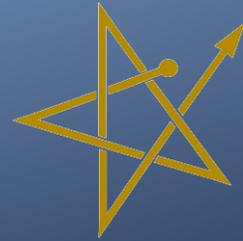


# NPRST



**Navy Personnel Research, Studies, and Technology**

5720 Integrity Drive • Millington, Tennessee 38055-1000 • [www.nprst.navy.mil](http://www.nprst.navy.mil)

research at work

**NPRST-TN-14-1**

**July 2014**

## **System Design for Navy Occupational Standards Development**

**David Dickason**  
*Navy Personnel Research, Studies, and Technology*

**Michael Marano**  
*SERCO, INC*

Approved for public release; distribution is unlimited.



# Report Documentation Page

*Form Approved*  
*OMB No. 0704-0188*

Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.

1. REPORT DATE <b>JUL 2014</b>	2. REPORT TYPE	3. DATES COVERED <b>01-10-2010 to 30-09-2012</b>			
4. TITLE AND SUBTITLE <b>System Design for Navy Occupational Standards Development</b>		5a. CONTRACT NUMBER			
		5b. GRANT NUMBER			
		5c. PROGRAM ELEMENT NUMBER			
6. AUTHOR(S)		5d. PROJECT NUMBER			
		5e. TASK NUMBER			
		5f. WORK UNIT NUMBER			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) <b>Navy Personnel Research, Studies, &amp; Technology (NPRST/BUPERS-1),Bureau of Naval Personnel,5720 Integrity Drive,Millington,TN,38055</b>		8. PERFORMING ORGANIZATION REPORT NUMBER			
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)		10. SPONSOR/MONITOR'S ACRONYM(S)			
		11. SPONSOR/MONITOR'S REPORT NUMBER(S)			
12. DISTRIBUTION/AVAILABILITY STATEMENT <b>Approved for public release; distribution unlimited</b>					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT <b>Same as Report (SAR)</b>	18. NUMBER OF PAGES <b>56</b>	19a. NAME OF RESPONSIBLE PERSON
a. REPORT <b>unclassified</b>	b. ABSTRACT <b>unclassified</b>	c. THIS PAGE <b>unclassified</b>			

NPRST-TN-14-1  
July 2014

# **System Design for Navy Occupational Standards Development**

David Dickason  
Navy Personnel Research, Studies, and Technology

Michael Marano  
SERCO, INC

Reviewed, approved, and released by  
David M. Cashbaugh  
Director

Approved for public release; distribution is unlimited.

Navy Personnel Research, Studies, and Technology (NPRST/BUPERS-1)  
Bureau of Naval Personnel  
5720 Integrity Dr.  
Millington, TN 38055-1000  
[www.nprst.navy.mil](http://www.nprst.navy.mil)

# REPORT DOCUMENTATION PAGE

Form Approved  
OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing this collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden to Department of Defense, Washington Headquarters Services, Directorate for Information Operations and Reports (0704-0188), 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to any penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number. **PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ADDRESS.**

<b>1. REPORT DATE (DD-MM-YYYY)</b>			<b>2. REPORT TYPE</b>			<b>3. DATES COVERED (From-To)</b>			
07/21/2014			Technical Note			1 OCT 2010 – 30 SEPT 2012			
<b>4. TITLE AND SUBTITLE</b>						<b>5a. CONTRACT NUMBER</b>			
System Design for Navy Occupational Standards Development						N00189-10-D-Z003-0004, N00189-10-D-Z003-0002			
						<b>5b. GRANT NUMBER</b>			
						<b>5c. PROJECT ELEMENT NUMBER</b>			
<b>6. AUTHORS</b>						<b>5d. PROJECT NUMBER</b>			
David Dickason, NPRST Mike Marano, SERCO, INC						<b>5e. TASK NUMBER</b>			
						<b>5f. WORK UNIT NUMBER</b>			
<b>7. PERFORMING ORGANIZATIONS NAME(S) AND ADDRESS(ES)</b>						<b>8. PERFORMING ORGANIZATION REPORT NUMBER</b>			
Navy Personnel Research, Studies, & Technology (NPRST/BUPERS-1) Bureau of Naval Personnel 5720 Integrity Drive Millington, TN 38055-1000									
<b>9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)</b>						<b>10. SPONSOR/MONITOR'S ACRONYM(S)</b>			
Navy Manpower Analysis Center (NAVMAC) Code 10 – Workforce Classification Millington, TN						<b>11. SPONSOR/MONITOR'S REPORT NUMBER(S)</b>			
<b>12. DISTRIBUTION/AVAILABILITY STATEMENT</b>									
<b>13. SUPPLEMENTARY NOTES</b>									
<b>14. ABSTRACT</b>									
<p>This document describes the current process of Navy enlisted Occupational Standards development and a proposed system architecture to enhance that process. The proposed system design seeks to improve the functioning of the current OCCSTDs development process. The report starts with a description of the current process as a reference for system design details presented later in the document. This system architecture seeks to enhance the existing OCCSTDs development process through developing three main components: 1) A central repository of OCCSTDs data; 2) a collaboration framework that enables collaboration between NAVMAC personnel and Navy SMEs, and; 3) a reporting framework for publishing OCCSTDs data.</p>									
<b>15. SUBJECT TERMS</b>									
Occupational standards, system architecture, collaboration.									
<b>16. SECURITY CLASSIFICATION OF:</b>						<b>17. LIMITATION OF ABSTRACT</b>	<b>18. NUMBER OF PAGES</b>	<b>19a. NAME OF RESPONSIBLE PERSON</b>	
UNCLASSIFIED						UNLIMITED	56	Wendy Douglas	
<b>a. REPORT</b>	<b>b. ABSTRACT</b>		<b>c. THIS PAGE</b>		<b>19b. TELEPHONE NUMBER</b>				
UNCLASSIFIED	UNCLASSIFIED	UNCLASSIFIED		(901)874-2218					

## Foreword

This document describes the current process of Navy enlisted Occupational Standards development and proposed system architecture to enhance that process. It was sponsored by the Navy Manpower Analysis Center (NAVMAC). We would like to thank a number of NAVMAC (Code-10) employees without whom this effort would not have succeeded, including, Mr. Thomas Crain, Deputy Director, Workforce Classifications Department, LCDR Juan Carrasco, Michele Jackson, and Johnny Powell.

David M. Cashbaugh  
Director

# Contents

<b>Foreword</b> .....	<b>iii</b>
<b>Introduction</b> .....	<b>1</b>
Problem .....	1
Objective .....	1
Description of Current OCCSTD Development Process.....	1
<b>Overview of Process</b> .....	<b>2</b>
Data Management .....	2
Subject Matter Expert Review.....	3
Data Analysis .....	5
Implementation .....	6
<b>Proposed System Design</b> .....	<b>7</b>
System Overview .....	7
Design Considerations.....	8
Assumptions and Dependencies .....	8
General Constraints .....	8
Goals and Guidelines .....	9
Development Methods.....	9
Architectural Strategies .....	9
System Architecture.....	10
System Deployment.....	12
<b>Detailed System Design: Navy Occupational Data Store (NOD)</b> .....	<b>13</b>
NOD Interactions and Uses .....	13
NOD Design Strategy .....	14
NOD Stored Procedures .....	17
Deployment Strategy .....	18
NOID-NOD Data Migration Strategy .....	19
<b>Detailed System Design: Collaboration Framework (CF)</b> .....	<b>19</b>
Site Organization and Navigation Strategy.....	20
Collaborative Workspaces.....	21
Collaboration Framework Interactions and Uses.....	27
Deployment Strategy .....	29
Survey Integration Strategy .....	30
<b>Detailed System Design: Reporting Framework (RF)</b> .....	<b>33</b>
Types of Reports Required .....	33
Report Formats .....	34
Generating Reports.....	35
Microsoft SQL Server Reporting Services .....	36
Report Tracking with SQL Server Reporting Services .....	37
Planning a Report With SQL Server Reporting Services .....	37
Report Viewer .....	39
<b>Summary</b> .....	<b>41</b>

<b>APPENDIX A</b> .....	<b>A-0</b>
NOID Data Elements.....	A-1
<b>APPENDIX B</b> .....	<b>B-0</b>
Occupational Standard Feedback Form .....	B-1
<b>APPENDIX C</b> .....	<b>C-0</b>
Task Survey Process .....	C-1

## Figures

Figure 1. Current OCCSTD Development Process .....	2
Figure 2. Data Management Phase of Current OCCSTD Process .....	2
Figure 3. SME Review Phase of Current OCCSTD Process .....	4
Figure 4. Data Analysis Phase of Current OCCSTD Process .....	6
Figure 5. Production Step within Implementation Phase of Current Process .....	6
Figure 6. Publication Step in Implementation Phase of Current Process .....	6
Figure 7. Venn Diagram: Primary System Components .....	11
Figure 8. Deployment Overview .....	12
Figure 9. Use Case Diagram: NOD.....	14
Figure 10. Entity Relationship Diagram for NOD.....	15
Figure 11. Historical Entity Relationship Diagram for NOD .....	16
Figure 12. SharePoint Sub-Sites .....	20
Figure 13. Collaboration Workspace Example.....	22
Figure 14. OCCSTDs Task List Example .....	23
Figure 15. OCCSTDs Task Discussion Example (Flat View) .....	23
Figure 16. Use Case Diagram: Collaboration Framework.....	28
Figure 17. Process Diagram: Collaboration Framework .....	29
Figure 18. QMWISe Integration .....	31
Figure 19. Repository Pattern Applied to Web Services.....	32

# Introduction

This Technical Note describes the production process for the development of Navy Enlisted Occupational Standards (OCCSTDs) for all ratings, paygrades E4 - E7, and a system architecture intended to enhance the existing OCCSTDs development process.

An OCCSTD describes the Navy's minimum requirements and skills for a Navy enlisted rating, in the form of tasks, skills, and abilities, as endorsed by the rating's primary resource or warfare sponsor.

## Problem

Currently the process for OCCSTDs development is manual and utilizes various tools, software, definitions, and measures. There is no capability for creating a collaborative workspace to allow multiple analysts or subject matter experts to work on developing or retaining the task statements that are the foundation for OCCSTDs. The majority of the task analysis data that forms the basis for OCCSTDs currently resides in a Microsoft Access database called the Navy Occupational Interim Database (NOID; see Appendix 1 for data elements).

## Objective

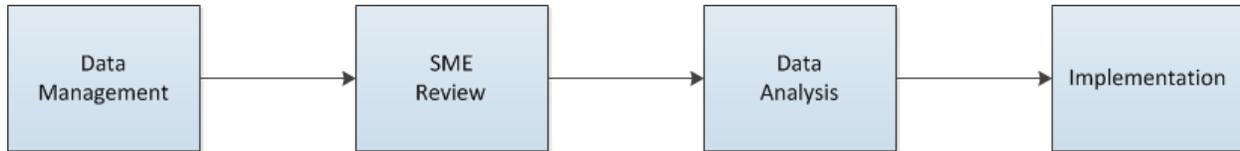
There is a high-priority need to increase access and visibility, reduce errors and approval times, institute administrative metrics, and standardize the occupational classification workflow and data processes across the Navy's enlisted force. In particular, the new capability should better enable NAVMAC (Code-10) to: analyze past data; capture new Navy occupational data; analyze, manage, and store the data; make data accessible to outside sources; produce Navy Total Force (NTF) classification standards; and publish validated standards.

## Description of Current OCCSTD Development Process

The proposed system design seeks to improve the functioning of the current OCCSTDs development process. The first report section will provide a description of the current process as a reference for design details presented later in the document. This process section will describe the current development process which uses subject matter expert (SME) panels to review and revise task inventory lists in lieu of a more traditional job-task analysis survey. The proposed system design can be used with either approach (SME or survey) and follows the process description.

## Overview of Process

OCCSTDs are developed in four sequential, broad phases as illustrated in Figure 1.



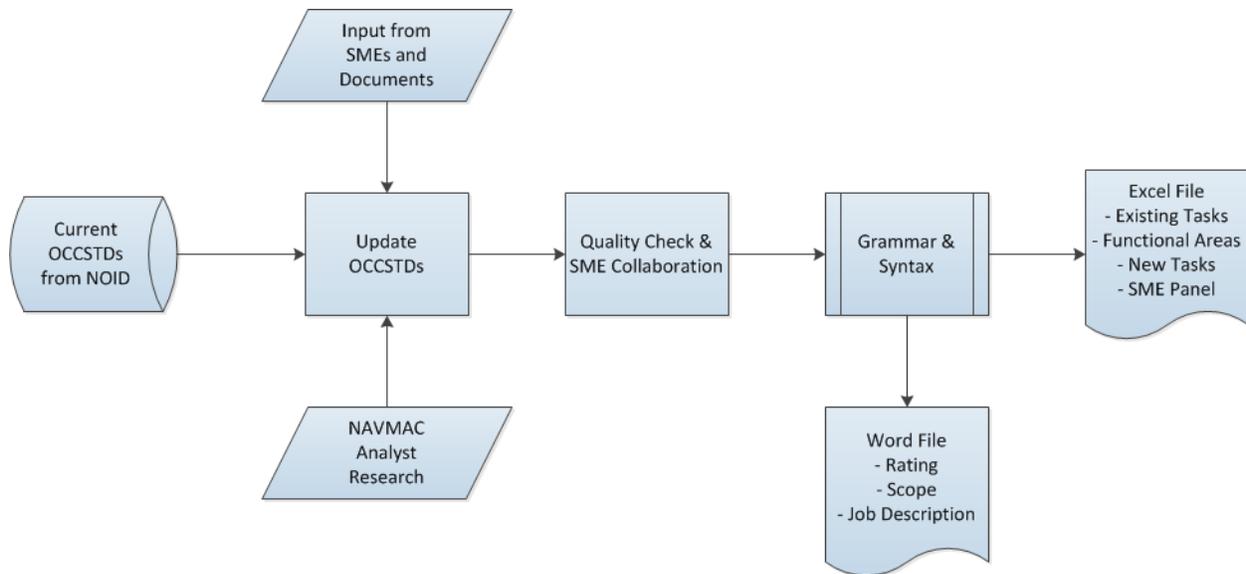
**Figure 1. Current OCCSTD Development Process**

Each phase of OCCSTD development is described in the following sections.

### Data Management

The Data Management phase includes a review of existing OCCSTDs and a comprehensive collection of more current information for the rating of interest. The output of this phase is a proposed task inventory to be provided to subject matter experts<sup>1</sup> for review and comment.

Figure 2 shows the steps included in this phase.



**Figure 2. Data Management Phase of Current OCCSTD Process**

<sup>1</sup> Currently, the Learning Center or Center of Excellence responsible for the rating under consideration convenes a panel of subject matter experts.

Existing OCCSTDs are maintained in a Microsoft Access database, the Navy Occupational Interim Database (NOID). The data elements that comprise the NOID are listed in Appendix 1. To start the review of a rating's OCCSTD, NAVMAC analysts retrieve that existing information from the NOID. The NOID data is augmented by information from NAVMAC research and input from SMEs who work in or supervise rating activities. The augmented information may include training or maintenance documentation from manuals, formal and informal correspondence, and known hardware systems changes. Task related information is typically received in both hardcopy and electronic formats making the process labor intensive and tedious.<sup>2</sup> Extracting task data from document sources can be labor intensive as it requires the analyst to search for verb-object pairings (such as "open-valve" or "remove-filter"). Appendix 2 provides an example of the preferred format for submission of information by SMEs. The result of merging existing OCCSTDs with input from NAVMAC research and SME input is the "Update OCCSTDs" box shown in Figure 2.

The updated OCCSTDs undergo a quality control check to ensure the information is accurate and consistent. NAVMAC analysts work closely with the subject matter experts during the quality check. Upon completion of the quality check, the updated OCCSTD is formatted into a strict structure of nouns, modifiers, and verbs that constitute the official grammar and syntax system.

The final step in the Data Management phase involves capturing the updated and properly structured OCCSTD in a Microsoft Excel file that includes four work sheets: existing tasks from the previous review, functional area of the rating under review, a list of Skills and Abilities as defined by O\*Net<sup>3</sup>, and space to capture the SME who will accomplish the formal review.

It is important to note that an audit trail is established to identify and justify any changes in task statement during development by analysts and SMEs. However, at present there is no mechanism to effectively retain information on a pending task within NOID. A pending task is a task that may or may not be included in the final task inventory. It is important to be able to accurately track wording changes and other task statement attributes (i.e., rationale for the change or who is advocating inclusion/exclusion of a particular task) so as not to lose process information in task inventory development.

## **Subject Matter Expert Review**

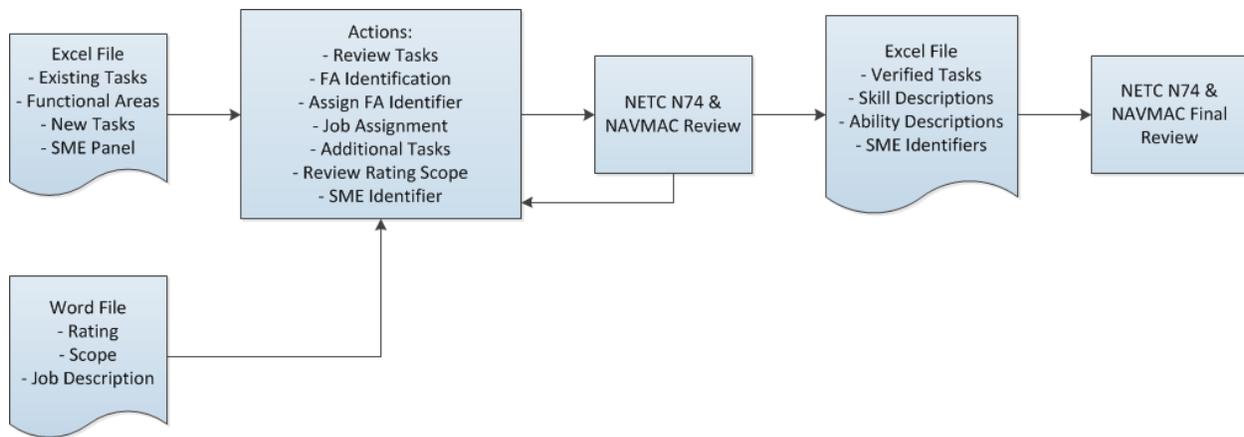
The next phase in the current OCCSTD development process consists of review of the preliminary task inventory list, scope, and job description(s) by a panel of subject matter experts outside of NAVMAC.

Figure 3 is an overview of the steps.

---

<sup>2</sup> It should be noted that this process has changed over time. Historically, NAVMAC sent analysts into the field to observe Navy personnel performing relevant tasks. Reductions in NAVMAC resources have led to the elimination of on-site observations.

<sup>3</sup> O\*NET, the Occupational Information Network, is a comprehensive database of worker attributes and job characteristics and is the replacement for the Dictionary of Occupational Titles (DOT).



**Figure 3. SME Review Phase of Current OCCSTD Process**

Upon receipt of the Microsoft Excel and Microsoft Word files<sup>4</sup> from NAVMAC , SMEs perform the following actions:

- Review the tasks in the Existing Task worksheet to determine continued agreement or disagreement. Comments are provided to support the recommendation. NAVMAC provides detailed definitions and rules to enable SMEs to distinguish among OCCSTD, Navy Standards, job-specific tasks, and NEC-related tasks.
- Review the Functional Area list provided in the Functional Area worksheet and assess if the area is current, should be revised, is obsolete, or needs a new Functional Area.
- Assign a Functional Area code to all current areas.
- Identify all the tasks that are performed in the particular jobs listed in the Existing Task worksheet.
- Identify additional tasks and ensure Grade, Functional Area, job assignment(s), and justification are provided.
- Review, update, or add Skills and Abilities as appropriate
- Review the existing rating scope and job description statements and recommend changes if necessary.
- Record name, rate, and rank of individual providing the information.

NAVMAC verifies the input received from the SME review. Verification involves further coordination with SME and consultation with resource and warfare sponsors. Upon completion of verification, NAVMAC forwards, through NETC N74, a second Microsoft Excel file to SMEs at the appropriate Learning Center. That file contains four worksheets: Verified Tasks, Skill Description, Ability Description, and SME Identifier.

<sup>4</sup> Various files containing pre-formatted worksheets are currently emailed back and forth between NAVMAC and SMEs in the review and development of task inventory statements.

SMEs use the Microsoft Excel file to accomplish the following:

- Review the tasks in the Verified Task worksheet.
- For each task, assign the two most appropriate skills listed in the Skills Description worksheet.
- For each task, assign the two most appropriate abilities listed in the Abilities description worksheet.
- Complete name, rating, and rank in the SME Identifier worksheet.

Previously, there were agreements that once the task inventory was complete, the task statements would be used to develop a job analysis survey. This would be the beginning of the Survey phase. Historically NAVMAC would survey 65% of a rating's population (stratified by paygrade and platform or activity) with a target response rate of 40-50% by paygrade. The survey sample could be developed concurrently with the survey development. Currently NAVMAC is not resourced and does not have a survey tool to collect task analysis data using survey methods.

## Data Analysis

The Data Analysis phase is designed to closely examine the more refined OCCSTDs and related supporting information. As noted previously, NAVMAC prefers to use a survey of individuals performing duties in the rating of interest to collect OCCSTD information and such a survey is expected to be used in an enhanced process. However, funding constraints now limit the collection and analysis to a panel of SMEs.

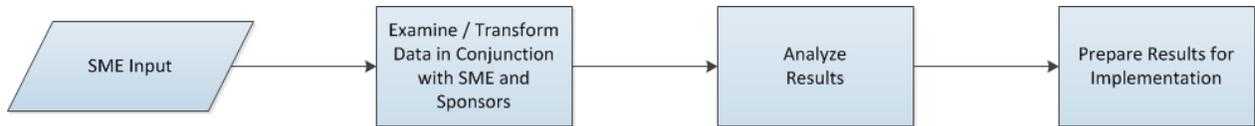
When a survey is used, detailed analysis is performed to ensure the validity of responses (e.g., minimum level of response and detection of undesirable response patterns). Descriptive statistics<sup>5</sup> are then generated and data is separated by groups and by certain statistics (e.g., by task, by paygrade, by percentage of responses, etc). Data is annotated (commented on) by the analyst and a list of tasks identified by respondents is created. This list includes validated new tasks to be added, obsolete tasks to be deleted (archived), and task statements that may need to be modified because of changes in systems since the last review. Valid tasks were identified as tasks with a 20% or higher response rate by paygrade. A supplementary list of tasks that did not have the minimum 20% response rate was also created. Results were archived along with annotations and summaries.

A comparable analysis is applied to the current input from SME panels. The focus is on comparing responses among panel members, inconsistent patterns within individual responses, and consistency between two or more organizations that may be involved in submitting input.

Figure 5 shows the general steps in the Data Analysis phase for the current process.

---

<sup>5</sup> These may include the demographics of the sample, sample size, response rate, confidence level, number of Commands involved, number of responses by task and paygrade, etc.



**Figure 4. Data Analysis Phase of Current OCCSTD Process**

## Implementation

The Implementation phase of the current OCCSTDs development process includes production and publication. Figure 5 shows the production step.



**Figure 5. Production Step within Implementation Phase of Current Process**

An important action during OCCSTDs production is categorizing validated tasks by functional area according to predefined criteria. A functional area provides a standardized grouping of similar tasks by paygrade. Currently, functional areas are determined during the internal review with confirmation from SMEs. Throughout the entire OCCSTD process (starting in Phase 1, Data Management), considerable effort is devoted to ensuring that the grammatical style and form of tasks are consistent across tasks and among other rating OCCSTDs. Once the validated tasks are categorized by functional area and paygrade in the correct style, “header” information (Job Codes, job description, DoD crosswalk information, etc.) is appended to the document and the draft OCCSTD is circulated within NAVMAC for comment as a quality assurance step. Once this is completed, preliminary OCCSTDs are generated for external review. Comments are tracked and, if justified, appropriate modifications to the OCCSTD are made. Once all the changes are consolidated, the proposed OCCSTDs are submitted to the Resource Sponsor or Primary Advisor for review and endorsement. Time for review and endorsement is limited so schedules are closely monitored. Once formal endorsement is given by sponsors, the OCCSTDs are promulgated and published.

Figure 6 shows the Publication step of the Implementation phase.



**Figure 6. Publication Step in Implementation Phase of Current Process**

Approved OCCSTDs are published in NAVPERS 18068F Volume I, *Navy Enlisted Occupational Standards, Manual of Navy Enlisted Manpower and Personnel Classifications and Occupational Standards*. Additionally various stakeholders require other formats such as EXCEL or .pdf. The last box in Figure 5 relating to database reconciliation refers to other Navy databases such as Total Force Manpower Management System (TFMMS) or the Navy's Credentialing Opportunities Online COOL Website.

## **Proposed Navy Occupational Data Collection and Management (NODCAM) Design**

### **System<sup>6</sup> Overview**

The following is excerpted from the full Workforce Standards System Design document.

The NODCAM architecture seeks to enhance the existing OCCSTDs development process. The system incorporates three main components: a central repository of OCCSTDs data, a collaboration framework that enables collaboration between NAVMAC personnel and Navy SMEs, and a reporting framework for publishing OCCSTDs data.

The proposed central repository is a relational database that provides the other system components with OCCSTDs data. It also allows other system components to create, update, and delete OCCSTDs data. The data store will also maintain a version history to facilitate auditing activities as well as the occasional rolling back of OCCSTDs data.

The collaboration framework provides a central meeting place for participants in OCCSTDs reviews. The framework enables reviewers to extract baseline OCCSTDs data from the relational data store into the NODCAM collaboration component at the onset of a review. This relational data store is a permanent repository of OCCSTDs data. During a review, participants will be able create, update and delete occupational information and store the results in both temporary (collaboration component) and permanent formats. Structured communication protocols supported by extensive auditing capability will help to maintain an organized OCCSTDs clearinghouse.

The reporting framework is designed to support the dissemination of required electronic and written documents, ad hoc reporting requests and formal publications of occupational information.

The NODCAM design also includes measures to incorporate two related activities: the reintroduction of a survey of sailors performing tasks inherent within the rating

---

<sup>6</sup> Starting with this section, "system" will be used throughout the report to refer to the NODCAM system.

under review and the Navy Job Analysis Management (NJAM)<sup>7</sup> initiative. A commercial-off-the-shelf product, *Questionmark* by Perception has been selected to support the survey, and its integration features are specified in this design. With a goal of producing a comprehensive system for all job classification elements, the NJAM initiative serves as a guide for this system design.

## Design Considerations

### Assumptions and Dependencies

NODCAM may serve as the foundation for future expansion and process improvements in the workforce standards domain. To allow for such efforts, the design of this system should be flexible enough to accommodate additional workforce classification elements. Some candidates for future classification elements include: Naval Standards (NAVSTDS), Navy Enlisted Classification Codes (NEC), and Navy Officer coding structure (e.g., Designators, Additional Qualification Designations (AQDs), Navy Officer Billet Classifications (NOBCs), and Subspecialties (SSPs)).

The occupational data for this system shall be derived primarily from an existing database, the Navy Occupational Interim Database (NOID). The NOID is a Microsoft Access database that has evolved over time along with the OCCSTDs development process itself and the associated reporting requirements. The NOID provides a solid foundation upon which to base components of the design. However, due to its evolutionary nature, care must be taken because the NOID may contain artifacts of past efforts that are no longer relevant to current OCCSTDs development efforts.

Concurrent with this design effort, a Commercial-Off-The-Shelf (COTS) product has been selected to support a highly desired survey component of the system. NAVMAC is currently working with *Questionmark Perception*<sup>8</sup> on the development of a question type that will satisfy requirements. This work should be completed prior to December 2011. This system design will identify possible integration points for the Questionmark product.

### General Constraints

The expected use and operating environment for the system will feature the following constraints:

- All system components shall adhere strictly to Navy Marine Corps Intranet development best practices and security standards and regulations.

---

<sup>7</sup> NJAM was a previous NAVMAC effort to identify representative requirements (rather than exhaustive) and serve as a basis for determining required areas and elements for present and future capability development. The intent of having an NJAM capability is to enable NAVMAC to have a standardized, automated, and self-contained in-house tool to plan, develop, and implement occupational workforce classification management.

<sup>8</sup> Additional details regarding the **Questionmark™ Perception™** application are available at: <http://www.questionmark.com/us/perception/index.aspx>

- Web-based components of the system shall be compatible with Internet Explorer 7 and above.
- The web-based collaboration framework should accommodate a minimum of 20 concurrent users, all of whom can be assumed to be working with the same data and web resources simultaneously.
- The web-based collaboration framework (and survey subcomponent) should be accessible from shipboard internet terminals that may have limited bandwidth. Due to this client bandwidth restriction, efforts should be taken to limit the number and size of HTTP requests and responses.

## **Goals and Guidelines**

The primary goal of this system design is to enhance the existing OCCSTDs development process by introducing standardized, streamlined, and less complicated procedures to obtain occupational data and track recommendations. The existing process is functional, but can be improved upon by applying the right technologies in the correct ways. During the course of this system design and development, the introduction of additional complexity for end users should be avoided at the risk of compromising user acceptance of the system.

## **Development Methods**

NODCAM is the product of an analysis phase, an interview phase, and a prototype phase. The policies, procedures, and organizational responsibilities inherent in the existing OCCSTDs development process guided the design of NODCAM. Whenever possible, terminology, implementation of features within software applications, and storage and maintenance protocols were defined to complement the existing process. Appendix 1 provides a description of the current OCCSTDs development process.

The OCCSTDs review and development processes are continuously refined in an ongoing NAVMAC effort to provide their stakeholders with the highest quality work. Due to the evolutionary environment in which OCCSTDs are created, an agile or rapid application development approach is recommended when implementing this design. At the core of this approach is the rapid development of iterative prototypes. This design document should serve as the foundation for the development of the first NODCAM prototype. Each prototype is reviewed jointly by all project stakeholders at regular intervals during Joint Application Development (JAD) sessions. The purpose of each session is to define a set of goals and requirements for the development of the next prototype. This approach should adequately address any differences between this design and the NAVMAC processes in place at the time of development.

## **Architectural Strategies**

This design strives to leverage the functionality offered by existing commercial products. If a well-established product fulfills most of the requirements of the NODCAM, it is preferable to customize that product rather than develop a new

application. This approach generally yields fewer overall system bugs and increases the overall satisfaction of system users. Furthermore, analysis of existing products must factor in their NMCI authorization status. In some cases, it can be more difficult to get a product authorized for use in the NMCI environment than it is to build the same functionality in an original application. For this reason, products that are already in use under NMCI are preferable over those that would require review and authorization.

Microsoft SharePoint 2010 Server<sup>9</sup> has been analyzed and determined to be a good solution for satisfying the collaborative requirements of the system. It is highly configurable, provides many desired features out of the box, and should serve as an excellent starting point for system development.

Microsoft SQL Server 2008 R2<sup>10</sup> has been analyzed and determined to be a good solution for satisfying the data-related requirements of the system. In addition to it being a requirement of SharePoint 2010 Server, the product has an established track record with the Navy of being a solid, full-featured database server. The product also comes bundled with SQL Server Reporting Services<sup>11</sup> and its associated tools, which should go a long way toward satisfying the reporting requirements of the system.

This design employs a use case-based approach to convey the system design. First, we describe the intended functionality of the NODCAM. Next, we discuss the implementation strategy to achieve realization of the intended functionality. When dealing with commercial products, we shall detail the features that we get out-of-the-box vs. the features that require customization, and how the customization can be achieved. Custom components shall have an object-oriented design. An object-oriented design approach helps promote testability, reusability and extensibility, and lays the groundwork for future expansion of the system.

## System Architecture

The envisioned system has three primary purposes:

- Act as a central repository of OCCSTDs data.
- Provide NAVMAC personnel with the ability to define and refine OCCSTDs. This can involve collaboration with Navy personnel external to NAVMAC.
- Provide a variety of Navy stakeholders, both internal and external to NAVMAC, with the ability to access OCCSTDs reports.

To serve those purposes three main system components have been identified:

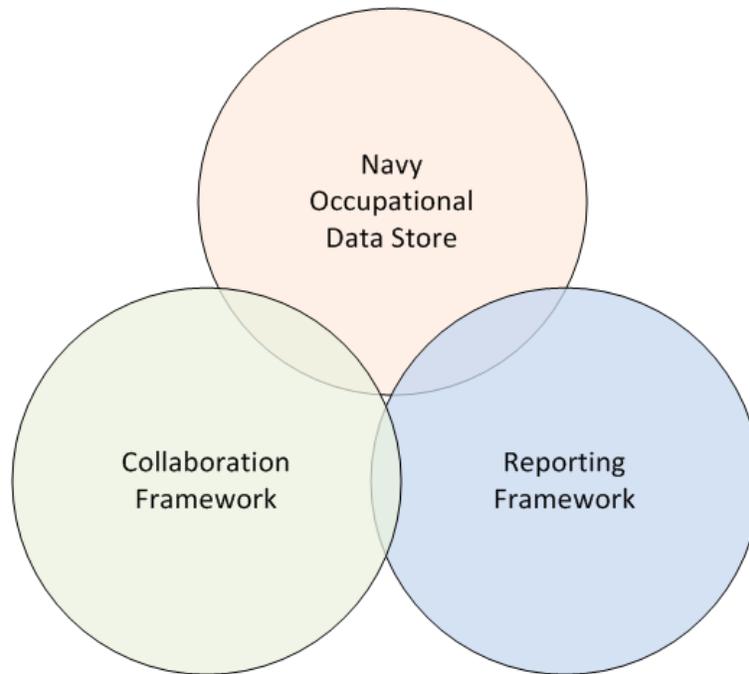
- Navy Occupational Data Store (NOD)
- Collaboration Framework (CF)
- Reporting Framework (RF)

---

<sup>9</sup> <http://sharepoint.microsoft.com/en-us/Pages/default.aspx>

<sup>10</sup> <http://www.microsoft.com/sqlserver/2008/en/us/r2.aspx>

<sup>11</sup> <http://msdn.microsoft.com/en-us/library/ms159106.aspx>



**Figure 7. Venn Diagram: Primary System Components**

The reporting framework could be considered a component of the collaboration framework. However, the collaboration and reporting framework should be viewed as mutually exclusive during the design. The reasons for separating these two components are:

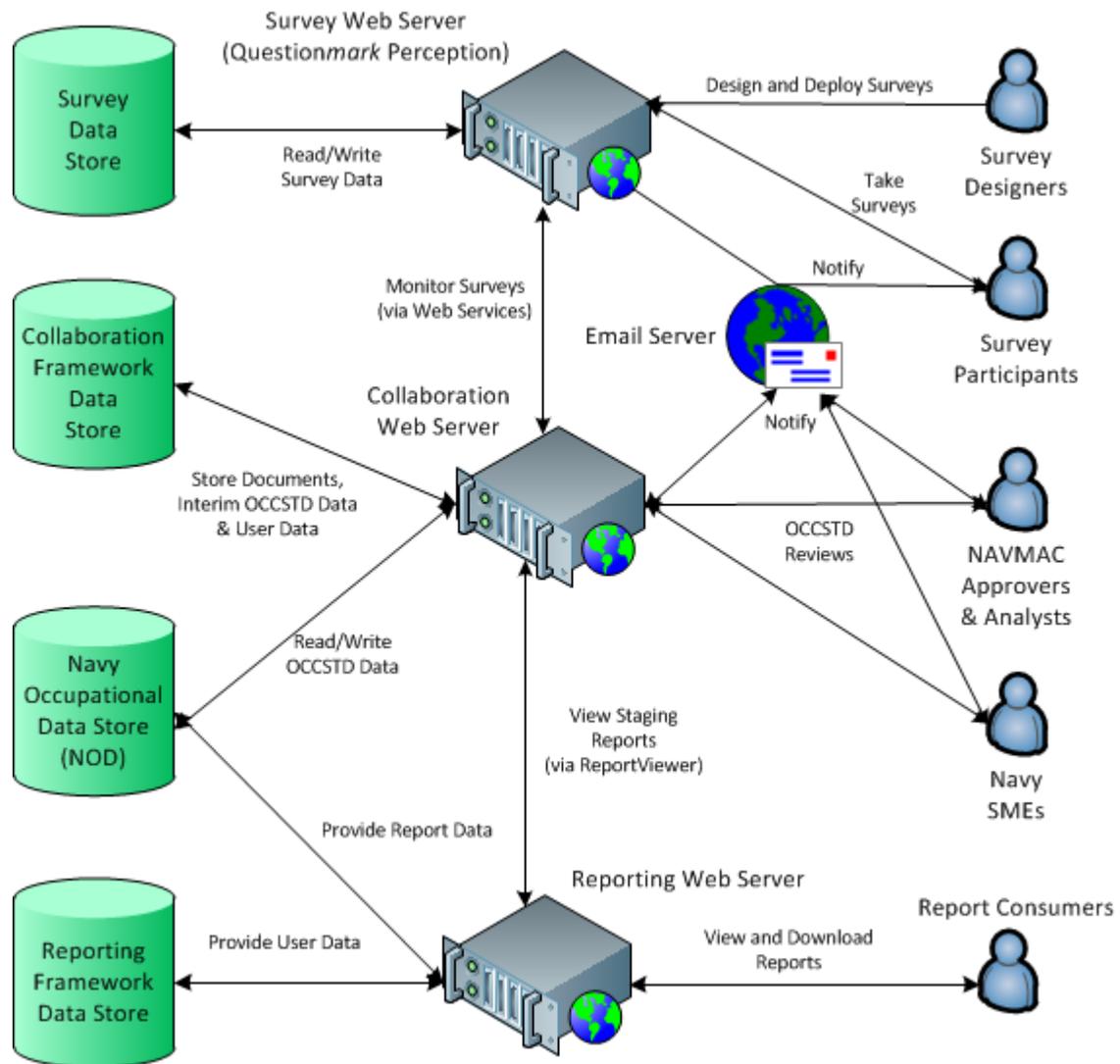
- **Licensing:** The collaboration framework will likely employ a COTS product that could have a per-user licensing agreement. Separation ensures that end users who are only interested in consuming reports are not included in that set of users.
- **Scalability and Reliability:** The reporting framework could have more end users than the collaboration framework. Separation allows for greater flexibility in the dedication of computing resources, thus making scalability easier and increasing the reliability of each component.
- **Security:** The reporting framework may have public and private regions whereas the collaboration framework should be strictly private, meaning all users need to be authenticated and authorized access prior to use. Separation allows for fine tuning access thereby reducing the surface area of each component to a minimum.

The NOD is loosely coupled with both the collaboration and reporting frameworks, but should be considered an independent entity in terms of its design. Storage of the OCCSTDs data on the CF in the form of SharePoint lists was considered during this design effort. However, the NOD database approach vs. storage in lists has significant advantages, namely: availability, maintainability, reliability, and reduced overhead and

complexity when writing queries for reports. This encapsulation of the data store also increases the portability and extensibility of the system.

## System Deployment

The overall system is composed of various components including logical database and web servers. A high-level deployment overview is depicted in the following figure.



**Figure 8. Deployment Overview**

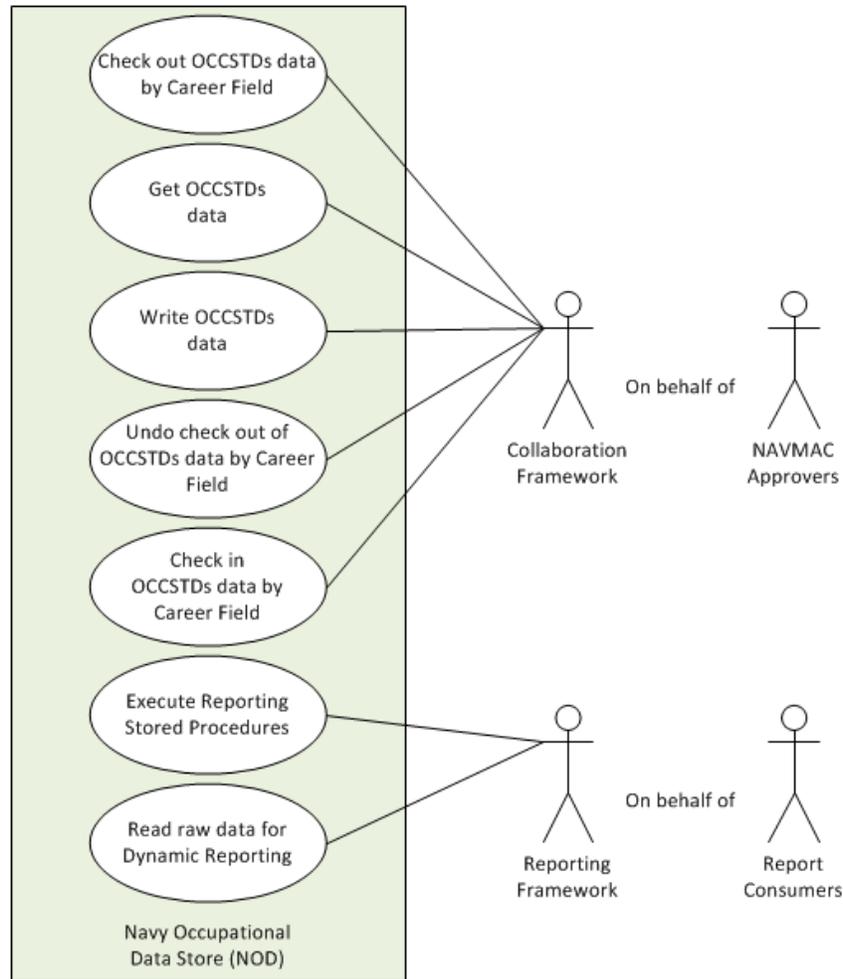
It should be noted that the above deployment overview represents logical database and web servers, not necessarily physical servers. Assuming that minimum hardware requirements are met, it is possible to consolidate logical database servers on a single physical server and likewise for web servers.

## **Detailed System Design: Navy Occupational Data Store (NOD)**

The NOD is a relational database that shall act as the central repository for all OCCSTDs data. This component should provide other system components with OCCSTDs data, via SQL queries and stored procedures. The NOD should also allow other system components to make recommendations to create, update, and delete OCCSTDs data via stored procedures. This component shall also maintain a version history to facilitate auditing activities as well as the occasional rolling back of OCCSTDs data. The system shall be designed such that changes to OCCSTDs data resulting from user activity in the Collaboration Framework are not visible to users of the Reporting Framework until they have been fully vetted and approved.

### **NOD Interactions and Uses**

The NOD interacts primarily with the other major system components (i.e., the Collaboration and Reporting Frameworks), as well as with the Database Administrator (DBA). It is assumed that the DBA has full control over the NOD. The following use case diagram depicts the uses and interactions of the NOD.



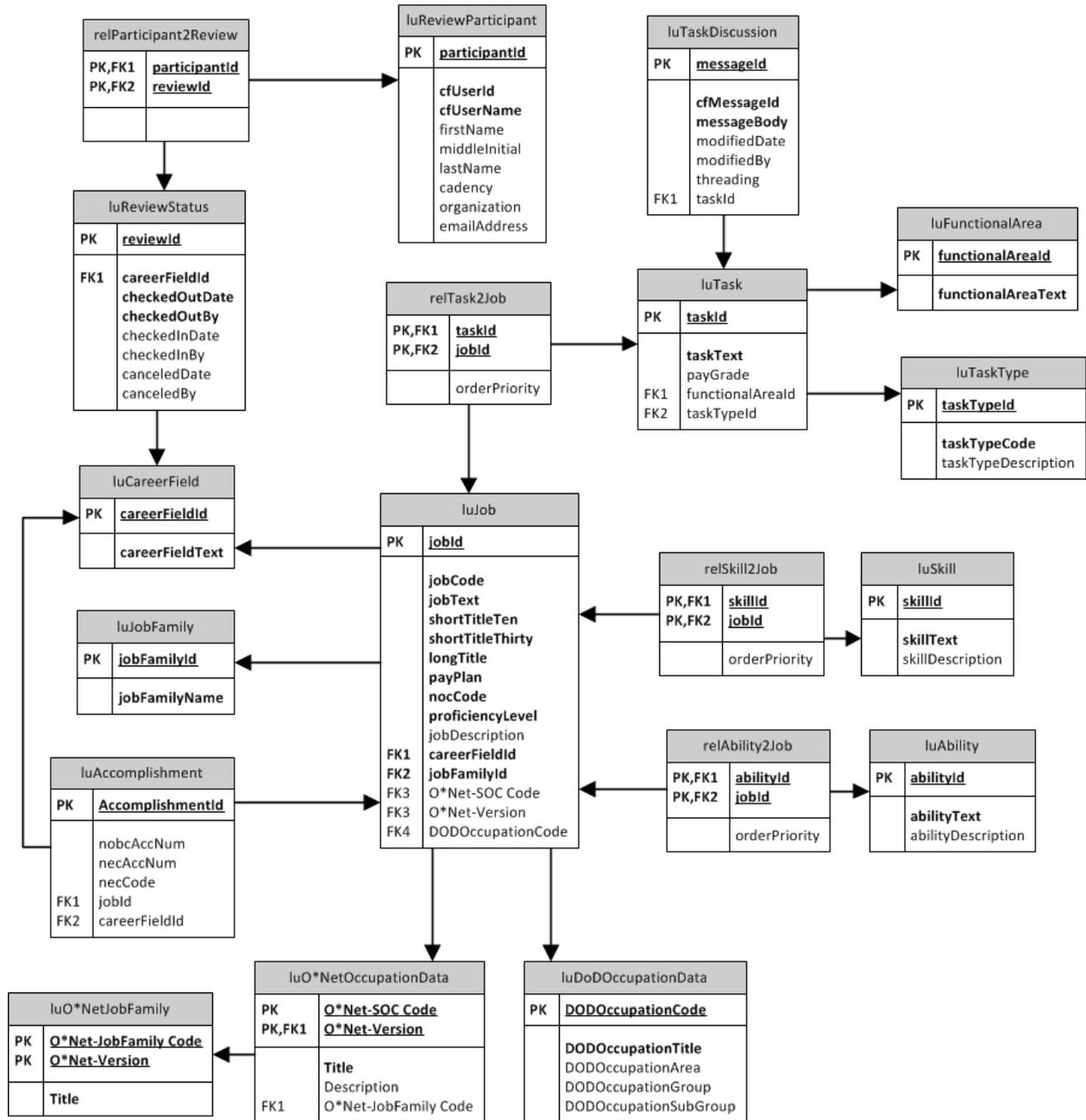
**Figure 9. Use Case Diagram: NOD**

At the onset of an OCCSTDs review, the Career Field will be “checked out” of the NOD. Upon the conclusion of the OCCSTDs review, the Career Field will be “checked in” to the NOD. This check-in/check-out approach ensures that data written to the NOD during the course of a review are associated with that review. Write access to the database should be limited to stored procedures whenever possible. For example, any INSERT, UPDATE or DELETE commands against the database should be performed exclusively through the use of stored procedures that also perform the desired level of logging and historical categorization. Doing so helps to preserve the integrity of the data and its history. The needs of data consumers should be anticipated, and read requests to the database should be accomplished via views and/or stored procedures whenever possible.

### **NOD Design Strategy**

The NOD database design should be consistent with a normalized relational database model. The following Entity Relationship Diagram (ERD) depicts the basic

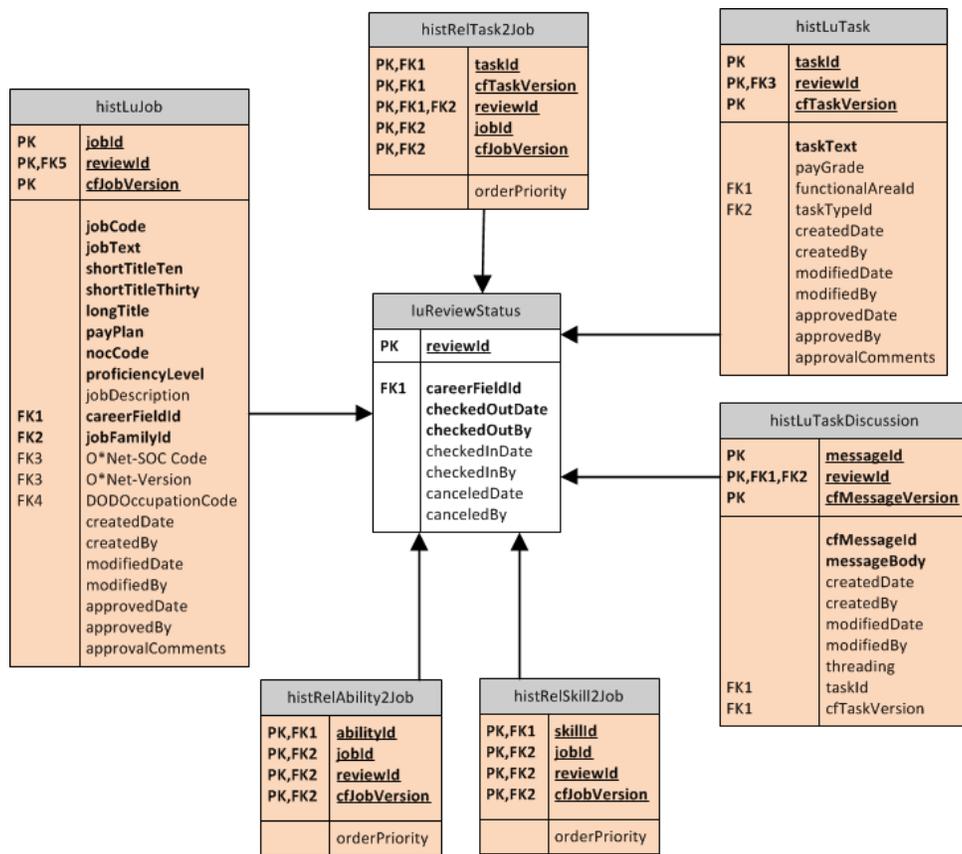
relationships among tables. Lookup tables are prefixed *lu* for clarity. One-to-many relationships are modeled through the use of foreign keys in one of the involved lookup tables. Many-to-many relationships are modeled through the use of relational tables (prefixed *rel* for clarity).



**Figure 10. Entity Relationship Diagram for NOD**

The warehousing of historical data can be accomplished by modeling a set of tables that mirrors the structure of the desired tables. Additional fields, such as the review

identifier and the collaboration framework version, should be added to the historical tables. The existing table identifier combined with the review identifier and the collaboration framework version identifier should act as a compound primary key for the historical table. Any data written to the Task lookup table, for example, would also be written to its historical counterpart. In this way, the data in the desired tables are never completely deleted. *Major* versions (changes from one review to the next) can be tracked by comparing the most recent collaboration framework versions from one review identifier to the next. *Minor* versions (changes that occur during a review) can be tracked by comparing collaboration framework versions that share a review identifier. The following ERD depicts a subset of historical mirror tables (prefixed *hist* for clarity) and their relationship with the review status table. Note some tables have been omitted to allow for a more straightforward presentation.



**Figure 11. Historical Entity Relationship Diagram for NOD**

In addition to historical mirror tables, it is also recommended that a historical log table be implemented. The historical log table basically maintains a running log of actions (INSERT, UPDATE, DELETE, etc.) executed against the database. A running log makes it easier for a DBA to discern what was changed in the database during a given time period. The structure of all tables including the historical log table is defined in greater detail in Appendix 2 of this document.

## NOD Stored Procedures

The NOD stored procedures developed in support of this system can be broken into the following classifications:

- Career Field check-in/check-out related procedures. These procedures enable the OCCSTDs review process by allowing the CF to “check out” and later “check in” a given Career Field (i.e. enlisted rating). Examples include:
  - *checkOutCareerField*
    - Inserts into the *luReviewStatus* table, setting the *careerFieldId*, *checkedOutDate* and *checkedOutBy* fields.
    - Outputs the NOD-generated review identifier.
  - *checkInCareerField*
    - Updates the *luReviewStatus* table, setting the *checkedInDate* and *checkedInBy* fields.
  - *undoCheckOutCareerField*
    - Updates the *luReviewStatus* table, setting the *canceledDate* and *canceledByFields*.
  - *addReviewParticipant*
    - Updates the *luReviewParticipant* and *relParticipant2Review* tables with participant data from the CF.
- Get OCCSTDs data procedures. These procedures basically return OCCSTDs data from the NOD. Examples include:
  - *getTasks*
    - Select Task data from the *luTask* and *relTask2Job* tables for a given *careerFieldId*.
  - *getJobFamilies*
    - Select *JobFamily* data from the *luJobFamily* table.
- Write OCCSTDs data procedures. The procedures are responsible for writing OCCSTDs data from the CF to the NOD. These procedures can even go so far as to prevent the modification of OCCSTDs data if the parent Career Field is not checked out. It should be noted that not all data is associated with a Career Field (and therefore an OCCSTDs review). Data that is not associated with a Career Field should not require that a review identifier is provided when modified. In order to convey this difference, the following examples include Tasks (which are associated with a Career Field) and Job Families (which are not). These procedures can be broken into 3 categories:
  - Create: Insert a completely new item into the NOD. Create procedures should return the NOD-generated identifier where applicable. Examples include:
    - *createTask*
      - Inserts given Task data into the *luTask* and *relTask2Job* tables.

- Inserts the data and review identifier into the *histLuTask* and *hisRelTask2Job* tables, to maintain a history of Task data.
- Inserts the action and pointer to historical data row(s) into the historical log table.
- Outputs the NOD-generated Task identifier.
- *createJobFamily*
  - Inserts given *JobFamily* data into the *luJobFamily* table.
  - Inserts the action into the historical log table.
  - Outputs the NOD-generated JobFamily identifier.
- *Update*: Update an existing item in the NOD. Examples include:
  - *updateTask*
    - Updates given Task data to the *luTask* and *relTask2Job* tables.
    - Inserts the data and review identifier into the *histLuTask* and *histRelTask2Job* tables, to maintain a history of Task data.
    - Inserts the action and pointer to historical data row(s) into the historical log table.
  - *updateJobFamily*
    - Updates given *JobFamily* data to the *luJobFamily* table.
    - Inserts the action into the historical log table.
- *Delete*: Delete an existing item from the NOD. For example:
  - *deleteTask*
    - Deletes Task by Task identifier from the *luTask* and *relTask2Job* tables.
    - Inserts the action into the historical log table.

## Deployment Strategy

The NOD should be deployed on a database server that meets or exceeds the minimum requirements for Microsoft SQL Server 2008 Standard Edition with Reporting Services.<sup>12</sup> Although the NOD typically should not need to support a large number of repetitive transactions or require large amounts of storage capacity, an ample amount of storage space should be available to the system. Since no data should ever be completely deleted from the system, over time the storage requirements will expand as older data is preserved in historical tables for possible rollback, reporting, and auditing purposes.

During the course of an OCCSTDs review, participants may need to review reports driven by data that is not yet finalized. To accommodate this, it may be desirable to maintain two instances of the NOD database, a *staging* instance and a *production* instance. OCCSTDs data modified during the course of a review will be written to the

---

<sup>12</sup> <http://msdn.microsoft.com/en-us/library/ms143506.aspx>

staging database first. This allows review participants to generate reports against the staging database without affecting the production reports. Once the reports have been reviewed and the associated OCCSTDs data is finalized, the data in the staging database can be replicated to the production database, effectively publishing the new OCCSTDs.

## **NOID-NOD Data Migration Strategy**

The primary challenge during data migration is the shift to a more normalized database. Once the NOD database has been created on a server and its structure is in place, all of the existing tables and data from the NOID database can be loaded into a temporary database on the same server. Having all of the data on the same server should make the data migration process easier for a DBA. The NOID data can be easily loaded into a temporary database on the NOD server via SQL Server Integration Services (SSIS).<sup>13</sup> An alternative to loading the NOID data via SSIS could be to query the data without loading it into SQL Server by using the *OPENROWSET* or *Linked Server* functionality of SQL Server.<sup>14</sup> Loading the NOID via SSIS is recommended, to simplify the queries that must be developed to facilitate data migration.

Once the complete system is fully functional, tested by users and approved for production use by NAVMAC personnel, the NOID database can be retired and archived. It is recommended to keep the NOID and any transitional structures used in the design of the new system until the new system is fully deployed in a production environment.

## **Detailed System Design: Collaboration Framework (CF)**

The Collaboration Framework (CF) streamlines the Enlisted Rating OCCSTDs review process by establishing a single meeting place for participants. The CF should provide the following:

- The ability to pull baseline OCCSTD data from the NOD at the onset of a review.
- The ability to make recommendations to create, update and delete OCCSTD data and write it back to the NOD at the conclusion of a review.
- Extensive auditing capabilities. Who changed what, when, and why?
- Structured communication between participants during the course of an OCCSTDs review.
- The ability to maintain an organized OCCSTDs document clearinghouse.

The CF should be developed utilizing Microsoft SharePoint 2010 Server as its foundation. SharePoint 2010 Server is capable of satisfying all of the above requirements, many with out-of-the-box functionality requiring only minor customizations.

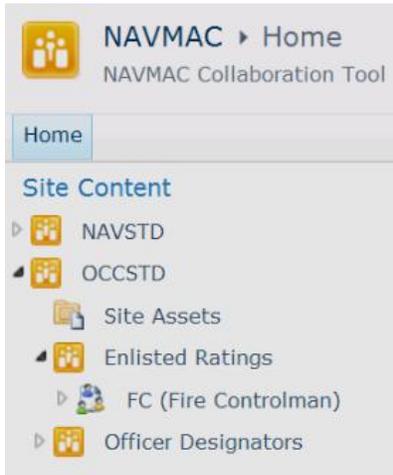
---

<sup>13</sup> More information regarding the use of SSIS to load Microsoft Access data is available at the Microsoft Developer Network (<http://msdn.microsoft.com/en-us/library/ms141209.aspx>)

<sup>14</sup> <http://msdn.microsoft.com/en-us/library/ms190312.aspx>

## Site Organization and Navigation Strategy

The initial design of the SharePoint site shall recognize that the CF could be extended in the future to support additional workforce classification elements. In order to ensure future extensibility, the SharePoint site should be organized into a site/sub-site tree structure, as shown in the following figure.



**Figure 12. SharePoint Sub-Sites**

In the site/sub-site tree structure, the lowest level of the sub-site tree consists of collaborative workspaces that are dedicated to respective Enlisted Ratings. In Figure 6, a collaborative workspace for the Enlisted Rating of FC (Fire Controlman) is displayed. Every Enlisted Rating should have its own collaborative workspace. The site tree could be structured as follows:

- NAVMAC Home
  - NAVSTD
    - Etc.
  - OCCSTD
    - Enlisted Ratings
      - AB (Collaborative Workspace)
      - AC (Collaborative Workspace)
      - AD (Collaborative Workspace)
      - AE (Collaborative Workspace)
      - Etc.
    - Officer Designators
      - Etc.
    - Navy Enlisted Classification Codes (NEC)
      - Etc.

Collaborative workspaces for all enlisted ratings shall be created prior to system deployment. Every Enlisted Rating shall have its own permanent, dedicated collaborative workspace. System administrators shall have the ability to create, modify and delete collaborative workspaces as needed. This approach provides users with the critical capability to manage documents and make minor changes to OCCSTDs data during non-review periods. Adoption of this approach does present some size and navigation challenges. SharePoint 2010 Server has some size constraints that must be noted.<sup>15</sup>

- The total number of sub-sites has a supported limit of 250,000.
- The number of sub-sites under a given parent has a supported limit of 2,000.

Internal testing by Microsoft has indicated that SharePoint performance begins to degrade as the above limits are approached. Presently there are approximately 88 enlisted ratings. Extending the subsystem design to additional classification elements could involve approximately 114 Officer Designators, 996 Officer Subspecialties, and 945 Navy Enlisted Classification Codes (NEC). Establishing a collaborative workspace for each of those classification elements appears feasible since the supported levels would not be exceeded. However, it is never appropriate to simply present users with a monolithic list of options. For example, while listing all 88 Enlisted Ratings may seem reasonable, listing all 996 subspecialties is not. When dealing with large lists, user interface tactics such as paging, filtering, and searching can help to alleviate the burden on both the server and the end user. Some potentially useful customizations to the default SharePoint navigation functionality could include:

- Prevent the left-hand navigation menu (*Tree View* or *Quick Start*) from listing a large number of sub-sites or list items.
- Treat the Enlisted Ratings sub-site as a landing page that maintains a list of all its child sub-sites (the ratings). Give users the ability to search the list by enlisted rating, or perhaps even filter the list by some other grouping such as OCCSTDs review status or job family.
- Prevent utility lists such as *Site Assets* from appearing in the navigation.
- Utilize static navigation menus that are not driven by site content.

For more information on customized navigation in SharePoint 2010, developers can review the Microsoft How-to: Customize Navigation.<sup>16</sup>

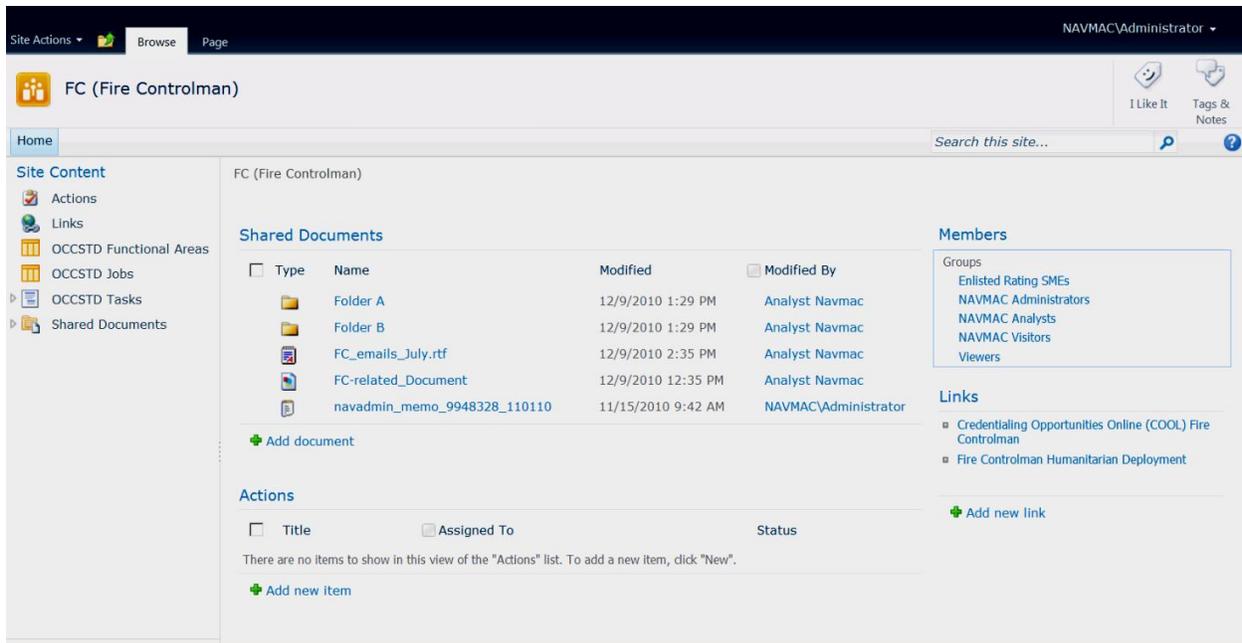
## Collaborative Workspaces

A collaborative workspace is essentially a customized sub-site, based on the standard SharePoint Team Site template. A Shared Document library, Calendar and Task list are ready to use immediately. Any default functionality determined to be unnecessary can be removed easily from the workspace. An example workspace is pictured in the following figure.

---

<sup>15</sup> <http://technet.microsoft.com/en-us/library/cc262787.aspx>

<sup>16</sup> <http://msdn.microsoft.com/en-us/library/ms558975.aspx>



**Figure 13. Collaboration Workspace Example**

The remaining functionality necessary to facilitate the OCCSTDs review process will need to be accomplished through the use of custom components (Custom Lists, Web Parts, Workflows, etc.). The design of custom components should be flexible enough to allow for the entire sub-site to be saved as a template and re-used as often as necessary during the development and maintenance of the system. The collaboration workspaces that correspond to each enlisted rating should not need to be manually constructed one-by-one. Their design should be sufficiently abstract to allow for programmatic provisioning. This approach should enable NAVMAC System Administrators to easily add new collaborative workspaces when necessary, for example upon the introduction of a new enlisted rating. To avoid duplication of work during development, the collaborative workspace template should be fully developed, tested, and approved prior to its provisioning to other enlisted ratings.

The OCCSTDs review shall largely be accomplished through the use of Custom Lists, Web Parts, and Workflows. The OCCSTDs Task list could be considered the centerpiece of OCCSTDs reviews. The OCCSTDs Task list is most efficiently implemented as a Custom List based on the SharePoint Discussion Board template. An example of the default view for a custom OCCSTDs Task list is shown in the following figure.



**Figure 14. OCCSTDs Task List Example**

Every OCCSTDs Task in the above figure is essentially the subject of its own discussion. Review participants can collaborate by participating in web-based discussions on each task. An example discussion is shown in the following figure.



**Figure 15. OCCSTDs Task Discussion Example (Flat View)**

During the prototype phase of this design, it was noted that the stock SharePoint Discussion Board does not allow users to easily make changes to task properties such as task text or functional area from the Flat or Threaded message views. The Flat and Threaded message views should be customized to allow analysts to make changes based on SME feedback without excessive navigation. By default, the user must click back up a level to the task list, check the row, and click the edit icon in the ribbon to bring up the task modification form. This can be simplified by simply adding an icon to the message

view that presents the task modification form for the parent discussion (the Task) in a modal box with just one click, instead of three.

In addition to discussing tasks, participants also should have the ability to rate each task in terms of its frequency or importance. Using the figure as an example, tasks can be rated on a thumbs-up/thumbs-down basis. Tasks with many up-votes are assumed to be accurate, while tasks with many down-votes are less so. This rating feature could help streamline the OCCSTDs review process by giving SMEs the ability to provide feedback with a single click. SMEs ratings provide other review participants with a snapshot of the overall sentiment surrounding a task/functional area/job/etc.

Every OCCSTDs Task has associated properties, such as a minimum Pay Grade, a Functional Area, and a Job or Jobs to which it applies. In SharePoint, these one-to-one and one-to-many associations are modeled as Lookup columns in a Custom List. These lookup columns reference other custom lists, effectively acting as a foreign key would in a database. The collaborative workspace should utilize the following custom lists:

<b>OCCSTDs Tasks (Custom List based on <i>Discussion List</i> Template)</b>			
<b>Column Name</b>	<b>Type</b>	<b>Allow Blank?</b>	<b>Allow Multiple?</b>
Task Text	Single line of text	No	NA
Pay Grade (PG)	Choice (Dropdown E1-E9)	No	No
Functional Area	Lookup (reference Functional Area list, Functional Area Text field)	No	No
Job(s)	Lookup (reference Job list, Job Text field)	Yes	Yes
Discussion	Multiple lines of text	Yes	NA
Rating (0-5)	Average value of all ratings	NA	NA
Number of Ratings	Number of ratings submitted	NA	NA
NOD Task ID	Number	Yes	NA
Approval Status *	Moderation Status	NA	NA
Approver Comments *	Multiple lines of text	NA	NA
Last Updated *	Date and Time	NA	NA
ID *	Counter	NA	NA
Modified *	Date and Time	NA	NA
Modified By *	Person or Group	NA	NA
Created *	Date and Time	NA	NA
Created By *	Person or Group	NA	NA
Version *	Single line of text	NA	NA
Replies *	Lookup (internal reference)	NA	NA
Folder Child Count *	Lookup (internal reference)	NA	NA

\* Read-only columns utilized by SharePoint

<b>OCCSTDs Functional Areas (Custom List)</b>			
<b>Column Name</b>	<b>Type</b>	<b>Allow Blank?</b>	<b>Allow Multiple?</b>
Functional Area Text	Single line of text	No	NA
NOD Functional Area ID	Number	Yes	NA
* Read-only columns utilized internally by SharePoint have been omitted to avoid repetition. These columns are essentially the same as those specified in the OCCSTDs Tasks list. The Replies and Folder Child Count fields would not be present as the Jobs list is not intended to be derived from a discussion list.			

<b>OCCSTDs Jobs (Custom List)</b>			
<b>Column Name</b>	<b>Type</b>	<b>Allow Blank?</b>	<b>Allow Multiple?</b>
Job Text	Single line of text	No	NA
Job Code	Single line of text	No	NA
Short Title (10)	Single line of text (max 10)	No	NA
Short Title (30)	Single line of text (max 30)	No	NA
Long Title	Single line of text	No	NA
Job Description	Multiple lines of text	Yes	NA
Pay Plan	Choice (Enlisted, Officer, Civilian)	Yes	No
Proficiency Level	Choice (A,J,M)	Yes	No
O*Net SOC Code	Lookup (reference O*Net Occupations list, O*Net SOC Code, Title, and Version fields)	Yes	No
DOD Occupation Code	Lookup (reference DOD Occupations list, DOD Occupation Code and Title fields)	Yes	No
Job Family	Lookup (reference Job Family list, Job Family Name field)	Yes	No
NOD Job ID	Number	Yes	NA
* Read-only columns utilized internally by SharePoint have been omitted to avoid repetition.			

<b>OCCSTDs Skills to Jobs (Custom List)</b>			
<b>Column Name</b>	<b>Type</b>	<b>Allow Blank?</b>	<b>Allow Multiple?</b>
Job	Lookup (reference Job list, Job Text field)	No	No
Skill	Lookup (reference Skill list, Skill Text field)	No	No
* Read-only columns utilized internally by SharePoint have been omitted to avoid repetition.			

<b>OCCSTDs Abilities to Jobs (Custom List)</b>			
<b>Column Name</b>	<b>Type</b>	<b>Allow Blank?</b>	<b>Allow Multiple?</b>
Job	Lookup (reference Job list, Job Text field)	No	No
Ability	Lookup (reference Ability list, Ability Text field)	No	No
* Read-only columns utilized internally by SharePoint have been omitted to avoid repetition.			

The following custom lists transcend individual Enlisted Ratings and can therefore be placed in one of the collaborative workspaces' parent sites (e.g. NAVMAC Home -> OCCSTDs -> Enlisted Ratings) to avoid repetition.

<b>OCCSTDs Skills (Custom List)</b>			
<b>Column Name</b>	<b>Type</b>	<b>Allow Blank?</b>	<b>Allow Multiple?</b>
Skill Text	Single line of text	No	NA
Skill Description	Multiple lines of text	Yes	NA
NOD Skill ID	Number	Yes	NA
* Read-only columns utilized internally by SharePoint have been omitted to avoid repetition.			

<b>OCCSTDs Abilities (Custom List)</b>			
<b>Column Name</b>	<b>Type</b>	<b>Allow Blank?</b>	<b>Allow Multiple?</b>
Ability Text	Single line of text	No	NA
Ability Description	Multiple lines of text	Yes	NA
NOD Ability ID	Number	Yes	NA
* Read-only columns utilized internally by SharePoint have been omitted to avoid repetition.			

<b>O*Net Occupations (Custom List)</b>			
<b>Column Name</b>	<b>Type</b>	<b>Allow Blank?</b>	<b>Allow Multiple?</b>
O*Net SOC Code	Single line of text	No	NA
O*Net Version	Single line of text	No	NA
O*Net Title	Single line of text	No	NA
O*Net Description	Multiple lines of text	Yes	NA
O*Net Job Family Code	Single line of text	Yes	NA
O*Net Job Family Title	Single line of text	Yes	NA
* Read-only columns utilized internally by SharePoint have been omitted to avoid repetition.			

<b>DOD Occupations (Custom List)</b>			
<b>Column Name</b>	<b>Type</b>	<b>Allow Blank?</b>	<b>Allow Multiple?</b>
DOD Occupation Code	Single line of text	No	NA
DOD Occupation Title	Single line of text	No	NA
DOD Occupation Area	Single line of text	Yes	NA
DOD Occupation Group	Single line of text	Yes	NA
DOD Occupation Subgroup	Single line of text	Yes	NA
* Read-only columns utilized internally by SharePoint have been omitted to avoid repetition.			

<b>Job Families (Custom List)</b>			
<b>Column Name</b>	<b>Type</b>	<b>Allow Blank?</b>	<b>Allow Multiple?</b>
Job Family Name	Single line of text	No	NA
NOD Job Family ID	Number	No	NA
* Read-only columns utilized internally by SharePoint have been omitted to avoid repetition.			

<b>Career Fields (Custom List)</b>			
<b>Column Name</b>	<b>Type</b>	<b>Allow Blank?</b>	<b>Allow Multiple?</b>
Career Field Text	Single line of text	No	NA
NOD Career Field ID	Number	No	NA
* Read-only columns utilized internally by SharePoint have been omitted to avoid repetition.			

All of the above lists should be implemented as SharePoint Custom lists and not as SharePoint External lists. External lists have a variety of limitations that make them unsuitable for use in the CF, such as the inability to be the target of a Lookup column. SharePoint Custom lists, on the other hand, provide a number of valuable features out of the box, such as version tracking and approval workflows. The use of Custom lists means that the collaboration framework must be customized to ensure that the items in these lists remain synchronized with the NOD.

In addition to the lists described above, collaborative workspaces shall feature a custom Web Part responsible for orchestrating the OCCSTDs review process. The Review Status Web Part is described in further detail in the following sections of this document. Collaborative workspaces shall also include a Report Viewer<sup>17</sup> Web Part to enable the review of system reports from the staging instance of the NOD prior to final OCCSTDs publication. Depending on the level of integration chosen for the survey application, a Survey Review Web Part may also be included in collaborative workspaces. A strategy for survey integration is discussed in more detail later in this document.

## **Collaboration Framework Interactions and Uses**

CF users can be classified into one or more of the following roles:

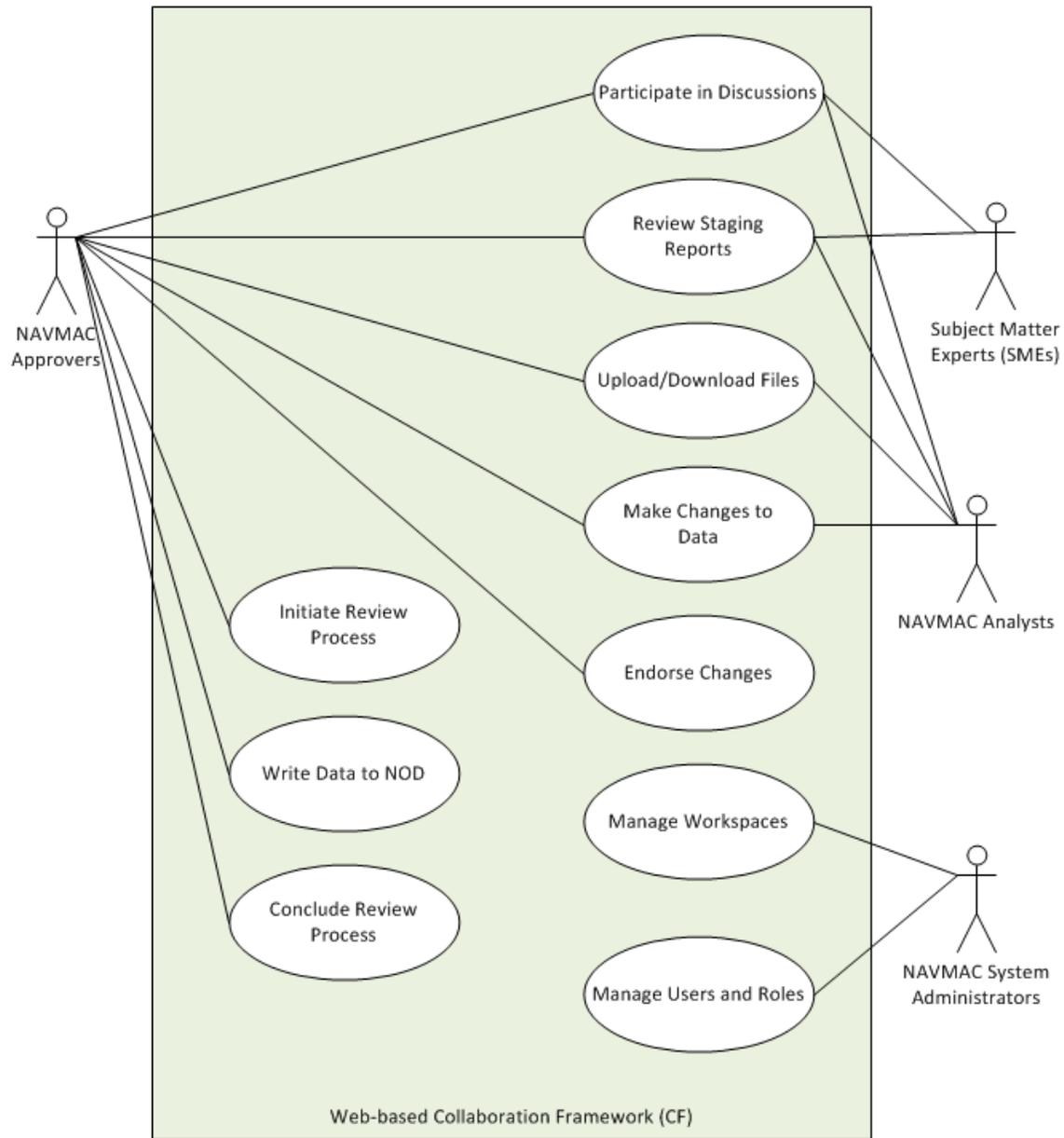
- **System Administrators:** Technical personnel responsible for the management of the CF.
- **Approvers:** Senior NAVMAC personnel with the authority to finalize the work products of an OCCSTD review (the definition of tasks, functional areas, skills, abilities).
- **Analysts:** NAVMAC personnel responsible for the development of the work products of an OCCSTD review.
- **Navy Subject Matter Experts (SMEs):** Navy personnel, external and internal to NAVMAC, responsible for assisting in the development of and providing feedback on the work products of an OCCSTD review.

Other users can be setup as needed with specific privileges as directed by NAVMAC.

The following use case diagram depicts the ways in which each of the above roles could interact with the system.

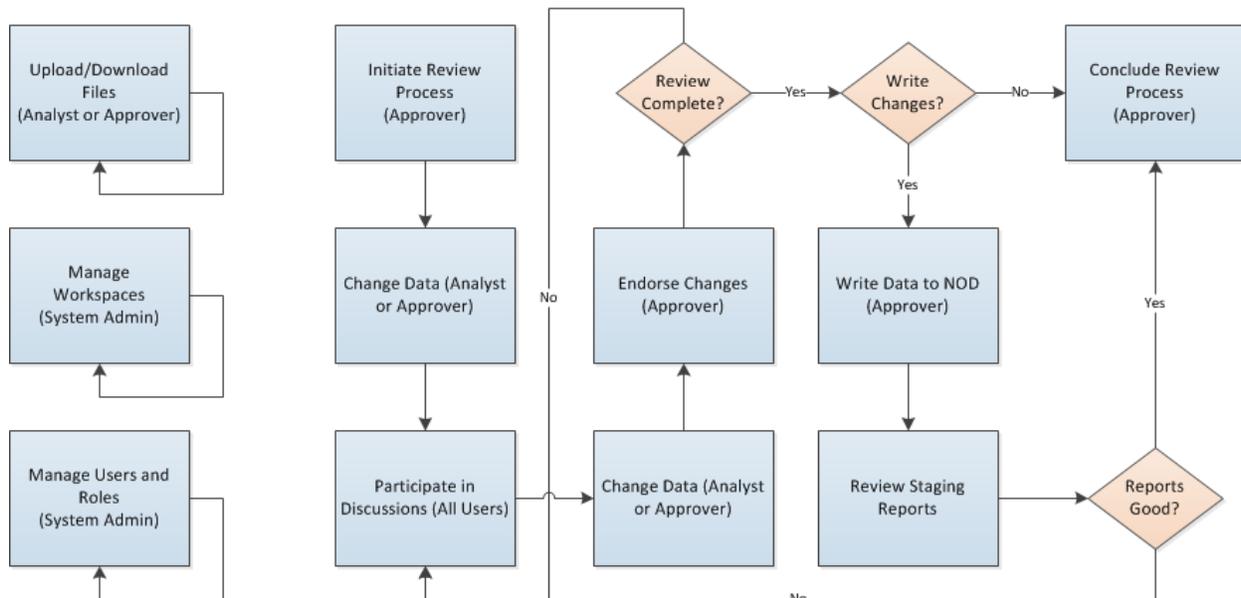
---

<sup>17</sup> <http://msdn.microsoft.com/en-us/library/ms159772.aspx>



**Figure 16. Use Case Diagram: Collaboration Framework**

The following process diagram depicts the intended flow of the above use cases.



**Figure 17. Process Diagram: Collaboration Framework**

The following table maintains an index of the CF system use cases illustrated in the above diagrams. It should be maintained and extended as the system matures. The complexity and priority fields for each use case can be used as an aid in project planning.

Use Case Index					
Use Case ID	Use Case Name	Primary Actor	Scope	Complexity	Priority
1	Manage Users and Roles	System Admin	In	Med	1
2	Manage Workspaces	System Admin	In	High	1
3	Upload and Download Files	Analyst	In	Low	1
4	Initiate Review Process	Approver	In	High	1
5	Change Data	Analyst	In	Med	1
6	Participate in Discussions	SME	In	Med	1
7	Endorse Changes	Approver	In	Med	1
8	Write Data to NOD	Approver	In	High	1
9	Review Staging Reports	Approver	In	High	1
10	Conclude Review Process	Approver	In	High	1

## Deployment Strategy

The CF should be deployed across two servers, web and database, that meet or exceed the minimum requirements for Microsoft SharePoint 2010 Server<sup>18</sup> Standard Edition. It should be noted that the physical database server hosting the NOD also could be used to host the database portion of the CF. If the email-related features of

<sup>18</sup> <http://technet.microsoft.com/en-us/library/cc262485.aspx>

SharePoint are to be leveraged, a SMTP mail server should be accessible to the web server hosting SharePoint.

## Survey Integration Strategy

During the course of an OCCSTDs review, a survey may be administered to sailors holding the Enlisted Rating under review. The survey shall be developed and deployed using the COTS product: *Questionmark Perception*. The Questionmark Web Integration Services environment (QMWISE)<sup>19</sup> provides system developers with the ability to tightly integrate the survey component of the system with the collaboration framework.

A phased approach to integration between the survey component and the collaborative framework is recommended to avoid “re-inventing the wheel.” It is assumed that the Questionmark user interface components that enable users to create and deploy surveys are mature, feature-rich products, and therefore should be leveraged. Initially, integration efforts should be minimal. As the survey process via Questionmark matures and its strengths and weaknesses become apparent, the CF integration strategy should be refined in ways that improve the process.

A possible chronological plan for integration between the CF and the survey component follows:

1. **Manual Integration:** NAVMAC personnel create/deploy surveys, associate participants, and monitor results via Questionmark applications and tools. NAVMAC personnel then shares survey results with review participants via the CF.
2. **Automated Monitoring:** NAVMAC personnel create/deploy surveys and associate participants via Questionmark applications and tools. OCCSTDs review participants monitor the survey results through the CF. The CF gets the results automatically from the Questionmark server via QMWISE services.
3. **Automated Provisioning and Monitoring:** NAVMAC personnel create/deploy surveys, associate participants, and OCCSTDs review participants monitor the survey results through the CF. The CF accomplishes all of this automatically via QMWISE services.

While manual integration may prove to be sufficient, the ability to monitor survey results via the CF could prove convenient during the course of a review. For example, survey results could indicate that a task is not core to a rating or that a task is no longer relevant. Furthermore, the ability to programmatically convert collaborative workspace lists (e.g. Tasks, Skills and Abilities) into surveys and to deploy them on the Questionmark server could also prove highly desirable.

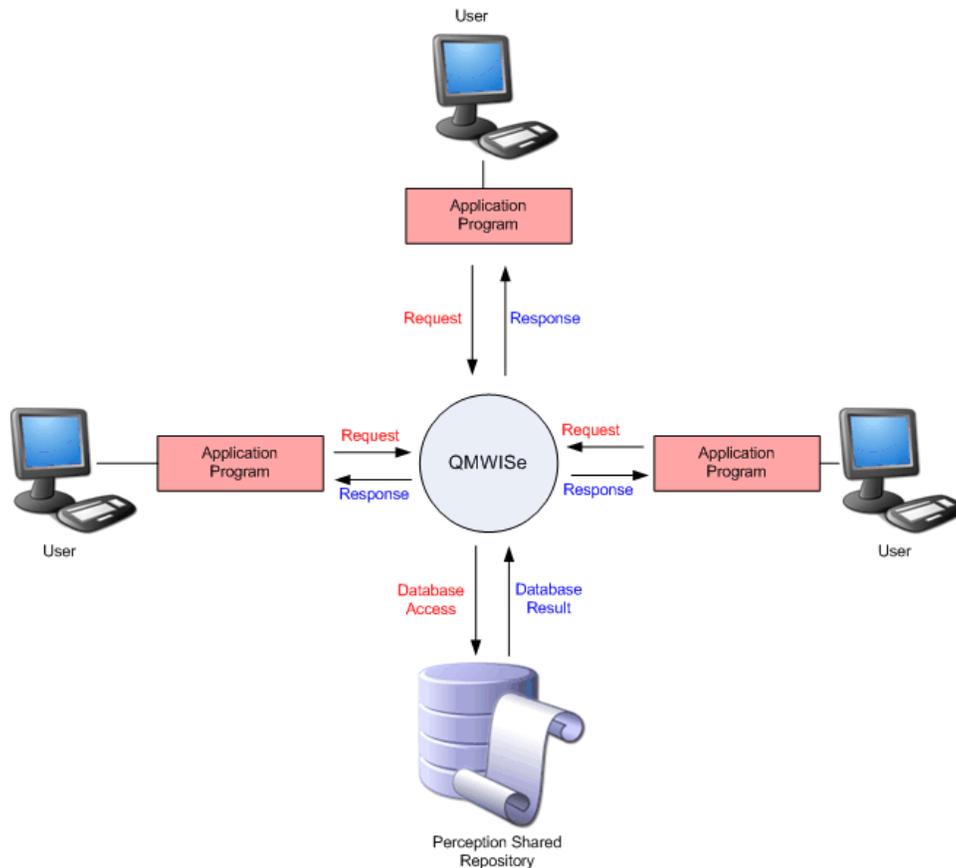
QMWISE is an Application Programming Interface or API that controls how Perception is used from an outside application using web services. The web service methods are a set of named interfaces with precise specifications that enable specific tasks, such as a report for a survey response rate to be requested of the web service. In this case QMWISE web service would be utilized within the collaborative framework to

---

<sup>19</sup> <http://www.questionmark.com/us/perception/qmwise.aspx>

request information from the survey repository in Perception. QMWISe uses open standards such as XML and Simple Object Application Protocol (SOAP) so it applies across different platforms and is compatible with emerging future technologies.

The following figure, taken from the Questionmark web site, shows the way in which QMWISe enables integration between Questionmark and external programs:



**Figure 18. QMWISe Integration**

Review participants are represented as one of the “User” symbols above, and are employing the CF as the “Application Program.” Requests for information are sent to the Questionmark Perception repository through QMWISe. To investigate survey response rate a user would generate an information request in the collaborative framework (formatted in XML and conforming to SOAP standards) and send it to QMWISe for processing. A response or fault message is returned to the user within the collaborative framework. The QMWISe API can be configured to log every request and response in support of auditing and debugging efforts.

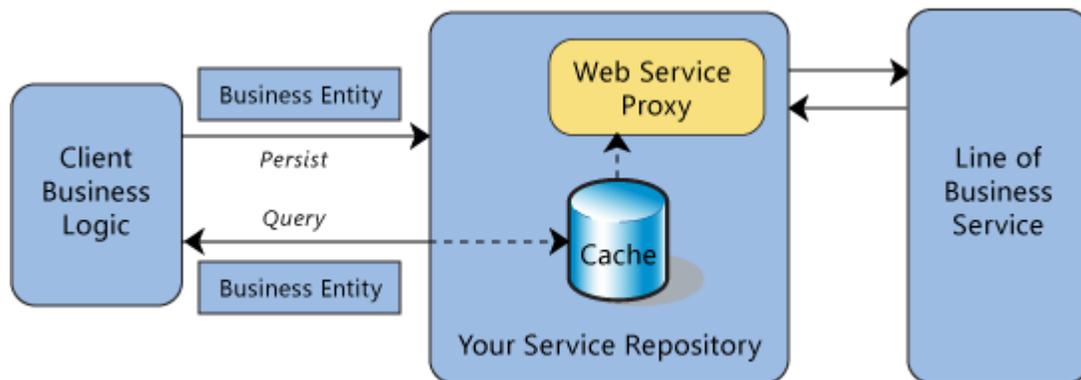
QMWISe supports the transfer of information via HTTP using command line parameters. To help overcome the cumbersome features of that protocol (i.e., using lists of data elements) XML lists can be placed in “envelops” with standard headers and standard formats and thereby enable users to follow a well-defined, standard process.

Security features of QMWISe are built-in and depend on headers for both request and response messages. The headers include a “ClientID” provided by the system administrator and a “Checksum” parameter that is generated from a string formed by concatenating the ClientID with the encrypted password for the administrator.

A collection of survey information within the Questionmark Perception database is called an Assessment. To investigate the response rate for a specific survey, a NAVMAC user would first need to associate the appropriate assessment identifier with the CF collaborative workspace. Then the results for the assessment of interest must be retrieved and presented in a coherent format. That can be accomplished by the following steps:

- Call the GetAssessmentList web service method to view all the assessments in the repository and note the identifier for the assessment of interest.
- Use the assessment identifier to call the GetResultListByAssessment web service method to return full details of results associated with the assessment (survey in the example) with that identifier.

It is highly recommended that the above steps be implemented on the SharePoint CF side through the use of the Repository design pattern. The Repository pattern, applied to web services, is illustrated in the following figure from Microsoft.



**Figure 19. Repository Pattern Applied to Web Services**

All interactions between SharePoint and Questionmark are accomplished through the use of a Survey Repository. The Survey Repository should cache data whenever possible to avoid making an excessive amount of web service calls to QMWISe. For example, it is likely that we only need to call the GetAssessmentList web service method once during a CF user’s session. The CF Survey Repository would then cache the result and return it instead of calling the web service again for the entire session, unless manually overridden by the user through a refresh request.

The integration approach described above, including the survey monitoring example, can be applied to other integration features as necessary. QMWISe provides a full set of

web service methods, allowing for the abstraction of just about any Questionmark feature.<sup>20</sup>

## Detailed System Design: Reporting Framework (RF)

As noted in the discussion on system architecture, the Reporting Framework could be considered a component of the Collaboration Framework, and should be designed if possible to support future integration. However, because of licensing, scalability, reliability, and security factors the reporting framework should be designed as a separate, exclusive component.

Key questions that must be addressed in designing the Reporting Framework are:

- What types of reports should be supported?
- What types of output report formats should be supported (e.g., Adobe PDF, Microsoft Excel, XML, HTML)?
- How do users define input sources, register the need for, and generate each of the required reports?

The reporting framework design should allow for expansion to more classification elements such as: Naval Standards (NAVSTDS), Navy Enlisted Classification Codes (NEC), and Navy Officer Designators and Subspecialties. The Navy Job Analysis Management (NJAM) initiative seeks to produce a comprehensive system for classification elements using the OCCSTDs development process as a basis. With that focus, the NJAM Project Description<sup>21</sup> provides insight to guide the reporting framework design.

### Types of Reports Required

During OCCSTDs development process, SMEs and NAVMAC analysts will need to extract and share information. For example, at the onset of an OCCSTDs review a baseline list of tasks is requested from the NOD. Such “in-process” data needs are addressed in the Collaborative Framework and are not considered part of the Reporting Framework in this document. However, as the OCCSTDs Implementation Step unfolds, several types of reports will be required. The types of reports needed to support the production and publication phases within the Implementation Step are:

- **Production:** Review and survey results are to be translated into new task statements and Job Codes, KSA’s, DoD crosswalks, and other applicable Occupational Classification elements are incorporated. The draft OCCSTDs must then be reviewed and submitted for approval using the following types of reports:

---

<sup>20</sup> Additional information regarding the QMWISE API is available at <http://developer.questionmark.com/home/file.php/2/qmwiseapiguide/Default.htm>

<sup>21</sup> Jackson, Michele; Powell, Johnny; and Carrasco, Juan; *Navy Job Analysis Management Project Description*, NAVMAC, January 2010.

- Lists of validated tasks, sorted by Functional Area, based on user defined criteria.
- Reports that append job codes, job descriptions, scope, DoD crosswalk information, and other occupational classification elements to tasks.
- Summary of NAVMAC internal and external comments of changes to tasks.
- Lists of OCCSTD elements added, deleted, or modified.
- Historical reports of occupational classification standard elements.
- Audit history summarizing changes to classification elements.
- **Publication:** OCCSTDs approved in the production phase are to be published in NAVPERS 18068F Volume I, Navy Enlisted Occupational Standards, Manual of Navy Enlisted Manpower and Personnel Classifications and Occupational Standards. OCCSTDs must also be made available in various formats for stakeholder purposes.
  - Published Reports
    - OCCSTDs
    - Job Code Crosswalk
    - Career Field to Job Family
    - Task Commonality
    - Ability Crosswalk
  - Standard reports of study results at predefined levels of detail such as:
    - Pay grade
    - Rating
    - Job code
    - Functional Area

## Report Formats

The Reporting Framework should support the review and approval actions of the OCCSTDs development process. That process involves circulating drafts within NAVMAC, creating and distributing approval packages, and promulgating transmittal and approval letters. To accommodate the various documents, reports, and enclosures, a wide range of common file formats should be supported.

Common file types that should be supported within the reporting framework include:

- Text Files: Microsoft formats (.doc, .docx), plain text formats (.txt), mail messages (.msg), and log files (.log).

- Data Files: .csv, .dat, .efx, .ppt, .pptx, and .sdf.
- Spreadsheet Files: .wks, .xls, .xlsx.
- Database Files: .accdb, .db, .mdb, .pdb, .sql
- Web Files: .asp, .htm, .html, .xhtml, .cer, .csr

Other file formats used by developers and system administrators may also be required.

## Generating Reports

There are three broad roles for individuals likely to be involved in generating reports:

- **System Administrators:** Within the reporting framework system administrators will need to establish roles for authorized users and assign one or more roles to those users. Furthermore, role-based access levels may apply to the entire system or be specific to certain data items and reports. An advanced feature for which system administrators may need to create protocols is “data-driven subscriptions.” Data-driven subscriptions permit queries to external data sources that are controlled by a subscription at specified run times. For example, NAVMAC may wish to subscribe to DoD job classification systems or the Department of Labor (DoL) Occupational Classification Network (O\*NET).
- **Report Designers:** Based on knowledge of roles and authentication rules, data connection capability for both internal and external data sources, and query procedures, report designers will develop a set of standard reports to support OCCSTDs development. Report designers will need to define the look and feel of reports and determine how users select the parameters that restrict the data to be included in each report. Additionally, report designers may also need to provide means to explore each cell within a report in more detail (drill-down) and to export data to other applications.
- **Report Consumers:** Report consumers will be charged with helping accomplish the OCCSTDs process by identifying the type of report to be used, adding or grouping parameters to define the exact information to be displayed, and generating the report. Users may also examine selected data within the report in more detail or link the report information to other data sources. For example, prior to OCCSTDs publication, a user may generate an OCCSTD document for a specific rating. Report users may also need the capability to produce ad hoc reports in which they apply query techniques to examine special issues that may arise during OCCSTDs production and publication. Any reports generated by report users should be exportable to other applications.

To identify an effective reporting services tool the following criteria are assessed:

- Can the required reports be designed using the tools provided?
- What input would a user need to provide in order for the system to generate the report?

- What output is reported back to the user?
- What components of the reporting framework and overall system are involved in using input and producing the desired output?

## Microsoft SQL Server Reporting Services

SQL Server Reporting Services is a server-based reporting platform that provides a wide range of ready-to-use tools and services to create, deploy, and manage reports. Reporting capability can be extended and customized through programming features.

SQL Server Reporting Services work within the Microsoft Visual Studio environment and are fully integrated with SQL Server tools and components. Interactive, tabular, graphical, or free-form reports from relational, multidimensional, or XML-based data sources are possible. Furthermore, scheduling report processing, accessing reports on-demand, and creating ad hoc reports is supported using a variety of viewing formats. Exportation of report information to other applications and subscription to external published reports and data sources is also possible. Reports can be viewed over a Web-based connection, as part of a Microsoft Windows application, or SharePoint site.

Specific features in SQL Server Reporting Services align with the requirements for an OCCSTDs development system.

The Reporting Services Configuration Tool would enable system administrators to specify service accounts, create or upgrade the report server database, modify the connection properties, set virtual directories, manage encryption keys, and configure the report server for unattended report processing and e-mail report delivery. Those features would support report designers and report users within NAVMAC as well as allow for data-driven subscriptions.

Report designers could work with system administrators to take advantage of the Report Manager.<sup>22</sup> Report Manager is a web-based tool with which to set permissions, manage subscriptions and schedules, and work with reports and models. Report Manager can also be used to view reports.

Use of Report Manager requires sufficient permissions. Report Manager provides different pages and options depending on the role assignments of the current user that can be useful for the different types of users involved in the OCCSTDs process such as NAVMAC analysts and SMEs from warfare and resource sponsors. Users with permissions to view reports can access links to open reports.

Report Designer and Model Designer are two design tools available within SQL Server Reporting Services Business Intelligence Development Studio. The design surfaces in the tools include tabbed windows, wizards, and menus used to access report and model authoring features. The design tools become available when a Report Server Project, a Report Server Wizard, or a Report Model Project template is chosen. Once

---

<sup>22</sup> Report server administrators can use Management Studio to manage a report server alongside other SQL Server component servers. Management Studio provides almost identical functionality as Report Manager, but with additional support for managing other server types in the same management workspace.

reports are designed they would be available to report users according to the permissions set in the Reporting Services Configuration Tool.

Report users can take advantage of the Report Builder to create ad hoc reports that use published models as a data source. Reports generated with Report Builder can be saved to a report server and exported to other applications.

## Report Tracking with SQL Server Reporting Services

SQL Server Reporting Services (SRS) is fairly robust with respect to logging features. SRS maintains an execution log of all reports, from which management reports can be generated. Management reports can detail the type of reports requested, when, by whom and with what parameters. Sample Server Management Reports for SRS 2008<sup>23</sup> are bundled with the product and should be reviewed by developers who desire to add management reporting functionality to the system.

## Planning a Report With SQL Server Reporting Services

The following example shows how issues associated with report type, report format, and report generation affect subsystem development and how system administrators, report designers, and report users could work together to address them in a specific design using SQL Server Reporting Services.

Suppose as part of the production stage and prior to submitting an OCCSTD document for review, NAVMAC analysts see a need to determine if tasks within an enlisted rating under review are also identified for other jobs in other career fields. That information is currently available in the Task Commonality Report.

Suppose further, NAVMAC analysts decide to continue with the existing report title “Task Commonality Report.” They, in conjunction with system administrators and report designers, could proceed with enhancing the utility of the report by answering the following questions, based upon Microsoft’s Reporting Services “Planning a Report”.<sup>24</sup>

- **In what format do you want the report to appear?**

Reports can be produced online in a browser such as Report Manager and be exported to other formats such as Excel, Word, or PDF. The final form your report takes is an important consideration because not all features are available in all export formats. The design team could decide to make the Task Commonality Report available online in HTML format and also in Word, PDF, and Excel format for export.

- **What structure do you want to use to present the data in the report?**

Presentation choices are tabular, matrix (similar to a cross-tab or PivotTable report), chart, free-form structures, or any combination of those options. The present text columns are selected for all formats except Excel which will use the

---

<sup>23</sup><http://msftsrprodsamples.codeplex.com/wikipage?title=SS2008!Server%20Management%20Sample%20Reports&referringTitle=Home>

<sup>24</sup> <http://technet.microsoft.com/en-us/library/dd220520.aspx>

identically named columns with one record for each job entry. The Excel format better supports sorts and counts that may be useful in the review.

- **What do you want your report to look like?**

Report Builder provides many report items that can be added to reports to make it easier to read, highlight key information, and help the audience navigate the report. By establishing the appearance, the design team can determine whether items such as text boxes, rectangles, images, and lines are desired. The design team also considers whether to show or hide items, add a document map, include drill-down reports, sub-reports, or link to other reports. In the initial enhancement the design team determines no additional appearance effects are needed and the current text columns are effective. However, it is desirable to link the “Job Code” field to reports of other tasks for the code.

- **What data do you want your readers to see? Should the data or format be filtered for different audiences?**

The scope of the report could be made specific to users, locations, or to a particular time period. Report data may be filtered using parameters so only desired information is retrieved and displayed. For the enhanced Task Commonality Report the design team decides to give users the option of viewing all information or filtering by “Career Fields” under the control of resource or warfare sponsors.

- **Do you need to create your own calculations?**

SQL Server Reporting Services supports creation of calculated fields. That feature is useful if the data source and datasets do not contain the exact fields needed in the report. For example, it may be necessary to calculate a performance metric from basic data. Expressions are also available for conditional formatting and other advanced features. No calculations or conditional formatting is needed in the Task Commonality Report.

- **Do you want to hide report items initially?**

It is possible “hide” report items, including data regions, groups and columns, when the report is first run. That feature is helpful for initially presenting a summary table, and then allowing drill down into the detailed data. That capability is not needed in the Task Commonality Report.

- **How are you going to deliver your report?**

The Task Commonality Report must be saved to a local computer for continued work. However, since the report is to be shared it must also be saved to a report server that is configured in native mode or a report server in SharePoint integrated mode. Saving it to a server lets others run it whenever they want to. In the future, system administrator will consider establishing a subscription to the report or arrange automated e-mail delivery of the report.

## Report Viewer

The Report Viewer can add full-featured reports to custom applications. Reports may contain tabular, aggregated, and multidimensional data. Report Viewer controls are provided so that you can process and display the report in your application.<sup>25</sup> The Report Viewer is also available as a SharePoint Web Part,<sup>26</sup> easing integration between the RF and the SharePoint-based CF.

The Report Viewer control works by combining user input, data from a data source, and a report definition to produce a custom report. The following code snippet shows a custom web form with a Report Viewer control added to it.

```
<%@ Page Language="C#" AutoEventWireup="true" CodeFile="Default.aspx.cs" Inherits="Default" %>
<%@ Register assembly="Microsoft.ReportViewer.WebForms, Version=9.0.0.0, Culture=neutral,
PublicKeyToken=b03f5f7f11d50a3a" namespace="Microsoft.Reporting.WebForms" tagprefix="rsweb" %>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" >
<head runat="server">
  <title>Report Viewer Sample</title>
</head>
<body>
  <form id="form1" runat="server">
    <div>
      <rsweb:ReportViewer ID="ReportViewerSample" runat="server" Font-Names="Verdana"
        Font-Size="8pt" Height="400px" Width="992px">
        <LocalReport ReportPath="Report1.rdlc"></LocalReport>
      </rsweb:ReportViewer>
    </div>
  </form>
</body>
</html>
```

---

<sup>25</sup> [http://msdn.microsoft.com/en-us/library/ms251671\(v=VS.100\).aspx](http://msdn.microsoft.com/en-us/library/ms251671(v=VS.100).aspx)

<sup>26</sup> <http://msdn.microsoft.com/en-us/library/ms159772.aspx>

The following code snippet shows the code-behind for the above web form, it handles the binding of the Report Via to a data source.

```
public partial class _Default : System.Web.UI.Page
{
    protected void Page_Load(object sender, EventArgs e)
    {
        ReportViewer1.ProcessingMode = ProcessingMode.Local;

        ReportDataSource rds;
        DataTable tbl = FillStrength();

        rds = new ReportDataSource("DataSetReport usp Report2", tbl);
        ReportViewer1.LocalReport.DataSources.Add(rds);

        ReportViewer1.LocalReport.Refresh();
    }

    protected DataTable FillStrength()
    {
        DataTable tblData = new DataTable();
        string myConnectionString =
ConfigurationManager.ConnectionStrings["NODConnectionString"].ConnectionString;
        using (SqlConnection connection = new SqlConnection(myConnectionString))
        {
            SqlCommand cmd = new SqlCommand("dbo.usp_Report2", connection);

            cmd.CommandType = CommandType.StoredProcedure;
            SqlDataAdapter myAdapter = new SqlDataAdapter();
            myAdapter.SelectCommand = cmd;
            myAdapter.Fill(tblData);
        }

        return tblData;
    }
}
```

## Summary

The primary goal of this system design is to enhance the existing OCCSTDs development process by introducing standardized, streamlined, and less complicated procedures to obtain occupational data and track recommendations. We have presented a system design that we believe fulfills this goal. There are many aspects of this design that may be elaborated upon or modified. As NAVMAC's processes and procedures evolve, care should be taken to work closely with NAVMAC personnel to verify that their processes and job responsibilities are being represented and supported. The choice behind leveraging Microsoft products was largely based on the Navy's current use of them as well as their ease of use and customization features.

The commercial products mentioned in this system design are listed below. Costs are not referenced. Costs are based on how the system will be deployed. In addition, Microsoft has established various partnerships with the DoD which may be leveraged to reduce licensing costs. Once the system deployment details such as hosting have been decided upon, a more accurate cost estimate can be determined. The minimum requirements for the system are as follows.

- Production Environment Software Requirements
  - Microsoft SharePoint 2010 Foundation (for web server)
  - Microsoft SQL Server 2008 R2 Standard Edition (for database server)
  - Microsoft SQL Server Reporting Services (for database server); free, bundled with SQL Server
  - Microsoft Windows Server 2008 R2 (2 licenses, 1 OS for web server and 1 for database server)
- Production Environment Manpower Requirements: The following roles can be filled by part-time employees, or even by a single part-time employee with the right skill set.
  - Database Administrator (DBA): Responsible for handling day-to-day database maintenance activities. Activities could include the occasional execution of ad-hoc queries, performing database backups, monitoring reports, troubleshooting, etc.
  - SharePoint Administrator: Responsible for handling the day-to-day SharePoint Server maintenance activities. Activities could include environment updates, issue resolution, site creation, user training, backup, restore, and performance analysis.

During development of the system, and more importantly once the system has been deployed to the production environment, all development work should be done in a development environment separate from the production environment. This allows for the development of new features (e.g. the addition of support for NAVSTDs to the system) without impacting users of the production system. Once a new feature has been fully tested and accepted, it can be deployed to the production environment at minimal

inconvenience to users of the system. Ideally, the development environment should mirror the production environment as closely as possible. This can be accomplished through virtualization to reduce the number of physical servers required. Microsoft Developer Network (MSDN) subscriptions can prove extremely useful when setting up the development environment. A MSDN subscription basically allows for unlimited use of Microsoft software provided it is for development or testing purposes.

- Development Environment Software Requirements
  - Microsoft SharePoint 2010 Foundation (for web server)
  - Microsoft SQL Server 2008 R2 Standard Edition (for database server)
  - Microsoft SQL Server Reporting Services (for database server); free, bundled with SQL Server
  - Microsoft Windows Server 2008 R2 (2 instances. 1 for web and 1 for database)
  - Microsoft Visual Studio 2010 Professional (to support development)
  - Microsoft SharePoint 2010 Designer (free)
- Development Environment Manpower Requirements
  - Database Administrator (DBA) skilled in:
    - MS-SQL / SQL Server
    - MS Reporting Services
    - SQL Database Security
    - Stored Procedures
  - Junior-Mid Software Engineer(s) (1 or 2) skilled in:
    - Microsoft .NET
    - SharePoint a plus
  - Senior Software Engineer (Team Lead) skilled in:
    - Microsoft.NET
    - SharePoint Design and Development
  - Tester skilled in:
    - Dependent on testing tools available, bare minimal, a resource who can manually interact/work with the application to verify the system is working properly in response to the requirements

More information regarding the hardware requirements for MS SharePoint 2010 is available at <http://technet.microsoft.com/en-us/library/cc262485.aspx>. If the physical database server will also host the NOD and the Survey component data store, ensure that the memory, storage and processing power is in excess of what is specified for SharePoint alone. Likewise for the physical web server, if it will host the CF, RF and Survey component, its specifications should exceed the hardware requirements for SharePoint alone. Ideally, the CF, RF and Survey tool would each have a dedicated physical (or virtual) web server.

# **APPENDIX A: NOID Data Elements**

(NOTE: Many of the data element descriptions are being revised.)

Field Name	Description
AbilityDescription	The description of the Ability Text
AbilityText	Enduring attributes of the individual that influence performance and enable the performance of tasks.
AbilityTextID	Unique code identifying the Ability; an alpha-character followed by digits(s).
CareerFieldDescription	Job Family Description (e.g., Aerographer's Mate; Explosive Ordnance Disposal; etc.)
CareerFieldText	Job Family Name; AG, EOD, etc. Crosswalk was based on accomplishment codes (not used in table). Career field test is rating, GS series or NOBC
DODOccupationArea	The DoD Occupational Area as defined by the Dept of Defense. Data obtained from DoD website <a href="https://www.dmdc.osd.mil/owa/odb/odb">https://www.dmdc.osd.mil/owa/odb/odb</a> .
DODOccupationCode	The DoD Occupational Code as defined by the Dept of Defense. Data obtained from DoD website <a href="https://www.dmdc.osd.mil/owa/odb/odb">https://www.dmdc.osd.mil/owa/odb/odb</a> .
DODOccupationGroup	The DoD Occupational Group as defined by the Dept of Defense. Data obtained from DoD website <a href="https://www.dmdc.osd.mil/owa/odb/odb">https://www.dmdc.osd.mil/owa/odb/odb</a> .
DODOccupationSubgroup	The DoD Occupational Subgroup as defined by the Dept of Defense. Data obtained from DoD website <a href="https://www.dmdc.osd.mil/owa/odb/odb">https://www.dmdc.osd.mil/owa/odb/odb</a> .
DODOccupationTitle	The DoD Occupational Title as defined by the Dept of Defense. Data obtained from DoD website <a href="https://www.dmdc.osd.mil/owa/odb/odb">https://www.dmdc.osd.mil/owa/odb/odb</a> .
EffectiveEndDate	Effective end date for the archiving of data.
EffectiveStartDate	Effective start date for the data.
FunctionalAreaID	An alpha-character used to identify the Navy Functional Area within a job. JS is Job Specific, NS is Navy Standard, NE is NEC
FunctionalAreaText	The title of the Functional Area assigned to that job.
JobDescription	Brief description of what the job encompasses.
JobCode	Six-digit system generated code.
JobText	Name of the job – AG – Oceanographic Forecaster. Associated to the enlisted rating and similar to the Job Long Title.
NEC_Additional_Guidance	Adhoc NEC data associated to the work.
NOCCCode	Navy Occupational Code – initially developed from the concatenation of the O*NET SOC code and the JobFamilyID.
O*NETJobFamilyCode	The JobFamilyCode as defined by O*NET website.

Field Name	Description
O*NETJobFamilyName	The JobFamilyName as defined by O*NET website.
O*NET-SOC2006Code	
O*NET-SOC2006Title	The SOC code and title as defined by O*NET website.
OccSTDCode	The OCCSTDs code associated by concatenating the FunctionalAreaID and the TaskTypeID for CORE.
OccSTDPaygrade	The OCCSTDs associated to the paygrade as determined by Survey
PayPlan	Identification category, i.e. Enlisted, Officer, or Civilian.
ProficiencyLevel	An alpha-character that identifies Apprentice, Journeyman, and/or Master (AJM) level.
ShortTitle (10 characters)	10 character abbreviation of long title
ShortTitle (30 characters)	30 character abbreviation of long title
SkillDescription	Description of skill text
SkillText	Developed capacities that facilitate learning.
TaskText	Task statement
TaskTypeCode	Unique identification code for a task type.
TaskTypeID	Unique standardized ID for the task type.

**APPENDIX B:  
Occupational Standard Feedback Form**

Name:			Rate/Rating:	
email:			Command:	
Phone:			DSN:	

New Task (Yes/No)	Task ID#	Paygrade

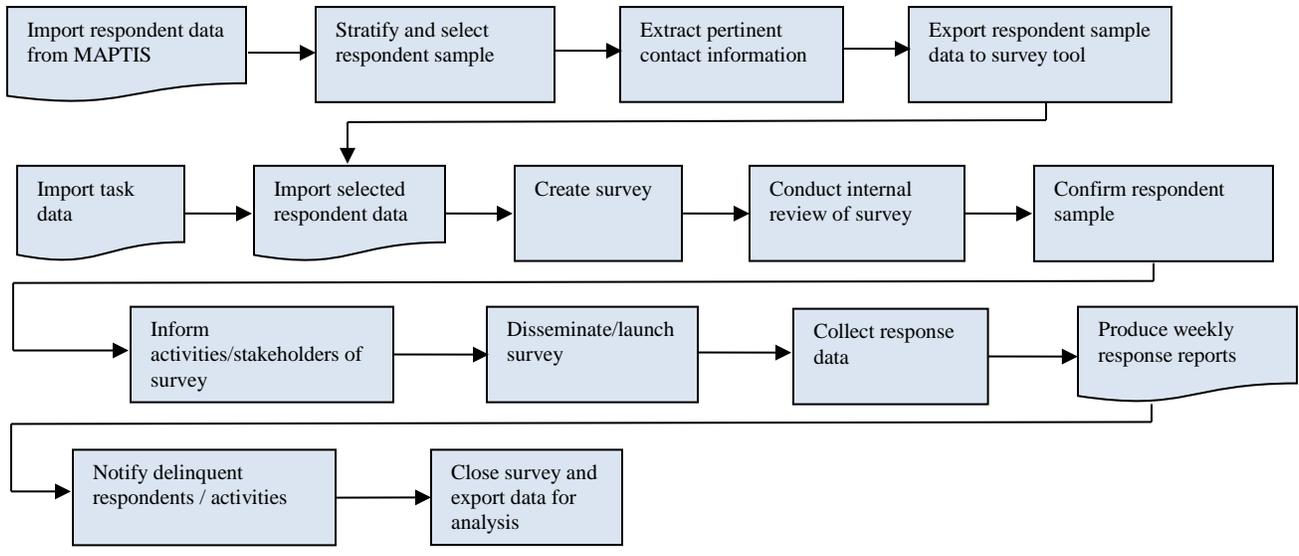
Task or other OCCSTD element (as currently is):
Comment and/or Recommended Change:
Justification:

27

---

<sup>27</sup> The form repeats for however many new tasks personnel need to enter.

# **APPENDIX C: Task Survey Process**



## **Distribution**

AIR UNIVERSITY LIBRARY  
ARMY RESEARCH INSTITUTE LIBRARY  
ARMY WAR COLLEGE LIBRARY  
CENTER FOR NAVAL ANALYSES LIBRARY  
HUMAN RESOURCES DIRECTORATE TECHNICAL LIBRARY  
JOINT FORCES STAFF COLLEGE LIBRARY  
MARINE CORPS UNIVERSITY LIBRARIES  
NATIONAL DEFENSE UNIVERSITY LIBRARY  
NAVAL HEALTH RESEARCH CENTER WILKINS BIOMEDICAL LIBRARY  
NAVAL POSTGRADUATE SCHOOL DUDLEY KNOX LIBRARY  
NAVAL RESEARCH LABORATORY RUTH HOOKER RESEARCH LIBRARY  
NAVAL WAR COLLEGE LIBRARY  
NPRST SPISHOCK LIBRARY  
OFFICE OF NAVAL RESEARCH (CODE 34)  
PENTAGON LIBRARY  
USAIR FORCE ACADEMY LIBRARY  
US COAST GUARD ACADEMY LIBRARY  
US MERCHANT MARINE ACADEMY BLAND LIBRARY  
US MILITARY ACADEMY AT WEST POINT LIBRARY  
US NAVAL ACADEMY NIMITZ LIBRARY