

Configuring an Alternative Database for SAS[®] Web Infrastructure Platform Services

By default, SAS Web Infrastructure Platform Services is configured to use SAS Framework Data Server. If you prefer, you can choose to use a third-party database. Here are the versions of the alternative databases for SAS Web Infrastructure Platform Services that were used for testing in SAS 9.3:

- Oracle 11g FP2
- PostgreSQL 8.7 and 9.0
- MySQL 5.0 and 5.5.11
- IBM DB2 Version 9.7 FP2
- Microsoft SQL Server 2008

If you prefer to use a third-party vendor database instead of SAS Framework Data Server, then the database must be configured with the tables needed by SAS Web Infrastructure Platform Services. The JDBC connection information for the database must be provided to the SAS[®] Deployment Wizard. The following information must be provided:

- Host name of the database server
- Port number of the database server or listener
- Database name or SID
- User ID and password
- Directory location of JDBC drivers for the database

The SAS Deployment Wizard can create the tables that are needed by SAS Web Infrastructure Platform Services. Select the **Automatically create tables and load data** check box to use this feature. If you prefer to create the tables yourself, then clear the check box and submit the SQL statements after the wizard finishes. The SQL statements are located in the *SAS-installation-directory\SASWebInfrastructurePlatform\9.3\Config\Deployment\Data* directory.

The rest of this document describes database vendor-specific information such as sample database and user creation commands, and JDBC URLs.

MySQL

Before running the SAS Deployment Wizard, you must have a user ID with permission to access and create tables. The following steps show sample steps for creating a user for MySQL:

1. Log on to MySQL as the root user.
2. Create a user account:

```
create user sharedservices@'%' identified by 'password';
```

3. Grant access to the database for the new account:

```
grant all on sharedservices.* to sharedservices@'%';
```

Note: Do not create the database. The SAS Deployment Wizard issues the CREATE DATABASE statement when it runs.

You must also install the JDBC drivers from MySQL Connector/J on the machine that is hosting the Web application server. The SAS Deployment Wizard configures the Web application server with a JDBC data source that uses a URL that is similar to the following example:

```
jdbc:mysql://host:3306/SharedServices
```

If you want the SAS Deployment Wizard to automatically configure and load tables, then confirm that the user account can perform the query `select 1` successfully before you start the SAS Deployment Wizard.

Oracle Database

Before running the SAS Deployment Wizard, you must have a database instance and user ID with permission to access and create tables. The following steps show sample steps for creating a database instance and user.

1. Create an Oracle instance. This can be done with the Database Configuration Assistant for Oracle database. When the wizard prompts you for the SID, enter `sharedservices`.

Note: SAS strongly recommends that you use UTF-8 as the character encoding for the database.

2. Connect to the Oracle instance:

```
sqlplus sys@sharedservices as sysdba
```

3. Create a user account:

```
create user sharedservices identified by password default tablespace  
user temporary tablespace temp;
```

4. Grant access to the database for the new account:

```
grant connect,resource to sharedservices;
```

The Oracle Database JDBC drivers are located in `ORACLE_HOME/product/version/install/jdbc/lib`. The SAS Deployment Wizard configures your Web application server with a JDBC data source that uses a URL that is similar to the following example:

```
jdbc:oracle:thin:@host:1521:SharedServices
```

If you want the SAS Deployment Wizard to automatically configure and load tables, then confirm that the user account can perform the query `select 1 from dual` successfully before you start the SAS Deployment Wizard.

IBM DB2

Before running the SAS Deployment Wizard, you must have a database instance and user ID with permission to access and create tables. When the database is created, the table structure requires specific bufferpool sizes. The following steps show sample steps for creating a database instance.

1. Access the DB2 Command Line Processor.
2. Create a database:

```
create database shareddb on /db2 alias shared pagesize 32 k
```

3. Connect to the database:

```
connect to shared
```

4. Create a bufferpool and tablespace:

```
create bufferpool shared_bp size 8192 pagesize 32 k

create temporary tablespace sys_temp_shared pagesize 32 k managed by
database using (file '/db2/shared/ts/shareddb/sys_temp.ts' 500m)
autoresize yes bufferpool shared_bp

create large tablespace shared1 pagesize 32 k managed by database using
(file '/db2/shared/ts/shareddb/shared1.ts' 500m) autoresize yes bufferpool
shared_bp

create user temporary tablespace user_temp_shared pagesize 32 k managed
by database using (file '/db2/shared/ts/shareddb/user_temp.ts' 500m)
autoresize yes bufferpool shared_bp
```

5. Grant database administration authority:

```
grant dbadm on database to user sharedservices
```

Note: IBM DB2 uses operating-system user IDs by default. Make sure that you identify a user ID that has an account on the operating system.

The IBM DB2 JDBC drivers are located in *DB2_HOME*/java. The SAS Deployment Wizard configures your Web application server with a JDBC data source that uses a URL that is similar to the following example:

```
jdbc:db2://host:50002/shared
```

If you want the SAS Deployment Wizard to automatically configure and load tables, then confirm that the user account can perform the query `select 1 from sysibm.sysdummy1` successfully before you start the SAS Deployment Wizard.

PostgreSQL

SAS performed testing with version 8.7 and expects any version newer than 8.7 to operate correctly. SAS requires that you use the 9.0 or newer version of the PostgreSQL JDBC driver. Before running the SAS Deployment Wizard, you must have a database instance and user ID with permission to access and create tables. The following steps show sample steps for creating a database instance and user:

1. Log on to the system with the PostgreSQL user ID, and create a role:

```
create role sharedservices LOGIN PASSWORD 'password';
```

2. Create the database:

```
createdb SharedServices owner sharedservices encoding 'UTF8'
```

3. Confirm that the **listen_addresses** parameter in the `postgresql.conf` file permits connections from hosts other than localhost, and that the `pg_hba.conf` file is configured to permit access from the machine that is hosting the Web application server. Your needs may differ from those described here. For more information about those configuration files and additional configuration options, see the PostgreSQL documentation.

You must install the JDBC driver for PostgreSQL on the machine that is hosting the Web application server. The SAS Deployment Wizard configures the Web application server with a JDBC data source that uses a URL that is similar to the following example:

```
jdbc:postgresql://host:5432/SharedServices
```

If you want the SAS Deployment Wizard to automatically configure and load tables, then confirm that the user account can perform the query `select 1` successfully before you start the SAS Deployment Wizard.

Microsoft SQL Server

SAS performed testing with SQL Server 2008. The following steps show sample steps for creating a database instance and user.

1. Use the SQL Server Installation Center to create a database instance. Configure it to use mixed mode authentication.
2. Use `sqlcmd` to log on to the instance with the `sa` account and create the `sharedservices` user:

```
use master
go
create login sharedservices with password = 'password'
go
create user sharedservices for login sharedservices
go
grant all to sharedservices
go
```

Note: Do not create the database. The SAS Deployment Wizard issues the `CREATE DATABASE` statement when it runs.

You must also install the JDBC drivers from Microsoft on the machine that is hosting the Web application server. The SAS Deployment Wizard configures the Web application server with a JDBC data source that uses a URL that is similar to the following example:

```
jdbc:sqlserver://host:1433;databaseName=SharedServices
```

If you want the SAS Deployment Wizard to automatically configure and load tables, then confirm that the user account can use the master database and issue a `CREATE DATABASE` statement, as well as perform the query `select 1`. This command must run successfully before you start the SAS Deployment Wizard.

SAS provides the option to either automatically create and upgrade the database or for your database administrator (DBA) to perform that function manually.

1. If your database administrator (DBA) is going to manage the schema creation and upgrade process manually, the `'sharedservices'` user needs `SELECT`, `INSERT`, `DELETE`, and `UPDATE` permissions for the `'SharedServices'` database. For example:

```
grant SELECT on SharedServices to sharedservices
grant INSERT on SharedServices to sharedservices
grant DELETE on SharedServices to sharedservices
grant UPDATE on SharedServices to sharedservices
```

2. If you want the SAS software to automatically create/update the database and limit the `'sharedservices'` user access after initial creation, the user requires `SELECT`, `INSERT`, `DELETE`, `UPDATE`, `ALTER TABLE`, `CREATE TABLE`, `CREATE INDEX`, and `DROP`. For example:

```
grant SELECT on SharedServices to sharedservices
grant INSERT on SharedServices to sharedservices
grant DELETE on SharedServices to sharedservices
grant UPDATE on SharedServices to sharedservices
```

(code continued)

```
grant ALTER TABLE on SharedServices to sharedservices
grant CREATE TABLE on SharedServices to sharedservices
grant CREATE INDEX on SharedServices to sharedservices
grant DROP on SharedServices to sharedservices
```

Because the SAS software is acting as the administrator in this case, it might require other permissions in the future. It might therefore be more straightforward to grant the user all permissions on that particular database.

```
grant all on SharedServices to sharedservices
```

Note that these grants to the SharedServices database need to occur after the Software Deployment Wizard process has created the database and tables in this method.