

SILVERRUN

INSTALLATION GUIDE

S I L V E R R U N
E N T E R P R I S E
S E R I E S

Repository Installation and
Setup Under ORACLE™

Grandite

AUGUST 2007

SILVERRUN Enterprise Series

Version 2.8

Installation Guide

Grandite

Copyright © 2004 - 2007 Grandite
All rights reserved.

Installation Guide for the ORACLE Repository, version 2.8

No part of this document may be reproduced or transmitted in any form or by any means, electronic or mechanical, without the express written consent of Grandite.

Information in this document is subject to change without notice.

Any comments on this product or its documentation may be addressed to:

Grandite
1220 Lebourgneuf blvd.
Suite 120
Quebec (Quebec)
Canada G2K 2G4

You can also send your comments by way of the Internet by accessing our Web page or by sending an e-mail to:

www.grandite.com

support@grandite.com

All trademarks are the property of their respective owners.

Table of Contents

Preface	7
First Installation	9
Server Configuration	9
Database Software	9
Communication Software	9
Hardware	10
Disk Space	10
Before You Start	11
Step 1: Creation of Tablespaces and SR_ADMIN User	14
Step 2: Creation of the SILVERRUN Metamodel	15
Step 3: Creation of SILVERRUN Users	16
Step 4: Creation of Synonyms	17
Upgrading the Repository to Version 2.8	18
Before You Start	19
Step 1: Validation	20
Step 2: Upgrade of the ORACLE Catalog	21
Step 3: Creation of Synonyms	22
Step 4: Upgrade of the Application	23
APPENDIX	25
First Installation Scripts	26
Upgrade Scripts	27
Utility Scripts	28

The *SILVERRUN Enterprise Series Installation Guide* describes the procedures to install the SILVERRUN Enterprise Series Repository in an ORACLE environment. It also shows how to upgrade an existing Repository to version 2.8. The installation guide applies both to SILVERRUN-RDM Enterprise series and SILVERRUN-BPM Enterprise series.

Installation of SILVERRUN Enterprise series is twofold: first, the SILVERRUN Repository is created on the ORACLE server, then the application (RDM or BPM) is installed on each of the workstations. The installation of the Repository on the server is conducted by the SILVERRUN Administrator, assisted by the DBA (database administrator), while the installation of the application on workstations can be done by users.

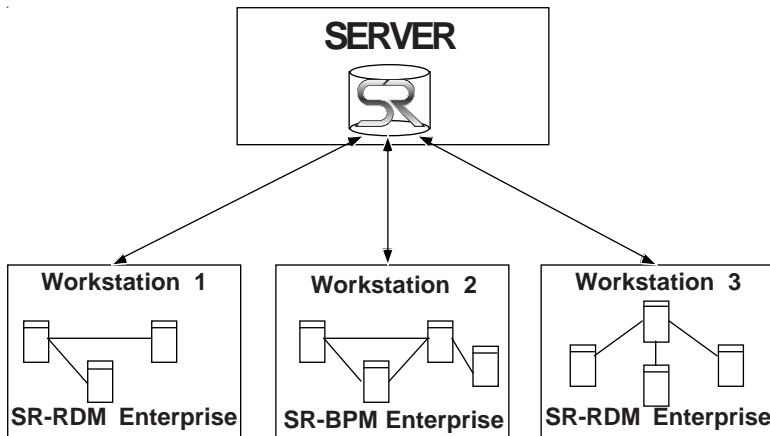
For information on other aspects of SILVERRUN Enterprise series, see the other guides provided with your application.

SILVERRUN Enterprise series has modeling features for development workgroups. Although users have the option of working in stand-alone mode, the application offers a dynamic way of designing, building and managing an enterprise information architecture:

- users work online with the SILVERRUN Repository,
- users have concurrent access to the SILVERRUN Repository.

The SILVERRUN Repository resides on a computer called "server". The server provides shared services to all workstations – the "clients" – connected on the network. Client workstations run the SILVERRUN Enterprise series application and share the SILVERRUN Repository located on the server.

The SILVERRUN Administrator is the person responsible for creating and maintaining the SILVERRUN Repository, as well as controlling access to it by the different users.



Different workstations running SILVERRUN Enterprise series access the SILVERRUN Repository

By using ORACLE as RDBMS support for the Repository, SILVERRUN Enterprise series allows you to benefit from multi-platform support, various networking possibilities and an extensive range of products and services.

When working with SILVERRUN Enterprise series, the first step is to have ORACLE properly installed. Once the RDBMS is installed, you are ready to install the Repository on the server. For information on the installation of ORACLE, please refer to the ORACLE documentation.

Server Configuration

The SILVERRUN Repository under ORACLE can reside:

- on an RDBMS server (dedicated server): generally, it is one of the best ways to get the most performance from your system, although performance depends on the number of tasks executed by the server and the number of applications running under ORACLE,
- on a network server: this configuration is suitable if the server has to execute few tasks – printing, communications, etc.

The SILVERRUN Repository must be installed on version 7.0 or later of ORACLE. Version 1.0 or later of SQL*Net™ must be correctly installed.

Database Software

SILVERRUN Enterprise series is compatible with any communication software supported by SQL*Net – TCP/IP, NPX, etc. Please refer to the ORACLE documentation for the list of communication protocols supported by SQL*Net.

Communication Software

Hard- ware

The SILVERRUN Enterprise Series Repository can be installed on any platform supporting ORACLE. Therefore, you can install it on micro, mini or mainframe servers. Please refer to the ORACLE documentation for a list of the supported hardware.

Disk Space

The disk space required to install the Repository depends on the size of your models.

Tablespace for Models

A model on the Repository requires approximately 3 times more disk space than if it was in binary format as a disk file. For example, a 2MB SILVERRUN-RDM file would use about 6MB when transferred on the Repository.

Rollback Segment

The size of the rollback segment depends on the number of occurrences in the transaction. If a rollback segment is not large enough, ORACLE will send an error message.

As an indication, with the **Copy to Repository** command, a 1MB binary format file will use a rollback segment of about 4MB when it is transferred to the Repository.

Before You Start

- ORACLE must be installed on the Repository server and the database must be operational.
- SQL*Plus™ and SQL*Net must be installed and operational on the workstation on which the installation scripts will be run.
- Determine on which ORACLE instance you want to work; can you use an existing instance or do you want to create a new instance exclusively for the SILVERRUN Repository.
- Decide whether you want the *simplified* or the *optimized* installation. The *simplified* installation creates a single tablespace for both tables and indexes required to store SILVERRUN information. The *optimized* installation creates two tablespaces, one for tables and one for indexes; the two tablespaces can be created on two separate disks.

Simplified Installation

TABLESPACE:
SILVERRUN_REPOSITORY_DATA

all SILVERRUN
indexes and tables

Optimized Installation

TABLESPACE:
SILVERRUN_REPOSITORY_INDEXES

only SILVERRUN
indexes

TABLESPACE:
SILVERRUN_REPOSITORY_DATA

only SILVERRUN
tables

- Make sure you have a rollback segment available for SILVERRUN. The length of the rollback segment should be revised depending on the operations performed by the application (see **Server Configuration**).
- Determine the path on the server for the tablespaces needed for tables and indexes.
- Decide how much disk space to allocate to tablespaces for tables and indexes, according to the size of your models (see **Server Configuration**).

- Prepare a list of users who will be granted access to the SILVERRUN Repository. Database user names will need to be created for new users. User privileges will need to be granted for all users, according to the following roles:

ROLE	PRIVILEGES
SR_USERS	user is granted modification privileges, i.e. adding, modifying and deleting occurrences from all data models stored in the Repository
SR_GUESTS	user cannot make any modifications to models, only browsing is allowed

- The first two steps of the installation require the cooperation of someone having DBA privileges: make sure that person is available when needed.
- The SILVERRUN Repository installation procedure creates an SR_ADMIN user name (SILVERRUN Administrator) for the database. Designate someone in your organization to be responsible for the SILVERRUN Repository.
- The installation scripts are on the Repository installation diskette, or the Repository installation section of the CD-ROM. A complete list of scripts for installation and upgrade can be found in the appendix.
- In the following pages you will find the complete installation steps.

When a script is run, a file is generated with the same name as the script but with the extension .LST. This file is a report of the script execution indicating if operations have failed or succeeded. The .LST file may be opened with a text editor.

Installing SILVERRUN Enterprise series 2.8

Step 1 Creation of tablespaces and SR_ADMIN user

Step 2 Creation of the SILVERRUN metamodel

Step 3 Creation of SILVERRUN users

Step 4 Creation of synonyms

Step 1: Creation of Tablespaces and SR_ADMIN User

SCRIPT NAME

Select the appropriate script:

INIT_S.SQL

if you want the simplified installation.

This script creates the tablespace SILVERRUN_REPOSITORY_DATA for tables and indexes, two roles (SR_USERS and SR_GUESTS) and the SR_ADMIN user id for the SILVERRUN Administrator.

INIT_O.SQL

if you want the optimized installation.

This script creates the tablespace SILVERRUN_REPOSITORY_DATA for tables, the tablespace SILVERRUN_REPOSITORY_INDEXES for indexes, two roles (SR_USERS and SR_GUESTS) and the SR_ADMIN user id for the SILVERRUN Administrator.

AUTHORIZED USER

With DBA privileges.

PROCEDURE

If you need to rename the default tablespace name(s), change them in the appropriate script. If you use tablespace name(s) different from the default, tablespace name(s) will have to be changed in the other installation scripts where they are mentioned.

In the appropriate script, indicate the path where tablespace(s) should be located.

Before you run the script, adjust the disk space allocated to a tablespace if necessary.

Run INIT_S.SQL or INIT_O.SQL.

Step 2: Creation of the SILVERRUN Metamodel

SCRIPT NAME	TABLES.SQL	Creates SILVERRUN tables.
	Select the appropriate script:	
	INDEX_S.SQL	Creates indexes for the simplified installation.
	INDEX_O.SQL	Creates indexes for the optimized installation.
	Select the appropriate script:	
	GRTUSER.SQL	Grants privileges to the SR_USERS role.
	GRTGUEST.SQL	Grants privileges to the SR_GUESTS role.
AUTHORIZED USER	SR_ADMIN.	
PROCEDURE	Run TABLES.SQL.	
	At the end, a validation message displays. For example:	
	<pre>Number of tables created inside the SILVERRUN Repository COUNT (*) - - - - - 150</pre>	
	Depending on the installation, run INDEX_S.SQL or INDEX_O.SQL.	
	At the end, a validation message displays. For example:	
	<pre>Number of indexes created inside the SILVERRUN Repository COUNT (*) - - - - - 513</pre>	
	Run GRTUSER.SQL.	
	Run GRTGUEST.SQL.	

Step 3: Creation of SILVERRUN Users

SCRIPT NAME	NEWUSER.SQL	Creates user and grants SR_USERS role.
	NEWGUEST.SQL	Creates user and grants SR_GUESTS role.

AUTHORIZED USER With DBA privileges.

PROCEDURE New users on ORACLE:

- To create and grant a new SILVERRUN user with modification privileges (SR_USERS role), run NEWUSER.SQL followed by the name and password of the new user.

example: NEWUSER user_name password

- To create and grant a new SILVERRUN user with browsing privileges (SR_GUESTS role), run NEWGUEST.SQL followed by the name and password of the new user.

example: NEWGUEST user_name password

Existing users on ORACLE:

- To grant the SR_USERS role to an existing user, use the GRANT command with the name of the user.

example: GRANT SR_USERS TO user_name

- To grant the SR_GUESTS role to an existing user, use the GRANT command with the name of the user.

example: GRANT SR_GUESTS TO user_name

SR_USERS	SR_GUESTS
user is granted modification privileges, i.e. adding, modifying and deleting occurrences from all data models stored in the Repository	only browsing is allowed, no modifications to models may be made

Step 4: Creation of Synonyms

SCRIPT NAME	SYNONYMS.SQL	Creates tables synonyms for each SILVERRUN user.
--------------------	---------------------	--

AUTHORIZED USER	All users.
------------------------	------------

PROCEDURE	Under each SILVERRUN user name, run SYNONYMS.SQL.
------------------	---

A validation message should display. For example:

```
Number of synonyms created for the user
COUNT ( * )
- - - 150 -
```

Upgrading the Repository to Version 2.8

You may ignore this upgrade section if your Repository is currently installed for SILVERRUN Enterprise series 2.8.0.

Upgrading to SILVERRUN Enterprise series 2.8 is done in 4 steps.

Step 1 Validation

Step 2 Upgrade of the ORACLE catalog: the Repository structure is changed to accomodate new features of SILVERRUN-RDM or BPM Enterprise series (new tables, new indexes, etc.)

Step 3 Creation of synonyms

Step 4 Upgrade of the application

Before You Start

- SR_ADMIN privileges are required to upgrade the SILVERRUN Repository.
- Backup the old Repository and keep the backup in a safe place until users have worked with the new version on a regular basis.
- During the upgrade procedure, no user may be connected or attempt to connect to the Repository. Users attempting to connect to the Repository will receive a warning message saying that the version of the Repository is not compatible with the version of their SILVERRUN application.
- Find out if the installation of the previous version was simplified (one tablespace for tables and indexes) or optimized (one tablespace for tables and one tablespace for indexes).
- The upgrade scripts are on the Repository installation diskette, or the Repository installation section of the CD-ROM. A complete list of scripts for installation and upgrade can be found in the appendix.
- Make sure you use the scripts in the RDM_28 sub-directory (for RDM and BPM Enterprise series).

Step 1: Validation

SCRIPT NAME	INIUPGRD.SQL	Outputs: 1- the version of the current Repository on the server or the last step executed in the upgrade procedure, 2- user names currently connected to the Repository, if any.
AUTHORIZED USER	SR_ADMIN.	
PROCEDURE	Run INIUPGRD.SQL. A message displays at the end of the operation: if the version identified is SILVERRUN-RDM Enterprise series 2.7.6 or SILVERRUN-BPM Enterprise series 2.7.6, you may continue the upgrade procedure. If the version is 2.8.0, the new Repository version is already installed on the server.	

Step 2: Upgrade of the ORACLE Catalog

SCRIPT NAME

Select the appropriate script:

UPGRD_S.SQL

if existing installation is simplified, i.e. one tablespace for data and indexes.

This script upgrades the Repository and creates tables and indexes in the **SILVERRUN_REPOSITORY_DATA** tablespace.

UPGRD_O.SQL

if existing installation is optimized, i.e. two tablespaces, one for data and one tablespace for indexes.

This script upgrades the Repository and creates tables in the tablespace **SILVERRUN_REPOSITORY DATA** and indexes in the tablespace **SILVERRUN_REPOSITORY_INDEXES**.

AUTHORIZED USER

SR_ADMIN.

PROCEDURE

If you need to rename the default tablespace name(s), change them in the appropriate script.

According to the type of installation you have, run **UPGRD_S.SQL** or **UPGRD_O.SQL**.

Step 3: Creation of Synonyms

Note: This step is not necessary for SILVERRUN 2.8.0, since there are no new tables.

SCRIPT NAME	SYNUPGRD.SQL	Creates new tables synonyms for each SILVERRUN user
AUTHORIZED USER	All users.	
PROCEDURE	Each user must run SYNUPGRD.SQL under his user name in order to have access to the new tables.	

Step 4: Upgrade of the Application

APPLICATION	SILVERRUN Enterprise Series 2.8
AUTHORIZED USER	All users.
PROCEDURE	Install the SILVERRUN Enterprise series 2.8 application (RDM or BPM) on every workstation.

APPENDIX

SILVERRUN Enterprise Series 2.8 Installation, Upgrade and Utility Scripts

First Installation Scripts

Directory / Folder	Script name	Description
INSTALL	INIT_S.SQL	For the simplified installation, the script creates: <ul style="list-style-type: none"> • the tablespace for tables and indexes (SILVERRUN_REPOSITORY_DATA), • two roles (SR_USERS and SR_GUESTS) • the SR_ADMIN user id for the SILVERRUN Administrator.
	INIT_O.SQL	For the optimized installation, the script creates: <ul style="list-style-type: none"> • the tablespace for tables (SILVERRUN_REPOSITORY_DATA), • the tablespace for indexes (SILVERRUN_REPOSITORY_INDEXES), • two roles (SR_USERS and SR_GUESTS), • the SR_ADMIN user id for the SILVERRUN Administrator.
	NEWUSER.SQL	Creates user and grants SR_USERS role.
	NEWGUEST.SQL	Creates user and grants SR_GUESTS role.
	GRTUSER.SQL	Grants privileges to the SR_USERS role.
	GRTGUEST.SQL	Grants privileges to the SR_GUESTS role.
	TABLES.SQL	Creates SILVERRUN tables.
	INDEX_S.SQL	Creates indexes for the simplified installation.
	INDEX_O.SQL	Creates indexes for the optimized installation.
	SYNONYMS.SQL	Creates synonyms of tables for each SILVERRUN user.

Upgrade Scripts

Directory / Folder	Script name	Description
UPGRADE\RDM_28	INIUPGRD.SQL	Outputs: 1 the version of the current Repository on the server or the last step executed in the upgrade procedure, 2 user names currently connected to the Repository, if any.
	UPGRD_S.SQL	If the existing installation is simplified, i.e. one tablespace for data and indexes, the script upgrades the Repository and creates tables and indexes in the SILVERRUN_REPOSITORY_DATA tablespace.
	UPGRD_O.SQL	If the existing installation is optimized, i.e. two tablespaces, one for data and one tablespace for indexes, the script upgrades the Repository and creates tables in the SILVERRUN_REPOSITORY_DATA tablespace and indexes in the SILVERRUN_REPOSITORY_INDEXES tablespace.
	SYNUPGRD.SQL	Creates new tables synonyms for each SILVERRUN user.

Utility Scripts

Directory / Folder	Script name	Description
UTILITY	DRPINDEX.SQL	Drops all SILVERRUN indexes except those on zModule.
	DRPSYNON.SQL	Drops all synonyms for a user.
	DRPTABLE.SQL	Drops all SILVERRUN tables except those on zModule.
	ORASTATS.SQL	Obtains statistics on the performance of ORACLE and suggests enhancements.
	REPINFO.SQL	Extracts data to ensure that the installation of the Repository has been done successfully.
	STORGIND.SQL	Alters the storage of SILVERRUN indexes.
	STORGTAB.SQL	Alters the storage of SILVERRUN tables.
	TRUNCATE.SQL	Deletes all occurrences of the SILVERRUN Repository except those of zModule.
	USERINFO.SQL	Extracts data to ensure that the creation of a user has been done correctly.

