

# VAR Utility Resources 2.1.x - Installation Instructions

- [Assumptions](#)
- [Summary](#)
- [Service and Data Source Dependencies](#)
- [Create WebLogic Server](#)
- [Property Overrides](#)
  - [Authorization MDWS Keys Configuration](#)
  - [Authorization Services and OAuth Configuration](#)
  - [VAR OAuth Configuration](#)
  - [MongoDB Configuration](#)
  - [Default Messages Configuration](#)
- [Proxy Configuration](#)
- [Import Clinical Services Collection \(if it has not been done\)](#)
  - [Submitted Request Limit \(Enterprise Level\)](#)
- [Register Sites Supporting VAR \(if it has not been done\)](#)
- [Enable Primary Care direct scheduling](#)
- [CDW Check List](#)
- [Deployment](#)
- [Validation](#)
- [Rollback procedures \(2.1.2 to 2.1.1\)](#)

## Assumptions

- Operations personnel is familiar with JEE applications and knows how to deploy applications into a WebLogic environment.
- Operations personnel knows how to take backups and revert deployments if there are failures.
- Operations personnel knows how to procure environment-specific property values.

## Summary

This document outlines the instructions for installing the **VarUtilityServices**.

<b>Service Name</b>	VAR Utility Services
<b>Enclave</b>	Staff Only
<b>WebLogic Target</b>	VARUtilityResources - PS Cluster Web Logic 10.3.6
<b>Java Runtime</b>	JRE 1.7

## Service and Data Source Dependencies

**VarUtilityServices** has dependencies on the following resources:

Dependency	Type	Dependency Type	Description
Authorization Services v4.4.x	Service	Runtime	Verify this service is deployed on the cluster.
MongoDB	Data Source (MongoDB Database)	Runtime	Verify this data source exists and is accessible. Otherwise create it.
OIA_MobileHealth	Data Source (CDW)	Runtime	Verify this data source exists and is accessible via JNDI lookup under the name "OIA_MobileHealth"; otherwise, create it.

## Create WebLogic Server

Create an appropriate weblogic server. Current naming convention is as follows:

<hostname><ApplicationName> e.g. swin10j7di01VARUtilityServices

This is being done to remove confusion on which server the application is deployed on.

## Property Overrides

**VarUtilityServices** uses Spring property manager to configure property files specified in the classpath.

1. Verify that `/opt/applications/{ClusterName}` (e.g. `ClusterName=VARUtilityResources`) folder exists and is specified in the class path using Web Logic Node Server startup configuration. Otherwise, create the folder and add it to the classpath.
2. Verify that `/opt/applications/{ClusterName}/VarUtilityServices/v2p1m2/properties` folder exists, otherwise create it.
3. Create a `/opt/applications/{ClusterName}/VarUtilityServices/v2/p1/m2/properties` folder for mongo. properties only

Create the following property files under `/opt/applications/{ClusterName}/VarUtilityServices/v2p1m2/properties`.

Add mongo.properties under `/opt/applications/{ClusterName}/VarUtilityServices/v2/p1/m2/properties`.

## Authorization MDWS Keys Configuration

### mdws-keys.properties

```
mdws.keys.authorized=SD SUPERVISOR
```

## Authorization Services and OAuth Configuration

### oauth.properties

```
oauth.validateTokenUri=https://<environment-address>/AuthorizationServices  
/rest/validateToken
```

## VAR OAuth Configuration

Obtain values of **VAR Utility <Client ID>** and **<Secret>** registered with **Authorization Services**.

ClientId and Secret needs to be added to the oauth-clients.xml in Auth Services configuration.

TBD: May need CCB ticket??

### var-prod-oauth.properties

```
oauthDetails.clientId=varUtility  
oauthDetails.clientSecret=<VAR Utility secret>  
oauthDetails.redirectUri=https://<environment-address>/va-tool-set-2.0.0  
oauthDetails.tokenUrl=https://<environment-address>/AuthorizationServices  
/oauth/token  
oauthDetails.scope=read
```

## MongoDB Configuration

### **mongo.properties**

```
// Comma separated ip and port to support replica sets
mongo.host=<MongoDB ip address with port #>
mongo.user=<var-utility service-specific account> (See PMDS ticket for
this service)
mongo.secretkey=<password>
mongo.database.name=var-utility
mongo.ssl=true
mongo.useAuthentication=true
```

## **Default Messages Configuration**

### **defaultmessages.properties**

```
default.messages.
id=appointmentNoPreferredDatePopup|typeOfCareDropdown|noClinicRadioButtons
default.messages.text=Scheduling Clerks may be able to find additional
dates and times for you. To request assistance, change the scheduling
method you selected to "Request help scheduling" and submit the request
details. A clerk will contact you to provide assistance.|If you don't see
a Type of Care that matches your needs, contact your facility for
assistance.|If you don't see the clinic that matches your needs, contact
your facility for assistance.
```

## **Proxy Configuration**

Use the configurations below to setup Apache proxy. Restart the proxy after making changes to the /etc/httpd/conf.d/va-tool-set.ssl configuration file (e.g., `sudo service httpd restart`).

Verify that the following Include statement present in /etc/httpd2/conf.d/ssl.conf file, if not add the entry. If the statement exists no need to add /modify the statement.

Please add the following proxy configuration to the apache service to forward traffic to the deployment on WebLogic.

### **Apache Proxy Configuration (/etc/httpd/proxy-sites/va-tool-set.ssl)**

```
<Location /VarUtilityServices/v2>
    SetHandler weblogic-handler
    WebLogicCluster <environment-ip-address:port>
    WLProxySSL ON
</Location>
```

## **Import Clinical Services Collection (if it has not been done)**

Use the instructions provided by the following link to import the [Clinical Services Collection](#).

## Submitted Request Limit (Enterprise Level)

Set that the value for "submittedRequestLimit" in the above Clinical Services collection for each type of care to represent the Enterprise-level value. Facility-level values are set using the VA Tool Set application for each Facility. The Facility value is always less than or equal, and cannot exceed the Enterprise value.

## Register Sites Supporting VAR (if it has not been done)

Use **mongoimport** utility to import the list of facilities (sites) supporting the VAR application. Import data into the **siteSupportingVAR** collection in the **var-utility** database. A sample file illustrating the json file data structure is provided below. Refer to MongoDB documentation for syntax of the command to import a json file into the designated collection in the designated database specified. A sample partial command is provided below. Please issue the complete command for the environment being installed.

```
mongoimport --ssl --db var-utility --collection siteSupportingVAR --drop --jsonArray --file <sites supporting VAR json file> -u <username> -p <password> --authenticationDatabase <authentication database>
```

### Sample Supported Sites

JSON for PROD: [field-testing-sites-supporting-var.json](#)

JSON for other environments (dev-int): [sites-supporting-var-442.json](#)

JSON for other environments (SQA): [sqa-sites-supporting-var.json](#)

JSON for Demo: [sites-supporting-var-demo.json](#)

JSON file code below includes correct sites needed for installation into PROD.

### sites-supporting-VAR.json

```
[
  {
    "_id": "405",
    "name": "WHITE RIVER JUNCTION"
  },
  {
    "_id": "523",
    "name": "BOSTON"
  },
  {
    "_id": "541",
    "name": "CLEVELAND"
  },
  {
    "_id": "608",
    "name": "MANCHESTER"
  },
  {
    "_id": "689",
    "name": "CONNECTICUT"
  }
]
```

## Enable Primary Care direct scheduling

Use mongo shell to enable direct scheduling of Primary Care appointments. In VAR version 4.2.4 and VAR Utility version 2.1.1, Primary Care direct scheduling is no longer enabled by default, but requires explicit configuration for sites which support it. The script below will ensure that Primary Care direct scheduling remains enabled for those facilities which do not already have this setting applied. Refer to MongoDB documentation for syntax of the command to execute a script against the the var-utility database specified. A sample partial command is provided below. Please issue the complete command for the environment being installed.

```
mongo --ssl -u <username> -p <password> --authenticationDatabase <authentication database> var-utility set-  
primary-care-support.js
```

Download the the script: [set-primary-care-support.js](#)

## CDW Check List

Ensure that the **CDW** instance connected to this installation has the correct version of the following stored procedures and functions:

1. `OIA_MobileHealth.App.ap_GetChildFacilities_v1`

## Deployment

**VarUtilityServices** is a standard VAMF web service that has been developed and tested on WebLogic. After verifying the dependencies and property configuration, install the *Deployable Artifact* from the VDD page on the WebLogic cluster/server.

Deployment of WAR file can be performed either manually through the WebLogic Admin Console or through the following sample script:

```
# Please make sure the paths are correct and the parameters are replaced  
with correct values before executing the script on command line  
  
source /opt/oracle/middleware/wlserver_10.3/server/bin/setWLSEnv.sh  
java weblogic.Deployer -adminurl t3://{adminserverip}:{adminserverport} -  
user {userid} -password {password} -undeploy -name "VarUtilityServices-  
2.0.0"  
  
java weblogic.Deployer -adminurl t3://{adminserverip}:{adminserverport} -  
user {userid} -password {password} -deploy -upload -targets  
VARUtilityServices -name "VarUtilityServices-2.0.0" -source  
"VarUtilityServices-{version with build number}.war"
```

## Validation

Ensure that the WebLogic deployment state is **Active** and that <https://<environment-address>/VarUtilityServices/v2/> returns "**Welcome to VarUtilityServices**" with the correct manifest.

## Rollback procedures (2.1.2 to 2.1.1)

Steps to rollback the installation:

1. Undeploy **VarUtilityServices** from the Web Logic Cluster using Web Logic Admin console.
2. Remove `/opt/applications/{ClusterName}/VarUtilityServices/v2m1p2` folder
3. Install Var Utility Resources (v2.1.1) using these instructions

