Genre-aware Systems Development
A Case Study of a Web Content Management System
Jonas Lundberg, Linköpings universitet
Linköping University Electronic Press, Department of Computer and Information Science
jonlu@ida.liu.se

Abstract: In this paper, I describe how genre analyses were used to achieve system requirements, for the first system iterations of the shapeCMS parallel publishing system. I argue that genre-aware systems development delivers a set of core functionalities in terms of form, content, and purpose that a system producing genre instances typically need to support. I also discuss genre change, with a focus on system dependence on five layers of separation between form and content. The layers are (0) contents, (1) the site structure layer, (2) the connection layer (3) the page structure layer, (4) the element layer, and (5) the presentation layer. The four example web sites designed to illustrate the approach, also draws on the four organizational principles of information architecture and digital genres, namely search, structure, hypertext, and stream.

Introduction
Two problems that face many publishing systems are discussed in this paper. The first is to achieve a good-enough set of initial requirements. The second problem is that of genre change. It is common to view technical development as a driving force, or as an enabler of genre change (Shepherd & Watters, 1998). Also design is sometimes seen as a force of genre evolution (Pedro & Carlos, 2003). However, genre can also drive the evolution of technology, by forcing systems to adapt to changing genre rules, carried out using competing systems. Previous work in the area of genre analysis has shown, for different genres, what page elements are most central to that genre, at specific times (Ihlström & Lundberg, 2004). That gives a set of central elements, which are needed for creating generic instances of the genre, fitting the genre at that time. The elements also have some variation in their presentation, which adds another challenge, to allow some variation in the element presentation. However, research has also shown that genres tend to change over time, so that a system matching the requirements at one moment in time might slowly become obsolete. Thus, when constructing a web publishing system, genre change should also be taken into account, to make the system somewhat elastic so that it can cope with modifications of the genre.

The main idea in this paper is that genre instances can be seen as the composition of elements, defined in terms of form, content, and purpose, composed according to genre rules. By identifying a core set of elements, recurring in most genre instances, a core set of system requirements can be achieved. Adding new non-typical elements, or elements from other genres, can then give the system unique characteristics.

The following modifications of a genre are considered here:
- Deviation from genre rules by adding new elements.
- Variation in the form and composition of genre elements.
- Variation in visual style.

In this paper, the design of the shapeCMS content management system is discussed. It has a five-layer separation of form and contents, and support of a set of basic page elements. Using these elements, the core elements of online newspapers, blogs and discussion forums can be rendered, using a basic set of contents (text and images).

Four principles of web site organization
In the literature, four basic organization principles are described; structure, hypertext, search and streams. The first three are described in information architecture literature (Rosenfeld & Morville, 2002), whereas streams are described in literature on digital genres (e.g. Ihlström & Lundberg, 2004). These principles can be viewed as user activities, but also as page elements. Some elements are mainly used to support specific activities, such as the search function, or navigation elements (menus, breadcrumbs). However, many elements serve several purposes. The activity of navigation is supported also by other elements, for instance. For instance, streams, which show updated contents from specific sections, can both be used to get to the sections in general, and to specific contents.

Structure is the container of contents. The site structure, which contains locations that you can move to, is an example of structure. Also pages, which contains fixed page elements, such as navigation, advertisements, and news contents. The structure can be either hand-made, or generated from for instance meta-data. Static structures allow learning the position of items. Structures thus can have hierarchical relations, e.g. ice hockey being a sub-section of sports.

The search space, consists if all contents, and their metadata, including relations between contents. A search function can query this space, and show its contents. If the space is rendered on the page, then a visual scan can search the space (now as a structure, a normal web page) manually. If the search is rendered over several pages, then the user needs to
navigate when searching (just like browsing a site). Alternatively, the user can narrow down the search, or widen it, if too few items were found.

**Hypertext** refers to relations by association, for instance news items associated with each other as belonging to an event, such as a tropical storm. These associations typically are not connected to the general site structure, but are connections between particular items.

**Streams** embody the time aspect of web sites. Many sites, such as blogs, newspapers, discussion forums, and e-commerce sites, have relatively static structures, whereas the contents change dynamically. For instance, although latest news is always shown on the same page, and on the same position on that page, the actual news items shown there change over time. To find the latest news, thus, the user can monitor a latest-news stream, instead of navigating a structure to find all new items, or to formulate a search that presents the latest items. Two common streams on online news sites are the latest news stream and the headlines stream, showing selected contents of currently high news value.

### Genre analysis and cases

A genre analysis was re-used here, which was previously conducted to identify the core news-carrying elements of online newspapers and their composition, with the sample focusing on front-pages (Ihlström & Lundberg, 2004). A complementary analysis was also done on article pages (Lundberg, 2004). Also, an analysis was made of blogs and discussion forums. Blogs typically have a simpler structure, consisting of a front-page, optional section pages, and entry pages. This analysis was less through, sampling a set of randomly selected blogs and discussion forums, and the main blog engines, such as Blogger. Discussion forums included a forum based on the Venice engine (electricminds.org), the dagensskiva.com forum layout, and also the popular phpBB engine.

For **online news**, the core elements were a) the **news stream**, presenting news according to publishing time. The b) **headlines** element presented recent news, ordered by importance as decided by editorial staff. Also, c) the **navigation** elements, presented the main sections of the papers.

For **blogs**, the main element was the a) **blog entry stream**, presenting the full text of blog entries ordered by time, and sometimes grouped by date. Moreover, there were b) **blog entry comment streams**, presented on blog comment pages, with the main blog entry on top.

For **discussion forums**, the main elements were a) the **discussion stream**, composed of lists of presenting the full text of discussion entries, ordered by time. In some cases, all entries were presented on one page, and in other cases, the stream was split over several pages. Another element was b) the **topic overview** listing discussions. The discussion overview was organized either in the order of appearance with the latest on top, or by the latest topic with a post in it, and could sometimes be re-ordered by the user. The composition and style of the elements differed between the different papers, blogs, and discussion forums.

Using these elements, two newspaper layouts were created, together with one blog layout and one discussion forum layout (Figure 1). The blog layout also featured a genre-mix, a front page for several blogs with a headlines element and a latest entry blog stream, presenting the latest entry for each blog (Lundberg, 2006).
The shapeCMS content management system

The shapeCMS content management system features a 5-layer separation of form and content:

0. Contents: Each content container can have any number of metadata entries. Any metadata entry can be used as site structure metadata. Thus, a specific content can be associated with any number of locations at the same time. (Metadata is described by ontology, metadata, position. Position is only used for site structure metadata.)

   1. Site structure layer: defines a metadata set, and hierarchical relations in the set, for instance ice hockey being a subclass of sports.
   2. Connection layer (site structure and page structure connection): selection of page structure patterns for a location metadata set
   3. Page structure layer: selection and composition of page elements
   4. Element layer: content selection and markup
   5. Presentation layer (visual style of elements, e.g. XSL-FO+FOP, XSLT+CSS)

The system builds on the idea, that search can generate the data sets needed for each view – given metadata that enables precise searches. Thus, the system can generate page elements from the search space. Since many contents can be associated with the same location metadata, a position attribute can be used to order the contents. This enables streams to be generated based on position (which can be static or edited). Editing static positions, and adding and removing tags from items, enables the headlines stream. Using just the location tag, ignoring the position tag, but instead using a creation or publishing time tag, creates a latest news stream. Site structure is defined by any metadata set (layer 1) interpreted as location metadata. That means that hierarchical relations (e.g. chess being a kind of game) define sections and subsections. If a metadata set is instead interpreted as for instance stories (e.g. about an athlete taking drugs in the Olympic Games), several items tagged with the same metadata form a hypertext.

To render the site structure (layer 1) as a site, it is necessary to have descriptions of the page to be shown on each location (layer 3), and to associate each location with a page description (layer 2). Each page description contains references to a number of element descriptions (layer 4). Each element description firstly contains a database search (number of items to show, how to order the items, and what location metadata to use, with the default being the same as the location metadata set for the site). Secondly, it contains selection rules and markup for each content item, for instance to only show the heading of each item and to mark it up with a heading tag. Using the markup tags, the presentation layer (5) assigns a visual style to the contents.
Page generation thus follows the following sequence
1) Match file type (for instance pdf or html)
2) Interpret URL as location metadata. (layer 1)
3) Find page layout for URL, file type, and user rights (layer 2)
   (shapeCMS allows different connection layers to be used depending on user rights and file type)
4) Find page element layouts referenced in the page layout (layer 3)
5) Display database-driven elements and items stored in the file system. (layer 4)
   (display cached items instead of creating them, if they are not invalidated by an update)
   a. For each database-driven element: i) Select database contents, ii) Display the head of the element, iii) Display the body of the element, iv) Display the foot of the element
   b. For each item stored in the file system: Load it, transform it (optional) and display it
6) Presentation transformation (for instance, create PDF from XML, or create HTML from XML, browser transforms page by applying CSS.). (layer 5)
7) Browser requests other media, for instance images

Note that subsequent requests for the same page can be taken from a cache, containing all already generated pages. Thus, pages only need to be rendered again after changes have been made.

**Discussion**

Figure 1 and in more detail in Lundberg, 2006, show the core elements identified for news sites, blogs, and discussion forums, when rendered. For the news layouts, that is the navigation and headlines element (the news stream is also shown in (Lundberg, 2006)). The purposes of these elements are to support navigation, to provide an overview of the contents of the papers, and to provide an overview of recent news with high news value. Considering variation, the headlines element is shown in three variations on the two pages. All three use the same basic organization principle, ordering by news value as assigned by a human editor (layer 4) and select content from the same basic set (layer 0). They vary mainly regarding form (layer 4). Whereas the top news page show headlines and puff for the first item, and only headlines for other items, the bottom page shows the headline and puff for all items. The bottom news page also shows related items regarding the same “story”, for the top item. Those items are associated with that particular item, rather than with the stream. The lower part of the top news page show a headlines elements divided into a table of contents. These contents are also selected using a different editorial valuation than the top headlines (layer 4). Thus, this layer allows a very different form of contents to be displayed (variation), and to change it without any editorial intervention for the contents themselves (layer 0) providing the basic markup is used in the data structure (e.g. to select the “puff”, a puff must be defined at layer 0). Also, the page views can be altered without altering the contents, by altering the page patterns (layer 3). However, creating a completely new organization by editing or creating new location metadata (layer 2) also demands editorial intervention also at the contents themselves (layer 0). Moreover, to alter the visual style of the elements, layer 5, the end transformation can be changed, preserving the other layers. Thus, each layer enables a specific kind of genre change, and also variation, in the form, composition, and visual style of the contents. The only layer that mixes form and contents is layer 4. That layer could be subdivided further, although from a genre perspective, this layer defines genre elements and therefore it might be more useful as a single layer. This exemplifies the four principles of information architecture and research on digital genres, namely search, structure, hypertext, and streams. Implementing all four of these indicates that the system architecture is relatively general.

However, other kinds of genre change, such as the introduction of new organizational principles for streams, can demand new functionality and/or new metadata in addition to new search criteria. In Table 1, the selection and ordering principles of the elements used here are first summarized. Then, I list other possible elements. Considering, for instance, a most highly rated items element, this elements demands functionality for rating items, and that the ratings are stored in the database. As another example, a more of the same element demands that user behaviour is tracked and stored. Thus, the system must be extensible with new functionality and the database format must be changeable, to allow this kind of genre change regarding new items with different selection and ordering principles. Also, if a new system is ordered, the system requirements must contain the selection and ordering principles that are to be used in the first site version (at least), since this example shows that some kinds of streams demands certain functionalities and data structures. This also has implications for the practice of testing systems with users very late, since if their expectations are violated by the omission of some core genre element, then this affects system functionality, and not only the presentation of contents already selected. If done too late, it might cause unforeseen delays in the development process. Also, for all genres, additional elements are commonly used, in combination with the core elements.
## Genre-aware Systems Development

<table>
<thead>
<tr>
<th>Element</th>
<th>Selection and ordering principle</th>
</tr>
</thead>
<tbody>
<tr>
<td>News Stream</td>
<td>Order by creation time (ascending, descending)</td>
</tr>
<tr>
<td>Headlines Stream</td>
<td>Order by number assigned to the item/metadata (ascending, descending)</td>
</tr>
<tr>
<td>Blog entry stream</td>
<td>Order and group by date first, and then by time</td>
</tr>
</tbody>
</table>

**Other possible elements (examples)**

<table>
<thead>
<tr>
<th>Element</th>
<th>Selection and ordering principle</th>
</tr>
</thead>
<tbody>
<tr>
<td>Personalized stream</td>
<td>Selection associated with specific user, or user group, rather than with only a location</td>
</tr>
<tr>
<td>Most read / bought stream</td>
<td>Selection associated user behaviour, such as page views, for some set of contents.</td>
</tr>
<tr>
<td>Most highly rated</td>
<td>Selection associated with user ratings of items.</td>
</tr>
<tr>
<td>Most-read, for a specific search</td>
<td>Selection associated with page views, in the context of a specific search.</td>
</tr>
<tr>
<td>More of the same</td>
<td>Selection based on a category (metadata), presenting more of the category of which the user has viewed the most items</td>
</tr>
<tr>
<td>Most linked to</td>
<td>Selected based on the number of associated items, using some metadata as criteria</td>
</tr>
<tr>
<td>Most-read, version</td>
<td>Most-read, restricted to a specific version of the content</td>
</tr>
<tr>
<td>Most-read today</td>
<td>Most-read, restricted to a specific date</td>
</tr>
</tbody>
</table>

**Table 1. Elements**

Considering the presentation layer (layer 5), creating a new format, might need new functionality to achieve the transformation from the content with markup to a presentation. It should also be noted that a genre change considering the style of contents (layer 0), for instance a change in popular language, would most likely demand editorial work to achieve the change, whereas the system might stay the same.

## Conclusions

Genre-aware systems development can both provide initial requirements (core elements) and make the system flexible enough to cope with genre change. The five-layer architecture described here, which was implemented in the shapeCMS system, shows that each layer contributes with an aspect of flexibility to change. The layers separating form from content (layer 0) were, (1) the site structure layer, (2) the connection layer, (3) the page structure layer, (4) the element layer, and (5) the presentation layer. The system was illustrated by four example sites, which made use of the four organizational principles of information architecture and digital genres, namely search, structure, hypertext, and stream. Also the analysis of genre elements, and potential new elements, shows that new elements will demand new functionality, rather than merely be achieved by a new search request to the database. That both emphasises the need to include the core elements for a genre in the initial requirements, and the need to have a system open to extensions in functionality. Moreover, the analysis shows that some changes to a genre will demand editorial work, but will otherwise leave the system unchanged.

## Acknowledgements

Thanks to Mattias Arvola for discussions on the concepts used to describe the system layers.

## References