Performance Management Advisors Deployment Guide

Creating the Oracle Schema for Advisors

4/5/2026

Contents

Creating the Oracle Schema for Advisors

This page describes how to create a generic Oracle schema for Advisors. Each individual Oracle schema in an Advisors implementation has its own creation script.

All Oracle scripts are creation scripts except those that contain the word `migrate` in the name. Any existing schema with the same name must be dropped prior to running the scripts. Use the migration scripts when upgrading your software version. Always review the "readme" files, if supplied, along with the database scripts. The "readme" files can contain important details, specifics, and exceptions related to a particular release, which are not reflected in the general documentation.

If, due to security restrictions, administrator or security administrator access cannot be granted, the local Database Administrator (DBA) should implement the steps described in the procedure.

The procedure applies to an Oracle user who has permissions to create tablespaces, users, and to grant permissions. Follow your enterprise’s policies in production environments. If necessary, have the DBA create tablespaces, users, and grant permissions. Use scripts relevant to your environment after the DBA completes the work.

Examples of Schema/User Names

Advisors Component	Schema/user name	Notes
Platform	advisors_platformdb	Required for Advisors implementations.
CCAdv/WA		Use the Platform and Metric Graphing schemas.
FA		Uses the Platform schema.
Metric Graphing	advisors_mgdb	Metric Graphing schema. Required to run the CCAdv/WA dashboards and CCAdv XML Generator.
Advisors Genesys Adapter	advisors_gametricsdb	AGA metrics schema. Used by AGA to transfer Genesys real-time statistic values to CCAdv/WA. This schema is required for CCAdv and WA server installations only.

Before You Begin

You must perform all of the steps in the procedure on a machine where you have Oracle client or Oracle instant client installed. The installation scripts require SQL*Plus, which is installed as part of the Oracle client installation or added in addition to the Oracle instant client installation.

Verify that you have your system or session `ORACLE_HOME` or `TNS_ADMIN` environment variable and

`tnsnames.ora` content set properly. If you have full Oracle client installed, you can verify the connectivity to the database by running the following command:
`tnsping <alias for the oracle instance contained in the local tnsnames.ora file>`

It is important to use *<alias for the oracle instance contained in the local tnsnames.ora file>* as a response on all prompts where the database scripts ask you to *<Enter the database alias>*.

For example:
Your `tnsnames.ora` contains the following entry:

```
wolf =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = inf-wolf.qalab.com)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = orcl.qalab.com)
    )
  )
```

On the machine with full Oracle client installation, you can check the connectivity by typing the following command:

```
C:>tnsping wolf
```

The successful message will look as follows:

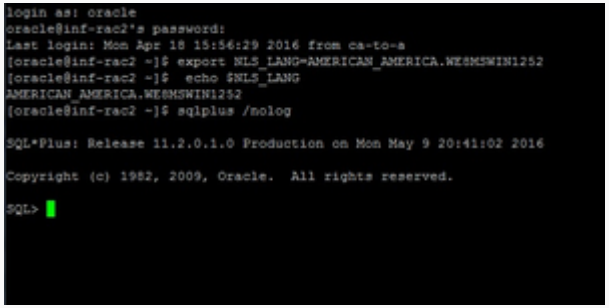
```
Used TNSNAMES adapter to resolve the alias
Attempting to contact (DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(HOST = inf-wolf.qaslab.com)(PORT = 1521))
(CONNECT_DATA = (SERVER = DEDICATED) (SERVICE_NAME = orcl.qalab.com)))
OK (0 msec)
```

Procedure

Procedure: Creating the Advisors Oracle Schema

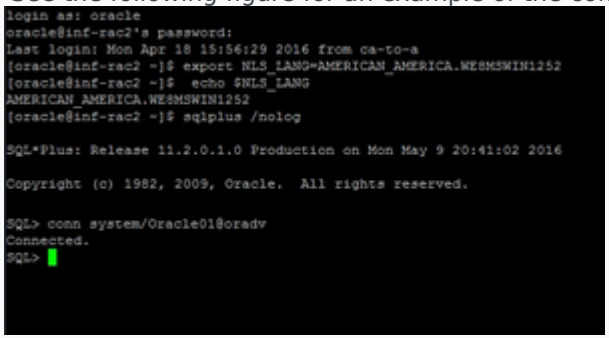
Steps

1. Copy all of your Oracle database scripts to a folder on the machine where you have the Oracle client installed. The path name for this location must not contain spaces.
2. On the machine where the Oracle client is installed, open a command prompt and change directory to the folder where the database scripts now reside.
3. Review the "readme" files located in the script directories.
4. Database scripts are encoded in Windows-1252 format. Before you start SQL*Plus, be sure to set your session to a value with this encoding. See the Oracle NLS_LANG FAQ for more information.
   Set the NLS_LANG variable and start SQL*Plus.
   The figure below shows an example of the commands for Linux and Oracle 11g.



SQL Command Prompt

5. Using a user account that has DBA privileges (for example, SYSTEM), enter the following at the prompt to connect to the Oracle instance:
   conn *<User>/<Password>@<alias for the Oracle instance contained in your local tnsnames.ora file>*
   See the following figure for an example of the command entry.



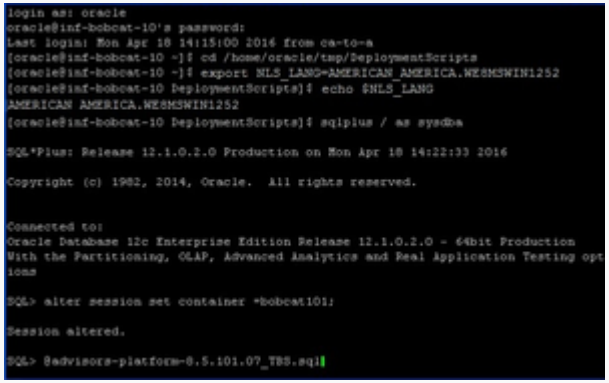
SQL Command Prompt 2

6. If the tablespaces are already present, you can go to Step 7. Otherwise, create tablespaces as described in this Step.
   You can either edit the tablespace script in order to adapt it to your environment, or you can create the tablespaces manually. Genesys recommends that you create at least a dedicated data tablespace and a dedicated temporary default tablespace for each Advisors user/schema.

   a. You, as a privileged user, or your DBA if you do not have privileged user access, must run the tablespace script contained in the installation

package (the script name ends with _TBS.sql). To run the tablespace script, enter @<*script name*> at the SQL*Plus prompt. For example:
`@advisors-platform-8.5.xxx_TBS.sql`, if you are creating a Platform schema; or
`@gc-metrics-8.5.xxx_TBS.sql`, if you are creating an AGA METRICS schema; or
`@mg-8.5.xxx_TBS.sql`, if you are creating a metric graphing schema.

See the following figure for an example of the command entry. The figure shows an example that uses Linux. The name of the script supplied in the installation package contains the specific release number of Advisors Platform that you will be installing.

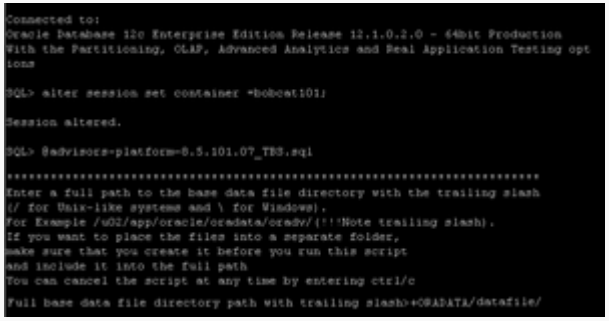


SQL Command Prompt 3

b. When prompted, enter the full path to your base data file directory including the trailing slash. This is the path on the server where ORACLE is installed; you are indicating where to put the files that will contain the tablespace data. The script will either:

   - Create the tablespaces if they do not yet exist, or
   - Skip the creation if the tablespaces are already present.

   Note that the script will preserve your SQL*Plus connection, which you can reuse later in this procedure.
   The following figure shows an example.



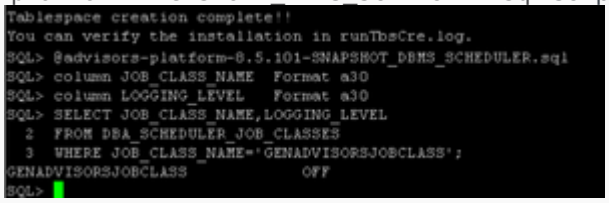
SQL Command Prompt 4

c. Verify the results of your script execution:

   i. Using a separate command prompt/terminal session, examine the `runTbsCre.log` file. You can find this log file in the same directory as your installation scripts.

   ii. Browse your data file location to ensure that the files were created. Alternatively, you can run the following query from any Oracle client connected as the system user:
   `SELECT * FROM dba_data_files`

7. Starting with Advisors Platform release 8.5.101.17, you must create a job class with the name `GenAdvisorsJobClass` before the creation of the Platform schema objects. Only a privileged user, either you or your DBA, can create the job class. The privileged user must run the `advisors-platform-<version>_DBMS_SCHEDULER.sql` script supplied in the installation package. Verify the results as shown in the following figure.



SQL Command Prompt 5

8. Create the user/schema and schema objects.

[+] Show steps to create the user/schema and schema objects separately

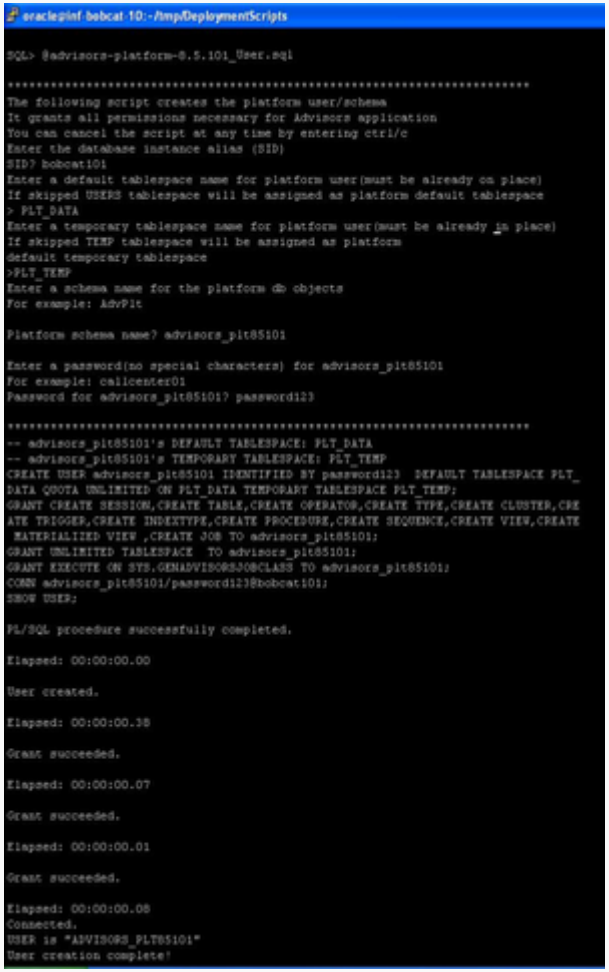
a. You, as a privileged user, or your DBA if you do not have privileged user access, must run the user creation script that is contained in the installation package (the script name ends with `_User.sql`). To run the user creation script, enter @<*script name*> at the prompt. For example:
`@advisors-platform-8.5.xxx_User.sql`, if you are creating a Platform schema; or
`@gc-metrics-8.5.xxx_User.sql`, if you are creating an AGA METRICS schema; or
`@mg-8.5.xxx_User.sql`, if you are creating a metric graphing schema.

The script prompts you to enter the user/schema name, the password, the default data and temporary tablespace names, and the SID. Genesys recommends that you create dedicated data and temporary default tablespaces for each Advisors user/schema. Make sure that the tablespaces are created and that you know the names before you start the user/schema creation procedure.

In the local client `tnsnames.ora` file, find the alias for the Oracle instance, and enter it at the SID prompt. For example, if your local client `tnsnames.ora` file contains the following entry for the target Oracle instance, you would enter `bobcat101` at the `SID>` prompt (note that the alias name is case-sensitive):

```
bobcat101 =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = inf-wolf.qalab.com)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = orcl.qalab.com)
    )
  )
```

See the following figure for an example of the command entry.


oracle@inf-bobcat-10:~/tmp/DeploymentScripts
SQL> @advisors-platform-8.5.101_User.sql
****
The following script creates the platform user/schema
It grants all permissions necessary for Advisors application
You can cancel the script at any time by entering ctrl/c
Enter the database instance alias (SID)
SID? bobcat101
Enter a default tablespace name for platform user(must be already on place)
If skipped USERS tablespace will be assigned as platform default tablespace
> PLT_DATA
Enter a temporary tablespace name for platform user(must be already in place)
If skipped TEMP tablespace will be assigned as platform
default temporary tablespace
>PLT_TEMP
Enter a schema name for the platform db objects
For example: AdvPlt
Platform schema name? advisors_plt85101
Enter a password(no special characters) for advisors_plt85101
For example: callcenter01
Password for advisors_plt85101? password123
****
-- advisors_plt85101's DEFAULT TABLESPACE: PLT_DATA
-- advisors_plt85101's TEMPORARY TABLESPACE: PLT_TEMP
CREATE USER advisors_plt85101 IDENTIFIED BY password123 DEFAULT TABLESPACE PLT_
DATA QUOTA UNLIMITED ON PLT_DATA TEMPORARY TABLESPACE PLT_TEMP;
GRANT CREATE SESSION,CREATE TABLE,CREATE OPERATOR,CREATE TYPE,CREATE CLUSTER,CRE
ATE TRIGGER,CREATE INDEXTYPE,CREATE PROCEDURE,CREATE SEQUENCE,CREATE VIEW,CREATE
MATERIALIZED VIEW ,CREATE JOB TO advisors_plt85101;
GRANT UNLIMITED TABLESPACE TO advisors_plt85101;
GRANT EXECUTE ON SYS.GENADVISORSJOBCLASS TO advisors_plt85101;
CONN advisors_plt85101/password123@bobcat101;
SHOW USER;
PL/SQL procedure successfully completed.
Elapsed: 00:00:00.00
User created.
Elapsed: 00:00:00.38
Grant succeeded.
Elapsed: 00:00:00.07
Grant succeeded.
Elapsed: 00:00:00.01
Grant succeeded.
Elapsed: 00:00:00.08
Connected.
USER is "ADVISORS_PLT85101"
User creation complete!

Creating the User/Schema and Schema Objects Separately: SQL Command Prompt 1

b. After the script completes and SQL*Plus exits, examine the `runUsrCre.log` file (located in the same directory as your installation scripts) to verify the results.

c. Connect as the owner of the Platform schema and execute the object creation script that is contained in the installation package (the script name ends with `_ObjectsPlus.sql`). To execute the object creation script, enter @<*script name*> at the prompt. For example:
`@advisors-platform-8.5.xxx_ ObjectsPlus.sql`, if you are creating a Platform schema; or
`@gc-metrics-8.5.xxx_ ObjectsPlus.sql`, if you are creating an AGA METRICS schema; or
`@mg-8.5.xxx_ ObjectsPlus.sql`, if you are creating a metric graphing schema.

The script prompts you to enter tablespace names for various groups of tables and indexes, as well as the SID. Genesys recommends that you create dedicated default tablespaces for each Advisors user/schema and that, at the very least, you put the tables into those dedicated default tablespaces. The tablespaces must be created and available after the user/schema is created.

In the local client `tnsnames.ora` file, find the alias for the Oracle instance, and enter it at the SID prompt. For example, if your local client `tnsnames.ora` file contains the following entry for the target Oracle instance, you would enter `bobcat101` at the SID> prompt (note that the alias name is case-sensitive):

```
bobcat101 =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = inf-wolf.qalab.com)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = orcl.qalab.com)
    )
  )
```

See the following figure; the figure shows empty entries at all prompts for tablespaces, which means that all the data and indexes will go to the default tablespace, which, in this case, is `PLT_DATA`. For better performance, you can separate indexes, group the tables by I/O patterns. Each prompt for a tablespace represents a table or index group.

Creating the User/Schema and Schema Objects Separately: SQL Command Prompt 2

If you prefer, you can use SQL Developer, instead of SQL*Plus, to create objects within the schema that you created earlier. You must connect as the owner of the corresponding schema, and then execute the object creation script (the script name ends with either `_ObjectsDefault.sql` or `_ObjectsCustom.sql`). The difference between the two scripts is:

- the `_ObjectsDefault.sql` script silently creates all objects and places them into your default tablespace.
- the `_ObjectsCustom.sql` script issues prompts, allowing you to place the table groups or indexes into different tablespaces. This script requires an explicit tablespace name on every prompt, even if you want to place the table group into your default tablespace.

d. After the script completes and SQL*Plus exits, examine the `runUsrCre.log` file (located in the same directory as your installation scripts) to

verify the results.

[+] Show steps to create the user/schema and schema objects in one step

If you have privileged user access, you can create the user/schema and the objects in one step. You must use SQL*Plus – and only SQL*Plus – to execute the script.

a. You, as a privileged user, or your DBA if you do not have privileged user access, must run the script contained in the installation package (the script name ends with `_Schema.sql`). To run the script, enter @<*script name*> at the prompt. For example:
`@advisors-platform-8.5.xxx_Schema.sql`, if you are creating a Platform schema; or
`@gc-metrics-8.5.xxx_Schema.sql`, if you are creating an AGA METRICS schema; or
`@mg-8.5.xxx_Schema.sql`, if you are creating a metric graphing schema.

The script prompts you to enter the user/schema name, the password, the default data and temporary tablespace names, and the SID. Genesys recommends that you create dedicated data and temporary default tablespaces for each Advisors user/schema. Make sure that the tablespaces are created and that you know the names before you start the schema creation procedure.

In the local client `tnsnames.ora` file, find the alias for the Oracle instance, and enter it at the SID prompt. For example, if your local client `tnsnames.ora` file contains the following entry for the target Oracle instance, you would enter `bobcat101` at the `SID>` prompt (note that the alias name is case-sensitive):

```
bobcat101 =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = inf-wolf.qalab.com)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = orcl.qalab.com)
    )
  )
```

After the user is created, the script prompts you to enter tablespace names for various groups of tables and indexes. Genesys recommends that, at the very least, you put the tables into the dedicated default tablespaces that you created for each Advisors user/schema. The tablespaces must be created and available before you execute the `_Schema.sql` script.

See the following figure; the figure shows empty entries at all prompts for tablespaces, which means that all the data and indexes will go to the default tablespace, which, in this case, is `PLT_DATA`. For better performance, you can separate indexes, group the tables by I/O patterns. Each prompt for a tablespace represents a table or index group.

Creating the User/Schema and Schema Objects in One Step: SQL Command Prompts

b. After the script completes and SQL*Plus exits, examine the `runUsrCre.log` and `runObjCre.log` files (located in the same directory as your installation scripts) to verify the results.

This concludes a general Oracle schema/user setup where each created user is the owner of the corresponding schema: Platform, AGA metrics, or metric graphing. You can now specify the user in the relevant Advisors installation wizard screens related to database connectivity.

Database access for runtime users with least privileges

Starting with Advisors release 8.5.202, you have the option to configure database access through runtime users with least privileges, rather than through users who are "schema owners" . The procedure to create runtime users is implemented on top of the general Oracle schema/user setup. See Least Privileges: How to Configure Advisors Database Accounts with Minimal Privileges for more information.