

Document Control Number: eTIMS3-SDD-R0-CO-(U)
November 2008

SOFTWARE DESIGN DOCUMENT

For the

Electronic Tooling Information Management System (eTIMS) 3.0

Contract Number N00383-04-D-900H, 0006
CDRL Sequence No: A0004

Prepared for:
Navy eTIMS Program Manager
700 Robbins Ave
Philadelphia, PA 19111-5098
Code 071.09

Prepared by:
Lexes Associates, Inc.
401 Park Drive
Warner Robins, GA 31088

DISTRIBUTION D: Distribution authorized to DOD and DOD Contractors only; (Determined on 15 October 1992). Other requests shall be referred to the Naval Inventory Control Point – Philadelphia.

TABLE OF CONTENTS

SECTION 1 - SCOPE.....	1
1.1 IDENTIFICATION.....	1
1.2 SYSTEM OVERVIEW.....	1
1.3 DOCUMENT OVERVIEW	2
SECTION 2 - REFERENCED DOCUMENTS.....	3
SECTION 3 - PRELIMINARY DESIGN.....	4
3.1 CSCI OVERVIEW	4
3.1.1 Online System Overview	4
3.1.2 CSCI Architecture.....	5
3.1.3 CSCI Equipment Environment.....	5
3.2 CSCI SOFTWARE ENVIRONMENT	6
3.2.1 Client Work-Station Operating System.....	6
3.2.2 Client Application Software.....	6
3.2.3 Web Server Software	6
3.2.4 Firewall Security Software.....	6
3.2.5 Database Software	6
3.3 SYSTEM STATES AND MODES	6
3.4 MEMORY AND PROCESSING TIME ALLOCATION	7
SECTION 4 - DETAILED DESIGN	8
4.1. ETIMS SYSTEM OVERVIEW	8
4.2. ETIMS SEARCH SUBSYSTEM.....	8
4.3. ETIMS TOOL SUBSYSTEM.....	16
4.4. ETIMS PART/STE SYSTEMS SUBSYSTEM	21
4.5. ETIMS STE SYSTEM SUBSYSTEM.....	26
4.6. ETIMS A-LIST SUBSYSTEM.....	26
4.7. ETIMS C-LIST SUBSYSTEM.....	28
4.8. ETIMS RECEIVE TOOLS SUBSYSTEM	30
4.9. ETIMS SHIP TOOLS SUBSYSTEM	33
4.10. ETIMS REPORTS SUBSYSTEM	36
4.11. ETIMS APPLICATION ADMINISTRATION SUBSYSTEM	37
4.12. ETIMS SYSTEM ADMINISTRATION SUBSYSTEM	40
4.13. ETIMS DISPOSITION DOCUMENT	40
4.14. ETIMS DATA REFRESH DOCUMENT.....	67
SECTION 5 - COMPUTER SOFTWARE CONFIGURATION ITEM (CSCI) DATA.....	90
5.1. CSCI DATABASE DESCRIPTION	90
5.2. FAILURE CONTINGENCIES	90

FIGURES

Figure 1 eTIMS System Diagram	4
Figure 2 eTIMS System Use Case.....	8
Figure 3 Search Subsystem Functional Flow Diagram	11
Figure 4 Search Use Case	12
Figure 5: Search Use Case Option	13
Figure 6 Search Results Use Case	14
Figure 7 Search Activity Diagram	15
Figure 8 Tooling Subsystem Functional Flow Diagram	17
Figure 9 Tool Movement Functional Flow Diagram.....	18
Figure 10 Tool Use Case	19

Figure 11 Tool Dup Use Case.....	20
Figure 12 Part Subsystem Functional Flow Diagram	21
Figure 13 Part Subsystem Use Case	22
Figure 14 Maintain Part Use Case	23
Figure 15 Maintain Part Number Activity Diagram	24
Figure 16 Add Part Number Activity Diagram.....	25
Figure 17 STE System Subsystem Functional Flow Diagram.....	26
Figure 18: A-List Subsystem Functional Flow	27
Figure 19: C-List Functional Flow.....	28
Figure 20: Amend C-List Functional Flow	29
Figure 21: Release C-List Functional Flow	30
Figure 22: Receive Tooling Functional Flow	31
Figure 23: Receive Tooling Use Case.....	32
Figure 24: Receive Subsystem Sequence Diagram.....	33
Figure 25: Ship Subsystem Functional Flow	34
Figure 26: Ship Subsytem Sequence Diagram	35
Figure 27 Reports Subsystem Functional Flow Diagram	36
Figure 28 Application Administration Subsystem Functional Flow Diagram	38
Figure 29: Support Functional Flow	39
Figure 30 System Administration Subsystem Functional Flow Diagram	40
Figure 1 Disposition Process Overview.....	41
Figure 5 SIMPLE MODEL PROCESSING	47
Figure 8 BOM EXPLOSION	51
Figure 9 Tool dup disposition processing.....	52
Figure 10 Tool-id processing.....	53
Figure 11 TOOL CODE PROCESSING	53
Figure 12 MANUAL REVIEW FLAGS	55
Figure 15 FIVE YEAR HISTORY DESIGN FLOWCHART	61
Figure 16 COPY DMIL CODES.....	64
Figure 17 DMIL PRIORITY TABLE	64

TABLES

Table 1 Search Sub-function Documents	16
Table 2 Tool/Tool Dup Sub-function Documents	21
Table 3 Part Sub-function Documents	25
Table 4 STE Sub-function Documents.....	26
Table 5: A-List Subfunction Documents.....	27
Table 6: C-List Subfunction Documents.....	30
Table 7: Receive Subsystem Documents	33
Table 8: Ship Subsystem Documents	35
Table 9 Reports Sub-function Documents	36
Table 10 Application Administration Sub-function Documents.....	40
Table 11 System Administration Sub-function Documents	40

SECTION1 - SCOPE

1.1 Identification

This Software Design Document (SDD) describes the design for the Computer Software Configuration Item (CSCI) identified as the Electronic Tooling Information Management System (eTIMS) V3.0 supporting the Program Management Office's Production Shutdown Automated Information System (AIS). The SDD provides:

- **The system requirements detailing the eTIMS CSCI. This document will identify the requirements to be satisfied and will serve as a basis for mutual understanding between the user and developer;**
- **Information on performance requirements, preliminary design considerations and user impacts;**
- **Information on system hardware architecture;**
- **A functional baseline for the system to be developed;**
- **A software development methodology including requirements, design, implementation and deployment phases; and**
- **A basis for development of system tests.**

1.2 System Overview

The electronic Tooling Information Management System (eTIMS) Automated Information System (AIS) was developed to support the Production Shutdown for the USAF F-15, USAF F-16 and multiple U.S. Navy aircraft. As part of the production shutdown responsibilities, the Program Office must make disposition determinations on the government owned production special tooling/test equipment to ensure capability is retained to support future programs, while not incurring the added expense of storing excess tooling. To meet these responsibilities, the Program Offices require an automated information system. The system will be designed, built and implemented will provide the capability to match a Part Number to its required tooling and permit the inventory management of retained ST/STE.

The Program Offices are responsible for integrated weapon system support of DOD aircraft and its subsystems from acquisition to retirement. This responsibility includes the management of postproduction programs to provide for the support of DOD aircraft, aircraft systems and programs throughout its operational life. Given current operational

requirements, production and spare parts tooling will be needed for aircraft modification crash damage, spare parts and special project requirements well into the twenty-first century. To meet individual program goals, eTIMS was designed to perform multiple functions. First, large quantities of data are drawn from the manufacturing and logistics systems of defense contractors such as Boeing, Lockheed Martin, Northrop Grumman and their major subcontractors. This manufacturing data is then combined with supply and requirements data from government support organizations. Algorithms are then developed to identify and retain the necessary tooling required to meet defined production and spares requirements. The system then provides complete inventory management for all retained tooling and test equipment along with the complete part breakdown data with the tooling required for manufacture of each part. ETIMS gives program managers unique capability to support the weapon system throughout its operational life. ETIMS also provides an audit trail of disposition decisions which provides program managers and contracting officers a record of the disposition of all Government Special Tooling (ST).

1.3 Document Overview

This document describes the Electronic Tooling Information Management System (eTIMS) CSCI. The architecture, functional allocation, functional control and data flow, and detailed design are described. The document also details inputs, data dictionary definitions, processing, and outputs.

SECTION 2 - REFERENCED DOCUMENTS

The project references that apply to the history and development of the eTIMS CSCI are listed below:

- **Department of Defense (DOD), Military Standard, DI-MCCR-80012A, Software Design Document Data Item Description, dated February 29, 1988.**
- **Statements of Work for US Air Force/Navy Program Offices to develop an automated Tooling Information Management System.**
- **Functional Description (FD) for the Navy Tooling Information Management System Automated Information System (AIS), eTIMS-SDD-RO-CO-(U), dated 20 September 1999.**

SECTION 3 - PRELIMINARY DESIGN

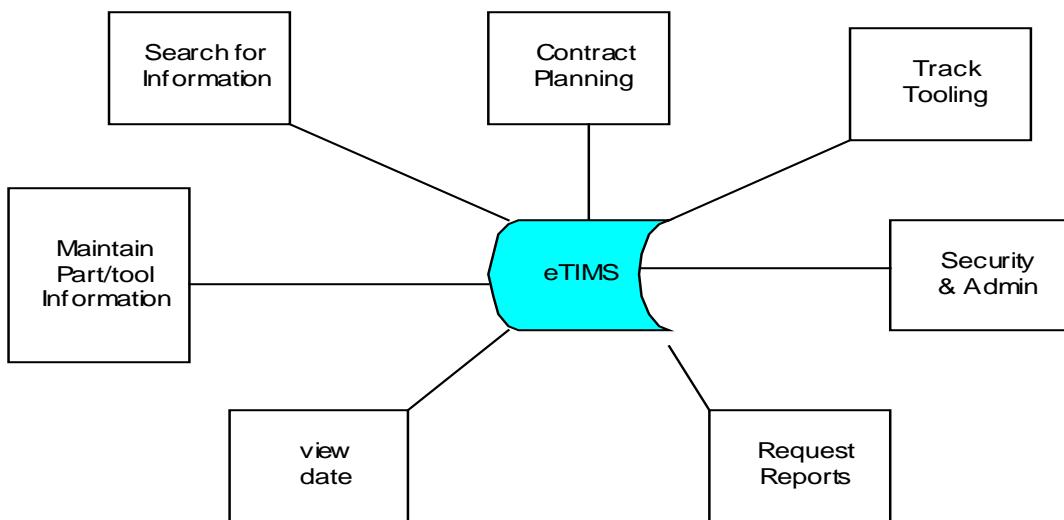
3.1 CSCI Overview

ETIMS allows for parts and tooling data to be *Weapon System specific*. Each defined Weapon System will have its own Part Numbers and tools, its own Tooling Disposition Rules and its own tooling formats. If a Weapon System has separately defined STE, it will have its own STE sub-system (Special Tooling Equipment).

ETIMS also allows for those instances when a Part Number, or a tool or a piece of STE is used on more than one Weapon Systems. In those rare cases the part/tool/STE will be available to each Weapon System that needs it.

Online User Access into a Weapon System's data is controlled by a "System Administrator" and is granted only after the proper form has been filled-out, submitted, and approved. A one-time User Id and password must be issued by the System. The user will associate a DOD CAC or ECA certificate when logging into the system the first time. After this initial logon, only PKI level III certificate logon will be permitted. A user can have access into more than one Weapon System, and each weapon system that the user has access into will have its own level of accessibility assigned.

3.1.1 Online System Overview



707338

Figure 1 eTIMS System Diagram

The system consists of programs that are the user's window into ETIMS data. They allow the user to:

- Search for Part Numbers, tooling and STE using a wide variety of search arguments, including the ability to combine search arguments.
- Update resident part, tooling and STE data.
- View Disposition results.
- Plan for tooling to be offered for use on government contracts.
- Mark tooling and/or STE as it is received into storage and/or sent out for use on government contracts.
- Control security and maintain Admin tables that are used in a variety of background functions.
- Automatically notify the appropriate users about tooling shipments.
- Report on the use of tooling.
- Create ad hoc tooling reports.
- Save tooling query results to external files.
- In selected weapon systems, link to tooling drawings.

3.1.2 CSCI Architecture

The eTIMS CSCI will be developed as web application. The web application architecture is comprised of (1) the application software that resides on the web server, (2) web server, and (3) the database server. The users will access the application via the secured access on Internet, through a security firewall and then to the web server. The web server will manage all communication between the application and the database server.

The web architecture offers the best design for security, connectivity, scalability and maintainability. The benefits of this configuration include:

- **Unlimited access from any location with internet access and a web browser**
- **No client installation of software necessary**
- **Easier deployments of enhancements and new capabilities**
- **Database information and application software objects are secure**
- **Supports object inheritance**
- **Isolates the database**
- **Enhanced performance by caching software objects and data on web server**
- **Development environment is tried proven technology**

3.1.3 CSCI Equipment Environment

The user's work-station will be personal computers (PC) must have Internet Explorer 7.0

or greater, Excel 2003 or greater, and must allow JavaScript execution. The user workstations will need secured access to the Internet (HTTPS). eTIMS will be protected by a Sidewinder® firewall appliance. The eTIMS will use Oracle® 10G for data persistence.

3.2 CSCI Software Environment

The eTIMS CSCI includes the software required to manage the database, the software required to develop the application, the web server software for managing all communication between the application software and the database management system and the software required to manage the firewall security requirements.

3.2.1 Client Work-Station Operating System

The eTIMS CSCI client work-stations will run a Microsoft Window® XP or greater.

3.2.2 Client Application Software

The application software for the eTIMS CSCI will be Internet Explorer 7.0 and Excel.

3.2.3 Web Server Software

The web server software for the eTIMS CSCI will be IIS 6.0 or greater.

3.2.4 Firewall Security Software

The firewall security server software for the eTIMS CSCI will be Sidewinder® Firewall Security Server V 7.

3.2.5 Database Software

The database management system (DBMS) software for the eTIMS CSCI includes Oracle® 10G running on a Microsoft ® Windows 2003 operating system.

3.3 System States and Modes

The accuracy and validity of the part-to-tool relationships are directly related to the accuracy of data provided by the prime contractor. Disposition recommendations are based upon a part usage relationship. For full disposition to be performed, the tool must tie to the part that it makes. If this relationship does not exist, then only limited disposition can be performed using tooling data, such as tool code, only. The accuracy and validity of the contracting usage data in the eTIMS is directly related to the diligence of the eTIMS Program Managers. Shipping and location data can only be as accurate as the data entered by the Tooling Monitor/Program Managers or the storage sites and

packing/shipping import files. ETIMS will assist in this effort by thoroughly validating all data input for format and proper range of values. Data failing this test will be rejected. Drop down selection boxes and radio buttons will be used where possible to preclude invalid data entry.

The eTIMS, as currently defined, will not directly interface with or directly receive data from any other automated system. The system will receive periodic imports from automated data systems from the prime contractor, Department of Defense, Defense Logistics Agency and the U.S. Air Force/Navy, which will be loaded into the system.

Data integrity will be protected by limiting invalid user access to the system and by validating on-line data changes. Access to the system is limited to only those users with valid CAC/ECA certificates associated to the user ids. The eTIMS system will verify format and value ranges for all inputs. Data not meeting the format or value criteria will be rejected.

3.4 Memory and Processing Time Allocation

The speed at which the data is retrieved from the database for display on the user's workstation is contingent on the reliability and speed of the user's internet access and routing prior to accessing the eTIMS T-1 node on the internet. Information entered on-line by the user will be immediately updated in the databases. Reports will be printed when requested by the user. An appropriate error message will be immediately displayed when eTIMS data validation logic identifies any error condition.

The application and database servers are more than sufficient for development, deployment and management of the eTIMS application and database. The eTIMS Oracle® 10G Server database will be configured with a 16 gigabyte of RAM on the server. This memory capacity will allow sufficient space to accommodate the access and management of all part/tooling data. The selected server is also capable of being upgraded in both storage and processing capacity should future system requirements dictate.

SECTION 4 - DETAILED DESIGN

4.1. eTIMS System Overview

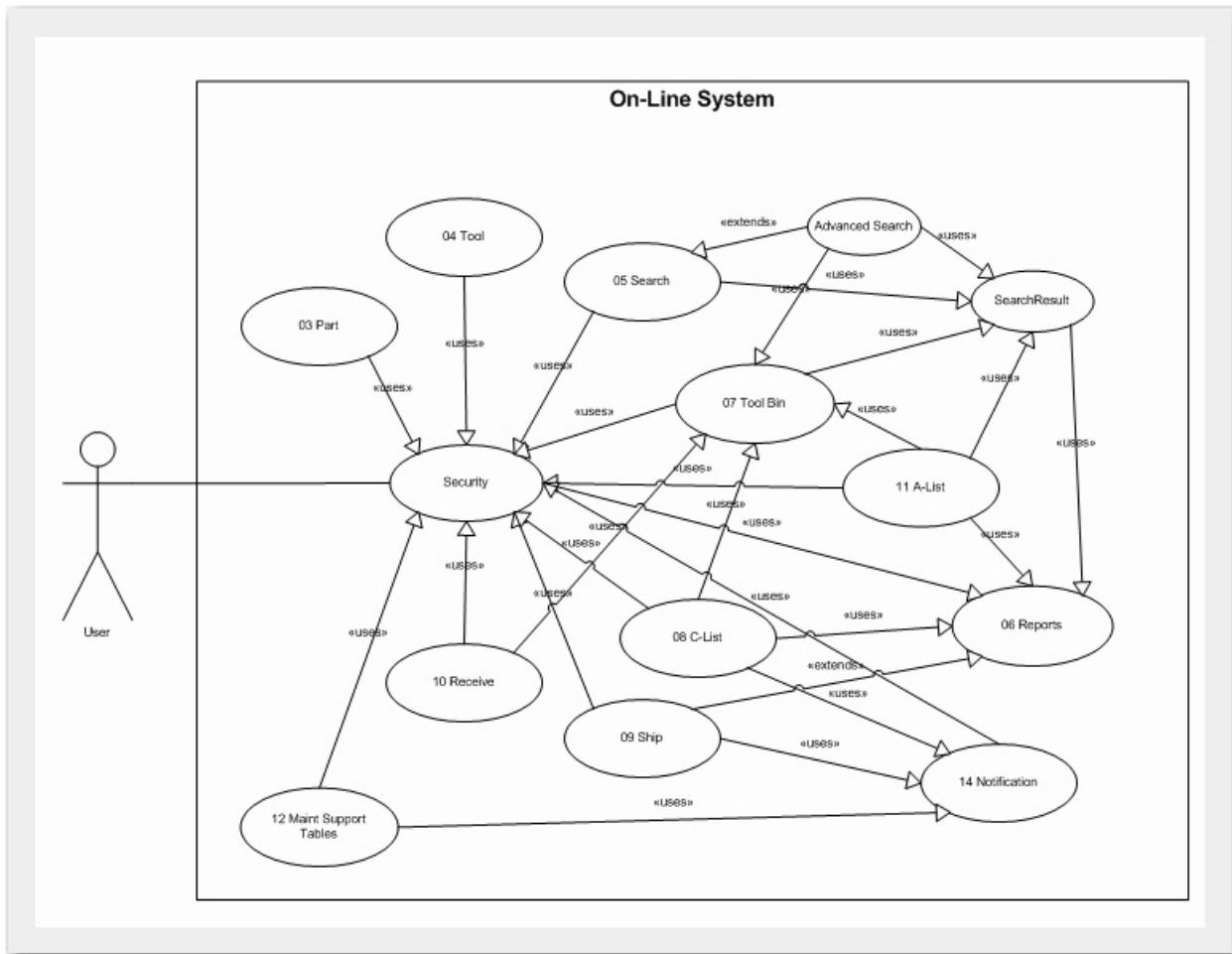
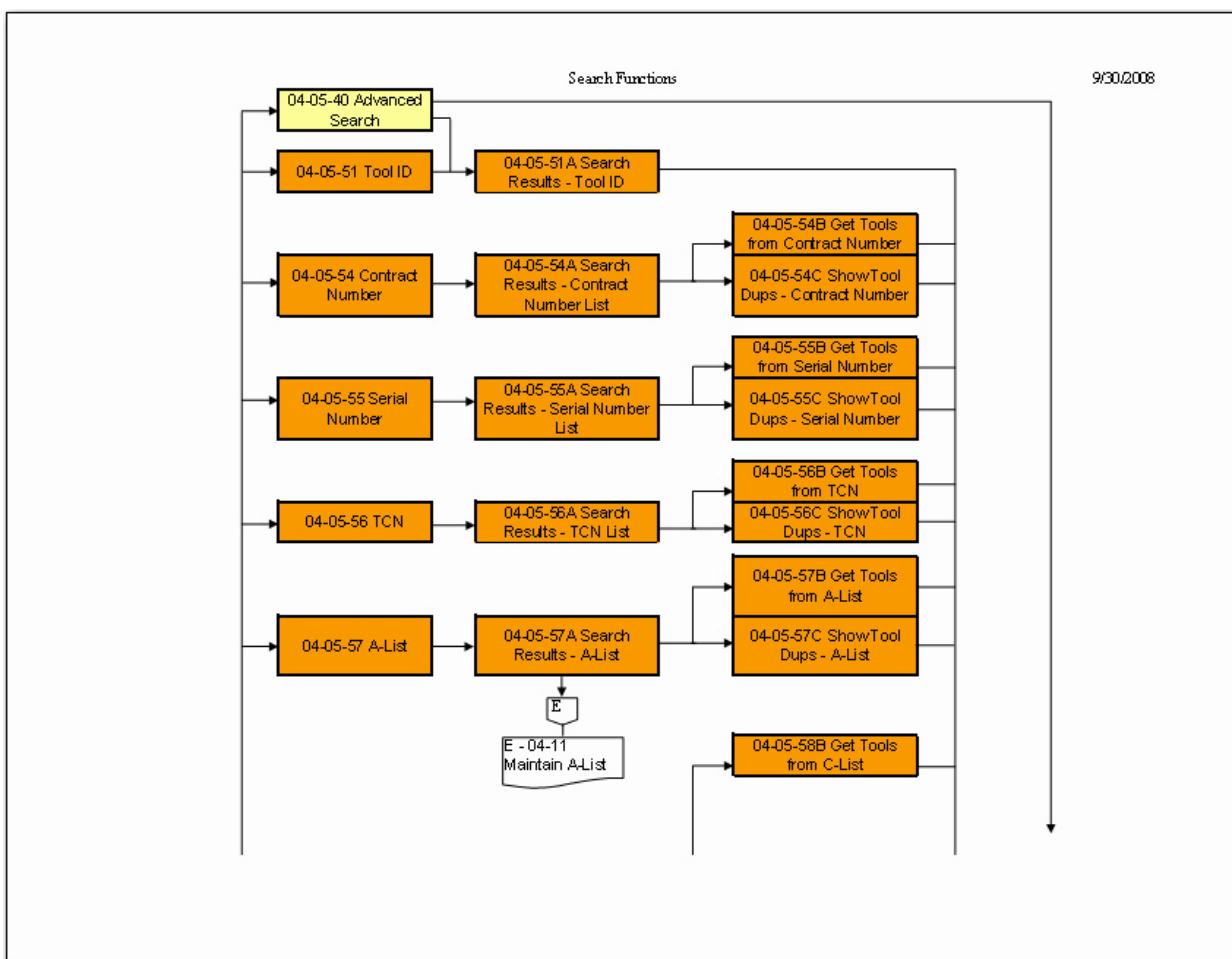
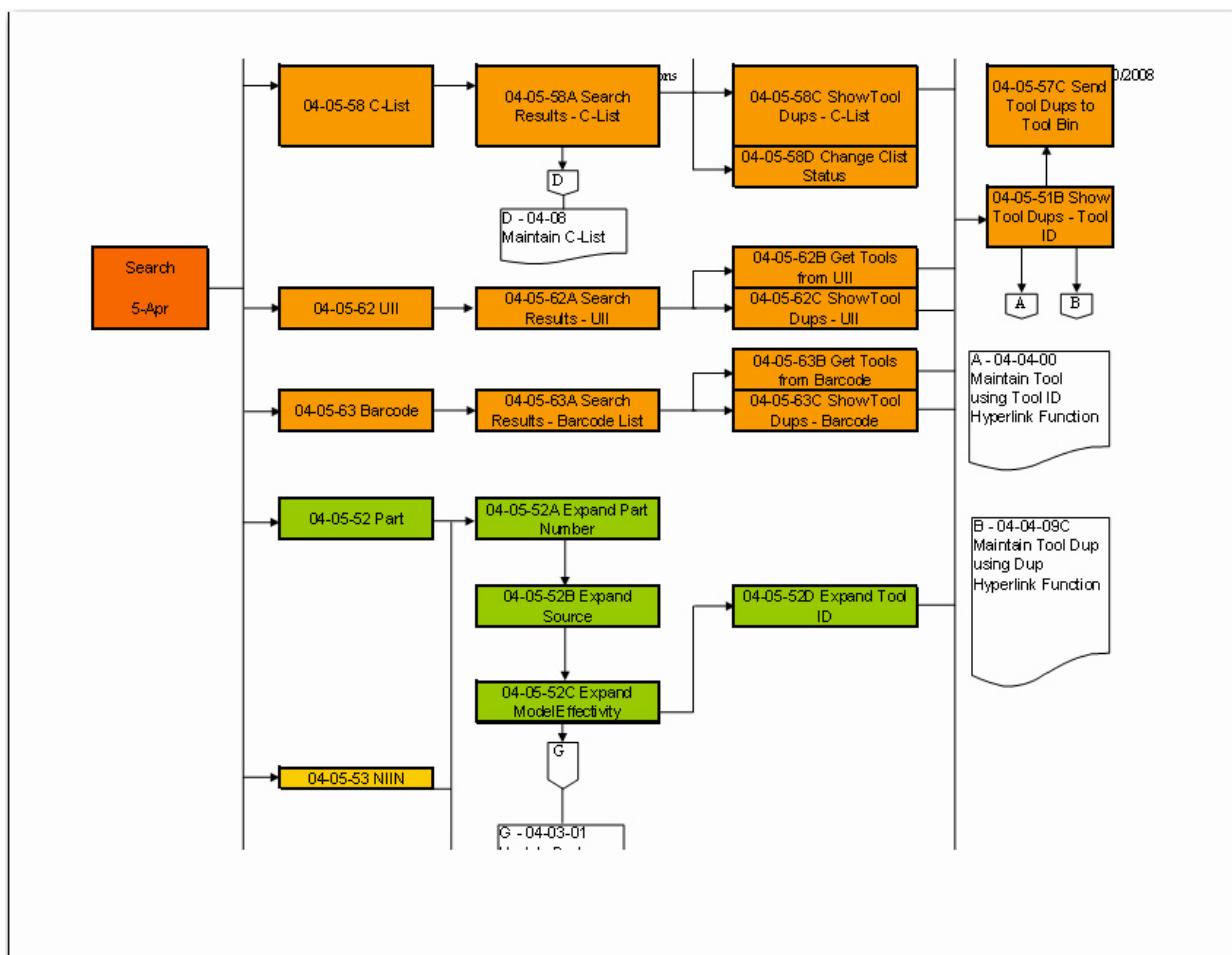


Figure 2 eTIMS System Use Case

4.2. eTIMS Search Subsystem

The Search Subsystem comprises all the methods for retrieving data from the eTIMS database. There are thirteen retrieval methods in search. The goal of the search module is to narrow down the list of tools to only those that the user requires.





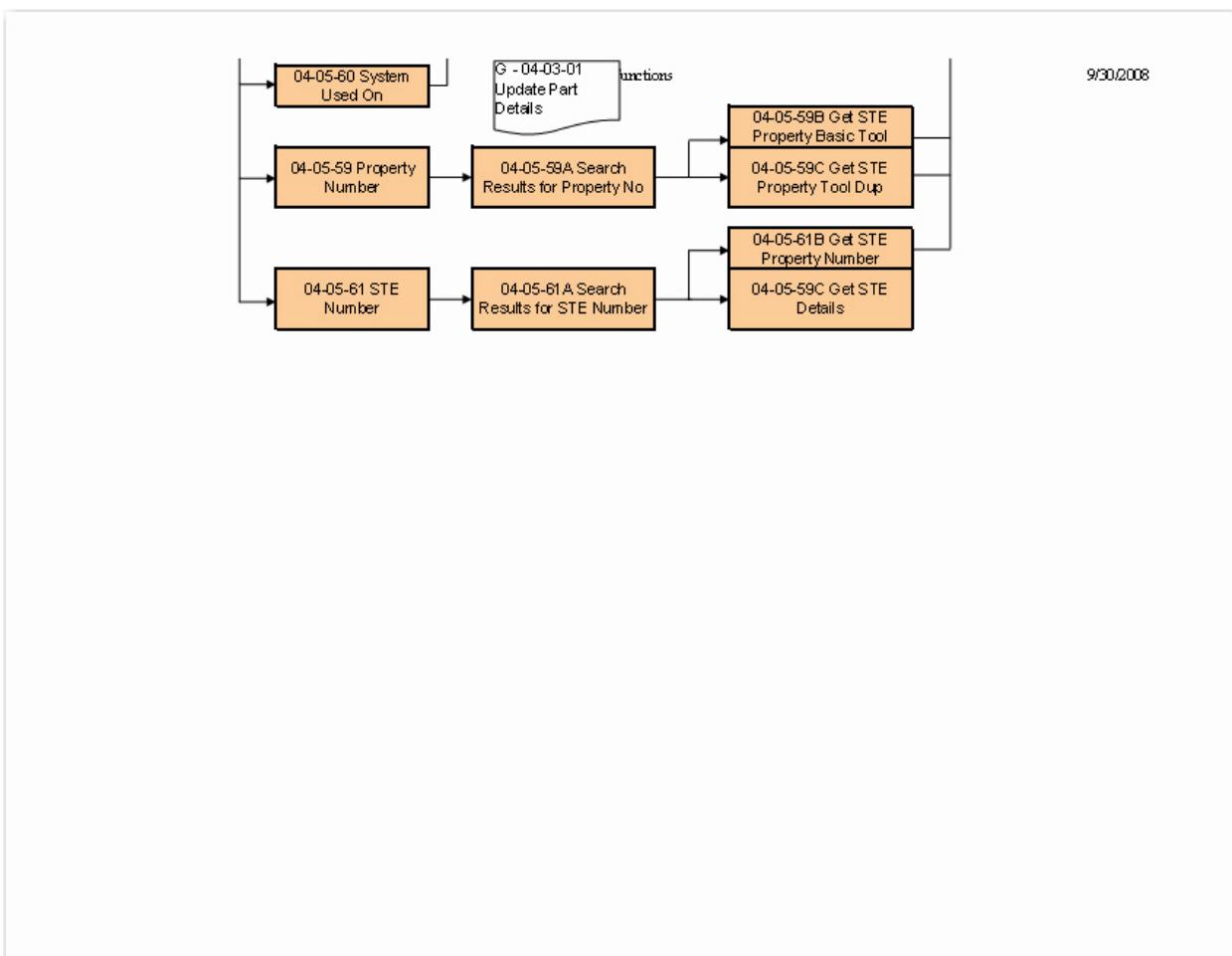


Figure 3 Search Subsystem Functional Flow Diagram

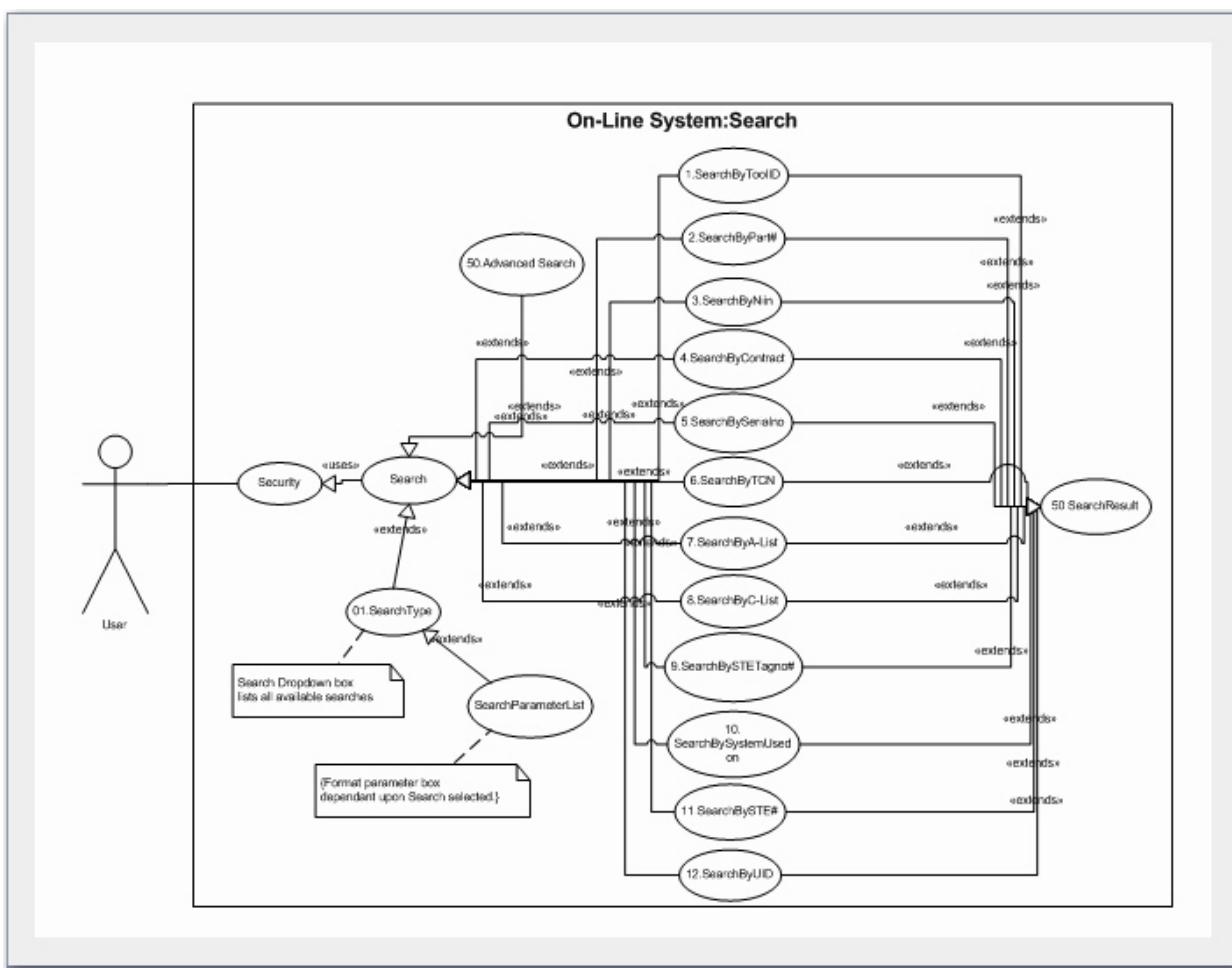


Figure 4 Search Use Case

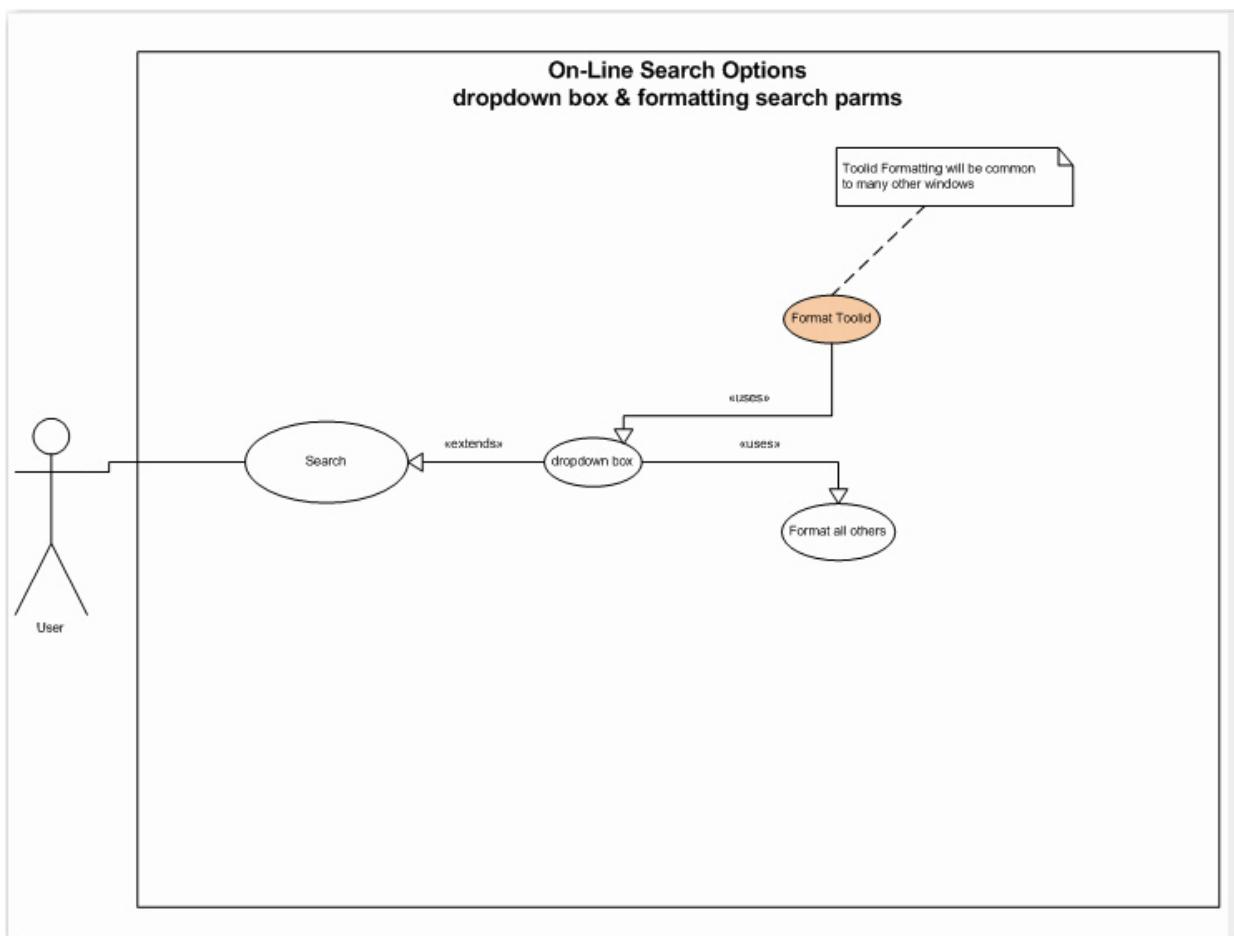


Figure 5: Search Use Case Option

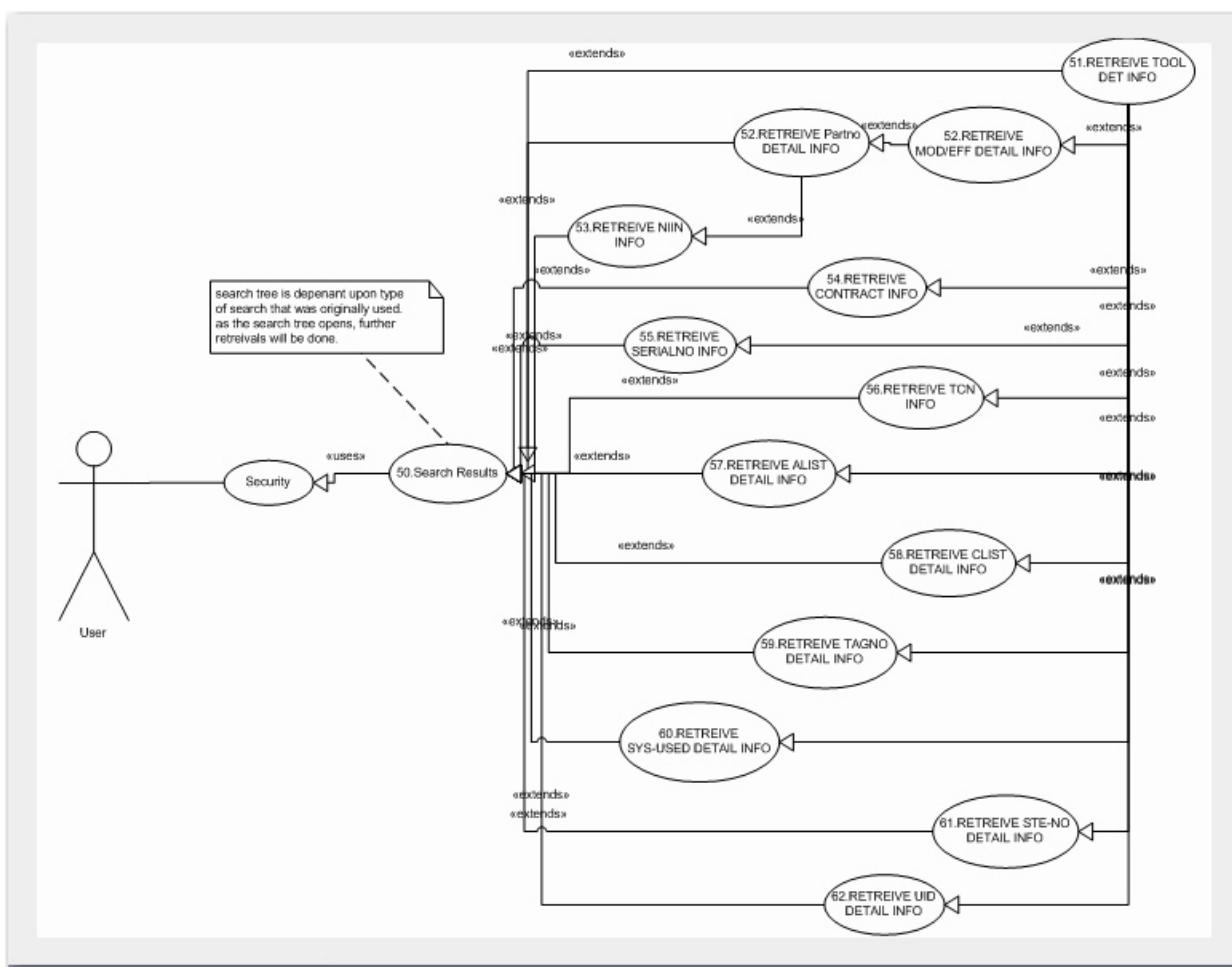


Figure 6 Search Results Use Case

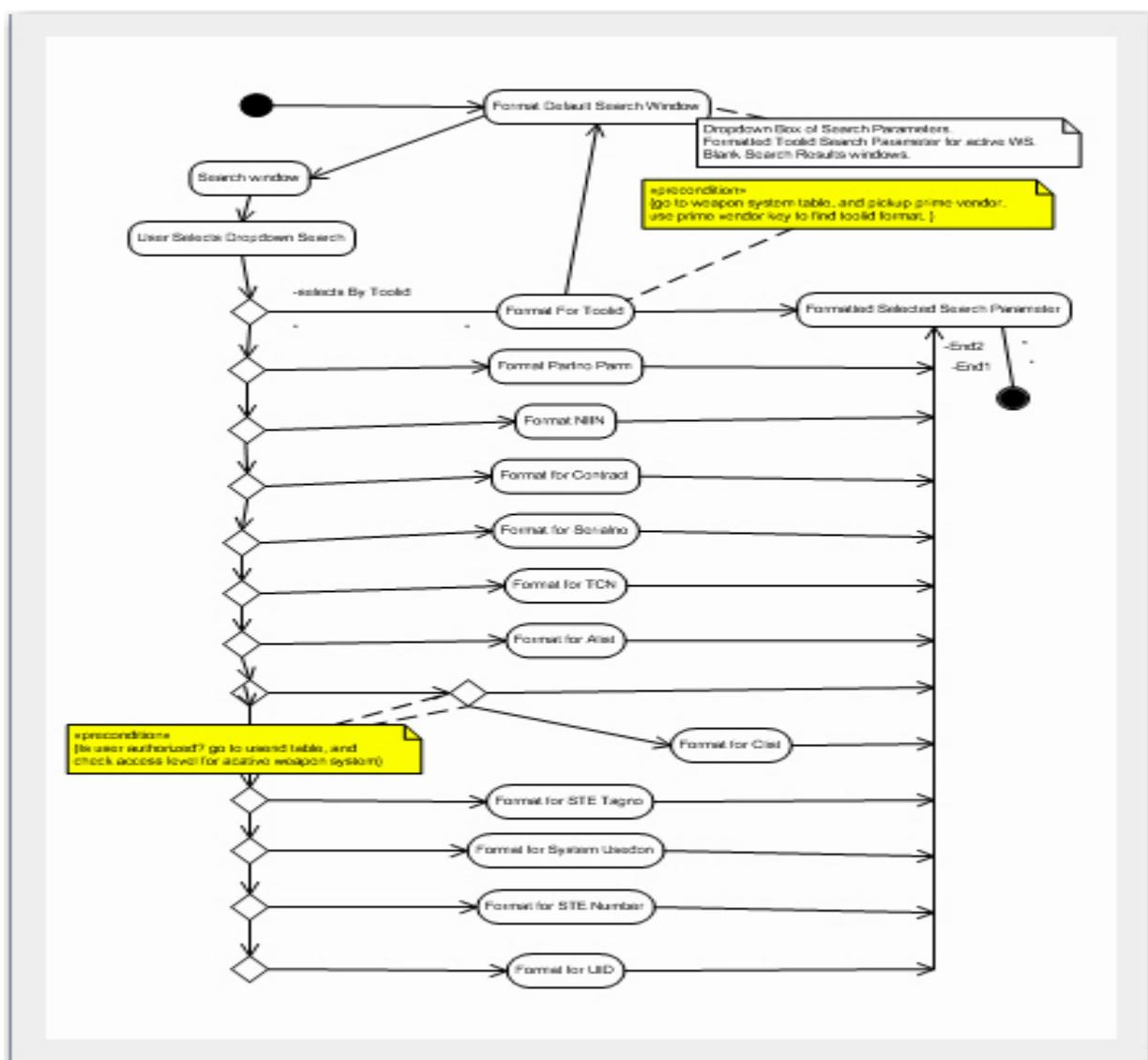


Figure 7 Search Activity Diagram

Separate documents were created for each function. Refer to the table below to determine available subfunctions.

Document Name

- 04-05-51 Search for Tool ID.doc
- 04-05-51A Search Results – Tool list.doc
- 04-05-51B Show Tool Dups - Tool.doc
- 04-05-51C Send Tool Dups to Tool Bin.doc
- 04-05-52 Search for Part Number.doc
- 04-05-52A Expand Part Number.doc
- 04-05-52B Expand Source.doc
- 04-05-52C Expand Model Effectivity.doc
- 04-05-53 Search for NIIN. doc
- 04-05-54 Search for Contract Number. doc

Document Name

- 04-05-57C Show Tool Dups – Alist.doc
- 04-05-58 Search for CLIst.doc
- 04-05-58A Search Results - CList.doc
- 04-05-58B Get Tools from CList.doc
- 04-05-58C Show Tool Dups – CLList .doc
- 04-05-58D Change CList Status.doc
- 04-05-59 Search for STE Property Number.doc
- 04-05-59A Search Results - STE Property Number.doc
- 04-05-59B Get Tools from STE Property Number.doc
- 04-05-59C Show STE Property

	Number.doc
04-05-54A Search Results – Contract Number list.doc	04-05-60 Search for STE System.doc
04-05-54B Get Tools from Contract Number .doc	04-05-61 Search for STE Number.doc
04-05-54C Show Tool Dups – Contract Number.doc	04-05-61A Search Results - STE Number.doc
04-05-55 Search for Contract Number. doc	04-05-61B Get Tools from STE Number.doc
04-05-55A Search Results – Serial Number list.doc	04-05-61C Show Property Numbers - STE Number.doc
04-05-55B Get Tools from Serial Number.doc	04-05-61D Get STE Details - Usage.doc
04-05-55C Show Tool Dups – Serial Number.doc	04-05-62 Search for UII.doc
04-05-56 Search for TCN. doc	04-05-62A Search Results – UII List.doc
04-05-56A Search Results – TCN list.doc	04-05-62B Get Tools from UII.doc
04-05-56B Get Tools from TCN.doc	04-05-62C Show Tool Dups - UII.doc
04-05-56C Show Tool Dups – TCN.doc	04-05-63 Search for Bar Code.doc
04-05-57 Search for AList. doc	04-05-63A Search Results – Bar Code.doc
04-05-57A Search Results – AList list.doc	04-05-63B Get Tools from Bar Code.doc
04-05-57B Get Tools from AList.doc	04-05-63C Show Tool Dups – Bar Code.doc
04-05-40 Advanced Search – Usage.doc	

Table 1 Search Sub-function Documents

4.3. eTIMS Tool Subsystem

The Tool Subsystem displays the detailed information about the selected tool. The goal of the tool module is to give the user all the detailed information known about the selected tool.

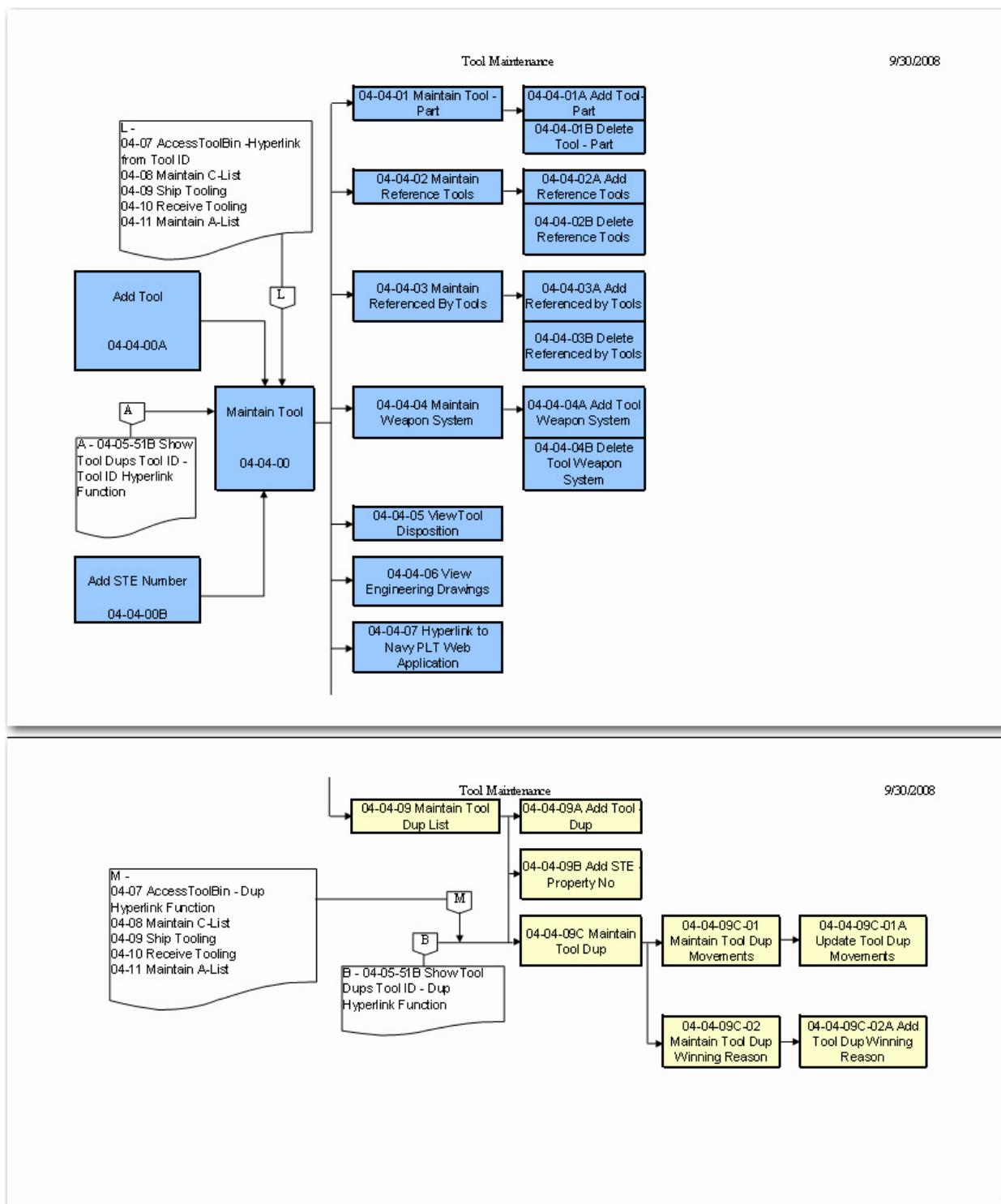


Figure 8 Tooling Subsystem Functional Flow Diagram

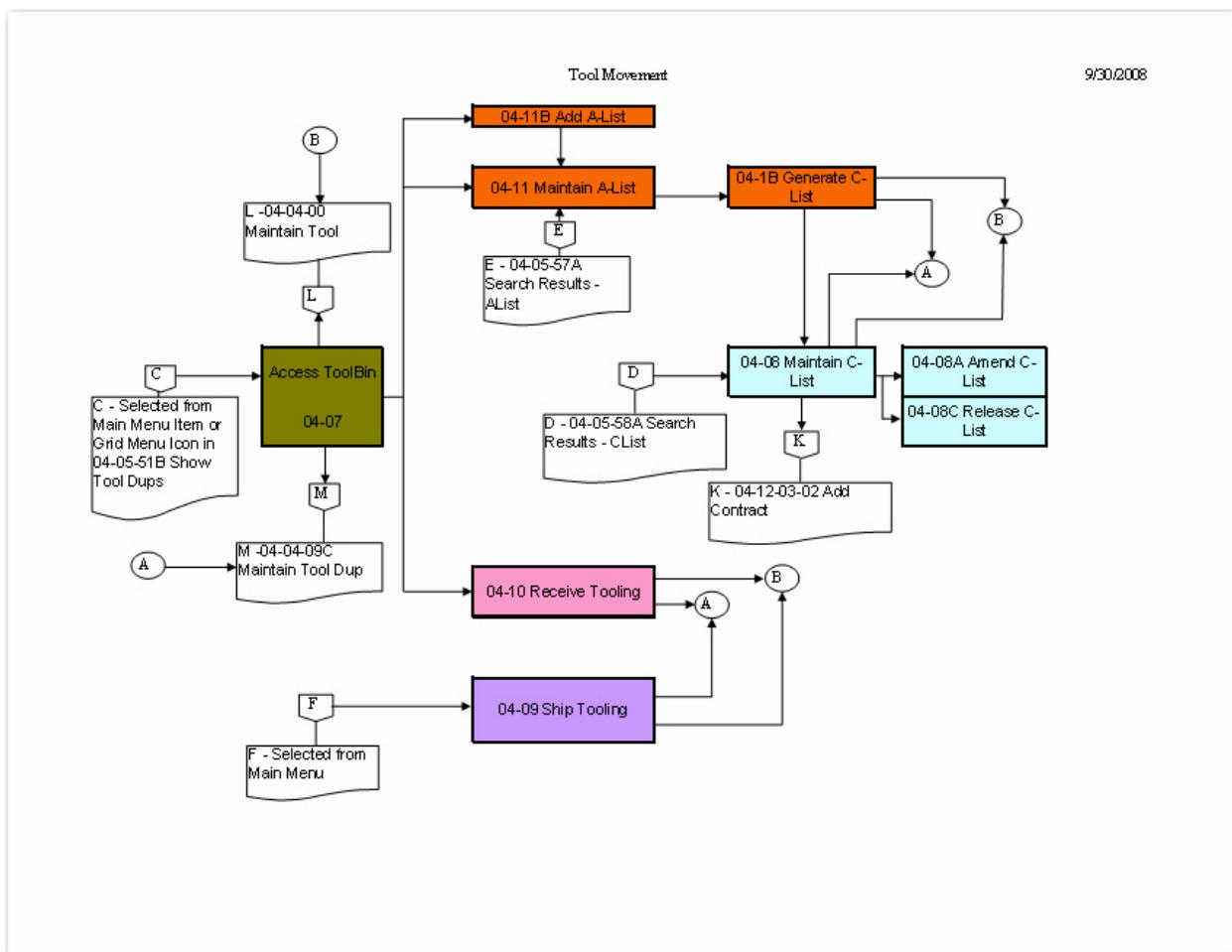


Figure 9 Tool Movement Functional Flow Diagram

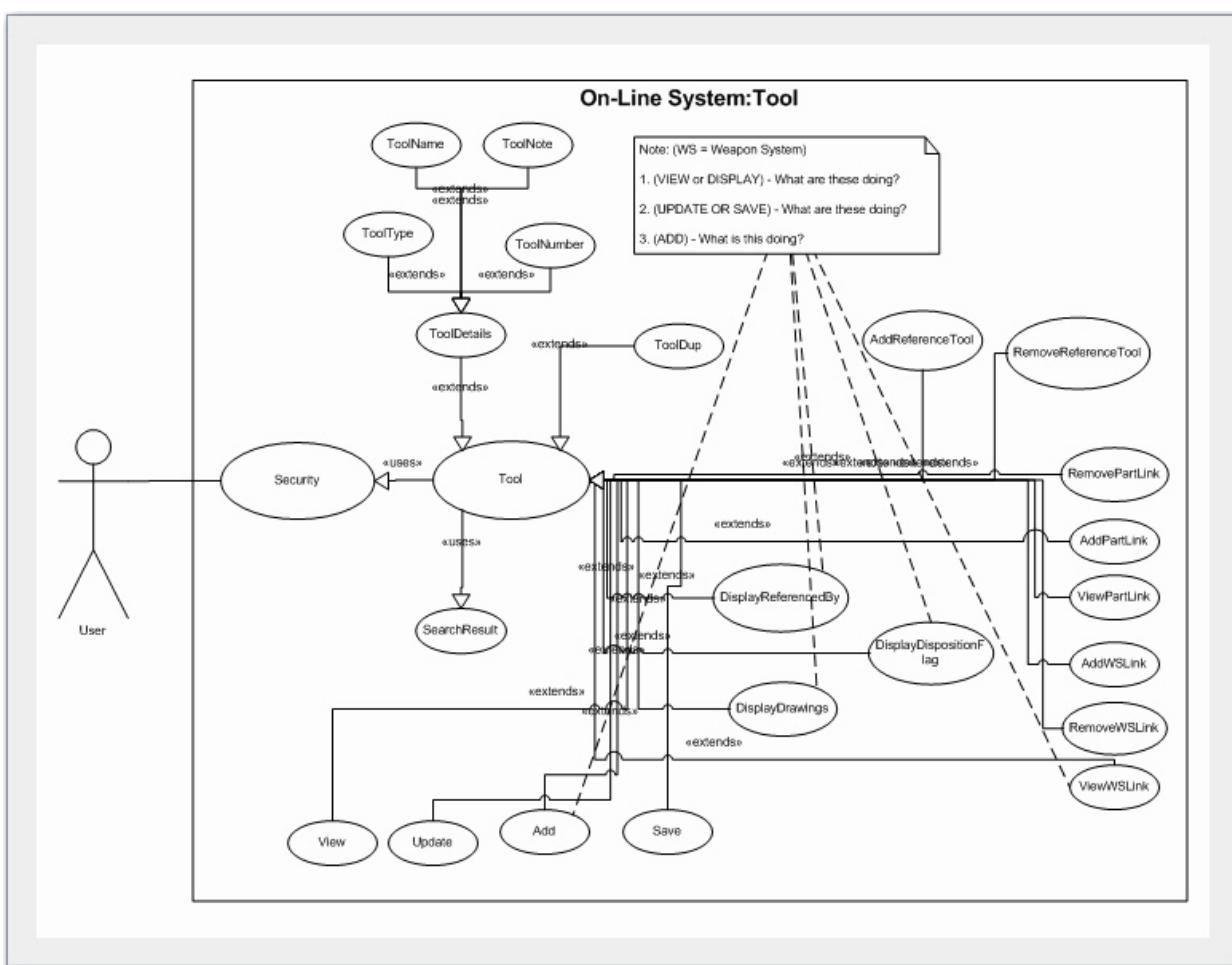


Figure 10 Tool Use Case

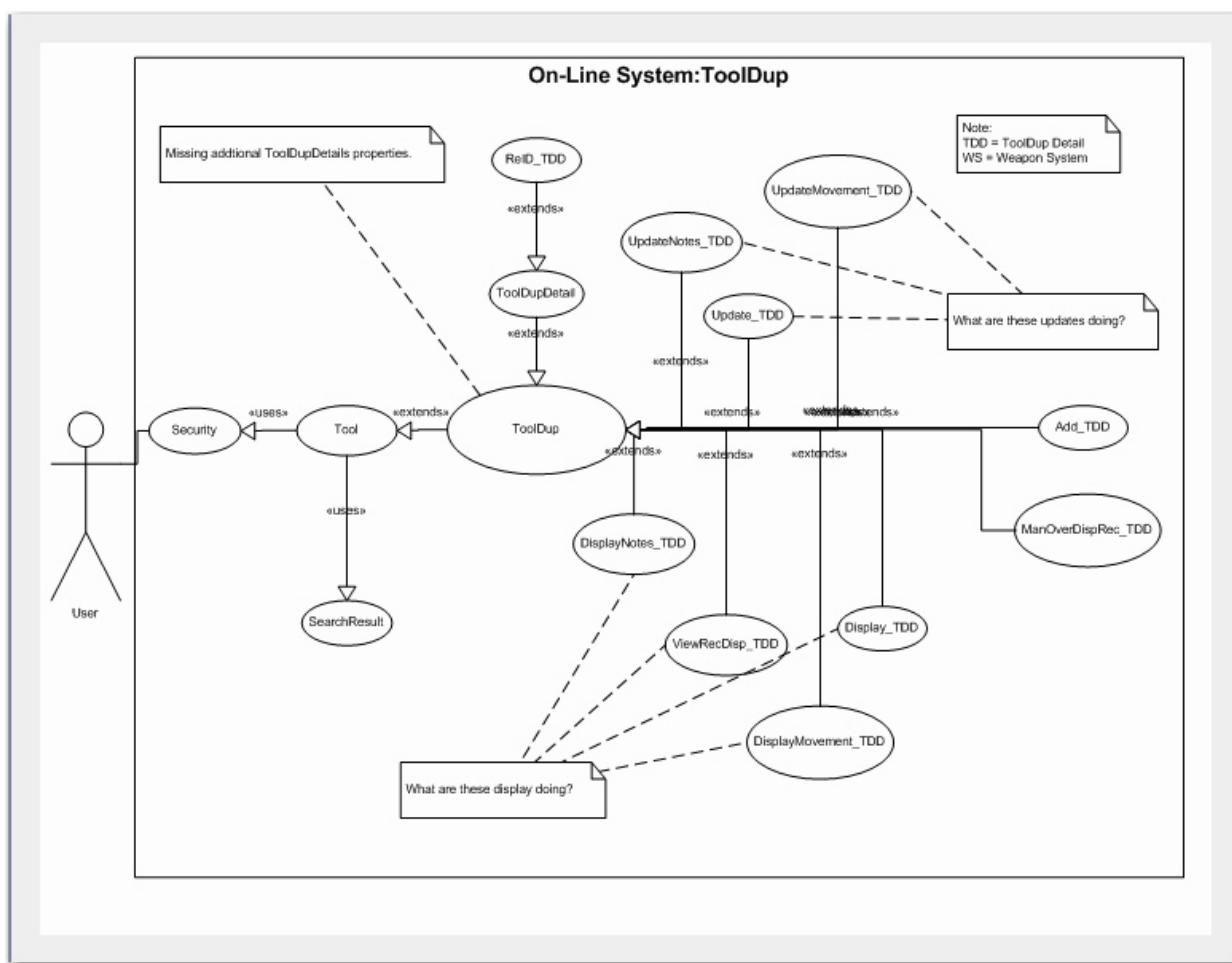


Figure 11 Tool Dup Use Case

Separate documents were created for each function. Refer to the table below to determine available subfunctions.

Document Name

04-02-00 Maintain Tool.doc

04-02-00A Add Tool.doc

04-02-00B Add STE Number.doc

04-02-01 Maintain Tool-Part.doc

04-02-01A Add Tool-Part.doc

04-02-01B Delete Tool-Part.doc

04-02-02 Maintain Reference Tools.doc

04-02-02A Add Reference Tools.doc

04-02-02B Delete Reference Tools. doc

04-02-03 Maintain Referenced by Tools. doc

04-02-03A Add Referenced by Tools.doc

Document Name

04-02-04B Delete Referenced by Tools.doc

04-02-05 View Tool Disposition.doc

04-02-06 View Engineering Drawings.doc

04-02-07 Hyperlink to Navy PLT Web Application.doc

04-02-09 Maintain Tool Dup List .doc

04-02-09A Add Tool – Dup – Usage Scenario.doc

04-02-09B Add STE – Property No – Usage Scenario.doc

04-02-09C Maintain Tool Dup.doc

04-02-09C-01 Maintain Tool Dup Movements.doc

04-02-09C-01A Update Tool Dup Movements.doc

04-02-09C-02 Maintain Tool Dup

04-02-03B Delete Referenced By Tools .doc	Winning Reason.doc
04-02-04 Maintain Tool Weapon System.doc	04-02-09C-02A Add Tool Dup Winning Reason.doc
04-02-04A Add Tool Weapon System. doc	04-02-09C-03 Reidentify Tool Dup.doc
	04-02-10 Get Current Tool Movement.doc

Table 2 Tool/Tool Dup Sub-function Documents

4.4. eTIMS Part/STE Systems Subsystem

The Part Subsystem allows the user to display the detailed part information for the selected part/source.

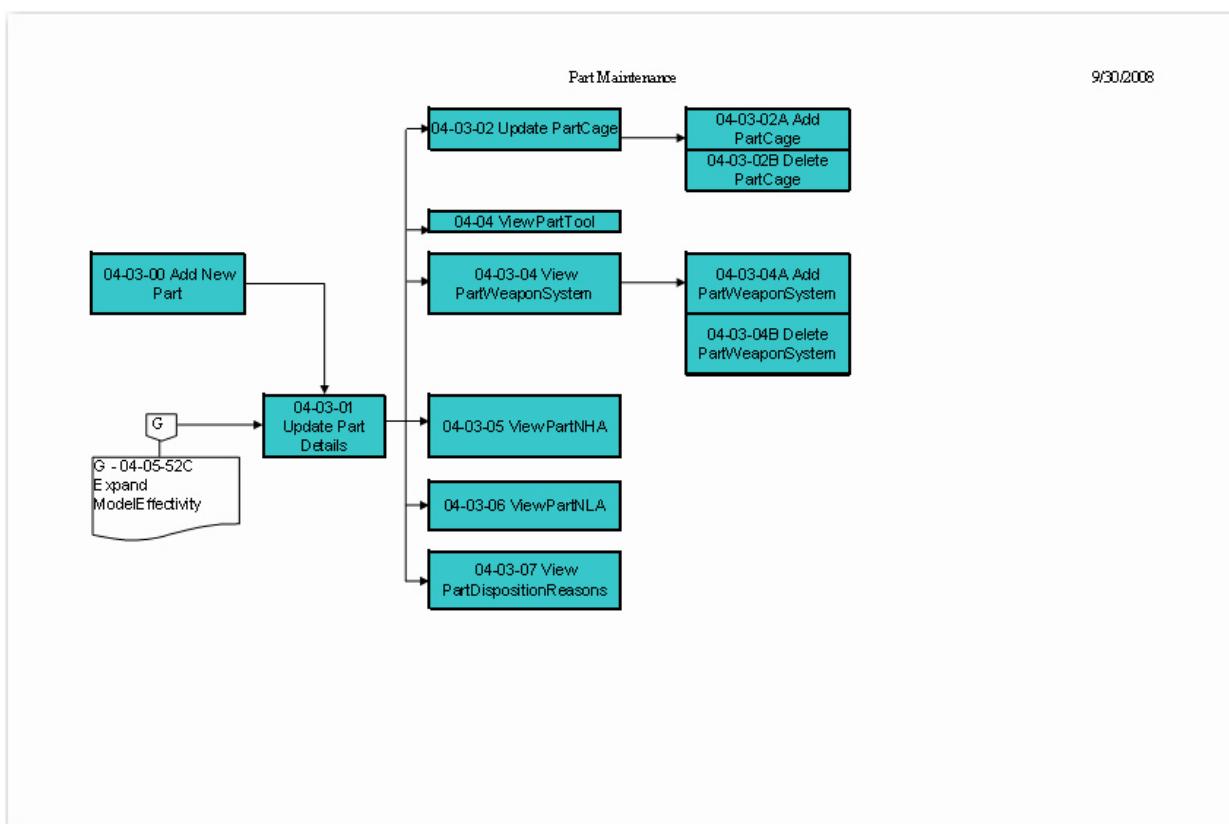


Figure 12 Part Subsystem Functional Flow Diagram

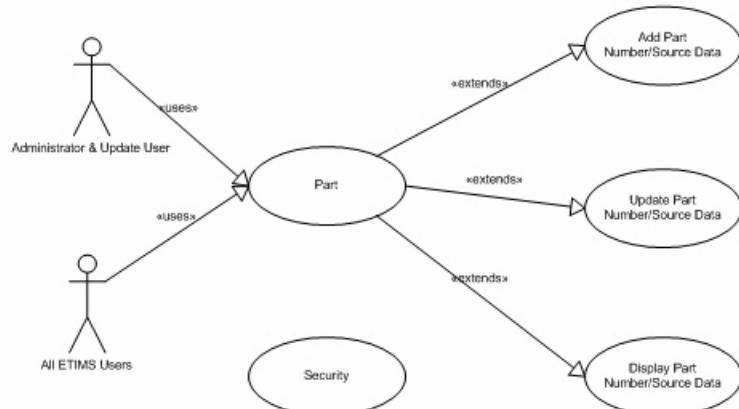


Figure 13 Part Subsystem Use Case

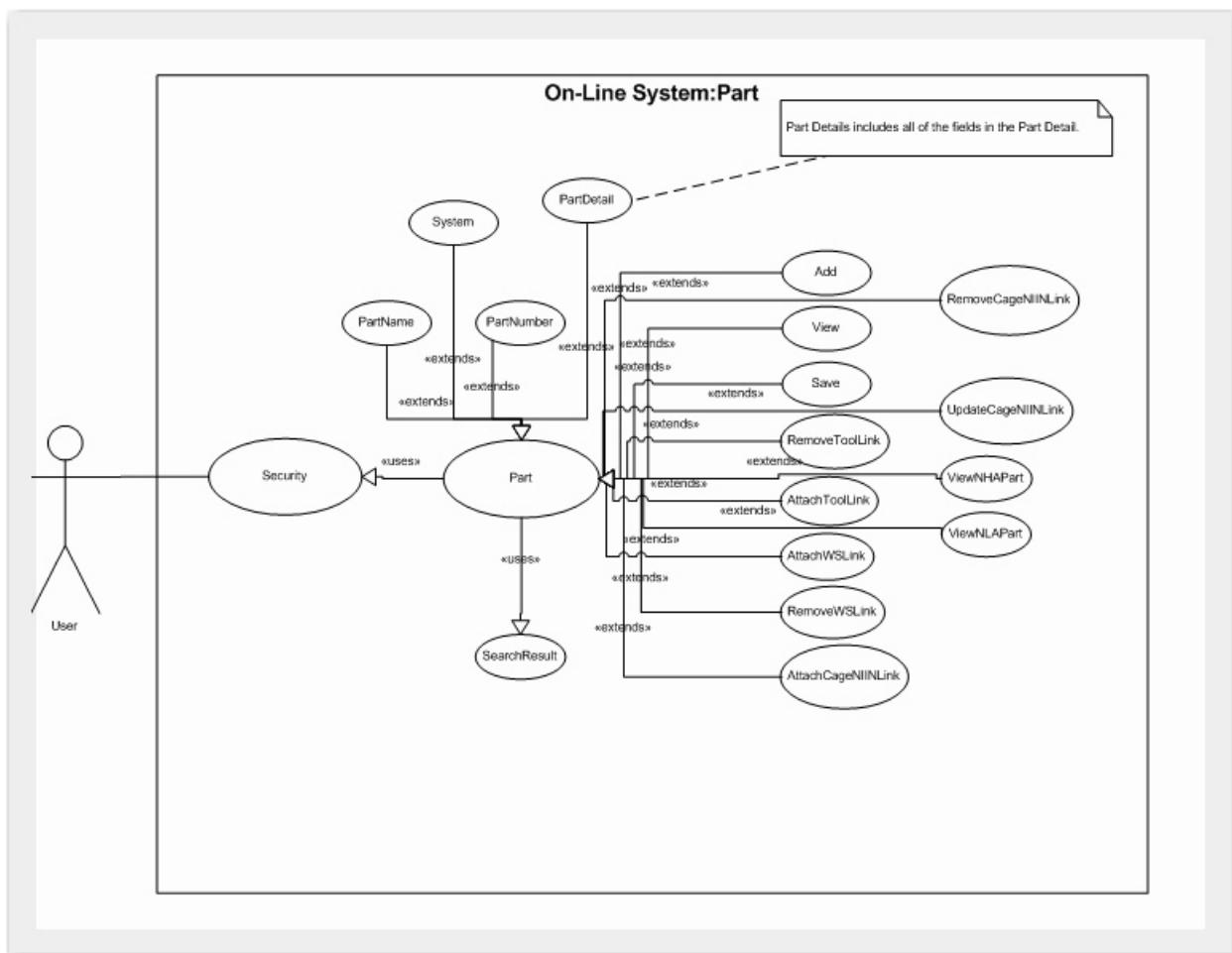


Figure 14 Maintain Part Use Case

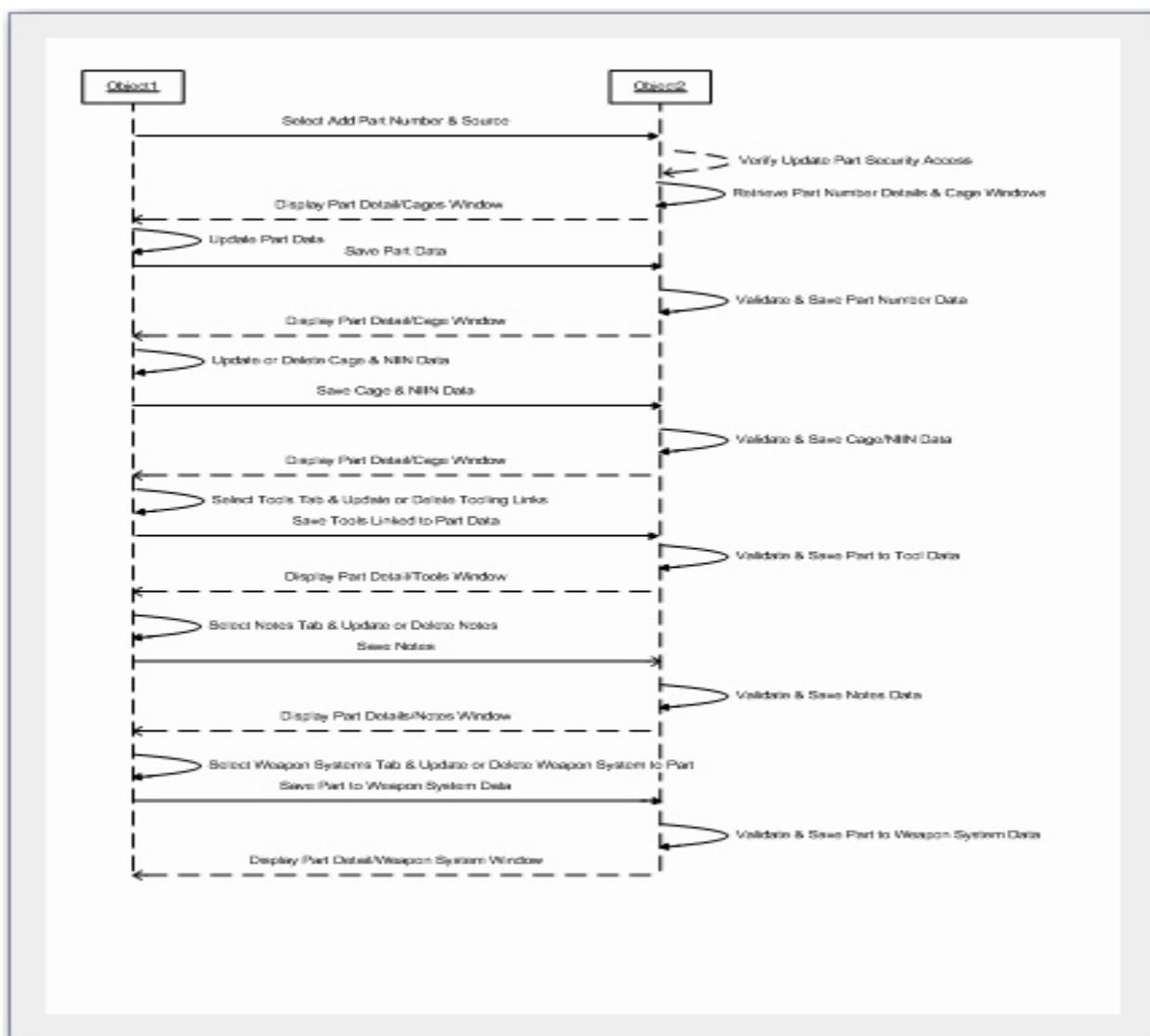


Figure 15 Maintain Part Number Activity Diagram

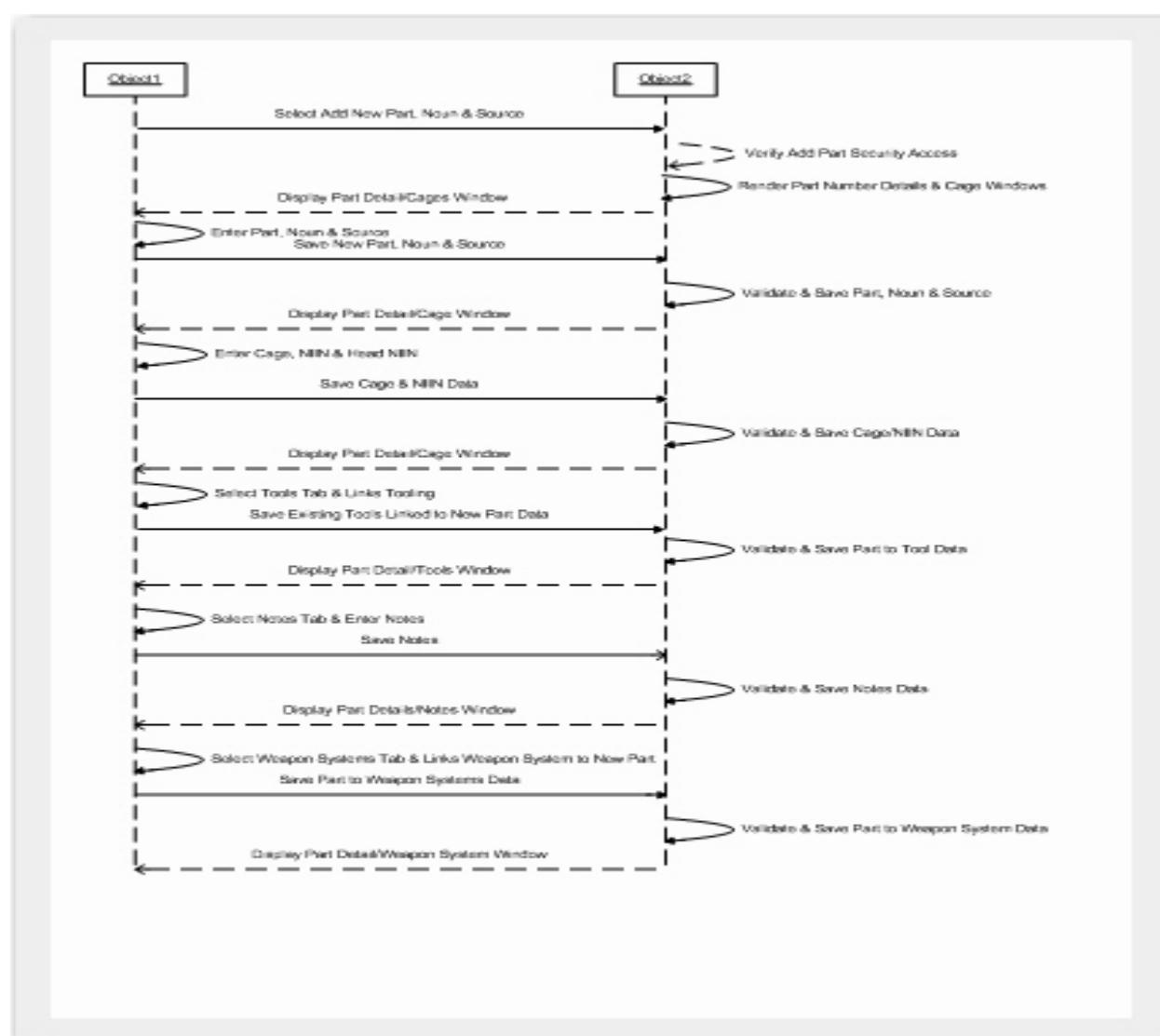


Figure 16 Add Part Number Activity Diagram

Separate documents were created for each function. Refer to the table below to determine available subfunctions.

Document Name

- 04-02-00 Add New Part – Usage Scenario .doc
- 04-03-01 Update Part Details – Usage Scenario.doc
- 04-03-02 Update PartCage – Usage Scenario .doc
- 04-03-02A Add PartCage – Usage Scenario .doc
- 04-03-02B Delete PartCage – Usage Scenario.doc
- 04-03-02C View PartCage – Usage Scenario .doc
- 04-03-03 View PartTool – Usage Scenario .doc

Document Name

- 04-03-04 View PartWeaponSystem – Usage Scenario.doc
- 04-03-04A Add PartWeaponSystem – Usage Scenario.doc
- 04-03-04B Delete PartWeaponSystem – Usage Scenario.doc
- 04-03-05 View PartNHA – Usage Scenario.doc
- 04-03-06 View PartNLA – Usage Scenario .doc
- 04-03-07 View PartDispositionReasons – Usage Scenario.doc

Table 3 Part Sub-function Documents

4.5. eTIMS STE System Subsystem

The STE System Subsystem allows the user to display the detailed part information for the selected part/source.

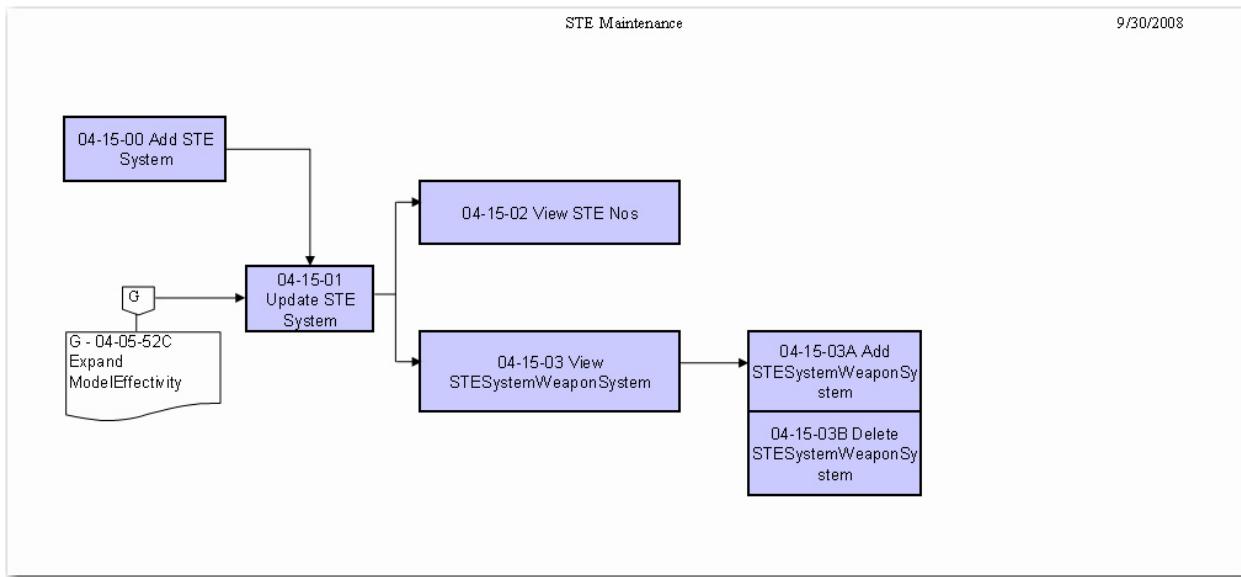


Figure 17 STE System Subsystem Functional Flow Diagram

Separate documents were created for each function. Refer to the table below to determine available subfunctions.

Document Name	Document Name
04-15-00 Add STE System.doc	04-15-03 ViewSTESystemWeaponSystem – Usage Scenario.doc
04-15-01 Update STE System.doc	04-15-03A AddSTESystemWeaponSystem – Usage Scenario.doc
04-15-02 View STE Nos.doc	04-15-03B DeleteSTESystemWeaponSystem – Usage Scenario. doc

Table 4 STE Sub-function Documents

4.6. eTIMS A-List Subsystem

The A-List Subsystem allows the user to create work lists of tool/dups that can be used to create shipping lists.

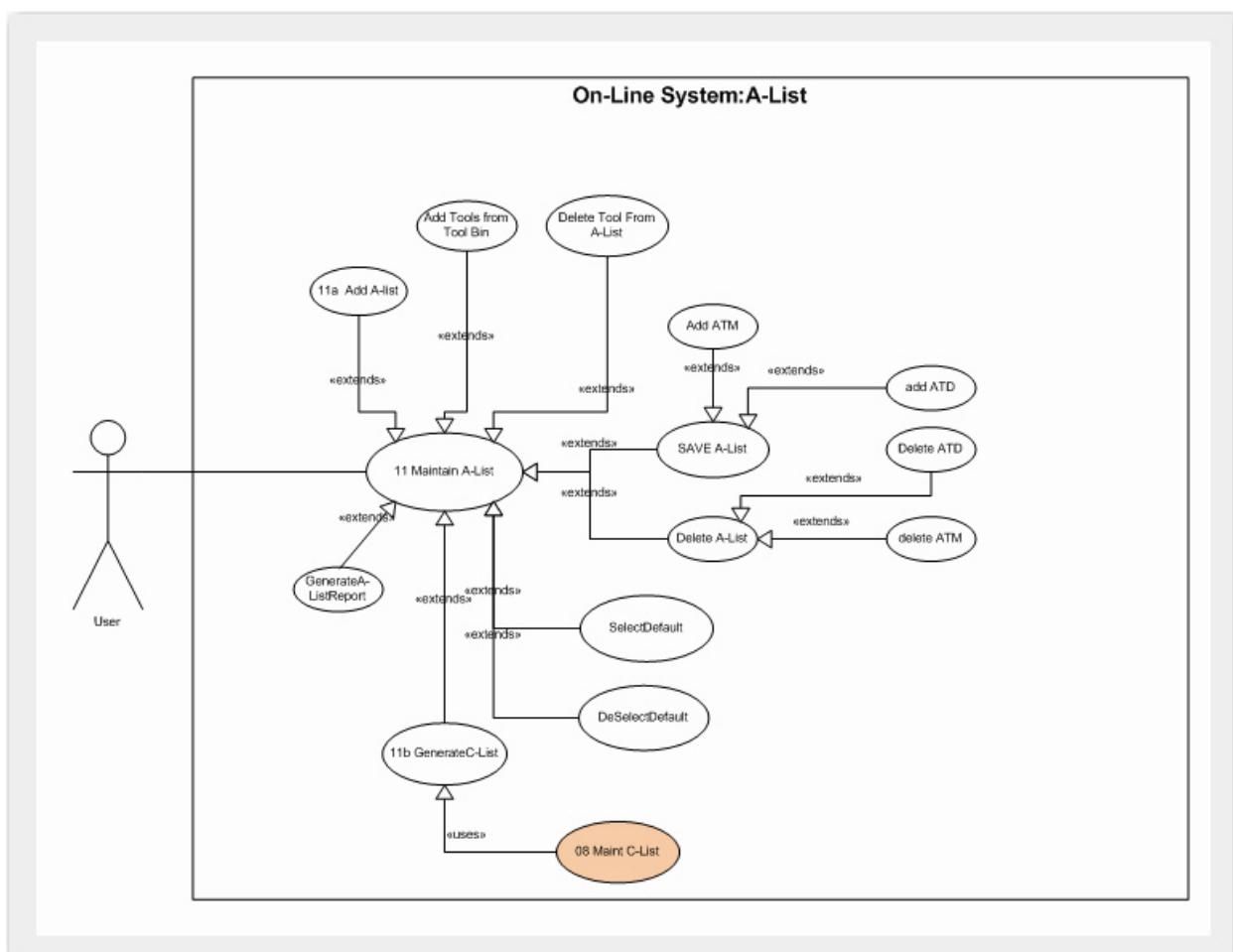


Figure 18: A-List Subsystem Functional Flow

Separate documents were created for each function. Refer to the table below to determine available subfunctions.

Document Name

04-06 Maintain Clist – Usage.doc
04-06A Add Clist – Usage.doc

Document Name

04-06B Generate Clist – Usage.doc

Table 5: A-List Subfunction Documents

4.7. eTIMS C-List Subsystem

The C-List Subsystem allows the user to create shipping lists for use by the storage facility or other locations that have the selected tooling.

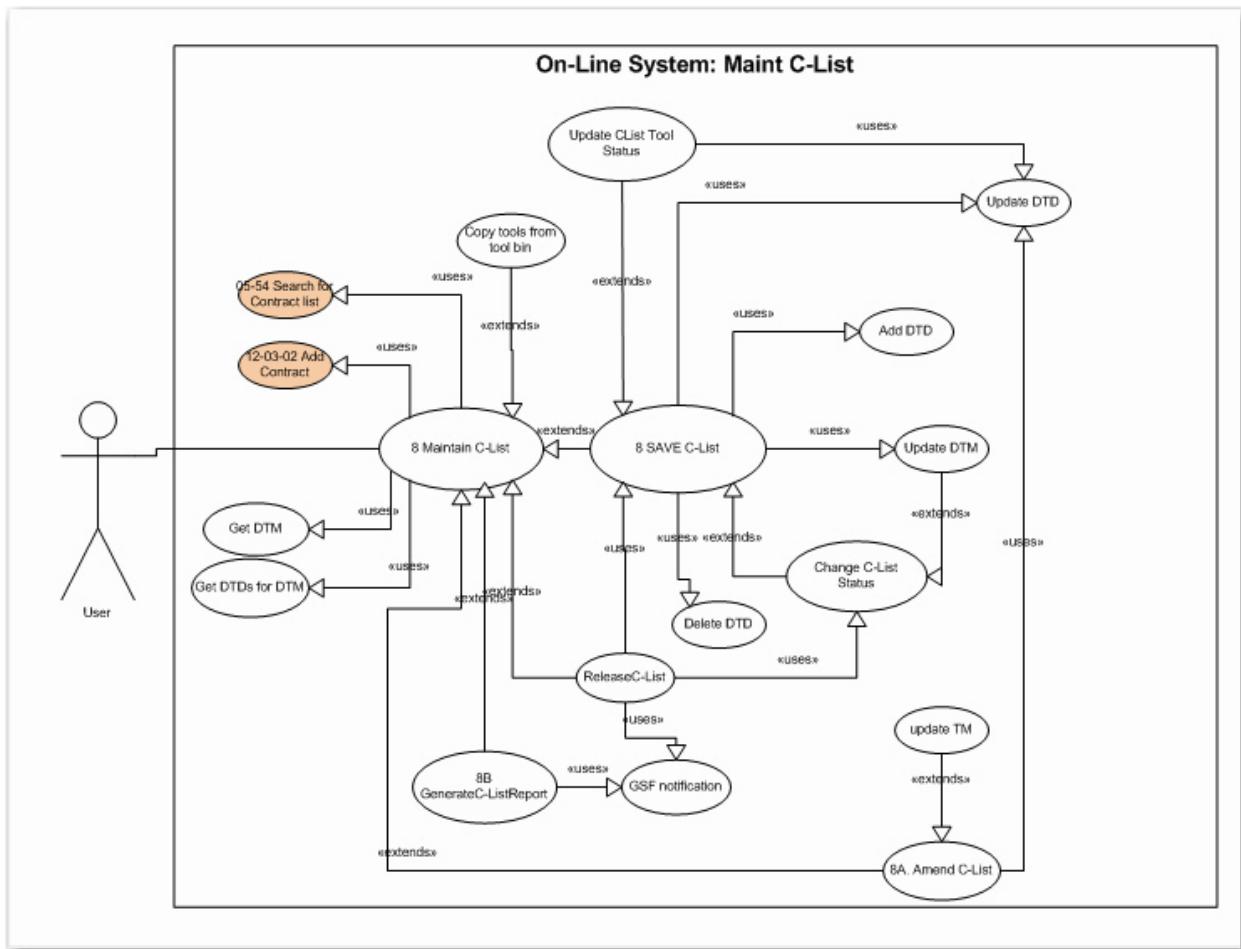


Figure 19: C-List Functional Flow

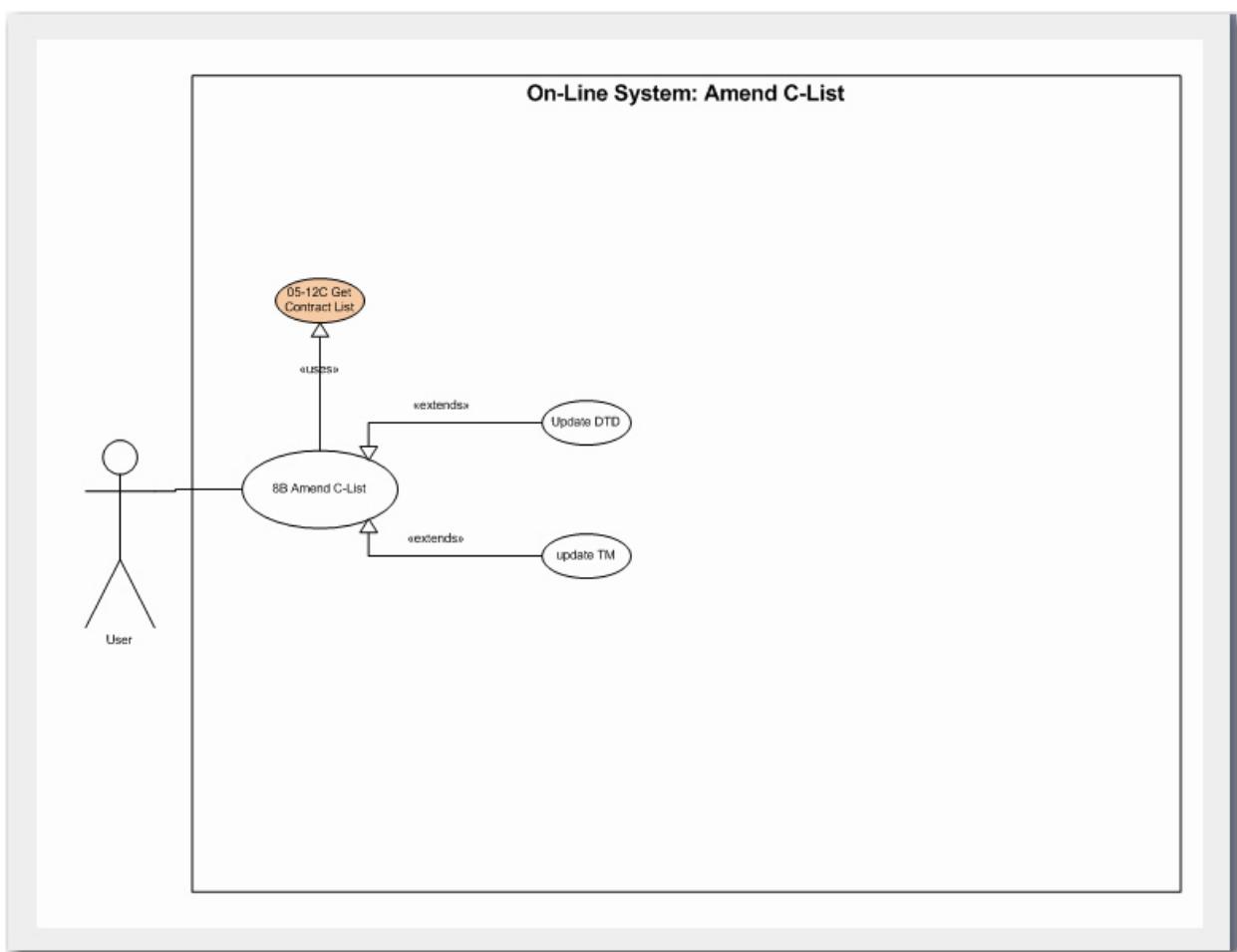


Figure 20: Amend C-List Functional Flow

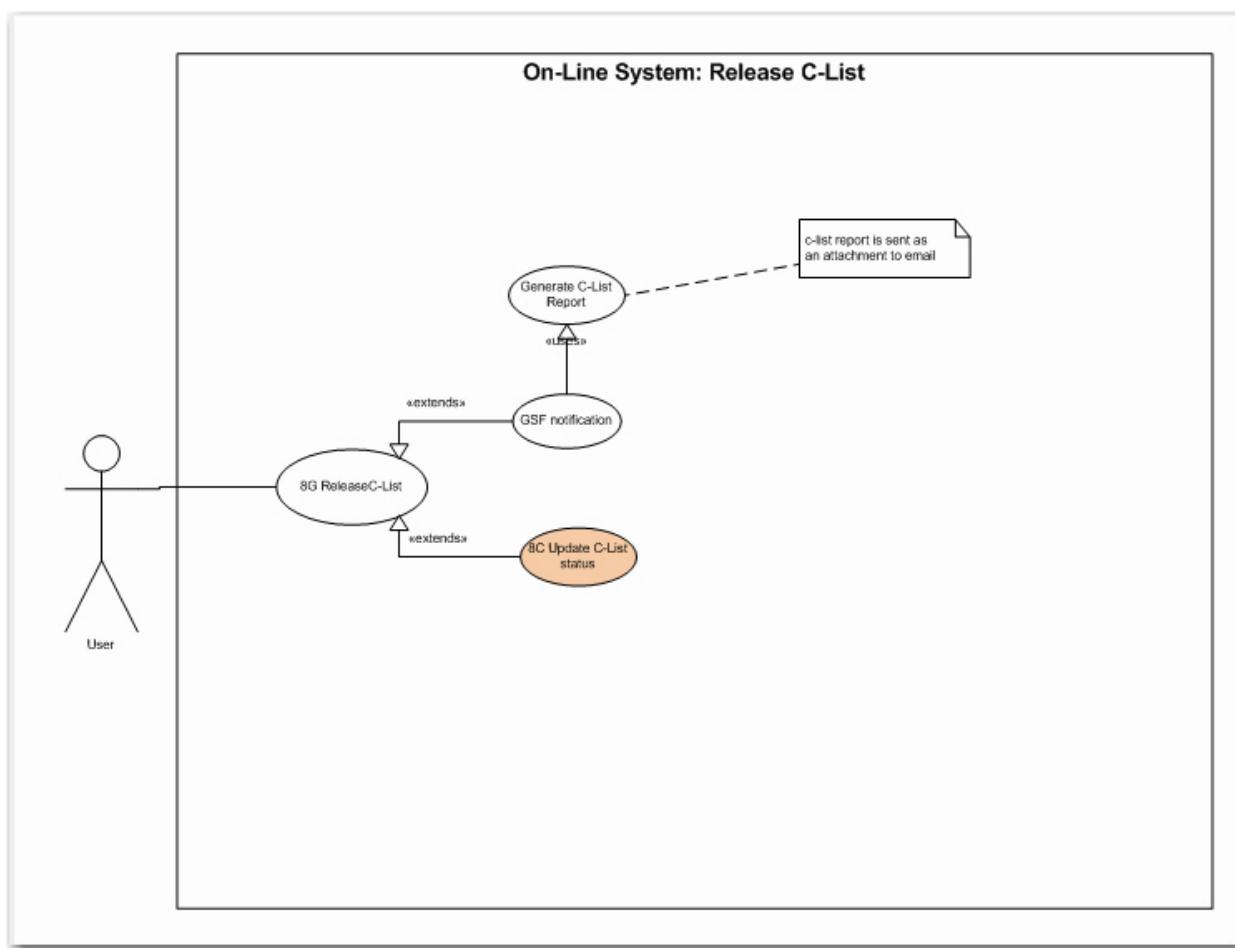


Figure 21: Release C-List Functional Flow

Separate documents were created for each function. Refer to the table below to determine available subfunctions.

Document Name

04-08 Maintain Clist – Usage.doc
04-08A Amend Clist – Usage.doc

Document Name

04-08B Release Clist – Usage.doc

Table 6: C-List Subfunction Documents

4.8. eTIMS Receive Tools Subsystem

The Receive Subsystem allows the storage facility to mark tools as stored in their facility.

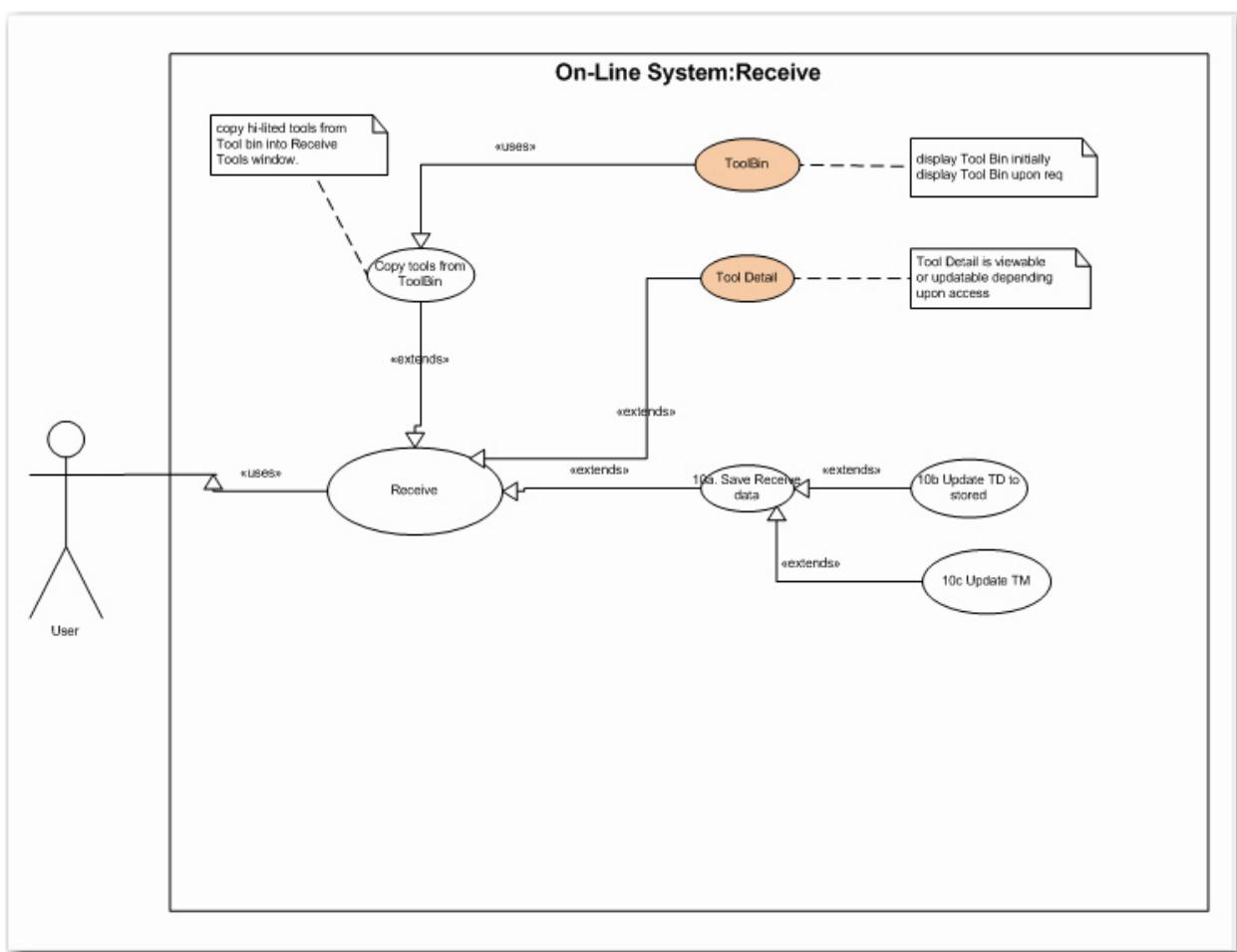


Figure 22: Receive Tooling Functional Flow

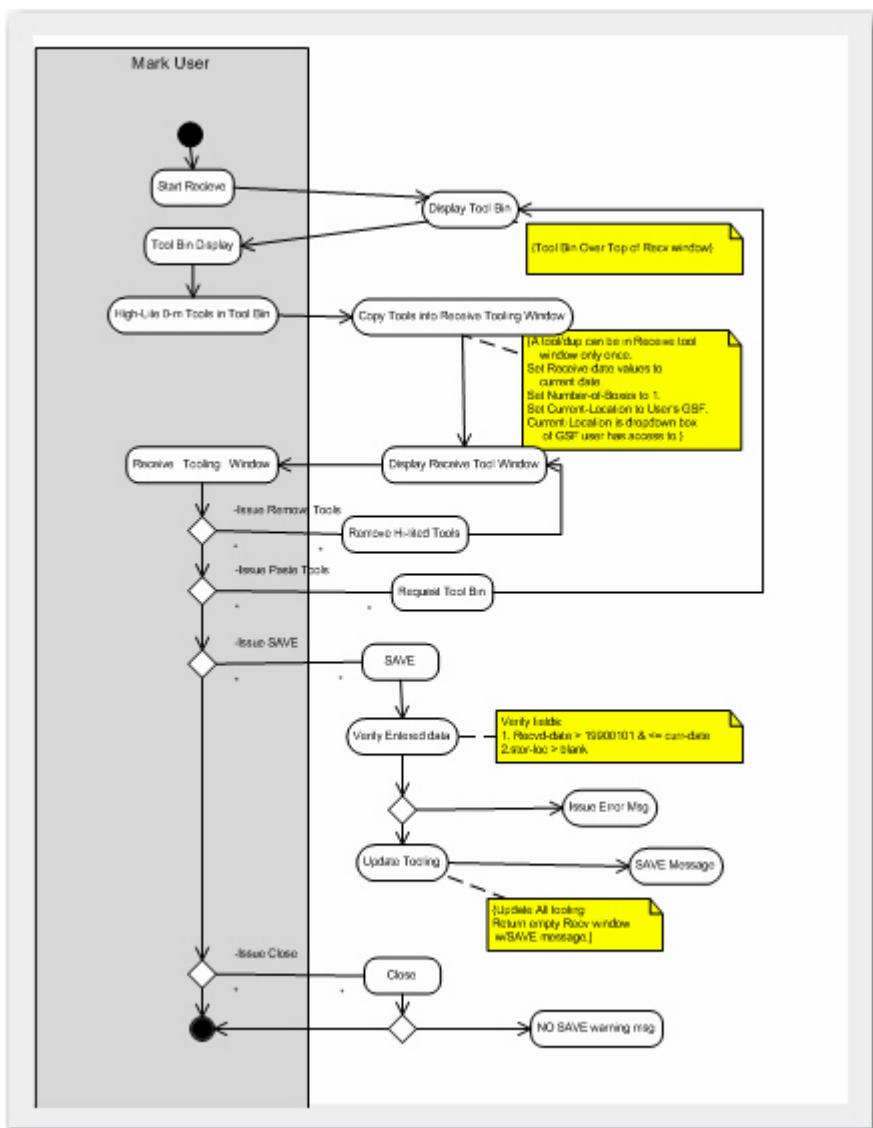


Figure 23: Receive Tooling Use Case

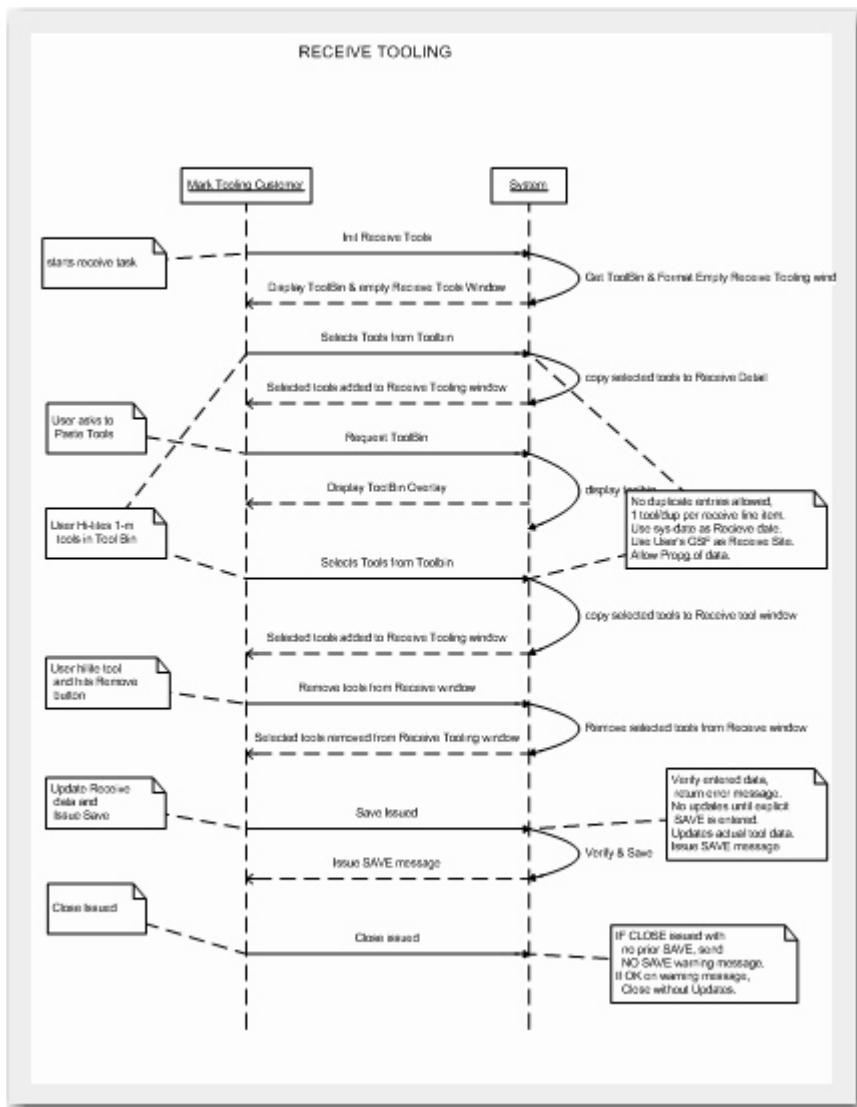


Figure 24: Receive Subsystem Sequence Diagram

Separate documents were created for each function. Refer to the table below to determine available subfunctions.

Document Name

04-10 Receive tooling – Usage.doc

Document Name

Table 7: Receive Subsystem Documents

4.9. eTIMS Ship Tools Subsystem

The Ship Subsystem allows the user to mark tools as shipped to a vendor for use.

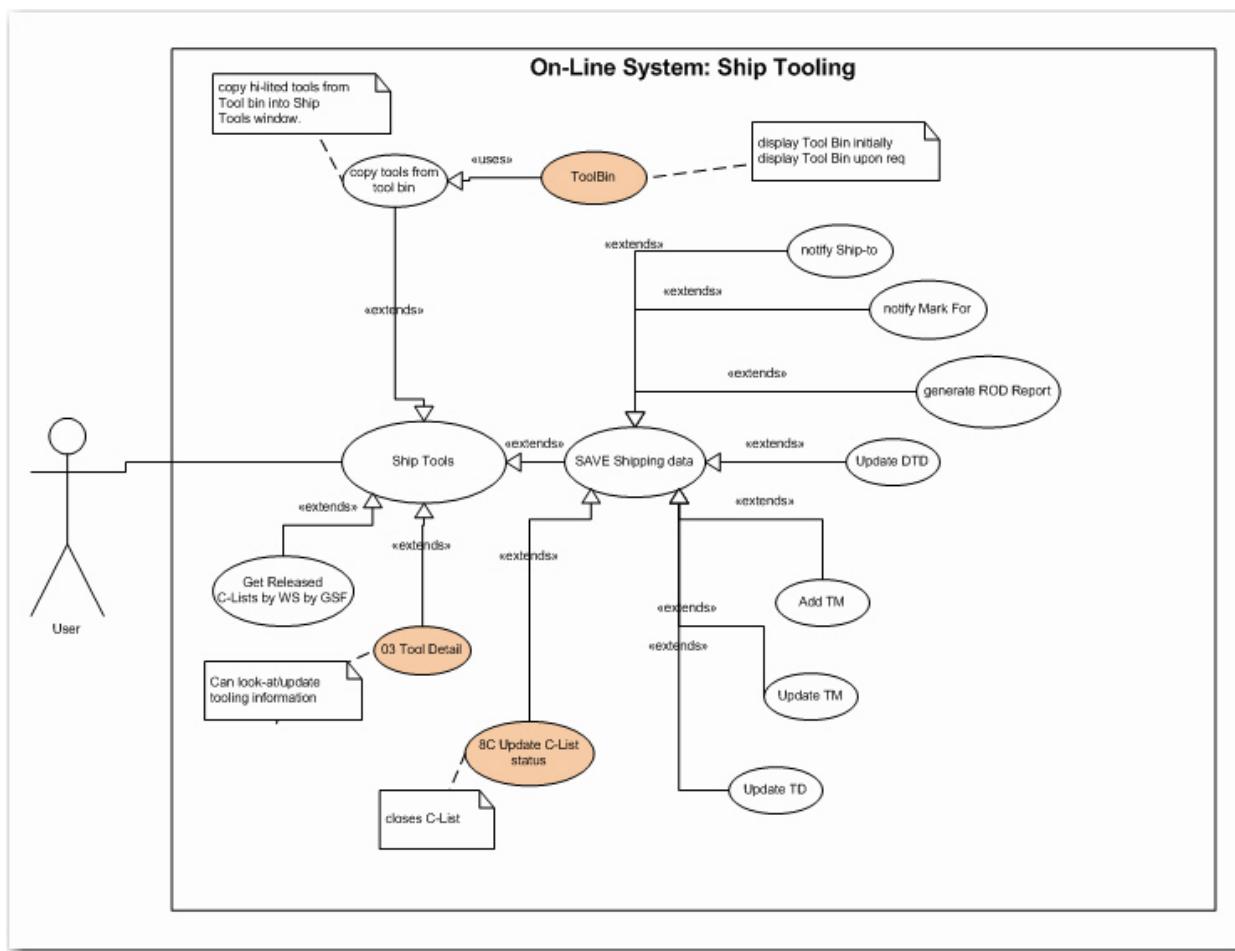


Figure 25: Ship Subsystem Functional Flow

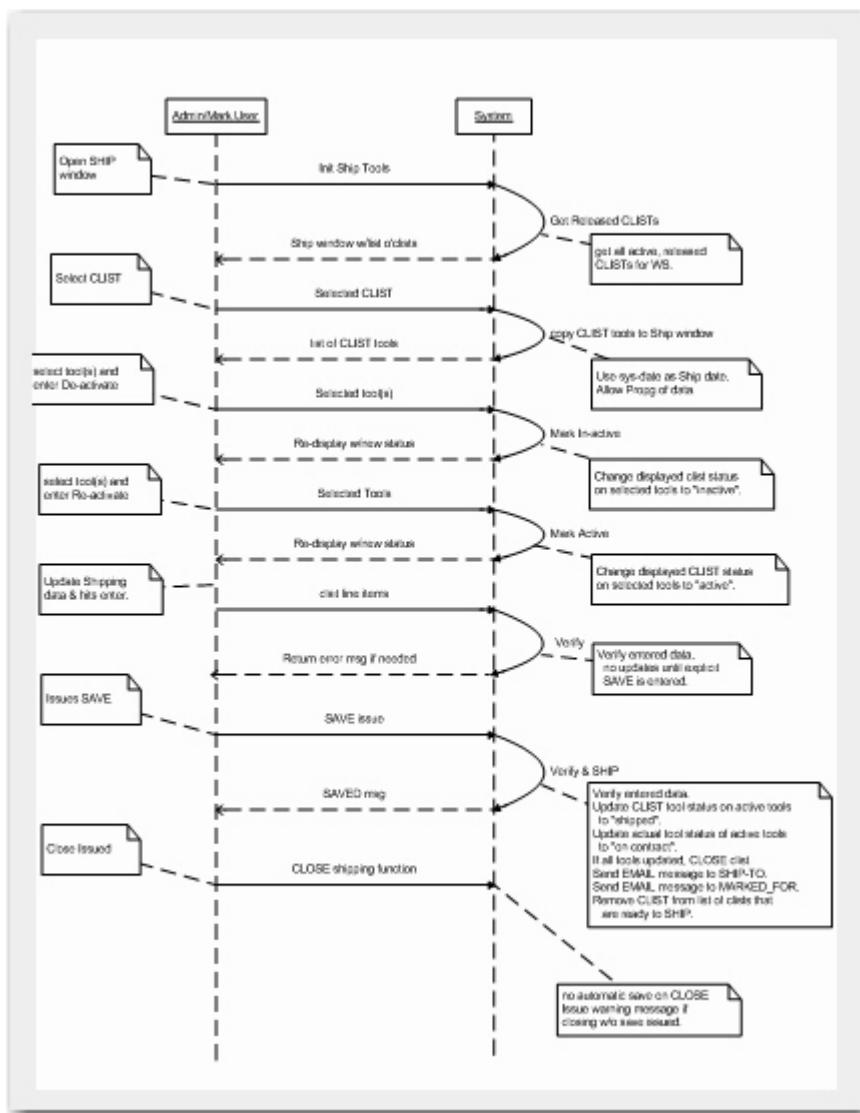


Figure 26: Ship Subsystem Sequence Diagram

Separate documents were created for each function. Refer to the table below to determine available subfunctions.

Document Name

04-09-Ship Tooling – Usage.doc

Document Name

Table 8: Ship Subsystem Documents

4.10. eTIMS Reports Subsystem

The Report Subsystem allows the user to generate canned reports from the system.

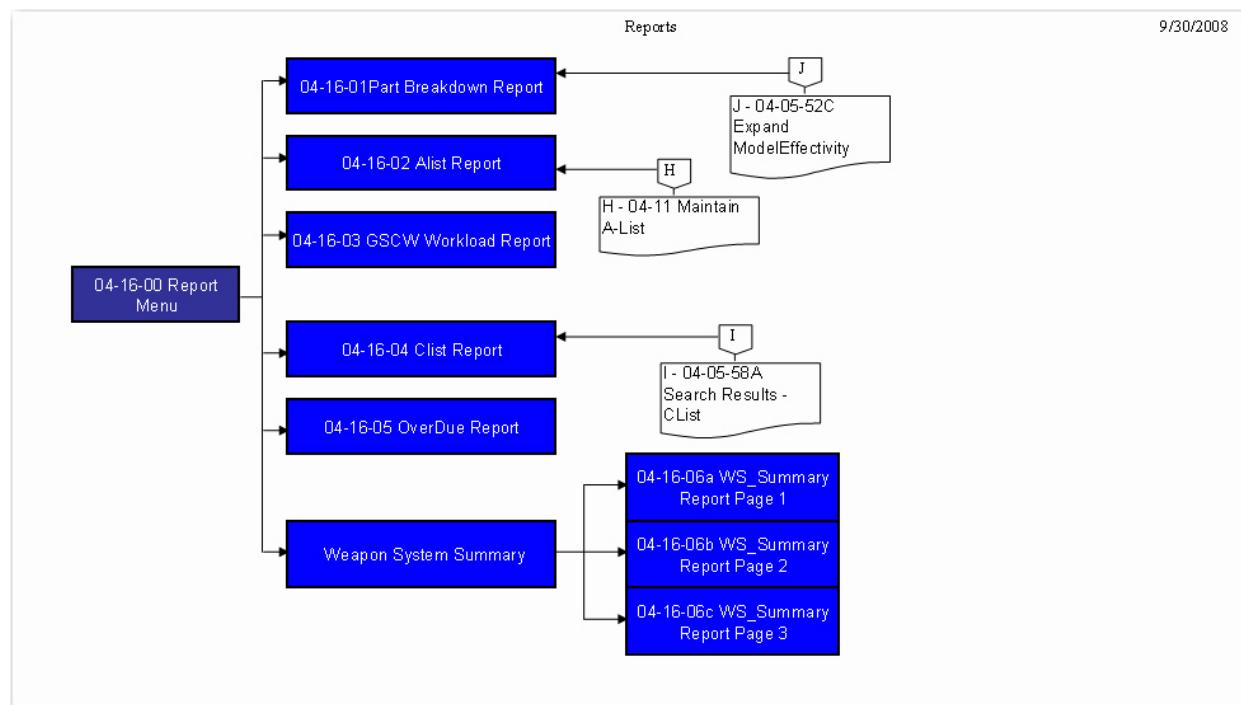


Figure 27 Reports Subsystem Functional Flow Diagram

Separate documents were created for each function. Refer to the table below to determine available subfunctions.

Document Name

04-15-00 Reports Menu – Usage

Scenario.doc

04-16-01 Part Breakdown Report – Usage

Scenario.doc

04-16-02 Alist Report – Usage Scenario.doc

04-16-03 GSCW Workload Report – Usage Scenario.doc

04-16-04 Clist Report – Usage Scenario.doc

Document Name

04-16-05 OverDue Report – Usage

Scenario.doc

04-16-06A WS_Summary Report Page 1.doc

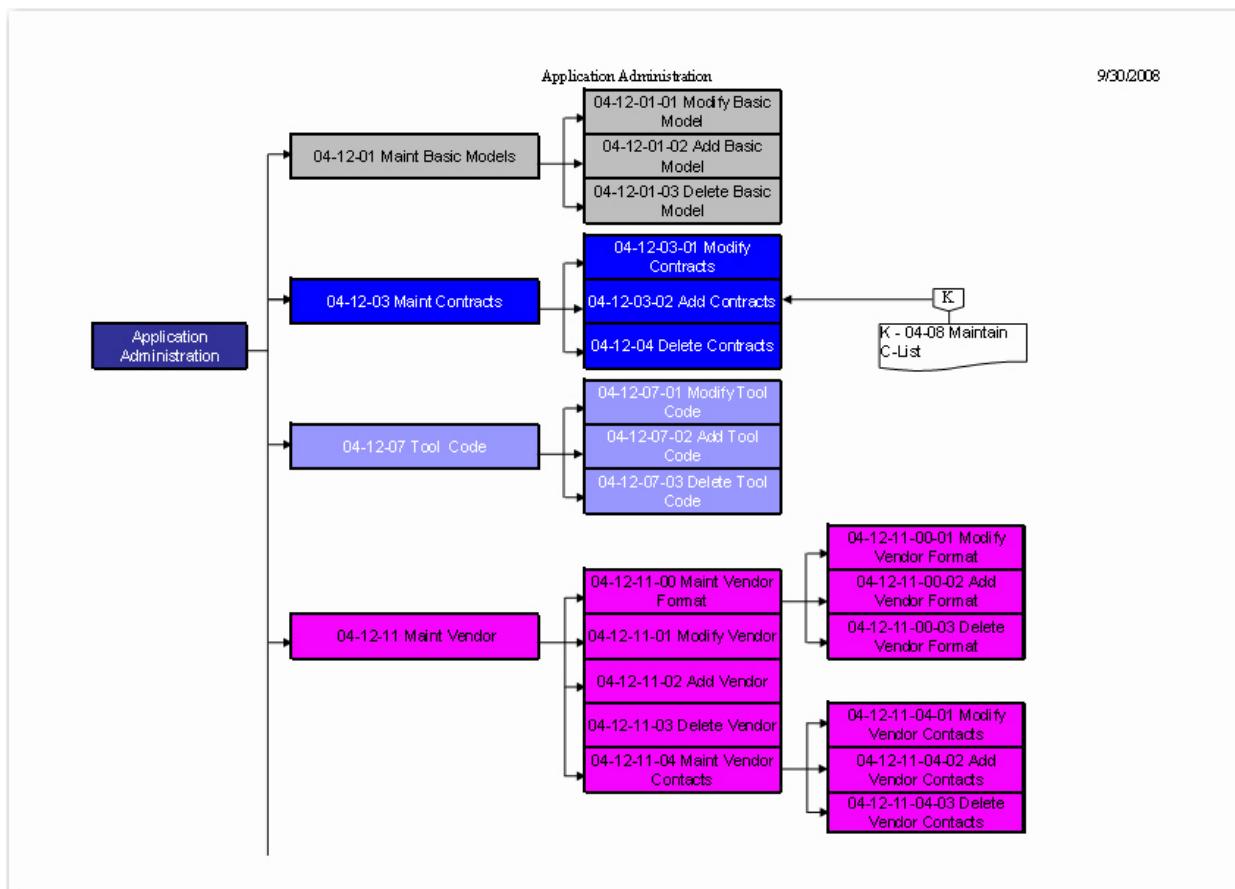
04-16-06B WS_Summary Report Page 2.doc

04-16-06C WS_Summary Report Page 3.doc

Table 9 Reports Sub-function Documents

4.11. eTIMS Application Administration Subsystem

The Application Subsystem allows the users with the appropriate access level to manage system tables.



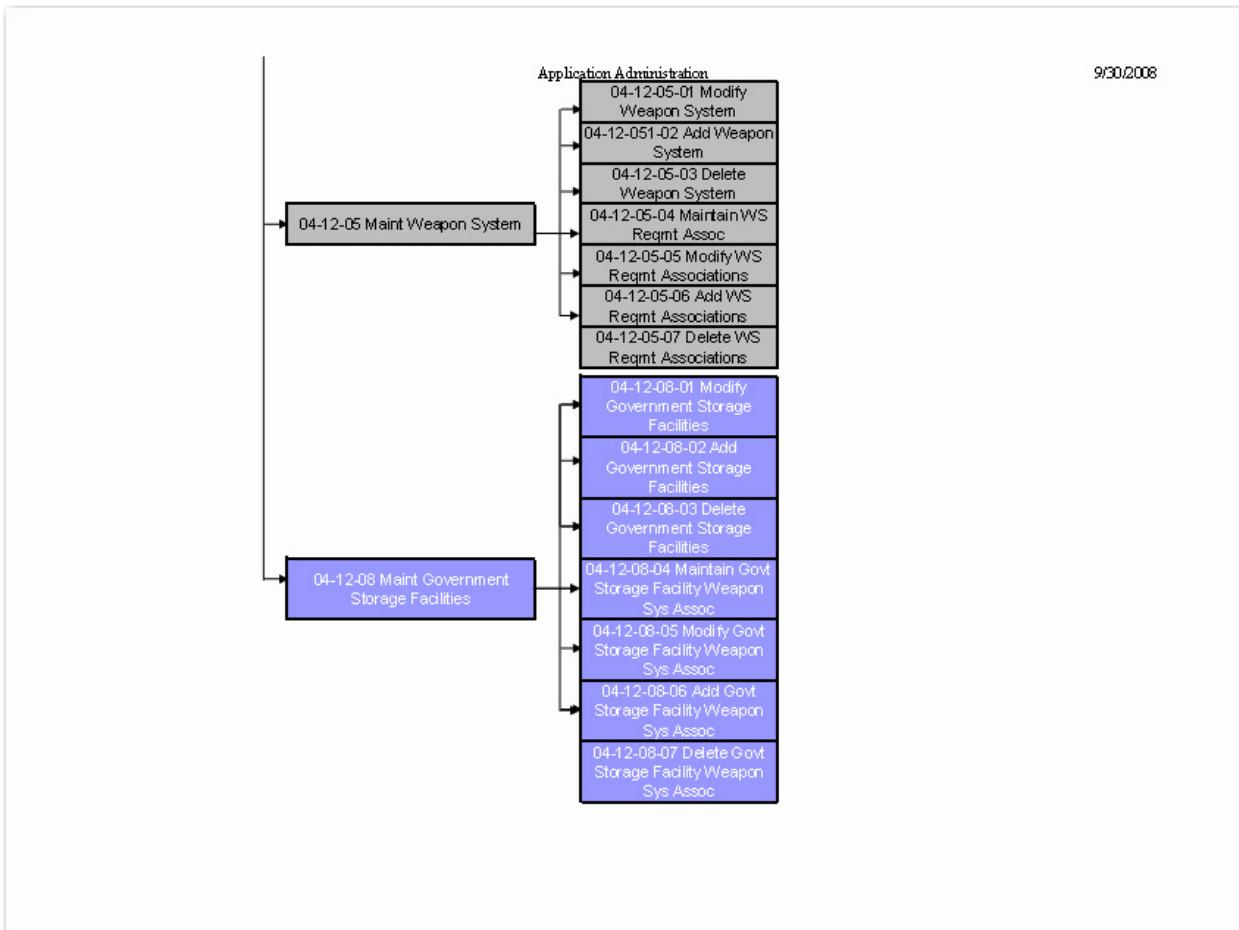


Figure 28 Application Administration Subsystem Functional Flow Diagram

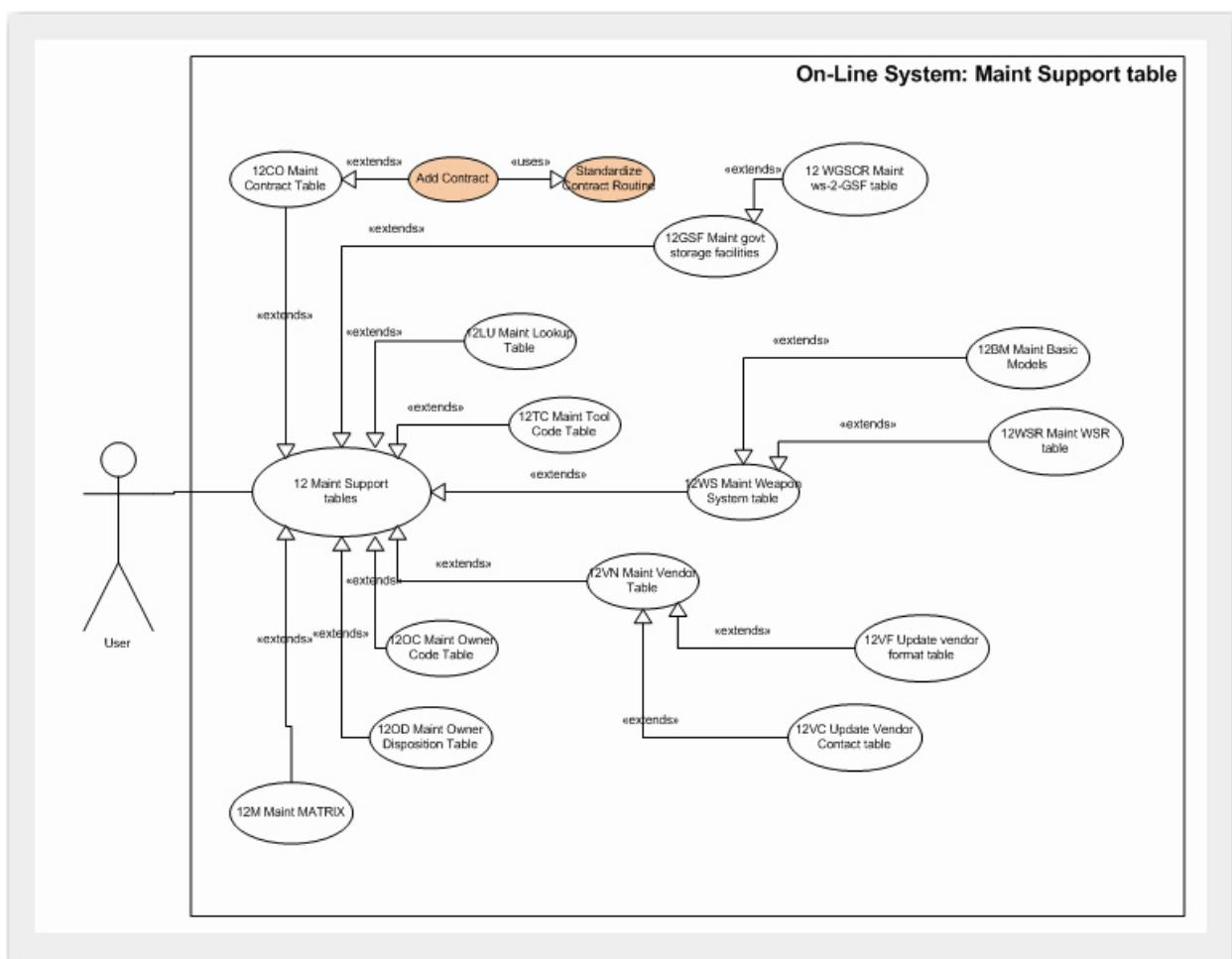


Figure 29: Support Functional Flow

Separate documents were created for each function. Refer to the table below to determine available subfunctions.

Document Name

04-11 Maint Vendor.doc

04-11-04 Maint Vendor Contacts.doc

04-11-04-01 Modify Vendor Contacts.doc

04-11-04-02 Add Vendor Contacts.doc

04-11-04-03 Delete Vendor Contacts.doc

04-11-03 Delete Vendor.doc

04-11-02 Add Vendor.doc

04-11-01 Modify Vendor.doc

04-11-00 Maint Vendor Format.doc

04-11-00-01 Modify Vendor Format. doc

Document Name

04-11-08 Maint Government Storage Facility.doc

04-11-08-01 Modify Government Storage Facility.doc

04-11-08-02 Add Government Storage Facility.doc

04-11-08-03 Delete Government Storage Facility.doc

04-11-08-04 Maint Govt Storage Facility Weapon Sys Assoc.doc

04-11-08-05 Modify Govt Storage Facility Weapon Sys Assoc.doc

04-11-08-06 Add Govt Storage Facility Weapon Sys Assoc.doc

04-11-08-07 Delete Govt Storage Facility Weapon Sys Assoc.doc

04-11-05 Maint Weapon System.doc

04-11-05-01 Modify Weapon System.doc

04-11-00-02 Add Vendor Format.doc	04-11-05-02 Add Weapon System.doc
04-11-00-03 Delete Vendor Format.doc	04-11-05-03 Delete Weapon System.doc
04-11-03 Maint Contract.doc	04-11-05-04 Maintain WS Reqmt Associations.doc
04-11-03-01 Modify Contracts.doc	04-11-05-05 Modify WS Reqmt Associations.doc
04-11-03-02 Add Contract.doc	04-11-05-06 Add WS Reqmt Associations.doc
04-11-03-03 Delete Contracts.doc	04-11-05-07 Delete WS Reqmt Associations.doc
04-11-07 Maint Tool Codes.doc	04-11-01 Maint Basic Models.doc
04-11-07-01 Modify Tool Code.doc	04-11-01-01 Modify Basic Models.doc
04-11-07-02 Add Tool Code.doc	04-11-01-02 Add Basic Models.doc
04-11-07-03 Delete Tool Code.doc	04-11-01-02 Delete Basic Models.doc

Table 10 Application Administration Sub-function Documents

4.12. eTIMS System Administration Subsystem

The System Administration Subsystem allows the users with the appropriate access level to manage system tables.

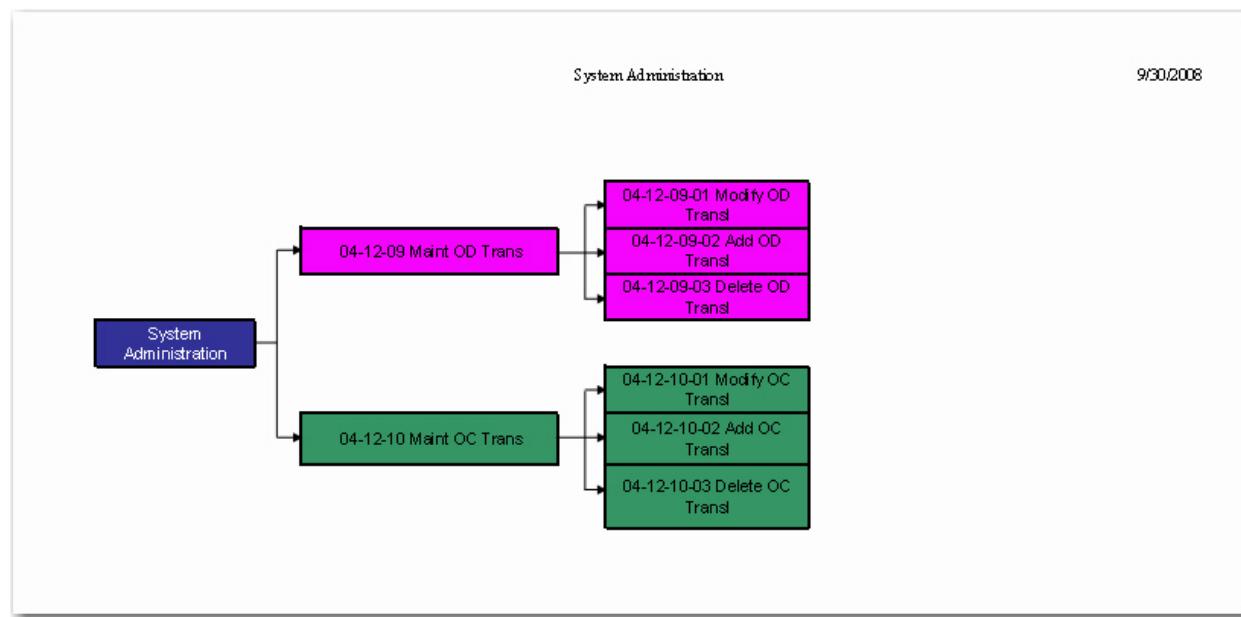


Figure 30 System Administration Subsystem Functional Flow Diagram

Separate documents were created for each function. Refer to the table below to determine available subfunctions.

Document Name

- 04-12-09 Maint OD Trans.doc
- 04-12-09-01 Modify OD Transl.doc
- 04-12-09-02 Add OD Transl.doc
- 04-12-09-03 Delete OD Transl.doc

Document Name

- 04-12-10 Maint OC Trans.doc
- 04-12-10-01 Modify OC Transl.doc
- 04-12-10-02 Add OC Transl.doc
- 04-12-10-03 Delete OC Transl.doc

Table 11 System Administration Sub-function Documents

4.13. eTIMS Disposition Document

DISPOSITION PROCESSING OVERVIEW

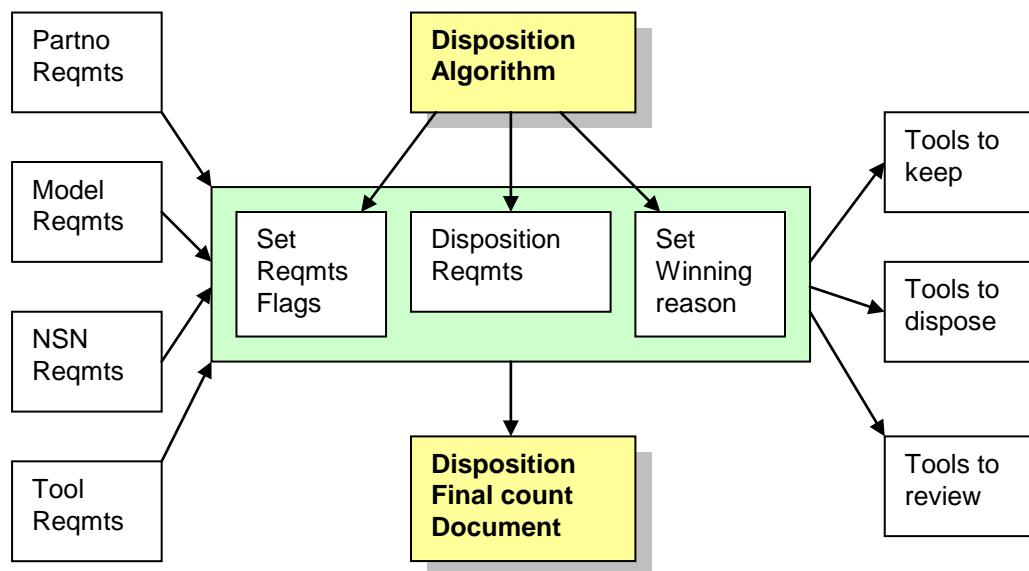


Figure 31 Disposition Process Overview

The purpose of dispositioning tooling is to create recommendations for Special Tooling after the prime vendor no longer has a need for the Tooling under the contract it was purchased for.

The eTIMS Tooling Dispositioning Process is the automated decision-making process where Past Requirements and Future Estimates for consumption/repair against Part numbers, NSNs and/or Tooling are gathered and applied to the Tool/Dup level.

The input into the Dispositioning Process is (1) the Disposition Algorithm (2) ETIMS and (3) various [Disposition Requirement Files](#) that are generally obtained from the prime vendor and NADEP/NAVICP/PMA representatives.

The Dispositioning Process for a given weapon system is defined in a J.A.D (Joint Application Development) session led by Lexes personnel and attended by government and prime vendor representatives. Subsequent sessions expand and refine the process.

The goal of these sessions is to produce a “[Disposition Algorithm](#)” document which an eTIMS analyst maintains. This document defines what files will be used to set which Disposition Requirements Flags, in what priority the flags will be set, what constitutes dispositional tooling and where to ship tools. See sample below.

After all Requirements files are received and logged into the Disposition Algorithm Document, the Disposition Process starts.

First, using the Disposition Algorithm, the requirements files are matched against eTIMS and [Disposition Requirement Flags](#) are set. Requirement flags can be set at the

Part Number and/or the Basic Tool level.

Next Disposition uses the eTIMS Part Number BOM and Part/Tool and Tool/Tool linkages to explode the Disposition Requirement flags just set to all levels.

Lastly the “winning reason” recommendation is set at the individual Tool/Dup level. Recommendations to keep or dispose of specific active tools are made and can be either:

- retain this Tool for anticipated future use, or
- dispose of this Tool no future use expected, or
- manual review of this Tool is necessary.

This manual defines the various disposition processes that can be used. A given weapon system may or may not use all of the methods discussed here. Details for a specific Weapon System will be contained in a weapon system specific section.

No dispositioning is done or is planned to be done on STE.

Requirements files

Requirements files can be 1 of 3 types: Keep files, Dispose files or Review files. **Keep Requirements** files are typically a list of parts, Tools, NIINs, Models, etc, that have either have a history of use or an anticipated future use. **Dispose Requirements** files are lists of things the government doesn't want to keep. **Review Requirements** files are parameters or lists that will create a list a Tooling to be manually reviewed and decided upon.

Each Requirement File will be assigned a **Disposition Flag** (for example Keep_SPR) and conversely, a Disposition flag can have zero to many Requirements files. Requirements file flags can operate at several levels such as NSN, Model, Part Number, Tooling Id, etc. Requirement flags are assigned and "pushed down" to lower data levels. All Requirements Files try to resolve down to the Tool/Dup level. If a Requirements File finds no matches, a Reject file will be created. Requirement Files match to part data or tooling data, each covered separately.

- Part Number match files
- NIIN match files
- Model code match files
- Tool-dup match files
- Tool-id match files
- Tool code match files

Disposition Processing

The Disposition Process is different for each weapon system, although the Basic logic is the same throughout. **Dispositioning does not cross weapon systems** but is confined

to the parts and Tooling within the given weapon system. Tooling and parts common to more than one weapon system can receive multiple disposition recommendations based on the algorithms of each weapon system. Disposition Processing has several logical processing sections.

There is a [Pre-disposition Setup](#) step that removes the old Basic Tool requirement flags (for the current weapon system only). They will be completely reset using the most current disposition logic.

It also revisits the [Manual Disposition Over-ride flags](#). Manual disposition over-rides usually have the ultimate say over what happens to a Tool. Regardless of what the disposition process recommends eTIMS will take the manual over-ride as the final say over what to do with a Tool.

The first Disposition Processing step is to match the Disposition Requirement Files against the eTIMS tables using the match logic outlined in the Disposition Algorithm and set the appropriate Disposition Requirement Flags. See [Requirements files](#).

The second step is to explode the Basic Parts flags to the sub-assembly Part Number levels. See [PART NUMBER EXPLOSIONS](#)

The third step is to copy all the Part Numbers Disposition Flags to the attached Basic Tools.

Fourth, all the Basic Tool disposition flags are set, disposition finds the master Tooling required by those Basic Tools and copies them to the master Tool's Basic Tool requirements.

Last disposition determines a "winning reason" at the Tool Dup level for [Dispositional Tools](#). The term Dispositional Tooling was coined to provide an easy handle for discussing the Tools that would receive a disposition recommendation. The definition of what Tools qualify for disposition recommendations changes with time. For some runs, for example, stored Tools may be dispositioned and for other runs stored Tooling is strictly off limits.

Disposition Output

Outputs of the Dispositioning Process include:

- Requirement Flags set at the Tool-id level
- A Recommended Disposition assigned to each dispositional Tool/Dup,
- The "winning reason" why Tool/Dup was recommended to be kept,
- The recommendation of where to ship Tools recommended to be kept,
- Tooling recommended to be disposed of, may receive dispose instructions,

- depending upon the DMIL codes available
- The Disposition Counts document reflecting Requirement Flag Tool counts, and
 - Various disposition export files and reports sent to the weapon system's PMA and/or the prime vendor.

Dispositional Count Document

Disposition Count Analysis Document is compiled by the programmer based on the load update numbers and reviewed by the analyst. A new sheet is created for each load, listing how many records were received, how many were flagged as possible problems and how many were rejected, etc.

Save Files.

FIELD NAME	LENGTH
TOOL-ID	32
Dup-SEQ	8
DISP-FLAGS-SET*	VARIES
One column for each disposition flag used in the weapon system.	
WINNING REASON	3
CURR-CONTRACT	25
CURR-CAGE	6
SHIP-TO	20
SCHED-ID	10
STATUS	20
Part-ATTACHED-IND	1
TOOLS-REFER-IND	1
OWNER-CODE	20
Cost (fab-hrs*rate) + fab-amt	10
ORIGINAL TOOL-ID	32
TDKEY	12

*EXAMPLE of WSR flags for active weapon system.

Flags								
KTM	DTC	ASO	DLA	MDI	CC	RAFUP	RAFSP	CST

Dispose & Review Files.

FIELD NAME	LENGTH
TOOL-ID	32
Dup-SEQ	8
CURR-CONTRACT	25
CURR-CAGE	6
SCHED-ID	10
STATUS	20
Part-ATTACHED-IND	1
TOOLS-REFER-IND	1
OWNER-CODE	20
ORIGINAL TOOL-ID	32
TDKEY	12

PART PROCESSING

There are 3 types of Part oriented Disposition Requirements files and they all set the Disposition Flags at the Basic Part level (PDR). These files are:

- [Model MATCH FILES](#)
- [Part Number MATCH FILES](#)
- [NIIN MATCH FILES](#)

MODEL MATCH FILES

The purpose of Model Code Requirements is to flag all tooling associated with a given model(s). To do this, disposition goes thru the Part-2-NHA Model-effectivities looking for matching Model codes. It is self-evident that *BOM explosions do not need to be done for Model match requirements*.

There are 2 types of Model Match requirements files that can be done in disposition:
Simple Model Match and Model UNIQUE Match.

Simple Model Match

Flag all the model “XXX” tools. Disposition flags all Basic Parts that have a model-effectivity under Model “XXX”. Then it flags the Basic Tools that are associated with the flagged Basic Parts via Part-2-Tool & Tool-2-Tool linkages.

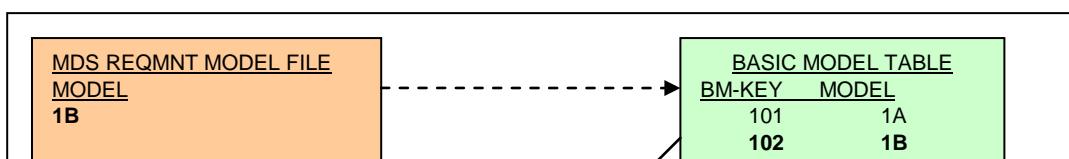
Step 1. First match the model to the Basic Model (BM) table to find the BM_KEY of the model.

Step 2. Use the BM_KEY to find the Model-EFFECTIVITY (ME) linkage records that have a matching ME_BM_KEY.

Step 3. Use the ME_PP_key to match to the PART-2-Part (PP) table and flag every associated BP_KEY found → every PP_NLA_BP_KEY & PP_BASE-BP-KEY is used to insert a PDR record using the WPR_KEY of the Requirement Flag of the file being processed.

Step 4. Use the WPR_BP_KEYS to find the Part-2-Tool linkages to the Basic Tools and flag all the Basic Tools (WTR) associated with the flagged Basic Parts.

Step 5. Use the WTR_BT_KEYS to find the Tool-2-Tool linkages to find required reference tools. Insert WTR-flags for the reference tools.



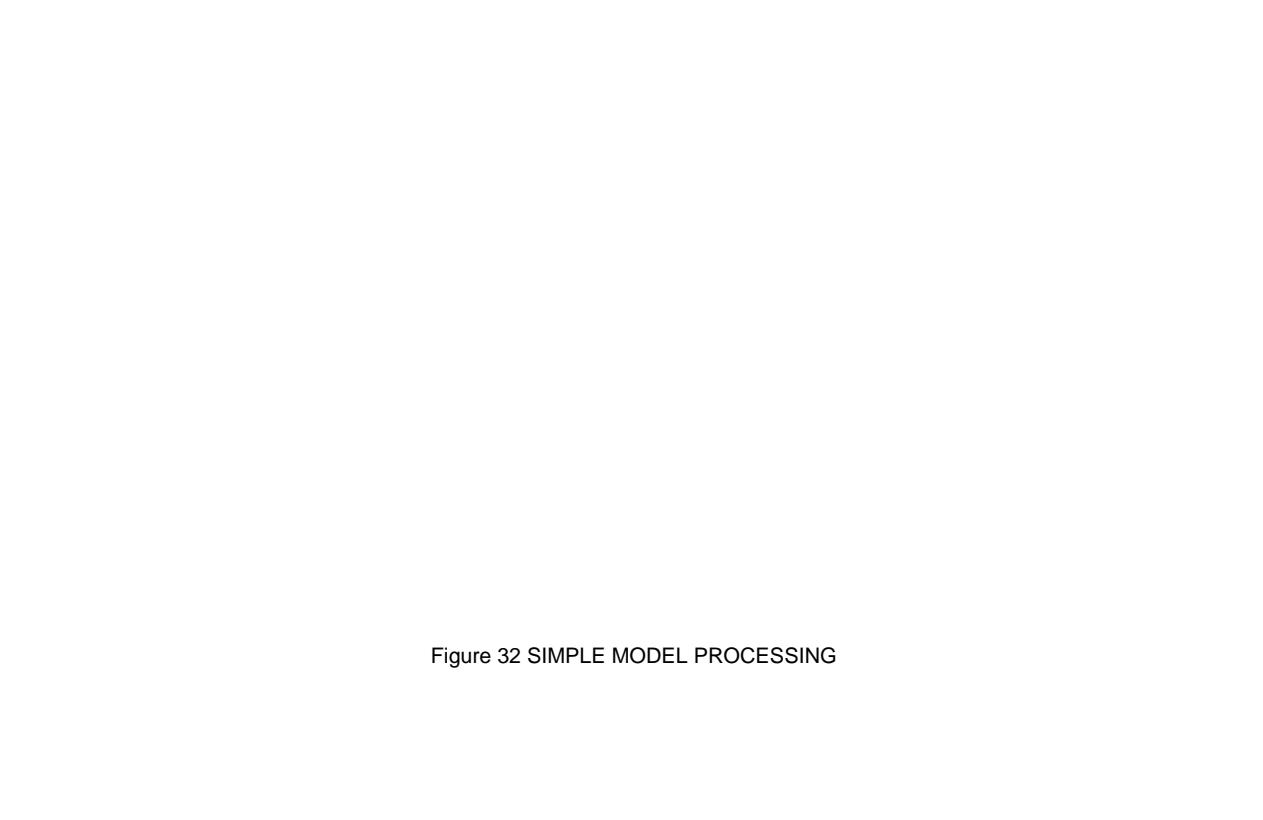


Figure 32 SIMPLE MODEL PROCESSING

Model Unique

FLag tools associated with Model “XXX” **only**. Disposition flags all Basic Parts & Basic tools that same as Simple Model match. Then it re-examines the Basic Tools flags and ***removes the flag if the Tool is linked to a Part that is linked to any other Model within the weapon system.***

Step 1. First match the model to the Basic Model (BM) table to find the BM_KEY of the model. Same as Simple Model step.

Step 2. Use the BM_KEY to find the Model-EFFECTIVITY (ME) linkage records that have a matching ME_BM_KEY. Same as Simple Model step.

Step 3. Use the ME_PP_key to match to the PART-2-Part (PP) table and flag every associated BP_KEY found → every PP_NLA_BP_KEY & PP_BASE-BP-KEY is used to insert a PDR record using the WPR_KEY of the Requirement Flag of the file being processed. To this point, the Same as Simple Model step, but now there is an additional step where **disposition REMOVES the PDR records where the BP_KEY has a Model within the current weapon system, other than the one being**

flagged.

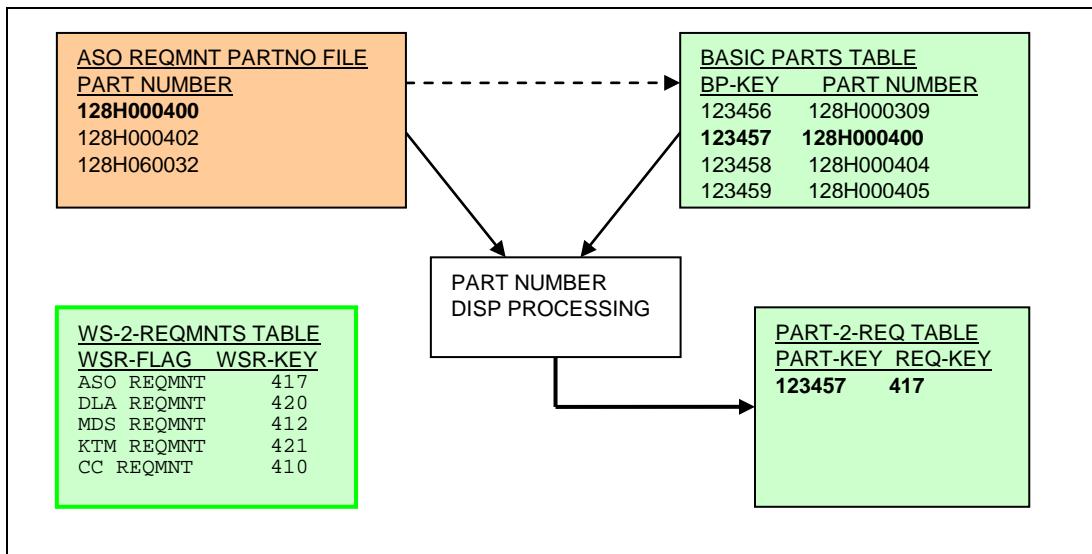
Step 4. Use the WPR_BP_KEYS to find the Part-2-Tool linkages to the Basic Tools and flag all the Basic Tools (WTR) associated with the flagged Basic Parts.

Step 5. Use the WTR_BT_KEYS to find the Tool-2-Tool linkages to find required reference tools. Insert WTR-flags for the reference tools.

PART NUMBER MATCH

There are 2 types of Part Number match files that can occur: match the Part to the eTIMS Part Number and set the flag, or match a “partial” Part Number to eTIMS Part Numbers and set the flag for each Basic Part that matches the partial Part Number. Partial part number matches are against the leading characters of the BP_PART_NUMBER.

Very Simple logic, match the incoming Disposition Requirement Part Number against eTIMS Basic Parts table BP_PART_NO. Once a full or partial matching Part Number has been found, an insert is made in the PDR table using the BP_KEY just found and the WPR_KEY of the Requirement Flag of the file being processed.



NIIN MATCH FILES

All NIIN files match the full 9 characters of the NIIN. NIIN files go thru the NIIN to get to the part numbers to hang the requirements flags. The exact same logic here is used as the Part Number logic; with a step added to get to the Part Number BP_KEY. Disposition must match on the Basic NSN table to find the Part Number key.

Match the incoming Disposition Requirement's NIIN against eTIMS Basic NSN table BN_NSN_NIIN. Once a match is found, disposition uses the BN_KEY to find Part Numbers linked to that NIIN. Disposition uses the Part_CAGE table to do this, inserting 1 PDR record for each PC_BP_KEY found (using the WPR_KEY of the Requirement Flag of the file being processed, of course).

DLA Requirements Algorithm

Since the algorithm for establishing a DLA requirements flag is somewhat complex, we will define it here.
IF ACQ Advice Code = "V" or "T" or "Y"

Then ignore DLA record
Else IF consumpt > 1,
Or IF stock lvl > zero,
Or IF Date of last demand was within past five years
Then set DLA requirements at NIIN/Part level.

PART NUMBER EXPLOSIONS

After all the part number match programs have completed, **non-model requirement flags** are copied from assembly part number to all sub-assembly part numbers with the BOM of the assembly. Model and effectivity are not taken into effect during this processing.

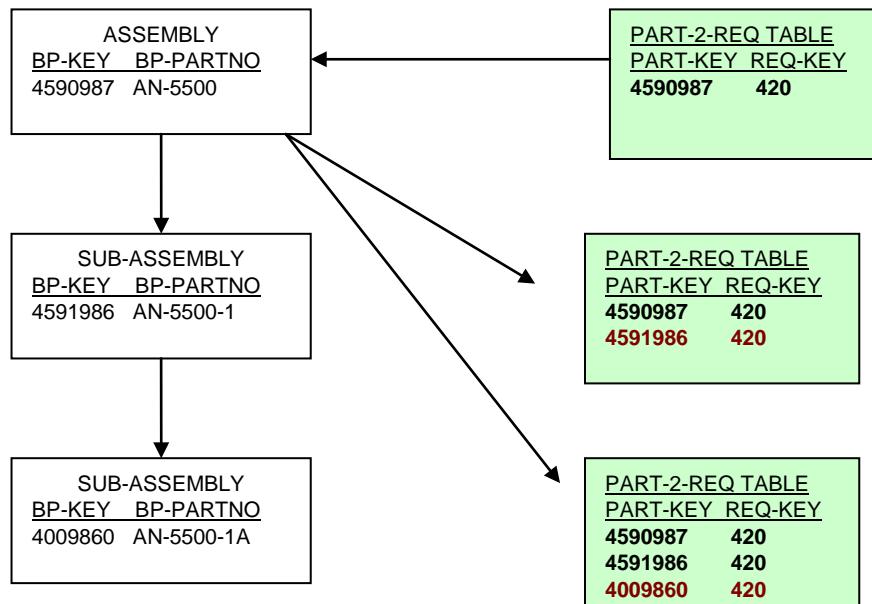


Figure 33 BOM EXPLOSION

COPY PART REQUIREMENTS TO BASIC TOOL

After all requirements flags at the part number level have been set that can possibly be set, the requirements set for a given part number are copied to any and all linked Tool-ids. **This of course excludes Unique Model Flags.**

TOOL PROCESSING

There are 3 types of Tool oriented Disposition Requirements files and they all set the Disposition Flags at the Basic Tool level (TDR). These files are:

- Tool Dup MATCH FILES
- TOOL-ID MATCH FILES
- Tool Code MATCH FILES.

Tool DUP MATCH FILES

These matches are to a specific Tool Dup. Someone, somewhere knows that this specific Tool must be kept. These types of Disposition Requirements files are rarely used.

Tool Dup matches can either be by serial number or by Tool-ID and Dup number and is different from the Tool-id match in that it sets both the flags at the Basic Tool level and at the Tool Dup level.

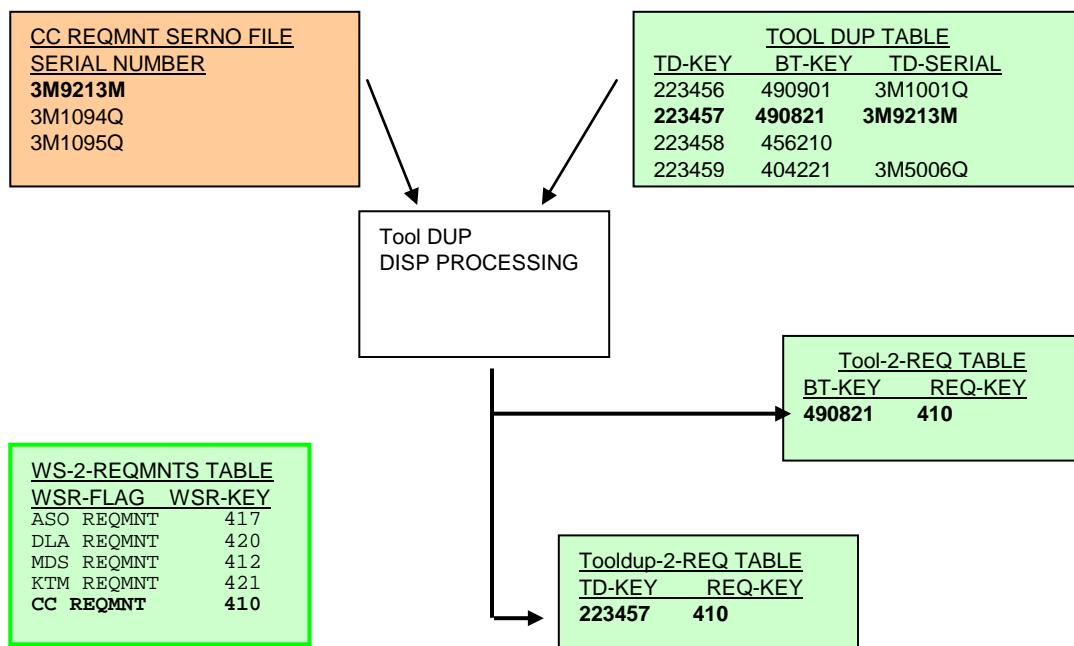


Figure 34 Tool dup disposition processing

Tool-ID MATCH FILES

These matches are to a Tool-id level. Tool-id matches can either be by full Tool-id or partial Tool-id and only sets the flag at the Basic Tool level.

It matches to the Tool-id, full or partial, and adds to the TDR table.

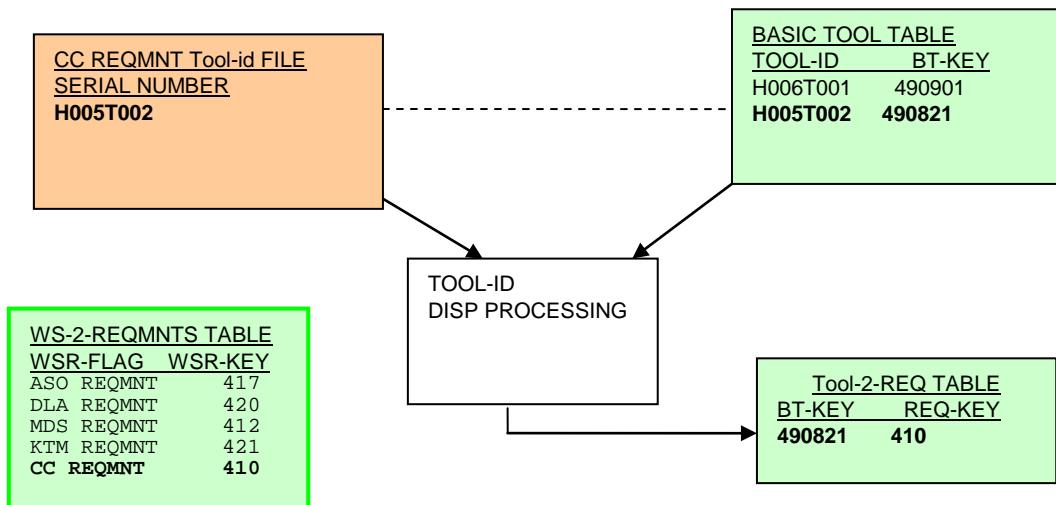


Figure 35 Tool-id processing

Tool CODE MATCH FILES

Tool Code matches are almost identical to partial Tool-id matches, with the exception that partial Tool-id match to the first X positions of the BT_TOOL-ID, and Tool Code matches match only to the embedded Tool Code within the Tool-id. Tool Codes locations vary by prime vendor and in order to find the exact location within the Tool-id, disposition must access the vendor format table looking for the Tool Code entry and use those positions within the Tool-id to match. Just as with the Tool-id match, Tool Code matches only sets the flag at the Basic Tool level.

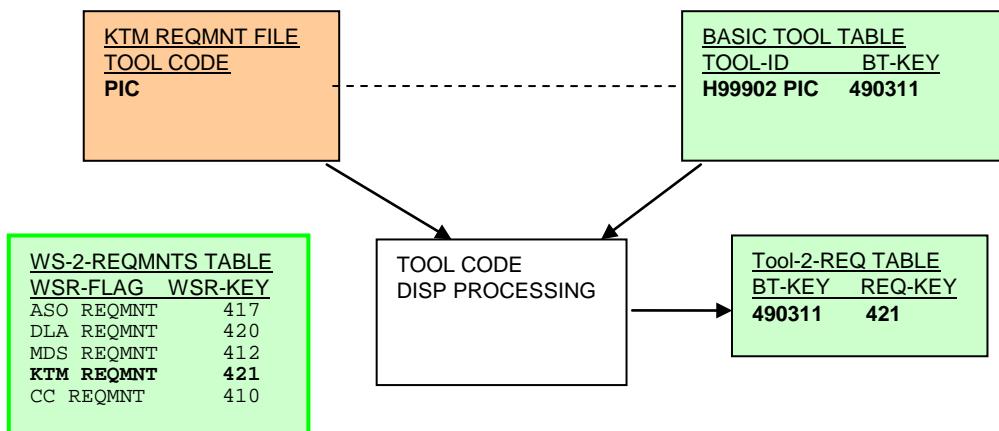
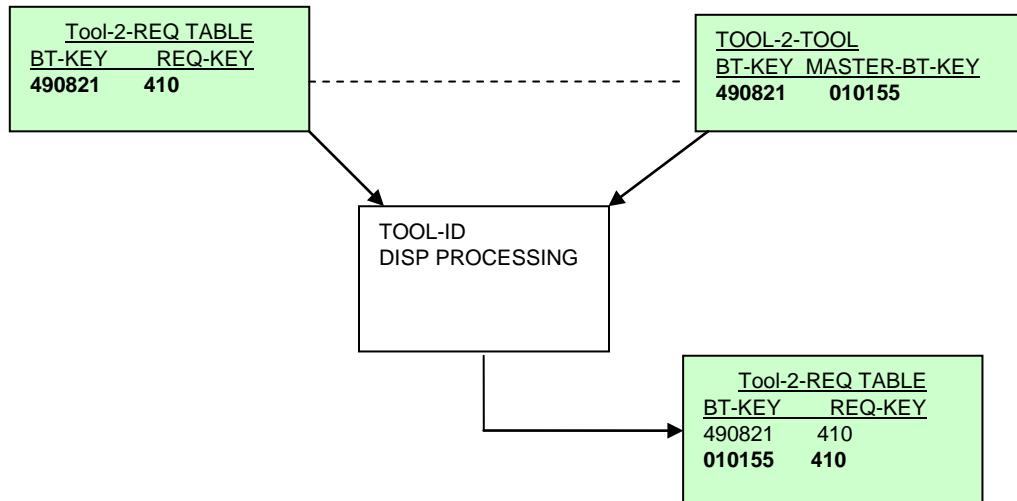


Figure 36 TOOL CODE PROCESSING

FIND BASIC TOOL'S MASTER TOOLS

After all Basic Tool level Disposition Requirement Flags have been set from both the part requirements files and the tool requirements files, disposition must find tools that need master tools and extend those requirements to the master tooling. **This would exclude Unique Model Flags.**

To do this it looks at the flags in the TDR table and matches them against the Tool-2-Tool table, inserting additional flags for the master tooling as needed.



Disposition Manual Review File

The Manual review files or the "Look-at-me" files are run with the normal Disposition Requirements files. The kicker is this step between all basic Tool flags being set and setting a winning reason at the Tool-Dup level.

ETIMS disposition has a step where it looks at every BT-KEY that has been given a Manual Review flag. If there are other Disposition Flags for that BT_KEY, then the Manual Review flag is removed. If there are no other Disposition flags set for that BT_key then the manual review flag stays in place.

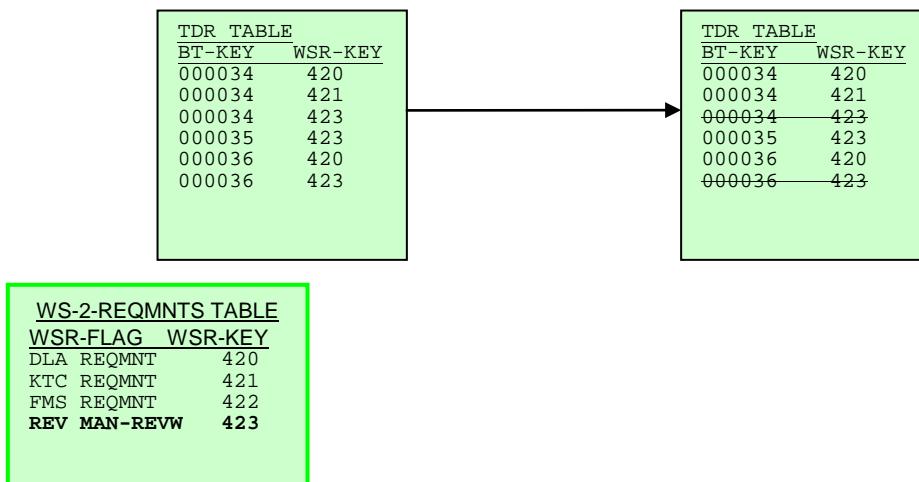


Figure 37 MANUAL REVIEW FLAGS

HIGH COST

The High Cost requirement flag executes exactly like the tool/dup flag, starting with looking at the tool/dup level, and adding Basic Tool flag if the tool/dup meets the requirement. Since High Cost is a manual review requirement, it can be assigned to a manual review requirement.

DISPOSITION TO TOOL/DUP LEVEL

Tool-id Requirements are prioritized and a "winning reason" is assigned to Disposable Tool under the Tool-id. At this time the recommended ship-to storage site will be assigned.

If the dup is linked to a "Dispose" requirement file (only) or if the maximum number of keep copies is already reached, the Tool/Dup will be recommended to be disposed.

If there is no discernable reason for keeping a Tool/Dup AND the Tool is an OutsTool, the Disposition recommendation will be Unknown.

If there is no discernable reason for keeping a Tool/Dup AND the Tool is NOT an OutsTool, the Disposition recommendation will be Dispose.

PRE-DISPOSITION UPDATE PROCESSING

REQUIREMENTS		match	UPDATE						
FLAG	REQUIREMENT		Part name	Part DMI L	Part cost	Add Part CAGE	NSN FSC	NSN COG	NSN AAC
ASO	ASO_3M	PN				Y	Y	Y	
ASO	ASO_5yr_Demand	PN	Y	Y		Y	Y	Y	Y
ASO	ASO_PN_BUYS ?	PN					Y	Y	
NGS	NGC_5yrHist_Dnumbers	PN			Y				
DLA	DLA_E2	NIIN			Y		Y	Y	Y
ASO	ASO_STRAT_E2	PN	Y	Y		Y	Y	Y	Y

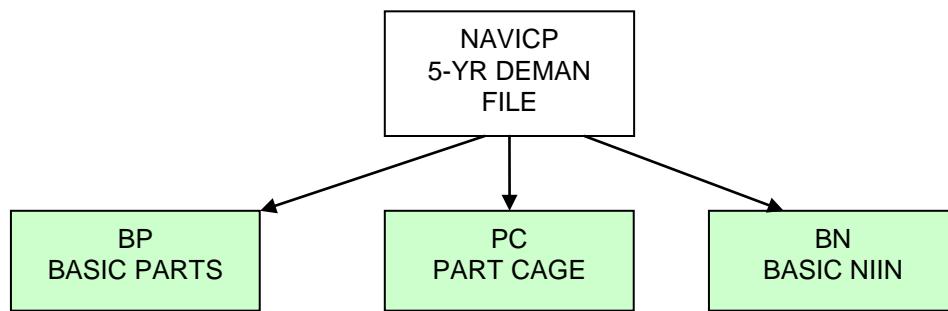
ETIMS is instituting new Pre-Disposition Procedures beginning with the E-2 weapon system. Some of these processing should already be in place for the older weapon systems and some of the processing is new. If the files noted above are available, they need to be included in the Pre-dispositioning update process.

ETIMS will match Requirements files against eTIMS and pickup what data the files have to offer and store them on eTIMS. The processing logic is simple: match on Part Number/NIIN and update fields flagged. Except for D-Mil D-Mil has its own replace logic that is explained in the section [DMIL Replacement Logic](#).

The STRAT & Battle Creek file processing is not changed. The change to this processing is to load the Part-CAGE updates prior to running the STRAT/BATTLECREEK processes.

After the files in the above table have processed, then the Dmil code updated in the Basic Parts tables will need to be carried down to the Part's tooling and the Tool's tooling.

Government 5 YEAR DEMAND FILE



Pre-processing Requirements

Sort by Part Number/NIIN, eliminating duplicate records.

Input layout

FIELD NAME	START	LENGTH
CAGE	01	520
PARTNO	11	20
NIIN	31	9
COG	41	2
FSC	51	4
NAME	61	15
DMIL	86	1
AAC	96	1

Update Pass 1

- Match Part Number to BP-Part-no.
 - If a match is found and if the BP_PART_NOUN is blank, replace the BP_PART_NOUN with the incoming part name.
 - If a match is found, replace the current BP_LU_DMIL_CD_KEY using the [DMIL Replacement Logic](#). Replace the BP_LU_DMIL_CD_KEY with the corresponding LU-key found under CATEGORY= “DMIL” and the incoming DMIL code.
 - **Do not update the BP_UPDATE fields unless a change in the data has actually occurred!**

BP Mapping

BP PART NO found

BP_PART_NOUN	Part Name
BP_LU_DMIL_CD_KEY	LU_KEY after lookup to LU-table "DMIL" / DMIL code
BP_UPDATE_DT	current date
BP_UPDATE_USER_ID	userid

Update Pass 2

- Match PARTNO to BP_PARTNO.
- If a match is found, add the CAGE using the BP_KEY to the PC-Part-CAGE table

Update Pass 3

- Match NIIN to BN_NSN_NIIN_NO.
- If a match is found and if the BN_FSC, BN-AAC OR BN-COG are blank, replace with incoming values.
- Do not update the BN_UPDATE fields unless a change in the data has actually occurred!

BN Mapping

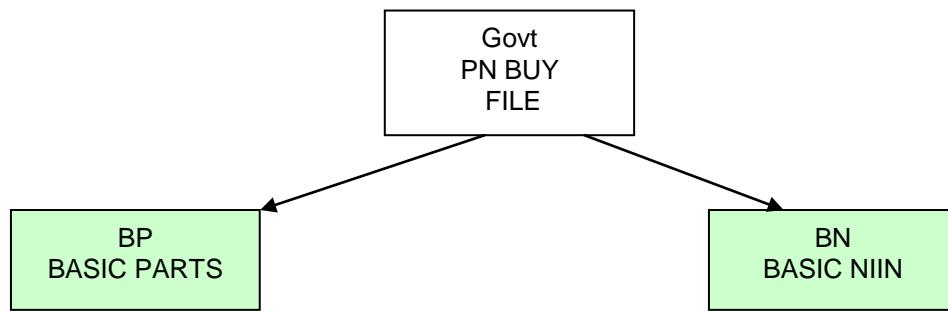
BN_NSN_NIIN_NO	
BN_NSN_FSC_NO	FSC
BN_ACQ_ADV_CD	AAC
BN_NSN_COG_NO	COG
BN_UPDATE_DT	current date
BN_UPDATE_USER_ID	userid

Expected Output

Total report:

- total number of records incoming
- total BP records updated
- total BN records updated

Government PN BUYS



Pre-processing Requirements

Sort by Part Number/NIIN, eliminating duplicate records.

Input layout

FIELD NAME	START	LENGTH
NIIN	01	09
COG	12	2
FSC	16	4
Part Number	20	20
NOMENCL	42	25

Update Pass 1

- Match Part Number to BP-Part-no.
- If a match is found and if the BP_PART_NOUN is blank, replace the BP_PART_NOUN with the incoming part name.
- Do not update the BP_UPDATE fields unless a change in the data has actually occurred!

BP Mapping

BP_PART_NO	found
BP_PART_NOUN	Part Name
BP_UPDATE_DT	current date
BP_UPDATE_USER_ID	E2-5yrDem-Disposition

Update Pass 2

- Match NIIN to BN_NSN_NIIN_NO.
- If a match is found and if the BN_FSC OR BN-COG are blank, replace with incoming values.

- Do not update the BN_UPDATE fields unless a change in the data has actually occurred!

BN Mapping

BN_NSN_NIIN_NO	
BN_NSN_FSC_NO	FSC
BN_NSN_COG_NO	COG
BN_UPDATE_DT	current date
BN_UPDATE_USER_ID	E2-5yrNavy-Disposition

Expected Output

Total report:

- total number of records incoming
- total BP records updated
- total BN records updated

5 YEAR HISTORY DESIGN FILE

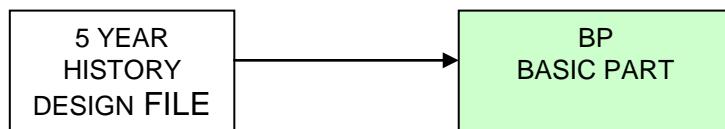


Figure 38 FIVE YEAR HISTORY DESIGN FLOWCHART

Pre-processing Requirements

Sort by Part Number, eliminating duplicate records.

File Layout

FIELD NAME	START	LENGTH
PARTNO	29	20
UNIT PRICE	88	13
NIIN	10	9
FSC	3	4
PNAME	56	20

Update

- Match Part Number to BP-Part-no.
- If a match is found and if the BP_PART_NOUN is blank, replace the BP_PART_NOUN with the incoming part name.
- If a match is found and if the BP_UNIT_COST is zero or null, replace the BP-UNIT-COST with the incoming part cost.

Do not update the BP_UPDATE fields unless a change in the data has actually occurred!

BP Mapping

BP_PART_NO	
BP_PART_NOUN	Part Name
BP_UNIT_COST	PART COST
BP_UPDATE_DT	current date
BP_UPDATE_USER_ID	E2-5yrDem- Disposition

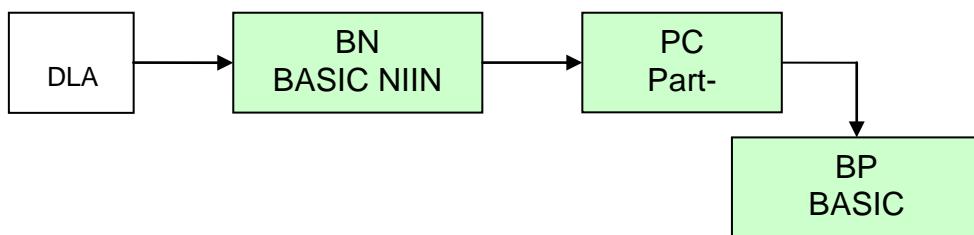
Expected Output

Total report:

- total number of records incoming
- total records updated

DLA FILE

This file has always been used in the past for Tool disposition purposes only. We are going to start picking up information from the file to be used in eTIMS online.



Pre-processing Requirements

Sort by NIIN.

File Layout

FIELD NAME	START	LENGTH
------------	-------	--------

FIELD NAME	START	LENGTH
FSC	01	04
NIIN	07	09
NOMENCLATUREr	17	01
ACQ ADVICE CDE not T,V,Y	39	1
FILLER	16	1
QTRLY FCST >zero	47	09
FILLER	26	01
STOCK OBJECTIVE >zero	60	11
FILLER	36	01
DATE-DEMAND MM/DD/YYYY	76	10

Validity Requirements

Match to **Basic NSN** (BN) by NIIN. If not found, skip record. If found, continue to update.

Update

- If BN_ACQ_ADV_CD is blank, replace with incoming AAC.
- If BN_NSN_FSC_NO is blank, replace with incoming FSC.
- Find corresponding **Basic Part** (BP) record via Part-CAGE linkages (there may be 0-m). If there are no linkages, skip the remainder of the processing.
- If BP_PART_NOUN is blank, replace with incoming Part-name.
- If BP_UNIT_COST is zero/null, or less than the incoming cost, replace with the incoming acq-cost.

Expected Output

Total report:

- total number of records incoming
 - total Tooling record updated
-

COPY DMIL CODE FROM BASIC PARTS TO BASIC TOOLS.

Retrieve all basic parts with DMIL codes. Find all tools attached to them. Compare the basic Tool's DMIL code priority with the new DMIL code and replace according to the DMIL replacement logic explained earlier.

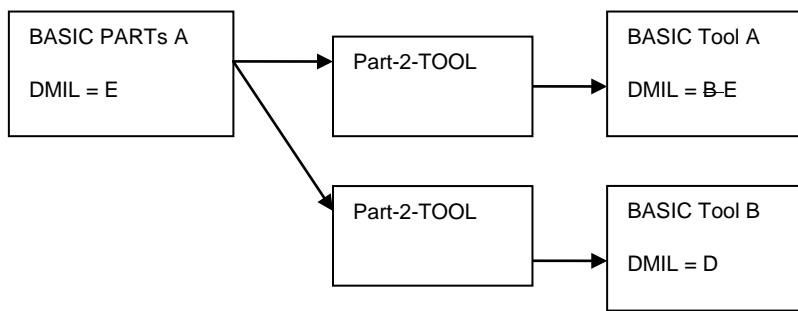


Figure 39 COPY DMIL CODES

DMIL REPLACEMENT LOGIC

A DMIL (Demilitarization) code tells the government how to dispose of the Part/Tool. The lower the priority code the more effort must be made in disposing of the item. eTIMS can replace a DMIL code with a DMIL code having a lower priority but not the opposite.

If the incoming record matches to a eTIMS record, say Basic Part and the BP_DMIL_CD is empty, replace with the incoming DMIL code.

If the incoming record matches to a eTIMS record, again Basic Part and the BP_DMIL_CD is NOT empty compare the priority numbers of the 2 DMIL codes contained within the LU table (under the LU-CATEGORY-CD of "DEMIL"). The DMIL priority code is located in the LU_ACTUAL_DISP_CD field.

If the BT_DMIL/BP_DMIL (being replaced) has a LOWER priority number than the incoming or replacing BP_DMIL, do not replace.

DMIL	A	DMIL not Required. Non-Munitions List	7
DMIL	B	DMIL not Required. Trade Sec Cntl req.	8
DMIL	C	Remove/Destroy installed Key Pts	4
DMIL	D	Total Destruction - melting,cutting, etc	1
DMIL	E	DMIL instructions by Dod DMIL Pgm Off	2
DMIL	F	DMIL instructions by Item/Tech Manager	3
DMIL	G	AEDA items - declass/remove marks/info	5
DMIL	P	Declass/remove marks/info - no AEDA	6
DMIL	Q	DMIL not required. USDOC items.	9

Figure 40 DMIL PRIORITY TABLE

STRAT & BATTLECREEK

There are 2 Standard Post-processing files that will need to be processed before every Disposition: the processing for the STRAT file and the BattleCreek file. Both of these files update part data and the linkages between Part Numbers and National Stock Numbers.

These processes have been documented elsewhere, but we will give an overview here.

The STRAT file is downloaded from the Mechanicsburg mainframe. RIC codes are

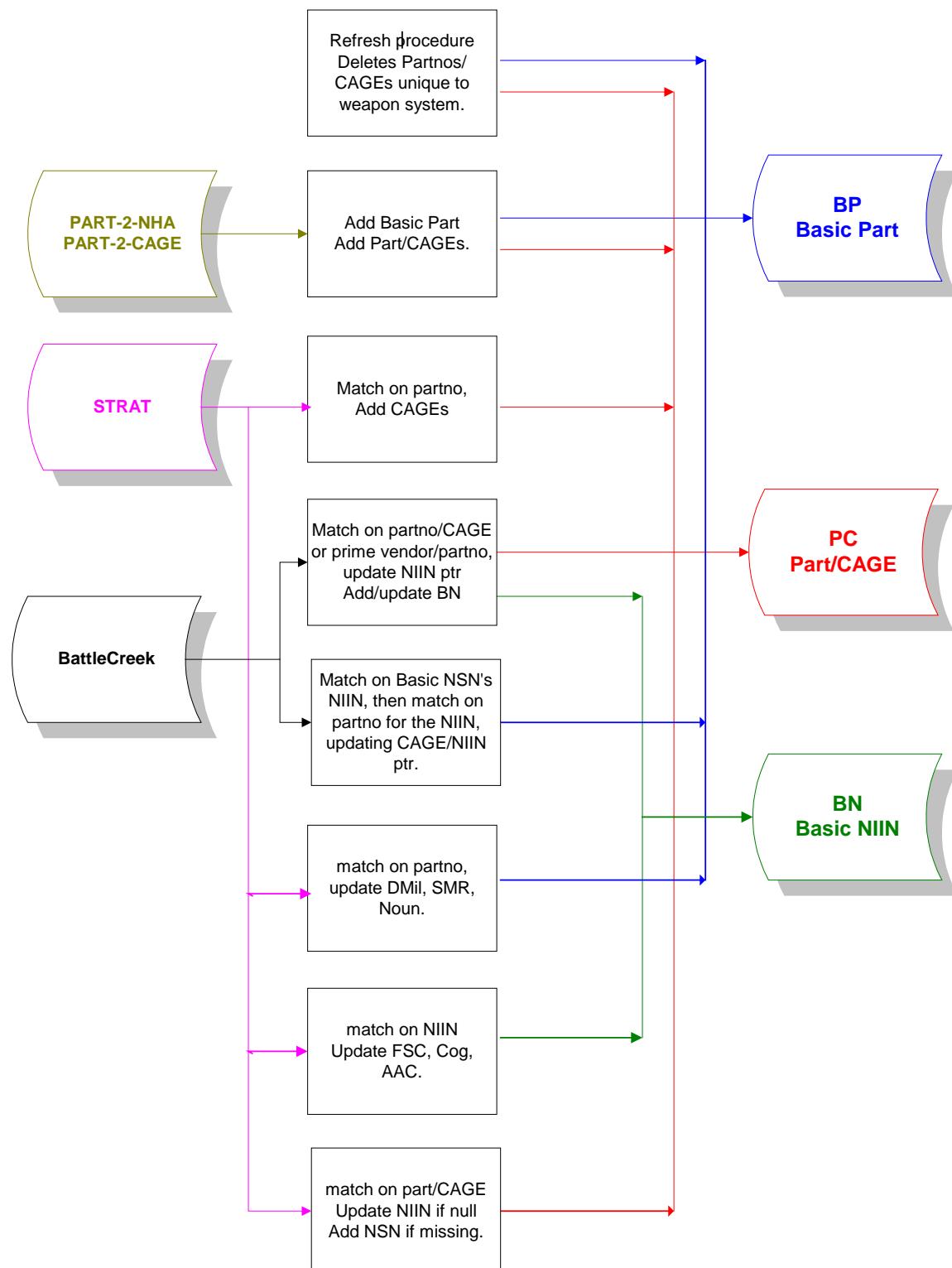
used to gather the data. One file is created for each RIC code. The files are sorted together, eliminating records that are 100% duplicates of each other. The resultant file is downloaded to the server.

STRAT Layout

FIELD NAME	START	LENGTH
NIIN	01	09
COG	10	02
FSC	12	04
NAME	16	19
FAM-GRP	35	04
FAM-REL	39	01
SOURCE	40	02
SMR	42	04
DMIL	46	01
ACQ-ADV-CD	47	01
REORDER / EOQ	48	14
CAGE	62	05
PARTNO	67	32
PROCURMT CODE	99	01
3M-REQ (bad data ignore)	100	01
STRAT-REQ (not = N)	101	01

Note-1

A list of **RIC codes** representing physical areas of the weapon system is given by Navy representatives to ETIMS. The Systems Administrator processed the RIC codes through a batch program resident on the Mechanicsburg mainframe. This process creates individual RIC code files that are sort/merged into one STRAT file. This file is downloaded to ETIMS where it is matched by part/CAGE, updating as needed.



4.14. eTIMS Data Refresh Document

LOAD REFRESH FILES INTO TEMP TABLES

There will be an estimated 9 files from LMC for the load/refresh. Load each into a temp-table for processing.

The files include but are not restricted to:

1. [TOOLING INFORMATION](#)
2. [TOOL-2-TOOL](#)
3. [DUP-2-DUP](#)
4. [PART-2-NHA](#)
5. [PART-2-TOOL](#)
6. [PART-2-CAGE](#)
7. [PART TYPE-VERSION-2-SHIP](#)
8. [STE](#)
9. [SUB-VENDOR FILES](#)

Tooling Information Files

- [VENDOR FILE](#)
- [STATUS TRANSLATION FILE](#)
- [DUP-2-DUP FILE](#)
- [TOOL INFORMATION](#)
- [TOOL-2-TOOL](#)

Vendor Files

The LMC Vendor File will need to have 2 one-time processes performed with it.

1. Load the file into a **Vendor Translation Table (VT)**.
2. Load the Vendor Translation table (VT) into the **current eTIMS Vendor table (VN)**.

The format of the incoming Vendor file is:

Field	Length	Type
Vendor Code	6	A/N
Vendor Name	86	A/N
Street Address	204	A/N
City	48	A/N
State	2	A/N
Zip	10	A/N
Country	35	A/N
CAGE	5	A/N

New table: Vendor Translation table (VT)

The logical key will be Prime-vendor and Vendor-code combined.

1. Load the **Vendor-Translation Table**.

2. Match the Vendor File to the Vendor Translation table (VT), matching on Prime Vendor and Vendor Code. Add record if missing.

THE LAYOUT OF THE VT TABLE IS:

Field	Length	Type
VT-PRIME-VENDOR	5	81755
VT-CAGE	5	A/N
VT-VENDOR-CD	6	A/N
VT-VENDOR Name	86	A/N
VT-ADDRESS	204	A/N
VT-City	48	A/N
VT-State	2	A/N
VT-Zip	10	A/N
VT-Country	35	A/N

Load eTIMS Vendor Table (VN)

LMC tooling data contains a 6-digit “vendor code” that must be translated to Government CAGE codes (whenever possible). To do this we will use the old Vendor Translation table to convert LMC Vendor Codes to eTIMS CAGE codes (VN-SITE-CD).

Match the just loaded Vendor Translation Table (VT) to eTIMS Vendor Table (VN).

1. We do not have a valid CAGE.

If VT_CAGE = BLANK, then

match **VT_VENDOR_CD** to VN_SITE_CD.

- a. If a match is found, update the name, address, city, state, zip and country fields, ONLY if those VN fields are blank.

- b. If a match is not found, add the entry **using the VT_VENDOR_CD as the value in both VN_SITE_CD & VN-SHORT-NAME.**

2. We do have a valid CAGE code.

If VT_CAGE is not = BLANK, then

match **VT_CAGE** to VN_SITE_CD.

- a. If a match is found, update the name, address, city, state, zip and country fields, ONLY if those VN fields are blank.

- b. If a match is not found, add the entry **using the VT_CAGE as the value in both VN_SITE_CD & VN-SHORT-NAME.**

STATUS TO STATUS TRANSLATION (STS)

LMC will be sending more status codes (owner disposition) than we really need. We have combined the 26 possible status codes to 7 status codes (see the following table).

This table will only be loaded ONLY once BUT will only be used again for Refreshes. It has no online value.

New Table: Status Translation Table

This new table will contain:

1. STS_VN_KEY 9(5) fk VALUE 4525.
2. STS_Old_Status X(8).
3. STS_Status_Desc X(25)
4. STS_TRANSLATE_TO x(2)
5. STS_OD_KEY 9(5).

LMC STATUS	LMC DESCRIPTION	TRANSLATE TO	EQUIVALENT eTIMS STATUS
Blank	Issued	AV	At Vendor
AT CAL	AT CALIBRATION LAB.	AV	At Vendor
AT QAC	IN QAC LAB.(FWA/ATI ONLY)	AV	At Vendor
AVAILABL	AVAILABLE FOR ISSUE FROM A CRIB,STORE,OR WHSE.	AV	At Vendor
CAL REQD	CALIBRATION REQUIRED BEFORE USE	AV	At Vendor, add to notes
CHCNTLVL	CHANGE OF RECORD CONTROL LEVEL.	AV	At Vendor
DECONTLD	DECONTROLLED.NO LONGER ACCOUNTABLE ASSET.RECORD ARCHIVED.	SC	Scrapped
DISPOSED	DISPOSITION INSTRUCTIONS HAVE BEEN COMPLIED WITH	SC	Scrapped
EXCESS	DECLARED EXCESS IN-HOUSE SCREENING BEGUN,USE HAS CEASED.	EX	Pending Disposition
ID ERROR	IDENTIFICATION ERROR TO BE CORRECTED		reject
IN TRANS	SCHEDULED TO BE OR IN PROCESS OF MOVING WITHIN AF PLANT#4.	AV	At Vendor
INFAB	STE/ST IN PROCESS OF BEING FABRICATED	AV	At Vendor
INSCREEN	SUBMITTED ON INV. SCH. OR PSR OR LIST FOR OPTION TOOLING	PD	Pending Disposition
ISSUED	ASSIGNED TO A RESP.DEPT OR ISSUED FROM A CRIB FOR USE.	AV	At Vendor
LD&D	LOST,DAMAGED,DESTROYED:RELIEF GRANTED OR CASE CLOSED	LD	Lost
LD&DL	LOST,DAMAGED,DESTROYED:RESPONS E REC'D, HELD LIABLE	LD	Lost
LD&DP	LOST,DAMAGED,DESTROYED:SUBM'D TO CUSTOMER, PENDING RESPONSE	LD	Lost
NON STAT	NOT STATUS QUALIFIED	SC	Scrapped
POREVIEW	FOR FWV CONTROL OF CLOSING PO'S - VALID FOR 12 MONTHS	AV	At Vendor
REJECTED	PART HAS BEEN REJECTED BY CERT LAB	AV	At Vendor
REPAIRRQ	REPAIR REQUIRED:OUT OF SERVICE	AV	At Vendor
REQUEST D	TOOL REQUEST TRANSMITTED AND ACCEPTED	AV	At Vendor
SHIP FWT	SHIPPED:TO BE RETURNED TO LFWC.PARS2 RETAINS RECORDS.	AV	At Vendor
SHIPPED	SHIPPED:PARS2 RECORD ARCHIVED	SH	shipped
STOCK	IN STOCKROOM OR CRIB.SYNONYMOUS WITH THE STATUS AVABL.	AV	At Vendor
STORED	INACTIVE PROPERTY RETAINED FOR FUTURE REQIREMENTS	AV	At Vendor
SURPLUS	DISPOSAL INSTRUCTIONS HAVE BEEN ISSUED	PD	Pending Disposition
UNLOCATE	70 LD&D REPORT SUBMITTED TO PROPERTY MANAGEMENT DEPT. 012-P	PL	Pending Lost

Tooling Information File (BT, TD, TM)

The Tooling Information file will add/update the F16 eTIMS tooling data. Certain translations will have to be done on the data prior to being able to find and update the record in eTIMS. Two files will be created that will both feed into the Update Matrix processing routines.

INPUT LAYOUT.

<u>Field</u>	<u>Length</u>	<u>Type</u>	<u>Notes</u>
Tool-id	8	A/N	
Tool Code	4	A/N	Maybe blank (production aides)
Description	40	A/N	
Ownership Code	1	A/N	
Status Code	8	A/N	OWNER DISP
Dollar Cost	15	D13.2	only for Govt or RT
Current CAGE/Vendor Code	6	A/N	
Acquisition Contract	16	A/N	
Current Contract	16	A/N	
Disposition Date	10	YYYY-MM-DD	
PARS2 Part Number	32	A/N	only used if no tool code
Item on which Used	32	A/N	only used if no tool code
Archive Flag	1	A/N	active or dispositioned - ties to status

Output expected

There are 3 things expected from the Tooling Information Update process:

1. Updated eTIMS data
2. Reject File, image of input record, with appropriate error messages.
3. Totals Report. Including but not restricted to:
 - a. TOTAL RECORDS PROCESSED
 - b. TOTAL RECORDS REJECTED
 - c. TOTAL TOOLS UPDATED
 - d. TOTAL TOOLS ADDED.

Pre-processing tasks.

The pre-processing tasks include validity checks against designated fields and conversion of specific LMC field values to eTIMS values and formatting the update record to feed into the Update Matrix.

There are 6 pre-update verification tasks needed to both prep the update record and prepare FK tables:

1. **Tool-id.** Left adjust, eliminating leading blanks. Cannot be all blanks or all zeros. If record contains all blanks or zeros as the value for Tool-id, reject record with message "invalid Tool-id" and write to Reject file.

2. **Status Translation table.** Using the incoming STATUS CODE (OWNER DISPOSITION) code, match to the STS table using 81755/4525 as vendor. Failure to find a matching STS code is a show-stopper for any given tooling record.

a. If a match is found, attach the STS_TRANSLATE_TO to the record to be used later.

b. If the OD code is not found, **reject the record** with the message "unknown' OD code (STS)'.

3. **Owner Disposition table.** Using the STS_TRANSLATE_TO code found above, match to the OD table using 81755/4525 as vendor. Failure to find a matching OD code is a show-stopper for any given tooling record.

a. If a match is found, attach the OD_key and OD_MATRIX_CD to the record to be used later.

b. If the OD code is not found, **reject the record** with the message "unknown' OD code'.

4. **Owner Code table.** Must be present within the OC table. Lookup entries in the OC-Table using 4525/81755 as the Vendor key.

a. If found attach the OC_LU-KEY value with the tooling record and continue processing.

b. If the owner code is blank, reject the record with the message 'no owner code'.

c. If the owner code is not found in the table, reject the record with the message "unknown ownership code".

5. **CAGE/Vendor-code.** Match the incoming vendor code against the [Vendor Translation Table](#), or the Vendor table, depending on the length of the vendor code coming in.

If the length is 6 digits, we may assume it is a vendor code and needs to be translated to a CAGE using the vendor translation table. If the length is only 5 digits, we may assume it is a CAGE and go directly against the vendor table.

a. **If the incoming CAGE/Vendor-code field is 6 digits,** match to the Vendor Translation table's VT-VENDOR-CD using 81755(4525) as the Prime Vendor Code.

- i. If a match is found and VT-CAGE = blank, do not replace the incoming CAGE, the vendor code should have been added to the vendor table during pre-processing.
- ii. If a match is found and VT-CAGE is not = blank, replace the record's Vendor-code with the value in VT_CAGE.
- iii. If a match is not found and the record has an inactive status, blank out the record's Vendor code.

- iv. If a match is not found and the record has an active status, place the incoming CAGE. Place the incoming CAGE/Vendor-code on a list to be worked by the analyst.
- b. **If the incoming CAGE/Vendor-code field is 5 digits**, match directly to Vendor Table (VN).
 - i. If a match is found, the data is good, continue processing.
 - ii. If a match is not found, reject the record, and place the incoming CAGE/Vendor-code on a list to be worked by the analyst.
- 6. **Contract table.** LMC will not send us a separate Contracts file, so we must examine the Tooling information file to pick-up and pre-update the eTIMS Contracts table (CO).
 - a. There are 2 contract fields within each Tooling Information record: Acquisition & Current.
 - b. From each contract field, remove dashes and slashes and left-adjust the contract number using the Standard Contract Adjusting Routine.
 - c. Match the current contract against the eTIMS Contract table (CO).
 - i. If a match is found, Examine the CO-WS-OWNER-CD-KEY. If not = 16 or null, update to null value. Pickup the CO-CO-KEY and attach to the record.
 - ii. If a match is not found, add to the CONTRACT table using values below, note the assigned CO-CO-KEY and attach to the Tooling record.

CO-CONTRACT-NO	= Edited contract number
CO-VN-KEY	= 4525
CO-WS-OWNER-CD	= 16
CO-UPDATE-DATE	= Sys-date
CO-UPDATE-USER	= F16 Refresh

Add or Update?

F16 will use the incoming tool-id as the TD_Dup_SEQ.

F16 will use a combination the incoming tool-id and the incoming tool code for the eTIMS Tool-id.

Once the ST records have passed the pre-processing preparations and validity tests, we need to split the file into (1) records that will update existing eTIMS data and (2) records that will add to existing eTIMS data.

We can do this by finding if there is already an eTIMS record for the incoming tooling record.

1. If there is a matching TD_DUP_SEQ then we only have to deal with updates to the matching TD record. Update the tooling data as defined within the LMC Tooling Update Matrix, using the **Matrix_code** retrieved earlier.

2. If there is no matching TD record, then the incoming record is an **ADD Tool**.

Update the tooling data as defined within the LMC Tooling Update Matrix (also see below).

Adding new tool

1. The incoming record requires an add of the Basic tool (BT), an add of the Tool Dup (TD) and an add of the required TM(s). eTIMS will use the incoming Tool-ID as the BT_Tool_ID.

2. The TM record requirements and the appropriate eTIMS status will be determined by the LMC REFRESH Matrix values (RDT & RDR tables). The LMC matrix operates using the **Matrix_code** retrieved earlier.

Basic Tool Mapping

BT-TOOL-ID	IN-TOOL-ID+TOOL-CODE (POS 1-12)
BT-NOUN	IN-DESCRIPTION
BT-ORIG-TOOL-ID	IN-TOOL-ID
BT_LU_TOOLTYPE_CD_KEY	124
BT-SIZE	0
BT-WEIGHT	0
BT-UPDATE-DATE	CURRENT-DATE
BT-UPDATE-USER	F16-REFRESH
BT-STE-IND	N
BT-TOOL-ID-VENDOR	BLANK

WTR, TOOL-2-WS FIELD MAPPING

Every BT added must have a connection to a weapon system

WTR-KEY	generated
WTR-BT-KEY	From above
WTR-WS-KEY	16
WTR-UPDATE-DATE	CURRENT-DATE
WTR-UPDATE-USER	F16-REFRESH

TOOL DUP FIELD MAPPING

TD-DUP-SEQ	IN-TOOL-ID
TD-LU-OWNER-CD	OC-KEY from previous lookup
TD-OWNER-DISP	STS_OD_KEY from previous lookup.
TD-LU-STATUS-CD	(as defined by the matrix processing rules).
TD-BUILD-SITE-CODE	81755
TD-FAB-COST	IN-COST
TD-ORIG-CONTRACT	IN-ACQ-CONTRACT
TD-SERIAL-NO	IN-TOOL-ID
TD_ACQUISITION_DT	Current date
TD-SCRAP-DATE	Only if tool is inactive use IN-DISPOSITION-DATE
TD-SCRAP-USER	Only if tool is inactive use F16-REFRESH
TD-UPDATE-DATE	Current date
TD-UPDATE-USER	F16-REFRESH

TOOL MOVEMENTS FIELD MAPPING

1. every TM record will have the following fields filled in:

TM-UPDATE-DATE	Current date
TM-UPDATE-USER	F16-REFRESH
TM-ADD-DATE	Current date
TM-ADD-USER	F16-REFRESH
TM_TRANSACTION_DT	Current date

2. **Site records:** every TD added must have a site record.

TM_LU_TRANSACTION_CD	1
TM_CURR_PRIOR_CD	C or P as determined by matrix
TM_SHIP_DT	null
TM_SHIP_FROM_SITE_CD	Verified/converted IN-CAGE (or vendor cd). If blank default to 81755
TM_VN_SHIP_TO_SITE_CD	VN-KEY of above
TM_CO_KEY	CO-KEY OF ACQ contract found during pre-processing

3. Ship record, insert as determined by matrix.

TM_LU_TRANSACTION_CD	5
TM_CURR_PRIOR_CD	C
TM_SHIP_DT	disposition date
TM_SHIP_FROM_SITE_CD	Verified/converted IN-CAGE (or vendor cd). If blank default to 81755

TM_SHIP_TO_SITE_CD	none
TM_CO_KEY	CO-KEY OF ACCT contract found during pre-processing

4. Vendor record, insert as determined by matrix.

TM_LU_TRANSACTION_CD	2
TM_CURR_PRIOR_CD	C
TM_SHIP_DT	disposition date
TM_SHIP_FROM_SITE_CD	Verified/converted IN-CAGE (or vendor cd). If blank default to 81755
TM_SHIP_TO_SITE_CD	VN-KEY of above
TM_CO_KEY	CO-KEY OF ACCT contract found during pre-processing

TOOL-2-TOOL FILE

Designates which tools are master tools for other tools

Input Layout

The format of the incoming file is:

Field	Length	Type
Tool ID	8	A/N
Master Tool ID	8	A/N

Only after the Tooling Information file has finished processing may this file process.

Output expected

There are 3 things expected from the T2T Update process:

1. Updated eTIMS data
2. Reject File, image of input record, with appropriate error messages.
3. Totals Report. Including but not restricted to:
 - a. TOTAL RECORDS RECEIVED
 - b. TOTAL DUPLICATE RECORDS
 - c. TOTAL RECORDS REJECTED
 - d. TOTAL TOOL-IDS NOT FOUND
 - e. TOTAL REFERENCE TOOL-IDS NOT FOUND
 - f. TOTAL TOOL-2-TOOL LINKAGES ADDED.

Processing

The major difference in processing this Tool-2-Tool file as opposed to the Tool-2-Tool files we have processed for other Prime Vendors is the LMC file relates a tool to a tool **at the Dup level**. ETIMS will have to find the Basic Tool key for each tool in the file.

Find the proper TD by matching each tool-id against the Dup-Number Index. Using the **TD_BT_KEY** found for each tool-id insert into the TTR table where

1. TTR-TOOL-ID = incoming tool-id's
TD_BT_key
2. TTR-REFERENCE-TOOL-ID = incoming Master Tool-id TD_BT_-key.

- [BASIC MODEL](#)
- [BASIC PART NUMBER](#)
- [PART-2-NHA](#)
- [PART-2-TOOL](#)
- [PART-2-CAGE](#)

F16 Model Codes & Descriptions (BM)

	BM_MODEL_TYPE_CD	BM_MODEL_DESC
►	1C	F-16C blk 40/42
	1D	F-16D blk 40/42
	3M	F-16N blk 30
	3N	F-16TN blk 30
	5C	F-16C blk 20/30/32
	5D	F-16D blk 20/30/32
	61	F-16A blk 01/05/10/15
	62	F-16B blk 01/05/10/15
	CC	F-16C blk 50/52
	CD	F-16D blk 50/52

LMC will not give us a separate Model Code file. Therefore we must add these values using the information listed in the table above (in yellow). This will be a one-time effort. Processing Part-2-NHA will automatically add any new models received.

PART-2-NHA FILE

Layout of the input file is:

<u>Field</u>	<u>Length</u>	<u>Type</u>
Part Number	20	A/N
Part Name	15	A/N
NHA Part Number	20	A/N
Type/Version	3	A/N
Ship-From	4	A/N
Ship-Thru	4	A/N
Qty Per NHA	5	A/N

Basic Part Data

As with other weapon systems, **the first step** prior to any validity, will be to add the Part Numbers and Part Names as found, using 81755 as the Part Source.

Right-adjust and eliminate spaces from Part Number. Match incoming Part number against the BP_part_no.

If a match is found, and the incoming part name is greater than blank and is different from the BP part name, replace Part Name.

If no match is found, add to BP table, using LMC/81755 as the Part Source and connect to the F16 Weapon System (WPR).

BASIC PART FIELD MAPPING (BP)

BP-PRIME-VENDOR	4525 (81755)
BP-PART-NO	IN-PARTNO (adjusted)
BP-NOUN	IN-NAME
BP-STE-SYSTEM-IND	N
BP-UPDATE-DATE	CURRENT-DATE
BP-UPDATE-USER	F16-REFRESH

PART-2-WS FIELD MAPPING (WPR)

Every BP added must have a connection to a weapon system

WPR-KEY	generated
WPR-BP-KEY	From above
WPR-WS-KEY	16
WPR-UPDATE-DATE	CURRENT-DATE
WPR-UPDATE-USER	F16-REFRESH

PART STRUCTURE DATA

After we have added all possible part numbers, we will verify the record, collapse effectivity and add the BOM structures.

BOM VALIDITY

Basic validity needs to be done on these records. Sort the file eliminating duplicates. If 2 records are received that are **exactly** identical (Part Number, NHA, Model, From Effectivity and Thru Effectivity), remove duplicates. Qty-Per may be ignored for this test.

1. Part number cannot = blanks or zeros or asterisks, etc.
2. Part number must be found in BP table
3. NHA part number must be found in BP table.
4. IF NHA part number = Part Number, blank out the NHA part number.
5. Ship-from must be numeric and not zero.
6. Ship-thru must be numeric and not zero.
7. Type-version must be in basic Model (BM). If it is not and is not = "ALL", then add it with no description.
8. IF Qty-per is not numeric, substitute 1.
9. **Remove Bogus End-item Records.** AFTER we have passed all the records thru the above validities, there is one more hurdle.
 - a. Separate the end-item records from all the others making 2 files: an end-item file and a non-end-item file.
 - b. Compare the 2 files, matching on part number, model, ship from and ship thru.
 - c. If you find a match, blank out (delete) the matching record within the end-item file. When all the possible matches have been eliminated from the end-item file, re-combine the files.

Collapse effectivity.

This part is VERY IMPORTANT! The records must be sorted in this sequence or the collapse will not work. Sort the new-validated file by:

- Part number
- NHA part number
- Model
- From effectivity
- Thru effectivity.

After sorting the verified PART-2-NHA file will be put thru the **standard BOM processing routines**. There are 2 versions of the standard BOM processing routines (1) insert as received (2) collapse effectivity and insert.

For records with the same part number, NHA and model, combine effectivity up to and not including gaps.

PNO/NHA/ MODEL	FROM	THRU	Combine into
A/B/C	0001	0007	
A/B/C	0008	0009	0001-0009
A/B/D	0001	0012	
A/B/D	0012	0025	

A/B/D	0025	9999	0001-9999
A/B/E	0022	0031	0022-0031
A/B/E	0033	0042	
A/B/E	0043	0056	0033-0056
A/B/F	0001	9999	
A/B/F	0042	0044	
A/B/F	0056	0066	0001-9999

PLEASE NOTE THE LAST EXAMPLE A/B/F. THIS POSSIBILITY IS DIFFERENT THAN ANYTHING WE HAVE GOTTEN FROM Boeing. THE 1-9999 Record.

Add Part-2-Part records (pp)

If NHA part number equals the literal “ENDITEM” OR “END-ITEM” blank out the NHA field.

PART-2-

		PART FIELD MAPPING
(PP)	PP-KEY	generated
	PP-BASE-BP-KEY	NHA BP KEY
	PP-NLA-BP-KEY	PART BP-KEY
	PP-UPDATE-DATE	CURRENT-DATE
MODEL-	PP-UPDATE-USER	F16-REFRESH

EFFECTIVITY FIELD MAPPING (ME)

Lookup the BP_KEYS for the Part number and the NHA Part Number and insert into the PP table.

ME-KEY	generated
ME-PP-KEY	PP-KEY of record created above
ME-FROM	FROM EFFECTIVY
ME-THRU	THRU-EFFECTIVITY
ME-BM-KEY	BM-KEY of model
ME-UPDATE-DATE	CURRENT-DATE
ME-UPDATE-USER	F16-REFRESH

PART-2-TOOL FILE**Input layout**

<u>Field</u>	<u>Length</u>	<u>Type</u>	<u>Notes</u>
Part Number	20	A/N	
Engineering Tool Id	8	A/N	
Plan Number	20	A/N	for reference only

Output expected

There are 3 things expected from the P2T Update process:

1. Updated eTIMS data
2. Reject File, image of input record, with appropriate error messages.
3. Totals Report. Including but not restricted to:
 - a. TOTAL RECORDS RECEIVED
 - b. TOTAL DUPLICATE RECORDS
 - c. TOTAL RECORDS REJECTED
 - d. TOTAL PART NUMBERS NOT FOUND
 - e. TOTAL TOOL-IDS NOT FOUND
 - f. TOTAL PART-2-TOOL LINKAGES ADDED.

Validities

Sort the file eliminating duplicates. If 2 records are received that are exactly identical, remove duplicate.

Part number must be greater than blank. If part number is blank and plan number is not blank, move plan number to part number field and use it as the “best guess part number”.

Tool-id must be greater than blank.

If both of the above conditions are met, continue processing, else skip that record.

Right-adjust the incoming part number and eliminate blanks. Match to BP_PART_NO.

1. If no match is found write **entire** record to reject file, with message “Part Number not found in eTIMS”.
2. If a match is found, use the BP_KEY to insert.

Match the incoming Tool-id to the **Tool/Dup index**.

1. If no match is found, write **entire** record to reject file with the message ‘Tool-id not found in eTIMS’.
2. If there is a match found, use the TD_BT_KEY to insert.

If both keys are found, update the PT table setting

- PT_BT_KEY = BT key found above.

- PT_BP_KEY = BP key found above.
- PT_UPDATE_DT = current sys date
- PT_UPDATE_USER = F16 Refresh.

Input datasets

Process the F16 Part-2-Tool data sets in the following order.

Maintain separate totals and reject files.

PART-2-CAGE FILE

<u>Field</u>	<u>Length</u>	<u>Type</u>
Part	32	A/N
Filler	1	
CAGE	5	A/N

The CAGE/Vendor Code I be the same as that done by Boeing processing. Add to the PC table as follows:

- PC_BP_KEY = BP-KEY of incoming part number
- PC_PART_CAGE_CD = CAGE.
- PC_UPDATE_DT = current sys date
- PC_UPDATE_USERID = F16 Refresh

STE File Layout.

The format of the incoming file very similar to the tooling information file, and will use almost the same logic as the tooling information updates.

<u>Field</u>	<u>Length</u>	<u>Type</u>	<u>Notes</u>
Asset number	8	A/N	
Parent Asset number	8	A/N	May be blank
Ownership Code	1	A/N	
Status Code	8	A/N	
Dollar Cost	15	D13.2	
Current CAGE/Vendor Code	6	A/N	
Used-on	32	A/N	
PARS2 Part Number	32	A/N	
Description	40	A/N	
Acquisition Contract	16	A/N	
Current Contract	16	A/N	
Serial number	15	A	
Acquisition Date	10	YYYY-MM-DD	
Archive Flag	1	A/N	

Output expected

There are 3 things expected from the STE Information Update process:

1. Updated eTIMS data
2. Reject File: image of input record, with appropriate error messages.
3. Totals Report. Including but not restricted to:
 - a. TOTAL RECORDS PROCESSED
 - b. TOTAL RECORDS REJECTED
 - c. TOTAL TOOLS UPDATED
 - d. TOTAL TOOLS ADDED.

Pre-processing tasks

The pre-processing tasks include validity checks against designated fields and conversion of specific LMC field values to eTIMS values.

There are 6 pre-update verification tasks needed to both prep the update record and prepare FK tables:

1. **ASSET NO.** Left adjust, eliminating leading blanks.

a. Cannot be all blanks or all zeros. If record contains all blanks or zeros for the value for ASSET-NO, reject record with message "invalid ASSET" and send to Reject file.

b. If ASSET-NO starts with 2 zeros and a letter, remove the leading zeros.

Example 00ME1309 → ME1309

2. **Status.** Translate Status code using [STS table](#). Using 4525 as vendor key, match incoming status to STS_old_status and pick up STS_OD_KEY.

3. **Owner Disposition.** Using the incoming OWNER DISPOSITION code, match to the OD table using 81755/4525 as vendor. Failure to find a matching OD code is a show-stopper for any given tooling record.

a. If a match is found, attach the key to the record to be used later.

b. If the OD code is not found, **reject the record** with the message "unknown' OD code'.

4. **Owner Codes.** Must be present within the OC table. Lookup entries in the OC-Table using 4525/81755 as the Vendor key.

a. If found attach the OC_LU-KEY value with the tooling record and continue processing.

b. If the owner code is blank, reject the record with the message 'no owner code'.

c. If the owner code is not found in the table, reject the record with the message "unknown ownership code'.

5. **CAGE/Vendor-code.** Match the incoming vendor code against the [Vendor Translation Table](#), or the Vendor table, depending on the length of the vendor code coming in. If the length is 6 digits, we may assume it is a vendor code and needs to be translated to a CAGE using the vendor translation table. If the length is only 5 digits, we may assume it is a CAGE and go directly against the vendor table.

a. **If the incoming CAGE/Vendor-code field is 6 digits,** match to the Vendor Translation table's VT-VENDOR-CD using 81755(4525) as the Prime Vendor Code.

- i. If a match is found and VT-CAGE = "Y", do not replace the incoming CAGE, the vendor code should have been added to the vendor table during pre-processing.
- ii. If a match is found and VT-CAGE is not = "Y", replace the record's Vendor-code with the value in VT_CAGE.
- iii. If a match is not found and the record has an inactive status, blank out the record's Vendor code.
- iv. If a match is not found and the record has an active status, place the incoming CAGE. Place the incoming CAGE/Vendor-code on a list to be worked by the analyst.

b. **If the incoming CAGE/Vendor-code field is 5 digits,** match directly to Vendor Table (VN).

- i. If a match is found, the data is good, continue processing.
- ii. If a match is not found, reject the record, and place the incoming CAGE/Vendor-code on a list to be worked by the analyst.

6. **Contract table.** LMC will not send us a separate Contracts file, so we must examine the Tooling information file to pick-up and pre-update the eTIMS Contracts table (CO). There are 2 contract fields within each Tooling Information record: Acquisition & Current.

- a. From each contract field, use the **Standard Contract Formatting routine** to remove dashes and slashes and left-adjust the contract number.
- b. Match each against the eTIMS Contract table (CO).
 - i. If a match is found, Examine the CO-WS-OWNER-CD-KEY. If not = 16 or null, update to null value. Pickup the CO-CO-KEY and attach to the record.
 - ii. If a match is not found, add to the CONTRACT table using values below, note the assigned CO-CO-KEY and attach to the Tooling record.
- c. CONTRACT TABLE FIELD MAPPING

CO-CONTRACT-NO	= Formatted contract number
CO-VN-KEY	= 4525
CO-WS-OWNER-CD	= 16
CO-UPDATE-DATE	= Sys-date
CO-UPDATE-USER	= F16 Refresh

ADD OR UPDATE (FINDING THE MATCHING TD RECS)

Once the records have passed the pre-processing validity and preparations, we need to split the file into records that will update existing eTIMS data and records that will add to existing eTIMS data. We can do this by finding if there is already an eTIMS record for the incoming tooling record.

[Lookup the incoming tool-id against the TD_Dup_SEQ index.](#)

1. If there is a matching TD_DUP_SEQ then we only have to deal with updates to the matching TD record. Update the tooling data as defined within the LMC Tooling Update Matrix.
2. If there is no matching TD record, then the incoming record is an ADD Tool.

Adding STE

1. STE records to be inserted are a straight 1-for-1: for each STE record add 1 BT, 1 TD and 1-2 TM records (depending upon the Matrix).
2. ETIMS will use the LMC Matrix Add tool logic in order to know what tool movements records to add, what eTIMS status codes to use, etc. We have no STE Dup-2-Dup table.

BT-TOOL-ID	IN-ASSET
BT-NOUN	IN-DESCRIPTION
BT-ORIG-TOOL-ID	IN-ASSET
BT_LU_TOOLTYPE_CD_KE	125
Y	
BT-SIZE	0
BT-WEIGHT	0
BT-UPDATE-DATE	CURRENT-DATE
BT-UPDATE-USER	F16-STE-REFRESH
BT-STE-IND	Y
BT-TOOL-ID-VENDOR	BLANK

STE MOVEMENTS FIELD MAPPING (TM)

Every TM record will have the following fields filled in:

TM-UPDATE-DATE	Current date
TM-UPDATE-USER	F16-STE-REFRESH
TM-ADD-DATE	Current date
TM-ADD-USER	F16-STE-REFRESH
TM_TRANSACTION_DT	Current date

Site records: every TD added must have a site record.

TM_LU TRANSACTION_CD	1
TD-DUP-SEQ	IN-ASSET
TD-LU-OWNER-CD	OC-LU-KEY from previous lookup
TD-OWNER-DISP	STS_OD_KEY from previous lookup.
TD-LU-STATUS-CD	(as defined by the matrix processing rules).
TD-BUILD-SITE-CODE	81755
TD-FAB-COST	IN-COST
TD-ORIG-CONTRACT	IN-ACQ-CONTRACT
TD-SERIAL-NO	IN-SERIALNO
TD_ACQUISITION_DT	Current date
TD-SCRAP-DATE	Only if tool is inactive use IN-ACQ-DATE
TD-SCRAP-USER	Only if tool is inactive use F16-REFRESH
TD-UPDATE-DATE	Current date
TD-UPDATE-USER	F16-STE-REFRESH
TM_CURR_PRIOR_CD	C or P as determined by matrix
TM_SHIP_DT	none
TM_SHIP_FROM_SITE_CD	Verified/converted IN-CAGE (or vendor cd). If blank default to 81755
TM_VN_SHIP_TO_SITE_CD	VN-KEY of above
TM_CO_KEY	CO-KEY OF ACQ contract found during pre-processing

Ship record, insert as determined by matrix.

TM_LU TRANSACTION_CD	5
TM_CURR_PRIOR_CD	C
TM_SHIP_DT	disposition date
TM_SHIP_FROM_SITE_CD	Verified/converted IN-CAGE (or vendor cd). If blank default to 81755
TM_SHIP_TO_SITE_CD	none

TM_CO_KEY CO-KEY OF ACCT contract found during pre-processing

Vendor record, insert as determined by matrix.

TM_LU_TRANSACTION_CD	2
TM_CURR_PRIOR_CD	C
TM_SHIP_DT	disposition date
TM_SHIP_FROM_SITE_CD	Verified/converted IN-CAGE (or vendor cd). If blank default to 81755
TM_SHIP_TO_SITE_CD	VN-KEY of above
TM_CO_KEY	CO-KEY OF ACCT contract found during pre-processing

ESTABLISH STE USED-ON (BP)

After we have successfully added the STE tooling data, we need to add the BP STE Usage records.

Right adjust IN-USED-ON, eliminating leading spaces.

Match adjusted IN-USED-ON to BP-part-no.

- a. If it is found, continue processing.
- b. If a no match, add the BP-record.

STE SYSTEM USED-ON BASIC PART FIELD MAPPING

BP-PRIME-VENDOR	4525/81755
BP-PART-NO	IN-used-on
BP-NOUN	BLANK
BP-STE-SYSTEM-IND	Y
BP-UPDATE-DATE	CURRENT-DATE
BP-UPDATE-USER	F16-STE-REFRESH

CONNECT STE & THEIR USED-ONS (PT)

After all possible STE USED-ONS have been established or found in BP, process the STE records again, this time adding the part-2-tool linkages.

If the record's USED-ON is blank skip this record.

Right-adjust the record's USED-ON, eliminating leading spaces. Match to BP_PART_NO and pickup the BP-key.

Match the incoming ASSET-NO to the **Tool/Dup index**. If no match is found, reject the record with the message 'Tool-id not found in eTIMS'. If there is a match found, use the TD_BT_KEY to insert.

If both keys are found, update the PT table setting

PT_BT_KEY	BT key found above
PT_BP_KEY	BP key found above
PT-UPDATE-DATE	Current date
PT-UPDATE-USER	F16-STE-REFRESH

CONNECT STE SETS (TTR)

Process the STE file again, this time connecting the Tool-2-tool linkages.

If the record's PARENT-ASSET-NO is blank, skip processing this record.

Match the **PARENT-ASSET-NO** against the TD-Dup-Number Index.

- If PARENT-ASSET-NO is not found, write record to exception file with message "Parent Asset number not found".
- If the PARENT-ASSET-NO is found, store the **TD_BT_KEY** found to be used in the TT insert.
- Match the **ASSET-NO** against the TD-Dup-Number Index.
- If the ASSET-NO is not found, write the record to the exception file with message "asset number not found".
- If the ASSET is found, store the **TD_BT_KEY** found to be used in the TT insert.
- If both PARENT-ASSET-NO and ASSET NUMBER were successfully found, insert a TTR record as follows:

TTR-TOOL-ID	PARENT-ASSET-NO BT-KEY
TTR-REFERENCE-TOOL-ID	ASSET-NO BT-KEY
TTR-UPDATE-DATE	Current date
TTR-UPDATE-USER	F16-STE-REFRESH

SECTION 5 - COMPUTER SOFTWARE CONFIGURATION ITEM (CSCI) DATA

5.1 CSCI Database Description

The (DBMS) AIS will be developed using the Oracle® 10G database management system. The system databases will be in a relational structure. Each table and column used in the eTIMS system will be further defined in the Data Dictionary.

5.2 Failure Contingencies

Since the eTIMS CSCI will be developed on a server with hard disk RAID (Redundant Array Inexpensive Disk) back-up controller and software, a hard drive failure should not cause data loss or system downtime. The system is being replicated to a backup site. Replication occurs as information changes in the system. If an event occurs where the primary site is not available the backup site will be activated and the users access the site without any knowledge. When the primary site is available again, the information is replicated to the primary site and the users are moved back without their knowledge. As a last resort, backup copies of the application software, database structure and data should be maintained in both an on-site and off-site location. In the event of a failure of the software currently being used to manage the system, a backup copy can be quickly loaded and put into service.

Data integrity will be protected by limiting invalid user access to the system and by validating on-line data changes. Access to the system is limited to only those user's with valid CAC/ECA certificates tied to the user ids. The eTIMS system will verify format and value ranges for all inputs. Data not meeting the format or value criteria will be rejected.