6

Search and display

The previous chapter examined the various types of digital file formats that are commonly used in digital collections found on the web. The first half of this chapter will focus on the conventional ways users interact with these files when they are assembled together into documents. It is important to understand how documents of the types we have identified – textual documents, still images, moving images and sound, and data sets – work both within collections and within the larger context of the web, because virtually every decision you will make during the planning of your collection will have an impact on how users will find items in the collection, how the items will be displayed and how users will navigate the various parts of the collection. Understanding your options for presenting your content on the web will allow you to make more informed decisions about the metadata you will use in your collection, the features you want in a content management system, and in deciding on the ways in which you put into operation the digitisation or conversion phases of your project. This chapter will define some basic decisions that have to be made early in the process of planning your digital collection.

Even though it is impossible to define a 'typical' user or set of interactions a user has with a website, it is reasonable to assume that users of organised collections of documents search and/or browse for documents, are presented with documents they find, navigate through those documents (if the documents consist of multiple parts) and navigate from document to document within the collection. Other activities may also be common, depending on the features that the collection's developers have included; users may be able to add documents or parts of documents to personal space in the system ('my favourites' etc.), save documents, annotate them, e-mail them and so on. Each general type of document poses its own challenges, and within each type, there are many variations that pose significant challenges to collection developers and users. When building digital collections of textual documents, for example, it is considerably simpler to present single-page letters than newspapers, because newspapers comprise multiple, separate articles that can span several pages. In this particular example, decisions made early in the planning process can have significant implications for how the collection is used.

It is important at this point to distinguish between a collection's functional requirements and the digital library content management system's user interface functional requirements. The collection's functional requirements describe broad structural aspects of the group of documents as a whole and of individual documents within the collection. Those of the content management system's user interface build on the collection's functional requirements and add specific aspects of functionality that are independent of the objects and metadata that make up the collection. For instance, the ability for users to view individual scanned pages of a book is a function of the collection, whereas the means provided for zooming in on a single page or for e-mailing the page or book is a function of the digital library content management system.

The fields of information architecture and software engineering provide relevant tools for the developers of digital document collections. In the second half of this chapter, I will borrow from these fields and build on the knowledge they have to offer by applying the techniques of use cases and storyboarding to the planning of digital collections. The fields of information retrieval, text retrieval and cognitive science have much to offer in the form of experimental models for improving the accuracy, scalability and sophistication of the systems used to provide access to digital collections. I will refer to several promising examples of search and retrieval applications from these fields as well.

Still images

Still images are probably the most common type of content that libraries and other types of cultural organisations make available on the web. Photographs (historical, architectural, etc.), photographs of paintings and sculptures, maps, posters and postage stamps all fall into this category of content. Many libraries develop online image collections because, in general, image collections are easier in many ways to implement than other types of collections.

Searching for images

Because most images do not contain any textual content that can be queried, users must rely on metadata such as creator, title and description for resource discovery.[1] Some search and retrieval interfaces to image collections do not rely on metadata but can perform queries based on the content of images. Examples of this type of interface (commonly known as CBIR or content-based image retrieval) include IBM's QBIC (Query By Image Content)[2] and imgSeek,[3] an open-source application that does not yet operate over the web. This type of searching has not been deployed widely and is still considered experimental, although a QBIC interface to a collection of paintings in Russia's Hermitage Museum is available.[4]

Display and navigation issues

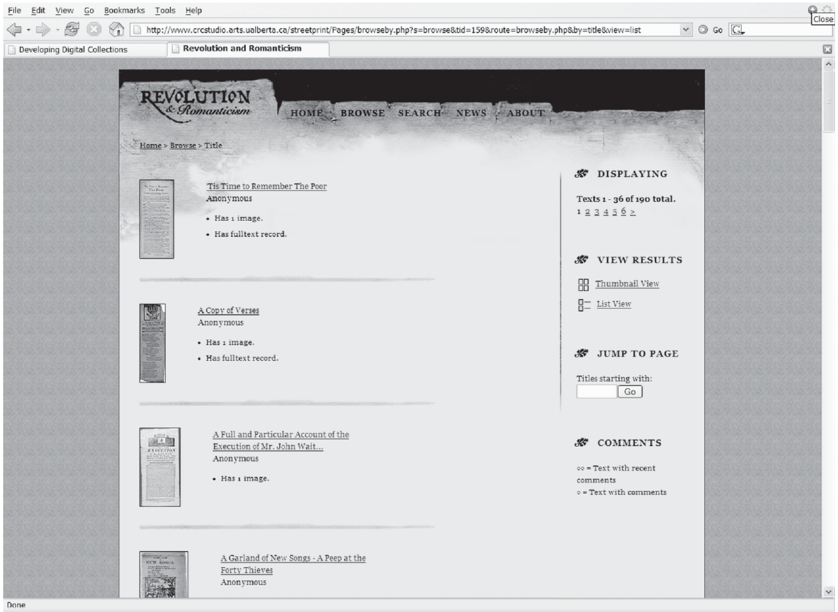
Result sets generated by user queries or by browsing metadata elements frequently consist of one or two descriptive elements and thumbnail versions of the images (typically 200 pixels wide or high), and hyperlinks to either a full descriptive record or to a large version of the image itself. Brief records in results lists are often laid out in a table; this layout allows the user to see more thumbnails at once than other layouts, as illustrated in Figure 6.1.

The presentation of images on the web is usually fairly simple. Typically, the entire image can be shown on the user's screen at one time, and because there is no need for the user to navigate with the document, the navigational aspects of the web page can remain minimal, but often include mechanisms for moving on to the next record in the current result set, navigating back to a search or browse interface, and returning to the collection's homepage or appropriate top-level page of the collection's website.

Determining the size of the image to provide to the user is not straightforward, as the size of computer monitors varies widely. Typical monitor resolutions at the time of writing range from 800 × 600 pixels (common on 14-inch monitors) to 1280 × 1024 pixels (common on 17- and 19-inch monitors). Ideally, the images you provide should be as large as possible as the low image density of computer monitors (72–96 DPI) can obscure the details in an image, particularly if the image is a black and white or greyscale image. As illustrated in Figure 6.2, the impact that screen size has on the layout of the image display is dramatic.

Figure 6.1 List and grid views of the same result set, as provided by the Streetprint Engine content management system.[5] Used with permission of the CRC Humanities Computing Studio, University of Alberta.

(a)



(b)

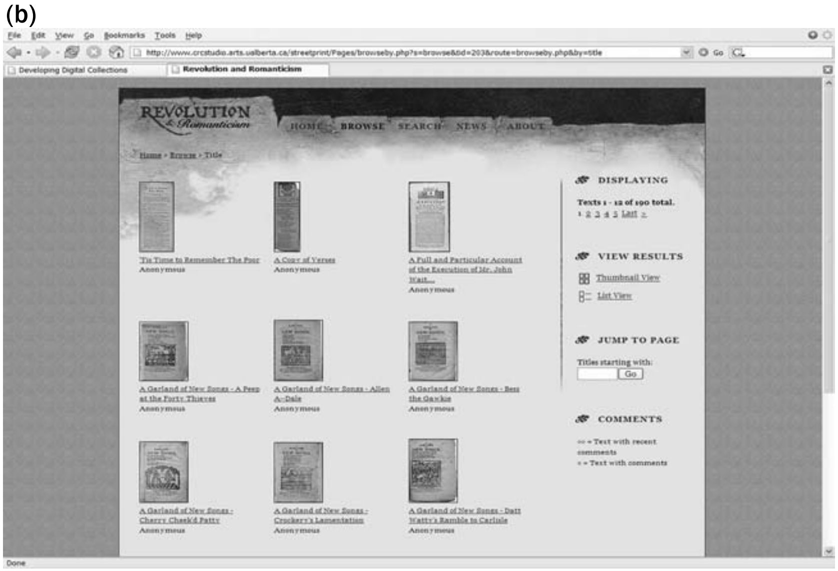
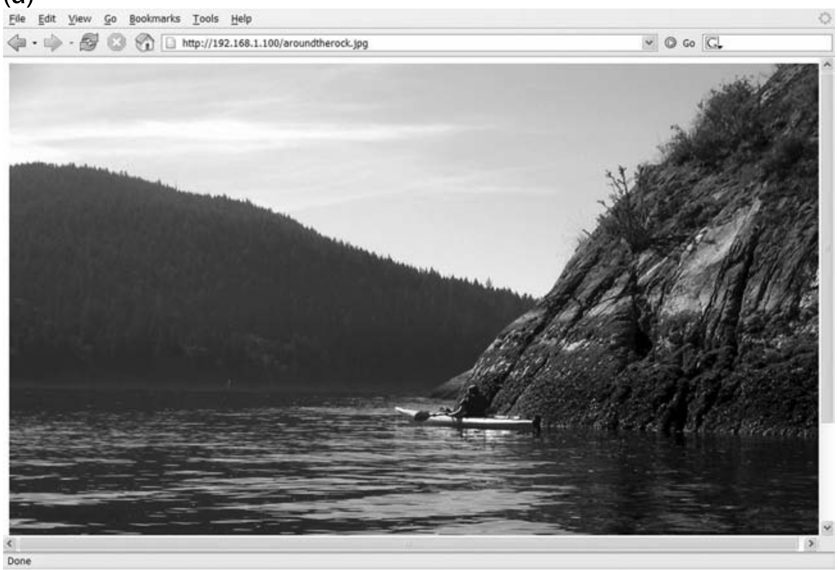
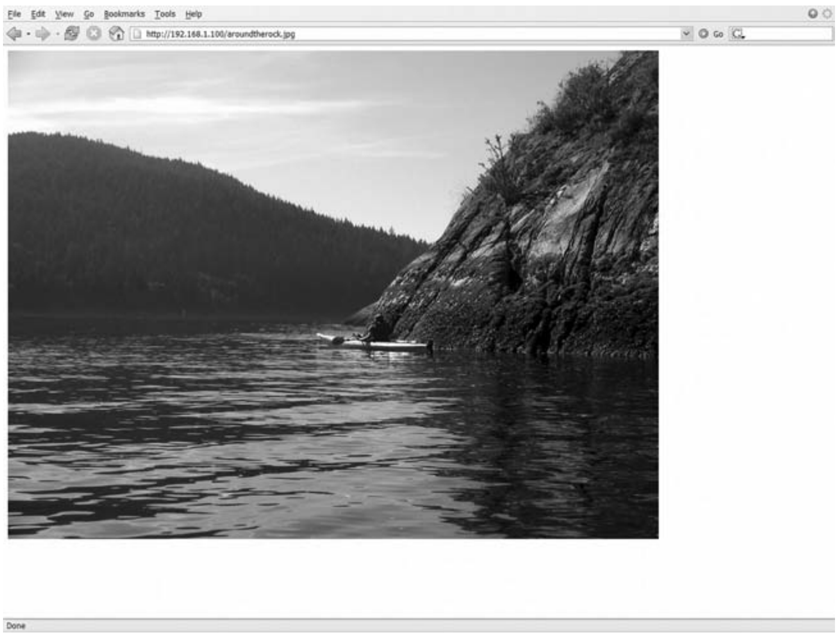


Figure 6.2 **Screen space filled by the same image on monitors of two different sizes: (a) 1024 × 768 resolution monitor; (b) 1280 × 1024 resolution monitor.**

(a)



(b)



The image of the kayak, water and mountains is 1000 pixels wide by 750 high. It is obvious that the proportion of screen space consumed on the smaller monitor is much higher than that consumed on the larger one. If the display contained any descriptive metadata, navigational elements or other elements on the web page, they would be totally obscured on the smaller monitor.

Some image collections present the full-sized images by linking from the descriptive record or thumbnail version to the full size file (for example, a .jpg) without wrapping it in HTML. Although doing this does allow the largest possible version of the file to be displayed, the context, even if it is a simple description of the content of the image, is temporarily lost. In addition, doing this requires the user to rely on the browser's 'back' button for navigation by removing the possibility of linking to the next record in the result set, a different size of version of the current image, etc. In general, if you decide to provide a link to the image file outside of any HTML wrapper, perform tests with potential users to see how they react to this type of presentation.

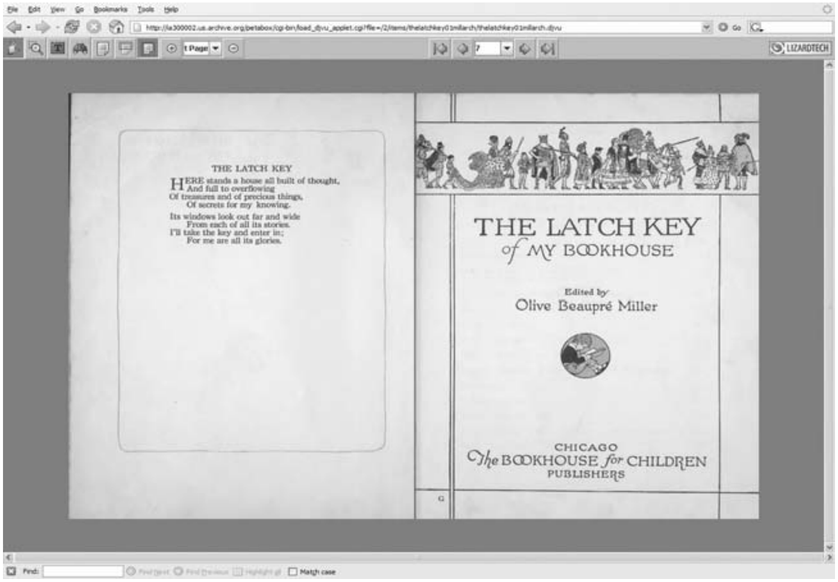
Most current web browsers will automatically resize images so that they fit on the screen properly, but this behaviour may be disabled by the user or not available at all in some browsers. Frequently, resizing the image may actually degrade its clarity or appearance. Therefore, you should not adopt the strategy of increasing the size of your images on the assumption that the web browser will always resize it to match the user's screen size.

One popular technique for accommodating a variety of monitor sizes is to provide several versions of the same image, all identical except for their pixel dimensions. In this way, users can view the version that best suits their monitors. Providing links to the different versions takes up little screen space. It is also possible to allow the web browser to detect the size of the monitor using JavaScript and to display the appropriate version of the image automatically. Creating multiple versions of the same image for this purpose (and for use as thumbnails in result set displays) is easy to accomplish using standard image manipulation software and takes up little hard drive space on the web server.

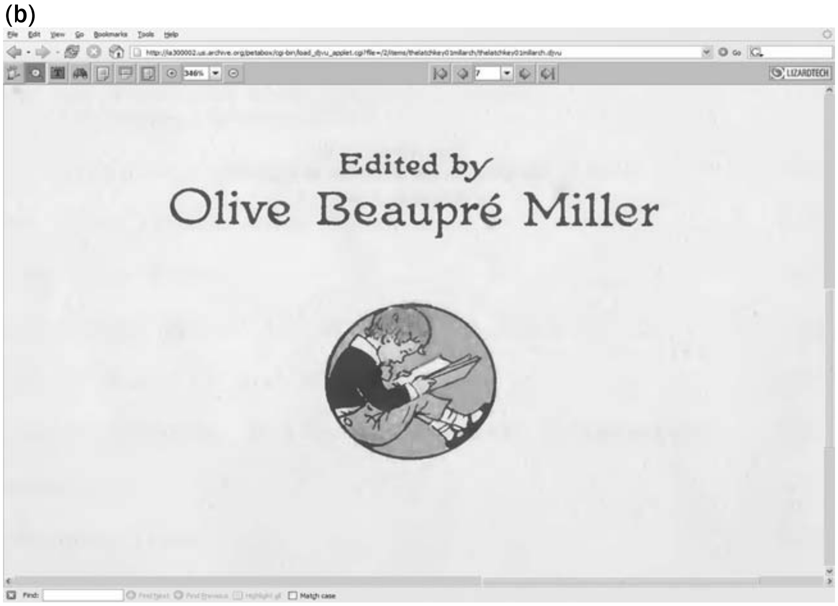
As indicated in the previous chapter, some image file formats such as JPEG2000[6] and DjVu[7] allow practical delivery of high-density image files. These types of file format are most effectively applied when users may want to view the entire image and zoom in to see specific areas of the image at higher magnification (Figure 6.3). Typical applications for this technology include rare books, maps, architectural plans and paintings. At the time of writing, both JPEG2000 and DjVu require browser plug-ins.

Figure 6.3 A double page image (above) from *The Latch Key of My Bookhouse*, and a magnified view (below), as viewed using the Internet Archives' DjVu applet. Used with permission of the Internet Archive.

(a)



(b)



Up to this point we have been discussing static images – those that are created during a certain part of the production workflow and that are placed on a web server as files to be displayed whenever they are linked to from within an HTML page. The file that is presented to the user is identical to the file that is sitting on the web server waiting to be requested. However, images can be created dynamically, or *at the time of retrieval*. A simple example of this type of image is a map that shows the route from one street address to another. The map showing this route is not predefined, the route is added on top of an underlying map. Another query using a different origin and destination would generate a map that differed from the first one only in the line representing the route.

The most common use of this type of image is in collections of textual documents that display scanned images of the pages in response to a user's query. I will discuss textual collections and documents in detail below, but textual pages displayed as images is very common. Several commercial systems, including Olive Software's Olive collection management system,[8] will highlight in the page image the keywords that the user has entered. Google Print also uses this technique.[9] In these cases, the pixel coordinates that define the boundaries of the word in question are used to define the highlighted area. Other applications of this technique might include drawing a border around a person in a photograph whose name was used in a query, and identifying particular species of plants in photographs containing multiple specimens.

Some image-based documents comprise multiple images. Postcards, which many people would consider still images, are actually more like multi-page full-text documents than still images because their reverse sides are often of as much interest as their front sides. As we will see later in this chapter, some documents do not fit easily into convenient categories such as 'still image' or 'textual documents'.

Textual documents

After still images, textual documents are the next most common type of content that libraries make available online. Apart from containing text, this type of document is usually characterized by having multiple pages. Books, periodicals, and handwritten documents such as letters and postcards are commonly digitised or are born digital and made available in organised collections on the web. Some academic libraries are now developing collections of theses and dissertations produced at their institutions and making these widely available over the web.

Searching for textual documents

Unlike images, which require some sort of metadata to allow them to be retrieved by a user, textual documents can be retrieved using keyword searches on the text of the document or by searching metadata. Both approaches have their uses: full text searching is useful when the user knows specific words that might appear in the text, and metadata searching is useful when the user knows some attribute of a document that will aid in retrieval, such as the author, title or broad subject. Alternatively, searches can be performed on a combination of metadata and full text.

If you plan to allow your users to search full text, you must decide on the granularity of the full text search: the entire document, logical subdivisions such as chapters, individual pages, paragraphs, etc. The most common search granularity is the single page (i.e. when a user searches for multiple keywords in the 'full text', records will be returned if the words all occur on the same page). In this case the results are usually displayed as a list of pages matching the user's query. If more than one page within the same document matches the user's query, the results list may contain the document title only once or may contain an entry for each matching page.

Page-level granularity may not always provide effective retrieval. To illustrate the importance of selecting the appropriate full text search granularity, we can contrast what 'full text searching' means for single-page documents such as memoranda and what it means for complex documents such as long technical manuals. In the former case, each document contains only one page, and the full text of each document is its entire content. Full text searching on a collection of these documents will probably prove to be effective because all searches are performed within a limited scope. For technical manuals, which might contain multiple chapters, sections, subsections, tables, diagrams (with captions) and footnotes, defining the most effective 'full text' search granularity is problematic; here, searches for the same three keywords within the entire document (which could be hundreds of pages long), within a single chapter, within a single page or within a single illustration caption would probably yield very different results: more hits would be retrieved within the larger scope of 'full text' but they may not prove to be very relevant to the user. The challenge in planning a collection of this type of document is to define 'full text' at the level that would provide the best balance of search recall and precision.

Ideally, users would be able to select whether they want their full text searches to be on single pages, within chapters (when searching books) or within the entire document (without or in combination with searching descriptive metadata). Some search and retrieval systems allow this flexibility, particularly those that use XML as their native data structure. Of course, these options may not be of interest to most users but the small percentage that choose 'advanced' search options may be able to take advantage of this flexibility.

It is important to keep in mind that the technologies involved in making large quantities of full text searchable are fairly specialised, most notably (a) in their ability to perform queries efficiently on large numbers of documents, and (b) in their ability to calculate the 'relevance' of particular documents to a searcher's query. Particular technologies should not determine how collections are organised, but the size of a textual collection can have a direct impact on how much time is required for a user to perform a query. A number of products and platforms exist that will easily handle several million full text documents, but in some cases these tend to sacrifice granularity and flexibility for speed. For example, the well-known Greenstone Digital Library software can handle very large quantities of text and can perform queries very quickly, but has limited sorting capabilities.

Display and navigation issues

Results lists for textual documents can be similar to those for images; for example, if the page images make up a textual document, thumbnails are often used to represent the page returned by the query. Including the images of book covers in result lists is also popular. In some cases, standard bibliographic information is used in results displays, often consisting of author, title and data elements. As alluded to above, if users search at page-level granularity, the page (thumbnail or simple hypertext link) is sometimes displayed in the result list; another option is to display the page and also a link back to the first page of the document.

The display of and navigation through textual documents digitised for access on the web pose a number of challenges. The following are issues that planners of digitised collections of texts have to consider. Many of these issues apply to collections of born-digital texts as well.

First, many people do not enjoy reading long passages of text on screen. Many factors contribute to this, ranging from general feelings that a printed book offers a pleasing tactile experience to a dislike of sitting in front of a computer monitor. Some of these issues can be mitigated by providing versions of the text that facilitate easy reading, such as page images that fill the screen. Formats such as Adobe Acrobat allow flexible page resizing, so users can choose a page size that they find easiest to use. Also, texts marked up in structured formats such as XML can be dynamically resized when presented on the screen using Cascading Style Sheets and other technologies, offering readers various degrees of usability. Some collections that feature images of each scanned page also provide a simple textual version that facilitates easy reading, especially when the scanned page is not clear or is damaged.

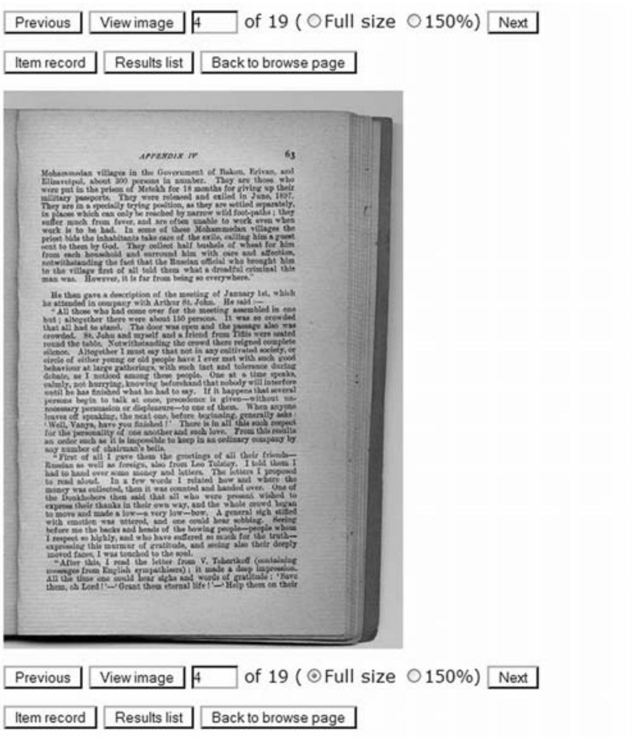
Given that many people dislike reading texts on a computer screen, it is important to consider how users will print documents in your collection. Documents that are made up of many individual files, such as single-file page images, are very awkward to print. Documents that are contained in one file, such as an Adobe Acrobat file, tend to be much easier to print.

The second major challenge when dealing with textual collections is that the physical act of turning the pages of a book or other text must be translated to useful mechanisms within a piece of software. 'Page turners' of various sorts allow users to navigate inside long texts using a familiar metaphor. These can take the form of simple image-based turners that provide 'Next' and 'Previous' buttons on a web page (see Figure 6.4), to Adobe Acrobat files that contain thumbnail versions of each page in the file that link to the full size version of the page.

Documents that are highly structured, such as those marked up in XML formats including TEI format or DocBook, can be presented to end users in multiple ways, being converted to multiple formats either before being made available on a website or dynamically as they are requested by the user. Two examples of collections of XML-encoded documents that are provided in multiple output formats are the University of Virginia Library's Etext Center's Ebook collection[10] and the Alex Catalog of Electronic Texts.[11] Both of these collections offer versions of well-known works of literature in output formats including HTML for standard web browsers, e-book formats suitable for portable hand-held devices such as Palm and Microsoft Windows-based PDAs (Personal Digital Assistants), and Adobe Acrobat.

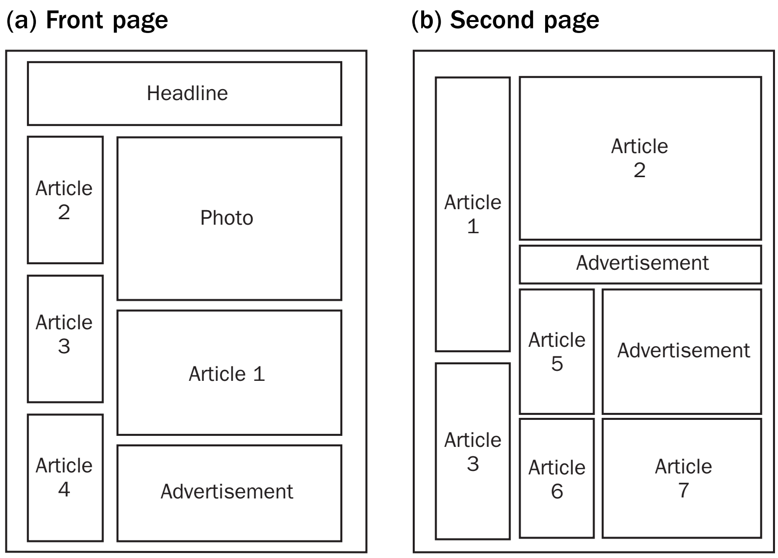
XML is the preferred format for complex documents that refer to other documents or to other parts of themselves, such as scholarly editions, variorum editions, and works within a larger context or group of significantly related documents. John Foxe's *Book of Martyrs Variorum Edition Online*[12] is an example of this type of document. This edition, hosted at the Humanities Research Institute of the University of Sheffield, is marked up in TEI, which is converted to HTML for presentation on the web. People's names and other references are identified by TEI tags, which are translated into hypertext links in the HTML.

Figure 6.4 A simple page turner from a collection of texts that provides one image for each page of text. Used with permission of Simon Fraser University Library.



Standard newspapers provide an excellent example of the variety of ways textual documents can be presented online. The most obvious way to provide access to a newspaper would be to scan each page and make them available on the web. As planners of this hypothetical collection, the first decision we would have to make is how large the page images should be: as newspaper pages are very large compared with most computer monitors, we would have to make sure that the images were large enough so the text is legible but small enough so that users can navigate within each image easily. The second set of choices revolves around navigating between various parts of a single article: many newspapers split articles up over more than one page. Figure 6.5 illustrates the front page and second page of a newspaper, simplified but nonetheless realistic.

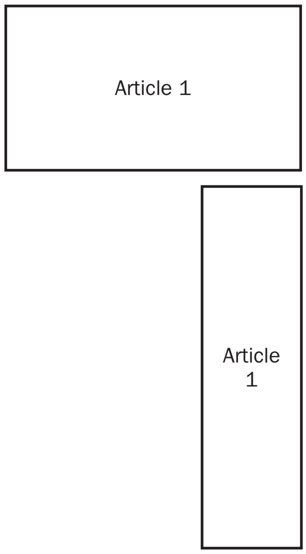
Figure 6.5 A simplified diagram of how standard newspaper layout can pose challenges for navigating between parts of a single article.



In order to allow our users to read all of Article 1, we would have either to allow them to turn to the next page (assuming they would expect that the article was continued on the second page) or to provide an explicit link from the bottom of Article 1 on the front page to the top of the continuation of Article 1 on the second page. Another option would be to piece both parts of Article 1 together for the user and provide the entire article at once, as in Figure 6.6.

The disadvantage of presenting the parts of the article using either of these options is that doing so for every article in the newspaper would be very labour intensive and therefore expensive. Software exists that will automate identifying parts of newspaper articles that span multiple pages, but it would have its own costs, which would not necessarily be lower than manually identifying, cutting out and reassembling individual articles. We could avoid these issues by providing the user with a transcription of the article without any images of the pages as they exist in the original newspaper, but then the context of the article as it appeared in the original would be lost.

Figure 6.6 A single newspaper article removed from its original context.



Moving images and sound

As networks become more robust and as improved file formats and software to read these formats become more common, video and sound recordings (together known as temporal media) are becoming increasingly common in digital collections created by libraries and other organisations. An excellent example of such a collection is the Internet Archive's Moving Image Archive,[13] which at the time of writing contained nearly 20,000 moving images and nearly 28,000 sound items.

Searching for moving images and sound

Like still images, video and sound files do not contain text that can be queried, so users must rely on metadata in order to find documents. And as also with still images, some progress is being made to develop automated methods of extracting 'full text' from temporal media that can then be searched or otherwise used for retrieval purposes. A leading example of this research is the work being done at the Centre de Recherche Informatique de Montréal.[14] Until this type of technology becomes widely deployed, the verbal content of temporal media will need to be manually transcribed and used in the same way as the content of textual documents. Technologies that automate generation of closed captioning for television are beginning to appear on the market, and mass-market voice recognition applications such as Dragon Naturally Speaking[15] are already common and may prove to be useful in converting the verbal content of temporal media into searchable text.

One type of metadata that is specific to temporal media is MPEG-7, which was designed specifically to address the unique retrieval challenges of multimedia content. As we saw in Chapter 4, MPEG-7 offers a standardised XML vocabulary for identifying structure, time points, and qualities such as colour and texture that allow precise retrieval and navigation within a moving picture or audio file. There are very few publicly available examples of MPEG-7 search and retrieval systems, however.

Display and navigation issues

A number of innovative display and navigation tools are included in the Internet Archive's interface to its moving images collection:

- On the details page for a moving image, a series of still images extracted from the moving image at one-minute intervals is displayed, giving the user an indication of the progression and content of the moving image.
- Beneath the rotating still images, a link with the text 'View thumbnails' leads the user to a grid view of all of the extracted images.
- Some videos in the collection can be viewed at an accelerated speed; for example, a moving image that is 15 minutes long can be viewed in 3 minutes, allowing the user to evaluate its usefulness or relevance.

The Internet Archive and other collections of moving images and sound provide multiple versions of the same file, each balancing quality and size, which are in general inversely related to each other. For example, typical moving images are provided in the following variations:

Streaming

- 64 kb MPEG4
- 256 kb MPEG4

Downloading

- 64 kb MPEG4 (109 MB)
- 256 kb MPEG4 (244 MB)
- MPEG1 (904 MB)
- MPEG2 (5.7 GB)

'Streaming' refers to a method of delivery whereby the files are sent from the web server to the user's computer in a constant stream, allowing the movie to start before the entire file is downloaded. The file is not saved on the user's computer but is played through special viewer software on the user's computer. 'Download' refers to delivery of the entire file at once; the user must wait until the entire file has been downloaded until the movie starts. The file is saved to the user's hard drive so it can be played later. The higher the quality or density (here expressed in kilobits or more properly kbit/s), the larger the files.

Sound files in the Internet Archive are provided in much the same way as video, allowing users to choose the highest quality version they can use given the speed of their Internet connection:

Download

- Lossless 621.6M
- Hi-Fi 151.9M
- Lo-Fi 49.1M

Stream

- Hi-Fi
- Lo-Fi

'Lossless' is the best quality file (also the largest); 'lo-fi' is, as the name suggests, the lowest quality but is also the smallest.

The reason it is important to provide a number of versions of moving image and sound files is that the files tend to be very large. As the examples from the Internet Archive show, the highest quality version of the video files is 5.7 gigabytes, the equivalent to a commercial movie DVD. Even the lowest quality version of the same file is over 100 megabytes, which would be unusable over a conventional modem.

Delivering moving image files over the Internet is not always reliable owing to the huge file sizes. Like still images, moving images are created at a specific pixel height and width, and resizing or enlarging them in the player software can cause the image to appear stretched and blurry. Therefore, providing several versions of each file allows users to select the best quality version that their Internet connection can transfer reliably.

Data sets

Data sets are the raw numerical data generated by research in the physical and social sciences. Non-numerical data such as qualitative survey responses can also belong to this type of digital content. Libraries of various types maintain collections of raw data, much of it already in digital formats but not easily accessible over the web, such as tapes, CD-ROMs and a variety proprietary formats. In the last few years, however, an increasing number of data sets are becoming available over the web. Factors that are increasing the need for libraries to create and maintain collections of data sets include increased interdisciplinary use of GIS data, a growing focus on 'e-science' and 'e-social science', and policy changes by national funding agencies in many countries that require grant recipients to make data generated with their funding support widely available.

Searching for data sets

Data sets, and collections of data sets, tend to be highly domain-specific, and therefore it is difficult to generalise about the search tools that accompany these collections. For example, the data sets available through NASA's Goddard Earth Sciences Distributed Active Archive Center, such as the *SIMBIOS-NASDA-OCTS On-line Data Products*, require knowledge of the particular variables that are described by the data, such as 'normalized water-leaving radiance', 'chlorophyll *a* concentration', and 'integral chlorophyll, calculated using the Level-2 values chlorophyll *a* divided by K_490'.[16] These and other variables are options that the user selects as part of the query.

Many collections of data do not have search interfaces as specialised as those described above. A much simpler user interface to raw data, in this case, XML files that describe molecules, is provided by the University of Cambridge's *WorldWideMolecularMatrix*.[17] This interface (which is the one provided by the DSpace repository software) provides a single search field and also allows users to browse the collection by title, author or date.

Although not a collection maintained by a library, Google Earth is a good example of information visualisation,[18] the use of visual search interfaces to facilitate the querying and use of numerical and other types of data. Users of Google Earth can enter street addresses and 'zoom right in' to a photo or map, 'tilt and rotate the view to see 3D terrain and buildings', and perform other types of queries on Google's vast datastore of geospatial data, satellite imagery and maps.[19]

Display and navigation issues

Tabular data in plain text format, tabular data in Microsoft Excel files, structured data in XML files and specialized formats associated with particular data-processing software applications are all common file formats contained in data collections. Some files that are part of data sets are extremely large (in the same range as larger moving image files), but issues associated with the size of the files are sometimes mitigated by systems that allow the retrieval of subsets or portions of the entire data set. Also, users of data sets are likely to be situated at universities and other institutions that will probably have fairly robust networks. In many cases, specific software applications (far more specialised than, say, Excel) are required to view or process the retrieved data.

In general, because the users of specialised data collections tend to have access to the knowledge, tools and infrastructure required to use the data, planning and design of these collections tends to focus less on the types of display and navigation issues associated with collections of images, textual documents, and moving images and sound, and more on creating specialized interfaces to extract the desired data.

Mixed document collections

Classifying all document collections as images, text, temporal media and data sets is somewhat artificial. Many collections contain several of these types of document, and some documents (known variously as 'complex' or 'compound' documents) comprise more than one type of content (e.g. textual documents that contain images). Some documents, such as sheet music, do not fit easily into the categories I have defined. The preceding analyses of each type's particular qualities and challenges is intended to make collection planners aware of some of the issues they will face when they start planning their collection, and to illustrate some of the complexities involved in creating organised collections of documents.

Collections that contain documents of more than one type ('mixed' collections) pose their own challenges. The first set of issues relates to the tools you will provide to the user for searching. Many search interfaces allow the user to limit his or her search to a particular type of document, which, as long as the metadata describing the documents contains the necessary information, may provide useful functionality. However, providing the same search granularity to users of collections containing heterogeneous documents as to users of homogeneous collections can be problematic. For example, a collection that contains both textual documents and photographs could provide users with options for searching on metadata elements such as 'writer/photographer', 'title' and 'subject'. If full text searching is provided for the textual documents, then the search tools should take into account the user's inability to search the full text of images. Because photographs do not typically contain full text, choosing clear and effective labels for search interface components can be challenging (as the example of 'writer/photographer' suggests). Although many users may not be confused by seeing a 'full text' field in the search form if they know the collection contains photographs, others may be. As usual, careful user testing early in the development phases of a collection's interface and search tools can help avoid usability problems later.

A second challenge is defining how search results will be displayed. We have seen how the various types of documents lend themselves to the use of different types of compressed or compact displays of results lists – image collections typically use thumbnails, textual collections use bibliographic information, collections of sound files use icons representing loudspeakers, and so on. The results list produced by a search that returns multiple types of documents will contain multiple types of compressed records. This is not necessarily negative, but does complicate decisions about how to display useful results lists. Another issue is how the results are grouped. In other words, are the results ordered by an attribute that applies to all the documents in the collection (such as date) or are all of the texts displayed together, then the images and then the sound files. This may not seem like a problem to many collection planners but some bodies of material may lend themselves to this type of grouped display, and some users may prefer to see the results displayed in this manner.

Planning your collection's interface

Planning the user interface to digital collections is a complex process. In addition to the issues surrounding the various types of digital content described in this chapter, factors that can have a direct influence on how your collection 'works' in a general sense include issues surrounding the production of the content, financial resources available for the development and maintenance of the collection, available technical infrastructure, the nature of the metadata applied to the items in the collection, and political factors such as donors' desires to see their material treated in ways that may not be consistent with best practices or user friendliness. To assist you in making informed and defensible decisions about providing access to your content, the remainder of this chapter will focus on techniques of use cases, storyboarding, and paper prototyping as they can be applied to online collections.

Use cases and functional requirements

It is useful to distinguish between the functional requirements of the software used to deliver your collection – the content management system – and the functional requirements of the collection itself. Reasons for separating the two types of requirements include:

- If you have not yet selected a content management system, determining the functional requirements of your collection will help you clarify certain aspects of the software during your selection process.
- You may migrate your collection from one content management system to another over the years; new versions of your current system may offer unplanned ways to access and use the collection; and in some cases, features disappear in newer versions of your current system.
- You may want to present your content to a number of different audiences (e.g. you have identified the two target audiences of a historical photo collection as historians and young children), and you may publish the content in multiple content management systems.

I will look closely at functional requirements of content management systems in the next chapter. The specific capabilities of these systems do have a real impact on how users interact with the documents in your collection, but ideally those capabilities should not drive how you plan and organise your collection; in other words, content and general search, display and use requirements should drive the overall architecture of your collection; the capabilities of a particular content management system should not. In the remainder of this chapter I will discuss how you can systematically define the aspects of your collection of documents that are independent of the software you use to deliver the collection on the web.

One technique for determining functional requirements for software applications is the *use case*. Use cases are scenarios that describe how a system should interact with the end user.[20] Even though our goal is to define how users should interact with our collections independent of any software, the same methodology that software designers use can be applied to collections of documents. This methodology typically involves using highly structured and standardised templates that identify the key aspects of each case; for example, some common template elements include a *summary* (a brief description which captures the essence of the scenario), *preconditions* (describing any conditions that must be true when a user enters the scenario), *triggers* (which describe the starting conditions when the scenario is entered), *basic course of events* (the essential actions in the scenario, in their required order and often numbered), *alternative paths* (exceptions, what happens when errors of various kinds occur), *postconditions* (describing the state of the interface at the end of the scenario) and *notes* (any additional information). Although there are a number of competing templates and ways of applying them, the goal of the use case methodology is to isolate and describe the ways that the application (or in our case, the collection) is intended to be used.

Table 6.1 is an example use case for an image collection describing the user's ability to find images by artist's name.

Use cases exemplify and document functional requirements. Each use case should describe a single aspect of your collection, and after you have completed writing all of your use cases, a listing of all of the individual summaries can act as your functional requirements list. To extend the example above, if you want your users to be able to find images through other access points such as subject descriptors or place names, you would write a use case for each of those access points, with the goal of documenting a complete list of desired access points.

The goal in writing use cases is to help us think systematically about certain aspects of our collection independent of the software used to deliver the collection. It is important to keep this distinction in mind when writing use cases, as it is very easy to fall into the habit of describing a user's interaction with a collection of documents and individual documents within the collection in terms of the user interface elements, application

Table 6.1 **A sample use case**

Summary	In this collection, users must be able to search for an artist's work by the artist's name
Preconditions	User must know how to spell the name of the artist. User must know how to enter the name of the artist (i.e. in direct or inverted order)
Triggers	User must be at a page on the collection website that contains the search field
Basic course of events	User enters the name of an artist, for example 'Vincent van Gogh'
Alternative paths	No records are found: user is presented with a list of names that are similar to the one entered
Postconditions	User has found all records by requested artist
Notes	The ability to search by artist's name requires a corresponding metadata element. In order for our search interface to support artist names entered in direct order, our metadata will need to use cross-references.

features and even application 'look and feel'. Use cases can be deployed in designing or evaluating a content management system (after all, they come from software engineering), but at the planning stage of a digital collection they describe required search capabilities, document attributes and navigational mechanisms, not a piece of software. At the end of this phase of planning, you will be able to state the following about each of your functional requirements: 'Users of this collection should be able to ...'.

Methods of generating use cases include interviewing potential users to determine how they would like to use the collection, and trying similar collections and enumerating the types of tasks that you can (or cannot) perform using their interfaces. It is important to be specific when generating use cases so that all of the parts of the template as illustrated above can contain sufficient detail to produce useful functional requirements. Another potential issue to be aware of is that your ideas of how your collection should work are probably not the same as a *typical* user's ideas. The classic example of this problem is advanced search functionality. Librarians frequently request the ability to use Boolean logic, granular limiting options and truncation, but most users of online collections never use anything more than the simple search functionality provided on the collection's homepage. Including both is perfectly acceptable, but if limited resources require deciding between the two types of search interface, the one most likely to be meaningful to typical users is probably the better choice.

An important functional requirement is adherence to standards that promote accessibility for users with disabilities. I will cover this in the next chapter.

Storyboarding and paper prototyping

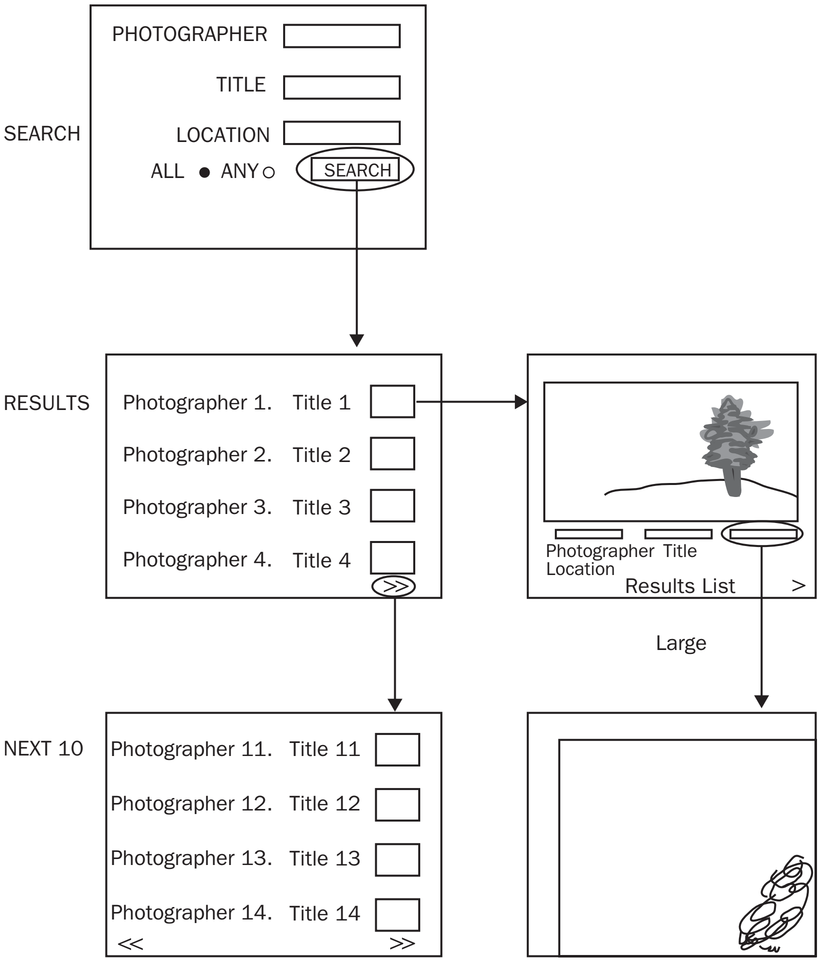
After the functional requirements for a collection and its constituent documents have been defined, the requirements can be validated by using two techniques, storyboarding and paper prototyping. Combined, these techniques will minimise the likelihood that important aspects of your collection will be overlooked, and that the functional requirements for the collection are possible given the resources available for the development and maintenance of the collection.

Storyboarding is the process of representing graphically the typical tasks involved in using your collection. The technique originated in film making and has been adopted by instructional designers, software interface designers and others to allow them to test various combinations of screen layouts, conditional operations and navigational components (Figure 6.7).

Storyboarding can help planners of digital collections in the following ways:

- *By ensuring that functional requirements can be translated realistically into online documents and collections*: Storyboarding's purpose is to allow planners to walk through (more precisely, to *draw* through) tasks such as searching for documents, displaying them, and navigating within and between them. These tasks will have been identified in the functional requirements. If each functional requirement can be drawn as part of a storyboard diagram, planners will be able to demonstrate the tasks associated with each functional requirement.
- *By raising issues related to production of the documents*: Functional requirements such as whether or not each page will be displayed as its own image, or alternatively whether the entire document will be displayed as a single file, may determine workflows and procedures during production of the documents. Chapter 10 will deal with workflow in detail, but decisions made during the functional

Figure 6.7 **A simple storyboard illustrating the steps required to get from the search page to the display of an image.**



requirements stage of the planning process will have a direct bearing on how the documents and their constituent files will be created, and storyboarding can validate these functional requirements.

- *By helping identify the descriptive and structural metadata required for the proposed architecture*: For complex objects (i.e. those that comprise more than one file), functional requirements relating to searching, display and navigation will determine what types of structural metadata will be required to enable these tasks.

Storyboarding can highlight the necessary descriptive metadata and the necessary structural metadata for fulfilling specific functional requirements, such as those associated with page turners, hyperlinking between sections of a document or linking between texts.

- *By highlighting functional requirements of content management software*: We have been separating the functional requirements of the collection and the functional requirements of software used to deliver that collection to end users. Storyboarding can illustrate how specific ways of searching, displaying and navigating might be embodied in general terms, independent of any particular content management application. The basic use cases and resulting functional requirements can help form the basis of the functional requirements of software packages that will either have to be developed locally, found among open-source applications or procured from vendors.

Paper prototyping is related to storyboarding, but involves users who interact with simple mock-ups of the collection. As the name suggests, these mock-ups are made of paper. Paper prototyping is a further refinement of storyboarding in that it takes testing away from the whiteboard and into the hands of users. Individual screens depicting user tasks that were represented as rectangles in a storyboard are transformed into corresponding sheets containing the hand-drawn (or roughly laid out using graphics software on a computer) equivalents, which are manipulated by typical users so that the steps in the tasks and the transitions between tasks can be tested. In software development, this interactivity even extends to having separate slips of paper representing various states that a menu can have, hypertext links on a web page, and buttons and other HTML elements. People acting as 'the computer' swap slips of paper representing changes in these interface components in response to user actions and choices, while someone else observes the user to see where he or she has difficulty interacting with the prototype. If any problems arise, they are documented and taken back to the interface designers.

Within the context of planning digital collections, it is not necessary (and not advisable) to achieve the same detailed level of interactivity with test users as software developers would achieve. For planners of digital collections, the purpose of paper prototyping is to allow representative users to validate the choices depicted in the storyboards. Collection planners may find it enlightening to observe how users interact with search forms, browse lists, navigation buttons and drawings depicting documents; again, it is important to remember that it is the collection and its constituent documents that are being tested, not a particular software application. Also, it is extremely important to remember that any problems users have with the prototype are the result of a poorly designed interface (or prototype) and are *not* the fault of the user. Staff who are running the test session must train themselves to remind the user of this tenant of paper prototyping whenever the occasion arises.

Multiple user interfaces

Some collections may warrant multiple user interfaces. For example, a collection of historical diaries may have one interface that is suitable for specialists (hobbyists, genealogists, historians) and another that is suitable for children. The first interface may feature advanced search tools and may present the diaries such that users can read the entire book using a page turner built into the website; the second interface may present selected pages from a group of diaries that all describe the same event, linked from an artist's pictorial rendition of the event. Multiple interfaces that provide access to the same content (in this case the same scanned pages from a group of books) allow you to provide the most effective access to the collection possible, as specialised interfaces can meet the specific needs of a particular group of users without the risk of being cumbersome, difficult and ineffective for general or non-specialist users.

Providing multiple interfaces to the same content is only possible if you produce that content with either all the planned interfaces in mind or without reference to a particular interface. The second approach is obviously the desirable one, as you are unlikely to be able to predict what specialised interfaces you might want to develop in the future. As Clifford Lynch puts it:

> [L]earning materials, interpretation and presentation seems to me to typically – or at least often – have shorter lifespans than the primary source materials that they draw upon. If you look at the processes of scholarship they include a continual reinterpretation of established source material (as well as the continued appraisal of new source materials). Source material persists and generation after generation of scholars and students engage it, yet we typically rewrite textbooks every generation or so at least.[21]

Each interface you develop will have its own development, cost and maintenance issues, but multiple interfaces to the same content can often be justified if the benefits to specialised audiences are defined, tracked and documented. In addition, specialised interfaces allow the use of features that enable tasks other than simply viewing images, reading texts or downloading raw numerical data. For example, CONTENTdm allows users to compare two images selected by the user side by side;[22] MDID features a drag-and-drop 'light table', which uses the metaphor of the traditional tool used to view and arrange photographic slides.[23] An increasingly popular feature is the ability for users to annotate items in a digital collection and to discuss the item with other users, within the context of the collection's website. The variety of features that allow users to perform engaging tasks can be overwhelming to planners of digital collections because some features (but not all) require unproven, non-standard or proprietary technologies. Technologies such as Macromedia's Flash[24] are extending the capabilities of what standard web browsers can allow users to do without specialised software. Multiple, specialised interfaces allow collection planners to test new technologies without necessarily committing to integrating them into the collection's sole interface.

A good example of a collection that uses multiple user interfaces is the International Children's Digital Library,[25] which provides three different ways to read each book: 'standard' (one page image at a time, with links for navigating to the next and previous pages), 'comic' (a zooming interface in which all pages are laid out in panels like a comic book) and 'spiral' (a zooming interface like 'comic' but the pages are laid out in a spiral pattern that rotates so that the current page is brought to the front of the spiral). Books that are still under copyright are also provided in Adobe Acrobat format.

Summary: making decisions about search and display

Grouping the content that we are likely to organise into collections on the web into still image, text, video and sound, and raw data allows us to identify and address issues in searching for these types of content and in displaying them to the user. Although this classification is admittedly simplistic, it can prove useful in planning our collections as long as we acknowledge that many collections contain content of various types. Also, users tend to want to do more than simply search for and display documents; they may want to manipulate them in various ways, save them for use later, comment on them, e-mail them to friends or print them for reading offline. Planning how we anticipate users to interact with our collections in general can be challenging because it is difficult to separate the 'collection' or 'documents' from the 'software' and in fact it is not possible to do so completely. We can assume that the digital content we have created will have a longer life than any particular content management or digital library system we use to deliver that content, however, and we may even want to provide more than one 'view' of our content in order to meet the needs of specific audiences. Being conscious of the particular qualities and attributes of the basic types of digital content typically organised by libraries into online collections can assist us in our efforts to organise our collections so they can be used effectively.

Every library will develop its own protocols and methods for planning a digital collection, influenced by any number of internal and external factors. I will look at some of those factors closely in the chapter on project management. This chapter introduced several techniques from the fields of software and usability engineering that can be applied to the planning of digital collections. These techniques can help us identify possible issues that might impact our work significantly in other stages of developing our collection, and demonstrate that the tools we intend to provide to users are effective. Most importantly, these techniques can help us validate assumptions we make when organising content into coherent digital collections.

Further reading

Horton, W. (1994) *Designing and Writing Online Documentation*. New York: John Wiley & Sons.

Even though Horton's book pre-dates the web as it currently exists, the book focuses on aspects of online documentation that are directly relevant to designing collections of digital documents, such as planning strategies, access methods, differences in reading texts in print and online, and effective use of multimedia. Horton also cites a large body of scholarly research and examples from commonly known software systems that is still relevant although somewhat outdated.

Nielsen, J. (2000) *Designing Web Usability: The Practice of Simplicity*. Indianapolis: New Riders.

Jacob Nielsen is a leading usability expert and the author of the popular useit.com columns on web usability. His book *Designing Web Usability* documents websites that exhibit good and bad user-centred design, and although it does not address the particular challenges involved in planning collections of documents, Neilsen's constant focus on the user should be a model for anyone who is involved with putting content online.

Rosenfeld, L. and Morville, P. (1998) *Information Architecture for the Word Wide Web*. Sebastopol: O'Reilly.
This book was the first to articulate in detail the principles of 'information architecture', the study and practice of user-centred organisation of information. The chapters on 'Organizing information' and 'Designing navigation systems' are directly relevant to the development of digital collections and the websites that bring them to users. The authors are librarians by training and this background is evident.

Snyder, C. (2003) *Paper Prototyping: The Fast and Easy Way to Design and Refine User Interfaces*. San Francisco: Morgan Kaufmann.
This comprehensive handbook is the standard resource for people who want to employ paper prototyping in software development, but most of the techniques Snyder describes can be used outside of software application development and can be applied to the planning and development of digital collections. Snyder covers everything from the political implications of paper prototyping to the types of paper and glue you should use in your testing.

Witten, I. and Moffat, A. (1999) *Managing Gigabytes: Compressing and Indexing Documents and Images*. San Francisco: Morgan Kaufmann.
A well-respected textbook on building digital libraries, *Managing Gigabytes* focuses on the techniques involved in high-performance text retrieval (although as the title suggests, image retrieval is also covered). The popular Greenstone digital library software (*http://www. greenstone.org*) is the result of Witten's application of information retrieval research to a sophisticated and feature-rich digital library content management system.

Notes

1. A comprehensive overview of theories of and applications in image retrieval is given by: Jorgensen, C. (2003) *Image Retrieval: Theory and Research*. Lanham, MD: Scarecrow.

2. *http://wwwqbic.almaden.ibm.com/*
3. *http://imgseek.python-hosting.com/*
4. *http://www.hermitagemuseum.org/fcgi-bin/db2www/qbicSearch.mac/qbic?selLang=English*
5. Revolution and Romanticism, *http://www.crcstudio.arts.ualberta.ca/streetprint/*
6. *http://j2karclib.info/* [but this is covered in the preceding chapter]
7. *http://www.djvuzone.org/wid/index.html* [but this is covered in the preceding chapter]
8. *http://www.olivesoftware.com/*
9. *http://print.google.com/*
10. *http://etext.lib.virginia.edu/ebooks/*
11. *http://infomotions.com/alex2/*
12. *http://www.hrionline.ac.uk/foxe/*
13. *http://www.archive.org/details/movies*
14. *http://www.crim.ca/en/*
15. *http://www.scansoft.com/naturallyspeaking/*
16. *http://daac.gsfc.nasa.gov/data/datapool/OCTS/index.html*
17. *http://www.dspace.cam.ac.uk/handle/1810/724*
18. The best introduction to information visualisation is: Bederson, B.B. and Shneiderman, B. (2003) *The Craft of Information Visualization: Readings and Refelections*. San Francisco: Morgan Kaufmann.
19. *http://earth.google.com/*
20. Wikipedia contributors, 'Use case,' *Wikipedia: The Free Encyclopedia*, *http://en.wikipedia.org/wiki/Use_Case*
21. Lynch, C. (2002) 'Digital collections, digital libraries and the digitization of cultural heritage information,' *First Monday* 7:5, *http://www.firstmonday.dk/issues/issue7_5/lynch/*
22. *http://www.contentdm.com/*
23. *http://mdid.org/*
24. *http://en.wikipedia.org/wiki/Macromedia_Flash*
25. *http://www.icdlbooks.org/*