REPUBLIQUE DE COTE D'IVOIRE
UNION – DISCIPLINE – TRAVAIL

MINISTERE DE LA FAMILLE
DE LA FEMME ET DE L'ENFANT

PROGRAMME NATIONAL DE PRISE EN CHARGE DES
ORPHELINS ET AUTRES ENFANTS RENDUS
VULNERABLES DU FAIT DU VIH/SIDA

OVC Data System Guidelines for Côte d'Ivoire

JUNE 2021

June 2021

Acknowledgments

Abbreviations

BD-OEV   OVC database

MFFE   Ministry of the Family, Women and Children

OEV   orphelins et enfants rendus vulnérables du fait du VIH (orphans and children made vulnerable due to HIV)

OVC   orphans and vulnerable children

PEPFAR   President's Emergency Plan for AIDS Relief

PN-OEV   National OVC Program (Programme national de prise en charge des orphelins et enfants rendus vulnérables du fait du VIH)

USAID   United States Agency for International Development

INTRODUCTION

The Côte d'Ivoire Ministry of Women, Families and Children (MFFE) oversees the National Program for Orphans and Vulnerable Children (PN-OEV). Its mandate is to develop OVC policies and to ensure their implementation in OVC-related programs. PN-OEV's main mission is to ensure that quality services are delivered to OVC and its goal is to bring about a qualitative change in the well-being and lives of these children, in accordance with national standards.

This revised guide to the administration of the revised national OVC database (BD-OEV, or referred to as the "OEV Web") was developed with technical support from Data.FI and financial support from USAID/PEPFAR. The installation chapter provides information on how to install the OVC Web application in various contexts, including online central server, offline local network.

The OEV Web application is based on the OpenMRS platform (version 2.3.2 Platform). This platform runs on all platforms for which there is a Java JDK, which includes the most common operating systems, such as Windows, Linux, and Mac. OpenMRS runs on the MySQL database system. OpenMRS is packaged as a standard Java web archive (WAR file) and therefore runs on all servlet containers such as Tomcat and Jetty.

The Data.FI team recommends a Linux-based operating system (either Ubuntu Server 16, 18 or 20), the MySQL database system version 5.6, and the Tomcat Servlet container as the preferred environment for server installations.

INSTALLATION

Server specifications

OpenMRS requires your server to have an appropriate amount of RAM, number of processor cores, and a fast disk. These recommendations should be considered rules of thumb, not exact measurements. OpenMRS scales linearly with the amount of RAM and the number of processor cores. The more you can afford, the better the application performance will be.

Minimum RAM requirements:

- 2 GB for a small instance
- 12 GB for a medium instance
- 64 GB for a large instance

Processor cores: Four processor cores for a small instance, eight or more processor cores for a medium or large instance.

Hard disk: SSD is recommended as a storage device. The minimum reading speed is 150 Mb / s, 200 Mb / s is good, 350 Mb / s or better is ideal. In terms of disk space, at least 100 GB is recommended, but will depend entirely on the amount of data contained in the tables of data values. Analysis tables require a significant amount of disk space. Plan ahead and make sure your server can be upgraded with more disk space if necessary.

Hard disk requirements: SSD hard disks with a minimum read speed of 150 Mb/s and minimum hard disk space of 100 GB

Software required for the server

The OVC database requires a number of applications to be installed. These applications are related to the OpenMRS core platform. OpenMRS therefore requires the installation of the following applications to function properly.

- Java JDK 7 and higher

- MySQL database version 5.6 or later

- Tomcat servlet container version 7 or later

Server configuration

This section describes how to set up a server instance of OpenMRS on 64-bit Windows, Linux with MySQL as the database system, and Tomcat as the servlet container. This guide is not intended to be a step-by-step guide per se, but rather to serve as a reference on how OpenMRS can be deployed on a server. There are many possible deployment strategies, and they will vary depending on the operating system and database you are using, as well as other factors.

For this guide, we assume that 8 GB of RAM is allocated for MySQL and 8 Go of RAM is allocated for Tomcat / JVM, and that a 64-bit operating system is used. We recommend that the available memory be split roughly equally between the database and the JVM. Don't forget to leave some physical memory for the operating system to perform its tasks, for example about 2 Go. Steps marked as optional.

MYSQL 5.6 installation

To install the MySQL database server on the server you will need to run the following command:

```
$ sudo apt-get install mysql-client-5.6 mysql-server-5.6
```

The previous command allows you to install the server and the tools allowing the administration of the server. During this installation you will be requested the password of the ROOT user. Be careful when writing down this password so that you never lose it.

JAVA installation

You run a self-installing executable file to unpack and install the JDK on the computers. The recommended Java JDK for OpenMRS is OpenJDK 8 (for OpenMRS version 2.3.3 and later). You can install it with the following command:

```
$ sudo apt-get install openjdk-8-jdk
```

Check that your installation is correct by calling:

```
$ java -version
```

Installation of Apache Tomcat 8

To install the Tomcat servlet container, we will use the Tomcat user package by calling:

```
$ sudo apt-get install tomcat8-user
```

This package allows us to easily create a new Tomcat instance. The instance will be created in the current directory. An appropriate location is the OpenMRS home directory.

```
$ sudo tomcat8-instance-create tomcat-openmrs
```

```
$ sudo chown -R openmrs:openmrs tomcat-openmrs/
```

This will create an instance in a directory called tomcat-openmrs.

Next, edit the file tomcat-dhis/bin/setenv.sh and add the lines below:

JAVA_HOME defines the location of the JDK installation.

JAVA_OPTS passes the parameters to the JVM.

- Xms defines the initial memory allocation to the Java heap memory space.
- Xmx sets the maximum memory allocation to the Java heap memory space. This should reflect the amount of memory you want to allocate to the OpenMRS software application on your server, but also the available memory.

Check that the path to the Java binaries is correct as they may vary from system to system. On AMD systems, for example, you might see /java-8-open jdk-amd64. Note that you need to adjust these values to your environment.

```
JAVA_HOME='/usr/lib/jvm/java-8-openjdk-amd64/'
JAVA_OPTS='-Xms4000m -Xmx7000m'
```

The Tomcat server configuration file is located in the file in the directory accessible as follows: tomcat-openmrs/conf/server.xml.

The element that defines the connection to OpenMRS is the Connector element with port 8080.

You can change the port number in the Connector element to a desired port if necessary. The relaxedQueryChars attribute is required to allow certain characters in the URLs used by the OpenMRS interface.

```
<Connector port="8080" protocol="HTTP/1.1"
  connectionTimeout="20000"
  redirectPort="8443"
  relaxedQueryChars="[]" />
```

OpenMRS deployment

The next step is to install the OpenMRS application on the previously configured Tomcat server.

This application can be downloaded from the application link and is a file with the .WAR extension. To download it, you will have to run the following command on the server command line:

```
$ wget
https://sourceforge.net/projects/openmrs/files/releases/OpenMRS_Platform_2.3.2/ope
nmrs.war/download
```

Then place the downloaded .WAR file in the webapps folder of the Tomcat server previously configured by typing this command:

```
$ mv openmrs.war tomcat-openmrs/webapps/openmrs.war
```

You can finally start the server and therefore the application by running the following command:

```
$ tomcat-openmrs/bin/startup.sh
```

OpenMRS configuration and installation of required modules

OPENMRS CONFIGURATION

Once the application is installed and working properly, it provided us a configuration page that allows us to define the basic configurations for the proper functioning of our application.

To be able to do this we need to have access to the application, and that is done by going to the server link on which we are working or if it is on a local server then we open a browser and launch the following link:

http://localhost:8080/openmrs ou http://127.0.0.1:8080/openmrs

Follow the different configuration steps and wait until the process is finished and the application asks us to log in as a user.

IMPORTING MODULES

OpenMRS is a modular application that has a database of several hundred modules. These modules can be downloaded from the official website of the platform modules: https://addons.openmrs.org .

For the proper functioning of the OVC web application, we essentially need the following official modules:

- Webservices. Rest v2.26.0 (see appendix)
- Legacy UI Module v1.8.1 (see appendix)
- Serialization Xstream v0.2.14 (see appendix)
- Reporting REST v1.11.0 (see appendix)
- Open Web Apps Module v1.12.0 (see appendix)
- Reporting v1.21.0 (see appendix)
- Calculation v1.2.1 (see appendix)
- HTML Widgets v1.10.0 (see appendix)

These modules are simple OMOD (OpenMRS Module) files that must be imported into the modules folder of the running application.

Two additional modules must be imported: one module for services and the other for user interface management. These modules are:

- **OVC Rest Module v1.0**: for specific services related to the OEV Web application (see Appendix),

- **Weboev**: the interface module which is a simple .ZIP file that must be imported via the Open Web Apps module (see appendix).

APPENDIX

OpenMRS has a modular architecture that allows modules to be easily added or removed from the system. Modules have full access to the system and can modify or improve the system behavior (https://wiki.openmrs.org/display/docs/Modules). Below are the repositories of the source codes of the different modules.

Webservices. Rest v2.26.0
(https://wiki.openmrs.org/display/docs/Webservices.rest+Module)

Legacy UI Module v1.8.1
(https://wiki.openmrs.org/display/docs/Legacy+UI+Module)

Serialization Xstream v0.2.14
(https://wiki.openmrs.org/display/docs/Serialization+XStream+Module)

Reporting REST v1.11.0
(https://wiki.openmrs.org/display/docs/Reportingrest)

Open Web Apps Module v1.12.0
(https://wiki.openmrs.org/display/docs/Open+Web+Apps+Module)

Reporting v1.21.0
(https://wiki.openmrs.org/display/docs/Reporting+Module)

Calculation v1.2.1
(https://wiki.openmrs.org/display/docs/Calculation+Module)

HTML Widgets v1.10.0
(https://wiki.openmrs.org/display/docs/HTML+Widgets+Module)

OVC Rest Module v1.0
(https://bitbucket.org/abelnzi/ovcrestmodule/)

Weboev
(https://bitbucket.org/abelnzi/bdoev/)