# Stella Application Stack

## Overview

Stella is comprised of web applications hosted on Apache Tomcat. Users do not interact with Tomcat directly. Rather, all HTTP / HTTPS requests are handled by Apache HTTP Server which then relays requests to Tomcat. Apache HTTP Server handles the encryption via SSL.

## Apache HTTP Server / httpd

Apache HTTP Server (Apache httpd) is the application's web server. While Apache Tomcat (hereafter simply called Tomcat) hosts the applications and generates web pages, the Tomcat layer is not exposed to the outside world. All requests on port 80 and 443 are routed through Apache httpd.

### Apache httpd as a Service

Apache httpd is configured as a service and can be viewed with the chkconfig command.

]# chkconfig –list | grep http  
httpd 0:off 1:off 2:on 3:on 4:on 5:on 6:off

The service can be started or stopped using the standard RHEL syntax

]# service httpd

Usage: httpd {start|stop|restart|condrestart|try-restart|force-reload|reload|status|fullstatus|graceful|help|configtest}

### SSL Hosting

Port 80 is visible simply to redirect users to SSL on port 443. Apache httpd is configured to generate a redirect for any request on port 80, regardless of URL, to the server root / on 443.

Apache httpd uses the mod\_ssl module to support SSL. Apache httpd is configured with the SSLCertificateFile, SSLCertificateKeyFile, and SSLCertificateChainFile in the config file /etc/httpd/conf.d/jimi.conf. Any updates to certificate files should be reflected in jimi.conf and httpd restarted.

### JK Connectors

The Tomcat redirector module, mod\_jk, facilitates connecting Tomcat to the web server Apache httpd. It requires 2 files, mod\_jk.so (installed in /etc/httpd/modules) and workers.properties in /etc/httpd.

The workers.properties file defines how many connections (workers) are maintained in a pool of workers. There are also pooling and connection information contained in this file, which should not need to be changed.

The VirtualHost entry for Stella informs Apache httpd to use the workers file for the connection to the tomcat application. This is in the file /etc/httpd/conf.d/jimi.conf. The version of mod\_jk installed uses the newer ajp13 protocol as ajp12 has been deprecated.

## Apache Tomcat

Apache Tomcat is the server that runs the Stella applications. It is a servlet container and web server and is used to host the Stella .war files. Stella is deployed via .war files which are a zipped and compiled project.

### Configuration

Tomcat is installed as a service by using a “Java Service Wrapper.” Tomcat is deployed in the folder /usr/share/hli/apache-tomcat-7.0.56 where there is a symlink simply called tomcat

tomcat -> apache-tomcat-7.0.56

Tomcat has already been configured to run Stella. The app files are hosted in /usr/share/hli/tomcat/webapps

The Java Service Wrapper is where Tomcat startup configuration is done. This is done via the wrapper.conf file located in <tomcat-root>/wrapper.conf. The following server elements are defined in the wrapper.conf file:

Java classpath

wrapper.java.command=/usr/lib/java/java\_hli/bin/java   
wrapper.java.classpath.1=/usr/share/hli/tomcat/lib/wrapper.jar  
wrapper.java.classpath.2=/usr/share/hli/tomcat/bin/bootstrap.jar  
wrapper.java.classpath.3=/usr/share/hli/tomcat/bin/tomcat-juli.jar

Memory management

wrapper.java.additional.2=-Xms1284m  
wrapper.java.additional.3=-Xmx1284m  
wrapper.java.additional.4=-XX:MaxPermSize=512m  
wrapper.java.library.path.1=/usr/share/hli/tomcat/lib

Logging configuration

wrapper.java.additional.1=-Djava.util.logging.config.file=/usr/share/hli/tomcat/conf/logging.properties  
wrapper.logfile.format=LPTM  
wrapper.logfile.loglevel=INFO  
wrapper.logfile.maxsize=1m  
wrapper.logfile.maxfiles=10

### Running Tomcat as a Service

To use the Java Service Wrapper to run Tomcat as a service, several pieces must be in place. There is an init script placed in /etc/init.d called tomcat which is a symlink to a script to start the wrapper.

/etc/init.d/tomcat -> /usr/share/hli/tomcat/bin/tomcat

The tomcat file must be configured with the location of the other wrapper elements on the system. This is also where a user can be specified under which tomcat is run.

BASE\_DIR="/usr/share/hli/tomcat"  
WRAPPER\_CMD=$BASE\_DIR"/bin/wrapper"  
WRAPPER\_CONF=$BASE\_DIR"/conf/wrapper.conf"   
RUN\_AS\_USER=tomcat

## Server Management

### Stop / Start / Restart

Tomcat and Apache httpd servers are deployed as services on Redhat. Standard Redhat service manament commands are used to stop and start. The services are registered to start, which can be checked using the chkconfig command.

]# chkconfig --list | grep tomcat tomcat 0:off 1:off 2:on 3:on 4:on 5:on 6:off tomcat6 0:off 1:off 2:on 3:on 4:on 5:on 6:off

]# chkconfig --list | grep httpd httpd 0:off 1:off 2:on 3:on 4:on 5:on 6:off

It is recommended that Stella first be stopped before issuing a restart to the Redhat server. Normally the stop command happens very quickly but can take longer if the application is in the middle of a file import. Simply issue the command:

]# service tomcat stop

You can use tail -f to tail the logfile to watch for 'wrapper stopped'

]# tail -f /usr/share/hli/tomcat/logs/wrapper.log

Once tomcat is stopped, you can restart the host server.

### Deploying a new version of Stella

Applications deployed to Tomcat are held in the /usr/share/hli/tomcat/webapps directory. They are deployed as single files jimi.war and stella.war. When Tomcat sees a new application it is inflated to a folder of the same name. So inside webapps there is a jimi.war file, but also a folder structure named jimi.

First step is to stop Tomcat. ]# service tomcat stop

Now backup the currently deployed war. Rename it to something other than .war so it doesn't get used by the server. For example: ]# mv jimi.war jimi.war.bak

Remove the jimi directory entirely as we'll get a new directory created when Tomcat starts back up.

Copy the new jimi.war file into the webapps directory. Start tomcat: ]# service tomcat start

The new jimi directory will be created. Tail the log file

]# tail -f /usr/share/hli/tomcat/logs/wrapper.log

Watch for 'Server Started in x seconds.' The server is now back up and running and can be accessed via the web. There is no need to restart Apache httpd server. The JK Connectors will automatically reconnect to Stella when it is started.



# File Movement and Processing

## Overview

This section serves to as a technical overview of all processes involved with the day to day operations and ongoing monitoring of both JIMI and the Stella products in regards to how data is loaded into those systems.

## Servers involved and their role:

### dc08sc1jdimstla – Windows 2008 R2 Standard Server

This is a Windows 2008 R2 Standard Server. It provides for temporary storage of data and/or transport of files that will be fed into one of the target systems (JIMI or Stella). The primary mechanisms for storage and transport are:

IBM Lotus Domino Server R 8.5.3 – This Domino server replicates a database from the HarrisLOGIC data center across a common VPN. Files sourced from HLI are attached to records in this database. The encrypted database is replicated to the Dallas County Domino server where those attachments are saved to a local folder and then deleted from the Domino database. The local folders where these files are saved reside within the Windows File Share referenced in the following item.

Windows File share: [\\dc08sc1jdimstla\exports](//dc08sc1jdimstla/exports/) – This file share makes all files transported to the Dallas County data center via Domino database replication available on the Dallas County network. This share is secured and accessible using the run-time Active Directory account for JIMI Bot. Access allowed:

SYSTEM – Allows the IBM Lotus Domino server to detach files.

SVC\_JIMI\_BOT – Allows the Windows file share in II.a to be mounted to the file system of the stellawebapp server.

#### Folders and their purpose:

* admin – Files containing encounter data targeted for Stella. Will be imported regardless of stated agency
* backups – used for temporary storage when troubleshooting
* client – Files containing demographic data targeted for Stella
* encounters – Containing folder. Subfolders will hold encounter data files target for Stella
* encounters\1629093118 – Files containing encounter data for the agency: ACS
* encounters\1629093119 – Files containing encounter data for the agency: CSP
* encounters\Z000000001 – Files containing encounter data for the agency: JAIL (not currently used)
* jimi – Containing folder. Subfolders will hold data files target for import by JIMI.
* client – Files containing demographic data. Source: HLI extract of NTBHA feed obtained from Value Options. Transferred via IBM Lotus Domino application.
* infoint – Files containing bookin data. Source: Dallas County AIS system. Currently transferred via IBM Lotus Domino application after FTP from Dallas County systems to HLI SFTP server. Transfer mechanism to be replaced by direct transfer from Dallas County system to this folder.
* meds – Files containing medication service data. Source: HLI extract of NTBHA feed obtained from Value Options. Transferred via IBM Lotus Domino application.

### stellawebapp.dallascounty.org

This server is the primary application server that runs both application UI and the backend import processes. This section serves to highlight the import processes.

### /var/imports/

This is the mount point for the Windows share: [\\d08sc1jdimstla\exports](//d08sc1jdimstla/exports/). This provides access to files contained in the remote Windows share by processes and applications running on the stellawebapp server. The folder structure contained in this folder is controlled by the hosting Windows server and should not be altered using the Linux operating system.

### /root/cronScripts/ and /root/cronlogs

Shell scripts contained in this folder run on a schedule (via root crontab) to move files out of the mounted folder structure onto a local file system where they will actually be processed. The shell scripts in this folder are named in a manner to tell you which schedule they follow. Each script in this folder runs shell scripts contained in one of the subfolders jimi or stella. Results from each of those is piped into a plain text .log file contained in the /root/cronscripts folder, divided into a subfolder according to application: jimi or stella.

Files are first moved to a local 'holding' folder which serves the purpose of actually moving the files across the wire onto the local file system.

Files are subsequently moved from the local 'holding' folder into a 'watched' folder where processes running in the Tomcat server perform the actual import. Here is the current crontab configuration:

**\*/5 \* \* \* \* /root/cronScripts/copyFilesMaster-5Min.sh**

**0 18 \* \* 5 /root/cronScripts/copyFilesMaster-Friday-1800.sh**

**0 08 \* \* 6 /root/cronScripts/copyFilesMaster-Saturday-0800.sh**

**0 20 \* \* 6 /root/cronScripts/copyFilesMaster-Saturday-2000.sh**

**copyFilesMaster-5Min.sh** runs:

* jimi/copyInfoInt.sh >> /root/cronlogs/jimi/copyInfoInt.log
* stella/copyStellaClient.sh >> /root/cronlogs/stella/copyStellaClient.log
* stella/copyStellaEncounters1629093118.sh >> /root/cronlogs/stella/copyStellaEncounters.log
* stella/copyStellaEncounters1629093119.sh >> /root/cronlogs/stella/copyStellaEncounters.log

**copyFilesMaster-Friday-1800.sh** runs:

* jimi/copyClients.sh >> /root/cronlogs/jimi/copyClients.log

**copyFilesMaster-Saturday-0800.sh** runs:

* jimi/copyMeds.sh >> /root/cronlogs/jimi/copyMeds.log

**copyFilesMaster-Saturday-2000.sh** runs:

* jimi/copyEncounters.sh >> /root/cronlogs/jimi/copyEncounters.log

### Specific Shell Script detail

* copyInfoInt.sh –Handles files from the AIS system targeted for JIMI.
* copyClients.sh –Handles files from the NBTHA feed containing demographic data targeted for JIMI.
* copyMeds.sh –Handles files from the NBTHA feed containing medication data targeted for JIMI.
* copyEncounters.sh - Handles files from the NBTHA feed containing provider encounter data targeted for JIMI.
* copyStellaClient.sh –Handles files from the NBTHA feed containing demographic data targeted for Stella.
* copyStellaEncounters1629093118.sh –Handles files from the ACS Crisis EMR system containing encounter data targeted for Stella.
* copyStellaEncounters1629093119.sh –Handles files from the CSP Crisis EMR system containing encounter data targeted for Stella.

# System Monitoring:

## Overview

This section serves to point out where data flows can or should be monitored in order to insure a smooth flow of data to both systems.

## Windows File share: [\\dc08sc1jdimstla\exports](//dc08sc1jdimstla/exports/)

The following folders within this network file share *may* be monitored for a backlog of files though this is not a requirement. A backlog of files here might indicate that:

* Stellawebapp server has dropped the mount and is not moving files out of these folders.
* The Tomcat instance running on stellwebapp is down or hung.

### JIMI

* jimi\infoint – Files should be processed every 5 minutes, 24 x 7
* jimi\client – Files should be processed every Friday night at 6:00 PM
* jimi\meds – Files should be processed every Saturday morning at 8:00 AM
* jimi\encounters – Files should be processed every Friday night at 6:00 PM

### Stella

* client – Files should be processed every 5 minutes, 24 x 7
* encounters\1629093118 – Files should be processed every 5 minutes, 24 x 7
* encounters\1629093119 – Files should be processed every 5 minutes, 24 x 7

## stellawebapp.dallascounty.org

The following folders *should* be monitored for ANY files which indicates a failure to import:

### Stella

* /var/import/stella2/client/.failed
* /var/import/stella2/encounter/1629093118/.failed
* /var/import/stella2/encounter/1629093119/.failed
* /var/import/stella2/encounter/Z000000001/.failed

### JIMI

* /tmp/jimi/client/.failed
* /tmp/jimi/infoint/.failed
* /tmp/jimi/meds/.failed

The following folders *should* be monitored for the regular creation of files which indicates successful import processing:

### Stella

* /var/import/stella2/client/.done
* /var/import/stella2/encounter/1629093118/.done
* /var/import/stella2/encounter/1629093119/.done

### JIMI

* /tmp/jimi/client/.done
* /tmp/jimi/infoint/.done
* /tmp/jimi/meds/.done

The graphic on the next page highlights how files are currently moved from the HarrisLOGIC data center to the Dallas County data center.



## Some suggested alternatives to the Windows hosted IBM Lotus Domino server currently used only for the purpose of transporting files from the HLI data center to the DC data center:

1. A new or existing SFTP process. Wayne referenced something currently in use during our meeting but I could not catch the name. HLI and other participating agencies could be setup with their own account for sending files. A mechanism would then have to be developed to move files from that server to the stellawebapp file system. If Windows based, the existing mount of a network file share approach can be leveraged. If Linux hosted, merely mounting the volume between the two Linux servers would suffice.
2. Create a Windows File Share on an existing server as a replacement for the current file share ( or use the existing Windows host without the Domino server component ). This assumes the existence of a production server that could also host this share. Inbound files from HLI (or other participating agency ) could then place files directly into this share across a VPN leveraging an agency specific user account. Current processes use the Jimi/Stella runtime account to mount the Windows File Share to the stellaweb file system and should be migrated to use an HLI specific account/runtime that leverages a more refined level of access.