Lab Exercise: Install & Configure OHDSI Tech Stack

Instructor:	**Lee Evans** **LTS Computing LLC**
Time:	1.5 hours
Prerequisites:	**Laptop computer with 8GB of memory,** **VirtualBox installed and open** **OHDSI-in-a-box Virtual Machine image installed and running in VirtualBox** **(Virtual Machine image login id = "ohdsi" and password = "ohdsi")**
Student Knowledge:	Comfortable using command line & Chrome web browser Some exposure to the OHDSI web tools & RStudio

Learning Objective:

Upon completion of this lab exercise, you will be able to install and configure the OHDSI technology stack using the OHDSI Broadsea docker images.

During this lab, you will learn how to:

- Install the OHDSI Broadsea docker images
- Configure the OHDSI Broadsea images to connect to a postgreSQL database
- Run the Broadsea images
- Verify the OHDSI Atlas & OHDSI RStudio applications are running
- View the log files

Overview of Tasks:

- Install the OHDSI Broadsea docker images
- Configure the OHDSI docker images using the docker-compose.yml files
- Start the OHDSI docker containers and view the log files
- Configure the SOURCE and SOURCE_DAIMON tables using SQL
- Restart the OHDSI webtools docker container
- Verify the OHDSI Atlas & OHDSI RStudio applications are running
- View the log files in the Broadsea-Webtools container

Exercise 1

Install the OHDSI Broadsea docker images

Objective

The goals of this exercise are to:

- **Demonstrate how to pull the OHDSI Broadsea docker images from DockerHub**
- **List the docker images on your machine to verify the docker images are installed**

Instructions

Note: For this tutorial, the images are already pre-installed in the Virtual Machine and you should see a message that says the latest version is already installed. Usually it would take several minutes to download these two large docker images from the DockerHub web site.

1. Type the following command to get to the working directory for this tutorial:

```
cd ~/ohdsi_stack
```

2. List the installed docker images with this command: (the above two images will be included in the list):

```
docker images
```

Note: The list of docker images should include the ohdsi/broadsea-webtools image and the ohds/broadsea-methodslibrary

Exercise 2

Configure OHDSI docker images using the docker-compose.yml file

Objective

The goals of this exercise are to:

- **Locate the docker-compose.yml configuration file for the OHDSI Broadsea images**
- **Edit the docker-compose.yml file to specify the database connection**

Instructions

1. Double click on the Leafpad text editor icon on the desktop.
2. In the text editor click on the open file menu option.
3. Navigate to the file "/home/ohdsi/ohdsi_stack/docker-compose.yml" and click the open button to open it.
4. Edit the docker-compose.yml file to set the values shown in bold red below:

```
version: '2'

services:

  broadsea-methods-library:
    image: ohdsi/broadsea-methodslibrary
    ports:
      - "8787:8787"
      - "6311:6311"
    volumes:
      - ./home/rstudio:/home/rstudio
      - ./site-library:/usr/lib/R/site-library

  broadsea-webtools:
    image: ohdsi/broadsea-webtools
    extra_hosts:
      - "database:10.0.2.15"
    ports:
      - "8080:8080"
    volumes:
     - .:/tmp/drivers/:ro
     - .:/tmp/achilles-data-reports/:ro
    environment:
```

```
- WEBAPI_URL=http://127.0.0.1:8080
- env=webapi-postgresql
- datasource.driverClassName=org.postgresql.Driver
- datasource.url=jdbc:postgresql://database:5432/ohdsi
- datasource.ohdsi.schema=ohdsi
- datasource.username=ohdsi_admin_user
- datasource.password=admin1
- spring.jpa.properties.hibernate.default_schema=ohdsi
- spring.jpa.database-platform=org.hibernate.dialect.PostgreSQLDialect
- spring.batch.repository.tableprefix=ohdsi.BATCH_
- flyway.datasource.driverClassName=org.postgresql.Driver
- flyway.datasource.url=jdbc:postgresql://database:5432/ohdsi
- flyway.schemas=ohdsi
- flyway.placeholders.ohdsiSchema=ohdsi
- flyway.datasource.username=ohdsi_admin_user
- flyway.datasource.password=admin1
- flyway.locations=classpath:db/migration/postgresql
```

Note: In the above docker-compose.yml configuration file we are configuring the following:

- *postgres database jdbc connection string, database userid and database password*
- *flyway database connection info and schema where WebAPI tables will be created/maintained*
- *WebAPI services IP address and port*
- *RStudio port*
- *OHDSI Web applications port (e.g. Atlas)*
- *Spring Hibernate default schema*
- *Spring Batch table prefix*

Exercise 3

Start the OHDSI Broadsea containers

Objective

The goals of this exercise are to:

- **Change the working directory to the directory where the docker-compose.yml file is located**
- **Start the ohdsi/broadsea-methodslibrary and the ohdsi/broadsea-webtools containers**
- **Verify that the containers are running**

Instructions

1. In the command line window that we opened in Exercise 1, enter the following command:

   ```
   cd /home/ohdsi/ohdsi_stack/
   ```

   Note: This will change the working directory to the directory where the docker-compose.yml file is located.

2. Enter the following command to verify that the docker-compose.yml file is in the current working directory:

   ```
   ls docker-compose.yml
   ```

   Note: The docker-compose.yml file name should be shown. If you see a "file not found" error message, then ask one of the faculty members for assistance.

3. Type the following command to start the two Broadsea containers:

   ```
   docker-compose up -d
   ```

4. Enter the following command to verify that the containers are running:

   ```
   docker-compose ps
   ```

   Note: If the containers are started successfully you will see the following two items listed:

   ```
   ohdsistack_broadsea-methods-library_1   /usr/bin/supervisord -c /e ...   Up
   1410/tcp, 0.0.0.0:6311->6311/tcp, 0.0.0.0:8787->8787/tcp
   ohdsistack_broadsea-webtools_1          /usr/bin/supervisord            Up
   0.0.0.0:8080->8080/tcp
   ```

   Note. If you don't see the above items, then ask one of the faculty members for assistance.

Exercise 4
Configure the SOURCE and SOURCE_DAIMON tables using SQL

Objective

The goals of this exercise are to:

- **Start the pgadmin3 Postgresql SQL client**
- **Connect to the CDMV5 postgresql database in the pgadmin3 SQL client**
- **Open the SQL queries (in the source_source_daimon.sql file) in the pgadmin3 client**
- **Execute the SQL queries to populate the SOURCE and SOURCE_DAIMON WebAPI configuration tables**
- **View the SOURCE table to verify the data that was inserted by the SQL queries**
- **View the SOURCE_DAIMON table to verify the data that was inserted by the SQL queries**

Instructions

1. Double click on the Pgadmin III (Elephant) icon on the OHDSI Virtual Machine desktop to start it.
   Note: It may take up to a minute for the Pgadmin III application to start.

2. Double-click on the database server connection called "ohdsi" to open the connection to the postgresql database. *Note: There's already a database server connection configured in the Pgadmin III client called "ohdsi".*

3. Click on the SQL magnifying glass in the pgadmin III menu bar to open a new query window

4. Click on file, open in the pgadmin III menu, double click on the ohdsi_stack directory and double-click the file called source_source_daimon.sql to open it.

5. Right click on the SQL code and click the select all option (to select all the text in the SQL query window) and click on the execute icon (small green right facing triangle) in the pgadmin III menu-bar to execute the selected SQL queries. After the queries have finished executing close the pgadmin III SQL query window.

6. In the tree on the left of the main pgadmin III window, click on tables to expand the list of tables in the OHDSI database Schema and then scroll down and right click on the SOURCE table and click on the "view data" option.
   Note: A results table showing the SOURCE table rows will be shown.

7. Resize the SOURCE table results table columns by dragging the column divider lines to the right with your mouse so you can see the new data we just inserted into the SOURCE table.

8. In the tree on the left of the pgadmin III screen right click on the SOURCE_DAIMON table and click view data.
   Note: A results table will be shown.

9. Resize the SOURCE_DAIMON results table columns by dragging the column divider lines to the right with your mouse so you can see the new data we just inserted into the SOURCE_DAIMON table.

Exercise 5

Restart the OHDSI Broadsea Webtools docker container

Objective

The goals of this exercise are to:

- **Restart the OHDSI Broadsea Webtools docker container, so it will load the WebAPI configuration data we just populated in the SOURCE and SOURCE_DAIMON tables.**
- **Verify that the OHDSI Broadsea Webtools Docker container is running.**

Instructions

1. In the command line window that we opened in Exercise 1, enter the following command to temporarily stop the docker container (it may take up to 30 seconds to stop):

   ```
   docker stop ohdsistack_broadsea-webtools_1
   ```

2. Enter the following command to restart the docker container:

   ```
   docker start ohdsistack_broadsea-webtools_1
   ```

3. Wait 30 seconds and then enter the following command to verify the docker container is running:

   ```
   docker ps
   ```

   Note. If you don't see "ohdsistack_broadsea-webtools_1" listed, then ask one of the faculty members for assistance.

Exercise 6

Verify the OHDSI Atlas & OHDSI RStudio applications are running

Objective

The goals of this exercise are to:

- **Verify the OHDSI Atlas & OHDSI RStudio applications have been successfully deployed and are up and running by opening them in the Chrome web browser.**

Instructions

1. Double click on the Chrome icon on the OHDSI VM desktop to start the Chrome web browser.

2. Enter the following URL to open the OHDSI Atlas web application:
   Note. Wait up to 5 minutes for the Atlas application to launch in the browser!

   http://localhost:8080/atlas/

3. Open a new tab in the chrome browser by clicking on the rightmost mini-tab at the top right

4. Enter the following URL to open the OHDSI RStudio web application and enter the default userid "rstudio" and password "rstudio" when the "Sign in to RStudio" page appears in the browser:

   http://localhost:8787

   Note. Wait up to 1 minute for the RStudio application to launch in the browser!

   Note. If the Atlas or RStudio web applications do not launch successfully then ask one of the faculty members for assistance.

Exercise 7

View the log files in the Broadsea-Webtools container

Objective

The goals of this exercise are to:

- **Open a bash shell inside a running Broadsea-webtools docker container**
- **View the log files**

Instructions

1. In the command line window that we opened in Exercise 1, enter the following command to connect to a running Broadsea-webtools docker container and open an interactive bash shell inside the container:

```
docker exec -it ohdsistack_broadsea-webtools_1 bash
```

2. Enter the following command to view the stderr log file inside the docker container:

```
tail /var/log/supervisor/*stderr*
```

Note. Include the "" wildcard chars in the above command.*

3. Enter the following command to view the stdout log file inside the docker container:

```
tail /var/log/supervisor/*stdout*
```

Note. Include the "" wildcard chars in the above command.*

Note. If you encounter any issues with connecting to the running Broadsea-webtools container in step 1 then ask one of the faculty members for assistance.

Appendix

If you want to install/update the Broadsea container, the commands to pull the OHDSI Broadsea docker images from the DockerHub website are as follows:

```
docker pull ohdsi/broadsea-webtools
docker pull ohdsi/broadsea-methodslibrary
```