Perspectives on Taxonomy, Classification, Structure and Find-ability

Thoughts from the Consortium for Service Innovation, a work in progress by Greg Oxton, John Chmaj and David Kay.

> *"We are all in pursuit of relevance."*
>
> *– John Chmaj*

The goal of using taxonomy, classification schemes and structure is to improve our ability to find relevant content in a large collection of content and to improve our ability to learn from the patterns and trends that emerge from a large collection of content without manually review each piece of content.

The Goal—Complement (Not Replace) the Human Mind

The human mind has an amazing ability to make connections and sense out of information, observations and experiences, which to a machine (or program) would appear to be totally unrelated. This is in part because of the subtleties of context and our mental ability to infer, interpolate, extrapolate and interpret our experiences and interactions at multiple, non linear levels of conceptualization. In addition to multi layered conceptualization the human mind benefits from all five senses. A smell, a sound (or song), a sensation or feeling can each, or in combination, trigger a connection to experience. Machines struggle to deal with multi-dimensional concepts and non-linear relationships, and they are largely limited to a single "sense" in the form of text. Machines have a sever disadvantage due to the imprecise nature of language we use to represent thought and concepts. At the same time, the human mind does not have complete or perfect recall of all that we know; machines do.

The goal is to create a knowledge practice that complements what individuals know with what the collective experience of the organization or community. To accomplish we are seeking to create content that is good enough to be findable and usable by a target audience. Taxonomy, classification and structure all play a role in find-ability.

The Difference Between Taxonomy, Classification and Structure:

- Taxonomies attempt to be exhaustive; they position all known things in a hierarchy and/or relationship map. One key point here is that taxonomies have evolved from organizing the relationship between physical things (tangibles) like living things or the earth's fundamental elements. We are now applying taxonomies to abstract things (intangibles) like the meaning of words and phrases or concepts. Abstract things by their very nature have an element of ambiguity that makes it hard to apply the same structures, definitions or relationship mapping that works well for the physical world.
- Classification is a way to organize things or objects into categories or buckets based on similarity. Classifications are generally not exhaustive. In practice, the categories are often predetermined. Generally, they have one or two dimensions at best. That is, the categories are exclusive: an object can be in category A or B but not both.

- Structure (as we talk about it in KCS terms) is a simple, less rigid or absolute way to organize information and does not require that we anticipate all the potential things that might fit into a level of the structure. Structure in the KCS model gives the words and phrases some context or role that improves readability. If the search engine uses the structure, it can help find-ability.

Taxonomy (a definition): a structure for classification, nomenclature to describe a catalogue, a relationship map

A wiki on taxonomy http://en.wikipedia.org/wiki/Taxonomy

Taxonomy (examples):
- The classification of living things (e.g. organism/domain/phylum/class/order/family/genus/species)
- The classification of the earth's minerals—periodic table
- Computers—network/system/device/module/part/component

Taxonomy and knowledge—taxonomies used in knowledge tools are often very detailed relationship maps for the meaning of words and sometimes concepts. The content is tagged programmatically such that it aligns with or fits with the map. When searches are done the search engine looks uses the map to identify

Observations on Manual Classification of Knowledge

> *"A sad note: it seems the smaller the target document set a KB technology is set up for, the more likely they are to use category filtering as a primary relevance mechanism. Lazy and sad."*
> *– David Kay*

Classification of content at a certain level can be helpful in improving the relevance of search response:

- The degree to which classification helps is a function of how distinct or separate the domains of content really are. The more distinct or separate, the more value in segmenting them.
- This is not exactly the same as helping users find what they need. Rather, we are excluding content that we anticipate they do not need.
- High level classification of content plays a part, but must be balanced with other system and content functions in the 'pursuit of relevance.'
- Experience has shown that in some environments scoping is essential at the first or second iteration in achieving some form of reasonable subset of content to operate against in more detail. Categories for products, general issues, often content types, and several other potential dimensions can greatly facilitate the initial scope definition of relevant resources to help address an issue. The problem people get into, is overdoing it—trying to get classification schemes to make the full bridge to the subtle, slippery ways in which people try to articulate and resolve problems. So, classification schemes can only go so far, but for some environments, they are essential.

At the risk of oversimplifying, here are the three key things we do at knowledge discovery time with categories:

1. **Filter**—remove results that aren't in the right part of the category tree
2. **Browse**—navigate through content based on categories
3. **Rank**—move matching categories higher up a results list

Done deftly, in theory, category-based ranking is hard to disagree with. In practice, because users will generally only look at a page or two of results (if that), ranking morphs into filtering, so it has to be used as an extra-credit sort of influence, and gently. As we have learned the hard way, if a user gives us five words and we find four of them pretty close together in the document's title, we'd better return that document on top, even if our category smartness tells us something else is better.

Similarly, browsing may be more or less useful, but at least it is benign. The customer's in charge, and as long as the categories map into the user's view of the world at some level (they have good "scent"), then there's nothing but good in this use of categories—mod the accuracy of the classification. Users are not going to use this too much in huge document sets, so it is appropriately self-limiting.

The danger really comes with filtering. We need to focus on this point in a statement of best practices. Using filtering as the key way of driving relevance is incredibly dangerous—this is why the homegrown parametric search engines built on Access or something similar fail miserably as tech support or service desk tools. Not to mention decision trees, which effectively do the same thing. Here again we have to acknowledge that there are some environments (particularly fixed or static ones) where decision trees have value. (For a great decision tree application, check out playing 20 questions with Darth Vader at http://www.sithsense.com/flash.htm.) Unfortunately, most of the support environments we have encountered are extremely dynamic and the relevance and maintenance of the tree structures make them unsuitable.

The problem is that if there is any mismatch at all between how the user and the organization view an aspect of the problem, filtering fails. Silently. We have no idea that we just filtered out a relevant result. At the point of search, we do not know the root cause.

On the other hand, there are circumstances where filtering is useful. Users are capable of knowing the primary product line they are dealing with. Users can know, with some past experience, which content source is most relevant for their search. And, organizations can know this stuff with confidence as well. (E.g., version numbers to which a fix applies.)

Certainly, a best practice for filtering (except for gross product family) is to make it available after the search results appear. (Think search, then browse, the way Yahoo! generally worked in its earliest days.)

So, categories are not bad, but filtering based on categories certainly can be. In a small KB (1000 docs? 3000 docs?), we simply do not need it. In a huge KB (100,000s), it can be helpful, but it needs to be used very thoughtfully based on the environment and an understanding of the context of the users who will be searching. The context of the user is

often hard to anticipate so a process of continuously improving the categories and filters by paying close attention to users' experience.

Additional considerations about classification

- The risk of predefining categories…at what level of detail or accuracy can we anticipate the structure or relationships of things we do not yet know? Wouldn't it be better to let the content self-organize?
- Manual classification of incidents or solutions by support agents; if we have enough categories for the data to be actionable, we have too many for the analyst to deal with…If we have a number of categories that the support analyst can deal with (5-7 items), we have not enough specificity to be actionable.
- Manual classification is not multi-dimensional; it does not support the numerous ways in which content can be related.

Automation Opportunities:

- Programmatically tagging content (creating meta data) based on a predefined taxonomy improves search results and allows us to organize the content in multi-dimensional ways. This is particularly valuable for unstructured content. The taxonomy requires maintenance as new associations are identified.
- Data mining tools that detect patterns in content based on the content! Emerging technologies that can organize content based on what it is without a predefined taxonomy. This allows the content to self organize in ways that we might not have anticipated.

Thoughts on Structure (from the KCS practices)

A format to give content a little bit of context:

- Problem/Question
- Environment
- Fix/Answer
- Cause (optional)
- Metadata (date created, last modified, # of time used, life cycle state

Searching and Find-ability

- Google searches for the existence of words in a blob of text—if we search for "install Linksys router in network with Apple and Window XP," Google will respond with the most frequently referenced documents that include any of those words…e.g. "install windows XP" or "install windows emulator on Apple PC" without regard to the role the words play.
- Rather than searching a blob of text find-ability is improved if we search problem statements against problem statements and environment against environment
    - Problem: install router
    - Environment: Linksys, Apple, Windows XP

  The response will only be issues related to installing a router in this environment.

  A few challenges with this approach

1. Most content is not in the KCS structure, in which case we need one approach to search well structured content and a different approach to search unstructured content
2. Aligning the search engine with the structure can not be absolute, it has to be a balance