Michael Adcock

LIS 537

December 7, 2007

Final thoughts on thesaurus construction

My thesaurus construction experience was somewhat unique, because I not only learned about the processes of building a thesaurus, but also chose to use and customize a piece of software to aid our group in the task. The decision to use and create this software solution came later in the process, and was not immediately accepted by our group members, including myself. However, over time we realized the potential of this solution, and I had a clearer understanding of what changes should be made to the software. Ultimately, our efforts proved to be more successful than we expected, and it became apparent that the tool we used to create the thesaurus was valuable to the end user as well.

We initially approached the thesaurus construction assignment like most of our classmates. We defined a domain and user group that seemed reasonable. (In hindsight, I think it might have been more fun to go with one of our earlier ideas to look at alchemy. However, amateur astronomy was certainly a practical domain, even if slightly less interesting.) We located what we thought were appropriate sources, including books, magazines, and websites, and harvested terms from them. We used Excel to store the terms since it was easy to use and allowed for different types of sorting. In addition to the terms themselves, we included the source of each term, and the position in the source (page number with a section code – for example G127 for page 127 in the glossary.) In a few cases we also included a note that indicated if there was an obvious relationship to another term (that might not be obvious to us later.) Since three of us were working on the project, we decided to import our Excel spreadsheets into a single Google Document spreadsheet online.

It was at this point that I stumbled upon a full list of the terms[[1]](#footnote-2) from the Astronomy Thesaurus[[2]](#footnote-3) online. After some discussion with the group, I imported these terms into our existing list, giving us several thousand additional terms. Although our domain was amateur astronomy, we felt that some general astronomy terms would be important, and the inclusion of this list would help support general terms we found in our more targeted sources. Once we were happy with the ridiculous[[3]](#footnote-4) number of terms we had gathered, we sorted them and collapsed them down into single entries, and added a count value to each term. Because we kept a copy of the original expanded list of terms (with their sources), we did not include source information in the collapsed list. We decided to use the term counts as an indication of literary warrant – if the term had a high count value it had been found in multiple sources. We did not, however, make all our decisions about term inclusion based on term count. It provided a useful rule of thumb later in the process when we were trying to decide which terms to upward post, or when we were faced with finding a single preferred term among several synonymous terms.

Once we had the list of terms with counts, we sorted them and separated out all the terms that appeared more than once. (We kept the list of single instance terms, for possible inclusion later as additional lead-in terms.) The resulting list included about 600 terms, which we decided to try to split into roughly equal chunks for each of our group members. We briefly considered using Soergel cards, but the sheer number of terms we had already collected deterred us. To make things manageable, we divided the terms into somewhat arbitrary groupings like “stars”, “equipment”, “solar” (things in our solar system), and so on. The terms were placed into groups by a quick search for the definitions of each term, although we failed to capture these definitions at the time. Had we saved these definitions, it would have lessened our work later on. These groupings were then divided amongst ourselves so we each owned about 200 terms. The plan was to identify which terms could be excluded from the thesaurus entirely, based on usage. We also wanted to identify definitions for most terms, and through searching for those definitions identify related terms and relationships. In order to reduce our terms, we redefined our user group and decided that more theoretical concepts and objects that could not be seen by amateurs would be discarded. We also decided that all proper names would be discarded and moved into an authority file.

While I had been secretly[[4]](#footnote-5) considering proposing the use of a TiddlyWiki for our project, it wasn’t until we started working on our own sets of terms that I realized the decision should be made at this point. I saved the terms from our Excel file[[5]](#footnote-6) into a comma-delimited file. Then I wrote a quick and dirty script to pull in these terms, their count values, and the arbitrary group labels, and produce a chunk of HTML that could be pasted into a blank TiddlyWiki project file. Since each entry (or “tiddler”) in a TiddlyWiki is just a certain kind of formatted HTML code, this was a fairly easy process. However, it’s worth noting that a future revision of this TiddlyThesaurus[[6]](#footnote-7) should include a user friendly method for importing terms. It’s certainly possible to type them all in directly, but I believe the actual harvesting process can be done most quickly with a spreadsheet like Excel.

Once the terms were in the TiddlyWiki (as tiddlers), and the term counts and arbitrary groupings were applied (as tags), I began to play with the system. The built-in search feature made it easy to find terms, and the other navigation features were useful too. I decided I would try to add functionality to handle relationships, and try to enforce the standard rules that apply to a thesaurus. Before I started modifying any code, I showed the TiddlyWiki to the rest of my group and explained what I thought could be done with it. I planned to only add features that we needed for the project, and keep things as simple as possible. The base TiddlyWiki was rather elegant, and I hoped that any additional features I included wouldn’t damage that elegance. We were all skeptical at first, but we agreed to start using this new system to enter definitions for our terms. Because there was no way to enter the relationships for terms, we postponed this until I got the code working.

We encountered some issues at first, regarding saving, that resulted in the loss of some work. But we worked through these issues, and as I got some of the thesaurus specific features implemented (including scope notes), we all agreed that the tool taking shape before us was becoming more useful. Once relationships could be defined, it was even easier to move through the terms. However, manually verifying that relationships were properly balanced (every RT reference has a RT reference back, etc.) seemed like a time consuming process[[7]](#footnote-8). Checking for such things in code and automatically listing any problems seemed relatively easy. I considered automatically creating the necessary relationships as new relationships were defined, but I was afraid that this could cause confusion and chaos if we made mistakes. So I created a way to check for problems, and generate a report of the issues with links back to the terms that needed to be fixed. As we discovered new issues, I added checks for those as well. I also tried to anticipate other problems that had not occurred and check for those. We ended up with a sort of diagnostic page that will check for, and list, all sorts of potential problems with terms and term relationships.

As we continued to update terms in the TiddlyThesaurus, I started to work on displaying the two schedules automatically. The system contained the terms and their relationships, so it had all the information needed to display the schedules. In both cases, I wrote code that would iterate through the terms and display the schedules properly. At this point, we were still missing several things. We had not defined a facet structure[[8]](#footnote-9), we didn’t have a way to create node labels, and we didn’t have a way to display notation.

Creating a node label concept was as easy as creating the concept of preferred terms – we just tagged the terms with a special “node\_label” tag, and I modified the code to display such terms surrounded by angle brackets. These terms were ignored in the alphabetical schedule, and when they appeared as broader terms, the real term above was substituted as the broader term. A similar process was used for narrower terms.

It was easy to generate notation. Since the hierarchy display in the classified schedule was generated automatically, the position in the hierarchy could be used to define the appropriate notation. The notion we chose was very simple, but it could have been possible to define a more complicated notation scheme. As a matter of personal opinion, I don’t really understand the value of notation beyond the ability to jump from a term in the alphabetical schedule to its position in the classified schedule. It seems to only really be valuable in a printed thesaurus, and less important in an electronic thesaurus. Perhaps I just don’t fully appreciate the purpose of notation.

Although our process surrounding facets was a bit unorthodox, I believe the result was acceptable, and in line with the CRG and other examples. Since the TiddlyThesaurus would display the most up to date classified schedule as we made changes to relationships between terms, we watched the tree structure form as we tweaked individual terms. We spent some time discussing how to apply the CRG facets, but realized that our existing “pieces” of hierarchy already suggested something of a structure. With this in mind, we defined our top level facets after a lot of discussion. Then our process revolved around taking each of the disconnected terms (which appeared at the top level), and choosing where they should be placed in the hierarchy that evolved. As strange as it might sound, this process was actually fairly intuitive. There were only a handful of terms that were truly difficult to situate. These remaining issues were addressed by upward posting, or by a small amount of juggling relationships around. In all of these cases, we referred to the definitions of the terms.

While there wasn’t much planning involved in the creation of the TiddlyThesaurus software, the result was a system that addressed the needs we encountered during the process of creating our thesaurus. The only regret I have about the thesaurus content is the arguably questionable “Observation” facet. This worked for the purposes of the assignment, but I wasn’t really comfortable with it. The TiddlyThesaurus code could also be cleaned up, as my goal during the project was to create reliable functionality as quickly as possible. I also believe that in some ways, using the TiddlyThesaurus encouraged us to create a better thesaurus. I don’t feel that it constrained us unnecessarily, but it forced us to produce a standards compliant thesaurus. It also allowed us to explore changes in the structure of our thesaurus schedules with a minimal amount of effort or risk. Rather than spending a lot of time and effort on monotonous tasks like applying notation, or manually checking broken relationship links, we could focus on the more interesting questions surrounding how terms should be related, and how the structure of the thesaurus as a whole should be built. Sadly, I can’t take much credit for these results – these last couple of benefits were happy accidents. In any case, I thoroughly enjoyed this assignment, both for the procedural challenges of creating a thesaurus, and the programming challenges I faced along the way. It allowed me to move freely between those two worlds, and I realized this is exactly the sort of thing I’d like to do in the “real” world.

1. <http://www.aao.gov.au/lib/thesaurus.html> [↑](#footnote-ref-2)
2. <http://msowww.anu.edu.au/library/thesaurus/> [↑](#footnote-ref-3)
3. After some reflection, I really don’t believe the number of terms we collected was too large. It would have been completely unmanageable without some sort of tool to assist us, but with such a tool I think it worked out well. I think my whole group would agree that we had very good coverage and the initial term count also helped us to decide how to better define our domain and users. [↑](#footnote-ref-4)
4. It’s not like I had been plotting and scheming. Well, maybe a little bit… But I didn’t want to propose using the TiddlyWiki until I was fairly confident it would work. I certainly didn’t want to encourage us to lock up our terms in a system that wouldn’t ultimately do what was required! [↑](#footnote-ref-5)
5. As mentioned earlier, we used Google Documents for a while, but ultimately we went back to the Excel file when we created the arbitrary groupings of terms. The Google Document acted more as a backup that we could all access. [↑](#footnote-ref-6)
6. That is what I’m calling the software. Catchy, no? [↑](#footnote-ref-7)
7. It also, frankly, seemed like a waste of time. [↑](#footnote-ref-8)
8. All of the relationships we defined were between terms, but there was no overall structure. In effect, we had pieces of a tree structure, but these pieces were not connected. [↑](#footnote-ref-9)