WHERE THE RICE COOKS: CONNECTING TEXT AND COMMUNITY AT *RICEPAPER* MAGAZINE THROUGH DEVELOPING A CONTENT MANAGEMENT SYSTEM

by

Jessica Gin-Jade Chan
Bachelor of Fine Arts, University of Victoria, 2003

PROJECT SUBMITTED IN PARTIAL FULFILLMENT OF
THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF PUBLISHING

in the

Faculty of Arts & Social Sciences
Publishing Program

SIMON FRASER UNIVERSITY

Fall 2005

APPROVAL

Name: Jessica Gin-Jade Chan

Degree: Master of Publishing

Title of Project: Where the Rice Cooks:
Connecting Text and Community at *Ricepaper* Magazine
Through Developing a Content Management System

Supervisory Committee:

____
John Maxwell
Senior Supervisor
Instructor, Master of Publishing Program
Simon Fraser University

____
Ron Woodward
Senior Lecturer, Master of Publishing Program
Simon Fraser University

____
Jim Wong-Chu
Industry Supervisor
Publisher, *Ricepaper* Magazine
Suite 12, 2414 Main Street
Vancouver, British Columbia Canada V5T 3E3

Date Approved: Sept 07/2005

SIMON FRASER UNIVERSITY

PARTIAL COPYRIGHT LICENCE

Title of Project:

Where the Rice Cooks:
Connecting Text and Community at *Ricepaper* Magazine
Through Developing a Content Management System

Author:

Jessica Gin-Jade Chan

Sept. 07/2005
Date Signed

ABSTRACT

This report is a case study that explores *Ricepaper* magazine's efforts to develop a content management system (CMS) at a time when the small non-profit publication was reorganising its internal administration and publishing process. Magazines traditionally facilitate the fluid relationship between community and content. This report is a case study that examines how *Ricepaper* Magazine ("*Ricepaper*")—a small non-profit publication—used technology and software to streamline its work cycle and build intra-community networks. It documents and explores the magazine's attempts to develop Ricecooker, a software system that was conceptualised to improve communications and unify all department processes with a centralised content management system (CMS).

By detailing an in-depth look at *Ricepaper*'s development experience, this report undertakes an examination of all the attendant issues that accompany such a project—organisational evolution, decision-making, system limitations, unforeseen problems, and unexpected solutions. It will also look at how *Ricepaper* has utilised alternative communication technologies to semi-automate their workflows, promote cross-unit interaction, and deepen the magazine's relationship with readers and staff communities.

Overall, this report offers insight to: the production workflow at a volunteer-based publication, and the management of an evolving magazine. Finally, some reflections are offered with regard to planning for change, and how process management can inform software development.

DEDICATION

For My Mother

ACKNOWLEDGEMENTS

I would like to extend my sincere thanks to John Maxwell and Ron Woodward. I could not have hoped for a better Advisory Committee, who provided me with much direction and insight. They were willing to take a leap of faith on my accelerated writing schedule, and generously took time out to share their knowledge and enthusiasm in my work. Their unwavering belief in me has been a key ingredient towards making this project report a possibility.

Thanks also to Dr. Rowland Lorimer, who has guided me throughout my studies at the Master of Publishing Program. He has been, and continues to be a pillar and trailblazer in the Canadian Publishing industry.

Jo-Anne Ray, my Program Advisor, is another person who has enriched my graduate studies. I thank her for the advice she always has on hand, and the extra care she has put into making this program a smooth transition for me.

I am grateful for the opportunity of working at *Ricepaper* Magazine. The experience has been a rewarding challenge for me in both my education and career goals. Special thanks go to my Publisher, Jim Wong-Chu, who not only brought me to *Ricepaper*, but also had the awareness for what I could do there. I would like to acknowledge the mentorship he has shown me over the past three years.

Sincere gratitude also goes to my core staff—Carol Lee, Jenny Uechi, Michelle Siu, Jonathan Lin, Carl Queton, and Hennie Choy—for the patience and forbearance they have shown me all through this project. These are the people who showed me what a lot of dedication and a love for publishing can accomplish—here's to them, and our many years ahead.

TABLE of CONTENTS

LIST of TABLES

LIST of FIGURES

CHAPTER 1. Introduction

This report is a case study that explores *Ricepaper* magazine's efforts to develop a content management system (CMS) at a time when the small non-profit publication was reorganising its internal administration and publishing process. The magazine's staff took more than two years to conceptualise and develop "Ricecooker", a home-brewed software solution for managing content and inter-departmental processes. Staff from all departments and all levels were involved, spending hundreds of volunteer hours conceptualising, meeting and planning how to use technology to better meet the needs of the evolving magazine. They researched ways to streamline workflow, encourage inter-departmental communications, and manage their expanding volunteer resources—changes that they believed were necessary to stabilise the magazine's production processes, and ensure long-term sustainability.

Throughout this period, I held a number of different roles at the magazine—Assistant Business Manager, Executive Editor, and most recently, Editor-in-Chief—roles that have allowed me to participate in the CMS development project.[1] This report synthesizes my first-hand observations of the Ricecooker development experience, and provides an analysis of the project, and its subsequent effect on the organisation.

A custom-designed CMS solution, our management believed, would be able to do all this. However, the ideal CMS solution had to be within the magazine's limited financial resources. Trying to balance need and limitation, the management made the decision to go ahead with Ricecooker, a CMS project that would change the daily functions of nearly every department in the magazine.

[1] As the current Editor-in-Chief, I am part of the magazine's three-person senior management. Throughout this report, I refer to the "senior management" whenever I describe collective decisions and actions taken by this group of staff.

In this document, I present a case study that explores *Ricepaper*'s efforts to conceptualise, develop and test Ricecooker. This report focuses on the interactions and processes of *Ricepaper*'s most established departments: editorial, art and production.

I will begin by exploring how magazines are utilising different technology to reach out to their communities, and then proceed to a description of CMS technology, and how *Ricepaper* came to determine its need for one. I will examine the decision-making process of the staff and management at *Ricepaper*, and the steps that they took to accommodate organisational evolution, and solve unforeseen problems. Lastly, I will examine the impact of this development experience on the magazine's staff, production workflow, and organisational culture.

This report will not evaluate the ultimate success of Ricecooker, as the magazine's staff are currently programming another version of the CMS. Instead, I will focus on the development process, and its impact to date. Through this, I will attempt to place CMS technology in a specific publishing context by looking into the relationship between textual content, workflows, and a magazine's staff community.

The intent of this paper is to describe how organisational growth and technological change can be managed within the limitations of a small non-profit magazine. *Ricepaper* provides an interesting case study as a volunteer-driven magazine that used technology to automate their processes, enhance communications, and deepen their relationship with readers and staff. The organisation's creative strategies, its decision-making process, and its ability to embrace change are exemplified by its decision to develop Ricecooker. This report should be of interest to those who manage a non-profit publication, for magazine publishers adopting new technology, or for publishers trying to find creative solutions to manage workflow and communications issues.

1.1 A Magazine and Its Community

Magazines facilitate the fluid relationship between its staff, readers and editorial content. They facilitate for their readers the creation of both personal identity and communities of shared interest. Throughout history, "people with shared interests use communications technologies (both hi- and lo-tech) to help form themselves into self-created and self-organizing groups. To a significant degree, these are held together by documents circulating among members, keeping each conscious of being a group member and aware of what others are up to".[2] Magazines service their communities in much the same way—printed pages are bound together, and circulated to a community of people, linked by interest, identity, geography, or a combination of these factors.

[2] John S. Brown and Paul Duguid, "The Social Life of Documents", *First Monday*, vol. 1, no. 1, 1996.

In considering the role of magazines in our society today, it is not possible to ignore the advent of information reproduction technologies—photocopiers, faxes, the internet—that have allowed people with shared interests to form a "social world" with relative ease and autonomy. According to Brown and Duguid, this development made it possible for scattered groups of unacquainted people to form robust social worlds with one another: "From hound dog owners to herbologists, and from fans of *The Avengers* to *Star Trek*'s 'Trekkers', the easy circulation of shared communications has helped build well-coordinated social groups with a strong sense of shared identity". [3]

Typically, a magazine captures and serves its community through its printed pages. The definition of "community" becomes important when discussing the details of a magazine and its readership. For the purposes of this report, I define a magazine's community according to the three core characteristics offered by Tharon W. Howard, a communitarian theorist. Howard argues that all communities share

> first, to varying degrees, some set(s) of "beliefs and values". Second, a community's members must communicate in such a way that "relations between members should be direct and they should be many-sided". And third, communities have a characteristic that [Taylor] calls "reciprocity". Members of a community make short-term sacrifices in order to receive the long-term benefits of membership in the community.[4]

A magazine's community consists of its reading public and its in-house staff. Members of this reading public generally "subscribe" to a common set of beliefs and interests, often set out in the pages of the magazine. These readers might communicate directly with the magazine staff and other readers through letters to the editor, or publicity events. As well, readers or subscribers might discuss the magazine, its content, and any related issues or interests. All these different forms of interaction link the members of the magazine's community.

In magazines where the final product is produced by both paid staff and volunteers, the boundaries distinguishing reader from staff become blurred. In Canada, the periodical publishing industry relies largely on volunteers and unpaid staff. In 2003, just over 5,200 volunteers helped to produce magazines. In contrast, the industry had nearly 6,500 full-time employees, and just over 3,000 part-time employees.[5]

Typically, magazines are produced "from the few" and distributed "to the many," based on the understanding that "the few" and "the many" are a community that share a common set of interests and values. This concept of "narrow-casting"—specific information of assured

[3] Ibid.

[4] Tharon W. Howard, *A Rhetoric of Electronic Communications.* (London, England: Ablex Publishing Corporation, 1997), Pg. 65.

[5] Statistics Canada, The Daily: Periodical Publishing [online], June 8, 2005.

value to definable and reachable audiences—has come to be ever more central to periodical publishing since the late 1950s.[6]

"Narrow-casting" is a term that characterises the relationship between the magazine medium and their audiences. The key is magazines' special relationship with their community—one that is ideally characterized by unusual loyalty, affinity, shared personal interests and ideologies. Taken all together, magazines have a unique ability to provide high-value information targeted at self-defined audiences. Based on this, it can be argued that magazines seem uniquely suited to take advantage of the digital (internet/interactive) future.

1.2 Magazine Publishing and Content Management

Traditionally—that is, before the advent of digital media—communications from the magazine to its community was limited to the content that was published in its physical pages. However, the digital age has changed content production, and along with it, increased the competition in the publishing industry. Just as Brown and Duguid have observed:

> As the number of documents multiplies dramatically, and their reach is extended by information technology, the challenge of engaging an intended audience grows too. The swelling number of documents and the shrinking amount of time available for each one raises the problem of what Richard Lanham calls the "economy of attention".[7]

In an increasingly crowded attention economy, the challenge of reaching an intended audience requires creativity on the part of magazines. Instead of the traditional "text to page" format, today's market demands the production of editorial content that can subsequently be delivered in different forms. Magazines are no longer transferring their content verbatim to the Internet. Many are using their Web sites and various print medium to update content between publishing cycles, to provide a forum for communities of interest to interact together, and to engage people in interactive experiences such as games, polls, and contests.[8]

Magazine publishers who have a lot of related content are looking for ways to re-combine it into new and different products. The most obvious example is by reselling print content for the web, trading it with another print publication, or repackaging overflow content as a print supplement. In order to do this in a quick and cost-effective way, a "reservoir" of immediately usable content is necessary. By holding this content in an organised digital storehouse, in-house staff can increase their ability to meet the changing demands of their reading community. However, the simultaneous creation of digital and hard copy content can be time-consuming and expensive if the magazine staff do not integrate the separate

[6] Mark Glaser, "The Future of Magazines: Net Could Empower Readers," *USC Online Journalism Review*, May (2005).
[7] Brown and Duguid, "The Social Life of Documents."
[8] Mark Glaser, "The Future of Magazines: Net Could Empower Readers."

procedures into a single process. Integration is difficult to implement, particularly if the staff have disorganised data sources, or if they do not manage their content properly. All these constitute the foundation of content management.

1.3 The Concept of Content Management

It is best to think of content management as a broad concept that covers all aspects of publishing content with digital tools. Given that content management is inextricably tied to publishing activities, the natural question that comes to mind is: "How is content managed?" In their book *Content Management Systems*, Suh et al. have identified that content management does just three things in the publishing environment:[9]

A) Asset Management

In publishing, each unit of content is an asset. Day in and day out, publishing staff create and manage these assets. In order to determine which of these assets should be disseminated to a wider audience, publishers submit this content to an evaluation and organisation process called "asset management". This process formalises and prepares the assets for the next few steps.

B) Transformation

Once content assets are available, publishers make choices about how to present that content, and whether it should be packaged for print or online distribution. After they make this decision, publishers usually attempt to shoehorn their content into some pre-established design templates. This process of templating is called "content transformation", and often takes place during the magazine production period. The right application of design to content does more than just make it look attractive; it enhances its effectiveness and impact.

C) Publishing

The final step in content management is to deliver the message. This publishing step considers the publication's community, and makes sure that the content is available to them. Content is published in whatever formats the community may need (print, HTML, database feeds), and for various medium and devices (print magazines, browsers, or legacy systems).

The publishing phase is primarily technical and logistical. It deals with getting the transformed content out to the intended audience. This includes deploying static web pages, circulating teaser content on e-mail list serves, or releasing the printed magazine to newsstand distributors. The publishing phase is heavily impacted by the choices made in the previous two

[9] Phil Suh et al., *Content Management Systems* (e-book: Glasshaus, 2003).

activities. As content progresses from asset management to transformation, and finally to publishing, progressively fewer people are involved in the process. At the same time, these individuals will require an increased level of technical knowledge.

1.3.1 A People-Centric Activity

Content management is an activity that tends to take place in an organisational environment that includes goals, organisational culture, and decision-making processes.[10] As such, content management tends to, in some ways, reflect the people who undertake the activity. In the broadest sense, organisations and people all use web sites to communicate. While this communication is not the same as speaking on the phone or writing a letter, it is, ultimately, person-to-person communication. Content management is about facilitating the technical aspects of this type of communication. It is also important to note that content management is also a people-intensive activity. Content managers tend to focus on the technology at hand, while neglecting the fact that this technology exists to help staff get their jobs done faster, and more efficiently. Content management is about applying technology to real human problems. This means that workers will need to be trained, and management will need to explain what a CMS does. In implementing a CMS, an organisation's management needs to set expectations, and conduct internal evangelism. Effective CMS always take into account the human aspect.

1.3.2 A Set of Processes

Content management marries and merges human and technical processes. In any organization, there are protocols and procedures for the flow of work. Content management is no different–it is an agreement between workers. An agreement or process-map concerning how things are done is not a new concept for organisations. However, in content management, this agreement has to be more explicit, and the enforcement of the process needs to be programmed into the workflow.[11] This workflow enforcement can change the nature of the original protocol. Hence, if it is poorly conceived, the technical enforcement of this process will only serve to calcify it. In publishing, processes change all the time. As such, content management has to follow an accurate, but flexible workflow that reflects the current process agreements. Based on this, I venture to say that a CMS should ultimately be a process-management system.

[10] Ibid.
[11] Ibid.

1.3.3. An Underlying Infrastructure

Content management is the infrastructure that supports the many mundane tasks of people and their processes. For instance, a proper content management system (CMS) can remind staff to submit, approve, and paste content into a template. The automation of these processes can support the most important aspects of content management, and allow people to work at a higher level. Authors can focus on what they write, rather than on waiting for an editor to make changes for them. Designers can conceptualise graphics and art, rather than focus on tedious, repetitive formatting and layouts.[12]

1.3.4. Premeditation

A CMS should not be adopted as software that will magically solve all existing process issues. Prior deliberation is necessary, simply because content management is a process that affects all levels of people, their processes, and working habits. The planning stages for such a system involve:

- thinking about and identifying the organisation's problems,
- listing the requirements needed to solve these problems,
- developing a process to meet these requirements,
- researching a CMS that might match these processes and requirements.[13]

1.4 Managing Content in a Publishing Environment

Professionals in the publishing industry can argue that they have always managed content. Publishing is, after all, about knowing what to publish, producing this content, and distributing it to the market. The intermediary step of "producing" involves managing content through a production life cycle.[14]

In magazine publishing, content generation normally originates from the author's personal computer. When the draft content is sent to the editor, they might save the file on their office computer. If an article goes through several reincarnations, the author and editor may create subsequent drafts, and store the many versions of the same document in separate computer systems. During this process, should someone ask after a specific stage in the article's development, it might be a challenge to locate that particular file version quickly.

Throughout the publishing cycle, freelance authors, editors, proofreaders, designers, and printers will all be the custodians of content at some point, and each will perform a specific

[12] Ibid.
[13] Ibid.
[14] James Wright, "Exploiting Your Assets," *Publishers' Research Quarterly*, Fall (1999): 84-94.

function on the content. However, this means that knowledge about the content is broken down to a functional level. The only way department heads can find out what stage the content has reached is by contacting the freelance staff who is in charge of that particular function.

During this process, the content can be said to have only a virtual existence, as its various bits and pieces are scattered in the hands of various third party freelancers. These third parties constantly move the content towards a final stage of amalgamation in the production cycle. Department heads mostly just manage the process, never actually working on the content itself. As such, it may be difficult for them to monitor this process, as they could lack critical knowledge about details on the scattered information. When department heads have less than perfect knowledge of day-to-day progress, it becomes a challenge to control the overall production cycle. Poor monitoring can further exacerbate the entire process, especially if parties further along the work chain build slack into the schedule, creating bottlenecks.

All this goes to show that publishing professionals manage their content informally. Rather, their expertise lies in the management of those who perform functions on the content. This management style has created a virtual organisational structure that gives publishers certain skill sets, while leaving other types of expertise to their partners.[15] Once upon a time, this heavy reliance on others might have seemed cost effective because it freed up department heads to go about the rest of their business. Nevertheless, the overall impacts of this management style cannot be ignored: there lies the potential for inconsistent delivery and delays. At the same time, this set of problems make magazine publishers ideal candidates for content management solutions.

1.5 The Importance of Understanding the Production Process

Disorganised digital production, with files scattered over many computers and servers without any control is a recipe for disaster. A CMS in itself will not solve these problems, but it does create an appropriate organisational framework to tackle the issues holistically, rather than disjointedly. Since all working files are managed centrally in the content store, software management, support and security procedures become easier tasks.

Installing a CMS is not simply a matter of designing a database and dumping content into it. As will be seen from *Ricepaper*'s experience, this runs the danger of turning this digital content storehouse into nothing more than a burial ground. An effective CMS manages the content from birth to archive. It links content to metadata and identifiers, as well as gives users the capacity to track versions, enforce standards, and control quality throughout the content development process.

[15] Ibid.

Creating a good CMS is tightly linked with production systems. A good CMS can require the extensive re-engineering of key processes in the product life cycle. This involves not only the re-organisation of working relationships in the organisation, but also creates new demands on the skills of publishers. This is the reason why it is important for end-users to be involved in the development and requirements gathering process. As I have outlined in this chapter, the relationship between people and processes is crucial in content management. "Participatory design"[16] gives users the opportunity to be involved in the design of a system that will ultimately transform the way they perform their daily tasks. This also helps to ensure that the user will be able to relate what they do, and what they produce to the final CMS. To put this in context, content management often requires publishers to commit to a substantial re-thinking of their production process. This is because the whole point of the process has changed—from producing a magazine to producing content for a digital repository which can be manifested in a number of different forms.

As I mapped out in Section 1.3, content management gives an underlying infrastructure to a set of processes between human and technological interaction. As such, if a magazine publisher is not completely in control of their current production process, a first step must be taken to understand and map these procedures. Before an organisation can change current production procedures, it is necessary to understand them.[17] Given that the expertise of publishers lies in managing people who perform functions on the content, the decision to adopt a CMS requires publishers to take a different attitude towards production management.

[16] Pelle Ehn. *Work-Oriented Design of Computer Artifacts.* (Stockholm: Arbitslivscentrum, 1988), 188.

[17] L. Schwartz, "Must Change, Will Change: Process Re-engineering in Publishing," *Publishing Research Quarterly* Issue 15, no.3 (1999): 99-108.

CHAPTER 2. *Ricepaper*: An Overview

Ricepaper is a volunteer-driven arts and culture magazine that describes itself as a "national forum which showcases and develops literary talent in the Asian-Canadian community."[18] Based in Vancouver, *Ricepaper* produces four issues per year at approximately 56 pages per issue. Printed in black and white and saddle-stitched with a glossy four-colour cover, the magazine currently prints 1,100 copies for national distribution. Approximately half of its distribution consists of paid copies (newsstand sales and subscribers).

The magazine has an estimated national circulation of 3,500,[19] with approximately three readers per copy of every circulated issue.*Ricepaper*'s readership is primarily between the ages of 25 and 35. Of these readers, 60 percent are Asian, and 40 percent are professional and educated people. The readers are "informed, sensitive and actively involved in the community and with cultural issues."[20]

During the last two years, the organisation has reviewed current and future needs for the publication, and has undertaken projects in the areas of information technology development, communications, production and editorial management.

2.1 History

Like many arts and literary magazines, *Ricepaper* had humble beginnings. The magazine began in 1995 as a newsletter of the Asian Canadian Writers Workshop (ACWW). ACWW is a non-

[18] Ahnsu Consulting Group, "Regional Marketing Plan for *Ricepaper* Magazine: A Review, Recommendations and Activities to Develop a Regional Marketing Strategy" (report presented at the annual marketing review meeting for *Ricepaper*, Vancouver, BC, April 15, 2005), 5.
[19] Ibid, 6.
[20] *Ricepaper* Magazine. "Canada Council Report on the Publishing Program," (grant application submitted to the Canada Council for the Arts, Ottawa, ON, February 2005.

profit Vancouver-based organisation that assists Asian Canadian writers in publishing, and artists and performers in showcasing their works in diverse venues. To this day, ACWW remains the official publisher of *Ricepaper* magazine.

Over the years, *Ricepaper* has outgrown its newsletter format, and has taken shape as a full-fledge maturing arts and culture non-profit magazine that has captured a national reading audience. Now in its tenth year, the magazine provides in-depth perspectives on arts, literature and culture by highlighting the creative works of Asian-Canadians. It serves as a collaborative platform for and about Canadian writers, artists, performers and filmmakers, with a special emphasis on those living in British Columbia.

Today, *Ricepaper* is a volunteer-run publication that endeavours to serve its readership and community by providing a space where established and emerging artists can reach a wider audience, and where new entrants into Vancouver's magazine publishing community can gain hands-on experience and mentorship.

2.2 Mission Statement and Values

Mission Statement

In maintaining its continued growth and leadership role in its community, *Ricepaper's* mission statement is to:

> Reflect the diverse interests of Asian-Canadians, and provide an alternative to the mainstream media for both readers and advertisers.

For the past few years, *Ricepaper* has served to:

> Connect the community and challenge the parameters of how Asian-Canadians are perceived and defined.[21]

In a market that is overflowing with special interest titles, the magazine has managed to retain and attract new readers because of its grassroots approach to meeting the needs of its reading community. Recognizing that many of its readers are aspiring artists and writers, *Ricepaper* and ACWW offer monthly writing workshops, taught by acclaimed Asian writers (members of ACWW). As well, ACWW provides manuscript evaluation services to *Ricepaper* subscribers, and helps connect these aspiring writers with appropriate publishers.

[21] Arts Ink Consulting, "*Ricepaper* Strategic Plan and Implementation Plan" (report presented at the annual strategic planning meeting for *Ricepaper*, Vancouver, BC, April 30, 2004), 15.

In addition to being a successful niche publication, *Ricepaper* prides itself on offering a rewarding work environment for its staff and volunteers. As a volunteer-run organization, *Ricepaper* makes it a priority to provide learning opportunities for its volunteers. While volunteers may belong to a primary department, they may often take on tasks from another department. This opportunity to move between departments gives volunteers an all-rounded experience in small magazine publishing.

With eight paid staff and over 60 volunteers working full-time and part-time at *Ricepaper*, it is a credit to the organization that many of its volunteers have worked there for a third of the magazine's publishing life. Often, volunteers join the organization for a few years, depart, and return to continue their contributions to the publication. Their work ethic and commitment to *Ricepaper* echoes the organization's mission statement, which comprises three core values:

> a) *Ricepaper* volunteers value a commitment to the Asian-Canadian community, and also the opportunity to gain publishing experience.
> b) *Ricepaper* management values integrity and leadership, perseverance and a willingness to learn, and a consistent approach to performance and procedures.
> c) *Ricepaper* management and volunteers value an innovative approach to social and cultural change. [22]

2.3 The Multi-Community Challenge

Ricepaper has a unique volunteer base, consisting of professional and amateur publishing staff located across Canada. As well, the magazine caters to the needs of a geographically, professionally and culturally diverse readership, many of whom are also volunteer staff. In producing a product that captures the attention and meets the needs of these communities, *Ricepaper* has to compete with the vast number of other Asian-targeted media on the market. Brown and Duguid, authors of *The Social Life of Documents*, note, "The central issue here is for the intended audience to be able to recognize documents intended for them."[23] In *Ricepaper*'s case, the magazine has four main challenges:

1) ***The magazine serves as a link between each of its Asian sub-communities.*** *Ricepaper*, unlike other ethnic publications, does not target only one ethnic community. Rather, its readership encompasses several traditionally disparate Asian communities, which have

[22] Ibid,16.

[23] Brown and Duguid, "The Social Life of Documents."

their own separate languages, and interact primarily within their own ethnic circles. *Ricepaper* is the "umbrella" publication that links these Asian communities, and represents them in the English media.

2) ***The magazine reaffirms the cultural identity of its readership through its published literature.*** *Ricepaper*'s content reflects the Asian Canadian community by redefining the landscape of contemporary Canadian literature and arts. As a "cultural object", the magazine influences its readers—"not just connecting, but coordinating social performance" in this community.[24] By showcasing the work of emerging and established artists, the magazine gives them acknowledgement and status in the Canadian arts scene.

3) ***The magazine bridges its ethnic community with the publishing industry.*** *Ricepaper* has attracted the interest of the publishing industry—partners who have joined its community by serving as consultants and joining its staff. These experienced staff mentor aspiring publishers through internships in editorial, production, marketing, and business management. In the last decade, the magazine has been the career springboard for many volunteers.

4) ***The magazine manages a diverse volunteer community.*** These volunteers are separated by geography, and have varied skills and priorities. Some offer their time and labour because they believe in *Ricepaper*'s publishing mandate. Some come on board in the hopes of learning a new skill, or breaking into the publishing industry. Others volunteer because *Ricepaper* keeps them on the forefront of news and events in their social circle. Variations of this reciprocal relationship all contribute towards the *Ricepaper*'s volunteer community. In the words of *Ricepaper*'s Publisher, Jim Wong-Chu, "Volunteers are the life-blood of *Ricepaper*. We would be foolish not to take care and value the staff who spend their time and skills to help build this magazine. This is why any relationship we have with our staff must be mutually beneficial."[25]

2.4 The Magazine at a Glance

Ricepaper is a volunteer-run organization that is structured along the lines of corporate hierarchy. A product of parent organization the ACWW, *Ricepaper* is governed by ACWW's nine-person Board of Directors. A Publisher, Associate Publisher, Editor-in-Chief, Assistant Managing Editor and department heads oversee the daily operations of the magazine.[26]

[24] Ibid.

[25] Wong-Chu, Jim (Publisher, *Ricepaper*). Interview by author. Vancouver, BC, June 20, 2005.

[26] *Ricepaper* Magazine. "Canada Council Report on the Publishing Program."

The magazine relies heavily on the contributions of volunteer staff and interns to manage, develop, and deliver the projects and objectives of the publication. There are eight volunteer-run departments of varying sizes: editorial, art and production, circulation, advertising, marketing, finances, office administration, and information technology (IT) services. Figure 1 provides an organization chart that details the reporting relationship and communications between the various departments.

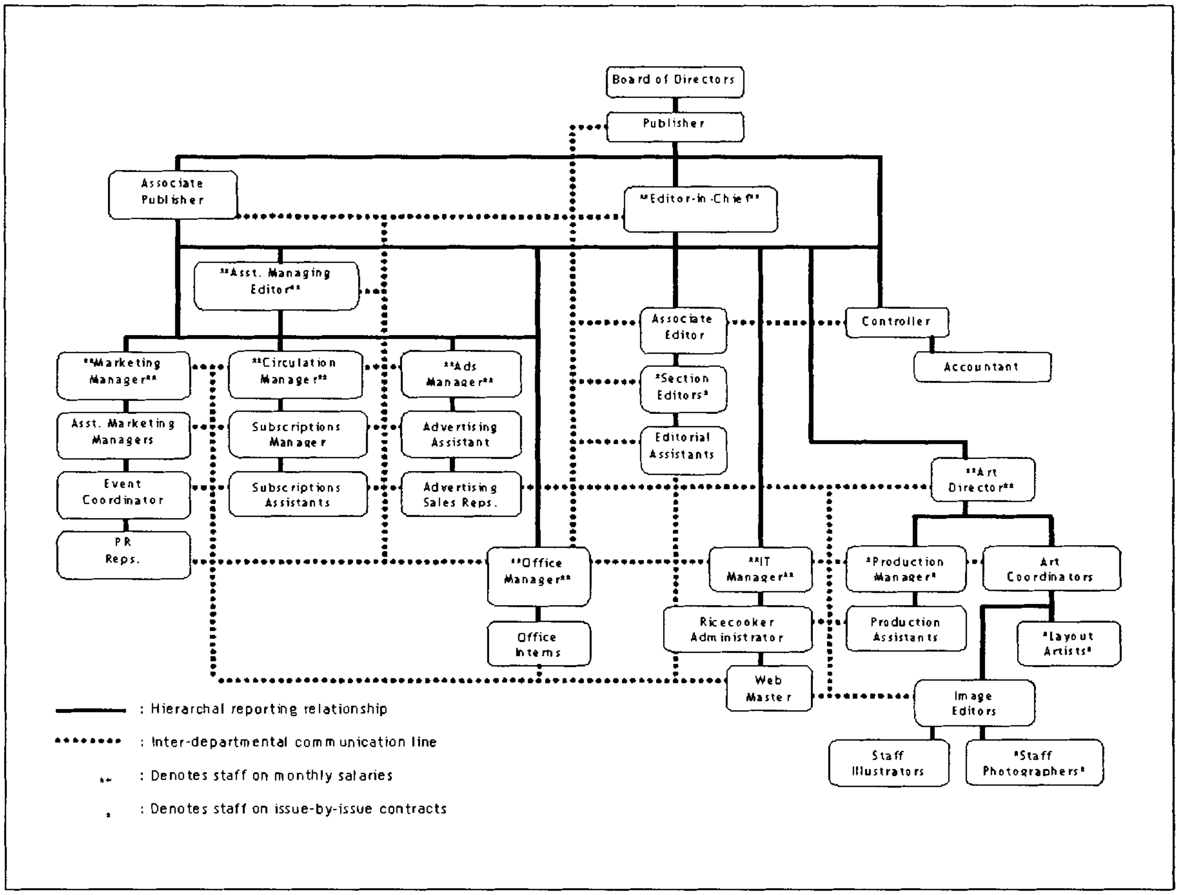


Figure 1. *Ricepaper* organisation chart, April 2005

The editorial, art and production departments are the two largest departments, with eighteen and sixteen volunteers respectively. The IT services department is one of the smallest, with only four volunteers. However, it is a significant player in the day-to-day workflow of *Ricepaper*. It is worth noting that total volunteer hours for the IT department roughly equals 35 hours a week, approximately equivalent to one full-time IT staff.

Volunteers are assigned to a primary department, or combined primary and secondary departments (for instance, editorial and production). The organization's hierarchy is organized according to two levels of management:

a) ***Core senior management***, which consists of the Publisher, Associate Publisher, Editor-in-Chief, and Assistant Managing Editor. This management level directs the creative vision of the magazine, solicits new grants and sponsors, and makes critical operating decisions.
b) ***Inter-departmental middle management***, which is the link between senior management and most volunteers. Each of the eight department heads manages the process and activities within their department, and supervises the growth and training of their own volunteers. They execute and forecast future directions for their individual departments based upon the vision set out by the senior management. They are also responsible for maintaining a consistent flow of interdepartmental communication within *Ricepaper*.

The above hierarchy clearly defines each position, and outlines its place in the magazine's chain-of-command. Sometimes, staff may temporarily occupy more than one of these positions. For instance, the Advertising Manager is also the Office Manager, as these are the two positions that require a volunteer to constantly be present at the office during business hours. The dotted lines in Figure 1 map out *Ricepaper*'s inter-departmental communication patterns. Formal and informal communications are exchanged between all departments, and at all levels—between senior management, department heads, and interns. Interaction between the staff does not rely on their physical proximity to one another, as most of the communication is exchanged via electronic discourse. It should be noted that sometimes, communication between departments may also be through an indirect channel, such as an assistant.

While *Ricepaper* has a hierarchical structure, this exists only to clarify the magazine's reporting lines and positions of authority. In reality, the magazine's daily operating culture is fairly "flat"—the staff regularly communicates with one another, updating people who are in junior positions, as well as managers in senior-level roles. The magazine's hierarchy is only formally enforced during important decision-making sessions, or when staff need to go up the chain-of-command to seek resolution for an ongoing dispute. For instance, debate over the merit of a particular article is brought to the Editor-in-Chief, who has veto power and final say over all content that is published in the magazine.

2.5 *Ricepaper*'s Management Style

Ralph Hancox describes the functional method used by many "traditional periodical publishing organisations" as being akin to "the assembly line method of production."[27] The functional method breaks down work into separate tasks that are assigned to individuals or groups. This can result in a conflict of interest between functional employees and inhibit communication.

[27] Ralph Hancox, *Topics in Publishing Management*, (Vancouver: Simon Fraser University, 2003), section 2.5.2.

Contrary to this traditional management style, *Ricepaper* employs a fluid process to manage its departments and processes. The magazine's "flat" operational culture encourages inter-departmental collaboration. As such, departments do not follow a rigid function-by-function protocol. Though the staff do adhere to formal protocols, they often perform tasks that might overlap inter-departmental boundaries. For instance, the Production Assistant who monitors ad booking and submission dates takes on responsibilities in both the production and advertising departments.

2.5.1 Inter-department al Fluidity

This fluid management style has made it possible for *Ricepaper* to avoid many of the pitfalls common to functionally managed publishers. As Hancox explains of such organisations: "editorial, graphic arts, and marketing territories: may be established and defended with the objective of 'sealing off' a function from unwelcome interference. Information is exchanged only on a 'need to know' basis."[28]

While *Ricepaper* has a large volunteer base, only one-third of its volunteers are active members. As such, many of its department heads and staff have to shoulder two or more hats. This creates overlap between the functions of each department, and minimises inter-departmental territorial competition. *Ricepaper* staff tend to work together collectively to meet their targets. For instance, if the advertising department might—at two weeks before the ad booking deadline—be short on advertisers, staff from the marketing and circulation team will volunteer to help pursue other ad sources. This inter-departmental collaboration is particularly helpful during critical production deadlines, staff absences and emergencies.

2.5.2 An Open Physical Layout

Representing this open management style, even the physical layout of *Ricepaper* approximates the way workflows from department to department. The magazine's office consists of a 400-square foot one-room office with four Pentium III PC desktops, arranged around the room. This humble set-up is the physical home to all eight departments, with IT, advertising, circulation, art and production being the primary users of the office. On a day-to-day basis, volunteers drop into the office and set themselves up any available computer or desk. They do not sit in cubicles, nor do they only sit in their respective departments. There is no physical delineation between departments, but this close proximity encourages the volunteers to share information.

[28] Ibid.

For instance, even circulation and production—two departments that do not interact on a constant basis—understand one another's functions.

The office can accommodate up to 15 people at any one time, though only a small proportion of the magazine's 60 volunteers—five to eight people—might use the space daily. Most of them chose to work and communicate by e-mail, or discuss their projects on the magazine's internal forum. In fact, they only visit the office during production crunch time. The workspace is then converted into a temporary "publishing sweatshop", with people working (and sleeping) in the office around the clock. Due to the limited office space, the staff hold their annual general meetings and larger events at venues outside the office.[29]

2.5.3 Managing a Volunteer-based Magazine

At *Ricepaper*, volunteers may come and go, or remain in the background, only contributing to projects every few months. Others may be consistent workers, and volunteer on a regular basis. *Ricepaper* has eight part-time salaried staff, 18 freelancers, 16 regular volunteers, and approximately 20 "backup" volunteers who contribute to the magazine on an issue-by-issue basis. People can move between departments, straddle responsibilities, and take on roles with greater or lesser responsibilities—according to their skills, commitment level, and availability.

To illustrate, *Ricepaper*'s previous Advertising Assistant conducted his sales calls during business hours. When he departed, the volunteer who replaced him was less available to perform these tasks during the day. As such, she delegated a portion of her calls to the Sales Representatives. This made her responsible for training them, and if necessary, recruiting qualified volunteers for the role. The responsibility of the Advertising Assistant—meeting a pre-determined advertising goal—did not change. Instead, the Advertising Assistant became responsible for finding new ways to meet the advertising targets.

Given the limited availability of its volunteers, it is not always possible for *Ricepaper* to place the best-qualified individuals in positions of responsibility. As such, it is important to administrate the magazine with some flexibility, and manage change organically. "Organic management" is a term specially coined by *Ricepaper*'s department heads. It refers to the magazine's directed, but also "hands-off" management style, where change is supervised and guided, but not forcibly manoeuvered in a certain direction. Volunteer mobility and organisational fluidity often means that the most effective process tends to be the one that they develop naturally amongst themselves. By carefully steering and monitoring these evolving processes, the management ensures that change is guided in a desirable direction, but operations are allowed to develop in accordance with the natural growth of the organisation.

[29] These events are usually held at a volunteer's residence, a boardroom in the local community centre, or at a restaurant.

2.5.4 Managing Change Organically

The fluid nature of *Ricepaper's* volunteer pool means that organisational change is constant, and organic. This is reflective of an observation made by De Bruijn et al. in their book, *Process Management*, "many organisations no longer lend themselves to hierarchical steering. Problems are so complex that the managers of many organisations lack the expertise and hands-on experience to make the right decisions alone."[30] As such, *Ricepaper's* management has to direct the process of decision-making intelligently. It becomes important for staff to anticipate difficulties, and develop solutions to meet these challenges. This management style allows, to some extent, processes and protocols to evolve organically.

In *Ricepaper*, rigid function-by-function procedures do not sit well with volunteers. Since people contribute to the magazine during their limited spare hours, they do not want to spend hours following a formal procedure set out by management. Rather, volunteers work best when some degree of flexibility and creativity is permitted—hence, organically developed processes tend to yield greater productivity. *Ricepaper* follows a process approach to change, which is based on fundamentals that are made explicit to volunteers beforehand. These are:

- Agreement about the decision-making process is made by the department heads and volunteers beforehand;
- The emphasis in these prior agreements isn't on the final product, but on how this change will be effected;
- These agreements leave each of the department heads and volunteers sufficient room to promote their own interests.

This process allows everyone involved to contribute their experience, needs, and evaluate problems collectively. Solutions are developed organically by this method and result in the refinement of a procedure, or substantive redesign of workflow. These decision-making methods let the magazine's management shape processes and customise workflows to best suit their staff, skills, and schedules. *Ricepaper's* Advertising Manager, Michelle Siu, explains the situation eloquently:

> With a volunteer organisation, you have to be creative, because you cannot just assume that you have someone to take care of something. A lot of the time, you have to find someone to do a task, because jobs don't always fall to the same person.[31]

[30] Hans de Bruijn, Ernst ten Heuvelhof, and Roel in 't Veld, *Process Management* (The Netherlands: Kluwer Academic Publishers, 2002), 5.

[31] Siu, Michelle (Advertising Manager and Office Manager, *Ricepaper*). Interview by author. Vancouver, BC, June 24, 2005.

2.6 Challenges of *Ricepaper*'s Management Style

Ricepaper's constant modification of processes can cause weaknesses in terms of workflow efficiency and organisation stability. The affect of these weaknesses are detailed below:

- ***In an organic process, fluid roles and responsibilities create unclear reporting lines.*** At *Ricepaper*, most volunteers, especially those in management roles, would consider themselves as part of more than one department. For instance, the Production Manager may also be a junior Ad Sales Representative. This creates project unaccountability, especially when the staff are uncertain of the boundaries drawn between their work and their roles.
- ***Interdepartmental overlap results in conflict, and lopsided interest representation.*** There have been times when "all the staff on the organisation chart could even be listed two or three times."[32] If a person holds posts in departments with conflicting goals, they might represent the interests of one unit over the other. Unequal representation can lead to low departmental moral, and the gradual disintegration of an existing unit. If this happens, the magazine becomes "lopsided", due to the non-existence of some departments.
- ***Uneven workload distribution results in staff burnout.*** The constant reassignment of volunteers to different roles can result in an unequal distribution of human resources, and especially overburden those who belong to smaller departments. Over an extended, this can lead to volunteer burnout, low morale, and the eventual departure of staff.
- ***Constant staff mobility makes it difficult for volunteers to take responsibility for projects.*** The constant mobility of volunteers between projects makes it hard to hold anyone accountable. Sometimes, "people picked up projects to avoid more important things."[33]
- ***Staff turnover results in the loss of critical operating knowledge.*** *Ricepaper* volunteers do not keep records of their ever-changing processes. As such, the departure or reassignment of volunteers means that important operating knowledge may be lost. This leaves remaining staff to reinvent the wheel, and sometimes, make old mistakes that could have been avoided.
- ***Processes change, according to the people in positions of power.*** The new managers might attempt to install their preferred processes in the department, executing an unnecessary top to bottom overhaul of previous processes.
- ***Organic processes spearhead change but create fluctuations in Ricepaper's organisational health.*** This has become evident, as the magazine has experienced troubles throughout its CMS development and testing periods. Some of these issues will be explored later.

[32] Choy, Hennie (Former Senior Editor, *Ricepaper*). Interview by author. Vancouver, BC, June 29, 2005.
[33] Ibid.

The above problems are weaknesses in *Ricepaper*'s management style. The next section outlines *Ricepaper*'s current workflow, focussing specifically on the editorial to production process.

2.7 A Look at *Ricepaper*'s Current Workflow[34]

Ricepaper follows a work cycle that divides the 52 weeks in a year into four 13-week schedules. The work cycle is a comprehensive list of most tasks involved in the standard running of a small quarterly, divided into logical departments. Inter-departmental tasks are mapped into overlapping "mini cycles", with the eight department heads leading the process. Department heads do not necessarily have to perform all these tasks, but in the end are entirely responsible for them. Week one in the work cycle begins after the last issue arrives in the office, with the process continuing week by week until the next issue is printed and shipped. A week-by-week break down of *Ricepaper*'s 13-week publishing cycle is detailed in Appendix A.

The production schedule of the magazine, often created eight months in advance of the magazine's actual publication date, is essential to the workflow at *Ricepaper*. The schedule is a critical part of the magazine's project workflow, because it indicates deadlines, as well as the critical intercepts for people who are directly involved in magazine production.

The production schedule begins with the closing date of editorial content; only after all editorial content is finalized can production of the magazine begin. The Art Director, designers, and editors may start on pre-production many months ahead of an issue deadline—conceptualizing and planning stories and photos. During the production period, the art department designs, the editors vet, the ads arrive, and the production coordinators collect and monitor all these processes. *Ricepaper* spends eight weeks in pre-production, and five weeks in production every work cycle.

Near completion of the production process, final proofs of the magazine are printed and circulated. Advertising ensure that all ads have been placed on the correct pages, circulation double-checks its distributor lists, editorial proofs for grammar and style, and art verifies image quality. By the end of this process, half-a-dozen or so staff members will have signed-off on the proofs, including a final authorization from the Editor-in-Chief. The Production Manager then guides the proofs through the next few stages, in which corrections are done, files sent to the printer, electronic proofs are received, and authorization is given for the printer to print the magazine.

At *Ricepaper*, most staff generally work from home, or on separate computer systems in the office. This makes it difficult for production staff to access the different MS Word and Excel

[34] "Current" refers to the workflow process at *Ricepaper* before the development of Ricecooker. It should be noted that following the development of Ricecooker, this workflow was often in flux, even though the primary stages outlined in this section continued to serve as the skeletal framework for the overall production process.

files as they progress between the various stages of editing, design, and ad sales. So often, the production assistants are left with two choices: 1) Spend their time hunting down the different documents; or 2) wait for the various versions to make their way into the correct e-mail in-box. Since the production assistants are responsible for tracking all the information that is listed in the Article Information Matrix,[35] it often becomes necessary for them to call up each editor or designer for updates. This can take days, even weeks, eating up production time. From beginning to end, this describes the tedious "search and wait" method that characterises much of the magazine's workflow prior to the development of Ricecooker. Appendix C provides a detailed look at the relationship between the editorial, art, and production departments.

It is also worth nothing that these processes do not take into consideration added complexities, such as editorial rewrites, the difficulty of reaching certain staff, or the submission of incorrect content files. Most importantly, it fails to establish a direct and constant communication line between the editorial, art and advertising departments. As the next section shows, this can often prove to be disastrous.

2.8 Weaknesses of Current Workflow

The inefficient tracking process detailed above works around the obstacles caused by constant staff mobility, *Ricepaper*'s independent computer systems, and each department's different working and communication methods. The magazine's management believed that content management technology was a solution that could replace these weaknesses caused by the current workflow:

- ***Multiple document versions make it difficult for production staff to track the progress of each article.*** Writers submit electronic documents to editors, who then print out the file and edit it *by hand.* More often than not, editors only send the electronic versions of their working files to Production, neglecting to transfer any changes that were made on the hard copies. This confuses the production assistants, as the document would appear as though it has not been edited.
- ***The variety of reporting methods used by every department creates extra work for production staff.*** Every department reports and checks-off its status update differently—on hard-copy forms, via e-mail, over the phone. The lack of a centralized reporting system makes creates extra work for production staff, who have to track down updates and consolidate all them into a single file.

[35] See Appendix B. Sample Article Information Matrix.

- ***Scattered paper and electronic files make it difficult for volunteers to locate information quickly at a moment's notice.*** This is especially problematic when the opportunity comes up for the magazine to trade a repackaged version of its content to a third party. Since the information cannot be located, the magazine has to pass up the opportunity to market itself.
- ***The duplication of hard and soft copy files creates extra post-production work.*** All paper and electronic files need to be sorted, and duplicates destroyed. This creates a confusing archival process *for the production, office management, and IT departments.*
- ***The lack of transparent communication makes it difficult for volunteers to understand their own role in relation to the rest of the departments.*** As a result, they tend to view production staff as a hindrance, and are less cooperative when asked for status updates.
- ***Information is incomplete.*** Editors do not always fill in their status information of the Article Information Matrix thoroughly. If production assistants are unable to get the data from the editor, articles sometimes go to layout with missing information.
- ***Multi-entry work increases the risk of error.*** The working line-up is updated by Section editors, the Editor-in-Chief, and then to production assistants, who distribute this document every week. This results in mistakes and typos, which can be a huge problem if author names and contact information is recorded incorrectly. A lot of time is wasting in tracking down information and comparing notes from one department with another.
- ***Each department can become overly-dependent on the working-line-up, and rely on it as a substitute to actual communication.*** During pre-production, the editorial, art, and advertising departments consult the working line-up to obtain updates on one another's progress. While the line-up provides sufficient information on basic article specifications and status, further details, such as a change in editorial direction, are not noted. This lazy communication between the art and editorial departments results in lengthy corrections towards the end of production, creating extra stress and work for all involved.
- ***Designers must wait for editors to complete their work before beginning their own.*** Even if illustrations or photography are completed early, layout cannot proceed because there is no text to input. When all the materials come through at once, the designers become overwhelmed with work.
- ***The process is slow.*** It can take many days between every department's sign-offs on the final layout proofs. The situation can also be exasperated if different department heads take the only set of proofs out of the office to work on from home. Separate proof sheets are sometimes lost, or reordered inaccurately.

Underlying the above issues is every department's limited knowledge of their own role in the overall production process. During production crunch-time, it creates a situation where the magazine is constantly in "crisis mode."[36] Production staff are constantly picking up the slack for other departments, leaving them less time to monitor the overall workflow.

When the senior management looked to possible solutions to solve management issues and improve the magazine's workflow, we decided a central information repository was crucial. The next chapter looks at the process that *Ricepaper* went through in deciding what was needed to order to fix the inefficiencies with the current workflow. It also describes the effort the volunteers put into conceptualising, developing and implementing Ricecooker 1.0.

[36] Wong-Chu, interview.

CHAPTER 3. The Catalyst for Change

As seen in Chapter 2, there are so many variables in magazine publishing that it is rare for the production process to go perfectly. *Ricepaper*, more often than not, encounters problems that delay production, and sometimes the magazine does not stay on schedule. Volunteers may get sick and drop out, and already overworked staff may step in to take on their work. Ads may arrive late, or clients may request changes that need to be accommodated in very little time. This creates problems, as magazines are shipped off to distributors late, resulting in a late release on the newsstands.

To illustrate, about three years ago, the sudden departure of *Ricepaper*'s Editor-in-Chief caused a large vacuum in the organization. Not only did this core person supervise all the product generating departments—editorial, art and production, he was also the only custodian of all production related knowledge. Documentation on production procedures had never been recorded during his time at *Ricepaper*. His departure resulted in a subsequent disintegration of the magazine's production workflow. Whole departments were left without direction, and tasks were performed midway without follow-up personnel to take them on to the next stage. This lack of guidance and supervision left many volunteers disoriented. The staff lost confidence in the magazine, and gradually departed. This created an even larger vacuum in the publication's editorial, art and production departments.

Ricepaper's repeated delays resulted in the loss of readers and advertisers—valuable revenue sources—as its reading community gradually lost faith in the magazine. Loyal readers, who used to visit newsstands regularly to check out the latest issue, stopped looking for the magazine every three months. Advertisers, whose ads only reached readers long after the initial appeal of their products, no longer saw the value of advertising with *Ricepaper*. This irregularity caused frustration to subscribers, and even led loyal customers to think that the magazine had closed down. As a reader wrote in a fit of annoyance to the magazine, "I guess you guys are no longer around, since I never see you on the newsstands anymore. But I'm still waiting for my subscription to come through the mail. If you aren't going to send it, I'll like my cheque back, please."[37]

How was it possible for the departure of the Editor-in-Chief—a person who had only served one year in that position—to wreak such chaos on the magazine? The answer lies in *Ricepaper*'s lack of a transparent communications and tracking procedure. Documents were scattered and often lost, making it difficult for the two Publishers to track the progress (or regression) of every department. Individual staff provided updates and reports separately, without cross-referencing their information with supporting documentation from other departments. On average, the staff spent more time trying to locate and verify data than actually performing their tasks. As well, this disorganised state of affairs also made it harder to pinpoint specific weaknesses in the workflow. Every working relationship was based on an assumption of trust—that all reported information was true, and that working knowledge was passed on, and not withheld. So long as a new issue came out every couple of months, it generally looked as though the magazine was doing well. This was the reason why *Ricepaper*'s Publishers did not notice these underlying issues early on. No one really knew just how messy the production process was, since all this information was kept by the one person had an overview of everything. When this person left, processes disintegrated further.

As the above illustrates, it is critical for a magazine to monitor all inter-department workflows, and ensure that production stays on schedule. Failure to do so makes the publication unaccountable to distributors and vendors, resulting in the loss of faith, readers, and advertisers. Over a prolonged period, this might even lead to the eventual closure of the magazine. More importantly, the workflow information and project status should be shared by all departments, and not contained in the memory of one individual. This applies particularly to *Ricepaper*, especially since the volunteer (and sometimes informal) nature of its organisation means that most operating details reside in the minds of a select number of active volunteers.

Shaken by the effects of the above incident, the magazine's Publisher, Board of Directors, and remaining staff strongly felt that the time had come to consider a different approach to production management. In hindsight, Wong-Chu notes:

> It wasn't evident to us then, but it was unwise to give one individual so much power and responsibility. We had to decentralise authority, increase transparency in our reporting, and move towards a structure-oriented communication network.[38]

[37] Chen Dong Cai, e-mail message to the editor, July 13, 2003.
[38] Wong-Chu, interview.

With these goals, the magazine's senior management steered the department heads towards finding a transparent solution that would manage production and content.

3.1 The Search for a Solution

It was obvious that *Ricepaper* needed to find a solution that could handle the changing needs and demands of the magazine's production and operations. However, it wasn't immediately evident that a content management system might be able to solve the magazine's problems. While the departments made lists of the shortfalls in *Ricepaper*'s production management, the editorial department began investigating a more holistic solution.

Former Senior Editor Hennie Choy observed the inefficiencies of the multiple document versions. She wanted to find solutions that would create greater control over document versions, and offer a structure for inter-department processes. As well, she saw that security measures were needed to protect data, as "people, even brand new volunteers, were being given very sensitive information. The stressful nature of *Ricepaper*'s volunteer positions meant that its volunteer turn-over could be pretty high."[39] Yet at the same time, volunteers needed to access documents without having to go through multiple third parties, or even having to come into the *Ricepaper* office. Many staff were "getting overworked", and Choy felt the urgency to find a solution that would be "a last push to try to fit *Ricepaper* into their lives."[40]

Meeting with the IT Manager, she mapped out *Ricepaper*'s need for a system that would serve as a central holding pen for all pre-production information. According to her, such a system would need to:

- ***Continue to facilitate transparent communications between departments.*** The system would function as a central knowledge exchange tool, detailing all past, ongoing and future projects. Volunteers from all departments would be able to log into the system to view projects of interest, sign up to work on them.
- ***Serve as a central content repository to manage the magazine's information goldmine.*** The system would be used to store and archive departmental documents and magazine content. Operating knowledge and task tips would be stored in this repository, and help preserve the information flow between current and future staff. This would minimise the loss of knowledge during staff transitions. As well, the central repository would establish an archive of current and new material for future commercial exploitation.
- ***Allow for version control.*** The system would store and track alternate versions of the same document in centrally, making it efficient for staff to access (and if necessary, repackage) a

[39] Choy, interview.
[40] Ibid.

particular document version at any particular time. For instance, a Marketing Assistant who needs a clipping from a specific article for a press kit need not contact the original editor. Instead, he would be able to obtain what he needs from the system.

- ***Provide a means for volunteers to maintain consistency in file formats, even while working from distant locales.*** The online central file repository would make it possible for volunteers to access and make changes to *Ricepaper* files from a remote location. Staff would no longer have to work off the only existing hardcopy of a document. All changes could be made directly in the electronic files, and saved on a central server. As well, this would decrease the time volunteers spend in commuting to *Ricepaper*, and give them more flexibility to fit the magazine into their own lives.
- ***Establish access boundaries to proprietary information.*** Only *Ricepaper* staff would be given access to the password-protected system. In the event of a staff member's departure, their individual staff passwords could be changed. Such a system would also help retain information within the organisation, instead of having it leave the magazine whenever its knowledge keeper departed.

Choy believed that *Ricepaper* needed a system that could solve many of the problems caused by the combined weaknesses in the magazine's management and workflow process. Ideally, such a system would store and manage the flow of all editorial content. In October 2002, Choy, along with then IT Manager Jerry Young, initiated the development of Ricecooker Version 1.0, a homebrewed solution for content management. Appendix D. provides a detailed timeline of *Ricepaper*'s entire CMS Development Process.

3.2 Ricecooker Version 1.0: The Concept

Like many projects at *Ricepaper*, the idea for Ricecooker's development was the result of one volunteer's concept that had evolved naturally, gained momentum, and eventually grown into a possible solution. While Choy and Young were the original perpetrators of the project, their idea had grown out of countless meetings and talks with staff from other departments.

In spite of this, Choy only decided to build a system that would primarily meet the needs of the editorial department. This decision was not a conscious choice to neglect the rest of the departments, but an attempt to address the magazine's problem at its root. Choy felt that most of the problems she witnessed stemmed from the improper management of editorial content: "I felt that if we had better control over our editorial processes, we would also have control over the rest of our workflow."[41]

[41] Ibid.

The initial idea for Ricecooker 1.0 was simple: a password-protected web site that would serve as an online repository of editorial information. Article files, editorial calendars, and status reports would be organised and stored under individual folders, which would contain all the different versions of the same file. With Ricecooker, editors would enter article updates directly into a working line-up stored on the site. By logging in, editors, designers, and production assistants would be able to download, at any point during production, the updated version for any article. As well, they would be able to check the status, editorial changes, and article specifications without going through various third party information keepers. As Choy put it, the concept behind the Ricecooker 1.0 site was "like FTP for Dummies."[42]

3.2.1 The Trial and Error Stage

Due to *Ricepaper*'s immediate need for a content management solution, Choy felt that it was important to build and implement an immediate solution to address these issues. As the research for possible solutions wore on, volunteer morale continued to sink further. Staff were continually overworked, and many projects were falling to the wayside. The pressure was on to find a quick-fix solution, and Choy felt that time was not a luxury that the magazine could afford. Due to this urgency, the development of Ricecooker 1.0 followed an idiosyncratic path.

Choy, acting as Project Facilitator, and Young, acting as Project Developer, led the process of developing Ricecooker and educating the staff in its use. As Project Leaders, Choy and Young enlisted the help of all departments during the pre-planning of Ricecooker 1.0. They did not intend to develop functions to meet every department's needs and wants immediately. Instead, they would build a centralised online content repository to address the organisation's most immediate need, and add future features and functions if new problems arose. In an attempt to cut down on development time, Choy and Young also decided to execute Ricecooker's development, usability testing, and implementation stages simultaneously.

This was a surprising and ambitious decision, considering Choy and Young were both inexperienced in developing such a project. In addition, the technical skills of their development team—volunteers from the magazine's small IT department—were limited to systems administration and basic web-building—HTML coding, JavaScript, and Dreamweaver. No one was familiar with PHP or MySQL—languages that could be used to program a more advanced content management system. In spite of these limitations, development, and shortly after, implementation began on Ricecooker 1.0.

[42] Ibid.

3.3 Basic Development and Implementation

Ricecooker 1.0 was primarily developed as a web interface for storing data. It consisted of a string of hand-coded HTML web pages, each representing a particular directory: Line-ups, Article Status, Issues. A password-protected home page linked to these directories, each of which listed sub-categories of folders. Ricecooker 1.0's interface was modelled after the GUI in a Windows browser, so that "volunteers would find the workspace familiar, and less threatening."[43] The site was designed to serve as a "24-hour virtual library" for the magazine, where volunteers would be able to download files, make changes, and upload a new version to the content repository. Figures 2 shows Ricecooker 1.0's data storage interface.

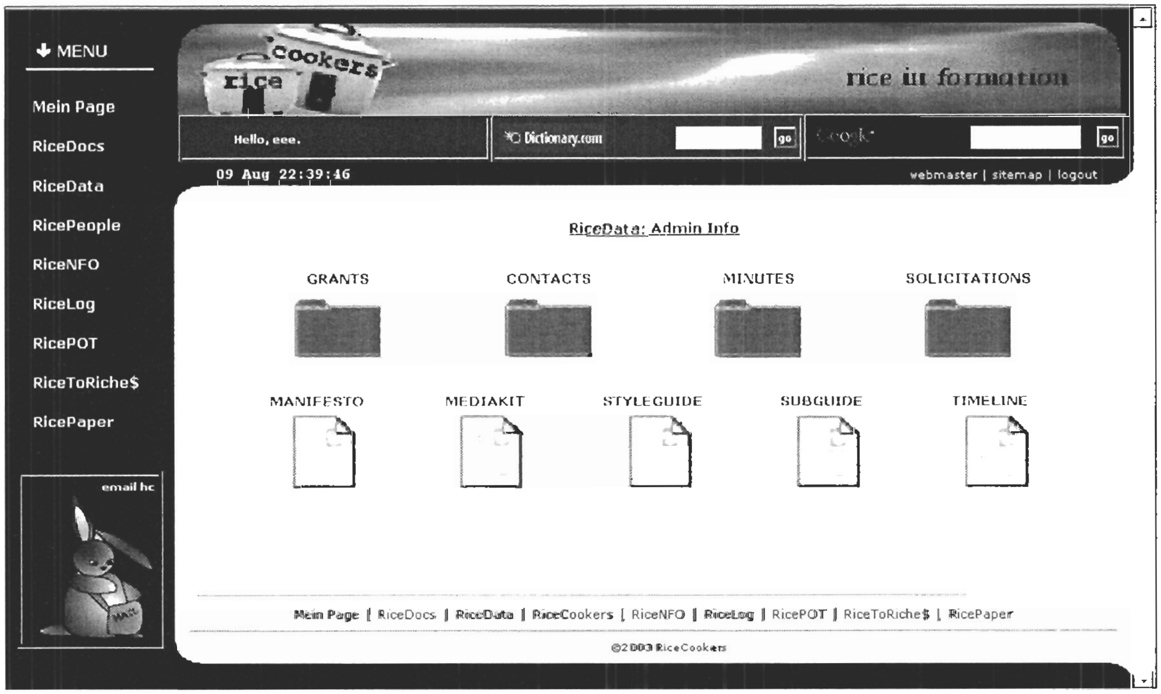


Figure 2. Ricecooker 1.0's Data Storage Interface.

However, Ricecooker 1.0 was hosted off Young's university web space, and had a limited capacity. As such, the volunteers could only use the site to store text, and not graphic files.[44]

A) Educating the Staff

Once the primary development on Ricecooker was complete, a "Users Committee" was created, originally consisting of 13 members—all the department heads, and half of the editors and

[43] Ibid.

[44] There was less of a need to store artwork at this point, as the magazine's text to image ratio was 70:30. The magazine was also using a text-centric template, where editorial took precedence over art and design.

designers. For three months they met twice a week for one-hour workshops on how to use Ricecooker 1.0, and to discuss matters related to the site. These meetings were learning sessions, though they gradually evolved to include usability testing. Staff would make note of site glitches, suggest improvements, and brainstorm ways to use Ricecooker.

The workshops followed the actual production cycle of the magazine. Stretched over a 13-week period, the volunteers got the opportunity to learn as they completed actual work—using Ricecooker 1.0 to access files, store data, and update each other on their weekly progress. The purpose was to get all the staff engaged in production-related work to understand the structure of Ricecooker, its functions, and the uses for this new tool. For the members of the Users Committee, these meetings were quite time-consuming, as learning to complete tasks with the new system now took twice as long. In fact, many volunteers did not want to learn how to use Ricecooker. Since the workshops matched the pace of the magazine's actual production, results were not always immediately apparent. Volunteers often pushed to revert to "the good old pen and paper route."[45] Some of them even became resentful of the meetings, for they had "joined the magazine to become editors and designers, not guinea pigs for some web site."[46]

B) Usability Testing and Staff Feedback

Another goal of the Project Leaders was to build a solid understanding of every production-related detail at *Ricepaper.* They asked the Users Committee to create job process definitions, which entailed providing specific breakdowns of their jobs. Every procedure in the editorial, advertising, art and production departments had to be recorded and numbered.

During workshop sessions, volunteers sat with members from their departments, and simulated production scenarios. The different departments were asked to "test" Ricecooker by running actual data through the site, just as they would during the magazine's production. Throughout this simulated process, staff were asked to match up their numbered tasks with the functions and directories depicted on Ricecooker's site map. For instance, an editor who oversaw final stage editing would match this task with a folder that contained final edits.

This was the first time the Project Leaders had tried to match the organisation's needs directly with the site they had built. At the end of these user feedback meetings, they saw the site they had designed did not entirely fit the volunteer's needs. However, they were hesitant to send Ricecooker 1.0 back to the drawing board. While the site was not ideal, the Project Leaders felt that it was important to show everyone that they were taking action to address *Ricepaper*'s

[45] Choy, interview.
[46] Ibid.

urgent need for a content management solution. Prolonging the implementation of Ricecooker 1.0, they felt, would cause staff morale to sink even lower. Based on this, they made a decision to push forward with the site. Instead of redesigning Ricecooker 1.0 entirely, they decided to address the site's shortfalls by adding features that could meet staff needs. The main challenge in Ricecooker 1.0's implementation, they felt, was not the faulty system. Rather, the challenge was in working with staff to modify the site to match the magazine's existing processes.

C) Implementation Issues and Solutions

Brainstorm sessions were held during the first month where staff discussed possible ways to modify Ricecooker to fit their needs. These sessions, which often followed the Ricecooker workshops, tended to address the latest frustration staff had experienced in trying to use the site. Consequently, most of the ideas that were conceived at these sessions tended to be "Band-Aid fixes"—solutions that could solve a temporary problem, but not necessarily address deep-rooted issues. Exasperating thc situation was the fact that all suggested solutions had to be programmable in HTML or JavaScript. Often, staff would suggest new gimmicks or features they had seen on other commercial web sites, and not judiciously researched content management solutions.

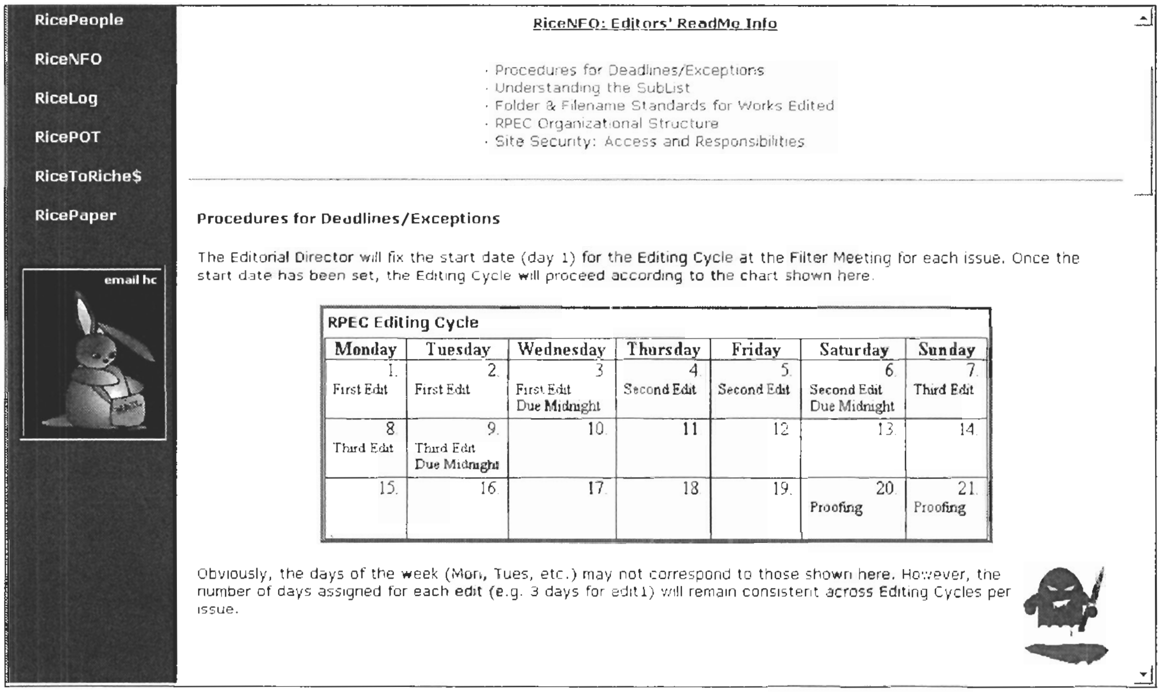
For instance, Ricecooker 1.0 was not specifically designed to track workflow status, but it had to be used for this purpose. Since the web site was designed as a static information repository, and not a database, it was not possible for users to search and pull status reports on the various workflows. In an attempt to remedy this, a Status Board was added to the home page. Restricted only to production staff, the Status Board consisted of a HTML form that allowed staff to input status reports, and post them on the main page. However, this new feature did little to improve the old production tracking system as production assistants still had to contact all editors and designers to obtain updates in order to compile the status report. What was needed was a function that automatically updated the status of each article every time an editor or designer made changes to the file. Ideally, this would establish greater transparency between inter-departmental status reporting.

In reaction to this, a scrolling message board was added to Ricecooker's home page. All staff were allowed to use this area to post status updates and make departmental announcements. However, the scrolling nature of the board meant that important messages were often bumped down as newer messages were posted. Announcements would be lost, unless the production assistants saw fit to repost the announcements on the Status Board.

These random and poorly conceived additions resulted in a unorganised site with functions that did not inter-connect with one another. At the peak of its usage, Ricecooker had polling functions, feedback forms, message boards, chat boards, an entertainment area, and 50 directories of vital data, including banked submissions, query sheets, contracts, staff and author profiles. As illustrated above, these "functions" would sometimes create greater obstacles than the ones they set out to fix. In fact, many of the staff who were engaged in production activities did not know how to use these features, or had no idea that they existed.

3.4 Cooking with the Cooker

Many of the above issues, while apparent with hindsight, were not evident to the Project Leaders and the general staff at first. In was only after the fourth month of implementation that Project Leaders gradually became aware of the inherent limitations of Ricecooker. However, they were also encouraged by the positive feedback from the staff who did use the site to accomplish their tasks. Editorial and production calendars were posted online, increasing accessibility to the documents. All deadline changes were immediately updated on the site, eliminating the need to mail-out copies of the calendar whenever timelines were adjusted.



RiceNFO: Editors' ReadMe Info

· Procedures for Deadlines/Exceptions
· Understanding the SubList
· Folder & Filename Standards for Works Edited
· RPEC Organizational Structure
· Site Security: Access and Responsibilities

Procedures for Deadlines/Exceptions

The Editorial Director will fix the start date (day 1) for the Editing Cycle at the Filter Meeting for each issue. Once the start date has been set, the Editing Cycle will proceed according to the chart shown here.

RPEC Editing Cycle

Monday	Tuesday	Wednesday	Thursday	Friday	Saturday	Sunday
1. First Edit	2. First Edit	3 First Edit Due Midnight	4. Second Edit	5. Second Edit	6. Second Edit Due Midnight	7. Third Edit
8. Third Edit	9. Third Edit Due Midnight	10.	11	12.	13.	14.
15.	16.	17.	18.	19.	20. Proofing	21. Proofing

Obviously, the days of the week (Mon, Tues, etc.) may not correspond to those shown here. However, the number of days assigned for each edit (e.g. 3 days for edit1) will remain consistent across Editing Cycles per issue.

Figure 3. Sample Ricecooker 1.0 Editorial Calendar.

By viewing the changes to each document file on the site, production assistants were able to assess the editorial progress for each article. These updates were then compiled into a status report, which was posted on the web site.

The staff also used the various message boards on the system to post announcements and alerts throughout the production cycle. For instance, the Production Manager might write something like this: "designers working on the layout for pages 46, 47, and 52 should note that the content for these sections is still not ready. Please commence work on pages 35 to 39 first".

Not surprisingly, the editorial, art and production departments were the ones to use Ricecooker 1.0 the most, since the site had been designed specifically for their processes. Each Section editor could access articles on a rolling basis, instead of waiting for the previous person in the workflow to pass the file to them. As well, designers could log online and print-off the outline or draft of an article well before the piece was submitted for layout. This gave everyone additional time to plan and conceptualise content ideas.

In an attempt to entice volunteers to use Ricecooker regularly, the Project Leaders created an "entertainment area" on the home page. The department heads posted cartoons, riddles, and trivia here, thus enticing volunteers to use the site for fun as well as for work. Volunteers would use the entertainment area to exchange personal notes, advertise the opening of their new art show, or post an announcement for their upcoming garage sale. Once volunteers realised they could use Ricecooker to reach and mingle with a larger community, they started checking the site daily. During Ricecooker 1.0's first production cycle, only seven staff attempted to use the system. Eight weeks later, there were 16 active registered users—approximately 67% of the magazine's volunteers.[47] On average, about half of the actual production work was accomplished online. Tasks such as status reports, inter-departmental communication, document filing and archival, were all performed on Ricecooker 1.0. However, it should be noted that most of this was limited to the editorial and production departments.

Ricecooker 1.0 had limited web space. Given that most art files were large, it was not possible to store and track multiple design versions online. As such, the art staff mainly used the web site to share thumbnails of their design concepts with the editors. A mix of hardcopy and electronic formats still had to be used for storing and circulating design files—discs, portable media drives, e-mail, filing cabinets, portfolio holders—a disorganised system that often created much confusion. There was no official cataloguing system that differentiated between unsolicited submissions and commissioned artwork. Sometimes designers might use the wrong image for a layout. An unknowing intern could accidentally delete an entire bank of art scans or mistakenly discard a mixed-up pile of photographs. The art and production staff still had to search for scattered image files most of the time, and then manually match these to their textual counterparts. This was a frustrating process in itself.

3.5 Cooking with the Cooker: Some Benefits

The implementation and subsequent modifications of Ricecooker 1.0 had many attached issues. Even though the site was not a "cure-all" for *Ricepaper*'s workflow problems, it did:

[47] The magazine had approximately 24 volunteers left after the departure of the previous Editor-in-Chief, as compared to their original staff of 49. It would be another year before the magazine was able to rebuild its human resources. Currently, there are 60 staff at the magazine.

- ***Function as a central repository for most pre- and post-production text files.*** While a seemingly insignificant step, this was the change that made it possible for the magazine to merge its disparate electronic and paper trails, standardize files, and help push the organisation towards greater self-sustainability. Operating knowledge and department logs could be kept on the server, thus minimising knowledge loss during staff transitions.
- ***Encourage the beginnings of an on-line Ricepaper community.*** The Project Leaders designed a personalised "ricecooker icon" and profile page for every active volunteer. While these developments might seem trivial, it was an important and creative step towards using Ricecooker to integrate the volunteer community. This gave the volunteers a glimpse into how an online *Ricepaper* community could extend the magazine's sphere of influence. They realised that the site could be used to build a remote community for new and existing staff, thereby fostering a sense of belonging for those situated in distant locations.
- ***Serve as a forum for transparent communications and document the magazine's first textual discourse.*** After the volunteers had used Ricecooker 1.0 for a month, the Project Leaders added two chat boards—one for art and production and another for editorial—to the site's home page. The intention was to give staff the opportunity to discuss their concerns frankly with everyone else. The rule was, if someone had a criticism, they had to post a constructive suggestion to improve the problem. Since every discussion was logged, staff were more hesitant about criticising for the sake of criticism.
- ***Minimise circuitous file requests.*** To remedy this, a chart was posted on Ricecooker, detailing all task assignments according to departments and chain of command. Instead of randomly asking (an annoying) people, volunteers would know who to approach directly, making it "worthwhile to have the website, as it meant not bothering someone."[48]
- ***Motivate staff by holding them accountable to their duties.*** Status Update Logs were posted on Ricecooker 1.0, creating greater transparency in workflow reporting. Unfulfilled duties now became much more noticeable than they were before. As departments were encouraged to conduct "balances and checks" with each other, volunteers were now accountable, not only to their department heads and the Production Manager, but to everyone else in the workflow. This open accountability kept staff on their toes, as no one wanted to be the person to stall processes.
- ***Provide greater working flexibility for volunteers.*** Ricecooker made the magazine's data readily accessible 24 hours a day, seven days a week. This decreased the time volunteers spent in commuting to the office, and gave them the option of tailoring their volunteer hours

[48] Choy, interview.

according to their personal schedules. This minimised stress levels, and improved the magazine's work environment.

- ***Serve as a reference library.*** New volunteers could learn a task by downloading someone's completed work, and studying how the previous person approached the task. This was particularly useful for the editorial department, who later started storing "gold standard" articles on the site for learning and reference purposes.
- ***Establish access boundaries and protect confidential information.*** Only *Ricepaper* staff were given access to the password-protected site, which stored internal files like contributor data, author solicitations, meeting minutes, and grant applications.
- ***Give volunteers insight to the whole magazine and its operations.*** Ricecooker 1.0 gave staff the option of viewing data for other projects within the magazine. The site map—a "snap-shot" overview of various directories—gave volunteers an overview of the activities and tasks that contributed to the operations of *Ricepaper*.

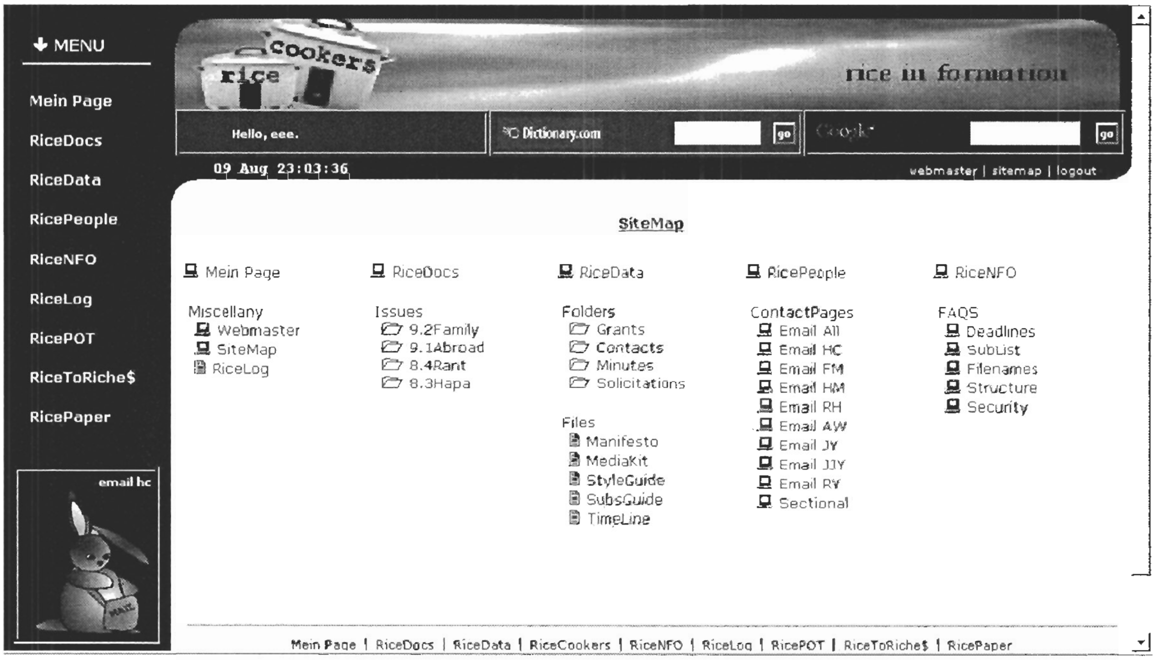


Figure 4. Ricecooker 1.0 Site Map.

- ***Inspire other departments to reorganise and streamline their workflows.*** Ricecooker had originally been designed to meet editorial, art and production needs. Staff from the advertising department occasionally used the site to update their ad status, but the site was not built to handle all of their workflows. The Project Leaders had not wanted to integrate all departments into Ricecooker 1.0, as they feared that the site would become a "data dumping ground and organisational nightmare."[49] Instead, they provided the Ricecooker template to other departments, and encouraged them to build their own web sites. If every department had such a site, they reasoned, managers could simply log into each site to obtain updates for all of the magazine's operations. Eventually, the marketing department started their own site, modelling it after Ricecooker 1.0. While the site only served the needs of the marketing staff, other department heads were granted access to the site's chat boards and message centre.

On the surface, Ricecooker 1.0 seemed to bring about several unanticipated benefits. It appeared to meet most of the original goals set out by Choy: 1) Serving as a central content repository; 2) facilitating transparent inter-departmental communication; 3) maintaining consistency in working and archival files; 4) providing volunteers with working flexibility; 5) allowing for greater tracking and version control; 6) establishing access boundaries and protecting proprietary information. These positive developments marked the start of many short and long-term benefits for the magazine. However, Ricecooker 1.0's unstructured development and disorganised implementation meant that the site had many underlying issues. These would ultimately complicate the magazine's workflow and upset its established procedures, leading staff to consider a massive design overhaul for Ricecooker 1.0.

3.6 Cooking with the Cooker: Limitations and Drawbacks

Ricecooker 1.0 was a relatively robust website. All of its functions worked, and the site was relatively bug-free. Rather, the system's main problem was its inherent design limitations. This was not surprising, especially since Ricecooker 1.0 was not designed to handle the type of work that the staff needed it to do. The Project Leaders had believed that by partially integrating the magazine's departments—editorial, advertising, art and production—communication and information sharing would be greatly improved. In turn, they hoped this would increase workflow efficiency and eliminate the duplication of tasks. However, the web site's limitations proved to be a significant drawback:

- ***Ricecooker was not a production-tracking tool.*** This was the site's biggest limitation. It was thought that consolidating all status updates online would make it significantly easier to track workflow status. However, no thought had been given to *how* these status reports would be updated or compiled. As a content management solution, Ricecooker 1.0 actually did little to "manage" content or workflow. Instead, production staff had to contact all the involved staff to obtain updates, and then manually enter this data into the line-up and Article Matrixes.

[49] Ibid.

- ***Duplications in tasks still existed, even though all the data was now available on one server.*** Since the many functions in Ricecooker 1.0 were not linked to each other, new data that was entered on one section of the site did not mean that everything that referenced this data would be automatically updated. Editors might fill out the line-up, but forget to update the Article Matrixes. Production staff would then have to input the data on their behalf.
- ***Ricecooker did not automatically inform its users when a new file was updated.*** If a file was uploaded prior to its due date, the next person in the work cycle often would not know the file was accessible to them, unless they happened to check the site ahead of time. Ricecooker 1.0 could not be set to notify users that their files were ready. Instead, the production staff would have to manually post a reminder, or send a mass e-mail to everyone.
- ***The site had no "check-in, check out" function to track the user currently working on a document.*** As such, it was difficult to know if another staff had merely downloaded the file to input a quick change, or if the previous person in the workflow was still working on the file. Two people could download the same file and work on it simultaneously, and then upload two separate sets of changes, completely disregarding file versions and production protocol.
- ***Though based on the Windows Explorer GUI, Ricecooker 1.0 did not have an intuitive interface.*** The workshop sessions did little to prepare them for the challenge of actually working with the site. Ricecooker's directories were organised according to production stages, but it was not easy for users to find relevant files, because documents would be moved to different directories as the files progressed through the workflow.
- ***There was no user manual for the site.*** Since the Project Leaders had tried to combine the usability testing and implementation periods, it had not been possible to write a manual for a web site with constantly changing features. Instead, when users ran into problems, they would ask another fellow volunteer, which would slow productivity even more because two people were taken away from their tasks to figure out a software problem.
- ***Ricecooker 1.0's storage space was limited.*** Since *Ricepaper* planned its editorial four issues ahead, the volunteers had to take past issues offline after production, and archive them to CD.
- ***The site did not have a user-oriented administration back-end.*** Since most department heads did not have sufficient technical knowledge, either Young, or one of the IT staff had to make the major changes on Ricecooker 1.0. For instance, users could not delete their announcements and messages from the site, as these had to be manually removed from the site's code. Everything had to go through the IT staff, who were bombarded with requests to remove postings and amend announcements.
- ***Since Ricecooker 1.0 was designed for the use of three departments, this created unbalanced workflow tracking at the magazine.*** The remaining departments—marketing, circulation, office management, finance, IT—continued to use a combination of paper and electronic files

to do their work. In their eyes, "if the old way still worked for other departments, then it should still be good enough for editorial and production". Unsurprisingly, this created much resentment between the departments. In the end, the Project Leaders and Users Committee realised that it would be beneficial to standardise all inter-departmental workflows, and link all departments together under Ricecooker.

3.7 Summary: An Evaluation of Ricecooker 1.0

Ricepaper's existing workflow had weaknesses that were caused by a number of problems: Separate computer systems, scattered documents, duplication of tasks, and issues with production tracking. Ricecooker 1.0 improved the situation in many ways, but mainly because the volunteers tried to work around the site's limitations. They modified workflow processes to compensate for the site's shortfalls, sometimes even becoming "salves to the system."

Ricecooker 1.0 was a convenient web site that could store data, publish information, and allow staff to share knowledge. However, it was not a production-tracking tool. As such, users could only use it in a limited manner to "manage" content and workflow. While the site was passable as a temporary solution, it had no room for future growth. In fact, Ricecooker 1.0 had already been modified to the maximum and it still could not meet the magazine's needs. In many ways the site was just a centralised substitute for the old way of doing things. In the end that was just it: Ricecooker 1.0 had been conceived as a web-based storehouse, and in truth, it only served that function. No number of new features could make up for the site's inherent limitations—a consequence of poor planning, undisciplined development, and no usability testing. The staff's refusal to acknowledge this early in the implementation process only exasperated the magazine's existing workflow problems. As mistakes continued, processes were further delayed and new obstacles were created. These mistakes and obstacles often generated extra work as staff had to solve the new problems that cropped up.

Eventually in March 2003, the magazine staff realised that Ricecooker 1.0's drawbacks would always be a hindrance to the magazine's production operations. Just five months after launching Ricecooker 1.0, the Users Committee voted for a complete overhaul of the site. They also recommended redeveloping the site into a more sophisticated and robust content management system. Ricecooker 2.0 had to be a pre-meditated project involving proper planning by skilled developers.

CHAPTER 4. Finding Structure Within Chaos

As described in the previous chapter, Ricecooker 1.0's limitations left no room for future extendibility, and its many shortfalls necessitated yet another system overhaul. This is not to say that the web site failed to leave any enduring positive impact on the magazine and its staff. Ricecooker 1.0 established the beginnings for the magazine's first online community—a development that later proved to have great impact on the organisation's culture and operations, as Chapter 5 explores in greater detail. Right from the beginning, Ricecooker 1.0 subtly transformed the organisation's culture, enlightening the staff to the importance of inter-departmental transparency, the luxuries of telecommuting, and the benefits of a centralised content exchange system.

With these benefits in mind, the magazine's management began casting about for replacement options. They already knew that it would be necessary to build a completely new system—Ricecooker 1.0 was a structurally flawed web site, not a content management system. As staff continued to struggle with Ricecooker 1.0 it became obvious that the web site only served one aspect of the magazine's needs—that of a central content repository, a function that might just as well be fulfilled with any basic FTP freeware. While the Project Leaders did not officially take Ricecooker 1.0 offline, the web site's demise was inevitable—a natural process that occurred as staff gradually stopped expecting Ricecooker 1.0 to be a tracking tool. As the magazine's demand for archival space grew, Ricecooker 1.0's limited storage capacity became a more pressing issue. The time had come to replace Ricecooker 1.0 completely.

4.1 Redeveloping Ricecooker

In early April 2003, I was appointed as *Ricepaper*'s new Editor-in-Chief. By that point the magazine had reached a critical point in its production—what management had once considered a workable temporary solution was slowly turning into a workflow headache. Files were mixed-up, versions lost, and days of work would sometimes be deleted due to someone's oversight. Message boards were crammed with announcements that were often buried further down, even before anyone read them. Eventually, staff stopped using the message and status boards, reverting instead to e-mail communication. During this interim period, an excel spreadsheet—the only means used to track workflow status—was circulated via e-mail and updated by staff.

Despite this, the senior management and department heads were unwilling to rush the staff into redeveloping Ricecooker. This time the magazine wanted a more sophisticated piece of software: Ricecooker 2.0 had to be a custom-designed CMS, carefully planned and developed by qualified personnel. Nevertheless, all of us had already put a lot of thought into conceptualising Ricecooker 1.0—work that now served as an invaluable redevelopment resource. However, *Ricepaper* still had a tall order to fill, especially given the limited financial and technical resources. While the magazine did have an IT team with three staff, only Jonathan Lin, the IT Manager, had some experience with CMS development.

During my previous term as the magazine's Executive Editor, I had personally been a part of the original Users Committee, participating in the magazine's development of Ricecooker 1.0. Due to my new senior managerial commitments, I was no longer on the Committee, so my role in the development of Ricecooker 2.0 was quite different. Instead of being part of the usability testing process, I now took charge of the administrative logistics for the project—soliciting vendor quotations, liaising with potential development partners, and ensuring that the project did not stall the regular production of the magazine.

The primary concern for the redevelopment project was that it had to be accomplished at little cost. Commercial companies specialising in content management software had quoted the magazine a range of prices: US$25,000 for basic software, US$60,000 for a semi-customised system, or US$120,000 for a completely customised CMS, including implementation and technical support.[50] Even preliminary estimates from local developers ranged between $12,000 and $35,000 for translating Ricecooker 1.0 into a workable CMS.

The departure of the previous Editor-in-Chief had dealt a serious financial blow to *Ricepaper*. The magazine, still recovering from its loss of advertising and subscription revenue, was not in a position to allocate a huge budget for development costs. It was evident that *Ricepaper* would not be able to hire a team of CMS developers on a commercial contract. Rather, the best option was to seek out trained, but less experienced, development partners such as students in educational or vocational training programmes who would be willing to work on the project in return for non-monetary benefits.

[50] Prices are based on estimated telephone quotations to *Ricepaper* from a variety of CMS vendors in June 2003.

4.1.1 Request for Development Partners (RFDP)

The vast range of project quotations indicated that every developer and vendor had interpreted *Ricepaper's* CMS needs differently. It would be necessary for the magazine to define and streamline the list of requirements. The first step was for the IT staff to meet with the Users Committee—department heads and mid-level staff who were already familiar with the needs of each department and had a clear picture of what a system like Ricecooker had to do for the magazine.

The members of the Users Committee were asked to consider their specific tasks and duties, and from these, create a list of needs and wants for each of their departments. Details for every process, along with ideas of how these processes could best be managed, were mapped out and used to create a list of possible functions for Ricecooker 2.0. These details were used to create a request for development partners (RFDP), a formal document that was directed to Vancouver-based schools and institutions with Computer Science and Software Engineering programs. By sending an RFDP to these schools and institutions, *Ricepaper* was indicating that they were interested in offering its CMS project as a learning and development opportunity to students in these programs. The magazine also hoped that programs would respond with a list of their applicable courses, and demonstrate interest in collaborating with the magazine to develop Ricecooker 2.0. The RFDP was also sent to universities, specifically targeting schools with directed studies courses.

The Users Committee led the requirements gathering process for the RFDP, which was a significant undertaking for *Ricepaper*. Staff were invited to submit their needs and wants, and encouraged to get involved in the planning of the system. For the first time people gained insight into the potential of the magazine, seeing how labour and time were wasted as untapped resources. With this new knowledge, the Users Committee was able to visualise an "ideal" CMS that would eliminate unnecessary functions and increase workflow productivity. The editorial department had the longest list of requirements on the RFDP. They listed about thirty functions that they needed, including article tracking, matching contributors with their subscription renewals, printing Article Matrix reports, and a rights database. The advertising sales department wanted a software system that could track advertisers and agencies, create printable and electronic contracts, and serve as a contact management system. The production department's list of requirements focused mainly on job costing, project tracking and printing specifications. The initial check sheet was a 54-page document. It listed in detail the varied requirements that *Ricepaper* envisioned in an ideal CMS, organised by units and cross-referenced by interdepartmental processes. The list was so extensive that the project would have easily gone close to six figures if it had been a commercial contract.

The department heads on the Users Committee examined the detailed list of needs and wants and tried to pare it down into a list of requirements for the RFDP. It soon became evident that the proposed CMS was based on a combination of: 1) actual processes used by the departments; and 2) ideal processes that were based on how staff thought their department should operate. Most of the editorial, art and production workflows, for instance, were simply mapped out according to the processes they followed while Ricecooker 1.0 was still in use. However, for the departments that had never been integrated into Ricecooker 1.0—marketing, circulation, finance, office management and HR—workflows had to be visualised, and made to "fit" the staff's idea of an ideal CMS system. Some of the daily procedures between the marketing and production department, for instance, were based on paper and phone communication, and still did not utilise electronic document exchange. The challenge was for staff to conceptualise how this form of communication would fit into the overall electronic workflow, and translate this into functions that could be programmed in Ricecooker 2.0.

The RFDP was finished in July 2005, and was sent out to twenty instructors and professors who taught computer science and software development at post-secondary institutions and universities. The final RFDP included a six-page CMS check sheet[51] that the Users Committee had compiled for Ricecooker 2.0, which listed the five main functions that *Ricepaper* wanted in its new CMS. It was a triumphant moment for the staff at *Ricepaper*, as the project was a cumulative effort from a number of staff from across several departments. The next step for *Ricepaper* was to wait for responses and to consider the development partners they had for designing Ricecooker 2.0.

The results did not quite meet expectations: only one school responded. Jim Wong-Chu, *Ricepaper*'s Publisher, speculated that the reason many schools did not respond to the RFDP was because they simply could not provide the kind of partnership that *Ricepaper* was looking for; the magazine had been *too specific*, and demanded a professional-level development partnership from educational institutions.[52] When *Ricepaper* asked the instructors why they did not respond to the RFDP, one instructor commented, "the requirements your magazine requested are significantly more sophisticated than the software that our students are asked to develop."[53] Another university professor said, "your project demands implementation support from our students. This is something beyond the scope of what our course can provide."[54] Even instructors who had worked on CMS development projects with other organisations could not meet *Ricepaper*'s specific needs. One instructor even suggested that as a magazine, *Ricepaper*'s software needs were "too different from the CMS requirements of other industries."[55]

[51] See Appendix E. Ricecooker 2.0 Needs and Wants Check Sheet

[52] Wong-Chu, interview.

[53] Developers' Responses to the RFDP, a collection of e-mail messages to the author. August 10, 2003.

[54] Ibid.

[55] Ibid.

Many of the instructors taught software development courses that required students to develop an end-of-term software project, but most of these courses preferred their students to work on projects for hypothetical organisations. Other instructors encouraged their students to engage in development projects with actual non-profit organisations, but were daunted by *Ricepaper*'s extensive list of development, implementation and post-implementation support demands. As a publishing organisation, *Ricepaper* had specific requirements, including the ability to handle subscription reminders, batch invoices, sales commissions, as well as other magazine-specific production practices. Jim Wong-Chu admits that early on, he knew from some of the school's course descriptions that they would not be able to meet all of *Ricepaper*'s requests. Nevertheless, he had hoped to receive more responses to the magazine's RFDP.[56]

After months of preparation in accumulating knowledge of the magazine's workflow, researching and creating a RFDP, the only school that was willing to enter a development partnership with *Ricepaper* was Vancouver Film School (VFS).

4.1.2 No-Cost Development: Vancouver Film School (VFS)

VFS, an institute that strives to be "the world's pre-eminent centre for both training and higher learning in all areas related to media and entertainment production,"[57] initially seemed like an unlikely fit with *Ricepaper*'s CMS project. Surprisingly, a Senior Instructor at VFS contacted me in August 2003. He was looking for potential non-profit organisations to participate as "clients" for an interactive media class he was teaching in September 2003. *Ricepaper*'s CMS project provided the perfect learning opportunity for his students to "use interactive media to build an environment that focuses on the people who are involved in the creation of the magazine and to create a great experience for those who read it."[58] He also saw *Ricepaper*'s CMS development as an opportunity for his students to exercise their marketing skills and multi-media design talent. After further discussion, both parties finally agreed on a compromise that would meet both *Ricepaper*'s CMS needs, and the learning needs of the VFS students. Instead of one project, the students would engage in two related projects: Project 1) A revamp of *Ricepaper*online.com, *Ricepaper*'s public web site; and Project 2) A content management and production workflow tool, otherwise known as Ricecooker 2.0.[59]

The fact that only one institution responded to the RFDP, and of that, compromises had to be made on the initial project, disappointed some people on the Users Committee. Committee members felt that all of their hard work in coming up with the RFDP should have elicited more

[56] Wong-Chu, interview.
[57] Vancouver Film School, About VFS: Vision statement [online], 2005.
[58] Users Committee, Notes from VFS Client Group Project Meeting. Vancouver, September 25, 2004.
[59] Since this report is on the development of the Ricecooker CMS, it will mainly focus on Project 2.

responses. Nonetheless, it was a consensus to proceed with VFS, as "it was the only partner willing to work with *Ricepaper*'s requests, and at no financial cost to the magazine." [60] Both parties made an agreement—the development on Ricecooker 2.0 would start on September 1st, 2003, and the new CMS would go-live on December 30th, 2003—a four month development, implementation and training period.

4.1.3 Refining Requirements and Expectations

Throughout September 2003, the VFS students conducted interviews with every *Ricepaper* department head, and staff from various levels of management. Based on these interviews and the initial RFDP, the students developed a set of internal and external project objectives:

Internal Objectives

- Increase efficiency production tracking;
- Improve the timeliness in the creation of the magazine;
- Improve quality by increasing production efficiency and creating more time for staff to implement improvements on content;
- Overall: Maximise the magazine's limited resources.

External Objectives

- Use the magazine's web presence to increase circulation;
- Build a stronger online community presence;
- Automate the submissions process;
- Use the internet's viral nature to market to a wider audience.[61]

Based on these objectives, the original RFDP requirements were refined, and presented to the *Ricepaper* User's Committee on September 25, 2003. Two departments—finance, office management and HR—would not be integrated into the CMS. Upon further discussion with VFS, *Ricepaper*'s management had felt that it was best *not to* store the confidential data from these departments on the centralised Ricecooker CMS. Instead, data from these departments would be converted into electronic format, and stored on a separate restricted server at the *Ricepaper* office.

The final requirements focussed on delivering a CMS system that would cater mainly to the magazine's editorial, art and production departments, with supporting features for the advertising, circulation, marketing and IT departments. Ricecooker 2.0 would be developed as an "internal tool (intranet/extranet) to simplify the following phases of the magazine

[60] Users Committee, Notes from VFS Client Group Project Meeting.
[61] Ibid.

construction: 1) editorial submissions and proposals, 2) editing, 3) production tracking."[62] Linked directly to the *Ricepaper* website, the CMS would also tie back end deployment with front end content delivery. The basis for Ricecooker 2.0 would be a centralised database, consisting of issues, articles, subscribers, event listings, sales contacts, and the magazine's PR list serve. In addition, the system would consist of:

- A knowledge base to manage flat data files;
- A scheduler;
- Production and project calendars;
- Data input templates;
- Report output templates;
- A publishing mechanism to port various forms and platforms (e.g. .PDF, web etc.);
- An administration section to manage user accounts and access privileges; and
- A media assets management section (e.g. photo gallery).

It should be noted that during the planning stage for Ricecooker 2.0, *Ricepaper's* Users Committee placed less emphasis on building communication tools into the CMS. It was thought that staff would be kept up to date by logging into the system and viewing project progress and comments from other staff. As such, the Users Committee opted to discard the message and chat boards that had been used extensively (and chaotically) in Ricecooker 1.0.

4.2 Ricecooker 2.0: Development Realities

Since members of the Users Committee had worked to streamline their CMS requirements, they had expected Ricecooker 2.0 to be a reasonable (and efficient) development process. The project had after all, been placed in the hands of a qualified and enthusiastic team. However, the development and subsequent operational realities were a bit of a surprise to everyone.

Developing and building the CMS proved to be more ambitious than anyone had anticipated. The four-month development and implementation timeframe was exceedingly tight, and proved to be a huge challenge. At the end of September 2003, the VFS students finally completed the system plans for Ricecooker 2.0—a whopping 61-page document that included schematic diagrams, wire frames, interactive storyboards, design comps, content inventories, and validation scenarios. Despite the trimmed down requirements, the VFS team still needed extra time. Working well into the original usability testing and pre-implementation period, they took two and a half months longer to complete the system. In the end, time constraints made it impossible for them to deliver all components of the system: The advertising, circulation,

[62] Ibid.

marketing and web sections wcrc underdeveloped. While data fields for each of these departments had been created, data input forms and output templates still did not exist. As well, thc VFS team did not have the time to build an interface that would allow users to upload content from the CMS directly to the public web site. Instead, "hooks" were placed in the system's code so that these functions could be developed in the future, and linked up with the rest of the CMS.

By the time Ricecooker 2.0 was finally unveiled to the Users Committee on February 19th 2004, the projcct was already a month and a half past the initial go-live date. Since the project had progressed far into thc students' second school term, there was barely any time left for the team to conduct usability testing, or to assist with the implementation of Ricecooker 2.0. With the exception of a quick demonstration, the students would not be able to provide any further technical or training support. As the VFS Senior Instructor explained, "students can develop professional-level systcms for clients, but the limited timeframe of the courses makes it less possible to provide the kind of follow-up support clients would like to have."[63] In the remaining weeks before graduation, the VFS team would only have time to create system back-up disks, and complete the migration of Ricecooker 2.0 from VFS to *Ricepaper*'s ncw server.[64]

4.2.1 Testing the Cooker... by Trial and Error

Ricecooker 2.0 was deployed at the end of March 2004. However, this successful server migration did not mean that the CMS was now ready to be used. The VFS team had not provided *Ricepaper* with a users' manual. While the Users Committee had a 61-page system plan for reference, none of its members knew how to use the ncw CMS. Still fresh from their previous expcrience with Ricecooker 1.0, both senior management and the Uscrs Committee were determined to put the new system through a vigorous usability testing phase.

Usability testing for Ricecooker 2.0 was stretched over a six-week period, during which a "condensed" production work cycle was carried out with thc new CMS. Members of the Users Committee ran actual text and graphic files through the system, and tricd out the different functions by following the schematic diagrams and interactive storyboards provided in VFS's system plan. Volunteers input differcnt data combinations, and used the system's search and output functions to pull customised status reports for every department. By trying out the functions in a series of different combinations, the usability testers tricd to work out how the system could be used to complement each department's workflow.

[63] Ibid.

[64] Thc magazinc had recently secured a new sponsorship deal with an internct-hosting provider. With two new servers and increased storage space, they now had the capacity to store *both* editorial and graphic files on Riccooker 2.0 if they used the CMS.

Working simultaneously alongside the usability testers was a small team of volunteers from the Users Committee, who documented and matched each function according to its corresponding stage in the magazine's workflow cycle. The record of these matches was then used to compile a step-by-step manual for the rest of the staff. The plan was to use this booklet as a training manual to educate the rest of the staff on the "best way" to use Ricecooker 2.0 to facilitate each of their department's daily processes. Over six weeks, the entire Users Committee spent a total of 18 hours on testing and documenting the system.

4.2.2. Bug-Infested Rice is Not Nice

Through the testing process, it was discovered that several of the functions were either faulty, or did not make sense with the magazine's workflow. The Users Committee's largest concern was Ricecooker 2.0's tendency to "reassign" and "eat up" data. This was particularly confusing for editors, as article files would often "reassign" themselves, and data matrixes and status reports would then be associated with the wrong article. For instance, a search for an article on Rice Rockets might turn up an article on tap-dancing Asians instead. An even more frustrating discovery was the fact that uploaded files would often never show up on the system.

The production calendar function was also rife with bugs— missing graphics, lack of date indicators, faulty scheduler, error messages. Sometimes, data would even revert to the default setting. These kinds of bugs were everywhere in Ricecooker 2.0, making it literally impossible for users to depend on the system as an authoritative data source. By the end of their testing period, the Users Committee had identified approximately 72 bugs, with more that were still undiscovered. Overall, they considered Ricecooker 2.0 to be an "inflexible" and "non-intuitive" system. In the words of a member from the Committee, "The breadcrumb navigation was bad, version control was problematic, and the flat-plan function kept bringing up blank pages. Let's just say that the system didn't translate what we had envisioned very well."[65]

4.3 Salvaging the Cooker

With new bugs constantly popping up in the system, it was apparent that Ricecooker 2.0 was just simply not ready to move into the implementation stage. At the end of May 2004, Jim Wong-Chu and I determined that Ricecooker 2.0 could not be used in *Ricepaper*'s workflow. Since the VFS students were no longer available, all further redevelopment and programming work would have to be executed by our own staff. Meeting with the Users Committee and IT department, we mapped out a four-step strategy to debug and salvage Ricecooker 2.0.

[65] Lee, Carol (Art Director and Production Manager, *Ricepaper*). Interview by author, Vancouver, BC, June 24, 2005.

1) ***Recruit more IT staff to debug and implement Ricecooker 2.0.*** We realised that it would be necessary to have a team of in-house developers who were familiar with PHP and MySQL, the programming languages that VFS used to build Ricecooker 2.0. An active recruitment drive was conducted for experienced volunteer developers: the Users Committee sent out open calls to universities and colleges, contacted professional associations, and posted notices in online developer forums. This time round, it was much easier to attract volunteer developers, as the project involved redesigning Ricecooker 2.0, instead of building a new CMS from scratch. A redevelopment team was formed, consisting of seven members, with IT Manager Jonathan Lin as its leader. In comparison, the VFS development team was two times larger, with 16 students. Given this difference in workers, the redevelopment project was extremely challenging. However, *Ricepaper* was still relatively inexperienced with CMS development, and we did not realise how ambitious the redevelopment process would be. Rather, we approached the situation in the same way we had handled Ricecooker 1.0—through trial and error.
2) ***Amalgamate the editorial, art and production departments into a single temporary unit.*** This new unit, made up of the product-producing departments, was nicknamed APE (Art, Production, editorial). The purpose of the step was to consolidate resources, so that all tech-savvy art and production volunteers could be reassigned to help the Users Committee with the CMS project. This idea resulted from the development of Ricecooker 2.0, a phase that had created many overlapping roles between the staff from these departments. Throughout the development process, art and editorial staff had contributed to production tasks once their own workflows reached a lull, in order to allow production staff to assist with requirements gathering and usability testing. The consolidated APE unit was a natural extension of this existing "staggered" work method used by the art, production, and editorial volunteers.
3) ***Streamline Ricecooker 2.0 by cutting down the system to its bare essentials to meet Ricepaper's most critical needs.*** Both the IT Manager and Users Committee felt it would be best to strip Ricecooker 2.0 down to a manageable scale, and focus on repairing its best-developed functions—editorial, art and production. Since Ricecooker 2.0 was designed to be a system for integrated data, it was of utmost importance to keep the core database and input templates, which contained 80 different fields for article details, production specifications, contributor data, and status information. The CMS also had existing search, automatic tracking, and staff reminder capabilities—aspects that the Users Committee felt were critical for *Ricepaper*'s production process. Another crucial feature to retain was document version control. Ricecooker 2.0 was built to update the working line-up and status reports each time a user uploaded or downloaded a new file to the system—a time-saving function for editorial and production staff. Of utmost importance, however, were

the existing flat plan and production calendar functions that governed the time and layout assignment for the entire magazine. Ricecooker 2.0's flat plan function was particularly unique: the system could automatically generate a new flat-plan whenever users updated specific page assignments in the article input template.

4) ***Create an online forum to facilitate communications between the Users Committee and the Ricecooker redevelopment team.*** This new forum, nicknamed "Ricepot", would serve as the information centre for all developments concerning the Ricecooker system. The idea was similar to the original discussion boards in Ricecooker 1.0, except that Ricepot was not intended to be a component of the redeveloped CMS. Despite the fact that Ricecooker 2.0 did not have a built-in communications tool, the volunteers did not feel the need to integrate a forum function into Ricecooker 3.0 at this point. Ricepot was a separate system meant for the use of the Users Committee and CMS redevelopment team. The purpose was to increase communications efficiency between the usability team and redevelopment team. Proper coordination between everyone was necessary to ensure that functions were not just fixed, but also compatible with the magazine's processes.

The Users Committee and the IT staff on the redevelopment team were now entrusted with the task of designing a workable CMS, and advising on all development options. It should be noted that the IT staff's primary duty was to build the system, while the Users Committee's responsibility was to test and direct the entire development project. In the event that conflict of ideas arose between the two groups, it was the Users Committee, and not the IT staff, who had the authority to make decisions on behalf of all the departments and volunteers. It was estimated that the debugging and recoding process would take approximately three months. In the meanwhile, the rest of the magazine's staff would continue to complete production tasks by using their current methods: circulating documents via e-mail, communicating by phone and meetings, and storing files on the *Ricepaper* FTP site.

4.4 Why Redevelopment Still Could Not Salvage Ricecooker 3.0

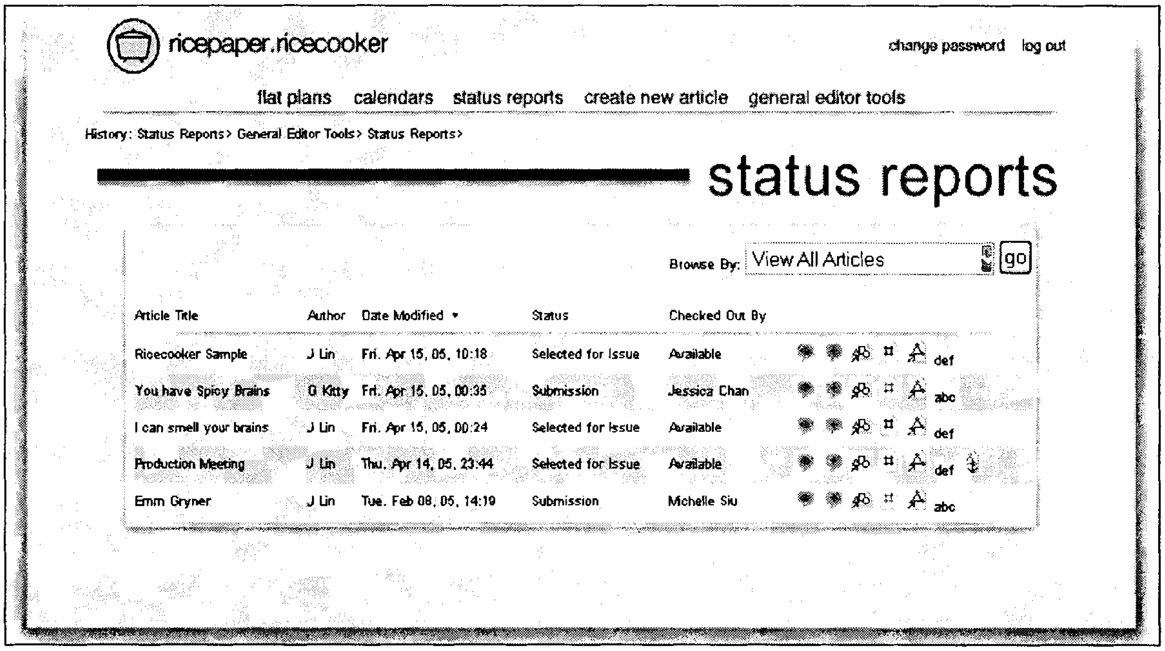
By scaling down Ricecooker 2.0 and repairing all the bugs in the system, the IT staff were able to create Ricecooker 3.0, a functioning but ultimately unsuitable system. They also attempted to integrate most of the usability testers' suggestions for functional improvement without changing too much of the system's original infrastructure. However, Ricecooker 3.0 still failed to match of *Ricepaper*'s needs. The system was limited by the following:

1) ***Version control of files was too rigid.*** Whenever a user downloaded a file, the CMS automatically recorded this as one "edit" attempt, assigning a new version number to the document. As well, the system only allowed users to track and store up to five drafts of the

same file. When an article reached the end of its editing phase, there was often no room left to save its later versions online. This was because the CMS falsely recorded new "versions" each time designers downloaded a file for viewing, but not editing.

2) ***There was no system administration panel.*** All basic system permissions had to be changed manually in the code, as there was no user-oriented administration back-end. The IT staff had to process all password changes, authorise new users, manage accounts, and set access levels for all users. Even something as simple as deleting a file had to be performed by the System Administrator. All these limited the department heads' capacity to manage content and monitor staff activity with the CMS.

3) ***A complicated user interface.*** While aesthetically pleasing, the user interface was difficult to navigate. In an effort to minimise the amount of text on the screen, the VFS team had designed the CMS with several tiny icons, each representing a different function.



Browse By: View All Articles

Article Title	Author	Date Modified ▾	Status	Checked Out By	
Ricecooker Sample	J Lin	Fri. Apr 15, 05, 10:18	Selected for Issue	Available	def
You have Spicy Brains	G Kitty	Fri. Apr 15, 05, 00:35	Submission	Jessica Chan	abc
I can smell your brains	J Lin	Fri. Apr 15, 05, 00:24	Selected for Issue	Available	def
Production Meeting	J Lin	Thu. Apr 14, 05, 23:44	Selected for Issue	Available	def
Emm Gryner	J Lin	Tue. Feb 08, 05, 14:19	Submission	Michelle Siu	abc

Figure 5. Ricecooker 2.0 Interface

As Carol Lee astutely observed, "Once you figured it out, it was actually not hard to use Ricecooker 3.0. But if we left it up to the staff to figure out on their own, they probably would never use it."[66]

4) ***The flat plan function was limited in its ability to differentiate between ads and editorial.*** The VFS team had never worked in a publishing environment, and so had interpreted many

[66] Lee, interview.

of the staff's requirements in a rigid, function-by-function manner. Take for example, the Production Manager's request for a flat plan function. The VFS team created a tool that allowed the Production Manager to allocate editorial and ads to specific pages. However, this tool did not allow the Production Manager to designate the placement of fractional ads or editorial. If two ads appeared on the same page, there was no way for a designer to tell the size and exact placement of the ads. The VFS team had built a tool to fit the magazine's exact specifications, but "the human factor of being able to move things around was not there at all."[67]

4.5 Ricecooker 3.0: Ambition vs. Reality

The initial design of Ricecooker 2.0 took place in September 2003, and continued for a period of six months. In comparison, Ricecooker 3.0's subsequent redesign took half the amount of time, starting in June 2004 and ending just three months later. Prior to Ricecooker 2.0, it had been difficult for *Ricepaper* to develop its own CMS with only three part-time IT volunteers. Ricecooker 3.0, however, was redeveloped under very different circumstances. IT staffing was doubled, bringing more programming expertise to the magazine. These volunteer developers pooled their collective skills and time together, each working approximately 5 to 9 hours a week to salvage Ricecooker 2.0. Totalled up, the entire team worked about 35 to 56 hours a week, roughly equalling the estimate for two full-time programmers. [68]

This fresh influx of professionally trained developers renewed the staff's overall confidence, reigniting their passion for the CMS project. The inadequacy of Ricecooker 2.0 was a catalyst that actively pushed the management and staff to address the problems with the system. Unwilling to let all their development efforts go to waste, they still intended to put the CMS to use in the future. The experience with VFS also made them recognise the need for the magazine to rely on itself for the development, and implementation of a functional CMS. More importantly, this experience changed the collective mind-set of the staff: Instead of regarding Ricecooker as a secondary priority, they now saw the CMS as a system that was as important as producing the magazine itself. If it is any indication, the volunteers spent two whole months on redefining how the CMS should be modified in Ricecooker 3.0! (Ironically, the intellectual effort they spent on analyzing Ricecooker 2.0 amounted to twice the time spent on the original planning.) As well, more volunteers joined the Users Committee as usability testers. Over the three-month redevelopment period, 15 staff took turns to identify bugs in the system and

[67] Ibid.

[68] The number of hours dedicated each week also depended on the magazine's production cycle, during which the IT staff had to put more time towards system maintenance, troubleshooting, and electronic archival work.

suggest improvements to various functions. An average of 15 to 20 volunteer-hours were logged each week in order to redevelop Ricecooker 3.0.[69]

It was soon apparent that the IT developers would have to change the underlying infrastructure that the VFS team had programmed, especially if Ricecooker 3.0 was to be modified to match *Ricepaper*'s publishing functions. The biggest headache though, was the realisation that the system just did not go hand-in-hand with the magazine's workflow. Even though Ricecooker had originally been custom- designed to recreate and simulate *Ricepaper*'s production processes, almost a year had passed since the magazine's staff had mapped out their roles and workflows to the VFS team. Ricecooker 2.0's initial delay, combined with the lengthy usability testing, debugging and recoding period, had led to many transformations in the magazine's production process. The most noticeable of these was the amalgamation of the art, production and editorial departments into a single APE unit. In fact, the IT developers found it challenging to develop a CMS to cater to the magazine's fluid interdepartmental processes. As Jonathan Lin, *Ricepaper*'s IT Manager, noted:

> It is not possible to build a system that is fitting for only one particular situation, nor can a task be programmed to match a particular role in the organisation. For instance, Section editors previously only took care of the structural editing process. However, the APE workload distribution changed that. Now, Section editors participate in all stages of article development—from text and art conception, structural editing, fact-checking, to the final layout and proofing. Ricecooker 3.0 was tweaked to take care of the earlier version of the editing process. Yet by the time we were finished, the magazine's processes no longer matched the original concept. Even if we hadn't created the APE unit, something else would have changed in the workflow. This is because staff roles are always changing in *Ricepaper*, and you can never be certain that the same person, or even position, will always be responsible for the same kinds of tasks. The problem with Ricecooker 3.0 was that users got holed in by the system's limitations every time change occurred. There just wasn't the flexibility to cope with all these changes.[70]

While Lin used the amalgamation of the APE unit as an example, his observations were relevant to all processes at *Ricepaper*. Large and small changes occurred so consistently at the magazine that it was necessary to build a system that could anticipate change, and grow with the organisation. In reality, Ricecooker 3.0 could be used for the magazine's art, production and editorial processes. However, senior management realised that if the system was implemented, the staff would once again be in a situation where they would constantly have to work around the Ricecooker 3.0's limitations. If the magazine wanted a CMS that would cater to its long-

[69] Users Committee considered each redevelopment of the CMS to be a brand-new version. Hence, instead of naming Ricecooker 3.0 "Ricecooker 2.1", they felt that the new system should be given its own independent numeric name.

[70] Lin, Jonathan (Information Technology Manager, *Ricepaper*). Interview by author. Vancouver, BC, 24 June 2005.

term growth, they had to reprogram Ricecooker 3.0 with functions that could accommodate their fluid publishing operations.

This was a very important realisation for the *Ricepaper* staff. The VFS team had designed a CMS that catered to publishing as most non-publishers understood it: a sequence of individually performed functions. In *Work-Oriented Design of Computer Artifacts*, Pelle Ehn criticises this type of systems-based thinking as a design method that reduces the job of many workers to "algorithmic procedures", rather than helping to enhance their skills and processes. Ehn notes that objectifying and formalizing systems descriptions lock users out of the design process, and do not support participative communications between the users and designers.[71] This describes the problem with Ricecooker 2.0's development, which was largely an inquiring and detached reflection by the VFS students. While the Users Committee and some *Ricepaper* volunteers—the ultimate users and decision-makers—were certainly involved, they left it up to the VFS team to interpret *Ricepaper's* production needs. The Users Committee overlooked the need to provide the VFS students with the non-explicit, practical understanding of the magazine's publishing process. Since the VFS team were not familiar with *Ricepaper's* fluid operating structure, what they saw from the requirements list was a traditional, function-oriented publication: Managers proposed specifications for particular work units, appointed measurable standards of performance to their assigned staff, and the process was repeated for the next work unit in the sequence.[72] Hence, the VFS team's design of Ricecooker 2.0 was in fact a description of how they saw *Ricepaper's* production processes, and a solution to solve production problems—according to their interpretation. At the same time, Ricecooker 2.0 was indeed what the *Ricepaper* staff had asked for in their paper specifications. The fact that the CMS did not work entirely with the magazine's workflow attests to the importance of obtaining feedback from end-users throughout the development process.

Since the purpose of a CMS is to manage process, it becomes problematic if a system is programmed with function, instead of process in mind. As previously alluded to in Chapter 1, the relationship between content and process in a CMS is inextricable. This meant that if the *Ricepaper* staff still intended to use Ricecooker 3.0, they had to translate the system's function-oriented features into process-oriented ones.

The top-down method of systems design—in which the VFS developers were given free rein to compose a 61-page specification, and then code to it—seemed like a good idea in theory. However, "in practice, it may foster heroic designers to whom no one listens, as well as narrow goal oriented designers that follow the methodology instrumentally, but leave the

[71] Ehn, *Work-Oriented Design of Computer Artifacts*, 51.
[72] Hancox, *Topics in Publishing Management*.

humanistic behind".[73] This certainly was the case with Ricecooker 2.0, especially with *Ricepaper*'s "hands-off" approach, and the VFS team's intent on building to specifications. It is worthy to note, however, that while the top-down development model has received substantial criticisms in software development circles, most people outside of this industry are still unaware of its many drawbacks.

4.6 Some Lessons Learned... and Benefits

Ricepaper's senior management saw that Ricecooker 3.0 was not the software solution to the problems they were having with inefficient workflow. However, the process of developing a home-brewed content management system did strengthen the magazine's staff community, and brought greater self-awareness to the organisation. While Ricecooker 3.0 could not accommodate the magazine's needs, the bug-free system was an encouraging sign to the *Ricepaper* staff—they had been able to fix the CMS on their own. While the lack of financial resources constrained *Ricepaper*'s CMS development options, the staff had become more experienced at soliciting and acquiring low-cost development assistance. As well, the staff learnt some invaluable lessons through participating in the development process.

1) ***During the Ricecooker 2.0 development crisis, Ricepaper had amassed a team of skilled IT programmers.*** The magazine also subsidised professional training courses for two team members, who subsequently returned to conduct PHP and MySQL workshops for the rest of the team. 3.0. *Ricepaper*'s staff now saw that it was crucial to work with developers who had an understanding of magazine publishing and their workflow processes.
2) ***Volunteers saw how their concepts could be translated into working functions in a CMS.*** Both Ricecooker 2.0 and Ricecooker 3.0 had automated tracking features. While these did not work seamlessly, volunteers saw how these functions could automatically cross-reference workflows, helping to ensure better workflow continuity. This made volunteers realise how important it was for the magazine to have a functional CMS. Thus, despite the setbacks, they were even more determined to redevelop Ricecooker into a working system.
3) ***Experiencing three reincarnations of Ricecooker made the magazine's staff less resistant to technological change.*** By participating in the conceptualisation and development of Ricecooker, *Ricepaper*'s staff had become familiar with the process of integrating their workflows into a CMS. This empowered the magazine's staff, and created a self-aware community of flexible workers. Armed with this new knowledge, staff were also able to determine, to a greater degree of accuracy, the exact requirements they needed in a CMS.

[73] Ehn, *Work-Oriented Design of Computer Artifacts*, 188.

4) ***Sometimes, end-user involvement is crucial in the development process.*** Through a process of trial and error, volunteers gradually came to understand the importance of participating in the development process. As Ehn states so eloquently, "Ideally, the design process involves as participants all those who can be directly affected by the system, its stakeholders. The designer is someone that through encouragement and facilitation enables the participants and stakeholders to deal more effectively with their organisational messes."[74] Up until this point, the volunteers had always thought that CMS development was a project that depended on the skills of third party developers. By actively participating in the development process, they realised that the solution to their problem was in their own hands, and they could solve it.

5) ***When creating a list of CMS requirements, it is important to note the difference between the actual, and the ideal.*** The requirements list that *Ricepaper* submitted to VFS contained a number of proposed functions that had been modelled after idealised processes, and were not based on how the production department interacted with other units. As such, the VFS team created a system with functions that did not fit into the production department's process—because their idealised procedures never materialised.

6) ***Establishing the Ricepot forum as a communication tool resurrected Ricepaper's online community.*** While this forum had initially been set up to facilitate communications between the usability testing and development teams, members from these teams soon started using Ricepot to communicate with the other departments. This slowly re-established the magazine's online community—bringing back one of the positive influences of Ricecooker 1.0. While Ricecooker 1.0's community had often been a chaotic mess of online discourse, the appointment of Ricepot moderators helped maintain order in the new community. Later on, both the forum and community would have a great influence on the magazine's future development, as detailed in the next chapter.

4.7 Continuing Development: A New Approach

The benefits that Ricecooker 2.0 and 3.0 brought to the organisation were not direct improvements to the magazine's workflow. However, both systems did change the mindset of the staff, and affected their outlook on future CMS development.

It is difficult to understand a technology without having a functional understanding of how it is used. Furthermore, that understanding must incorporate a holistic view of the network of technologies and activities into which it fits.[75] The VFS students, being developers, treated the

[74] Ibid, 189.

[75] T. Winograd and F. Flores, *Understanding Computers and Cognition—A New Foundation for Design* (Norwood: Ablex, 1986), 5-6.

design of Ricecooker 2.0 as a technological device in isolation of actual magazine processes. They did not have an overview of how the CMS technology had to service the magazine and its consistently evolving activities. However, as clients and end-users, the *Ricepaper* volunteers also did not take more initiative to involve themselves more extensively with the development process. This was the reason why both Ricecooker 2.0 and 3.0 were not built in the context of *Ricepaper's* larger network of practices. To quote Winograd and Flores, "In designing for computer artefacts we should not primarily be concerned with meeting technical specifications for efficient use, but with understanding conditions for the future user's readiness-to-hand in using the artefacts being designed".[76] The *Ricepaper* Users Committee and IT staff saw that a future concern should be to anticipate the breakdown that could occur in the use of the CMS, a potential problem that could occur whenever staff were reassigned, and processes readjusted.

The Users Committee felt that the next version of the system had to be programmed to provide optimum structure, and accommodate operational change. They re-envisioned the next CMS as an adaptable system that could integrate new functions and processes as and when the need arose. Two years ago, building this level of flexibility into the system might have been more challenging, as most CMS were only customisable during the development process, and not after users implemented the system. IT staff felt that the redevelopment of Ricecooker called for "agile programming", a different approach to the design process. Exploring this a little further, IT Manager Jonathan Lin writes:

> There's a concept for programming workflow that involves tasks, users, roles and permissions. Right now, the Ricecooker 3.0 system works by assigning different users specific permissions, but these users do not have particular roles or responsibilities. For instance, only the Art Director has been given the permission to assign page numbers in the flat plan. However, she can have this permission level, but choose not to care about a particular area of administration. In a CMS where roles are programmed into the system, a user who has been assigned a role cannot chose to ignore a particular task. This has always been the problem with *Ricepaper*—staff are constantly mobile, and job reassignments often mean that they neglect their original tasks. This type of programming does allow us to change a person's role, but the trick is to combine all these roles into a workflow chain, where the Ricecooker Administrator can mix and match the roles to meet the magazine's needs. Let us say in the beginning, we only have one phase in our editorial process—structural editing. Later on, copy-editing joins the workflow, which in turn creates new roles for our structural editors, who now also become copyeditors. The Ricecooker Administrator can then add this new role to the user's existing profile, giving that person the permission and responsibility of completing the copy-editing job. This is an example of how the CMS can be programmed to "anticipate" our needs. Being able to split roles apart is very important, as these roles tend to reflect our workflow processes. A person can be a writer at the beginning of the workflow, but also be an editor at the end of the cycle. This

[76] Ibid.

> brings structure to our processes, as we can track who does what task. It also allows editorial assignments to be flexible. With anticipatory systems design, we can evolve organically as processes change.[77]

Ideally, the next CMS would facilitate inter-departmental communications, and automate reporting procedures and project tracking. As well, the system would be a database that protects proprietary information, but still shares documents seamlessly between the various departments. More importantly, it needed to centralise the online management of all departments, and sustain the magazine's evolving processes.

All along, the Users Committee had an unwavering intention to develop a working and satisfactory CMS. Even when their attempts to salvage Ricecooker 3.0 failed, they did not question the feasibility or necessity to redevelop the system—yet again. However, the decision to push forward with another development attempt could put strain on the magazine's processes. The next chapter explores the interim measures that the *Ricepaper* staff undertook to ensure the continuation of the magazine's production activities. It also evaluates *Ricepaper's* decision to continue redeveloping Ricecooker, and looks at other possible content management solutions for the publication.

[77] Jonathan Lin, e-mail message to the Users Committee, June 18, 2005.

CHAPTER 5. Calling Ricepot to the Rescue

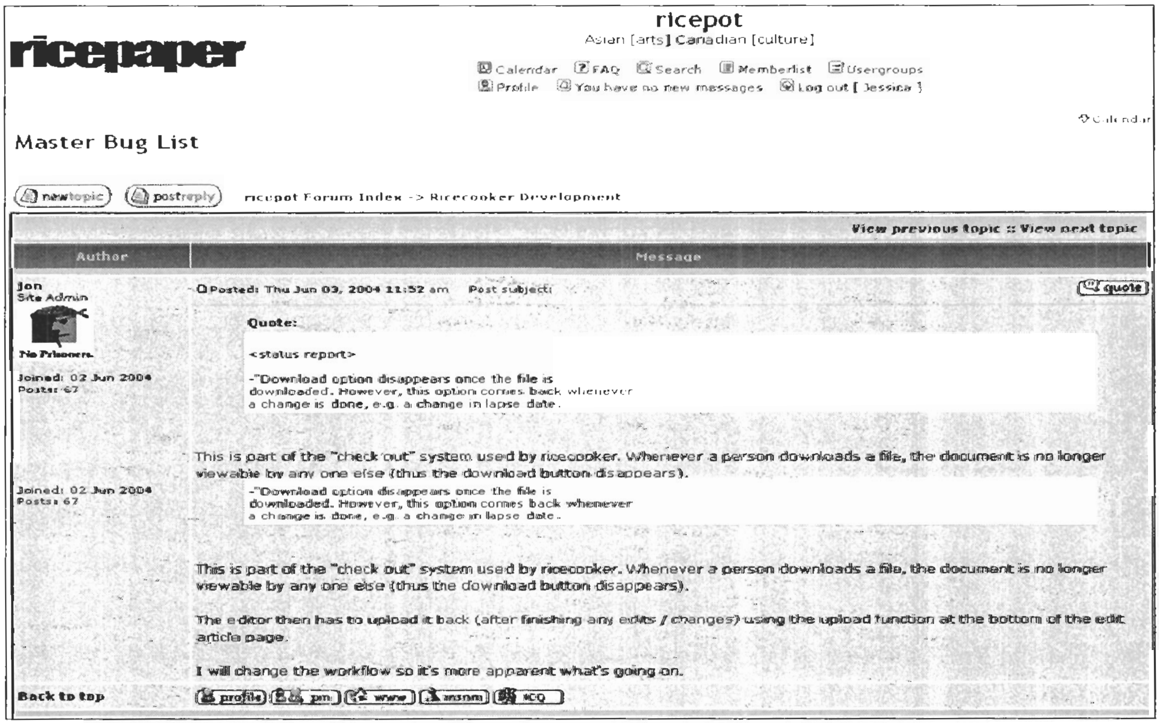
While the Users Committee and IT staff assumed that the redevelopment of Ricecooker should continue, both the Publishers and I had some initial concerns. Developing a forth version of the CMS—"Ricecooker 4.0"—meant that the Users Committee would have to continue with usability testing, systems research and CMS design. Given that the Committee consisted of all department heads, plus one-third of the editors and designers, this would take a valuable portion of the staff away from the actual operations of the magazine.

The Users Committee, which was first formed in November 2002 to conduct tests on Ricecooker 1.0, had spent almost two years on testing and researching requirements for the CMS. At an estimated eight hours a week, this meant that the Users Committee had poured about 640 hours into the system thus far. While the staff had consistently produced the magazine alongside their development work, the senior management questioned if they had already spent too much time and effort on the project. In fact, the Publishers even suggested dropping the CMS entirely, especially since no one was actually using Ricecooker 3.0. However, the Users Committee and IT staff vehemently objected to this suggestion, as they were reluctant to see their development efforts go down the drain. More than anything, they argued, the volunteers needed a CMS, and the only way *Ricepaper* could afford content management software was to develop it ourselves. Thus, despite the development realities of Ricecooker, we decided that fixing a flawed system was better than having no system at all. As it was, the need for a CMS was increasing rapidly with the production of each issue of our magazine. There was a pressing need for an interim central repository—documents needed archiving, data had to be accessible, while workflow processes needed tracking.

In June 2004, *Ricepaper*'s IT staff had created Ricepot, an online discussion forum. As detailed in Chapter 4, the Users Committee and IT staff had felt the need for a forum that would facilitate communications during the redevelopment of Ricecooker 3.0. The idea for Ricepot sprung from the message boards that were originally on the Ricecooker 1.0 web site. Back then, these html-coded boards were rudimentary communications tools that allowed users to post and respond to a single discussion thread. Ricepot, however, was a sophisticated version of its predecessor. Created from phpBB, an open source bulletin board package, it was a customisable forum based on the PHP server language.[78] Since phpBB worked "out-of-the box", it was simple for the IT staff to install Ricepot—a task that was completed in a week.

Unlike the basic message boards in Ricecooker 1.0, Ricepot could support public and private forums, an unlimited number of posts, and had a private messaging system. It also had moderation features that let users edit topics, delete and move posts. This differentiated Ricepot from the previous message boards: the former allowed moderators to maintain order on the forum, while the latter was one continuous discussion thread that later fell to disuse due to sheer disorganisation. It is important to note that Ricepot was not a crude version of Ricecooker. Ricepot was a forum used for staff communications, while Ricecooker was a CMS meant for tracking production processes. The two were entirely different software. What is interesting is *Ricepaper*'s attempt to use Ricepot to fulfil the same purposes as Ricecooker.

The art, production and editorial (APE) staff on the Users Committee initially used Ricepot to post results from their usability tests. Figure 6 shows a sample screen-shot of a Ricepot bug discussion thread.



Master Bug List

Jon
Site Admin

Joined: 02 Jun 2004
Posts: 67

Posted: Thu Jun 03, 2004 11:52 am   Post subject:

Quote:

<status report>

-"Download option disappears once the file is downloaded. However, this option comes back whenever a change is done, e.g. a change in lapse date.

This is part of the "check out" system used by ricecooker. Whenever a person downloads a file, the document is no longer viewable by any one else (thus the download button disappears).

Joined: 02 Jun 2004
Posts: 67

-"Download option disappears once the file is downloaded. However, this option comes back whenever a change is done, e.g. a change in lapse date.

This is part of the "check out" system used by ricecooker. Whenever a person downloads a file, the document is no longer viewable by any one else (thus the download button disappears).

The editor then has to upload it back (after finishing any edits / changes) using the upload function at the bottom of the edit article page.

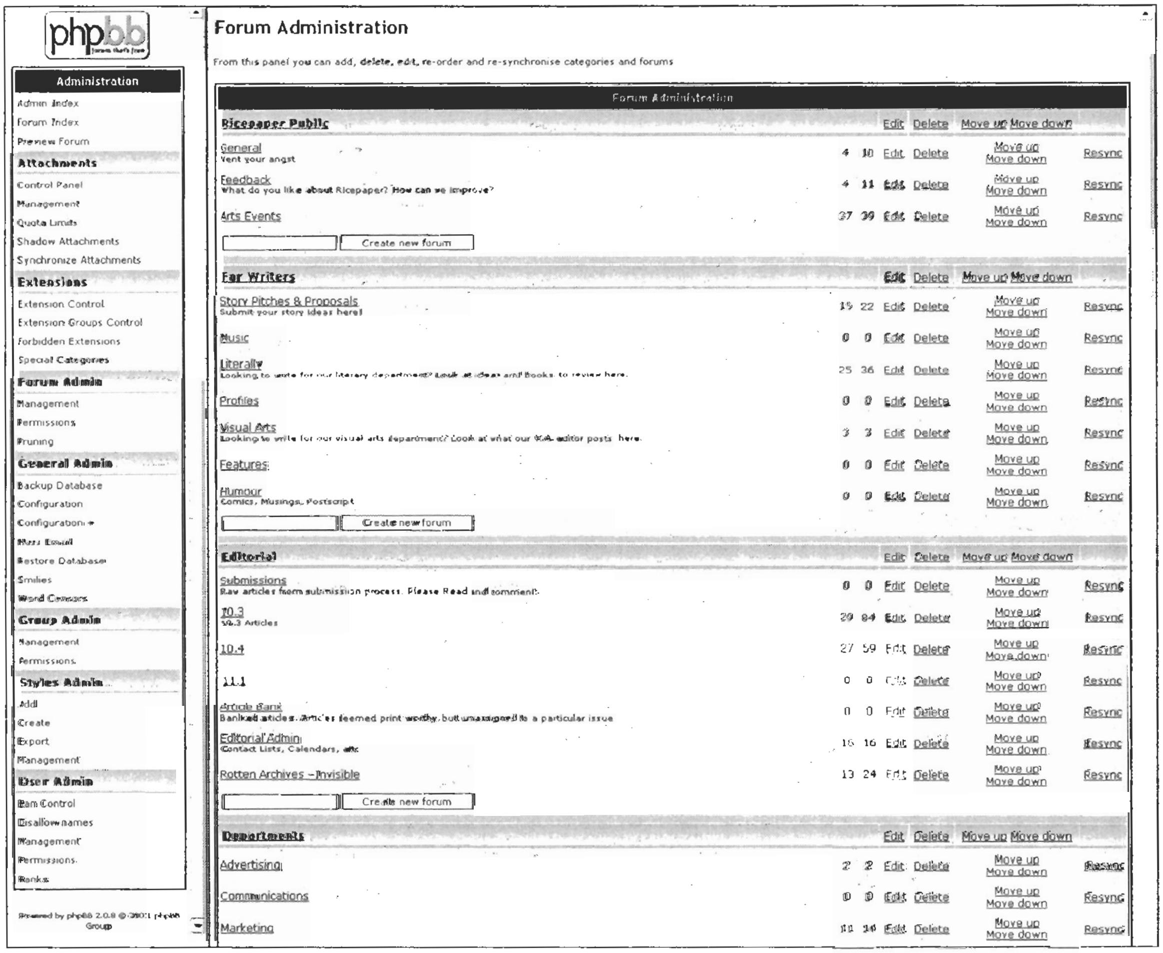
I will change the workflow so it's more apparent what's going on.

Figure 6. Sample Ricepot Bug Discussion Thread.

[78] phpBB: Creating Communities [online], 2005.

Over the course of using the forum, the APE staff gradually expanded their use of the system to their own individual departments. Instead of just using Ricepot to post online discussions, the APE staff utilised the forum as a central document repository for posting production calendars, and in-progress articles, making these documents readily accessible to all staff. By August 2004, just two months after the launch of Ricepot, they were already using Ricepot to provide inter-department status updates, and track the editorial status of on-going articles. Once an editor completed the structural editing for an article, they would indicate the new status of the document by changing the title of the article in the main article index. The new entry would read: "Structural complete: Article X." When copy editing started, the editor would indicate this by changing the article name to: "Copy edit ongoing: Article X." Thus, just by looking at the main article index, the APE staff could see the status for every article.

Another benefit that Ricepot offered was a flexible, straightforward administration back-end. This function made it easy for any staff with some technical knowledge to administrate and maintain Ricepot, by simply setting the appropriate permissions in the system.



Forum Administration

From this panel you can add, delete, edit, re-order and re-synchronise categories and forums

Forum Administration						
Ricepaper Public			Edit	Delete	Move up Move down	
General Vent your angst	4	10	Edit	Delete	Move up Move down	Resync
Feedback What do you like about Ricepaper? How can we improve?	4	11	Edit	Delete	Move up Move down	Resync
Arts Events	37	39	Edit	Delete	Move up Move down	Resync
Create new forum						
For Writers			Edit	Delete	Move up Move down	
Story Pitches & Proposals Submit your story ideas here!	15	22	Edit	Delete	Move up Move down	Resync
Music	0	0	Edit	Delete	Move up Move down	Resync
Literally Looking to write for our literary department? Look at ideas and Books to review here.	25	36	Edit	Delete	Move up Move down	Resync
Profiles	0	0	Edit	Delete	Move up Move down	Resync
Visual Arts Looking to write for our visual arts department? Look at what our VA editor posts here.	3	3	Edit	Delete	Move up Move down	Resync
Features	0	0	Edit	Delete	Move up Move down	Resync
Humour Comics, Musings, Postscript	0	0	Edit	Delete	Move up Move down	Resync
Create new forum						
Editorial			Edit	Delete	Move up Move down	
Submissions Raw articles from submission process. Please Read and comment.	0	0	Edit	Delete	Move up Move down	Resync
10.3 10.3 Articles	29	84	Edit	Delete	Move up Move down	Resync
10.4	27	59	Edit	Delete	Move up Move down	Resync
11.1	0	0	Edit	Delete	Move up Move down	Resync
Article Bank Banked articles. Articles deemed print worthy, but unassigned to a particular issue	0	0	Edit	Delete	Move up Move down	Resync
Editorial Admin Contact Lists, Calendars, etc	16	16	Edit	Delete	Move up Move down	Resync
Rotten Archives – Invisible	13	24	Edit	Delete	Move up Move down	Resync
Create new forum						
Departments			Edit	Delete	Move up Move down	
Advertising	2	2	Edit	Delete	Move up Move down	Resync
Communications	0	0	Edit	Delete	Move up Move down	Resync
Marketing	11	14	Edit	Delete	Move up Move down	Resync

Figure 7. Sample Ricepot Administration Panel

New functions did not have to be hard-coded into the system by the IT staff. This gave the users greater administrative flexibility, as they could add new departments whenever the need arose.

Such developments and features showed that Ricepot could be as a temporary replacement for Ricecooker during the redevelopment of the CMS. The plan was that Ricepot would serve as an alternate online communications tool to facilitate workflow between every department. As a centrally accessible system, it could deliver critical information about the magazine's daily operations.

5.1 Migrating a Magazine's Operations Online

Senior management adopted an informal department-by-department approach towards moving the various units on to Ricepot. This was largely due to the differing workloads of every department throughout the magazine's 13-week work cycle. Lull periods in the circulation department could mean intense periods of editing, layout and proofing in the APE departments.

In September 2004, department heads sent e-mail announcements to the staff of each department, explaining what Ricepot was, and detailing management's eventual plan to move all departments on to the system. This was an important step, as many staff members (not only the new interns) did not actually understand what Ricepot was. While the members of the Users Committee had been using the Ricepot for three months, other volunteers in the magazine were not fully aware of the fact that *Ricepaper* was planning another system transition. The department heads sent these e-mail announcements all through the implementation period, providing *Ricepaper* staff with useful information to assist them with their migration progress. As more and more staff adopted the forum, the department heads gradually converted their e-mail announcements to postings on Ricepot. These announcements contained information such as instructions on how to post replies, regular reminders to check the forum, and updates for the instructional PowerPoint presentations on the forum. The department-specific presentations served as online workshops, giving staff the option of learning to use Ricepot during their own time. Each department head also conducted seven on-site training sessions for their volunteer staff at the *Ricepaper* office.

As mentioned in the previous section, the amalgamated APE department was the first to use Ricepot as a workflow management system—to post production specifications, track and comment on articles, update status, and distribute meeting minutes and flat plans. Most of the APE staff had already gone through many IT transitions, and as such, they took to Ricepot's intuitive interface with ease. Nevertheless, a small portion of the staff resisted the transition. While these APE staff were already accustomed to working via e-mail and exchanging electronic documents, their transition to Ricepot would not be an entirely smooth process.

Carol Lee remembers the transition to be a period of reminders:

> Some staff would use Ricepot, and some would not. I had no control over how often they used it, but it was up to me to remind them. In the end, it came down to this—if staff wanted to be part of the online community, they would check the forum. A volunteer's participation in the forum depended a lot on their working attitude and their views towards responsibility. Eventually, most of the staff switched to using Ricepot because they *had to.* They realised it was the only place that stored the information they needed to complete their task. [79]

Since Ricepot did not have functions that allowed department heads to monitor the usage activities of their staff, there was no way for management to make sure they were checking the forum. Essentially, the forum was system built on "trust."[80] department heads also urged their staff to use the forum by showing them how to use the forum to connect with the magazine's staff and readers. An illustrator, for example, might post an announcement on both the public and staff forums, inviting anyone from the *Ricepaper* community to attend his latest gallery opening.

Early on, it had been determined that the editors, many of who were members on the Users Committee, would probably be the most adept at adopting the forum. Unlike some of their peers in the APE unit, they understood the critical importance of such a system, and were committed to executing the forum. This would prove to be the crucial ingredient towards getting staff to "buy-into" the Ricepot system. As an editorial staff member commented, "you feel less resentful about the transition when you actually know why you are spending time to learn to use a new system to do the tasks you have previously mastered by another method."[81]

5.1.1 Implementing Transparent Communications

Despite some struggles during the four-month implementation period, *Ricepaper* still managed to migrate the majority of its departments on to Ricepot. A few changes in communications were immediate for the magazine:

- ***department heads no longer needed to e-mail status updates to all their staff.*** Volunteers were able to find this information themselves by accessing the Ricepot system and viewing the information on-screen.
- ***The staff switched from e-mail reporting to forum reporting.*** This meant that all files and correspondence were located in a central place. Reporting lines became more transparent.

[79] Lee, interview.
[80] Ibid.
[81] Wong, Richard (Editorial Assistant, *Ricepaper*). Interview by author. Vancouver, BC, May 14, 2005.

- ***Ricepot helped facilitate the assignment of staff, resources, and skills to specific projects.*** To illustrate, the image editor would post up a list of all articles that required a match with specific design skills. Interested designers and photographers then would pool their resources together to work on the assignment. By promoting discussion between staff, Ricepot increased the collective creative power of individual departments.
- ***The forum provided the option of communicating publicly and privately.*** A "Personal Message (PM)" function allowed staff to communicate delicate issues directly with one another—such as if an editor did not want other staff to view the sensitive comments they had on a particular writer.
- ***Ricepot provided an "at a glance" snapshot of the organisation.*** All staff were encouraged to explore different sections in the forum to orientate themselves with an overview of the magazine's entire operations.
- ***Ricepot changed the Ricepaper work environment.*** Instead of meeting in person at the office, staff could discuss projects with one another even before they met in person. By scanning the discussion boards, managers were also able to get a sense of how their staff interacted with one another—providing them with valuable insight on the most effective way to divide and team up their staff.

Ricepot centralised information and brought greater transparency to inter-departmental communications. For the first time, staff were interested in the workings and operations of not just cross-unit workflows, but the operations of the *entire* magazine. The staff appreciated Ricepot because it introduced a new level of equality and open discussion to the organisation. Junior staff could pitch their ideas to senior staff, while volunteers could participate in the discussions of other departments. Overall, Ricepot created a "greater sense of belonging"[82] amongst the staff, and promoted a more cohesive workforce in the magazine.

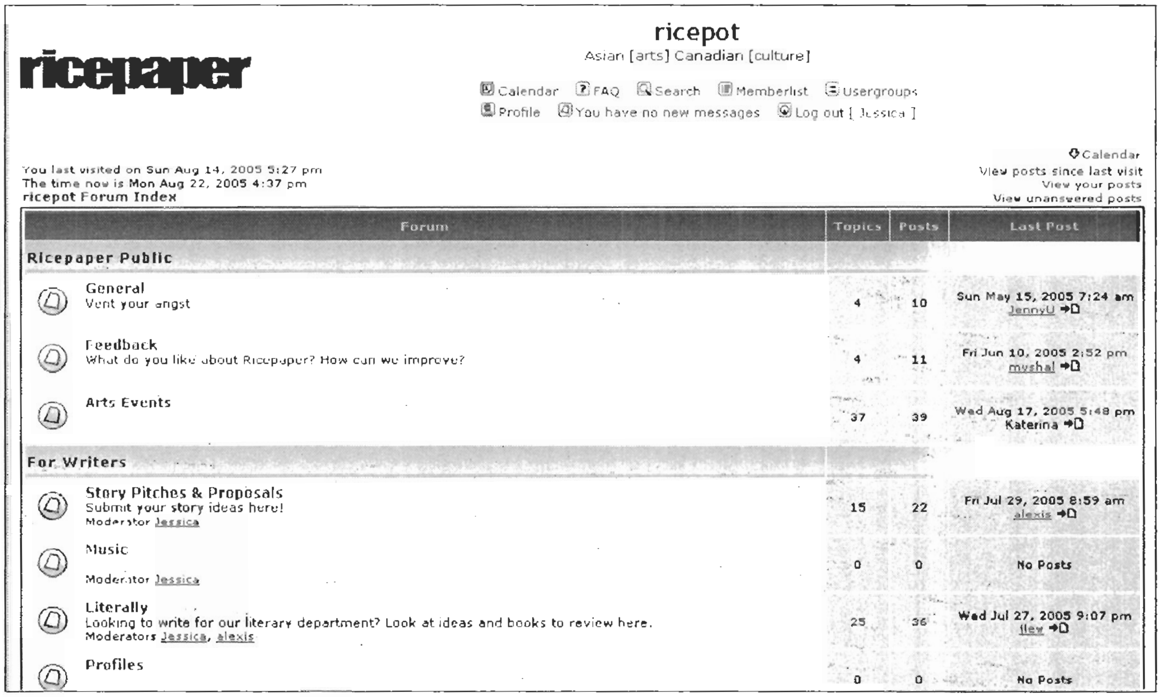
It had always been the Publisher's intent to move a portion of the magazine's operations online. However, this plan had not been possible until the birth of Ricepot. No previous incarnations of Ricecooker were able to support the online mass migration of every department's processes and intersecting workflows. Even Ricecooker 1.0, which the staff used for seven months, was not comparable to Ricepot. The former only had the capacity to house editorial documents and processes, while the latter could accommodate the data and workflows of all eight departments. More importantly, the implementation of Ricepot was successful, despite the minimal staff training and absolutely no usability testing. As the previous chapters show, no amount of usability testing or staff education for Ricecooker had managed to ensure such a smooth migration process.

[82] Wong, interview.

Throughout the uptake of the Ricepot, most of the staff were very much involved with the new system—moving data online, and experimenting with new ways to use the forum. As such, they did not give much thought to Ricecooker, which had not worked for them. In fact, some staff even thought that Ricepot was a reinterpretation of Ricecooker! With the exception of the Users Committee and IT Staff, most of the other volunteers gave no further thought to the original CMS plan. They did not raise any questions about the interim switchover to Ricepot, nor did they express any regret over the failure of Ricecooker 3.0. There was no need to—they were happy using Ricepot to accomplish their activities.

5.2 Ricepot: Two Communities, One Home

Following the migration of *Ricepaper*'s operations on to Ricepot, a "public" forum was added to the system. This allowed external readers to post comments about the magazine's content, and meet other members of the *Ricepaper* community. This section of Ricepot was linked to the magazine's website, www.*Ricepaper*online.com.[83]



ricepot

Asian [arts] Canadian [culture]

You last visited on Sun Aug 14, 2005 5:27 pm
The time now is Mon Aug 22, 2005 4:37 pm
ricepot Forum Index

Forum	Topics	Posts	Last Post
Ricepaper Public			
General Vent your angst	4	10	Sun May 15, 2005 7:24 am JennyU
Feedback What do you like about Ricepaper? How can we improve?	4	11	Fri Jun 10, 2005 2:52 pm myshal
Arts Events	37	39	Wed Aug 17, 2005 5:48 pm Katerina
For Writers			
Story Pitches & Proposals Submit your story ideas here! Moderator Jessica	15	22	Fri Jul 29, 2005 8:59 am alexis
Music Moderator Jessica	0	0	No Posts
Literally Looking to write for our literary department? Look at ideas and books to review here. Moderators Jessica, alexis	25	36	Wed Jul 27, 2005 9:07 pm llew
Profiles	0	0	No Posts

Figure 8. Ricepot General Public Feedback Forum

Only *Ricepaper* staff with could log into the private sections on the forum to view confidential information about the magazine. In separating both the public and private forums, *Ricepaper*

[83] The public layer of Ricepot is available at http://www.ricepaperonline.com/forum/.

was actively acknowledging the presence of two online communities: One that was consisted of staff, and one that was made-up of external readers. The forums became the textual embodiment of *Ricepaper*'s online community. The creation of the public site encouraged audience feedback and discussion, while the existence of the private site promoted the exchange of ideas between departments. Sometimes, the editors further developed audience suggestions into full-fledged editorial content. In such cases, the magazine's consumers also became knowledge contributors, thereby creating a communal exchange of ideas. Discourse of this nature fostered a sense of community amongst its readers, and increased the contact that staff had with their readers. As the central repository that housed all these communications, Ricepot became the virtual "home" for the two communities.

Opening up communication channels to the public had a great effect on the *Ricepaper* volunteers. *Ricepaper*'s reading community was placed in a humanised context: for the first time, the staff got to interact with the people who read the magazine they produced. By participating directly in online discussions with their readers, they came to realise the impact that their work had on this reading community. Encouraged, inspired, and sometimes offended by suggestions on the public forum, *Ricepaper*'s volunteers came to develop a greater sense of ownership towards their magazine. For example, The Marketing staff used the public forum to conduct basic surveys on reader preferences, and to find out where readers obtained their copies of *Ricepaper*. These responses were then used to fine-tune the magazine's circulation plan. Assignment editors also started consulting the public forum more regularly to obtain content ideas, sometimes even polling the readers on the public forum to assess the potential popularity of a story concept. Jenny Uechi explains the reason behind this change in the staff's approach, "We felt motivated to improve the magazine product, because we knew that readers were responding to our work."[84]

5.2.1 Administrating the New Community

While the *Ricepaper* staff used Ricepot primarily to perform work-related tasks, they also logged on to the forum to stay connected with the rest of their peers. As volunteers, they benefited by staying in touch: 1) Publishing newbies could learn more about the magazine's operations; 2) Ricepot gave them access to both the staff and readers; 3) The forum reinforced the magazine's sphere of cultural influence, and fostered a sense of communal belonging. IT Manager Jonathan Lin says it best:

> When we created Ricepot, we wanted to recreate the same intimacy that already existed in our office community. The point was to get our staff to

[84] Uechi, Jenny (Assistant Managing Editor, *Ricepaper*). Interview by author. Vancouver, BC, June 28, 2005.

> feel comfortable enough to post messages like, 'She just had a baby!' or 'Hey, I'm moving next weekend, I need help!' Many of us volunteer at *Ricepaper* because we work with a community of like-minded and caring friends. We wanted to take that working relationship on to the next level, by translating that into Ricepot.[85]

Indeed, Ricepot provided the chance for volunteers to connect with one another beyond the confines of the office walls. Staff who were unable to visit the office on a regular basis could still communicate with their peers. The forum also allowed open discussions to flow between groups of people and individual staff members. With an integrated communications forum, everyone could see the opinions of everyone else. This meant that there had to be standards set out for the Ricepot community. The department heads had to establish rules about respectful discourse and interaction. As with any community, Ricepot users each had their roles, and had to follow certain online etiquette.

The forum was also reorganised into three levels of access: 1) General public, 2) unrestricted staff, 3) restricted management. As explained in Section 5.2, the general public layer, or "open Ricepot", allowed everyone to view the forum community on the *Ricepaper* public website. This layer was not password protected, and was accessible to all staff and external readers of the magazine. The second level of access was open to all *Ricepaper* volunteers. All staff could read, post, and download documents from the unrestricted staff sections. However, only department heads could edit, delete, or move posts. As well, only certain volunteers, such as production staff or editors, had the permission level to upload articles to the forum. A volunteer's access level on Ricepot was determined according to their position, the length of time they had spent at *Ricepaper*, as well as the type of task they needed to execute. Lastly, the highest level of restricted access was placed on the departments that handled the bulk of confidential data—Finance, HR and Contracts. This level of password protection also applied to other internal information, such as advertising spreadsheets and circulation reports. Only the senior management and department heads had the authority to read, edit, and make new posts in these restricted sections.

The IT staff added this extra layer of precaution to prevent any unauthorised staff from maliciously altering another person's post, or randomly replacing an online document with their own version. The Production Manager, in particular, had been very concerned about limiting upload access in order to protect version control. As Carol Lee expressed:

> I am very careful about who we let into the depths of Ricepot. If a new intern unknowingly goes inside and accidentally changes a document version, that can seriously mess up our tracking cycles. The last thing we want is for someone to go online, take our content, and reuse it for an

[85] Lin, interview.

> unauthorised purpose. This is why our department heads need to approve any new volunteers before the IT staff grant them access to the forum.[86]

The above quote succinctly highlights the importance of protecting *Ricepaper*'s "data goldmine." Not all volunteers needed access to every level in the forum. It was vital to protect the deeper levels of the organisation, by choosing whom to let into Ricepot, and how they should use it. Through implementing the series of cautionary measures outlined above, *Ricepaper*'s IT staff and management secured greater control over its online community.

5.3 Creating a Digital Office Culture

Perhaps the biggest change brought about by Ricepot was the possibility of running the magazine by distance management. Ricepot geared the *Ricepaper* staff up to embrace a "digital workplace". In many ways, the forum functioned like a "virtual office" for the staff that were located outside the Vancouver area. It housed files and archives, and served as a meeting space for staff discussions. The majority of the art and production staff used the forum to talk about preliminary concepts before meeting in person to finalise the design for the entire magazine.

In an effort to promote "working via remote", *Ricepaper*'s management encouraged their staff to adopt other communication technology to complement their use of Ricepot. IT Manager Jonathan Lin indicated that this was not such a dramatic change for the staff, considering most of them were already accustomed to working in an environment that emphasised electronic reporting. Even before the advent of Ricepot, volunteers were already using creative methods to communicate with different kinds of technology: instant messaging, teleconferencing, and online whiteboard meetings. However, the crucial difference was this: Ricepot, when used together with these other communication tools, made it possible for staff to contribute to the magazine without stepping foot inside the office. In fact, the senior management conducted 100% of all their meetings via teleconferencing, while 45% of small group meetings took place online.

Alternative meeting solutions did not replace in-person gatherings entirely. However, they did transform the manner in how the staff got their work done. By using Ricepot in companion with other communications technology, staff created a previously unprecedented working flexibility in each department. The editorial staff, for instance, were able to cut their meeting time by more than 75%! For other departments, the combined use of these technologies meant that they could run a good percentage of their operations online, or as the department heads called it, "via remote."

[86] Lee, interview.

Ricepot also helped to automate a good portion of the editorial submission process. Since the editors posted most of the text documents on the forum, this eliminated the need for editorial assistants to follow document trails in order to make sure everyone received the required information. As well, editors no longer had to meet in person, since they could conduct most of their discussions online. This did not just apply to the submission process, but the *entire* editorial cycle. At the peak of all online activity, the editorial staff accomplished approximately 85% of all editorial processes via remote! In comparison, 95% of the IT department operated online, 60% of Art and Production processes were distance-managed, while approximately 45% of Circulation, Advertising, and Marketing projects were handled via remote. Office Management, HR and Finance—the departments that dealt with the most confidential and proprietary data—had a less prominent online presence, operating only 25% of their activities on the web.[87]

This is not to say that these departments did not have the capacity to migrate more of their operations online. Rather, digital technology offered each department the option of modifying their working methods to suit their needs. The art and production staff, unlike editors, opted to manage a smaller portion of their activities online. This was because design was different from text, and tended to be more effective when viewed in person. Nevertheless, Ricepot still enabled the magazine to take about half of its operations online, a significant change that attests to the forum's efficacy.

5.4 Ricepot: An Interim or Permanent Solution?

While the Ricecooker CMS was originally conceptualised as an independent system, the introduction of Ricepot to the magazine brought about a new conundrum. Were the two systems separate entities, or could they be used as companion work tools? While our Users Committee—department heads, senior editors and designers—had always emphasised that Ricepot was only an interim solution, the staff—volunteers who ran the magazine's daily operations—saw otherwise. As the volunteers continued to use the forum, they started to view Ricepot as the decided content management system for the magazine. The positive impact of Ricepot created an unprecedented morale high in the organisation. As such, the staff were divided on the Users Committee's decision to continue redeveloping Ricecooker for a fourth time. Some were resistant to the plan, as they felt that Ricepot might only meet short-term needs, but "staff have managed to work around these limitations so far."[88] On the other hand,

[87] Uechi, interview.
[88] Wong, interview.

the Users Committee felt encouraged by Ricepot's unexpected success, and were even more determined to create a better CMS to ensure long-term stability.

Ricepaper, as a whole, is not resistant to change; the managers and staff all approved the initial decision to develop Ricecooker 2.0 and 3.0. Despite issues with the CMS, the staff managed to invent creative ways to use Ricepot to meet their short-term needs. More impressively, the magazine's volatile phase did not cause its volunteers to falter. Instead, the staff managed to rebuild, maintain, and even improve their production workflows. With this in mind, our senior management organised an organisation-wide poll on Ricepot.[89] The following question was posted on the forum: "Would the staff be willing to contribute their time and resources to the redevelopment of Ricecooker 4.0?"[90] Out of the 43 staff who participated in the poll, 60%—approximately 26 staff—voted in favour of continuing with the plan. Those who supported the redevelopment of Ricecooker 4.0 were relatively satisfied with Ricepot. They just wanted the opportunity to adopt a more sophisticated system that would be perfect for the magazine. Despite this, they still saw that Ricepot had many positive traits. Instead of tossing out the Ricepot system, they suggested, it could be beneficial to integrate the forum's best features with Ricecooker 3.0. Considering this, the Users Committee spent a month running both the systems through a simulated production cycle, using them separately and then simultaneously, in order to compare advantages and drawbacks.

5.5 The Cooker vs. The Pot: Independent or Interdependent?

The Users Committee assessed Ricecooker 3.0 and Ricepot with two goals in mind: 1) to evaluate the possibility of using Ricepot in companion with Ricecooker; 2) to determine if the functions in either system could make up for the limitations in their counterpart. The Committee also considered the possible duplications if the two systems were to be combined.

It was evident from the beginning that Ricecooker 3.0's automated tracking system and Ricepot's communications forum were the respective strengths of both systems. However, while it was possible for a forum Administrator to go into Ricepot to manually update the status of a particular project, it was not possible for an Administrator to go into Ricecooker and manually import the discussion of a project into the CMS. The Users Committee also found that using the CMS in companion with the forum was a tedious process. As an example, editorial users would log into Ricepot to view submission comments, and then switch back to Ricecooker to check the status for each article. If they only logged into one or the other system, they would not have the full picture of an article's progress. While all the staff could update both systems regularly, this

[89] This was conducted using the "Poll" function that came with phpBB. The poll took place in March 2005, after approximately seven months of using Ricepot.

[90] *Ricepaper* Magazine, Online Volunteer Poll, March 25, 2005.

would duplicate their tasks. Such an arrangement also had a greater potential for human error. The alternative—maintaining scattered information banks in both systems—would be time-consuming, and also defeat the purpose of having a central data repository. For the most part, staff felt inclined to consult one centralised system, rather than cross-reference between two systems for updates.

Table 1 provides a condensed comparison of Ricecooker 3.0 and Ricepot, based on the Users Committee's evaluation of both the systems.[91]

Ricecooker 3.0	Ricepot
Specifications and tracking reports are automatically generated. Individual staff can make updates directly in the system; updates will be cross-referenced with the relevant data.	Separate word documents tracking reports need to be manually created. Staff need to cross-compare data to produce relevant status reports.
Generates an "at-a-glance" list of status updates for all editorial articles.	No "at-a-glance" status list; information is buried within each discussion thread.
Central information repository. Archives of past issues are automatically created and stored on the server.	Central information repository. Archives of past issues need to be manually organised and stored in a "restricted access" section on the forum.
Rigid version control. The pre-programmed tracking system only allows users to store a limited number of document versions.	Flexible version control. The forum allows users to store an unlimited number of document versions.
Different access levels need to be manually programmed into the system by IT staff.	Permission settings can be set by department heads in the built-in administration panel.
CMS follows fixed step-by-step functions; does not allow users to reorder processes.	Forum does not follow a rigid order; users can make changes according to their needs.
No administrative flexibility; new sections cannot be created at will, massive re-programming required.	Allows for administrative flexibility; new departments can be added by adjusting settings in the administration panel.
"Comments" function only displays the most recent feedback; system does not track the date or user who makes the comment.	A history of all comments, their origin, and profiles of their users are displayed.
No forum; does not allow public and staff to interact with one another. Cannot accommodate an online community.	Forum promotes fluid communication between staff and reading public. Helps build a vibrant virtual community.
Staff cannot communicate privately with one another.	"Personal Message" function allows users to communicate privately with one another.
Limits interdepartmental communication; repositions all decision-making in the hands of the core staff.	Transparent communications forum supports interdepartmental interaction; all staff can participate in decision-making process.
Flat plans are automatically created.	Flat plans need to be manually drawn and posted on the forum.

Table 1. Ricecooker vs. Ricepot: A Comparison at a Glance

[91] Users Committee, "Ricecooker 3.0 and Ricepot: An Evaluation by the Users Committee" (Internal report, *Riccpapcr*, April 2005).

As can be seen, both Ricecooker 3.0 and Ricepot had their inherent benefits and drawbacks. In evaluating both systems, it is important to prioritise the various functions that each can offer. On the surface, Ricecooker 3.0 would appear to be an ideal system for a volunteer-based magazine: the system could automatically generate updates and cross-reference data, thereby cutting the labour spent on compiling update reports. Upon further consideration though, this was not a critical need for *Ricepaper.* If fact, weren't the benefits of Ricepot an indication of what the magazine really needed—an expandable system that was simple to administrate, and facilitated transparent communications? Either way, Ricecooker 3.0 failed to deliver on all counts. The CMS had an inflexible structure that would require IT staff to spend hours of programming just to add a new section or function to the system. There was also no guarantee that these additions would be able to work with seamlessly with the rest of the CMS. Furthermore, Ricecooker 3.0's greatest drawback was the fact that it limited communication between the departments. For example, in order to view someone's opinion on an article, the staff had to shift through three layers of data—an article matrix, various document versions, layout specifications—before they could download the file to read the embedded comments. Comment functions were buried so deep within the system that they were more likely to obstruct, rather than facilitate interaction. It would take a major overhaul of the system's infrastructure in order to correct this issue.

Ricepot on the other hand, was a flexible and efficient communications engine. As detailed in section 5.1.1, the forum increased the transparency of the magazine's operations, and changed the way in which the staff worked with one another. More importantly, this new level of open communication facilitated a steady exchange of ideas—allowing staff to pass on knowledge, increase their level of participation, and ensure continuity and stability in the magazine's processes. This was a vital step towards retaining operating knowledge within *Ricepaper*, and ensuring that the subsequent departure of staff would not bring the magazine's workflow to a standstill. Moreover, the staff had become less dependent on the IT team because they could set permissions and maintain the forum on their own. Ricepot's highly flexible structure also made it possible to customise the system in accordance with the magazine's constant growth.

A quick look at the chart above shows that of the 12 critical needs under comparison, Ricepot fulfilled nine, whereas Ricecooker 3.0 only fulfilled four. The forum's only limitation was the need for staff to track all files and workflow manually. Despite this, the staff had already used it to monitor their processes efficiently for seven months!

Ricepaper's previously opaque communication methods had isolated the departments from one another, and depended too much on one individual source of information. This block between the staff's communication was a primary cause for the disintegration of the magazine's operating processes two years ago. Given this experience, it should have been more important to maintain transparency within *Ricepaper*, and less critical to automate its processes. After all, while Ricecooker 3.0 could fulfil an idealised want, it failed to meet the staff's needs the way Ricepot could. However, the Users Committee had a different way of evaluating the situation.

Instead of dismissing Ricecooker 3.0 completely, the Users Committee felt that the CMS could be utilised as a stand-alone system—if the magazine was willing to consider some drastic changes. Workflows needed readjustment to fit the system's rigid, pre-programmed functions. Everyone had to be more disciplined about storing multiple document versions on the system. As well, departments would have to rely strongly on alternative technologies to communicate with one another.

Despite their relatively lenient views of Ricecooker 3.0, the Users Committee had a much harsher assessment of Ricepot. While the staff already used the forum as a tracking system, the software was inefficient and labour-intensive for this purpose. In a final report by the Users Committee, Ricepot was evaluated as a "system that worked, but only because of the resourceful nature of *Ricepaper*'s staff, who had constantly found new ways to utilise a traditional forum".[92] They concluded that the energy the staff had spent on working around Ricepot's limitations could eventually result in volunteer burnout.

Ricepot did require the staff to make adaptations to their workflow. However, these adjustments were not so different from the previous changes that they made whenever the magazine adopted another reincarnation of Ricecooker. Each time, the staff needed to adopt labour-intensive solutions to make up for the deficiencies in the trial system. With Ricepot, the greatest challenge was in workflow tracking. Since the forum could not automatically generate status lists, the production staff had to go into every discussion thread in order to view updates, and compile status reports. By staggering its volunteers into separate production shifts, *Ricepaper* had been able to conduct constant status checks, while protecting its staff from overwork. This type of labour-consuming work method would have been impossible if the magazine did not have a large pool of staff. Nevertheless, as Jim Wong-Chu once said, "The one resource that we do have is human labour. We may not have the financial means to purchase ideal equipment, but we can put more labour into achieving similar results".[93] This is a telling comment, as it suggests that our senior management was aware that the staff could work around the limitations of Ricepot with no real threat of burnout.

However, the Users Committee thought otherwise, and advised the senior management and staff that it was in *Ricepaper*'s best interests to continue with redeveloping Ricecooker 4.0. They deemed that Ricepot functioned best as a communications system, and not as a CMS. In their eyes, a communications system promoted discussions and interaction amongst its users.

[92] Users Committee, "Ricecooker 3.0 and Ricepot: An Evaluation by the Users Committee."
[93] Wong-Chu, interview.

On the other hand, a content management tool had to meet five criteria: 1) Monitor tracking processes, 2) cross-reference relevant information, 3) circulate status reports, 4) store all files centrally, and 5) protect confidential data with customisable access levels. Given that Ricepot did perform or facilitate all of the above functions, it was ironic that the Users Committee was reluctant to acknowledge it as an effective CMS. As a central data repository, the forum saved time: editorial and production assistants no longer had to track document trails, and staff did not have to wait for an intermediary person to forward the required documents to their in-box. The transparent communications system also eliminated the need for "middle-man" jobs—anyone could post and distribute documents directly on the forum, allowing the production staff to monitor the submission process directly online. By discussing and interacting with key-decision makers, volunteers felt a greater sense of involvement. They were motivated to take ownership for their projects, thus increasing accountability within the organisation. As can be seen, while the IT staff did not develop Ricepot as a CMS, the forum could still cater to *Ricepaper*'s content management needs.

The Users Committee's negative views towards Ricepot were partially due to their belief that content management technology needs to be specially customised. As well, the team was unwilling to give up the Ricecooker project after investing almost two years in the development process. In their mind, "it was obvious that Ricepot could not be the final product of all that development work. The forum marked the beginnings of free-flowing information, but its benefits were only a glimpse into the possibilities that could be offered by a *true* CMS".[94]

After their experience with Ricecooker 1.0, the Users Committee had negative views towards working around software limitations. As well, they felt disappointed by Ricepot. Though the forum met most of the magazine's needs, its interface and concept were nothing like the ideal CMS that the Users Committee had originally envisioned. They felt it was in *Ricepaper*'s interests to pursue the "best possible system, designed especially for magazine publishing".[95] Ricepot, they pointed out, was built from an open-source bulletin board package which anyone could access off the internet. As such, it would ultimately be inferior to Ricecooker 4.0, which would be a customised CMS built specifically to accommodate the idiosyncrasies of a magazine publishing environment. It was on these grounds that the Users Committee pushed for the redevelopment of Ricecooker 4.0.

[94] Siu, interview.

[95] Users Committee, "Ricecooker 3.0 and Ricepot: An Evaluation by the Users Committee."

5.6 Assessing the Pot

Was Ricepot truly problematic as a content management solution? No. The implementation of the forum brought about greater stability to the magazine's work cycle, and gave the staff greater confidence in using technology to solve their content management problems.

Chapter 3 described how *Ricepaper*'s work cycle had disintegrated when the sudden departure of its previous Editor-in-Chief shook the magazine's production and editorial processes. Ricepot helped the magazine to rebuild these workflows by facilitating regular inter-department communications. Having stabilised production work cycles made it possible for the magazine to deliver their product to subscribers and newsstands on time. This increased regularity marked many positive changes for *Ricepaper*. Previous advertisers returned, more distributors were willing to carry the magazine, subscriptions increased, and new staff volunteered their skills.[96] The magazine was in a position to plan and conduct timely marketing campaigns for each of its forthcoming issues, as the staff could now hold to their publication schedules. This brought new promotion partners on board, thereby widening *Ricepaper*'s sphere of influence. As can be seen, an effective content management system can affect a magazine's entire workflow, thereby influencing its health and wellbeing.

As of the time of writing this report in August 2005, four months have passed since the Users Committee made the decision to redevelop Ricecooker 4.0. During this time, the staff have continued to use Ricepot as an interim work tool until the new CMS is completed. The development period for Ricecooker 4.0 is tentatively scheduled for completion in December 2005. At this point, however, the Users Committee is still in the pre-development research stage, and has yet to map out any concrete system plans. It is still too early to say whether Ricecooker 4.0 will meet all the staff's expectations, or if this will be even our final CMS project. It is possible that new system could be a success. However, there is also the ever-looming question: what if Ricecooker 4.0 turns out just like its predecessors? The staff could continue redeveloping further versions—Ricecooker 5.0, 6.0. 7.0—without ever creating their ideal CMS. Considering that Ricepot already functions seamlessly with the magazine's workflow, is it necessary for the staff to expend resources on redeveloping the system yet again?

As the magazine's Ricecooker experience demonstrates, building a home-brewed content management solution is a process of trial and error. Sometimes, when the development process fails, it can be particularly hard to let go of something that the staff have painstakingly worked on for an extended time. It is not always easy for volunteers to embrace new

[96] Approximately 15% of previous advertisers have returned, subscriptions have doubled, two new distributors have taken on the magazine, and 20 new staff have joined *Ricepaper* in the last year.

technology. This is especially the case when the new software does not fit their view of an ideal work solution.

At times, the most suitable CMS solution is the one that evolves naturally. In *Ricepaper*'s case, this alternative is Ricepot, a piece of software that does not resemble the Users Committee's original development plans. However, they overlook this solution for various reasons: 1) They did not play an active part in developing the technology; 2) they assume that the solution is inferior to their initial plans; 3) they expect that they need complex and sophisticated technology in order to meet their requirements. In many ways, Ricepot is the anti-thesis to all the above. The forum is successful as a content management tool, not because the *Ricepaper* staff designed it for this purpose, but due to the creative and flexible ways in which they have utilised the technology. As Mark Armentrout, an expert in IT management, has identified, "competitive advantage is not a result of the type of technologies selected, but dependent on how the technologies are used".[97]

As the Users Committee continues their redevelopment of Ricecooker 4.0, I believe that it is necessary for the staff to realign their definition of content management back to people and processes. During the development process, the Users Committee may focus solely on designing a CMS to specification, and lose sight of how their final system needs to accommodate their users. With hindsight, it is easy to see that the Users Committee could benefit from having a broader and more flexible approach towards CMS design. As Jim Wong-Chu states astutely, "The best solution may not always look like the way that you expected it to, but that does not mean it will not work".[98] Instead of focussing on Ricepot's few limitations, the staff need to take pride in what they have accomplished with the system so far. Only then, will they be able to re-evaluate the value of Ricepot, and use it to inform their future development efforts.

[97] Kimball Fisher and Mareen Duncan Fisher, *The Distance Manager: A Hands-on Guide to Managing off-Site Employees and Virtual Teams* (New York: McGraw-Hill, 2001).
[98] Wong-Chu, interview.

CHAPTER 6. Conclusion

This report represents a collaborative effort between *Ricepaper*'s management, IT staff, and VFS to effect change in the magazine's operations through the development and implementation of content management technology. This report provides not only the first comprehensive documentation of *Ricepaper*, but also explores the process of developing a home-brewed content management solution. This concluding chapter will link *Ricepaper*'s experience back to Chapter 1's discussion about the relationship between magazines, their communities, and CMS technology. It also provides some final reflections on how the Ricecooker development process has influenced staff and transformed the organisation.

The process of designing and developing a CMS is one of constant negotiation, with a host of uncontrollable circumstances. This is especially the case in a small magazine with a constant influx of volunteer staff. Processes sometimes change, ever so slightly, but just enough to require major changes to be made to the CMS. This is the primary reason why organisational self-awareness is critical when considering such a project. As a relatively young publication, *Ricepaper* had a limited view of its operations and the evolving nature of its organisation. While the magazine gradually did acquire some organisational-awareness, this was hard earned—through three rounds of CMS development, vigorous testing, and research. To date, the *Ricepaper* staff have already spent 21 months working on the project—an estimated total of 800 hours on the combined design and implementation of Ricecooker and Ricepot. At a hypothetical wage of $20.00 an hour, this would mean that *Ricepaper* has invested a total of $16,000.00 on the project so far.

6.1 The Need to Manage Content: A Catalyst for Change

A notable case in point is *Ricepaper*'s challenging situation all through the CMS development process. As a small non-profit magazine, *Ricepaper*'s search for a content management solution would already have been constrained by many factors: a constantly evolving staff, processes, and limited resources. In addition, the magazine tried to develop and experiment with content management technology during what could be considered one of the most volatile periods of its existence. It is important to mention that the magazine was not just undergoing a gradual and guided organisation change; the sudden departure of its previous Editor-in-Chief had affected the entire publication, and even brought the production process to a standstill. Although the magazine had already been in operation for seven years, this transition pushed the publication back into an infantile period in its life cycle. The volunteers had to account for delays in the production cycle, while additional human resources had to be recruited, old positions replaced, and everyone had to figure out new ways to accomplish previous processes.

As such, the new management struggled to balance the needs of basic survival with the long-term vision for operational enhancement. As Jim Wong-Chu explains, "Back then, we were always in crisis mode. Our management was always buried in the day-to-day task of saving the publication. There was little time for them to lift up their heads, take a step back, and direct the entire vision of the magazine".[99] *Ricepaper* was not in the most ideal situation when it first decided to develop its own CMS. At the same time, the need to provide a framework for the reconstruction process was the reason why *Ricepaper* required a content management solution. In fact, this urgent need for a CMS was a catalyst that pushed *Ricepaper*'s staff to analyse their processes and address the issues at the root of the problem.

6.2 Finding Organisational Awareness

In many ways, *Ricepaper*'s four-round development experience was an exercise in fostering organisational self-awareness. Every reincarnation of the CMS project was a process of introspection, as each system compelled the staff to adjust their workflows and learn more about inter-departmental processes. The fact that the staff found new ways to work around the limitations every time is a good indication that they had learnt how to accommodate the magazine's evolving operations. Through *Ricepaper*'s development experience, it has become evident that the adoption of new CMS technology is most effective when an organisation is mature, aware of its needs, and has a stable environment that is able to accept transformation. Without these pre-requisites, there is a possibility that the resulting changes will disintegrate into chaos, just as *Ricepaper*'s experience with Ricecooker 1.0 demonstrates. Fortunately for the magazine, its management had the judgement, and its staff had the patience, to treat its various shortfalls as learning opportunities instead of absolute failures.

The magazine's partnership with VFS is a good example of this; the development process made it necessary for staff to map out workflows and organisation charts, in order to

[99] Wong-Chu, interview.

give their partners a visual representation of how they *thought* their magazine operated. Despite this, the VFS team still constructed Ricecooker 2.0, a CMS that did not fit the magazine's processes. It was only by analysing and modifying the system that *Ricepaper*'s staff came to realise their own mistake. Throughout the development process, they had not maintained close communication with the VFS team. As well, they had asked VFS to design an almost impossible system—because the documents they had provided depicted an idealised view of the magazine, and not actual problems. Instead of showing the VFS team *Ricepaper*'s fluid operations, the staff had carefully constructed a fairytale magazine with tidy departments and systematic processes. This was due to two reasons: 1) The lack of organisational transparency often made it difficult for staff to see their magazine as it really was; 2) The volunteers were hesitant to reveal the deficiencies in an organisation that they worked so hard to re-build.

Given the unrealistic scope of *Ricepaper*'s requirements, the VFS team had a mammoth CMS development project on their hands. With hindsight, it is amazing to think that they were able to develop Ricecooker 2.0 in just six months. Without the VFS team's efforts, it is unlikely that *Ricepaper* would have gotten this far in their CMS development. While the system that VFS developed might not have met *Ricepaper*'s needs, Ricecooker 2.0 provided a critical foundation for the design of future content management tools. As well, the requirements gathering and development process gave the *Ricepaper* staff a better comprehension of the magazine's evolving needs, resulting in better staff consciousness, thus empowering them to rebuild the magazine and continue the challenge of developing their CMS.

6.3 Strengthening the *Ricepaper* Community

Through developing a CMS, the volunteers learnt a lot more together than they could ever have learned alone. Participating in the collaborative sharing of knowledge has been a satisfying experience for the staff, teaching them how to pool their skills and resources together, thereby strengthening the collective working power of the staff community.

In particular, the creation of Ricepot centralised information and brought greater transparency to the whole organisation. Entire operations and reporting protocols were laid bare, promoting a "flat" organisational culture by introducing a new level of equality. Today, the magazine's decision-making method has become a more democratic process. Interns, as well as department heads have the opportunity to contribute to *Ricepaper*'s operating plans, and voice their opinions about improvement strategies. Senior managers interact directly with junior assistants instead of through middle managers. Overall, Ricepot fostered a stronger sense of communal ownership, thus improving staff morale and creating a more cohesive workforce. This positive change has also increased *Ricepaper*'s appeal, helping to draw new staff and retain the organisation's professionally trained volunteers—editors, Writers, designers, IT Developers, Marketing staff—thereby strengthening the *Ricepaper*'s pool of skilled labour.

Most importantly, the advent of Ricepot has initiated an ongoing dialogue between the magazine and its readers. Previously, the staff had minimal contact with readers. Ricepot has changed this by humanising the interaction between the staff and reading communities, drawing together all the people involved with the magazine. This has enabled *Ricepaper* to deepen its relationship with the public, and accommodate their ever-changing reading tastes. To illustrate, readers recently sent in many requests for more content on Asian arts and culture trends. In response, *Ricepaper* added "J-Pop", a new editorial department on popular Asian art, to the magazine. Ever since the introduction of this section, more producers and publicists have contacted the magazine with requests to be featured in this department. The magazine's latest subscription campaign brought in a 12% increase in gift subscriptions from existing subscribers. On the accompanying survey form, about half indicated that their interest in *Ricepaper* had increased due to the magazine's coverage of popular art. One subscriber wrote, "Thank you for J-Pop. The new content tells me you are truly listening to my suggestions. I am glad to see more faces like my own represented in your magazine."[100] By improving its product, *Ricepaper* managed to raise its profile within the arts community, attracting industry interest and new subscribers, thereby expanding its sphere of influence.

6.4 Utopia vs. Reality

As the previous sections describe, *Ricepaper*'s CMS development efforts have brought about many positive changes to date. Despite this, the Users Committee still has its eye on continuing with the redevelopment of Ricecooker 4.0. In evaluating the entire Ricecooker development process, it is necessary to question this single-minded quest for the perfect CMS. After having invested so much time and work in the project, it is understandable that the staff want to create the best possible system they can. However, the process of building a work tool should not be a gift from a small group of managers and designers, but involve the participation and influence of all staff.[101]

A CMS has to serve a community of users, not only the needs of a small group of individuals within that community. It may not have been the wisest choice for *Ricepaper* to focus the entire responsibility of developing a CMS on one group of individuals. As a collective, the Users Committee has the authority to direct the Ricecooker project and make related decisions on behalf of all the staff. However, they are not necessarily representative of the entire

[100] Amelia Chua, letter to the editor, April 21, 2005.
[101] Ehn, *Work-Oriented Design of Computer Artifacts*, 270.

end-users community. Of the 15 members, eight are department heads, and seven are from the APE departments. While everyone is welcome to join the Users Committee, new members have to agree to remain with the project for at least a year. As such, while the turnover rate for the Committee is relatively low, it is difficult to attract more staff to the project. This unbalanced decision-making body is the reason why Ricecooker was designed largely from a middle-management perspective. The Users Committee felt strongly about having a system with built-in automated tracking capabilities because that function would minimise their workloads as information coordinators.

As middle managers, the Users Committee also had a different level of commitment from the junior volunteers. Many of the staff on the Users Committee had supervisory roles within the magazine and did not have to use the CMS on a daily basis. For instance, they only use Ricepot to coordinate workflows and monitor updates three times a week, since the junior staff handle most files and data directly. In comparison, the junior staff use Ricepot almost daily, as they have to communicate with staff from all levels of the magazine to obtain the correct data. As well, many staff on the Users Committee have been with the magazine for almost two years and intend to stay on longer. Unlike the junior staff that might only remain with *Ricepaper* for a few months, these middle managers will be around to witness the future results from the redeveloped CMS. This is the reason behind their different opinions towards Ricepot: the Users Committee wanted a system that would bring long-term benefits, while the other volunteers wanted a system that they could use immediately.

Many realistic factors affect the development of a CMS—financial resources, labour, time—these all determine the scope and extent of the final system. As a non-profit magazine, *Ricepaper*'s main priority is to meet production schedules, and ensure the timely delivery of its publication to distributors, subscribers, and advertisers. For this purpose, the magazine needs a basic CMS, not a complex piece of software. Currently, Ricepot does function as a CMS. It already facilitates critical workflow processes—data exchange, status tracking, interdepartmental communications—and ensures transparency and fluidity between the staff. Why, then, is the Users Committee so determined to push forward with the redevelopment of a new CMS, ignoring the efficacy of Ricepot? The answer lies in the gap between their idealised CMS and their disappointment with what the current forum can offer. The effort to redevelop Ricecooker 4.0 is an attempt to create a CMS that will bridge the gap between what they envisioned, and what they are able to attain. In short, Ricecooker 4.0 represents another opportunity to pursue "content management utopia".

Participation in idealised redesign enables the participants to raise their intrinsic and extrinsic values, especially their ideals, to consciousness. This type of idealised process tends to invoke the curiosity of participants and usually generates excitement because it expands their conception of what is possible.[102] Idealised redesign gives the Users Committee the opportunity to imagine and create, and these activities are challenging and fun. This type of idealised reflection has certain advantages, especially in that it motivates the staff to develop the best possible system to improve existing workflows. However, in pursuing the ideal, they might lose sight of the final goal—building a CMS that meets realistic needs. As Chapter 5 illustrated, the Users Committee was so focussed on creating a perfect system that they overlooked other alternatives that could solve *Ricepaper's* content management problems.

6.5 The Importance of Participatory Design

Change is difficult. Change is especially difficult when it involves over 60 different people who are constantly evolving—adopting different skill sets, working and moving fluidly between roles, changing departments and processes. Throughout *Ricepaper's* transition period, most of the staff were busy rebuilding workflows. Even though they urgently needed a content management solution, magazine production had to continue. Given their limited finances and time, the staff could not afford to drop their tasks to concentrate on developing a CMS. Instead, they had to depend on a small group of management-appointed representatives—the Users Committee—and trust them to develop an appropriate CMS. While *Ricepaper's* CMS development did involve its staff, they were only able to adopt what I call "partial participatory design", since a small group of users was designing for the needs of a larger user community.

As Carol Lee acknowledges, "getting effective results from technology depends on how humans use the system".[103] Creating a good CMS is tightly linked with production processes, and may require users to retool their workflows to accommodate the new system. This is why it is important for all staff to be involved in the development and decision-making stages, especially if they are going to be using the CMS at the very end. End users should have knowledge of how the CMS can benefit the *whole* organisation, not just a limited view of how the technology can affect specific tasks or individual departments. Many oversights occurred during the Ricecooker project because the majority of the staff were not involved throughout the development. Their lack of direct involvement inadvertently created a gap between what was needed and what was designed.

For instance, many of the staff had initially felt that Ricepot met their requirements, and were reluctant to continue with the redevelopment of Ricecooker 4.0. This changed after attending a Users Committee information session. 60% of the volunteers voted in favour of continuing with the redevelopment project, largely because they were given information that

[102] Ibid, 190.
[103] Lee, interview.

was heavily influenced by the Users Committee's point of view. According to one volunteer, "we were satisfied with Ricepot, but didn't feel comfortable about voicing this as we were not involved in developing Ricecooker. We felt persuaded to consider *Ricepaper*'s long term interests—though to be honest, many of us had no idea what these were".[104] This is not to say that the Users Committee did a poor job of developing Ricecooker. In fact, many of the department heads on the Committee had to run their own departments in addition to their development duties. In light of *Ricepaper*'s busy transition period, it is understandable why the Users Committee overlooked certain aspects of the CMS development process.

While their planning for each version of Ricecooker involved people from all departments, they did not include volunteers from all levels of the organisation. With the exception of a few volunteers from the production team, no junior-level staff participated in the development and testing stages of Ricecooker. Unlike senior and mid-level management, they did not have an overview of how the system affected each department. All they saw was how the CMS created more work with each transition phase.

Pelle Ehn believes that it is necessary to consider both the technology and the intended user during the design process. To quote him, "In designing the production system one has to investigate both the technical system and the social system and their interrelations on work group level".[105] However, *Ricepaper*'s entire CMS project was mainly focused on designing the software. During the development of Ricecooker 2.0, the Users Committee spent most of their attention on gathering requirements and ensuring that the VFS team designed according to specification. With Ricecooker 3.0, their focus shifted to usability testing and debugging the system. Throughout these processes there remained little time for them to concentrate on the relationship between the CMS and its ever-changing users. As a result, the *Ricepaper* staff could not relate the work they performed to the final system.

6.6 Some Final Reflections

As I have highlighted throughout this report, intersecting processes need to be fluid, and people need to be aware of their roles in the overall workflow. I now venture to say that this level of transparency is also necessary throughout the development process of a CMS. All staff—senior management, department heads, project coordinators and interns—should be informed about the design of a system that will influence every level of the organisation.

As this case study of *Ricepaper* demonstrates, developing a CMS is a process-oriented project, much like magazine production. The entire development process is divided into

[104] Wong, interview.
[105] Ehn, *Work-Oriented Design of Computer Artifacts*, 261.

stages—requirements analysis, programming, usability testing, de-bugging, implementation—all involving people in different or overlapping roles. Sometimes these people may have conflicting viewpoints, much like the creative process in content production. A programmer might interpret a set of requirements differently from their client or the Project Leaders, and end-users might clash over what they need from the final system. This is why staff participation and interaction is necessary. Communication can help all parties understand the organisation's overall needs and facilitate the translation of these requirements into user-oriented functions.

As the Users Committee continues with the next stage of development and implementation, it is vital to ensure that proper knowledge translation and staff involvement are encouraged. Rather than everyone learning about the technology on an "as needed" basis, *all* staff who will be using the CMS should be educated about the development process. This way the staff will ideally play a larger role in creating the system that they will ultimately use.

However, I should acknowledge that the *Ricepaper* staff might only be able to contribute a limited amount of time and effort to the redevelopment project. In a volunteer magazine like *Ricepaper*, it is a challenge to ask volunteers to take on CMS development work in addition to their other duties. This is why the Users Committee continues to play a central role in the development process—as project coordinators and knowledge facilitators. Even if the volunteers can only participate in a limited manner, the Users Committee needs to update them on the development process and give these end-users an opportunity to provide their feedback. This way, these volunteers will have a better idea of where the organisation will be in a few months time and be prepared for what might be expected of them in the future stages.

Over the past two years, the process of experimenting with different content management solutions has caused erratic and unpredictable changes within the magazine. Despite their heavy workloads, the Users Committee and staff persisted with the CMS project, demonstrating an amazing level of dedication for a team of volunteers. The three reincarnations of Ricecooker and the birth of Ricepot are a salute to their labour of love.

Meanwhile, *Ricepaper* has plenty of growing pains to endure. The staff look forward to the day, which some believe could be as far as two years later, when Ricecooker will be fully developed and implemented. The path towards this goal will be challenging, and they will have to make many difficult decisions. Nevertheless, the staff remain optimistic. In the words of Jim Wong-Chu, "We have weathered every possible storm that might hit a small organisation like us. Our staff thrive on change. We won't just create Ricecooker; we will have our rice and eat it too".[106]

[106] Wong-Chu, interview.

APPENDICES

Appendix A. *Ricepaper*'s 13-Week Work-Cycle

Week	Editorial Editor-in-Chief & Section Editors	Art & Production Art Director & Prod. Manager	Advertising Ad. Manager & Sales Reps.	Circulation Cir. Manager & Subs. Manager.	Marketing/PR Marketing Manager	IT & Web IT Manager	Office/HR Office Manager	Finance Controller & Accountant
1	* Post-mortem meeting of previous issue; best practice notes taken. * Contributor copies mailed w/cheques & thank-you notes. * Sec. Eds. meet Ed-in-Chief to plan next issue. * Assess Sec. Eds.' logs of unsolicited subs & queries.	* Post-mortem meeting of previous issue; best practice notes taken. * Pre-production begins. * A. Dir. meets Ed-in-Chief to determine the look of next issue. * House ads planned & design begins. * Monitor editorial & art processes; ongoing.	* Post-mortem meeting of previous issue; ongoing ads noted. *Advertiser copies mailed w/ invoice & rate cards. * Ad Manager meets sales reps to set new ad targets. * Evaluate Sales Rep's logs of potential, new & ongoing advertisers.	* Post-mortem of previous issue; new circ targets established. * Finalize subscriber, ads, & contributors list. * Mags. sent to subscribers, associations & institutions. * Call CMPA to check if shipment received.	* Post-mortem on marketing initiatives for previous issue. * Meet w/ Circ. & Ad to determine marketing goals for this cycle. * Launch party guest list finalised, invitations sent. * Issue press release sent out; press release filed w/ notes.	* Meet with Prod, Marketing & Ads to determine new uses for web, & IT needs. * Details of new issue posted & updated; ongoing. * Launch press release posted; open invite sent to list-serve. * All partner sites contacted with updated issue info.	* Post-mortem; best practices noted. * Coordinate volunteers for launch events. * Work w/ Circ to coordinate office mail-out. * Archive hardcopy material from previous issue. * Space found to store previous issue.	* Issue cheque to printers for payment of previous issue.
2	* Editorial commissioned. * Deadlines set for all draft copy. * Writers work on outlines.	* Photo shoots planned. * Article outlines received. * Illustrators contracted.	*Ad sales. * Send renewal offers to advertisers in previous issues.	* Mags sent to secondary dist. * Enter renews. * Promo distribution planned.	* Launch details finalised. * Press release of latest issue sent.	* BCAMP site updated. * Databases checked. * Web stats sent to Marketing.	* Archives started. * Open call sent for subs.	* Collect revenues from pre-launch events.
3	* Unsolicited subs chosen & followed-up; rejections sent.	* Photographers contracted & photo shoots planned.	* Last-call-to-advertise release sent out. Follow-up calls.	* Contact CMPA & sec. dist. for report on previous issue.	* E-mail launch of new issue & retailers to stakeholder lists.	* Post call for submissions/contests for next issue on web.	* Staff contracts reviewed, workload assessed.	* Meet Heads to discuss income & expenses for current.
4	* Spreadsheet started w/ existing work & rough estimates.		* Payment reminders sent out to advertisers in previous issue.	* Finalise cir. chart for prev. issue; forecast current circ.	* Send out subs/contest call for next issue.	* Staff laptops updated with software & anti-virus.	* Meet Pub /EIC to discuss reassignment staff.	* Prepare quarterly financial report.

Week	Editorial Editor-in-Chief & Section Editors	Art Art Director & Prod. Manager	Advertising Ad. Manager & Sales Reps.	Circulation Cir. Manager & Subs. Manager.	Marketing Marketing Manager	IT/Web IT Manager	Office/HR Office Manager	Finance Controller & Accountant
5	* Drafts in, Sec Eds. vote & comment.	* Final ad deadline. All ads placed.	*Final ad booking deadline.	* Contact new dist. partners; ongoing.	* Distribution of issue to cult. centres, events & BCAMP prog.	* Archives cleaned, last issue burned to 3 DVDs.		* Quarterly report sent to Heads.
6	* Notes made—back to writers. * All editorial for issue selected, rest to bank.	* Rough illust. in, changes ordered. * Ready illust. scanned +prepped. * Ready photos sent for edit. * House ads done. * All photos+ final art in.	* Ad contracts mailed out. * Update database; recurring ads marked. * Prelim design/copy of house ads approved.	* Compile monthly circ. Report update. * Solicit subscription partners. * Renews sent to pending expiries.	* PR conducts surveys of previous issue. * Survey results updated to media kit. * Uncollected revenues noted & sent to Controller. * Media list updated.	* Pre-production systems adjusted. * House ads posted on web. * IT & Prod. discuss protocols.		* Compile report on all revenues & expenditures for last issue.
7	* Issue control sheet up-to-date.	* Photo-editing/art manipulation In progress.	* Ad contracts signed and returned.	* Update database with new subs.	* Plan promo swag for current issue.	* IT help staff on stand-by for production staff.	* Every 2 weeks: Bank run.	* Quarterly report circulated to Dept. Heads.
8	* Continue work w/ writers. * Copy-fit updated. * All material in from Sec. Eds.	* All editorial in. * Prod. starts. * Rough layouts started.	* House ads done. * Preliminary ad line-up sent to Prod. Manager. * Draft flat plan. * Ad reminders sent.	* Mid-cycle mail-out (to new subs) * Contact bookstores to replenish mags.	* Mail details for new issue. * Meet Heads to discuss launch. * Media list updated.	* Support production cycle, maintain systems, post news on web.	* Track staff recruitment. Update contact lists; e-mail regular updates.	
9	* Copy editing & proofreading. * All material to Art/layout. * Bumped writers. * Editorial Q&A. * Editor-in-Chief approves all layouts.	* All ads in, designs finalised. * Final ads placed on Ad line-up. * All layouts finalised. * Design Q&A. * Changes made. * 2nd-to-last Q&A. * Changes to final layout.	* Final Ad line-up submitted to Prod. Manager. Flat plan tweaked & finalized. * Art & Advertising meet to discuss ad environment. * Ads tweaked & ready.	* Collect revenues from distributors. * Print CMPA's circ report. * Conduct inventory of previous issues. * Archive previous issues.	* Monthly: Media clippings updated; quote file for promo book/mags; news/announce ment releases. * Swag for current issue finalised, ready for distribution.	* Bonus web ads selected & posted (noted in ad line-up). * Final web material to layout person * Updates given to web master.	* Ongoing: meetings; news on list-serve & office calendar; minutes taken, distributed & posted, filing; mail sorting; arrange staff's professional development.	* Ongoing: Record and process receipts, monthly staff invoices & department expenses.

Week	Editorial	Art	Advertising	Circulation	Marketing	IT/Web	Office/HR	Finance
	Editor-in-Chief & Section Editors	Art Director & Prod. Manager	Ad. Manager & Sales Reps.	Cir. Manager & Subs. Manager.	Marketing Manager	IT Manager	Office Manager	Controller & Accountant
10	* All final requests for next issue's review books/DVDs/CDs sent to office manager.	* Final Q&A. * Leftover layout work, photos and illust. (if any) to art archives.	* Verification of ads against ad line-up. * Missing ads placed; holes filled with house ads or trades.	* Payments for new/renewals (previous issue) collected. * Stuffing party set/ volunteer RSVP.	* Launch & pre-launch budget set. * Potential co-launch proposals drafted, co-partners contacted.	* All editorial versions & files archived on server. * All art files & proprietary images archived in Gallery.	* Once a year: Work with Publisher to compile budgets for grants; research potential new grants.	* Compile 3-year financial forecast; apply for GST returns.
11	* Meeting w/Sec. Eds. * Editorial bank checked, next issue rough planning.	* Regular proof printing & file preparing. * Proofs under strict version control. * Final proof.	* Ad payments for previous issue in, sent to Controller. * Uncollected accounts sent to Controller.	* Uncollected accounts noted & sent to Controller. * 2nd round of renewals sent to expired subs.	* Launch planned. * Pre-launch fundraising ready. * Issue press release written.	* All contributor & ad contracts scanned & archived electronically in server, under protected access.	* Order review books/DVDs for next issue. * Submit office expense & staff claims receipts.	* Note all uncollected accounts & map out collections strategy with indiv. depts.
12	* Longer term editorial commissioned. * Compile contributors report for current issue.	* Files finalized. * Files to printer. * Proof back from printer: Authorised. * To Print. * Production ends.	* Send ad revenue for current issue to Controller. * Invoices sent to clients in this issue.	* Send circ/sub revenue for previous issue to Controller. * Labels & renew letter ready before new mags. arrive.	* Launch party time/venue booked. * Party readers booked. * Launch invites printed.	* Server cleared of old data. * Proprietary archives marked & accessibility levels set.	* Monitor postage costs. * Coordinate w/ Controller on received revenues & outstanding accounts.	* Contributor cheques finalised & sent to Dept. Heads. * Organise & documents for tax reporting & grants.
13	* Planning for next issue continued.	* Magazines arrive in office. * Thank-you e-mail to all volunteers.	* Compile ad report. * Print advertiser thank-you notes.	* Mags arrive; stuffing party starts.	* Send anticipated fundraising revenue for current issue to Controller.	* New issue material laid out in web templates, & ready for upload.	* Update & circulate this chart!	* Consolidate collected revenues for departments in previous issue.

Appendix B. Sample Article Information Matrix

File Name	MargCho.ak.sg.jc.aw.FINAL.doc
Author Name (by line)	Alan Woo & Jessica Gin-Jade
Author Contact Info	1234@ricepaper.com
Author Bio Line	See body text.
Images (Y/N)	Yes.
# of Images	4
Images In?	Yes.
Photographer	Austin Young
department	Cover Feature
Word Count	2,601
Structural editor	Alexis Kienlen

editor's Initials	Type of Edit (Rewrite/Structural/Copy Edit)	Date
AK	Structural 1st Edit	June 28, 2005
SG	Structural look at after Alan's edits	July 6, 2005
JC	Approved for copy-editing, sent to author for sign-off	July 13 2005
AW	Author signed-off	July 15 2005
JC	Approved for layout	July 18 2005

editor's Initials	Suggested Titles	Suggested Subtitles	Pull quotes short	Pull quotes long	Sell Lines for Cover
JC	Margaret Cho: The High Priestess of Everyman's Cause	Remember that righteous hot bitch on a bar stool who wouldn't shut up? (Of course you do.) The funny one who yakked on about kissing women, fisting, and crystal meth? The good news is that she's publishing a controversial new book. The bad news? She's gotten married.	Receiving endless hate mail after slamming George W. Bush, Cho responded the way one might expect: she posted everyone's email addresses up for all to see.	"As you know by now, I'm Korean… I don't have a store or anything (well, not anymore)… I have very typically Korean features—I look like one of those girls that you would see in a Korean grocery store, on a calendar, holding a box of soy milk… 'Dreenk Vitasoy. Eees good for you!'" —from *HBO Comedy Half Hour: Margaret Cho* (1994)	Scratch Attack! MARGARET CHO targets Bush, and explains why she loves Canada, but will not live here.

Appendix C. The Relationship Between Production, Editorial and Art

editorial & Advertising	Production	Art Direction
* Editorial brainstorm session; potential articles listed, editorial bank checked. * Conceptual meeting; Art + Production + Ad Heads conceptualise ideas for upcoming issue.	* Production Manager attends brainstorm session and conceptual meeting.	* Art Director attends brainstorm session and conceptual meeting.
* editorial Submissions (rolling basis); articles enter editorial Board. * Each editor reads and assesses articles. * All commented articles assessed by Editor-in-Chief who draws up the final article line-up. * All articles MUST be in by the submission deadline, by which editors should send a copy of all articles to production assistants. * Ad booking deadline.	* On-going: production assistants contact editors and designers on a weekly basis; track files and save them on server. * If no reply is received from either art or editorial after one week, production assistants MUST notify department heads.	* Illustrators/Photographers/designers MUST produce mock-ups and send these to the Art Director before proceeding with layout. * Upon approval, a copy of the final design MUST be sent to production assistants. * Where an article may not be ready until the editorial deadline, editors must notify the photo editor if a photo shoot is required. * Artwork and photo shoots begins.
* Advertising MUST inform Art of any design work that might be required for house ads.		*Design for 3rd party & house ads begins.
* Editor-in-Chief, Art Director, Production Manager meet to draft flat plan. * Editor-in-Chief MUST inform Production Manager of any delays with articles.	* Production assistants prepare advertising submissions for layout.	* Designs for all house ads are completed. * 3rd parties contacted for images after the finalised line-up.
* Structural and Tonal editing, Fact-checking (1 month). * All advertising files submitted.	* Production assistants check if article files are available on the server/office; prepare all articles and art files ready for layout.	* Article drafts sent to designers, who contact editors for layout suggestions. * designers receive 3rd party ad files.
* Copy-Editing Round 1 (4 days).	* After structural editing is complete, production checks matrix information and make warning flags (e.g. problems with writer, article subject to major revamp).	* Draft layouts done. * All approved layouts MUST be packaged and sent to Art Director, who will compile all layouts.
* Copy-Editing Round 2 (4 days). * Editor-in-Chief approves layouts.	* Sign-off., Pre-flight check; files to printer.	* Changes to all layouts submitted. * Text re-flowed (if necessary). * Art Director approves layouts.

Appendix D. Timeline of *Ricepaper's* CMS Development Process

Oct 2002	*Ricepaper* staff initiate the development of Ricecooker 1.0
Mid-Oct 2002	Development begins on Ricecooker 1.0 web site.
Mid-Nov 2002	Users Committee created.
Nov 2002 to Jan 2003	Concurrent implementation & usability testing of Ricecooker 1.0 (13-week period).
Feb 2003	Staff realize limitations of Ricecooker 1.0 after using the site for four months.
Mar 2003	Staff vote for a complete overhaul of Ricecooker 1.0
Apr 2003	Author appointed Editor-in-Chief of *Ricepaper.*
Jul 2003	RFDP- sent out to 20 instructors and professors
1 Sep 2003	Work begins on refining Ricecooker 2.0's requirements
25 Sep 2003	The refined RFDP requirements are refined and presented to the *Ricepaper* Users Committee
1 Oct 2003	VFS begins the development of Ricecooker 2.0.
30 Dec 2003	Original completion date for Ricecooker 2.0; VFS requests more development time.
19 Feb 2004	VFS completes Ricecooker 2.0. The CMS is shown to the Users Committee.
End of Mar 2004	Ricecooker 2.0 deployed.
Apr 2004	Usability testing begins on Ricecooker2.0.
Mid-May 2004	Usability testing of Ricecooker 2.0 complete; identification of bugs in the CMS.

End of May 2004	Users Committee meet with *Ricepaper's* senior management to strategize methods to salvage Ricecooker 2.0; a four-step strategy is devised.
1 Jun 2004	Debugging of Ricecooker 2.0 commences; the CMS's features are streamlined to create Ricecooker 3.0.
	A basic Ricepot discussion forum is created for the purposes of exchanging debugging and redevelopment experiences.
7 Jun 2004	Users committee start using Ricepot to test and discuss the redevelopment of Ricecooker 3.0.
Jul and Aug 2004	Users Committee start using Ricepot to discuss the work for the art, production, and editorial (APE) departments.
End of Aug 2004	Debugging process completed, Ricecooker 3.0 is bug free. However, Users Committee realizes the system has many inherent limitations. Senior management assesses Ricecooker 3.0, and decides that it cannot be used. Decision made to continue redeveloping Ricecooker 3.0 into Ricecooker 4.0.
Sept 2004	Decision made to use Ricepot as an interim solution to fill the vacuum created by Ricecooker3.0.
Mid-Sept 2004	Migration of magazine operations begins—each department is gradually moved on to Ricepot.
Mid-Dec 2004	Online migration of magazine operations complete. All eight departments use Ricepot in varying degrees.
Mar 2005	Poll conducted on Ricepot. 60% of the staff voted in favour of continuing the plan to redevelop Ricecooker 4.0.
Mid-Dec 2004 to present	*Ricepaper* staff continue using Ricepot as an interim solution. Redevelopment work continues on Ricecooker 4.0.

Appendix E. Ricecooker 2.0 Needs and Wants Check Sheet

Basic Required Features

1. Data input templates

Data flat files

Separate web-based data input templates are required for the following flat data files:

- Articles in progress
- Magazine printing spec sheet
- Creators
- Contacts (Marketing, Institutions, Agents, Suppliers, Distributors, Rights)
- Writers' creators' status
- Per-Issue editorial line-up
- Per-Issue production details
- Rights status
- Sales tip sheets

Publication schedule/progress tracking checklists

Web-based tracking checklists are required for monitoring the progress of title projects. While the same format might be used for the tracking checklist for each issue, every imprint should have a tracking checklist that is entirely separate from the other issues. For instance, while Issue 9.1 and Issue 9.2 might be able to use the same tracking checklist input form; the data on these checklists should be categorised by issues. Hence, a search for Issue 9.1's checklists will not turn up Issue 9.2's checklists. These checklists would allow project administrators to indicate the status of individual issues by clicking check boxes to indicate the completion of selected tasks. As well, the checklists should let project administrators input dates and status notes on every individual project.

Publication schedule/progress tracking checklists (continued)

Tracking checklists would be made relational with the above data flat files. For instance, a user viewing a list of "Issues in progress" should be able to click on individual articles to view each article's status and publication schedule. The following is a list of required tracking checklists:

- Publication/editorial calendar
- Proposal tracking/Titles in progress

2. Database administration management

Separate accounts should be created for each *Ricepaper* staff department (i.e. editorial, Production, and Administration). Specific permissions should be set for each account, thus only permitting users to use data for certain purposes. For instance, an editor's account could be set to allow them to input editorial data into the article flat file. However, the same editor would be granted "read only" access to data in the production spec sheets.

The database should also be programmed such that certain users are only authorized to change specified data fields, and not all the data within a flat file. This requires the database to have a separate web based "accounts management" section that allows one database administrator to set permissions for all staff accounts.

3. Data output functions

Data search functions

Generating standardized reports for often-used data

Users should be able to streamline their searches by limiting their searches to publication season, contributors, articles, and type of status update (e.g. archival, current, etc.). The database should allow users to search and generate standardized reports for often-used data.

Generating standardized reports for often-used data (continued)

Users should be able to choose from a pre-set list and generate data reports for often-used data such as article/production schedules and printing sheets. The following is a list of often-used data:

- Printing spec sheets
- Article status reports
- Issue status reports
- Creators
- Contacts (Marketing, Institutions, Agents, Suppliers, Distributors, Rights)
- Proposals/submissions status
- Production schedules
- Publication calendars
- Rights status
- Sales tip sheets

Customizing reports for specific data

Another useful function would be to allow users to generate customized reports from the database. In addition to the pre-set list of often-used data, users should be able to "pull" specific data fields from a data list in order to generate customized reports.

Output features and formats

There should be standard data output features and formats for all data. For instance, users should be able to search for and generate a list of articles in progress. This generated list should be viewable onscreen, printable, and exportable to .PDF format. These three data output features and formats should be built into the database such that all data generated from searches can be output as such.

4. "Smart" data

Data updates

The database should automatically record the date for when a data file was last updated. This data update will always be generated along with standardised and customised data reports.

Reminder system

The database should have a reminder system that will inform users when certain important data fields have been changed (e.g. contributor bios, word count, article status, working titles). Such marked fields would be "red flagged" for a set period of time, thus informing users that the data has been changed. Users should also have the option of either having reminders delivered to their e-mail in-box, or simply logging onto the system to view the updates. A separate "database administration options" page should allow the database administrator to set and change these reminder warnings.

5. Internal Staff Communication

Staff web-log

A web-log should placed on the home page of the database would allow staff to communicate and update each other in projects in progress.

Required Ricecooker 2.0 Data Fields

Article Status

- Title
- Subtitle
- Author
- Contributor
- Commissioned
- Unsolicited/Submitted
- Approved for publication (yes or no)
- Version
- Series (Cont' from prev. issue?)
- Issue assignment
- department
- Word count
- No. of pages allotted
- Illustrations
- Photos (yes or no)
- Image format (.doc, .txt, .PDF, .jpg)
- Previously published in other journals?
- Original pub. date
- Original publisher (if any)
- Type of rights sold to *Ricepaper*
- Competition/comparative articles
- Potential marketing tips

Marketing/Advertising Sales tip sheets

- Article title
- Subtitle
- Author
- Bio
- Previously published works
- Issue
- Theme (if any)
- Season
- Release date
- Main features/ Sales handle
- Marketing points/Description
- department
- Word count
- Insertion
- Illustrations
- Distributors
- Newsstand Category
- Special projects (if any)
- Previous advertising/publicity
- Online/special markets
- Competition/comparison titles
- Critical praise (quotes)
- Table of contents/Excerpt

Required Ricecooker 2.0 Data Fields

Printing Spec Sheets

- Issue
- Trim size
- Page count
- Text paper (matte, gloss)
- Text paper weight
- Printing (1/1, 4/4, 4/1, specify)
- No. of colour images
- No. of b&w images
- Binding (hb, pb, plc)
- Cover stock weight
- Lamination (gloss, matte)

Rights Status

- Article/excerpt title
- ISBN (if book)
- Publisher
- Contact person/Position title
- Address/Shipping address
- Phone, ext./Fax/e-mail/Website
- Rights held (1st serial, 2nd serial)
- Contract date
- Pub. date/Duration of license
- Lapse date/Price paid

Creators (Categories)

- Art Direction by
- Author
- General Contributor
- Designed by
- Edited by
- Essays by
- Illustrations by
- Photographs by
- Photos by
- Text by
- Translated by

Creators General Fields

- First name/Middle name/Last name
- Bio
- Agent (if any)
- City/country of residence
- Address/Phone, ext./Fax/e-mail/Website
- Work place/institution/org. name
- Position held (title)
- Creator's pub. history/Reviews, Articles
- Upload creator's image (if any)
- Upload book cover image (if any)

Contacts (Categories)

- Subscribers
- Advertisers
- Institutions/Other Publishers
- Companies/Associations
- Suppliers (freelancers, etc.)
- Printers
- Distributors
- Agents/Publicists/Media
- Special sales contracts
- Friends of *Ricepaper*

Contacts General Fields

- Name
- Address
- Shipping address
- Phone, ext./Fax/e-mail/Website
- Delivery instructions
- Category
- Company/organization name
- Position within the company (title)
- Customer number
- Notes

BIBLIOGRAPHY

BOOKS

de Bruijin, Hans, Ernst ten Heuvelhof, and Roel in 't Veld. *Process Management*: Why Project Management Fails in Complex Decision Making Processes. Netherlands: Kluwer Academic Publishers, 2002.

Fisher, Kimball, and Mareen Duncan Fisher. *The Distance Manager, A Hands-on Guide to Managing off-Site Employees and Virtual Teams.* New York: McGraw-Hill, 2001. [cited 22 August 2005]. Available from: <http://netlibrary.com>.

Hancox, Ralph. *Topics in Publishing Management.* Vancouver: Simon Fraser University, 2003.

Howard, Tharon W. *A Rhetoric of electronic communities.* London: Ablex Publishing Corporation, 1997.

Suh, Phil, Dave Addey, David Thiemecke, and James Ellis. *Content Management Systems.* Glasshaus, 2003. Available from: <http://librarybooks 24X7.com>.

Winograd, T. and F. Flores, F. *Understanding Computers and Cognition—A New Foundation for Design.* Norwood: Ablex Publishing Corporation, 1986.

ARTICLES

Brown, John S. and Paul Duguid. "The Social Life of Documents" [online]. *First Monday.* Vol. 1, no. 1 (1996). [cited August 22, 2005]. Available from:<http://www.firstmonday.dk/issues/issue1/documents/index.html#9>.

Glaser, Mark. "The Future of Magazines" [online]. *USC Online Journalism Review.* May 2005 Issue. [cited August 22, 2005]. Available from: <http://www.ojr.org/ojr/stories/050524glaser>.

Schwartz, L. "Must Change, Will Change: Process Re-engineering In Publishing". *Publishing Research Quarterly.* Vol. 15, no.3 (1999): 99-108.

Wright, James. "Exploiting Your Assets". *Publishers' Research Quarterly.* Fall (1999): 84-94.

RICEPAPER ARCHIVES

Ahnsu Consulting Group. "Regional Marketing Plan for Ricepaper Magazine: A Review, Recommendations and Activities to Develop a Regional Marketing Strategy." Report presented at the annual marketing review meeting for *Ricepaper*, Vancouver, BC, April 15, 2005.

Arts Ink Consulting. "*Ricepaper* Strategic Plan and Implementation Plan." Report presented at the annual strategic planning meeting for *Ricepaper*, Vancouver, BC, April 30, 2004.

Ricepaper Magazine. "Canada Council Report on the Publishing Program." Grant application submitted to the Canada Council for the Arts, Ottawa, ON, February 2005.

Users Committee, *Ricepaper* Magazine. "Minutes from Users Committee Meeting." Internal Document. Vancouver, August 20, 2003.

Users Committee, *Ricepaper* Magazine. "Notes from VFS Client Group Project Meeting." Internal Document. Vancouver, BC, September 25, 2004.

Users Committee, *Ricepaper* Magazine. "Ricecooker 3.0 and Ricepot: An Evaluation by the Users Committee." Internal Report. Vancouver, BC, April 2005.

PRIMARY SOURCE MATERIALS

Choy, Hennie (Former Senior Editor, *Ricepaper* Magazine). Interview by author. Vancouver, BC, June 29, 2005.

Lee, Carol (Art Director and Acting Production Manager, *Ricepaper* Magazine). Interview by author. Vancouver, BC, June 24, 2005.

Lin, Jonathan (Information Technology Manager, *Ricepaper* Magazine). Interview by author. Vancouver, BC, June 24, 2005.

Siu, Michelle (Advertising Manager and Office Manager, *Ricepaper* Magazine). Interview by author. Vancouver, BC, June 24, 2005.

Uechi, Jenny (Assistant Managing Editor, *Ricepaper* Magazine). Interview by author. Vancouver, BC, June 28, 2005.

Wong-Chu, Jim (Publisher, *Ricepaper* Magazine). Interview by author. Vancouver, BC, June 20, 2005.

Wong, Richard (Editorial Assistant, *Ricepaper* Magazine). Interview by author. Vancouver, BC, May 14, 2005.

WEB RESOURCES

About VFS: Vision statement [online].Vancouver Film School website. [cited 22 August 2005]. Available from: <http://www.vfs.com>.

Periodical Publishing 2003 [online]. The Daily: Statistics Canada, 8 June 2005. [cited 22 August 2005]. Available from: <http://www.statcan.ca/Daily/English/050608/d050608a.htm>.

What is phpBB? [online]. phpBB: Creating Communities. [cited 22 August 2005]. Available from: <http://www.phpbb.com/index.php>.