



This PDF is generated from authoritative online content, and is provided for convenience only. This PDF cannot be used for legal purposes. For authoritative understanding of what is and is not supported, always use the online content. To copy code samples, always use the online content.

iWD Deployment Guide

Migration from 8.5.1x to 9.0.x

5/7/2025

Migration from 8.5.1x to 9.0.x

Prerequisites

- iWD 9.0 requires Java 1.8. Support for Java 1.6 and 1.7 is withdrawn in release 9.0.

General Migration Procedure

1. Backup all databases.
 2. Upgrade iWD dependencies to supported versions (see [Preparing for Installation](#)).
 3. Stop all iWD 8.5 components.
 1. iWD Manager's Tomcat.
 2. iWD Runtime Node.
 3. iWD Web.
 4. Interaction Server.
 4. Install all components.
 1. Upgrade GAX plug-in to a new version (at least 9.0.002.07) by uninstalling the previous GAX plug-in and installing the new one (details [here](#)).
 2. History Node
 1. Install History Node using [this procedure](#). Set up JMS and JDBC Database Access Points (DAPs).
 3. Data Mart
 1. History Node should be installed already.
 2. Migration
 1. Create a new database for Data Mart.
 2. Complete a fresh [installation of Data Mart as a separate application](#).
 3. Copy all necessary settings from the iWD Runtime Node 8.5 application.
 4. Copy all customized ETL scripts to the iWD Runtime Node 9.0 ETL directory (**../iWD Runtime Node 9.0/etl**). Note: If your Kettle plugins fail to work on an Oracle database, please set the following database parameters:
 - JDBC URL: jdbc:evo:oracle:@//<databasename>:1521/<SID>
 - JDBC Driver Class: evo.database.oracle.EvoDriver
- For more information about setting database parameters please refer to [iWD GAX Plug-in Datamart help](#).

5. Create a Configuration Server database JDBC DAP with configuration option **[iWD]/role = cfgserver** and add it to the Runtime Node connections (details [here](#)).
 6. Add a **jdbc** section with options **url** and **driver-class** to all created DAPs listed in the connections (details [here](#)).
 7. Change or set the Data Mart application to a new one for all necessary solutions using **GAX / iWD / Datamart / <Solution>**.
 8. Using GAX, change the **ETL Scripts Directory** to a new one for each solution.
 9. Check that your Data Mart Business Structure Solution has a timezone set.
 10. Check that in the Data Mart application the **schedules/historical** and **schedules/intraday** are set as valid Quartz cron expressions.
 11. [Optional] Disable the EventLog as it was replaced by History Node and JMS event logger.
4. Perform [Data Migration from iWD 8.5.1x to iWD 9.0](#) if necessary.
 5. Stat Server Extensions
 1. Install the new 9.0 version. No other changes are required.
 6. iWD Manager
 1. iWD GAX Plugin must be updated already.
 2. Make sure iWD Manager 8.5 is stopped. Disable its autostart.
 3. Import new the iWD Manager 9.0 application templates (App and Server) as described in [Installing iWD Manager](#) and make applications from them.
 4. Install the iWD Manager 9.0 application itself to a separate folder.
 5. Copy the required settings from the old iWD Manager application to the new iWD Manager server application by adding a connection from iWD Manager Server to Interaction Server.
 6. Copy the required settings from the old iWD Manager application to the new iWD Manager Application by adding the necessary permissions for the iWD access group (to make users able to login, to read filter list, and so on).
 7. Migrate options from the old **iwd.properties** file to iWD Manager Server Application Options. The detailed description of the properties can be found [here](#).
 - Note: Parameter **iwd.cfgConnectionTimeout** is replaced by four independent parameters for each connection:
 - **cfgConnectionTimeout** for the Configuration Server connection.
 - **ixnConnectionTimeout** for the Interaction Server connection.
 - **ucsConnectionTimeout** for the Universal Contact Server connection.
 - **msgConnectionTimeout** for the Message Server connection.
 8. Migrate logging settings.
 1. If you use Centralized Logging, make sure that you've created a connection from iWD Manager Server to Message Server.
 2. Configure logging in accordance with the details in [Logging](#).
 9. Make sure that host and port of the primary Configuration Server, and the application name of the iWD Manager Server, are presented in the **General -> Command Line Arguments** of the

iWD Manager 9.0 Server.

10. Run the separate [iWD Manager Configuration Database to Configuration Server migration procedure](#).
 11. Open GAX and make sure that the **Scripts -> iWD Manager** folder is created and permissions to that folder are granted for the **EVERYONE** group.
7. iWD Web
1. Install a new iWD Web application as described in [this series of topics](#).
5. For each configured Solution you must complete following steps:
1. Add the corresponding History Node using **GAX > iWD > Business Structure > '<Solution>' > History Node Settings**. For more details please refer to the [iWD Business Solution Configuration](#).
 2. Make sure that the JDBC URL of every Interaction Server used by the Solution is configured in DAP. For more details please refer to the [iWD Business Solution Configuration](#).
6. Start:
1. All iWD 9.0 components.
 2. Interaction Server.

iWD Manager Configuration Database to Configuration Server Migration Information

iWD Configuration Database migration is performed manually by a Python script similar to that used in [8.5.0 to 8.5.1 migration](#). To run the script, you need to provide the iWD Manager Configuration Database connection parameters (URL, login and password), and the GAX URL, username and password. The Python script takes the data from the IWD Configuration Database and sends it into the Configuration Server via the GAX API.

Procedure

1. Install Python 2.7.
2. Add environment variables:
 - Path=C:\Python27;C:\Python27\Scripts
3. For Linux users some additional libraries may be required. To resolve this please deploy the following RPMs with the version appropriate for your OS version:
 - unixODBC-*.rpm (for example, for 64-bit RHEL6: unixODBC-2.2.14-14.el6.x86_64.rpm)
 - freetds-*.rpm (for example, for 64-bit RHEL6: freetds-0.91-2.el6.x86_64.rpm)
4. Download the required Instant Client packages for your platform. All installations require either the Basic or Basic Light package.
5. Unzip the packages into a single directory such as "instantclient_12_2".
6. Set your environment's library loading path (for example **LD_LIBRARY_PATH** on Linux, or **PATH** on Windows) to the directory created in Step 2.
7. Start your application.

8. Refer to any detailed instructions on the download page of your platform for more information. Note: Python and Oracle client versions must both be either 32- or 64-bit.
9. If there are any data with Unicode characters to be migrated, ensure that the GAX application is configured to support UTF-8 encoding. You can do the following:
 - For Windows, add `-Dfile.encoding=utf-8` to `JavaServerStarter.ini` under the **[JavaArgs]** section.
 - For Linux, add `export JAVA_OPTS="$JAVA_OPTS -Dfile.encoding=utf-8` to `setenv.sh`.
10. Restart GAX.
11. Unzip the migration.zip file. This should be located under the **config/ folder** in iWD Manager's installation directory.
12. Install dependencies.
13. From the folder where migration.zip was unpacked, run the following command:

```
pip install --upgrade -r requirements.txt
```
14. Run the migration script for each tenant, include the Environment tenant:

```
python main.py -g <gaxurl> -u <gax username> -p <gax password> -t <tenant> -y <db type> -d <db host> -r <db port> -n <db name> -s <db user> -w <db user password> --verbose
```

The script parameters are as follows:

- `-g, --gaxurl`—GAX URL address
- `-u, --username`—GAX user name
- `-p, --password`—GAX user password
- `-t, --tenant`—Tenant Name
- `-y, --dbtype`—iWD Configuration Database type (either Oracle or MSSQL)
- `-d, --dbhost`—iWD Configuration Database host
- `-r, --dbport`—iWD Configuration Database port
- `-n, --dbname`—iWD Configuration Database name
- `-s, --dbuser`—iWD Configuration Database user
- `-w, --dbpass`—iWD Configuration Database password
- `--verbose`—verbosity level

Oracle example:

```
python main.py -g http://192.168.123.132:8080/gax -u default -p password -t selenium -y oracle -d 192.168.123.123 -r 1521 -n xe -s iwd_user -w iwd_password --verbose
```

MSSQL example:

```
python main.py -g http://192.168.123.123:8080/gax -u default -p password -t selenium -y mssql -d 192.168.123.123 -r 1433 -n iwd_config -s iwd_user -w iwd_password --verbose
```

History Node Specific Migration Information

- [iWD History Node Migration](#)

TLS Configuration

To configure TLS, please refer to [Configuring TLS for iWD](#).