Real Time Location System (RTLS)

Enterprise Systems Engineering (ESE)

Asset Tracking Interface

Installation, Back-Out, and Rollback Guide

November 2017

Department of Veterans Affairs

Office of Information and Technology (OI&T)

Table of Contents

1. Introduction

The Installation, Back-Out, Rollback Guide defines the ordered, technical steps required to install the product, and if necessary, to back-out the installation, and to roll back to the previously installed version of the product. It provides installation instructions for the Asset Tracking interface as managed through the Real Time Location System (RTLS) project.

1.1. Overview

This installation guide covers the installation and configuration of the RTLS Asset Tracking (AT) interface application. The AT interface application synchronizes AEMS-MERS equipment details and locations with Intelligent InSites RTLS.

The environment needed to run the applications is multi-server and includes the following distinct deployment components:

- RTLS VistA Patch (KIDS build)
- WebLogic VistaService Web Services
- RTLS Mule ESB domain component
- RTLS Mule ESB application component
- RTLS-VistA Interface Configuration Tool

Deployment and configuration of any of these components has dependencies on at least one of the others (i.e. the Mule ESB application is dependent on VistaService Web Services; VistaService is dependent on the VistA patch).

The following sections provide descriptions of the deployment components.

1.1.1. RTLS VistA Patch (KIDS Build)

The RTLS patches and a companion patch for the Engineering package (VIAA*1*1, VIAA*1*3 and EN*7*100) support the implementation of the project and are distributed in a Kernel Installation and Distribution System (KIDS) build. Multiple Remote Procedure Calls (RPCs) are used to support the user interfaces with the packages touched by this project.

The RTLS patch is exporting a new option, 'Make Web Call to Mule' [VIAA MAKE WEB CALL TO MULE], to send Engineering file changes (MUMPS Style Cross Reference events) to the RTLS database via a Mule server. This option is a queue process that checks every number of minutes, configured by the site, to see if there are any record changes from files EQUIPMENT INV (#6914) and ENG SPACE file (#6928) to transmit to RTLS.

A new file, PENDING RTLS EVENTS (#6930), is also exported by the patch. The PENDING RTLS EVENTS file holds record changes before transmission to RTLS and when the Mule server is off-line. When the Mule server comes back online, the next run of the queue transmits every record and the file is cleaned up immediately. While the file may be populated with record changes at any time, its use is limited to very shorts periods of time and will not require disk space at the local site. The PENDING RTLS EVENTS file introduces a redundancy to save record changes that could not be sent to the RTLS database when the Mule server is offline for any reason.

Data dictionary changes to the EQUIPMENT INV file (#6914) and the ENG SPACE file (#6928) are exported by the Engineering patch.

The following options in the Engineering package generate events after record changes have been completed by users of the package. Multiple fields in the EQUIPMENT INV file (#6914) and the ENG SPACE file (#6928) will be equipped with MUMPS cross references to monitor data changes.

- New Inventory Entry [ENINVNEW]
- Multiple Inventory Entry [ENIN-ENTER-MULTI]
- Inventory Edit [ENINV EDIT]
- Enter New Room Space Data [ENSPROOM]
- Display/Edit Room Data [ENSPROOMD]

Additionally, any changes completed via the Enter/Edit option of VA FileMan for any of the fields monitored will create an event that will be sent to RTLS.

For more information and instructions on how to install the VistA patches, see the RTLS AT Patch Installation Guide.

1.1.2. WebLogic VistaService Web Services

Communication with VistA is managed with a series of RESTful web service methods designed for the RTLS solution called VistaService. VistaService is an interface to VistA creating a uniform calling method for underlying MUMPS RPCs. The RESTful web service methods provide a mechanism to encapsulate the VistALink calls into a standard format. This improves code maintainability, de-couples the service from the interface so multiple interfaces can use it, and provides for future scalability of the interface.

VistaService has methods for creating, retrieving, and updating VistA data over HTTP using RESTful semantics. RESTful methods are called using a URL.

1.1.3. RTLS Mule ESB Domain Component

Part of the RTLS solution includes an open source Enterprise Service Bus (ESB) called Mule ESB. Mule ESB is a Java-based ESB and integration platform to facilitate message routing and data transformation.

The RTLS Domain component provides a central configuration for all domain applications to share common resources, such as the HTTP listening port. It must be installed before any domain applications are installed which reference the shared resources.

1.1.4. RTLS Mule ESB Application Component

The RTLS AT interface application is orchestrated by a series of ESB flows. A Mule ESB flow is a way to manage or orchestrate services. Flows provide a flexible method to build integrations by choosing building blocks from a standard list of defined components. By separating configuration information from the underlying code using building blocks, building, deploying and maintaining the interface is easier to manage.

Within the ESB flows, RESTful web services are employed on both ends to facilitate communication between RTLS (i.e., Intelligent InSites) and VistA. Communication with VistA is managed by the VistaService Web Service methods.

1.1.5. RTLS-VistA Interface Configuration Tool

The RTLS-VistA Interface Configuration Tool is a software utility used to configure the schedule of the AEMS-MERS/RTLS interface. The tool is designed for administrative users of RTLS to have a single location to view and manage network connections and associated jobs within the VISN. The tool is a web application hosted on the interface WebLogic server and is displayed as a Snap-in screen on the Intelligent InSites user interface.

1.2. Purpose

The purpose of this guide is to provide VA facility Information Resource Management Systems (IRMS) personnel, and the VistA Laboratory Information Manager (LIM) with the necessary technical information required to understand the installation and implementation of the RTLS AT interface application. This guide provides instructions for installing and configuring the following components of the applications:

- WebLogic VistaService Web Services
- RTLS Mule ESB domain component
- RTLS Mule ESB application component
- RTLS-VistA Interface Configuration Tool

The installation of the VistA patches is covered in the RTLS AT Patch Installation Guide.

The RTLS AT interface application is a distributed solution with multiple components each with its own deployment. Due to the complexity of the components and operating environment, as well as software release changes and local VA environment protocols, each installation may be different. Because of these differences, individual installations may vary from the instructions provided in this guide. Where possible, every effort has been made to explain the purpose of steps to aid in understanding.

1.3. Scope

The scope of this guide is limited to the technical steps required to install, configure, and implement the non-VistA components of the RTLS AT interface application. This includes:

- Installation and configuration of the VistaService Web Services
- Installation and configuration of the RTLS ESB Application Component
- Installation and configuration of the RTLS-VistA Interface Configuration Tool

This guide does not include:

- Installation and configuration of the WebLogic Domain or Mule Enterprise Edition
- Installation and configuration of Intelligent InSites

Intelligent InSites is a Commercial off-the-shelf (COTS) product. Installation and configuration are managed by the vendor as part of the overall deployment plan.

1.4. References

- RTLS AT Patch Installation Guide
- VistALink System Management Guide
- VistALink Developer Guide
- VistALink Installation Guide
- RTLS ESE Deployment Plan
- RTLS ESE Formal Test Results

2. Pre-installation and System Requirements

The following points are important to note about the RTLS AT interface application installation covered in this guide and the companion guide, the RTLS AT Patch Installation Guide:

- The guides represents the instructions for installing the RTLS AT interface application in a test environment.
- Installation will always occur within a test environment before being approved for installation in production.
- The difference between a test and production installation is minimal.
- Users do not have to log off the system during installation.
- The average amount of time required to install the software is 4 hours.
- This software does not have to be installed during off-peak hours.

2.1. Platform Installation and Preparation

2.1.1. Non-VistA Required Software

The following list of software represents the minimum versions required for installation:

Software Application	Version
Mule Enterprise Edition	3.7.3
Oracle JRockit	R28.2.5-4.1.0 64 bit (for WebLogic)
Oracle WebLogic Enterprise Edition	10.3.6 64 bit
Red Hat Enterprise Linux (RHEL)	6.7
Sun/Oracle Java Development Kit (JDK)	1.8
MMC Console	3.4.3
Microsoft SQL Server	2012
Intelligent InSites RTLS	4.7

2.1.2. Installation Sequencing Information

The installation must be performed in the following order:

Step 1: RTLS AT VistA Patches
• Follow all instructions in the RTLS AT Patch Installation Guide
Step 2: WebLogic VistaService Web Services
• VistaService Web Services Pre-Installation (section 2.8) • VistaService Web Services Installation (section 3.1)

Step 3: RTLS-VistA Interface Configuration Tool
• RTLS-VistA Interface Configuration Tool Installation (section 3.2)
Step 4: RTLS ESB Components
• RTLS ESB Application Component Pre-Installation (section 2.9) • RTLS ESB Application Component Installation (section 3.3)
Step 5: Configure SSL/TLS
• Configuring SSL/TLS (section 4.1.1)

2.2. Download and Extract Files

The HPE RTLS Administration Support Organization is responsible for installing the RTLS AT Interface Application. The software installation files are stored within the VA's Rational Repository and described within the RTLS Version Description Document (VDD).

2.3. Database Creation

The RtlsInterfaceAdmin application runs on SQL Server. See section 3.2.1 for instructions on how to create the database.

2.4. Installation Scripts

The installation scripts for the deployment of the RTLS AT application interface are interspersed throughout this guide.

2.5. Cron Scripts

N/A

2.6. Access Requirements and Skills Needed for the Installation

The RTLS AT interface application is multi-server and includes distinct deployment components. The skills needed for the installation is varied by deployment components:

- **Web Services and Mule ESB components** - the audience is a system administrator with a strong working knowledge of Linux, distributed systems, Java Enterprise Edition (JEE), and virtual environments.
- **Configuring SSL/TLS** - the audience is a systems administrator with a strong working knowledge of both Linux systems administration and enterprise security.

The installation of the RTLS AT Interface Application, with the exception of the RTLS VistA components, will be performed by the HPE RTLS Administration Support Organization along with representatives from peer VA organizations. The installation and configuration of all RTLS VistA components will be performed by VA personnel. Contact the HPE RTLS System Administration Support Team via email: vartlsadmins@hpe.com.

2.7. Environmental Variables

The following table provides a list of environmental variables used throughout this guide. An environmental variable represents the name of a directory. Directory names can vary across installations. The variables provide a method of standardizing the install process.

Environmental Variable	Definition/Origin
{environment}	Used to enable stacking multiple WebLogic environments on a Virtual Machine (VM).
{MW_HOME}	Directory for middleware home (eg /data/Oracle/Middleware).
{WL_RTLS}	Directory for RTLS WebLogic script home (eg /data/rtls/weblogic). This directory is typically outside of the Middleware directory structure.
{APPROOT}	Directory for VistaServices (eg /data/rtls/vistasvcs) and the RTLS ESB application depending on the component being installed.
{domain_home}	Directory for WebLogic domain home (eg {MW_HOME}/user_projects/domains/{environment}_domain).
{MULE_HOME}	Directory where Mule is installed.

2.8. VistaService Web Services Pre-Installation

VistaService is a WebLogic application. Pre-installation steps for the VistaService Web Services are the configuration tasks needed to establish the application within the WebLogic domain. The installation and configuration of the WebLogic domain is outside the scope of this document.

To perform the configuration tasks needed to establish the VistaServices web services application within the WebLogic domain, complete all of the steps in the following sections.

2.8.1. Setup Security

Create a new group and user as the application owner for VistaServices. For ease of maintenance, consider creating group IDs (GIDs) and user IDs (UIDs) consistent across environments. GID and UID synchronization is required if using Network File System (NFS). Your local systems administrator will provide guidance. The instructions below use 550 as an example only. Change this value as appropriate for your installation. Execute instructions as root.

1. Create the VistaServices group and user:

   groupadd -g 550 vistasvcs

   useradd -g 550 -u 550 -c "Vista Services AppId" -d /home/vistasvcs -s /bin/bash vistasvcs

2.8.2. Create WebLogic User Config Files for VistaServices

Create WebLogic scripts for automatically configuring, starting, and stopping VistaServices by copying generic scripts shipped with WebLogic and modifying them.

1. Copy the supplied WebLogic scripts to the designated scripts folder in your environment (i.e. {WL_RTLS}).
2. Edit the start_{environment}.py and change the following:

   Change base_cluster to vs_cluster

   Change base_server_ to vs_server_
3. Edit the stop_{environment}.py and make the same changes as step 2.
4. Verify that the entry in wlst.sh script references the valid location of wlst.sh under the WebLogic install (i.e. {MW_HOME}/wlserver_10.3/common/bin/wlst.sh $*)
5. Setup the userkey files for the id used to run the scripts (typically WebLogic):
   a) Modify install\create_userkey.py with the environment_name to name the files ({environment}) and the WebLogic password that was specified when the domain was created.
   b) From {WL_RTLS} directory, run wlst.sh install/create_userkey.py
   c) Respond by entering y to the one prompt that displays.

      Note: It only prompts the first time a particular userconfig is created.
   d) Remove the password from create_userkey.py
6. Add the WebLogic startup script command to system startup file:

   Startup file = /etc/rc.local

   Add:

   {WL_RTLS}/vm_startup_initd.sh
7. Restart the server and verify WebLogic servers came up.

2.8.3. Create a Certificate for the Intelligent InSites Connector Server

There are many different methods to create an SSL certificate request. The following generalized steps are provided as one example using Microsoft Management Console (MMC). Your installation may vary. The Microsoft Management Console (MMC) with the Certificates snap-in is used to view and manage SSL server certificates. Execute instructions as root on the server the certificate will be installed on:

1. Start MMC and select Add/Remove Snap-in.
2. Select Certificates. Select Add.

3. When asked for the type of account, Select Computer Account and Local Computer.
4. In the Personal Folder, go to All Tasks and click Request New Certificates.
5. Select the Active Directory Enrollment Policy as the Certificate Enrollment Policy.
6. On the Request Certificates page, select VA Web Server (Manual Enroll).
7. Provide the certificate identification information for the following parameters.

Parameter	Definition	Example
CN	Common Name – The fully qualified domain name (FQDN) of the server you are requesting the certificate for.	<servername(FQDN)>.<server id>.med.va.gov
O	Organization	Midwest Health Care Network
OU	Organizational Unit	<City> Vet Center
L	Locality	<City>
S	State	<State>
C	Country	US

Note: The above information is required with all of the certificate request tools.

8. Click on Enroll.

   The wizard displays a message that the certificate has been enrolled and installed on the computer. The certificate will not be available immediately. When available (may be up to 24 hours), the certificate will appear within MMC under Personal folder → Certificates.

   The Intelligent InSites connector software is aware of MMC so no keystore is necessary. Intelligent InSites interacts with MMC to verify the certificates when needed.

9. Use the information from the enrollment process to send a request to VA to issue the certificate. Follow the VA PKI SSL/TLS Request process (http://vaww.pki.va.gov/ssltls/).

2.9. RTLS ESB Application Component Pre-Installation

2.9.1. Setup Security

Create a new group and user for Mule ESB. For ease of maintenance, consider creating group IDs (GIDs) and user IDs (UIDs) consistent across environments. GID and UID synchronization is required if using Network File System (NFS). Your local systems administrator will provide guidance. The instructions below use 503 as an example only. Change this value as appropriate for your installation. Execute instructions as root.

1. Create the ESB group and user:

   groupadd -g 503 esb

   useradd -g 503 -u 503 -c "ESB" -d /home/esb -s /bin/bash esb

2.9.2. Create Application Directories

Create an RTLS application directory owned by the esb group.

1. Create the directory and change ownership as root.

   mkdir /data/rtls

   chown -R esb:esb /data/rtls

2. Change to the esb owner id so all new directories and files will be owned by esb.

   su – esb

3. Create directories outside of the {MULE_HOME} directory to use as {APPROOT} later in the install instructions. This will be for RTLS ESB application usage. Creating the directory outside of {MULE_HOME} allows new versions of {MULE_HOME} to be installed without losing environment-specific information.

   mkdir /data/rtls/esb

   mkdir /data/rtls/esb/{environment}

2.9.3. Configure Mule ESB Enterprise Edition for RTLS

2.9.3.1. Add Java Properties

1. Add the following properties to {MULE_HOME}/conf/wrapper.conf

   wrapper.java.additional.4=-Desb.config.dir="/data/rtls/esb/base"

   wrapper.java.additional.4.stripquotes=TRUE

   wrapper.java.additional.5=-Ddeployment.security.TLSv1.1=true

   wrapper.java.additional.6=-Ddeployment.security.TLSv1.2=true

   wrapper.java.additional.7=-Djdk.tls.client.protocols=TLSv1.2

   wrapper.java.additional.8=-Dhttps.protocols=TLSv1.2

Note: The numbers appended to the above property statements vary based on how many additional properties are configured in the file. Modify to the next available number.

2.9.3.2. Modify Existing Java Properties

Increase the logfile size, the number of archive files, and the Java memory. The values below are recommendations only. Edit the {MULE_HOME}/conf/wrapper.conf file properties as follows:

1. Increase the logfile size to 20 MB

   wrapper.logfile.maxsize=20m

2. Increase the number of archive files to 99

   wrapper.logfile.maxfiles=99

3. Increase the Java memory to 1024 MB

   wrapper.java.maxmemory=1024

3. Installation Procedure

The following sections provide instructions for installing the individual components that comprise the RTLS AT interface application.

3.1. VistaService Web Services Installation

The section defines the installation procedure for the following sub-components:

- **VistaLinkConsole-1.6.0.028.ear** – This is a VA Developed WebLogic application that enables an Administrator to monitor the health of VistaLink connectors. Once installed, the application is accessible through the WebLogic Administration console.
- **VistaLinkSamples-1.6.0.028.ear** (non-production environments only). – This is a VA developed sample application that can be used to test VistaLink configurations. It is only for use in test environments.
- **vlj-1.6.0.028** – This is a VA Developed J2EE Resource Adapter that defines VistaLink connections and binds them to the WebLogic Server Java Naming and Directory Interface (JNDI). VistaService uses these connections to execute RPCs on a given Vista instance.
- **vistaassettrackservice.war**– This is the HP developed application that implements a set of RESTful Web Services invoked by the ESB in the RTLS solution.

3.1.1. Add WebLogic ID to the Vistasvcs Group

Add the WebLogic ID to the vistasvcs group so the VistaServices web services can write to the log files, have access to the staging area, etc.

1. Add WebLogic ID to vistasvcs group

   usermod -a -G vistasvcs weblogic

3.1.2. Create Application Directories

1. Create the application directory (i.e. {APPROOT}) to hold VistaServices and set permissions. As root, create the directory, change the owner and permissions.

   mkdir /data/rtls/vistasvcs

   chown vistasvcs:vistasvcs {APPROOT}

   chmod 775 {APPROOT}

3.1.3. Copy install Files and Extract

Copy the VistaServices setup file into a new application directory and untar it. Unzip the VistaLink components in the staging area.

1. As vistasvcs, make a new directory, copy the setup file and untar it:

   mkdir {APPROOT}/{environment}

   cd {APPROOT}/{environment}

cp the vistasvcs_setup.tar.gz in to this directory

tar -xzf vistasvcs_setup.tar.gz

2. As vistasvcs, unzip the VistaLink components into the staging area:

   cd staging

   unzip vlj*.zip

3.1.4. Configure HEV_CONFIG/log4j.xml File

1. The log4j.xml file is a configuration file for establishing parameters important for logging application failures. Edit the file and set the file location for logging as well as the level of logging. Set the location of the log4j.xml file in the <param> element.

   Logging file location = {APPROOT/{environment}/HEV_CONFIG/log4j.xml

   The file directory added to the param element should match the variable name set in setDomainEnv.

   e.g. <param name="File" value="${log4j.log.dir}/...."/>

2. Set the level of logging appropriate for the log4j environment as follows:

   Production environment : <level value=”warn”/>

   Testing environment: <level value=”debug”/>

3.1.5. Modify VistA Configuration Files

Two files must be revised to include entries for each VistA instance the RTLS solution needs to communicate with:

- gov.va.med.vistalink.connectorConfig.xml – The schema definition for this file is connectorConfig.xsd. A connector element needs to be defined for each Vista instance to be configured. See the VistaLink System Management Guide, section 2.3 VistALink Connector Configuration File for details.
- weblogic-ra.xml - contains WebLogic-specific deployment descriptor configuration settings (such as initial and maximum pool sizes) and JNDI-related properties that must be modified for each adapter, to distinguish one adapter from another in JDNI. See the VistaLink System Management Guide, section 2.2.4, 1.6 Deployment Descriptor: weblogic ra.xml for details.

1. Change the permission of the connector config file:

   chmod 666
   {APPROOT/{environment}/HEV_CONFIG/gov.va.med.vistalink.connectorConfig.xml

2. For newly added site connector names, edit the following file and create new <connection-instance> elements by copying an existing one, and modifying the Java Naming and Directory Interface (JNDI) name with the new name. Update will include two instances for each connection.

   {APPROOT}/{environment}/staging/vlj-*/META-INF/weblogic-ra.xml

3.1.6. Deploy VistaLinkConsole and Other Applications

Applications are deployed using the WebLogic Administration Console. The Administration Console is the web-based management interface for a WebLogic domain. For more information on how to use the WebLogic Administration Console, visit the online Oracle documentation library (http://www.oracle.com/technetwork/indexes/documentation/index.html).

Deploy the applications and resource Adapter from {APPROOT}/{environment}/staging

1. Deploy VistaLinkConsole to the AdminServer
2. Deploy the exploded vlj-1 directory as a resource adapter on the AdminServer and on the Cluster.
3. Deploy the VistaServiceEar to the cluster.
4. For test environments, deploy vistaLinkSamples* to the cluster. For Prod, may not want the samples loaded.

3.1.7. Setup and Configure Security

The create_vistasvcs_security.py script creates new user groups, security groups, and roles to support VistaServices Web Services and the RTLS-VistA Interface Configuration Tool. Verify the parameters in the VistaServices Security script, change where necessary and then run the script. Execute commands as weblogic from {WL_RTLS}:

1. Verify params in wlst.sh install/create_vistasvcs_security.py
2. Run the script

   wlst.sh install/create_vistasvcs_security.py
3. Update the vistaConfig6.dat and insitesConfig6.dat files on the ESB server and configure the system connection information appropriately (See 4.1.2).

Once the application is deployed the services are started.

3.2. RTLS-VistA Interface Configuration Tool Installation

To install the RTLS-VistA Interface Configuration Tool, complete all steps in this section.

3.2.1. Create SQL Server Database

The RtlsInterfaceAdmin application runs on SQL Server. Complete the following steps as the database administrator to create the database:

1. Execute the following script to create the appropriate permission to create the database.

   grants.sql
2. Create a database named admin_tool for the Application data store.
3. Create a database named rtls_scheduler for the Quartz scheduler data store.
4. Execute the following script to create the primary data store for the application. By default this script uses the admin_tool database to create the schema.

MSSS2008R2_admin_tool_with_enterprise_data_Consolidated_04162014.sql

5. Execute the following script to create the Quartz scheduler database used by the application. By default this script uses the admin_tool database to create the schema.

   rtls_scheduler_sqlserver_03282014.sql

3.2.2. Define SQL Server Data Sources

Define the two SQL Server data sources using the Weblogic Admin Console. Create one Java Database Connectivity (JDBC) data source for each RtlsInterfaceAdmin database using the same user as detailed below. When prompted, select “Supports Global Transaction and Choose One-Phase Commit”.

Application Data Store	
Parameter	**Value**
Name	MSSqlRtlsAdmin
JNDI Name	jdbc/MSSqlRtlsAdmin
Database Type	MS Sql Server
Database Driver	Oracle's MS SqlServer Driver (Type 4) Versions 7 and later
Database Name	admin_tool
Host Name	Fully qualified SQL Server host name
Port	1433
Database User Name	user created for this database
Password	account password
Driver Class Name	weblogic.jdbc.sqlserver.SQLServerDriver
URL	jdbc:weblogic:sqlserver://[HOST_NAME]:1433
Target	Assign Data Source to the "All Server of the Cluster" option

Quartz Scheduler Data Store	
Parameter	**Value**
Name	MSSqlRtlsScheduler
JNDI Name	jdbc/MSSqlRtlsScheduler
Database Type	MS Sql Server
Database Driver	Oracle's MS SqlServer Driver (Type 4) Versions
Database Name	rtls_scheduler
Host Name	Fully qualified SQL Server host name
Port	1433
Database User Name	user created for this database

Quartz Scheduler Data Store	
Password	account password
Driver Class Name	weblogic.jdbc.sqlserver.SQLServerDriver
URL	jdbc:weblogic:sqlserver://[HOST_NAME]:1433
Target	Assign Data Source to the "All Server of the Cluster" option

3.2.3. Deploy the RTLS-VistA Interface Configuration Tool

Deploy the RTLS-VistA Interface Configuration Tool using the WebLogic Administration Console.

Deploy the application from {APPROOT}/{environment}/staging

1. Deploy RtlsInterfaceAdmin.war to the cluster.

3.2.4. Configure RTLS-VistA Interface Configuration Tool

After the RTLS-VistA Interface Configuration Tool is installed, use the tool to define your site (e.g. a VA facility) and the jobs needed to interface between AEMS-MERS and Intelligent InSites. The RTLS-VistA Interface Configuration Tool is documented in the RTLS ESE Technical Manual. The instructions for how to define a site and set up jobs is documented in the Technical Manual. Follow the high level steps below:

1. If your VA facility is the first facility within a VISN to install RTLS, define the VISN.
2. Add your site.
3. Add and schedule the jobs needed to establish the interface between AEMS-MERS and RTLS.

Warning: The VistA Engineering EQUIPMENT INV file # 6914 requires backup before the interface jobs run to provide a snapshot of file for rollback if needed. See the RTLS AT Patch Installation Guide for backup instructions.

3.3. RTLS ESB Component Installation

To install the RTLS ESB application for the first time, complete the steps in section 3.3.1- 3.3.5. If you have already installed the RTLS ESB application and need to re-install, see Subsequent install section 3.3.6.

3.3.1. Initial Installation

1. Copy the setup tar file to {APPROOT}

   copy setup_esb.tar.gz to {APPROOT}

2. Extract and run the setup tar file.

   tar -zxf setup_esb.tar.gz

The following table lists the directories extracted and a description of each.

Directory/File	Description
{APPROOT}/prop	A directory to hold server-specific properties (i.e. assettrax.properties). The directory persists between application deployments.
{APPROOT}/schemas	A directory to hold application supplied schema.
{APPROOT}/report	A directory to hold the current audit-type reports for the application.
{APPROOT}/reporthistory	A directory to hold the historical audit-type reports for the application.

3.3.2. Populate the RtlsDomain Property File

The Mule ESB application domain component has a property file.

The file name and location are as follows:

{APPROOT}/prop/rtls.domain.properties

Property Key	Property Value	Definition/Origin
rtls.domain.host	localhost	Server name or IP address of listener.
rtls.domain.port	8082	This is the authorized port for RTLS applications to listen on.
rtls.domain.path	esb	Base path of all RtlsDomain ESB applications.
rtls.domain.timeout	60	

3.3.3. Deploy the RtlsDomain ESB Domain Component

The Mule ESB domain component declares shared resources that are utilized by all ESB applications. This component must be installed before any RTLS ESB applications. Any previously installed (non-domain) RTLS applications MUST be uninstalled and, after deploying the RtlsDomain component, the domain-enabled version of the RTLS applications must be installed. Domain applications are programmatically bound to a specific version of the RtlsDomain component.

To deploy the domain component, the RtlsDomain-1.0.zip file should be placed in the ${MULE_HOME}/domains directory.

3.3.4. Populate the Asset Tracking Property File

The Mule ESB application component has a property file. Use the information gathered from the installation and configuration of the application components and pre-requisite software to populate the fields in the file. Only the properties values should be modified. The following table provides a list of the properties to be defined. Within the assettrax.properties file, userids and passwords are encrypted using Triple-DES cryptography (PBEWithMD5AndTripleDES) which is then encoded as base64 string values.

The file name and location are as follows:

{APPROOT}/prop/assettrax.properties

Property Key	Property Value	Definition/Origin
async.strategy.maxBufferSize	5000	Asnc processing strategy maximum buffer size. ESB application internal usage. Do NOT modify.
async.strategy.maxThreads	500	Asnc processing strategy maximum threads. ESB application internal usage. Do NOT modify.
async.strategy.minThreads	50	Asnc processing strategy minimum threads. ESB application internal usage. Do NOT modify.
async.strategy.threadWaitTimeout	-1	Asnc processing strategy wait time out. ESB application internal usage. Do NOT modify.
client.ssl.password	cacerts.password	Password for the cacerts certificate. Provided by the system administrator.
client.ssl.path	/path/to/cacerts/certificate	Path to the cacerts certificate. Provided by the system administrator.
insites.locator.file	C:/Projects/Configuration/prop/insitesConfig6.dat	File path to the Insites connections configuration file.
insites.not.responding.service.interval	0	Intelligent Insites property configuration time interval usage. Do NOT modify. Corresponds to the configurable interval for an internal Insites process that determines whether or not an asset's active tag transmitter has been detected on the Wi-fi network during the past interval period (specified in

Property Key	Property Value	Definition/Origin
		hours).
base.path	esb	Base path for all RtlsDomain services
local.assettrax.path	assettrax/services	path prefix to all services, appended onto the base.path
local.assettrax.host	localhost	The host name for the application (server name) (must resolve to an address that the rtls.domain.host is configured for in the rtls.domain.properties file)
local.assettrax.port	8082	The shared port number for all RtlsDomain applications (must match the rtls.domain.port value in the rtls.domain.properties file)
local.assettrax.timeout	30000	ESB internal server transaction timeout usage. Do NOT modify.
locationreportdir	/data/rtls/esb/report/	Directory of where the unmapped report for Intelligent Insites is temporary stored.
locationreporthistorydir	/data/rtls/esb/reporthistory/	Directory of where the unmapped historical reports for Intelligent InSites are stored.
rtls.timeout	300000	Connection timeout for RTLS
security.filter.realm	default	authentication realm
security.user.id		Basic authentication userid for the Mule ESB endpoint.
security.user.id		user id
security.user.password	basic.authentication.password	Basic authentication password for the Mule ESB endpoint.
vista.locator.file	/data/rtls/esb/prop/vistaConfig6.dat	Path to the vistaConfig.dat which defines the primary/substation relationships and the connection to the corresponding WebLogic service.

Property Key	Property Value	Definition/Origin
vista.timeout	300000	Connection timeout for VistA

3.3.5. Start and Verify the Mule ESB Application

For each of the ESB servers, start the RTLS ESB application and verify it is running correctly.

1. Start Mule server: ${MULE_HOME}/bin/mule start.
2. To verify mule started successfully, check to see if the esb-assettrax-1.0.x.x-anchor.txt file was created under the ${MULE_HOME}/apps.
3. Open the log file "assettrax.log" under ${MULE_HOME}/logs. Ensure the log file looks similar to the content in the figure below.

Figure 1 Sample of Successful Result for assettrax.log File

```
************************************
* Application: esb-assettrax-1.0.??                          *
* OS encoding: \, Mule encoding: UTF-8                *
*
*
* Agents Running:                                                          *
*   JMX Agent                                                                 *
*   DevKit Extension Information                                *
*   Batch module default engine                                 *
*   Wrapper Manager                                                    *
************************************
```

3.3.6. Subsequent Installation

To re-install the RTLS ESB application, complete the following steps.

1. Make a copy of the currently installed esb-assettrax-1.0.x.x application as a backup. If the original application package is available, use that as the backup copy.
2. Go to the directory ${MULE_HOME}/apps and remove the application by issuing the command "rm esb-assettrax-1.0.x.x-anchor.txt".
3. Shutdown Mule EE by issuing the command {MULE_HOME}/bin/mule stop.
4. Make changes to the assettrax.properties file, if needed.
5. Place the new esb-assettrax-1.0.x.x.zip file under ${MULE_HOME}/apps.
6. Start Mule server ${MULE_HOME}/bin/mule start.
7. Check assettrax.log to ensure esb-assettrax-1.0.x.x application is up and running.

4. Implementation Procedure

4.1. System Configuration

4.1.1. Configuring SSL/TLS

There are many different individual steps involved in configuring SSL. Expressing the full configuration of SSL is beyond the scope of this guide. A couple of items to note:

- WebLogic includes a FIPS approved crypto package for SSL that is configured for the solution.
- The current version of Mule ESB does NOT have an approved crypto module, or a facility to integrate one. The solution deploys an Apache proxy to offload the FIPS approved crypto SSL with localhost-only communication over non-SSL. A future upgrade to Mule ESB 3.5 would allow integration of FIPS SSL support with purchase of a 3rd party FIPS crypto provider.
- The AT Patch Installation Guide contains the recommended location for the imported keystore.

The following generalized steps are provided to give the reader an understanding of the process to install a certificate.

1. Modify configuration files to point to the new certificate.
2. Import the certificate to create a trusted chain within your keystore by importing the Root, Sub, and then the new SSL certificate.
3. Check to make sure the new certificate is working.

4.1.2. Configuring Site Information

To deploy the Asset Tracking interface, the network connection information is defined in both the vistaConfig.dat file and the insitesConfig6.dat. To establish the new connection, a Reload Locator Service is called. This allows the connection to be established without having to stop and re-start the RTLS ESB application.

The entries below should be defined, one per primary station. The VISN field is informational only, allowing locating the correct group of entries in a multi-VISN (i.e., AITC) installation easier.

The format of the Config.dat files is as follows:

VISNxx; primary station; <sites>;<timezone>;reportUser; <hostname>|<port>|<userid>|<password>

Where:

- VISNxx = VISN identifier
- Primary station = the primary station that is associated with a VistA instance

- sites = a comma delimited string of all CL stations/substations hosted within the primary station's VistA instance
- time zone = the time zone designation. Valid values are:
    - America/New_York for Eastern
    - America/Chicago for Central
    - America/Denver for Mountain
    - America/Los_Angeles for Pacific
- ReportUser = a comma delimited string of Intelligent InSites users who will receive reports.
- hostname = the Fully Qualified Domain Name (FQDN) of the WebLogic server
- port = port
- userid = the userid for the RTLS ESB application to communicate with WebLogic webservices
- password = password

Note: The connection is encrypted. The following is an unencrypted example:

```
VISN23;437;437;
America/New_York;joesmoe;visn23weblogicserver.med.va.gov|443|HPUser|HP_UserPassword
```

The following is an encrypted example:

```
VISN23;437;437;
America/New_York;joesmoe;ENC(PLUzckiJdiXiwQ3RP3fbRozQcnOV4B+1REwnjzgAb)
```

Follow the steps below to define a new site for the Asset Tracking Interface:

1. Copy the EncryptConnectionString.class file needed to encrypt the connection string from the code repository to the server.
2. Edit the {MULE_HOME}/vistaConfig.dat file properties as follows:
    a) Add a new line to represent the VISN if it doesn't exist.
    b) Add the site.
    c) Encrypt the hostname, port, userid and password using the EncryptConnectionString.class file.
    d) Insert the encrypted string into the properties file.
3. Save the file.
4. Run the Reload Locator Service: https://{server hostname}/locatorReload
5. Remove the EncryptConnectionString.class file from the server.

Note: For security reasons, it is important for the EncryptConnectionString.class file to be removed from the server after the string has been encrypted.

4.1.3. Creating Location External References in InSites

The AEMS-MERS-RTLS interface relies on each location within Intelligent InSites having an <esf-room-number> location external references. Follow the steps below to establish the external references within InSites:

1. Invoke the following endpoint on the ESB:

   https://{server hostname}/esb/assettrax/services/location/extref/migration?station=<station>&vista=<VistA Instance>

4.2. Database Tuning

N/A

5. Back-Out Procedure

5.1. Back-Out Strategy

The high level steps defining the back out strategy for the RTLS AT interface application are as follows:

1. Monitor installation – use logging and other diagnostic methods to monitor and check the installation as the components are individually installed.

2. Document errors – Document and resolve errors that occur during the installation process.

3. If any of the installed RTLS AT application components has a severe error and needs to be backed out, the RTLS System Operations Manager contacts the RTLS Technical Director to communicate that Back- out/Uninstall Procedures are being followed.

4. Follow the procedures for individual components as documented in Section 5.6.

5. The RTLS Technical Director will notify VA OIT staff that the documented Back Out process was followed.

6. Document the entire scenario in an Issue log.

5.2. Back-Out Considerations

5.2.1. Load Testing

N/A

5.2.2. User Acceptance Testing

User Acceptance Testing was performed and accepted on all of the RTLS ESE Interfaces, including the Asset Tracking interface. To view the consolidated daily reports delivered at the conclusion of testing each day, see the RTLS ESE Formal Test Results.

5.3. Back-Out Criteria

Back-out will be necessary if any of the installed RTLS AT application components has a severe error.

5.4. Back-Out Risks

The risk involved with backing out of the RTLS AT interface applications is low because RTLS can function without the interface running.

5.5. Authority for Back-Out

The RTLS AT interface application is a subset of the larger national RTLS deployment. The acceptance of all hardware and software is documented in a formal facility acceptance plan. The facility Point of Contact (POC) responsible for acceptance would also be the person with authority to require back-out and accept potential risks.

5.6. Back-Out Procedure

The following sections provide instructions on how to remove the components from the system.

5.6.1. VistaService Web Services Back-Out

The steps necessary to back-out the VistaService Web Services include deleting the deployment using the WebLogic Administration Console and then manually modifying the configuration files to remove content related to the services.

1. Undeploy the VistaService Web Services.
2. Edit the following files to remove the references to VistaService Web Services:

Reference Step	File(s)
3.1.4 Configure HEV_CONFIG/log4j.xml File	Log4j.xml
3.1.7 Setup and Configure Security	Files referenced in: create_vistasvcs_security.py

5.6.2. RtlsInterfaceAdmin Application Back-Out

The steps necessary to back-out the RtlsInterfaceAdmin application include deleting the deployment using the WebLogic Administration Console, removing the JDNI bindings related to the application, and dropping the associated databases.

1. Undeploy the RtlsInterfaceAdmin application.
2. Remove the JNDI data sources related to the application
3. Drop the following SQL databases created during the install:

   admin_tool

   rtls_scheduler

5.6.3. RTLS ESB Application Component Back-out/Uninstall

1. If the application is up and running, issue the following command from ${HOME_MULE}/apps directory:

   $rm esb-assettrax-1.0.x.x-anchor.txt

   The application is removed.

2. If there is no esb-assettrax-1.0.x.x-anchor.txt, issue rm –r esb-assettrax-1.0.x.x-anchor directory.
3. If there is no esb-assettrax-1.0.x.x-anchor directory, issue rm esb-assettrax-1.0.x.x-anchor.zip

5.6.4. RTLS ESB Application Component Back-out/Uninstall

1. If the application is up and running, issue the following command from ${HOME_MULE}/apps directory:

   $rm esb-assettrax-1.0.x.x-anchor.txt

   The application is removed.
2. If there is no RTLSServices-anchor.txt, issue rm –r esb-assettrax-1.0.x.x directory.
3. If there is no RTLSServices directory, issue rm esb-assettrax-1.0.x.x.zip

6. Rollback Procedure

The RTLS AT interface application does not have a transactional database. For information on back-up and rollback of the AEMS-MERS inventory data, see the RTLS AT Patch Installation Guide.

6.1. Rollback Considerations

N/A

6.2. Rollback Criteria

N/A

6.3. Rollback Risks

N/A

6.4. Authority for Rollback

N/A

6.5. Rollback Procedure

N/A

7. Appendix A: Troubleshooting

To diagnose an installation problem, use the following general troubleshooting steps. In addition, check the RTLS CL Patch Installation Guide Frequently Asked Questions (FAQ) section.

7.1. Run the Deployment Health Check

To verify that the communication is working between all distributed components, the following health checks are available.

ESB Application is up and running:

…/esb/assettrax/services/ping/txt

ESB Application can communicate with the Vista site:

…/esb/assettrax/services/ping/txt?system=vista&siteId=500

ESB Application can communicate with the Insites site:

…/esb/assettrax/services/ping/txt?system=insites&siteId=500

If there is a problem, the response will provide some information about what to check.

7.2. Check the Log Files

WebLogic, Mule ESB, and the RTLS Interface Application use a Java logging framework called log4j. The log files are defined in the log4j.properties file and also included below for convenience. The log4j.properties file can be used to configure different logging levels and also to enable and disable logging as needed. The log files may capture information about exceptions that arise during installation.

Log File	Description
/data/rtls/vistasvcs/{env}/log	WebLogic log file.
/data/mule/{env}/logs	Mule ESB log file.
/data/mule/{env}/logs/{flowname}.log	Log files containing information related to the Mule ESB flows.

Note: In the log file names above, {env} represents an environmental variable set during installation. {flowname} represents the process flow name associated with the interface component.

Template Revision History

Date	Version	Description	Author
February 2016	2.1	Changed title from Installation, Back-Out, and Rollback Plan to Installation, Back-Out, and Rollback Guide as recommended by OI&T Documentation Standards Committee	OI&T Documentation Standards Committee
December 2015	2.0	The OI&T Documentation Standards Committee merged the existing *"Installation, Back-Out, Rollback Plan"* template with the content requirements in the OI&T End-user Documentation Standards for a more comprehensive Installation Plan.	OI&T Documentation Standards Committee
February 2015	1.0	Initial Draft	Lifecycle and Release Management

The Template Revision History pertains only to the format of the template. It does not apply to the content of the document or any changes or updates to the content of the document after distribution. It can be removed at the discretion of the author of the document.