Database installation guide

v1.3.0

Install SQL Server

Note: This installation guide is based on SQL Server 2017.

Download the desired version of SQL Server (for example, the Express version from https://www.microsoft.com/en-us/sql-server/sql-server-downloads).

When the installation wizard opens, select the manual configuration option in order to fine tune the installation process.

On the Features selection:

- In Shared Feature section, select SQL Client Connectivity SDK
- For installations with Data Warehouse, select Analysis Services, Reporting Services and Integration Services (not available for the Express edition)

On the Instance configuration, the default name (SQLEXPRESS) can be used, unless it is already used by another instance.

On Database engine configuration, select Mixed Mode (SQL Server authentication and Windows authentication) in Authentication Mode.

Continue the setup process until the installation is complete.

Configure SQL Server

Open the SQL Server Configuration Manager

- On the left panel, select SQL Server Network Configuration → Protocols for SQLEXPRESS (or the name of your SQL Server instance) → Enable Named Pipes and TCP/IP
- Select SQL Server Services → right click on SQL server (instance name) and select Restart

Initialise openIMIS database

To facilitate the setting up of the openIMIS database, it is suggested to install SQL Server Management Studio (SSMS). The following procedure is based on SSMS, but you can use the standard SQL Server prompt to proceed with the setup.

First, download the openIMIS database SQL files and migration scripts from Github repository (the source code ZIP file).

In SQL Server Management Studio:

- Create a new database for the openIMIS instance (i.e. openIMIS.X.Y.Z where X.Y.Z is the openIMIS database version).
- Execute the initial database creation script provided in the release sources (from the `Empty databases` folder).

Choose the database to restore based on your context. There are 3 types of databases (structurally identical, but they are pre-configured differently):

- Online: this is the default choice when deploying a central online server
- Offline: this mode is sometime used for remote insurance offices without connectivity. Note: the synchronisation of data with the central server is manual.
- Offline HF: this database can be used in remote health facilities without connectivity. Note: the synchronisation of data with the central server is manual.

Restauration can be done in SQL Server Management Studio

or the below command line can be used in a shell

Becareful to addapt the querries to your setup, in the command lines example those assumptions were made:

- `Online` database version 1.20 is to be restored
- The database is called `IMIS_DATABASENAME`
- The SQL server is called `SQL_Server_Name`

```
1    SqlCmd -E -S SQL_Server_Name -Q "RESTORE DATABASE [IMIS_DATABASENAME] FROM
     DISK='X:\PathToBackupFile\openIMIS_ONLINE_v1.2.0.bak'"
```

In case the database need to be restored on a server that doesn't have the same file structur as the initial server (e.g. from Windows to Linux), the new location of the mlf/mlf files can be specified

```
1    SqlCmd -E -S SQL_Server_Name -Q "RESTORE DATABASE [IMIS_DATABASENAME] FROM
     DISK=N'/tmp/openIMIS_ONLINE_v1.2.0.bak' WITH MOVE N'CH_CENTRAL' TO
     '/var/opt/mssql/data/IMIS.mdf', MOVE N'CH_CENTRAL_log' TO
     '/var/opt/mssql/data/IMIS_log.ldf'"
```

- **Execute the SETUP-IMIS stored procedure**

In SQL Server Management Studio

- In the Object Explorer: openIMIS (the openIMIS db name) → Programmability → Stored Procedures → dbo.SETUP-IMIS
- Right click on the dbo.SETUP-IMIS and execute the procedure.

or run this command in a shell

```
1    SqlCmd -E -S SQL_Server_Name -d IMIS_DATABASENAME -Q "EXECUTE [SETUP-IMIS]"
```

The result returned from the procedure should be 0.

If you prefer to initialize the database using the shell:

Be careful to adapt the queries to your setup, in the command lines example those assumptions were made:

- `Online` database is to be initialized
- The database is called `IMIS_DATABASENAME`
- The SQL server is called `SQL_Server_Name`

```
1   SqlCmd -E –Q "CREATE DATABASE IMIS_DATABASENAME"
2   SqlCmd -E -S SQL_Server_Name -d IMIS_DATABASENAME –i X:\PathToSQLFile\openIMIS_ONLINE.sql
```

Create a dedicated user with full privilege on the openIMIS database only:

- In the Security → Logins → right click and select "New Login…"
- In General page:

  - Give a login name (i.e. ImisUser)
  - Select SQL Server authentication and provide a password
  - Unselect Enforce password expiration
  - Change default database to openIMIS

- In User Mapping page:

  - Map IMIS db to ImisUser user
  - Give the role of db_owner

Upgrade the openIMIS database

Before updating the database make sure the database is not reachable (off line) for the applications (web, mobile,…).

If you want to update a production instance:

- Please duplicate the database (create a full backup of the database)
- Execute the steps below on the copy of the database
- If the migration script succeeded on the copy, then you can apply the migration script to the production instance.

This approach will prevent impacting the production if the migration script failed because of customizations in your openIMIS instance and it will give you an idea of the time required to update the database.

If an existing openIMIS database exists already, follow the next steps to upgrade it to the desired version:

- Do a backup of the database

  Use the backup tools available in SQL Server Management Studio

or run this command in a shell

Becareful to addapt the querries to your setup, in the command lines example those assumptions were made:

- The database is called `IMIS_DATABASENAME`
- The SQL server is called `SQL_Server_Name`

```
1    SqlCmd -E -S SQL_Server_Name -Q "BACKUP DATABASE [IMIS_DATABASENAME] TO
     DISK='X:PathToBackupLocation\[Name_of_Database].bak'"
```

- Download the openIMIS database SQL files and migration scripts from Github repository (the source code ZIP file).
- In SQL Server Management Studio, run the migration script on the openIMIS database

or using the shell:

```
1    SqlCmd -E -S SQL_Server_Name -d IMIS_DATABASENAME -i "X:\PathToMigrationScript\openIMIS
     migration v1.2.0 - v1.3.0.sql"
```