaning in favour of an *embodied approach*, (King and others 2014) argue that while propositional account is flawed, it is nevertheless indispensable. CDG/+ agrees, and seeks to amend the *propositional approach* so as to resolve the most seriously problematic objectivist assumptions that underlie an unrehabilitated approach to propositions based on *predicate calculus*. Briefly these assumptions are that:

- **Readers** can distinguish between *propositions* and *non-propositions*; they can assess which expressions can bear *truth values*; and they can identify *truth values*. 
• **Constants** have an *extension* (referent) in the *real world*.

• **Individual constants** are uniquely referring *expressions* and have a known *extension* (determined referent).

• **Qualifiers** designate the *extent* to which the *class* is evaluated by the procedure designated by *class constants*.

• **Extension** (determination of referents) of *constants* must be known from the given expression.

• **Reference** is just designating the extension of a constant (determining the referent) in the *real world*.

By far the biggest problem noted is that traditional approaches based on *predicate calculus* have no machinery for taking *context* into account. *Constants* are not always uniquely referring expressions with a known extension. *Reference* is not a straightforward matter of being able to determine the *referent* of a *constant* in the real world. *Truth-values* cannot always be obtained. While each of these problems is significant, CDG proposes that they can be resolved by treating *context* as central in the resolution of *reference* and by rehabilitating the method used to assign *truth conditions*. (Solutions to these problems with propositions and the alternate definition of propositions used in CDG are discussed at length in my forthcoming monograph on CDG+.) Thus, CDG employs a different conception of proposition.

### 3.4 - Reasoning

This section considers three theories of reasoning, describing their controversy in artificial intelligence (AI) and cognitive psychology. This controversy parallels and further specifies the challenges already noted between philosophy and linguistics in semantics which have lead to the emergence of cognitive semantics approaches, and between compositional and constructivist approaches to discourse processing in language comprehension.

#### 3.4.1 - Formal Rules for Reasoning

The first class of theories on reasoning assumes that reasoning depends on *formal deductive rules of inference*, like logical calculus. The benefit of *rules* is that they explain in principle how people can *reason about anything* regardless of its
content. Problems with this approach arise when incorrect inferences are made from conditionals, and when trying to derive common inferences from daily life using such formal mechanisms. So much reasoning is not deductive that formal rules are widely considered too brittle even with non-monotonic remedies. But by far, the biggest problem for formal systems of reasoning is propositional content of everyday (Johnson-Laird and Girotto 1998). Another problem that can be especially troublesome for analysts using deductive reasoning over propositions is that people are not typically aware of how they came to their conclusion (Johnson-Laird and Girotto 1998; Khemlani and Johnson-Laird 2009; Quelhas and others 2010).

3.4.2 - Content Rules for Reasoning

The second class of rule-based theories distinguishes itself from deductive rules by explicitly recognizing the central importance of content through the formulation of content-specific rules of inference. This approach produces large knowledge bases of rules derived from interviewing human experts, and was widely used to create expert systems. This approach is, by analogy, similar to amassing exemplars within a connectionist network where inference is more like recall. (Johnson-Laird and Girotto 1998)

The benefit of content-specific rules is that they explain how contents of premises affect reasoning. Unfortunately these approaches offer no explanation of deductive or non-deductive reasoning (‘Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness’ 1983c).

3.4.3 - Mental Models for Reasoning

that nearly all theories of mental models concern the same underlying reality that the mind is a symbolic representation system (Johnson-Laird 1983).

Johnson-Laird generalizes mental models as working models of the world used by humans to understand ('Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness' 1983e). More specifically, they are conceptual spaces that humans use to work out probabilities and primarily deductive inferences. Insofar as mental models involve the creation of analogue mappings that are isomorphic with the original situation, these representations can be used directly to reason about the original situation.

Johnson-Laird’s original mental models resemble the formal models of semantics in their commitment to compositionality and in the way some form of truth conditionality obtains within the model (Coulson 2001). While semantics has largely moved beyond the assumption that human language is fully compositional, the idea has had a big impact on disciplines beyond linguistics and philosophy, as in cognitive science, which has, on some accounts, bogged down in the traditional linguists’ compositional focus (Coulson 2001).

Mental models have three principles:

1) Reasoners use the meaning of premises and their knowledge to build mental models of the possibilities evoked by the premises

2) Mental models are as iconic as possible, but include some symbolic components

3) Mental models represent what is true but not what is false. Reasoners can flesh out model explicitly.


The principle value of mental models (or in Johnson-Laird’s terms, simply models) lies in the fact that they are simultaneously a mental representation and a
mechanism for reasoning, testing and evaluating ('Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness' 1983g). Johnson-Laird insists that mental computation requires a higher level of organization than is afforded by purely propositional representations, which do not by themselves make explicit to consciousness any high level relational structures, such as those common in discourse ('Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness' 1983h). Werth wholeheartedly agrees, proposing that Cognitive Discourse Grammar in fact models high level relational structure in the natural course of online discourse processing (Werth 1999f).

3.4.4 - Discourse Models for Reasoning

Johnson-Laird similarly argues against Euler circles or Venn diagrams as visual forms of mental models, and proposes instead the discourse model ('Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness' 1983i). He argues that the evidence shows that mental models built from language are similarly structured to those that are built by perception or memory, and that reasoning exploits discourse-built mental models in the same way as it does any other mental models (Johnson-Laird and Girotto 1998). He points out that the major problem with most other theories that use mental models as a representation of knowledge is their radical incompleteness ('Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness’ 1983j). Also, theories of discourse account typically only for how fragments of language can be translated into models ('Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness’ 1983k). Werth acknowledges this critique and offers the CDG Discourse Processing Model to remedy both of these shortcomings.

This name choice is problematic for discourse scholars, since Johnson-Laird’s discourse models neither theorize the language-phenomenon of discourse nor handle extensive stretches of text, but only uses linguistic input to build mental models from the meaning taken from individual premises. Thus, in order to reduce potential for confusion, I will henceforth refer to Johnston-Laird’s formulation of mental models built from linguistic input as discourse mental models.

Johnson-Laird observes that the case for discourse mental models has been made in formal semantics, linguistics, artificial intelligence, and in
psycholinguistics (Johnson-Laird 1983). In his view, discourse mental models render explicit the structure of the situation\textsuperscript{15} as it is perceived or imagined. The explicit content of the discourse is usually a blueprint for a state of affairs in that it relies on the reader or listener to flesh out the missing details using inference (Johnson-Laird 1983). This processes of inference can be modelled using a transparent notation that is structurally similar to the state of affairs being modelled in such a way as to create analog models of the situation described in the text (Werth 1999g; 1999b).

Unfortunately, unless people use visual proxies for the mental models they use for reasoning, they typically have no conscious access to the models themselves (‘Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness’ 1983l). This introspective deficiency is used by critics of mental models. However, in recognition of this problem analysts are trained to externalize their mental models explicitly by using various representational schemas to manipulate the evidence and generate hypotheses in the sense-making and cognitive-functional flow models of analysis. This introspective deficiency is thus not a fatal flaw of mental models, but is rather accepted as a limitation of human cognition, and one that analysts require help from VA tool builders to overcome (Heuer 1999).

3.4.5 - Rich Worlds for Reasoning

Thus, Johnson-Laird’s theory provides mental models for linguistic content. However, despite their name, discourse mental models do not robustly theorize the treatment of discursive language use or its abstraction as texts (Werth 1999f). Nevertheless, Johnson-Laird’s approach can, with some modification, accommodate the rich mental models or text worlds proposed by the original Text Worlds approach. That is to say, while mental models are not rich models, rich models can in fact be defined within Johnson-Laird’s theory of mental models. This section notes important distinctions between Johnson-Laird’s discourse mental models and CDG’s rich model text worlds.

Johnson-Laird’s theory of discourse mental models is based on three principal ideas that vary in their degree of accord with the approach taken in Cognitive Discourse Grammar. First, insofar as a discourse refers to a situation, a mental
model represents the situation described in the discourse (‘Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness’ 1983m). Werth concurs on both points.

Second, Johnson-Laird claims that the meaning of a discourse is the set of all possible situations it could describe (‘Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness’ 1983n). This is substantially different from the rich worlds Werth describes. Rich worlds are not about describing or representing the set of all possible situations. Similarly, the standard model-theoretic approach to truth-conditionality is not taken by Werth. In Text Worlds, the meaning of the discourse is nothing like Johnson-Laird claims. Instead, the discourse describes a situation, and the meaning of the propositions comprising the description of that situation is jointly agreed by participants (Werth 1999c). Negotiated propositions from the text create the text world, which provides the world setting into which all subsequent references in the text refer (Werth 1999c; 1999k). The meaning is thus the one situation described by the text and this situation is instantiated as propositions describe action in that Text World (Werth 1999h).

Third, according to Johnson-Laird, as long as a model of the discourse can be embedded in a model of the real world, a discourse is judged to be true (‘Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness’ 1983o). Alternatively, we see in Text Worlds that reference is from a text to a Text World; and in case of prototypical discourse which also includes references into the real world, Cognitive Discourse Grammar provides for references to the speakers’ jointly shared representation of reality, the Discourse World (Werth 1999k; 1999a). But in Text Worlds there is no condition that it must be possible to embed any given Text World into any given Discourse World. The discourse is taking place in the actual world, and the contents of the text of that discourse do not have to refer to anything except the proper domain of their reference (e.g., their text worlds) in order to warrant their truth of their existence.

In a Text World, propositions are assessed for truth with respect to that world. But true in the Star Wars galaxy is not necessarily true in any complex conditional counterfactual hypothetical situation an analyst might imagine, and that complex counterfactual need not be true in this real world for it to be true in its own text.
world. Thus, claims of truth require compact, portable descriptions of the situation within which truth holds. TextWorlds provides just such a mechanism.

The rich worlds of Text Worlds assume that inferences made over personal and culturally shared frame-knowledge will lead to justifiable (valid or true) conclusions. This assumption could prove problematic if the inferencing process were instead left unrecorded and unevaluated. However, TextWorlds explicitly accommodates and records both evoked frame-knowledge and inferences, by incrementing both into the Common Ground of accepted facts. This permits subsequent on-line modification and updating (e.g., correction, or deactivation) of known or inferred Common Ground facts via interaction with TextWorlds visualizations. Consequently, inferences made through frame knowledge can prime concepts even prior to their explicit articulation in the text. Frame priming cannot be accounted for in standard approaches to Discourse Analysis.

3.5 - Decision-making

Normative prescriptive models of decision making are currently the basis of analytical training in both Intelligence (Heuer 1999) and Public Policy Analysis (Nagel 2001). A common and recurring critique of these idealized prescriptive models is that, far from describing the typical workflow, these models present an illusory standard of the decision and analysis process (Howlett and Ramesh 1995).

3.5.1 - Classic Decision Making

Johnson-Laird observes that the classical theory of decision making, despite its presumed status as a specification of rationality, does not begin to explain the mental processes underlying decisions (‘Mental Models: Towards a Cognitive Science of Language, Inference, and Consciousness’ 1983p). Hoffman suggests that the classic Decision-Analytic model is solely representative of novice decision-makers (Hoffman and Militello 2009), who must function under conditions of certainty while evaluating alternatives (Edwards and others 2007). Since Visual Analytical tool users do not typically operate in conditions of certainty, classical decision making theory is not the best
3.5.2 - Naturalistic Decision Making

The paradigm of **naturalistic decision making** can be seen as a reaction to the normative and prescriptive models of decision-making (Hoffman and Militello 2009; Klein 2008; Mosier and Ute 2007). **Naturalistic decision making** approaches involve (Hoffman and Militello 2009):

1) A focus on the examination of *decision making* in everyday situations, both routine and non-routine situations, both simple and complex;
2) A focus on *decision making* by experienced, knowledgeable individuals;
3) The examination of *decision making* in real-world job contexts that are especially important to business, government, and society at large. (Hoffman and Militello 2009)

This paradigm has produced various **new models of decision-making**, including recognition-primed decision-making, a macro-cognitive model of expert reasoning, and the theory of situation awareness (Hoffman and Militello 2009; Klein 2008). Recognition-primed decision-making explains how **schemas** enable quick situation recognition and action-oriented decision-making. Research in naturalistic decision making shows how chess-masters, soldiers, firefighters, business intelligence analysts and many other experts build these **schematic knowledge structures** and use them to rapidly make sense of a situation\(^{16}\) and formulate an action.

3.5.3 - Recognition Primed Decision Making

Experts solving problems in familiar domains can use **general knowledge** captured in **schemas** as rules or concepts to **find good solutions quickly** (Thagard 2005). However, findings in the psychology literature indicate that while experts are good at what they do in their established work context, they are **rarely** so good out of that context or with equivalent tasks in a parallel context (Hoffman and Militello 2009). This observation is the basis for the Recognition-Primed Decision-making approach.

Overall, the Recognition Primed Decision-making model combines aspects of the Duncker (Duncker 1945) model, mental modelling, progressive deepening of conscious deliberation, the **classic Decision-Analytic Model** and broad findings
indicating that direct recognition of situational factors leads to action plans (Hoffman and Militello 2009).

Drawing from studies of expertise of firefighters, nurses, military commanders, and weather forecasters, Hoffman (Hoffman and Militello 2009) concludes that experienced expert decision-makers typically employ recognition-primed decision-making, wherein features of the situation are rapidly evaluated and compared with recalled events that serve as prototypes. In this regard, recognition-primed decision-making model closely resembles the constructivist account of meaning construction used in Cognitive Discourse Grammar.

It is worth noting that the analytic, decision-making, and reasoning models described to this point universally share a reliance on schemas, and on prototype events stored in long-term memory as frames. The Sense-Making model of analysis (Russell and others 1993; Pirolli and Card 2005) implicitly recognizes that analysts work with procedural schemas of the cognitive task domain. The Cognitive Flow model of analysis goes farther, and explicitly accords schemas for propositional and procedural knowledge a central role in the day to day analysis process. The Cognitive Discourse Grammar model of discourse comprehension theorizes rapid context-reading, situation-building and sense-making based in part on schemas containing both procedural and propositional knowledge and in part on the propositions contained in text. (Thagard 2005; Hoffman and Militello 2009; Russell and others 1993; Pirolli and Card 2005)

3.5.4 - Prototype Recall in Decision Making

By aligning the Recognition-primed Decision-Making model with Cognitive Discourse Grammar model of discourse processing, we conclude that experts resolve situation features in both real world and text world contexts by rapidly evaluating and comparing with recalled events that serve as prototypes in order to come to actionable decisions.

Understanding descriptions of the process of decision-making offered by approaches beyond classic decision-analytic theory will set the stage for developing Visual Analytics tools that are primed to help analysts record and improve how they make decisions using discourse models and employing knowledge schemas.
3.6 - Cognition Synopsis

This Chapter provides an introduction to a selection of research domains in the Cognitive Science Literature that are principally relevant to understanding the process of language usage and comprehension. This Chapter has likewise highlighted important concepts and approaches to theorizing higher-order cognition, particularly in Discourse Comprehension, Semantics, Reasoning, Decision-making, Knowledge Representation. The rationale for including selections from these literatures is simply that any transdisciplinary research must establish foundations broadly to ensure that references resolve locally.

In this chapter, I have laid the foundation for the argument that understanding the process of meaning construction offered by approaches beyond formal semantics using a range of knowledge representations will set the stage for the development of visual and analytical tools that are primed to help analysts record, understand, and improve the innate and personal process they use to build meaning using language.

Even the idea from formal semantics that all language is innately qualified would allow developers to build tools that help analyst to identify the precise level of generality at which each claim is warranted, by identifying the specified domain for each processed linguistic reference as falling within the TextWorlds model. This could prove useful in analytical reasoning beyond simply improving reading workflows during analysis.

Thus, the research by cognitive and frame semanticists like Fillmore, Lakoff, Coulson and Werth has potential to inform visual and analytical tools that focus on Evoking and creating schemas in long term memory, in analytical Reading, and Search, particularly of evaluative judgment.

On the basis of constructivist accounts of language comprehension, VA tools that aim to help analysts with the meaning construction phase of reading in analysis must take existing personal knowledge frames into account. If frames, or other schemas from long-term memory are cued by text, then a system which helps an analyst index, interact with, and record frame cuing and interaction could contribute to better understand of meaning comprehension during analysis.
Efforts to scaffold analytical sense-making should also consider how existing lexical resources like PropBank (Eleni and Aravind 2004), FrameNet (John B Lowe and others 1997; Baker and others 1998; Feldman and Narayanan 2004) and VerbNet (Enhong Chen and Wu 2005), in addition to other lexical resources like WordNet (Gangemi and others 2002; Tonelli and Pighin 2009) might be dynamically accessed to provide contextually cued and empirically grounded resources to support analytical meaning construction while reading.

It is clear that the CDG+ idea that frames and schemas guide language understanding based on personal and cultural experienced stored in long-term memory is not unique in cognitive semantics or constructivist accounts of on-line meaning formation from discourse processing (Fillmore 2006). However, while I could be wrong, it certainly seems like the methodological description and computational implementation of the cognitive semantic, constructivist Cognitive Discourse Grammar model of discourse processing with the explicit purpose of supporting and scaffolding real on-line meaning construction would be novel in the fields of Cognitive Semantics, Discourse Processing, Cognitive Linguistics Analysis, and Visual Analytics.

This interactive predictive model (Thomas and Cook 2005) of analytical reading comprehension conceptual structure networks will require VA tools to computationally prepossess text in order to present and probabilistically rate the most likely segregations of propositional figure & ground that constitute a viable basic readings of texts for the analyst to adopt as their own, and subsequently interact with, refine, and truly personalize.

Research on reasoning foregrounds the cognitive processes of deductive inference, model building, model manipulation, reference determination, truth assignment, inductive inference, and analogy. This literature underscores the important contribution of reasoning in the Cognitive-Functional Flow Model of Analysis, particularly in the cognitive-functional steps of judging, inferring, building, organizing, schematizing, instantiating, interacting hypothesizing, reasoning.

Understanding approaches beyond formal deductive logic will set the stage for developing Visual Analytics tools that are primed to help analysts capture and improve their reasoning process, using rich discourse models. Visual Analytics can
build tools for **Judging truth** in the context of the conditions specified by the text. These **rich world models** provide explicit mechanisms for reasoning by:

1) Modelling the **relational structure** of text at and above the level of propositions.
2) Modelling the **situation** described in a text.
3) Formulating model-based inferential conclusions.
4) Enabling articulation of models and incorporation of facts that disprove **model-based inferential conclusions**.
5) Assessing validity of model-based inferential conclusions.
6) Qualifying acceptance of non-verifiable **model-based conclusions**.

Most importantly **rich world mental models** provide a space created by text for reasoning about the contents of the text through **interaction** with the model.

The research domains of **Semantics, Language Comprehension**, and **Discourse Processing** provide insight into processes of **judgment** in **meaning construction**, **reference determination**, **truth assignment**, **inductive inference**; processes of **memory evocation** of personal **procedural** or **propositional schemas**, creation of individual **propositional situation-types schemas**, creation of **generalized situation-types schemas** as **frames** of cultural knowledge stored in long-term memory; processes of memory-scaffolding in the creation of a **propositional knowledge base** that is automatically annotated with the appropriate domain of reference to enable subsequent reuse across texts. Each of these cognitive processes will be central in the approaches taken in the following Chapters toward addressing different aspects of the Problem Situation.
Chapter 4 - The Analysis Literature

The author of this dissertation has also, over the course of a graduate research career, trained as a policy analyst and worked professionally as a data analyst while conducting research and developing expertise in Visual Analytics. This practical working experience in analysis is brought to bear in selecting domains for consideration and review in Chapter 4, The Analysis Literature. Consequently, this Chapter focuses on the analytic domains of Cognitive Linguistics Analysis, Discourse Analysis, Computational Text Analysis, Visual Analytics, and Intelligence Analysis.

Since the stated goal of the field of Visual Analytics is to produce a class of tools whose purpose is to generate insight and novel discovery, this discussion in this Chapter will focus on mechanisms from these analytical domains that support innovation using higher-level cognition to produce analytical conclusions, advice and other analytical products. This discussion ultimately aims to clarify how the cognitive domains discussed in the previous Chapter 3, The Cognition Literature, can impact the design and evaluation of visual analytic information systems.

The discussion of this Chapter wraps up with a consideration of the Sense-Making model of intelligence analysis described by Pirolli & Card (Pirolli and Card 2005) and noting certain of its limitations as opportunities for refinement and further study that are realized in Chapter 5, Flow, with the articulation of the Cognitive-Functional Flow Model of General Analysis. The Flow model describes analysis in common and general terms as both a process and a product that is produced over phases of preparation and execution. Each task is viewed as an overlapping network of flowing sub-tasks.

To begin, in common usage, analysis is both product and process. Especially in its conventional manual and analog configurations, this process of analysis involves building and manipulating models by continuous data selection, collection, and evaluation; most often with the purpose of providing time-critical situation-specific insight and advice. Thus, analytical products can range in complexity and elegance from simple labels or messy piles to highly involved statistical, linguistic, social, or economic models. They can look like anything from interim utilitarian
externalizations of cognition on the back of a crumpled napkin to polished official reports and summaries prepared for circulation.

To the trained eye, examination of analytical work product may provide some general insight into its productive process. But pragmatically speaking, analytical products neither necessarily nor universally provide a retraceable road map to their construction. In analytical practice this is known as provenance. Thus, it would be beneficial if analytical products could retain a breadcrumb trail, or history of the procedures employed to produce them. It is this observation that has motivated the creation of the prototype system for procedural tagging undertaken in Chapter 7, m\(^3\) [em-cubed].

Describing the analysis process in the modern computational context, analysis is an iterative, multi-stage (Hutchins and others 2003; Pirolli and Card 2005; 1995), multi-tool process. In both (analog and digital) cases, the productive process and its cognitive steps are ephemeral and perdurant whereas its product is typically endurant. My dissertation describes, designs and implements methods and systems for turning process into usable product, from the perspectives of Cognitive Linguistics Analysis, Discourse Analysis, Visual Analytics, and general all-source analysis.

Figure 4.1. A Word Cloud view of Chapter 4 - The Analysis Literature. A. Frequent Deictic and relational terms. B Frequent conceptual terms.
4.1 - Cognitive Linguistics

The fundamental hypotheses of Cognitive Linguistics (Croft and Cruse 2004d; Geeraerts 2006) are synthesized to four points:

1) Language is not an autonomous cognitive faculty.
2) Grammar is conceptualization.
3) Knowledge of language emerges from language use.
4) Linguistic meaning is perspectival, dynamic, flexible and encyclopedic.

(Croft and Cruse 2004d; Geeraerts 2006)

The first hypothesis implies that the conceptual structures and processes proposed for language should be structurally the same as those found in non-linguistic human cognition. Broad findings in cognitive psychology are the basis of this Cognitive Linguistics principle that linguistic meaning is derived from non-autonomous cognitive processes. These modern findings still support the principled assumption in CDG, of the structural similarity between general human knowledge and linguistic knowledge.

Hypothesis 2, that grammar is conceptualization also underscores a close connection to cognitive psychology. Research on categorization (Anderson 1991) has been inspired by (Rosch and Mervis 1975) and research on construal operations (e.g., the manner of selecting from a range of possible meanings) (Talmy 2006; Riemer 2005; Croft and Cruse 2004a; 2004c; Fauconnier 1998) was inspired by the work of Gestalt psychologists (Wertheimer and others 2012; Helson 1926; Humphrey 1924; Koffka 1922; Werthheimer 1912).

The hypothesis (3) that knowledge of language emerges from language use draws from the literature on embodied cognition (Fischer and Zwaan 2008; de Vega and others 2008b; Bittencourt 2007; Zwaan 2003; Wilson 2002; Rosch and others 1992). Geeraerts (2006) contends that in Cognitive Linguistics, coverage of this hypothesis represents both a weakness and an opportunity for improving cognitive linguistics approach (Geeraerts 2006). Thus, the focus of language researchers must somehow be extended from focus from the mind of language user to include 1) the actual communication function of language, 2) a deeper understanding and theory of social interaction in discourse, and 3) the implications of social and cultural structures retained in memory on language understanding. I
will argue that the CDG+ Model contributes in this area in *Cognitive Linguistics*, by addressing these three needs in turn.

Hypothesis 4 is really a series of hypotheses on linguistic meaning. First, *Cognitive Linguistics* attempts to analyze the fact that **perspectivizations** are observed everywhere in language, whereby we each see the same story construed from different **point of view**. Since meanings change as the world and our experience of it changes, the structure of language must therefore not be rigid and stable, but flexible. Finally, insofar as linguistic meaning entails human interaction with the world, it involves the whole human in its construction, which both draws from and reflects our overall experience, including non-linguistic world knowledge that is integrated with our various other cognitive capacities in long-term memory. CDG+ explicitly shares each of these theoretical premises with *Cognitive Linguistics*.

Thus, linguistic analysis and methods which conform or contribute to these basic hypotheses are here considered to be Cognitive Linguistic analysis. While canonical Cognitive Linguistics employs various methods (fMRI, EEG) common in discourse psychology, other methods like **Discourse Analysis** are less common.

### 4.2 - Discourse Analysis

**Discourse analysis** distinguishes itself from *general linguistic analysis*, which focuses on analyzing **sentence-level phenomenon**, by analyzing language phenomenon **above the level of the sentence**. Practitioners of *discourse analysis* in linguistics generally share the objective of improving **understanding of language** and **characterizing extended language events**.

In *general linguistic* usage, a **discourse** refers to any **unit of text above the sentence**. What this has meant in *general linguistic practice* is that the data analyzed as **discourse** often consists of a **set of sentences** which the **analyst** has constructed for the purpose of testing or illustrating some linguistic phenomenon, which do not necessarily exhibit the qualities expected of an extended language event. Such **artificial approaches** to discourse will not be considered here. Moreover, while in academic contexts more broadly, **discourse** may refer an
artefact of social power dynamics as a critique of social power imbalances, this usage of discourse will not be employed here.

In Cognitive Discourse Grammar (CDG), discourses are language events that are drawn from real language use, as is motivated by Cognitive Linguistic principles. This view is common among discourse analytical methods. Texts are abstract records of discourse language events. These texts must be whole but may be brief and necessarily include context.

The linguistic field of discourse analysis often uses transcribed conversation as its preferred data. However, such prototypical dialogue situations are by no means the only language data studied by discourse analysts. In this field, discourse is typically defined as the turn-based process of linguistic exchange between speakers & listeners. This is, however, simply the most obvious and prototypical real world case of an extended language event. When recorded and archived as a text, as opposed to experienced as dialogue, discourse may refer to the current collection of texts in use in an extended language event (e.g., 18th century Russian Literature or modern popular culture).

Naturally, the unit of study in discourse analysis varies according to the data analyzed, but discourse units are typically form-based, (e.g., Sentence, clause, turn, tone unit, utterance), content-based (e.g., Propositions), or action-based (e.g., Speech acts, communicative acts (Austin 1962)). Consequently, the precise phenomena of interest studied within discourse range greatly, but typically include turn-taking in conversation, or the establishment and maintenance of coherence and cohesion in discourse.

In the alternative approach to discourse advocated here, the data used for discourse analysis will include any text or collection of texts that the analyst intends to read together with the actual active process of analytic reading. Thus, when viewed as an interactive process, discourse is the process of reader engagement with an author’s text with the purpose of coming to an understanding of either the linguistic artifact itself or author’s intentions.

In Extended Cognitive Discourse Grammar (CDG+), as in many discourse analysis approaches, discourse is theorized in terms of utterances that express propositions. Thus, the first phase of analysis is accomplished at the level of the
proposition, while subsequent steps also require functional analysis of both interpropositional relations and speech acts (Austin 1962).

The phenomena studied within discourse range greatly but in the focal area specified by Cognitive Discourse Grammar, these include the establishment and maintenance of coherence and cohesion in discourse, deixis and resolution of reference and anaphora, and determination of implicature.

Thus, discourse is not simply conversation, nor is it just text. Instead, it is a general process of cognitive engagement with the conceptual contents expressed using language by participants in both immediate (spoken) and abstract (written) language events.

4.3 - Computational Text Analysis

There is naturally a wide array of computational approaches to text analysis and to managing analysis within collections of text. I have selected for discussion here one important method for dealing with the challenge of search over large collections of text, from among the many widely practiced computational methods of text analysis. This Section 4.3, on Computational Text Analysis provides a brief discussion of the key features of the Vector Space Model, and the methods of Dimensionality Reduction (or Projection Techniques) that are required for its use. This brief section has been included to assure the reader that the author is broadly aware and deeply familiar with existing computational methods of text analysis.

Though the focus of this dissertation research is on the analysis accomplished while actually reading, the discourse processing model I have selected for closer examination in Chapter 8, CDG+, is also a space-creating method that produces dedicated spaces, or Worlds, to index the events described within the text. These Worlds are also multidimensional spaces, though they are so in the same way that the real world is a multidimensional space. Namely, within the standard dimensions of time and space, objects have their multiple dimensions embedded within them as metadata features that can be stratified as belonging to increasingly specified dimensions of human experience. Thus, each dimension of human experience supervenes upon previous dimensions; so these Worlds are radically
different multidimensional spaces from the random, feature-generated multi-
dimensional spaces of produced by the class of methods described in this Section.

4.3.1 - Vector space model

The Vector Space Model (VSM) is an influential and powerful framework for
storing, analyzing, and structuring documents that was developed by Salton in
1974 (Salton and others 1975). Though it was originally developed for application
in information retrieval, today not only is the model a widely used framework for
indexing documents based on term frequencies (Salton and others 1975), the
vector-space model is also relied upon by the majority of text clustering algorithms
(Beil and others 2002).

In the VSM, each text document $d$ is represented by a vector of frequencies of
the remaining $m$ terms: $d = (tf_1, ..., tf_m)$ (Beil and others 2002). This is accomplished
in three stages; 1) document indexing, 2) term weighting, and 3) computation of
document similarity coefficients. These stages are discussed below.

**Document indexing**

Document indexing creates a vector in a high dimensional space to represent
each document or query (Börner and others 2003). The number of unique terms in
a document corpus determines dimensionality of the document vector space
(Börner and others 2003). Preprocessing often uses a stop list of common words to
remove high frequency and non-significant words from the document vector
(Börner and others 2003). This stop-word removal has implications for the CDG+
Discourse Processing Model, and the TextWorlds Method of Textual Analysis, since
the most commonly removed stop words are structural connection terms (and, or,
but, not) and indexicals (the, that, this, there, here). (The implications of stop-words
are discussed further in Chapter 8, CDG+.)

**Term weighting**

Term weighting evaluates term importance for document representation (Börner
and others 2003). Most weighting schemes like the inverse document frequency
assume that the importance of a term is proportional to the number of documents
the term appears in (Börner and others 2003). Since long documents usually have a
much larger term set than short documents, document length normalization is employed to reduce the likelihood that long documents will be retrieved more often than short documents (Beil and others 2002).

**Computing Document Similarity Coefficients**

Given any of the methods (described in the following Section) for constructing and weighting a vector space over a document collection, the next step of computing similarity coefficients evaluates a metric for this space “to represent similarity in the content of the documents” (Jardine and van Rijsbergen 1971). Very common document similarity coefficients include the Euclidean distance or cosine measures (Willett 1988) (Steinbach and others 2000) (Salton and McGill 1986). These distance coefficients have been used very extensively in cluster analysis, owing to their simple geometric interpretation (Willett 1988).

The similarity between any two documents (or between a query and a document) can then be determined by the distance between vectors in a high-dimensional space (Börner and others 2003). In general, word overlap or co-occurrence between documents indicates similarity (Börner and others 2003). The following Section provides an overview of select measures for document similarity, including Term frequency/Inverse Document Frequency TF*IDF, Euclidian distance, cosine coefficient, overlap coefficient, and query-sensitive similarity measures.

**TF*IDF**

Term frequency/Inverse Document Frequency (TF*IDF) document similarity coefficient metric is computed for each term, as the ratio of number of occurrences of terms in the current document to the total number of terms in the current document, multiplied by the number of terms in common between two documents, the formal equation is:

$$TF*IDF = TF(t) * IDF(t)$$

Where Term Frequency or $TF(t) = \frac{\text{Number of times term t appears in a document}}{\text{Total number of terms in the document}}$; and Inverse Document Frequency or $IDF(t) = \log_e(\text{Total number of documents / Number of documents with term t in it})$. 


Euclidian distance

The measure of Euclidean distance is based on the sum of the squares of the differences between a pair of documents on every dimension (Jardine and van Rijsbergen 1971). However, a major limitation of the Euclidean distance in the information retrieval context is that it can lead to two documents being regarded as highly similar to each other, despite the fact that they share no terms at all in common (Willett 1988). Consequently, the Euclidean distance is thus not widely used for document clustering (Willett 1988).

Cosine Coefficient

The Cosine Coefficient, possibly the most popular similarity measure in text clustering applications, is based on the differences in the angles of the document’s vectors from the origin of the space (Börner and others 2003; Jardine and van Rijsbergen 1971). It defines the similarity between two documents by the cosine of the angle between their two vectors (Börner and others 2003). It resembles the inner product of the two vectors, normalized by the products of the vector lengths (square root of the sums of squares) (Börner and others 2003). Sullivan (Sullivan and others 2004) claims that the cosine coefficient is known to be more sensitive (than the overlap coefficient) to variations in document length.

Overlap Coefficient

The Overlap Coefficient is computed as the number of terms in common between two documents, divided by the number of terms in the smaller document (Sullivan and others 2004). Sullivan (Sullivan and others 2004) cites recent research that supports the general utility of the overlap coefficient, especially for representing one-dimensional document similarity. However, this coefficient was not broadly cited in the reviewed literature.

Query-sensitive similarity measures

Tombros et als (Tombros and van Rijsbergen 2004) point out that conventional measures of inter-document relationships, including the cosine coefficient, fail to detect the inherent similarity between co-relevant documents because they ignore the specific context (i.e., the query) under which the similarity of two objects is judged. In (Tombros and van Rijsbergen 2004) they introduce query-sensitive
similarity measures that bias inter-document relationships towards pairs of documents that jointly posses attributes included in the query.

**Vector Space Model Summary**

Each of these methods of determining document similarity coefficients may be used in the many algorithmic variants of Vector Space Model. To review, the VSM presents the standard assessment of document similarities based on word matches (Börner and others 2003). However, it cannot detect variations in word meaning (Börner and others 2003). This is known as the "vocabulary mismatch problem" (Börner and others 2003). Other methods (e.g., Latent Semantic Analysis) that examine the context of words have been proposed to resolve this problem. Also, over time, the vector space model has developed many variations, which address specific shortcomings raised by continuous research.

**4.3.2 - Dimensionality Reduction (Projection) Techniques**

The high-dimensional data produced by methods using the Vector Space Model must be projected into a lower-dimensional space prior to clustering or visualization for computational text analysis (Zhaxue Huang and Lin 2000). Many of the standard data-reduction approaches used for generic data clustering also applicable for computational approaches to text clustering. According to Poulovich, (Paulovich and others 2008), projection techniques are generally either based on linear combinations of data attributes, defining them in a new orthogonal basis of small dimension, or on procedures that attempt to minimize a function of the information lost during the projection. Thus, most approaches to dimensionality reduction for computational text analysis can be classified as 1) **Topology Preserving Algorithms** (Shiping A Huang and others 2005), 2) **Non-linear (Minimization) Projection Techniques** and 3) **Hybrid (Linear/Minimization) Projection Techniques**.

**Topology Preserving Algorithms**

The purpose of **Topology Preserving Algorithms** is to represent high dimensional data spaces in a low dimensional space while preserving as much as possible the structure of the data in the high dimensional data space (Shiping A Huang and
The approaches selected for discussion here include the Random Projection Method, Singular Value Decomposition (or Eigenvalue/Eigenvector Decomposition), and Principal Component Analysis.

Random Projection method

The random projection method is used by the WEBSOM algorithm to compress a very high dimensional but primarily sparse term frequency vector (Azcarraga and others 2004). The elements of the document vector are normalized term frequencies. By creating a random matrix whose elements are normally distributed per column, one can compute the projection of the original document vector onto a much lower dimensional space. Azcarraga (Azcarraga and others 2004) claims that this method compresses the number of dimensions to under 1 percent of the original number, yet preserves most of the original information content necessary for effective text classification and archiving.

Singular Value Decomposition or Eigenvalue/Eigenvector Decomposition

This technique is widely used in scientific computation (Börner and others 2003). The method for its production is as follows: Given an $N \times N$ matrix $A$, if there exist a vector $v$ and a scalar value $\lambda$ such that $Av = \lambda v$. The vector $v$ is an eigenvector, and the scalar value $\lambda$ is a corresponding eigenvalue (Börner and others 2003). Eigenvector decomposition reduces the dimensionality of a high-dimensional space while preserving its structure (Börner and others 2003). Thus, for any collection of points in a high-dimensional space, the eigenvalues of their covariance matrix reveal the underlying dimensionality of that space (Börner and others 2003). Though mathematically robust, this method does not reliably place dissimilar objects far apart and also tends not to produce discrete clusters (Börner and others 2003), and was therefore not broadly used in the reviewed text analysis literature on text clustering. SVD eigenvector analysis techniques encompass Factor Analysis methods like Principal Component Analysis (Börner and others 2003), which are briefly discussed in the following section.

Factor Analysis & Principal Component Analysis

Factor analysis is an exploratory data analysis technique for multivariate data first introduced in the 1930s (Börner and others 2003). It is primarily applied to
reduce the number of variables and detect structure in the relationship between variables (or classify variables) (Börner and others 2003). Its factors can be interpreted, whereas those of Latent Semantic Analysis cannot (Börner and others 2003).

Principal Component Analysis (PCA) is an important and widely used method in factor analysis (Börner and others 2003). PCA transforms a number of potentially correlated variables into a smaller number of uncorrelated variables called principal components. Principal components are not interpretable, as compared to those of Factor Analysis. As (Börner and others 2003) succinctly explains: “The first principal component accounts for as much of the variability in the data as possible and each succeeding component accounts for as much of the remaining variability as possible.” When applied to the tasks of clustering documents according the the similarity of their contents, this method does not force each object into a single cluster; rather, objects can be classified in multiple factors (Börner and others 2003).

Identifying the principle components uses the following basic procedure (Shlens 2014):

1. Organize data as an \(m \times n\) matrix, where \(m\) is the number of measurement types and \(n\) is the number of samples.

2. Subtract off the mean for each measurement type.

3. Calculate the Singular Value Decomposition (SVD) (e.g., the eigenvectors of the covariance matrix).

The principal components \(\{\phi\}\) of a dataset are therefore simply the eigenvectors of the covariance of that dataset, sorted by their descending eigenvalues (Bailey 2012).

**Critiques**

One limitation of the classic Vector Space Models like PCA is that it does not distinguish between variance due to measurement noise and variance due to genuine underlying signal variations (Bailey 2012). Another is that, because these statistical methods collapse the innate structure of language into a feature vector, they typically cannot detect variations in word meaning (e.g., for polysemous
terms, or between incompatible senses of a given term) in what is known as the "vocabulary mismatch problem" (Börner and others 2003).

Shiping shows that relying only on the first several principal components or Multidimensional Scaling dimensions in text document visualization often leads to significant information loss (Shiping A Huang and others 2005). Landauer (Landauer and others 2004) notes that the first several principal components rarely capture all potentially interesting relations. In such cases, Shiping (Shiping A Huang and others 2005) suggests that using up to the 5th or more dimensions can improve classification and visualization accuracy, and Landauer has tuned Latent Semantic Analysis to the first 300 principle component dimensions (Landauer and others 2004).

Despite the caveats, it is most important to note that these methods represent n-dimensional data by a small number of salient dimensions, making it possible to display multivariate data on a two dimensional surface like paper or a computer screen (Börner and others 2003), and to more efficiently compute similarity indices between documents or vectors.

**Topology Preserving Projection Summary**

This section has reviewed Topology Preserving Algorithms, including Random Projection Method, Singular Value Decomposition, and Principal Components Analysis. However, these linear techniques often fail to detect non-linear structures inherent in the data like arbitrarily shaped clusters or curved manifolds (Paulovich and others 2008). Such data calls for the more computationally expensive (non-linear) Minimization Projection Techniques (Paulovich and others 2008), which are summarized the following section.

**Non-linear (Minimization) Projection Techniques**

Non-linear projection techniques solve a minimization problem such that the distances between points in the low-dimensional subspace match the given similarities as closely as possible (Börner and others 2003). Techniques summarized here will include Multidimensional Scaling, Sammons Mapping, Force-directed Placement, and Projection Pursuit, which are typically to determine the placement of cluster elements when visualizing text content in clusters.
**Multidimensional scaling**

Multidimensional scaling (MDS) is a class of techniques that tries to map instances into a d-dimensional space (d ≤ m) while trying to preserve some distance relations (Paulovich and others 2008). It attempts to find the global structure in a set of proximity measures between instances (Börner and others 2003). The general procedure follows three steps. First, the distances of all objects are determined. Next, a goodness-of-fit measure called stress is maximized to produce a low-dimensional scatter plot of the instances. Finally, the dimensions are rotated and interpreted, recalling that axis orientations are arbitrary (Börner and others 2003).

MDS is valuable for analyzing any kind of distance or similarity matrix (Börner and others 2003). However, given its high computational costs, it is suited only to small data sets (Börner and others 2003). It also cannot display relationship data, and can be difficult to interpret (Börner and others 2003). Some of these shortcomings have been addressed in modified approaches (Börner and others 2003).

**Sammons Mapping**

Sammons Mapping is a well known MDS technique. Initially it defines a function indicating the amount of information lost in the projection, and then applies an iterative non-linear optimization method based on the gradient of this function to find a (local) minimum (Paulovich and others 2008). This method can unfold data belonging to manifolds of high-dimension, though it can fail for highly twisted spaces once large distances are taken into account in the optimization (Paulovich and others 2008). It also suffers from high computational complexity and is therefore computationally expensive. To make this method more efficiently scalable, several strategies have been advanced for projecting only a subset of multidimensional instances and interpolating the remaining points (Paulovich and others 2008).

**Force-directed Placement**

The simplest MDS technique, Force-directed Placement (FDP) was originally proposed as a graph-drawing heuristic that attempts to bring a system of items connected by ‘virtual’ springs into an equilibrium state (Paulovich and others 2008). In the initial state, items are placed randomly and the spring forces
iteratively push and pull until equilibrium is reached (Paulovich and others 2008). To make FDP properly an MDS technique, the spring forces must be proportional to the distances among the d-dimensional points and to the difference between the dissimilarity among the m-dimensional instances (Paulovich and others 2008). While this method is also computationally intensive, various improvements have been suggested in the literature that bring scalability at the cost of precision (Paulovich and others 2008).

**Projection Pursuit**

Projection pursuit (PP) is an exploratory data analysis method that projects high-dimensional data into a low-dimensional subspace in a combinatorial search for a projection that best represents the high-dimensional data structure (Landauer and others 2004). This method is suited to non-normally distributed data, and to sparse matrices (Mao Gao and Wang 2007). Its key claim is that it finds an ideal projection direction (Mao Gao and Wang 2007; Keke Chen and Liu 2006).

**Minimization Projection Summary**

This section has reviewed (non-linear) Minimization Projection Techniques, including Multidimensional Scaling, Sammons Mapping, Force-directed Placement, and Projection Pursuit. However, some data requires hybrid (Linear/Minimization) Projection Techniques, which are presented in the following section.

**Hybrid (Linear/Minimization) Projection Techniques**

The specific Hybrid (Linear/Minimization) Projection Techniques selected for discussion here include Least-Square Projection, Self-Organizing Maps, and Latent Semantic Analysis.

**Least-Square Projection**

Least-Square Projection (LSP) is claimed to encompassing the best features of both linear and non-linear projection methods (Paulovich and others 2008). LSP is fast, it can be successfully employed on high-dimensional sparse spaces, it can manage complex correlations, and it produces a precise final positioning of points. It offers high precision when handling document full-text, yet it is fast enough to handle a reasonable amount of documents (Paulovich and others 2008). While an
interesting technique, interested reader is referred to (Paulovich and others 2008) for a complete account of the method.

**Self-organizing maps**

Self-organizing maps (SOM) are quite possibly the most influential neural network paradigm in information visualization (Börner and others 2003). During the learning phase, a self-organizing map algorithm iteratively modifies weight vectors to produce a map (usually 2D) in the output layer that will exhibit as best as possible the relationship of the input layer (Börner and others 2003). The SOM algorithm can deal with large volumes of noisy data (Rauber 1999). However, reading and interpreting the structure is difficult, expensive, manual process (Rauber 1999). The map learns distinctions between clusters, however exact cluster extents and characteristics cannot be determined from the standard SOM representation (Rauber 1999). Some enhanced versions assist in determining cluster structure (Rauber 1999; Ramanathan and Guan 2007), while others have automated the task of identifying and labelling clusters by their characteristics (Rauber 1999).

**Latent Semantic Analysis**

Manual methods of relating textual information involve significant intellectual effort to create links (e.g., author-selected keywords and titles, authority-assigned indexing terms, or bibliographic citations) in text (Landauer and others 2004). As a computational alternative, LSA compares the similarity of whole documents rather than just of titles abstracts by keyword co-occurrence. LSA is a corpus-based technique that employs statistical machine learning (Börner and others 2003) for text analysis (Deng and others 2005). It was developed to resolve the ‘vocabulary-mismatch’ problem of Principal Component Analysis (Börner and others 2003). Accordingly, LSA handles synonymy (e.g., variability in human word choice) and polysemy (e.g., variability in the meanings a single word can take) (Landauer and others 2004) by considering the context of co-occurring words (Börner and others 2003).

(Börner and others 2003) describes the 4 steps used in LSA. First, the method counts the number of occurrences of each word across the corpus (e.g., collection of documents). 1) These tallies are stored in a words-by-articles frequency matrix
(e.g., words as rows, documents as columns). 2) Then cell values are weighted using information-theoretic (e.g., TF*IDF, Euclidean Distance) post-processing. 3) Next, the matrix is processed using the advanced statistical technique, Singular Value Decomposition (SVD) to build an n-dimensional abstract semantic space with one vector for each original word. 4) LSA represents each document as the order-independent average of the vectors of the words it contains (Börner and others 2003). Thus, LSA discovers the global Euclidian structure (Deng and others 2005) as the SVD eigenvector of the weighted corpus term frequency matrix.

LSA extracts latent terms using singular value decomposition (SVD) (Landauer and others 2004). A latent term may correspond to a salient concept that may be based upon several keywords, for example, the concept of ‘text clustering visualization’ (Börner and others 2003), but latent terms are not explicitly interpretable, and would not actually be labelled as using a concept name in the standard algorithm. (Glenberg and Mehta 2008) points out that LSA’s effective reduction of dimensions is consequent to its novel use of SVD. Landauer observes that LSA correlations with human text-to-text similarity judgments are often empirically highest at $N \approx 300$ dimensions, and using only the top 60,000 words in the corpus (Landauer and others 2004).

Within Cognitive Science, in quarters that support an updated approach to cognition using abstract amodal symbols, some theorists take LSA as a model for the construction of meaning in language. Naturally, this assertion is hotly contested from elsewhere in Cognitive Science, especially from the supporters of embodied cognition.

**Critiques**

Deng et als (Deng and others 2005) points out that LSA seeks to uncover the most representative features rather than the most discriminative features for the purpose of document representation. Since discriminating documents with different semantics is the ostensible goal of clustering, Deng questions its optimality for application in document clustering (Deng and others 2005)).

Landauer acknowledges that without human help, LSA “often does not adequately represent the variability of meanings conveyed by predication,
anaphora, metaphor, modification, attachment, quantification, logical or mathematical propositions, or negations (Landauer 2007)."

Finally, Greenberg et al (Glenberg and Mehta 2008; de Vega and others 2008a) and other embodied cognition researchers question how much of meaning is described in the co-occurrence of LSA, and show empirically that co-occurrence is not sufficient to derive meaning from abstract, amodal, terms. Experimental evidence from many quarters effectively counter the claim that high-dimensional vector representations derived from the language stream can be an adequate account of human meaning (Glenberg and Robertson 2000).

4.3.3 - Summary

Within Cognitive Science, there is an argument from quarters that support an updated approach to cognition using abstract amodal symbols, which presents an important Vector Space Model, Latent Semantic Analysis, as a model for the construction of meaning in language. However, this assertion is hotly contested from other quarters in Cognitive Science, especially from the supporters of embodied cognition. CDG+'s TextWorlds are radically different multidimensional spaces from the random, feature-generated multi-dimensional spaces produced by the class of methods described in this Section.

4.4 - Visual Analytics

Digital media and information have grown at incredible rates over the past several decades — and the volume, velocity, and complexity of this data growth will only increase into the future. This rapidly expanding universe of information creates well-documented challenges for even highly trained knowledge workers and information users (Thomas and Cook 2005). Visual Analytics (VA) was initially envisioned as a transdisciplinary approach to this complex problem space. But after ten years, it is clear that cognitive science, computer science, and policy sciences have not yet converged in VA with a shared focus on producing applications to address real-world challenges facing analysts today. Transdisciplinary convergence, collaboration, shared focus, and real-world challenges facing analysts seem all but forgotten in the focus on producing applications.
4.4.1 - Challenges

Many serious challenges faced by modern analysts, at the inception of VA continue to impact the evolution of the field today. Enduring wicked problems include gaining on-the-ground access to analysts in the wild; selecting appropriate units of study for analysis and validating measures for them; and developing translational expertise between the lab and the field. These questions — what is analysis, how best to gain access to analysts, and how to effectively translate between domains within VA — pose real challenges to VA researchers from any discipline, regardless of backgrounds in computer science, cognitive science, or social science.

If Visual Analytics is committed to being "the science of analytical reasoning facilitated by interactive visual interfaces" (Thomas and Cook 2005), it must take a structured, scientific approach to understanding how analysts reason and how interactive visual interfaces are used throughout the analysis process. VA must consider that similarities in cognitive substrate and procedure do not net out to a singular, invariable or ideal analysis procedure. Only once the dynamic cognitive and functional nature of the digital analysis process is better understood should VA attempt to improve analytic outcomes with interactive visualization. If the tool-building super-majority in VA cannot undertake such foundational research on their own, they should collaborate and otherwise support its publication and wide dissemination within the field.

4.4.2 - Analysts

In VA, the term analyst has historically referred to intelligence, law enforcement, or military analysts and peripherally to policy or financial analysts, and only recently to other analysts. Given the difficulty of studying the prototypical VA analyst (John R. Taylor 2003) in situ, VA researchers often abstract from the characterization of the analytic context and task to identify proxy analysts. Across these domains the analytic task is characterized as “applying human judgments to reach conclusions from a combination of evidence and assumptions (Thomas and Cook 2005).” Thus, for the purposes of this research, any highly data-intensive and integrative, abstract work, such as that undertaken by analysts or scholars (e.g.,
discourse analysts or doctoral students) qualifies such those executing it as valid exemplars of general analysis.

**4.4.3 - Intentions**

In order to facilitate high-quality human judgment with a limited investment of the analysts’ time, VA research officially aims to create tools that support the most complex and time-consuming portions of the analytical process and that create interactive visual representations that instantly convey the important content of information, within a dynamic, context-responsive environment. Successful VA tools must build upon a deep understanding of the cognitive and perceptual principles underlying these reasoning, perceptual mechanisms, and interaction processes in order to "provide [goal]-appropriate interactions that allow analysts to have a true discourse with their information."

**4.4.4 - Focus**

While the field of VA began with a recognition of the multi-faceted nature of the practical challenges facing real-world analysts, efforts in the early years (2005-2015) of VA have focused more on system-building (Thomas and Kielman 2009) than on building an interaction science of human analytic discourse and cognition. This observation presents the uncomfortable question: How can VA expand its widely held definitions of VA’s primary object of study beyond visualization & tool building to include explorations of interactive, cognitive, & analytic discourse that may not be implemented to the level expected by a tool-building majority?

**4.4.5 - Purposes**

A reasonable first step might be to reaffirm consensus on the purpose of this field. VA is ultimately about facilitating a productive range of physical and conceptual actions by real people in a challenging information contexts. To this end, understanding what analysts do and the context within which they work is key. As such, it should not be difficult to accept the suggestion that VA needs to publish contributions that develop and extend methods for theorizing, observing, testing and evaluating the process of human/tool/data/human interactions among analysts doing real-world analytic work; both with and without using visual
analytic tools & techniques; at the individual level and within a larger social & organizational context. Thus, a minor mind-expanding exercise may be all that is needed to afford various emerging approaches to research on the analytic process and analytic work context their proper place in the VA literature.

4.4.6 - Metaphors

If further motivation is required to escape this particular local minimum, it may also be helpful to consider how VA has metaphorically conceived of its raison-d’être. Metaphors like information overload have proven useful for building the common ground (Clark 1992) or common operating picture necessary to build Visual Analytics as a field — by contributing to a shared understanding of the problem domain. But human problems constantly evolve beyond our metaphors’ capacity to expand understanding.

To VA researchers working within the familiar metaphorical parameters of the data deluge and massive complexity, offers an alternate view of the VA problem space, characterized simply by vast indeterminacy. By this view, in our rapidly expanding universe of information “we are presented with an almost infinite landscape of indistinguishable data elements (Kielman and others 2009).”

4.4.7 - VA Reboot

In this revised context, the challenge for VA is to enable its users to productively focus their mental energies on the determination of features that will turn indistinguishable data elements into a coherent, semantically effective & timely whole. Kielman’s indeterminacy removes the problem focus from survival mode or simply not becoming overwhelmed and foregrounds various core cognitive tasks like feature detection, classification, and generalization that are foundational to more advanced analytic tasks (e.g., Hypothesis generation, hypothesis testing, modelling, rationalization, inference & conclusion).

Thus, the problem is not simply whether users can isolate salient features in a visualization (as in the perceptual sciences), or how we can improve visualizations and interactions to better focus analysts attention on elements or subsets of interest (as in information visualization); but how can we capture and mobilize basic sense-
making at relevant points in the analytic process, and do so effectively across analytic contexts, using visual analytic tools to scaffold and support the full range analytic work.

4.5 - All-Source Intelligence Analysis

Pirolli & Card’s notional or conceptual model of the intelligence analysis process, widely known in the field of Visual Analytics (VA) as the Sense-Making model, remains the most comprehensive empirical descriptive model of investigative analysis used in VA to date. It is taken as a standard model of analysis in VA. Few other studies have attempted to characterize the operational structure of the processes involved (Pirolli and Card 2005), so it stands in marked contrast to prescriptive or normative accounts of the analysis process (Gevamay and Wildavsky 1997).

This model was created based on the empirical basis of a cognitive task analysis and protocol analysis of tasks in the intelligence analysis domain. This task domain is a vast and varied (Heuer 1999; Kees 1992). Pirolli argues the analysis process must be empirically described before leverage points can be identified that will suggest radical improvements to cognitive tasks in any given domain (Pirolli and Card 2005). These empirically described leverage points should then inform the design and evaluation of VA information systems.

4.5.1 - Expertise schemas

Intelligence analysis should be understood as an expert behaviour (Hoffman and Militello 2009; Hoffman and Millitello 2009; Pirolli and Card 2005). From the literature on expertise it is clear that experts can be expected to have drawn from extensive experience to have built a set of schemas containing patterns that typify important task features and processes. Ericsson has found that such domain specific schemas (elsewhere called scripts, frames, Idealized Cognitive Models) are more important in expert performance than personal capacities (Ericsson and Lehmann 1996).

Naturalistic decision-making (Hoffman and Militello 2009; Klein 2008) has developed a model of recognition-primed decision-making that explains how these
schemas enable quick situation recognition and action-oriented decision-making. Research in naturalistic decision making shows how chess-masters, soldiers, firefighters, business intelligence analysts and many other experts build these schematic knowledge structures and use them to rapidly make sense of a situation and formulate an action.

Pirolli suggests expert schematic knowledge structures are less likely to have problems fitting information, so compared to novices, experts spend less time creating schemas and more of their time encoding information into working schemas (Pirolli and Card 2005). Thus, Pirolli notes that much day to day intelligence work consists of extracting information and repackaging it without much actual analysis (Pirolli and Card 2005). Of course, experts also develop expert skill schemas for information extraction and restructuring that may assist at any point in those procedures.

Hutchins et als (Hutchins and others 2007) provide a thorough Cognitive Task Analysis of intelligence analysts that focuses on particularly difficult scenarios or unusual tasks. They typify this type of work as frequently forcing the analyst to work out of context (Hutchins and others 2007). Working out of context requires on-the-spot, time-pressured learning, since the experts’ schemas are domain and task specific and do not necessarily function well out of context.

Consequently, novel problem situations necessitate inventing new approaches or learning new tools whenever adapting existing methodologies and updating existing schemas fails to satisfactorily resolve the unusual task. Thus, working out of context highlights gaps in expertise. Hutchins establishes that content expertise and even procedural expertise are no guarantee for the domain knowledge and methodological tool handling expertise required to resolve novel problem situations. This analysis reinforces the (Pirolli and Card 2005) characterization of intelligence analysis work and provides further leverage points for analytical tool builders, who must now be aware that they must focus on helping analysts to understand and build new procedural and propositional knowledge schemas.

4.5.2 - Sense-making Model

In the Sense-Making model, analysis begins with external data which is searched, filtered, collected, read, extracted, related, schematized, evidenced,
modelled, hypothesized, supported, presented, and re-evaluated. This process has one set of activities around finding the information and another that cycles around making sense of the information; and iterative loops between each (Pirolli and Card 2005). Pirolli’s diagram of the Sense-Making model of analysis segregates data flows and process flows, and arranges these in an \((x,y)\) matrix by degree of effort (x) and the degree of information structure (y). The model also differentiates directions of flow as bottom-up vs. top-down.

4.5.3 - Foraging

The foraging loop includes searching, filtering, reading, extracting and schematizing information. The tasks in this loop are presented as a trade-off between exploring, enriching, and exploiting information in the available information-space (Hutchins and others 2007; Pirolli and Card 2005; Hutchins and others 2003; Pirolli and Card 1995; Russell and others 1993).

Exploring increases coverage in the monitored space, or improves search recall in information retrieval terms. Enriching reduces the set of items previously collected for analysis by excluding some items. Exploiting the items in the reduced set occurs with more thorough reading, information extraction, inference generation, and pattern identification (Pirolli and Card 2005). Thus, greater information space coverage comes at the expense of eventually needing to process and exploit the material (Pirolli and Card 2005).

Pirolli (1995) and Russell (1993) both show that computational innovation can reduce the absolute and relative time costs of various information operations (Russell and others 1993). An ideal VA solution would reduce cost structure of the explore/enrich/exploit trade-off, by employing computational or crowd-sourced processing to provide broad-band low fidelity assessment of incoming data tightly coupled with narrow-band, high-fidelity processing by expert analysts.

The decision as to which data deserves further assessment includes cognitive tasks of reading, recognizing, evaluating content. Ideally, the cost of selecting filtered imported data to exploit should be externalized to computational pre-processing filters, which should identify important information to highlight visually with pre-attentive coding (Pirolli and Card 2005).
The costly decision to **shift focus** to a new area, or to **start analysis over** can be mandated by feedback or motivated by the data. The cost of **shifting attentional control** can occasionally be reduced by keeping search operations continuously open, and tracking progress within the search task, and in the users’ selection from results, thus eliminating the need to retread previously covered ground.

### 4.5.4 - Sense-making

The **sense-making loop** includes iterative development of a **mental model** from the schema that best fits available **evidence** (Klein and others 2006b; 2006a; Weick and others 2005; Pirolli and Card 2005). Pirolli identifies many intelligence activities as **sense-making** tasks, which include **information gathering** (Bae and others 2006), **information schematization** (Faisal and others 2009), **cultivating insight** by **manipulating the schema representation** (Stasko and others 2009; Hampson and Cowley 2005; Center for the Study of Intelligence (U.S.) 2009), and the creation of summary **knowledge products** or recommended **actions** (Thomas and Cook 2005) based on the insight.

Many **choke points** in the **sense-making loop** concern **problem structuring** (Heuer 1999), **evidentiary marshalling** (Pirolli and Card 2005), and **decision-making** (Hoffman and Militello 2009). It has been argued that these processes are affected by well known **cognitive biases**, wherein **human perception** is biased towards interpreting information according to **existing expectations** into **existing schemas** (Edwards and others 2007; Heuer 1999; Caverni and others 1990). Many other deviations and biases from **normative rationality** are noted by Tversky (Amos Tversky and Kahneman 1974). At such **choke points**, clarification of the **analysis process** can potentially contribute to the design and evaluation of analytical support tools.

**Problem structuring** is a **phase of analysis** (Wildavsky 1987) that encompasses the **generation**, **exploration**, and **management** of **hypotheses**. While undertaking **problem-structuring** and **decision analysis**, people often **fail to generate hypotheses** (Heuer 1999; Pirolli and Card 2005). **Contextual pressures** of limited time and unlimited data often inhibit rigorous **evaluation of competing hypotheses**. Finally, without a full sets of a) **alternative hypotheses** or b) **problem features**, the **diagnosticity** of **evidence** is incomplete. In these cases, analysts often forget that
they cannot determine the extent to which any piece of evidence makes a relative contribution in weighing the likelihood of scenarios (Heuer 1999). Ideal VA solutions would improve coverage in possibility space of the set of generated hypotheses (Pirolli and Card 1995), or even build generalized hypotheses based on a combination of exemplars, recency, and existing functional and propositional knowledge bases. They should also improve coverage of evidence-establishing problem features to establish diagnosticity of evidence only after generating a full set of alternatives.

Evidentiary marshalling includes organizing and selecting evidentiary facts to support or disconfirm hypotheses. However, reasoning over evidence has an exponential cost structure (Pirolli and Card 2005). Thus, rapid growth of relational patterns rapidly exceeds working memory. Of course, limits in human working memory (Styles 2005; Kramer and others 2007) have strong implications for the number of hypotheses, the range of evidence, and the number of evidentiary relations that can simultaneously be attended. The ideal solution from VA must structure evidentiary elements so as to focus analyst attention on the structure of the organized hypotheses and evidence, thus offloading maintenance of this conceptual structure to computational memory and freeing the analysts’ working memory for pattern identification.

Decision-making ultimately involves selecting a prediction or action from the set of alternatives (Pirolli and Card 2005), but more generally also includes judgments at all points in the analysis process, from search-term selection (Rubin and others 1999), and feature-based results filtering (Turetken and Sharda 2005), word-sense selection, propositional evaluation (Werth 1999a) and knowledge frame activation (Werth 1999c).

To help the analyst at a low level of word sense judgments during the sense-making process of analytical reading, an ideal solution would present a range of lexically and semantically framed word sense options overlaid on an underlying neural network, and using contextual cues from TextWorlds for node excitation and inhibition, highlight potentially relevant sense options while inhibiting out-of-bounds options. Moreover, potentially novel sense creations could be initiated between competing senses to create innovative sense blends. A similar approach
could be applied at a higher level, to propositions in determining both their entailments and inferences from activated frame knowledge.

4.5.5 - Critiques

Pirolli (2005) and Card (2006) describe the Sense-Making model over many pages in several sense-making model articles. These texts discuss at least twenty further cognitive procedures involved with analysis that, for unexplained reasons, remain unlabelled in their diagrams of the Sense-Making model of analysis. Several important cognitive phases of analysis are omitted in the sense-making model and its description (Wright and others 2006). For example, hypotheses are included as a transitional data state, rather than a complex, active cognitive process in its own right. Also, important procedural steps like search and re-evaluation are iterated over different content at various points in the analysis process: no less than five of sixteen top-level cognitive processes in the sense-making model are labeled as Search for X. Thus, search seems to be, to use the model’s terminology, a cross-cutting process. But it is not noted as such. Re-examination and re-evaluation, both procedures of analytical judgment, also seem to be unacknowledged as cross-cutting procedures. Ultimately, the cognitive processes involved in these wide-ranging recursive cognitive tasks are not satisfactorily unpacked.

Of course, there are many possible explanations for these omissions. Some of the discussed-but-unincluded cognitive tasks are tightly coupled with subsequent processes that may have been simplified for convenience. Also, the authors of the sense-making model have sorted their cognitive task analysis findings into a model that attempts to integrate four intersecting schemas as looping iterations within and between information foraging and sense-making; as typifying processes as either bottom-up vs. top-down; as balancing a trade-off of increasing cost vs. increasing effort; and as somehow also differentiating data flow vs. process flow (Pirolli and Card 2005). Some of the discussed-but-unincluded cognitive tasks may not have been conceptualized as having the necessary features required to fit any one of these implicit ‘vs.’ schematic filters. Finally, some cognitive tasks, like schema evocation and reasoning, are subconscious and therefore inaccessible to analysts except through a range of novel elicitation methods.
Thus, what is missing in Visual Analytics is a model that represents the not just operational but also cognitive processes involved in analysis in real detail. A productive collaboration will be required in order to empirically investigate the extent to which high level cognition is involved at each step in a revised cognitive-functional analysis process. Unfortunately, collaboration has not taken off (Ribarsky and others 2009) as intended (Thomas and Cook 2005) between cognitive scientists and Visual Analytics technologists, whose contributions are both required to produce the translational research necessary to improve and implement a cognitive-functional model of analysis processes for wide use in Visual Analytics. A renewed commitment to a more nuanced understanding of the analysis process will be required to translate this knowledge into the practical design and evaluation of visual analytic technologies (Ribarsky and others 2009).

4.6 - Analysis Synopsis

The discussion in this Chapter has introduced relevant domains from analytical practice and higher-level cognition and described how they contribute to a better understanding of the role of higher-level cognition in the generic, text-focused analysis process. This Analysis Literature chapter began by considering selected domains and concepts in Analysis, starting with a brief introduction to the domains of Cognitive Linguistics, Discourse Analysis, Computational Text Analysis, and Visual Analytics. I then introduced and briefly discussed the Sense-Making model of the analysis process (Pirolli and Card 2005). Its authors noted leverage points for the design and evaluation of Visual Analytics tools. These leverage points were then discussed and elaborated, as were limitations in the sense-making model of analysis.

Each proposed model discussed here, from discourse comprehension, reasoning, and decision-making, intelligence analysis and Visual Analytics postulates, defends, and thus relies rely upon the analysts’ mental interaction with constructs from long-term memory repositories that have variously been called schemas, mental spaces, mental models, and situation models.

I argue that creating functioning, interactive digital versions of these conceptual knowledge repositories should be a top priority of analytical tool builders,
including those in Visual Analytics, since these constructs transact a host of dynamic working memory procedures that must be difficult to actively maintain in memory, let alone simultaneously maintain in meta-memory, while engaging in challenging analytical tasks like analytical reading.

So, since we cannot expect analysts to read and understand a text while simultaneously cultivating a meta-cognitive self-awareness of that reading process that includes reporting or annotating each of the sub-conscious cognitive steps attached to the reading and comprehension process; a better understanding of the cognitive flow of analysis cannot be attained purely by continuous real-time reporting through introspection. Thus, building towards an interactive computational discourse model of reasoning would allow analysts to interact with this model-building process directly and reduce the cognitive load of reasoning by using rich, interactive discourse models. This line of enquiry is taken up in Chapter 9, TextWorlds+.

In the next Chapter, Flow, I introduce the Cognitive-Functional Flow Model of Analysis as an alternative model based on 1) the descriptions of (Pirolli and Card 2005), 2) the wider literatures on Intelligence Analysis and Policy Analysis and 3) a modified version of the concept of cognitive flow (Crandall and others 2006; James 1918).
Part II: Modelling Cognitive-Functional Analytical Process

To review, Part I, Foundations, has established the conceptual background as a common foundation that forms the basis for the research projects reported in subsequent Parts of this dissertation, which includes Parts II, Modelling Cognitive-Functional Analytical Process, Part III, Supporting Ubiquitous Shallow Analytical Judgment, and Part IV, Modelling Conceptual Structures in Deep Analytical Reading.

In Part II, Modelling Cognitive-Functional Analytical Process, the dissertation moves forward from introducing the background problems, concepts, and domains that support the overall Research Programme, toward introducing and improving upon models of the whole analysis process, and creating methods for gathering data on digital functional workflow dynamics.

By taking a human scale view of the analysis process problem space, this stance naturally affords a multi-level focus that moves from understanding the whole, to its widest parts, and then to its deepest parts, keeping the human analyst reader in focus throughout. Thus, starting in Part II, Modelling Cognitive-Functional Analytical Process, with a focus on analysis at the highest, holistic level, the research reported in Chapters 5, Flow, and Chapter 6, FlowSpaces, contributes to building cognitive functional procedural models and the collection of task-tool usage datasets on digital tool workflow.
Chapter 5 - Flow

*a.k.a The Cognitive-Functional Flow Model of Analysis*

This Chapter begins by reviewing progress through this dissertation to this point. Part I, Foundations, has presented the background and foundations of the two core challenges of Cognitive Limits in human analysts, and Functional Limits in analytical tools, which together define the analytical process problem space. Chapter 1, Introduction, introduces this problem space and Chapter 2, The Problem Situation, elaborates and situates four related dissertation research projects in relation to specific levels of the problem situation. Chapter 3, The Cognition Literature, and Chapter 4, The Analysis Literature, introduce the domains that have contributed to the discovery of the way forward toward a solutions to this problem space.

In Part II, Modelling Cognitive-Functional Analytical Process, the dissertation moves forward from introducing the background problems, concepts, and domains that support the overall Research Programme, toward introducing and improving upon models of the whole analysis process, and creating methods for gathering data on digital functional workflow dynamics.

Taking a human scale view of the analysis process problem space naturally affords a multi-level focus that moves from understanding the whole, to its widest parts, and then to its deepest parts, keeping the human analyst reader in focus throughout. Thus, starting with a focus on analysis at the highest, holistic level, Part II, Modelling Cognitive-Functional Analytical Process, begins by drawing upon 1) my practical experience as an analyst using analytical tools, including those offered by VA, and 2) the Literatures on pragmatic analytical procedures and higher level cognitive processes. Both will be found to agree that certain important cognitive and functional tasks are missing from the standard Sense-Making model of analysis that is widely used in VA. These Chapters will make the argument that the use of the Sense-Making model as a specification for analytical tool design provides one explanation for the clear gaps in analytical functions offered by VA tools today. The corollary to this critique is reasoned hope that published extensions to the notional Sense-Making Model of analysis could have a similar impact on the focus of tool-
builders, and result in the production of analytical tools that cover a greater range of real-world analytical tasks.

Thus, in Part II, Modelling Cognitive-Functional Analytical Process, the research reported in Chapters 5, Flow, and Chapter 6, FlowSpaces, will respectively contribute to building cognitive functional procedural models and the collection of task-tool usage datasets on digital tool workflow.

Specifically, in this Chapter 5, Flow, I will address identified omissions in the Sense-making model of analysis (Section 4.5.2, Sense-Making model) by creating the Cognitive-Functional Flow Model of General Analysis. The Flow Model additionally reconceptualises the analysis process such that each cognitive and functional task is conceived as overlapping networks of requisite and related cognitive and functional sub-tasks. Thus, when a task occurs, its requisite sub-tasks must occur, in some order, and its related sub-tasks may occur, or may be skipped entirely. Cognitive-functional Task Models specifying requisite sub-tasks and sub-task dynamics must be developed and plugged in for each cognitive or functional task in the Flow Model of general analysis. Chapters 8,9&10 undertake the specification of a Task Model for analytical reading.

To begin, Section 5.1, General Analysis, will introduce and argue for the notion of general analysis and typify the generic analytic process in terms of iteration through specified cognitive and functional steps, from shallow through deep execution of analytic tasks. I will describe the typical analysts process in terms knowledge of goals, data and methods required to produce practical insights. Despite the etymology of the term, analysis requires more than the ability to cut; it also requires the ability to identify, classify, and specify relations between parts. Though the productive process of analysis is not usually seen as creative, I argue for the utility of notion of analytical creativity (Csikszentmihalyi and Sawyer 2014; Simonton 1988), which is defined, here, beyond expertise, in terms of the analysts ability to envision new goals, to create new data and models, and to develop new tools (Dougherty and Keller 1982) and methods. In this Chapter I will also argue for the utility of analytical creations in facilitating the process of making analytical discoveries.

In the following Sections (5.2 and 5.3) on Cognitive Flow and Functional Flow, I will briefly introduce the cognitive-functional Flow Model of General Analysis, and
go on to discuss in more depth the rough segmentation in the Model, between shallow Preparatory and deeper Execution phases of analysis. Thus, Section 5.3.1 the Preparation Phase, discusses in three sub-sections that identify cross-cutting and emergent features of the preparatory phase of analysis; Section 5.3.1.1, Classify, notes how each of the core preparatory tasks involves cross-cutting classification supported by shallow analytical judgment, and Section 5.3.1.2, Localize, motivates the unmet need to provide systematic methods to capture those judgments in the collection phase, and Section 5.3.1.3, Focus, to subsequently enable, at the right moment, the rediscovery and reuse those judgments in both ongoing and future analytical tasks.

Section 5.3.2, Execution Phase, will introduce the Execution Phase of Analysis, arguing that the distinction between shallow and deeper analysis is roughly matched by a transition in the object of analysis from containers to content. Section 5.3.2 likewise divides discussions among three subsections, which discuss the targeted deeper analytical tasks; Evaluate, Annotate, and Read.

Section 5.3.2.1, Evaluate, will introduce the notion of deeper analytical judgments, as produced by deeper evaluation of content as opposed to the shallow feature-identification of content discussed in the Section 5.3.1.1, Classify. It will attempt to establish that content evaluation is not simply the province of the subjective reader, since it can also be approximated by natural language processing.

The following Section 5.3.2.2, Annotate, introduces the task of annotation as a means of capturing the products of these deeper, subjective content evaluations. This Section highlights challenges with standard user-driven approaches to manual externalization using conventional annotation methods. Specifically, this section considers problems for formal methods and for free-form concept-mapping methods, and positions the TextWorlds Analysis Method of extended CDG+ discussed in Chapter 8, as prepared to both accommodate and constrain the systematic externalization of internal representations of the conceptual structures of text as structured node-link diagrams with a specified referential scope.

The final Section 5.3.2.3, Read, will introduce the notion of analytical reading as the challenging and inescapable task required for the deeper analysis of text.
Figure 5.1. A Word Cloud view of Chapter 5 - Flow. A. This Chapter is highly connective, specifying extensive (e.g., and) and possessive connections and properties (e.g., of), then referential (e.g., the) and purposive relations (e.g., to), containment (e.g., in), and alternation relations (e.g., or) and definitions (e.g., is). B. This Chapter focuses on handling of analytical media in analysis.

5.1 - General Analysis

In the course of my academic research and professional experience I have explored the iterative process of preparing and executing analysis on a broad spectrum of media from the perspective of several complimentary fields of analytic practice. These disparate analytical fields do not typically interact or have obvious intersections but nevertheless share certain commonalities that may be used to effectively characterize analysis generally regardless of domain. Insofar as so many aspects of the core activities of analysis match across domains, we may speak of the generic analyst without sacrificing the particular details that make analysis in each domain unique. Thus, we may describe general analysis as typically iterating among various cognitive-functional steps along paired continuums of depth and completion, that range from the shallow analysis required for preparation through the deeper analysis involved in the execution of traditional analytic activities.

Traditionally, the analyst is a disciplinary specialist who knowledgeably selects from a pool of data-appropriate processing methodologies to produce practical, strategic insight. Thus, in this definition, analysis is both a goal-focused
adjudication and application of relevant methods to data and a context-relevant packaging of analytical products. In this context, analytical expertise is supported by both background cultural knowledge, personal experiential knowledge (Glenberg 1997) and procedural knowledge. Knowledge of **goals** includes both objectives and metrics for success (Gevamy and Wildavsky 1997; Russell and others 1993). Knowledge of **data** includes both its features and context. Knowledge of **methods** includes both preparation and execution of data processing tasks. Consequently, any analyst’s ability to productively engage in the **process of analysis** can be limited by insufficient knowledge of relevant goals, data, or methods.

The acquisition and activation of this knowledge is supported by core cognitive tasks, like schema activation, edge detection, feature identification and entity recognition, which are essential to pattern matching and thus to evaluation and classification (Zacks and Barbara Tversky 2001). So while the etymology of term ‘analysis’ is rooted in the manual embodied (Zwaan 2003) task of cutting (Zacks and Barbara Tversky 2001), effective analysis can only be accomplished with a thoughtful approach, that not only determines where to cut, but also creates labeled classes (Rosch and others 1976) from features and specifies relative connections, (e.g., relations) among parts.

While good analysis must be productive, analysis is not typically considered a creative act. Thus, good analysts are expected to employ the process of analysis to create analytical products (e.g., reports) rather than to produce analytical creations (e.g., aesthetic data models like visualizations, or data-driven artwork like data sculptures (Marchese and Banissi 2013; Nam and others 2007; Jack Zhao and Moere 2008; Kenderdine and others 2011; 2013; Bailey 2012; Briggs and Blythe 2013)). Nevertheless, as argued in this Chapter and exemplified throughout this dissertation (most notably in Chapter 6, **FlowSpaces**), creativity has an important and under-acknowledged role to play in selection of search terms and class labels, in the identification of relations between entities, in the building of schemas, hypotheses, and arguments, and in the discovery of insights. So analytical creations could also have something important to contribute to the initial preparatory and exploratory phase of analysis, beyond their acknowledged value as models created in the execution phase of analysis.
Analytical creativity (Julmi and Scherm 2015; Csikszentmihalyi and Sawyer 2014; Sawyer 1992; Martindale 1990; Simonton 1988) thus steps beyond analytical expertise. For the purposes of this discussion, analytical creativity is evident in the analyst’s capacity to envision new goals, to create new data and models, and to develop new methods for reaching those goals from the given data. Analytical creativity is, perhaps, best displayed as the analyst’s ability to layer, extend, and augment existing procedures and to develop new methodologies, including the design of new mechanisms (Dougherty and Keller 1982) for creating, shaping, analyzing and purposing new observational datasets.

In this introduction, I have argued for the notion of general analyst and typified the generic analysis process in terms of iteration through specified cognitive-functional steps, from shallow through deep execution of analytic tasks. I described the typical analysts process in terms of their knowledge of the goals, data and methods required to produce practical insights. I have established that despite the etymology of the term, analysis requires more than the ability to cut; it also requires the ability to identify, classify, and specify relations between parts. Finally, while the production of analysis is not usually seen as creative, I argue for the utility of the notion of analytical creativity (Csikszentmihalyi and Sawyer 2014; Simonton 1988) defined beyond expertise in terms of the analysts ability to envision new goals, to create new data and models, and to develop new methods, and also for the utility of analytical creations in facilitating the process of making analytical discoveries.

In the following sections I will propose in more detail the Cognitive-Functional Flow Model of Analysis as an alternative extension to the standard Sense-Making model. Section 5.2, Cognitive Flow, bases cognitive-functional Flow model on 1) Pirolli, Card, & Crandall's, descriptions of (Pirolli and Card 2005), 2) the wider literature on Intelligence Analysis, and the analysis process 3) a modified concept of cognitive flow (Crandall and others 2006; James 1918). Section 5.3, Functional Flow, elaborates upon the common tasks with distinct products in preparation and execution phases of analysis.

As we progress through this Chapter, the reader will come to understand that analysis requires the ability to start forming goals and evaluative criteria even from a deficit of knowledge, and to incrementally build up a schema, or knowledge
representation model, even while searching and evaluating results (Kang and Stasko 2012). These goal-filtered data models ground and guide the analysis process and evolve throughout. Any revision of goals, data models, and methods, may be based on either new information, changing priorities, or top-down directives (Patton and others 2015).

5.2 - Cognitive Flow

In the foregoing introduction to this Chapter, I argued for the notion of general analysis and typified the generic analytic process in terms of iteration through specified cognitive-functional steps, from shallow through deep execution of analytic tasks. I described the typical analysts' process in terms knowledge of the goals, data and methods required to produce practical insights. Despite the etymology (e.g., word origins) of its Latin components, analysis requires more than the ability to cut; it also requires the ability to identify, classify, and specify relations between parts. Though the production of analysis is not usually seen as creative, I argue for the utility of notion of analytical creativity (Csikszentmihalyi and Sawyer 2014; Simonton 1988) defined beyond expertise in terms of the analysts ability to envision new goals, to create new data and models, and to develop new methods and tools (Dougherty and Keller 1982), and also for the utility of analytical creations in facilitating the process of making analytical discoveries.

In this section I will introduce and briefly detail the cognitive-functional Flow Model of general analysis. The Flow Model is based on the premise that humans do not accomplish cognitive or functional tasks is a strictly serially segmented manner. Rather, cognitive tasks overlap and run in parallel. This simultaneous overlap and change is described a functional-cognitive flow, a related yet distinct conceptualization from the experiential cognitive flow described by (Csikszentmihalyi 1991). Activities in the cognitive-functional Flow Model include the major tasks of (Evoking) memory, Search, (Discovery), Reading, (Building), (Presenting) and Re-evaluating, which each contains a range of prototypically related sub-tasks or supporting tasks. Though the model layout may appear linear or hierarchical, the dynamics of the model are not determined by this layout. Component tasks may or may not be activated as required.
Finally, this section provides a visual summary of the activities in the cognitive-functional Flow model, and indicates when an action occurs more than once in the analysis process and on what objects it is executed. Activities that occur the most frequently across the preparatory and execution phases of analysis are therefore considered cross-cutting tasks. These cross-cutting tasks have a greater number of unique child objects upon which the analyst acts. The extended Flow model can be used (Pirolli and Card 2005) to focus consideration on how current research in higher level cognition can be applied to identify more overall leverage points for analytical tool-builders.

5.2.1 - Cognitive-Functional Flow Model of General Analysis

This section describes individual Tasks in the Cognitive-Functional Flow Model of Analysis, (Figures 5.2.1 and 5.2.2), catalogues the relations between its primary Tasks (e.g., Predicates) and the Objects upon which they act (e.g., key arguments) (Figure 5.2.3), and describes cross-function dependency relationships between steps and related processes and sub-tasks (in Figure 5.2.4) (Kang and Stasko 2014).

Accordingly, this model is based on the premise that, in general, humans do not execute cognitive tasks in a strictly segmented serial manner (Klein and others 2006b; Kang and Stasko 2014). Certain functions, like search, are in fact a composite of several cognitive tasks working sometimes serially, and sometimes in overlapping segments, and occasionally in parallel. The term cognitive flow, as it is used here, describes this simultaneous overlap and change in cognitive processes.

![Figure 5.2.1. The top level Cognitive-Functional Flow Model of Analysis, an extension of the Sense-Making model. Original Sense-Making model elements are in bold. Primarily cognitive tasks in parentheses are extracted from the articles describing the model (Pirolli and Card 2005) and from the literature on cognition and analysis.](image-url)
over time (Klein and others 2006b; Marchionini 2006; James 1918) This usage of the term flow is related yet distinct from the experiential cognitive flow described by (Csikszentmihalyi 1991). Awareness of this multi-lapping flow process is a challenging exercise in metacognition that is difficult to achieve (Bailyn 1977).

The Cognitive-Functional Flow Model of Analysis is a critical reorganization of the sense-making model of analysis (Pirolli and Card 2005). Original ‘notional’ or functional steps from the sense-making model are described in bold. Parenthesized tasks in this model are primarily cognitive. These tasks were extracted from articles describing the sense-making model, and are supported by the wider literature on intelligence analysis process (Heuer 1999; Bodnar 2003; Hampson and Cowley 2005; Wright and others 2006; Hoffman and Millitello 2009). Certain parenthesized functional tasks were noted in the articles introducing the Sense-Making model, but were not included in the model.

While the configuration of this model may seem to imply a static or linear process, the dynamics of flow depicted in Figure 5.2.2 is not a strict hierarchical order of sub-task execution, but rather a prototypical sequence usually followed by flowing and overlapping related tasks usually observed in relation to the successful execution of a major task.

Thus, for major tasks, flow among related tasks not necessarily hierarchical, since related tasks (and even some sub-tasks) can be bypassed. For example, when reading, the natural shallow flow may be to skim in a search-like mode without ever actively engaging any of reading’s typical subsequent functional tasks (e.g., the bold tasks in Figure 5.2.2, Extract and Relate), while similarly keeping cognitive tasks and products (parenthesized in Figure 5.2.2) internal.

Alternatively, the analyst may choose to externalize and capture aspects of their cognitive process of reading and discovery as interim analytical products (e.g., notes, extracts, models). For example, while engaged in deeper analytical reading (as opposed to shallow skimming); upon (discovering) something interesting, active (judgment) - a search sub-task - leads to (exclusion) or (collection), whereupon the later becomes grounds for extraction, (annotation), and the former for moving on. In this example, the major task of search was bypassed, so we assume that the analytical reader simply had the text ready at hand, and did not need to spend time looking for it. Yet, search’s sub-task network of related cognitive tasks (e.g., (select),
(judge), (exclude), (collect)) was activated and used to support extraction and (annotation) while reading. Thus, the sub-task dynamics of search seem to be naturally folded into in the process of deeper, goal-focused reading.

So, in Figure 5.2.2, the Cognitive-Functional Flow Model of General Analysis makes no claim for specified task sequences that could be construed as restrictions on the dynamics of flow within the model. Task sequences are prototypical, not absolute or universal. Task groupings are observed as prototypical relations between frequently co-occurring related tasks that may be requisite for top-level tasks. The flow model simply recognizes this skipping, layering & looping dynamic and identifies the functions which are necessarily undertaken as components of analysis.

The recognition and specification of the precise network of sub-tasks required for any major or minor cognitive or functional task is an area for future work. That work begins in Part IV, Modelling Conceptual Structures in Deep Analytical Reading, with the identification of a Task Model that defines the required sub-tasks network for analytical reading.

This Section concludes with a concise description of the major and minor task/functions in the flow model depicted above (Figure 5.2.2). For a thorough elaboration of each function and the objects upon which these actions are undertaken, See Appendix B. the Full Cognitive Flow Model of Analysis.

(Evoke)

Analysis begins by unconsciously (evoking) internal schemas that provide procedural task knowledge and propositional domain knowledge.
Figure 5.2.2. The major tasks in Cognitive-Functional Flow Model of Analysis, expanded to view their sub-group network of overlapping related tasks. Cognitive-Functional Task Models that specify a required sub-task network and its dynamics must be identified for each major and minor task in the Cognitive-Functional Flow Model of Analysis. Chapters 8 and 9 will thus consider the merits of the Task Model for analytical reading derived from the Cognitive Linguistic discourse processing model, Cognitive Discourse Grammar.
Evidence is organized. Based on the analyst’s current knowledge, even prior to initial search, (building) commences with problem situation schema building. After this initial goal-setting and problem structuring, (building) involves formalizing iterated reorganizations of evidence as schematic representations, cases, theories, hypotheses, and reports.

- Schematic representations are (instantiated) with facts and (interacted) with to reach conclusions.
- Cases or theories are (hypothesized) as evidence is (judged) and used to (disconfirm) hypotheses or support them.
- Arguments are (reasoned) as supporting evidence is (argued) and likely eventualities are (concluded).
- Reports are (written) as conclusions are required and supported.

Search

Procedural, propositional, problem situation and goal schemas identify which data-sources and search terms are searched first. Search results are (selected), (judged) and (excluded) or (collected). This sub-group of search related minor tasks is also applied throughout the analysis process, as cross-cutting tasks that are applied to every object selected for examination or consideration, so as to determine both its inherent features (e.g., pdf, long, article, topic=...), and is ascribed qualities (e.g., interesting, important, relevant).

Read

judgment of relevance iterates as texts are read, facts are extracted, implications are (inferred), extractions are (annotated), and entities are (related)

(Discover)

At any point in this process — and ideally at many points — insights are (discovered).

Present

Finally, the case is presented to a client audience, with whom directive feedback is (negotiated).
Re-evaluate

Any phase or product may be re-evaluated as directed or required.

5.2.2 - Analytical Targets

This section provides a visual summary of the major and minor tasks in the cognitive-functional Flow model, with particular attention for the Objects upon which these tasks are executed. Figure 5.1.3 indicates, by supervision, when an action occurs more than once in the analysis process, and in the context of application to the subvening objects (e.g., children arguments). Activities that occur the most frequently across the execution of analysis in various contexts therefore have a greater number of child objects upon which the analyst acts. This permits relative weighting of tasks in the Flow model and the identification of cross-cutting tasks which occur in both preparation and execution of analysis.

In this diagram, (Figure 5.3.1) descriptions of each bold function in the Cognitive Flow Model are initially drawn from (Pirolli and Card 2005). Important cognitive steps that are overlooked in the original Sense-Making model have their descriptions drawn from a predicate-argument analysis of Pirolli & Cards articles describing the Notional Model of Analyst Sense-Making (e.g., The Sense-Making Model) and are supported elsewhere in the Intelligence Analysis, Policy Analysis literature (Heuer 1999; Hampson and Cowley 2005; Jonker and others 2005; Kapler and others 2005; Cowley and others 2006; Crossno and others 2007; Eccles and others 2007; Hutchins and others 2007; Bier and others 2008; Center for the Study of Intelligence (U.S.) 2009; Chin and Kuchar 2009; Decker and others 2009; Kang and others 2009; Stasko and others 2009; Fischoff and others 2011) and from the targeted selections from the cognitive science literature detailed in Chapter 2.

As an extension of the Sense-Making model, the cognitive-functional Flow model can be used to focus consideration on how current research in higher level cognition can be applied to identify leverage points for analytical tool-builders to support challenging cognitive tasks with computational remedies (Pirolli and Card 2005). Since the Flow Model explicitly includes more cognitive and functional tasks, the extended Flow model would seem to offer more opportunities for
Figure 5.2.2.1. Objects upon which Cognitive Work is executed in the Cognitive Flow Model of Analysis, an extension of the (Pirolli and Card 2005) Sense-Making model. Process elements are action predicates and thus take the arguments listed. This diagram shows the arguments typically taken by process predicates.
identifying points of leverage in analytical tool design for tool-builders. For example, Figure 5.2.2.1 can be used to identify and label frequently reused tasks as cross-cutting tasks. Such cross-cutting activities occur the most frequently across several analytical steps. When applied in various contexts, cross-cutting tasks can therefore be expected have a greater number of child objects or targets upon which the action is undertaken.

In following Chapters, this dissertation will focus on capturing the cross-cutting task of applying shallow analytical judgments to each item (selected). Readers who would enjoy a thorough explanation of each step visible in conceptual diagrams of Figure 5.1.2 are encouraged to refer to Appendix B. The Full Cognitive Function Flow Model of General Analysis.

This section 5.2.1 Cognitive Functional Flow has provided a visual summary of the activities in the cognitive-functional Flow model, and indicates, by supervision, when an action occurs more than once in the analysis process, and in the context of application to its own subvening objects (e.g., children arguments). Activities that occur the most frequently across the execution of analysis in various contexts therefore have a greater number of child objects upon which the analyst acts. As an extension of the Sense-Making model, the cognitive-functional Flow model can be used to focus consideration on how current research in higher level cognition can be applied to identify more overall leverage points for analytical tool-builders (Pirolli and Card 2005).

This section has provided a visual summary of the activities in the cognitive-functional Flow model. The next Section 5.3, Functional Flow, details important functional steps of analysis by preparatory and execution phases. Section 5.3.1, Preparation Phase, identifies in three sub-sections specific cross-cutting and emergent features of the preparatory phase of analysis; Section 5.3.1.1, Classify, notes how each of the core preparatory tasks involves cross-cutting classification supported by shallow analytical judgment, and Section 5.3.1.2, Localize, motivates the unmet need to provide systematic methods to capture those judgments in the collection phase, and Section 5.3.1.3, Focus, motivates the unmet need to subsequently enable, at the right moment, the rediscovery and reuse those judgments in both ongoing and future analytical tasks.
Section 5.3.2, Execution Phase, will introduce the Execution Phase of Analysis arguing that the distinction between shallow and deeper analysis is roughly matched by a transition in the object of analysis from containers to content. Section 5.3.2 likewise divides discussions among three subsections, which discuss the targeted analytical tasks; Evaluate, Annotate, and Read.

Section 5.3.2.1, Evaluate, will introduce the notion of deeper analytical judgments, as through deeper evaluation of content as opposed to the shallow feature-identification of content discussed in the Section 5.3.1.1, Classify. It will attempt to establish that content evaluation is not simply the province of the subjective reader, since it can also be approximated by natural language processing.

The following Section 5.3.2.2, Annotate, introduces the task of annotation as a means of capturing the products of subjective content evaluation. This Section highlights challenges with standard user-driven approaches to manual externalization using conventional annotation methods. Specifically, this section considers problems for formal methods and for free-form concept-mapping methods, and positions the TextWorlds Analysis Method of extended CDG+, discussed in Chapter 8, as prepared to both accommodate and constrain conceptual structures in text as structured node-link diagrams with a specified referential scope. The final Section 5.3.2.3, Read, will introduce the notion of analytical reading as the challenging and inescapable task required for analysis of text.

5.2.3 - Synopsis of Cognitive-Functional Flow Model

This section has provided a summary description of the activity functions in the cognitive-functional Flow Model of general analysis. This model is based on the premise that humans do not accomplish cognitive or functional tasks is a strictly serially segmented manner. Rather, both cognitive and functional tasks overlap and run in parallel. This simultaneous overlap and change is described a cognitive-functional flow, which a related yet distinct concept from the experiential cognitive flow described by (Csikszentmihalyi 1991). Activities in the cognitive-functional Flow Model include the major tasks of (Evoking) knowledge, Search, (Discovery), Reading, (Building), (Presenting) and Re-evaluating, which each may contain a
range of prototypically related or supporting tasks. Though the model layout appears linear or hierarchical, the dynamics of the model are not determined by this layout. Each related task in a major task sub-group may or may not be activated as required.

The following section provides a visual summary of the major and minor tasks in the cognitive-functional Flow model, with particular attention for the Objects upon which these tasks are executed. Figure 5.2.3 indicates, by supervention, when an action occurs more than once in the analysis process, and in the context of application to the subvening objects (e.g., children arguments).

5.3 - Functional Flow

To review, the previous Section 5.2, Cognitive Flow, has introduced the core cognitive and functional tasks required to undertake the preparation and execution of analysis. The Flow Model of Analysis is thus based on the way that cognitive tasks overlap and run in parallel, such that activities flow into each other and, consequently, component activities are sometimes difficult to identify (except in exceptional case when they run in non-overlapping parallel).

This Section segments the Cognitive-Functional Flow Model of Analysis into preparatory and execution phases, and discusses the rationale for this distinction. This rough segmentation of analytic preparation and analytic execution is motivated by the sense-making, intelligence analysis, and public policy analysis literatures (Hutchins and others 2003; Pirolli and Card 2005; 1995). This distinction mirrors the layered analytical method described in Chapter 9, TextWorlds+, which segments textual information into the contextual ground that sets the stage for the unfolding action, insofar as analytic preparation sets the stage for analytic execution. Most importantly, this figure/ground distinction in generic analysis highlights some important gaps in the Visual Analytics literature and tools, which are explored in subsequent sections.

Certain key cognitive functions appear in both analytic preparation and execution phases. These are cross-cutting tasks. For example, shallow judgment of media containers in preparatory analysis is primarily about the creation, use, and refinement of the classification schemas required to classify documents. In
executive analysis, **deeper judgment** is seen in **content evaluation**, which integrates series of **classification schemas** applied to deeper levels of media contents. In preparatory analysis, **collection** is the external product of the search process which selectively localizes resources. In executive analysis, both **manual annotation** and **automated markup**, and **explicit model building** externalize this deeper content-evaluative process.

This Section has discussed the rationale for segmenting the cognitive-functional Flow Model into preparatory and execution phases. The Following Section 5.3.1, **Preparation Phase**, has three sub-sections that identify cross-cutting and emergent features of the preparatory phase of analysis. Section 5.3.1.1, **Classify**, notes how each of the core preparatory tasks involves cross-cutting classification supported by shallow analytical judgment. Section 5.3.1.2, **Localize**, motivates the unmet need to provide systematic methods to capture those judgments in the collection phase. Section 5.3.1.3, **Focus**, motivates the unmet need to subsequently enable, at the right moment, the rediscovery and reuse those judgments in both ongoing and future analytical tasks.

**5.3.1 - Preparation Phase**

The previous section has motivated the segmentation of the cognitive-functional Flow Model into preparatory and execution phases, and discusses the rationale for this distinction. This section will now turn to a brief description of the **Preparation phase** of analysis.

The **preparatory phase** includes **information foraging** and extends into **sense-making** (Pirolli and Card 2005). In this phase, analysts **seek, identify, collect, organize, assess, and convert** (Ferilli 2011) **multi-modal media** (text, audio, video) into the various formats required by the **digital analytical tools** the analyst will eventually use to facilitate the **execution phase** of their **analysis**. Decisions made in the preparatory phase are considered in the Flow Model to be important, if shallow, **analytical judgment** tasks.

The purpose of this analytical phase is to prepare: to create an organized personal repository that automatically captures any available meta data for important analytical media, as well as any personal analytical judgments made of
collected analytical media, and reduce the cost of getting analytical media ready to analyze in analytical tools.

The following three sub-sections will discuss the emergent features of the preparatory phase in the cognitive-functional model of analysis, noting how each of the core preparatory tasks involves cross-cutting classification supported by shallow analytical judgment. These sections motivate the unmet need to provide systematic methods to capture those judgments, and to subsequently enable, at the right moment, the rediscovery and reuse those judgments in both ongoing and future analytical tasks.

5.3.1.1 - Classify

The previous section offered brief introduction to the Preparation Phase of analysis, with reference to nearly equivalent concepts (information foraging and sense making) used by the Sense-Making model of analysis (Pirolli and Card 2005), but noting overlap between them. This section introduces the notion of cross-cutting shallow analytical judgments, and considers their role and impact in preparatory analysis.

In the preparation phase of analysis, the Cognitive-Functional Flow Model of Analysis identifies the steps of search, collection, organization, assessment, and conversion as key. These tasks are primarily functional tasks. Each depends on a network of requisite sub-tasks. Sub-tasks that are requisites in more than one major task are considered cross-cutting tasks.

While the entire requisite sub-task networks for each of these major preparatory task will not be examined here, this section will elaborate on one such cross-cutting cognitive task. This is the underlying application of shallow analytical judgments to various objects that may be manipulable digital items, or other, more ephemeral products of cognitive processes (e.g., schemas, properties or assessments). When applied to digital items, these shallow judgments are micro-assessments of media containers and contents that analysts seem to be able to do almost without thinking (MacInnes and others 2010) whenever they seek, select, sort, and rename, or relocate in order to organize media.

All shallow analytical judgments in analytic preparation seem to involve an implicit classification decision, such that incoming information is in or out of a
given set according to identified features in some sort of classification schema. All such classification schemas must be created or learned; updated; and possibly integrated with other systems. It may be an ad-hoc (e.g., #hashtags), or formally established (e.g., the Dewey decimal system). The following paragraphs consider the role of cross-cutting shallow analytical judgments and classification across each of the tasks identified as prototypically belonging to the Preparation Phase of Analysis.

In search and collection, analytical judgments are used to identify and select many objects, including data sources, search terms, query results, diagnostic (filtering) criteria, items to collect, and item properties. The functional tasks of search and collection rely upon cognitive functions (e.g., analytical judgments) that draw extensively on the analysts’ expert knowledge schemas. Knowing where best to start looking, what conceptual network of search terms will produce good results, and what makes a result worth keeping are each the product of experience.

In the organization of collections, analytical judgments are used to identify and select local collection management strategies. This draws upon the user’s experience with storage and retrieval systems, and occasionally highlights a discomforting friction between personal preferences and institutional standards and best practices. So there is the way institutions manage collections, and there is the way individuals manage their collections. Personal decisions about how best to label, organize, file and store resources are often very different from the librarian or information science professionals approach.

Nevertheless, in both cases the ostensible goal is to facilitate subsequent retrieval of collected resources. So the organization of collections typically involves the assessment and classification of resources, wherein analytical judgments are initially made to determine features to index. Subsequent search of local collections may then select items or portions for retrieval from collected resources. Analysts must therefore know their data’s metadata in order to determine its usefulness in analysis. Collecting all available institutional metadata on analytical media is smart, since it leverages available resources, and reduces the number of classifications the user must execute themselves. And subsequently, in the deeper evaluation phase, knowing details like the forum of the publication permits the reader to evaluate the source.
In data conversion, analytical judgments are used to identify and select conversion pathways. This draws upon tool-using expertise to bring the state and format of raw data into correspondence with required input states and formats. Thus, conversion tasks requires expertise with both end-stage executive analytical tools and broad-spectrum conversion utilities.

To summarize, search negotiates a target feature set as criteria for collection in a gathered set. Organization negotiates and leverages a standardized feature set for the gathered set to establish relations between gathered sets, while assessment extends the standardized feature set to include a personal, evaluatively generated feature set for the gathered set, and for retrieval ultimately negotiates a filtered subset from standardized + evaluative feature sets on the gathered set. Conversion adjusts the gathered set features to match the feature-set requirements of favored analytical tools. It should be clear that each one of these shallow analytical judgments that is required prior to final executive analysis are critical to overall analytical success (Heuer, 2010) of the analysis process.

The previous Section 5.3.1, The Preparation Phase, has introduced the core cognitive-functional steps of preparation. It has noted that each preparatory task requires the application of shallow analytical judgments, which are evaluations of items by their properties according to a classification schema. Classification is thus central to preparatory analysis (Patton and others 2015). The following Section 5.3.1.2, Localize, will rationalize the requirement for collecting and managing analytical multi-modal media, noting the challenges, costs, and benefits of collecting existing metadata and assigning personally generated evaluated features as tags, which are leveraged as a mechanism for automating file management.

**5.3.1.2 - Localize**

The previous section has introduced the core cognitive-functional steps of preparation, noting that each requires the application of shallow analytical judgments, which are evaluations of items by their properties according to a classification schema. This section will now introduce the requirement for creating collections, managing multi-modal analytical media, and present the notion of enriching local collections of multi-modal analytical media using tags.
Analytic preparation typically begins as the cognitive-functional steps of the preparatory phase are executed on network or internet accessible data in a process that selectively makes salient media locally accessible.

Despite the redundancy created by having local duplicates of network accessible media, this localizing process is necessary given the impermanence of network accessible media and the requirements of many current analytical tools which operate only on local data. Thus, analysts must create and curate personalized collections of media to support analysis. Unfortunately managing files and collections is time-consuming. However, not managing them increases retrieval costs and reduces the likelihood of rediscovery of salient media during subsequent local search. Accordingly, analytical media management must somehow be made easier and cheaper for analysts, and should only be considered for salient analytical media that the analyst has determined to be potentially relevant.

Local analytical media collections can be enriched with institutionally identified and personally identified analytic media features by harvesting all available metadata for analytic media in both its original modality and any available digital versions, and by recording on-the-fly micro-assessments of media containers and contents. Grabbing existing metadata and tagging evaluative features produces value to the overall analysis process particularly for subsequent shallow analytic tasks like naming, filing, grouping, classifying, and managing media, since grabbed and tagged media features can be used to automate management tasks and thus reduce the costs of curating local analytical media collections. Of course, metadata and tags require management too. The challenge for analytical tools is thus to reduce management costs for both data and metadata acquisition.

This section has rationalized the requirement for collecting and managing analytical multi-modal media, noting the challenges, costs, and benefits of collecting existing metadata and assigning evaluated features to tags, as a mechanism for automating file management. The following section introduces the notion of rediscovery requirements, wherein collection maturity may allow analysts to reduce search scope to local analytical media collections.
5.3.1.3 - Focus

The previous section has rationalized the requirement for collecting and managing analytical multi-modal media, noting the challenges, costs, and benefits of collecting existing metadata and assigning evaluated features to tags, as a mechanism for automating file management. This section now introduces the notion of rediscovery requirements, wherein collection maturity may allow analysts to reduce search scope to local analytical media collections.

Ultimately, however, for local analytical media collections to produce any value whatsoever, they must be included in subsequent searches where their items must be selected for deeper analysis. So whereas first search executes only on networked internet data, with the purpose of collecting relevant resource; with increasing experience, good recall, and sufficient collection maturity, the analyst’s search focus should turn inward to locally accessible repositories. In such cases, the first three tasks (search, selection, & retrieval) of the preparatory phase may subsequently be iterated on local analytical media collections before beginning deeper analysis. Analytical readers can always skip over shallow analysis and jump straight into deeper analysis, but the preparatory analysis phase is a conventional procedural filter that is used by analysts to keep themselves from engaging too deeply with irrelevant or inferior sources.

This section has briefly considered the notion of rediscovery requirements, wherein collection maturity may allow analysts to reduce search scope to local analytical media collections. It notes that shallow preparatory analysis is not always required, yet this phase has utility as a filter data source filter.

5.3.2 - Execution Phase

To review, the previous Section 5.3.1, Preparation Phase, has discussed in three subsections the emergent features of the preparatory phase in the cognitive-functional model of analysis, noting how each of the core preparatory tasks involves cross-cutting classification supported by shallow analytical judgment. It has motivated the unmet need to provide systematic methods to capture those judgments, and to subsequently enable, at the right moment, the rediscovery and reuse those judgments in both ongoing and future analytical tasks.
This section will now turn to a brief description of the Execution phase of analysis, arguing that the distinction between shallow and deeper analysis is roughly matched by a transition in the object of analysis from containers to content.

In the **execution phase** analysts work with digital tools to **deeply analyze** various data streams, iterating through sense-making steps to **discover**, **support**, **challenge**, and **report** their **conclusions**. The distinction between **shallow** and **deeper analysis** is roughly matched by a shift in the **object of analysis**. The **shallow preparatory analysis** is typically focused on **media containers** (e.g., files) whereas the **execution of deeper analysis** delves deeper into various levels of **media contents**.

This Section 5.3.2, **Execution Phase**, has introduced the Execution Phase of Analysis arguing that the distinction between shallow and deeper analysis is roughly matched by a transition in the object of analysis from containers to content.

The following Section 5.3.2.1, **Evaluate**, will introduce the notion of deeper analytical judgments, as with the evaluation of content as opposed to the shallow feature-identification of content discussed in the Section 5.3.1.1, **Classify**. It will attempt to establish that content evaluation is not simply the province of the subjective reader, since it can also be approximated by natural language processing.

The following Section 5.3.2.2, **Annotate**, introduces the task of annotation as a means of capturing the products of subjective content evaluation. This Section highlights challenges with standard user-driven approaches to manual externalization using conventional annotation methods. Specifically, this section considers problems for formal methods and for free-form concept-mapping methods, and positions the TextWorlds Analysis Method of extended CDG+, discussed in Chapter 8, as prepared to both accommodate and constrain conceptual structures in text as structured node-link diagrams with a specified referential scope.

The final Section 5.3.2.3, **Read**, will introduce the notion of analytical reading as the challenging and inescapable task required for analysis of text.
5.3.2.1 - Evaluate

To review, the previous Section 5.3.2, Execution Phase, has introduced the Execution Phase of analysis and argued that the distinction between shallow and deeper analysis is roughly matched by a transition in the object of analysis from containers to content.

This section now turns to introducing the notion of deeper analytical judgments, as with the deeper evaluation of content as opposed to the shallow feature-identification of content classification discussed in the Section 5.3.1.1, Classify. It will establish that content evaluation is not simply the province of the subjective reader, since it can also be approximated by natural language processing.

In analytic execution, the cognitive functions of reading, extracting, inferring, annotating, relating, schematizing, hypothesizing, reasoning, and writing are key. And each of these functional steps requires the execution of deeper analytical judgments upon the contents of digital items and the externalization or capture of that process. And while content evaluation is typically considered a process that is entirely subjective to the reader, as in the assessment of media salience or quality; it can also be executed objectively, as with the feature identification of natural-language text preprocessing.

Thus, the objective feature identification of NLP text preprocessing is transformational preparatory analysis that converts raw text data into annotated language data. But each NLP preprocessing task is taken as a proxy for some real mental processing phenomena that is part of language comprehension. Thus, since NLP involves resource-intensive analysis of the text, and since it is taken to model aspects of language comprehension, NLP text preprocessing is here considered a part of deeper analysis.

This section has introduced the notion of deeper analytical judgments in the evaluation of content as opposed to the shallow feature-identification of content discussed in the Section 5.3.1.1, Classify. It has established that content evaluation is not simply the province of the subjective reader, since it can also be approximated by natural language processing.

The following section, Annotate, will introduce the notion of annotation as a means of capturing the products of subjective content evaluation, and will highlight
challenges of standard user-driven approaches to manual externalization using conventional annotation methods. Specifically, the next section considers problems for formal methods and for free-form concept-mapping methods, and positions the TextWorlds of extended CDG as prepared to both accommodate and constrain conceptual structures in text as structured node-link diagrams with a specified referential scope.

5.3.2.2 - Annotate

To review, the previous section, Evaluate, has introduced the notion of deeper analytical judgments in the evaluation of content as opposed to the shallow feature-identification of content discussed in the Section 5.2.1. It has established that content evaluation is not simply the province of the subjective reader, since it can also be approximated by natural language processing.

This section introduces the notion of annotation as a means of capturing the products of subjective content evaluation, and highlights challenges with standard user-driven approaches to manual externalization using conventional annotation methods. Specifically, this section considers problems for formal methods and for free-form concept-mapping methods, and positions the TextWorlds of extended CDG as prepared to both accommodate and constrain conceptual structures in text as structured node-link diagrams with a specified referential scope.

The traditionally subjective process of content evaluation is typically captured in conventional user-driven approaches to annotation like syntax diagramming, semantic or logical formalization, note-taking, or concept-mapping. But there are real problems with these conventional annotation methods. First, the ability to visualize the syntactic tree structure of a text does not clarify its semantic content or conceptual structure. Moreover, conversion of regular text into formal semantic statements does not generally produce a formalism that is somehow more accessible or easier than reading. In short, formal equations and hierarchical structures alone have not been found to contribute to the generation of insight for typical analyst readers (Johnson-Laird and Girotto 1998; Glenberg and others 1999).

Moreover, the softer note-taking and concept-mapping procedures are criticized as not scaling to big data (Proulx, Tandon et. als. 2006). But there are even more
problem with free-form conceptual annotation, or concept mapping. Concept mapping begins as a tabula rasa — where reader focus selects ideas from the text to insert and label, usually not systematically, in a hierarchically structured node-link diagram.

So nodes are typically elaborated using only other node-entities in subset/superset (or parent/child/sibling) relations. In these diagrams, the semantics of concept-map things (Nodes) and relations (links) is either entirely unspecified or else is systematically underspecified simply labels or arrows on links. Thus, concept maps typically have fairly poor class consistency, since things can be classed as nodes or links, or as both simultaneously in different parts of the same concept map. Classification of nodes or links with labels is similarly unconstrained. In natural language, people often describe features initially as a procedural noun and subsequently as a verb, etc. Unfortunately, simultaneous use of different lexical facet of a single given concept in single concept map creates confused visualizations. Finally, there is the problem of underspecified referential scope, whereby the intended level of generalization of each node or label is neither obvious nor specified. Thus, it is not clear whether a node represents a term, phrase, whole sentence, series of sentences, paragraph, series of paragraphs, or a whole text.

Accordingly, in standard free-form concept maps, things are usually nodes and their connections are lines but not always, and that is up to the reader/evaluator. All things exist in a hierarchical relationships that have only one thing at its center, even if that is not how things really are. And both the initial classification of things as nodes and links and any subsequent classification of nodes and links is uneven and unconstrained, and is left up to the reader, who has no means of specifying which aspect or portion of the source text are seen as justification for the asserted node and link relations.

Thus, for concept maps to achieve communicative success as stand-alone summaries of analysis, they must be built with semantically aware constraints to ensure class consistency among its elements; features of their elements must be evenly and richly specified using a second level of node and link feature-classification that also employs semantically aware constraints to ensure feature-class consistency; they must be extended to accommodate non-hierarchical
relations; and means must be provided to specify both the scope and the source of reference; and their semantically aware constraints must treat variations in linguistic form as semantically meaningful facets of a single concept and normalize these variations, yet retain their semantic values, rather than stemming and discarding semantic meaning.

Thus, it should be clear that excavation of conceptual structures of text will require the systematic application of a theoretically based approach to the semantic value of syntactic structures. Moreover, it will require interactive means of fusing objective textual structures with both baseline lexicographical semantic structures and personal knowledge structures. Cognitive Discourse Grammar not only offers a theory-based approach to constraining the representation of the semantic value of syntactic structures, but also provides an interactive means of layering computationally derived objective text structures onto computationally derived baseline lexicographical semantic structures and selectively updating and fusing these with personal knowledge structures.

This section has introduced the notion of annotation as a means of capturing the products of subjective content evaluation, and highlighted challenges with standard user-driven approaches to manual externalization using conventional annotation methods. Specifically, this section considered problems for formal methods and for free-form concept-mapping methods, and positioned the TextWorlds of extended CDG as prepared to both accommodate and constrain conceptual structures in text as structured node-link diagrams with a specified referential scope.

The following section, Read, will introduce the notion of analytical reading as the challenging and inescapable task required for analysis of text.

5.3.2.3 - Read

To review, the previous section, Annotate, has introduced the notion of annotation as a means of capturing the products of both shallow analytical judgments and subjective content evaluation, and has highlighted challenges with standard user-driven approaches to manual externalization using conventional annotation methods. Specifically, this section considered problems for formal methods and for free-form concept-mapping methods, and positioned the TextWorlds of extended CDG as prepared to both accommodate and constrain
conceptual structures in text as structured node-link diagrams with a specified referential scope.

This section will now turn to introducing analytical reading. Overall, Figure 5.3.2.3 shows typical functional tasks involved in the analytic preparation and execution phases of analytic reading. On the left, Figure 5.3.2.3 also shows basic flow of functional (and roughly cognitive) tasks involved in the deeper execution phases of analytic reading. A different flow order between task will apply for shallow, preparatory reading.

Functional stages of analytical reading include preparatory media classification, wherein surface details of media containers are considered in the course of collecting, selecting labels for, & organizing analytical media; active reading, wherein the reader interacts with a text to extract relevant portions, infer facts, annotate findings, and relate entities and increment them into reader’s knowledge-base; and content evaluation wherein text content is assessed and normalized descriptions or summaries are generated at various level of detail.

Cognitive discursive tasks of readers are schematized as the Analytical Reading Task Model, which is derived from the extended Cognitive Discourse Grammar (CDG+) Discourse Processing Model. Tasks allocated to both participants (e.g., writer/speaker and to the reader/hearer) by CDG+ and the Task Model are developed fully in Chapter 8, CDG+, and Section 9.3.4, Analytical Reading Task Model.
Figure 5.3.2.3.1. Analytic Reading: preparatory and executive analytic cognitive functions of analyst readers. (This diagram is read top to bottom and clockwise from the center node.)
5.4 - Recommendations for Analytical Tool-Builders

The focus of modern Visual Analytics tools and the VA literature is on moments of obvious interactive analysis within specific tools. Thus, in VA, analysis is framed narrowly as what you execute with the analytical tool on the data.

5.4.1 - Recommendation 1: Dedicated Preparatory Analytical Tools

However analytic preparation involves the application of shallow analytical judgments, including simple classifications and shallow evaluative assessments.
across many preparatory cognitive-functional steps before analytical tools from VA can even be used.

Current analytical tools do not universally support nor capture the shallow analytical judgments made during the preparatory phase of analysis that includes the tasks of search, select, collect, sort, tag, convert, and manage analytical media. The next Chapter 6, FlowSpaces, is intended to show empirically just how much of an analyst’s time is occupied with these preparatory analytical tasks. I argue that analytical tool builders, both within and beyond Visual Analytics, can no longer afford not to help analysts with the preparatory analytic tasks that occupy so much of their time, if they want to ensure the relevance of their tools analytical tool-users both today and in the future.

Therefore, as a field Visual Analytics must collectively re-conceptualize analysis further back into the preparatory phase of analysis. Analytical tool-builders should then extend or re-calibrate existing analytical tools so as to better support this extensive, yet under-supported preparatory phase.

Ultimately, this dissertation research recommends that analytical tool-builders should consider building dedicated preparatory analytical tools for the search, collection, tagging, and management of multi-modal analytical media. Analytical tool-builders can draw upon leverage points identified by an extended model of analysis, like the Cognitive-Functional Flow Model of Analysis to build new tools that support and capture the shallow analytical judgments made in order to search, select, collect, sort, tag, convert, and manage multi-modal (e.g., multi-media that has multi-media metadata) analytical media. Beyond recommendations, this dissertation research also provides some prototypes of one way to support the preparatory analysis phase with dedicated tools. Thus, to start analytical tool-builders on their way, I propose and prototype several tools that address these gaps in Chapter 7, m³ [em-cubed].

### 5.4.2 - Recommendation 2: Dedicated Analytical Reading Tools

Most current analytical tools are derived from statistics-based information retrieval tools. Such tools are actually quite helpful for cutting through a large collection of texts in order find texts of interest to focus on. However, almost all VA tools are alike in that when it comes to reading, the analyst is on their own.
Top shelf VA tools that handle text, like INSPIRE, StarLight, and JigSaw enable users to select individual documents from large collections of texts using a combination of search queries and direct selection of elements in a visualization. The user searches both visually and traditionally (e.g., using Boolean search queries) among externally identified text features. These tools typically use a statistical approach to text that relies on dimensionality reduction and historically have not make use of semantic or lexical resources for handling synonymy or other linguistic and semantic phenomenon, like linguistic meaning.

However, if meaning is not inherent in linguistic communication, but is in fact an interactive, unfolding process of discursive interaction between the reader, the text, and their unique context; then purely statistical or even latent semantic methods of handling textual language data would seem to be missing something important (Glenberg and Robertson 2000). While statistical approaches to text are considered a more mature approach, and though not all natural language processing tasks are considered solved problems, Thomas (Thomas and Cook 2005) suggests that greatly improved results may be gained by combining the strengths of statistical and semantic approaches to transforming the content of textual documents for interactive visual representation in VA tools. To this I would add that the previous previous Chapter 3, The Cognition Literature, Section 3.2, Semantics, shows how conventional formal semantic models have little to add to theories of linguistic meaning, so care must be taken in selecting a semantic approach that provides useful insight into natural language usage in real world contexts.

Thus, neither a document-formatting-based structural analysis nor a linguistic Natural Language Processing fuelled Discourse Analysis of text contents is part of standard text mining or Visual Analytics of text toolsets. Such tools do not exist to help analysts when they are engaging analytical reading. So tools must be built to support analytical reading, by, for example, visually modelling reader’s conceptualizations and thereby externalizing mental models used to reason over textual contents.

In order to create Visual Analytic tools that support analytical reading, VA tool-builders must understand what the human mind does and requires during online discourse processing, or reading comprehension, and attempt to scaffold this process as closely as possible. Accordingly, Chapter 8, CDG+, introduces a method
that visualizes the process of analytical reading by explicitly modelling the evolution of personal conceptualizations of text contents.
Chapter 6 - FlowSpaces

To once more review progress through this dissertation to this point; Part I, Foundations, has presented the background and foundations of the two core challenges of Cognitive Limits in human analysts, and Functional Limits in analytical tools, which together define the analytical process problem space. Chapter 1, Introduction, introduces this problem space and Chapter 2, The Problem Situation, elaborates and situates four related dissertation research projects in relation to specific levels of the problem situation. Chapters 3, The Cognition Literature, and 4, The Analysis Literature, introduces the domains that have contributed to the discovery of the way forward toward a solutions to this problem space.

Part II, Modelling Cognitive-Functional Analytical Process, begins by drawing upon 1) my practical experience as an analyst using analytical tools, including those offered by VA, and 2) the Literatures on pragmatic analytical procedures and higher level cognitive processes. Both are found to agree that certain important cognitive and functional tasks are missing from the standard Sense-Making model of analysis that is widely used in VA. The argument is made that the use of this model as a specification for analytical tool design provides one explanation for the clear gaps in analytical functions offered by VA tools today.

Thus, in Chapter 5, Flow, I have repaired those omissions in the Sense-Making model of analysis (Section 4.5.2, Sense-Making model) by creating the Cognitive-Functional Flow Model of General Analysis. The Flow model additionally reconceptualises the analysis process such that each cognitive and functional task is conceived as overlapping networks of requisite and related cognitive and functional sub-tasks. Thus, when a task occurs, its requisite sub-tasks must occur, in some order, and its related sub-tasks may occur, or may be skipped entirely.

In the visual conceptual model of the Cognitive-Functional Flow Model of Analysis (Figure 3.1.2) sub-task networks are only shown for top-level tasks, though they do exist for all tasks. Thus, for example, Search always involves (Selection) (judgment) and one of either (Exclusion) or (Collection). But in order for (Selection) to happen, text-based Search almost invariably requires some level of Reading. Moreover certain types of goal-directed Reading look a lot like Search. So the ‘flow’
dynamic accounts for this smearing of boundaries, or simultaneous overlapping among tasks, their requisite sub-tasks, and related or embedded tasks.

In this Chapter 6, FlowSpaces, I have looked for ways to take a crisper yet more malleable view on functional transitions between digital Tasks and Tools by focal Activity (Ashley Williams 2009; Norman 2006; Gay and Hembrooke 2004; Christensen and Bardram 2002; Bannon and others 1983) focus. My approach steps beyond the linguistic class of Application Names, to produce a more systematic yet still essentially brittle conceptual classification system that classifies Applications according to the Prototypical Task that they enable.

But then, much as a blacksmith reheats iron slowly to reduce its brittleness and quenches it to fix a desired state of hardness, my approach also secondarily classifies each application usage by its current context of Activity focus, and easily updates focal Activity class just by dragging and dropping Application Windows between virtual Desktop Spaces. This reduces the brittleness of the Task class assignment, recognizing that a dynamic flow between tools from very different categories is often required to accomplish a single focus Activity. Thus, by maintaining focus on one Activity, even switches between Tools and Tasks can be productive and are often essential, and not just interruptions as is widely thought in the literature (Garrett and Danziger 2007; Czerwinski and Horvitz 2002; Czerwinski and others 2000). FlowSpaces data lets me identify productive Tool Task alternation sequences.

The approach I have taken is based on my experience as an analyst and as a tool-user. This entire FlowSpaces Research Project on digital tool usage was designed and conducted prior to discovering the principles of Activity Based Computing (Voida 2008; Czerwinski and Kaptelinin 2007; Bardram 2006; Bardram and others 2006; Bardram 2005) and prior to discovering a cognitive ethnomethodological study published in 1982 on how blacksmiths conceptualize and organize their tools (Dougherty and Keller 1982): by activity. Blacksmiths’ conceptualization of tool classes is personal, locational and practical. Thus, both in method (Keller was a trained blacksmith) and in conclusions, Dougherty & Keller’s taskonomy is the analog precursor to FlowSpaces.

Thus, to apply Dougherty’s observations to the analysis process problem space, when asking an analyst how to group their tools, “what leads to highly effective
means of \textit{analysis} is flexibility in \textit{the approach taken to tool} classification.” Dougherty concludes that analysts won’t use linguistic Name classes or even Function classes (e.g., Tasks) as their organizing principle, because both are too general to be practical. Yet, at the same time the functional classes are also artificially narrow, since they fail to incorporate the data, processes, and products associated with the primary functions of the tasks of analysis.

So Dougherty suggests that Analysts will either 1) \textit{avoid} or 2) \textit{further specify} Name & Function classes, extending them by encoding dimensions of context that are crucial for the current goal-focused Activity, just as I have done. Dougherty concludes that “constellations of conceptual units arising in response to a task at hand” are a better organizing principle for characterizing the analyst’s knowledge structures. Here, those constellations are ad-hoc sequences of Activity-focused app-task alternations.

Before the contribution of FlowSpaces, digital analytical tool usage is complex multi-Activity domain with very poorly identified functional steps and transitions. Here, neither prescriptive account of the ideal functional tasks or task-sequences, nor an isolated empirical sampling of functional transitions within a single tool user population would tell the whole story. The method I intuitively created, incidentally and accidentally corroborates Dougherty’s method and conclusions.

Neither the specific literatures on multi-tasking, workflow, nor that of User Interaction more generally has published longitudinal studies or data on multi-year, multi-focal, multi-tool, multi-tasking. Since no prior systems existed for logging hierarchical event data for core-periphery digital tool usage in functional task-space, I argue that the design, construction, and customization of a reusable system-of-systems for this purpose is a practical and usable research success that should also contribute to improved knowledge of digital tool usage sessions for both analytical tool-builders and analytical tool-users. I trust that in time this knowledge will similarly contribute not only to better coverage of unsupported tasks in analytical tools, but also to meta-cognitively improved tool usage sessions by self-reflective tool users.

Thus, the initial phase at this general level of my research focused on building a system-of-systems ‘meta-tool’ to capture, convert, and produce a longitudinal dataset. Moreover, by examining this dataset on “functional usage patterns of
digital tool usage sessions by prototypical task and current activity focus,” we can exemplify the **exploratory data analysis method** supported by visual analytic tools, and we can identify new empirical patterns for future statistical analyses.

For example, exploratory analysis revealed at least three distinct transitional patterns, or scenarios, in the transition data. We can now see where prototypically non-analytical tools take a cyclical role in analysis; we can see contiguous app-usage patterns, that may indicate mature within-app toolsets that succeed in supporting analytical focus and optimal flow experience (Csikszentmihalyi 1991); and finally we can identify which tools and usage sessions presage a functional shift to a new, distinct cycle in the digital tool usage workflow. These insights were not hypothesized a-priori, but were discovered using a data-grounded methodology of visual analysis (Ninci and others 2015; Ximenes and others 2009). Validity of the patterns derives from the coverage of the data, which is nearly population data coverage (all events for 1439 of 1460 Days) for events. These findings are consonant with the literature on workflow design (Kohlhase and others 2013; Chinthaka and others 2011) and the while visual analysis method is widely practiced in psychological studies using a single-case design, and is therefore appropriate.

---

**Figure 6.1.** A Word Cloud view of Chapter 6 - **FlowSpaces.** The most frequent terms in this chapter A. without and B. with stop-word removal. The most frequent terms excluded by stop-list filtering are relations that indicate deictic attachment (e.g., the, this, that), ownership (e.g., of), inclusion (e.g., and),
alternation (e.g., or), state (e.g., with), location (e.g., in, at, to, on), purpose (e.g., for), conditionality (e.g., if), consequence (e.g., then) and means (e.g., by).

6.1 - Overview

This Chapter begins by relating the general purpose of this research, identifying the functional transitions between digital tools by Activity, and relating it to the post-hoc discovery of principles of Activity Based Computing and an ethnomethodology that produced the concept of the taskonomy.

After providing this section-by-section overview, this Chapter continues by providing a summary of related previous work. Since this review of several relevant literatures identified no published long-term studies on multi-focal multi-application multi-task digital tool workflows, FlowSpaces aims to be the first published study in this area.

The next section introduces nine core concepts that are essential to understanding the research project described here.

This Chapter then continues with a discussion of The Problems, (Section 6.4) that have motivated the design and creation of the FlowSpaces data collection system and dataset.

Next, a novel system-of-systems and method for capturing real-time digital tool-usage and user-interface-usage data is introduced as a proposed Solution (Section 6.5) to The Problem (Section 6.4).

The Observational Case Study, (Section 6.6) was conducted over four years, capturing nearly population level data on nearly all Tool and most Space usage cycles. This study captures count and categorical data on various other user interface events (like mouse clicks, keyboard shortcuts, and keystrokes.

The next section discusses in detail the Longitudinal Workflow Data (Section 6.7) that were captured in the Observational Case Study. This discussion includes the pattern-based, statistical, and artificial intelligence-assisted methods of text analysis employed to convert semi-structured system-log-based textual data into multi-tool, multi-tasking, workflow event data. This Section further specifies the various Levels of data available within the FlowSpaces dataset.
In the visual analytical Exploratory Analysis Case Study, described in Section 6.8, I note how the design of analytical creations has in this case reduced knowledge deficits vis-a-vis goals, data and methods. Next, the scenarios predicted by various combinations of Activity Focus, Tool Focus, and Task Focus are elaborated.

Possible avenues for future analysis and appropriate analytical methods for the data became clear through exploratory analysis. Future work will use this dataset to train a predictive model for workflow and window states.

6.2 - Previous Work

This Research Project began by reviewing the literatures on Human-Computer Interaction and Usability (Schroeder 2008) and Visual Analytics to attempt to identify published long-term studies on multi-application multi-task digital tool workflows. While it is certainly possible that this literature review was not exhaustive, this dissertation has found no published long-term studies on multi-focal multi-application multi-task digital tool workflows.

Nevertheless, these literatures do offer a wide range of related work, regarding both the virtual desktop UI improvements and the empirical observation of Activity focus, task-direction, and application usage over time.

For example, the original concept of the workspace emerged in response to analyses of the activities performed by users of computer systems that showed complex patterns of interleaved activities (Jin and Dabbish 2009; Bannon and others 1983).

The original virtual desktop system extended the desktop metaphor as Rooms (Henderson and Card 1986)

IBM Research produced ActivityExplorer (Muller and others 2004), an activity-focused collaborative technology that is positioned between ad-hoc communication technologies of email and chat and the formal structure of shared collaborative workspaces. A long-term (100 day) study looked at sharing activities between 30 users of a small number of shared objects (e.g., messages, chats, files, folders, annotated screen shots, and to-do items), organized into activity threads that contain any type of object without hierarchy with membership/access to
objects dynamically assigned. Similarities with the work here include the longer-term data and Activity focus, though the collaborative and object-sharing focus is not reproduced in FlowSpaces.

Shorter term approaches to understanding the functional steps in an analysts’ digital and analog tool usage have been published in VA literature, most notably using the Glass Box, a software environment for the collection of workstation data while analysts perform information searches and report production, in order to better model and understand the intelligence analysis (IA) process (Laura Curtis 2008; Cowley and others 2006; Haack 2006; Greitzer and others 2006; Dizard 2005; Cowley and others 2005; Greitzer 2005; Quadrel 2004; Gorton and Haack 2004; Elm and others 2004). Unfortunately, in 2006, funding cuts shuttered GlassBox research and development. Similarities with the work here include a longer-term focus (a cohort of 8-week long studies), and an effort to view the task sequences. However, FlowSpaces adds a layer, tracking explicit Activity focus and refocusing/repurposing of tasks to contribute to a different Activity focus. It also captures data on the user’s engagement in any activity using any tool, not just search and report writing.

Related work with a wider focus on capturing high fidelity multi-media records in the areas of lifelogging include MyLifeBits, a personal archive database for everything in a person’s life (Gemmell and others 2006; 2002), LifeLog (Takata and others 2008), and the original Memex memory extender of (Bush 1945). These projects share the long-term commitment to capturing personalized, experiential digital traces.

Thus, the goal of this research project is conduct and publish the first such task logging longitudinal study. This study and dataset contribute toward an improved understanding cognitive-functional Flow dynamics in digital tool usage. This research project has thus focused on capturing, analyzing, and visualizing multi-year multi-focal multi-tool multi-tasking tool usage dynamics in the context of Activity Space usage.

Given the gap in the literature, it was not surprising to find a parallel gap in digital tools designed to capture this complex layering of multi-tasking dynamics. Thus, in order to meet research goals, this research project has required the development of a unique data collection system for observing and recording virtual
Desktop Name (e.g., Space Name), Application Name, Application Window Title, and Prototypical Application Task.

6.3 - The Concepts

In order to ground the research discussed in this Chapter, I begin this Section by introducing nine core concepts, Activities, Tasks, Tools, Tool Usage Cycles, Activity Spaces, Activity Space Usage Cycles, Activity Focus, Tool Focus, and Task Focus.

6.3.1 - Tasks

Tasks are specific functions that prototypically belong to a single Activity (e.g., a general functional area). However, some Tasks are cross-cutting, such that they occur in various Activities. In these cases, an extra series of prototype classification decisions are made to assign the cross-cutting Task to more than one Activity by assigning second through 16th-place Activities a proportion weight. For example, Reading is an Activity and read!emails is a Reading Task. But this task is folded into the aggregate email! Task that encompasses check!emails, receive!emails, read!emails, write!emails, and send!emails among others. Thus, the read!emails Reading Task primarily belongs to the Conversing Activity, since the email! Task accomplishes the goals of the Conversing Activity. So, considering only these two scenarios, read! Task is prototypically a Reading Activity, and secondarily a Converse Activity, assigned roughly at an 80/20 split. So the user estimates that reading email accounts for 20% of all reading. In fact, various levels of reading occur in every Activity, from Search, through Design, and the precise assignment of proportions is most illuminating in the first several major categories of Activity.

6.3.2 - Activities

Activities are general functional areas in which a range of Tasks exist to accomplish the goal of the Activity. For example, when a tool user wants to communicate, they must decide whether they prefer to converse by email, text message, instant message, video chat, phone, etc. This method selection determines which Tasks will be executed to accomplish this goal (e.g.,
receive/read/write/send email, text, or instant message; or receive/make/have video chat or phone call) and consequently contributes to the decision of which Tools will be selected to execute the Task and conduct the Activity.

6.3.3 - Tools

Tools are executable software applications that prototypically belong to a single functional Activity area, though most support a range of Tasks across several Activities. Thus, a prototype classification decision is made to assign each Tool to one of these general functional areas, according to the users’ assessment of what Activity they primarily use the tool to accomplish. For simple, single-Task tools, this prototypical classification usually poses no problems. However, for complex multi-Task tools (e.g., toolbox apps like MS Word, or Excel), this prototypical classification can be challenging. Future work is envisioned to handle this complexity empirically (See Section 6.10, Future Work).

6.3.4 - Tool Usage Cycles

Tool Usage Cycles measure the time and duration of the usage of each tool. When Tools are used, the duration of usage is calculated from the logged tool usage cycle Start Time and the logged tool usage cycle End Time. Thus, each Tool Usage Cycle lasts until either the App Window Title changes, or the App Window focus changes to a new Tool.

6.3.5 - Activity Spaces

Activity Spaces are task-specific virtual Desktop Work Spaces that are dedicated to a single Activity. Each Tool, once it has been manually assigned a prototypical Activity, is bound to launch in its prototypical Activity Space. Activity Spaces can be entered intentionally by direct navigation or incidentally by Tool usage. A Tool’s prototypical Activity Space is entered whenever that Tool is first launched and subsequently whenever the user reassigns focus to any App Window of that Tool (e.g., by selecting the application icon from the dock, transitioning between apps by keyboard shortcut, selecting between open windows on the desktop, etc.).
New Activity Spaces may also be entered either when the user selects another Tool that is not classified as belonging to the current Activity Space. Finally, Activity Spaces can be navigated through direct selection using relative keyboard shortcuts (⌥←, ⌥→,...etc.; option + arrow key) or by absolute numbered keyboard shortcuts (e.g., ⌥1 for Space 1, ⌥2 for Space 2,..., etc.), or by selecting a space from a visual overview of all 16 Activity Spaces reached by keyboard shortcut (⌥.; option + period key).

6.3.6 - Activity Space Usage Cycles

Activity Space Usage Cycles measure the time and duration of the usage of each Activity Space. Thus when Activities are undertaken, the duration of Activity-focus is calculated from the logged space usage cycle Start Time and the logged space usage cycle End Time. Each Activity Space Usage Cycle lasts until a New Activity Space is entered.

6.3.7 - Activity, Tool, & Task Focus

Activity Focus is measured by length of stay within or rate of transition between Activity Spaces.

Tool Focus is measured by the length of usage of a Tool or the rate of transitions between Tools.

Task Focus is measured by the length of stay within or rate of transition between Prototypical Tool Tasks.

6.4 - The Problem

The problem addressed in this large, multi-year, multi-layered study has cognitive, social, epistemological, technical, theoretical, and pragmatic dimensions.

For nearly forty years, cognitive science has theorized that human brain seems best equipped to handle 7+-2 ‘chunks’ of information at a time (Clark and Chase 1972). More recent studies have shown that much depends on the chunk size, and thus on the features of data being chunked. Thus, it is not surprising that it can be difficult to maintain focus through shifting contexts (Radvansky and others 2011;
2010; Radvansky and Copeland 2006), and effectively keep track of where you have been, what you have done, what your goals are, and what you must do next. It is not easy to maintain an awareness of specific trajectory and goal states while engaging in the task at hand, especially when that task is complex and fluidly crosses functional task boundaries.

Thus, while experts may have an implicit understanding of their process and seem to simply ‘know’ what to do next, expressing that process explicitly can be difficult (Crandall and others 2006). Software engineers and tool builders often build tools for experts, without themselves being experts in that domain of practice. It should not be surprising then, since digital tool users cannot easily explain their process, that digital tool builders do not understand the nature of digital tool usage workflows. In the context of this dissertation, this problem is particularly notable in preparation and execution of analysis on multi-modal analytical media, which the last Chapter 5, Flow, has established is not adequately theorized in the field of Visual Analytics (VA).

6.4.1 - Social & Epistemological Levels

At social and epistemological levels, the problem is that we don't all know what the experts know. Thus, it would seem that both tool builders and tool users alike require a low-cost empirical technical system and analytical methodology for gathering, analyzing, visualizing and modelling multi-tool multi-task digital workflow data. The availability of this prototype system and the generalized method could help tool builders and users to identify problem areas in digital multi-tool multi-tasking workflows, and even use those problems as leverage points for improved system design and tool usage.

6.4.2 - Technological levels

At the technological level, the default OS X Spaces provides baseline functionality for creating multiple desktop Spaces and binding applications to launch within a specified desktop Space. This virtual desktop system with application launch bindings is the foundation of the FlowSpaces system-of-systems. (For a thorough discussion of the usability problems identified in the default Spaces configuration, See Section 6.4.5, Usage Example)
Most importantly, to distinguish FlowSpaces from the default Spaces configuration, it is important to establish that Spaces does not produce a record of work, such as a log of space to space transitions, whereas in FlowSpaces the system exists only to record transitions in user Activity focus.

More noticeably, though arguably less importantly, Spaces does not go far enough toward assisting users in navigation and way-finding (e.g., shifting focus between virtual desktop Spaces). FlowSpaces easily accomplishes improvements to the base user interface of Spaces, by incorporating visual cues for way-finding, like desktop labels, customized desktop wallpapers and customized Docks. A unique, semantically charged customized desktop wallpaper is applied to each Space. Desktop Labels are superimposed on the bottom-left-most corner of each Space. The Space is labeled for its dedicated Activity (e.g., in the Space dedicated to search tools, the label Seek is applied; in the Space dedicated to analysis tools, the label Analyze is applied, etc.). Unique Docks are created for each space as ready-to-hand collections of all the tools used to accomplish the Activity of each virtual desktop work Space.

6.4.3 - Pragmatic & Grounded Theoretical Levels

At pragmatic & grounded theoretical levels, the problem is that the cost of digital work with computational tools increases with the complexity of digital workflow (George Davey Smith and others 2003). Any field, like VA, which claims to be about supporting analysis must consider how to reduce tool usage costs for the user, and should therefore come to theoretical terms with workflow complexity. For example, one particularly sparse proxy for workflow complexity is the number of concurrent applications open, compounded by the number of windows open in each application. This can be calculated at a gross aggregate level per User Activity Cycle from the collected FlowSpaces data, by counting the number of applications used per User Activity Cycle, and the number of windows used per application per User Activity Cycle.

The features of the data captured by FlowSpaces have allowed this study to take an even more detailed view. Accordingly, the collected FlowSpaces data allows the researcher to focus not only on the number and frequency of transitions between apps and windows, but also on the number and frequency of transitions between
Tasks and Activities as primary factors in digital workflow complexity. Section 6.8, The Exploratory Analysis Case Study, will discuss the latter two factors in detail.

This even more nuanced view of digital workflow complexity appreciates that the size of the pool of background elements alone does not determine the complexity of the digital workflow; though it certainly does make sequential scanning (e.g., search among open app windows) for the next focal target more difficult. Thus, the search difficulty for the next focal target is another important factor contributing to the complexity of digital workflow, that we will attempt to quantify in Section 6.10, Future Work.

6.4.4 - Usage Example

As a system-of-systems, FlowSpaces employs and extends a range of open-source and commercial applications. Component applications within FlowSpaces have been selected to 1) augment user interface for the default Spaces app; 2) to

Figure 6.4.4.1. The default OS X Spaces interface offers up to 16 contiguous, unlabeled, undifferentiated, numerically indexed desktop spaces.
collect data on the transition dynamics between tasks, applications, and app windows, and on mouse and keyboard user interactions with the system; and 3) to provide user-interface mechanisms for connecting apps with task-classes and task-spaces.

6.4.4.1 - Apple's Spaces

Spaces is a built-in feature of the OS X 10.5+ operating system that allows users to create and navigate multiple desktops to organize their application windows into groups so as to “reduce desktop clutter.” (Apple, Inc. OS X 10.6.1 System Preferences “Exposé & Spaces” Preference Pane Application 2013)

The premise is that separation improves manageability for typical work situations wherein multiple application windows are open simultaneously. That is to say, with many windows from many applications open and overlapping each other, users would have more difficulty identifying the next window required in the next step in their workflow. However, when open windows are segmented into a manageable number of groups and these groups are indexed appropriately for visual search, users should find what they need more easily.

Figure 6.4.4.1.^#. CMD+Tab Application Switcher
Consider the following case as motivation for using Spaces. Whenever Spaces is inactive and the user has many applications with multiple windows open simultaneously, the desktop is a mess. Finding an open application window using only visual search is not feasible when windows overlap each other.

For such cases, OS X offers the experienced user several built-in options for finding and selecting the next needed open application window, like mousing over the Dock or using the CMD+Tab **Application Switcher** (pictured above). But with so many applications and windows open even the Dock and Application Switcher are fairly cluttered. In their favour, serial search between well distinguished visual elements like application icons over a small space like the dock is easier than search over homogenous images like window thumbnails over a wide area, like the whole desktop. But since these options select applications, not specific windows, they also require a second keyboard shortcut CMD+~ (tilde) to launch the Window Switcher, possibly several times to cycle through the current focus application’s open windows to find the target window. So from a movement-based efficiency perspective, the Dock is preferable to the Application Switcher simply because it does not require a keyboard shortcut to make it visible.

Using Exposé is another built-in option. Launching Exposé shrinks open windows into a labeled matrix. While the Exposé matrix of miniaturized windows may usually provide sufficient visual search cues to facilitate selection of the next required application window, it loses effectiveness with each additional open window. In OSX prior to Yosemite, the Exposé matrix is indexed only by window names rather than by application and window name. Since users cannot look in one place for all the windows of a given application, the Exposé matrix is still just a neat, straightened mess.

This messy situation can be salvaged by filtering the matrix by application with TAB, Command+TAB, or direct dock icon selection. TAB will cycle through open applications and display only the open windows for the current application, while Command+Tab will launch the Application Picker, which requires selection by mouse or keyboard prior to filtering.

Whenever users cannot easily differentiate between open windows visually, they must search matrix labels and potentially have to recall the thumbnail locations in order to visually compare between potential target windows. Thus the
Exposé matrix will fail to make window selection easier when too many windows are open, when windows from a given application look similar, when the user cannot recognize a window from its thumbnail, when the user does not know the target window name, and when window names are too long to display in the matrix. When the next application window is ultimately selected from the Exposé matrix, the selected window is moved to the top of the pile of open windows.

![Figure 6.4.4.2. Column A - Launching Exposé shrinks open windows into a labeled matrix. Column B - Selecting from the Expose matrix zooms back in, with the selected window on the top of the ‘pile’ of open windows.]

As an alternative to the messy multiple-window in a single-desktop scenario, **Spaces** allows users to automatically sort open windows into groups by application type, so the user avoids both the initial messy desktop state and the straightened mess of the initial Exposé overview matrix and jumps directly to a filtered overview matrix, which can be further filtered by application.

Spaces could also be considered a software implementation of a multiple-monitor hardware configuration wherein each physical screen is dedicated to a single application or group of applications, albeit with a couple important distinctions. First, in Spaces only one desktop is visible at a time, whereas multi-
monitor configurations constantly display the extended workspace on all screens simultaneously.

Second, up to 16 contiguous, full-resolution desktop workspaces can be configured in Spaces, whereas very specialized hardware configurations are required to push more than 6 physical monitors at greater than 1080p resolution.

Thus, compared to physical multiple-monitor setups, OS X Spaces allows users not only more pixels to work with but also a more focused, dedicated space in which to work, using consumer-grade hardware at a fraction the cost.

Figure 6.4.4.3. Column A: Keyboard Shortcut zooms out to Overview Mode. Column B: While in Overview Mode, mouse selection of another Space zooms in to new Space.
Spaces is off by default. When turned on, it defaults to 4 spaces, though the number of spaces is configurable by adding and removing rows and columns.

Launching the Spaces application zooms from the current single desktop to a fullscreen overview of all Spaces. With no windows open it is a featureless matrix which is differentiated only by a darker highlight to indicate the current Space (See Figure 6.4.4.1)

When application windows are open on any desktop and the Spaces overview is launched, open windows shrink and zoom off to their relative location in the Spaces overview matrix. This transition typically occurs in under 1 second (See Figure 6.4.4.3).

While in this overview mode, mouse or keyboard selection can move focus to another space. Hovering the mouse over another space changes focus, while a mouse click activates the selected Space, causing the display to zoom from the overview back to the selected desktop workspace.

Figure 6.4.4.4. When application windows are visible in Spaces overview mode, users may drag and drop single or multiple (using Option-click) application windows between Spaces.
When many applications are open and overlap each other, the user may hit the Exposé key to separate all open windows in the Spaces overview.

Figure 6.4.4.5. A) Overlapping windows, B) sorted by Expose while in Spaces.
When changing Spaces directly without using the All-Spaces overview mode, by instead using keyboard shortcuts (or special mouse button bindings), a ‘HUD’-style matrix briefly appears, animating a transition of focus from the current to the destination Space.

![Figure 6.4.4.6. Heads up Display (HUD) Style matrix animates transition](image)

Applications can be manually assigned to open only in a specified space, in every space, or by default to open in the currently active space. Whenever a Spaces-bound application is launched, the visible desktop automatically switches to its assigned Space.

Thus, navigation through Spaces is achieved:

1) by launching space-bound applications (from the Dock, Finder or Spotlight), or
2) by selecting already-open space-bound applications (from the Dock, Application Switcher, or Expose), or
3) by launching the Spaces Overview (above) by Dock icon or keyboard shortcut), or
4) by direct selection of Target Destination Spaces using numbered keyboard shortcuts.

Spaces is built using only window identity and location to differentiate between Spaces. But past a certain point, it is not possible to recall the identity of all open application windows by their titles let alone exactly where they are open. So
navigation between Spaces is very difficult whenever users forget *where they were* when they opened an (unbound) application window and also with untitled windows and windowless UI elements like the menubar, dock, and expose.

Figure 6.4.4.7. Default interface for assigning Applications to a specific Space

The default Spaces configuration offers no differentiation between Spaces in Matrix Overview mode (zoomed-out), especially when no application windows are currently open. In overview mode with only a couple windows open in a couple Spaces, a user might reasonably be expected recall which Space their application windows are in, and which windows they would like to switch to.
In its default Single Desktop mode (zoomed-in), Spaces offers no persistent visual or textual cues to indicate which Space the user is currently in. The same dock and desktop wallpapers appears on all desktops. (This shortcoming has been addressed in OSX Yosemite and better.) The only indication of which Space the user is in is an option to enable a menubar icon that provides an index number for the currently active Space. (See the first icon in the image below.)

![Figure 6.4.4.8. The first menubar icon indicates the user is in Space 4.](image)

However, whenever the dock is reloading, this menubar icon is blank. In this state, Spaces offers no indication of which Space is currently active. This blank icon might even mislead the user, as it is identical to that used when Spaces is turned off.

![Figure 6.4.4.9. The first menubar icon indicates that Spaces is off.](image)

The HUD transition animation only displays ‘relative location’ information on a switch between Spaces for a very brief moment. That is, it reinforces the location of the Space the user has switched to relative to the location of the last Space. This ‘relative location’ information might be useful when using only a small number of spaces, but knowing that you are now ‘beside’ where you just were does not tell you what windows are open there, or what you were doing there when you were there last.

Furthermore, as mentioned above, the Dock - the Icon-laden strip along the bottom or side of the screen that allows quick access to favourite applications, documents, and folders - remains constant across all Spaces.

Also, you cannot ‘send’ an application (or a single window from an application with many windows open) to a user specified Space using only the keyboard (it is possible to do so using a combination of the mouse clicks and keyboard shortcuts, or iteratively dragging an application window over the Desktop Space’s edge).

Finally, the assignment (e.g. binding) of applications to Spaces is not simple, intuitive, or error-proof.
Fixing each of the above limitations would improve Spaces by reducing or eliminating interruptions to user flow right out of the box. But several implemented tools offer real improvements over the basic functionality offered by unmodified Spaces. These improvements are offered by the HyperSpaces and DockSpaces applications.

**6.4.4.2 - HyperSpaces**

As mentioned above, default OSX configuration of Spaces segments the desktop workspace indexically using only a numerical value arranged in a matrix (e.g. as Spaces 1 through 4; or 1 through 16). Using HyperSpaces, I have added both textual labels and visual cues for each space. Now, each Space is visually distinct and linguistically accessible with:

1) A unique, visually iconic Desktop wallpaper for each *task-space*.
2) A semantically useful name for each *task-space*.
3) A Label on each Desktop wallpaper.

Most importantly, HyperSpaces creates a Growl notification for each Space transition. This is logged to create the second level of data - the ‘task-space’ state.

**6.4.4.3 - DockSpaces**

Customized docks are an Activity-focused shortcut for finding appropriate tools.

The first version of FlowSpaces (OSX Snow Leopard), adds shortcuts to customized tool collections to each Space in the form of custom docks. I first manually collected the most frequently used and/or ‘favourite’ tools from each task type onto the leftmost side of custom Dock for each Space. Next I exhaustively added all the tools that have been classified as a given *tool type* (e.g., tools for Search are classified as ‘seekers;’ tools for Analysis are ‘analyzers,’ etc.) into a folder with a customized drawer icon, that appears as a labelled (e.g. “task + modifier”) dock Stack on the rightmost side of each custom Dock for each Space.

Thus, each Dock contains all the tools available for working in the given Space, with preferred tools arrayed for easy access directly on the Dock and the rest in a succinctly labeled interactive drawer.
On the left-most side (and middle) of each Dock, blank spaces are added to provide some separation between the Activity-Space-dedicated tools (on the left) and all the currently active running apps (toward the middle, highlighted with a glow beneath them). On the right, I have also added a ‘stack’ (e.g., a visual shortcut interface to a folder) for recently used applications; a ‘drawer’ (e.g., a folder shortcut) for ‘favourite’ places (e.g. Shortcuts to Folders I open frequently) and a stack for the contents of my Documents folder.

6.4.4.4 - Workspace Arrangement

I have arrayed the workspaces so as to be conducive to workflows that require a transition between Spaces. The Layout of Spaces in FlowSpaces conceives of the relations between workflows less of a hierarchy and more of an idealized order of task precedence. This order is not considered absolute and is geared towards describing a users’ first-time interactions with files. (i.e. A different Spaces order might better suit a user who no longer actively collects files.)

The order of Spaces is derived from a manual classification of nearly 900 applications according to the ‘type of task’ they *enable* or *do* for the user. The
order of the Spaces 1-16 corresponds to the order of Tasks in a meta-workflow that
moves the user from resource acquisition through processing to analysis and
presentation. Using 16 Spaces allows for greater resolution in segmenting this meta-
workflow into specific workflows for each distinct ‘task types.’ And having only 16
Spaces means that some aggregation (e.g. of tasks conceivable as sub-tasks) of tasks
is unavoidable.

The order is conceived of as additive, such that the tools at each successive task
types are generally expected to also provide or build upon the functionality of prior
task types. (e.g. If the Space 5 is for ‘managers’, these manager apps will necessarily
allow some search, collection, sorting, & organization.)

6.4.4.5 - Application-Space Assignment

While this aspect of Spaces is not easy to use, it is not the biggest challenge.
Since Application-Space bindings are stored in the Dock; I required some way of
sharing the bindings I have created across all docks. For example, when in Space 1,
if I assign 5 applications to Space 1 and then switch to Space 2, I will find that my
bindings do not persist, since Spaces 1 and 2 have different Docks. While I
laboured with this app-space-binding UI challenge for a very long time, it proved
nearly impossible to reproduce all the bindings with perfect fidelity using the app-
space assignment UI provided by OSX. There were simply too many manual
operations required and too many opportunities to make mistakes.

However, by editing the Dock PLIST files directly, it is much easier to simply
edit bindings within one Space and copy them into each subsequent custom Dock
PLIST file. I also used Keyboard Maestro and AppleScript to built into FlowSpaces
some custom coding and scripting to check for consistency of Application-Space
bindings across all Docks (and the currently running Dock).

6.4.5 - Integration Example

This section provides an extended example of the process I used to generate
and integrate data from existing tools, highlighting the challenges and problems
with using each component in FlowSpaces. I have used off-the-shelf and open-
source packages to capture data where possible, and written my own applications
and macros as required\textsuperscript{24}. Consequently, each off-the-shelf data capture
application approaches the event-capture process with somewhat different goals and methods. One captures the application and window names while recording user interface manipulations of mouse and keyboard for each window with sub-second resolution. But this higher precision comes at a cost of greater application instability. Another captures to a variable width (two to five-second) resolution, capturing only the duration of user focus on each application window by window name. Here, precision is traded for better valid-event coverage and fewer crashes.

6.4.5.1 - Slife

Slife logs the application and window names in focus, the event duration, and the event start time. Through a standard cocoa interface, Slife allows the user to manually add categories of activity, to set and track goals for each activity, to manually add applications to be logged, and to manually assign each application an activity and a colour.

![Slife Cocoa interface to add and set targets for Activities.](image)

*Figure 6.4.5.1.1. Slife Cocoa interface to add and set targets for Activities.*
Manual assignment of Applications to Activities by first creating and selecting an Activity, then checking a list.

Its preferences allow the user to set an idleness threshold to automatically subtract idle time from the duration of idle-contiguous events, and to set the observation rate anywhere between every five seconds to every minute. Data is stored in a SQLite database, which can be set to purge at user selected intervals.
The challenges with Slife have to do with its manual set up and maintenance, its data retention, data precision, performance, and its visualizations. Regarding set up, the user can simply start Slife and it will log all applications used while Slife is running. However, until the user sets up activity classes and assigns colours to activity classes, the visualizations and data are not particularly helpful. Adding activity categories in the UI is fairly straightforward. However, other tasks in the UI, including adding individual applications, classifying applications with an activity, and assigning applications a colour are difficult and tediously manual. As long as Slife is running, whenever new applications are installed and used on the system they will be logged by Slife. However, periodic maintenance is required to classify and colour these new apps. To improve the user experience of this tool, options should exist to batch-add applications, to assign activity immediately when the application is first logged or added, and to automatically assign colour based on the assigned activity class.

Slife runs most efficiently as a background process. It can be set to display a menu icon in the menubar to remind the user that it is running and to offer a quick way to classify the application and document windows currently in focus. However, if and when it crashes silently, the user may not be aware that the application is no longer running and that data is no longer being logged. The risk of data loss can be mitigated by automating Crash recovery by writing a launch daemon to check if the app is running and launching if not running. User awareness could be improved by creating a larger optional status window display the current focus data.

Slife’s data retention policies are related to its performance issues. Since the whole database is loaded into memory each time Slife is launched, purging data on a daily or weekly basis would be ideal for performance, so long as some backup or rollover options were available. However, since backup is not available, I opted to set Slife to never delete events. Over time, this has lead to performance issues. On average, six months’ worth of logged data creates a database file of just over 100MB. So system slowdown is markedly noticeable, more so in the latter half of the year.

Logged events are first stored in memory until committed to the database at each application switch or when closing Slife. Each application switch results in a
3 to 5+ second spike in CPU usage while idle times are calculated and removed from the logged event duration. While a 5s CPU spike is generally tolerable from a performance standpoint, no events are logged during these 5s interims between logged events. From a precision standpoint, not recording events during a 5 second interim means potentially losing track of many events and is not ideal.

When Slife crashes before committing in-memory events to the database, that event data is lost. When the database is being used by both Slife and another application, database corruption is possible, and file locking is highly probable. In the case of the latter, Slife will stop recording data and appear to ‘hang’ until the other application has closed its connection into the SQLite database.

Through its interface Slife allows the user to see daily and monthly views on the collected data by activity, goal, application, and document. However, interactivity in the visualizations is limited to toggling the temporal granularity of the visualization from daily to monthly views, and brushing by selecting a mark in the view for callout, or an item from a list to visualize its proportional contribution to the daily or monthly totals. Visualizations are another area where significant improvements are possible.

![Slife Visualization of Application Usage by Month](image)

*Figure 6.4.5.1.3. Slife Visualization of Application Usage by Month. Icon indicates App and Color indicates Activity.*
Figure 6.4.5.4. Slife Visualization of Application Usage by Day. Icon indicates App and Color indicates Activity.

6.4.5.2 - Backtrack

BackTrack records durations for in-focus application windows. It logs application and window names, the event duration, and the event start time, as well as typed text and mouse clicks. Through a standard application interface the user can inspect a table of events. BackTrack offers no visualization beyond this large table.

Its preferences can be set to log keystrokes, Command+V results, mouse clicks, and screenshots. Many options exist to limit logging to selected applications, to purge data at intervals, and to export data.

Challenges with BackTrack center upon its precision and reliability.

BackTrack samples data at the sub-second level and consequently collects over three times as many events as Slife.
Events lasting only milliseconds are logged yet durations are calculated to the nearest whole second. Since events are logged by application and window title, that ephemeral span where windows load just prior to retrieving their titles creates a nearly universal pattern of named windows of some real duration preceded by an untitled window of 0 duration.

Untitled events lasting less than one second with contiguous application values need to be merged and their field values updated.

BackTrack also logs ‘modal’ and ‘sheet’ windows, which have no titles but are produced by an application requiring the user to select an option, such as a ‘Save As’ dialogue box. These untitled windows often have a duration greater than 0 and should also be merged with previous events that share application value.

If the user interacts with another (background) window without changing the in-focus window, keyboard and mouse clicks in the background window will be logged to the in-focus window. Occasionally under heavy load, BackTrack will lose track of an in-focus window switch and continue logging key and mouse clicks to the previously in-focus window. In such cases switching back and forth between two or more windows of an application will often properly align the key and mouse click log with the currently in-focus application window.

BackTrack keystroke logging records the unicode character produced by any combined keystroke like a keyboard shortcut. Consequently, the text log includes characters where none were actually produced. These un-typed characters need to be removed from the log, or replaced with the actual keystroke of the keyboard shortcut which produced them.

BackTrack hangs when another application accesses its SQLite database while it is trying to commit a user session from memory to the database. Also, BackTrack does not log any events that occur while it is hung or is trying to write to the database. The duration of the write to database depends on how long BackTrack has been logging. If the application is closed before its data can be successfully committed to the database, the data will be lost.

Prolonged CPU spikes are expected during the calculation and commit process, which can sometimes last hours or even days, if the system has not committed data for several weeks.
6.4.5.3 - Growl

Growl is an event notification system for OS X. It is an open framework that allows applications developers to display formatted information from an application to the user. Notifications can be logged, displayed on screen, or forwarded via SMS, email or TCP to another device.

![Growl Notification displayed on screen](image1)

Growl checks for user activity in thirty-second intervals and internally sets its status to Inactive when no user activity is recorded in the previous thirty seconds, and back to Active when user interface activity is detected.

The developers of some applications have Growl notifications built in, like Hyperspaces, DockSpaces. Notifications for these applications was set to write into an XML log.

6.4.5.4 - Keyboard Maestro

Keyboard Maestro lets users create macros, or procedures that are executed when particular environmental variables or user input is detected. FlowSpaces uses Keyboard Maestro to record when certain keyboard shortcut are executed, and to provide a redundant log for transitions between windows and applications consequent to these shortcuts.

6.4.5.5 - Splunk>

Splunk> is a real-time temporal event analytics platform that is designed to facilitate the conversion of time-stamped computer-generated system logs into analyzable transaction data.
The information contained within system logs is often semi-structured. This means that it is separable into fields & values, though stored in string format. Splunk is an integrated environment that offers customized parsing and data extraction, as well as data warehousing management for datasets extracted in real time from time-stamped text-stored data. Thus, the splunk> system is a programmable, command-line data-extraction and visualization environment.

FlowSpaces uses splunk> to integrate data and extract events from the XML and TXT event logs generated using Growl & Keyboard Maestro. FlowSpaces also uses Splunk> to identify system events from system logs, and to extract event data from the SQLite databases created by Slite and BackTrack (application focus & keystroke) logging applications.

For each new data type, the splunk> analytical platform requires extensive customization. Splunk> users must also build a custom event extraction routine for each element of data that requires extraction, for each data type and source.

Once data is being reliably ingested, data must be improved with post-extraction computation, classification, & analysis using command-line queries.
These results are added to the raw & extracted data, and it is all shuffled, reorganized, converted, formatted, and exported using command-line queries.

Figure 6.4.5.5.2. Proportional Events logged across Activity Spaces

Figure 6.4.5.5.3. Total Events logged across Activity Spaces, (highlighting Analysis in Yellow).
The fully customized Splunk> system is available for download, and the sequence of commands used to generate the dataset is included in Appendix C - The Splunk> Integration, Processing, & Export Search, in Splunk> Search Language.

6.5 - The Solution

FlowSpaces is a system-of-systems approach to workflow logging and tool management and retrieval that extends the built-in functionality of Spaces.

With FlowSpaces I address the weaknesses in Spaces discussed in the previous Section 6.4, The Problem, to record task-state-transitions or workflows within and between tasks and tools. Additionally, FlowSpaces creates a highly usable connected series of dedicated, visually distinct, tool-ready, activity-specific workspaces for distinct, yet related, kinds of digital work activities, or tasks.
Figure 6.5.1. The FlowSpaces system-of-systems augments the default OS X Spaces configuration and functionality using HyperSpaces, DockSpaces, Keyboard Maestro & Growl to log digital Application Usage, by Prototypical Task and by current Activity Space Focus. Thus, FlowSpaces is an event-log generating, visually distinct, conceptually labelled, task-specific, tool-collection, and Activity-focusing virtual Desktop workspace system.

6.5.1 - User Interface

In FlowSpaces, each Task has a dedicated Activity-specific desktop workspace, or Activity Space. Each Tool is an application that is manually classified by Task (using Chapter 7, m3 [em-cubed]), and is manually bound to launch by default in its named Activity Space. Each Activity Space therefore is a unique location that houses a unique and dynamic collection of task-specific tools (Dougherty and Keller 1982), that is manually curated in a customized, task-specific dock for each task-space. (For further details and to see each dock, See Section 6.4.5.3 Custom Docks.) Furthermore, as part of the m³ [em-cubed] system-of-systems described in the next Chapter 7, each tool is manually tagged with its prototypical task and object (Dougherty and Keller 1982) to facilitate retrieval using local search.

6.5.2 - User Interaction
FlowSpaces logs application usage by Window Name and Prototype Task in the context of the current focal Activity. Since Applications have a range of uses beyond their prototype Task and can also be used in workflows beyond their prototype Task, FlowSpaces makes it easy to specify when a Tool is conscripted for use beyond its prototype Task.

By simply dragging application windows between Activity Spaces, FlowSpaces users may easily and temporarily reclassify the dragged Application Window as belonging to the new focal Activity. This reclassification lasts for as long as the dragged Application Window remains in use in the target Activity Space. Thus, FlowSpaces uses the **currently active desktop context** to transiently reclassify tools. This reclassification by *current app-usage context* does not change the prototypical tool classification initially assigned by the user, but provides a simple mechanism for indicating boundary-spanning or peripheral uses of a task-specific tool within a workflow that centers upon another activity-specific desktop workspace.

![Figure 6.5.2. Dragging app windows between task-spaces reclassifies those windows with the users’ current real-time task focus.](image)

Thus, FlowSpaces offers two levels of classification; the first is a manual classification of the typical task for each tool, and the second is a silent and automatic classification of the real time activity of each usage session. The
assumption is that the default bindings of Tools to Activity Spaces is the default focal area for all standard usages sessions of any given app. Thus, only peripheral usage cases require reclassification.

6.6 - The Observational Case Study

To review, the purpose of this phase of the Research Project was to create a dataset that would provide tool users and tool builders with insight on the aspects of the analysis process which involve the use of digital (analytical) tools, and to be able to differentiate between multi-tasking workflows based on Activity.

This dataset does not clarify tool functionality to the level of functional calls in the execution of code. Such functions exist below the level of human perception. This study concerns only the functions of tools at the level at which the human tool-user is able to perceive and apply functions. Moreover, since most digital tools enable many functions, and since with the current data-collection methods it is not clear which gross-tool-function is actually in use at any given moment, the approach specified here generalizes across the subset of actual functions for each tool to identify a single prototypical function for each tool. This prototypical functionality for each Tool is called its Task.

So each Tool has a Prototypical Task, from which is also derived a prototypical Activity Focus. So any time an application is launched, the prototypical Task determines the default Activity Focus. But the user can easily specify a new activity focus, by moving to a new Activity Space while ‘holding on to’ (e.g., command-clicking) an Application Window (or dragging windows between Spaces in an Overview mode).

6.6.1 - Study Design

The Observational Case Study is a single-case design (Ninci and others 2015; Hurtado-Parrado and López-López 2015; Wieringa 2014; Franklin and others 2014; Parker and Vannest 2012; Elliott 2010; Perdices and Tate 2009; David Morgan and Robin Morgan 2009b; 2009a; Ximenes and others 2009; Zendler and others 2001; Stocks and Monica Williams 1995; Dahlstrom 1983; Kazdin and Tuma 1982; Horne and others 1982). This longitudinal study tracks a wide range of
digital events related to the sequence of digital Application (e.g., Tool) Usage Cycles (e.g., sessions) by the tool user’s current Activity focus over a four-year period.

### 6.6.2 - Technical Preparation

Before data collection was possible, each **application** installed on the user’s system required manual classification within each major component in the FlowSpaces system-of-systems (For a detailed discussion of each application used in this system-of-systems, refer to section 6.4.5, *Integration Example*.) Thus, I began by manually classifying all tools on a user’s system (e.g., using the Tags, Slife, OS X Spaces, FlowSpaces, and DockSpaces apps.) (See also sections 6.3.1 **Task** for further details on the classification protocol.)

This Tool-Task ‘taskonomy’ (Dougherty and Keller 1982) became the basis for building the FlowSpaces system-of-systems to collect longitudinal data as initial Case Study to measure and capture analyst tool usage over time across each Activity class of tools. (See section 6.5, *The Solution*, for further details.) But it also became the basis for further explorations of how to usefully and useably constrain free, open-concept tagging using predicate structure, hierarchy-embedded tags, and process-state updating tags in Chapter 7, m3 [em-cubed].

### 6.6.3 - Data Logging

This section offers an overview of the process of data capture, first describing what data are captured, then briefly describing the means used for data capture.

Every Application Usage cycle is logged by Window Title, by Prototypical Task, and by current Focal Activity. Beyond App-Usage Cycles, three further hierarchical events are logged. First is the duration of usage of each Activity-specific Custom Dock. Second is the duration of stay in the current focal Activity Space. Next is the duration of Active System Usage cycles. (Conversely, cycles of system inactivity are also logged.) Several more descriptive events are logged, that provide count data for keyboard shortcuts, keystrokes, and mouse clicks, as well as the text-stream produced by keystrokes, and the keyboard shortcut key-combinations.
I have used off-the-shelf and open-source packages to capture (e.g., Slife, BackTrack) and process (e.g., splunk>) the data whenever possible. (For a complete discussion of the tools used for data generation and processing, refer to Appendix W. FlowSpaces.)

Though both logging apps commonly store data in an SQLite database format, each off-the-shelf data capture application approaches the event-capture process with somewhat different goals and methods. BackTrack captures the application and window names while recording user interface manipulations of mouse and keyboard for each window with sub-second resolution. But this higher precision comes at a cost of greater application instability (and consequently lead to some data-loss). Slife captures data to a variable-width resolution (e.g., a two to five-second), capturing only the duration of user focus on each application window by window name (e.g., not the sub-second array of events prior to an active window receiving its name and focus in the User Interface). Here, precision is traded for better valid-event coverage consequent to fewer crashes.

However, only some applications (e.g., HyperSpaces and DockSpaces) produced application event notifications (using Growl), which I captured into an XML log file. Whenever the existing system logs did not capture the targeted system and application events I required to generate event endpoints, I wrote my own log-producing macros (using Keyboard Maestro and Growl). These macro-logs mostly capture various targeted keyboard shortcuts, primarily those that specify the mode of navigation between Activity Spaces. In particular, these logs identify intentional Activity Space Selection events as differentiated from incidental Activity Space transitions due to changed focus to an application bound to another Activity Space.

6.6.4 - Data Ingest

Identifying the log files that contained key events, generating customized logs, and gathering these log files was only the first step. This raw semi-structured data then had to be processed with a custom-built, hand-coded text analytics system (using splunk>). I thus used splunk> to customize my own log file and database processing data-ingest system. This system enabled me to write my own event-extraction protocol, and hand-code my own data-processing queries, as required (using splunk>). This process is further detailed below.
Splunk> is a real-time temporal event analytics platform that is designed to facilitate the conversion of timestamped computer-generated system logs into analyzable transaction or event data. The information contained within system logs is typically semi-structured. This means that data stored in string format (e.g., text) is separable into fields & values. Splunk> offers an integrated environment for customized parsing and data extraction, as well as data warehousing management for datasets extracted in real time from time-stamped text-stored semi-structured log data. Thus, the splunk> system is a programmable, command-line data-extraction and visualization environment.

To be able to do anything analytical with the logged data, the wider Splunk> ingest system had to be first set up to access and process the range of system log files, user-generated log files, SQLite databases detailed in the previous section. So FlowSpaces uses splunk> to extract events from three log sources. First from the custom XML and TXT event logs generated using Growl & Keyboard Maestro (See Appendix W for a discussion of each tool used in the system-of-systems built to capture FlowSpaces data). Next, system events are extract from OS system logs. And finally, application event data are extracted from the SQLite databases created by Slife and BackTrack (application focus & keystroke) logging applications.

For each new data type (e.g., Growl XML log, system logs, App-specific SQLite Databases) the splunk> analytical platform requires extensive customization. Beyond customizing unique log source processing templates, Splunk> users must also build a custom event extraction routine for each element of data the system designers identify as requiring extraction.

Once data is being reliably ingested, data must be improved with post-extraction computation, classification, & analysis using command-line queries. These results are added to the raw & extracted data, and it is all shuffled, reorganized, converted, formatted, visualized and exported using command-line queries. Next, using tableau, the extracted and enriched transaction data was analyzed and visualized.

Thus, using splunk>, raw data was enriched, transactionalized, and layered into the five levels described in the next Section. This was accomplished in the command line search language, manually coding queries incrementally. As
discussed above, conversion queries can be found in Appendix Z. Splunk> Data-shaping search.

The fully customized Splunk> system used by FlowSpaces is available for download.

6.6.5 - Analysis & Visualization

The features of this dataset were not known at the outset. Consequently, statistical methods of analysis could not be preselected on the basis of study design. The method employed for analysis of the data is standard in the psychology literature for single-case designs. Visual analysis is complicated in this case because of the scale of the dataset, so special care had to be taken, and special (stacked timeline) visualizations had to be designed to provide an overview of the data that allows visual analysis.

Data features only became apparent through exploratory data analysis using Visual Analytical tools. These, interactions with the analytical creations made during exploratory analysis suggested improvements to the data collection system and refined goals, such as identifying co-used tool pairs and gaps in tool functionality.

6.6.6 - Data Utility

Viewing FlowSpaces data in real-time (using Splunk>) was instrumental to the iterative improvement of both the data collection method & workspace design. FlowSpaces data was analyzed & visualized data classes & distributions using Tableau for exploratory data analysis. This results of this visual analysis will motivate future work, including predictive modelling of user focus & app transition states, and the development and testing of more specific hypotheses.

6.7 - The Longitudinal Workflow Data

The original data collected by FlowSpaces spans four years, hundreds of GB in log files, and billions of logged events of a single user’s multi-tool multi-tasking workflow patterns of 16 classes of work within academic and industry settings.
These raw data are reduced, filtered, and refined using splunk> to custom-build a real-time log text analysis system (For details, see Appendix Z. Splunk> Data-shaping search.)

FlowSpaces records the duration of time spent in each task-space using any task-classified applications, as well as the duration of time spent in each focal window of every application. Other data, like window titles, keyboard shortcuts, mouse clicks, and keystrokes are also recorded.

Thus, FlowSpaces produces five distinct levels (or streams) of data. These are discussed below.

6.7.1 - System Activity State

First, the system activity state registers whether or not the user is actively using the computer. A new indexed system activity session is created upon detection of any of the following logged system events: user login, system startup, system crash recovery, clamshell (e.g., Laptop lid) open, system wake, or screensaver deactivation. A session is closed upon any of the following: user logout, system shutdown, system crash, system sleep, laptop clamshell close, or screensaver activation. Within each activity session, user interface activity is monitored to

Figure 6.7.1. Floating Average of tool usage by Activity Space, Tool, and Task.
determine the duration of actual activity. Subsequent mouse clicks or key strokes not logged within 30 seconds of the last logged activity are used to mark an activity session end (at T-30sec) and the beginning of a new activity session; unless the subsequent session begins within a further 90 seconds. Thus, sessions “timing out” by default (as is common while reading text on screen) but beginning again within 90 seconds are grouped as a single system activity session, and a running total of timeouts (e.g., the count of joined timed-out events) is logged to each activity session. Durations of joined timed-out events are adjusted to account for time-out durations. System activity session index data is used to classify lower level event data. Thus, each lower level event data is typically contained in a system activity session. (Though due to data quality issues, some events are logged between activity sessions, e.g., whenever the Growl or Keyboard Maestro logging frameworks failed.)

6.7.2 - Activity-space State

Second, the Activity-Space session registers which Activity-specific desktop space is currently active. A new indexed Activity-Space session is created at transitions between task-spaces. Durations of task-space sessions are calculated post-hoc from timestamped logs, as is the sequence of N+1 subsequent to N-2 prior Activity-Space transitions. Thus, each task-space session includes the task-space states from previous and subsequent sessions. Activity-space session index and previous and subsequent state data are used to classify lower level event data. User task sessions can be interrupted by system activity sessions, so that neither inactive uptime nor system downtime are counted in task-space session durations. Thus, task-space sessions can be abbreviated at either end by the extent of temporal overlap between itself and a system inactivity session (e.g., The inverse of the system activity session), or even have multiple overlapping system inactivity session durations subtracted from them. This reduces the time logged within any given task-specific desktop to account for both system shutdowns and occasionally leaving a system on, but inactive, overnight.

6.7.3 - Tool-Usage State
Third, the **tool usage state** registers which application is currently active. A new **tool usage session** is created at transitions between applications. Durations of **tool activity sessions** are calculated post-hoc from timestamped logs, as is the sequence of prior **tool usage** transitions. **Tool usage sessions** can also be abbreviated by **system inactivity sessions**.

### 6.7.4 - App window State

Fourth, the **app window state** registers the window title of the currently active application. A new **app window session** is created at transitions between windows. Durations of **app window sessions** are calculated by logging applications, and are validated post-hoc from timestamped logs, as is the sequence of prior **app window** transitions.

### 6.7.5 - UI events

Finally, the **UI events** (keystrokes, mouse clicks, keyboard shortcuts) are recorded for each app window session. Aggregates of keystrokes, and sums of mouse clicks and various keyboard shortcuts are calculated for various higher level sessions.

### 6.7.6 - Hierarchy and Containment

Thus, a single **system activity session** will contain >0 focal **Activity Space usage sessions**. However, since that **Activity Space session** could have been identified many **System Activity sessions** previously, the system is designed to look around in the data, to read N-1 to N-3 prior session classes at each relevant level.

A single **Tool Usage session** is typically contained within 1 **Activity Space session**. However, it may also overlap 2 or more Activity Space sessions, whenever a single application window is dragged between 2 or more desktop spaces.
6.8 - The Exploratory Analysis Case Study

Previous sections have detailed the collection of a longitudinal single-case dataset on functional patterns of digital tool usage sessions by prototypical task and current activity focus. This is a very deep, and very rich dataset. The discussion here will not consider all the possible directions for analysis that this dataset provides. (But for some illuminating possibilities, refer to the following Section 6.10, Future Work.)

This section introduces the method of exploratory data analysis (Sarvghad and Tory 2014; Cheng and others 2013; Nogami and others 2012; Lam and others 2007; Berthold and Hall 2003; Kaski 2001) and visual analysis (Ximenes and others 2009; Eler and others 2009) and exemplifies the combination of the exploratory data analysis method supported by visual analytic tools with single-case visual analysis to identify new empirical patterns for future statistical analyses. This section will show that by observing transitions between tasks and applications within a focal Activity Space, I can identify distinct patterns of tool usage to characterize different types of application.

For example, exploratory analysis revealed at least three distinct transitional patterns, or scenarios, in the transition data. Using the FlowSpaces data, it is possible to see where prototypically non-analytical tools take a frequent role in analysis. Next, we can see contiguous app-usage patterns, that would seem to be indicating mature within-app toolsets that succeed in supporting analytical focus and optimal flow experience (Csikszentmihalyi 1991). And finally, we can identify which tools and usage sessions presage a functional shift to a new, distinct Activity cycle in the digital tool usage workflow.

These usage patterns were not hypothesized a-priori, but were discovered using a data-grounded methodology of visual analysis (Ninci and others 2015; Ximenes and others 2009). Validity of the patterns derives from the coverage of the data, which is nearly population data coverage (all events for 1439 of 1460 Days) for application usage and Activity Space usage events. These findings are consonant with the literature on workflow design (Kohlhase and others 2013; Chinthaka and others 2011) which indicate that the design of technical and scientific workflows is complicated by non-standard sequences and boundary cases. Overall, the visual
inspection and analysis method is widely practiced in psychological studies using a single-case design (Ninci and others 2015; Kazdin and Tuma 1982; Horne and others 1982), and is therefore appropriate here, with one caveat: single-case studies rarely have this density, richness, and ‘big’ timescale. Specialized views had to be designed in Tableau to achieve non-aggregated overall view of the data.

6.8.1 - Reflections

FlowSpaces has logged over a half-million uses of nearly 900 apps; totaling 6107 hours of active system usage over 4 years of event data. Details of the Data Logging methodology are provided in the previous Section 6.6.4. All events have durations that have been adjusted as required for verified user activity (power cycles, screensaver, sleep, mouse or keyboard input, and network wakes). Thus, only app usage sessions with verified user interface activity are considered.

Figure 6.8.1.1. Stacked Timeline View of Tool Usage over time. The minutes of the day are on the horizontal axis, while each day in a four-year period is on the vertical axis.
In order to visualize data with this level of detail, an extremely high resolution display is required. Therefore, the following images, despite their full-page size, are *thumbnails* provided for illustrative purposes only. The full resolution images and interactive versions are online (http://www.yoroles.com/researcher/dissertation/). In this *stacked timeline view*, each horizontal line represents the course of one day. Horizontal progress aligns to each of the 1440 minutes in a 24-hour period.

Redundant logging systems produce similar data to different levels of temporal granularity. Below, the rightmost column is logged by BackTrack at a millisecond resolution, without automatically reducing app durations to account for user and system inactivity. The BackTrack dataset can therefore not easily be perfectly aligned to the Slife dataset, and while it records nearly twice as many events (due to its finer grained temporal settings), it offers a face-value validation of the longer-running and sparser Slife data, which is used for most of the data analysis discussed in this section.

*Figure 6.8.1.2. Redundant logging system verified the basic usage patterns over time, though different temporal granularities in the records create different density patterns (the rightmost column does not bin data below a given duration threshold, while the leftmost column does).*
Figure 6.8.1.3. Parreto Distribution of aggregated Tool Usage shows proportional contribution of each Application to the overall tool usage.

A ranked proportional summation of event durations seems follow an extreme Parreto distribution. Thus, in this longitudinal case study covering four years of Application Usage cycles (N=587,000), about 5% of apps (e.g., 45 apps) are used 91% of the time (12% of apps, or 108 apps, are used 97.5% of the time).
Figure 6.8.1.4. Parreto Analysis by Activity Space shows the relative contribution of each application to the total App Usage logged in each Activity Space.

A Parreto analysis of proportional Tool Usage by Activity Space (Figure 6.8.1.4) shows that some Activities are dominated by a small group of applications (e.g., Search), while others activities are accomplished with a wide collection of tools (e.g., Manage). Thus, the proportion of all Tools used in a given Activity Space
required to account for 97.5% of the captured usage sessions in each Activity Space a range from 9% to 27%. This is clearly

6.8.2 - Transitions

When event duration data are viewed by Activity without a floating average, the data presents as noisy, with extreme ranges, and most data points close to the shortest end of the duration scale.

Figure 6.8.2.1. Raw App Usage Event Durations by Activity have an extreme range. Density patterns suggest a lower average. In 2012, the collection procedure was extended to capture the activation of Screen Savers. This notably reduced the variance in event durations.

A more visibly linear pattern, below, is produced by taking the floating average of the durations of all app uses in the given minute (by referring application and activity space), considering averages similarly derived for the previous 12 hours, displayed on a logarithmic duration scale.
Figure 6.8.3.2. Floating Average of App Usage Event Durations by Last Application and Activity Space. While a linear pattern is evident for continuously and frequently used App Transition Pairs, the extreme variation in referring Applications that are used neither continuously nor frequently produce a background noisy distribution which cannot be resolved into a linear pattern using a floating average.

Each dot is an application icon that represents 1 minute in the last 4 years. For each minute, one app icon is displayed for every Referring Application to the Current App. The raw count of times a given app/referring-app pair has been used in that minute is indicated by the size of the icon. Dots recognizable as icons are therefore indications of a continuous transition between uniquely titled windows within an application, or App Window transitions. The largest icons indicate a rate of App transition roughly every 4 seconds. Color halo indicates the Prototypical Task assigned to each App. The size of the color halo indicates the count of how many times Referring Apps used in the given minute.

When a floating average is taken, the appearance of horizontal and vertical lines, and ‘noise’ or fuzz in the data each indicate a distinct phenomenon (a pattern in application usage actions) that is nearly continuous (horizontal), an app usage that begins suddenly de novo (vertical), or simply does not occur at close enough intervals or frequently enough to produce a linear pattern (noise).
Transitional patterns can be disaggregated by referring app and next Activity Space. (Figure 6.8.1.1.3.) The first column indicates transitions between windows in a single app. These are continuous or focused uses of an app. Lines are thicker and icons larger, indicating continuity and high frequency. These indicate focused, high-flow single-app usage sessions. Analytical tool builders should note, your app does not show up here, it is not enabling flow within the given Activity or task-space. Apps that do show up here internally offer sufficient tools to permit deep activity focus.

Figure 6.8.2.3. App Usage Event Durations by Transition Type. First column is Within App (Window) Transitions. Second column is Between App transitions. Third is between Activity Space transitions.

The second column indicates multitasking transitions between apps that maintain the same Activity class (or desktop) as the referring app, where the referring app is not the current app. This represents Switching between apps while maintaining focus on a core Activity, or focused multi-app single-tasking. Tool-builders should note, that apps here are used in concert with other apps. Builders should look at what referring/next apps offer that their app does not and consider
building that functionality into their app to create better and deeper within-activity flow and focus.

The third column indicates transitions between Activity spaces (or Desktops). This represents progression or steps in Workflow. These apps foment a shifting Activity State. (e.g., Right before I moved out of this Activity, I was using this App).

Rows indicate a collection of Activities that are differentiated in the raw data, but are joined here to even the distribution of lesser used activities, like reading, annotation, design, etc.

![Figure 6.8.2.4. Moving Average duration by Referring App of the Writing app, Scrivener.](image)

The next two images ZOOM IN to focus in on three Activity groups and do not distinguish between Transition Types as above.
Figure 6.8.3.5. Aggregated Activity Spaces Emulate & Analyze, Seek, and Read, Note & Write. Each Aggregated Activity Space exhibits a unique usage pattern.

Figure 6.8.3.6. Floating Average of the most-used Search application. Note its use in both Writing and Analysis & Emulation contexts.
6.8.3 - Interpretations

FlowSpaces offers unique method for collecting and processing data to identify predicted scenarios formed by combinations of Activity Focus (e.g., length of stay within an Activity Space or rate of transition between Activity Spaces) Tool Focus (e.g., the length of usage of a Tool or the rate of transitions between Tools), and Task Focus (e.g., the length of stay within a Prototypical Tool Task or rate of transition between Prototypical Tool Tasks).

The first expected case is the prototypical Tool Usage case whenever Tools are used for Tasks that match the Tool’s prototypical Activity. For example, when a reader Tool is used to read updates in a subscribed RSS feed in the Reading Activity Space. When a single Tool App Window is in continuous use for a long time, this first case is called single-focus single-app single-tasking (sss). When multiple Tools are in use in a single Activity Space and each Tool belongs to that Activity Space, this case is called the single-focus multi-app single-tasking scenario (sms).

Second is the peripheral Tool Usage case, whenever Tools are used for Tasks beyond the Tool’s prototypical Activity within another Activity Space. For example,
when a reader Tool is used to read! updates in a subscribed RSS feed in the Analysis Activity Space, as a part of a larger focused Analysis workflow involving various Tools not prototypically assigned to the Analysis Activity. This second case is called the single-focus multi-app multi-tasking scenario (smm).

Third is the narrow case of retrieving Tools to be used for Tasks beyond the Tool’s prototypical Activity. For example, when a reader Tool is launched and then dragged to the Analysis Activity Space to be used (to read! updates in a subscribed RSS feed) as a part of a larger Analysis-focused workflow. This third case is called the multi-focus single-app multi-tasking scenario (msm).

Fourth is the case of transitioning between Activity Spaces to use Tools within their prototypical Activity Space. For example, when a reader Tool is used to read! updates in a subscribed RSS feed in the Reading Activity Space; and then a collector Tool is used to collect! updated RSS feed data in the Collecting Activity Space; and then a converter Tool is used to convert! the feed data, stripping it of HTML tags in the Converting Activity Space; and then an analyzer Tool is used to analyze! the feed data in the Analyzing Activity Space.

When the App Window Title indicates the same data is being used in different Activity Spaces, this fourth case is called the single-data multi-focus multi-app multi-tasking scenario (smmm). When the above example happens with a different Object of action for each Tool by Activity Space (e.g., not all updated RSS feed data), when the number of transitions between Activity Spaces exceeds a certain threshold in a specified unit of time, and/or the duration of stay in each Activity Space is lower than some specified threshold, this case is called multi-focus multi-app multi-tasking (mmm).

Thus, this Chapter has thus examined in depth how the progression of Tool usage workflows within and between Activity Spaces can be captured, analyzed and visualized in order to better understand multi-tool multi-task Tool Usage Cycles in the context of Activity Space Usage Cycles.

6.8.4 - Limitations

Many tasks contribute to shallow preparatory analysis. For example, naming files, creating file structures, moving files into file structures, tagging files with terms
from an open taxonomy, and classifying files within a closed or constrained ontology are all equivalent schema-building, updating, & maintaining tasks that are important preparatory analytic tasks.

The data collected by FlowSpaces should ideally record these tasks regardless of the workspace in which they occur. However, since the OS provides a common User Interface for many of these simple file management tasks using untitled ‘modal dialogue windows’ that are not logged by Slife and Backtrack, FlowSpaces does not record these key preparatory analytical tasks. The FlowSpaces data could be significantly improved by segmenting out SAVE AS… modal dialogue windows & actions.

Even without recorded data on these particular preparatory analytical tasks, the composite layout of FlowSpaces (e.g., ‘higher’ workspaces accommodate all ‘lower’ tasks) means that use of certain tools or workspaces explicitly classified ‘lower’ than analysis are also an indicator of preparatory analysis. For example, use of a tagger Tool for tagging requires shallow analytical judgments and classification.

Thus, analysis is accomplished using tools whose primary class is not analysis and in workflow states beyond the desktop workspace dedicated to Analysis. More profoundly, shallow preparatory Analysis happens in all workspaces, with all tools.

6.9 - Recommendations for Analytical Tool-Builders

The Exploratory Analysis Case Study identified three interesting user patterns that should help both tool builders and users improve their respective work:

1. When the user is using an app and only shifts focus within that app, maintaining task focus, that app can be seen as providing all the functionality required for the task; and thus as facilitating ‘highly focussed flow’ (or tight flow) state within the workflow.

2. When the user is using a pair (or larger group) of apps that collectively provide the functionality required for a task. This app shift maintaining activity focus should tell tool builders to build functional collaborative (import/export) pathways between co-used apps; or to implement co-used app functions in order to enable tighter in-app activity focus.
3. When the user is transitioned automatically or intentionally, to another workflow state; builders of current apps can look at next apps and next activities as targets for building improved functional & collaborative (import/export) pathways between subsequent activities. Also, apps used just prior to a Workspace shift (and the apps whose activation caused the activity shift) can be looked at as precursor events. Future Work will look at predictive modelling of workflow states or app selections.

6.10 - Future Work

The FlowSpaces dataset has sufficient richness and depth to provide research directions for years to come. Some interesting potential avenues for future work are outlined briefly in this section.

6.10.1 - Understanding the role of text in the user interface of analytical tools

In future work, I will take a finer grain view of this multi-tasking workflow dataset to focus on the production and processing of text during analysis. The non-

![Figure 6.10.1.1 - Linear model of word count by Application duration](image)
linear editing process and the possibility of changing the insertion point with the keyboard or mouse have important implications for the recorded log of text production. Chiefly, this means that the aggregation of logged keystrokes does not produce a coherent text. Thus, the recorded log requires disaggregation of focal adjustments (e.g., cursor repositioning via mouse or arrows, selections). This disaggregation of UI-hotkeys from standard keystrokes has already been done in the data. Also, the incoherent sequence of the recorded log makes the statistical ‘bag of words’ approaches to analysis of logged text a reasonable analytical approach.

6.10.2 - Extracting tasks from tool UI

In the current FlowSpaces system, Tasks are either manually assigned, or analytically extrapolated by agreement between the prototypical Task and Activity assignments or by identification of peripheral usage in non-prototypical Activity Spaces. However, it is possible to automatically and exhaustively determine each Task that the Tool claims to support by mining its User Interface labels. First, each Tool’s application window user interface (e.g., Menu Items and Button Labels) is scraped. Next, each label in the Menu structure is analyzed to derive a predicate-argument structure for the task (e.g., the action and the object of the action). This procedure should identify both implicit (e.g., File>New Project, where the task create! is unstated, becomes create![File]-for-New Project) and explicit tasks (e.g., Document>Snapshots>Take Snapshot becomes take![Snapshot]-of-Document).

Thus, each Task is listed in relation to the object upon which the Task is undertaken (e.g., task![Object]-Proposition-Locus) and Tasks are arranged in descending frequency. The selection of the prototypical class for each tool could then be selected from an exhaustive empirical set of locally valid options, rather than from a pre-existing global class structure.

6.10.3 - Automating tool classification

In future work, the manual Tool classification step could be augmented or eliminated (e.g., automated) by various means. For example, a principle components analysis of the Tool’s primary Tasks (identified using the method
proposed above in section 6.10.2) could be used to present the user with the top
tasks (e.g., task!/Object]-of-Locus) for each app.

The user could then select from the entire ranked population of Tasks enabled
by each Tool when assigning the prototypical Activity class. Alternatively, the
activity Assignment could be automated using supervised learning methods to
select the top Task and assign secondary or even tertiary Task classes as well.

6.10.4 - Activity space navigation study

Does the mode of Space Selection affect the duration of activity focus?
Do expert users navigate directly to their target spaces via keyboard shortcuts,
or do users select Spaces by launching apps by Spotlight, Finder, or the Dock?

6.10.5 - Quantify next focal target search difficulty

The search difficulty for the next focal target is another important factor
contributing to the complexity of digital workflow. (Greitzer 2005). Future work
will attempt to quantify difficulty of the search task.

6.10.6 - Test Hypotheses

The patterns identified in the FlowSpaces single-case study can be formulated as
hypotheses for future studies. Example hypotheses for future testing might include:

H1 Tools are most frequently used for their prototypical task in their default
Activity context.

H2 Accomplishment of a single focal Activity often requires a dynamic flow
between tools from several different task categories.

H3 Experts maintain meta-cognitive awareness of focal Activities and mobilize
existing knowledge regarding ‘what to do with peripheral tasks that are needed to
accomplish this activity’ while executing analysis. Training in these two areas will
help novices improve their ability to execute analysis. Cuing of focal Activities and
ranked presentation of historical workflow trajectories will help experts improve
their analysis process.

H4 Low task familiarity, low UI familiarity (e.g., awareness of what can be done
with the UI, and experience using the UI) and low meta-cognitive awareness of
both the current focal task and its relation to peripheral, satellite tasks will reduce the number of Activity-focused transitions.

Future work here would initially entail extending the case study to see if initial 4-year observations generalize to a larger number of analysts. Eventually the study could also be extended by setting up a small experimental study to observe where respondents execute work: in the default space, or in the expected focal space. Evaluate the impact of task familiarity, UI familiarity, and meta-cognitive awareness of relations between focal activity and peripheral task requirements on the conceptual location (Activity Space) of work.

6.10.7 - Long-term study of multi-level work fragmentation

(Mark and others 2005)’s research on interruptions in workflow consider that work fragmentation has two main aspects: the length of time spent in a continuous activity, and frequency of interruptions of that activity. The FlowSpaces dataset has both a higher level concept of an Active Cycle, and an index of how many <90 second interruptions are reconnected per continuous Activity Cycle. Moreover, it is also possible to take a wider view of continuity between longer interruptions, so that more distal Activity cycles could also be reconnected. Future work can report on both the brief inactivities and the longer absences.

6.10.8 - Predictive modelling of activity space navigation & tool selection

Next steps should also include building a Hidden Markov Model that predicts what Activity Space and Application the user is most likely to use next based on the multidimensional FlowSpaces dataset. The workflow monitoring and prediction system could then be extended to query statistically unlikely procedural choices to have the analyst provide the ground truth factors for surprising path selections.

6.10.9 - Network modelling of activity space navigation & tool selection

The FlowSpaces dataset has been designed so that each event ‘knows’ its prior and posterior neighbours, as well as several features of those neighbours. Thus,
transitions can be analyzed and represented as a network of triples, though minor revisions to the data structure may have to consider removing within-App transitions.

Figure 6.10 - Gephi Network analysis of first two years of data as a transition network.

6.11 - FlowSpaces Synopsis

To review, FlowSpaces is a low-cost, passive digital tool-usage workflow monitoring system that records activity-focused multi-tasking app-usage with minimal ongoing user input. The system produces a rich, deep, and wide dataset designed as an embedded networks of triples that record transitions between active system usage and inactivity. When the user is active, FlowSpaces records transitions between functional Activities, applications within those workflow states, Window states within applications, and user interface activity in Application Window. Each event ‘knows’ about its preceding and following events, at the levels of the contiguous work session and current task-space.

Focused Activity workflow states are indicated by user-initiated or automated transitions to activity-specific Desktop Workspaces. All apps are bound to 1 prototypical activity, so launching them automates a focal shift to that Workspace.
However, any app window can easily be dragged to any other Workspace. Manual dragging app windows between virtual desktop Activity Spaces reclassifies windows to the ‘actual current’ workflow Activity state (from the user’s perspective).

Custom Docks for each Activity Space collect all the tools tagged with the focal Tasks of each Activity Space. Thus, the combination of FlowSpaces Activities and Docks contribute toward what Hill et. als., call ‘“activity hubs,’ which marshal all the tools and information needed to perform work, [so that] users will work in a more complete context for their actions and be burdened by fewer manual integration tasks.” (Hill and others 2006)

Clearly, however, more work is required to marshal together all tools and data required for analytical work. Moreover, as (Dougherty and Keller 1982) predicts, digital tool users including analyst will need further contextual specification before a functional classification of tools can be maximally useful. In partial answer to this requirement, DockSpaces+ relies on $m^3$ [em-cubed] to enable the classification of each Application (e.g., Tool) by both the prototypical action that it enables, and the prototypical object upon with the action is applied. (Further non-prototypical actions and objects of the tool can also be specified). Thus, Word(TM) is an editor, since its core task is edit!documents, while the core task it enables the user to accomplish is write!documents. And Google Chrome is a seeker, since enables the user to seek!information (though from the perspective of the tool, its primary function is to display!documents and permit the user to read!documents).

Thus, the $m^3$ [em-cubed] Research Project, described in the next Chapter, outlines how further specification of tools can be used to produce and record ad hoc Activity-focused workflows, and how these workflows can be chained together to accomplish analytical (and other functional) goals. Moreover, by marshalling tools and data together by Activities, we can see how multiple tasks and tools are required to achieve analytical goals.
Part III: Supporting Ubiquitous Shallow Analytical Judgment

To review, Part I, Foundations, has established the conceptual background as a common foundation that forms the basis for the research projects reported in Parts II, Modelling Cognitive-Functional Analytical Process, Part III, Supporting Ubiquitous Shallow Analytical Judgment, and Part IV, Modelling Conceptual Structures in Deep Analytical Reading, of this Dissertation.

In the foregoing Part II, Modelling Cognitive-Functional Analytical Process, the dissertation moved forward from introducing domains and concepts toward introducing and improving upon models of the whole analysis process, and creating methods for gathering data on digital functional workflow dynamics.

In this Part III, Supporting Ubiquitous Shallow Analytical Judgment, moves focus from the highest level to the intermediate level, where some Activity-specific and many cross-cutting preparatory analytical tasks are under-supported in current analytical tools.

Taking this human scale view of the analysis process problem space imbues this Dissertation Research Project with a multi-level focus that moves from understanding the whole, to its widest parts, and then to its deepest parts, keeping the human analyst reader in focus throughout. Thus, Part III, Supporting Ubiquitous Shallow Analytical Judgment, of this research moves to a focus on analysis at the intermediate, preparatory level, which includes discovering and prototyping ways to support cross-cutting cognitive tasks like ubiquitous shallow judgments.
Chapter 7 - \( m^3 \) [em-cubed]

a.k.a. (multi-modal-meta) analytical media management

This chapter begins by reviewing progress through this dissertation to this point. Part I, Foundations, has presented the background and foundations of the two core challenges of Cognitive Limits of human analysts, and Functional Limits of analytical tools, which together define the analytical process problem space. Chapter 1, Introduction, introduces this problem space and Chapter 2, The Problem Situation, elaborates and situates four related dissertation research projects in relation to specific levels of the problem situation. Chapter 3, The Cognition Literature, and Chapter 4, The Analysis Literature, introduce the domains from cognitive science and text-focused analysis that have contributed to the discovery of the way forward toward the solutions proposed for this problem space by each research project.

Part II, Modelling Cognitive-Functional Analytical Process, began by 1) drawing upon my practical experience as an analyst using analytical tools, including those offered by Visual Analytics, and 2) drawing upon the Literatures on pragmatic analytical procedures and higher level cognitive processes. Both were found to agree that certain important cognitive and functional tasks are missing from the standard Sense-Making model of analysis that is widely cited in VA. Thus, the two Chapters in this Part of the Dissertation made the argument that the use of the so-called Sense-Making model as a specification of the analysis process for the design of analytical tools provides one explanation for the clear gaps in analytical functions offered by analytical tools today, and that empirical observation of functional flow in tool usage can help to begin to address those gaps.

Thus, in Chapter 5, Flow, I addressed those omissions in the Sense-Making model of analysis by extending Pirolli & Card’s Sense-Making model into the Cognitive-Functional Flow Model of General Analysis. The Flow Model additionally reconceptualises the components of the analysis process such that each cognitive and functional task is conceived as a network of overlapping requisite and related cognitive and functional sub-tasks. Thus, when a task occurs, its requisite sub-tasks must occur, in some order, and its related sub-tasks may
occur, or may be skipped entirely. The ‘flow’ dynamic accounts for this smearing of boundaries, or simultaneous overlapping among tasks, their requisite sub-tasks, and related or embedded tasks. Elsewhere this phenomenon has been called task interleaving (Jin and Dabbish 2009; Bannon and others 1983).

In the previous Chapter 6, FlowSpaces, I have looked for ways to take a more precise yet more malleable view on functional transitions between digital Tasks and Tools by focal Activity (Ashley Williams 2009; Norman 2006; Gay and Hembrooke 2004; Christensen and Bardram 2002; Bannon and others 1983). My approach addresses the inherent brittleness of a prototype functional classification system (Dougherty and Keller 1982). So beyond classifying Applications according to the Prototypical Task that they enable (Bardram 2006), FlowSpaces secondarily classifies each application usage by its current context of Activity focus, and allows easy updates to the focal Activity class by simply dragging and dropping Application Windows between virtual Desktop Spaces.

This move recognizes two important facts. First, in the case studied deeply here, most often tools are used in their prototypical task in their default Activity context. But, secondly, a dynamic flow between tools from several different task categories is often required to accomplish a single focal Activity. These findings suggest hypotheses for future studies, provide a rationale for extending the case-study N, and also make a case for extending the research with experimental studies. (See Section 7.7, Future Work). While the user maintains focus on one Activity, even switches between Tools and Tasks can be productive. So Within-Activity task transitions are often essential, and not just interruptions as is widely discussed in the literature (Garrett and Danziger 2007; Czerwinski and Horvitz 2002; Czerwinski and others 2000). FlowSpaces data can be used to identify intentional, productive, Activity-focused Tool Task alternation sequences.

This Chapter 7, m³ [em-cubed], pushes the focus downward and inward from overall digital tool usage workflow on a macro-time scale, to analytical tool-usage workflows on a local mezzo-scale, and to specifically focus on capturing the products of cross-cutting shallow analytical judgments made during preparatory analysis. I will attempt to show how user conceptualization and workflow state can be captured in tags and other low-cost annotations. Products of the shallow classification tasks of identifying and labeling features of the analytical media (e.g.,
judgments) can be captured as Tags applied the document container (in the file system), and can subsequently be used as triggers to automate cascades of workflow processes. Strategies for managing a growing collection of annotations are proposed.

Then, the somewhat deeper process (though still of an intermediate depth) of shallow ‘skim’-reading can also begin to identify ‘chunks’ of text as internal features of the analytical media, which can be captured as Highlights applied within the document container at the level of the document contents. These annotations can be embedded within the file as modifications to the file; stored as external, separate metadata files; or as in the case of Tags, saved as hidden file attribute metadata stored in the file system and accessible through command line interfaces or apps. Standard Open Source Annotation applications (Skim (Hoffman and Millitello 2009)) and storage frameworks (OpenMeta (AndersenIronic Software 2009)) can be customized to address management challenges for both Tags and Highlights. Related strategies are applied to create manageable and semantically meaningful subgroups for both Annotation types.

FlowSpaces and m³ [em-cubed] are retrospectively identified as deeply consonant with both Activity Based Computing (Voida 2008; Czerwinski and Kaptelinin 2007; Bardram 2006; Bardram and others 2006; Bardram 2005) and hence with Activity Theory (Ha and others 2007; Nardi 1996; Kaptelinin 1996; Blackler 1993) and also with the ethnomethodology used to derive the particularized tool taskonomy of technological experts (Dougherty and Keller 1982) discussed in Chapter 6, FlowSpaces. The method I intuitively created, in FlowSpaces and m³ [em-cubed], incidentally corroborates Dougherty’s method and conclusions, and conversely, Dougherty’s published findings offer oblique support for the method I used to develop the FlowSpaces and and m³ [em-cubed] approach.

Similar to other approaches taken in Activity Based Computing, FlowSpaces and m³ [em-cubed] can be considered components of an alternative semantic Desktop. So I argue that the combination of FlowSpaces Activity Spaces and task-specific Docks contribute toward what Hill et. als., call ‘activity hubs’ which marshal all the tools and information required to perform work, so that “users will work in a more complete context for their actions and be burdened by fewer manual integration
tasks” (Hill and others 2006). FlowSpaces offers features beyond what is offered in Activity Based Computing systems, such as ActivityBar and Activity Based Computing (ABC) User Interface for Windows XP (Bardram and others 2006) and the Giornata Activity-Based Desktop Interface (Voida 2008) and The Placeless Documents system (Dourish and others 2000).

Clearly, however, more work is required to marshal together all tools and data required for various types of analytical work. Moreover, as (Dougherty and Keller 1982) predicts, digital tool users including analysts will require further contextual specification before a functional classification of tools can be maximally useful. DockSpaces+ addresses this requirement by using \( m^3 \) [em-cubed] to classify each Application (e.g., Tool) by both the prototypical action that it enables and the prototypical object upon with the action is applied\(^{27}\) (non-prototypical actions and objects of the tool can also be indexed). This further specification of the functional classification system allows the dock to have dedicated spaces (e.g., stacks or drawers) full of tools for each class of object, and for each related type of activity (e.g., document editors, database editors, etc.).

Thus, the \( m^3 \) [em-cubed] Research Project, described in this Chapter, outlines how further specification of the functional classification system for tools can be used to label and record ad-hoc Activity-focused workflows in real time as the workflows are happening. Furthermore, these workflows can be chained together to accomplish analytical (and other functional) goals. Moreover, by marshalling tools and data together by Activity, tool users and tool builders can begin to see how multiple tasks and tools are required to achieve focal analytical goals, and both can begin build collections of useful tool usage patterns.

Recall from the introduction to the previous Chapter 6, FlowSpaces, how by applying Dougherty’s method and observations to the analysis process problem space, we may ask the analyst how they group their tools. In this case, Dougherty might observe that in order to achieve a highly effective analytical process the analyst must be allowed to take a flexible approach to the classification and organization of their tools. Dougherty predicts analysts won’t use linguistic Name classes or even Function classes (e.g., Tasks) as their primary organizing principle, because both are too general to be practical; and yet, at the same time, functional
classes also narrowly exclude the data, methods, and products associated with the primary Tasks in the process of analysis.

So what Analysts require is flexible means to *further specify and extend* Function classes by encoding in them their relevant *dimensions of context* (See Section 8.5.5, *CDG+ Meta-Contextual Frame*, for a full discussion of the major deictic dimensions and the contextual dimensions of human experience) that are crucial for the current goal-focused Activity (Dougherty and Keller 1982). Dougherty concludes that, “constellations of conceptual units arising in response to a task at hand” are a better organizing principle for characterizing the analyst’s knowledge structures regarding tool classification and organization (Dougherty and Keller 1982). Here, those *constellations* are ad-hoc sequences of Activity-focused app-task alternations. Or more simply, ad hoc groups of tools needed to get the job done.

Thus, the initial phase at this Research Project focused on building a system-of-systems ‘meta-tool’ to allow the user to specify and extend basic Functional Classes assigned to Tools with their prototypical target objects, and moreover to capture ad hoc evaluative judgments regarding file features, and to capture workflow progress and states. I have previously (in Chapter 5, *Flow*) introduced a general model of analysis wherein analytical reading is an important yet under supported phase in the analysis process.

Moreover, the general *Cognitive-Functional Flow Model of Analysis* recognizes that the analyst’s conceptualization of analog documents, digital files, and their multi-modal contents must naturally occur at many levels and at many preparatory points prior to the deeper reading moment. Consequently, I have argued that analytical tools in general must generally improve their overall support of preparatory analysis, which includes both analytical media management and the capture of surface analytical judgments that are a part of simple classifications and shallow evaluative assessments in various forms of multi-modal annotation metadata.

Thus, the specific cognitive tasks selected from the cognitive-functional Flow Model of general analysis, for application in this Chapter, include search, classification, evaluation, & shallow judgment. Preparation naturally also includes
many functional tasks, like naming, filing, tagging, managing, converting, and editing files.

I have also argued that analytical tools that aim to support analysis of text data must improve their support for various levels of analytic reading, which includes both preparing digital content for reading as well as initial, shallow skimming of digital media, and subsequently executing various deeper analytical methodologies upon their contents.

The prototypes, discussions, and recommendations offered in this Chapter can be used to start analytical tool-builders on their way. In this Chapter I propose required yet missing functionality that begins to address preparatory analysis and analytic reading gaps in Visual Analytics tools today.

Figure 7.1. A Word Cloud view of Chapter 7 - m3 [em-cubed]. The most frequent terms in this chapter A. without and B. with stop-word removal. The most frequent terms excluded by stop-list filtering are relations that indicate deictic attachment (e.g., the, this, that), ownership (e.g., of), inclusion (e.g., and), alternation (e.g., or), state (e.g., with), location (e.g., in, at, to, on), purpose (e.g., for), conditionality (e.g., if), consequence (e.g., then) and means (e.g., by).
7.1 - Overview

This Chapter begins by relating the general purpose of this research back to each previous Chapter, and to my Research Question 2: **How can transparent, low-cost observational methods be designed to produce empirical datasets that capture functional and cognitive processes?**

First, I have reiterated the post-hoc discovery of the remarkably good fit between this Research Project and the principles of Activity Based Computing (ABC) and an ethnomethodology that produced the concept of the taskonomy. Where this work distinguishes itself from Prior Work in ABC, is that in FlowSpaces the transition between Activities is treated more organically, since with virtual desktop system, there is no need to ‘pack up’ the files and windows, or to ‘unpack and set up the activity’ space upon deciding to return to an Activity.

The focus here is to extend the Activity perspective established in Chapter 6, FlowSpaces, from organizing tools for Activity-focused workflows to also considering the organization of analytical media resources (e.g., files, documents, and datasets), and annotating each analytical media resource with the contents of shallow analytical judgments and classification decisions (e.g., identified features and group memberships) and the current state of each file in the specific workflows required to process the file through the entire preparatory phase. Thus, the goal here is to prototype components in a system-of-systems capable of capturing and supporting the targeted preparatory analytical phase task of shallow analytical judgment.

After providing this section-by-section overview, this Chapter continues by providing a summary of related previous work. Of note here is the fact that this review has identified no Visual Analytics tools focussed on Analytical Data Preparation, which involves tasks to Collect, Manage, Convert, and Sanitize Data. Various media collection management tools are therefore considered, from iTunes, to Zotero (Harding 2013; Wareesa-ard 2013; Vaidhyanathan and others 2012; Murimboh and Hollingdale 2012; Lisa Spiro 2008; Hull and others 2008), and Mendeley (Mohammadi and Thelwall 2014; Russo and others 2013; Vaidhyanathan and others 2012; Erlandson 2010; Hull and others 2008), as well as
tools that focus narrowly on the collection of meta-data but not media, or meta-
databases like RefWorks, Reference Manager, and EndNote.

Given the manual effort required for the collection and management of files
(e.g., search, filtering, retrieving, file naming, filing, feature extraction, classifying,
searching for bibliographic data, linking database records to digital objects in a file
system,) and for maintaining linking between meta-databases and files maintained
externally in file systems, the recommended way forward is an integrated (multi-
modal-meta) media management system, like iTunes or Papers3 (like Zotero, or
Mendeley). The key feature of the best of manager applications is the automatic
identification, acquisition, comparison and ingestion of institutional sources of
metadata, based on automatic partial, shallow feature extractions (e.g., Title,
Author, Pub Year) of metadata from file meta fields and content analysis.28

The next section introduces five core concepts that are essential to
understanding the Research Project described here: Users, Annotations, Tags,
Highlights, and Workflow Task States. This Chapter then continues with a
discussion of The Problem that has motivated the design and creation of m³ [em-
cubed] analytical media management as a prototype system-of-systems for
collecting and organizing analytical data and metadata. The Solution presented
here is considered to be a prototype system-of-systems for the real-time capture of
assigned vs. completed workflow status on digital file, annotating files with the
procedural status, capturing conceptual evaluations of analytical media, and
managing analytical media prior to deeper analysis. Thus, the system-of-systems
prototyped here includes The Tagger, which uses Open Source infrastructure to add
Tags and Highlights, and The Leveler, which connects various levels of document
metadata.

Future Work will work toward The Binder to manage and organize analytical
media files, particularly by wrapping similar files together.

Even with the contribution of FlowSpaces and m³ [em-cubed] to help capture
and visualize the flow of tasks, tools, and files in work processes, digital tool usage
remains complex multi-Tool, multi-Task, multi-Activity phenomenon. For this
reason, I suggest that capturing the processing state of files as they move through
workflows may help users return to a complex workflow and pick up without
incurring typically high task resumption costs (due to the need for extensive review). Future work will evaluate these improvements by testing these hypotheses.

With FlowSpaces and \textit{m$^3$ [em-cubed]} to help, this complex domain now has well defined general functional steps and transitions, and now also even some specific workflow sequences recorded in file tags. In the light of findings from the FlowSpaces data on focused transitions in analytic workflows and with the goal of generating and substantiating hypotheses to test in future work, the following section aims to typify the chief concerns and experiential progress of the \textbf{analyst tool user}, in order to build these realities into the \textit{m$^3$ [em-cubed]} prototype \textit{system of systems}.

7.2 - Previous Work

7.2.1 - Extending the Desktop Metaphor

The ActivityBar of (Bardram 2006) provides the analyst with the ability to define, pause, and re-engage in Activities defined as ad hoc collections of Applications, Windows, and Files. FlowSpaces defines Activities initially defaulting to the Prototypical Task assigned to each Application and subsequently by drag-&-drop reassignment of Application Windows to alternate Activity Spaces. \textit{m$^3$ [em-cubed]} exists to create, discover, and catalog productive sequences of Application transitions as Workflows, as discovered through the application of tags for each functional step in the workflow. Other relevant extended semantic desktop approaches include the \textit{Activity Based Computing (ABC) User Interface for Windows XP} (Bardram and others 2006), the \textit{Giornata Activity-Based Desktop Interface} (Voida 2008) and \textit{The Placeless Documents system} (Dourish and others 2000).

7.2.2 - Tagging workflow state and importance as reminders for resumption of interrupted tasks

The approach taken here is to provide a user with a low cost mechanism for tagging files with both features and workflow states. This approach effectively embeds stateful reminders onto files. This implicit expectation is that this should be
helpful in reducing the cost of task resumption. In a study on user’s memory for daily computing activities after a 24 hours and 1 month, Czerwinski found that approximately half of the events highlighted by the study participants as memorable contained a high-level intention or goal, such as a focal Activity, as opposed to the application or document that they were using.

These higher-level events could be supported by a reminder system if the user were provided with a means for efficiently editing or refining initial suggestions of event activity that might be provided by an automated system. Whatever the mechanism provided to support these types of events, the user interface affordance must be fairly 'lightweight', so that the user could easily leave a simple reminder notation with the system prior to a task switch or in the face of an interruption. (Czerwinski and Horvitz 2002)

I propose that tagging is just the lightweight mechanism to allow both the initial refinement of the actual event activity, and the subsequent low-cost retrieval of that annotation through a tagging application UI, command line interface, or file system integration.

7.2.3 - Taskonomy

This Tool-Task taskonomy (Dougherty and Keller 1982) became a post-hoc justification of the method employed for building the FlowSpaces system-of-systems to collect longitudinal data. This initial Case Study measured and captured analyst tool usage over time across each category of tools. (See sections 7.5, The Solution, for further details.) But it also became the basis for further explorations of how to usefully and useably constrain free, open-concept tagging using predicate structure, hierarchy-embedded tags, and process-state updating tags in this Chapter 7, m3 [em-cubed]. The tool-task-object constraint developed here is also similar to approaches in other related fields. For example, Roth employs a similar predicate-argument structure approach to defining actions and their objects in the context of cataloguing interactions for geographical visualizations (Roth 2013).

7.2.4 - Annotation
Marshall discussed how to facilitate the transitions from paper books to the digital library in terms of technologies that support managed annotations on primarily textual digital analytical media (Marshall 1997) (Bae and others 2006), while Arbib argues for the superiority of digital annotation (Arbib and others 1999). Various semantic desktop extensions employ semantic tagging (Voida and Mynatt 2009; Voida 2008; Czerwinski and Kaptelinin 2007; Bardram and others 2006). Even recent Visual Analytics tools are building internal capacity for annotation on visualizations produced within a VA tool (Xia and others 2014; Yang Chen and Jing Yang 2013; Sanfilippo and Chikkagoudar 2013; Javed and Elmqvist 2013; Legg and others 2013; Hoferlin and others 2012; Kandogan 2012), and even treating annotations as sources of data in analysis (Dunsmuir and others 2012; 2010; Kadivar and others 2009). But approaches to annotation in VA tools are all tool-internal, so they do not extend beyond the single VA tool and generally are not able to be shared or accessed except by and within the VA tool.

7.2.5 - Highlighting Chunks

Stoffel uses color-highlights to differentiate structural elements of documents (e.g., Header, Titles, SubTitles, Authors, Abstracts, Introductions, page Numbers, etc.) in his VA approach to Automated Document Structure Analysis (Stoffel and others 2010). While at the intermediate shallow skim reading level, my concerns chiefly focus on user-driven semantic chunking of text into tagged phrases, the m³ [em-cubed] prototype for simple highlighting also uses a structured color scheme to differentiate between different semantic concepts.

7.3 - The Concepts

In order to ground the discussion of m³ [em-cubed] offered in this Chapter, I begin this Section by introducing five core concepts; Users, Annotations, Tags, Highlights, Workflow States.

7.3.1 - Users

The m³ [em-cubed] user is a highly skilled computer-literate analytical reader who regularly collects, filters and processes analytical media, which here includes
regularly reading digital reading materials. The assumption, common among analytical tool builders, that users are already experienced or highly skilled analysts (Hoffman and Millitello 2009; Hoffman and Militello 2009; Hutchins and others 2007; Klein and others 2006a; Crandall and others 2006) can be unpacked in a thought experiment by briefly comparing novice and expert approaches to beginning analysis with a blank slate. Consideration of the process whereby skill development and experience building occur should identify foundational skills and procedures as leverage points for analytical tool builders (Pirolli and Card 2005; Russell and others 1993).

On any new computer system, the bare operating system contains neither the files nor the tools necessary to prepare and analyze the data resources of interest to the analytical tool user. The user must therefore begin to seek out, acquire, organize, (Hoffman and Millitello 2009) and use data files and the tools required to manipulate them. While the novice will take considerable time executing first searches, or basic searches without the benefit of existing collections, experienced users will bring both tool-using patterns and data resources with them to a fresh installation. This is to say, experienced users already know what to do, or rather, what they do with their files. Since no single application will ever do or enable everything that the user needs or wants done, experienced users are generally familiar with a range of tools that accomplish their goals for the data resources they have collected.

Thus, given any new computer system, all analytical users primarily face three tasks: Collection, Organization, and Learning. At the highest level of generalization, the two main types of collections of interest in this Chapter are applications (or tools) and data resources (or files). Users learn about new tools and files, where to find them, how to use them, how to evaluate them, and how to manage them. Over time, users can be expected to develop specific strategies for organizing and using these various types of collections.

Thus, the collection process for both files and tools is highly idiosyncratic and unique to each user. It is this history of individual choices made while managing a collection which make the process user-specific. FlowSpaces + m³ [em-cubed] can be used to examine this process in detail, to identify the cost structures (Pirolli and Card 1995; Russell and others 1993), of specific organizational strategies, in order
to select ideal configurations (of tools and files) and avoid costly ones. Since not everyone can be expected to have the time or inclination to curate every collection of data resources and their complimentary data-manipulation tools, I hypothesize\textsuperscript{29} that the quality of collections will vary with degree of expertise. The FlowSpaces data shows that, for the single user studied, most collections (of both tools and files) are neither highly nor evenly maintained.

7.3.2 - Workflow Task States

In m\textsuperscript{3} [em-cubed], workflow Task States are captured as tags that indicate the status of a file with reference to some Activity, which use a semi-structured naming convention to distinguish between related states. Many workflow states are possible, though the prototypical workflow state marks a file as either assigned a to-do status as ‘do!’ or a completed status as ‘done’.

7.3.3 - Workflows

Workflows are tagged sequences of tasks. Workflows are logged by tagging workflow tasks states, and ‘discovered’ by reviewing the subset of stateful tags applied to a single file.

7.3.4 - Annotations

Annotations are a general class of user-generated meta-data produced by shallow analytical judgments ranging from feature-based classification, through to shallow content evaluation during ‘skim reading,’ and down through deeper analytical reading (Marshall 1997).

7.3.5 - Tags

Tags are extensible metadata attributes attached to the file at the file-system level. They may hold whatever strings the user wishes to attach to content.

7.3.6 - Highlights

Highlights are user-specified chunks of text that are assigned a user-specified label and distinguished as a colored area superimposed over a text file. Typically,
highlights could be of any scale or unit length, though in this work I focus on phrase-size chunks, working towards creating links between highlighted chunks to facilitate assignment of emergent Rhetorical Structure Theory ‘propositional’ relations that systematically determine how highlighted chunks are related to each other (Mann and Sandra Thompson 1988; Taboada 2006).

7.4 - The Problem

As a user becomes more experienced, they come to know their files and tools more closely. The result is simultaneously a refined sense of a tool’s capacity to effectively manipulate files and a subtle appreciation of the properties of particular data resources. Knowing which tools are appropriate for which manipulations of which files with which properties is the mark of an experienced user.

However, knowing how to effectively and efficiently pair tools with files is the mark of an expert user. Experts will presumably recognize the value of organized collections and while they may benefit from reminders or automatic assistance with maintaining organization of their collections, they do not need to be told why they should do so. Moreover, experts will presumably not waste time collecting or organizing non-essential file features, or optimizing for nonessential procedures.

Analytical tool design should prompt users toward the best available procedures and data, by optimizing the organization of tools and files so as to highlight the relevant capabilities of tools and to foreground the salient features of data resources. If analytical tools cannot highlight their own capabilities outside of actual usage sessions, then alternate external mechanisms should be designed to do so. Likewise, for data resources.

Thus, the highest level purpose of the FlowSpaces + m³ [em-cubed] preparatory digital work environment is to facilitate the acquisition of tool-using expertise and file-tool pairing expertise; to develop complex, multi-step file-tool workflows; and to record both file trajectories and tool contributions through various essential micro-level preparatory analytic workflows. A second important purpose is to facilitate timely retrieval of appropriate file and application resources.

m³ [em-cubed] is an important extension to the case study in OS-wide customization an Activity monitoring started in FlowSpaces. This Research Project
aims to help users to systematize and automate their collection and organization processes. Attempting to do so for other reader-analysts is a massive undertaking that involves creating the means for recording and refining strategic techno-functional paths through a growing set of tools, not just on a massive timescale as in FlowSpaces, but also in a mezzo-focus, as on the tasks required to process just one single file.

Keep in mind, the current $m^3$ [em-cubed] prototype is only a step on the way toward building a larger, integrated digital work environment whose ultimate purpose is to support online analytical reading and multi-level classification of native digital analytical media content. Naturally, understanding the tasks and tool-usage patterns that users must use in order to get to that point of readiness for analysis could, upon reflection and optimization of those usage paths, facilitate actually getting to that point more quickly.

Based on observations of Zipf’s law in action with email contacts and in other digital contexts, (William E M Lowe 2003) I have hypothesized that typically users use a small number of tools very frequently — and most tools not at all. This fact is clearly born out in the FlowSpaces data produced by a single deep case study. I suggest that a similar usage frequency distributions are likely to apply to files — though the pool of frequently used files is likely to be far larger, and their usage frequencies far lower. The FlowSpaces data have also shown this to be the case.

So the problem in a nutshell is this: users have many files, which they want to be able to retrieve, but wont likely retrieve often. They have far fewer applications, but still a great many, and will have to retrieve some of them quite often. The utility of the greatest number of files and applications is in question, therefore, especially if no reasonable means of retrieval exists at the precise moment at which retrieval becomes a priority.

### 7.4.1 - Retrieving Best Files

The problem is somewhat different for files as compared with applications. For files, typical feature-based indices may provide the cues needed for retrieval, but individual keyword based search/matching of contents is hit and miss given the synonym problem. System based features (e.g., file modification time, byte size,) will not usually be expected to facilitate retrieval except in exceptional cases.
Search through manual file system traversal may be successful, but is often costly in terms of time. Even search using assigned keywords, labels, or tags may not retrieve the desired content if the user does not consistently use precisely identical tag names. Thus, a structured tag system should be explored, that offers reasonable and light constraints on tag names in exchange for greater expressivity within protected semantic domains to facilitate retrieval. This is what the current m3 [em-cubed] prototype offers.

Furthermore, m3 [em-cubed] assumes that, as resources for analytical tools, files have a type of natural resource relation with tools, such that tools link files as inputs through tool-enabled Actions to output products of analysis. Maintaining clear awareness of these input-output relations is important in reporting the provenance of analysis (D N Williams and others 2013; Walker and others 2013; Gotz and Zhou 2009; 2008). Files thus have Tool/Activity mediated relations with other files, such that input through an application and output as a converted version could be noted on both input and output files. Files also have relations with the tool which effected the conversion (or other processing) and with the conversion Task, independent of the tool used to accomplish it. If these relationships can be captured as Tags, then the relations that a file has with other files, processing tools, and processing Tasks may support targeted retrieval.

7.4.2 - Retrieving Best Tools

For applications, the challenge of retrieving the right tool at the right moment has different dimensions. Unless the user has hundreds or thousands of apps, usually the process of locating an application is fairly straightforward. But the simplicity of local search for applications is belied by the fact that the experienced user has developed a mental index for their tools, which allows them to associate present states with future goal states, mapped relationally through some tool/task state.

What I propose is that this implicit mapping of tools as means of accomplishing goals is most efficient and expedient rationale for functional classification. Extending this functional class (Dougherty and Keller 1982) to include the objects of these functions may even provide just enough specificity to be diagnostically useful, and possibly even provide all the user needs to differentiate between, for
example, Editors, Readers, or Taggers; as PDF Editors, Photo Editors; Database Readers, RSS Readers, and POS Taggers, File Taggers. Importantly, in the long tail of tool usage, the tool names (e.g., application names) may or may not (Dougherty and Keller 1982) be known to be associated with these functional classes; though it would seem that frequently used applications would be more strongly associated. Alternatively, if you don’t use an application and you don’t know its name (and if you don’t recognize its icon) then you probably don’t know what its good for.

The ‘right’ tool depends on the context; what data must be processed, any limitations of the resources, and what tools are available, and any known limitations or quirks of particular applications. So if you do not know the name of the tool or its functional class, how can you be sure that you are using the best tool? The brittleness of prototypical classification systems noted in Chapter 6, FlowSpaces, raises its head again here. Tool usage patterns risk lock-in to sub-par familiar patterns, when they do not include active search for best options. On the other hand, discovering new tools (or re-discovering unused tools locally) can be a costly and frustrating process of learning the quirks and limitations of new tools. This trade-off must be to find best options while lowering the cost of discovery, since the cost of learning tools may be fixed.

This time, the solution would appear to be to saturate the classification system with all the functions and their natural objects for each application in the user’s collection and to use some reasonable automatic method of sorting among them. This strategy is discussed as Future Work in FlowSpaces. To recap, the UI of an application (e.g., button labels & menu items) is scraped to extract related Tasks and Objects (e.g., Save>File). Many apps share a great many basic application predicate>argument functions; so the set of Task>Objects features for each application can be compared against those of the entire collection of application to find for each application its unique tasks and clusters of tasks.

Thus, when identifying an application’s prototypical Task>Object function, the tool user may select from a ranked list of Task>Object application features, that takes the Task>Object feature space of the entire user’s collection of tools into account. And finally, when searching for an application, the tool user may target a Task>Object feature, to retrieve a ranked list of applications. The user may also target an application, to retrieve a ranked list of Task>Object features. Or the user
can navigate the entire composite Task>Object feature space to locate an application. In each case, search evaluates the entire Task>Object feature space to identify the right tool regardless of the user’s recall of an app name, app icon, or prototype app task.

Taking a step outward from local to mezzo search (before global search), if Task>Object feature spaces could be objectively shared between colleagues or communities of practice, then a user could query a wider ranked consolidated feature space to identify the right tool for the job based on a wider feature space, the assigned rankings and the launch usage patterns of trusted communities.

Finally, considering global search, consolidation of Task>Object feature spaces for a large collection of tools indexed to to operating system could offer users just in time tool discovery and offer tool builders a targeted consumer base.

7.4.3 - Reading on Paper

When reading text on paper, analysts can physically cut out or highlight portions of interest, color code different topics, features, or patterns; write in the margins, or on sticky notes or note cards; and they can use other physical and spatial organizational structures (like shoeboxes; pigeonholes; cork-boards, yarn, and pushpins; sharpies and craft paper; or fountain pen on napkins to doodle or diagram). But working this way with paper is a time-intensive process.

7.4.4 - Reading on Screens

When working with digital texts, analogues to paper methods either 1) do not exist or 2) where they do exist, they seen as are challenging or somehow unpleasant to work with. They often create entirely separate analytical artifacts that are disconnected from both the analytic process and their original data resources, which creates another digital object requiring management. Or, worse, they are embedded within a file and entirely inaccessible unless the file is opened by the right reading software.

Today’s analytical reader ought to be able to identify meaningful patterns in text beyond simple (or even complex!) character-level pattern matching (e.g., keyword search). And they ought to be able to do so in layers on top of existing digital copy.
For example, when reading in an unfamiliar domain, on a first skim through a given text the reader-analyst may just be interested in identifying **key terms** that are **defined** in the text. The analyst’s rationale may be something like this: *any term that the author goes to the trouble of defining must be important.* Perhaps the authors choice to define a term marks some divergence from the ‘common’ usage of the given term, or specifies and selects from among a range of senses precisely which sense is intended. (Thus aiding the novice analyst in paring down polysemous possibilities.)

As the analyst reads, they may note that the author has extended or (provisionally) modified the definition somehow. The analyst should be able to use the same mechanism to add these extra details to the previously identified definition, in effect ‘extending’ or ‘qualifying’ that first ‘definition’ pattern.

Thus, with the user having identified a few different ‘definition’ patterns, the system should be able to use the lower-level features (e.g., character based, syntactic, semantic patterns) of the given stretches of text to abstract rules for identifying ‘patterns for identifying definitions.’ These patterns should then be used to automatically attempt to identify candidates. Definitions found by the tool on behalf of the user can then be vetted and validated by the user.

Thus, even the shallow reading system ought to become a learning system wherein the analyst’s actions begin to teach that system what patterns are of interest.

### 7.4.5 - Analytical Preparation

In Chapter 5, *Flow*, Section 5.3.1, *Preparation Phase*, has detailed the tasks of classification, collection, and search. To review, the purpose of this preparatory analytical phase is to prepare for deeper analysis. This preparation should entail the creation of an organized personal multi-modal analytical media repository that automatically captures any available meta data for important analytical media, as well as any personal analytical judgments made of collected analytical media, so as to leverage evaluative judgments and other capture metadata to reduce the cost of getting analytical media ready to analyze in analytical tools. The problem, then, for analytical tools is determining how best to support these shallow classification and meta-data collection tasks.
7.5 - The Solution

At the highest level, the purpose of this Research Project is to ensure analytical preparation is worth the effort required to Collect and Organize, both file and tool resource collections and to Learn how and when to use tools. Accordingly, m³ [em-cubed] is a system-of-systems prototype that currently supports manual workflow micro-logging for improved file and tool management and retrieval. The long-term goal is to provide a wider integrated analytical reading environment that actively supports this challenging task.

With the m³ [em-cubed] prototype I address the challenges of using Tags, Highlights, and other Annotations, on both file and tool resource collections, at both document container and document content levels. Thus, the m³ [em-cubed] SoS is a proof of concept for showcasing the uses for a customized, extensible open-source tagging infrastructure and lightly controlled taxonomic schema for capturing ‘shallow analytical judgments’ and simple classification of document containers (e.g., files). Moreover, m³ [em-cubed] also offers a corollary rule-based system that drives automation for document filing and facilitates subsequent local media retrieval. Finally, for ‘light’ discourse analysis, or ‘skim’ reading, the m³ [em-cubed] proof of concept presents a customized, extensible color-based content-annotation system for text-based PDF media, which maps color-space to upper-level domain-independent ontology of human context (See Section 8.5.4, CDG+ Deictic Matrix, for a full description of the contextual dimensions of human experience).

7.5.1 - Capture Shallow Analytical Judgments as Annotations

The m³ [em-cubed] prototype proof-of-concept is designed for low cost capture of shallow analytical judgments in the preparatory analysis phase (and in other functional contexts) using a system-of-systems approach to:

1. capture and update procedural workflow states on digital files using Tags
2. capture and update shallow cognitive evaluation, classification, and labelling on digital files using Tags;
3. create a record of Tag assignments, by file, over time,
4. capture and update somewhat deeper cognitive evaluation, classification, and labelling on digital file contents during shallow analytical ‘skim’ reading using conceptually labeled Highlights.

Both tags and highlights are low-cost assignments, though search through Tag or Highlighter Label lists prior to assignment of these annotation types creates a distinctive cognitive load for the classification task. This is considered to be an acceptable tradeoff here (compared to unconstrained tagging) though this cost should nevertheless be minimized. Thus, to reduce load of search through tag or highlighter list prior to assignment, \textit{m\textsuperscript{3}} should design an organization & presentation system for tags and highlights. One nice method is to filter the list in real-time based on character by character keystroke entry.

7.5.2 - Capture Actions and Relations of Files, Tools, and Tasks as Tags

The immediate goal of the \textit{m\textsuperscript{3}} \textit{proof of concept} is to help user acquire and refine their tool and file using expertise. Tagging each tool and file for the Actions it enables while similarly Tagging any recent Relations between files and tools (as evidenced by recent or proposed workflows involving them) will improve the quality of tool & file collections.

The long-term goal of the wider proposed \textit{m\textsuperscript{3}} (multi-modal-meta) system for analytical media management is to give the user a place to keep their connected tool and file collections in a simple, user-directed, adaptive environment that automatically curates the users’ analytical media & tool collections. This environment automates actions based on connections among data files, between data files and tools, and among tools.

At its most basic level, the proposed infrastructure bins three things: ‘connected’ data files (e.g. Is-a-duplicate, is-similar, is-distally-related, and other ‘is-a’ relationships); prototypical workflows for pushing given data files through appropriate tools; and sequences of workflows for converting data files through to a desired format for unrestricted data manipulation.

7.5.3 - Capture Workflow Sequences
In their simplest form, Workflows are sequences of tags applied to files at task assignment and completion points. Thus, a workflow sequence is captured as time stamped stateful tags which can be modified or extended at any time. In \textit{m}^3 \textit{[em-cubed]}, workflows exist to catalog and facilitate all of the file manipulations and conversions that are necessary as preparatory steps towards achieving the ultimate purpose of analysis in analytical tools (like the proposed digital reading environment proposed in Section 7.5.7).

\textbf{7.5.4 - Bin Connected Files}

Document bundles are meant to wrap up data resources from their initial raw digital form through as many conversions/formats as are necessary to produce a version of the data that the user may then manipulate, annotate, and analyze using analytical tools.

Take, for example, the conversion of an out-of-print public-domain paper-printed book to enriched, analyzable, and annotatable digital copy. The proposed data wrapper will bin together raw scanned .TIFF files; post-processed versions that remove binding and edges and split facing into single pages; aggregated into an image-only .PDF file; OCRed, corrected, & archived as an .OPD file; exported into .TXT, .RTF, .DOC, or various text-based .PDF file; post-processed to normalize layout & formatting into .XML (content) and .XSLT (normalized XML style sheet) files; and pre-processed through various natural language processing steps (POS-tagging, shallow parsing, discourse element chunking) that enrich the basic XML or JSON data structure; and finally internet-searched, parsed, and matched for bibliographic metadata in .ENL, .RIS, or .BIB formats.

In a traditional file system each of the files listed above is considered a separate entity. A single analog document may consequently have \textit{hundreds} of digital files, including versions and differently sized chunks (e.g., chapters, pages). Traditional file management systems offer a ‘folder hierarchy’ as the grouping metaphor for organizing disparate files that ‘belong’ together. However, managing what amounts to different interim versions, properties, and ‘chunks’ of the same data resource embodied in multiple digital files of different formats using only the traditionally available means and metaphors quickly becomes cumbersome and unwieldy to manage manually through a long conversion process like the one detailed above.
Even in more concise collection workflows — like that exemplified by Papers\textsuperscript{3} for collecting Journal articles from online repositories — the traditional folder system fails to automate or accommodate file versioning. While internal file changes (like PDF annotations or highlights) can be saved as modifications within the original PDF file, if the original file requires conversion (say from image-only PDF or Acrobat-OCRRed PDF to a properly OCRRed format), then both the conversion process and the files generated by it must be managed externally (e.g. external to both the Papers application and to the original file’s location in the systems folder hierarchy).

Therefore, there exists a real need for collecting and unifying multiple digital embodiments of a single analog media resource or document. Existing file de-duplication tools do not consider duplicate content in different file formats. Existing file management tools do not handle files with different content modalities and the same file name & file type/extension.

### 7.5.5 - Build towards an integrated Analytical Media Management and Analytical Reading Environment

Existing Visual Analytics tools help carve through large collections, primarily using statistical methods from information retrieval to systematically and visually identify a specific subset of documents the analytical reader is likely to be interested in. But VA tools typically leave the next step - reading through the text - completely up to the analyst.

Reading is a manual process that occurs in linear real time. This ‘reading step’ is so time-consuming that it is usually presumed that reducing the breadth of the document collection is the only point of leverage to redress this bottleneck in analyst capacity. The strategy can by summed up as follows: by focusing in and avoiding ‘peripheral’ content; most VA approaches assume analysts can still afford to read deeply in ‘key’ areas. However, it is not clear that existing VA tools are keyed to the serendipitous discovery, insight generation, and capture of these ‘online discovery & classification’ moments while reading.\textsuperscript{30}

Ultimately, Chapter 9, TextWorlds+, Section 9.7, Future Work, describes how the ideal analytical reading environment might leverage the automatically
identified syntactic, stylistic, and rhetorical structure of a text. The system offers conceptual scaffolding, within an enriched and augmented analysis environment that is tuned to the analytical reading moment. But a lower level of engagement with textual contents while ‘skim’ reading also requires envisioning of new tools, some of which are easily achieved by incremental improvements to existing tools, as has been established by the prototype

7.6 - The Prototype

This prototype system is designed to work with any analytical tools, or more accurately between and behind them. The $m^3$ [em-cubed] tagger provides a common storage point for simple Tag annotations on all data files of any type using any analytical tool, at any point in the analytical workflow.

7.6.1 - tagger

I propose that simple classifications can be improved by integrating ubiquitous local file-tagging at levels ranging from the whole digital media container down through its detailed contents.

This proposal is supported by the explosive literature on the semantic web (Uschold 2003; Mika and others 2004; Atzenbeck and Nürnberg 2004; Enhong Chen and Wu 2005; Katifori and others 2007; Kammergruber and Ehms 2010), which maintains that individual-focused tagging schemes may be improved by exposure to social tagging efforts; and that such folksonomies can be improved by layering on thesaurus-like relations; which can be improved by developing ontology-like inference among categories (Weller 2007).

When considering how best to improve simple classification for analysts, ascending this continuum of semantic technologies is advised, though the ‘ideal’ schema is not necessarily the most formal one. In the approach used here, light restrictions are placed upon action words, to keep identifiable conceptual facets distinct. The implications of this design choice are explored in the following extended example.

In this part of the $m^3$ [em-cubed] prototype SoS, the precise workflow actions and sequences of actions are recorded as metadata on the file itself. All files are
acquired and processed in de-facto workflows which may be either a standard or customized procedural configuration. Thus, to capture workflow states, unique state tags simply use the past tense of the Activity verb for completed workflow task states (e.g., copied), and the bare infinitive form (e.g., ‘copy!’ with no ‘to-’ in front) with a unique semantically loaded terminating character (here the bang ‘!’ is used to indicated a directive towards an action).

This combined lexicographical and typographical distinction underlines an important conceptual distinction that, if followed, will allow the user to keep different facets of a single concept distinguishable yet retrievable from the same search. Thus, as files progress through a workflow and are processed by various tools, their prospective to-do state can be updated as completed, by automatically replacing ‘do!’ tag with ‘done.’

This simple difference, of selecting a different verb form to denote future or past actions, combined with the addition of a single non-standard character (e.g., the bang ‘!’) is sufficient to disambiguate most verb/noun ambiguities. For example, if I want to duplicate a file, for efficiency I would probably choose the shorter term ‘copy’ as my root action. By terminating the action-item with a bang character ‘copy!’, I not only leverage cultural and lexicographical standards whereby commands are emphasized with the exclamation point, but I also successfully differentiate the copy of a letter sitting on my desktop, which also happens to have the ‘copy’ tag applied, but with respect to its features and not its requirement-to-process. Thus, action is known as ‘action!’ , while the action (e.g., mechanism) is semantically preserved. Thus, m³ [em-cubed] provides range of other evocative single character prefixes are selected to preserve semantic spaces within the undifferentiated pool of tags.

Similarly, other workflow states are distinguished using unique semantically loaded single characters. For example, the user could indicate export from an application ‘←Word’, or export from an application to another application ‘Acrobat←Word’ etc.

By recording procedures in this fashion, a file’s origins, duplicates, conversions, siblings (duplicates of a different file type) or processing steps are all always available (e.g. stored as metadata on the file itself). Detailed tracking information - of file progress through a workflow, and of files ‘related-to’ files (e.g. Duplicates in
other formats) is retained and this metadata persists independent of file location (and ideally persists even for duplicated files.) This metadata is accessible from the command line and is with some duplication is also visible in the file browser UI (e.g. A ‘double-entry’ system must be set up to copy this metadata into the Spotlight Comments field).

Nevertheless, analysts must frequently negotiate established external classification systems. This fact may be the single best argument for facilitating development of personalized indexing systems that are integrated with official, standardized indexing systems. As the analyst’s digital work proceeds it frequently entails interfacing with existing official taxonomies, ontologies, and similarly ordered systems, so there is a pressing need to align ‘free-form’ personalized tagging with these existing structured and semantic resources. While adopting someone else’s classification system may not be desirable or feasible, drawing upon officially classified resources can

The challenge of organizing ongoing collections is familiar to scholars in the fields of information, archival and library sciences. Many commercial and open-source software packages exist to organize both digital and analog collections - but these typically focus on single media types. The long-term goal is thus to extend the m3 [em-cubed] prototype into a management application for analytical media with the management strategy and aesthetic of iTunes or Papers.

7.6.2 - leveler

In the best meta + media management tools available to date, most metadata exist in databases that are not explicitly linked to a digital file. However, ‘in the wild’ certain metadata also exists embedded inside a file. And with the right tools, like the m3 [em-cubed] prototype system-of-systems described here, users can also layer metadata on top of a file by extending the file system’s XATTR file metadata infrastructure to enable custom tagging. Thus, the options for metadata location are inside, outside, and beside (or on top).

Thus, enabling systematic coherence between file metadata and file content will ideally involve creating multiple content mark-up layers on the best available digital version. In m3 [em-cubed], for text documents, the best available digital
version means the version of the document that is the closest to the original, whose
textual contents are selectable, whose layout & style reflects the original, and of the
smallest size in bytes.

So the method proposed this part of \( m^3 \) [em-cubed] begins with the retrieval of
published media metadata from institutional and commercial repositories and
various social clouds. It proceeds to an examination of the digital files to extract
and annotate features of both the digital container and digital contents and then to
embed this metadata into the file where possible. Finally, with both the generic
institutional metadata and the particular details of the specific digital embodiment
captured as personally applied annotation metadata, multi-layered annotation of
the textual content within the document container may begin.

Organization of document collections requires ongoing conceptualization (and
capture) of document, file, & content features. For example, the preparation of
documents for digital annotation occasionally necessitates file conversions — and
each file preparation workflow begins by identifying the current state of the digital
document. The proposed analytical media manager would capture these
conceptual assessments and even automatically facilitate certain conversion
processes.

For example, if a file arrives as an image-only PDF, several potential workflows
become options, depending on the preferred target file type. Once one or more of
these workflows are executed, the original file has been through (at least) one
workflow and annotations have recorded each action taken. Execution of a range of
possible conversion destinations create a range of sibling duplicate files, most often
in a variety of different file formats; but some in multiple files that share a single file
format and have their contents accessible in a different modality (e.g. Like an
image-only PDF, an OCR'd text-overlay-on-image PDF, and a text-only PDF).

7.6.3 - binder

In a digital environment the cost of creating copies is low and, consequently,
duplicate files abound. Considering time costs alone, digital to digital conversion
costs are significantly lower than for analog to analog conversion, whereas digital
to analog conversion time costs remain high.
However, the full cost of copies must consider downstream costs in file management; and particularly the efforts taken for file de-duplication. At both personal and enterprise levels, file archiving and secure backups have a time, disk, and dollar cost, so file de-duplication can reduce the cost of maintaining backups.

I propose a novel (in using this term I mean, by collecting together the following features, the resulting composite of recognizable separate features will be something new and emergent) single-source multi-format file binder as a file-management paradigm that will centralize analytic media resources for individual media entities with multiple digital instances

- in multiple formats (e.g. TIF, DOC, DOCX), or
- in single formats with various content modalities (e.g. PDF image, PDF ClearScan OCR, PDF layered OCR, PDF text), or
- with multiple related stand-alone sub-entities (e.g. Months of Data, Songs on an Album, Chapters in a Book, Articles in a Journal, Clips from a Video).
- transitional states indicated by drafts or schematic data transformations.

Thus, a single binder will collect all container types (e.g., computer file types) for a single source document; all content types (e.g., a scanned pdf, and an OCRed pdf) within a given container type; and will organize all structural content divisions and facilitate incremental content transformations and conversions. An intelligent agent will match content requests from applications to the file-handling capacities of the requesting tool to automatically select the best existing container-content-type combination, or initiate automated conversion to generate better container-content-types. This analytic media management approach will create a default collection point for incoming media, a natural starting point for local search, and reduce local indexing congestion (where multiple versions of files appear multiple times in local search results).

### 7.7 - Future Work

At this point, the prototype m³ proof-of-concept captures data by manually assigned tags applied by a thoughtful analyst. However, analysts miss files, and don’t always tag, so automated tagging of files as they proceed through workflows (Voida and Mynatt 2009; Voida 2008) should be explored. For example,
whenever a “Save as…”, “Export”, or “Import” menu option is selected (in any application). Tags indicating both the original file type (e.g. DOC), the destination file type (e.g. RTF →), and the current App could be automatically added.

Visualization of the manually tagged workflow state data could reveal connections between files that are not obvious from the file system alone. File de-duplication or identification of ‘preferred-versions’ could be facilitated using this data.

These tags could thus be useful to the downstream task of ‘binning’ or binding all file siblings (duplicates of a different file type, or of the same file type with the same content yet a different content contextually) together. The proposed leveler and binder functions of the analytical media manager both remain as goals for future work.

7.7.1 - Recommendations

Reflection upon the current capabilities of proof-of-concept prototypes have suggested future possibilities and paths for improved developed. These lessons learned can be distilled into Recommendations for four high level features that all analytical tools should consider implementing, as well as suggesting areas for a new class of analytical-preparation tools. Finally, specific improvements and extensions to the m³ [em-cubed] prototype are considered.

Analytical tool design should internally

1. Prompt users toward the best available procedures and data.
2. Optimize the organization of tools and files by Activity.
3. Highlight the relevant capabilities of tools.
4. Foreground the salient features of data resources.

Consider analytical tool designs that provide external meta-mechanisms to

1. Highlight tool capabilities and file properties.
2. Help users find best options while lowering the cost of discovery.
3. See if sharing quirks lowers the cost of learning new tools.
4. Capture, analyze and optimize tasks and tool-usage patterns in analytical preparation to facilitate quicker entry to deeper analysis.
5. Minimize the tradeoff costs between search through Tag or Highlighter Label lists prior to assignment of these annotation types and the benefit of employing a lightly constrained vocabulary in Annotations.

6. Improve designs of visual presentation & organization systems for tags and highlights.

**7.7.2 - Experiment on Analysts preferences**

The explorations of the m³ [em-cubed] prototype has revealed a number of assumptions which could be framed as hypotheses and tested in small experimental studies to:

1. Test Dougherty’s prediction that analysts won’t use linguistic Name classes or even Function classes (e.g., Tasks) as their primary organizing principle.

2. Evaluate experts’ recognition of the value of organized collections, and whether experts benefit from reminders or automatic assistance with maintaining organization of their collections.

3. Evaluate the hypothesis that the quality of collections will vary with degree of expertise.

4. Evaluate the hypothesis that the file’s tagged relations with other files, with processing tools, and with processing Tasks can support targeted retrieval.

5. Evaluate if sharing quirks lowers the cost of learning new tools.

**7.7.3 - Build Smart Color Picker**

Build a custom OS X color picker that automatically assigns incrementally different colors to children Concepts in an Annotation Concept hierarchy. Thus, if a new Concept is created in the Classification Hierarchy, it will take its color automatically based on its parents and siblings. So, for each new concept label, the smart color picker Auto-selects Colors based on the Labels Location in the Color Picker's Hierarchy.

**7.7.4 - Build Highlight Linker**

Extend an Open-Source PDF viewer (Skim (Hoffman and Millitello 2009)) that supports a range of within-file Annotations, build a directional annotation linker.
This will allow the user to assign visual, labeled, directional links between Annotations. Ultimately, this will permit assignment of Rhetorical Structure Relations to the chunks identified in shallow content evaluation during the skimming phase of analytical reading.

7.7.5 - Build Auto Highlighting

Use features of manually identified chunks to generate rules for automatic identification of further chunks. For example, this would enable automated identification of ‘stretches that look like definitions’ from textual and linguistic cues derived from textual analysis of stretches the user has previously identified as definitions. This is similar to Stoffel’s automated structure identification, except the target classes are user-generated, and not simply document-formatting based.

7.8 - m³ [em-cubed] Synopsis

The current m³ [em-cubed] prototype was designed to inexpensively capture shallow analysis in preparatory data analysis over time, using a system-of-systems approach to:

1. capture and update procedural workflow states on digital files using tags
2. capture and update shallow cognitive evaluation, classification, and labelling on digital files using tags;
3. create a record of tag assignments, by file, over time,
4. capture and update deeper cognitive evaluation, classification, and labelling on digital file contents using highlights.

Both tags and highlights are low-cost assignments, though search through (tag or highlighter) lists prior to assignment of either creates a cognitive load for the classification task. Thus, to reduce load of search through tag or highlighter list prior to assignment, m³ [em-cubed] should design an organization & presentation system for tags and highlights.

This Chapter also proposes and works toward a larger system for (multi-modal-meta) analytical media management system that is proposed in Future Work. The first goal of the larger proposed system is to eventually provide automated
management and binning of multi-format analytical media resources, by identifying and collecting duplicates and nearly identical versions having different container file types or different content modalities. Binned files can better leverage existing metadata to automate organizational tasks like sorting, naming, and filing. Creating single source bins and collecting duplicates and versions of digital files should reduce noise in local search and facilitate local retrieval workflows.

The second goal is to capture and improve structured external and internal metadata resources for analytical media, which includes providing smooth import export trajectories among various complimentary and competing media and metadata management applications. This is primarily important for keeping tools that manage only metadata synchronized with those that also manage files. The recommended move towards integrated meta + media managers should reduce any needs for leveling between various management tools, which should provide smooth import-export trajectories internally without the need for external customized database synchronizing middleware.

The third goal provides a tagging interface for the missing preparatory analysis phase to enable simple classification simultaneously along any relevant feature or functional dimensions on any given multi-modal analytical media resource.

The fourth goal provides an improved annotation interface for the missing preparatory analytic reading phase, to enable shallow visual classification of salient semantic, rhetorical, or stylistic function of selected sections of a text with highlighting and to enable the identification deeper relations between visually classified or highlighted portions of text.

The final goal leverages natural language preprocessing of content text structure as well as several available lexical and semantic resources in user-driven annotation of the texts’ conceptual contents. This deeper content analysis is developed more fully in Chapter 9, TextWorlds+. 
Part IV: Modelling Conceptual Structures in Deep Analytical Reading

To review, Part I, Foundations, established the conceptual background for this research. Chapter 2, The Problem Situation elaborated and situated four related dissertation research projects in relation to specific levels of the problem situation. Chapter 3, The Cognition Literature, clarified the role of specific areas of higher-level cognition in the general process of analysis and analytical reading. Chapter 4, The Analysis Literature, offered a curated introduction for analysts and analytical tool-builders to various important analytical approaches, to identify the tasks involved in the analysis process in general and text analysis in particular.

Part II, Modelling Cognitive-Functional Analytical Process, began by drawing upon the author’s practical experience as an analyst using analytical tools, including those offered by Visual Analytics (VA), and the Literatures on pragmatic analytical procedures and higher level cognitive processes. Both were found to agree that certain important cognitive and functional tasks are missing from the standard Sense-Making model of analysis that is widely cited in VA. Thus, the two Chapters in this Part of the Dissertation make the argument that the use of the Sense-Making Model as a specification of the analysis process for the design of analytical tools provides one explanation for certain gaps in analytical functions offered by VA tools today. Empirical observation of functional flow in tool usage may help to begin to address those gaps.

In the foregoing Part III, Supporting Ubiquitous Shallow Analytical Judgment, the discussion moved in focus from the highest level to the intermediate level, where certain focal Activity tasks and many cross-cutting preparatory analytical tasks are under-supported in current analytical tools. The previous Chapter 7, m³ [em-cubed], offered a prototype system for capturing cross-cutting shallow analytical judgments as metadata in the form of tags and highlights, regardless of the focal Activity or current Application in use. This builds an external repository of judgments that is not locked into the internal annotation format of any given analytical tool. In Chapter 7, the focus moved downward and inward from overall digital tool usage workflow, to analytical tool-usage workflows, and to specifically
focus on capturing the products of cross-cutting shallow analytical judgments during preparatory analysis.

The previous Chapter 7 showed how user conceptualization and workflow state can be captured in tags and other low-cost annotations. Products of the shallow classification tasks of identifying and labeling features of the analytical media (e.g., judgments) can be captured as Tags applied to the document container (in the file system), and can subsequently be used as triggers to automate cascades of workflow processes. Strategies for managing a growing collection of annotations were proposed. Then, also, the somewhat deeper process (though still of an intermediate depth) of shallow or skim-reading can also begin to identify ‘chunks’ of text as internal features of the analytical media, which can be captured as Highlights applied within the document container at the level of the document contents. These annotations can be stored within the file as modifications to the file; as external, separate metadata files; or just as in the case of Tags, as file metadata stored in the file system. Standard Open Source Annotation application and storage mechanisms can be customized to address management challenges for both Tags and Highlights. Related strategies are applied to create manageable and semantically meaningful subgroups for both Annotation types.

Taking a human scale view of the analysis process problem space naturally affords a multi-level focus that moves from understanding the whole, to its widest parts, and then to its deepest parts, keeping the human analyst reader in focus throughout. So in Part IV, Modelling Conceptual Structures in Deep Analytical Reading, the focus moves to the deepest level of language comprehension during analytical reading. The broadest contributions at this deepest level include the refocusing of discourse analytical methods to precisely mirror the dynamics of discourse processing models in order to externalize the mental ‘situation models’ created internally while reading and produce interactive visualizations of the conceptual structures of text. All of this is attempted in order to fix the gaps in visual analytic tools’ support for analytical reading.

Specific novel Contributions to CDG made here include:

1) introducing general textual analysts, analytical tool-builders, and discourse-curious linguists to CDG+,
2) the repair and extension of CDG as CDG+;
3) the positioning of CDG+ as a bona fide Discourse Processing Model;
4) the specification of the CDG+ Context Model;
5) the specification of the CDG+ Meta-Contextual Frame to facilitate reader classification of terms;
6) an introduction to CDG+ Manifest Semantic Analysis;
7) an introduction to CDG+ Personal Ground;
8) the specification of the CDG+ Task Model of Analytical Reading.

These Chapters describe a single process and its product twice. The first time, in Chapter 8, CDG+, defines how the process of discourse comprehension unfolds regardless of a person’s particular visualization goals or degree of self awareness or self reflection. The second time, in Chapter 9, TextWorlds+, explains the novel method used for developing a mindful meta-cognitive awareness of the process that will permit the analyst to derive an externalized proxy of the internal mental models that they produce while reading.
Chapter 8 - CDG+

This Chapter begins by reviewing progress through this dissertation to this point. Part I, Foundations, has presented the background and foundations of the two core challenges of Cognitive Limits of human analysts, and Functional Limits of analytical tools, which together define the analytical process problem space. Chapter 1, , introduces this problem space and Chapter 2, The Problem Situation, elaborates and situates four related dissertation research projects in relation to specific levels of the problem situation. Chapter 3, The Cognition Literature, and Chapter 4, The Analysis Literature, introduce the domains from cognitive science and text-focused analysis that have contributed to the discovery of the way forward toward the solutions proposed for this problem space by each research project.

Part II, Modelling Cognitive-Functional Analytical Process, began by drawing upon 1) my practical experience as an analyst using analytical tools, including those offered by Visual Analytics (VA), and 2) the Literatures on pragmatic analytical procedures and higher level cognitive processes. Important cognitive and functional tasks are missing from the standard Sense-Making model of analysis that is widely cited in VA. The use in VA of this model as a specification for analytical tool design could thus explain certain clear gaps in analytical functions offered by VA tools today.

Thus, in Chapter 5, Flow, (Section 4.5.2) I addressed those omissions in the Sense-Making model of analysis by extending Pirolli & Card’s model into the Cognitive-Functional Flow Model of General Analysis. The Flow model additionally reconceptualizes the components of the analysis process such that each cognitive and functional task is conceived as a network of overlapping requisite and related cognitive and functional sub-tasks. Thus, when a task occurs, its requisite sub-tasks must occur, in some order, and its related sub-tasks may occur, or may be skipped entirely. The ‘flow’ dynamic accounts for this smearing of boundaries, or simultaneous overlapping among tasks, their requisite sub-tasks, and related or embedded tasks. Elsewhere this phenomenon has been called task interleaving (Jin and Dabbish 2009; Bannon and others 1983).

In the previous Chapter 6, FlowSpaces, I have looked for ways to take a more precise yet more malleable view on functional transitions between digital Tasks and
Tools by focal Activity (Ashley Williams 2009; Norman 2006; Gay and Hembrooke 2004; Christensen and Bardram 2002; Bannon and others 1983). My approach addresses the inherent brittleness of a prototype functional classification system (Dougherty and Keller 1982). So beyond classifying Applications according to the Prototypical Task that they enable (Bardram 2006), FlowSpaces secondarily classifies each application usage by its current context of Activity focus, and allows easy updates to the focal Activity class by simply dragging and dropping Application Windows between virtual Desktop Spaces.

This move recognizes two important facts. First, as is shown in the single case studied deeply here, most often tools are used in their prototypical task in their default Activity context. But, secondly, a dynamic flow between tools from several different task categories is often required to accomplish a single focal Activity. When the user maintains focus on one Activity, even switches between Tools and Tasks can be productive. So within-Activity task and tool transitions are often essential, and not just interruptions as is widely discussed in the literature (Garrett and Danziger 2007; Czerwinski and Horvitz 2002; Czerwinski and others 2000). FlowSpaces data can be used to identify intentional, productive, Activity-focused Tool Task alternation sequences.

The previous Chapter 7, m$^3$ [em-cubed], pushes the focus downward and inward from overall digital tool usage workflows, to analytical tool workflows and to specifically focus on preparatory analysis. This Chapter shows how the products of cross-cutting shallow preparatory analytical tasks can be captured in tags and other low-cost annotations. Specifically, m$^3$ [em-cubed] also shows how tags can be used to capture results from the process of identifying and labeling features of the analytical media on the document container, and can be used as triggers to automate cascades of workflow processes.

It should be clear, to this point, that this dissertation is chiefly concerned with identifying and resolving gaps in the support of analytical tasks by analytical tools today. So far, each Chapter has offered a progressive focusing from the broadest level of the entire analysis process to intermediate level preparatory analysis. At each level, the twin lenses of cognitive (internal) and functional (external) perspectives have considered the Tasks, Activities, Tools, Resources, and Products
that are each important parts of the analysis process, as specified by the cognitive-
functional Flow Model of general analysis.

Ultimately, however, the Flow Model is not complete. It requires a detailed,
empirically based, cognitive-functional Task Model for each analytical task in the
model, so that one day, better analytical tools will be built on a solid practical and
theoretical basis. By selecting analytical reading from those analytical tasks in the
unacknowledged, unsupported gap between current and future analytical tool
functionality, we now move the level of inquiry to the deepest level of the
execution of a specific and challenging analytical task.

Thus, the general purpose of this Chapter 8, CDG+, is to consider both this
under-supported analytical reading task and possibilities for improving support of
this task in analytical tools. To do so, however, from the perspective of Visual
Analytics, I have had to go ‘out of field’ to look at established models of language
comprehension from established domains of enquiry beyond where the general
analyst or analytical tool maker is likely to be familiar.

This Chapter presents only a meta introduction to the original Cognitive
Discourse Grammar approach, since it only introduces an outline of the approach,
and is intended to provide reasons for the reader to take a closer look. This is
simply a meta introduction for several important reasons. The first is that CDG is a
large framework. There is simply not sufficient space available to consider each of
the mechanisms and all of the dynamics of this framework. The second reason is
that CDG is a massively integrative approach, a system-of-theoretical-systems, if
you will. Simply meeting standard academic expectations for proper conceptual
attribution alone would occupy hundreds of pages; not to mention the specification
of each relevant theoretical element, mechanism, or concept from the approaches
CDG has integrated, and exactly how each is employed, critiqued, and modified in
CDG. The third reason is that CDG interleaves its own novel contributions
throughout the integrated theories. Thus, any specification of CDG’s original
theoretical, methodological, or mechano-dynamical innovations, made without the
extensive attribution of its wide and varied theoretical bases, could thus become
subject to the criticisms of naiveté, whereby common and widely researched
linguistic topics might seem to be naively and incorrectly attributed to the
progenitor of the CDG approach. But this is most assuredly not the case. The
original CDG both acknowledges a wide swath of literature in range of domains and contributes to them broadly. This Chapter begins with a condensed summary of how several common and widely researched linguistic topics have been integrated into the CDG framework.

CDG is a detailed cognitive and functional approach to externalizing the mental models produced in the process of discourse comprehension. CDG integrates both domain expertise and cultural knowledge into the process of determining linguistic meaning. It further integrates several varieties of inference cued by concepts explicitly referred to in the text. It explicitly prioritizes the production of visualizations of the internal representations of the conceptual structures the readers generate internally while reading. It is a detailed and layered approach that uniquely integrates more than 30 major areas and mechanisms of cognitive and linguistic phenomena.

For example, CDG integrates a thorough treatment of context (Emmott 1998; Kintsch 1998; Lenat 1998; Emmott 1997; Lyons 1995; Verdonk and J.J. Weber 1995; Gerrig and Murphy 1992; Alba and others 1981; Lyons 1981; Van Dijk 1977) with scene recognition (Rensink 2000; Vermersch P 1999; Fauconnier and Turner 1998; Mandler 1992; Ankenbrandt and others 1990; Landauer 1986; Rayner and others 1978; Rayner 1978; Pavlidis 1977; Simon 1975; Grice 1957) and event segmentation (Thompson and Radvansky 2015; Magliano and others 2014; Ribarsky and others 2012; Kurby and Zacks 2008; Zacks and Swallow 2007; Zacks and others 2007; Hard and others 2006; Speer and Zacks 2005; Barbara Tversky and others 2004; Zacks and Barbara Tversky 2001; Zacks and others 2001; Bryant and Barbara Tversky 1999) from embodied cognition (Chow and others 2014; Malafouris 2013; Feldman 2008; Lakoff and Johnson 2006; Wilson 2002; Emmott 1999b; 1999a; Glenberg and others 1999; Madhavan and Grover 1998; Rosch and others 1992).

CDG integrates the extension from Leibnitz’ Best Possible Worlds (‘Essais De Théodicée Sur La Bonté De Dieu, La Liberté De L'homme Et L’origine Du Mal’ 1747) to Lewis’ Possible Worlds (Richard Lewis 1996; David Kellogg Lewis 1986; 1973; David K Lewis 1968) and Kripke’s Modal Logics (Kripke 1963b; 1963a; 1959) in formal semantics with Eco’s Literary Worlds (Eco 1984) to Werth’s conception of Rich Worlds (Werth 1999f; 1995a).
CDG integrates Wertheimer’s **Figure** and **Ground** (Wertheimer 1922) from *Gestalt* psychology with the Johnson-Laird’s **Mental Models** (Johnson-Laird 1983) from reasoning and decision making, and Fauconnier’s **Mental Spaces** (Fauconnier and Sweetser 1996; Fauconnier 1985) from cognitive linguistics, together with Kintsch & van Dijk’s **Situation Models** (Kintsch and Van Dijk 1978) from discourse processing.


Thus, as an integrative approach, CDG owes a great debt to a great many language theorists. Yet CDG remains unique in its combination and contribution to modelling these established cognitive and linguistic phenomena, especially towards the ends of producing visualizations of the internal representations of conceptual structures readers generate internally while reading. The CDG approach is richly specified, but also richly productive.

The fourth and final reason for offering only a meta introduction is that CDG is neither perfect nor complete. This Chapter therefore identifies and addresses shortcomings in the original CDG approach, and formulates the extended and revised version of CDG, introduced here as CDG+. In general, its scope and complexity are real challenges to its comprehension and use. Several addressable concerns regarding logical consistency (e.g., in the selected ontology for modals) and word choice (e.g., ambiguous overuse of the terms ‘modal’ ‘modals’ ‘modality’) can only be properly addressed in a thorough introduction. The identification of certain gaps or oversights, and identification of where complimentary approaches have produced portable solutions that could be easily integrated into CDG (e.g., integration of RST analysis (Mann and Sandra Thompson 1988; Taboada 2006) to
identify inter-propositional relations) only really become obvious in the context of the original approach.

So the problems with which my Research Programme has grappled with and found traction for regard a general void of theoretical extensions to CDG, the under specification of a stepwise procedure for the discourse analytical method of Text World analysis, the underspecified starting point and static modality of Text World diagrams, the absence of any infrastructure attempting to capture the evaluations and truth value judgments of users. Thus, in this Chapter 8, CDG+, I have focused on providing a visual introduction to CDG, designing and using models never considered in the original CDG approach, which therefore constitute a practical and theoretical extension to CDG.

Thus for many reasons, CDG cannot be fully detailed here. Nevertheless, this overview provides a considered selection of both the major concepts, the signature mechanisms, and the dynamics of the original CDG approach. Recall that the Flow Model of general analysis still requires a detailed cognitive-functional specification for analytical reading, particularly identifying the requisite sub-task network required to do analytical reading.

No such Task Model exists ready-made, in CDG, but the requisite sub-tasks of the analytical reading process are specified directly in CDG. Thus, CDG/+ can be mined to produce a specification of the requisite sub-tasks of analytical reading. CDG+ takes on this task, and works to provide a detailed cognitive and functional specification for the subtasks involved in analytical reading in the form of the CDG+ Task Model of Analytical Reading.

For these reasons, each of which is plausible enough on its own, in this Chapter I will take the time to introduce analytical tool makers, including those in Visual Analytics, and general analytical tool-users to various reasons that they should look more closely into CDG/+. Both analysts and tool-makers may be surprised at the complexity of the ‘simple’ task of reading.
The following Chapter 9, TextWorlds+, will serve to discuss and evaluate the Solutions introduced in this Chapter in the form of the extended TextWorlds+ Analysis Method. Readers interested in working through detailed exemplars of the manual TextWorlds method are referred first to the next Chapter 9, TextWorlds+, and also to the many excellent exemplars provided by Werth (1999) or Gavins (2007). Readers interested in a term-based glossary of CDG+ Discourse Processing Model features, and of standard linguistic terminology used in CDG, are referred to Appendix V. Glossary. Those readers interested in an exhaustive identification of each lexicographical cue that TextWorlds claims indicates a role in building the text world diagrams or in the dynamics of the discourse processing model’s machinery may refer to method of analysis and Appendix B. Textual Cues required to facilitate the automation of TextWorlds Method.

![Figure 8.1.A. These most common words, or stop-words, are usually removed from vector space representations of document word frequency. However, many of these words serve an important role of indicating the deictic attachment of a term (determiners) or the semantic connections between terms. B. The top 300 words used in this Chapter.](image)

8.1 - Overview

This Chapter begins by reviewing the specific purpose of this Research Project, which is to improve upon manual methods of Modelling the Conceptual Structures produced during Deep Analytical Reading drawn from a cognitive approach to Discourse Analysis, and to work towards building computational methods of the same. The overall goal in this Chapter is to establish the utility, problems, and
solutions presented by the original **Cognitive Discourse Grammar** and to specify a range of specific contributions to the CDG approach. Thus, the next Section 8.2, **Previous Work**, explains the origins and current applications of CDG. This Section distinguishes between the original CDG approach and specific revisions and extensions to the original approach offered here as CDG⁺.

The following Section 8.3, **The Concepts**, introduces the original CDG approach, and then some of the main concepts integrated in CDG⁺ and indicates their originating domains. The subsequent Section 8.4, **The Problems**, explains the original problem situation that spurred the creation of CDG. It then goes on to consider shortcomings in the original formulation of CDG, in the subsequent stylistics versions of Text World Theory, and then moves on to specify several large problems that are resolved in CDG⁺. The next Section, 8.5, **The Solutions**, outline each major revision and extension I have made to the original Cognitive Discourse Grammar approach. Finally, Section 8.6, **CDG⁺ Synopsis** reviews the Research Project described in this Chapter.

The following two Sections elaborate upon the need establish a closer connection between **Discourse Analysis**, **Discourse Comprehension**, and **Visual Analytics**. First, Section 8.1.1, **Analytics and Comprehension**, situates the two primary approaches to analysis discussed here, **Discourse Analysis** and **Visual Analytics**, with reference to language understanding in general and discourse processing models in particular. Finally, Section 8.1.2, **Analytics and Discourse**, discusses the relationship between both analytical traditions and what their approaches may have to offer each other, beyond a simple trade in trending metaphors.

### 8.1.1 - Analytics and Comprehension

This Section attempts to establish a closer connection between **Discourse Analysis**, **discourse comprehension**, and **Visual Analytics**. For the purposes of my arguments here, I will begin by reviewing my definition of three domains. **Discourse Analysis** is a linguistic field that is focused on the meta-cognitive task of identifying, reflecting on, and theorizing the components, relations, and functions of discourse. **Discourse Comprehension** or **discourse processing** is a field of psychology that is focused on creating models of online language processing and
experimenting to evaluate the predictions and performance of the models in experimental settings. Visual Analytics is a software systems and visualization field that is focused on the production of visually interactive analytical tools.

The core argument of this Chapter is twofold. First, Discourse Analysis can be improved by examining not only what discourse is, its parts, and its functions, but by patterning the method of analysis after the process of Discourse Comprehension. Analyzing discourse using such methods will create models of the parts, functions, and productive dynamics of human language cognition. Thus, Discourse Analysis using analytical method patterned after Discourse Comprehension gets the analyst closer to externalizing the mental situation models created as internal representations of conceptual contents of discourse.

Second, Visual Analytics’ support of textual analysis can be improved by understanding what analysts must do as they read, and by supporting the range of analytical reading sub-tasks after the process of Discourse Comprehension. Thus, in order to be prepared to follow Heuer’s longstanding invocation to build tools that will help analysts to externalize both their mental models and any creative or updating processes required to maintain them (Heuer 1999), analytical tool-builders both within and beyond the field of Visual Analytics need to discover and come to understand a model of discourse processing that is both empirically plausible (Forsythe and Xavier 2006; Forsythe and others 2006) and pragmatically usable. Having examined its core theoretical mechanisms, I propose that Cognitive Discourse Grammar (CDG) is a reasonable candidate model of discourse processing, especially given its explicit focus on producing visual models of the internal structures built while reading.

This core argument and approach is shared by research in Cognitive Systems (Adams 2009; Forsythe and Xavier 2006; Graesser and others 2005; McCrickard and Chewar 2005). Cognitive systems researchers build plausible (Forsythe and Xavier 2006; Forsythe and others 2006) computational models of human cognitive processes as a basis for human-machine interactions. The process for model development integrates scientific understanding of human cognition with structured engineering practices. Whereas the focus in Cognitive Systems is on designing software to enable machines to interact with people as people interact with each other; the focus in my research is on designing tools that successfully support and
externalize the process of language comprehension and its products as interactive situation models, to enable people to interact with their own cognitive structures. Nevertheless, the application of the criteria of plausibility (Forsythe and Xavier 2006; Forsythe and others 2006) will be useful in considering CDG+ as a Discourse Processing Model.

In cognitive science more broadly, not only are the merits of different models of cognition debated, but the mode of each cognitive model is also hotly contested. For example, in the debate over models of long-term memory, those favoring computational models argue that “verbal theories, like chunking, that emphasi[ze] the role of long-term knowledge are not precise enough to make detailed predictions about experimental data, but computational implementations of the theories can bridge the gap.” (Jones and others 2014) Thus, from the experimental psychologists’ perspective, the price of claiming that CDG can be extended into a plausible (Forsythe and Xavier 2006; Forsythe and others 2006) CDG+ Discourse Processing Model is that it must eventually be specified computationally and tested experimentally. For this reason, analytical tool-builders should engage with both Cognitive Models, and Task Models, like those specialized for Discourse Comprehension and analytical reading.

8.1.2 - Analytics and Discourse

The focus on discourse in analytics in my dissertation is shared by Thomas, a founder of Visual Analytics, who placed great emphasis on analytical discourse (Thomas and Cook 2005). While the baldly metaphorical use of the term discourse to refer to the "interaction between the analyst and her data (Thomas and Cook 2005)" could be taken as evidence of the extent to which discourse is under-theorized in VA circles, Thomas devotes considerable effort to develop the concept of analytic discourse as the "technology-mediated dialogue between an analyst and her information to produce a judgment about an issue. (Thomas and Cook 2005)" In visual analytic discourse, the analytical reasoning process is the basis for the ongoing dialogue between analysts and their information. (Thomas and Cook 2005) insists that "enabling this discourse is at the heart of the Visual Analytics mission."
Granted, discourse linguists may be uncomfortable expanding their prototypical sense of the term discourse to include the evaluative, iterative (and thus only vaguely discursive) process of interaction between analysts and data. To review, in the linguistic tradition, discourse is the process of producing, communicating, and comprehending language exchanged (not necessarily bi-directionally) between an intentional speaker (or writer) and a motivated listener (or reader) in a particular context (not necessarily shared).

While substituting multi-modal analytical data for language and excising an intentional interlocutor entirely at first glance may seem anathema to a linguist, it could be read simply as an extreme problematization of theories of authorial intention: as a test of Discourse Analysis theory and methods. Is authorial intention necessary in discourse? Are the readers’ linguistic sense-making skills applicable to non-linguistic or retrogressive linguistic data? Should the sense-making process used for non-linguistic data be called discourse? These questions will require some clarification in order to resolve conflicting conceptualizations of discourse.

Intentional interlocutors are without question required for linguistic sense-making. In language, intentionally selected and conventionally composed terms can be taken to mean something, whereas random words or references cannot. Similarly, in both general multi-modal and scientific analysis, only intentionally selected and properly collected data can be taken to represent an observed situation, in a way that random facts and figures simply cannot. In both cases, context determines the validity of the selection, and convention dictates the validity of its structural composition. Thus, whether human or machine, the original data collector determines whether the selection and structure of data sufficiently meets these standards. So one could hold choosing what to observe to be tantamount to choosing what to say.

In both cases, the original, authorial interlocutor or collector may not be present to justify their selections or explain their intentions. Thus, good data, like good writing, must contain sufficient contextual details for the recipient analyst to make sense of it. However, convention in this area, as regards annotation of data with salient contextual information, is not strong as with linguistic data, where such annotations are built into both lexical selection and syntactic structure, and
consequently analysts are frequently faced with absent-author + context-unreported data.

Following arguments from Cognitive Linguistics, since linguistic sense-making capacity is treated as a special case that employs the same mechanisms as embodied human sense-making more generally, if humans can successfully navigate and make sense of the general world despite an absent author, then they ought to be able to do so with linguistic data. And in fact, they can. Thus, the sense-making process whereby analysts infer or extrapolate contextual data within which to evaluate absent-author data ought to be either patterned after or identical to the sense-making process of humans experiencing the world more generally.

It should be noted that absent-author data is by no means equivalent to authorless data. Thus, while humans seem to do just fine making sense of a world which, \textit{prima facia}, seems to have no author, from the embodied human perspective on the ground, humans are always surrounded by a situation with intersecting mathematical, temporal, spatial, physical, corporeal, mental, perspectival, sociocultural, linguistic, and semantic dimensions. While all dimensions do not apply for each entity or each situation, aspects of these contextual, experiential dimensions are universally inscribed in the point of view adopted by the embodied sense-maker. Thus, the selection of a point of view, and a relevant level from these contextual experiential dimensions effectively authors an interpretive stance.

The built environment, for example, has no placard nominating the creators or purpose of each roadway, building, or artefact. But the intentions of builders, designers, and makers are conventionally known; roads are for driving, doors are for entering, windows are for viewing, shelves are for holding, etc. Even the natural environment in its pristine form is not experienced as pure chaos. Nature exhibits regularities which are often perceived as rules. The conservation of mass, the forces of nature, the transcription of RnA, the splitting of cells, the firing of neurons, the forming of memories, and so forth, are all considered to be systems or procedures with target and transitional states. While these procedures clearly do not manifest intent, each process does function more or less to author a result.

Though typically physically remote in written discourse, it is nevertheless difficult to excise the author entirely. Something can usually be inferred about the originator of textual data. This could present a significant barrier to describing the
sense-making process of analysts facing non-linguistic data as discourse, since inferences about the ‘authorial process’ that generated the non-linguistic data are conventionally validated by hypothesis testing. Yet just because one may attach features to an author, this does not mean that author-less content (e.g. empirical non-linguistic data) cannot also be ascribed reader-inferred features.

Empirical non-linguistic data has no intentional creator, per se. Of course, the system or method of data capture usually has some human originator whose selection of a point of view amounts to authoring and observational and interpretive stance. But as long the data itself is not simply designed by humans or synthetically generated by computation, one cannot claim a linguistic sense of ‘authorship’ of the data. Instead, such data is taken to represent features of a situation. This data may or may not have a recognizable and manipulable structure within which patterns can be identified and discoveries made. When it does, it becomes possible to conclude things about the situation which gave rise to those patterns.

In this way, by substituting an original observer for an intentional author, the traditional view of discourse can be brought into better alignment with the notion of analytic discourse from VA. If discourse can survive the substitution of other data for language and the absence of an intentional author, then Discourse Analysis begins to focus on the reader’s attempt to derive meaning from data. In this sense, Discourse Analysis becomes more or less synonymous with analytic reading in particular and sense-making more generally. This move, in fact, reclaims an older (Middle English) meaning of the term discourse as the power or process of reasoning.

This argument has brought the wider analytical processing of non-linguistic data into alignment with Discourse Analysis. Naturally, when engaging with properly linguistic data (e.g., text that respects Gricean maxims) and even when dealing with deceptive texts, practically speaking, any general or specialized (e.g., intelligence and policy) analyst who reads a text is engaging in discourse with text; and their analysis is (at least intuitively) a product of Discourse Analysis. Now, their concerns and focus are not on exhaustive and accurate typification of discourse units or understanding of phenomena like coherence and cohesion, as would be the case for an academic linguist in the field of discourse analysis. But professional analysts
necessarily and continually evaluate texts and seek relevant content by making using the phenomena studied in discourse analysis; particularly the quick segmentation of figure and ground, the ongoing segmentation of scenes, and the determination and evaluation of lexical, semantic, logical, rhetorical, and other functional relations between its propositions, which are understood as the smallest relevant situations in the text. For this reason, I will now turn to examining a cognitive discourse analysis approach in order to attempt to identify the components required as requisite sub-tasks necessary to accomplish the task of analytical reading.

8.2 - Previous Work

Naturally, any attempt to articulate the machinery of the original Text World approach, and especially the approach taken here to extend the original approach into the CDG+ Discourse Processing Model (For background, see Literature Section 3.1.3, Discourse Processing Models) will necessarily cover ground charted by its originator, and also the conceptual territories of other linguistic and cognitive theorists. I may be wrong, but I think that the trick to successfully standing on the shoulders of giants is really just to hop off as soon as you’ve got the lay of the land. Then, just make sure that your review from the top offers something novel, challenging, and game-changing in the span of the few heartbeats you’ll have before you land. Scholars who can do this should always land on their feet.

8.2.1 - Text Worlds

Text Worlds: Representing Conceptual Space in Discourse (1999) is Werth’s definitive statement of his ideas on knowledge, frames, text worlds and situations that builds upon coherence mechanisms invented to account for situational factors in Focus, Coherence, & Emphasis (1984). It is a complex, detailed work and an outstanding resource for any analyst who is committed to manually excavating the conceptual structure of text as it emerges in the mind of the reader. In simple terms, using this approach, I can draw a picture of what’s going on in my head as I read. The purpose of this research is to make it possible that, eventually, any reader could use this method to do likewise.
Thus, while I may not be able to adequately summarize Werth’s masterwork in the span of a few short seconds, in this section and the next Section 8.3.1, Cognitive Discourse Grammar, I will provide a 1000-fold compression of Werth’s original approach, that simply highlights some of the most important elements of the original CDG approach.

Though Werth named this approach Cognitive Discourse Grammar, this original nomenclature has been largely overlooked. This is, however, understandable since in a monograph of over 150,000 words, the terms ‘Cognitive Discourse Grammar’ and ‘CDG’ are used only 17 times.

Thus, the perspective offered in this Dissertation on the original CDG approach can literally be found nowhere else in the literature. This fact is not simply because the approach’s name is fairly obscured by its own non-standard naming and referential practice within the theory itself. Discourse Analysis based on CDG is widely applied and published in the fields of stylistics and poetics. Yet, the stylistic version of the original approach, called Text World Theory, does not discuss CDG, using these terms specifically because practically speaking, these approaches primarily use the most popular mechanism from standard CDG; the Text World.

This is understandable, since stylisticians and cognitive poetic analysts are not trying to solve problems of reference in formal semantics, as Werth was attempting to do, nor are they trying to create a cognitively plausible (Forsythe and Xavier 2006; Forsythe and others 2006) framework for representing Discourse Comprehension externally. Consequently, the stylistics approach is not concerned with the whole framework. They are simply trying to use the just those parts of the original framework that can help them accomplish appropriate kinds of analysis in their own domain. So it could be understood if Text World Theory did not notice, or even for choosing not to use the proper name for the original CDG approach.

But I have noticed these fine details as specific opportunities to extend and improve upon the original CDG approach. The original CDG approach to Discourse Analysis claims to be capable of capturing a schematic form of the internal representations of the conceptual structures produced in the reader’s mind while reading any type of text.
In this Chapter 8, CDG+, I will extend the original claims even further, arguing that only reducing the cost of applying this analytical method will make its application universal. Anyone who reads, especially analytically and professionally, could potentially benefit from some assistance in sorting out conceptually what they are reading, and organizing those conceptual structures so as to be able to effectively share their reading perspective with others. I will thus argue for the plausibility of CDG+ as a de facto Discourse Processing Model. CDG does not make this claim on its own behalf. Yet, given how many features of the CDG approach match the features of discourse processing models published since Werth’s death in 1995, this argument seems reasonable.

Thus, in Section 8.5, The Solutions, I present a range of novel revisions and extensions to the original CDG approach. Of specific interest to the task of introducing CDG to unfamiliar audiences, the reader should note Section 8.5.1, CDG+ Context Model, and 8.5.2, CDG+ Discourse Processing Model, and 8.5.3, CDG+ Discourse Model. In these sections I expand upon the original CDG approach by offering visual models to situate the elements of the Discourse Model and the Discourse Processing Model which produces it.

Of particular interest to analysts who are currently using the original or stylistic Text World Theory analysis methods, the reader should note Section 8.5.4, the CDG+ Deictic Matrix, which provides a deictic starting point for Text World diagrams. The Deictic Matrix provides a clear, interactive way to evaluate the Text World along four deictic dimensions. Thus analysts can begin placing their Text Worlds with reference to a clear deictic origo.

Of note to any analyst who must regularly evaluate the modal status, for example the existential or epistemic truth claims, of the documents or data they work with, the reader should note Section 8.5.5 the CDG+ Meta-Contextual Frame and 8.5.6 the CDG+ Manifest Semantic Analysis. The purpose of these contributions is to provide a means to capture deeper evaluations and judgments of terms and Worlds that are made while reading.

Since the utility of the original CDG approach is proven in the fields of stylistics and cognitive poetics, I will argue that CDG+ can therefore also be packaged as a useful and theoretically informed, extended manual Discourse Analysis method, or
TextWorlds+ Analysis, and eventually implemented computationally. This extended method is introduced in Chapter 9, TextWorlds+.

8.2.2 - Text World Theory

Scholars in the fields of stylistics and cognitive poetics may recognize Gavins’ revision of Werth’s original Text Worlds approach as Text World Theory (2007). Text World Theory is “a stylistic tool for text analysis and comprehension (Lahey 2014).” However, stylistics analysts naturally focus on analysis of document texture (e.g., unique features) and not understanding the process of comprehension. Consequently, stylisticians accept but do not (generally) query or develop the cognitive basis of Text World Theory or its status as a model of comprehension. Moreover, most users of Text World Theory in literary studies would not even recognize Cognitive Discourse Grammar, nor would they see Text World Theory as belonging to CDG. They would also not necessarily view the original Text Worlds approach to language as specifying the elements of a Discourse Processing Model in its own right, though they would also not be likely to argue against this position.

8.3 - The Concepts

In preparation for the exposition presented in this Chapter, I have used Chapter 3, The Cognition Literature, and Chapter 4, The Analysis Literature, to challenge each of the core theoretical components of original CDG approach with reference to the wider literature; paying particular attention to any recent developments that might invalidate key structures in the CDG approach.

From my starting point with generalist disciplinary training in communication theory and professional training as a policy analyst, in order to read Text Worlds (1999), I had to be prepared to employ a broad, trans-disciplinary reading strategy, guided by focused domain-spanning search to backfill my own knowledge gaps. Accordingly, I propose that the path taken here may be useful to for anyone starting the same task from outside of the specialized domains of Cognitive Linguistic Analysis and Discourse Analysis.

Essentially, each new concept and mechanism in the original CDG approach created a trans-disciplinary query. Most queries followed the pattern similar to:
“Are these concepts and mechanisms still appropriate?” For example; Do cognitive scientists still theorize and experiment with frames, propositions, & mental models? More specifically, in the years since the publication of Text Worlds (1999); Has consensus emerged on whether we think with propositions, or mental models? Have any semanticists satisfactorily resolved published shortcomings with classical propositional approaches? Have discourse linguists ‘solved’ coherence? Are frames currently used to represent knowledge?

Given the research focus of this Research Programme, each query also raises questions as how best to implement computational support for any process or procedure described. For example; What computational approaches to coherence are available? What computational lexical resources for frames are available? Are computational methods available for automatically identifying frames?

The short answer to each of these questions provided in Part I, Foundations, is that frames, propositions, and mental models are still appropriate conceptual structures, since each is still employed by scientists and theorists, notwithstanding an ongoing debate in discourse psychology and philosophy about the structure of representations of real-time, online thought. Moreover, neither semanticists nor discourse linguists have resolved their respective core challenges (e.g., respectively referential extension and truth conditionality of propositions and coherence in real world language use), so I argue that since CDG present a plausible solution to these problems, the approach offered by CDG is still relevant.

Ultimately I have found that mental models remain the basis for most discourse processing models, in the form of situation models. However, not all discourse processing models use the predominantly symbolic propositional representation; embodied approaches use ‘event-representations,’ which seem something like frames. In linguistic discourse analysis, assessments of global coherence are still challenging for current approaches to coherence, especially computational approaches. Most computational approaches to coherence operate at the level of words and consider lemmatized term matches in a limited window of sentences. No other methods identified since Text Worlds (1999) integrate prior cultural knowledge into the evaluation of coherence, nor do they consider propositions, nor do they evaluate functional relations between propositions. None use a computationally-assisted Artificial Intelligence Common Ground as a manager for
the activation window, or a repository for reference chains, or as an automated search agent for semantic and functional relations between propositions and terms. **Deixis** and **Anaphora** are researched widely and approaches are even implemented computationally. However, while some topics are widely researched, the role of the reader in assigning modal states like probabilistic truth assessments, is neither well-defined nor well-supported.

Thus, the concepts of **Mental Models**, **Frames**, **Propositions**, **Coherence**, and **Common Ground**, have each already been excavated deeply in the Literature section to establish whether these theories and structures and thus far have withstood the test of time and may still be considered valid, reasonable mechanisms. The answer is clear: the core components of the original CDG approach remain valid and widely used theoretical constructs today. What is more; their composition together in one model remains unique. From this solid footing, I am well placed to address the Problems (Section 8.4) I have identified in the original formulation of CDG in the following, by extending the original CDG approach in Section 8.5, **The Solutions**. Thus, this section reviews and summarizes the integration of these concepts in the Cognitive Discourse Grammar Framework.

### 8.3.1 - Cognitive Discourse Grammar

As discussed in Section 8.2.1, **Text Worlds**, while Werth did select the term CDG as the name for his human-centered, cognitive approach to **Discourse Analysis**, he only used this name 17 times in a text of over 150,000 words. Despite this referential anomaly, this Research Programme refers to Werth’s original approach as CDG.

From the discussion of foregoing Chapters, the reader should have a good idea of what **Cognition** and **Discourse** are, but what is the ‘Grammar’ of CDG? Givon (Givón 2005; Gernsbacher and others 1995; Givón 1993; 1992) suggests that **grammar** represents a cognitive process of automation, akin to forging neural pathways for high-frequency actions (Werth, 1999 p.27). Thus, the ‘Grammar’ of CDG provides specification for relations and dynamics between parts in the framework, primarily as constraints on those dynamics. As a cognitively focused **Discourse Grammar**, CDG’s principle contribution is to specify a series of
constraints that guide the process of Discourse Comprehension. The notable discourse processing constraints proposed in CDG are listed here for reference:

4) The activation of FRAMES is constrained by explicit textual cues.
5) The production of textually based inferences is constrained by explicit textual cues.
6) The evocation of FRAME information is constrained by the evaluation of FRAME relevance, which is determined through analogical situation feature matching. Thus, the selection of FRAME information is constrained by textual cues which must include situational features.
7) The evaluation of propositional coherence is constrained by the relation each proposition holds with other propositions in their shared CDG/+

Common Ground. Thus, the CDG Coherence Constraint formally states that every proposition is part of a discourse that has a Common Ground, and this proposition is functionally related to some other proposition in that Common Ground, by full or partial synonymy, antonymy, hyponymy, metonymy or metaphor.

The detail-oriented reader will note that common feature among these CDG discourse processing constraints is that these dynamics processes are constrained by explicit textual cues. This is the primary theoretical contribution that allows CDG to gracefully handle the immensity and complexity of context. While adult language users presumably have extremely deep and wide repositories of background knowledge and a ready capacity for reasoning and inference, as far as language processing goes, what matters is the text, and the precise cues it provides to interactively activate, reinforce, and produce reader knowledge.

Werth bases his definition of Cognitive Discourse Grammar on the common linguistic distinction between a text, which consists of the stretch of language itself without taking into account the surrounding context; and a discourse, which is the stretch of language together with the context which supports it. CDG therefore takes its primary data to be discourses rather than sentences, and it takes these to reflect certain cognitive rather than abstract systems (Werth, 1999 p.50).

Cognitive Discourse Grammar claims that our ability to form and understand texts (e.g., as abstractions of discourse) has two primary sources. Language users have available various formal textual signals that indicate the textual function of joined expressions, or propositions, while various semantic connections relate these expressions to each other and to the world that form the background for the text.
Thus, CDG is a discourse grammar in which the basic propositional “units used are locally referential with respect to their current context” (Werth, 1999 p.27).” In CDG/+ models of context “are characterized not in terms of empty logical elements [or logical predicates, as in minimalist, philosophical Possible Worlds], but rather as richly defined situations, complete with characters, and specified objects having actual relationships, carrying out concrete actions in adequately depicted circumstances. In a word, CDG is experiential (Werth, 1999 p.78).”

This Rich World Model of CDG extends philosophical approaches to Possible Worlds and integrates them with the Situation Models of discourse processing and the Mental Models of inferential reasoning and decision-making to produce something like a Mental Space model from cognitive linguistics to represent the conceptual contents of text built as an internal mental representation while reading.

8.3.2 - CDG/+ Propositions

CDG’s basic units of data is thus the utterance and the simplest ‘situations’ that can be derived from cognitive processing of utterances, which are called propositions. As discussed in Section 3.3.5, these are not the logical propositional representations of predicate calculus or formal semantics. Rather, CDG Propositions39 (Appendix Y. CDG) are conceptualized as the semantic expression of a basic utterance asserting a minimal scenario or scene, containing arranged objects, or an event wherein agent acts upon patient or state changes (Werth, 1999 p.50).

Beyond offering an alternative conceptualization of propositions as the basic level of knowledge representation, Werth aims to characterize higher-level structures for knowledge provided by both 1) the text and 2) the discourse participants, and 3) their union by differentiated them as 1) Text Worlds and 2) Frames, and 3) Common Ground (Werth 1999c).

8.3.5 - CDG/+ Text Worlds

The Text World is the Ground within which action and description (the Figure) makes sense, and within which all reference resolves. Thus, the Text World is gradually specified by the ongoing discourse, and is constructed out of the
combined general knowledge of the speakers. The propositions contained within text refer to the situations which constitute the foreground of the text. A situation is a complex unit made up of entities in relationships (Werth 1999f). A situation derives its deictic framework, or background, from the context specified in the Text World, though it may contain further linguistically embedded deictic elements — like tense, aspect, and modality — that provide finer detail to fill in the general characterization provided by the Text World. Thus, words and words grouped as propositions also activate frames of cultural knowledge which guide reader interpretation (Werth 1999g).

8.3.3 - CDG/+ Knowledge Frames

Many early theorists in Cognitive Linguistics [Fillmore, Langacker, Lakoff, Kay] shared a concept of the FRAME (called variously idealized cognitive model, cognitive model, script, rules, procedures) as storage mechanisms for Knowledge in Long Term Memory. However, in the view of one discourse grammarian who worked in the early years of the cognitive linguistics, other cognitive linguistic approaches employed the general concept of the FRAME without specifying “an account of knowledge structure” particularly as it regards “discourse context” (Werth, 1999 p.46). Cognitive Discourse Grammar (CDG) is Werth’s response to this gap. CDG integrates a cognitive approach to Discourse Linguistics with an embodied approach to knowledge structures, upon the premise that any full explanation of human language must take both verbal and situational context of language event into account (Werth, 1999 p.50).

So discourse participants necessarily enter discourse-events with a certain amount of knowledge \( K \), which is composed of the propositions contained in their personal knowledge base (Werth 1999g). This knowledge is arranged and accessed as Frames, which are selectively activated by textual cues.

8.3.4 - CDG/+ Common Ground

In conversation, the prototype of discourse, the speaker role alternates between participants. In such cases it is clear that context is jointly negotiated (Clark 1992; Werth 1999c). But this negotiation also occurs whenever propositions put forward by a producer must subsequently pass through the filter of a recipient’s knowledge
via language comprehension (Werth 1999c). Thus, written discourse and monologues are also jointly negotiated, though the negotiation in such cases is admittedly more private (Werth 1999f).

In all cases, the jointly accepted context and contents of discourse are accessible to discourse participants as the Common Ground. CDG+ adds several features to the basic concept introduced by Clark. In CDG/+, the negotiated and mutually accepted propositions are stored in a dynamic, reasoning repository that is powered by artificial intelligence, called the CDG/+ Common Ground. Accepted propositions constitute the CDG/+ Common Ground (Werth 1999c).

If the CDG/+ approach is correct, then the core of several of the most troublesome existential and referential problems in philosophy and formal semantics (e.g., generally intentionality and compositionality; or more specifically truth conditionality, referential extension, and referential opacity) can be resolved in CDG within Text Worlds through one further assumption: Any situations explicitly expressed with language must necessarily be possible situations in the Text World of its discourse (Werth 1999e). Thus, new propositions must not only be consistent with the propositions that define the Text World, but they must also cohere with, and satisfy a coherence constraint posed by all previously introduced and currently active propositions in the Common Ground (Werth 1999a). Thus, readers must then use the active verbal context together with the propositions of the foregoing text to negotiate the acceptability of each proposition.

8.3.6 - CDG Synopsis

Thus, I argue that combining CDG’s specification of cultural knowledge frames, text worlds, and Common Ground produces a unique model of discourse processing. Though, since Werth did not make this claim, I will extend the original CDG approach as CDG+ and make the claim myself. I will further argue that computational implementation of this model would permit testing and refinement of the CDG+ Discourse Processing Model, and could contribute to better understanding how analysts reason with text (Werth 1999e) and how insight is generated in the course of analytical reading (Werth 1999m).

The computational implementation of this extended CDG discourse processing model is envisioned in Section 9.7 as future work. Chapter 8, CDG+ describes a
range of original contributions to CDG, including improved manual TextWolds+ discourse analysis method that is explicitly based on the extended CDG+ Discourse Processing Model. TextWorlds+ Analysis affords users a methodology and will eventually be implemented as a computational system that can be used to create a baseline TextWorld model that will identify situations in text and populate them with enough details, in co-ordination with the original text on which they are based, for an analyst to approach a text with a digest of events occurring in the text, indexed to the text-defined and user-identified situations in which they occur. With CDG+ TextWorlds computationally implemented it will then be possible to evaluate the cumulative benefit of building digital knowledge frames and reusing Common Ground during reading-intensive applications of the analysis process. Thus, while CDG articulates its dependence on Cognitive Science and Cognitive Psychology, if the cognitive perspective can be attributed to the accepted principles of Cognitive Linguistics, the domain in Cognitive Science most relevant to CDG+ is Language Comprehension.

8.4 - The Problems

It has been established that though Werth initially referred to his analytical approach and research program as Cognitive Discourse Grammar, he did not do so frequently or consistently throughout Text Worlds (1999). So despite the fact that Werth rarely employed his favored term for this work, and in light of the fact that the scholars who have in various measures followed Text Worlds (1999) have described their approach as Text Worlds Theory, I will employ Werth’s chosen term, CDG, as the proper name for the domain of practice and theory covering Werth’s original Text Worlds approach.

It has also been established that not only did Werth not use his own preferred name for his approach, but he did not position CDG as a Discourse Processing Model. Notwithstanding, in Chapter 3, The Cognition Literature, this dissertation has carefully considered both the source material on which CDG is based and the current state of the art in discourse processing models (For this discussion, refer to Section 3.1.3, Discourse Processing Models). With reference to existing discourse processing models, the CDG approach initially exhibits surface compatibilities with both the integration-construction models of Graesser and the Event-Indexing model.
of Zwaan (though its unit of cognition is the proposition and not the event model of Zwaan et al.s). Future work will examine these relations further.

In this Section, I first consider the original problem space that initiated the creation of CDG, and then I turn to identifying shortcomings in the original formulation of the CDG approach. Each of these problems present an opportunity to revise the original case for Text Worlds, to articulate the CDG+ Discourse Processing Model, to rebrand the analytical methodology, and to revitalize the analytical practice of building Text World as TextWorld Analysis by introducing its benefits to a wider audience of students, academics, and analysts. These Solutions are discussed in the following Section 8.5. I could be wrong, but I would argue that anyone who must read widely or deeply could eventually benefit from both the scene analysis and the detailed concept modelling methodology used to produce TextWorld, once the process has been reliably automated.

8.4.1 - Coverage of Linguistic Phenomena in Generative Linguistics

To this point, it has been established that Cognitive Discourse Grammar (CDG) is a novel human-experience-centered research programme that is an outgrowth of the Chomskyan paradigm that accords with the fundamental principles of Cognitive Linguistics. (Werth 1999b). Where CDG’s originator Werth differs from Chomsky on points of methodology, CDG proposes simply to expand the linguistic domain of study beyond syntax, in order to focus on areas not treated in traditional generative linguistics. Differences on issues of coverage are addressed by proposing to study linguistic data from real language use, rather than artificial examples; and proposing the treatment of full discourse, rather than just selective singular sentences. Thus, CDG is a cognitive discourse-modelling approach to language that specifies something akin to a Cognitive Grammar (Langacker 2008; 2009) for discourse, which begins to resolve the discourse-sized gap (Croft and Cruse 2004d) in Cognitive Linguistic research.

The standard generative approaches to discourse only consider the text, or the verbal context that surrounds sentences. This is understandable, since from within the generative paradigm, the context problem appears insurmountable. As an alternative, CDG claims to model both the surrounding text (as the archived verbal context of discourse) and the surrounding language situation, including everything
and everyone within and around the language event. (See Section 8.5.1, CDG+ Context Model, for a visual model of the role of context and situations in CDG/+.)

8.4.2 - Text Worlds Monograph

In an approach typical in functional and discourse linguistics, Werth introduces and then immediately and extensively exemplifies key concepts in an engaging and fairly assimilable style. However, in subsequent chapters, he often revisits and expands upon previously introduced concepts in the clarifying light of related exposition. Thus, for many central ideas and mechanisms of CDG, the entire theoretical picture is only achieved through an exhaustive reading (and re-reading!) of over 400 densely packed pages. And while many concepts and mechanisms in CDG are closely related or even dependent, the overall chapter and section structure of the text does not always well reflect the conceptual relations and structure of the CDG model. Moreover, though the text exemplifies the original Text World approach in rich detail, a detailed, step-by-step specification of the analytical Text Worlds method is not offered.

Moreover, the posthumous publication of Werth’s monograph on text-worlds was not based a completed manuscript, but on an incomplete draft and various previously published articles and chapters. Thus, the text suffers occasionally from inconsistent terminology in certain places where whole articles or chapters were reprinted. Consequently, CDG would seem to be based upon an inconsistent taxonomy of foundational terms (Gavins 2005), so it either contains several outright contradictions, or at very least it sorely taxes the polysemous nature of certain of its key terms (e.g., modality).

8.4.3 - Text World Theory

While many cognitive poetics and literary text-world analyses have been published since 1999 (Lahey 2014; Giovanelli 2013b; 2013a; GiovanelliMarcel 2013e; 2013h; 2013d; 2013f; 2013a; 2013b; 2013c; 2013g; Whiteley 2011; Giovanelli 2010; Narayan 2009; Semino 2009; Farrow 2008; Tendahl and Gibbs 2008; Filipović 2007; ‘Text World Theory - Edinburgh Scholarship’ 2007; Yingfan Gao and others 2006; Gavins 2006; Speer and Zacks 2005; Gavins 2005; 2003; Szilas 2003; Pollard 2002; Emmott 2000; Hidalgo 2000), only a few like Gavins
(2005, 2008) have contributed to the ongoing theoretical development of Text Worlds. Yet Gavins’ Text World Theory has focused primarily on the practical and diagrammatic production of Text Worlds to the exclusion of much of the machinery and mechanisms — like an AI-powered Common Ground, modal and literary sub-worlds, coherence constraints, emphasis, frames and inference, accommodation, incrementation — introduced by Werth. And as yet, no contemporary Text World theorists have initiated computational implementation of this method.

### 8.4.4 - Static Text World Diagrams

CDG+ makes an important distinction between the Text World diagrams that are manually drawn on paper (or composed in a diagramming program) as an external schematic representation of internally represented mental models, and the future of TextWorld Models that are automatically built in a computational reading system as interactive multi-dimensional digital models of internally represented discourse processing situation models.

Though the entire CDG+ Discourse Processing Model is responsible for creating the dynamics within which TextWorlds are generated in the course of Discourse Comprehension, stylisticians use a simplified analytical method to break off Text Worlds from the larger CDG apparatus. So Text World Theory builds static Text World Diagrams whereas CDG+ builds dynamic TextWorld Models. The description of CDG+ and the CDG+ Discourse Processing Model is therefore required to introduce the full capacity of the original Text World approach to text analysts, analytical tool-builders, and curious linguists.

### 8.4.5 - The Underspecified Method of Text World Analysis

As noted previously, the original CDG approach neither specifies a general method for Text World Analysis nor does it provide any step-by-step instruction for producing static Text World Diagrams. Consequently, the approach of modern stylisticians is an improvement over the original CDG approach, insofar as Text World Theory actually specifies a usable analytical method for accomplishing Text World Analysis.
However, Text World Theory selects only a single mechanism from the wider CDG+ Discourse Processing Model. It takes a simple version of Text World mechanism from Cognitive Discourse Grammar to produce a static Text World Diagram as an analytical view onto a given text. Stylistic Text World Theory analysis requires the reader to first identify elements contributing to the ground of the situation and then also the action and description of the situation. By following several rules for presentation of these elements, the analyst manually builds their static Text World diagrams, which are not fully specified models of human comprehension.

Alternatively, in the CDG+ manual method of TextWorld Analysis described in Section 9.5, the analytical method of producing a dynamic TextWorld Model is an emergent product of the challenging manual process of tracking the dynamics of reading comprehension as specified by the whole CDG+ Discourse Processing Model. As a stopgap prior to computational implementation of CDG, the CDG+ extended manual method of TextWorld Analysis provides digital templates not only for identifying the two major classes of TextWorld elements (e.g., World-Builders and Function-Advancers) but also for standard dynamics (e.g., world shifts, sub-world creation, coreference) in CDG+ Discourse Processing Model.

8.4.6 - The High Cost of Text World Analysis

To review, stylistic approaches to Text World Theory (Gavins 2005; 2000; Whiteley 2011) typically do not consider the entire CDG Model or its dynamics. Instead, they appear to use a simplified version of the basic classification tasks required to produce static Text World Diagrams and not dynamic TextWorld Models of comprehension.

In the view presented here, and it is one which I believe Werth would share, both the canonical Text World approach and the extended manual TextWorld+ Analysis method (See to Section 9.5, Manual Method) are in fact isomorphic with the online language comprehension dynamics of the CDG+ Discourse Processing Model. Nevertheless, simple forms of stylistic Text World Analysis are clearly possible without the wider machinery of the CDG+ Discourse Processing Model. Therefore, I must account for why the wider machinery of the CDG+ Discourse
Processing Model is even necessary. My answer hinges on the difference in utility of Text World Diagrams compared with that of TextWorld Models.

Moreover, I argue that it is possible to produce Text World diagrams with the simplified stylistic version of the original Text World approach only because the analysts’ human cognition is carrying the load of entity and proposition identification and indexing, coherence analysis, coreference resolution, emphasis placement, knowledge activation, inference, term disambiguation, semantic role labeling, probabilistic truth assessment, speaker evaluation, epistemic judgment, content assessment, and linguistic category assignment.

The high level of training and linguistic expertise of stylistic analysts means they may have a unique ability to ‘just see’ many linguistic structures and labels that the average analyst would not necessarily be able to identify. And, of course linguistic analysis is hard work and linguistic intuition only gets the analyst so far. Thus, not everyone would even be able to do even the simplified stylistic form of Text World analysis. This fact supports my primary reason for pursuing implementation of this method computationally — to offload the high expertise cost required for linguistic category identification to algorithms from Natural Language Processing, thereby making the execution of the task of reading, and the analytical task of creating a model from the conceptual contents encountered while reading, less costly and more accessible.

8.4.7 - Toward a Deeper Understanding of Discourse Processing Tasks

So regardless of whether the CDG+ Discourse Processing Model is verifiably the way that humans process language, and I cannot validate this important supposition here, we do know that these cognitive, analytical subtasks are taking place in support of understanding language. Thus, considering the anticipated range of reader participants in this present discourse and given their expected diversity along disciplinary lines, I argue that a thorough introduction to the entire CDG+ Discourse Processing Model is critical for both analysts and tool builders.

First, 1) any language-focused text analysts interested in implementing the manual method of extended TextWorld+ Analysis must know how the extended
CDG+ Discourse Processing Model works, before they can be trained in the TextWorlds+ Analysis method, since the dynamics of the discourse processing model produce the discourse model in the mind of the reader. Thus, the entire CDG+ Discourse Processing Model is required background for the novel extensions to the original method reported in Chapter 9, TextWorlds+, since it provides the background details required to proceed along to each next step. The extended CDG+ Discourse Processing Model generalizes how these subtasks function together to produce comprehension, and the CDG+ Discourse Model is the product of that comprehension process.

Moreover, 2) a deeper introduction to CDG+ is also important for informing analytical tool-builders of the dynamic cognitive complexity of the overlooked analytical task of reading. It is, 3) similarly required to convince linguists who may not be familiar with the domain of online language comprehension or cognitive discourse processing that CDG+ is a reasonable and worthwhile Model that provides a solid basis for pursuing research in this area, especially in the areas of computational and cognitive research on discourse.

Finally, 4) in order to build towards a computational version of TextWorld+ Analysis, these subtasks must be used as components in the design and implementation of a specific computational sequence (Section 9.7, Future Work) that approximates human language understanding to produce a baseline TextWorld as an interactive scaffold for real human language understanding. The system I build toward must therefore attempt to model language understanding in order to try to help readers to see how they understand language. I have considered using other discourse processing models; but they don’t provide a satisfactory account for context, they don’t integrate cultural knowledge frames, and they don’t integrate the results of inference, and they don’t prioritize the production of visualizations of internally produced discourse models.

8.4.8 - World Placement

While standard CDG provides reasonable and defensible directions for placing elements within Text Worlds, standard Text World analysis is underdetermined as regards where to begin placement of TextWorlds. While standard TextWorlds is clear that the basic temporal signature of the first sentences of a text should
determine the first TextWorld’s starting position, the convention is to begin positioning relative to the Present Time Zone.

Unfortunately, the standard approach presents some real practical problems and design challenges. In particular, this approach is problematic for texts that begin in the Past or Future, since standard Text World diagrams do not include the ‘true present’ as the temporal ‘origin.’ So the present of the narrative may in fact be a previous or anticipated present, wherein ‘the story’ present is ‘the real world’ past or future. The standard TextWorlds approach does attempt to visually distinguish the Present Time Zone with bold line weight, but this contrasting design only works when the Present Time Zone and some other contrasting Time Zone is in use. Thus, in the view presented here, the standard method offers insufficient distinction between the TextWorld’s actual starting point and its ostensible temporal origin. Until a Present Time Zone Text World is activated, it is not immediately apparent (e.g., simply by looking at the diagram) whether an actual CDG Time Zone is Past or Future. It may therefore be helpful to think of the standard CDG Text World placement diagram (Figure 8.4.4.1) as a model for standard, face-to-face discourse, where the Present Time Zone is the typical default.

In the standard approach to visualizing Text Worlds, the question of where to place each TextWorld with reference to existing Worlds and sub-worlds is also not strictly determined. The suggested convention (Figure 8.4.8.1) is to fan TextWorlds out with past Worlds beneath present, and future Worlds above, so as to minimize overlap, since both minor overlap and floating disconnection are each intended as a unique meaning-bearing configuration. Specifically, in traditional Text Worlds layout, minor overlap and underlap indicate participant accessibility in both proximal and remote past and future time zones, respectively. Non-overlap or floating placement on left and right indicates a modal or epistemic disjunction that permits only character accessible worlds in both proximal and remote past and future time zones, respectively (see Werth’s original Tense Relationship diagram (Figure 8.4.8.1 or Werth (1999, p. 247))).
8.4.9 - How to help where help is needed

Finally, despite all the dynamic complexity of interactive language comprehension modelled here, it is important to note that this entire complex of cognitive sub-tasks occurs within milliseconds of stimulus onset (Jaeger and Snider 2013; Poljac and Yeung 2012; Schmalhofer and Perfetti 2012; Liang and others 2010). Consequently, any efforts by analytical tool-builders toward modelling and supporting this easy, almost-free (Zwaan 2015) cognitive process of creating meaning must neither drag down the reader’s speed nor increase the task difficulty.
Now, these facts create critically important criteria for both tool-users and tool builders.

These facts support my argument that analytical tools must first do well where humans do well in order to bring analytical tools to the point where they can actually provide help to human readers with what humans cannot do so easily. Thus, whatever a human can do easily (e.g., process text and make syntactic, lexical, semantic classifications and connections nearly instantaneously) tool-builders must have the analytical tool approximate, in order to facilitate where humans slow down or have more difficulty (e.g., integrating valid new information into models (Liang and others 2010), negation (Glenberg and others 1999), and ascertaining and assigning modality classifications. For example, if humans can easily accomplish the first steps of a sub-task network (in the Flow Model, See Chapter 5, Flow) but are stuck in the linear reading process or just have more difficulty with integrating new information into their mental models, then tools that intend to offer support for analytical reading must first get this point, in order to focus on helping here.

The ‘easy’ steps (for human readers) must first be transacted externally and computationally. In my approach, extensive automated preprocessing will be required using text analysis at various levels of complexity from relation-extraction (Angeli and others 2015; Aryani and others 2013; Andrews 2004), semantic-role labelling (Gildea and Jurafsky 2002; Palmer and others 2005; Martin and Chao 2001), discourse parsing (Tofiloski and Brooke 2009), and anaphor resolution (Klebanov and Wiemer-Hastings 2002; Theijssen 2007; Thompson and Radvansky 2015) to shallow parsing (Oostdijk and van Halteren 2013; Lapata and Lascarides 1999; Chan 2006) and standard tagging of natural language classes (Manning and others 2014; Toutanova and others 2003; Toutanova and Manning 2000).

Then, the complexity of the various classification and phenomena identification processes employed for preprocessing must be simplified for the reader by the constrained insertion of those preprocessing results into standard structured elements in an interactive model of the conceptual contents of the text. This model represents an automated ‘reading’ or interpretation of the text, which the reader can interactively refine to produce a deeply personalized mental model of the conceptual structure in text.
8.4.10 - The CDG+ Problem Synopsis

This Section adds a description of the origin and features of the CDG approach to the foregoing description of its key concepts, particularly noting some of its Challenges. The first issue to note is the fact that while the author originally called this Research Programme and analytical framework Cognitive Discourse Grammar, this name cannot be found in the literature, even among approaches that cite Text Worlds (1999).

This section has first identified some small coherence and consistency problems with the original Text Worlds monograph that can largely be attributed to its posthumous publication. Most importantly, it has distinguished between the original CDG Text World approach, the current stylistics approach of Text World Theory, and the extensions and revisions contributed here as CDG+. This section has further distinguished between Text World Diagrams and TextWorld+ Models. Whereas the former are static handmade visualizations made for the purposes of summary and publication, the latter are computationally generated and dynamically interactive.

Though Text Worlds are just one part of the CDG+ Discourse Model, stylisticians use a simplified analytical method to effectively break off Text Worlds from the original CDG approach. Moreover, both the original CDG approach and Text World Theory build static Text World diagrams whereas CDG+ aims toward building increasingly interactive and dynamic TextWorld+ Models. Neither the original CDG approach nor stylistic Text World Theory have fully and satisfactorily specified a manual method for Text World analysis.

Moreover, both require a high degree of linguistic expertise which makes both approaches costly. Both novice and general analysts and analytical tool-builders are therefore expected to have procedural knowledge gaps regarding the foundations and procedures of the TextWorlds Analysis method. Therefore a more complete description of both the original CDG approach and the extended CDG+ Discourse Processing Model is thus required to introduce the full capacity of the original Text World approach to text analysts, analytical tool-builders, and discourse linguists. As will be established in the following sections, the CDG+ Discourse Processing Model constrains the dynamics within which the CDG+
**Discourse Model** is produced in the reader’s mind in the natural course of Discourse Comprehension.

Then, there is the problem of an underspecified deictic starting point in the original method for the creation of Text Worlds, and the fact that the original **CDG Common Ground** does not specify what happens to accepted propositions in the Common Ground once they lose their activation focus. Finally, as noted in Chapter 5, Flow, the **Cognitive-Functional Flow Model of Analysis** is not complete until its requisite analytical reading sub-task network is fully specified.

Before the reader can consider accepting the claims of a revised and extended CDG+, they ought to first have a clear understanding of the original approach.47 To this end, this Chapter should be read as a meta-introduction to CDG that offers motivation towards further reader-directed exploration. In the view presented here, the mental representation of the **CDG discourse model** emerges from the dynamics of a unique **Discourse Processing Model**, though the original approach never claims as much itself.

## 8.5 - The Solutions

This Dissertation Research Programme offers focused solutions to the problems detailed in the previous section by extending the original CDG approach in several ways. Lacking the space to introduce the entire CDG approach, this section will summarize and specify relations between the **CDG+ Discourse Processing Model** and its product, the **CDG+ Discourse Model**, beginning with the visual **CDG+ Context Model**, which specifies the four Situations that impact both CDG+ process and product Models.

Then, to address the problem of an underspecified deictic starting point in the original method for the creation of Text Worlds, I have created a variable **CDG+ Deictic Matrix**. Analysts simply select a point within the Matrix that encapsulates their judgment of the TextWorld’s status in four deictic systems.

Next, since the focus of original CDG is on building Text Worlds as an aggregate representation of reader judgments, it is curious that there is no explicit focus on supporting the capture of individual specific reader judgments while reading. Thus, I have designed the **CDG+ Meta-Contextual Frame** and the method
of *Manifest Semantic Analysis* for classifying both individual terms, propositions, and TextWorlds along multiple *deictic* and *contextual dimensions*.

Moreover, since the original *CDG Common Ground* does not specify what happens to accepted propositions in the Common Ground once they lose their activation focus, I have specified the longer-term collection of deactivated terms into the *CDG+ Personal Ground*, and drawn a connection between this discourse-specified long term memory bank and the formation of Frames in Individual Long Term memory through repetition and aggregation TextWorld Views or scenes.

Finally, since the *Cognitive-Functional Flow Model of Analysis* is not complete and requires the specification of requisite *analytical reading* sub-task network, I have drawn upon to the *CDG+ Discourse Processing Model* to specify the *CDG+ Task Model for Analytical Reading*.

**8.5.1 - CDG+ Context Model**

The critical argument of CDG/+ (e.g., both original and extended CDG) is that isolated context-free meaning is trivial. Context makes inescapable and critical contributions to the process of coming to understand linguistic meaning. As a pragmatic theory of discourse, CDG/+ distinguishes context as including the real world, the situation wherein interlocutors engage in interactive verbal exchange, the live *verbal* or archived *textual* contents of that exchange, and the recalled knowledge of participants. Extended CDG+ formalizes this *theory of situation* as the *CDG+ Context Model*, which is comprised of four related Situations: The Immediate Situation, The Interaction Situation, The Verbal Situation, and The Knowledge Situation. In this view, each use of language occurs in a context which is a dynamic, overlapping composite of these four situations. The manual method of extended TextWorlds+ Analysis is thus a reader-driven procedure for identifying and diagrammatically capturing the details and contributions made by each of these different levels of context. TextWorlds+ Analysis is based on the most recognizable mechanism of the *CDG+ Discourse Processing Model*, the *Text World*.

This section takes an incremental approach to introducing and discussing the CDG+ Context Model, as the foundation for the gradual introduction and organization of the elements of the proposed *CDG+ Discourse Processing Model* in
the following section. The CDG+ Context Model also draws out parallels between
the various levels of situation which have contributed to the extension of the
original CDG approach as the **CDG+ Discourse Processing Model**.

In the original Text Worlds approach, a world is a conceptual domain
representing a state of affairs. A situation is a mental representation of a specific
*state of affairs* defined by ongoing action within a constant setting. A situation is
thus a mental representation of what is known as an event (Radvansky and Zacks
2014; Maguire and others 2011; Kurby and Zacks 2008; Zacks and others 2007;
Barbara Tversky and others 2004; Zacks and Barbara Tversky 2003; 2001). A *world*
is composite of the interactions between successive *situations*. Worlds thus
 corresponds to situation models (Cardona-Rivera and Cassell 2012; Ferstl and
others 2005; Barsalou 2003; Fincher-Kiefer 2001; Zwaan 1999; Zwaan and
Radvansky 1998; Zwaan and others 1995b).

Thus, the *verbal situation*, or text, is an abstraction of the *interaction situation*,
or discourse; which is a carefully filtered subset of the *immediate situation*, or real
world; which together evoke the *knowledge situation*, or a collection of recalled
situation sets evoked from participant knowledge (See Figure 8.5.1.1).

The discourse world is the *mental representation* of the *state of affairs* of the
immediate real world situation in which speaker-hearer interaction is taking place;
but only of those elements that contribute to the language event. So neither the
original Text Worlds approach nor the extended **TextWorlds+ Analysis Method**
explicitly models the whole wide world of the *immediate situation*. Instead, the
*discourse world* is the the closest level to the real world, in the multi-level **CDG+
Discourse Model** of the *Verbal, Interaction, and Knowledge Situations* that are a
part of every extended, discursive language event.
Figure 8.5.1.1. A collapsed view of the four related types of Situations modelled by CDG+ Context Model. These main situations are also represented within the CDG+ Discourse Model that is produced by the CDG+ Discourse Processing Model.

The CDG+ Discourse Model (Introduced in Figure 8.5.1.2, and fully expanded in Figure 8.5.3.1) is a multi-level representation of the discursive interaction situation that accommodates all relevant levels of situation. At the highest level of the CDG+ Discourse Model, the situation is represented by the Text World (Figure 8.5.3.3). At the finest level, propositions (Figure 8.5.3.2) are micro-situations that encode the necessary details to make sense of utterances. Both the **Interaction Situation** and **Knowledge Situation** collect, capture, and arrange propositions in an endless array of evocative configurations. (For definition of propositions, see Section 3.3.5, Propositional Representations, and for discussion, see Section 8.5.3, CDG+ Discourse Model.) Expanding each of the Verbal, Interaction, and Knowledge situations of Figure 8.5.1.1 reveals their primary abstractions or main objects (Figure 8.5.1.2).
Figure 8.5.1.2. The CDG+ Context Model expanded to its first level reveals the top-level components of each Situation in the Model.
The knowledge situation theorizes knowledge structures for specific modes of knowledge. The interaction situation encompasses the extended language event of discourse, the discourse participants and their actions, and the multi-level discourse model created by discourse participants naturally in the course of interacting using language. The verbal situation includes the text as an archival abstraction of the discourse language event, its characters, and a shallow bibliographical model of the text’s surface features and metadata created by readers for indexing and retrieval purposes.

Digging down still further (Figure 8.5.1.3), we uncover two unique knowledge structures and two main modes of knowledge. We also see that the four primary components of the Discourse Model include knowledge frames, a Common Ground, the discourse world, text worlds, and sub-worlds. Note the wide range of activities required of discourse participants in the fulfilment of their roles producing and processing discourse. These activities are semantic predicates (See Figure 8.5.1.4) that may take a range of arguments. (See Figure 8.5.8. for the arguments taken to produce and process discourse.) The discourse itself is composed of a coherent series of individual utterances or speech acts (Austin 1962), while the text is composed of sentences.

Note that in TextWorlds, the primary purpose of discourse is the transfer of the contents of personal knowledge into shared cultural knowledge through language. And just as the verbal situation is an abstraction of the interactive discourse situation, the text is an abstraction of the discourse, and each sentence is an abstraction of its utterance. Finally, just as the discourse world is a subset of discourse-relevant elements from the real world, the evoked knowledge frames active in the discourse model are a subset of discourse-relevant frames from the total available set of knowledge frames.

The top-level actions of discourse participants are detailed in Figure 8.5.1.4 (See also CDG+ Task Model for Analytical Reading). Since Characters are portrayals of people, all Characters (within the expected normal range of mental faculties) are capable of taking these same actions as Interlocutors, only within the Text World rather than in the Real World. When telling stories of past events, for example, participants in the language event introduce and track characters, entities, relations, and actions in a text by means of reference and reference-chaining (e.g., anaphor
Figure 8.5.1.3. The CDG+ Context Model expanded to its second level. This reveals the major components of each Situation in the Context Model.
resolution) within *text worlds* and *sub-worlds*. All references are resolved as pointing to *entities* within these layered *worlds*. *Reference chaining* links terms (e.g., the specific propositions used to activate the representation of an entity) with their activated representation in the model. Characters are likewise capable of linguistic event indexing and reference chaining.

Expanding, or drilling down into the CDG+ Context Model one step further reveals structural elements particular to knowledge frames, and two further types of knowledge for both personal and shared modes of knowledge (Clark 1992).

This third level expansion also reveals details of the composition of each of the four main components of the Discourse Model. Both *knowledge frames* and *Common Ground* are propositional knowledge bases that function as long and short-term memory in the *Discourse Model*. The two components of discourse worlds (e.g., counterparts and cross-world anaphora) call for a brief explanation. Discourse is not restricted to referring simply to mutually apprehensible objects in the immediate environment. It may also refer to abstract, imperceptible objects in *remote situations* (See Section 8.5.4, CDG+ Deictic Matrix for a discussion of proximity and remoteness, and particularly Figure 8.5.5.1.13 for visualization of the same) that are not part of the actual shared discourse world. Thus, while the discourse world is the locus for interlocution, it often contains very few (or no) references to objects in the immediate, shared real world situation, because in many cases, there is no shared real world situation between interlocutors. It would contain objects manifest to the writer and the reader, but those sets of objects are often exclusive and the *Discourse World is thus often empty*. When it does contain a subset of objects from the immediate, shared, real world situation, these references occur in and from the text world to the discourse world. All objects in the discourse world are thus *counterparts* to their references in the text world, and this special case of reference between worlds is called *cross-world anaphora*.

Text World elements are divided into *Ground* and *Figure*, or those elements that define the contextual backdrop of the scene and those that describe the action and description that unfold within that scene. *Sub-worlds* are created for any reference whose epistemic ground diverges from that of the current text world. Their components are otherwise identical to those of text worlds. At each level, a *world* is initially defined by means of the *deictic* and *referential elements* nominated in
the text and fleshed out from participant knowledge in a process called world-building. The standard concept of *Deixis* from linguistics defines the conceptual

*Figure 8.5.1.4. The CDG+ Context Model expanded to its third level.*
domain appropriate for the processing of a particular discourse with reference to the speaker’s egocentric location in time and space. Then, a variant of the standard concept of Reference from linguistics introduces, tracks and links entities in the established conceptual domain (e.g., a world) using anaphoric and semantic connections to ascertain coherence.

So, the process of world-building establishes the setting (Fletcher and Chrysler 1990; Rinck and Ulrike Weber 2003; Zwaan 1999; Zwaan and van Oostendorp 1993) and entities in a situation and thus sets the basic contextual parameters within which entities in the text world can operate. The wider literature on discourse processing models (Rinck and Ulrike Weber 2003; Zwaan and others 1995b; 1995a) concurs that these objects and entities are central dimensions in situation models (e.g., situation model is the standard discourse processing terminology for the mental model created by a reader to handle the phenomena that TextWorlds handles (Zwaan 1999)).

Finally, by fully expanding the CDG+ Context model of the Interaction Situation (Figure 8.5.3.1), we see the Knowledge Situation in its entirety. Note the distinction between the two personal modes of knowledge amounts to a distinction between conceptual knowledge gained by abstract reflection and stored as propositions in knowledge frames, and the functional knowledge gained by physical experience and stored in script-like structures. Shared knowledge is either shared generally among the wider social group, as with linguistic and cultural knowledge; or mutually between people in shared experiential or mutually perceived real-world situations.

While the fully expanded view provides a surfeit of detail on both the CDG+ Context Model and the CDG+ Discourse Processing Model, the following discussion will focus on fleshing out the CDG+ Discourse Processing Model. Details in the CDG+ Context Model drawn from standard linguistic discourse and text analysis will not be elaborated further.

8.5.2 - CDG+ Discourse Processing Model

To review, the basic data48 of Cognitive Discourse Grammar is the discourse rather than the sentence; and as opposed to the generative approach, CDG takes these data to reflect certain cognitive rather than abstract systems. The pivotal
distinction between CDG and other early grammars and cognitive space accounts that emerged from within Cognitive Linguistics, like those offered by Fauconnier (Fauconnier 1985) or Langacker (Langacker 1987), is that others were based on a sentence perspective. Consequently, in Werth’s (Werth 1999b) view, so long as they maintain a sentence perspective rather than a discourse perspective, other approaches (like Mental Spaces and Cognitive Grammar) could not offer a fully integrated language theory.

This strong claim is supported by a simple argument: A sentence-based approach can only offer sense-schemas of individual sentences that are generalized, unanchored and ambiguous. Sentence-based analyses are generalized to the extent that they simply evaluate combinations of lexical items and syntactic values. They are unanchored such that their terms are deictically unattached to any specific context; and ambiguous since they cannot determine between multiple lexical entries.

What does this mean? An example will illustrate this claim. CDG theory specifies that Discourse Models are built initially in large part by specific deictic terms like ‘the, this, that.’ A sentence like E1 does not specify which dog, or where it rolled by. Using ‘the’ instead of ‘a’ as an article for the ‘dog’ would create a presupposition that the reader must already know of this particular dog (e.g., that the ‘dog’ is already in the Common Ground). Here, this is not the case so in this sentence ‘a dog’ is unanchored.

EXAMPLE 1 (E1): A dog rolled by and waved.

Many terms in E1 are ambiguous. ‘Rolling’ could be a tumbling roll, as down a hill. However, such roles are not usually sufficiently controlled to simultaneously produce an action recognized as a ‘wave’. ‘Rolling’ often indicates a movement in a vehicle of some sort, and a reasonably flat surface good for rolling on. So some sort of vehicle and path are generally indicated, but not specified. ‘Waving’ is physical action and a social activity, a movement of the arm or hand, and a recognition of an equal. This creates in the readers mind a likelihood of both a social entity, and one with a particular kind of body that enables the articulated motion of a whole arm and hand. Since the ‘wave’ recognizes another, the embodied personhood some other person or persons, possibly the speaker, is thus generally indicated, but not specified. In E1, there is no larger scene beyond the
A decontextualized sentence that permits any reference or inference to resolve to a specific sense-schema.

Since ‘dogs’ don’t usually ‘roll’, but walk; and almost never ‘wave’, the reader must wonder what kind of world this is. A cartoon, perhaps. Or maybe the YouTube sensation has learned a new trick. Or it is possible that this sentence doesn’t even refer to a canine, but to a sports fan, whose favorite team has finally made it to the Sugar Bowl, or to a marine in some military vehicle. Without a context, it is impossible to know what this sentence refers to, so this sentence is ambiguous.

Standard CDG argues that only a grammar sensitive to context can overcome these problems (Werth 1999f). Thus, a context-sensitive discourse-based approach can provide full, semantically rich meaning-schemas for related (e.g. contiguous and coherent) sentences that are highly specific and personalized by not only combining lexical items with syntactic values, but it also deftly handling polysemy and ambiguity by negotiating the best selection from multiple possible lexical entries. The discourse perspective also anchors all references using deixis to situate all expressions within a specified limited context within which it can robustly be claimed that all references make sense to discourse participants. That is to say, all references resolve to points located at different levels within the CDG Model.

In a Discourse Grammar, sentences are a part of a larger verbal context that describes and limits the referential scope of terms and includes background details and nominated entities. But, as we have seen in the CDG+ Context Model a thorough treatment

Thus, in Cognitive Discourse Grammar, the propositions contained within sentences are locally referential with respect to their current context and level within the Discourse Model (Werth 1999f). Contexts are characterized as richly defined situations, having characters, objects with specified relationships, executing real actions in adequately described circumstances (Werth 1999f).

In discourse, participants use this context in their attempts to process utterances such that the propositional information they contain bears upon the information already present in the Common Ground. Propositions that fit their context are coherent (Werth 1999b). Propositions that make complete sense are considered to
be **deictically anchored**, such that they refer into the world depicted by the discourse (Werth 1999b). Propositions that do not have a specified functional relationship to their context are **new information**, which the reader must then figure out how to integrate into the scene.

In order to effectively process discourse, *participants* need to be able to represent the notion of a **conceptual background**, which Werth provides as a **conceptual model/mental model/mental space/situation model** that is simply called a **Text World** (Werth 1999f). Thus, in Cognitive Discourse Grammar, a Text World is a deictic space, defined initially by the discourse itself, and specifically by the deictic and referential elements in it (Werth 1999b). Text Worlds are the extended scene within which action and description unfold.

As readers or hearers process utterances, and increment propositions into the **Common Ground**, individual terms and wider situations may cue the activation of repositories of cultural knowledge stored as **Frames** in long-term memory. These frames have detailed information about scenes, which can be drawn upon supply expected details that may not be provided in the text. Thus, in the above example, E1, frames for DOGS, ROLLING and WAVING were activated, but no inferences could be made since there was no wider context upon which to base a sense-selection, frame selection, or default slot information-selection decision.

Since it is based on a plausible (Forsythe and Xavier 2006; Forsythe and others 2006) model of Discourse Comprehension, the extended manual method of **TextWorlds+ Analysis** goes beyond any existing approaches to Discourse Analysis. While linguists can certainly accomplish discourse analysis without TextWorlds+ Analysis, they do not currently have at their disposal a method that matches a **model of Discourse Comprehension**, and that can explain more than local features of a text. What is missing in standard approaches to Discourse Analysis is an explicit, functional model of the context and active processing that is supplied by an active reader. In CDG/+, Werth follows a host of cognitive linguistics research that shows that **anaphora** (Cornish 2009), **deixis** (Hatch 1992; Lyons 1977; 1975; 1973), referential **opacity** (Bell 1973; David K Lewis 1968; Føllesdal 1961; Quine 1953), **truth-conditionality** (Semantic Leaps: Frame-Shifting and Conceptual Blending in Meaning Construction 2006b), **coherence** (Givón 1993) are all
challenging language phenomena that require the reasoning of an active human reader.

The CDG+ Discourse Processing Model thus makes it possible to study less prototypical language events like the written discourse of reader with writer that is negotiated via the text. So discourse is the ongoing interaction of readers making sense of texts in extended language events. In CDG/+, Text Worlds, Knowledge Frames, and Common Ground are simply artefacts of this conceptual exchange. Though the entire original CDG method cannot be fully described here, Figure 8.5.2.1 offers a conceptual outline of the major components in both the original and extended CDG/+ Model.

![Conceptual Outline of components in the original CDG model.](image)

**8.5.3 - CDG+ Discourse Model**

The novel CDG+ Context Model presented in the previous Section 8.5.1 provides visual context for the introduction of the CDG+ Discourse Processing Model, whose dynamics in turn produce the CDG+ Discourse Model. Figure 8.5.3.1 focuses on only the Discourse Model, while keeping the immediate
situation, the verbal situation, and the knowledge situation and the other components of the interaction situation in view.

Figure 8.5.3.1. An expanded view of the CDG+ Discourse Model in its position relative to the wider context of the CDG+ Context Model. The CDG Discourse Processing Model encompasses the dynamics of the interaction situation that draws exhaustively upon the verbal situation, selectively from the knowledge situation, and only occasionally from the immediate situation.

Whereas the original CDG approach was never explicitly positioned by Werth as a Discourse Processing Model for online language comprehension, my reading of both the current discourse processing literature and the entire Text Worlds monograph lead me to conclude that the CDG+ Discourse Model is a product of a
plausible (Forsythe and Xavier 2006; Forsythe and others 2006) Discourse Processing Model.

The standard CDG model is a schematized procedural model that is generated and modified in real-time by discourse participants in the Discourse Interaction Situation. The CDG+ Discourse Processing Model is a multi-level representation of the flowing referential dynamics between the knowledge, discourse interaction, and verbal situations. The CDG+ Discourse Processing Model has three primary structures and many supporting mechanisms. Primary structures include the Knowledge Frames, the Common Ground, and the Text World, which are introduced in this Section. The Text World is a graphical representation of a single view of the current state of the CDG+ Discourse Processing Model.

Looking at Figure 8.5.3.1, we can also now see how the Discourse Model, while a product of the Interaction Situation, actually draws extensively from cultural knowledge and the propositions contained within the text, and selectively from the real world to represent the conceptual structure of the text as knowledge frames, the Common Ground, and TextWorlds. The Ground of any Text World is its collection of world-building elements, which include times, places, doers, and objects. The Figure of any Text World includes all propositions that advance a (non-world-building) speech act function (Austin 1962), such as description of actions or objects, exposition, or argumentation.

A proposition is a basic sentence that asserts a minimal situation containing arranged objects or an event wherein agent acts upon patient or state changes. Each proposition is composed of a semantic predicate conjoined with at least one argument. A semantic predicate is a property, relation, or function (e.g. a verb, predicate adjective, or predicate nominal). An argument is an individual constant (e.g. Typically a proper name or definite Noun Phrase) or quantified class constant (e.g. All or Some). A semantic predicate may also be modified by further semantic predicates (e.g. Adjectives or adverbs, etc.). For example:

EXAMPLE 2: I saw everything clearly.
Representationally, both original Text Worlds and revised TextWorlds are structured and directed acyclic graphs that segment the source text into situations and aggregated meta-situations using custom-built language-based scene analysis. A text world represents the current state of affairs expressed by the propositions in the discourse. Propositions are represented as either path expressions between entities or descriptives that connect properties to entities within a world.

A sub-world represents any state of affairs that violates restrictions on discourse. For example, even in the briefest of sentences, due to the selection of any term which elaborates upon an unverifiable situation, from the reader’s perspective:

**EXAMPLE 3:** “I think…”

Thus, the reader or hearer is left with an individual’s’ report of an internal mental process. The reader must therefore accept the assertion, based on what they know of the speaker, or reject it, and assign it a probabilistic degree of falsity. The sub world exists to accommodate any modal expression, as well as a range of figurative expressions, such as metonymy and metaphor.
The original Text World diagram is a dynamic snapshot of a changing, textually-described current situation in an ongoing state-of-affairs. The Common Ground is a dynamic, artificial-intelligence-powered knowledge repository. It collects the propositional elements of successive situational snapshots, enables reasoning over them, and, using a neural network for activation, it also serves to filter ongoing discourse coherence. The activated propositions of the Common Ground represent the current state of affairs in the current and N recent text worlds. Interaction between reader and writer, mediated through the text during the process of discourse, creates interactions between cultural knowledge frames, the current
text-base, and the current Common Ground to dynamically update the current text world.

Thus, introduced entities and the propositions asserting facts about them are visually represented in a world layer and are simultaneously logged into the long-term, dynamic reasoning repository, or Common Ground, as and when they are accepted by participants in the discourse. The Common Ground of a text world thus contains not only entities and facts explicitly introduced by the text, but also the inferences and entailments of nominated entities and facts. Thus, the cultural, experiential, and other specialized knowledge of participants also contribute to the world-building process through inferences that help to define the conceptual domain within which all reference is determined.

Participant knowledge is accessed, collected, and represented as knowledge frames. Frames are conceptual structures that have slots for the generalized features that define a situation type. In the TextWorlds model, frames accrete whenever similarities between individual situational snapshots generalize into culturally recognizable situation-types and, in the context of discourse-processing, develop into culturally informed propositional knowledge frames. Inferencing over available cultural knowledge frames takes place within the specified world level. Thus, inferential conclusions must hold true in the given text world, though not necessarily in the real world.

When language is characterized in functional terms, each stretch of language is considered to be of a type (e.g. narrative, descriptive, discursive, interactive) that serves a language function, e.g. to advance the plot, scene, person, routine, argument, goal etc. Function-advancing propositions operate upon entities and
include the subsequent actions, events or arguments involving them, or any general predication made about them. All function-advancing operations take place within the text world domain; or, if the truth of propositions is inaccessible to the participants, within a dependent and limited sub-world domain. Participants are responsible for the truth, probability and authority of the content of the text- and sub-worlds that they have created.

8.5.4 - CDG+ Deictic Matrix

Ultimately, standard Text Worlds analysis does not have a satisfactory answer to the question of initial positioning and ongoing locative relations between TextWorlds (See Section 8.4.4, World Placement). The latter is a problem especially for longer texts, which are expected to have a greater number of SubWorlds and transition between TextWorlds. I have therefore re-examined and redesigned the structure and relation of the Systems in Figure 8.4.8.1 to reduce labelling redundancies and orthogonalize the relationships between these explicit (and implicit) dimensional systems.

![Figure 8.5.4.1](image)

*Figure 8.5.4.1. The revised CDG+ Deictic Matrix reduces labels and repositions the standard TextWorlds Tense and Remote Deictic System relationships.*

This deictic matrix includes the original four phenomena (Tense, Modality, Remoteness, and Accessibility) of Figure 8.4.8.1 and reduces redundant diagram elements while keeping aspects of the structural symmetry above and below the temporal-modal axis. However, the segment bounds of the vertical dimensions are not required to mirror each other. The Temporal system, above the line of symmetry may rotate left or right independently from the Modal system below. This modification supports the intuition that each existential or epistemic mode may, at different times be applicable in each Time Zone (See Figure 8.5.4.2).
The original ‘Tense Relationship’ diagram (Figure 8.4.8.1) clearly positions Text Worlds with reference to two dimensions: The Tense System which derives from the **temporal** dimension, and the Remote System which is derived from a generally **spatial** and specifically **linear** dimension. However, CDG/+ uses the generic **linear** dimension of distance as a template (Vandelanotte 2010) to situate ‘hedging’ as a class of distancing behaviors that are best understood as **social** deictic phenomena. So CDG/+ specifies Remoteness (renamed Proximity in the extended CDG+ approach) as a system for **social deictic** distancing.

Thus, two conclusions are apparent. First, in standard CDG, Tense and Remoteness/Proximity are considered systems of **deixis** (Lyons 1973), wherein elements are positioned with reference to some other point (Mey 2008; Fillmore 2006; Croft and Cruse 2004b; Zwaan 2003; Hatch 1992). A brief consideration of deixis is therefore relevant here. In language, the prototype, **spatial deixis**, is comprised of terms typically indicating **position** relative to the speaker (though that center can be moved by the speaker). Whereas **position** is fairly obviously **spatial**, other terms evoking relative proximity, distance, location, orientation, and direction are also included as spatial deixis (Dancygier and Sweetser 2012; Dancygier and Vandelanotte 2009). Second, recalling that Cartesian space is composed of individual dimensions intersecting orthogonally, analytically speaking we know that space is a composite dimension which requires two linear dimensions intersecting at right angles to form a plane and a third to provide height and
produce volume. Thus, the concept of a dimension may be viewed as an additive complex of lower dimensions.

Thus, distance, the single linear component of the spatial dimension, is used metaphorically in CDG+ as a template for the temporal and social systems of deixis (Dancygier and Sweetser 2012; Dancygier and Vandelanotte 2009). In applying distance as a metaphor, Werth follows (Lakoff and Johnson 1981) and a host of cognitive linguistics scholars since then. This approach has been applied to bridge from space to time by (Evans 2013; 2004), from time and space to social distance (Fleischman 1990; 1989), from time to epistemic distance (Dancygier 2006). Thus, given the wide availability of metaphorical extensions of distance in colloquial use, it is reasonable to make theoretical use of them.

Accordingly, standard CDG has capitalized upon the utility of employing the linear phenomenon of distance metaphorically (Levinson 1983) to situate and order other distinct areas of human experience, so as to make sense of reference. However, the standard CDG approach does not identify any further contextual dimensions, nor does it explore the relations between those it has identified.

Alternatively, the extended CDG+ approach employs the linear dimension metaphor to consider deictic and contextual dimensions beyond time, space, and society. Recall that the first step in the extended TextWorlds+ Analysis method, scene analysis, depends on the identification of entities belonging to the first four basic dimensions, which were listed as the temporal, spatial, physical and corporeal.

Upon inspection, however, the dimensions involved in spatial distances, areas, and volumes, physical matter and corporeal bodies are each noted as a dimensional complex composed of lower intersecting dimensions. This concept of the dimensional complex is the basis of a range of inferences. Each higher dimension can be assumed to be an intersecting complex of its lower dimensions. Thus, each time a physical object or entity is noted, we know by inference that it must have temporal and spatial properties, in addition to its particular physical features. And each time a biological entity is noted, it must have physical and spatiotemporal properties, in addition to its corporeal (e.g., bodily) features, and so forth.
Both the original CDG Text Worlds approach to deixis and the extended TextWorlds+ scene analysis critically depend upon the identification of elements as belonging to such contextual dimensions. Extended CDG+ proposes an iterative, prototype-based lexically-guided method for dimensional classification of terms (See 8.5.6, Manifest Semantic Analysis). Manifest Semantic Analysis initially employs a simple prototype classification judgment, then iterates that same task over each term in every definition of each sense of the original term, and ultimately offers a comparison between the initial snap judgment and a composite weighted ratio for each definitional sense to systematically determine a term’s prototypical dimensionality\textsuperscript{52}. However, since each definitional element (e.g., word or term in a definition) may have features which belong to various dimensions, dimensional classification is neither obvious nor simple.

And there are more dimensions to identify beyond the first four basic dimensions. Note that in Figure 8.4.8.1, the unlabeled origin of each rounded sub world in the above diagram is a modal expression or literary device such as desire, belief, purpose, hypothesis, condition, metaphor, metonym, flashback, negation. Thus, since each of these original sub worlds has its origin as an expression of mental or cognitive activity, it seems reasonable that the whole class of modal expressions could also be considered a Modal System derived as the fundamental product of the cognitive or mental dimension.

Cognitive Science and psychology have shown this fundamental cognitive system can be studied in terms of its bio-mechanical substrate (e.g., biological structures and chemical equilibrium) and electro-physical dynamics (e.g., neural activation) independently of its resulting cognitive products (e.g., judgments, choices, assumptions, opinions, emotions). So identifying, theorizing, and understanding the product is not the same thing as identifying, theorizing, and understanding its productive process. Consequently, revised CDG+ takes all SubWorld building modals as expressions of cognitive activity that are best understood as unique mental products that express an embodied, physical point of view in space-time; or perspective. The perspectival dimension is thus a composite of lower dimensions of space, time, matter, body, and mind. Perspective is a requisite for higher social, linguistic, semantic and mathematical dimensions. In CDG+, the Modality system is thus considered to be derived from the perspectival
**dimension**, wherein a required cognitive substrate is activated to evaluate and assign epistemic and existential states, like truth, certainty, actuality, or falsehood, probability, or possibility.

Thus, extended CDG+ has identified what it considers to be the smallest number of dimensions required to situate and ultimately to comprehend both relative and absolute human references. The extended CDG+ Contextual Dimension system includes the temporal, linear, planar, spatial, physical, corporeal, mental, perspectival, social, linguistic, semantic, and mathematical as contextual dimensions. Unfortunately, the deictic matrix does not provide sufficient space to accommodate each of these experiential and conceptual dimensions of context.

### 8.5.5 - CDG+ Meta-Contextual Frame

This limitation became the impetus for a search for a simple yet extendable representational design for meta-contextual space, that could not only systematically relate the major (e.g., proximal, temporal, and modal) deictic systems but also accommodate the wider range of contextual dimensions.

To review, the assessment, evaluation and judgment of the **Proximal, Temporal,** and **Modal Deictic Systems** is an important aspect of textual analysis during analytical reading that is currently not well supported by analytical tools. More generally, recall that in the preparatory and skim reading phase, analytical readers and tool-users frequently make many shallow judgments that are not captured explicitly, yet which are important to further processing or decision-making. Those shallow analytical judgments can be captured in Tags and Highlights, as was prototyped in Chapter 7, \( m^3 \) [em-cubed]. Thus, not only should analytical tools support the capture of important evaluations while the reader is engaged in the deeper phase of analytical reading, but great care must be taken to enable the kind of judgments that are actually required.

In light of the wide linguistics literature on modals and modality, which has established that the phenomena of modality are of great importance in language usage, the original CDG approach to Text Worlds analysis should thus be extended to capture these judgments of modality. This is a natural fit, since CDG’s treatment of modality (e.g., as creating sub-worlds) is one of the core contributions of original
Text Worlds approach. Modality in CDG thus relies upon the most important contributions of original CDG, which are 1) that Worlds define referential scope of utterances, and 2) that Participants evaluate multiple deictic dimensions, not just a single modal dimension, through probabilistic evaluation of truth and speaker authority. In essence, those deictic judgments are core multi-modal meta-data for each utterance. Thus, the efforts to capture shallow analytical judgments in m$^3$ [em-cubed] are related to efforts, here, to capture deeper meta-contextual judgments in deeper analytical reading. While such judgments are central in the original CDG approach, no mechanisms are envisioned or provided to capture those individual judgments on an ongoing, emergent, individual basis. Rather, an aggregate of those judgments is simply collated as the Text World Diagram. This deficit is addressed by CDG+ Meta-Contextual Frame, which is a mechanism for capturing deictic and contextual dimension judgments for individual terms (Marin Arrese 2015).

Recall that CDG/+ (e.g., both original and extended CDG) draws upon cultural knowledge Frames, which on one view, are simply statistical composites of multiple TextWorld meta-situations. TextWorlds express a similar composite structure, as they are composites of multiple scenes, or TextWorld Views, which are transient local situations that provide a selective contiguous view into the TextWorld. Since CDG/+ implicitly relies upon the concepts of frames, worlds, and views, it seems reasonable to begin the search for an extensible representational design for meta-contextual space by first looking at how frames, worlds, and views are generally represented conceptually and visually. I then consider how their cognitive models represent context.

What is a frame? Physical frames are typically thought of as a particularly large, shallow box, usually with one large central hole in it, that either contains some material planar projection, or affords a view from within one structure onto its exterior, like a window. What is a world? At the macro level, physical worlds are planets, which are roughly spheres. What is a view? From the local, physical human perspective on the surface of any such giant planetary spheres, worlds are flat and locally circumscribed by the horizon; or more commonly by proximally intervening structures in the natural or built environment. The horizon’s relative visual aspect is a function of the viewers’ absolute distance from the water-level surface of the planetary sphere in relation to any intervening structures. So in purely
physical terms, the spatial & physical position of the viewer determines the breadth and scope of an individual’s spatio-physical point of view relative to surface features of the world (e.g., altitude and other large-scale physical objects). For example, you cannot see much from a position the bottom of a 10-ft hole, in a dense forest, or in a classroom, whereas a position atop a ship’s crow’s-nest, skyscraper or mountain easily affords you a wider vista. Thus, a view is a circumscription or abbrevi√cation of the world to the immediate perceptual and experiential milieu. A frame similarly and further circumscribes the view with an opaque structure that serves to highlight and focus attention on a relevant portion of the view. In this way, a room can frame a screen or window, or a city skyline can frame a mountain vista.

Finally, visual representations of cognitive frames typically have slots for expected data, some of which are occupied with default values, and are minimally represented as a square with cells or fields (e.g., as a table) (Karp 1992). Visual representations of cognitive TextWorlds and TextWorld Views have both slots, nodes, and links; slots for background contextual details, nodes for entities, and links for active and descriptive relations. In both cases, visual representations of context look something like a slotted box.

Alternatively, rather than recreating another flat wrapper for flat content, or another slotted box, in this section, I explore the possibility and implications of an isomorphic three-dimensional meta-contextual frame for annotation of higher-dimensional content. A successful design of a 3-dimensional frame should support the intuitions of containment, focus, and viewpoint yet also provide a metric space that provides sufficient space for each contextual dimension and any entities belonging to them (Evans and Chilton 2007; Chilton 2014).

8.5.5.1 - Meta-Contextual Frame Design

My design for a contextual frame begins as a ball. The surface of the ball is rotated isometrically (Guiard and others 2006) and is then trisected by orthogonal cutting planes to excise slightly more than one quarter of one hemisphere. This simple tessellation affords a minimally obstructed view of the interior space. When viewed from the specified angle, the remaining exterior surface of the ball constitutes a rounded triangular frame onto the largest interior circle (e.g., the
bounded plane that segments the ball into top and bottom hemispheres). The frame contains a three dimensional Euclidean metric space that centers upon this hemiplane. This interior space uses Cartesian co-ordinates to locate content as points. The interior surface of the ball demarks the three-dimensional horison of this interior space.

The CDG+ contextual frame begins empty, yet provides orthogonally intersecting dimensional structure as conceptual scaffolding (Figure 8.5.5.1.2). One open quadrant of the top hemisphere is always in focus (e.g., facing the open...
frame), while the remaining three remain highly visible but not focal. The three primary axes intersect with the horizon at six polar points, each of which is visible through the open frame. Two are recognizable as the Topmost and Bottommost Poles (e.g., North and South). The remaining four demark absolute quadrants on the circumscribed hemiplane. In combination, cutting planes extended from these orthogonal axes define top and bottom hemispherical quadrants. The axes begin labeled for the three deictic systems, Temporality, Proximity, and Modality.

The three major dimensions meet at the center of the framed space. It is from this point, the Origo, that the speaker makes their utterance. While this concept is

Figure 8.5.5.1.2. The CDG+ contextual frame is an extension and formalization of the temporal, proximal (e.g., remote) and modal deictic systems found in standard TextWorlds.
noted in standard CDG, it is not conceived of as the point at which all relevant dimensions of context, and particularly where the major deictic systems, intersect. Nevertheless, in both standard and extended TextWorlds, the Origo is the center of all deictic or relative reference.

The full three dimensional space of the top hemisphere may be used to organize content. However, since these axes intersect orthogonally, any use of this vertical space must respect the metric of that vertical dimension. And since the Proximal

Figure 8.5.5.1.3. In standard TextWorlds, the speakers’ contextual position is called the Origo. However, in standard TextWorlds, the temporal, modal, and proximal deictic systems are not conceived as intersecting to produce both the Origo and an extensible multi-dimensional classification space.
deictic system (Figure 8.5.5.1.7) is a reduction of the spatial dimension to a single linear dimension that is metaphorical applied to distinguish further dimensions of human experience, another focal dimension must always be specified for proximity, to determine which kind of distance or proximity is specified. Otherwise, proximity will assume its prototypical spatial distance upward (e.g., height); whereupon the surface of the plane should be deformed to reflect an isomorphic mapping of the speaker’s position and the elevation of the immediate

\[
\text{Proximity}
\]

\[
\text{Origo}
\]

\[
\text{Temporality}
\]

\[
\text{Modality}
\]

*Figure 8.5.5.1.4. While standard TextWorlds does not consider further deictic or contextual dimensions, the extended CDG+ contextual frame accommodates both the major deictic system (e.g., Temporal, Proximal, Modal) and the full range of experiential dimensions of context.*
natural or built physical environment.\textsuperscript{54} However, the Proximal system of the CDG+ contextual frame was designed to accommodate more abstract contextual features using a unique relative metric for each temporal, linear, planar, physical, corporeal, mental, perspectival, social, linguistic, semantic, and mathematical distance, so the hemiplane does default as a map of the speakers spatio-physical location.

The hemiplane horizon accommodates each of these experiential dimensions of context (Figure 8.5.5.1.4). Thus, each of the minimally necessary\textsuperscript{55} contextual dimensions takes a point along the hemi-surface horizon that is equidistant from its

\begin{center}
Figure 8.5.5.1.5. The group of mental requisite dimensions of context. Entities, Actions, and Properties classified in this region require a mind.
\end{center}
neighbors given the selected total (e.g., N=10) of dimensions. Extending from that point through the Origo, each dimension also intersects the horizon on the far side of the interior surface. Rotating any contextual dimension toward the Proximal defines a unique circular segmenting plane that intersects both the Origo and the topmost and bottommost poles at a specified angle of rotation from the previous dimension. This is the proximal prototype plane of each contextual dimension.

Each ascending dimension of context is construed as a composite that includes each lower dimension and extends beyond them in some critical way. Thus, space is composite of a three lower dimensions, two that intersect to form a plane,

*Figure 8.5.5.1.6. The group of mental non-requisite dimensions of context. Entities, Actions, and Properties classified in this region do not require a mind.*
intersecting with one more (e.g., height) to produce volume. Similarly, the physical
dimension adds matter to space, the corporeal adds life to matter, the mental adds
cognition to biology, the perspectival adds modality to mind, the social adds others
to judgment, the linguistic adds language to interaction, the semantic adds meaning
to discourse, the mathematical adds numeracy to literacy, and the temporal adds
indexicality to experience. Thus, contextual dimensions can be classified according
to whether they require a given dimension of context (e.g., the group of mental
requisite dimensions) or whether they do not require the given dimension (e.g., the
group of mental non-required dimensions.)

Figure 8.5.5.1.7. The CDG+ contextual frame centers upon prototypical core values of the
three major deictic dimensions and extends through to their polar extremities.
Each deictic system is typified by its core dimensional values, which define the immediate context of the Origo. These core dimensional values not only prototypically situate the speaker temporally and spatially in the here & now, but also by epistemic and existential degree in the modes of actual, real and true events. This accords with the default mutual assumption of interlocutors that everyone will speak truthfully about actual, real events, starting from the here and now, and refer into other temporal, modal or remote situations only as required. So the whole hemiplane is dedicated to determining the range of existential and epistemic modal features for each included element in discourse, with reference to the core assumption that the world (and thus the hemiplane) is a true account of actual events. Nevertheless, some features of a true account may not be known with certainty, or may be, to some degree, false. Thus, after the initial challenge of determining these temporal, multi-dimensional-proximal, and modal meta-features for each discourse element, there is the ongoing challenge of tracking reference in cases where these temporal, proximal, epistemic, and existential modes change as discourse unfolds. Any such changes indicate a new TextWorld and thus require a new hemiplane. (Refer to Figure 8.5.5.1.13, for the interactions and dynamics of hemiplane transitions.)

Each major deictic system has one or more core values, which must be extended along the appropriate axis in order to determine the semantically related values of its polar extremities. The quadrant segmented hemiplane provides four concentric radial zones for the immediate (e.g., components of the Origo), proximal (e.g., components of the concept), distal (e.g., opponent concepts), and negated (e.g., ‘non’-concepts) semantic relations of each term. So the Origo is defined by the intersection of its immediate deictic and experiential component dimensions. For the purposes of illustration, that point is extended as the area around the central point of Origo. For example, the extremities of the temporal dimension of deixis extends backward from the now into the past (Figure 8.5.5.1.11) and forward from the present towards the future (Figure 8.5.5.1.11).

So the Present Time Zone is the first radial band from the center (Figure 8.5.5.1.8), while the Future and Past Time Zones respectively occupy the forward and backward halves of the second hemiplanar radial band. These past, present, and future Time Zones are reflected very explicitly in the tense system of most
Figure 8.5.5.1.8. Now contributes the temporal dimension to the Origo. While actually a single point, this point is expanded to an area for the purposes of illustration.

Figure 8.5.5.1.9. The Present Time Zone occupies the first radial band from the center and thus encompasses and extends beyond the Origo Now point. It is also defined along the transition point between the Past and Future Time Zones at the line demarking the deictic system of Modality.
Figure 8.5.5.1.10. The Past Time Zone occupies the backward half of the second hemiplanar radial band.

Figure 8.5.5.1.11. The Future Time Zone occupies the forward half of the second hemiplanar radial band.
Figure 8.5.5.1.12. The Temporal dimension of deixis with reference to the speaker’s Origo.

languages. For example, completed, ongoing, and potential actions each explicitly specify temporal deixis, whereas purely descriptive momentary states, lasting conditions, and transitory manners provide further oblique temporal cues that contribute to both the definition of the setting as world-building elements and the description of action as function-advancing language elements.

As discussed earlier, the extremities of the proximal dimension are prototypically derived from the linear dimension with comparative reference to the speaker’s spatial location. The proximal dimension thus has the widest application, since it is used metaphorically in each linear or polar comparison between two or more related terms (e.g., opposition, antonymy, meronymy etc.).
Recall that the proximal dimension (See Figure 8.5.5.1.13) intersects with each contextual dimension to produce a unique prototype proximal plane for each dimension. This proximal plane collects strongly prototypical values and attracts peripheral values. For example, the social deictic phenomenon of politeness is something like a reaction to a perspectival judgment that softens the modal force of a statement from certainty to probability. Thus, social distance is noted along the social proximal plane (See Figure 8.5.5.1.14).

Alternatively, three of ten contextual dimensions coincide with the three major deictic systems. These major deictic dimensions are 1) the temporal dimension of context, which corresponds perfectly with the deictic system of Temporality; 2) the
linear dimension of context which corresponds but does not overlap with the deictic system of Proximity, and 3) the perspectival dimension of context, which corresponds perfectly with the deictic system of Modality.

Thus, the perspectival dimension of context is isomorphic with the deictic system of Modality which prototypically assigns existential/epistemic meta-featural values. The perspectival/modal dimension may also be used to classify any Entity, Property, or Action as the product of the cognitive substrate (e.g., as a sensation, sensation, 

Figure 8.5.5.1.14. The Perspectival Proximal Plane, which describes perspectival distance. The major deictic dimension of Modality coincides with the contextual dimension of Perspective. Thus, Perspective, or any other contextual dimension, can be selected in conjunction with the major deictic dimension of Proximity, to specify the kind of proximal/remoteness cline intended for any particular designation of remoteness.
opinion, idea, concept, attitude), as the feature belonging to such products (e.g., sensorial, opinionated, ideational, conceptual, attitudinal), or the productive act itself (e.g., sense, opine, ideate, conceive, express attitude). This or any other dimension of context may be selected in conjunction with the deictic system of Proximity to describe tailored variant of distance along the selected dimension of context.

The core modal system prototypically weaves the **existential** and **epistemic** modal clines together (Figure 8.5.5.1.18) while also accommodating other modalities (Figure 8.5.5.1.22). While the core modal values of **actuality** and **certainty** (Figure 8.5.5.1.15) are not coextensive along their entire ranges, any claim of actuality usually carries the force of an implicit claim of certainty, unless specified otherwise. While **actuality** is defined at the point of intersection of the **modal** and **temporal** systems, in the current CDG+ Meta-Contextual Frame, that point is expanded to an area (Figure 8.5.5.1.15) for the purposes of illustration. Since **actuality** applies not only to **present** events, it is also defined along the line demarking the deictic system of Temporality, along the Past and Present Time Zones and into the future.

![Diagram](image)

*Figure 8.5.5.1.15. The core existential and epistemic modal values situate the speakers’ Origo prototypically with reference to Certain knowledge of Actual events.*
Figure 8.5.5.1.16. Probability is the mid range of the Certainty scale.

Figure 8.5.5.1.17. Possibility covers a wide area that includes Actuality, Probability and Improbability.
Each core modal value (e.g., actuality & certainty) has a unique polar value. The pole for **actuality** is **possibility** and for **certainty** is **probability**. The range of **possibility** (Figure 8.5.5.1.17) covers anything that could have happened, could be happening, or could happen, whereas **actuality** (Figure 8.5.5.1.15) covers everything that has happened and everything that is. Possibility thus marks the lower range of values on the Actuality scale, while the temporal dimension and the Origo mark maximal actuality.

All events that have actually occurred at some past time are possible and have a probability. However, probability ultimately describes anything less than 100% certainty as regards the **truth** or **believability** of past, present, and future events. For
Figure 8.5.5.1.19. Improbability marks the lower end of the Certainty scale.

Figure 8.5.5.1.20. Impossible events have an extremely low probability, whereas non probabilistic events have no assigned probability.
example, most events today will occur beyond the scope of my epistemic access. I will accept reports of these events, qualified to whatever degree I accept the claimant’s knowledge and trustworthiness. Probability thus reflects the common degree to which I will believe or support both the facts of my own experience and those presented by others. This will be some value less than 100%, unless the judge has reasonable epistemic grounds to claim certainty.

Though these core modal values have unique polar values, their overlap at the Origo enables the formation of a prototypical composite three-valued system of modality (Figure 8.5.5.1.18). The actual-possible and certain-probable dimensions

![Figure 8.5.5.1.21. For each core value, opponent values are captured in the second radial hemiplanar band. Lexical opposites may be noted in the third radial band. Negations are stored in the outermost band that intersects with the hemi-planar horizon.](image-url)
are lapped to produce the primary prototypical possible-actual/certain-probable modal system. For each term occupying an interior position, the contextual frame reserves radially adjacent spots in the *periphery* for semantically related terms indicating Opposition or Negation, each in their appropriate hemiplanar radial band. The reserved location for opponent terms is in the second radial band (Figure 8.5.5.1.16 and 8.5.5.1.17).

*Figure 8.5.5.1.22. The existential-epistemic Modal dimension of deixis prototypically situates discourse contents, with reference to the speakers’ Origo, as Certain knowledge of Actual events. However, many other Modal dimensions exist, such as Truth, Belief, Realization, Verification. In the Meta Contextual Frame, each Modality is attracted to its prototypical contextual dimension. Thus, most modals cluster towards the mental and perspectival dimensions. Some, like doing and realization include a component of making which can be accomplished through physical, corporeal action. Others, like verification, have a strong social component.*
Negations are captured in the outermost radial band, at the hemiplane horizon. In fact, negations are points on the horizon line, though here that line is extended into a radial band for illustration. Non probabilistic events have no assigned probability value, whereas improbable events have low assigned probability. Impossible events generally have a probability approaching zero. In general, semantic opponents and negations are not displayed at all, unless they are determined to be relevant by the reader.

The major deictic system of Modality thus has a composite existential/epistemic dimension that matches the three-valued dimensional values of the deictic system of Temporality. This allows a single composite modal value to be assigned. Alternatively, the entwined existential and epistemic modal dimensions can be segmented normally at the Origo, and independent existential and epistemic modal values may be assigned.

Beyond the core modal dimensions of Actuality and Certainty, several other important modal dimensions exist, including Factuality, Truth, Belief and Reality (Figure 8.5.5.1.22). The semantic opponents of these core terms can be arranged temporally, as completed vs. ongoing vs. potential features or actions. They can also be arranged by their primary or prototypical experiential dimension. For example, since verification of true events requires external corroboration of judgments by other people, usually using language to determine factuality; truth and factuality are considered and positioned as a sociolinguistic modality.

Thus, the core deictic network relates the core deictic values of the Origo to each other and to their respective opponent terms (Figure 8.5.5.1.23). The primary deictic system and wider dimensions of context are positioned so as to intersect at the Origo, with polar extremities of these dimensions defining points along the circumference of the great circle of the sphere.

The intersecting temporal and modal dimensions of the hemi-plane provide dedicated areas for classifying terms by selecting a value along each dimension (Figure 8.5.5.1.24). Recall that the hemi-plane surface is segmented by radial bands which produce a set of semantic relational slots for each dimension. Each dimension reserves its slots to accommodate specific, relevant semantic relations of the classified term.
Any term that is placed in this space will be an Entity, an Action (relation), or a Property (adjective or adverb). Temporal-modal meta-features of Actions are identified primarily in terms of their Potential (e.g., future), Continuation (e.g., present), or Completion (e.g., past). Specified temporal slots are reserved for terms with these meta-featural values. However, these temporal slots overlap with a differently-shaped set of modal slots reserved for existential/epistemic values. These overlapping modal meta-featural values range from Possible, Actual, and Probable.

Figure 8.5.5.1.23. The core deictic network includes the Origo-defining core values (bold) and their opponent terms. For each of the three major deictic systems of Temporality, Proximity and Modality, the major dimension is defined by both its core and polar extremities, as opponent-core-opponent values. Negations are only included in the outermost radial band as required.
Primary slot labels for Actions are Continuation for the first radial band, Potential for the half of the second radial band that extends into the future time zone, and Completion for the half of the second radial band that extends into the past time zone (Figure 8.5.5.1.24). The outer two radial bands are reserved for secondary slot labels, which match those used by all other types of term (Entities, Properties). The outermost radial band is always reserved for Negation. The second radial band is always reserved for Opposition. The first radial band also takes the term type as a secondary slot label (e.g., Action).

Figure 8.5.5.1.24. The radial bands of the hemi-plane have dynamic labels that take their slot label based on the type of term being classified. Actions are classified into layered temporal-modal featural slots whose labels are based primarily on their temporal values.
In practice, Temporal and Modal meta-features are somewhat intertwined. Past actions are complete in time whereas present actions are ongoing. Both are therefore possible, no matter how improbable. Future actions range from impossible (0% probability) to improbable (e.g., some indeterminate near-0% non-0% probability) to possible (e.g., indeterminate non-0% probability) to probable (e.g., some indeterminate near-100% non-100% probability) to certain/actual (e.g. 100% probability).

Certain properties are produced by the completion of actions, while others are entity features that range from unchanging permanence of innate qualities to the

![Diagram](image)

*Figure 8.5.5.1.25. The Meta-Contextual Frame uses concentric radial bands to classify entities, properties, and relations. States are point samples of underlying conditions, past and present.*
dynamic perdurance of *ascribed attributes*. Properties are thus derived from what is done, what is, and what is said of what is. Properties may have their own temporal qualities pertaining to their own Potential, Continuation and Completion. Similarly, properties may have their own modal qualities, pertaining to their own Possibility, Actuality, or Probability. Thus, potential properties (Figure 8.5.5.1.24) are likely features based on previous experience.

Properties are also further classified temporally according to their duration. Thus, **Conditions** (Figure 8.5.5.1.27) are spans that indicate ongoing actual states of being in the here and now (corresponding to Continuation), while **Manners** (Figure 8.5.5.1.26) are usually ongoing states of acting that typically match the span of the

![Diagram of Temporality and Modality](image)

*Figure 8.5.5.1.26. Manners are ongoing states of action whose span matches that of the action.*
action. In this conceptual system, a **State** (Figure 8.5.5.1.25) is a point, or a momentary sample from an ongoing condition. While the point values of states are epistemically agnostic as to the underlying truth and temporal meta-values of the ongoing condition, they are quite often used of entities or objects whose probable current condition may be inferred from a past state value. (e.g., The light switch was turned off. [Assuming no-one else has turned it back on, the light remains off]).

Thus, in the Meta-Contextual Frame, assignment of a term to any specific point on the hemi-plane amounts to the assignment of four or more meta-featural

![CDG+ Contextual Frame](image.png)

*Figure 8.5.5.1.27. The CDG+ contextual frame affords position-based classification in multiple dimensions simultaneously. Each point on the hemiplane denotes a unique 5-ary value that specifies a temporal, existential/epistemic, experiential/contextual and proximal value assignment.*
contextual values, with at least one from each of the Temporal and Modal dimensions. Since the default value for the Proximal dimension is set to the Origo [0] for all points on the hemi-plane, further meta-featural values are selected from the highest (e.g., most complex) contextual dimensions that best describes the term, in the reader’s informed opinion. Or, if the term is determined to lie between subsequent contextual dimensions, a partial value is calculated for the two closest contextual dimensions, that accounts for its relative position between those two dimensions.

Recall that the hemi-plane surface is segmented by radial bands which produce **semantic relational slots**. Both Actions and Properties divide antonyms into two classes: semantically related oppositional terms and lexically related negations. Though basic, these are extremely important semantic relations that should not be confused. Opposition connects terms with morphosyntactically distinct lexical entries (e.g., active vs. passive), whereas negations typically uses a negative affix or a not-construction onto the original positive term (e.g., non-active, inactive). Often opponent terms, or opposites are used as antonyms, which have a relatively low value attribution while negations indicate a more absolute non-value attribution.

The CDG+ contextual frame provides structured conceptual organization and unique affordances for tangible multi-dimensional classification. Rotations about any of the three axes shift the internal space, but not the frame itself. So the frame remains, while the axis labels and contents shift. The user simply swipes left or right, up or down along the frame to rotate the internal space.

**8.5.5.2 - Remote Compression & World Perspective Adoption**

Other touch-based interactions are also possible. If a lexical cue or a perspectival judgment registers the situation as diverging from actual truth, for example as an epistemically inaccessible modal world, the entire world can be shifted along the proximal dimension. Thus, the CDG+ contextual frame allows the user to compress layers upward or downward. To reclassify the current hemispherical metacontextual space in remote context for a specified dimension, the user simply has to swipe upward along the Y dimension, and the whole hemispherical working surface is compressed as it is relocated along the Proximal dimension.
My use of the term COMPRESS emerged from playing with the constraints of the spherical contextual model I designed. In any sphere, the movement of the hemiplane along any core dimension reduces the area of that plane, if the inner surface of the sphere is used to constrain the plane and disallow any extension beyond the sphere. Between the choices of clipping or compression, I chose compression. Though the term compression is also used as the primary mechanism

*Figure 8.5.5.2.1. The user swipes upward from the Origo to compress the hemiplane into a remote parallel plane.*

339
in conceptual blending theory (Dancygier 2012; Dancygier and others 2012; Dancygier and Sweetser 2012; Dancygier and Vandelanotte 2009; Dancygier 2006), it is used here in the sense of a physical reduction of scale of the meta-contextual hemiplane, resulting from displacement along the primary deictic dimension of Proximity to create a remote world.

Figure 8.5.5.2.2. Dragging upward moves the workspace against a virtual spring that provides resistance and settles at a 70% of the original hemiplane, occluding only the rear quadrant.
Werth talks about ‘remote worlds’ and suggests that most discourse regards remote worlds. While distancing concepts are widely cited in the linguistic analysis literature, Werth does not define remoteness, nor does he specify its precise role or dynamics. In CDG+ remote worlds are worlds whose position on the proximal dimension of deixis has an extreme value on the proximal scale. Thus, since remoteness is just a state assigned to a World based on a position on the proximal scale, and the proximal scale requires the selection of further relevant dimensions of context to be properly specified, a proper understanding of remoteness requires the addition of another dimension of context to have precise meaning.

Figure 8.5.5.2.3. Contextual ground for internal ‘remote’ sub-world.
Thus, when speaking from the Origo about the here and now, the world created by the speaker is not remote (rather it is the definition of proximal). But a speaker whose here and now is that of Tibetan monks circa 1910 is remote in time and space, to me, the hearer/reader. And the here and now of space pirates operating from a base on the third moon of Quernon is even more remote, in even more dimensions (e.g., not actual, real, possible or even probable to me; but actual, real, possible and true in the story).

Since any discourse can be about any historical or fictional world, and not necessarily or even usually of the real world, some degree of remoteness applies to all non-prototypical discourse (e.g., here and now, real world) references. This degree must be assessed by the participants in the discourse, and ultimately, by the reader, since the writer has had their turn and made their assessment known in their statement.

So the meta-contextual frame must be able to accommodate the fact that, from the reader’s perspective, most worlds will not be located on the hemi-plane (which is reserved for real-world, here & now, actual, real, true accounts, events stories etc., and more generally for accounts given from the here & now). So, in order to register the world as a fiction or history, the user can “tag the world,” or assign the appropriate degree of remoteness to the world by compressing upward along a single or multiple dimensions of context.

Downward compression from the hemi-plane has a different function. Swiping down below the great circle archives the world state, regardless of its remoteness status. This keeps the ‘basement’ of the contextual frame for a compressed sequential history of views, which can be accessed and navigated to access earlier points in the narrative/discourse.

Thus, worlds classified/tagged as remote will stay suspended (by degree of proximity/remoteness, above the hemi-plane) so long as the reader elects to maintain their own true deictic perspective or decides to actively maintain the story’s objectively true relation with objective-here-and-now reality, rather than suspending their own disbelief and allowing an acknowledgedly remote story overtake their own actual, real, certain, true position. So the reader chooses when, and when not to adopt the Story or World perspective.
However, from the STORY or WORLD perspective, each world has its own internal consistency and accuracy and currency, regardless of whether, to the hearer or reader, it appears remote. Thus, each world, after it has been classified by the reader as remote along some dimension(s), can be engaged on its own merits. Typically, even remote worlds will presume their own ‘actual real certain true’ position; because even stories remote in time are about people, who seem to presume truth. So the historical experiences of Tibetan monks can be discussed accurately or inaccurately. The worlds of Star Wars or Harry Potter each have their ground truth. Lies and mistakes are possible in each.

Adopting the remote WORLD perspective happens whenever the reader suspends their disbelief. And when a reader adopts an alternate deictic WORLD perspective, all the details of that world can be shifted back to the hemi-plane, as accepted as actual, real, true to that world. The contextual frame facilitates this adopted perspectival state of accepted non-reality or accepted remote-reality by decompressing the world back to the Origo (or to within a few units from the Origo). Whenever marked and returned to the hemi-plane, the view (e.g., through the triangular frame) into the world is widest, and interaction is most easily accomplished.

So after the reader has assigned the world a degree of remoteness (compressed upward from the hemi-plane) and adopted the world perspective (decompressed downward back to the hemiplane), the remote world indicates its meta-contextually remote status by change of color, or by some other perceptually unavoidable cue highlighting its divergent/accepted deictic status. Remote worlds can have remote worlds (e.g., histories, possible futures, fictions, errors, thoughts etc.). The extended method of TextWorlds+ Analysis calls these SubWorlds, and identifies and discusses SubWorlds primarily in relation to modal expressions.

8.5.6 - CDG+ Manifest Semantic Analysis

The CDG+ Meta-Contextual Frame provides a novel interface for needed forms of user-driven content evaluation, including probabilistic truth assessment, evaluation of speaker authority & reliability, and multi-dimensional meta-contextual judgment of the highest primary and peripheral contextual dimensions for each term.
This section begins to explore the possibility of using such content evaluations to automatically validate automated sense selections. The idea is that such user evaluations could be used to create an attested link between an evaluated term (as present, proximal, true, trusted, social) and any evaluations in the immediate discursive context of currently activated propositions in the Common Ground. Say, for example, the user was easily able to assign their personal dimensionality assessment to each word in a text, in sequence. In this case, each evaluative judgment is a sort of spot-classification task. However, in this case, this judgment task is not context neutral, because the previously activated words in the Common Ground can be expected to have a resonant effect on the classification task. The chief characteristics of this experimental evaluative context, then, is that it is a real context of language usage.

So then, following this approach, each word in the CDG+ Common Ground would be assigned a dimensional level; and each of those judgments would be contextually related. At various scopes, in both traditional linguistic units (e.g., phrase, sentence, paragraph, section, chapter, book, etc.) and also in the proposed alternative CDG+ Common Ground propositional units, these individual judgments could then be aggregated, weighted, and ranked to the unit. Thus, any term requiring disambiguation would have both a contextually related spot judgment of that term’s highest relevant dimension. Moreover, a composite, weighted, and ranked unit judgment representing the immediate context of the ambiguous term can be calculated for a variable-sized unit of text (e.g., based on the requested unit scope, in words, propositions, sentences).

Next, consider an alternative evaluative context, wherein terms are presented for classification with resonance inhibiting intervals. In this evaluative context, we may safely assume that each dimensionality judgment is independent of every other. This judgment task would be as close as possible to context neutral, though clearly many internal and external variables will affect human judges.

Then, as a final evaluative context, consider a case where the classification of a term is immediately followed by the classification of each term in each definition of each sense of that initial first term, in sequence. If a dictionary, (or WordNet) has only one definition for a given term, then the single definition is used. If the lexical resource has more definitions for different senses of the term, each is processed in
sequence. This, then, is the prototypical case, where the judgment is not cued by real world language use nor is it insulated from it. Rather, the *evaluative context* for this judgment task is based on the abstracted, *shallow context* of a term definition.

Upon completion of this shallow-context sense-bundle contextual judgment task, two things (at least) become possible. The first and simplest possibility is that, upon a more holistic reflection on the entire definition (including each sense), the user could decide to revise their opinion on which dimensional level should receive their vote as best characterizing the original term (as a class that holds true despite the variability of classes characterizing each sense of that term). The second and most important possibility is that, for the reader facing an ambiguous term (or for the AI system having selected a sense), a ranked unit judgment representing the immediate discursive context of the ambiguous term can be compared to the ranked unit judgment representing the shallow context of each possible sense of the ambiguous term. The sense selected could thus be confirmed or corrected (automatically or manually) based on 1) easy (low-cost) continuous assignment of highest dimensional level of terms, and 2) a sufficiently wide, previously established sense-bundle prototypical contextual classification pool.

The clear presumption here is that the immediate context of any ambiguous term requiring sense selection (e.g., the terms used around the ambiguous term) will manifest a dimensional signature that can be appropriately evaluated and approximately matched against the relevant dimensional signatures of the definitional context (e.g., the terms used to define each considered sense of the ambiguous term) for the possible senses. In this way, this method shares some assumptions with Latent Semantic Analysis. LSA uses the paragraph as the standard unit for comparison. Manifest Semantic Analysis employs another unit, derived from activated terms in short-term memory (e.g., an activation window of N terms from the Common Ground) to produce a different result.

Initially, lexicographical resources that contain ‘usage context exemplars’ for each term may prove helpful, though few such examples are typically attested per sense. CDG+ MSA, then, would be an automated way of creating a lexicon of attested examples for each sense selection. A larger personalized collection of attested senses should eventually prove more useful for helping automate correction of sense-selections.
Thus, initially, this method will require two sources of classification data. This method first requires the continuous classification of each term in a text sequence, whereby the reader (or an active learning system) evaluates each term to assign it to the highest available contextual dimension that the (human or computer) judge feels best prototypically situates that term. This method anticipates a significant difference between classifying the ascendant dimensionality of a term in the generalized context-neutral condition (e.g., spot classification) vs. the prototype dimensions in the shallow contextual condition (e.g., classifying components of definitions) vs. the deep contextual condition (e.g., classifying components of the current contextual situation from the Common Ground). We hope to use that difference to simplify the correction of sense selection to a single judgment of the current term’s highest relevant level of dimensionality.

Additionally, this method requires the execution of an extensive background project wherein dimensional judgments are applied first to terms, and then to each term in the definition of each sense of the initial term. Context neutral judgments of the highest relevant dimension (e.g., the ascendant class) of the first term are then evaluated against an aggregate representation of all judgments of all terms in the definition of each sense of the term; so a weighted multi-dimensional signature can be generated for 1) each sense of the term, for 2) the sense-bundle (e.g., the group of all sense definitions) of the term, and eventually also for 3) the synset-bundle (e.g., the group of all synonym senses) of the selected sense of the term. Fit between the ‘ascendant’ (e.g., highest relevant) and ‘prototypical’ (e.g., most frequent) context neutral sense-assumptions of users can then be weighted against either the sense-bundle (e.g., all the senses for the term) or just a single preferred sense. The evaluation algorithm could also be tuned to include the definitions from synset matches, and from other semantic relations in a lexicographical resource (e.g., parts, wholes, etc.) Thus, the scope of the unit for a term would become not only the definition of a specified sense, but the dimensional signature of specified sense plus that of all matching senses in the synset. In this way, a wider basis for comparison may afford a wider unit selected from the immediate context of the Common Ground.

Thus, I have created an interface for capturing prototypical dimensional classification in the shallow context of term definitions. This manual judgment
begins with a screen flash to inhibit resonance, and then permits the (nearly context free) ascendant judgment of the initial term. Manual retrieval of the WordNet list of sense definitions is followed by the manual parsing and entry of each term in each definition for each sense into a template by linguistic roles. Each term is placed in its correct location in the template (according to its part of speech and propositional role) and then the template evaluates whether that term has previously been classified. If so, the previous ‘ascendant’ dimensional classification is reused. If not, then the human judge must select and assign the highest relevant of 12 contextual dimensions of human experience.

For each sense definition, the interface calculates the sum of the classifications assigned to each term in that sense definition, providing a compact mini-vector that identifies the frequency distribution for the top eight contextual dimensions. Thus, the ‘ranked unit judgment’ for each sense definition may potentially attain, through gross aggregation, secondary, tertiary, etc., levels of dimensional classification. Also, ties may result. This is interesting, because it highlights emergent focalized dimensional complexes for each sense definition that arise out of the consideration of the context provided by each sense definition of a term. For example, the term SHIP is a ascendantly judged as a social object though it is a physical vehicle technology used to transport physical objects through physical space, over great distances, and also used as a tool of social and economic power, for strategic military and economic purposes. Thus a SHIP is prototypically a physical thing, serving social and perspectival uses in space and time. This is the linguistic version of its more compact dimensional mini-vector.
Figure 8.5.1 Interface for the Manifest Semantic Analysis prototype. Each term of each definition is classified by part of speech and assigned a colour-coded dimensional class.
Thus, a term may be assigned a context neutral spot-classification of the judge’s opinion regarding the highest relevant contextual dimension for the term, in general, affixed to no context except that which cannot be escaped (e.g., the judge’s personal knowledge). Any such context neutral, shallow context, or real world context prototypical class assignment is one dimensional, of the form $X$ is most often $Y$. It is not the simplest judgment possible (e.g., as would be the case with a binary choice that require correcting interaction only if it is wrong) but may eventually become simplified to a binary choice (e.g., as more judgments have been collected, fewer are required).

The difficulty of selecting from 12 options is amplified by the connections between the classes (since they are additive) but is mitigated by the principle restricting the choice to the highest relevant dimension. That is, where the judge may have difficulty due to a complex multi-dimensional context (e.g., as in the SHIP example above), the highest dimension is selected. It is left to the shallow context and real world context situations to identify the prototypical dimension additively.

Once a wide selection of terms has had each word in each of their sense-definitions similarly judged, the context-specific the spot-evaluation of the individual term, which the system compares with the previously classified senses for the given term. Successive multi-dimensional meta-contextual judgments of terms, both by individuals and by multiple raters, will then permit the creation of a standardized multi-dimensional contextual signature for each individual sense of a term listed in a computational resource like Princeton’s WordNet 2.0. The standardized meta-contextual multi-dimensional signature could then be used as a pattern to evaluate the local context (e.g., sentence) and select an appropriate sense.

**8.5.7 - CDG+ Personal Ground**

The activated sub-set of terms in the CDG/+ Common Ground is roughly equivalent to a socially-shared short-term memory, but in CDG+ the wider pool of deactivated yet related propositions also has a role to play. In CDG+, deactivated propositions from the CDG/+ Common Ground are collected to model the ongoing particularized, linguistic long-term memory of the reader in a structure called the
CDG+ Personal Ground. Evidence from language processing experimentation shows that these contents cannot usually be recalled with precision without cues, but cues instigate successful search for and re-activate propositions or gist (Graesser and others 2003a; Garrod and Sanford 1982). Various AI-driven processes can be hypothesized for the structuring and organization of deactivated long-term propositions in the CDG/+ Common Ground (e.g., latent semantic analysis, spreading activation networks). Whichever method is selected, the key processing constraint imposed by CDG+ is that, whatever methods are used to index, sort and aggregate propositions, search must be able to retrieve actual propositions (so not LSA).

For example, TextWorlds are a situation-based scene indexing tool for representing propositions that have been mutually accepted into the CDG/+ Common Ground. Each proposition can thus be used to retrieve the series of TextWorlds that have represented it. What can one do with a pile of matching TextWorlds? A sequence of TextWorld Scenes that match on a given N threshold of situation features and that contain matching cultural knowledge evoked from Frames can be ‘generalized’ to create a unique situational FRAME in the long-term memory CDG+ Personal Ground. If the FRAME is either activated in the CDG/+ Common Ground or exists in online lexicographical resources, the generalized set of TextWorlds can be used to augment the existing FRAME with both the textual passage which produced the TextWorld, and with the TextWorlds as visually organized and segmented Scene representations.

8.5.8 - CDG+ Task Model for Analytical Reading

The Cognitive-Functional Flow Model of General Analysis requires a detailed cognitive-functional specification of the network of sub-tasks that are required for the execution of analytical reading. Standard CDG identifies the tasks that are required to produce both Discourse Comprehension and a Text World mental model of the contents of that language understanding process. The following view (Figure 8.5.9.1) is taken from the Context Model. This view is presented with the intention of helping the reader to keep in mind the wider Situational Context of Discourse as they proceed through this Section to consider the actions undertaken by Discourse Participants towards processing discourse and achieving language
comprehension. These discursive tasks form the basis of the CDG+ Task Model for Analytical Reading.

While in discourse there are necessarily two participants, the CDG+ Task Model for Analytical Reading focuses primarily on the Reader.
Despite the reader-focus, the actions of the Speaker (or Writer) create the **text**. Following **Gricean Maxims**, Speakers select a **topic**, nominate **entities**, and make referential **utterances** that assert **new information**. Later, as they are processed by the Reader or Hearer, these **claims** will also be **negotiated**.

The Reader begins by **accepting** the **Topic selected** and **asserted** by the Speaker (or ignoring or changing the topic, but these Actions alter the course of the discourse, and are not considered further).
Figure 8.5.9.4. The Task Model of Analytical Reading. Readers accept the topic.

The Reader then turns to *identifying propositions* from the utterances made by the Speaker. In this model, there are three procedures Readers use to identify propositions. **Segmentation** identifies propositions within the explicitly stated utterances of the Speaker. **Evocation** activates and selects relevant matching Frames and permits selection of features and whole propositions from the relevant Activated Frames’ default slot values. **Inference** applies reasoning over either Frame elements or over Speaker statements to produce Reader-produced, Speaker-unstated propositions.

Figure 8.5.9.5. The Task Model of Analytical Reading. Readers identify propositions.

The Reader *segments* the **Utterance** (typically by phrase) to identify the Speaker’s explicitly uttered propositions.
Next, the reader follows a complex process to evoke relevant information from Activated FRAMES. First propositions activate FRAMES. Then, the reader must decompose the current proposition's elements. The results of this task (e.g., basic propositional elements and relations) are used in attempts to obtain a resemblance relation between the current situation described in the current proposition (or set of related propositions from a
single sentence) and the situation described in each Activated FRAME.

Each proposition (or individual term) in the utterance may activate FRAMES from the Reader’s long-term memory. When multiple FRAMES are activated, the Reader selects the most relevant FRAME(S), by attempting to obtain a resemblance relation match between the situation described by the FRAME and the current situation described by the proposition.

Figure 8.5.9.8. The Task Model of Analytical Reading. To evoke relevant information, textual cues activate Frames, from which Readers select relevant Frames or slot values (or update slot values).

Propositional elements are used to search the CDG Common Ground and the current set of Activated FRAMES, in order to evaluate propositional coherence, and obtain any resemblance relations between current propositions and activated frames.

Figure 8.5.9.9. The Task Model of Analytical Reading. To evoke propositions, Readers decompose propositional elements, which are used to Search CDG Common Ground to establish propositional coherence, and to search the Activated Frames to obtain resemblance relations.
Resemblance relations are obtained by collecting current situational features and comparing those against frame situational features. When matches are accepted, a FRAME is activated, default slot values from relevant activated FRAMES may be selected.

![Figure 8.5.9.10. The Task Model of Analytical Reading. To evoke relevant information, Readers obtain resemblance relations by collecting situational features and comparing these against frame situational features.](image1)

The Reader may now select from default slot values in a relevant Activated FRAME, in order to fill in details for the current situation from the matching FRAME for that situation.

![Figure 8.5.9.11. The Task Model of Analytical Reading. To evoke relevant information from activated Frames, Readers accept matching resemblance relations between current and Frame situation features. Accepted matches permit the selection of relevant frames and slot values.](image2)

Finally, moving up one level, back to the final means of identifying propositions; the reader may explicitly infer unstated propositions, which create SubWorlds dedicated to reasoned inferential assumptions of various types (e.g.,
Figure 8.5.9.12. The Task Model of Analytical Reading. To identify propositions, Readers infer unstated propositions, which create SubWorlds.

With all propositions of the current utterance identified, the reader next moves on to a range of tasks required to manage or work with the propositions they have just identified.

Figure 8.5.9.13. The Task Model of Analytical Reading. To manage propositions, Readers integrate, accommodate, evaluate, negotiate, identify, and add propositions.

The varied tasks that are executed upon the propositions identified in the previous step, include integrating propositions regardless of the procedure used to
produce them; accommodating **embedded (backgrounded) information** in matrix propositions (For Definition, Refer to Note 37); evaluating both **propositional coherence** and **propositional truth** (probabilistically); **negotiating propositional meaning** with the the Speaker; **identifying** propositions as belonging to and contributing to the **Figure** and the **Ground**; and **adding propositions** to their appropriate world level.

![Diagram of Analytical Reading Tasks](image)

**Figure 8.5.9.14. The Task Model of Analytical Reading.** To manage propositions, Readers integrate propositions regardless of their source. Readers accommodate embedded information in matrix propositions. Readers evaluate propositional coherence and truth. Readers negotiate propositional meaning. Readers identify propositions belonging to Figure and Ground. And Readers add propositions to appropriate World levels.

First readers **integrate propositions** regardless of the procedure used to produce them.
Next, Readers accommodate **embedded (backgrounded) information** in matrix propositions, by *marking* embedded information as *given information* and *incrementing* it directly *into* the **CDG/+ Common Ground**, effectively skipping the *evaluation* phase entirely and skipping ahead to the end of the *negotiation* phase.

Next, Readers evaluate both **propositional coherence** and **propositional truth** (probabilistically).
To evaluate propositional coherence, current propositions are compared against those in the CDG Common Ground.

This comparison evaluates whether the current proposition has a functional relation with any other proposition currently active in the CDG Common Ground. The functional relations sought include synonymy, partial synonymy, antonymy, hyponymy (e.g., the part-whole relation), metonymy (e.g., the is-a relation), and metaphor. Other functional relations may eventually be added, including those identified by computational Rhetorical Structure Theory (RST) Analysis (Mann and Sandra Thompson 1988; Taboada 2006).
Figure 8.5.9.18. The Task Model of Analytical Reading. To evaluate proposition coherence, Readers compare current propositions against those in the CDG Common Ground by evaluating functional relations between them.

**New information** includes any proposition that has no functional relation with existing propositions in the CDG Common Ground. Such information is therefore marked as new using the *emphasis placing mechanism*.

Figure 8.5.9.19. The Task Model of Analytical Reading. To evaluate propositional coherence, Readers mark new information using the *emphasis placement mechanism*.

**Cohering information** has been determined to have a functional relation with existing propositions in the CDG Common Ground. It is therefore used to *update anaphoric reference chains* (maintained in the CDG Common Ground), to *update deictic references*, and to *update previous predications* with any changed conditions to previously stated properties or actions. Updating actually takes place...
in the Figure and Ground-dedicated zones of the TextWorld or SubWorld, so updates reach from the evaluation phase back into the existing state of the model. Updating also occurs only after the acceptance of propositional meaning has been negotiated.

![Figure 8.5.9.20. The Task Model of Analytical Reading. To evaluate propositional coherence, Readers update coherent information reference chains, deictic counterparts, and TextWorld predications.](image)

Finally, reference is tracked by first searching the entity register of the CDG+ Common Ground. Next, world-building elements are searched and any major deictic shifts (in Time

![Figure 8.5.9.21. The Task Model of Analytical Reading. To evaluate propositional coherence, Readers track reference, by searching the entity register of the CDG Common Ground and the World Building Elements of each Text World to identify deictic shifts that require the creation of new Text Worlds, and actions or description chains that require updating.](image)
Zone, Epistemic or Existential Modality, or Proximity to the Here & Now) are identified as creating new TextWorlds. Finally, function-advancing propositions are searched in order to update previous predications with their ‘new’ yet cohering references.

Next, Readers assess probabilistic propositional truth. Reader evaluation of propositional truth includes both assessing the probable truth of a proposition (and assigning it a probability value), and subsequently also assessing the speaker’s authority.

Then, propositional meaning is actively negotiated between the Reader and the Speaker. As required, the Reader requests clarifications and more information from the Speaker, who responds by providing clarification and more information.
Figure 8.5.9.23. The Task Model of Analytical Reading. To negotiate propositional meaning, Hearers may request clarification or more information from Speakers.

Then the both Reader and Speaker search for implications of the newly provided information, after its propositions have been identified.

Figure 8.5.9.24. The Task Model of Analytical Reading. To negotiate propositional meaning, Hearers search for implications of further information provided by the Speaker.

When all queries are satisfied, propositional meaning is accepted, and accepted propositions are incremented into the **CDG Common Ground**. The CDG Common Ground contains Artificial Intelligence components that actively resolve within-world anaphoric identity references, and also handle resolution of the rarer cross-world anaphoric correspondences.
Figure 8.5.9.25. The Task Model of Analytical Reading. To negotiate propositional meaning, Readers accept propositional meaning and increment accepted propositions into the CDG Common Ground.

Next Figure and Ground are identified.

Figure 8.5.9.26. The Task Model of Analytical Reading. Readers identify propositions belonging to Figure and Ground.
To review, to this point, propositions have now been identified, segmented, evoked, inferred, integrated, accommodated, evaluated, negotiated, and accepted.

Figure 8.5.9.27. The Task Model of Analytical Reading. To identify propositions belonging to Ground, Readers classify propositions as belonging to the class of world-building elements, which includes times, locations, enactors, entities, and modal expressions.

At this point, propositions are now ready to be classified, in order to identify world building elements, specifically those entities belonging in the classes of Time, Location, Enactors, Objects, and Modals.

Next, the Figure is identified. Each proposition is classified to by Speech Act Function (Austin 1962) to identify function-advancing propositions belonging in the classes of action, process, modification, description. Each proposition finally classified to identify functional relations that inhere between propositions, particularly those that can be selected by the Speaker to produce a calculated rhetorical effect on the Reader (e.g., relations expressing Ground, Capacity, Argument, Causality, Counter-factuality, Conditionality, Judgment, Synthesis, Multi-nuclearity. Refer to Section 6.10.3, Interactive Evaluation for a discussion of the functional relations added in CDG+).
Figure 8.5.9.28. The Task Model of Analytical Reading. To identify propositions belonging to Figure, Readers classify propositions as belonging to the class of function-advancing propositions, which includes actions, processes, modifications, descriptions, and RST-style functional relations between propositions (Mann and Sandra Thompson 1988; Taboada 2006).

Next, if a World already exists and no deictic shift has occurred, propositions that have been classified as belonging to the Figure or Ground are added to appropriate locations in worlds, at the appropriate level, selecting the appropriate world level (e.g., the DiscourseWorld, the TextWorld, or the SubWorld) based on the Reader’s assessment of each proposition’s probabilistic truth-value. Recall that Reader evaluation of propositional truth included both the assessment of probable truth for each proposition (and assigning it a probability value), and subsequently also assessing the speaker’s authority, which includes evaluating the Speaker’s ability to take responsibility for the assessment of the truth value of their own claims.
In fact, the very first representation creation step first occurs when the discourse begins. However, the initial representation remains unspecified until that initial mental representation can be specified as a World. Specification of a representation as a World first requires several propositional processing steps in order to determine the correct World level of the representation. Further specification of the initial representation (or creation of a new representation) happens after the Reader’s **probabilistic assessment** of each **proposition’s probabilistic truth-value**. Also important to the creation and specification of representations as Worlds are the results of **reference tracking**, wherein the evaluation of propositional coherence of each new proposition requires the performance of **search** over existing World-building Elements to identify any major **deictic shifts** (e.g., in Time Zone, Epistemic or Existential Modality, or Proximity to the Here & Now), which are then taken as cues for creating new TextWorlds.
Figure 8.5.9.30. The Task Model of Analytical Reading. Readers create representations of conceptual structure, or mental models, in their minds.

Now the Reader has the evaluated, accepted propositions that have been added to the CDG Common Ground, and classified as belong to and contributing to the Figure and the Ground, which are ready to be added into a World. If a World has not yet been created, one is created now for the newly evaluated, accepted & classified propositions.

So, finally, readers create Worlds, selecting the appropriate world level based on the Reader’s probabilistic assessment of each proposition’s probabilistic truth-value. Also important here are any major deictic shifts (e.g., in Time Zone, Epistemic or Existential Modality, or Proximity to the Here & Now), which are cues for creating new TextWorlds.

Figure 8.5.9.31. The Task Model of Analytical Reading. To specify representations as Worlds, Readers create TextWorlds for actual events and SubWorlds for epistemically divergent events.
Actual, real, true, here & now events are assigned to TextWorlds.

Figure 8.5.9.32. The Task Model of Analytical Reading. To create TextWorlds, Readers add propositions to the appropriate zone, or update world-building elements and function-advancing propositions, or shift TextWorlds as deictic disjunctures are detected.

Propositions whose probabilistic truth value have been assessed and assigned by the Reader as improbable, unlikely, conditional, hypothetical, and those which are in any other way modally remote events are assigned to SubWorlds, which are indexed according to the variety of phenomena that produced the SubWorld (e.g., a hypothetical SubWorld, or a fictional SubWorld, or a metaphorical SubWorld etc.).

Figure 8.5.9.33. The Task Model of Analytical Reading. To create SubWorlds, Readers add propositions to the appropriate world zone, or update world-building elements and function-advancing propositions.

And thus, the task of analytical reading has proceeded through the processing of one propositional chunk of an utterance made by either a Speaker or Writer. Thus, in analytical reading, this process iterates multiple times per sentence. Specifying this Task Model has established the background mechanics of the sub-task network
that is involved in basic reading. Analytical reading is not substantially different at the mechanical level from basic reading; except that it is goal-focused and may deal with more variation in the quality of information evaluated, which may make probabilistic evaluation of probabilistic truth-values and speaker authority even more important.

Figure 8.5.9.34. The Task Model of Analytical Reading. An overview of the entire CDG+ Task Model for Analytical Reading.

8.6 - CDG+ Synopsis

The original CDG approach focused on extending Lewis’ minimalist Possible Worlds and Modal Logics into Rich Worlds, so as to be full of detailed, fully specified objects in relations explicitly patterned after those in the Real World. Rich Worlds can be seen as philosophical correlate of the Mental Simulations postulated in embodied cognition approaches to comprehension (Zwaan 2015; 1999; Graesser and others 2003a; Zwaan
The Worlds of the CDG Rich Worlds Model are also basically equivalent to the Mental Spaces of (Fauconnier and Sweetser 1996; Fauconnier 1985), and though Rich Worlds are one variety of Mental Spaces, they are not defined strictly in Fauconnier’s terms. Thus, the particular dynamics and mechanisms of the Rich Worlds in the CDG+ Discourse Model are related to Mental Spaces, but are unique in their focus on modelling situational context in the online (e.g., real time) processing of discourse. Consequently, mechanisms developed in Mental Spaces, such as Conceptual Blending (Dancygier 2012; Dancygier and others 2012; Dancygier and Sweetser 2012; Dancygier and Vandelanotte 2009; Dancygier 2006) and Frame Shifting (Semantic Leaps: Frame-Shifting and Conceptual Blending in Meaning Construction 2006c), also apply to the interpretive processes of readers, and therefore can also be integrated directly into CDG+.

Where the CDG Rich World Model is particularly unique, is in how it permits departures into structurally identical worlds that have an alternate epistemic or existential grounding (Vandelanotte 2004; Brisard and Conference 2002; Glenberg and Robertson 2000; Kaschak and Glenberg 2000; Glenberg and Langston 1992; Holly A Taylor and Barbara Tversky 1992; Glenberg and others 1987b), called Modal Worlds, or sub-worlds.

Beyond the Rich Worlds Model, CDG also offers a thorough and unflinching integration of insights from across cognitive, discourse, and standard approaches to language within a single framework. Even after twenty years, this novel framework of recognizable components remains a unique system-of-systems that focuses on specifying the formation and externalization of mental representations during online reading comprehension. The original CDG approach made no specific claims representing itself as a discourse processing model. However, given its focus, method, and the emergent dynamics of the mechanisms it integrates, I argue that extended version of CDG, CDG+, should be considered a plausible (Forsythe and Xavier 2006; Forsythe and others 2006) Discourse Processing Model.
Thus, in considering how the original CDG approach could be improved upon, it is readily apparent, given the integration of so many concepts and approaches in language, that complexity in the emergent framework is a challenge. At its core CDG provides the analyst with means to systematically approach the various levels of context in discourse and derive the contribution of each level to understanding the scene. I have therefore identified and visually modelled each contextual situation that CDG describes as contributing to both the Discourse and Discourse Processing Models. This contextual overview should help position the individual mechanisms of both the Discourse Processing Model and the Discourse Model which it produces. This should prove especially useful for the reader who is unfamiliar with this method or domain.

Thus, CDG+ offers a Context Model, which first specifies situational dynamics in the CDG+ Discourse Processing Model that condition the production of the CDG+ Discourse Model; and second, specifies the relations between the situations that are represented as levels in the Discourse Model. (See Section 8.5.1 CDG+ Context Model)

CDG+ thus offers a Discourse Processing Model, which is positioned in this Dissertation as a plausible (Forsythe and Xavier 2006; Forsythe and others 2006) model of the cognitive processes used to build an internal, representational Discourse Model during discourse processing for language comprehension. Critically, the CDG+ Discourse Processing Model relies on the activation of Knowledge Structures from memory (as van Dijk and Kinsch 1977 specified). CDG/+ specifies these Knowledge Structures as Fillmore’s Cultural Knowledge Frames, and thus integrates Fillmore’s Frame Semantics into a Rich World Model, that is in turn integrated with Johnson-Laird’s Mental Models and van Dijk’s Situation Models (Fillmore 2006; Fillmore and Baker 2001; Baker and others 1998; Fillmore and Atkins 1998; John B Lowe and others 1997; David Kellogg Lewis 1986; 1973; David K Lewis 1968; Johnson-Laird and Girotto 1998; Van Dijk 2006; Staczk and others 1985; Van Dijk and Kintsch 1983; Kintsch and Van Dijk 1978; Van Dijk 1977; 1976). Moreover, the selection of relevant Frames from activated Frames requires specialized analogical faceted search to match features of the recalled and activated Frame situation with the features of the current situation. Selected Frames may then be used as resources to provide detail for current
situations, specifically from the default scene-specific information contained in the
Slots of Cultural Knowledge Frames.

Equally importantly, the CDG+ Discourse Processing Model also relies on the
complex dynamic processing of an AI-managed CDG Common Ground, which
transacts a host of indexing and matching tasks in its role as a specialized type of
socially shared, textually cued, short-term memory. CDG/+ specifies these tasks as
anaphor resolution and reference-chain matching, deictic counterpart resolution,
identification of new information, placement of emphasis (on repeated old
information) and accent (on new information). CDG/+ thus integrates Grice’s
Cooperative Principles of Informativeness, Truthfulness, Relevance, and Clarity,
Lyon’s Deixis and Anaphor, Givón’s Coherence, Sperber & Wilson’s Relevance,
Føllesdal’s Opacity, Lewis’ Accessibility, each with significant extensions or
extensive critiques, into Clark’s Common Ground, which is itself significantly
extended as the CDG Common Ground (Grice 1975; Lyons 1995a; 1995b; 1981;
1977; 1975; 1973; Givón 2005; Gernsbacher and others 1995; Givón 1995; Givón
1993; Sperber and D. Wilson 1995; Føllesdal 1998; 1961; David Kellogg Lewis

Moreover, the original CDG approach integrates into CDG Common Ground its
own Selection rules for selecting warranted, inferred information from relevant
activated Cultural Knowledge Frames. CDG information evaluation rules determine
Coherence and result in the placement of Emphasis on old information and Accent
on new information. Incrementation rules govern the insertion of new information,
from the text and from inference, into the CDG Common Ground. Exceptions for
non-standard introduction of new information are handled using the principle of
Accommodation (Werth 1993; 1984; Werth International Conference on Social

CDG presents a novel Coherence Constraint which governs Incrementation of
new information both from the text and from inference into the CDG Common
Ground (Werth 1999c; 1999i). In CDG, evaluating Coherence requires active
comparison of each current proposition with those already in the CDG Common
Ground. First, the CDG Common Ground evaluates for functional relations
(synonym, antonym, meronymy, metonym) between its N most recently activated
propositions and the current proposition. The implication of a matched functional relation is that the current proposition is old, so it is marked to place *emphasis*. Old information is linked to its antecedent references in the CDG *Common Ground* by *Reference Chaining* (Werth 1999k). Conversely, non-matching propositions are evaluated as new, are marked with *accent* and are *incremented* into the CDG Common Ground.

The purpose of the original CDG framework is to create a visual proxy for the mental model created internally while reading. In CDG/+ this is the Discourse Model, which specifies three levels in the internal mental representation that is produced while making sense of language (e.g., equivalent to van Dijk & Kintsch’s three level situation model). The difference between the Discourse Processing Model and the Discourse Model is simple; the dynamics described as the Discourse Processing Model produce the Discourse Model.

Since I could not introduce the entire CDG approach in the available space, I have summarized and specified relations between the CDG+ Discourse Processing Model and its product, the CDG+ Discourse Model, in the visual CDG+ Context Model, which specifies the four Situations that impact both CDG+ process and product Models.

CDG+ offers the *Deictic Matrix* as an aid to analysts when starting to place their first TextWorld, to specify its starting temporal, epistemic, and proximal and accessibility status. Analysts simply select a point within the Matrix that encapsulates their judgment of the TextWorld’s status in four deictic systems. The Deictic Matrix addresses the problem of the unlabeled and presumed temporal origin in standard Text World Diagrams. The Deictic Matrix is an incremental improvement to the Standard CDG Tense and Remote System relationship specification that has reduced label redundancies and made boundaries within the matrix dynamic.

CDG+ offers a *Meta-Contextual Frame* and the CDG+ Manifest Semantic Analysis method as an improvement upon the Deictic Matrix, that swing the somewhat cluttered multi-dimensional Matrix around, popping it out of the two-dimensional flat matrix representational space. The result is a bounded three dimensional representational space, with a novel containing frame. Since in CDG the focus is on representing the aggregate of reader judgments, there is no explicit
focus on supporting the capture of specific reader judgments while reading. CDG+ Manifest Semantic Analysis explores the possibility of using dimensional content evaluations captured in the CDG+ Meta-Contextual Frame to automatically validate automated sense selections. Moreover, it provides an aggregation method to weight spot-classifications of single terms by the combined frequencies of the classified terms in their various definitions, to work out different dimensional signatures for each sense of a potentially ambiguous terms. Thus, I designed the CDG+ Meta-Contextual Frame and Manifest Semantic Analysis method for classifying both terms and TextWorlds along multiple deictic and contextual dimensions.

Since the original CDG approach does not specify what happens to accepted propositions in the Common Ground once they lose their activation focus, I have specified the longer-term collection of deactivated terms into the CDG+ Personal Ground. Finally, since the Flow Model of Analysis can only be complete with the specification of requisite analytical reading sub-task network, I have specified the CDG+ Task Model for Analytical Reading, which is extrapolated from the entire description of the CDG/+ discussion using a manual analysis of predicate argument structure to identify both tasks and the objects upon which the action is undertaken.
Chapter 9 - TextWorlds+

This Chapter, TextWorlds+, describes improvements to a discourse analytical methodology that is focused on the production of visualizations of the internal representational models produced while reading. This analytical methodology is derived directly from the revised and extended version (Chapter 8, CDG+) of the original integrative Cognitive Discourse Grammar, and particularly on the CDG+ Discourse Processing Model. The manual method of extended TextWorlds+ Analysis therefore benefits from improvements made to the original CDG approach, not least in achieving a stepwise procedural clarity not offered in the original approach. And, most importantly, in both its original and revised versions, this approach to Discourse Analysis depends upon active meta-cognitive reflection upon the act of online discourse processing and coming to a specific, personal comprehension of text during analytical reading.

In this Chapter (See Section 9.3, The Problem) the extended manual TextWorlds+ Analysis methodology is presented and acknowledged as a fairly costly (e.g., requiring a high degree of expertise in linguistic classification & phenomena), structured method of meta-cognitive reflection upon the particular, personal comprehension process that allows its user to visualize the conceptual, semantic, and relational structures they construct as mental models during reading. In general, it is argued here that providing structure to open-ended, free-form concept mapping marks an improvement over unconstrained methods, and one which holds promise for lowering the cost and increasing the effectiveness of sharing externalized mental models of conceptual structures of text between individuals.

The precise rationale for reduced cost and increased effectiveness of mental model construction and sharing is this: it should soon be possible to produce a ‘baseline’ model of the manifest (not latent) conceptual structure that any reader (e.g., with a specific Common Ground) would construct prior to any deeper engagement with their own particular knowledge stores. Experimental research on Latent Semantic Analysis models and human word-learning rates has shown that at a certain point, learning rates plateau from 10 wds/day to far fewer (1 to 4) beyond about 60,000 words learned. Thus, we may safely assume that an established
‘common’ background pool of cultural knowledge exists, and develop a precise specification of the lexicon and underlying corpus used for the production of each baseline model. So one may specify the precise collection of readings (e.g., the corpus) that contributed to the Common Ground upon which a particular baseline reading was produced.

Moreover, insofar as most humans who have read the core texts of a domain should produce a similar reading, it should also then be possible to establish how much reading, and of precisely which texts, is required to match or best a domain expert’s reading. This eventuality presents interesting possibilities for both curriculum design and measurement in learning, which for reasons of space will not developed more fully here.

At the point of having a baseline model, the TextWorlds+ User Interface would facilitate deep personalization of the baseline model, effectively becoming a personalized Discourse Co-Processing Agent (built upon the Discourse Processing Model) which begins as common to general readers becomes unique to a specific user. This becomes possible chiefly by drawing upon the user’s personal cultural knowledge structures and expertise schemas (e.g., Frames) to refine and specify the unique conceptual structure produced by their own reading.

At that point, it would become possible to measure the change in models, baseline and post-personalization, to measure the contribution of personal knowledge in Discourse Analysis. It would also become possible to measure the difference between my reading and yours; or between mine today and mine as a high school student; or to compare an individual’s progress to their cohort, or to the expected average. Measured change in externalized mental models would provide a novel empirical measure of learning.

This Chapter positions the improved manual TextWorlds+ Analysis method as a step towards lowering the cost of manual analysis by computational implementation, which will, in future work, integrate various Natural Language Processing techniques to offload the identification of linguistic classes and phenomena (e.g., anaphora and coreference resolution, concept identification, semantic role labelling, part of speech tagging) and leverage the results of these automated classification tasks to facilitate the automated construction of a baseline cognitive model that readers may then interact with and personalize.
Since the **TextWorlds+ Analysis** method is explicitly derived from the wider framework of Extended Cognitive Discourse Grammar, this Chapter begins with a review, briefly characterizing CDG and then summarizing relevant contributions made to the original approach that were first introduced in the previous Chapter as CDG+. The purpose of the CDG framework is to externalize the representational models produced internally during language comprehension. In its extended version, CDG+ offers a Context Model, a Discourse Processing Model, a Discourse Model, a *Meta-Contextual frame* within which to practice Manifest Semantic Analysis, and finally a Task Model for analytical reading so as to improve the **Cognitive-Functional Flow Model of General Analysis** and give tool-makers a better appreciation of the sub-tasks involved in analytical reading.

The **CDG+ Context Model** first specifies situational dynamics in the **CDG+ Discourse Processing Model** which condition the production of the **CDG+ Discourse Model**; and second, specifies the relations between the situations that are represented as levels in the Discourse Model. (See Section 8.5.1, **CDG+ Context Model**.) The CDG+ Discourse Processing Model is positioned a plausible (Forsythe and Xavier 2006; Forsythe and others 2006) model of the cognitive processes that are used to build an internal, representational CDG+ Discourse Model during reading. The CDG+ Discourse Model specifies three levels within the internal mental representation that is produced while making sense of language. The difference between the Discourse Processing Model and the Discourse Model is simply that the former produces the latter. The **CDG+ Deictic Matrix** provides a contextual label of the temporal, epistemic, and proximal and accessibility origin of each Text World Diagram.

The **CDG+ Meta-Contextual Frame** provides bounded three-dimensional Euclidean Space that contains a multi-dimensional Classification System for capturing evaluations of the meta-contextual dimensions of human experience and the major deictic categories of Temporal, Modal, and Proximal while reading.

The **CDG+ Manifest Semantic Analysis** explores the possibility of using dimensional content evaluations captured in the **CDG+ Meta-Contextual Frame** to automatically validate automated sense selections. Moreover, it provides an aggregation method to weight on-the-spot prototype classifications of single terms by the combined frequencies of the classified terms in their various definitions, to
work out different dimensional signatures for each sense of a potentially ambiguous terms.

The CDG+ Personal Ground captures all propositions from the CDG Common Ground upon their deactivation. The purpose of this capture is to begin to compliment the cultural background knowledge available to readers as FRAMES, with specialized personal background knowledge that remains available to readers after reading, stored as Personal Expertise Schemas and aggregated by repetition into Frames in a cold-storage personal version of the CDG Common Ground.

The CDG+ Task Model for Analytical Reading addresses a need in Flow Model of Analysis to specify the requisite sub-task network for each cognitive-functional task in the Flow Model. I have specified the CDG+ Task Model for Analytical Reading, which is extrapolated from the entire description of the CDG/+ discussion using a manual analysis of predicate argument structure to identify both tasks and the objects upon which the action is undertaken.

![Figure 9.1.A](image)

*Figure 9.1.A. These most common words, or stop-words, are usually removed from vector space representations of document word frequency. However, many of these words serve an important role of indicating the deictic attachment of a term (determiners) or the semantic connections between terms. B. The top 300 words used in this Chapter.*

### 9.1 - Overview

The common-place *generative folk theory* assumes that humans understand language based upon retrieval of meanings from specific words and sentences. As
an alternative to the ‘meaning is out there’ perspective, this Chapter shows in depth that the words and sentence structures, while important, are the just tip of the iceberg. And what’s more, that whole iceberg is in your head. Though the research reported in this Chapter will not reveal the whole iceberg, nevertheless, this Chapter will bring to light some generalizations that show what text processing requires - whether using manual methods, computational methodologies, or both.

The core idea presented here is that the extended method of TextWorlds+ Analysis described below (in Section 9.5 of this Chapter) is isomorphic with the model dynamics of online discourse processing specified in the Section 8.5.2, CDG+ Discourse Processing Model. So this manual method of Discourse Analysis, TextWorlds+ draws deeply upon the dynamics of the CDG+ Discourse Processing Model to produce a TextWorld Model. TextWorlds are worthwhile because they do not only do what an intelligent reader does (see Chapter 8, CDG+), they could potentially offer a way to give support to those who read to extract information quickly, in the form of a visual digest of scene-specific details. Other theories (including some from Cognitive Linguistics) may yield more accurate or precise readings, but do not facilitate the pragmatic process of professional analysts who require support in the challenging tasks of analytical reading.

This Chapter begins by offering an abridged and comprehensive summary of the Solutions offered in the preceding Chapter, in order to review the results of each specific contribution made toward revising the original CDG and Text Worlds approach and extending them into CDG+ and the extended manual and computational methods of TextWorld+ Analysis. In order to distinguish my contributions to the revised method from the original approach, I employ a typographical distinction between the original Text Worlds approach (and the simple Stylistic Text World Theory Analysis) which produced static Text World diagrams; and my extended Manual Method of TextWorlds+ Analysis (and extended Computational TextWorlds+ Method) that currently produces mixed modality, digitally interactive TextWorld Models. The reader may assume that extended TextWorlds analysis follows standard Text World approach wherever no revision is specified. Throughout this Chapter, note that the process is referred to as TextWorlds+ Analysis and the product is a TextWorld Model.

Section 9.1, provides a brief Overview of this Chapter 9, TextWorlds+. 
Section 9.2, The Concepts, reviews the immediate purpose of TextWorlds+ Analysis and the long term goals towards which this Chapter contributes. Eventually, computational implementation of the methods described and developed here will produce a baseline interactive digital TextWorld Model. Until that point, the manual practice of extended TextWorlds+ Analysis can facilitate the externalization of internal representations of the sense-making process that readers engage in while reading, albeit the hard way.

Section 9.3, The Problem, introduces the original manual Text Worlds approach and specifies several of its shortcomings, including its lack of a simple procedural guide to Text World Analysis.

Section 9.4, The Solution, presents solutions to those problems, as an improved manual method for the interim and a computationally implemented method for the long-term.

Section 9.5, Manual Method introduces the extended manual method of TextWorlds+ Analysis. The extended manual TextWorlds+ Analysis method involves four complimentary steps of scene analysis, propositional analysis, functional analysis, and shift analysis. These analytical phases are not discussed, as such, in standard TextWorlds approaches to Text World analysis, and thus mark an original contribution to the theory.

First, TextWorld scene analysis identifies the referential anchors (or first mentions) of world building scene entities. Next, TextWorld propositional analysis identifies the actions and descriptions as connections among the objects and enactors of each scene, whereupon TextWorld functional analysis identifies the unique speech-act function of, and between, contiguous propositions, while TextWorld shift analysis identifies transitional dynamics among references to entities, scenes, and TextWorlds. This introduction will set the stage for the presentation of a brief exemplar of TextWorlds analysis, in the next section.

Section 9.6 provides a brief Analytical Exemplar of the manual method of extended TextWorld+ Analysis.

Finally, Section 9.7, Future Work, sketches out a design for a computational version of this method. Section 9.6.1, Toward a Computational Method, introduces
computational TextWorlds methodology that I have designed to automate extended manual Text World analysis.

Finally, Section 9.8, 9.8 - TextWorlds+ Synopsis, reviews the arguments and contributions made in this Chapter toward the improvement of the original manual Text World approach, and the computational implementation of the extended manual TextWorlds+ Analysis Method as a Discourse Co-Processing Agent and novel AI-supported Interface to text.

9.2 - The Concepts

The purpose of TextWorlds+ Analysis is to identify the conceptual structures built by readers in the course of language comprehension. The TextWorlds created by this analytical method are intended to provide an interactive, visual representation that serve as a record of the major semantic classes automatically and easily identified by readers, the knowledge activated by references, the inferences supported by the text, and, finally, all the connections between entities, objects, and propositions. Thus, TextWorlds are the product and TextWorlds+ Analysis is the process of a specific method of analysis that reveals precisely what kind of text processing is required to produce discourse comprehension.

Whereas the future of TextWorlds is digital, automated and interactive, the history of Text World analysis is analog, manual and static. The efforts reported in this Chapter have contributed to the present hybrid state of the art in TextWorlds, which is digital, manual and interactive.

In the historical Text Worlds approach, Text Worlds are complex conceptual objects that function at various levels. At their most basic level, Text Worlds are a kind of Mental Model of texts (Niehaus and Young 2014; Van Dijk 2006; Fischoff and others 2011; Zwaan 1999; Johnson-Laird and Girotto 1998; Carley and Palmquist 1992; Glenberg and others 1987a) or Mental Spaces (Dancygier and Vandelanotte 2009; Geeraerts 2006; Brugman 1996; Sweetser 1996; Fauconnier 1985). The original Text Worlds approach produced a faceted (Jian Zhao and others 2013; Henry and others 2013; Gotz and others 2012; Ducrou and Eklund 2009; Collins and others 2009; Tunkelang 2009; Ando and others 2005; K Yang and Jacob 2003) view of the unique conceptual structure of texts by systematically
identifying, segmenting (Kurby and Zacks 2008; Brenner 2010), and connecting (Taboada 2006; Taboada and Mann 2006; Uzêda and others 2010; Theijssen 2007; Mann and Sandra Thompson 1988) elements of **figure** and **ground** (Jun Zhang and Jianhua Liu 1995; Tsur 2009; Ghosh and Roy 2011; Hsiao and Chang 2015) in text (Kitamura and Gunji 2011).

In their updated, present hybrid state, TextWorlds present as a scene-specific network-based (Jusufi and others 2013; Coffey and others 2003; Mylopoulos 1981; Minsky 1973) knowledge representation object (Marinov 2008; Brewster 2005; Mylopoulos 1981; Kuipers 1975). Manually created static Text Worlds are visual indices of situations in text that are identified through unassisted manual analysis, usually by literary scholars in stylistics (Gavins 2000; Whiteley 2011; Emmott and others 2007) or cognitive poetics (Vandaele and Brône 2009; Narayan 2009).

In both the present and future cases, digital TextWorlds are the interactive visual interface to a user-driven discourse analysis process that is based on a discourse processing model of human language conceptualization. In the context of the CDG+ approach, TextWorlds are thus also a core mechanism in the extended and revised version of the CDG+ Discourse Processing Model.

The prospect of computationally assisted TextWorld production will open the practice of TextWorld+ Analysis to a wider audience of practitioners, since the application of the original Text World method has historically required a high level of linguistic expertise. This line of reasoning is further elaborated in the following Section 9.3, The Problem. In all cases, both the extended TextWorlds+ Analysis method and original Text Worlds approach to identifying Text World model elements and dynamics should be considered as a first-person methodology for observing and recording the cognitive process used by readers to internally represent and externally model both the state of affairs surrounding any given language event and the conceptual structure and contents of that discourse, and all as fitting into layered, interactive worlds.

### 9.3 - The Problem

mechanisms of the approach. Instead, Werth describes the objects and mechanisms of his approach, and it is left to the reader to ascertain how best to procure and evaluate these entities. Whereas fellow scholars in cognitive discourse pragmatics may have a sense of how to proceed, the novice would not.

Gavins (2005) describes a manual approach to Text World Theory roughly as the identification of ‘world-building’ elements, or time zone, location, enactors, and objects, and then the identification of ‘function-advancing’ propositions, which include the texts’ unfolding action and description. In many respects, the analysis resembles the products of Mental Spaces Theory (Fauconnier and Sweetser 1996; Fauconnier 1985).

Analysts in cognitive stylistics and poetics are consistently able manually produce single static Text World diagrams with just these elements. Another model, blending has helped produce analyses of literary texts which represent the emergence of textual interpretation (Dancygier and Sweetser 2012).

In both cases, the approach is considered costly because its execution requires a range of expert linguistic skills, particularly in identifying linguistic classes of terms.

Moreover, to computationally model the content and dynamic evolution of mental models for text comprehension, analysts must identify much more than just Figure and Ground elements. For this reason, I have revised and extended the original Text Worlds approach as the extended manual TextWorlds+ Analysis method, or TextWorlds+ Analysis, and designed the computational TextWorlds+ methodology to augment and automate the extended, manual TextWorlds+ approach. This is discussed in the following Section, 9.4, The Solution.

9.4 - The Solution

The ultimate goal of this research is the design and development of a series of tools that work as a digital environment within which it is possible to predictively model, actively learn, and interactively visualize the formation and refinement of reader conceptualization models while reading. This Chapter describes my progress towards that end.
Automatically produced **digital interactive TextWorlds** are envisioned as interactive, semantic, conceptual networks built in near-real-time, whose visual state is synchronized to reading progress. The underlying conceptual structures that are visualized as TextWorlds are excavated and extrapolated from the syntactic and semantic structures of the text. Baseline TextWorlds will be partially constructed for the user, drawing upon natural language pre-processing techniques to identify the specific linguistic features that have been identified by the original Text Worlds approach as essential for the **interactive representation** of conceptual space in discourse (See Appendix B. Textual Cues required to facilitate the automation of TextWorlds Method). Reader interaction with this visual, analytical TextWorlds model will allow readers to update and refine their own conceptual models, with reference to their own knowledge bases, in order to capture, review, and revise their own process of Discourse Comprehension.

With these goals in mind, my current task has been to understand and articulate each element and dynamic of the original CDG model that serves as the foundation to TextWorlds analysis. This overview will position me, in the final section, Analytical Exemplar to step through a brief example in detail, to provide a manual approximation of the proposed automated methodology.

**9.5 - Manual Method**

At a gross level of generalization both CDG and CDG+ (henceforth CDG/+), simply segment the scene presented in a text in the same manner and using the same mechanisms as reader’s cognitive systems would do in a real world embodied experience (Wilson 2002; Louwerse and Peer 2009). Thus, the largest **chunks** (Jones and others 2014; Guida and others 2012; Bannard and Matthews 2008; Gobet and Clarkson 2004; Kübler and Hinrichs 2001; Gobet and others 2001; Gobet 1998; Davis 1984; Black and Bower 1979; Kleinberg and Kaufman 1971) in the CDG/+ Model are **Figure** and **Ground** (Wertheimer 1922; Koffka 1922; Humphrey 1924; Helson 1926).

In CDG/+, the set of entities that are identified as contributing to the Figure and Ground is called the **contextual entity set**. This **contextual entity set** thus includes entities, which are further classified as those contextual **entities required to build a**
World, and those descriptive elements that execute specific speech-act functions (Searle 2009; Ryckebusch and Marcos 2004; Sadock and others 2004; Jovanovic and op den Akker 2004; Searle 1974; 1969; 1962) of the discourse:

**Contextual, World-Building Elements**

- **Time** (when the situation takes place) (Werth 1999f)
- **Place** (the location of the situation) This represents the basic property of deixis, or spatial location with respect to the speaker. The extended TextWorlds DP Model also includes here conceptual extensions to space, particularly from abstract notions of containment metaphors.
- **Enactors** as sentient participants
- **Other entities** (e.g. Objects, concepts, masses, forces…)
- **Modals as modal verbs and other forms of modal expression**

**Descriptive, Function-Advancing Propositions**

- **Relations (holding between entities).** These relations may refer to predicate-functions as from formal predicate logic. However, most use of propositional relationships in CDG/+ are equivalent to surface verbs. The extended CDG/+ Discourse Processing Model also includes here functional relations determined by the reader to exist between propositions, patterned precisely after those identified by Rhetorical Structure Theory (RST) (Mann and Sandra Thompson 1988; O'Donnell 2000; Taboada and Mann 2006; Taboada 2006; Theijssen 2007; Radev 2010; Uzêda and others 2010).
- **Qualities** (applying to entities) These are typically adjectives and adverbs, which most often correspond to one-place predicates.

Analysts may use the CDG+ Context Model to situate elements of the CDG+ Discourse Model with reference to each situation that contributes to the CDG+ Discourse Processing Model. Thus, a discourse always takes place within an immediate physical situation but has as its topic a verbal situation which cues the construction of at least one text world (See Figure 8.5.1.4 CDG+ Context Model). A text world is thus both a particularly dynamic model of the discourse topic and its own particular kind of situation. The discourse world is a mental representation about the situation in which a participant engages in co-creating the discourse. It therefore has no more epistemic status than a text world; but it is the closest the analyst comes to reality, as represented within this model. Moreover, most discourses depict situations that are distinct from the immediate discourse situation.
Especially texts, whose discourse situation is split in time and space between author and reader. Thus, typical text worlds\textsuperscript{58} based on written texts have to do with \textit{states of affairs} which are \textit{deictically remote} insofar as they are in another place at another time. In any case, whether depicting a remote situation or a shared immediate context, the \textit{text world} is thus the \textit{mental representation} of any \textit{situation} that is \textit{depicted} by the \textit{discourse}. (Werth 1999f)

The manual extended TextWorlds+ Analysis method includes four complimentary steps: 1) \textit{scene analysis}, 2) \textit{propositional analysis}, 3) \textit{functional analysis}, and 4) \textit{shift analysis}. These specific steps, are not discussed, as steps in the execution of Text World analysis in either the original approach or the more recent stylistic approaches to Text World Theory. TextWorld+ \textit{scene analysis} identifies the referential anchors (or first mentions) of world building scene entities. Next, TextWorld+ \textit{propositional analysis} identifies the \textit{actions} and \textit{descriptions} as \textit{connections} among the objects and enactors of each scene. Next TextWorld+ \textit{functional analysis} identifies the unique speech-act function of, and between, contiguous propositions, in the fashion of Rhetorical Structure Theory analysis (Mann and Sandra Thompson 1988; O’Donnell 2000; Taboada and Mann 2006; Taboada 2006; Theijssen 2007; Radev 2010; Uzêda and others 2010). Finally, TextWorld+ \textit{shift analysis} identifies textual and conceptual cues indicating transitional dynamics for references to entities, scenes, and TextWorlds.

\section*{9.5.1 - Scene Analysis}

To begin the \textit{manual TextWorlds+ Analysis process}, the first objective is to establish the setting for the scene. Thus, manual TextWorlds \textit{scene analysis} starts by manually identifying the parts of speech (POS) of the first sentence. Basic standard POS analysis identifies nouns, verbs, and prepositions, as well as Subject and Predicate phrases. These linguistic classes are used to identify \textit{scene elements}, or the World-Building Elements of standard TextWorlds analysis. These include \textit{times}, \textit{places}, \textit{objects}, \textit{doers}.

\subsection*{9.5.1.1 - Temporal Analysis}

The original Text Worlds approach to the identification of World-Building Elements includes a specific process for identifying \textit{verb tense} on a phrase by
phrase (or proposition by proposition) basis. The TextWorlds analysis of verb tense first identifies the basic **Time Zone** for the TextWorld and then also highlights perspectival micro-shifts to internal and external **aspectual** sub worlds. A TextWorld’s basic **Time Zone** is determined using pairwise comparison of the Reichenbach (Reichenbach 1966; 1947) Speech Time, Reference Time and Event Time to decide whether the sentence’s main verb indicates action in the Past, Present, or Future. More specifically, the Text World’s basic Time Zone is determined by comparing between the time an utterance is made and the time referred to by the utterance. So the analyst first determines if the Reference Time is before, equal to, or after Speech Time. The **specific verb tense**, or **Aspect**, of each proposition marks small perspectival shifts, most often within the established basic Time Zone of the current TextWorld. Aspect is determined by comparison between the time the event took place and the general Reference Time (e.g., basic Time Zone), taking event characteristics of completion and duration into account.

![Table 9.5.1.1.1](image)

*Table 9.5.1.1.1 The determination of basic Time Zone and specific verbal Aspect is a stepwise comparison between Speech time, Reference time and Event time.*

With **verb tense analysis** complete, other lexical and semantic temporal indicators are identified (e.g., dates, times, temporal adverbs).

### 9.5.1.2 - Location Analysis

Next, indications of **scene location** are identified in order to establish the geospatial and physical locale of the TextWorld scene. These include explicit
geographic references, as well as environmental objects (e.g., nouns) that may be considered landmarks, and locative terminology such as prepositions with reference to which the place is defined. Whereas many prepositions contextualize parts of objects with reference to their whole, or arrangements of objects within a scene, at this point, only prepositions which define the location of the scene are considered.

9.5.1.3 - Entity Analysis

With location analysis complete, next, any active agents are identified (e.g. Nouns that are Subjects), as well as any other salient entities. Doers take the active role in the action described by the scene, whereas receivers are other objects that are either described or acted upon. As each entity is encountered, so long as it is uncontested in the discourse, and if it does not match any previously accepted and active entities, it is accepted as a new referential anchor into the entity Reference Chain Register of the Common Ground.

9.5.1.4 - Scene Analysis Synopsis

To this point, manual Text World scene analysis has identified what standard TextWorlds calls the World Building entities or times, places, objects, enactors. These are the basic scene elements required to build an initial distinct TextWorld. The revised and extended Text World scene analysis explicitly acknowledges that this stage depends on the identification of entities belonging to the first four most basic dimensions of human experience (e.g., the temporal, spatial, physical and corporeal). Thus, entities classified in these first four dimensions collectively provide the standard baseline for scene identification in TextWorlds.

Consequently, any language or genre that contain few concrete objects or descriptions, or few temporal or geographical references, can make identification of World-Building elements a difficult step in standard Text World analysis. Thus, in the revised and extended method, TextWorld scene analysis accounts for this challenge from abstract, methodological, or theoretical texts, first by loosening the traditional focus on geospatial and physical locale and extending the spatial focus to include intangible domains and other abstract containers. For example, “In the field of Cognitive Linguistics…” and “In this case,…” and “In Experiment 1,…” all
become valid abstractions of place that serve a World Building, scene-establishing function. Finally, revised TextWorld analysis takes the text-model (See Figure 5.2.2.3.) provided by document metadata, particularly bibliographical features, like publication dates, LOC classifications of publication subjects, authorial keyword-selections, or the disciplines and topics of periodicals, as a resource to provide missing World-Building entities. Standard TextWorlds Text World analysis does not include a text-model.

9.5.2 - Propositional Analysis

With the initial scene established, TextWorld analysis shifts to a focus on determining and visualizing propositional relationships. So, once scene analysis has identified referential anchors, propositional analysis connects these entities with their features and actions, starting with those provided explicitly in the text, by chunking the sentence into propositions (or predicate-argument phrases). The propositions produced by analyzing textual entities, actions and descriptions are augmented by activation of cultural knowledge frames cued by any of these textual entities, actions and descriptions. Thus, inferred entities, propositions, or whole situations are selected by each discourse participant from frames activated from their own repository of cultural knowledge frames.

9.5.2.1 - Anaphora Resolution

For each textually given or inferred entity, a match is attempted against existing Reference Chains. Non-matching entities are marked as new, and are added to the Reference Chain Register, while matched entities are added to existing Reference Chains. Thus all propositional arguments, textual and inferred, are indexed to the Reference Chain entity register, first resolving deictic and other coreference (e.g. anaphora) to the correct entities, before attaching features and actions to the correct entities.

9.5.2.2 - Probabilistic Truth Assessment

All propositions are then evaluated according to their degree of accessibility by discourse participants, regarding their probabilistic truth values, and according to participants’ personal and social evaluations of the speaker. The probabilistic truth
of propositions is thus determined to fall somewhere on a range from 100% probably true (e.g., certainly true), to 0% probably true (e.g., certainly false), where closer to 100% entails certainty and acceptance, and closer to 0% true entails contestation. In face to face discourse, this contestation is actively argued. In reading situations the contested assertion is proportionally marked as False, accepted as False, and incremented into the Common Ground. Inferences drawn from this conclusion are evaluated against the Reference Chain entity register and the Common Ground propositional index.

9.5.2.3 - Common Ground Incrementation

So, for each newly accepted proposition, a match is attempted against existing propositional entities in the Common Ground. Non-matching propositions are marked as new, and are added to the Common Ground, while matched propositional entities have their features and actions incremented into existing Propositional Conceptual Structures, expanding existing propositions to include new relations. This TextWorlds coherence analysis evaluates all accepted propositions, textual and inferred, against a range of semantic relations (synonymy, metonymy, antonymy) to determine any relation between each newly accepted proposition and the previously accepted and currently active propositions in the Common Ground. Thus, all propositions in the Common Ground, regardless of their textual or inferred origins, have been evaluated for probability and accepted by participants. Upon their acceptance, each proposition has been tested for coherence against both an entity register and a propositional index.

9.5.3 - Functional Analysis

After indexing and updating entity references, and then identifying, evaluating, and analyzing propositions for coherence, propositions are next analyzed for their speech act (How to Do Things with Words 1962a) or rhetorical function (Austin 1962). So, with basic propositional analysis complete, the TextWorld+ analysis process then moves on to functional analysis of the roles of and relations between propositions based, in part, on their contiguous textual location. For many simple propositions, a basic speech-act function is not dependent on the textual location of the phrase. However, for more complex propositions (e.g., propositions
composed of propositions) presentation order selects neighbors to produce emergent, unstated relational propositional values.

Rhetorical Structure Theory (RST) (Mann and Sandra Thompson 1988; Taboada 2006) has developed a method and computational tools to analyze and label these emergent relations. This method and, indeed relational propositions are not discussed in standard Text World analysis, so their treatment in extended TextWorlds is an entirely novel contribution to the method. Thus, revised TextWorlds+ treat three classes of propositional relations. First are the active relations between doers, their actions, and the objects of their actions. Next are those descriptive relations that hold between entities and their properties and finally those interactive relations that require a reader or hearer to determine to exist between propositions. A full list of the propositional relations identified by RST is available in Appendix C, Rhetorical Structure Theory.

9.5.3.1 - Element Placement

Once a proposition’s function has been identified, its relations are classified, marked, validated by user interaction, and it is positioned within the Text World. Standard Text World analysis determines the placement of individual Text World elements (e.g., propositional arguments and predicates) within Text Worlds and Sub-worlds strictly by relational class. Actions are depicted as downward flowing vertical arrows connecting doers to target objects or goal states. Descriptions are depicted as horizontal arrows connecting entities to properties and states.

However, standard Text World analysis does not exhaustively consider all relations which may exist between propositions, and instead focuses only upon a limited subset of semantic relations used in the course of building the TextWorlds propositional coherence analysis mechanism. Thus, it specifies no rules for depicting emergent relational propositions, nor even for the small set of semantic relations employed in coherence analysis. Therefore, revised TextWorld analysis not only considers a wider range of semantic and functional relations between propositions, it also specifies the placement of relational propositions as encapsulated related nodes that are labeled by speech-act function.

Thus, the results of propositional and functional analysis are placed within the results of scene analysis as schematic visualizations (e.g., structured concept maps)
of the *conceptual structures* the reader has generated of the conceptual space
detailed by the text. To this point, the TextWorld analysis process is thus a visual
approximation of context-bounded reader comprehension. All that remains is to
determine the **dynamics of transitions** within these conceptual structures and
between these conceptual spaces.

### 9.5.4 - Shift Analysis

So the TextWorld analysis process then moves on to evaluating transitional
dynamics between scenes and TextWorlds. Brief, Time-Zone-internal aspectual
shifts are expected to provide the speakers’ perspective on the action in question.
Larger shifts between Time Zones are also expected, whenever a specific verb tense
does not fit with the basic Time Zone of the current Text World, or consequent to
the use of literary devices and modal expressions such as memories, flashbacks,
predictions, etc. Changes in figurative or epistemic stance indicate a brief shift to a
perspectival, creative, or modal sub world. TextWorlds verb tense analysis expects
a return, or shift back to the primary Time Zone after such literary devices and
modal expressions. Alternatively, aspectual changes spanning Time Zone
boundaries and straight changes to the basic Time Zone are both prime indicators
of an enduring focal shift to a new TextWorld, and thus contribute directly to
TextWorld shift dynamics (which is considered post-scene analysis).

### 9.6 - Analytical Exemplar

Werth claims that the simple and stable mechanisms in the original Text Worlds
approach can be used to model any type of discourse. However, the original Text
Worlds approach does not explicitly address how genres that refer explicitly to the
real world are modelled as the discourse world. And since most of Werth’s
examples are from narrative rather than theoretical or abstract works, it is not
obvious that the original Text Worlds approach is suited, without adaptation, for
application to more technical or abstract genres. Both these problems become
obvious when building TextWorlds for observational scientific reports and highly
conceptual or theoretical works. This fundamental problem is addressed in manual
extended TextWorlds+ Analysis by relaxing the focus on geo-spatial locations and including abstract domains as valid place references.

In the published literature (Lahey 2014; GiovanelliMarcel 2013e; 2013h; 2013d; 2013f; 2013a; 2013b; 2013c; 2013g; Whiteley 2011; Giovanelli 2010; Narayan 2009; Semino 2009; Farrow 2008; Tendahl and Gibbs 2008; Filipović 2007; ‘Text World Theory - Edinburgh Scholarship’ 2007; Yingfan Gao and others 2006; Gavins 2006; Speer and Zacks 2005; Gavins 2005; 2003; Szilas 2003; Pollard 2002; Emmott 2000; Hidalgo 2000; Werth 1999h; 1999e; 1999c; 1999k; 1999g; 1999d; 1999f; 1999j; 1999l; 1999b; 1999i; 1999m; 1999a; 1999n; Ryan 1998; Werth 1995a; 1994; Suh and Trabasso 1993; Werth 1993; Enkvist 1989; Werth 1984), Text worlds analysis is typically executed exclusively on literary or narrative texts. There are notable exceptions (e.g. advertisements, sports reporting (‘Text World Theory - Edinburgh Scholarship’ 2007)) but in the main, the practice of this analytical technique has its home in cognitive poetics and stylistics.

Many analytical tasks in generic analysis involve extraction of timelines, facts, and scenarios from narrative reports. And in many cases, the creation of a narrative from such details is a recommended strategy for analysis of more abstract or conceptual texts. For these reasons, and, with deference to the narrative origins of TextWorlds, in this Section I will describe a brief example of Text World analysis on an introductory excerpt from a standard narrative genre.

This exemplar text is a narrative from the introductory paragraph of a short story:

Example 4 (E4)

I saw the large shouting man assault the door to the treasury and break it down. He looked murderous, even insane. I had never seen anyone so enraged. Even though I was looking through the window, I saw everything clearly. It was cold, so I could see his breath shooting out in steady gouts of steam.

To review the steps taken to prepare the text for visualization as a TextWorld, refer to Section 9.5, Manual Method, and Figure 9.7.1.1 in the following section, Toward A Computational Method.

Exemplar 1 begins by jumping right into the narrative without identifying the speaker, time zone, or place. It is left to the reader fill in these blanks. Text Worlds
accommodates such bald guesses and other more dependable inferences by wrapping inferred entities in square brackets, as a placeholder until better information becomes available and supersedes or confirms the current best guess. The entirety of **Text World 1** represents such a guess. Note that if we assume the speaker is situated in the present and in reality, only because the typical language event for first-hand verbal reports involves immediacy and presence. In the prototypical here and now of the speaker at the moment they speak, any hearer in the immediate vicinity is a potential interlocutor, and for this reason, this world is provisionally presented as a participant accessible world. However, should we discover, for example, that this passage represents a court transcript from 1812 or a dream your friend had yesterday, the location of **Text World 1** in the deictic matrix can be easily amended with precision, from present to past, and from participant accessible, to character accessible, from real to possible, and from proximal to remote, by simply moving the Text World to a new location in the CDG+ Deictic Matrix or the CDG+ Meta-Contextual Frame; or more specifically to a new location along the Proximal dimension.

Now, turning to **Sentence 1**, recall that chunking sentences into propositions relies upon the computational linguistics task of part-of-speech tagging, and then
applies semantic role labeling to approximate both the identification and labelling of each proposition with an RST-style relation (i.e., predicate role label) which holds between joined entities (i.e., arguments).

Several features of this first sentence are interesting. By comparing the speech time with reference time, we determine that the first sentence begins in the past time zone, with the non-deictic first-person pronoun providing a basis for the conclusion that the text is direct speech. Verb tense identifies the narrative events as previous to the speaker’s utterance, adjusts the matrix boundaries, and places Text World 2 in the appropriate deictic zone.

The method assigns world-building meta roles to each entity based on its syntactic role as subject or object. Thus, in Sentence 1, narrator recounts how the man is the doer, or enactor, while the door is a more or less passive object that receives the doers’ action. Receipt of or completion of the action is noted as the enactor’s goal. This goal could be a changed state, a completed action or an ongoing action that summarizes the doers purpose.

Note the use of definite determiners to refer to ‘the man,’ ‘the door,’ and ‘the treasury.’ Since these terms have not been mentioned before, they are explicit cases of unconventional assertions, which introduce new information as though it were previously established background information. These deictically attached entities serve a distinct world-building function.

Next is the evocation of a range of possible frames from the very particular location term, ‘the treasury.’ This is a rare term, that is probably more likely to be found in a historical fantasy novel than in a newswire article. A number of possible FRAMES can be activated by any textual cue, for example in this case the CASTLE frame, or the GOVERNMENT BUILDINGS frame. However, before a reader may
select information from a FRAME, its relevance must be evaluated to select the most relevant FRAME. In this case, features from the situation, including primarily ‘door-breakings,’ seem to fit a medieval action to the more medieval frame of CASTLE. It is not necessary that any or all readers must select this FRAME, nor that they adopt this particular rationale; but only that all activated FRAMES must be evaluated for relevance and that choice must be guided by situation feature matching. If, subsequently, the text provides more details that support another FRAME activation from the term ‘treasury,’ the assessment and indeed the whole analysis can be updated and back-propagated to correct previously reasonable inferences that are currently untenable.

Thus, from this activated, evaluated, and selected CASTLE frame, we retrieve an augmented view of the time as distant, historical past, likely in the Middle Ages. There are also a range of related places, things, doers, scripts, actions and descriptions which may be considered as primed for activation, should this FRAME be proven reliable.
Figure 9.6.3 Text World 2. The deictic matrix shifts to accommodate the unfolding presentation of past actions. The CASTLE frame is evoked by a single term ‘treasury’.

Of interest next is the parallel structure of both description and action. A pronoun standing in place of “the door to the treasury” provides some distance yet provides narrative continuity of an event that must have taken some time (e.g., Breaking down a presumably sturdy door) and that ultimately resulted in success for the attacker (e.g., The goal of breaking door is obtained). This coreference is indicated by double dashed lines.

In Text World 2, labelled actions flow vertically downward using a solid black tri-point arrow to link doers with the objects of their action. Optionally labelled descriptions flow horizontally rightward using an open line-point arrow to link
entities with properties including states, conditions, or manners. Optional labels include existential verbs, which may be [inferred] or explicitly stated. Labelled closed triangular arrows link entities with prepositional relations.

![Diagram of Common Ground](image)

Figure 9.6.4 Common Ground as of Sentence 1. Entities figuring in multiple actions or descriptions become containers for any subsequent actions and descriptions involving that entity and its coreferences.

The CDG Common Ground (CG) is an enduring record of activated and deactivated propositions, numerically indexed to sentence, TextWorld, sub world, perspective shift, and Speech Act (How to Do Things with Words 1962b) Rhetorical Function (Theijssen 2007; Taboada 2006; Taboada and Mann 2006; O'Donnell 2000; Mann and Sandra Thompson 1988). At the end of **Sentence 1**, the CG contains the elements in Figure 9.6.4.

Though brief, Sentence 2 also provides an interesting conceptual structure. The term ‘murderous’ hints at a second frame ANGER (See Figure 9.6.8) and provides some background to the intense, protracted historical assault on real property.
Figure 9.6.5 Sentence 2. Propositions are chunked.

Figure 9.6.6 Text World 2, S2. Inferred alternation and internal aspect shift are indicated by discourse marker badges connected to soft-bounded containers.
Evaluating verb tense, we note an aspectual change to past perfect. This indicates a description of a state at a particular point in time and is thus treated as an internal Aspect Shift, since the Time Zone remains in the past. This understated construction also leaves the reader to infer that ‘even insane’ describes how he looked and is in fact an alternate, more charged description. The standard alternative indicator [or] must be inferred. In this case, the usage of ‘even’ implies intensification, which is a sort of ad hoc graded comparison (e.g., good, better, best) between the previous description and the next.

Sentence three (S3) is fairly complex though brief. This negated graded comparison shifts focus from the main enactor to the narrator and makes a comparative reference to the narrator’s entire conscious experience. This shifts aspect from a state at a single point in time to reflect the Speaker’s view of state as a span. The term ‘enraged’ activates the ANGER frame previously hinted at in S2.

The term ‘never’ indicates a negation, which opens a remote sub-world. When processing the implications of S3, equivalent constructions are inferred that attach more easily to the existing Text World entities. This is made possible by recognizing that ‘anyone’ is the source of a comparison, while [the man] is the unstated target. The implication of this negated graded comparison places [the man] as [more enraged] [than] [anyone ever seen] by the speaker. This rearranged inferred construction is connected to ‘the man’ in TW2. Thus, the implications from a sub-world may provide information in the Text World.

The ANGER frame is relevant for both social and cognitive agents, and provides cued scripts, actions, and descriptions. The frame provides typical explanatory structures for why people might become angry, as well as related outcome conditions. One question left
Figure 9.6.8 Text World 2, Sentence 3. Inferences based on a negated graded comparison Sub World attach to ‘the man’ in TW2. The ANGER frame underscores and explains the destructive attack of S1 and momentary appearance state in S2.
open is what reason the attacker could have for their ‘destructive,’ ‘murderous,’
‘insane-looking’ ‘rage’.

The last word of S3 raises an important point here. Though the precise term
used is ‘enraged,’ which is an adjectival state description, we may infer that other
facets of the multi-faceted conceptual bundle of ‘rage’ are justifiably used or
substituted for the precise term ‘enraged.’ Thus, the reader who contemplates the
man’s reasons for his ‘rage,’ does not violating expectations by using another
related term. Thus, if someone is coldly furious, we may subsequently refer to a
winter of fury, and find that this expression resolves to the same concept, despite
the usage of a distinct propositional form.

Sentence 4 is a complex proposition that relates two propositions with an
explicitly marked relation. ‘Even though’ is subordinating conjunction that can be
used as a discourse marker that indicates an acknowledgement that while P2 might
reasonably be thought to inhibit P4, nevertheless P4. The ‘even though’ or
‘notwithstanding X, nevertheless Y’ construction is used to concede a lesser point
on the way to making a stronger one. This sentence is an example of the
contribution of RST to Text Worlds.

In extended TextWorld diagrams, each entity is a dynamic sub graph that is
updated to include each description and action of that entity, indexed by
TextWorld scene (e.g., enactor & goal). Just as in the Common Ground,
encapsulated entities can thus be collapsed to a single labeled point, effectively
removing the need to display all actions and descriptions in a contiguous chain. By
making this collapse automatic (on a 1 sentence delay) upon encountering deictic
or coreferential terms (e.g., like ‘he’ for ‘the man’), and by making re-expansion
possible through interaction, the great detail of action
Figure 9.6.10 Text World 2 S4. Not all propositional relations are clearly marked with discourse markers for each argument. Inference is often required to label each argument and propositional relation.
and description can be committed to longer-term storage, while the visual buffer is partially and selectively cleared synchronously with the text.

Up to this point, this exposition has not collapsed entities after encountering a coreference, because each sentence so far (S2, S3) has explicitly or implicitly reactivated the primary entity, which effectively resets the delay. In this sentence, all of the action and description to this point is encapsulated with the coreferential term ‘everything.’ Thus, in the next sentence the encapsulated entity ‘everything’ (Figure 9.6.19) can be collapsed and opened again by user interaction as required.

**Sentence 5**

**S5:** *It was cold, so I could see his breath shooting out in steady gouts of steam.*

S5 continues the argumentation begun in S4, presenting a reason and its consequence. Note that the coreference for ‘his’ (e.g., ‘the man’) has been collapsed in the encapsulated entity ‘everything,’ but this coreference is still indicated by double dashed lines. The CDG+ decision to recognize each component of the argument (e.g., Notwithstanding, Concession, Claim, or Reason and Consequence) as a separate encapsulated entity creates some duplication for the narrators’ self-references (e.g., “I saw,” “I was looking,” and “I could see.”).

Duplicate.

This analysis has shown that it is possible for an analyst without extensive linguistic training to produce a dynamic series of TextWorlds, facilitated by the use of interactive digital templates in OmniGraffle and manual lookup of various linguistic class data. The original Text Word approach is significantly extended here through the systematic inclusion of RST relations, which makes the analysis more practical for a wider range of analysts, since argument structures are easily mapped within the TextWorld.

This method can be extended to other genres of text to explore technical and theoretical texts and the abstract Worlds they create wherein time and place are subordinate to experimental manipulation or theoretical argumentation. What is important here is that rich conceptual structures emerge even from simple, brief narratives. This section clearly demonstrates that an approach that aims to produce interactive models should not simply be displayed as static diagrams in a digital or
printed document, so the CDG Models produced here are available online at DropBox Model Link.
9.7 - Future Work

Next steps will include integrating the textual cues extrapolated here (refer to Appendix B. Textual Cues required to facilitate the automation of TextWorlds Method) to identify the language function or phenomena required at each step into a finite-state classification dictionary; implementing the pre-processing algorithm described below, using the computational linguistic tools identified here (see refer to Figure 6.3.4.2) to extract and label the textual indicators and semantic structures required for each step in the General Architecture for Text Engineering (GATE, 2015). For a description of the dynamics of key model mechanisms in terms of the underlying computational mechanisms selected to accomplish their work refer to Figure 6.3.4.2.

With these theoretical and practical preparations in hand, in future work I will complete implementation of a complex text pre-processing system, coupled with a sophisticated user interface to serve simultaneously as a feed-forward model-refinement interface and an interactive personalized visualization of conceptual structure.

9.7.1 - Toward A Computational Method

I have designed a computational TextWorlds methodology to first use NLP text preprocessing to classify each text into its various salient linguistic categories. This preprocessed computationally annotated language data will be used to identify any explicitly deictic expressions to populate a sparse discourse world; to identify word building elements (times, places, objects, actors) and to identify major shifts in these component values that indicate an appropriate location to segment the text into a subsequent TextWorld; to identify modal and literary devices that initiate external Sub-Worlds and aspect which indicates internal perspective shifts; to chunk each sentence into propositions, and to identify the Speech-Act and Rhetorical Structure Theory (RST) language functions advanced by these propositions (Mann and Sandra Thompson 1988; Taboada 2006).
The following computational outline of the TextWorlds Analysis methodology identifies not only the steps necessary for the manual execution of TextWorlds analysis, but also the computational tools available to accomplish these tasks.

Figure 9.7.1.1 Overview of preprocessing workflow design in Orange.
Table 9.7.1.1 The TextWorlds method includes eight tasks that include fifteen steps.
**Task One** simply identifies the Part of Speech (POS) for each word of each sentence.

**Task Two** draws from deep and specific POS data to identify the Tense and Aspect of each verb.

With reference to the previous N sentences, Step 2a identifies when the basic tense (time zone) changes.

**Task Three** identifies sub world Shift markers, including aspect, and draws from an extensive list of modal terms.

**Task Four** chunks the sentence into propositions. This task is also called *dependency relation parser*, and is occasionally called *concept relation identification*, and can also be treated as *semantic role labeling*. As noted in the Features section of this Chapter, a proposition is most often simply two entities joined by a relation. However, since each propositional entity can also be a proposition, any given sentence can achieve a considerable structural complexity.

**Task Five** optionally executes shallow parsing to identify nested, hierarchical phrase (e.g., proposition) structures, and identifies named entities.

**Step Six** applies a discourse parser to provide coh-metrix and other approaches to local coherence,

**Step Seven** resolves anaphor and coreference using the best available algorithms and leverages those results in the creation of Common Ground short-term discourse memory.

**Steps 2a, 4a, 6a, 8a** visualize identified World Shifts, Propositional Relations, Word-building elements, and Discourse Relations.

In Future Work, computational support of TextWorlds will be also extended to automating access to **cultural knowledge resources** in the forms of FrameNet frames, VerbNet, and WordNet synsets, which embed social and procedural semantic relations in computer-accessible format. TextWorlds are also supported by an artificial intelligence short-term memory knowledge-base, the CDG Common Ground, which compares incoming propositions against previously accepted and currently active propositions to evaluate coherence, place emphasis on new
information, and resolve anaphoric references to the correct referent. This specialized AI component will have to be designed.

As a tool for reader-driven computer-assisted analysis, the final m³ [em-cubed] analytical reading environment will offer TextWorlds as an alternative interface for reader-selected digital texts. Thus, in the TextWorlds+ methodological pipeline, NLP text preprocessing will occur prior to the analyst reader’s interaction with the text through reading. In the ideal m³ [em-cubed] workflow, analytic reader evaluation & annotation should take place only after the following preparatory process:

1. A text has been digitized (Gemmell and others 2006) and converted to analyzable format;
2. its document-level and embedded file-level metadata retrieved, exposed, validated, and updated (Vaidhyanathan and others 2012);
3. management of its name and location automated (Vaidhyanathan and others 2012);
4. its document structure identified (Stoffel and others 2010);
5. its (shallow) syntactic structure tagged (Oostdijk and van Halteren 2013);
6. its rhetorical function structure labeled (Uzêda and others 2010; Theijssen 2007; Taboada and Mann 2006; O’Donnell 2000; Mann and Sandra Thompson 1988);
7. its functional style identified (Argamon and others 2007);
8. its baseline conceptual structure roughly outlined.

Ideally, most of these preparatory shallow tasks will be accomplished in m³ [em-cubed]. In this ideal context, the user would then have many Text Model metadata resources (See Figure 5.2.2.3) available to reinforce the organization and structure of their TextWorlds annotation. Any official external metadata resources also serve to constrain arbitrary quality of unassisted online reader classifications, labels, tags, or keyword identification procedures offered in m³ [em-cubed] to facilitate shallow/preparatory analysis.

9.8 - TextWorlds+ Synopsis

The long-term objective of improvements I have offered here to the manual method of Discourse Analysis from Cognitive Discourse Grammar known as the original Text Worlds approach, is to build the extended CDG+ method into a
digital tool for representing the conceptual structure of discourse. This tool for visualizing the analytical process of reasoning over discourse will support analysts as they engage in meaning construction, model construction, reasoning with models, and decision-making with models and knowledge bases (e.g., Cultural Frames and CDG Personal Ground and CDG+ Common Ground).

The extended TextWorlds+ Analysis method proposes a technology-supported process whereby analysts can interactively build domain specific schemas of text contents over time. Its implementation as an interactive inference-logging analytical reading platform would benefit analysts by externalizing and visualizing expert domain specific schemas of propositional content, which are key in expert performance (Ericsson and Lehmann 1996).

For the past thirty years, well-positioned observers (Heuer 1999) have noted that analysts urgently need tools that help them structure information, clarify assumptions, identify inferential chains, and specify uncertainty about their conclusions. Only recently has research in the emerging field of Visual Analytics begun to take these requirements seriously (Thomas and Cook 2005).

Given how much of the analytic process happens internally while reading (leaving very few traces or manipulable artefacts), and considering the analysts’ requirement to externalize, present, and defend their reasoning process and its conclusions, it is surprising to note that tools to help scaffold, organize, or visualize these implicit analytical tasks are not more widely available, particularly for analysts who work closely with text.

While various piecemeal computational tools and visual aides have long been used to complement many analytic procedures (Card and others 1999), very few tools exist to assist the analyst specifically while reading. To my knowledge, none exist to help the analyst by formalizing and automating the meta-observational process of exhaustively noting the formation and evolution of their own conceptualization of a text as it unfolds while reading.

The proposed implementation discussed in Chapter 9, TextWorlds+, will leverage established methods from Natural Language Processing, integrating their output into an interactive frame-based knowledge representation schema for discourse. An interactive system capable of producing basic, interactive TextWorld
models would circumvent the requirement for a high level of linguistic expertise required to undertake deep manual discourse analyses. The proposed system should provide readers with novel interaction with text that focus on identifying connections from their own knowledge bases, and evaluations of textual contents.

Beyond the eventual goal of computational implementation, even as a manually executed Discourse Analysis methodology, given a text and a reader, the extended TextWorlds+ Analysis method can model a start-to-finish account of the conceptual spaces a reader constructs. Johnson-Laird, Werth, Fauconnier, Lakoff, and Langacker have found that the mental and conceptual models readers create help their users to understand the world. It is clear that, regardless of the theory used to explain language, readers are in fact somehow able to process and resolve the ambiguities and challenges of reading.

But mental models are not easily articulated, and once they are, they become unwieldy to self-consciously interact with while working. Given the limitations of human working memory, working simultaneously with multiple mental models and describing their state, contents, and functional dynamics is just not a reasonable expectation of a human analyst, while they are analytically reading, especially for work. Thus, since it is not efficient to do the work twice (once to accomplish it and once more to think about what was done), since we cannot hold all this in our heads simultaneously, and given how quickly we accomplish on-line reading comprehension, the best solution is externalization of the process via computational implementation. The implemented model must provide the interactive mechanisms described in Cognitive Discourse Grammar - the Discourse Model, including TextWorlds, Frames, Common Ground - including all the properties and slots necessary for various classification and reasoning steps. The computational method would pre-process the text to identify all linguistic features that computational linguistics is currently equipped to provide.

I have argued that implementing a discourse model of reasoning should be a top priority for analytical tool builders. Doing so will enable analysts to interact with the process of mental model formation directly, to predict and project baseline comprehension networks for an analyst to use as starting point in model-building, and to reduce the cognitive load of reasoning with rich discourse models by standard means.
Part V: Discussion

The Research Programme presented in this Dissertation has addressed gaps identified in current analytical tools and models with concepts, methods, and models drawn from a transdisciplinary approach to this text-focused general analysis problem space.

Part I, Foundations, proposed a detailed focus on the problem space (Chapter 2, The Problem Situation) and on the domains of research (Chapter 3, The Cognition Literature) and practice (Chapter 4, The Analysis Literature) that provide focus, concepts, and models, that with transdisciplinary application, present solutions to this problem space.

Part II, Modelling Cognitive-Functional Analytical Process, took a broad view of analysis from the highest, most general level. The research in these two Chapters has contributed to building cognitive functional procedural models of analysis (Chapter 5, Flow) and to collecting, analyzing and visualizing digital workflow task-tool usage datasets (Chapter 6, FlowSpaces).

In Part III, Supporting Ubiquitous Shallow Analytical Judgment, the research moved to a focus on analysis at the intermediate, preparatory level, which included discovering and prototyping ways to support cross-cutting cognitive tasks like ubiquitous shallow analytical judgments (Chapter 7, m3 [em-cubed]).

Part IV, Modelling Conceptual Structures in Deep Analytical Reading, introduced and extended the original approach of Cognitive Discourse Grammar, (Chapter 8), noting several problems particular to the original CDG approach and proposing novel solutions to those problems as extended CDG+.

Chapter 9, TextWorlds+, proposed TextWorlds models as a novel, interactive, scene-based unit of text and the TextWorlds+ Analysis Method as the sole means of producing interactive digital TextWorlds.

Part V, Discussion, marks the progress through the Research Programme roadmap proposed in Chapter 1, Introduction. Chapter 10, Conclusion, reviews the key arguments supporting the overall Research Programme and offers further summary of each of the common Challenges, Motivations, Research Questions and Contributions of each Research Project. This discussion highlights the Findings,
Recommendations and proposed Future Work toward improved analytical tool designs from each Chapter.
Chapter 10 - Conclusion

The final Chapter of this Dissertation begins by reviewing the roadmap for the entire transdisciplinary Research Programme that was initially offered in the Introduction. Thus, for each major Research Project, this Chapter reviews problems, goals, products, questions and recommendations for future work.

The dissertation began by synthesizing Challenges in the analysis problem space into two Core Problems relating to the Cognitive Limits of human analysts and Functional Limits of analytical tools. These Core Problems were then further specified in relation to specific challenges facing analysts, tool makers, procedural models of analysis, and analytical tools (for the complete discussion, refer to Section 1.1, Challenges, Chapter 2, The Problem Situation, and Section 10.1, Challenges Reviewed).

Next, further Motivation for the Research Programme were characterized primarily as Goals for Research Products that this dissertation aimed to produce. In general terms, research activities and Products include extending and building conceptual and cognitive models, designing methods and prototypes, and building toward analytical tools. Thus, each model, method, and prototype tool produced here has a specific purpose that addresses a Core Problem, a particular Challenge in the analysis problem space, a general Research Question, and a project-specific Focus Question. (For the complete discussion, Refer to Section 1.2, Motivations, and Section 10.1.1, Motivations Revisited.)

Two general Research Questions were first presented to demark the theoretical grounds for these Research Goals and Products and the specific methods designed for their production. Accordingly, this Research Programme has generally inquired into 1) how theory and practice can inform tool design and specifically 2) how to transparently produce low-cost empirical datasets to model invisible functional and cognitive processes. At a more detailed level, specific major (and minor) Focus Questions were also posed for each Research Project (for the complete discussion, Refer to Section 1.4, Research Questions and Section 10.2, Research Questions Answered.)

Next, Research Contributions were listed by research project and purpose. These primarily include:
• Three Cognitive-Functional Models; including 1) a model of general analysis, 2) a model of discourse processing, and 3) a model of analytical reading.

• Two Prototype Systems; including 1) a text logging, event analysis system for gathering multi-tasking workflow data, and 2) a prototype system for recording shallow judgment and content evaluation using tagging and highlighting.

• Two Case Studies: including 1) a data-collection case study and 2) a data analysis case study.

• Three visual concept models; including 1) a contextual situation model for understanding the discourse processing situation, and 2) a visual model of the CDG+ Discourse Processing Model, and 3) a task model of participant actions taken during reading, according to the literature on discourse processing.

• One index of shortcomings in the original CDG approach.

• Four further novel methods that address specific problems with the original CDG approach; including 1) a Deictic Matrix for indexing the deictic origin of each scene in the Text World Discourse Model, 2) a Meta-Contextual Frame for capturing individual contextual and deictic evaluations of each discourse element, 3) outlines of a Method for Manifest Semantic Analysis, and 4) a specification for treatment of deactivated propositions in the CDG/+ Common Ground as constituting the CDG+ Personal Ground.

• Two methods for executing TextWorlds Analysis; including 1) an extended manual method and 2) outlines of computational method.

• And finally, one exemplar of the extended manual method of TextWorld Analysis.

Finally, to complete this review of the Research Programme roadmap, Chapter 1, Introduction, concluded with Chapter Overview and Executive Summary, each of which respectively provided a thorough and a succinct description of each Research Project by Chapter. The following five sections each review and discuss each of these sections of the original Introduction in some detail, contributing to a
broad review discussion of each a major Research Project by its problems, goals, products, and recommendations for future work.

10.1 - Challenges Reviewed

The challenges addressed in this Research Programme can be summarized as focusing on 1) difficult tasks in the analysis process (e.g., externalizing the results of classification and reading), 2) identifying gaps and omissions in a published reference model of the analysis process, and 3) on identifying shortcomings in the match between tasks in the analysis process and tasks supported by analytical tools. This research has focused on two particularly difficult tasks in the analysis process; namely, externalizing the results of both shallow analytical classification and of deeper analytical reading. Accordingly, this research has addressed two core problems at several levels. Cognitive Limits of human analysts and Functional Limits of analytical tools are addressed at the broad level of multi-tasking tool usage in general analysis and at the increasingly specific levels of analytical preparation and execution of analytical reading.

This research has first considered Cognitive Limits of human analysts in relation to the Procedural Complexity of the analysis process, and to the analyst’s Meta-cognitive Awareness of their own cognitive (internal knowledge) states and functional (external digital procedural) states. Thus, the difficulty of challenges associated with externalizing internal models and procedural progress should be expected to be related to the analyst’s Meta-cognitive capacity, their procedural expertise, and to variations in levels of their moment-by-moment meta-cognitive awareness. These hypotheses can be tested in future work.

Next, the Functional Limits of analytical tools were described at length, offering an extensive consideration of specific shortcomings classified generally in relation to 1) Toolmakers’ Expertise & Knowledge, 2) Challenging Data Features, and 3) Functional Coverage of Tasks in the Analysis Model. Specifically, this dissertation has argued that functional limitations of current analytical tools can first be attributed to deficits in the Analytical Tool-makers’ Expertise and Knowledge. Historically, tool-makers have noted challenges with both Eliciting Tool Requirements from analysts and with Implementing Relevant Findings from
Cognitive Science. This dissertation argues that these practical challenges are related to specific knowledge deficits, including the Analytical Toolmakers’ Knowledge of Real-World Analysis Process, their knowledge of Real-World Tool Usage, and their knowledge of what Findings are Relevant from Cognitive Science.

Of course, certain types of data are more difficult to analyze computationally, so Functional Limits of analytical tools are also related to the Scale, Modality, Complexity, and Type of Data considered. Each of these data challenges should thus be expected to compound the challenges associated with the computational analysis of text.

Finally, this Dissertation Research Programme has focused on gaps in the Functional Coverage of both individual analytical Task Models, and of the overall general Analysis Model, and addressed gaps in each by revising these Models at general and specific levels, and by producing datasets, prototypes, and recommendations for tool design.

10.1.1 - Gaps Motivated

Beyond serving as resources for identifying and resolving gaps in analytical tool coverage, the Literature Chapters also highlight gaps in two key analytical domains. Thus, Cognitive Linguistics researchers must be willing to extend beyond their experimental focus upon the observable neural dynamics of language users to also consider 1) the actual communication function of language, 2) a deeper understanding and theory of social interaction in discourse, and 3) the implications of social and cultural structures retained in memory on language understanding. It should be clear that the CDG/+ approach contributes in this area in Cognitive Linguistics, by addressing each of these three needs in turn. Moreover, in order to meet the field’s original and updated goals, Visual Analytics tool builders must be willing to reach beyond their existing tools and familiar approaches to identify analytical tasks, phases, and methods as targets for building analytical support.

While the handful of modern VA tools that analyze textual data do help to address the larger end of the data scale problem, where analysts must sift through large textual collections to find documents worth reading, at the other end of the scale (e.g., engaging with individual documents) analytical tools offer very few analytical affordances, particularly when preparing for analysis and in analytical
reading. What is missing in today’s focus on scalability and big data, is a detailed, multi-level focus on the human end of that scale whereby the experience and practice of using digital tools could be used to improve the design of tools to support analyst-driven analysis of difficult multi-modal analytical data sources like text.

At the most general level, the ‘human scale’ focus of this research underscores a desire to make a contribution toward understanding the cognitive and functional tasks required to step through the entire analysis process. This contribution is met by contributing to the knowledge of digital tool usage workflows; specifically, of which tools to use with which data, what procedures are required to prepare data for each tool, and how to both prepare and execute the required analytical work. This general level of research focus is provided by Chapter 6, FlowSpaces.

At an intermediate level, the ‘human scale’ focus means understanding that the preparatory phase of analysis involves basic cognitive analytical tasks (e.g., shallow judgments) which are not sufficiently supported in current VA tools. This level of contribution is therefore met by building means and mechanisms to more effectively capture and reuse this shallow preparatory sense-making, in Chapter 7, m³ [em-cubed].

At the deepest level, this ‘human scale’ focus means understanding, supporting, and modelling the language comprehension process that unfolds as the analyst reads the text. This level of research focus is provided by the Chapter 8, CDG+, and Chapter 9, TextWorlds+.

10.1.2 - Goal Levels Met

Overall, at each of the three levels of research focus identified above, (e.g., broad, shallow, and deep levels) the goal of this dissertation is to focus on identifying, understanding, and supporting the analysis process through the twin lenses of cognitive and functional tasks. Practically speaking, this has meant identifying general classes of procedural actions (e.g., activities, functions, or tasks), both digital and analog, undertaken in the process of general analysis and in analytical reading.
At the **broader cognitive level**, the focus is on understanding and identifying the **general analysis process at the level of cognitive and procedural functions** identified in the literature on cognition and analysis (Chapter 5, *Flow*).

At the broadest functional level, the focus is on understanding and identifying the **tool-supported analysis process at the level of workflows among digital task functions as shown in digital tool usage across dedicated Task Spaces** (Chapter 6, *FlowSpaces*).

At the intermediate cognitive level, the focus is on understanding and identifying the **preparatory analysis process at the level of specific tasks required to accomplish** and record shallow analytical judgements while evaluating documents and ‘skim reading’ contents in the preparatory phases (Chapter 3, *The Cognition Literature*, and Chapter 4, *The Analysis Literature*).

At the intermediate functional level, the focus is on identifying, integrating, and prototyping tools for each preparatory step the analytical reading process at the level of capturing shallow analytical judgment during the preparatory content evaluation and shallow reading phase (Chapter 7, *m3 [em-cubed]*).

At the deepest cognitive level, the focus is on understanding and identifying the **discourse comprehension process** at the level of identifying the specific tasks and dynamics required to understand linguistic exchanges in the execution phase of text-based document analysis, as patterned after (Chapter 8, *CDG+*).

At the deepest functional level, the focus of this research is on operationalizing the analytical reading process at the level of identifying specific tasks required to accomplish analytical reading in the execution phase of analysis, in order to externalize mental models created by the analyst in the course of analytical reading (Chapter 9, *TextWorlds+*).

This deepest functional level involves many tasks. The first requirement is identifying computational means of accomplishing tasks required to do analytical reading in the execution phases. The second requirement is to design an algorithm to execute computational proxies for steps in this sequence of requisite reading tasks. Finally, in future work, I plan to build a computational reading system for interacting with the products of the computational reading algorithm in the form of interactive Situation Indexed Event Indexed TextWorlds. (Chapter 9, *TextWorlds+*).
10.2 - Research Questions Answered

My first general research question (Q1) asked how cognitive theory and analytical practice can inform the design of analytical tools. My method (see Section 10.3, Methods Developed) for answering this question involved the development of a multi-modal (e.g., cognitive & functional) multi-level (e.g., broad, shallow, deep) multi-project (e.g., Flow, FlowSpaces, m3 [em-cubed], CDG+, and TextWorlds+) approach.

To consider the pragmatic dimension of analytical practice in answering Q1, I have used the foregoing theoretical levels of detail to situate three layered levels of analytical practice discussed in Section 1.2, Motivations. To review, these include the general overview level of the entire analysis process by steps or task, the intermediate, shallow level of preparation for deeper analysis, and the deepest level of analytical execution.

Moreover, I have also addressed the cognitive dimension of Q1 simultaneously from several complimentary levels of detail in Cognitive Theory. To approach the first broad level, Chapter 5, Flow, draws from cognitive theories of higher-level cognition to broadly describe general analytical process. At the same time, the approach to the second, intermediate level draws from cognitive theory of simple evaluative judgments to capture sense-making in Chapter 7, m3 [em-cubed]. And finally, to approach the deeper level of detail, Chapter 8, CDG+, draws from cognitive theories of discourse processing, which describes the narrow, specific analytical reading process.

The answer offered to Q1 proposes to improve knowledge of tool users and tool builders. Given that Visual Analytics (VA) tool-builders are trained as engineers and VA tool-users are trained as analysts, this research aims to improve trans-domain understanding of real-world analytic workflows with digital tools, so as to improve both tool design and tool usage.

Tool builders should understand that every analytical task in the analysis process has preparatory and execution phases, each with a variable network of related and requisite sub-tasks. At the functional level, neither preparatory nor execution phases have a fixed number or order of sub-tasks, though within a network of possible task transitions, a small number of sequences are highly
probable. However, at the cognitive level, some tasks should be considered as having requisite sub-tasks. For example, classification requires knowledge of item features and class features and the matching between them. Yet, most often, requisite cognitive steps are executed silently and (nearly) simultaneously without producing a functional record. Thus, the creation of functionality in digital tools that captures the products of silently executed preparatory and executive sub-tasks is one area where the design of analytical tools can be improved. This is addressed in Part III, Supporting Ubiquitous Shallow Analytical Judgment by Chapter 7, m3 [em-cubed] (multi-modal-meta) analytical Media Manager.

My second general research question (Q2) asked how transparent, low-cost observational methods can be designed to produce empirical datasets that capture functional and cognitive processes. In both real-world prototypes and ideal-world recommendations, my methodological approach aims to be almost entirely invisible to the user as the preferred means of reducing the cost of tool usage.

For example, in the FlowSpaces workflow monitoring system, while initial setup and periodic maintenance are involved, the requirement for ongoing interaction of user with the system is negligible. The user can simply continue to work as they always have; except that applications that have been classified with a task type will open by default in a task-specific desktop workspace. Interaction is only necessary to identify boundary-spanning tool usage. Similarly, the m3 [em-cubed] tagging system is designed to be activated within any application, as a meta-tagging framework that can attach the contents of shallow evaluative judgments to the file container or to document contents as required. Finally, the TextWorlds+ system for externalizing mental models will be designed to approximate invisible cognitive processing steps that readers engage in to produce meaning.

Thus, the answers offered to both Q1 and Q2 propose to exemplify the design of observational methods for capturing the process and products of functional and cognitive tasks, to build procedural models for them, and to prototype designs for analytical tools to support the same.

**Focus Q1: Flow - Extention to a Model of General Analysis**

In what cognitive, functional, and procedural areas can the Sense-Making model be extended to better characterize the process of general analysis?
Whereas the Sense-Making model originated in the analytical domain of ‘all-source intelligence analysis,’ which is an exclusive and narrow practice, general analysis is common and broadly practiced across domains by any persons collecting, integrating, and curating multi-modal data sources for deeper analysis. Though the field of Visual Analytics was originally focused on military and intelligence security analysis, in recent years focus has broadened to a wider range of analytical domains and tasks.

Nevertheless, in its first ten years the field of Visual Analytics has neither produced nor uniformly adopted an empirical framework for understanding analytical tool usage across each cognitive and functional step in the analysis process. Notwithstanding this lack of both a cognitive model of analysis and a universal lexicon of analytical functions, the Sense-Making model of analysis is widely cited within VA as a reference model for representing analytic process and workflow.

However, the Sense-Making model excludes a number of functional and cognitive steps that the wider literature on integrative all-source analysis clearly indicate are of central importance. Moreover, practical experience makes it clear that the general analysis process includes the preparatory functions of search, filtering and integrative schema-building, and the time consuming process of collecting and organizing data into datasets.

Though the Sense-Making model is based on the theory of information foraging, it does not sufficiently highlight the shallow cognitive tasks involved in analytical preparation for execution of deeper analysis. In particular, it fails to emphasize some of the most time-consuming cognitive tasks of the analysis process during preparatory sorting and classification, in sense-making while reading, and overall, in schematization throughout the entire analytical process.

Key functions omitted in the Sense-Making model include: goal setting & target specification; the selection, identification, evaluation, classification, labelling, collection and annotation of data features and objects; judgments of comparison, classification, relevance, coherence, inference, entailment, blending; moving, coordinating, organizing; re-representing; modelling; hypothesizing; testing; disconfirming; writing; framing; editing; and presenting data and evidence.
In response to these gaps, I have extended and refined Pirolli & Cards’ Sense-Making model of analysis to include cognitive and functional steps omitted from the original formulation (See Chapter 5, Flow). In order to rank among preparatory analytical task functions not typically viewed as part of analysis, I have also designed and implemented a computational framework for gathering data on multi-tool multi-tasking multi-activity workflows (See Chapter 6, FlowSpaces) in functional terms.

Focus Q2: FlowSpaces - General Analysis as Transitions though Activity Space over Time

What does general analysis look like over time?

At the broadest functional level, the purpose of FlowSpaces was first to develop a text-analytic system for extracting event data from computer-generated logs, and thereby to produce observational data on longitudinal digital-functional workflow. FlowSpaces then required the design of a visual-analytical methodology for multi-level longitudinal data, and the design interactive visualizations that reveal workflow patterns as schematized by the prototypical task functions of digital tools and by the interactive reclassification of current tasks.

FlowSpaces has thus designed and implemented a computational framework for gathering data on multi-tool multi-tasking multi-activity workflows (See Chapter 6, FlowSpaces). This framework has produced empirical data on multi-tool multi-tasking workflows which has raised its own important questions, such as: How should multi-tool multi-tasking workflows be described? What distribution does this data follow? What methods are appropriate for analyzing this data? (These questions were addressed in Section 6.8, The Exploratory Analysis Case Study.)

Good answers to these questions should enable analytical tool-builders to use this framework, model, and method to better understand the analysis process with particular reference to their own tools, and thus to uncover clear leverage points for improved analytical system design. VA tool-users could use this framework to gain an understanding of their own procedural workflow patterns, to identify bottlenecks or shortcuts in their workflows.
Figure 10.4.1. Stacked Timeline View of Raw Tool Usage Events over time. The minutes of the day are on the horizontal axis, while each day in a four-year period is on the vertical axis.

**FQ2.1: FlowSpaces - Functional Activity Spaces**

*In what functional areas does a general analyst use digital tools?*

The FlowSpaces system is calibrated to collect data on app usages within 16 functional areas; Seek, Collect, Sort, Tag, Manage, Read, Note, Write, Converse, Convert, Design, Develop, Emulate, Analyze, Administer, and Display.
FQ2.2: FlowSpaces - Usage Patterns of Digital Tool Usage

What usage patterns exist in digital multi-tool workflows?

The FlowSpaces system has gathered longitudinal data on a single user, so the unit of study is the (app usage) event rather than the analyst. The data is therefore not suited to making conclusions about analysts generally. The data is, however, useful for typifying a range of task-focused user interaction events. The interaction events of interest are application usage events in relation to both the previously used Application and Activity Space. Using visual analytical tools for exploratory data analysis, I have identified three distinct patterns in app usage transitions.

1) Continuous use (within space, within app transitions only occurring between application windows) indicative of focused usage sessions.
2) Multi-tool single-tasking use (within space, between app transitions) indicating focused co-usage of tools
3) Task Interrupted transitional use (between space, between app) indicating leaving both app and task space behind or transition to another task.
Figure 10.4.3. All App Usage Events on a single timeline aggregated with a 21-day floating average.

Figure 10.4.4. All App Usage Events by continuous usage, multi-app usage, and task-transitional usage, and by frequency-weighted grouping of Activity Spaces.

Viewed in aggregate with a logarithmic scale for duration, on a single horizontal timeline, (Figure 10.4.2) some linear trends become apparent, but with
much ‘noise’ and occlusion. Viewing the data through Activity Space classes resolves some of the occlusion, and reveals that each Activity class has a distinct pattern. Simultaneously viewing the data through Application Usage Transition classes (Figure 10.4.3) partitions out components of the non-linear noise and reveals a ‘shadow’ pattern.

First, (Figure 10.4.4) since application usage events are indexed by the Active Window Title and Application Name, when the former changes but the latter remains the same in a single Activity space, the data indicates focused app usage.

![Figure 10.4.4. App Usage Events filtered for continuous single-app usage in a focused single-tasking Activity Space. Most continuously used Spaces are Search, Manage, Converse, Analyze, and Write.](image)

Second, (Figure 10.4.5), if the Application in use prior is not the same as the current app, but Activity space remains constant, the data indicates a task-focused (single-task or Activity Space) multi-application usage scenario. Finally, (Figure 10.4.6), transitions between Activity Spaces indicate a shift in focus.
Figure 10.4.5. App Usage Events filtered for transitional multi-app usage in a focused single-tasking Activity Space. By sum of Event Counts, the most used Spaces are Search, Manage, Collect, Converse, Write, Analyze, Note, Emulate, Admin, Display.

Figure 10.4.6. App Usage Events filtered for transitional multi-app multi-tasking usage across multi-tasking Activity Spaces. This pattern shadows that of focused multi-app usage.

With a framework that makes it possible to produce an empirical characterization of multi-tool multi-tasking multi-functional digital tool usage
workflows, and given an extended and refined cognitive-functional Flow model of general analysis that explicitly recognizes the long and difficult pre-analytic preparatory process, it becomes possible to identify which cognitive or functional steps are under-supported by VA tools today.

Reviewing the top VA tools available today in the light of this empirical analytical workflow data and considering the extended Flow model of the analysis process, this research has identified a failure to adequately support some of the most time-consuming phases of the analysis process, particularly during preparatory sorting and classification, while reading, and overall, in schematization throughout the entire analytical process. With the identification of these gaps, it becomes possible to produce and implement recommendations for how analytical tool-builders can improve their tools as prototype stopgaps. These are carried out in m³ [em-cubed] and proposed for implementations in TextWorlds+.

Focus Q3: m³ [em-cubed] - Prototyping Support for Functional Gaps

In what functional areas can analytical tool-builders improve support of under-supported phases of analysis?

This dissertation has argued that analytical tools could better support procedural and conceptual annotation during preparation and execution of both shallow-level skim reading and deep-level analytical reading. Preparatory shallow analytic judgments (e.g., evaluations, assessments, and classifications) can easily be captured and schematized with labelling or annotation functionality like tagging. While reading, shallow analytical judgments can easily be captured using annotation functions like highlighting, callouts (e.g. sticky notes), and also with tags.

However, these judgments or simple classifications create new digital objects which multiply quickly. Thus, an effective solution requires low-cost options for managing both these objects and their connections to their originating/anchoring digital objects. Moreover, given how the linear process of reading remains an essential phase of analysis, and yet how few tools offer textual analysis, the field of VA must also focus on how best to facilitate the capture and modelling of deeper analytic judgments in order to model the production and refinement of mental models while reading.
A full response to these questions has involved the implementation of **automated workflow procedural tagging**, **ubiquitous file tagging**, **controlled taxonomic highlight labelling**, and the application of **Cognitive Discourse Grammar**’s **TextWorlds analysis** from **Cognitive Linguistics** to model and visualize the online discourse comprehension process involved in the deeper analysis of reading. These responses are positioned as Chapters that describe **FlowSpaces**, the workflow monitoring system, m³ [em-cubed], the procedural and shallow conceptual tagging system, and **TextWorlds+**, as a method for modelling and visualizing the discourse comprehension process described in **CDG+**.

**Focus Q4: CDG+**

What is **Cognitive Discourse Grammar**, what are its shortcomings, and how can it be improved for implementation in analytical tools to support analytical reading?

Cognitive Discourse Grammar is a Discourse Analysis approach based in the principles of Cognitive Linguistics that captures and externalizes the conceptual models of text produced in the course of discourse comprehension. It is an ambitious approach to visual representation of human conceptual structures, at the level of the Propositions contained within text and those activated by the text from the cultural Knowledge Frames of discourse participants (e.g., readers), and their functional Relations, as scene-indexed TextWorlds.

Its most basic shortcomings regard various small coherence and consistency problems with the original Text Worlds monograph that can largely be attributed to its posthumous publication, and can be resolved in consistent rearticulation. Most importantly, this research has distinguished between the original problems which produced the original CDG **Text Worlds** approach, those of the current stylistics approach of **Text World Theory**, and those whose extensions and revisions are contributed here as CDG+. Thus, this research has further distinguished between **Text World Diagrams** and **TextWorld+ Models**: whereas the former are static handmade visualizations made for the purposes of summary and publication; the latter are computationally generated and dynamically interactive.

Though **Text Worlds** are just one part of the **CDG+ Discourse Model**, stylisticians use a simplified analytical method to effectively break off Text Worlds from the original CDG approach. Moreover, both the original CDG approach and
stylistic Text World Theory build *static Text World diagrams* whereas CDG+ aims toward building increasingly *interactive and dynamic TextWorld+ Models*. Neither the original CDG approach nor stylistic Text World Theory have fully and satisfactorily specified a manual method for Text World analysis. This is resolved in CDG+ by specifying a manual method of TextWorlds+ Analysis (in Chapter 9).

Moreover, both require a high degree of linguistic expertise which makes both original and stylistic approaches costly. Both novice and general analysts and analytical tool-builders are therefore expected to have procedural knowledge gaps regarding the foundations and requisite analytical tasks of the original and stylistic TextWorlds Analysis methods. Therefore a more complete description of both the original CDG approach and the extended CDG+ approach is thus required to introduce the full capacity of the original and extended Text World approaches to unfamiliar text analysts, analytical tool-builders, and discourse linguists. Accordingly, the CDG+ Context Model encompasses layered aspects of the CDG+ Discourse Processing Model, which constrains the dynamics within which the CDG+ Discourse Model is produced in the reader’s mind in the natural course of Discourse Comprehension.

Then, there is the problem of an underspecified deictic starting point in the original intuitive approach to the creation of Text Worlds, and the fact that the original CDG Common Ground does not specify what happens to accepted propositions in the Common Ground once they lose their activation focus. These problems are respectively resolved in CDG+ by providing a CDG+ Deictic Matrix the Meta-Contextual Frame and the CDG+ Personal Ground. The former two enabled deictic and contextual classification of individual words, propositions, functional relations, scenes, and Text Worlds. The latter has extended the short-term memory of the CDG Common Ground into the longer-term memory as the CDG+ Personal Ground as and when the activation of mutually accepted propositions decays. Finally, as noted in Chapter 5, Flow, the problem of an incompletely specified Cognitive-Functional Flow Model of Analysis is partly addressed by specifying the requisite analytical reading sub-task network fully in relation to the CDG+ Discourse Processing Model.

Before the reader can consider accepting the claims of a revised and extended CDG+, they ought to first have a clear understanding of the original approach. To
this end, Chapter 8, CDG+, should be read as a meta-introduction to CDG that offers motivation towards further reader-directed exploration.

**Focus Q5: TextWorlds+**

*How can models of human discourse processing leverage existing advances in computational approaches to language analysis to build interactive models of human conceptualization?*

A TextWorlds’ real unit length is not solely defined by physical contiguity in text, but is instead a function of directed authorial and reader focus, and is thus variable. But TextWorld length can also be measured in words, propositions (e.g., connected phrases), sentences, paragraphs or sections.

**10.2.6 - Summary of Research Questions Answered**

This Section 10.3, Research Questions Answered, has collected the answers provided throughout this Research Programme to the two general Research Questions that were initially posed as theoretical grounds for specific Research Products produced here. These answers have introduced the methodology employed for coming to answer just how *theory and practice can inform tool design* generally and specifically how *empirical datasets can be inexpensively and transparently produced to model invisible functional and cognitive processes*. One further project-specific Focus Question is posed for each Research Project (and some minor project-specific focus questions for FlowSpaces).

**10.3 - Methods Developed**

This research is transdisciplinary and has thus sought solutions between disciplines. The transdisciplinary method used in this Research Programme has included the identification of gaps (See Appendix A - Transdisciplinary Gap Analysis) and a problem situation in one domain (Visual Analytics and VA tools) and the identification of complimentary solutions from another domain. Thus, this dissertation research has achieved its initial focus by identifying gaps in current VA tool functionality and literature concerning:
• tools or theory for capturing and visualizing mental model/schema creation in VA.
• tools or theory for adequately support the task of analytical reading, in either preparation or execution phases in VA.
• tools or approaches for capturing and visualizing workflow data in VA.

Thus, this Research Programme has reviewed both available analytical tools and the literature and has found no approaches or tools for capturing and visualizing mental model and schema creation in VA. Moreover, the literature and tool review have found that no tools adequately support the task of analytical reading, in either preparation or execution phases. Finally, the literature and tool review have found no tools or approaches that support capturing, analyzing, visualizing user interaction in longitudinal workflow data.

However, the Literature Review Chapters have identified extensive theoretical consideration and empirical investigation of these cognitive and functional analytical tasks (e.g., model formation & reading) in Cognitive Science (Chapter 3, The Cognition Literature) and Analysis (Chapter 4, The Analysis Literature) domains. These Chapters have thus summarized extensive findings of the topics of mental model formation, analytical reading, and shallow analytical judgment in complimentary Cognitive Science domains (e.g., analysis, decision-making, discourse processing & language comprehension).

Part I, Foundations, has therefore identified this gap dynamic as sufficient justification of the need for transdisciplinary application of theory from specialized domains in Cognitive Science to problem areas in Analysis and analytical tool design. Accordingly, this dissertation has argued that the challenges of Cognitive Limits of human analysts, and Functional Limits of analytical tools can be addressed by specific tool building and data generating goals. Thus, this research proceeds on the premise that the helping with currently-under-supported, difficult analytical tasks like analytical reading requires a new, transdisciplinary approach that integrates knowledge of this cognitive task from domains in Cognitive Science with knowledge of Analytical practice into the design of analytical tools. Likewise, helping with the externalization of cognitive and functional processes requires a new transdisciplinary knowledge-integrating approach to designing analytical tools capable of capturing and reflecting this data back to the user. This research should thus be understood as novel, theory-based tool and methodology design.
The particular research methodologies developed here to address the different levels of this research are a product of the tension between facts, requirements, and potentials. These existential modalities of actuality, necessity, and possibility are determined by discovering what is done, what must be done, and what could be done.

To study what is done, this dissertation research has created FlowSpaces as a workflow monitoring system to generate real time data on creative and analytic workflows. By arranging task-specific workspaces and building task-specific application collections into each workspace, this research has begun to build optimized subroutines into a system that changes what is done by implementing what could be done and making it possible to achieve new goals of easing transitions between related task types.

To establish what must be done at the optimized subroutine end of the scale, this dissertation research has created m³ [em-cubed], or multi-modal analytical media and metadata, the meta-data fuelled, analytical multi-media management & mobilization system. The m³ [em-cubed] prototype currently provides automated workflow procedural tagging to assist with analytical way-finding, ubiquitous file tagging to capture shallow conceptual analysis of media containers, controlled taxonomic highlight labelling to capture surface analysis of textual media contents.

To establish what must be done at the highest level of methodological description, this dissertation research has excavated, repaired and reformulated Cognitive Discourse Grammar as not only the extensive machinery behind a representational mental model of discourse contents but also as a manual methodology for TextWorlds analysis that can be implemented as a evaluable discourse processing model that helps analysts by providing Visual Analytics of in depth conceptual contents synchronized to reading progress.

In light of this discussion of the methods designed and developed here, the following sections outline the procedures undertaken to answer each general Research Question.

**Q1 Method Outlined**

1. Identify the conceptual models of the analysis process used as reference material by analytical tool-makers.
2. Review the cognitive and analytical literatures relevant to the analysis process.

3. Identify cognitive and functional tasks of the analysis process.

4. Revise the standard conceptual model of the analysis process with missing cognitive and functional tasks.

5. Assess the functional support of analytical tools relative to the revised conceptual model of analysis.

6. Identify analytical functions that are not supported by analytical tools.

7. Select one analytical function not (well) supported by analytical tools for prototyping (e.g., analytical reading).

8. Index digital tools by function.

9. Create a digital desktop workspace environment indexed by digital tool function.

10. Observe and record how the analyst uses digital tools over time.


12. Identify cognitive models of key analytical functions (e.g., analytical reading) in order to understand detailed cognitive-functional dynamics for those functions.

13. Use detailed cognitive models of cognitive-functional dynamics for selected analytical functions to identify analyst requirements pertaining to the support of that analytical function and its relevant sub-tasks in analytical tools.

14. Create a Task Model for analytical reading to summarize and operationalize analyst requirements for this task.

15. (In future work) Build cognitive models of key analytical functions (e.g., Task Models) into usable analytical tools, to properly support the cognitive-functional dynamics of that function and its constituent sub-tasks.

**Q2 Method Outlined**

This question pertains to satisfying feature criteria for the design of observational methods that create data on functional and cognitive process and prototype tools that support gap tasks. This section has considered the extent to which each of the data-gathering systems designed for my dissertation meet the criteria of **low-cost** and **transparency**.
FlowSpaces was designed to transparently capture functional flow in digital tool usage over time, using a system-of-systems approach to

1. Improve the user interface of the standard virtual multi-desktop application.
2. Create log data on user interaction events.
3. Develop text-analytic system for extracting event data.
4. Produce observational data on longitudinal digital-functional workflow.
5. Design visual-analytical methodology for multi-level longitudinal data.
6. Design interactive visualizations that reveal workflow patterns.

m³ [em-cubed] was designed to inexpensively capture shallow analysis in preparatory data analysis over time, using a system-of-systems approach to

1. capture and update procedural workflow states on digital files using tags
2. capture and update shallow cognitive evaluation, classification, and labelling on digital files using tags;
3. create a record of tag assignments, by file, over time,
4. capture and update deeper cognitive evaluation, classification, and labelling on digital file contents using highlights.
5. Both tags and highlights are low-cost assignments, though search through (tag or highlighter) lists prior to assignment of either creates a cognitive load for the classification task.
6. Thus, to reduce load of search through tag or highlighter list prior to assignment, m³ [em-cubed] should design an organization & presentation system for tags and highlights.

The manual and computational TextWorlds+ analysis methods have been designed and will be implemented to generate and interactively refine a profoundly deep analysis of the conceptual structure of text implicitly produced by goal-driven analytical readers over time. The manual Method of TextWorld Analysis uses a detailed cognitive-linguistic model of discourse and event processing (the CDG+ Discourse Processing Model) to track and model the process of reading. This method produces interactive, TextWorld visualizations. TextWorlds are thus based upon understanding and observing the dynamics of discourse processing.

So, tool-builders should build support for the cognitive-functional dynamics of reading and its constituent sub-tasks into a useful analytical tool that analyst readers can use to:
1. capture and update the situations identified in the text as TextWorlds
2. Identify the scene background
3. Identify the action & description foreground
4. Segment Events in the action
5. Interact with and update their TextWorlds cognitive models

10.4 - Objectives Accomplished

My first objective was to chart the disciplinary breadth and theoretical depth that together establish the foundation and domain for my technical and methodological designs and for my contributions to theory. This was accomplished in Chapter 1, Introduction, and in the Chapters of Part I, Foundations.

Next, in order to make a scholarly contribution to the practice and study of human cognition in analysis, I have presented both conceptual models of relevant cognitive-functional analysis processes, theory-based designs and development of original research methods, and prototype digital analytical tools. First, I have extended the Sense-Making model of analysis to include missing cognitive and functional tasks, and to acknowledge the key role of cognition throughout both preparation and execution phases of analysis Chapter 5, Flow. Second, I have defined and visualized multi-level multi-modal multi-media analysis. Third, I have collected and visualized multi-tool multi-tasking digital workflow data. Fourth, I have analyzed and leveraged multi-level digital workflow data to produce recommendations for improved tool design, in Chapter 6, FlowSpaces. Fifth, I have prototyped process-independent annotation tools for multimodal analytical data in Chapter 7, m3 [em-cubed]. Finally, I have extended the original Cognitive Discourse Grammar in Chapter 8, CDG+, and articulated and exemplified the manual TextWorlds Analysis method in Chapter 9, TextWorlds+.

Overall this research has established plausible methods for capturing and mobilizing basic sense-making at relevant points in the analytic process, across analytic contexts, in direct response to Research Question 2 (How can transparent, low-cost observational methods be designed to produce empirical datasets that capture functional and cognitive processes?)
10.4.1 - Research Products Produced

In general, this dissertation has focused on identifying (Flow, FlowSpaces, CDG+) and supporting (m3 [em-cubed] & TextWorlds+) the functional and cognitive tasks that are requisite to the preparation and analysis of the difficult modality of text. In the light of this simplified general motivation across Chapters, this section expands this classification by reviewing details of Research Project Goals for producing specific Research Products. Generally, these Research Product Goals include 1) Extending cognitive models, 2) Building conceptual models, and 3) Building (toward) analytical tools. These three Research Product Goals are further specified below.

**Extend a cognitive model** to include:

1. The cognitive and functional tasks and cognitive-functional flow dynamics involved in the general analysis process that are missing from the Sense-Making model of Intelligence Analysis.
2. The cognitive and functional tasks and dynamics involved in the analytical reading process that are missing from the Cognitive-Functional Flow Model of General Analysis.

**Build a conceptual model** for:

2. The CDG+ Model of Discourse Processing from the original Text Worlds approach.
3. The Task Model for analytical reading from the CDG+ Model of Discourse Processing.

**Build toward analytical tools** for:

1. Supporting each analytical task across both preparation and execution phases of the the analysis process, especially challenging and currently unsupported analytical tasks.
2. Supporting the process of externalizing internal cognitive processes.
3. Supporting the process of understanding ongoing shallow evaluative judgments and annotating shallow evaluative judgments.

4. Supporting the process of understanding ongoing language comprehension processes and capturing the dynamics of deeper semantic integration of knowledge with text contents.

5. Supporting the process of understanding ongoing procedural states and annotating digital procedural states

6. Capturing, analyzing, and visualizing the process of using digital tools in multi-tool multi-tasking sessions as nested within functional space-usage sessions.

10.4.2 - Core Problems Addressed

The first core problem of the Cognitive Limits of human analysts in relation to the Procedural Complexity of the analysis process is addressed by the goal of building toward analytical tools that help with the entire preparation and execution process of (currently un-supported and) challenging tasks in the analysis process.

Likewise, in relation to the analysts’ Meta-Cognitive Awareness of their own internal knowledge states, this first core problem is addressed by the goal of building toward analytical tools that help with the process of externalizing internal cognitive processes. This will help with meta-cognitive reflection on the process of building Mental Models & Schemas during analysis.

Also, in relation to the analysts’ Meta-Cognitive Awareness of their ongoing digital workflow procedural states, this first core problem is addressed by the goal of building toward analytical tools that help with the process of understanding ongoing procedural states and annotating digital procedural states. This will help with way-finding in the ongoing analysis process.

The second core problem of Functional Limits of analytical tools, in relation to Analytical Toolmakers’ Knowledge of Real-World Analysis Process, is addressed by the goal of extending the sense-making model of intelligence analysis to include cognitive-functional flow dynamics and missing cognitive and functional tasks. This will help with identifying gaps and friction points in the coverage of tasks within
the analysis process to use as leverage points for the improved design of analytical tools.

Likewise, relative to Functional Coverage of the Analysis Model, the second core problem is addressed by the goal of building analytical tools that help with real-world cognitive and functional analysis processes. This will help by supporting analytical tasks that are not currently (well) supported in analytical tools.

Next, in relation to Analytical Toolmakers’ Knowledge of Real-World Analytical Reading, the second core problem is addressed by the goal of extending the Cognitive-Functional Flow Model of Analysis to include a Task Model for analytical reading based upon the CDG+ Model of Discourse Processing. This will help with specifying and automating the specific cognitive and functional tasks and dynamics involved in the analytical reading process.

Finally, in relation to Analytical Toolmakers’ Knowledge of Real-World Tool Usage the second core problem is addressed by the goal of building an analytical tool to capture, analyze, and visualize all tool-usage sessions as nested within functional space-usage sessions to help with characterizing the functional flow of tools between tasks both within and beyond analysis.

10.4.3 - Datasets Generated

The common thread in this Research Programme across each of these levels of Research Focus is an effort to identify and refine methods and technologies to record and model analyst sense-making activities; generally while working with digital tools in both the preparation and execution of analysis, and specifically while working with digital texts at shallow preparatory and deeper analytical levels. This common thread works at each level to addresses Research Question 2, How can transparent, low-cost observational methods be designed to produce empirical datasets that capture functional and cognitive processes?

Consequently, most research and development time has involved designing and automating methods and technologies for the transparent collection of three types of personal dataset. Each dataset joins a deep individual focus with broad, multi-year event coverage to produce and analyze deep data on detailed user activities.
These datasets are personalized collections of **digital tool workflow**, **document** and **metadata annotation**, and **personal language comprehension**.

1) Personal **workflow data** is generated through a combination of manual and computational pattern recognition and event-extraction from computer-generated log files that record user activity while using applications across specific activity classes. This system is described in Chapter 6, **FlowSpaces**.

2) A prototype system for curating personal **multi-modal analytical data collections** (e.g., personal document corpora data) integrates each multi-modal document selected for collection over time with its **objective external metadata** and shallow **personal annotation data**. This prototype is described in Chapter 7, **m³ [em-cubed]**.

3) The CDG+ Discourse Processing Model is the basis for a manual method of incrementally building TextWorlds as conceptual models of reader comprehension. This method is exemplified in Chapter 9, **TextWorlds+**, while the entire Discourse Processing model is elaborated in Chapter 8, **CDG+**. Computational implementation of this method should eventually enable users to build and interact with dynamic, personalized sense-making models.

### 10.4.4 - Gaps Illuminated

This dissertation has argued that the editors and contributors of the **Visual Analytics** research agenda, **Illuminating the Path** (Thomas and Cook 2005), would certainly agree that it is not sufficient, having designed a street light, to ask passersby how well it illuminates the ground beneath, and upon a receiving a nod, to conclude that the path has been illuminated. Instead, tool-makers must understand the paths that users travel in their entirety, and pay special attention to those **rough dark patches** for which no illumination has yet been designed. This dissertation research, particular the **FlowSpaces** project, can help tool makers see the context of tool usage, and this method may even be used to help identify these ‘rough, dark patches’ in analytical tool usage and functionality. This focus on tools and methods for identifying ‘rough, dark patches’ discussed at length in the Chapter 6, **FlowSpaces**.
Based on the cognitive-functional Flow model of general analysis (introduced in Chapter 3), FlowSpaces (Chapter 4) presents a novel methodology and system for observing wider digital workflows in functional activity space. By analyzing this data, or customizing the FlowSpaces prototype system-of-systems, or by adopting aspects of the methodology developed here, VA tool builders could extend their evaluation of their own tools, to observe how users cycle between tools and activities, to identify activities and tasks that complement and extend the utility of their tool, and eventually, in future work, to identify how users conceptualize the utility of their tool.

FlowSpaces conceives of the process of digital work as an observable and (eventually) predictable workflow pattern that extends beyond the usage of a single tool. This dissertation has argued that workflow data collection systems, like FlowSpaces, have value for digital work domains beyond Visual Analytics and for tasks beyond analysis. Thus, while I have chosen to focus on analysis using visual and interactive tools, the technological and methodological framework I have designed can also be used to study design or development practice, search habits, interpersonal communication, multi-media consumption, reading, and writing, or any other activity for which digital tools have been used.

The data collection methodology and models developed in the course of this dissertation should offer a deeper understanding of patterns in the dynamics of analytic workflow. For example, the FlowSpaces data has at least established that analysis with digital tools involves regular iterations between multiple tools and functional activities and depends a range of cognitive procedures that are not easily excavated or scaffolded.

Accordingly, this dissertation has used the author’s professional experience with the class of analytical tools produced in the field of Visual Analytics to identify analytical reading as a particularly rough and dark patch, essential unsupported in this class of tools. (See Section 10.1 for a thorough description of this problem.) This research therefore focuses on identifying and supporting the requisite functional and cognitive tasks that are necessary steps in the preparation and analysis of the difficult modality of text. This focus is discussed at length in the Chapters 7, m3 [em-cubed], and Chapter 8, CDG+.
10.5 - Contributions Made

My research presents novel methods, prototype systems, and recommendations for improving empirical understanding of real-world tool usage and facilitates the development and practice of multi-level analytical methods in the context of applied cognition. This research has first endeavoured to help tool-builders to capture, simplify and comprehend the complex dynamics of digital tool usage throughout the analytic process, and also to see where transdisciplinary methods and models may fit as solutions to recognized weaknesses in VA tool coverage.

This section clarifies how each Chapter has made scholarly contributions to the empirical understanding of analysis, the pragmatic availability of technology, and the specification of analytical methodology. It outlines how this work has contributed to scholarly understanding of real world analysis and also contributes more pragmatically to the design and prototypes of a system of systems to better support analytical reading. Finally, it contributes to practical text-analytic methodology, extending and refining methods of both deep and shallow reader-driven analysis of text.

10.5.1 - Subject Domains

At various points this dissertation has offered novel strategic argumentation, trans-disciplinary theoretical synthesis, methodological articulation, data analysis, systems design, and computational implementation as contributions to the domains of Visual Analytics, Discourse Analysis, Cognitive Linguistics, Discourse Processing, and Knowledge Representation. Selected concepts and challenges from these domains are discussed in Chapter 3, The Cognition Literature, and Chapter 4, The Analysis Literature.

My research offers both pragmatic and academic contributions to these domains. The pragmatic contributions to improved analytical practice follow the four main functionalities in the system of systems outlined in Chapter 7, m3 [em-cubed]. The academic contribution of my research should be applied to the fields of Visual Analytics, through both theoretical specification and empirical first-person observation of analytical and tool-usage process. Chapters 8 and 9 should be read as applying to Cognitive Linguistics, Discourse Analysis, and Discourse Processing,
particularly through the revision and repair of the original Cognitive Discourse Grammar approach into a model of discourse processing and conceptual representation and also by advancing the methodology of TextWorld analysis toward implementation in GATE (Cunningham and others, 2014).

10.5.2 - Cognitive-functional Flow

In Chapter 5, Flow, I have argued that Visual Analytics has failed to appropriate results and methods from Cognitive Science toward the end of better understanding the analysis process in both cognitive and functional terms. This Chapter has introduced the Cognitive-Functional Flow Model of Analysis as a trans-disciplinary synthesis of theory from Decision-Making, Intelligence Analysis, and Discourse Comprehension. I argue that the Flow model of analysis provides more leverage points for VA tool builders than the traditional Sense-Making model of analysis and conclude that it should therefore be widely disseminated in VA.

The Flow model integrates a further typification of the analysis process as deeply involving cognitive tasks even in preparation of data for analysis. Once again, this claim diverges from the Sense-Making model, and substantiates the argument presented here, that Visual Analytics must release its focus on building tools for the final execution phase of analysis, and refocus on supporting the initial preparatory phase of analysis. This includes the capture and re-use of shallow analytical judgments on media containers and contents, the capture and visualization of workflow states, like conversion pathways, and the capture and visualization of analytic workflow over the entire range of preparatory and executive analysis. Thus, the Flow model provides a focus on gaps in VA tool coverage, which has subsequently been addressed in chapter 7, m^3 [em-cubed], which the capture of judgments and path states, while FlowSpaces implements the capture, analysis, and visualization of analytical workflow data.

10.5.3 - Empirical Workflows

In Chapter 4, I introduce FlowSpaces as a system designed and implemented to capture multi-tool multi-tasking workflow data. FlowSpaces is both a practical contribution, as an automated classification system for digital tools, and a corollary always-on tool usage logging system which tangibly de-clutters and optimally
focuses the working desktop application space, and an academic contribution that records multi-tasking tool usage, deconstructs the notion of isolated or discrete preparatory analytic workflows, and supports the notion of cognitive-functional flow. Thus, the original scholarly contribution of this work employs these pragmatic contributions to record an empirical description of both the preparatory and active phases of analysis over several years of broad spectrum digital work. This is made possible by a longitudinal case study of tool usage, which can be extended to a larger study in future research. I argue that the functional characterization of the digital analysis process as observed in over four years of captured FlowSpaces data provides insight into long-term task-tool dynamics, establishes a bottleneck in analytic preparation, and highlights leverage points in both preparatory and executive analysis for VA tool builders.

10.5.4 - Shallow Analysis

In Chapter 7, I introduce m³ [em-cubed] as a system of systems designed for analytical multimedia metadata management and mobilization, that facilitates the capture and reuse of shallow analytical judgments of:

- **digital media containers** (i.e. files) by integrating external meta-databases, os-level file metadata systems, file-type internal metadata fields, and the analysts’ existing filing and naming routines in a personalized file tagging and tag-management system. This part of m³ [em-cubed] has been implemented as a proof-of-concept prototype.

- **digital media contents** (i.e. texts) by extending the native text highlighting functionality of a pdf-viewer app like Skim to include automated assignment of functional relations between contiguous highlighted sections; and by building a drag-and-drop hierarchical annotation schema management system that automatically assigns colour based on taxonomic position into a system-wide colour-picking colour palette utility. This part of m³ [em-cubed] is based on shortcomings identified through extended use of existing tools and is proposed for future implementation.

- **workflow pathway states** (i.e. statuses) by capturing the current state of various task-specific functional workflows as completed or ongoing using the established tagging infrastructure to automatically tag files as they are processed through various workflows. This part of m³ [em-cubed] has been implemented as a proof-of-concept.
10.5.5 - TextWorld Analysis and CDG+ Model of online reading comprehension

I then argue that despite a 10-year focus on building tools for the final execution phase of analysis, Visual Analytics has largely failed to assist analysts in the core analytic task of reading. Analytical reading is typified as a process of goal-focused scenario-building and argument-building reading.

Finally, in Chapter 6 on TextWorlds, I introduce Cognitive Discourse Grammar, outline reasons necessitating its revision and repair, summarize the mechanisms composing the model, articulate the methodology for executing a TextWorld analysis, which includes the exhaustive identification of GDG cues and natural language processing modules that have permitted ongoing implementation in GATE.

10.5.6 - Discussion of Primary Contributions

Since tool makers are not necessarily analysts, since analysts are not always accessible and since analysts are not always able to explain the internal cognitive and digital functional aspects of their process, the tool-building discipline of Visual Analytics, generally employs the approach of using a model, like Pirolli & Card’s Sense-Making model of analysis, as a specification for building analytical tools. In Part II, Modelling Cognitive-Functional Analytical Process, I have drawn on my experience as a working analyst using various analytical tools to work through this model. I have reflected deeply on this model of analysis and identified problems and improvements.

Many professional analyst would agree that analysis includes task that are not explicitly included in the Sense-Making model of analysis. Nevertheless, analyst must still accomplish such tasks by whatever means necessary. Thus, to identify missing analytical tasks I have returned to the literature on the analysis process and to specific areas in Cognitive Science and even to the original publication of the Sense-Making model. First, however, reflecting on my experience as an analyst, I offer a reconceptualization of the analysis process in both cognitive and functional terms, wherein each analytical task is conceptualized as a composite bundle of related and requisite cognitive and function sub-tasks. Moreover, both the overall
analysis process and individual analytical tasks have both preparatory and execution phases. Thus, in Chapter 5, *Flow*, I use these deep reflections and gap observations to produce the **Cognitive-Functional Flow Model of General Analysis** as a revision and reconceptualization of the **Sense-Making model of Intelligence Analysis**. I then go on to characterize digital tool usage broadly in functional terms in Chapter 6, *FlowSpaces*.

Considering only analytical tasks from these unacknowledged gaps, I selected one broad yet shallow unsupported task upon which to focus in Part III, **Supporting Ubiquitous Shallow Analytical Judgment**. I thus consider the simple cross-cutting task of **analytical judgment** both conceptually and functionally, in order to identify how better support for this type of classification and evaluation task could be built into analytical tools. For this broad and shallow case at the intermediate level, I built a prototype system-of-system by integrating, extending, and customizing tools for applying external tags & other internal annotations. The goal at this level is to facilitate capture of shallow analytical judgments in a manner & form that could potentially benefit the ongoing work of analysts in practical ways (Chapter 7, *m3 [em-cubed]*).

Next, also considering only unsupported analytical tasks from the unacknowledged gaps, in Part IV, **Modelling Conceptual Structures in Deep Analytical Reading**, I selected the deep, difficult task of analytical reading to consider conceptually and functionally, in order to identify how improved support for this pivotal analytical task could be built into analytical tools. For this case required extensive consideration of theories and models of this task from various disciplines beyond **Visual Analytics**. VA offers little support and no theoretical insights into this task. Accordingly, I have constructed one cognitive and functional Task Model of analytical reading, by extending a model of Online Discourse Processing that offers both an explicitly cognitive-functional approach, a focus on visualizing the models produced by readers, and a well-developed set of mechanisms for visually representing the dynamics of reader comprehension within the model. This model is the called Cognitive Discourse Grammar (CDG). CDG’s original Text Worlds approach details how readers produce **rich mental model** of text contents while reading.
Analysts and tool builders in the field of Visual Analytics will not be familiar with the CDG model of human discourse processing, though the primary method of CDG is a recognized and widely published as an analytical approach in the literary analysis fields of stylistics analysis and poetics. Even linguists and discourse analysts are not likely to be aware of its strengths, features, utility, and possible applications beyond the field of stylistics. For these reasons, this dissertation devotes a significant amount of time and space to the description, extension, and application of Cognitive Discourse Grammar. I have identified shortcomings in the original CDG approach, and provide a revision and extension to the original CDG approach which I call CDG+.

In the research projects reported here, not only do I introduce the benefits and mechanics of CDG and CDG+ to several new audiences which would seem to have a clear need for it, I also offer several contributions to the development CDG as a theoretical research paradigm. I note and repair specified shortcomings. I position CDG as a fully fledged Discourse Processing Model. I produce a range of illuminating visual conceptual models of the context surrounding the CDG+ Discourse Processing Model, its components, and emergent dynamics. I create a new interactive classification rubric for deictic & multi-dimensional contextual Meta-Classification called the CDG+ Meta-Contextual Frame to situate TextWorlds and TextWorld contents along Major Deictic Dimensions and minor contextual dimensions. I also extend the original approach into a manual method for externalizing interactive TextWorld+ Models of conceptual contents of text that are produced as mental models in the reader’s mind while reading.

For the purposes of transdisciplinary application of CDG+ in Visual Analytics, I specify a Task Model for analytical reading, based on the CDG+ Discourse Processing Model, and report on progress made toward building this Task Model into a computational model and method for pre-processing texts using natural-language processing.

The main contribution at this deep level is the introduction of CDG+ to the field of Visual Analytics, defending its selection as a model of language comprehension, identifying and improving upon its shortcomings, providing exemplars of its function, and reflecting on how it can be implemented to provide robust support for analytical reading within analytical tools.
Future work will prototype tools for automating production of TextWorld Models and facilitating interaction with conceptual models formed while reading, to facilitate capture of deep semantic and conceptual structure in a manner & form that could potentially benefit the ongoing work of analysts in practical ways (Chapter 9, TextWorlds+).

10.5.7 - Contributions by Chapter

Thus, in order to address deficits in the Analytical Toolmakers’ Knowledge of Real-World Analysis Process, I have (in Chapter 5, Flow) extended the Sense-Making model of intelligence analysis to include cognitive and functional tasks from the literature on analysis and cognition.

Next, in order to address deficits in both 1) the Meta-cognitive Awareness of Analysts’ ongoing digital procedural states, and 2) the Analytical Toolmakers’ Knowledge of Real-World Tool Usage, I have (in Chapter 6, FlowSpaces) developed a novel data-capture & processing methodology and implemented a novel system-of-systems for capturing real-time digital tool-usage and user-interface-usage data as a necessary precursor to reflecting workflow patterns back to the analyst in a visual analytical system, and back to the toolmaker as functional tool evaluation.

Then, in order to capture evaluations that may help the analysts’ Meta-cognitive Awareness of both 1) their own internal knowledge states, and 2) their ongoing digital procedural states, I have (in Chapter 7, m3 [em-cubed]) prototyped and implemented a novel system-of-systems for capturing shallow judgments of file features, and for annotating digital procedural states to help with way-finding in the ongoing analysis process, both using the simple mechanism of tags.

Also, in order to capture evaluations that may help the analysts’ Meta-cognitive Awareness of their own internal knowledge states, I have (in Chapter 7, m3 [em-cubed]) built a novel colour-based taxonomy for capturing deeper judgments of file contents, using the simple mechanism of highlights. Also, designed for future implementation, a novel dynamic colour-picker, wherein drag&drop into hierarchical coding categories auto-assigns an incrementally unique colour, based on assignment to one of 10 core categories. Hue, Saturation, and Brightness are each further tied to specific dimensions, such that a precise colour assignment is
equivalent to a specific multi-dimensional feature set. Also, designed for future implementation, a highlight-linker, to enable the specification of relations between highlighted ‘phrases’ in a text.

Then, in order to address deficits in the Analytical Toolmakers’ Knowledge of Real-World Analytical Reading, I have (in Chapter 6, Cognitive Discourse Grammar) extended the Cognitive-Functional Flow Model of Analysis to include a Task Model for analytical reading based upon the TextWorlds Model of Discourse Comprehension. This will help with specifying and in future work automating the specific cognitive and functional tasks and dynamics involved in the analytical reading process. This current specification also produces the manual Method TextWorlds Analysis, which can be used as a means of manually capturing, modelling, and visualizing the dynamic construction of personal conceptual structures (e.g., mental models & schemas) in manually created, interactive, digital visualizations called TextWorlds.

Finally, in order to address the Meta-cognitive Awareness of their own internal knowledge states, I have (in Chapter 9, TextWorlds+) made important progress towards the future implementation of a novel method and system of capturing real-time online reading comprehension data, as a means of capturing, modelling and visualizing the dynamic construction of personal conceptual structures (e.g., mental models & schemas) in an interactive visual analytical system. Specifically, I have designed the Situation Model, the manual interactive digital TextWorld visualizations (e.g., interactive diagrams), and the Meta-Contextual Frame for deictic and contextual meta-analysis, and exemplified their usage.

So, in these many ways, this work makes scholarly contributions to the empirical understanding of analysis, to the pragmatic availability of technology, to the specification of analytical methodology, and to the extension of analytical theory. This work contributes to scholarly understanding of real world analysis. It contributes the design and prototypes of a system of systems for analytical preparation, and points towards the future implementation of interactive analytical reading system that can be interactively personalized to each analysts’ requirements. Finally, it contributes to methodology, extending and refining methods of both deep and shallow reader-driven analysis of text. I believe it will
remain relevant for as long as close analytical reading of texts remains a core process for digital readers and analysts.

This section has outlined my objectives for academic contributions to each domain discussed in the following chapter. Chapter 2, Literature establishes how the disciplinary foundations of my research draw upon particular subfields in the general study of Cognition and Analysis; particularly from the domains of Discourse Comprehension, Reasoning, Decision-Making, and Knowledge Representation, Visual Analytics, Intelligence Analysis, Discourse Analysis, Semantics, Cognitive Linguistics. This section has introduced the most significant contributions made to these domains by my dissertation research.

Specifically, practitioners in domains of analysis discussed below will benefit from externalizing and revising mental models generated while processing and analyzing text. The practitioners of text world theory approach to Discourse Analysis and the broader field of Cognitive Linguistics more generally will appreciate both pragmatic tools for engaging in these methods of analysis.

10.5.8 - Recommendations

Visual Analytics must release its focus on building tools for the final execution phase of analysis, and refocus on supporting the initial preparatory phase of analysis.

Analytical tool-builders can improve support of under-supported analytical tasks in the following functional areas:

1) Outside the document, on each file, by capturing:
   • Shallow analytical judgments of file features, captured as tags.
   • Shallow analytical judgments of file provenance, captured as tags.
   • Shallow analytical judgments of workflow progression states, captured as tags.

2) Inside the document, within textual PDF files, by capturing:
   • Shallow evaluative assessments of propositional boundaries, captured as highlighted chunks while ‘skim’-reading.
• Shallow evaluative assessments of **propositional functional roles**, distinguished via *color assignment*, by selecting a *concept-named-color* from a custom-built, hand-coded *colour palette* for each *highlighted chunk*.

In future, work towards building:

1) An **intelligent chunker** to automate identification of **propositional boundaries**, captured as *highlighted chunks*.

2) An intelligent, interactive **relation-picker** to automate identification of ranked options for **propositional functional roles** of highlighted chunks, captured as *highlighted chunk colours*.

3) An intelligent, interactive **colour-picker** to automate identification of ranked options for *color assignment* for highlighted chunks.

4) An intelligent, interactive **named-concept schema** to automate incrementally unique color assignment for *new named-concepts* added as children of parent classes, based upon their assigned location in a multi-dimensional class heirarchy.

5) A **color component mapping** between dimensions in HSB+Alpha color space to meta-feature class assignments in a multi-dimensional deictic contextual meta-classification space.

6) An intelligent **chunk linker** to automate identification of ranked options for **propositional functional roles** between highlights chunks & other PDF annotations, captured as *directional, coloured lines*, to enable deeper model-building concept map formation.

**10.6 - Summary of Discussion**

This Dissertation Research Programme has highlighted the challenges facing modern analysts and pointed to established problems in the field of **Visual Analytics**. These challenges mostly regard difficult tasks in the analysis process (e.g., externalizing the results of classification and reading), gaps and omissions in published understanding of the analysis process, and shortcomings in the match
between tasks in the analysis process and tasks supported by analytical tools. Thus, two core problems define this research: the Cognitive Limits of human analysts and the Functional Limits of analytical tools.

Cognitive Limits of human analysts are first described in relation to the Procedural Complexity of the analysis process, and to the Meta-cognitive Awareness of internal knowledge states and digital procedural states. Next, the following sub-section describes the Functional Limits of analytical tools, offering an extensive consideration of specific shortcomings classified generally in relation to

1) Toolmakers’ Expertise & Knowledge, 2) Data Features, and 3) Functional Coverage of Tasks in the Analysis Model. Specifically, for current analytical tools, limitations in these general classes relate to 1) the Analytical Toolmakers’ Expertise in Eliciting Tool Requirements and in their capacity as regards Implementation of Relevant Findings from Cognitive Science, and the Analytical Toolmakers’ Knowledge of Real-World Analysis Process, of their knowledge of Real-World Tool Usage and of; 2) technical challenges dealing with Data Scale, Modality, Complexity, Type, and 3) Functional Coverage of the general Analysis Model, of specific Task Models, and of tasks to facilitate Cognitive Externalization.

Then in the Section 10.2, Motivations Revisited, I have specified the motivations for this research in relation to both Core Problems, specific challenges, General Research Questions, and project-specific focus questions.

In general terms, the two goals of this dissertation are to extend and build cognitive models and to build (toward) analytical tools. Each model and tool has a specific purpose, that addresses a Core Problem, a specific challenge, a General Research Question, and to project-specific focus questions. Some also have minor project-specific focus questions.

Taking a ‘human scale’ view of the problem space allows this dissertation research to proceed from the widest, general level of functional workflow, to an intermediate, preparatory level of analytical reading, which includes shallow judgments, to the deepest level of language comprehension during analytical reading. Across each of these levels, this dissertation has worked towards collecting datasets to record and model analyst sense-making activities, generally while working with digital tools in both the preparation and execution of analysis, and
specifically while working with digital texts at shallow preparatory and deeper analytical levels.

Consequently, most research and development time has involved designing and automating methods and technologies for the transparent collection of three types of personal dataset. Each dataset joins a deep individual focus with broad, multi-year event coverage to produce and analyze deep data on detailed user activities. These datasets are personalized collections of digital tool workflow, document and metadata annotation, and personal language comprehension.
For a wider discussion of the role of existential modality in analysis, refer to Section 6.2.3

This environment aims to provide the analyst either with tools that do not currently exist (for analytical methods that are currently executed manually) or improved integration of existing tools (where tools do exist that offer elements of the ideal functionality, but they do not currently interoperate easily or seamlessly).

$^4$ $\text{em}^3 \text{ [em-cubed]}$ is a component based prototype system-of-systems. Its prototype leveler is a method for retrieving & managing the first level of whole, master-document level annotation is functioning and remains an active work in progress. Methods of document-, language-, rhetorical-, and conceptual-structure identification are documented in the literature, and will inform implementation of subsequent phases.

Not only does analysis strike a balance between theory and practice, but also half the domains selected here are theoretical, while the remaining domains are applied.

In the famous thought experiment, the idea of quantum entanglement between observation and measurement of state values is illustrated by a scientist, a cat, a box, a bottle, and a single atom of some radioactive isotope. After the scientist seals the isotope in the bottle and then seals the cat and the bottle in the box, she will never be able to ascertain the state of the cat. The cat could be alive, or it could have knocked over the bottle, been exposed to the isotope and be dead. Opening the box could release isotope, so while the box remains closed, the cat is simultaneously both alive and dead.

Of course, nobody ever wants to be the cat in this scenario, so writers generally follow the humane practice of substituting knowledge for radium and permitting the vagaries of comprehension take over for the ravages of radioactivity. Don’t be alarmed. Readers generally make it out alive from each textual structure they decide to enter, with only the state of their knowledge in question. You will too. Keep calm, and read on.

Writers must therefore do a lot of extra work to encode basic knowledge to different levels of accessibility within the text without knowing if definitions are required by the reader, or if exposition will be perceived as unnecessary review. To certain experts, the resulting structure of transdisciplinary research texts may occasionally become tiresome. To such expert readers with broad and deep expertise in several of these areas, I offer the following strategy for avoiding boredom or annoyance and for facilitating easy navigation to the core contributions and arguments of this research.

First, consider the text of this dissertation as a classroom with no prerequisites. In this context the writer is an educator whose first goal is to bring each reader to the point of sufficient expertise to engage productively with the arguments. This is very different from the standard academic dissertation. Selectively treat the text as a skills-based tutorial, skimming didactic sections and rejoining the class for the expository conceptual structure-building, or argumentation, as required. But please make judicious use of this open-door policy to skip ahead and not to skip out. Please do rejoin the class for the argumentation and the presentation of methods. I believe it will all be worth it.

Certain cues may help the expert skip ahead: “In the domain of …,” “In the context of….” In the Literature section the general domains wherein definitions apply is embedded in the text as labeled sections. Those section define terms and establish conceptual relations and structures. Outside of the literature section, if the chapter pertains to cognitive models, the domain of cognition can be presumed as the background for any newly introduced terms, unless otherwise indicated, and likewise for chapters on analysis. Specific research areas, sub-disciplines, or approaches within the domain of cognition will be specified locally in the text, as in the examples above. When they are,
match the domain to your expertise, and skim on ahead when you know that area well. This advice applies principally to Chapters 3 and 4.

8 Whereas sustained verbalization of process is certain to be a drag on productivity, further attempting to verbalize subconscious process can be expected to halt productivity indefinitely.

9 The term ‘data modality’ is used to express the enmeshed nature of basic data types (e.g., textual, numerical, audio, video), which are encased within basic container types (e.g., tangible physical media, digital file formats).

10 Many combinatorial possibilities exist for multi-modal data. However, the most relevant include cases of 1) single formatted entities (e.g., digital files) containing multiple data types (e.g., a JPG file which contains both structured metadata fields, their values, and the image data); 2) a single file transparently linked to another single file containing multiple data types pertaining to a single, individual data entity (e.g., files with external sidecar metadata); 3) single files containing multiple, individual data entities (e.g., database records that have multiple data values and types across many columns); 4) single files containing multiple records that may contain other files, (e.g., database records with embedded files); 5) singular data entities having multiple media and file formats (e.g., the first print edition, the digital ePub, and the audiobook) and versions. This research considers the role of both internal (e.g., metafied) and external (e.g., near-formed) multi-modality for analytical data.

11 The Vector Space Model (VSM) is an influential and powerful framework for storing, analyzing, and structuring documents that was developed by Salton in 1974 (Salton and others 1975).

In this model, each text document d is represented by a vector of frequencies of the remaining m terms: \( d = (t_1, ..., t_m) \) (Beil and others 2002). This is accomplished in three stages; 1) document indexing, 2) term weighting, and 3) computation of similarity coefficients.

Document indexing

A vector in a high dimensional space is made to represents each document or query (Börner and others 2003). The number of unique terms in a document corpus determines dimensionality of the document vector space (Börner and others 2003). Preprocessing often uses a stop list of common words to remove high frequency and non-significant words from the document vector (Börner and others 2003).

Term weighting

Term weighting indicate term importance for document representation (Börner and others 2003). Most weighting schemes like the inverse document frequency (see Section 3 Measures) assume that the importance of a term is proportional to the number of documents the term appears in (Börner and others 2003). Since long documents usually have a much larger term set than short documents, document length normalization is employed to reduce the likelihood that long documents will be retrieved more often than short documents (Beil and others 2002).

Computation of similarity coefficients

The similarity between any two documents (or between a query and a document) can then be determined by the distance between vectors in a high-dimensional space (Börner and others 2003). Word overlap indicates similarity [4](Börner and others 2003). Refer to Section 3, Measures for an overview of measures for document similarity.

This model presents an easy way to assess document similarities based on word matches (Börner and others 2003). However, it cannot detect variations in word meaning (Börner and others 2003). This is known as the "vocabulary mismatch problem" (Börner and others 2003). Other methods (e.g., LSA) that examine the context of words have been proposed to resolve this problem. Also, over time, the vector space model has developed many variations, which address specific shortcomings raised by continuous research.

12 Traditional definitions of ambiguity, polysemy, and vagueness form a sort of range. Ambiguity is defined as between 2 lexemes (or possible terms), polysemy as a single lexeme with multiple
distinct senses, and vagueness as a lexeme with a single non-specific meaning. In this approach, polysemy is a median point, part way between ambiguity and vagueness. (Tuggy 2006)

13 Norman’s definition of the term affordance is intended here. Software tools provide virtual action possibilities for a range of digital functions through user interaction.

14 This section introduces the Berkeley FrameNet project as an exemplary application of the frames theory in the domain of linguistics, and notes that other modern AI systems like CYC reflect a post-Minsky frame-like structure at certain levels.

15 Not the sentence.

16 Cognitive Discourse Grammar model of discourse comprehension theorizes rapid situational sense-making based on schemas containing both procedural and propositional knowledge.

17 Traditional definitions of ambiguity, polysemy, and vagueness form a sort of range. Ambiguity is defined as between 2 lexemes (or possible terms), polysemy as a single lexeme with multiple distinct senses, and vagueness as a lexeme with a single non-specific meaning. In this approach, polysemy is a median point, part way between ambiguity and vagueness. (Tuggy 2006)

18 Cognitive Discourse Grammar model of discourse comprehension theorizes rapid situational sense-making based on schemas containing both procedural and propositional knowledge.

19 Pirolli restates the Learning Loop Complex (Russell and others 1993) that described inner representational sense-making process used to digest a large amount of information for curriculum design. First there is a search for a good representation. Encoded information will not all fit into that representation. Adjustments to the representation aim at better coverage. Iterating this process ultimately yields a highly compact representation of the essence of the information relative to the intended task. (Pirolli and Card 2005)

20 Omitted cognitive tasks include: goal setting & target specification; the selection, identification, evaluation, classification, labelling, collection and annotation of features and items; judgments of comparison, classification, relevance, coherence, inference, entailment, blending; moving, co-ordinating, organizing; re-representing; modelling; hypothesizing; testing; disconfirming; writing; framing; editing; and presenting.

21 While at first glance this model seems overly simplistic, this interactive diagram reveals only the top level steps in analysis. Some of these steps encompass further cognitive operations or tasks that typically overlap or flow together.

22 Very often, the various task steps or phases of a job are not easily separable since they are not entirely conceptually discrete - or may be executed within (or by) a single program that offers a broad range of functionality. For example, MS Word provides page layout, text formatting, and file conversion options. Despite the great number of functions possible within ‘tool-box’ applications like MS Word, the current implementation of the FlowSpaces classification schema employs human judgment to determine prototypical task for each application. In this case, the application is determined to be prototypically for creating and editing written documents - so, for writing.

In the manual classification schema used for this implementation of FlowSpaces, ‘task types’ are nominalized to name a class of tools for the prototypical tasks they enable. The classification question is ‘What does this tool let users do?’ or, if they automate tasks for the user ‘What does this tool do?’ For example, since MS Word prototypically enables creation editing ‘Word document’ files, so it belongs to the tool class of editors. Alternatively, tools that primarily allow users to convert files from one format to another are converters. Though Word provides an admirable range of conversions, it is at its core about writing and editing. Since ‘writing’ is specific to textual documents; we look for an equivalent term that may describe a broader range of media, we opt to use the term ‘editor.’

At the most basic level, FlowSpaces rests on a pragmatic, human-initiated, usage-based typification of the primary task possible with any given software Application. Of course, software classification based on tool functionality is not in itself novel — software review websites and shareware/trial-ware download sites often do this at a level of granularity suited to the purpose of
expedient organization of thousands of application for retrieval over the internet. However, classes on most software review or vendor websites are not explicitly theoretically informed, and do not offer a gradient of classification, or a rated range of membership across all other categories.

FlowSpaces is designed to allow its users to deduce that gradient of classification and rated range of membership across all other categories.

Following prototype classification theory (Rosch and others 1972), the task type taxonomy I developed here considers major classes of task-types to be separable, if not entirely discrete. This is because in computer-enabled work, one task often ‘bleeds’ or ‘flows’ into the next. For example, when converting a file from one format to another, I may simply navigate to a file menu, select ‘Save as…’, and another file version is generated without ever leaving my initial task-space or application. The purpose of the taxonomy is to attempt to better demarcate the boundaries between tasks.

In April 2010 I began capturing application usage data using the open-source package Slife.
In January 2011 I began capturing application and user interface usage data using the paid application BackTrack.
In May 2012 I began capturing task-state desktop shift data with Growl.
In May 2013 I began capturing keyboard shortcut data with KeyBoard Maestro and Growl.

Other scenarios, like single-focus single-app multi-tasking (ssm), or switching between tasks within a single Tool in a given Activity Space, cannot be explored in the current dataset, until a mechanism is invented for logging the specific task in use within each Tool. And multi-focus multi-app single-tasking (mms) amounts to using various Tools from a single Activity class across multiple Activity Spaces. It is not easy to imagine a use-case scenario for this usage pattern.

The linear model of app duration is significant.

Model formula: 
application*( duration + intercept )

Number of modeled observations: 11904
Number of filtered observations: 0
Model degrees of freedom: 90
Residual degrees of freedom (DF): 11814
SSE (sum squared error): 283362
MSE (mean squared error): 23.9853
R-Squared: 0.971512
Standard error: 4.89748
p-value (significance): < 0.0001

Analysis of Variance:

<table>
<thead>
<tr>
<th>Field</th>
<th>DF</th>
<th>SSE</th>
<th>MSE</th>
<th>F</th>
<th>p-value</th>
</tr>
</thead>
<tbody>
<tr>
<td>application</td>
<td>88</td>
<td>46229.913</td>
<td>525.34</td>
<td>21.9026</td>
<td>&lt; 0.0001</td>
</tr>
</tbody>
</table>

For example, from the perspective of the tool itself, Word(r) is an editor, since its core task is to edit documents. Yet from the perspective of the user, at its core, Word(r) enables the user to write documents; so Word(r) is a document writer. Similarly, from the tool perspective, Google
Chrome is a browser, since its basic function is to retrieve documents and display documents and enable the user to read documents, particularly including various formats of network-accessible informational media. So from the tool perspective, the primary function of Google Chrome is therefore is usually poorly specified as browse[web documents]. However, from the user’s perspective, Google Chrome is a seeker, since it enables the user to seek information. Thus, Google Chrome is a network-accessible informational media seeker.

Thus, the manager app processes the document, attempting to identify DOIs, Titles and Authors, and conducts background searches, and automatically retrieves the best match to enter the record’s details locally. The manager app provides easy access to both alternate matches for manual selection, and to the best available public and institutionally available versions of full-text resources for download.

This hypothesis will be tested in future work.

For example, InSpire gives readers ‘groups’ - a hierarchical series of bins - to work with pre-classified data. Through interaction with its ‘Galaxy’ view, document subsets can be identified and added as ‘classes’. Its ‘hypothesis supporting’ content-ranking widgets and User Interface elements to capture insight generated while reading, but are . JigSaw gives readers several types of ‘entities’ extracted and labeled in a text. But the only relations it provides are ‘coincidence’ or ‘co-location.’

So this part of the project has required programmatically gaining access to all available levels of file metadata, and to the content itself; all of which is analyzed & evaluated to extract further metadata. Access to embedded file metadata must be read/write so that existing metadata may be read, evaluated, and possibly improved and augmented or overwritten. To this end, access to file-system-level metadata is also key, as is access of XATTR metadata.

Werth’s failure to employ his chosen name for his approach consistently throughout the Text Worlds monograph might be attributable to the challenging circumstances surrounding the posthumous publication of the monograph four years after Werth succumbed to cancer in 1995.

I have noticed because of the depth, intensity, and duration of my consideration of this framework, its utility, and its computational possibilities.

The formal articulation of Werth’s coherence constraint is defined formally as (Werth 1999c):

$$\forall p, \exists c \in D <c, p> \& F <p, c>$$

Where ‘p’ = a proposition, ‘c’ = the Common Ground, ‘D’ = a single discourse, ‘F’ = one of the functional conditions; full or partial synonymy, antonymy, hyponymy, metonymy or metaphor.

As regards the character of connections between propositions, TextWorlds concludes that what binds elements of a text together as coherent is better expressed in terms of functional relationships rather than logical ones. Thus, the functional conditions standard TextWorlds employs to determine coherence of new propositions with those in the Common Ground are; synonymy (full and partial), antonymy, hypernymy, hyponymy, metonymy and metaphor (Werth 1999c).

TextWorlds defines these functional relations in terms of comparisons between frames. Synonymy serves to overlap frames. Antonymy negates features in counterpart frames. Hyponymy requires a given frame to be part of a hierarchical structure in the other frame. Metonymy[1] is a relation of belonging that can range from part-whole, regular ownership, possession of a property or characteristic, to looser associations of contiguity (like bread and butter or pen and paper). In frame terms, metonymy represents an association between frames which may be well attested and commonplace, or special and happenstance, and therefore contextually bound to limited circumstances.

Metaphor is the mapping of one frame (typically an abstraction) to another more concrete frame (Werth 1999c).

You may think of them just as embedded labels that indicate the function of a given stretch of text. For example, textual signals can indicate such things as conceptual progress (e.g. discourse deixis), demonstrative entity specification (e.g. entity & person deixis), relationships between
propositions (e.g. rhetorical or argument structure), relationships between speaker and their spoken content (e.g. perspectivization and evaluative judgments).

Revised CDG+ adds functional relations which connect expressions to produce a functional value equivalent to an unstated expression.

These are based on word sense, word reference to the world built up by the text, and ultimately on the the real world within which the text exists as a linguistic artefact.

The original CDG approach also defines proposition more formally as a quantified expression containing an N-place predicate together with N arguments (Werth 1999a). Arguments may be other propositions. So at both the propositional level and the world level, situations may be embedded in other situations (as complex matrix propositions, and sub-worlds).

In the standard Text Worlds Model of CDG, the meaning of any CDG proposition is the conditions which produce its probabilistically assessed truth-value. All CDG propositions must posses this probabilistic version of a truth value, which is assessed by the reader with respect to the world level in which the proposition resides (Werth 1999a). For a discussion of the Truth Value Assessment mechanism of the CDG Discourse Processing Model, Refer to Appendix Y. CDG, Section 5.3, Interactive Evaluation.

This perspective includes assumptions regarding embodied nature of human cognition and the experiential nature of human language.

In a monograph of over 150,000 words, he employed this chosen name only ten times, plus seven more instances of the abbreviated form, ‘CDG’.

Thus, beyond Cognitive Linguistics, if a sentence-based linguist explores phenomena beyond the sentence, they “usually stop at the text” (Werth 1999f) p. 21) since the context surrounding the text is beyond the theoretical scope of standard generative approaches to linguistics.

The provided taxonomy does not distinguish between structural layers and functional relations in the “Modality Function of Language” that establishes setting. So the Modality Function situates information within relations of Viewpoint, Probability, and Interaction. Yet “modality is in the category of Interaction” and “specifically, it has to do with the assessment of such factors as truth, probability, and reliability (Werth 1999c, p.176).”

So either modality and Modality Function do not refer to the same things, and Modality-Function\Probability should be reclassified as Modality-Function\Interaction\modality\Probability\assessment, or they do refer to the same thing and Werth is just wrong.

For discussion, See note on Polysemy in Introduction.

In my view, given how many different definitions and contexts of use the term modality is used in (I have counted 8) the different uses of the term must be reduced and renamed to ensure Gricean maxims are honored.

(e.g., Modality as a contextualizing function of language that situates information with respect to the current context; modality as the relationship between the speakers and the text; modality as a class of world-building predicate; modal properties of necessity and possibility in logic; root modals expressing personal certainty while conveying obligation, permission, ability and volition; epistemic modality expressing situational certainty while conveying a degree of probability; modal contexts of probability and possibility; modal worlds that express possibility.)

In practice, the unstated original CDG approach to drawing a Text World Diagram is something like:

1) Review these exemplars of Text World Diagrams.
2) Notice these two areas are intended for different things and the general format used.
3) Now, recalling how everything works in the CDG approach;
4) Make something that looks like this.
Appendix Y presents the arguments, mechanisms, and dynamics of CDG sequentially, comprehensively, and clearly. By following this presentation, the reader will see how the CDG apparatus is a discourse processing model, and moreover will see how each sub-task in the analytical reading is identified. Readers will also realize that since this highly complex series of steps occurs nearly instantaneously online (e.g., within the mind at the moment of the action), support for this task must be able to do what we do so easily, and then help us with the more difficult longer-term challenges of integrating our thoughts with our existing models. This supports the argument I make for why support for analytical reading requires extensive computational preprocessing.

This choice of focus is motivated by the relationship between a text, discourse and their respective constituents. Since texts and sentences are an abstraction made for the purpose of analysis, whereas discourses and utterances are real language acts, Werth prefers to derive theoretical units from phenomenon which actually occur.

The idea of a text world is similar to concepts of ‘mental spaces’, ‘frames’, and ‘idealized cognitive models’ of Fauconnier, Fillmore and Lakoff. Text worlds specify such things a place and time details, the persons and objects present in the world, their properties and interrelationships. These elements activate frames - areas of memory that are related to areas of experience and knowledge encoded as complex conceptual structures - that interact dynamically with the text world and the propositions in the CG.

For example, the depiction of the standard approach to TextWorlds element placement (Figure 6.3.2.1 above) is highly redundant (e.g. Accessible n=12 PRESENT n=7, REMOTE n=7, TIMEZONE n=7, TextWorld n=7 Character n=6, Participant n=6, PAST n=6, FUTURE n=6, SYSTEM n=2).

Clearly, the dynamics of all matter that occupies space cannot really be collapsed to single linear dimension. Nor can the multi-dimensional dynamics of living beings. Nevertheless, the concept of both the physical and corporeal dimensions offer contingent segmentations of human experience, which are widely used, and useful. In fact, they have been used for centuries to define both practical pursuits and academic disciplines (e.g., plant, animal, mineral).

For each WordNet definition of a term, a rater classifies every term in the original term’s definition by judging its prototypical dimension. In this way, initial prototype judgment of the original term can be evaluated iteratively (across raters) and internally (within raters) to determine agreement between initial prototype judgment and the additive weighted ratio of all terms in a definition. Each new term encountered in a definition is added to the dimensional classification cue to be fully analyzed (e.g., by iterating through this prototypical classification process for each term in each definition of each sense of the original term).

Thus, the first time a term is encountered, a snap ‘prototype’ judgment is made, which can eventually be compared to the cumulative weighted composite class for each sense.

While Catherine Emmott has used the term ‘contextual frame theory’ since 1997, (Emmott 1997), and her work prior to that is cited by Werth (Werth 1999n). In order to acknowledge her prior work, I will henceforth refer to mine as a meta-contextual frame. Any future references to a contextual frame will take her conceptually prior meaning. My meta-contextual frame is an interactive tool for simultaneous meta-classification along multiple core deictic and contextual dimensions, so it should have enough specificity to avoid confusion with Emmott’s contextual frame theory.

In such cases, if the x,y co-ordinates of the represented spatial elevations were to remain fixed, the temporal dimension would no longer follows a linear path, and would instead follow the path of action.

Making the x,y co-ordinates of space relative to unfolding action turns the contextual frame into a temporal compass of sorts; since the ongoing narrative action will always unfold pointing toward the future. The representation of the environment will swing about in response to position in the environment, much as in a first-person game.
Since each experiential dimension is a composite of its lower dimensions, the user may reduce dimensional complexity by treating only the dimension of space, without consideration of its linear, planar, and height components individually. While each dimension could similarly be compacted, in practice the utility of this system is in fact in extricating and ranking the contributions of lower dimensions to each higher level concept (e.g., identifying the temporal, spatial, dimensions of an artefact of social power). Thus, the number of basic dimensions of human experience selected by the user should range from 10 to 12.

This background cultural knowledge should be specifiable as a collection of Frames produced in the course of the readings (and other linguistic experience) completed to a specified point in time or level (e.g., to the 3rd grade level; to the post-graduate level, etc.)

TextWorlds are distinct from manually created Text Worlds which are static, printed, schematic representations, in that TextWorlds are dynamic, digital interactive representations; hybrid forms, produced from digital templates within OmniGraffle, and interactive when viewed within that software.

Discourses about the discourse world discuss elements of the language situation that are obvious or accessible to participants. In such cases, the text world is a conceptualization of the part of the discourse world that is in focus for the purpose of the discourse. In the more common case, the text world is the conceptualization of a state of affairs in the memory or imagination of the speaker and/or hearer.

Norman’s definition of the term affordance is intended here. Software tools provide virtual action possibilities for a range of digital functions through user interaction.

Appendix Y presents the arguments, mechanisms, and dynamics of CDG sequentially, comprehensively, and clearly. By following this presentation, the reader will see how the CDG apparatus is a discourse processing model, and moreover will see how each sub-task in the analytical reading is identified. Readers will also realize that since this highly complex series of steps occurs nearly instantaneously online (e.g., within the mind at the moment of the action), support for this task must be able to do what we do so easily, and then help us with the more difficult longer-term challenges of integrating our thoughts with our existing models. This supports the argument I make for why support for analytical reading requires extensive computational preprocessing.

This highlight linker has not yet implemented, but general forms exist in the qualitative data analysis literature, tools like ATLAS.TI or NVIVO. My strategy was to building this entity linking view on top of the data created by PDF annotations, and create a simple PDF viewer that connects these surface semantic structures.
Bibliography


Andrienko, Gennady L, Natalia V Andrienko, S Rinzivillo, M Nanni, D Pedreschi, and others. 2009. ‘Interactive Visual Clustering of Large Collections of Trajectories’, in, pp. 3–10 <http://dx.doi.org/10.1109/VAST.2009.5332584> 


Bae, Soonil, Catherine C Marshall, Konstantinos Meintanis, Anna Zacchi, Haowei Hsieh, and others. 2006. ‘Patterns of Reading and Organizing Information in Document Triage’, in, xliii, pp. 1–27 <http://dx.doi.org/10.1002/meet.14504301160>


Bodnar, John W. 2003. Warning Analysis for the Information Age: Rethinking the Intelligence Process (Joint Military Intelligence College Washington DC Center For Strategic Intelligence Research)

Science, 247.4938: 44–48 <http://dx.doi.org/10.1126/science.2403694>


Brenner, C B. 2010. ‘Event Segmentation and Memory Retrieval in Reading Comprehension’


Chin, G, Jr, and OA Kuchar. 2009. ‘Exploring the Analytical Processes of Intelligence Analysts’, in


Cornish, F. 2009. ‘Indexicality by Degrees: Deixis, “Anadeixis” and (Discourse)
Anaphora’, A Symposium


type=2>

Cranor, L F. 2008. ‘A Framework for Reasoning About the Human in the Loop.’, Upsec


Cruz, Daniela, and Pedro Henrikues. 2010. ‘Exploring, Visualizing and Slicing the Soul of XML Documents’, in


Curtis, Laura. 2008. Glass Box (Rome, New York: Battelle Memorial Institute)


474


<http://dx.doi.org/10.1093/acprof:oso/9780199217274.001.0001>


Dumais, S T, G W Furnas, Thomas K Landauer, S Deerwester, and R Harshman.

Duncker, K. 1945. On Problem-Solving (Greenwood Press), lviii


Edwards, Ward, Ralph F Miles, and Detlof Von Winterfeldt. 2007. Advances in Decision Analysis (Cambridge Univ Pr)

Eleni, Miltsakaki, and Joshi Aravind. 2004. ‘Annotating Discourse Connectives and Their Arguments’, in


Emmott, Catherine, Anthony J. Sanford, and Eugene J. Dawydiak. 2007. ‘Stylistics Meets Cognitive Science: Studying Style in Fiction and Readers’ Attention From an Interdisciplinary Perspective.’, Style, 41.2 (Northern Illinois University, English Department): 204–24 <http://dx.doi.org/Article>


Evans, Vyvyan, and P Chilton. 2007. Language, Cognition and Space


Fauconnier, G. 1998. ‘Principles of Conceptual Integration’


<http://dx.doi.org/10.1207/s15516709cog2202_1>

Feldman, Jerome. 1975. ‘Bad-Mouthing Frames’, in


<http://dx.doi.org/10.1016/s0093-934x(03)00355-9>

<http://dx.doi.org/10.1007/978-0-85729-198-1_5>

<http://dx.doi.org/10.1016/S0304-3940(02)01196-5>


Fillmore, Charles J, and H Sato. 2002. ‘Transparency and Building Lexical
Dependency Graphs’, in, pp. 87–99


Fincher-Kiefer, R. 2001. ‘Perceptual Components of Situation Models.’, Memory and Cognition


Fischoff, Baruch, Committee on Behavioral and Social Science Research to Improve Intelligence Analysis for National Security, Cherie Chauvin, and National Research Council. 2011. Intelligence Analysis (Natl Academy Pr)


Fleischman, Suzanne. 1990. Tense and Narrativity (University of Texas Press)


Design Hillsdale)


Franklin, R D, D B Allison, and B S Gorman. 2014. Design and Analysis of Single-Case Research (Taylor & Francis)

Frege, Gottlob. 1980. ‘On Sense and Reference’, in, Translations From the Philosophical Writings of Gottlob Frege, 1st edn (Rowman & Littlefield)


Føllesdal, Dagfinn. 1961. ‘Referential Opacity and Modal Logic’


Gao, Yingfan, Runbo Ma, and Yushu Liu. 2006. ‘Fast Text Categorization Based on a Novel Class Space Model’, in (presented at the MICAI'06: Proceedings of the 5th Mexican international conference on Artificial Intelligence, Springer-Verlag)


Ghosh, Kuntal, and Anirban Roy. 2011. ‘Neuro-Visually Inspired Figure-Ground Segregation’, in (presented at the 2011 IEEE International Conference on Image Information Processing (ICIIP), IEEE), pp. 1–6 <http://dx.doi.org/10.1109/ICIIP.2011.6108932>


Giovanelli, Marcello. 2013b. ““La Belle Dame Sans Merci”’, in, Text World Theory and Keats’ Poetry : the Cognitive Poetics of Desire, Dreams and Nightmares (Bloomsbury Academic) <http://dx.doi.org/10.5040/9781472542083.ch-008>


Giovanelli, Marcello, Marcel. 2013h. ‘The Eve of St Agnes’, in, Text World Theory and Keats’ Poetry : the Cognitive Poetics of Desire, Dreams and Nightmares (Bloomsbury Academic) <http://dx.doi.org/10.5040/9781472542083.ch-005>


483


Gobet, Fernand, and Gary Clarkson. 2004. ‘Chunks in Expert Memory: Evidence for the Magical Number Four … or Is It Two?’, Memory, 12.6 (Taylor & Francis Group): 732–47 <http://dx.doi.org/10.1080/09658210344000530>

Gobet, Fernand, P Lane, S CROKER, P CHENG, G JONES, and others. 2001.
‘Chunking Mechanisms in Human Learning’, Trends in Cognitive Sciences, 5.6: 236–43 <http://dx.doi.org/10.1016/S1364-6613(00)01662-4>


Greitzer, Frank L, Paula J Cowley, and Ernest Hampson. 2006. ‘Glass Box Intelligence Analysis Research and Development’, Google.Ca


Ha, Daniel, Minjung Kim, Andrew Wade, William O Chao, Kevin Ho, and others. 2007. ‘From Tasks to Tools: a Field Study in Collaborative Visual Analytics’, in, pp. 223–24 <http://dx.doi.org/10.1109/VAST.2007.4389028>


Hintzman, Douglas. 1986. ‘Schema Abstraction in a Multiple-Trace Memory Model’, Psychological Review, 93.4: 311–428


<http://dx.doi.org/10.1109/MVA.2015.7153203>


Jakobs, Eva-Maria. 2003. ‘Reproductive Writing—Writing From Sources’, Journal of
James, W. 1918. The Principles of Psychology (Holt)


Kadivar, N, V Chen, D Dunsmuir, E Lee, C Qian, and others. 2009. ‘Capturing and Supporting the Analysis Process’, in, pp. 131–38 <http://dx.doi.org/10.1109/VAST.2009.5333020>


Karp, Peter D. 1992. The Design Space of Frame Knowledge Representation Systems, SRI Artificial Intelligence Center, p. 49


Katifori, Akrivi, Constantin Halatsis, George Lepouras, Costas Vassilakis, and Eugenia Giannopoulou. 2007. ‘Ontology Visualization Methods—a Survey’, Computing Surveys (CSUR), 39.4


King, Jeffrey C, Scott Soames, and Jeff Speaks. 2014. New Thinking About Propositions (OUP Oxford)

Kintsch, Walter. 1974. The Representation of Meaning in Memory (Hillsdale)


Role of Syntactic Cues and Causal Inferences’, in, From Learning Process to Cognitive Processes


Kuipers, B. 1975. A Frame for Frames: Representing Knowledge for Recognition, Dspace.Mit.Edu


Lakoff, George. 1968. ‘Counterparts, of the Problem of Reference in Transformational Grammar.’: 8


Leibniz, Gottfried Willhelm. 1747. ‘Essais De Théodicée Sur La Bonté De Dieu, La Liberté De L'homme Et L'origine Du Mal’


Manning, Christopher D, M Surdeanu, J Bauer, and J Finkel. 2014. ‘The Stanford CoreNLP Natural Language Processing Toolkit’, in


Marinov, M. 2008. ‘Using Frames for Knowledge Representation in a CORBA-Based Distributed Environment’, Knowledge-Based Systems

Mark Johnson, and George Lakoff. 1980. ‘Metaphors We Live by’, in, O’Brien - Language and Culture


497


Nam, Eun, Yiping Han, K Mueller, A Zelenyuk, and D Imre. 2007. ‘ClusterSculptor: a Visual Analytics Tool for High-Dimensional Data’, in, xli, pp. 75–82 <http://dx.doi.org/10.1109/VAST.2007.4388999>


Nardi, Bonnie A. 1996. ‘Activity Theory and Human-Computer Interaction’, Context and Consciousness (Massachusetts Institute of Technology)


Patton, C, D Sawicki, and James J Clark. 2015. ‘Basic Methods of Policy Analysis and Planning’ (Taylor & Francis)


Perdices, Michael, and Robyn L Tate. 2009. ‘Single-Subject Designs as a Tool for Evidence-Based Clinical Practice: Are They Unrecognised and Undervalued?’, Neuropsychological Rehabilitation, 19.6: 904–27 <http://dx.doi.org/10.1080/09602010903040691>


Pylyshyn, Zenon W. 1975. ‘Do We Need Images and Analogues?’, in (presented at the TINLAP ’75: Proceedings of the 1975 workshop on Theoretical issues in natural language processing, Association for Computational Linguistics)


Shlens, Jonathon. 2014. ‘A Tutorial on Principal Component Analysis’


Smith, George Davey, P Baudisch, G Robertson, Mary Czerwinski, Brian Meyers, and others. 2003. ‘Groupbar: the Taskbar Evolved’, in


Steele, Fiona. 2006. Event History Analysis, pp. 1–37


Hypothesis Testing the Cluster Hypothesis’, Information Retrieval


Szilas, N. 2003. ‘IDtension: a Narrative Engine for Interactive Drama’, in


Thomas, James J, and Joe Kielman. 2009. ‘Challenges for Visual Analytics’,
Information Visualization, 8.4: 309 <http://dx.doi.org/doi:10.1057/ivs.2009.26>


Tofiloski, M, and J Brooke. 2009. ‘A Syntactic and Lexical-Based Discourse Segmenter’, in


Toutanova, Kristina, and Christopher D Manning. 2000. ‘Enriching the Knowledge Sources Used in a Maximum Entropy Part-of-Speech Tagger’, in (presented at the EMNLP '00: Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics, Association for Computational Linguistics), xiii


Vandelanotte, Lieven. 2010. ‘“Where Am I, Lurking in What Place of Vantage?”: the Discourse of Distance in John Banville’s Fiction’, Barbara Dancygier and Jose Sanders (eds.), English Text Construction, 3.2 (John Benjamins Publishing Company): 203–25 <http://dx.doi.org/10.1075/etc.3.2.05van>


Voida, Stephen. 2008. ‘Personal Information Organization and Retrieval Using an Activity-Based Desktop Interface’, in


Williams, D N, T Bremer, C Doutriaux, J Patchett, S Williams, and others. 2013. ‘Ultrascale Visualization of Climate Data’, Computer, 46.9: 68–76 <http://dx.doi.org/10.1109/MC.2013.119>


Wright, William, David Schroh, Pascale Proulx, Alex Skaburskis, and Brian Cort. 2006. ‘The Sandbox for Analysis: Concepts and Methods’ (Montréal, Québec, Canada: ACM): 801–10 <http://dx.doi.org/10.1145/1124772.1124890>


Ximenes, Verônica M, Rumen Manolov, Antonio Solanas, and Vicenç Quera.


Appendix A - Transdisciplinary Gap Analysis

This dissertation has reviewed both visual available analytical tools available for analysing textual data, and the literature on textual analysis (Chapter 3, The Cognition Literature, and Chapter 4, The Analysis Literature) and has found very minimal discussion of:

1. Approaches to capturing and visualizing mental model, situation model, and schema creation in Visual Analytics tools.
2. Complimentary models of the analysis process (Badalamente and Greitzer 2005; Bodnar 2003).
3. Models of analytical reading.
4. Acknowledgement of the important role of analytical reading in the analysis process. (Except (Kang and Stasko 2014))
5. Moreover, the literature and tool review have found that no tools adequately support the task of analytical reading, in either preparation or execution phases.
6. Finally, the literature and tool review have found no tools or approaches to capturing, analyzing, visualizing longitudinal multi-tool multi-tasking workflow data.
7. However, the review has identified extensive consideration of these subjects (model formation & reading) in Cognitive Science domains (Chapter 3, The Cognition Literature) in language comprehension and particularly in discourse processing.
<table>
<thead>
<tr>
<th>Phase</th>
<th>Task</th>
<th>Object</th>
<th>VA Tools</th>
<th>Literatures</th>
</tr>
</thead>
<tbody>
<tr>
<td>Description</td>
<td>Description</td>
<td>Object</td>
<td>GeoTime</td>
<td>Visual Analytics</td>
</tr>
<tr>
<td>Preparation</td>
<td>Predicate</td>
<td>Argument</td>
<td>Tableau</td>
<td></td>
</tr>
<tr>
<td>Strategic,</td>
<td>Evoke (Activate)</td>
<td>Schemas</td>
<td>INSPIRE (5.4)</td>
<td></td>
</tr>
<tr>
<td>Procedural</td>
<td></td>
<td></td>
<td>JigSaw</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Hutchins and others 2007)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Pirolli and Card 2005)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Russell and others 1993)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Allbritton 1995)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Bier and others 2004)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Bier and others 2008)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Britton and Gülgöz 1991)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Chi and others 2005)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Coulson 2001)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Cruz and Henriques 2010)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Emmott and Alexander 2014)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Geeraerts 2006)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Glenberg and Robertson 1999)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Gómez-Rodriguez 2010)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Hard and others 2006)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Hintzman 1986)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Hutchins and others 2007)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Jacks 2003)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Judd 2015)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Kemp and others 2007)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Lahey 2014)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Langacker 2009)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Lea and others 2005)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(MacInnes and others 2010)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Miller and Myers 2002)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td>(Minksy 1973)</td>
</tr>
<tr>
<td>New</td>
<td>Search for</td>
<td>Data</td>
<td></td>
<td>Google Search Notifier</td>
</tr>
<tr>
<td>-----</td>
<td>-----------</td>
<td>------</td>
<td>---</td>
<td>----------------------</td>
</tr>
<tr>
<td>*</td>
<td>Select</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td>*</td>
<td>Judge</td>
<td>x</td>
<td>x</td>
<td>x</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>*</td>
<td>Exclude</td>
<td>(\times)</td>
<td>(\times)</td>
<td>(\times)</td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td><strong>Static data objects</strong></td>
<td>Collect</td>
<td>(\times)</td>
<td>(\times)</td>
<td>RSS aggregator</td>
</tr>
<tr>
<td>*</td>
<td>Build (Create)</td>
<td><strong>Structures</strong></td>
<td>(\times)</td>
<td>(\times)</td>
</tr>
<tr>
<td><strong>Problem Structure</strong></td>
<td>Build (Create)</td>
<td><strong>Schema</strong></td>
<td>(\times)</td>
<td>(\times)</td>
</tr>
<tr>
<td><strong>Into Schema</strong></td>
<td>Organize</td>
<td><strong>Data</strong></td>
<td>(\times)</td>
<td>(\times)</td>
</tr>
<tr>
<td>Convert</td>
<td></td>
<td></td>
<td>(\times)</td>
<td>Export to Excel, PDF</td>
</tr>
<tr>
<td>------------------------------------------------------------------------</td>
<td>---------------------------</td>
<td>------------------------------------------</td>
<td>---------------------------</td>
<td>---------------------------------------------------------------------</td>
</tr>
<tr>
<td>Create, Ingested</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Execute</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Collected, Ingested</td>
<td>Select</td>
<td>Data</td>
<td>3D Point Selection</td>
<td>alysis</td>
</tr>
<tr>
<td>dynamic</td>
<td>Evaluate (Judge)</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>dynamic</td>
<td>Filter (Exclude)</td>
<td>3D Data Filtering</td>
<td>Ad-hoc Data Filtering</td>
<td>Ad-hoc Data Filtering</td>
</tr>
<tr>
<td>dynamic</td>
<td>Filter (Exclude)</td>
<td>Mental Models</td>
<td></td>
<td></td>
</tr>
<tr>
<td>dynamic</td>
<td>Capture (Collect)</td>
<td>Mental Models</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Expert</td>
<td>Schemas</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Online real-time</td>
<td>Judge</td>
<td>Class Judgments</td>
<td>Color Sheet Tabs Calculate d Data Fields</td>
<td>labels, evidence marshalling, Make Groups Color Groups</td>
</tr>
<tr>
<td>Shallow Analytical Judgments</td>
<td>Manipulate</td>
<td>Classes</td>
<td>Evidence</td>
<td>Glyph Color, Shape</td>
</tr>
<tr>
<td>----------------------------</td>
<td>------------</td>
<td>---------</td>
<td>----------</td>
<td>-------------------</td>
</tr>
<tr>
<td>Manipulate</td>
<td>Classes</td>
<td></td>
<td></td>
<td>_glyph_color_shape</td>
</tr>
<tr>
<td>Deep Analytical Reading</td>
<td>Read</td>
<td>Portions</td>
<td></td>
<td>glyph_color_shape</td>
</tr>
<tr>
<td>Read</td>
<td>Portions</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Extract</td>
<td>Entitiess</td>
<td></td>
<td></td>
<td>Entity extraction</td>
</tr>
<tr>
<td>Extract</td>
<td>Entitiess</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Infer</td>
<td>Entitiess</td>
<td></td>
<td></td>
<td>Entity extraction</td>
</tr>
<tr>
<td>Annotate</td>
<td>Discoveries</td>
<td>Internal On Vis</td>
<td>Internal On Vis</td>
<td>Internal Rename Cluster Labels</td>
</tr>
<tr>
<td>Relate</td>
<td>Reports</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

(Bier and others 2004) (Bier and others 2008) (Chi and others 2005) (Pollock 2012) (Bier and others 2004)
Justification for my Selection of Solutions

This research proceeds on the premise that the helping with currently-under-supported, difficult analytical tasks like analytical reading requires a new, transdisciplinary approach that integrates knowledge of this cognitive task from domains in Cognitive Science into the design of analytical tools. Likewise, helping with the externalization of cognitive and functional processes requires a new transdisciplinary knowledge-integrating approach to designing analytical tools capable of capturing and reflecting this data back to the user. Finally, this research proceeds on the premise that the generation of useful, detailed descriptions of these linear, real-world process chains requires a new approach to capturing and reflecting this data back to the user. This research can thus be understood as novel, theory-based design.
Appendix B - The Full Cognitive Function Flow Model of Generic Analysis

Descriptions of each phase or stage in the Cognitive-Functional Flow Model of General Analysis are initially drawn from (Pirolli and Card 2005). Important cognitive steps that are overlooked in the sense-making model have their descriptions drawn from elsewhere in the Intelligence Analysis literature (Heuer 1999; Hampson and Cowley 2005; Jonker and others 2005; Kapler and others 2005; Cowley and others 2006; Crossno and others 2007; Eccles and others 2007; Hutchins and others 2007; Bier and others 2008; Center for the Study of Intelligence (U.S.) 2009; Chin and Kuchar 2009; Decker and others 2009; Kang and others 2009; Stasko and others 2009; Fischoff and others 2011).

1 - (Evoke)

(Evoke) functional and propositional expertise schemas

At a cognitive level, for expert analysts, the process begins with activating both the task-oriented (Pirolli and Card 2005) and data-oriented schemas necessary to begin the process of analysis.

Pirolli & Card’s paper on the sense-making model begins with a discussion of expertise schemas, but does not follow through by integrating the concept of the procedural schema into the model (Pirolli and Card 2005). This may be a principled choice, given that evoking schemas it is not a conscious action and that interviewed analysts would consequently not think to report it.

The sense-making model does use the term schematize to refer to an explicit internalized or externalized attempt to organize the particulars of a given case into a structured overview. Any re-representation of data into a
new schematic organizational form, internal or external, draws from both functional schemas of the analyst’s process expertise (Pirolli and Card 2005), as well as their propositional schemas of expert knowledge of relations and entity in a given domain. The terminology used in (Pirolli and Card 2005) is confused between functional schemas in memory based on experience of work and analytical schematization as arrangement and re-representation of elements to facilitate understanding; and data element schemas as interactive structural overview.

2 - Search

Search for data sources, information, relations of entities, items to read, selections of interest, evidential facts, inferential implications, re-representational schemas, schematic representations, theoretical explanations, convincing arguments, disconfirming evidence, alternate hypotheses, recommendations for action

Targets of search included in the sense-making model are listed in bold (Pirolli and Card 2005). Initially, various external (and internal) data sources are queried. Next, relations between extracted elements of information are sought. In the sense-making model, the term evidence is used to refer to evidentiary facts drawn from an evidence file into a re-representational schema. In the final case, search is for convincing arguments to support or disconfirm a hypothesis.

The typical sense of standard phrase to find supporting evidence for an argument is blocked by the sense-making model’s use of the term evidence for what is typically called facts. This misfortunate terminology presumes the existence of and connection to an argument, and makes search for evidence and support sound like the same thing.
2.1 - (Select)

(Select) data sources, search terms, query results, items to read, selections of interest, relations of entities, evidential facts, inferential implications, re-representational schemas, schematic representations, theoretical explanations, convincing arguments, disconfirming evidence, alternate hypotheses, recommendations for action, reporting formats, presentational styles

Though not explicitly included in the sense-making model, selection is a basic judgment that takes a variety of arguments at different points in the analysis process.

It should be obvious that selection is a key task in any interactive system (Bederson 2003, Card 1999). It should further be qualified as a decision leading to a subsequent context-specific action with the selected arguments.

2.2 - (Judge)

(Judge) relevance of data sources, search terms, query results, items to read, selections of interest, relations of entities, evidential facts, inferential implications, re-representational schemas, schematic representations, theoretical explanations, convincing arguments, disconfirming evidence, alternate hypotheses, recommendations for action, reporting formats, presentational styles

In the sense-making model, judgment of relevance iterates over different objects at different points in the process. Initially, retrieved search results are judged to be relevant or not, filtered and then placed into a sub-collection shoebox of relevant supporting data. Read textual contents are judged to be relevant or not, filtered, extracted, annotated and placed in an evidence file of relevant evidential facts.
Judgment presumes filter criteria; the ability to determine object features; as well as assessment of object features by filter criteria. It is the joint product of the available information and the additional facts and schemata the analyst brings to the analysis of this information (Heuer 1999).

2.3 - (Exclude)

(Exclude) irrelevant data sources, search terms, query results, items to read, selections of interest, relations of entities, evidential facts, inferential implications, re-representational schemas, schematic representations, theoretical explanations, convincing arguments, disconfirming evidence, alternate hypotheses, recommendations for action, reporting formats, presentational styles

Exclusion presumes object features that cannot meet filter criteria. Objects are partitioned for inclusion/exclusion according to the judgment of relevance. (Pirolli and Card 2005).

2.4 - (Collect)

(Collect) data sources, search terms, query results, items to read, selections of interest, relations of entities, evidential facts, inferential implications, re-representational schemas, schematic representations, theoretical explanations, convincing arguments, disconfirming evidence, alternate hypotheses, recommendations for action, reporting formats, presentational styles

Though this step in the analysis process is not explicitly specified in the sense-making model diagram, the end point of this process is represented as the transformational state of data when placed in the Shoebox.
Also, the text indicates that standing filters or user-driven searches collect relevant documents for further processing (Pirolli and Card 2005). Note that standing filters collect relevant documents only insofar as the analyst has specified the criteria used to judge contents, and the search agent only makes the specified selections and excludes non-matches.

Wright’s structured interviews with analysts indicate that working with a personal local knowledge base or shoebox is a common daily practice (Wright 2009, Wright 2006).

3 - Read

Read relevant texts from shoebox

Selections from shoebox collections are skimmed or read with the purpose of extracting evidence that can be used to support inference or disconfirm theories (Pirolli and Card 2005). Elements and stretches of text judged to be relevant will be extracted and annotated.

3.1 - Extract

Extract relevant selections of interest

Snippets that analysts (judge) as relevant during reading, or nuggets, together with any related low-level inferences are physically separated from their source document, annotated and collected into evidence files (Pirolli and Card 2005).

Extracted evidence may trigger new hypotheses or searches. Extracts may be used to support or disconfirm theories (Pirolli and Card 2005).

3.2 - (Infer)

(Infer) evidentiary implications
Any low-level inferences related to extracted evidentiary facts are annotated and collected into evidence files.

The text describes how extracts are used to draw inferences and that those inferences are stored as evidence (Pirolli and Card 2005), however inference is not explicitly included in the model diagram.

3.3 - (Annotate)

(Annotate) data sources, search terms, query results, items to read, selections of interest, relations of entities, evidential facts, inferential implications, re-representational schemas, schematic representations, theoretical explanations, convincing arguments, disconfirming evidence, alternate hypotheses, recommendations for action, reporting formats, presentational styles

Extracted elements of text are collected in an evidence file. In order to facilitate organization and reuse of facts and inferences, these selective extracts and their implications will require some form of labelling or feature assignment. In this way, the evidence file becomes a targeted, semantically meaningful collection of factual elements for which relations will be identified.

Selection and labelling are necessary in annotation, which Wright has identified as an important daily practice for analysts (Wright 2009, Wright 2006).

3.4 - Relate

Search for relations between entities, relations, evidential facts, inferential implications, theoretical explanations, convincing arguments, disconfirming evidence, alternate hypotheses, recommendations for action
Elements of information (evidential facts or inferential implications) in an evidence file may present patterns that generate hypotheses substantiating relations between entities or events (Pirolli and Card 2005). Such hypotheses may require further search and data extraction from the shoebox or from raw data (Pirolli and Card 2005).

4 - (Discover)

(Discover) patterns, insight

Discovery is also not explicitly included in the sense-making model. Discovery can be achieved by incremental means, or by nearly spontaneous problem-solving marked by ah-ha! moments. Such spontaneous insights emerge with seemingly little conscious preparation (Bowden 2005, Knoblich 2001, Knoblich 2006). Though there is no explanation for this phenomenon, it is known to involve tacit knowledge, inference, and information reorganization that includes discarding of old paradigms.

Incremental discoveries are made as the analyst reflects upon, relates, and re-represents evidentiary facts so as to identify patterns that are sufficiently strong enough to generate hypotheses that can substantiate relations between entities or events. (Pirolli and Card 2005)

5 - (Build)

(Build) schematic representations, cases, theories, hypotheses, reports

Build a theory or case by further marshalling of evidence (Pirolli and Card 2005). Interaction with the organized re-represented schematic of the evidential facts and inferential implications is intended to make it easier to draw conclusions (Pirolli and Card 2005). Hypotheses are tentative representations of these conclusions with supporting arguments (Pirolli and
Theories and cases are related collections of hypotheses (Pirolli and Card 2005).

The flow model restates the build action as encompassing four derivative tasks of organizing information, building schematic or diagrammatic representations, building theories based on valid conclusions of arguments, building justifications using information to support or disprove conclusions, and constructing reports (Pirolli and Card 2005).

5.1 - (Organize)

(Organize) or Marshal information, evidence, hypotheses, structure problem

The organization of information, particularly of evidentiary facts and inferential implications, is structured and captured in a re-represented schematization that can be used to more easily come to conclusions about the information (Pirolli and Card 2005).

Pirolli and Card frequently mention marshalling in the context of organizing. Since organizing is a more recognizable term, this label has been selected for use in the flow model.

5.2 - Schematize

Schematize or Re-Represent information to build a schematic representation

Schematic re-representations are structures within which analysts may organize information so as to make it easily used to draw conclusions (Pirolli and Card 2005).

Without the availability of easy-to-use tools and given poor tool integration, information is usually re-represented or schematized in the
analysts’ mind, or informally as sketches, and less frequently with elaborate computer-based methods that co-ordinate large data structures (Pirolli and Card 2005). Raw evidence can be organized into narratives or stories with typical topics or in answer to typical context-establishing questions (who, what, when, where, why, how) (Pirolli and Card 2005).

What Pirolli and Card miss is that evoking and creating schemas (schematization) is not limited to the re-structuring re-representational phase of analysis.

Cognitive Discourse Grammar offers text worlds a means to build the described schemas for both propositional and procedural knowledge (Werth 1999d). The text world schema establishes a built-in figure ground relation between relevant action and background context (Werth:1999e).

5.2.1 - (Instantiate)

(Instantiate) schema with evidential facts (or Search for evidence)

In this step, the analyst is looking for instances of evidential facts that fit the schema (Pirolli and Card 2005). Analysis or re-evaluation of theories may require re-examination of collected facts, or search for new facts (Pirolli and Card 2005).

This step is included in the sense-making model as Search for Evidence (Pirolli and Card 2005). Several reasons for modifying terminology exist. First, search is already included as a top-level, cross-cutting process in the Cognitive Flow Model. Most importantly, schemas provide structure that requires instances or content. And finally, the term evidence is conventionally used of arguments, which are built in another phase in the flow model.

5.2.2 - (Interact)
(Interact) with schematic representation

Interaction with the organizational schema that was built as an overview to the information (e.g. evidentiary facts and inferential implications) is meant to facilitate building a case, forming arguments, coming to conclusions and thus generating hypotheses in the first step in a problem-structuring meta-process, which also includes exploring and managing hypotheses (Pirolli and Card 2005).

5.3 - (Hypothesize)

(Hypothesize) likely situations, explanations, conclusions

Though this process is not explicitly included in the sense-making model, it is discussed at various points in the text. Evidentiary facts from an evidence file may upon reflection by the analyst present patterns that generate hypotheses substantiating relations between entities or events. Such hypotheses may require further search and data extraction from the shoebox or from raw data (Pirolli and Card 2005).

Hypotheses are the tentative representation of those conclusions with supporting arguments (Pirolli and Card 2005). Such hypotheses are created by the generation, exploration, and management process that define the problem structuring meta-process. This problem structuring phase of analysis is widely discussed in the policy and planning literature (Kang and Stasko 2014; Hutchins and others 2007; Pirolli and Card 2005; Gevamay and Wildavsky 1997; Russell and others 1993; Wildavsky 1987).

5.3.1 - (Disconfirm)

(Disconfirm) hypotheses
Establish disconfirming evidence that invalidates arguments that support the available theories (Pirolli and Card 2005).

5.3.2 - Support

(Interact) with Schema to Search for support

Re-evaluate and interact with the schema to justify conclusions. Analysis or re-evaluation of theories may require re-examination of the lower level schematic organization of facts. (Pirolli and Card 2005)

Conclusions are hypotheses supported by evidence.

5.4 - (Reason)

(Reason) interact with schematized evidence

Evidentiary reasoning involves interacting with a schema to marshal evidence to disconfirm or support hypotheses (Pirolli and Card 2005).

The sense-making model includes only the search for support (Pirolli and Card 2005), which may serve to underscore or reinforce the confirmation bias (Heuer 1999).

5.4.1 - (Argue)

(Argue) evidence, implications, likelihoods, conclusions

Organize (or marshal) evidence arranged to as to support cases, theories, hypotheses (Pirolli and Card 2005).

5.4.2 - (Conclude)

(Conclude) likelihoods or probabilities

Interaction with the organized re-represented schematization of the evidential facts and inferential implications is intended to make it easier to
draw **conclusions**. **Hypotheses** are tentative **representations** of these **conclusions** with **supporting arguments** (Pirolli and Card 2005).

5.5 - (Write)

**(Write) notes, reports**

Reporting is not highlighted in this model. However, the research agenda for Visual Analytics (Thomas and Cook 2005) and more recent reporting on the state of the field (Chinchor 2009) stress the importance of communicating analytic results, in order to have an impact on various audiences.

6 - (Present)

**(Present) or Tell Story**

The case is made to an audience in a presentation or publication (Pirolli and Card 2005).

6.1 - (Negotiate)

**(Negotiate) directives, feedback**

Inquiries or feedback from clients may require further evidence to support or disconfirm a theory, or even require re-evaluation of theories, arguments, schematic representations, evidence (Pirolli and Card 2005).

7 - Re-evaluate

**Re-evaluate**

Inquiries or feedback from clients may require marshalling further evidence to support or disconfirm theories, or even the generation and testing of more alternatives (Pirolli and Card 2005).
Appendix C - The Splunk> Integration, Processing, & Export Search, in Splunk> Search Language

index=flowspaces AND host="MBP*" AND eventtype="User *" OR eventtype="System *" OR (sourcetype=growl_log AND eventtype="Display Asleep") OR eventtype="Crash Recovery"
| dedup _time eventtype
| transaction eventtype maxevents=2 maxpause=20s keeporphans=true keepevicted=true mvlist=false
| eval et=if(eventtype="User Returned",1,
if(eventtype="User Idle",2,
if(eventtype="System Wake",3,
if(eventtype="System Sleep",4,
if(eventtype="System Shutdown",5,
if(eventtype="Crash Recovery",6,
if(eventtype="System Login",7,
if(eventtype="Audit Startup",8,
if(eventtype="Display Asleep",9,
if(eventtype="User Login",10,
if(eventtype="System Up",11,
if(eventtype="System Down",12,
if(eventtype="System Crash",13,
if(eventtype="App Active",14,
if(eventtype="UI Active",15,
if(eventtype="Observer Failed",16,
if(eventtype="Space Changed","17","oops"))))))))))))
| eval cycle_type=if((et=1 OR et=2 OR et=3 OR et=6 OR et=7 OR et=10),"active",if((et=4),"sleep",if((et=5 OR et=8),"power",if((et=9),"screensaver",if((et=11 OR et=12 OR et=13),"power",if((et=13),"crash",if((et=14),"app","untyped")))))),if((et=14 OR et=15),application,if(et=16,application,application)))
| eval begin_time
| eval
application=if(et=4,"Sleep",if(et=8,"Audit",if(et=9,"Screensaver",if(et=3 OR et=10,"Login",if(et=5 OR et=12,"Shutdown",if(et=6 OR et=13,"Kernel",if(et=7 OR et=11,"Startup",if(et=14 OR et=15,application,if(et=16,application,application))))))))))
| reverse
| streamstats current=t count as singles
| streamstats current=f window=1 global=f last(et) as 127_et last(singles) as 127_singles last(begin) as 127_time
| reverse
| streamstats current=f window=1 global=f last(et) as n27_et last(singles) as n27_singles last(singlescycle) as n27_singlescycle last(begin) as n27_time
| eval exclude=n27_time_time
| eval excludebit=if(exclude>7200,1,0)
| eval n27_timef=strftime(n27_time,"%m/%d/%y %T.%3N %p")
| eval singlescycle=if(excludebit=0,
if (et = 1 AND ((n27_et = 2) OR (n27_et = 4) OR (n27_et = 5)), singles + 1,
    if (et = 1 AND n27_et = 1 AND 127_et = 2, singles,
    if (et = 1 AND n27_et = 10 AND 127_et = 2, singles,
    if (et = 1 AND n27_et = 10 AND 127_et = 3, singles,
    if (et = 1 AND n27_et = 10 AND 127_et = 9, singles,
    if (et = 1 AND n27_et = 12 AND 127_et = 2, singles,
    if (et = 1 AND n27_et = 6 AND 127_et = 2, singles + 1,
    if (et = 1 AND n27_et = 7 AND 127_et = 2, singles,
    if (et = 1 AND n27_et = 9 AND 127_et = 2, singles + 1,
    if (et = 1 AND n27_et = 9 AND 127_et = 9, singles + 1
    if (et = 10 AND n27_et = 10 AND 127_et = 1, singles + 2,
    if (et = 10 AND n27_et = 10 AND 127_et = 10, singles,
    if (et = 10 AND n27_et = 2 AND 127_et = 1, singles + 1,
    if (et = 10 AND n27_et = 2 AND 127_et = 10, singles + 1,
    if (et = 10 AND n27_et = 2 AND 127_et = 2, singles,
    if (et = 10 AND n27_et = 2 AND 127_et = 3, singles + 1,
    if (et = 10 AND n27_et = 2 AND 127_et = 7, singles + 1,
    if (et = 10 AND n27_et = 2 AND 127_et = 9, singles + 1,
    if (et = 10 AND n27_et = 2 AND 127_et = 10, singles,
    if (et = 10 AND n27_et = 5 AND 127_et = 1, singles + 1,
    if (et = 10 AND n27_et = 6 AND 127_et = 1, singles + 1,
    if (et = 10 AND n27_et = 3 AND 127_et = 1, singles + 2,
    if (et = 10 AND n27_et = 4 AND 127_et = 1, singles + 1,
    if (et = 10 AND n27_et = 4 AND 127_et = 3, singles + 1,
    if (et = 10 AND n27_et = 4 AND 127_et = 10, singles + 1,
    if (et = 10 AND n27_et = 4 AND 127_et = 3, singles + 1,
    if (et = 10 AND n27_et = 2 AND 127_et = 1, singles + 2,
    if (et = 10 AND n27_et = 2 AND 127_et = 2, singles + 1,
    if (et = 10 AND n27_et = 2 AND 127_et = 2, singles + 1,
    if (et = 10 AND n27_et = 2 AND 127_et = 4, singles + 1,
    if (et = 2 AND ((n27_et = 1) OR (n27_et = 3) OR (n27_et = 7)), singles,
    if (et = 2 AND n27_et = 10 AND 127_et = 1, singles,
    if (et = 2 AND n27_et = 2 AND 127_et = 1, singles,
    if (et = 2 AND n27_et = 2 AND 127_et = 10, singles,
    if (et = 2 AND n27_et = 4 AND 127_et = 1, singles,
    if (et = 2 AND n27_et = 4 AND 127_et = 10, singles,
    if (et = 2 AND n27_et = 4 AND 127_et = 3, singles,
    if (et = 2 AND n27_et = 2 AND 127_et = 1, singles + 1,
    if (et = 2 AND n27_et = 4 AND 127_et = 10, singles,
    if (et = 3 AND n27_et = 1 AND 127_et = 1, singles,
    if (et = 3 AND n27_et = 1 AND 127_et = 2, singles,
    if (et = 3 AND n27_et = 1 AND 127_et = 2, singles,
    if (et = 3 AND n27_et = 1 AND 127_et = 2, singles,
    if (et = 3 AND n27_et = 1 AND 127_et = 2, singles,
    if (et = 3 AND n27_et = 1 AND 127_et = 4, singles,
    if (et = 3 AND n27_et = 1 AND 127_et = 1, singles,
    if (et = 3 AND n27_et = 10 AND 127_et = 1, singles,
    if (et = 3 AND n27_et = 10 AND 127_et = 4, singles,
    if (et = 3 AND n27_et = 4 AND 127_et = 1, singles + 1,
    if (et = 3 AND n27_et = 4 AND 127_et = 10, singles,
    if (et = 3 AND n27_et = 4 AND 127_et = 3, singles + 1,
    if (et = 3 AND n27_et = 4 AND 127_et = 4, singles + 1,
    if (et = 3 AND n27_et = 4 AND 127_et = 5, singles + 1,
    if (et = 4 AND n27_et = 3 AND 127_et = 1, singles,
    if (et = 4 AND n27_et = 3 AND 127_et = 10, singles,
if(et=4 AND n27_et=3 AND l27_et=10,singles,
if(et=4 AND n27_et=3 AND l27_et=2, singles,
if(et=4 AND n27_et=3 AND l27_et=3, singles,
if(et=4 AND n27_et=3 AND l27_et=9, singles,
if(et=5 AND n27_et=2 AND l27_et=1,singles
if(et=5 AND n27_et=7 AND l27_et=1,singles,
if(et=5 AND n27_et=7 AND l27_et=10,singles,
if(et=5 AND n27_et=7 AND l27_et=2,singles+1,
if(et=5 AND n27_et=7 AND l27_et=3,singles+1,
if(et=5 AND n27_et=7 AND l27_et=7,singles,
if(et=5 AND n27_et=7 AND l27_et=8,singles,
if(et=5 AND n27_et=8 AND l27_et=1,singles+1,
if(et=6 AND n27_et=7 AND l27_et=1,singles,
if(et=6 AND n27_et=7 AND l27_et=10,singles,
if(et=6 AND n27_et=7 AND l27_et=7,singles,
if(et=7 AND n27_et=10 AND l27_et=5,singles+2,
if(et=7 AND n27_et=2 AND l27_et=1,singles+1,
if(et=7 AND n27_et=2 AND l27_et=5,singles+1,
if(et=7 AND n27_et=2 AND l27_et=6,singles+1,
if(et=7 AND n27_et=2 AND l27_et=7,singles,
if(et=7 AND n27_et=5 AND l27_et=1,singles+1,
if(et=8 AND n27_et=7 AND l27_et=1,singles,
if(et=8 AND n27_et=7 AND l27_et=10,singles,
if(et=8 AND n27_et=7 AND l27_et=7,singles,
if(et=9 AND n27_et=5 AND l27_et=2,singles,
if(et=9 AND n27_et=9 AND l27_et=2,singles+1,singles )))))))))))))) )
))) ))))) ))))))))))))))))))))))))))))))))))
))) )))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))))),(time)
eval repairsingles=et+" +n27_et+" +l27_et
| streamstats current=f window=1 global=f last(singlescycle) as n27_singlescycle

| transaction name=Active
| eval firstduration=duration

| eval endtimet=(tonumber(mvindex(begin,1)))
| eval starttimet=(tonumber(mvindex(begin,0)))
| streamstats current=f window=1 global=f first(endtimet) as n27_endtime
| reverse
| streamstats current=f window=1 global=f last(endtimet) as l27_endtime
| streamstats current=t count
| reverse
| eval element=count

| eval 127_idletime=((tonumber(mvindex(begin,0)))-(tonumber(127_endtime)))
| eval 127_idletime=if(isnull(127_idletime),0,127_idletime)
| streamstats current=f window=1 global=f last(127_idletime) as n27_idletime

| eval merge=if(isnotnull(127_idletime) AND 127_idletime!=0 AND 127_idletime<16,1,0)

537
```
| eval n27_merge=if(isnotnull(n27_idletime) AND n27_idletime!=0 AND n27_idletime<16, 1, 0)
| eval dir=if(merge=1 AND n27_merge=1, "upanddown", if(merge=1 AND n27_merge=0, "down", if(merge=0 AND n27_merge=0, "stop", if(merge=0 AND n27_merge=1, "up", 0))))
| streamstats current=f window=1 global=f last(dir) as n27_dir
| streamstats current=f window=1 global=f last(n27_dir) as nn27_dir
| streamstats current=f window=1 global=f last(nn27_dir) as nnn27_dir
| reverse
| streamstats current=f window=1 global=f last(dir) as l27_dir
| reverse
| eval KILLME=if((tonumber((mvindex(excludebit,0)))==0),
  if(dir="stop", count,
  if(dir="up" AND n27_dir="upanddown" AND nn27_dir="down", count+2,
    if(dir="up" AND nn27_dir="upanddown" AND nnn27_dir="down", count+3,
      if(dir="down" AND (l27_dir="up" OR l27_dir="upanddown") AND (n27_dir="stop" OR n27_dir="up"), count,
        if(dir="up" AND l27_dir="down" AND n27_dir="down", count+1,
          if(dir="up" AND (l27_dir="stop" OR l27_dir="down") AND n27_dir="down", count+1,
            if(dir="up" AND (l27_dir="stop" OR l27_dir="down") AND n27_dir="upanddown", count+2,
              if(dir="upanddown" AND n27_dir="down" AND l27_dir="upanddown", count+1,
                if(dir="upanddown" AND n27_dir="up" AND l27_dir="upanddown", count+1,
                  if(dir="upanddown" AND n27_dir="upanddown", count+2,
                    if(dir="upanddown" AND l27_merge=up, count,
                      if(dir="upanddown" AND n27_dir="upanddown" AND nnn27_dir="upanddown", count+2,
                        if(dir="upanddown" AND n27_dir="down" AND l27_dir="upanddown", count+1,
                          if(dir="upanddown" AND l27_dir="down" AND n27_dir="up", count+1,
                            if(dir="upanddown" AND l27_dir="upanddown", count+2,
                              if(dir="upanddown" AND l27_dir="upanddown", count+3,
                                if(dir="down" AND (l27_dir="up" OR l27_dir="upanddown"), count,
                                  if(dir="upanddown", count,
                                    if(dir="stop", count, "othercondition")))))))))))))))))))))))))
| streamstats current=f window=1 global=f last(KILLME) as n27_KILLME
last(n27_dir) as nn27_dir
| eval KILLME=if(dir="up" AND n27_KILLME!=KILLME AND n27_dir="upanddown" AND nn27_dir="upanddown", n27_KILLME, KILLME)
| transaction KILLME keeporphans=true keepevicted=true mvlist=true

| reverse
| streamstats count as cycle
| reverse

| eval title=mvjoin(eventtype, " - ")
| eval starttime=if(mvcount(begin)>1,mvindex(begin,1),begin)
| eval do=exclude
| eval starttime2=strftime(starttime,"%m/%d/%y %l:%M:%S.%3N %p")
| eval endtime=if(mvcount(begin)>1,mvindex(begin,-2),begin)
| eval endtime2=strftime(endtime,"%m/%d/%y %l:%M:%S.%3N %p")
```
| eval cycle_gap=mvindex(l27_idletime,0)  
| eval last_idle=if(eventcount=1,l27_idletime,mvindex(l27_idletime,0))  
| streamstats current=f window=1 global=f last(last_idle) as next_idle  
| eval skip=if(isnull(duration) AND eventcount=1,1,0)  
| eval duration=endtime-starttime  

| inputlookup append=true max=50000000 march_apps.csv  
| sort limit=0 - _time  

| inputlookup append=true max=50000000 march_uis.csv  
| sort limit=0 - _time  
| eval app_idle=if(et=14,last_idle,null())  
| reverse  
| streamstats current=f last(sn) as l27_esn last(cycle) as l27_cycle  
| last(sc) as l27_sc last(n27_sn) as l27_n27_sn last(l27_sn) as l27_l27_sn  
| reverse  
| eval sn=if(isnull(sn),l27_esn,sn)  
| eval n27_sn=if(isnull(n27_sn),l27_n27_sn,n27_sn)  
| eval l27_sn=if(isnull(l27_sn),l27_l27_sn,l27_sn)  
| eval sc=if(isnull(sc),l27_sc,sc)  
| eval cycle=if(isnull(cycle),l27_cycle,cycle)  
| eval space_cycle=if(isnull(sc),l27_sc,sc)  
| streamstats sum(cycle_gap) as ixs by space_cycle  
| streamstats sum(app_idle) as iaxs by space_cycle  
| streamstats sum(app_idle) as iaxc by cycle  
| streamstats current=f last(nn) as n27_esn  
| eval app_d=if(et=14,duration,null())  
| reverse  
| eval ixs=if(et=17,ixs,null())  
| eval iaxs=if(isnotnull(ixs),if(et=17,iaxs-ixs,null()),if(et=17,iaxs,null()))  
| eval iaxc=if(cycle_type="active",iaxc,null())  
| eval sd=if(et=17,duration-ixs,null())  
| streamstats current=f window=1 last(starttime) as next_st  
| streamstats current=f window=1 last(endtime) as next_et last(app_idle) as n27_app_idle  
| eval cyclest=if(cycle_type="active",starttime,null())  
| eval cycleet=if(cycle_type="active",endtime,null())  
| reverse  
| streamstats current=f last(cycleet) as cycle_et by cycle  
| streamstats current=f latest(cyclest) as cycle_st by cycle
| streamstats current=f window=1 global=f last(space_idle) as l27_space_idle |
| reverse |
| streamstats current=f latest(next_st) as next_cycle_st by cycle |
| eval next_cycle_st=if(isnull(next_cycle_st),next_st,next_cycle_st) |
| eval cycle_st=if(isnull(cycle_st),starttime,cycle_st) |
| eval cycle_et=if(isnull(cycle_et),endtime,cycle_et) |
| eval iaxc2=iaxc |
| eval iaxc=iaxc2-n27_app_idle |

| eval killsd=if(et=17, |
| if(starttime>cyle_st, |
| if(starttime>cycle_et, |
| if(endtime>next_cycle_st OR endtime=next_cycle_st, (endtime-next_cycle_st), |
| if(endtime>cycle_et, 0,null())), |
| if(endtime>next_cycle_st OR endtime=next_cycle_st, (endtime-starttime)-ixs, |
| if(endtime>cycle_et, cycle_et-starttime,if(endtime>next_cycle_startime,(cycle_et-starttime)+(|endtime>next_cycle_st),if(starttime>cycle_et,(endtime-starttime)-ixs,null()),null()))), |
| if(endtime>next_st,if(endtime<cycle_et AND starttime>cycle_st,endtime-starttime,if(endtime<cycle_et AND starttime<cycle_st,if(starttime>cycle_et,0,(endtime-starttime)),(endtime-next_cycle_st)+(cycle_et-cycle_st))),cycle_et-cycle_st),null()) |
| eval duration=if(killsd>0 AND isnotnull(sd) AND sd>0 AND |
| killsd<sd,killsd,if((isnull(sd) OR sd<0) AND |
| isnotnull(killsd),killsd,if(et=17 AND sd>0 AND isnotnull(ixs),duration-ixs,duration))) |

| streamstats sum(app_d) as daxs1 by space_cycle |
| eval daxs=if(et=17 AND isnotnull(daxs1),daxs1,null()) |

| eval kl=if(et=17, |
| if(starttime>cyle_st, |
| if(starttime>cycle_et, |
| if(endtime>next_cycle_st OR endtime=next_cycle_st, "1 starts after current cycle & ends in next cycle, foreshorten starttime to next_cycle_st", |
| if(endtime>cycle_et,"2 starts after and ends after cycle. delete space duration","3 Starts & ends within current cycle. ignore")), |
| if(endtime>next_cycle_startime OR endtime=next_cycle_st, "4 ends after next cycle begins.", |
| if(endtime>cycle_et AND starttime<cycle_st,"5 starts before and ends within cycle. Find out how much before",if(endtime>next_cycle_startime,"6 Starts in current & ends in
541

next. compute overlap",if(starttime>cycle_st AND endtime<cycle_et,"7
starts and ends in current. ignore","DOH")))

if(endtime>next_st,if(endtime<cycle_et AND starttime>cycle_st,"8 starts
& ends in current cycle. ignore",if(endtime<cycle_et AND
starttime<cycle_st,if(starttime>cycle_et,"9 starts after current cycle &
ends in next cycle. So Zero-out","10 starts BEFORE current cycle and
ends BEFORE current cycle. so why is it in this cycle? erase?"),"11
Starts before current, s
o take cycle duration and add the overlap into
the next session.")),"12 starts before current cycle & ends between
cycles.Use cycle duration")

"nope"

| eval sd_to_cg=if(et=17,duration/do,null())
| eval daxs_to_d=if(et=17,daxs/duration,null())
| eval i axs_to_d=if(et=17,iaxs/duration,null())
| eval error=daxs+iaxs-duration

| eval transcount=eventcount
| eval eventcount=mvcount(et)

| eval duration=if(et=15 AND duration>7200, 7200, duration)

| eval sec=duration
| eval min=sec/60
| eval hrs=min/60
| eval ctt=cycle_type
| eval cycle_type=if(eventcount>1,mvindex(ctt,-1),ctt)
| eval at=application
| eval pa_chain=if(cycle_type="active",mvjoin(at, " ← "),pa_chain)
| eval application=if(eventcount>1,mvindex(at,-1),at)

| eval ett=et
| eval evtt=eventtype
| eval eventtype=if(eventcount>1,mvindex(evtt,-1),evtt)
| eval et=if(eventcount>1,mvindex(ett,-1),ett)

| eval sca=if(isnull(sca),sc+application,sca)
| eval scact=sca+cycle_type
| eval csc=tostring(cycle)+tostring(sc)
| eval ca=cycle+application
| eval cact=ca+cycle_type
| eval cct=cycle+cycle_type
| eval scct=sc+cycle_type

| streamstats c(cct) as ctXc by cct
| streamstats c(scct) as ctXsc by scct
| streamstats c(ca) as aXc by ca
| streamstats c(scact) as aXsXc by scact
| streamstats dc(application) as tacXc by cct
| streamstats c(sca) as ac by sca

| eval appXc=if(cycle_type="active",0,if(et=14,ctXc,null()))
| eval uiXc=if(cycle_type="active",0,if(et=15,ctXc,null()))
| eval spaceXc=if(cycle_type="active",0,if(et=17,ctXc,null()))

| eval appXsc=if(cycle_type="space",0,if(et=14,ctXsc,null()))
eval uiXsc=if(cycle_type="space",0,if(cycle_type="ui",ctXsc,null()))

activeXsc=if(cycle_type="space",0,if(cycle_type="active",ctXsc,null()))

streamstats current=f last(appXc) as n27_appXc last(uiXc) as n27_uiXc last(spaceXc) as n27_spaceXc last(cycle_type) as n27_cycle_type last(appXsc) as n27_appXsc last(uiXsc) as n27_uiXsc last(activeXsc) as n27_activeXsc

streamstats current=f last(n27_cycle_type) as nn27_cycle_type

eval

n27_appXc=if(n27_cycle_type="active",0,if(nn27_cycle_type="active",0,n27_appXc))

eval

n27_uiXc=if(n27_cycle_type="active",0,if(nn27_cycle_type="active",0,n27_uiXc))

eval

n27_appXsc=if(n27_cycle_type="space",0,if(nn27_cycle_type="space",0,n27_appXsc))

eval

n27_uiXsc=if(n27_cycle_type="space",0,if(nn27_cycle_type="space",0,n27_uiXsc))

eval

appXc=if(appXc=0 AND cycle_type="active",n27_appXc,appXc)

eval

uiXc=if(uiXc=0 AND cycle_type="active",n27_uiXc,uiXc)

eval

spaceXc=if(spaceXc=0 AND cycle_type="active" AND n27_spaceXc>0,n27_spaceXc+1,if(spaceXc=0,1,spaceXc))

eval

appXsc=if(appXsc=0 AND cycle_type="space",n27_appXsc,appXsc)

eval

uiXsc=if(uiXsc=0 AND cycle_type="space",n27_uiXsc,uiXsc)

eval

activeXsc=if(activeXsc=0 AND cycle_type="space" AND n27_activeXsc>0,n27_activeXsc,if(activeXsc=0,1,activeXsc))

reverse

streamstats current=t count as count2

reverse

streamstats dc(application) as iacXsc by sc

streamstats c(sca) as acXsc by sca

streamstats dc(application) as iacXc by cycle

streamstats c(ca) as acXc by ca

sort limit=0 - _time

table  _time starttime starttime2 endtime2 endtime next_st next_et next_cycle_st cycle_et skip eventtype evtt et ett eventcount
duration do sd_to_cg odsum killsd k1 sd 127_sd daxes next_idle ii

last_idle space_idle cycle_gap app_idle ixs_iакс daxes_to_d iaks_to_d
eror n27_ct 127_ct sec min hrs join jt cycle count count2 127_cycle ctt

ui sc 127_sc csc ca cact cct scct sca scact cycle_type ctXc appXc uiXc

spaceXc aXc aXsc ctXsc appXsc uiXsc activeXsc iacXc acXc iacXsc acXsc

sn ZACTIVITY zact application title at tt toast n27_sn n27_esn 127_sn