

SharePoint Team Site Configuration

Kevin Samms

Kennedy Space Center

Major: Computer Science

Data Mining and Knowledge Discover Summer Session

Date: 28 07 2014

SharePoint Data Mining Team Portal Configuration

Kevin O. Samms¹

University of Central Florida, Orlando, FL, 32816

The Data Mining project seeks to bring the capability of data visualization to NASA anomaly and problem reporting systems for the purpose of improving data trending, evaluations, and analyses. Currently NASA systems are tailored to meet the specific needs of its organizations. This tailoring has led to a variety of nomenclatures and levels of annotation for procedures, parts and anomalies making difficult the realization of the common causes for anomalies. Making significant observations and realizing the connection between these causes without a common way to view data is difficult to impossible. The Data Mining project is in its early stages of presenting a common visualization of normalized data through a web based portal. In this paper I detail what I have learned and what I have accomplished during my time as an intern toward the purpose of creating the project portal. I have gone through an iterative process of familiarization with the project and its goals, site design, research, modification and review on the road to creating a common method and location to view the visualized data created under this project.

Nomenclature

SAS	=	Statistical Analysis System
ETL	=	Extract, Transform, Load
SPD	=	SharePoint Designer
SP-SCAF	=	SharePoint with Standard Client Access Features
SP-E	=	SharePoint Enterprise edition
SME	=	Subject Matter Expert
XSL	=	Extensible Stylesheet Language
ASP	=	Microsoft's Active Server Pages technology
VB	=	Microsoft's Visual Basic high level programming language
CSS	=	Cascading Style Sheets

I. Introduction

The Data Mining project at NASA Kennedy Space Center (KSC) is in its early stages of creating visualizations of data combined from various NASA organizations and making that data available from a central location. The challenges in the visualization aspect of the project involve normalizing data from various systems where there are varying levels of detail in descriptions, varying levels of annotation, different nomenclature for the same parts and systems as well as different policies regarding how database fields are used. The presentation of visualized data within views appropriate to both the highly technical user and those seeking answers to common problems is another challenge. These challenges will be answered using three new tools: SAS (Statistical Analysis Software) and Tableau Desktop on the data developer side; Tableau Reader on the client side. There exists some ability for NASA to view data in a visual sense but the tool used is old technology not suited for the solving the modern Big Data problem or cleaning and connecting data from the various NASA organizations. The front-end portal aspect of the project involves the challenge of designing a common access point that is both easy to use and offers quick access to the visualized data. I came into the project having a familiarization with the visualization software, the data cleaning and transformation process, and the portal access technology known as SharePoint 2010. I have spent my time in this early stage of the project developing hands on understanding of SharePoint 2010 and the process of customizing this team site technology for efficient and simple access to the visualizations that are central to the success of the project.

¹ IT-G Intern, IT-G, KSC, and University of Central Florida.

I have gone through an iterative process of familiarization, design, research, modification and evaluation on the road to understanding SharePoint 2010. I began with a simple design for user access to the SharePoint site followed by a process of researching SharePoint capabilities, making modifications to the site, and then learning from that success or from further challenges. For the web page I used what SharePoint calls a Web Part Page and a few Data View web parts for presenting information to the user. A Data View Web Part is a list viewing tool which simply displays the information from a list or file library in typical row and column format. It is much like looking at the rows and columns of a database. A Web Part Page is just a web page which holds and allows the positioning of web parts. I eventually used other types of web parts as well as I discovered a need and found a web part to do the job. Of course the process of finding a new tool would not be as simple as taking one from the library of tools and just inserting it. My technical mentor and I had a vision of how the site should work. In contrast to that vision were the limitations of the version of SharePoint available for us to work with and the subsequent limitations in the available web part tools.

The challenge in realizing the early vision for the front-end portal, or simply the “site” as I will refer to it, was that I could not achieve the exact functionality envisioned using the tools and features that were available within our version of SharePoint. To meet this challenge I researched the available tools and searched for a way to achieve what we needed via alternative tools and methods. I would have to come up with a workaround solution that would allow us to realize the overall vision for the site but with the acceptance of the fact that the methods used may cause the user relatively more work and a less novel user experience. The workaround required that I “think outside the box” as my technical mentor put it. It was not all a compromise however, because over the first few weeks of this portion of the data mining project I had managed to get some SharePoint features activated and a web part modified at the coding level through communication with the SharePoint Team. Where the desired level of functionality was deemed highly necessary to the project and within the capabilities of SharePoint development we were able to bring in previously unavailable features.

Through the process of implementing the vision for the site I have been learning SharePoint 2010, practicing knowledge gained in COP 4710 Database Systems at the University of Central Florida (UCF) and gaining valuable experience in a professional working environment. Through both my technical and administrative mentors I have been gaining knowledge not only about the project goals but also about the details of how the data mining team functions. As the project moves forward I may have the opportunity to gain deeper experience with the data mining process through use of the tools which produce the visualizations that will be available via the site.

II. Evolution of Site Design

The site began with a conversation with my technical mentor in which we discussed text mining using SAS (Statistical Analysis Software) and how its statistical approach draws a connection between words typically found together in documents. From these “cluster words” topics are drawn by SAS. This became the basis of the Topics and Clusters list that I envisioned as a way for the user to find topics of interest and thus filter any list of files down to just those associated with a specific topic. That left the list of systems and their Tableau visualization files along with a list of any attachments that could possibly be associated with them. From these Ideas I designed the first layout of the site.

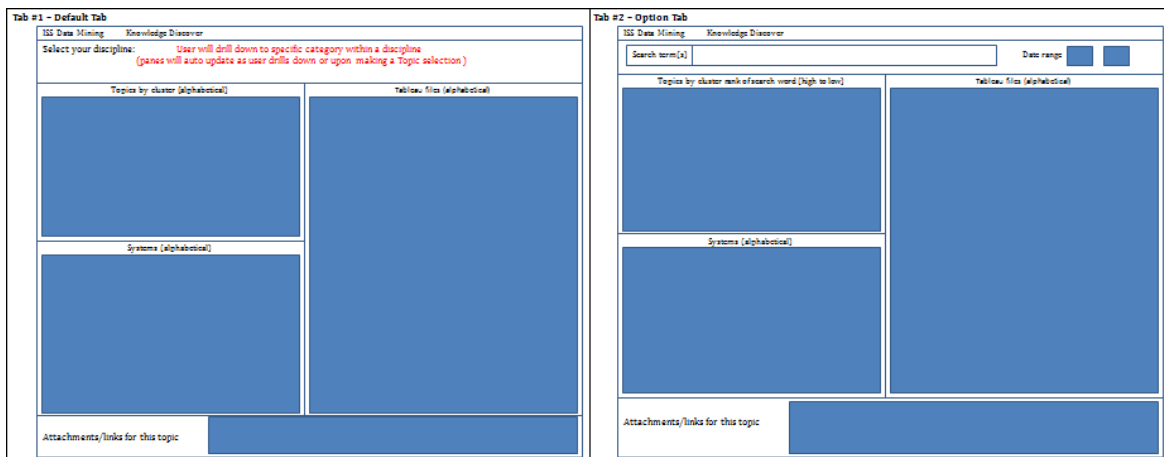


Figure1. Systems drill down tab (default view-left) and the topics search (option tab-right).

This early layout consisted of a default page with a search term area, a stacked view for the topics and systems, then the Tableau visualization files in a tall pane to the right followed by a pane on the page bottom to see attachments. This layout would allow the user to filter the views by topic and date range of the files. My technical mentor accepted this first design but required a modification in which the primary option page would be a page preconfigured to display content specific to the discipline of the user who is logged in to view the page. This seemed possible at the time since the functionality would only require making a profile modification requiring all users of the system to have their discipline listed in their profile settings. The list of choices would be preconfigured and this setting could then be used to key the Current User Filter which would then be used to focus the data views. However, this method could not be used. During the information gathering phase of any development, not only should one understand the requirements but the available tools should be verified as well.

After verifying our version of SharePoint and understanding its limitations, the use of the Current User Filter, drop down menus, and radio buttons were exchanged for links leading to content and an HTML web form part which could accept strings to be used in a query.

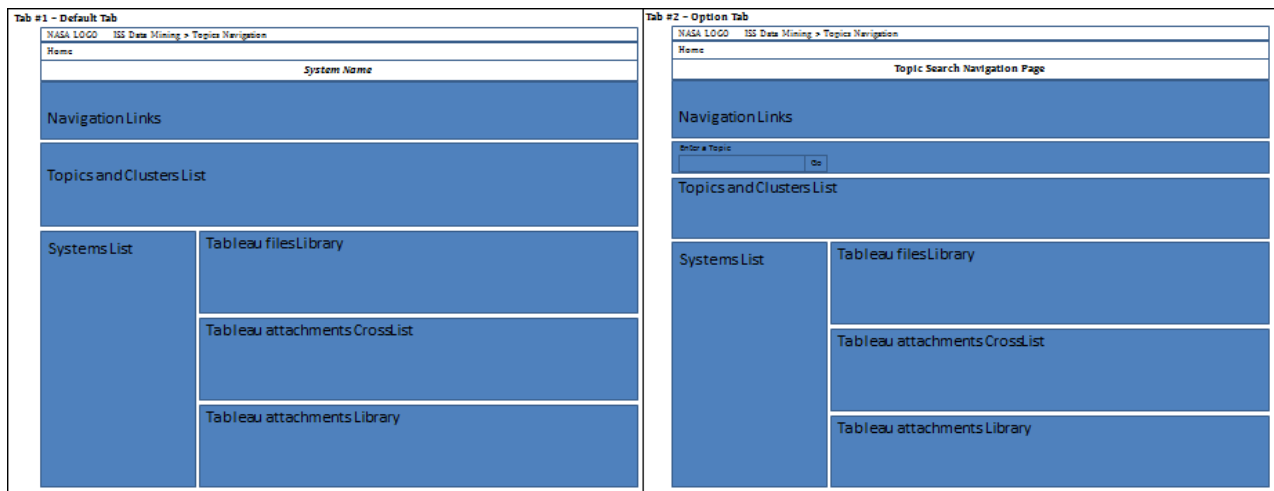


Figure 2. Connected web parts in system specific and topics search pages (left to right).

The overall idea of the available views did not change but the layout was modified to account for the fact that the data view web parts, once connected, act as filters to one another. The topics list filters the systems list which in turn filters the Tableau files library and the Tableau attachments cross reference list. The cross reference list then filters what files show in the attachments library. Each part acting as a filter gains a selection button column requiring that the user click one of those buttons in order to select a particular row as the active filter. This selection process is done following the series of web parts from the topmost filter down to the last in the chain of communication. This functionality in the parts creates a need for all of these views to be available to the user. Again the main difference between the two page options is that the primary page would be discipline specific whereas the option tab would allow exploration of the available visualization files and their attachments. A view of navigation links was added to allow for multiple pages with linking to each other and linking to a home page. At this stage, I had a better understanding of the capabilities of SharePoint 2010 and it was just a matter of further customizing aesthetic elements, reducing the duplication of files and finding ways to customize functionality to what we thought would make operating the site easier for our users.

The fourth and next to last evolution of the site saw the creation of separate subsites for the system specific users, now called basic users, and the exploratory view now designated as the view for advanced users. The basic users would get a Basic Users home page with system specific links (later changed to buttons) which open their system specific pages in new windows set to open at approximately half the width of the screen. It was determined that this half view helped to make the view of the data more orderly for the user. Opening the new page in a separate window allowed us to do this without interfering with the user's original browser window. Each of these smaller windows would also have their own close button at the bottom of the page. The advanced users get two search pages under the Advanced Users subsite. The first page allows search by topics only while the second page allows for search by topics or "cluster words" which are words typically associated with a particular topic. This comes from the SAS text miner ability to gather words typically found together in a document and group them under a single topic.

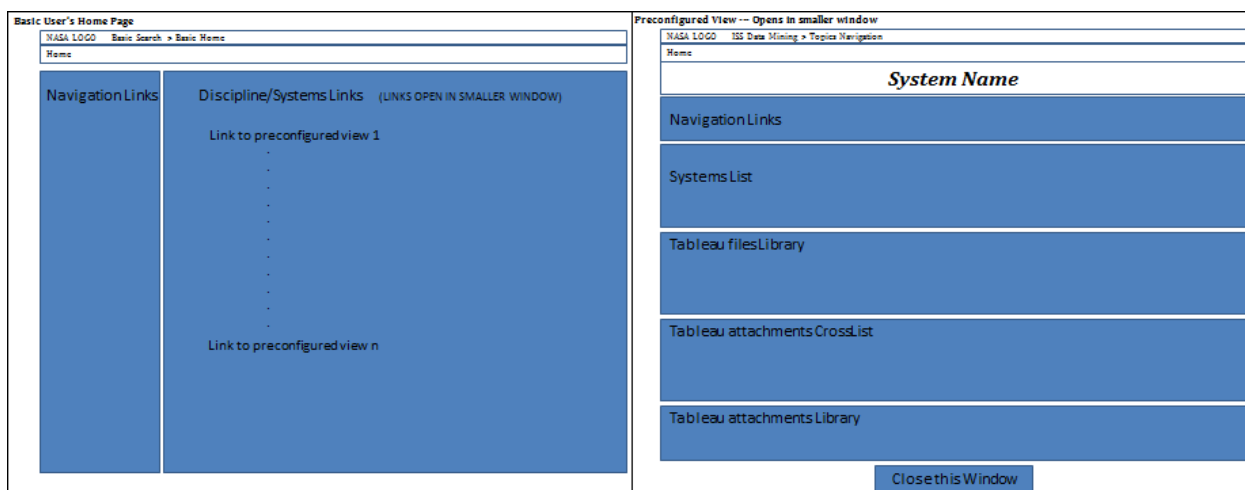


Figure 3A. Basic Users home page and system specific new window page (left to right).

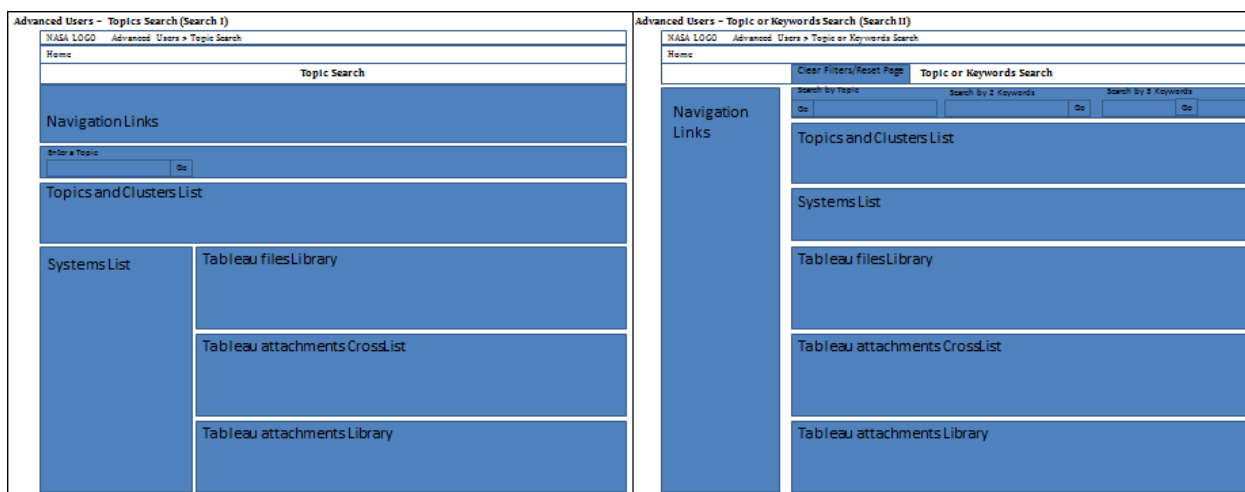


Figure 3B. Advanced Users pages – Topics search and Topics or Keywords search (left to right).

The final design of the site comes after this report and so will be documented as being much the same as the site in its third evolution above but with improvements such as radio style list filter selection buttons, linked files between libraries, a workflow for the approval of attachments, and the ability for users to request data visualizations of specific focus. The file linking will help with reducing space used on the server since all files will be centralized and other libraries that need a copy of a file will maintain only a link to that file. Therefore, these two subsites, basic and advanced, will not require the duplication of library data although the libraries in both sites use the same data. Both of these subsites will exist below the ISS Data Mining site; the main site in the hierarchy for the data mining project. Through communications with the SharePoint team I was able to have publishing services activated for the site. This will allow the use of the Content Query web part allowing both subsites to receive data from a central location. This will help to make subsite management a bit simpler. Training for users will come in the form of a written user's guide already in early construction and a video tutorial that will possibly make use of the video production support services available to the team.

III. The Implementation

In preparation for this internship my mentors requested that I familiarize myself with SharePoint 2010, Tableau Desktop and Reader, the Extract Transform and Load (ETL) process, and SAS (Statistical Analysis Software). As the project was in its early stages my experience in the use of these tools would progress from creating a portal for data access to taking part in developing the data itself. The time available and the progress of the project would determine what I could accomplish in this two month internship. Once on-property and with my workstation

prepared we began with a discussion about the concept for the site. This led to the first layout as described in the introduction; a page preconfigured for users by their discipline and a page for searching the available files in general. As the site moved through each evolution I documented the current state of the site, my actions, and my current direction on a weekly basis. These reports are a part of this paper and served as checkpoints to keep the team informed of my progress. My reports were typically of the format: current state, work completed and a quick bullet point summary version of everything for quick reading.

After the correct version of SharePoint 2010 was identified and its limitations relative to the desired functionality were known, I made an assessment of what could and could not be done. This assessment was done in context of the available tools so the team could better picture where the site design was going.

A. The Assessment and Work Completed in the First Two Weeks

- 1) If we were using SharePoint Server Enterprise we could do the following:
 - a. Utilize a Filter web part which could be added to a List View web part to filter its list content by the currently logged on user.
 - i. A Current User Filter would utilize a user property to filter the System list and thus display only the systems which apply to the specialty of that user. The remaining windows would key off of the system window and display only topics and Tableau files related to that system. No clicks required; the user logs in and immediately sees only content specific to their discipline.
 - ii. A Choice filter would do the same except the user has to click a box to bring up a list of choices. The choices would be the systems for which they could see the relevant data. Clicking on a choice then takes the user to the system specific view of the site and therefore all data relevant to that system. Two clicks required.
 - b. Modify site user properties to include a mandatory user profile property (i.e. “Discipline”) which a Current User Filter could use as a key for filtering the System list and thus display only the systems which apply to the specialty of that user. The remaining panes would key off of the system pane and display only topics and Tableau files related to that system.
 - c. Have web parts filter each other, as in a filtered list in one web part sends a row of data to another web part which then uses that data to filter its own list. In this way web parts can be made to communicate with each other/filter each other and thus create a page of information filtered by a selection in one of its parts. The entry point; the part that begins the communication would be the Current User Filter.
- 2) In the second approach I could utilize the features of SharePoint Designer which is what we will be installing on June 13, 2014. However, I am unclear as to whether Installing SharePoint Designer will enhance the capabilities of our SharePoint 2010 Standard to the point of being able to perform actions similar to the filtering by user scenario above. I will be researching what features the install of SharePoint Designer will bring to the project.
- 3) In the third approach to filtering content by a specific user I could research a workaround in which links are provided for the user to click on. These links would display the name of specific systems and take the user to pages preconfigured to display data relevant to those systems. This could possibly be a one click solution.

The third approach above, the workaround, is the one that I am currently researching. This may require a restructuring of the intended layout shown in the PowerPoint file associated with this report.

B. Work Completed and Summary

The work completed so far includes:

- 1) Two web part pages stored in a web parts page library (the primary and secondary pages).
- 2) A Tableau files library which currently holds text files with .twbx extensions to simulate Tableau files.
- 3) The Topics and Clusters custom list (columns: Topic (primary key); 2_word_cluster; 3_word_cluster)
- 4) The Systems custom list (columns: Topic (foreign key); System (primary key); Tableau File)
- 5) A Links list of one link to be embedded into the Search Navigation (secondary option) page. It allows a user on the Search Navigation page to go directly to the Current User Navigation (primary option) page. Name: Link to Current User Navigation page.
- 6) A Links list of one link to be embedded into the Current User Navigation (primary option) page. It allows a user on the Current User Navigation page to go directly to the Search Navigation (primary option) page. Name: Link to Search Navigation page.

- 7) Downloaded SharePoint Designer and Tableau Reader in preparation for the installations on Friday June 13, 2014.
- 8) Personal work tracking log to track hours and work done each day.
- 9) A PowerPoint showing the general layout of the primary and secondary web page options at the sandboxed site.
- 10) Microsoft Access databases file to demonstrate the linking of the “Topics and Clusters table” and the “Systems table” also on the sandboxed site.

To summarize where we are with the project, we have a sandboxed version of the production site. I will be constructing my solution on that site. I have the parts of the solution ready, as listed in the work completed section above, but I do not yet have the ability to get the parts to communicate the way that I intended. Our version of SharePoint is missing the following features which I required:

- 1) User profile modification.
- 2) The Filter web parts.
- 3) List View web parts ability to filter other List View web parts that are not using the same list.

As a result of not having the features above I am looking for a solution that will use web links to custom views. I intend to create a Links list to store the links to pages in which the filters for each page are preconfigured to show data relevant to particular systems. My next task is to create these preconfigured pages and then create a page to hold the links to them.

With the knowledge of what could and could not be done I devised a workaround using the tools available in our version of SharePoint 2010. The overall idea of how the site would work remained intact; a page specific to personnel belonging to a particular discipline and a page that allowed exploration of all systems and their data. The look and feel that we envisioned however would have to adapt to the available functionality.

Thus the site moved to its second conceptual evolution and first working implementation. The learning process continued as I researched and found ways to achieve the goals of the conceptual design despite not having the exact functionality desired. My report at the end of the third week of my internship informed the team of this first implementation and workaround solution. Testing of the site with common web browsers was also performed at this stage.

A. Third Week Report – Site Implementation

The site for the data mining project has moved from conceptual design to implementation. The site is composed of the ISS Data Mining home page with a navigation link to the primary design option, search by discipline/systems, and a link to the secondary design option, search by topic. All are web part pages and all are functional. The search by discipline/systems and the topics search pages have some inefficiency that I will attempt to correct in order to simplify the user experience. The topic search page has links to both the ISS Data Mining home page and the discipline/systems home page. The discipline/systems home page in turn has links to the ISS Data Mining home page, the topics search page, and links to each discipline/systems preconfigured page. A preconfigured page is one in which the System web part on the page is already set to filter its list values by a particular system thus focusing the user on working with content specific to that system. The two design options differ only in that the discipline/systems pages do not have an HTML form web part for entering a search string. A search string is not needed on a preconfigured page in which the query has basically already been done for a particular system. All other parts are common to both pages and are operated the same.

We included SharePoint Designer (SPD) into the project and it brought in the ability to easily add, configure, copy and delete pages. Although some page options and view modifications still had to be done from within our SharePoint 2010 with Standard Client Access Features (SP-SCAF). The software seemed a bit buggy and inconsistent in that some modifications to a page did not persist and had to be done in SP-SCAF. The variety of web parts was not increased. The communication between web parts was definitely enhanced in that web parts on different lists cannot communicate in SP-SCAF but through SPD communication across different lists and libraries it is now possible and the configuration of such communication is made much simpler through the SPD user interface (UI).

SPD does not elevate SP-SCAF capabilities beyond what it can do out of the box. As a result we still have the problem of not being able to automatically configure content based on the current user or give the user a menu to select their discipline. Those capabilities come with SharePoint Enterprise edition (SP-E) which we do not have. That edition has the Current User and Choice Filter as well as the ability to customize the user profile to include attributes that could serve to control current user filtering.

B. The Workaround

The workaround for the lack of functionality to utilize a choice menu or current user profile setting was to include the choices for discipline/system as links to pages which are preconfigured for the selected discipline/system. The links replace the functionality of a choice menu and the preconfigured pages replace the ability to utilize the current user and their profile settings to focus their site content. Whether or not we had the SP-E capabilities does not seem to affect the fact that the operation of the communication between these web part pages involves some inefficiency. Communication between web parts is a producer/consumer model. One part produces a row of data to be sent to a consumer web part which consumes and uses the row to filter a column of its own data. In this way, the column of one web part filters the data displayed in a column of another web part. The first web part in the chain of communication is initiated either by user action (enter a search string into the HTML web part and hit enter or click the “Go” button) or through preconfiguration. There are two problems with web part communication. The first is that the producer web part comes with a selector that must be clicked in order to update the next web part in the chain otherwise the next web part retains its old data. This necessitates that the user click on all selectors in the communication chain in order to ensure the data views are correct. The second problem is that if a web part has no data to display (it has no data for some category of the web part before it) its data display will collapse leaving no selector to be clicked. The result is that any subsequent web parts after the collapsed one will not only be non-updatable but will retain old data, thus making their portion of the data on the page possibly incorrect for the current page view. A possible solution for this involves researching the program code of the web parts in SPD and possibly coming up with a programmatic solution, i.e., custom code would have to be written to customize web part behavior.

C. Web Browser Testing

Web browsers tested on the site included Microsoft Internet Explorer 9 (IE 9) 64-bit, Firefox version 24.5.0 and Google Chrome version 35.0.1916.153m. Internet Explorer required two clicks to a Tableau file (click on file then click on “open” at prompt) and one click to open a Word document which will then only open in the browser. Firefox required one click to open a Tableau file and two clicks to open a Word document; one on the document and then click on “read only” or “edit” at prompt. Google Chrome required one click for each type and the document file opens in the browser as read only. Chrome requires the user to click on the file, select save as to begin the download, then in the browser click the downloaded file’s dropdown menu and choose “Always open files of this type”. Subsequent clicks on Tableau file will then open the file in its native program and thereby be of the minimum number of clicks. The procedure for Firefox is similar but Chrome is the better choice as it minimizes the number of clicks to a file.

D. Work Completed

- 1) ISS Data Mining home page changed to a web parts page with links to the discipline/systems search structure and the topics search page.
- 2) Created the Discipline/Systems home page and three example preconfigured subpages: Propulsion, Avionics and Mechanical.
- 3) Created Topics Search Navigation Page utilizing an HTML form web part. The user enters a search string into the form and presses enter or clicks the Go button.
- 4) Created library files for Tableau files and attachments. The library files can contain files of any type and the names can include letters, numbers and spaces. I believe file naming limitations are the same as for files in any standard Windows folder.
- 5) Created lists for topic search (Topics and Clusters List), Systems web part population (Systems List), a cross list relating Tableau files to their attachments (Tableau attachments cross list) and the SME list which will be the actual source for the Systems web part once we begin working with real data.
- 6) Unnecessary lists, libraries and pages created during testing and implementation have been deleted.
- 7) SharePoint Designer and Tableau Reader installed on Friday June 13, 2014.
- 8) Updating personal work tracking log to track hours and work done each day.
- 9) The PowerPoint showing the general layout of the primary and secondary web page options has been updated to reflect the necessary changes to the general layout of the site. The original is on file and the updated version is showing in the “Test site concept files” library at the ISS Data Mining home page.
- 10) A Microsoft Access database file to demonstrate the linking of the Topics and Clusters table, the Systems table and the Attachments table is also in the “Test site concept files” library. The linking and primary key, foreign key relationships could be better demonstrated within a MySQL instance.
- 11) Various methods and procedures have been documented for:
 - a. Web part page creation and modification.

- b. Browser customization to work with the files from the website.
- c. Team site modification (quick launch).
- d. SP-SCAF List creation using Microsoft Excel.
- e. Connecting SPD to the SP-SCAF site.
- f. Web part configuration and communication using SPD and SP-SCAF.

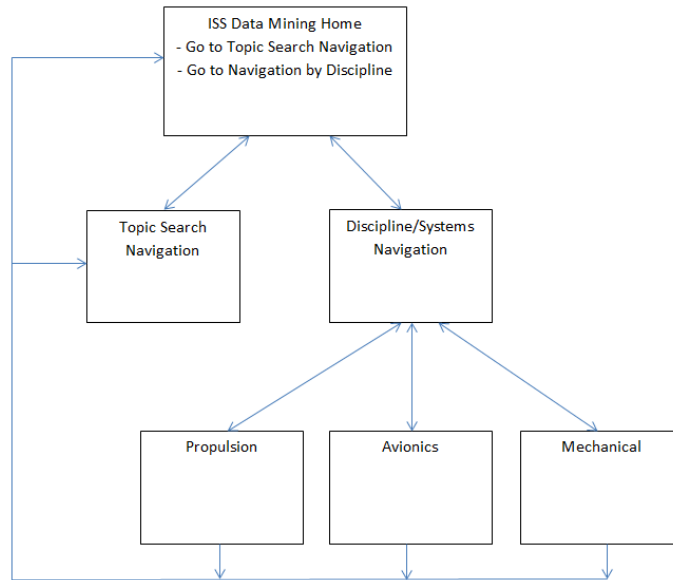


Figure 4. Site structure at the end of the third week.

The implementation required the application of database experience from the COP 4710 Database Systems class at the University of Central Florida. Considering the table information to be related and displayed to the user, multivalued attributes had to be removed and placed in separate tables. Thus a relational database schema in third normal form (3NF) was used to make the intended presentation of information to the user. Issues with the second evolution of the site were also noted. Two major issues were the inadequacy of the selection buttons inherent in the data view web parts for connected web parts and the issue of downline web parts displaying incorrectly when an upline web part had no filtering data to send downline. These issues caused me to begin researching Extensible Stylesheet Language (XSL) in order to understand what was going on in the code and whether or not I could find a solution to the problem.

So the fourth week of my internship was one of reading, continuing to update and modify the site, and support my mentor as meetings with other groups within the project continued. From the group meeting in this week I obtained valuable input regarding the site that led to its third evolution. My report at the end of this fourth week of my internship detailed my efforts, endeavors, and again served as a site status update to the team.

A. Fourth Week Report - The Site and Current Issues

The current site represents a workaround solution in which both the idea of filtering the user view of site content by their login and the idea of using choice menus are unavailable. Instead, the users will be presented with links from which they can choose the one that matches their discipline in order to be directed to discipline specific content. Users will also have the option of searching by topic or cluster words.

After the team meeting on Wednesday, June 25, 2014 the workaround has been further modified. There are now two subsites under the ISS Data Mining site: Basic Search and Advanced Users. The primary option of using links to direct a user to discipline specific content has become the Basic Search subsite. The secondary option of allowing the user to perform a search by topic or keywords has become the Advanced Users subsite. Currently the advanced subsite has two user directed search pages; a search limited by topic word (search I) and a search by topic or cluster words (search II). Both advanced search options require improvements. Search I requires “contains” word matching so it is not limited to an exact match search string only. Search II requires the same as well as an ability to detect ‘n’ search terms and choose the appropriate ‘n’ sized cluster word column.

Besides the needed improvements to the advanced searches, the current goals are: (1) create a workflow to approve document attachments to Tableau files; (2) fix the problem of a web part incorrectly displaying old

information when an upline web part has no information to display; (3) make downline web parts automatically select their first row of data. The last two problems require a programmatic solution. Problem #2 requires iterating through a web part list, getting the count and then communicating that information to a conditional test structure of a downline web part. The theory is understood but getting that done requires learning more about the code.

My current direction is to solve problems #1 and #2, then move on to the other issues.

B. Work Completed

- 1) Exhausted the search for a SharePoint Designer solution to the web part resetting problem.
- 2) Created a search by Topic or Keywords Search page (search II) to complement the Topic Search page (search I). This is to test a different display format and include text boxes to search each column of cluster words (called Keywords for the user).
- 3) Created a button to clear the search II text boxes. This fixes the problem of the topics list being stuck on empty display when switching between the different query string boxes to perform the available searches.
- 4) Found various books and websites to help in understanding web part coding and SharePoint coding in general. This is to help in understanding programmatic solutions to web part problems.

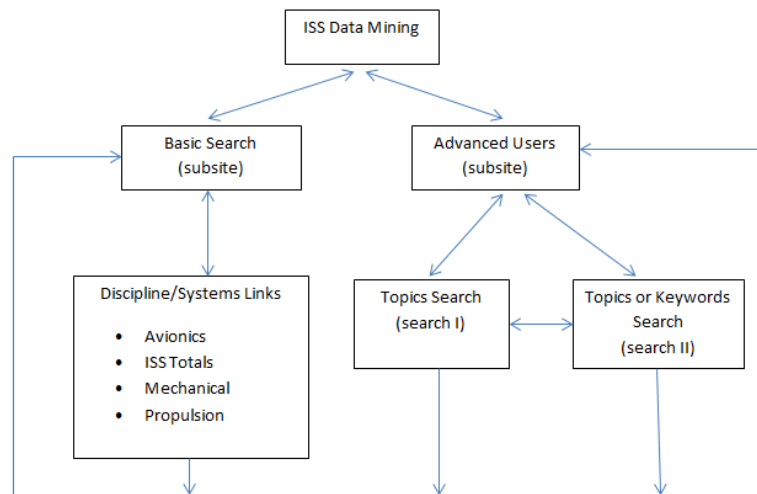


Figure 5. Site structure at the end of the fourth week.

At this point the site was in its third evolution and I was busy trying to understand XSL code as used in the SharePoint, researching related programming languages and assessing whether or not I had the experience to solve the functionality problems discovered in the fourth week. My technical mentor had moved the issue of setting up a workflow to my list of lower priorities in favor of solving the issue with the look and feel of the user interface. The selection buttons had to be made to look more like actual buttons and the web part communication issue should be solved. Similar to the fourth week, this fifth week was more about reading and research. It was a short week due to the holiday and some company events. My report at the end of this week detailed the status of the site and the results of my efforts in making custom changes at the coding level.

A. Fifth Week Report

Week 5 was the July 4th holiday week and was spent reading from various sources to learn more about how web parts work in SharePoint. I learned that C# and VB.net are in the .NET family of programming languages and are used to build SharePoint itself as well as used in building ASP.net pages. I also learned more about the SharePoint object model including the SPSite, SPWeb and SPList classes for manipulating and getting information from SharePoint objects. I discovered an important namespace, ddwrt, which is used to perform tasks such as accessing properties of a SharePoint list or firing events to connected web parts. Unfortunately this namespace is not documented by Microsoft and only a few references to it were found on the Internet. All references seemed to be between two people, Marc D Anderson who has an article on the site nothingbutsharepoint.com and Serge van den Oever's article on MSDN.

This was a reading week with some experimentation with the code in a SharePoint test page using SharePoint Designer. I was able to get some JavaScript running within the ISS Data Mining - Search II web part page. It

accesses a site list and gets the number of items in the list. I also used the Alert() function to create a pop-up message on the screen displaying the list count. Unfortunately this is not what I was looking for as I actually need to access a particular web part's reading of a list, not the list itself. In general I learned a lot more about how the code works (templates, functions, variables, etc.) and can now decipher the code a bit more.

B. Work Completed

- 1) Reading/researching SharePoint XSL code. Learned more about the classes SPSTite, SPWeb and SPList as well as “for loops” such as “foreach (Object_Class variable in variable.Lists)”.
- 2) Created my first JavaScript within the web part page “Data Mining – Search II” to access a site list, get its number of items and display that information via the Alert() function.
- 3) Discovered the ddwrt namespace. These functions perform tasks such as accessing properties of a SharePoint list or firing events to connected Web Parts. Unfortunately information regarding this namespace is difficult to find as it is not officially documented by Microsoft.
- 4) Focusing on learning the XSL language and ddwrt usage to capture web part query list results and work with that data to control other web parts.

By the sixth week my knowledge and experience with SharePoint and SharePoint Designer was well beyond the early familiarization at the start of this internship. I had gained experience with building and customizing pages using SharePoint itself, SharePoint Designer and had experience going into the XSL code to make custom changes. Some problems however seemed still beyond the reach of my current understanding and would likely require another request for expert support from the SharePoint team.

A review of my progress from my technical mentor led to some new visions for the look and feel of the site. My sixth report detailed my continued work and accomplishments on the way to satisfying these new goals from my technical mentor and continuing to develop the site.

A. Sixth Week Report

Week 6 was spent attempting to further penetrate the intricacies of SharePoint web part code. A number of classes such as altclass and hoverclass were found to be undocumented or have very little documentation to explain them.

My technical mentor and I discussed my progress early in the week and I was presented with a new set of tasks to be added to the ongoing task of learning how to control the Data View Web Part behavior. My goal for the ongoing task is to get each web part to select its first row of data on a page load and get the downline web parts to close when an upline part has no data to send to it. The new tasks and their current statuses are:

- 1) Highlight and bold first row of the Tableau files list.
 - a. Completed. I used SharePoint Designer to add “Conditional Formatting” to the list.
- 2) Remove Tableau files column from Tableau Files and Attachments Crosslist.
 - a. Completed. The Tableau Files column was removed from all basic site pages.
- 3) Change icon for Tableau files.
 - a. Completed. I copied a Tableau file picture from the Internet and resized it according to the directions at [http://msdn.microsoft.com/en-us/library/office/aa978104\(v=office.14\).aspx](http://msdn.microsoft.com/en-us/library/office/aa978104(v=office.14).aspx). I contacted the SharePoint team and passed the above directions to my contact. The final association of file to file type icon required a deployment effort by the SharePoint team.
- 4) Include a Tableau picture column.
 - a. Completed. The picture column was added using SharePoint's standard options.
- 5) Set the browser window on the basic site to scale the user's view to half size in order to make the page more readable, considering the width adjustments to the list columns in SharePoint Designer.
 - a. Completed, but with an issue going forward. Most browsers prevent the resizing of a page from the code so I instead modified the Basic Search site to open each SME page in a new window and manually set the sizing to about ½ screen widths.
 - i. Issue: Screen size was set using my current monitor. The new window should detect the screen size and set itself to ½ screen size for any monitor. However, this requires that the screen size be detected before the page loads and I have not yet figured out how to do that in the code.
- 6) Fix topic search to a “contains” search such that the user does not have to exactly type in a search term in order to filter the advanced search lists by a particular search term.

- a. In progress. I tried various methods with no success as documented in an email to the technical mentor on 7/14. The only other option that I could find is to use the Text Filter Web Part which only comes with SharePoint Enterprise edition. I contacted the SharePoint team asking if obtaining that web part is possible and whether there is an associated monetary cost. A service ticket was created for me regarding this issue.

The remainder of this week was spent continuing my research into web part code and behavior. I also documented various procedures used to accomplish the above completions.

At the seventh week of my internship the site consisted of two areas: Basic Search and Advanced Users. The basic site has a home page with links to SME specific pages which open in individual windows of size approximately half the size of a full page view. The first row of the Tableau files list is automatically highlighted. The Advanced Users site works as documented in previous reports; it has not undergone any new changes but I was awaiting information from the SharePoint team regarding a possible implementation of advanced filtering for this site. I was at this point researching methods for making changes to the site theme and background image in addition to other activities.

A. Seventh Week Report

This week was primarily about continuing to learn the XSL code used in SharePoint. I have abandoned the approach of using the Internet to learn about snippets of code in favor of learning the XSL language from a book; The XSL Companion by Neil Bradley. I am focusing on what I think are the relevant parts of the language. I hope this approach will bring me to an understanding of XSL sooner than, or at least complement, my former approach.

Regarding advanced search functionality for the topics search page I have contacted the SharePoint team and a request for information is currently under a service ticket. This week I also installed 3 new software programs: Tableau Reader 8.2, Notepad++ and Microsoft Visual Studio Web Developer. The Tableau install is a necessary upgrade from version 8.1 and is required in order to read the Tableau workbooks currently being output for the project. The other two programs are useful in the creation of SharePoint web parts; an activity I hope to begin learning once I have solved the current problems with the standard web parts used in the site. A topics modification to the Basic Search site was also completed. The former topics list has been replaced with the topics: IFI, PRACA, GFE PRACA and MOD AR.

B. Work Completed

- 1) Put a close button on the SME window pages.
- 2) Sent email to the SharePoint; an information request concerning procurement of a Text filter web part.
- 3) My technical mentor and I discussed project direction and topic changes to the Basic Search site. We also discussed my final report for this internship.
- 4) Modified Basic Search site to new specifications and kept Advanced Users site as is.
- 5) Reading XSL book to better understand SharePoint code.
- 6) Updated site concept files (PowerPoint).
- 7) Noted void link issue on Friday. Clicking on a link at the Basic Search site resulted in a javascript:void() instead of the URL stored in the link. Opening one of the links and clicking save caused the links to work again. I will contact the SharePoint team if problem persists.
- 8) Currently working on:
 - a. End of week and final reports.
 - b. Modification to site Theme and web part selection button picture.

In the eighth week of my internship I had moved away from the idea of modifying the web parts programmatically as this would have required that I begin learning web part programming. There was not enough time for that so I contacted the SharePoint team about changing the selection buttons for connected web parts to something that looked more like what a user would recognize as a button. Something that would draw a user's attention to the fact that a row of information had to be selected in order to filter the other connected lists.

We conducted a short demonstration in which my administrative mentor served as a tester for the site and gave us valuable feedback concerning the look, feel, and ease of operating the site. I had also contacted the SharePoint team regarding methods for connecting data view web parts to a single data source. This was to eliminate the duplication of work when maintaining the separate subsites. Regarding the buttons, it turned out that one of the SharePoint team developers had received a similar request over two years ago and so had the code ready for what I wanted. It had been implemented previously but then removed from the system due possibly to an update. My

request would get this code re-implemented once the SharePoint developers had the time to re-test and re-deploy the code. Regarding the duplication of work, the SharePoint team enabled publishing features for our site which would allow the centralization of files that I desired.

At the end of the eighth week the demonstration that we conducted showed us that we still had a lot of work left to do. However, we were confident that it could be done in time for another demonstration at the end of the ninth week. The site was up and running and I had experience rebuilding it a few times to incorporate new ideas. A user's guide had to be finished, the new publishing functionality had to be incorporated, this final report had to be completed, and a change to the button style would come at some point in the week when the SharePoint development team had time to perform testing. My technical mentor worked on the Tableau visualization part of the user's guide while I worked on its site navigation aspect. For all aspects of the project, the term "Tableau files" was being exchanged for the term "visualization files" in order to recognize the function more than the tool of the files that we present to the user.

A site governance issue had arisen with the changes to the look of the site. After I had figured out how to modify the site banner, breadcrumb and background and do so efficiently by linking custom CSS scripts to the web page code, it turns out the SharePoint team would only allow the background image. The other changes had to be reverted in order to maintain the uniformity of the site. The user's guide constructed in the earlier part of the week had to be modified to reflect this. Thanks to my browser testing in week three and subsequent updates, the part of the user's guide meant to show the user how to set up their browser would be an easy finish requiring only the addition of screenshots.

This final report will be turned in before the end of the ninth week and so will miss the documentation of the success of our next demonstration. As this report comes to a close the project will still have two weeks remaining and I predict a working site, a user's guide and visualization files complete with significant observations available for access to our test users. From that point we will include workflows for approving attachments to visualization files and a site feature to allow users to request additional visualizations targeted to their specific needs. Later in the project we may utilize our video production support service to complete a video tutorial for site users. At some point going forward I am confident that the team will also incorporate the advanced text filtering that will make the topics search pages more flexible for the advanced users. As my mentors have valued the work that I have done thus far I may have a future opportunity to work on the data mining and visualization development side of the project.

IV. Conclusion

SharePoint 2010 with Standard Client Access Features was not an insurmountable barrier to the kind of site functionality that we envisioned. It made the implementation more difficult in that workarounds had to be devised and the concept had to be adapted to the available functionality. For our purposes, SharePoint 2010 features previously unused were activated and the acquisition of a highly useful web part was being researched by the SharePoint team. We also saw program code tried years prior, now being reintroduced for our site. The communication within our team and with support services has made the site and access to our data visualization files not only possible but efficient and simplified for our users.

References

Books

¹Wright, S., Petersen, D., *Pro SharePoint Designer 2010*, 1st ed., Apress, New York, 2011.

²Buyens, J., *Microsoft Windows SharePoint Services Inside Out*, Microsoft Press, Washington, 2005.

³Bradley, N., *The XSL Companion*, 1st ed., Pearson Education, Massachusetts, 2000.