NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

AUTOMATED POLICY UPDATING USING A SEMANTIC WIKI

by

Dawn M. Swisher

December 2012

Thesis Advisor:	Karen L. Burke
Second Readers:	Zachary Peterson
	Christine Haska

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** December 2012	**3. REPORT TYPE AND DATES COVERED** Master's Thesis

4. TITLE AND SUBTITLE AUTOMATED POLICY UPDATING USING A SEMANTIC WIKI	**5. FUNDING NUMBERS**
6. AUTHOR(S) Swisher, Dawn M.	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING /MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	**10. SPONSORING/MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol number ______N/A______.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited	**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

The process of updating and maintaining policy updates at the local implementation level is currently a time-consuming manual process with vast room for improvement. A timely and accurate means of conveying policy updates from the higher levels in the government to the lower levels is needed.

This thesis describes a taxonomy as a first step to remedying the problem of maintaining current policies when upper-level policies are updated. This is especially important in regards to Information Assurance policies because of the fast-changing nature of the technology involved.

A taxonomy was defined and then implemented utilizing wiki technology for demonstration purposes. This research is further discussed in terms of its portability across the government as a means to automatically convey policy changes from upper levels to lower levels in order to maintain responsive forces backed up by current policies.

14. SUBJECT TERMS Policy, Semantic Wiki, Hierarchy, Taxonomy	**15. NUMBER OF PAGES** 49
	16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** UU

THIS PAGE INTENTIONALLY LEFT BLANK

AUTOMATED POLICY UPDATING USING A SEMANTIC WIKI

Dawn M. Swisher
Civilian, Department of the Navy
B.S., University of Colorado, Colorado Springs, May 2008

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN COMPUTER SCIENCE

from the

NAVAL POSTGRADUATE SCHOOL
December 2012

Author: Dawn M. Swisher

Approved by: Karen L. Burke
Thesis Advisor

Zachary Peterson
Second Reader

Christine Haska
Second Reader

Peter Denning
Chair, Department of Computer Science

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The process of updating and maintaining policy updates at the local implementation level is currently a time-consuming manual process with vast room for improvement. A timely and accurate means of conveying policy updates from the higher levels in the government to the lower levels is needed.

This thesis describes a taxonomy as a first step to remedying the problem of maintaining current policies when upper-level policies are updated. This is especially important in regards to Information Assurance policies because of the fast-changing nature of the technology involved.

A taxonomy was defined and then implemented utilizing wiki technology for demonstration purposes. This research is further discussed in terms of its portability across the government as a means to automatically convey policy changes from upper levels to lower levels in order to maintain responsive forces backed up by current policies.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I would like to express my very great appreciation to my advisor, Professor Karen Burke, for her support. Without her expert guidance I would not have been able to develop and complete this thesis. I would also like to thank Darlene Fromhart of Naval Air Warfare Center Aircraft Division for helping me understand the complexities of the problem addressed in my thesis, and Professor Zachary Peterson for his insight and support as a second reader. I would especially like to thank Dr. Christine Haska for her outstanding encouragement and advice throughout this entire process: She was continually involved and provided an applied perspective on this research. Finally, I would like to thank my husband, Dustin, for his unending patience and support throughout this entire thesis process.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. SCOPE

Although Information Assurance (IA) is important to the mission and IA policy provides the foundation for maintaining a secure posture, the process of updating, disseminating, and maintaining current IA policy remains a problem in the Department of Defense (DoD). As upper-level policies are changed and updated, the lower-level policies that reference them must also be updated to comply with the changes. Often those who are responsible for maintaining those lower-level policies are unaware of the upper-level policy changes, so inconsistency ensues. In addition, those who use the policies must research multiple DoD websites and often struggle with determining the correct version of the most current policies, particularly when inconsistencies are discovered. With all of the available sources of this information, policy updates must be de-conflicted. This research will be bounded at the upper level by the Department of the Navy (DON) and at the lower level by the local policies at the Naval Postgraduate School (NPS). Once the taxonomy is utilized for this subset of DoD policies, this thesis will discuss the portability of this model to include the DoD as the new upper-bound.

B. CURRENT POLICY DISSEMINATION PRACTICES

The links between lower-level policies and the upper-level policies they reference must correspond to the correct versions of the upper-level policies. As part of this thesis, there will be a review of the current manual processes for disseminating policy changes and how users of policies manually determine the most current versions of upper-level policies. This thesis will develop the methodology to automate the dissemination of policy changes. The process also will consider the requirement to "archive" policy versions that may be referenced in DON contracts.

C. RESEARCH GOALS

1. Taxonomy

This research carries multiple goals, which will be addressed in sections. The first goal is to develop a taxonomy of IA policies in the DON. The upper-bound of this taxonomy will be the IA policies found on the Department of the Navy Chief Information Officer website and related websites that house DON-level policies with the assumption that all updates and changes to policies above this level are correctly implemented. The lower-bound for this taxonomy will be the "local" level as represented by the IA policies at the NPS level. The taxonomy will describe the links from policies in the upper-bound to policies in the lower-bound, as well as links between policies residing at the same level. Mapping this relationship between policies is the first step to establishing a means of automatically conveying policy changes in a timely and efficient manner.

2. Automatic Conveyance of Policy Changes

The next goal of this research is to find a way to automatically convey when there have been changes to upper-level policies if there are lower-level policies that are potentially affected by the change. If possible, the changes themselves should be included in the information that is passed along to convey the change, and the administrator in charge of reviewing the policies should know specifically which lower-level policies are affected by the change. This makes the job of determining which policies need to be reviewed simple and efficient.

3. Policy Versioning

The third goal of this research is to maintain versions of policies that can be referenced if necessary. The space that is used to house the most current and up-to-date policies should also house the older versions of those policies in order to make them easier to find. This must be done in such a way as to clearly show which versions are current and which are out of date without taking away accessibility to older versions as they are needed. The outdated versions should always be read-only in order to preserve them in case it becomes necessary to roll back to them or if a contract is based on older policies that have since been replaced or updated. As it stands now, these versions can be difficult to find and their dates of use may be incomplete or unclear.

This research will attempt to disambiguate policy versions and dates of use, as well as make it easy to keep policies up-to-date without needing extensive research. Policies should have default review deadlines regardless of whether changes have occurred in referenced upper-level policies to ensure administrator users remain aware of current versions and make updates as necessary for operating purposes. When changes have been made, the changes should be distinguished by whether they are major content changes or minor grammatical changes, and there should be due dates for those changes to be pushed down. For example, small grammatical changes that do not affect content or meaning should be approved immediately without going through a chain of approval. On the other hand, content changes should be reviewed and approved on a minimum set schedule with bigger or more important changes implemented on a timely basis as the administrator sees fit. This could mean that content changes must be rolled out to lower-level policies at a minimum of once a year, or more often as needed. This would ensure that lower-level policies would not become years out of date, losing their relevance and effectiveness.

4. Proof of Concept (Implementation)

The fourth goal of this research is to build a proof of concept that demonstrates that the taxonomy is an effective structure for linking policies that reference one another so that the changes to upper-level policies are conveyed accurately to the administrator of lower-level policies that refer to them. A way of doing this is to set the administrator in charge of lower-level policies as a “watcher” of the upper-level policies they reference, and when changes occur, the administrator is notified. If related policies are grouped by code to indicate the links between them, this would ensure that the administrator knows which policies to review without having to trace back the relationships between policies from the references section of each policy. The proof of concept as described here entails a lot of front-end work to get the policies into a collaborative wiki format, but once it is in place and organized according to the taxonomy, it will require little maintenance to keep policies up to date and accessible. The ideal results will be achieved if there is a DON-wide compatible wiki format to house IA policies. Failing this, an administrator could simply keep a list of the upper-level policies and stay on RSS [Rich Site Summary] feeds or email lists to receive notification of changes to policies in the DON-level. Once changes have been made to policies at this level, the administrator will upload the changes to the wiki and the group notifications for affected policies will appear. This will also continue to provide policy versioning since the relevant policies and their versions will be stored and accessible on the wiki, which in the case of this research is housed on the local education network.

D. POLICY MAINTENANCE

This research will accomplish several things that are currently lacking in many government agencies at the local-level. Too often, the maintenance of current policies takes a back seat to all of the other important tasks and functions within an agency. Once policy maintenance begins to slip, it can get out of hand quickly and local policies may become years out-of-date before they are noticed. This neglect cannot be allowed to happen in an environment where policy is the foundation of day-to-day cybersecurity operations. This style of updating spreads the responsibility and knowledge of policy updates to more people. In many agencies, there is a “policy person” who is responsible for keeping local policy information up-to-date and who knows where to look for all pertinent information. This is a job that takes practice and extensive background knowledge, and is usually the responsibility of a particular person. This violates the practice of separation of duties that is described as, “A security principle that splits up a critical task among two or more individuals to ensure that one person cannot complete a risky task by himself [1].” While policy maintenance is not generally considered a risky task in and of itself, it is important to remember that the neglect of policy maintenance may become a liability when current policies are difficult to find in a timely manner, or if policies go out-of-date and the person responsible for this job is unavailable to address the problem.

With the infrastructure from this thesis in place, the process of updating policies will be accessible to anyone with administrator privileges to the wiki and policies will be easily maintained in the event that the policy manager is made unavailable to perform the duties of the job. The end goal of this research is to make IA policies maintainable no matter the staffing circumstances, and this infrastructure provides the means to revise the currently complicated role of policy maintenance.

THIS PAGE INTENTIONALLY LEFT BLANK

II. BACKGROUND AND RESEARCH

A. DIRECTION OF INFORMATION FLOW

This thesis presents a proposed answer to the question of how to create and implement a taxonomy of security policies in a semantic wiki in order to enable automated conveyance of updates from higher-level policies to lower-level policies. This can also be applied between policies at the same level with the exception of policies at the top level since information does not flow between them. The information flow in this model is from top to bottom, or between lower-level policies at the same level, and follows step-wise flow of information similar to that found in existing information processing flowcharts [2]. The concept of using hierarchies for decision making and showing the flow of information in a system is not a new one. There has been a vast amount of research applying hierarchical models to everything from marketing decisions to diagnosis [3] - [7]. The only new information this research brings is how to effectively apply this knowledge to the problem of information flow between policies at different levels within the DON. This introduces two challenges: 1) how to create the right taxonomy to portray the relationships between policies, and 2) how to implement that taxonomy within the semantic wiki so the information flows in the right direction.

B. WIKI COLLABORATION WITHIN THE GOVERNMENT

The wiki format has been successfully used within the government for collaboration on classified data. Intellipedia was created in 2005 and unveiled in 2006 [8], [9]. It has three levels that correspond to top secret, secret, and unclassified data that can be shared with users authenticated to each level. While the primary use of Intellipedia is collaboration on data of different classifications to allow for quick decision-making and dissemination of time-sensitive information, wiki technology also can be used to keep policies up-to-date. The functionality that a wiki provides is useful for conveying updates of upper-level policies to the lower-level policies that reference them. The fact that the government recognizes the necessity to migrate to more online collaborative environments is a good indicator that it is time to think about how this impacts the world of IA policy [10]. This so-called e-government is recognized as an important tool for collaboration and user participation [11], [12]. If wiki technology is utilized early in the IA policy process, this will ensure that cybersecurity technologies do not outpace the IA policies that govern their use.

Increasingly, support has been shown for the implementation of wiki technology for collaboration and dissemination of information within the government [13]. The potential for collaborative work is undeniable, and in light of this, the movement of government to wiki technology seems inevitable. The advantages of wiki technology for collaboration make it an ideal setting for maintaining IA policies in the fast-paced world of cybersecurity. In a world where constantly evolving technology requires frequent updates to IA policy, the government can no longer afford to maintain such policies in a paper or Portable Document Format (PDF) due to the exorbitant amount of time and effort that is required to make changes to it. IA policies at the level of the DON may be updated frequently with the advancement of technology. The lower-level policies that are derived from the DON-level policies for use in an implementation setting must also be updated on a regular basis. It is at the implementation level that policies are not managed appropriately due to difficulties with the propagation of updates to policies from a high-level to a low-level. If IA policies are not current at the implementation level, the systems that must abide by the policies are left weak and vulnerable to attack. When policy updates are ignored and local-level policies are neglected, the efficiency of government agencies is negatively affected in a way that could potentially compromise national security. At the very least, having IA policies that are out-of-date at the local level may have legal consequences for the government agency that is responsible for maintaining them.

C. WIKI TOOL CHOICE

Many wiki tools are available that would work for the maintenance of IA policy, but for the purposes of this research a particular wiki was chosen for a proof of concept. The Atlassian Confluence Wiki provides all of the necessary functionality for a proof of concept in addition to being the wiki of choice at the local-level for the Naval Postgraduate School. Since semantic wikis are similar in terms of the functionality that is required to perform the proof of concept for this research, the particular wiki that is used is not as important as the fact that the policies are stored and maintained on a wiki site. When the wiki is configured to link with a tool to track issues and create workflow such as Atlassian JIRA, it can perform the necessary functions to not only house policies, but to help keep them up-to-date as well. As a side-note, JIRA is not an acronym. This is just an in-house code name that was invented by the Atlassian JIRA creators [14]. Using an issue-tracking tool such as JIRA in combination with the wiki allows both current and old versions of policies to be housed on the wiki while JIRA functions to track updates and changes to policies. This combination provides the structure that is required to "push" the conveyance of changes to upper-level policies down to administrators of the lower-level policies that reference them. Any wiki and issue-tracker can be used as long as they integrate in an effective way to provide the following functionality:

1. Requirements of an Integrated Wiki/Issue-Tracker Tool

The important functions that must be available by a wiki and issue-tracker tool in order to implement this taxonomy in a useful way are as follows:

- Users must be able to subscribe to updates with a given frequency
- The updates that a user is made aware of consist of
    - Changes to policies including those that are added, edited, or deleted
    - Comments on a policy that are added, edited, or deleted
- The notifications can be customized to show the full content of pages changed as well as the differences between versions
- Minor changes (i.e., grammar or spelling changes) can be labeled as such so they do not cause notifications to be sent out.
- Users must be assigned to different privilege levels to allow administrators the ability to edit content and other users the ability to only make and edit their own comments. This is an important point because there are few people who are allowed to make changes to the content of IA policy.

- Notifications can be customized to alert the administrator to changes within particular groups of policies so that the policies that must be reviewed are clearly identified.

D. POLICY HIERARCHY AND TAXONOMY

Before the policies can be linked in the wiki, they must be ordered hierarchically in a taxonomy. Providing this framework at the outset before policies are placed on the wiki is the most efficient way to ensure that policies are linked correctly and that updates will be propagated accurately. The idea is for the wiki to manage the work to maintain IA policies and to enable the administrator to determine at a glance which lower-level policies are affected by changes to upper-level policies.

E. WHERE POLICIES ARE STORED ONLINE

A detail that cannot be determined by this thesis is where to house the upper-bound policies since this decision lies with the DON. Typically, IA policies at the DoD-level are hosted on websites such as the National Institute of Standards and Technology website and Defense Technical Information Center website, while DON-level policies are hosted on websites such as the Department of the Navy Chief Information Officer website and the Department of the Navy Issuances website. It is beyond the scope of this thesis to dictate where IA policies are stored, but this will be one of the topics discussed in the Future Works section. To solve the problem of referencing IA policies in the upper-bound within the wiki structure, the text of the most current policy will be copied and pasted into the wiki. Administrators will have to subscribe to RSS feeds and email lists to learn when DON-level policies are updated, which is the current practice for staying up-to-date on DON-level policies. The DON Issuances website already provides a useful taxonomy and version system to indicate which policies are in use and which are out-of-date. As updates occur, the administrator can “version” old policies so they will continue to be accessible on the wiki, but there will be no mistake as to which version is the most current. As changes are made, the administrator will be notified of all of the policies that must be reviewed based on the linking structure of the policies within JIRA.

Given the implementation of the taxonomy in the wiki, the updates will be conveyed accurately to ensure that policies do not go out-of-date as a result of neglect.

THIS PAGE INTENTIONALLY LEFT BLANK

III. TAXONOMY FRAMEWORK

A. CHOICES FOR TAXONOMY

The taxonomy that is used to determine how policies updates are pushed down to the local level from the DON level is distinct from the NPS Policy Hierarchy as described in Figure 1. Although the figure is straightforward in terms of describing the relationship between policies that exist in a simple tree structure, it does not address the difficulties of representing the flow of information between policies when the structure is more complicated, as is often the case within the government. The basic structure of linking policies from the upper level to the lower level is based on the one represented in Figure 1, but the difference comes in the case that policies reference one another at the same level, or where multiple upper-level policies are referenced by a single lower-level policy. In order for the taxonomy for this research to be expanded from the hierarchy in Figure 1, some choices must be made. When policies are arranged in a hierarchy, they can either be grouped by the policies from which they are derived (Figure 2), or by the policies that are derived (Figure 3). The models that were considered to represent how policy hierarchy should be implemented at the local-level are described as follows:

1. Group by Upper-Bound vs. Lower-Bound Policies

When policies are grouped by the DON-level policies from which they were derived, the importance is placed on the upper-level policies. This model seems more intuitive, but the strengths and weaknesses of both models had to be examined separately to make a decision on which should be used in the context of this research. The concept of a tree-style hierarchy and the resulting groups was derived from an example semantic partition tree of the *New York Times* front page [15]. While the application of a semantic partition in that case was used to discover the implicit schema for the hierarchical relationships between concepts, its usefulness can be expanded to the domain of IA policy hierarchy. In the top-down model, the highest-level policy is examined for changes, and the changes flow downstream to the lower-level policies. This is the same in both models, but the difference lies in the grouping of the policies. When policies are grouped by upper-bound, a lower-level policy may reside in the groups of multiple upper-level policies. On the other hand, when policies are grouped by lower-bound, each lower-level policy resides in its own group, and multiple upper-bound policies may reside in the same group as the lower-bound policy without having their own group. This second configuration of grouping by the lower-level policies places the emphasis on the lower-level policies that must be updated at the implementation level. The difference between the two models is illustrated by Figure 2 and Figure 3. The grouping of policies given in Figure 2 shows the simplicity of grouping by the upper-bound. In Figure 3, the grouping of policies by lower-bound gets complicated very quickly. This is due to the fact that policies go from more general at the upper-bound to more specific at the lower-bound. This causes fewer upper-bound policies to branch into a much greater number of lower-bound policies.

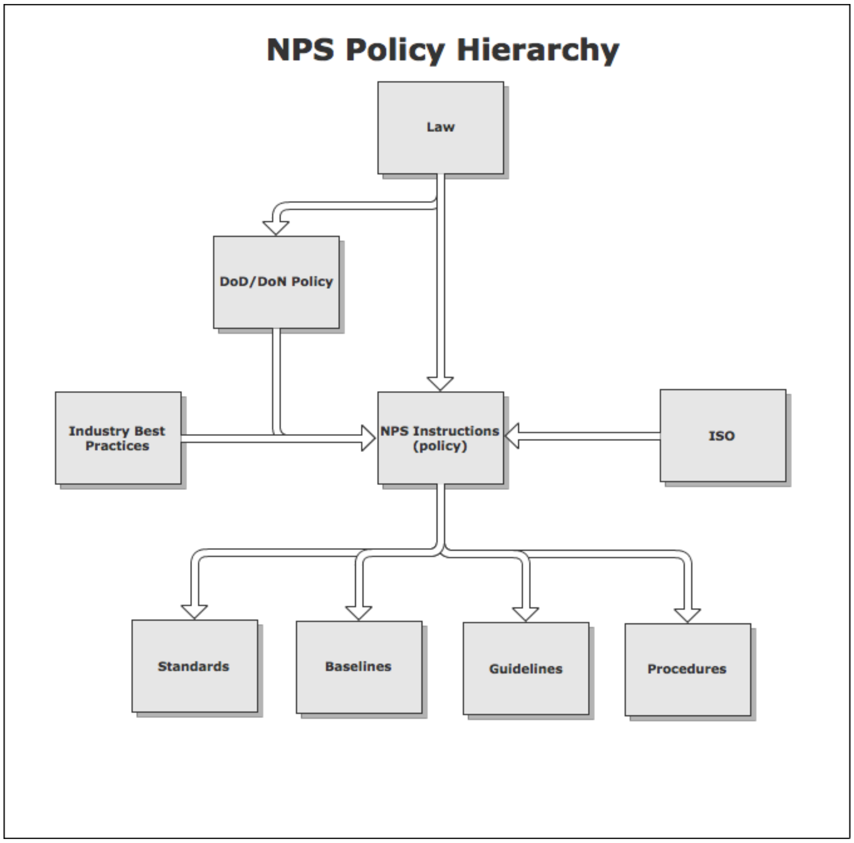


Figure 1. NPS Policy Hierarchy. From [16].

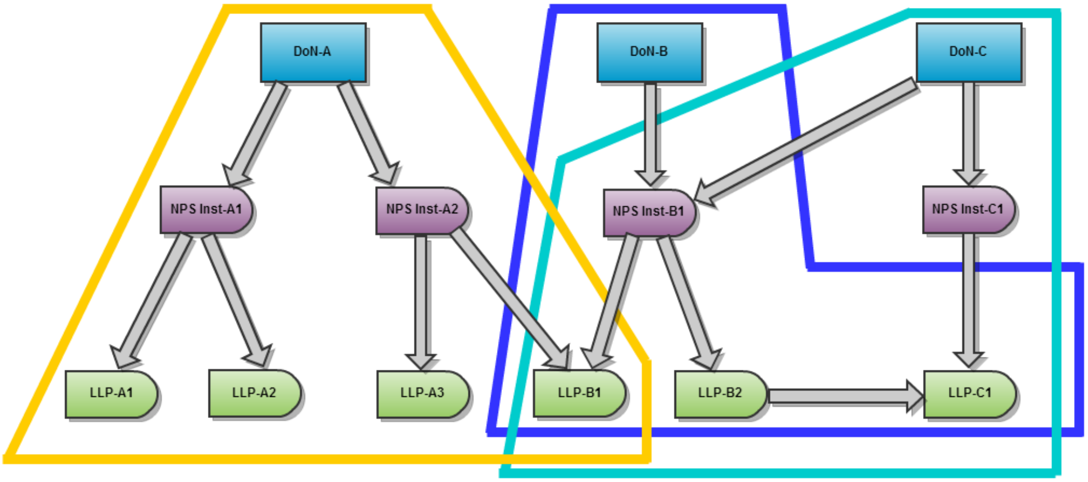


Figure 2. Group by Upper-bound

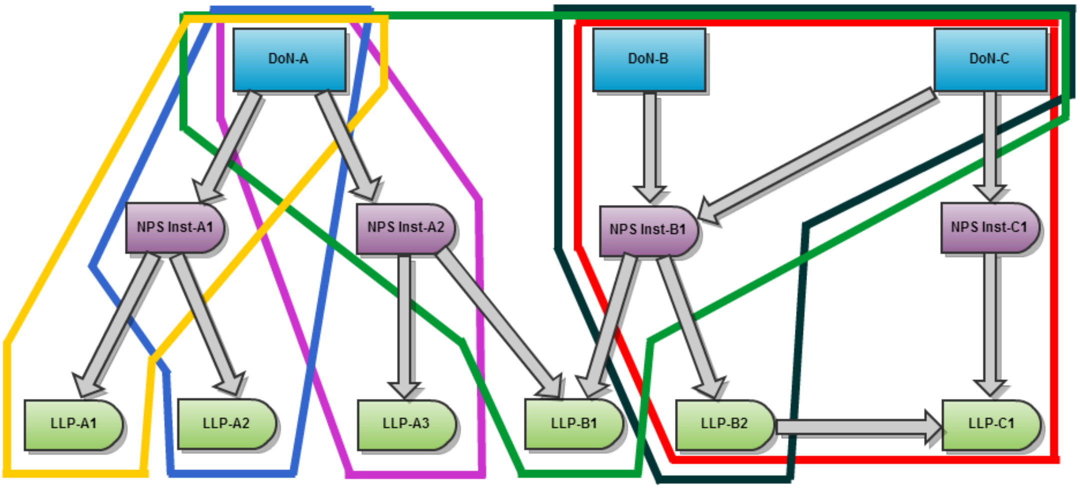


Figure 3. Group by Lower-bound

In the end, it was simpler and more effective to group by upper-bound policies because this was the easiest way to convey the correct flow of information from top-to-bottom. The lower-bound grouping requires the administrator to look at a lower-level policy when it is time for an update and determine which changes have been made to upper-level policies. This grouping of policies may cause multiple upper-level policies to be examined even when no changes were made to a particular upper-level policy. This allows all of the policy changes for the local policy to be done at once, but it is time-consuming in that it requires the administrator to examine all policies to determine where changes have occurred. In the top-down grouping, the administrator examines all policies downstream of the upper-level policy that was changed, and makes changes accordingly. In this model, a lower-level policy that references multiple upper-level policies that have changed will be examined multiple times. In spite of this, it is still more efficient to examine a lower-level policy multiple times in the event of updates to multiple upper-level policies than it is to examine upper-level policies even when they have not changed.

B. TAXONOMY OUTLINE

1. NPS Policy Hierarchy

The hierarchy begins with the policies at the DON level and works down to the local NPS level. The directionality for this hierarchy causes many of the groups to overlap to a certain extent. This overlap can be defined in such a way that the upper-level policy changes do not cause notifications to be created for the other upper-level policies. The notifications will only be created for the policies at levels that the administrator can modify. The hierarchy described in this way makes the clearest structure for the purposes of linking the policies in the wiki. The hierarchy that is implemented in the wiki is illustrated in Figure 4.

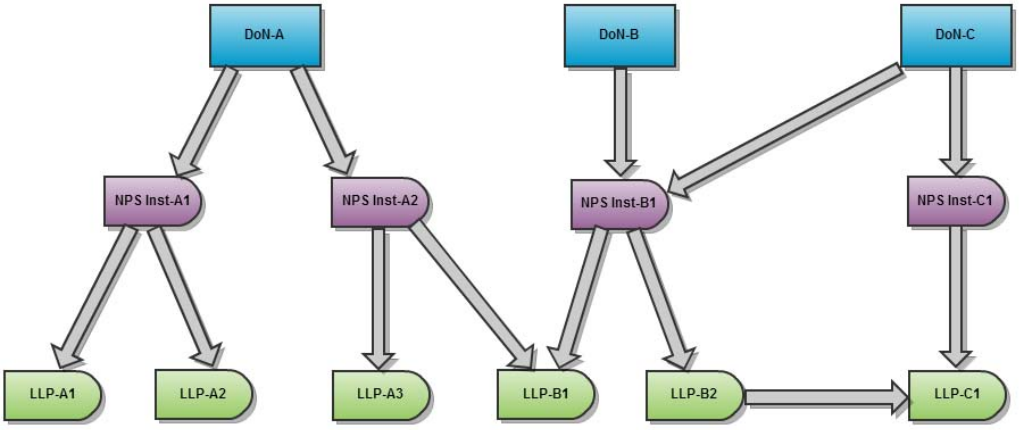


Figure 4. Policy Taxonomy

2. Flow of Information

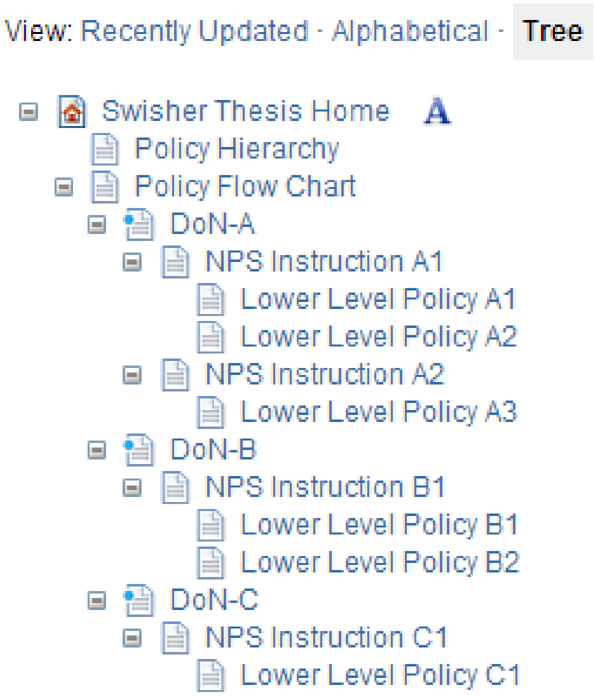
The hierarchy of IA policies in the DoD begins with the law at the top level, then the Executive Branch, the DoD level, the Joint Command, Service, and Agency level, then NPS Instructions and policy, and from there down to the level of Standards, Baselines, Guidelines, and Procedures at the local NPS level (Figure 1). The section of this hierarchy that will be examined is the DON for the upper-bound and the NPS local policies for the lower-bound. Multiple ways of grouping the policies were considered in this research with the focus placed on how to get the most efficient linking between policies. The efficiency of the linking is determined by how well the updates from upper-level policies are propagated to the lower-level policies. There are complex relationships between policies because they are often intertwined in ways that make it difficult to represent them in a graphical format. A given upper-level policy may link to many lower-level policies; a single lower-level policy may link to multiple upper-level policies; lower-level policies may link to each other. With all of this linking, the policies form a sort of web as opposed to a simple graph or flow diagram. Some of the updates that occur to upper-level policies may require updates to the lower-level policies that reference them, but some do not. Likewise, some of the updates to lower-level policies will require updates to other lower-level policies that reference them, but this is not necessarily the case. This taxonomy is necessary to describe the relationships between all IA policies from the upper-level to the lower-level, and between policies on the same level. The upper-level policies will not change as a result of changes to other policies in order to satisfy the requirement that updates flow only from upper-level to lower-level, and between policies at the same level where the relationship exists. The flow of updates between policies on the same level is allowed where necessary. This relationship usually occurs at the local-level.

3. Description of Taxonomy Identifiers

Since updates are accurately propagated within the wiki when this implementation of the taxonomy is used, the policies can be grouped as illustrated in Figure 2 with top-down information flow. The groups are defined by the upper-level policies in this example as follows: policies on the level of the upper-bound can be identified as DON-A, DON-B, and DON-C. The next level of policies that reference the policies on the upper-bound are identified as NAVPGSCOLINST. These are NPS instructions and for the sake of brevity will be referred to as "NPS Inst" and further distinguished by a letter and number as follows: NPS Inst-A1, NPS Inst-A2, NPS Inst-B1, and NPS Inst-C1. This is done to show that the NPS instruction came from a particular DON policy. In the small example given for demonstration purposes in this research, the letter that is assigned to the NPS instruction is assigned based on the DON policy that it references most or first, and the number differentiates between multiple NPS instructions that reference the same DON policy. In actual implementation of this model, the numbers and letters that are assigned may correspond with the DON level series number to unambiguously indicate the relationships between policies. From the DON Issuances website, the current taxonomy for DON-level policies utilizes the Standard Subject Identification Code (SSIC) and is described as, "the standard system of numbers and letter symbols used throughout the Department of the Navy for categorizing departmental records by subject. SSICs serve as the taxonomy for all departmental records," [17]. The next level down is the "local" NPS level, and this is the lower-bound of this research. This level represents the policies, guidelines, standards, and procedures that reside at the local implementation level. In the given example, these lower-level policies are labeled as LLP followed by a letter to signify the DON instruction they reference and a number to differentiate between them, much the same way the policies do at the level directly above this level. For example, Figure 4 lists the local policies as follows: LLP-A1, LLP-A2, LLP-A3, LLP-B1, LLP-B2, and LLP-C1.

4. Policy References

The flow of information between policies is represented by the directional arrows represented in Figure 4. The arrows show that the information from the DON-level policies at the upper-bound flow to the NPS instructions at the next level, and from there to the NPS local policies at the lower-bound. The language used to describe a relationship where an upper-level policy has an arrow pointing to a lower-level policy is that the upper-level policy is *referenced by* the lower-level policy and the lower-level policy *references* the upper-level policy. This distinction in language is an important one to make as this is the relationship that allows JIRA to define the flow of information between policies. Policies at the same level such as LLP-B2 and LLP-C1 can also be defined by this relationship as follows: LLP-B2 is *referenced by* LLP-C1 and LLP-C1 *references* LLP-B2. This shows that the information flows from LLP-B2 to LLP-C1. Another view of the hierarchy is shown by the tree diagram in Figure 5.



View: Recently Updated · Alphabetical · Tree

- Swisher Thesis Home
  - Policy Hierarchy
  - Policy Flow Chart
    - DoN-A
      - NPS Instruction A1
        - Lower Level Policy A1
        - Lower Level Policy A2
      - NPS Instruction A2
        - Lower Level Policy A3
    - DoN-B
      - NPS Instruction B1
        - Lower Level Policy B1
        - Lower Level Policy B2
    - DoN-C
      - NPS Instruction C1
        - Lower Level Policy C1

Figure 5. Policy Hierarchy Tree in the Wiki

5. Redundancies in Information Flow

In this model, it is necessary for lower-level policies to be able to reference one another. In this instance, this model for linking policies may cause some redundancies as the lower-level policies may be updated multiple times. This is a result of the automatic propagation of notifications as policies are updated. All of the notifications will go out as described by this path only if changes are actually made to the policies that are *referenced* by the other policies. Without a change to set off a notification, the subsequent policies in the path will not be affected. The notifications stop when the administrator stops making changes. Since this taxonomy is only necessary for the administrator to be alerted to changes, there is still a human-level check to determine whether an update to the affected policy is required. This check prevents infinite loops in the case of two policies referencing each other and renders circular referencing harmless.

IV. IMPLEMENTATION

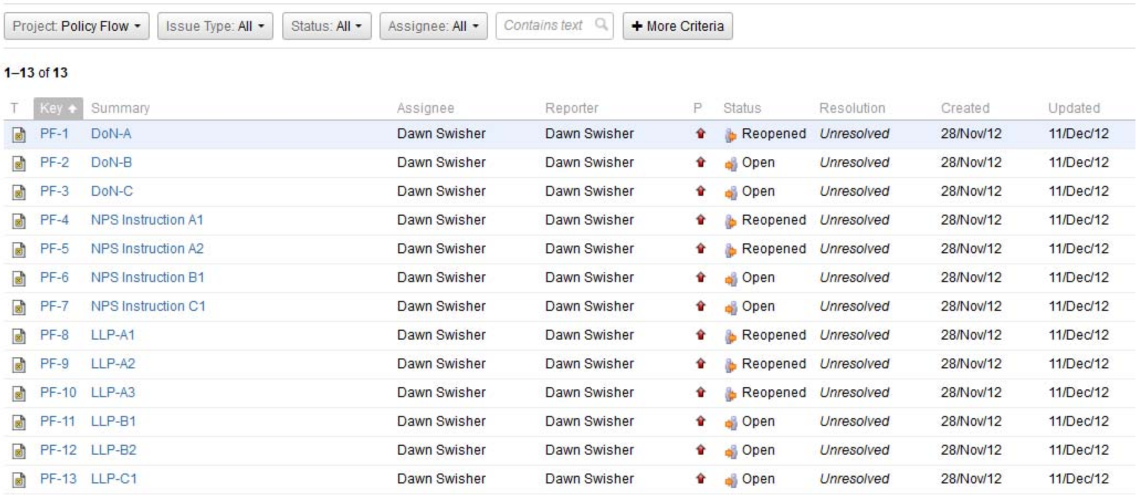
A. OVERVIEW

This research takes the previously described taxonomy and implements it in the context of a wiki combined with an issue tracker. There is no particular tool that is intended to be promoted by this research. The tool that was chosen serves the purpose of this research. Any wiki technology and issue tracker that work in combination with each other are appropriate if they have the required functionality as outlined in section II of this thesis. For the purposes of this thesis, the chosen tools are the Atlassian Confluence Wiki in conjunction with the Atlassian Confluence JIRA. The Atlassian Confluence Wiki (hereafter, referred to as wiki) is a semantic wiki collaboration tool that can be used by small groups or large agencies for sharing information [18]. The Atlassian Confluence JIRA (hereafter referred to as JIRA) works as a collaborative project tracking tool that is good for teams [14]. This choice was made based on availability and capabilities of the tools, but anything with the right functionality will do.

Using the wiki to house policies in this way enables the policy changes from the upper-level to the lower-level to be conveyed automatically. In order to implement this structure in the wiki it is necessary to perform a certain amount of front-end work to upload current policies into the wiki and create linked issues for all of the policies using JIRA. Once the policies have been uploaded and linked according to this taxonomy, it becomes a simple matter to maintain the policies so they do not go out of date. The amount of front-end work that is required to begin this process mitigates the amount of long-term work that it takes to manually search and update policies as they go out of date. The extensive amount of work and research that is currently necessary for administrators to perform these duties is greatly lessened by the policy management described by this research.

B. UPPER-BOUND

The upper-bound policies that are detailed by the small example given in Figure 4 are DON-A, DON-B, and DON-C. These policies are housed in the wiki and NPS instructions are added as child pages to the appropriate DON-level policies in the wiki. These policies have JIRA issues created for them as well. This continues until all of the policies have been added to the wiki and JIRA issues have been created for each policy (Figure 6). The policies should be added as child pages such that the higher-level policies are parent pages of the lower-level policies. This is achieved by adding lower-level policies to the upper-level policies as child pages, creating a tree hierarchy (Figure 5).



Project: Policy Flow ▾ Issue Type: All ▾ Status: All ▾ Assignee: All ▾ Contains text + More Criteria

1–13 of 13

T	Key	Summary	Assignee	Reporter	P	Status	Resolution	Created	Updated
	PF-1	DoN-A	Dawn Swisher	Dawn Swisher		Reopened	Unresolved	28/Nov/12	11/Dec/12
	PF-2	DoN-B	Dawn Swisher	Dawn Swisher		Open	Unresolved	28/Nov/12	11/Dec/12
	PF-3	DoN-C	Dawn Swisher	Dawn Swisher		Open	Unresolved	28/Nov/12	11/Dec/12
	PF-4	NPS Instruction A1	Dawn Swisher	Dawn Swisher		Reopened	Unresolved	28/Nov/12	11/Dec/12
	PF-5	NPS Instruction A2	Dawn Swisher	Dawn Swisher		Reopened	Unresolved	28/Nov/12	11/Dec/12
	PF-6	NPS Instruction B1	Dawn Swisher	Dawn Swisher		Open	Unresolved	28/Nov/12	11/Dec/12
	PF-7	NPS Instruction C1	Dawn Swisher	Dawn Swisher		Open	Unresolved	28/Nov/12	11/Dec/12
	PF-8	LLP-A1	Dawn Swisher	Dawn Swisher		Reopened	Unresolved	28/Nov/12	11/Dec/12
	PF-9	LLP-A2	Dawn Swisher	Dawn Swisher		Reopened	Unresolved	28/Nov/12	11/Dec/12
	PF-10	LLP-A3	Dawn Swisher	Dawn Swisher		Reopened	Unresolved	28/Nov/12	11/Dec/12
	PF-11	LLP-B1	Dawn Swisher	Dawn Swisher		Open	Unresolved	28/Nov/12	11/Dec/12
	PF-12	LLP-B2	Dawn Swisher	Dawn Swisher		Open	Unresolved	28/Nov/12	11/Dec/12
	PF-13	LLP-C1	Dawn Swisher	Dawn Swisher		Open	Unresolved	28/Nov/12	11/Dec/12

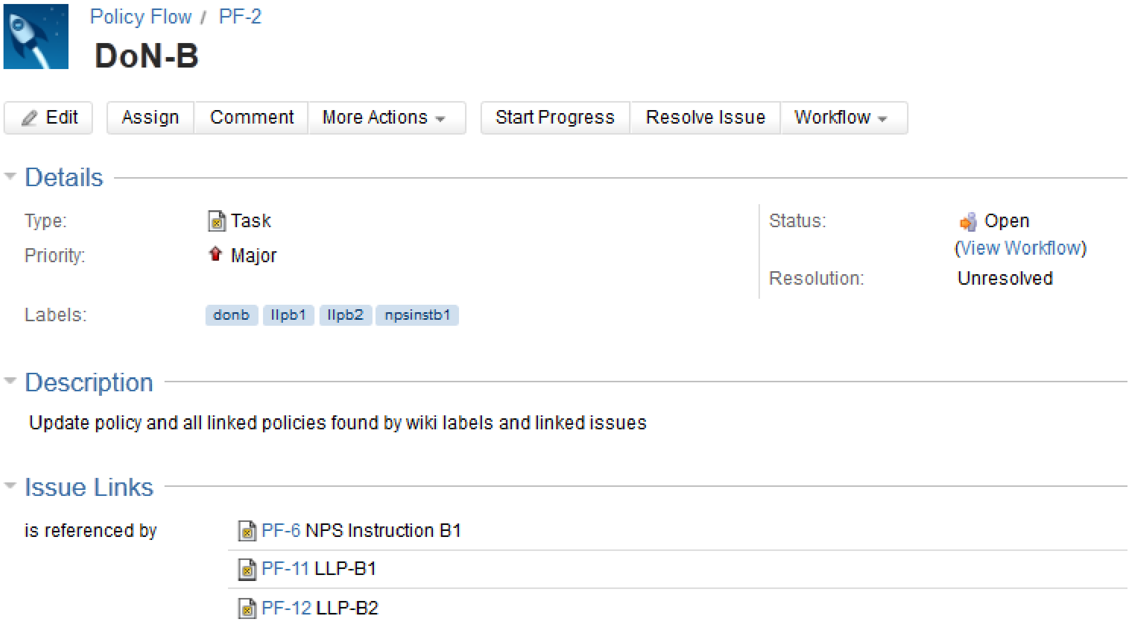
Figure 6. Atlassian JIRA Issues Page

C. LOWER-BOUND

In the instance that there is a child policy that references multiple upper-level policies, the lower-level policy will be added as a child page to the upper-level policy that it is named after. The naming scheme as defined in Section III describes this naming convention and makes the decision of where to store policies on the wiki unambiguous.

D. GROUPING OF POLICIES

The policies are labeled both on the wiki and on the JIRA site with all of the policies within their group. If information can flow between policies, they are given each other’s labels so that a simple lookup of each label shows all other policies within that group. For example, the DON-B policy has the following labels: donb, npsinstb1, llpb1, and llpb2 which represent all of the policies that may be affected by changes to the DON-B policy (Figure 7). All of the other policies within this group also have at least these labels. In the case of the NPS Inst-B1, it references both DON-B and DON-C, so it carries an additional label (Figure 8). Likewise, the LLP-B1 policy carries the labels for DON-A, NPS Inst-A2, and DON-C in addition to the other labels that DON-B carries (Figure 9).



DoN-B

Details

Type:	Task	Status:	Open (View Workflow)
Priority:	Major	Resolution:	Unresolved
Labels:	donb llpb1 llpb2 npsinstb1		

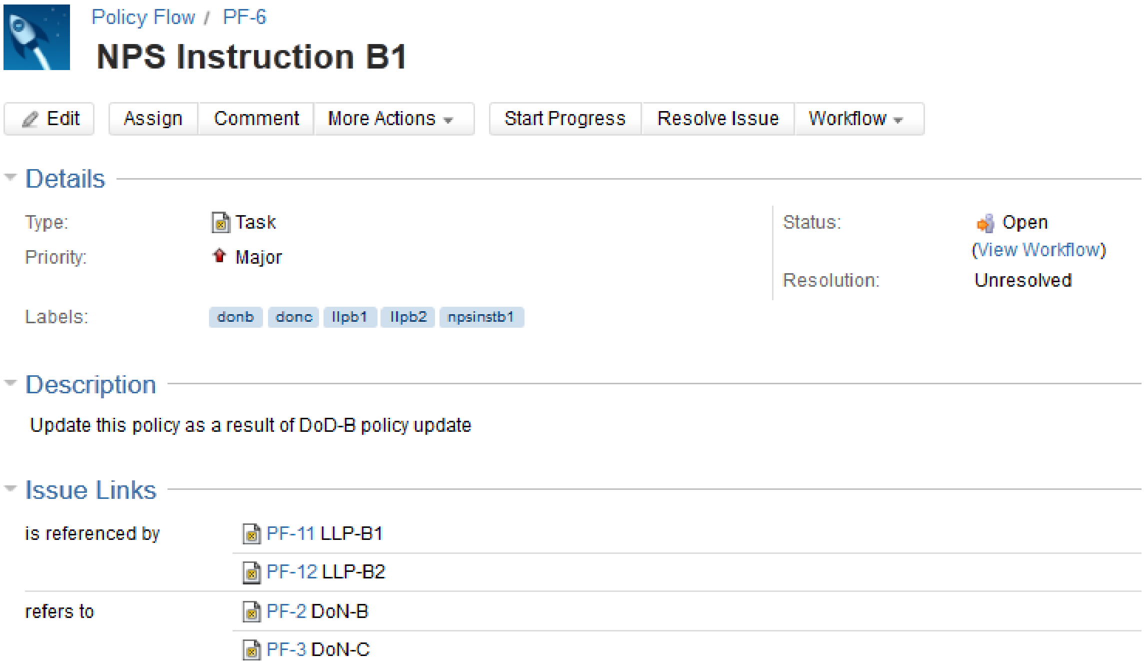
Description

Update policy and all linked policies found by wiki labels and linked issues

Issue Links

is referenced by	PF-6 NPS Instruction B1
	PF-11 LLP-B1
	PF-12 LLP-B2

Figure 7. DON-B in JIRA



Policy Flow / PF-6

NPS Instruction B1

Details

Type: Task

Priority: Major

Labels: donb donc llpb1 llpb2 npsinstb1

Status: Open (View Workflow)

Resolution: Unresolved

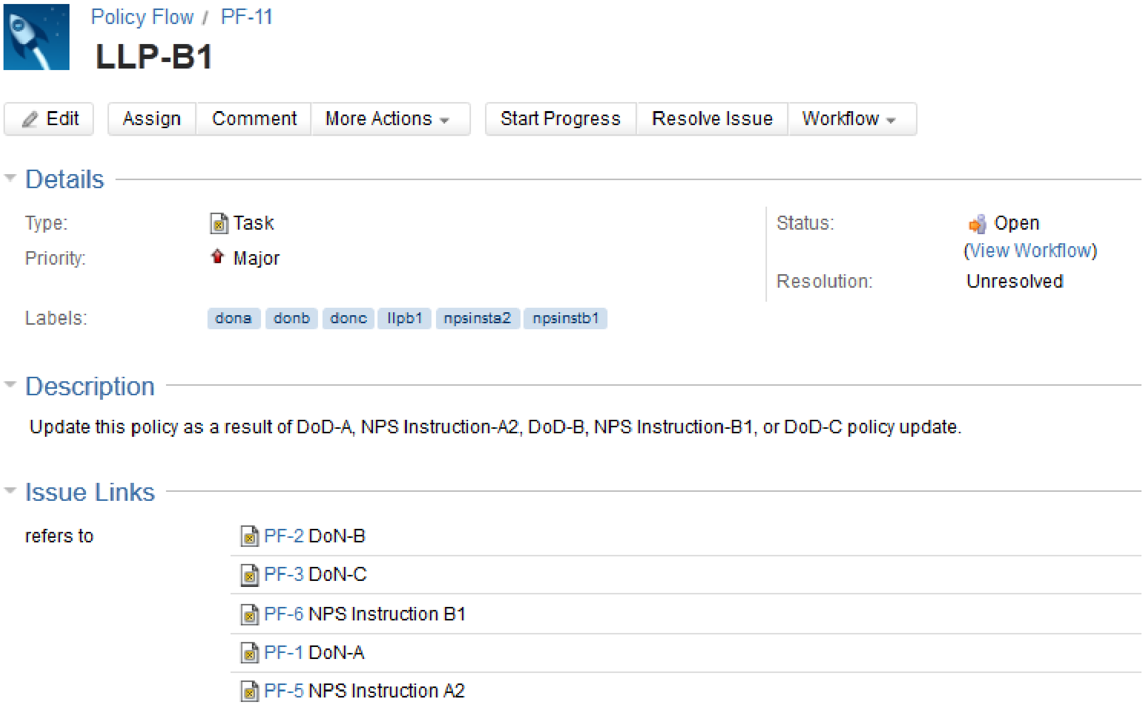
Description

Update this policy as a result of DoD-B policy update

Issue Links

is referenced by	PF-11 LLP-B1
	PF-12 LLP-B2
refers to	PF-2 DoN-B
	PF-3 DoN-C

Figure 8. NPS Instruction B1 in JIRA



Policy Flow / PF-11

LLP-B1

Details

Type: Task

Priority: Major

Labels: dona donb donc llpb1 npsinsta2 npsinstb1

Status: Open (View Workflow)

Resolution: Unresolved

Description

Update this policy as a result of DoD-A, NPS Instruction-A2, DoD-B, NPS Instruction-B1, or DoD-C policy update.

Issue Links

refers to	PF-2 DoN-B
	PF-3 DoN-C
	PF-6 NPS Instruction B1
	PF-1 DoN-A
	PF-5 NPS Instruction A2

Figure 9. Lower-Level Policy B1 in JIRA

E. LINKING AND FLOW OF POLICY UPDATES

It is important to note that the lower-level policies in the previous example have labels across groups for multiple upper-level policies, but the upper-level policies themselves remain free of these extra labels. This is because information flow from lower-level policies to upper-level policies is not allowed. The labels are not the only way to determine which policies should be updated based on changes to upper-level policies, however. The integration of the issue-tracker with the wiki is essential for the successful propagation of policy updates from upper-level policies to lower-level policies.

The capability of linking issues within JIRA allows policies to refer to each other in such a way that the flow of policy updates is obvious to users of this tool. In this example, the DON-level policies will link directly with the NPS instructions, and the instructions will link to the policies on the lower- level. This linking occurs within JIRA by assigning the policy at the upper-level as *referenced by* the policy at the lower-level, and the policy at the lower-level as *refers to* the policy at the upper-level (Figures 7, 8, 9). Policies at the same level may reference each other the same way with either a symmetrical or an asymmetrical relationship based on the hierarchy that is defined.

Since each policy housed in the wiki has a corresponding issue, and issues are linked as described above, the flow of policy information is well-defined. As a DON-level policy is updated, the administrator adjusts the corresponding issue in JIRA to reflect the status of the policy update (i.e., open, in progress, resolved, closed).

F. NOTIFICATIONS

When a change is made to a policy on the wiki or to an issue in JIRA, email notifications of the change are sent out as prescribed by the administrator. In this example, the administrator may subscribe to email notifications of changes that are made to policies within the wiki with the notification settings to include the changes within the email itself. This means that when a policy is updated, the administrator can receive an email notification that a change has happened, and the changes may be included within the email. The email lists stay current automatically because the admin can select the option to receive emails regarding any issue they create, or “mention” someone else to cause another user to receive emails regarding that issue. When the issue that corresponds to a particular policy is updated in JIRA, the administrator is notified of this as well. All of the linked issues that are associated with the issue for a particular policy that is changed within JIRA can be set to re-open if they were closed, and the workflow property in JIRA also allows due dates to be set for the tasks that are assigned. For example, if a DON-level policy is updated, the administrator will receive an email notification from the wiki of the change. The administrator goes to the wiki policy page and updates the JIRA issue to “Open.” A notification of the workflow change in JIRA is sent out. The administrator then goes to JIRA where all of the links to and from the changed policy are visible and tasks are assigned for affected policies to be reviewed. From there, it is a simple matter of completing the tasks assigned by JIRA to update the necessary policies. An example workflow diagram from JIRA is shown in Figure 10. The workflow can be customized to reflect the stage of progress that a policy is in. For example, if the workflow is altered to define a new state that indicates when a policy is in the update phase, this state is visible to everyone with the required permissions.

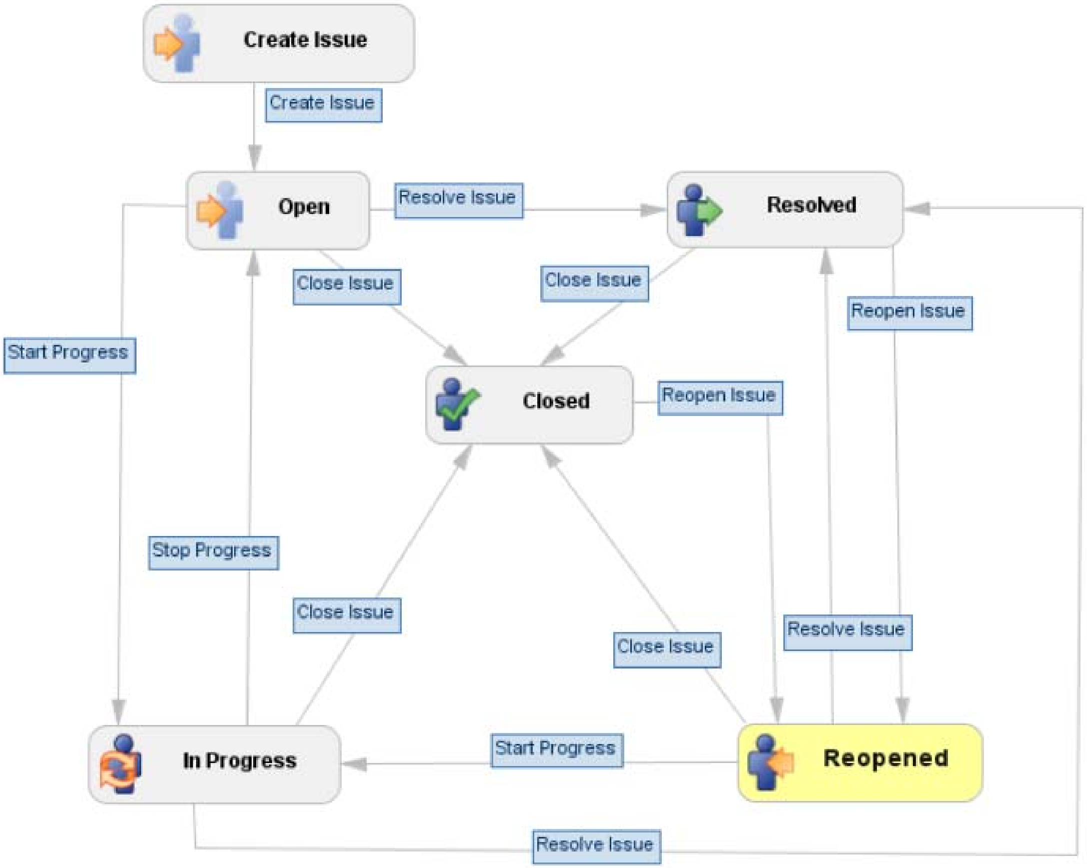


Figure 10. JIRA Issue Workflow

THIS PAGE INTENTIONALLY LEFT BLANK

V. ANALYSIS AND CONCLUSION

The taxonomy presented in this thesis is built upon the existing policy hierarchy as illustrated in Figure 1. The main difference stems from the choice that must be made in terms of the flow of information between policies when the linking between policies becomes complicated. The original policy hierarchy does not address the problem of how to group policies when information flows from multiple upper-level policies to multiple lower-level policies, and when lower-level policies reference each other. This has been a significant barrier for using wiki technology to link policies and have updates propagated from top to bottom in the IA policy domain. So far, each government agency deals with this problem as they see fit. This means that even though IA policies have been migrated to an online domain at the top (DON) levels, the lower-levels where implementation of these policies is performed suffer without the infrastructure to continue the chain of policy updates.

The weak link in IA policy maintenance lies within the implementation level, since many agencies continue to deal with PDF or paper versions of policies. In this day and age, there is no reason to resist the movement of IA policy maintenance to online collaborative environment, such as a wiki. Many government agencies have taken the first steps to moving their IA policies online, but with no clear workflow to keep them updated, they are still sifting through the same manual process that has plagued the government since the times of hard-copy policy maintenance. This research presents the infrastructure necessary to finally link policies and provide automatic conveyance of policy updates in a timely and cost-effective manner.

Given the amount of time and resources that must be spent on policy maintenance, the ideas described in this research present cost-effective solutions that require moderate up-front work to implement, and minimal effort to maintain. In addition to this, versions of IA policies remain available and are distinguished by version numbers that give a clear indication of which policy is the current one without denying access to older policy versions. This is a vast improvement over the current practice within government agencies, and the benefits are not limited to the IA policy family. This structure gives the foundation for policy maintenance that can easily be extended to any class of policy that exists in a hierarchy. The impact of the practices outlined by this research is far-reaching, and if this approach is tailored to deal with other hierarchical updating schemes, the efficiency of policy maintenance will dramatically improve. Having policies that are up-to-date is essential to the daily functioning of any organization.

VI. FUTURE WORK

A. POLICY WIKI WITHIN U. S. GOVERNMENT

This research has addressed the problem of IA policy maintenance at the implementation level of the government, but there is potential for this type of collaboration at every level within the government. The link that is weakest at the moment is the implementation level of policy maintenance. As this is dealt with, the next step is to have wiki collaboration at the DON level on all policies, just as Intellipedia is a wiki for government intelligence. From there, it is conceivable that this model be implemented at the DoD-level to ensure compliance with and maintenance of policies government-wide.

In the immediate future, this research will be applied at NPS by the Information Technology and Communications Services (ITACS) department. This will ensure that NPS maintains the readiness of our defense technologies and systems, and continues to enhance national security by maintaining current IA policies.

Further work is required to determine a suitable hierarchy across the government for all policies to be housed on the same or compatible wiki environments to allow smooth policy updates to be pushed from the top level to the lower levels. The focus of most policy work within the government has been at the DON level and above, but when policy requirements are ultimately executed at the implementation level, broken lines of communication can have a devastating effect on the operational readiness of our forces. This is an important focus of future research and development.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

[1] S. Harris, *All in One CISSP Exam Guide,* 5th ed. New York: McGraw-Hill, 2010.

[2] P. F. Merrill, "Hierarchical & information processing task analysis: A comparison," in *Journal of Instructional Development*, 1978, pp. 35–40.

[3] C. Y. Chung, M. Gertz, N. Sundarsesan, "Reverse engineering for web data: From visual to semantic structures," in *Proceedings of the 18th International Conference on Data Engineering (ICDE '02),* 2002.

[4] A. Maedche, S. Staab, "Ontology Learning for the Semantic Web," *IEEE Intelligent Systems,* vol.16, pp. 72–79, March 2001.

[5] O. Svab-Zamazal, V. Svatek, "Towards ontology matching via pattern-based detection of semantic structures in OWL ontologies," University of Prague, Dept. Information and Knowledge Engineering, 2008.

[6] D. Ariely, "Controlling the information flow: effects on consumers' decision making and preferences," *Journal of Consumer Research,* vol. 27, pp. 233–248, Sept. 2000.

[7] J. Rasmussen, "The role of hierarchical knowledge representation in decisionmaking and system management," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. smc-15, pp. 234–243, March/April 1985.

[8] "Intellipedia," in *Wikipedia*, n.d., http://en.Wikipedia.org/Wiki /Intellipedia [Accessed Nov. 14, 2012].

[9] Central Intelligence Agency, "Intellipedia marks second anniversary in CIA," March 20, 2008. [Online] Available: https://www.cia.gov/news-information/featured-story-archive/2008-featured-story-archive/intellipedia-marks-second-anniversary.html.

[10] Logen, C. "How to make a wiki government in dotGovWatch," Jan. 3, 2008. [Online] Available: http://www.dotgovwatch.com/?/archives/24-How-to-Make-a-Wiki-Government.html.

[11] McKinsey & Co., "E-government 2.0," McKinsey Quarterly, Summer 2009. [Online]. Available: http://www.mckinseyquarterly.com/E-government_20_2408.

[12] L. Ding et al., "TWC Data-Gov Corpus: Incrementally generating linked government data from Data.gov," in *WWW 2010,* April 20–24, 2010, Raleigh, North Carolina. ACM 978-1-60558-799-8/10/04.

[13] A. Chavan, "Developing an open source content management strategy for E-government," in *Proceedings of the 42nd Annual Conference on the Urban and Regional Informational Systems Association*, 2004, pp. 98–107.

[14] Atlassian, "JIRA 101," n.d. [Online]. Available: https://confluence.atlassian.com/display/JIRA/JIRA+101.

[15] S. Mukherjee, G. Yang, W. Tan, I. V. Ramakrishnan, "Automatic discovery of semantic structures in HTML documents," in *Proceedings of the Seventh International Conference on Document Analysis and Recognition (ICDAR'03)*, 2003, pp. 245–249

[16] Welliver, T. "NPS policy hierarchy figure in Naval Postgraduate School wiki," Dec. 1, 2011. [Online]. Available: https://Wiki.nps.edu/plugins/gliffy/viewlargediagram.action?name=nps-policy-heirarchy&ceoid=22414656&key=IA&pageId=22414656.

[17] Department of the Navy Issuances, n.d. [Online]. Available:http://doni.daps.dla.mil/default.aspx.

[18] Atlassian, "Confluence 101," n.d. Available: https://confluence.atlassian.com/display/DOC/Confluence+101.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. NAWCAD
   Patuxent River, Maryland

4. Karen L. Burke
   Naval Postgraduate School
   Monterey, California

5. Zachary Peterson
   Naval Postgraduate School
   Monterey, California

6. Christine Haska
   Naval Postgraduate School
   Monterey, California