eCompass: A Semantic Browsing Tool For Faceted Government Web Content

Assia Alexandrova[1] & Borislav Iordanov[2]
[1]Miami-Dade County, Government Information Center
[2]Kobrix Software, Inc.
{assia}@miamidade.gov
{borislav}@kobrix.com

Abstract

We present a tool for adaptive browsing within a large, decentralized government portal. A multifaceted information architecture is adopted for both the design of a standard, semi-stable information portal and a highly dynamic, contextualized, lightweight navigational aid.

Background

The miamidade.gov portal serves as the primary gateway to government information for the constituents of Miami-Dade County, Florida. It offers online transactions and service information from over 65 County agencies, which cover a comprehensive array of functions – from public safety (police, fire rescue, emergency management), to cultural activities, environmental preservation, social services, transportation (air travel, sea travel, and public transit), animal services, building and construction permitting, zoning, economic development, and many more. The volume of information on miamidade.gov has grown threefold over the last 5 years, and the County web publishers have been faced with the challenge of organizing the ever-increasing body of knowledge about Miami-Dade in a meaningful and comprehensive fashion.

Miamidade.gov represents a two-tier approach to information management. The first tier is comprised of over 250 websites maintained by agency webmasters and web publishers. The second tier is the enterprise web portal application, which displays brief summary items that introduce key contents from the agency websites in the first tier. The portal consolidates a typical large “conglomeration” of government websites from multiple subject areas and aims to fulfill the needs of multiple constituency types. To be effective and useful, the portal must balance the decentralized, dynamic mode of content contribution by multiple webmasters, but also achieve consistency in the presentation of content, through an information architecture choreographed for usability.

Unlike global internet portals and content repositories, miamidade.gov is an e-government portal exclusively focused on local issues, events and government services. Thus, the content remains generated by the County organization, for the most part, and its stable scope permits the careful, controlled modeling of the information space where depth needs to be a primary consideration.

Naturally, the organization of the portal content aims at ensuring that constituents of all types (e.g. residents, visitors, and businesses) can find what they need as quickly as possible. A comprehensive information architecture provides a foundation upon which the portal user interface is constructed and user-centric tools can be built. While a standard portal interface with a relatively stable layout is essential, a highly-structured information space provides opportunities for more intelligent and dynamic navigation tools. In the following, we will present one such tool developed at Miami-Dade County called eCompass. We begin by describing the portal information architecture.

Information Architecture

One of the main difficulties in information architecture for the web lies in the widely varying perspectives of users with different backgrounds and intents. The impossibility of satisfying all users with a single content classification model has resulted in “flat” classification (or tagging) schemes, where content is annotated with keywords without much further semantic import. "Folksonomies” emerging from collaborative tagging could well be applied to an enterprise or a government portal, but they are fraught with lack of “technical”, or domain precision and other usability issues (Mathes, 2004). In search mode, relying on an abundance of keyword variations to help the user find results has its benefits. But when the user browses from content item to content item, the more choices are offered the greater the likelihood of “conceptual overflow”. Thus, we have opted for a faceted information architecture with multiple independent categorization dimensions (as opposed to a single menu-like hierarchy), while still following a carefully chosen controlled vocabulary with clearly defined meaning.

The selection of facets for the multi-dimensional categorization of the Miami-Dade County content (depicted in Table 1.) is specific to our portal and it is based on many years of experience and user feedback. It reflects a public service delivery “model of the world.”

Facet Name	Description	Examples
Persona	Personas represent abstract groups of County constituents with common roles, characteristics or interests.	Realtor, Snowbird, Senior Citizen
User Goal	A user goal is a specific life aspect, or life objective which may be accomplished by a series of tasks/transactions online, or which provides a (demographic) characteristic of the user or makes explicit their preferences.	“making a difference in my neighborhood” “obtaining lower cost services and goods”, “ensuring a quality education for my children”
Landmark	A landmark is a popular public location often used as the site for large-scale and/or important community and County events. Landmarks do not have to be County, or Government facilities, they can also be open spaces, but they cannot be general geographic areas.	Performing Arts Center, Dadeland Mall, Vizcaya Gardens, South Dade Government Center
Actor	An actor can be either a specific individual, or an existing community or government organization. As opposed to the persona, which is a generalization of a group of users, actors are “real”, named people active in the community.	County Mayor, County Commissioner for District 6, Department of Environmental

		Service Management, United Way of Miami
Genre	A genre is a document, or asset type (not a file format). A genre is usually a structured document, and/or template, used for a particular business purpose. Two documents of the same genre will usually have the same format, and will follow the same pattern.	Press releases, FAQs, contact us directories, application forms, checklists
Section	A section is synonymous with portlet, i.e. it signifies the “box” on portal where content should appear.	Event Highlights, News, Legislation and Policy
Importance Type	The Importance type refers to the reason why a particular content item needs to have high visibility on the portal	County service interruption, natural disaster, time-bound community event, Legislative change
Program	A program is a service, or collection of services provided by a County agency to the public.	Dial-a-Life, Viernes Culturales, Government on the Go
Topic	A topic represents a general category of subject matter, or a specific life domain. Topics may have subtopics (level 2 topics).	Pets and Animals, Environment, Public Safety

Table 1. Miami-Dade County Facets

Almost all facets are multi-valued with the exception of Landmark and Program and only the Topic facet is hierarchical (i.e. topics are organized in general-to-specific relationships). In addition, some facets can themselves be categorized by other facets. For instance, both the Persona and Topic instances can have a set of User Goals associated with them.

Content items, such as news announcements, and online services can be tagged with any combination of these facets, which allows for the creation of distinct pages, or sub-portals, for each different perspective – for instance, there are pages for the Resident, Business, Activist, etc. personas, pages for each major topic group – e.g. Art and Culture, Public Safety, etc . The same information can therefore be accessed through numerous portal entry points which accommodate different user approaches to locating information, and follows the associative model of information retrieval (Fuchs & Rosati, 2005).

Inevitably, the tagging process is somewhat biased towards the organization of the portal web site itself. To ensure the listing of an item in a particular spot on the portal, publishers must be careful in annotating the item appropriately. In order to prevent this bias from constraining the tagging process to a particular perspective (that of the portal UI structure, or, ultimately, the institutional perspective of the County agencies), some facets like Section specifically model the portal structure. This allows us to separate the two main goals of the information architecture: (1) organizing the display of content in the portal in a fairly stable and predictable way (2) organizing content in an abstract conceptual space for purposes of search and adaptive, contextualized navigation..

Problem

Government sites are often constructed in such a way that the user must get acquainted with an introduction to a service before they can complete an online transaction, access a form, or perform anything consequential. Even when the targeted information is just 3 clicks away from the homepage of a site (Zeldman, 2001), the time to investigate content at each “click point” and the extent of content (typically written in isolation) that needs to be examined so that one can determine the next link to be clicked, confuses and overwhelms the user significantly. This leads to a sense of “lostness” (Gwidzka & Spence, 2005) far more frequently than in commercial, entertainment or other types of sites where background domain knowledge is not a prerequisite. Yet, constituents access government websites with a utilitarian motivation (Hansen, 2007), and typically seek to resolve a specific issue which has a practical impact on their lives. When they need to find information to achieve a concrete goal, they can easily get frustrated when they don’t locate it immediately, or when they don’t find it using their own terminology (e.g. “business license” vs. “ local business tax receipt").

In this context, we have identified two main causes leading to user dissatisfaction. First, the frustration associated with failure to locate needed information is due to the effort spent during actual browsing - the logistical work of page loading and associated browser behavior in addition to the inspection of the content. As evidenced in (Pirolli, 2005), web surfing behavior is characterized by ongoing cost-benefit assessments by the user of the utility of a particular piece of content. This offers a slightly different perspective on improving usability: minimize the cost of navigation itself. Secondly, while the multiple perspectives of content organization offered by the miamidade.gov portal increase the likelihood of the user successfully reaching their goal, the portal user interface ultimately embeds a “push” model whereby potentially relevant content is “prepackaged”. In itself, the model is valuable since it ensures a level of predictability of the display, but it is insufficiently reactive to user intent.

Therefore, recognizing the impossibility of a complete on-the-fly adaptation to a user’s worldview, we wish to facilitate the converse: offering the user a means to quickly and painlessly grasp the structure of the information space by extending the “space” metaphor through contextualized, "low cost", map-like explorations.

Approach

In the information seeking endeavor, searching and browsing are complementary activities and many systems have attempted to integrate them in a meaningful way (for examples see Quintarelli et al., 2006; Zhang & Marchionini, 2005). Our approach is a variation on this theme, based on the observation that "searching" can be seen as "filtering for relevancy".

We developed eCompass - an interactive graphical semantic navigation tool, similar to Visual Thesaurus (http://www.visualthesaurus.com), which allows users to explore the portal information space in a goal-directed manner, without having to browse the actual websites and get “locked” into a particular area.

In eCompass the user can enter search terms, focus in on the resultant content items, represented as interrelated nodes, and view other items which are closely related, or similarly classified. The tags annotating the content items are also represented as nodes in the interface, and by clicking them, the user will increase the weight of a particular category, and thus promote the visibility of items with equivalent metadata. When the user chooses to do so, they can view the webpage associated with a given content item in the browser.

Thus, eCompass is a dynamically-constructed browsing and navigation aid. The current page represents the focus of attention around which a "zoomed out" view of its proximal information space is displayed. The metadata associated with the current item serves as a set of constraint variables filtering the information space for relevance depending on the current context. The context is an aggregate of the user's profile, browsing options and browsing history expressed as a weight matrix on the tagging dimensions (a.k.a. facets).

The nodes and their relationships are presented graphically as an interactive hyperbolic tree (see Figure 1) with the following behaviors:

- When the user navigates to an item, a dynamic local map of similar items is constructed and displayed.
- Metadata nodes are marked with icons, which help the user visually identify what is metadata and what is actual URL-addressable content.
- Dragging a node, or and edge zooms in and out certain parts of the graph. If a metadata node is clicked, or the metadata slider is used, the graph recalculates the semantic tree rooted at content item currently in focus.
- Mouse hover over an item node displays the abstract of the content item, as recorded in our content management system.
- Mouse hover over a tag displays a description of that tag.
- Double-clicking on a content item node loads its webpage URL in the browser.

By “moving” from node to node, users will figuratively follow an information scent trail, which is dynamically defined and fine-tuned by all their navigation “gestures.” Other than clicking, double-clicking and dragging of the nodes, such gestures comprise manipulation of additional utilities that are part of the eCompass interface. Such utilities include: a history panel, which displays all the actions the user took, so they could step back to a previous navigation point; a metadata control panel, which allows the user to manipulate weights for all the metadata, zoom in and out of the current perspective, and other similar spatial operations, not unlike a GIS application toolbar; a search panel, which enables a new search, as well as the addition of terms to supplement an existing search.

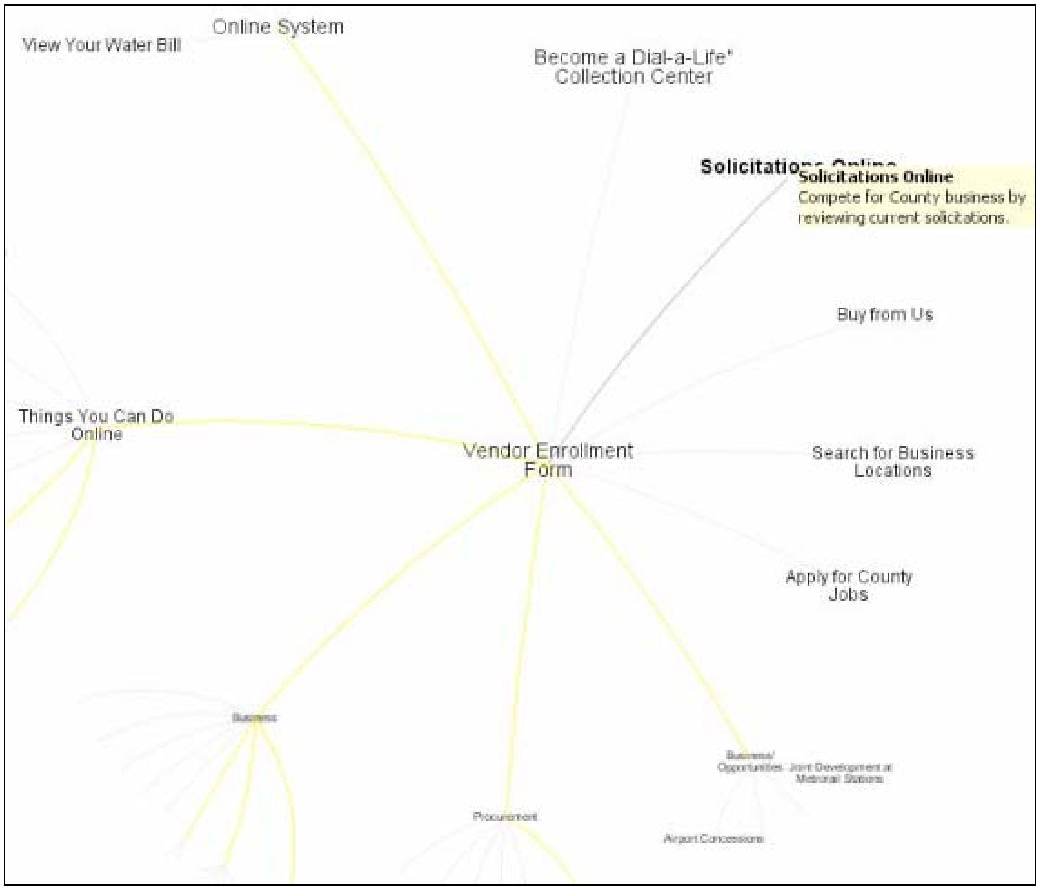

Figure 1. eCompass Hyperbolic Tree Partial Display

It must be noted that the hyperbolic tree is essentially a hierarchical display – it starts with a root node that branches out into sub-nodes where each branch corresponds to a semantic category (a metadata tag). However, the original data is not endowed with any explicit hierarchical structure since items are related through common tags in unconstrained ways. The tree-like categorical sub-divisions in the display are created dynamically as part of the visualization algorithm described below. This is analogous to the hierarchical categorization people apply when describing a particular domain of interest. In fact, according to cognitive psychology theories (Palmer, 1977; Car, 1998), humans are most at ease with hierarchically structured mental models. This cognitive limitation has translated into the static categorization of subject matters in primitive information architectures, which ultimately fail to capture all user perspectives, since the mental model of a user at any one time, depending on their current attention focus, may start at a completely different top-level category (modulo terminological differences). Categorizing along multiple dimensions, as done by facet-based information architecture, only solves the problem partially, since the user is nonetheless more comfortable with a hierarchical structure: their own hierarchical structure. Hence, it appears that a model where the information space is both dynamically and hierarchically-structured, and filtered for relevance to the task-at-hand following a high-dimensional categorical space, should be most helpful to users. That is the reason we have adopted the above strategy in eCompass.

The eCompass display design is based on the hypothesis that a hierarchical diagram is more amenable to direct perception, and leads to an unconscious, yet more effective internalization of the categorization of related items. While humans can simultaneously perceive complex relationships in their physical environment with ease, the communication of symbolic information (named entities, categories, labels, etc.) may involve some primitive, non-verbalized form of logical reasoning wherein a hierarchical sequence of sub-divisions of the examined space is grasped more quickly.

Implementation

This section describes the technologies and main algorithms used in the implementation of eCompass. All of the implementation components are either open-source projects or freely available upon request.

The tagging model is represented in the Topic Maps standard (http://www.isotopicmaps.org/). Each facet is represented as a topic that plays a type role having as instances all the tags belonging to that facet. Following a more or less standard practice in the topic maps community, tag descriptive attributes are represented as occurrences of its corresponding topic. Each item in the content database is represented as a topic as well and a separate topic-to-topic association is created for each tagging of the content item. Relationships between tags themselves are also represented as topic associations yielding a uniform representation of the information landscape. Furthermore, the design and implementation are independent of the Miami-Dade County domain and generally applicable to any high-dimensional categorized content repository. The topic implementation on the server-side is based on the HyperGraphDB database, a general purpose, transactional, in-memory database for storing and querying complex structures (http://www.kobrix.com/hgdb.jsp).

eCompass is implemented as a Java applet that communicates with its opening browser window and with the miamidade.gov server. Context changes are triggered by user click behavior, search options selections and the like. Each context change results in the dynamic construction of a proximity topic map on the server side which is then transmitted, following the standard topic map XML representation (http://topicmaps.org/xtm/) to the applet for display. The hyperbolic tree display is implemented by the hypergraph library (http://hypergraph.sourceforge.net) and the in-memory Topic Maps implementation TinyTM (http://tinytim.sourceforge.net/).

Thus, user actions trigger context changes which translate into the display of a new perspective on the portal content. The translation process involves two stages. At the first stage, we construct the set of most relevant items based on the current context. At the second stage, this set is partitioned hierarchically starting at the categories of the most relevant item. Thus, the focus item, when there is one, or the "best match" serves as the center of a contextualized semantic halo.

To compute relevance, we use a slightly complex weighting schema where each tag is assigned a pair of weights: the *system weight* measuring the relative importance of the tag within the information architecture and the *activation weight* measuring the importance of the tag for the current browsing session. Weight assignment works as follows. First, each of the $F_1, F_2\ldots$ facets has a predefined administrator assigned weight which is an arbitrary positive number reflecting the importance of the facet. Facet weights, denoted $W(F_i)$, are normalized to fit the (0, 1) interval. Then, the frequency of each tag within a facet gives rise to an inversely proportional tag weight denoted by $W(t_i)$[1]. The system weight of tag t is then $W_S(t)=W(F_t)*W(t)$, where F_t is the tag's facet. The activation weight is computed by maintaining for each tag an activation frequency $A_f(t)$ reflecting the relative number of clicks on the tag or on an item visually categorized under it. For logged in users, the activation frequency is initialized from the user profile, otherwise it starts at 0. The activation weight is then $W_A(t)=W(F_t)*A_f(t)$.

The system weight and activation weight are not directly comparable. It is important to keep those weight measurements separate as they follow different scales and the logic behind them is different. Thus, the weight of a tag is the vector $w(t)=(W_S(t), W_A(t))$ and weight accumulation is performed by vector addition. Because weights are not scalar numbers, we need to define an order relation. The following function implements ordering between weights:

```
compare(w,v)

Input
    w: a weight vector (x,y)
    v: a weight vector (a,b)
Output
    A number which is negative, 0 or positive depending on whether w is
    less than, equal to or greater than v.

If x < a and y < b  Return -1
Else if x > a and y > b  Return 1
Else if x > a Return x/a – b/y
Else Return y/b – a/x
```

Figure 2. Weight Comparison Function

A weight vector is less than (or respectively greater than) another if both of its components are less than (or greater than) both components of the other. We distinguish ambiguous cases by comparing ratios between components. If we wish to accord more credence to the active (or system) weight, we can just multiply the corresponding ratio by the appropriate factor. Whenever we perform sorting by weights in the algorithms below, we are using the compare function provided in Figure 2.

Given a set of active tags, that is tags with $W_A(t) > 0$, the information space is filtered for the most relevant items annotated with them. A contextual neighborhood is constructed according to the following algorithm:

```
getNeighborhood(tagSet, maxRelated)

Input
   maxRelated : Result set cardinality.
   tagSet:      The set of active tags.

Output
   neighborhood: the set of most relevant items according to the
                 current semantic context.

Let potentialWeight = Sum(w(t) foreach t in tagSet)
Foreach t in tagSet in descending weight order
    For each item I tagged with t
```

[1] Inversely proportional because less frequently used tags are assumed more specific and should therefore carry more weight.

```
        If I is in neighborhood already
            update w(I) = w(I) + w(t)
        Else
            Add I to neighborhood with w(I)=w(t)
            If (size(neighborhood) >= maxRelated and
              min(neighborhood) >= potentialWeight) then
               exit for loops
   update potentialWeight = potientialWeight – w(t)
return top maxRelated elements from neighborhood
```

Figure 3. Content Neighborhood Construction Algorithm

The algorithm proceeds optimistically by first examining related items that have "heavier" active tags. It maintains the maximum possible weight of newly discovered items in the *potentialWeight* variable and exits as soon as we have accumulated enough items and no unexamined ones could have a weight greater than the current minimum.

The second stage consists of building the semantic tree rooted at the focus. While the first stage serves a purely filtering purpose, the idea here is to dynamically create a hierarchical view of the set of items selected as relevant. It is implemented by the following algorithm:

```
buildTree(branchNode, itemSet, iFanOut, cFanOut, tagPath)

Input
    branchNode: Tree node at which the current tree branch is being
                constructed.
    itemSet: Items to display at this tree branch sorted in decreasing
             weight order.
    iFanOut: The number of items to display as children at each branch.
    cFanOut: The number of categories to display as children at each
             branch.
    tagPath: List of tags on the path from the focus to this branch.

For each item I of the top iFanOut elements in itemSet
    Create new node for I and connect it to branchNode

Let newItemSet = itemSet without the items just displayed

Let S = empty set
For each t in the "heaviest" cFanOut tags not in tagPath
    Add t to S
For each t in S
    newTagPath = tagPath with t appended at the end
    newItemSet = sortWithAdjustedWeights(newItemsSet, newTagPath, S)
    tNode = create new node and connect it to branchNode
    builtTree(tNode, newItemSet, itemFanOut,
              categoryFanOut, tagSet, newTagPath)
```

Figure 4. Semantic Tree-Construction Algorithm

The algorithm recursively constructs a set of overlapping but visually disjoint spanning trees of the topic graph. Weights are used to follow the Gestalt rule of proximity that indicates that items physically close together are perceived as being closely related. The outcome is visually presented as a set of overlapping hierarchies radiating out of the focus node where each branch is labeled by some tag. Items are frequently replicated at different top-level branches, but they appear at varying distances from the main focus depending on the path of tags leading to them. This is modulated by the *sortWithAdjustedWeights* function that reorders the set of remaining items to display according the current tag path from the root. The function looks as follows:

```
sortWithAdjustedWeights(itemSet, tagPath, siblings)

Let n = tagPath length
for i = 1 to n
    update W_{S,A}(t_i)= W_{S,A}(t_i) + i* W_{S,A}(t_i)/2*n
    (where t_i is the ith element in tagPath)
For each t in siblings
    update W_{S,A}(t)= W_{S,A}(t) - W_{S,A}(t)/2
resultSet = sort itemSet with the new weights
restore the old weights
return resultSet
```

Figure 5. Contextual Weight-Adjustment Function

The weight adjustment ensures that items appearing directly under a given branch are not likely to appear directly under sibling branches. It essentially "bumps up" the weight of a tag relative to its siblings purely for display purposes so that items distributed redundantly at different branches have their copies distanced visually.

Avoiding redundancy in the graph by simply displaying the connections as they are results in a highly cluttered, hence cognitively unbearable, diagram of relationships. Another option that we have explored is hiding the tagging information from the display altogether. This results in a much simpler display with a central item node and set of related items laid out in a spiral form (radiuses increase with the similarity distance), which might as well be displayed as a search result-like list. But then the categorization structure must be dynamically obtained through additional actions (i.e. mouse roll-over, clicks and selections), which requires the extra mental effort of keeping track of the categories of previously examined items. And if tags are listed together with the items, in a conventional tabular format, the display becomes cluttered with text and difficult to read since for each item that is being "scanned" visually, its categories must be read separately.

Conclusions and Future Development

A multifaceted information architecture yields many benefits in organizing a diverse and large content base such as the ability to create a flexible information portal integrating multiple perspectives. By capitalizing on a richly structured foundation and on the natural human cognitive capability to combine spatial perception with mental concepts, we created an alternative content navigation option that is both lightweight and adaptive. As shown in (Glover et al., 2002), metadata about a document often has a greater discriminative and descriptive power than text in the target document itself. eCompass offers an easy grasp of the "aboutness" of information assets from all domains featured on the e-governmental portal. Its dynamically synthesized hierarchical view substantially eases the cognitive burden of exploring the real underlying relationships.

Our current work and future plans focus on tracking usage of the tool by utilizing standard measures such as number of user sessions, average session length, number of referrals from eCompass to County URLs, and basic survey measures such as user ratings of the tool's usability and overall utility. In addition, we are planning on a pilot project to allow constituents to collaboratively tag content on their own, which could allow us to quantitatively assess the relative performance of controlled vocabulary vs.

folksonomy-based tagging. Further improvements involve integration with a semantic search engine and leveraging the topic maps scoping formalism to create alternative views on the facet ontology.

References:

Car, A. (1998). Hierarchical Spatial Reasoning: a GeoComputation Method. Available from: http://www.geocomputation.org/1998/18/gc_18.htm

Fuchs, S., & Rosati, L. (2005). The Italian and the English model of information retrieval used in the governmental websites. Available from: http://trovabile.org/download/information-architecture-for-public-administration.pdf

Glover, E., Tsioutsiouliklis, K., Lawrence, S., Pennock, D. & Flake, G. (2002). Using web structure for classifying and describing web pages. *Proceedings of the 11th international conference on World Wide Web*, 562-569**.**

Gwizdka, J., & Spence, I. (2007). Implicit measures of lostness and success in web navigation. Available from: http://www.psych.utoronto.ca/users/spence/Gwizdka%20&%20Spence%20(2007).pdf

Hansen, S. S. (2007). Web Site Usability Drives Credibility And Citizens' E-Government Attitudes. In *Proceedings from the International Communication Association 2007 Conference.* Available from: http://www.oracular.com/assets/docs/ICA07egovtweb.pdf

Mathes, A. (2004). Folksonomies - Cooperative Classification and Communication Through Shared Metadata. Available from: http://www.adammathes.com/academic/computer-mediated-communication/folksonomies.html

Palmer, E. S. (1977). Hierarchical Structure in Perceptual Representation. *Cognitive Psychology, 9*, 441-474.

Pirolli, Peter (2005). Rational Analyses of Information Foraging on the Web. *Cognitive Science*, 29:3, 343-373

Quintarelli, E., Resmini, A., & Rosati, L. (2006). FaceTag: Integrating Bottom-up and Top-down Classification in a Social Tagging System. Available from: http://www.facetag.org/download/facetag.pdf

Zeldman, J. (2001). Taking Your Talent to the Web: Making the Transition from Graphic Design to Web Design. New Riders Publishing.

Zhang, J. & Marchionini, G. (2005). Evaluation and evolution of a browse and search interface: relation browser. *Proceedings of the 2005 national conference on Digital government research, 89,* 179 – 188.