User Manual
System Landscape Directory

SAP NetWeaver 7.1 SPS 5

Copyright

Icons in Body Text

Icon	Meaning
	Caution
	Example
	Note
	Recommendation
	Syntax

Additional icons are used in SAP Library documentation to help you identify different types of information at a glance. For more information, see *Help on Help → General Information Classes and Information Classes for Business Information Warehouse* on the first page of any version of *SAP Library*.

Typographic Conventions

Type Style	Description
Example text	Words or characters quoted from the screen. These include field names, screen titles, pushbuttons labels, menu names, menu paths, and menu options.
	Cross-references to other documentation.
Example text	Emphasized words or phrases in body text, graphic titles, and table titles.
EXAMPLE TEXT	Technical names of system objects. These include report names, program names, transaction codes, table names, and key concepts of a programming language when they are surrounded by body text, for example, SELECT and INCLUDE.
Example text	Output on the screen. This includes file and directory names and their paths, messages, names of variables and parameters, source text, and names of installation, upgrade and database tools.
Example text	Exact user entry. These are words or characters that you enter in the system exactly as they appear in the documentation.
<Example text>	Variable user entry. Angle brackets indicate that you replace these words and characters with appropriate entries to make entries in the system.
EXAMPLE TEXT	Keys on the keyboard, for example, F2 or ENTER.

⬇ Configuring, Working with and Administering System Landscape Directory

Use

A system landscape consists of a number of hardware and software components that depend on each other with regard to installation, software updates, upgrades, and demands on interfaces.

The System Landscape Directory of SAP NetWeaver (SLD) serves as a central information repository for your system landscape. More information:

> 💡
>
> Bear in mind that the abbreviation SLD is not intended to define a product, since the System Landscape Directory is part of SAP NetWeaver. This abbreviation is only intended to improve readability.

> ⚠️
>
> If you are reading this documentation offline, bear in mind that this might not be its latest version and that it might contain broken links. You can find the latest version on SAP Help Portal at `http://help.sap.com/nwpi71` → *SAP NetWeaver Process Integration Library English* → *SAP NetWeaver Library* → *SAP NetWeaver Process Integration Library* → *Administrator's Guide* → *Technical Operations for SAP NetWeaver* → *General Administration Tasks* → *Software Life-Cycle Management* → *Configuring, Working with, and Administering System Landscape Directory.*

Procedure

Configuring Systems to Connect to the SLD

Systems that are installed in your system landscape can periodically send data to the SLD. To do so, you have to configure them.

More information:

Working with the SLD

With SLD you can view and manage technical systems, landscapes, business systems, products, and software components.

More information:

Administrating the SLD

The initially configured SLD is ready to provide its services to the systems that are connected to it. Nevertheless, you:

- Might want to change the initial SLD configuration

- Should regularly perform tasks that are vital for the consistent state of your SLD and the information it contains

More information:

System Landscape Directory

Purpose

The System Landscape Directory of SAP NetWeaver (SLD) serves as a central information repository for your system landscape. A system landscape consists of a number of hardware and software components that depend on each other with regard to installation, software updates, and demands on interfaces.

Integration

The figure below shows the flow of data between the SLD and other components.

* **Common Information Model – Object-oriented industry standard of the DMTF**
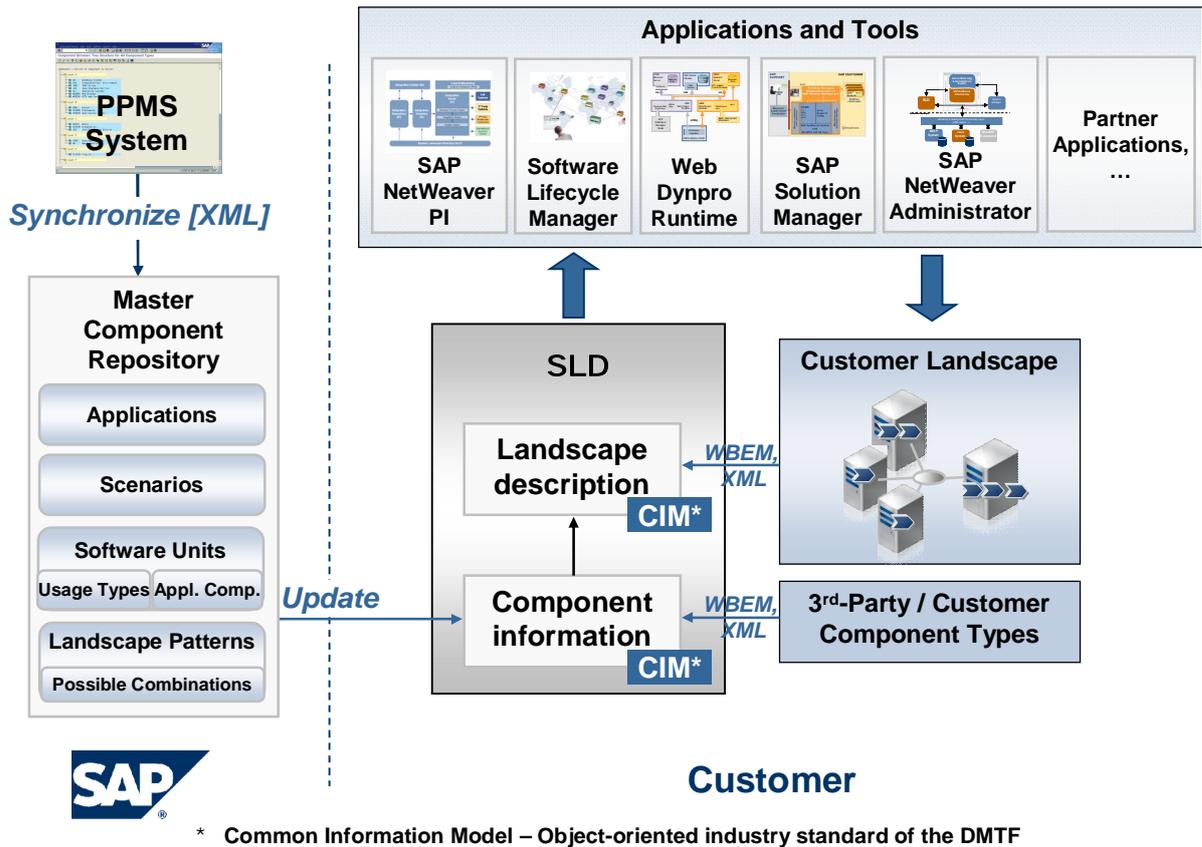
Information in the SLD is used by various SAP tools, such as SAP Solution Manager and SAP NetWeaver Administrator.

Features

- SLD stores information about all installable and installed components of a system landscape. SLD is based on the Common Information Model (CIM) standard. This standard is a general, extensible schema for describing the elements in a system landscape. SAP has enhanced the CIM standard with classes for SAP-specific content.

- SAP provides both a Web-based user interface (UI) for interactive access and client APIs for programmatic access in ABAP and Java.

- SLD provides information about installable SAP software, dependencies and recommended scenarios and regularly publishes updates on SAP Service Marketplace. You can browse this information in the software catalog views. You can also manually add information about third-party components to the SLD.

- SAP software components that are installed on technical systems are registered automatically and on a regular basis in the SLD. Thus the SLD always contains up-to-date information about the installed system landscape. You can view this information in the technical systems views, and can aggregate and group it in the landscapes views.

- In addition, you can use the SLD to reserve component names in the SAP NetWeaver Developer Infrastructure (NWDI). It contains the master list of all reserved names used in development, such as development component names, Java package names, and database object names.

- SAP NetWeaver Process Integration (PI) uses the SLD to store information about business systems, which are logically connected to technical systems.

More Information

Common Information Model

The Common Information Model (CIM) is a standard of the Distributed Management Task Force (DMTF) and is based on the object-oriented modeling approach. This standard provides an implementation-neutral schema to describe management information within a computing environment. For more information about CIM and DMTF, see `www.dmtf.org`.

CIM is designed to model hardware and software elements. It provides a hierarchical set of core classes, which can be extended with application-specific classes.

The SLD is currently based on CIM model 2.9.

Each CIM class has one or more key properties, and a number of additional properties. Each class has an arbitrary number of instances, which contain the actual data. An instance can be uniquely identified by its set of key values.

Classes can be grouped in schemas. The SLD model provides three schemas: component data for installable software, landscape data for installed software and hardware, and name reservation data used in software development.

A special feature of CIM is the association between the classes. Associations are also classes that have two reference keys, each of which identifies exactly one CIM instance. An association therefore connects two instances, as shown in the figure below:
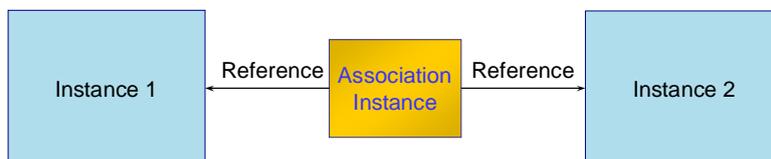
Architecture Overview of the SLD Data Suppliers

Besides component information, the SLD stores information about the current system landscape. A data collection process in each technical system in the landscape can provide this information to the SLD. This process is managed by the data supplier administration. The data suppliers collect data not only about SAP systems (SAP products, database parameters, hosts, clients, and so on), but also about any third-party products that are installed.

Data Collection Process

As shown in the figure below, the data suppliers deliver information about the installed elements in a particular system, update this data periodically, and send it to the SLD server by means of CIM over HTTP.
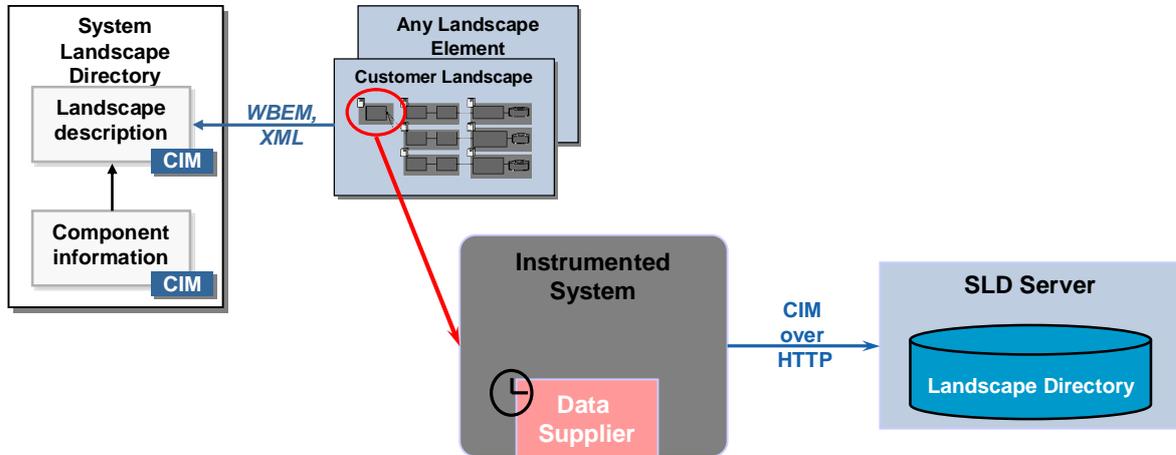
Currently, there are data suppliers available for all ABAP-based systems as of SAP Basis release 4.0B and for all Java EE based systems as of SAP NetWeaver 6.40.

Data Sending Process

The figure below shows how the SLD data suppliers send data to the SLD server:
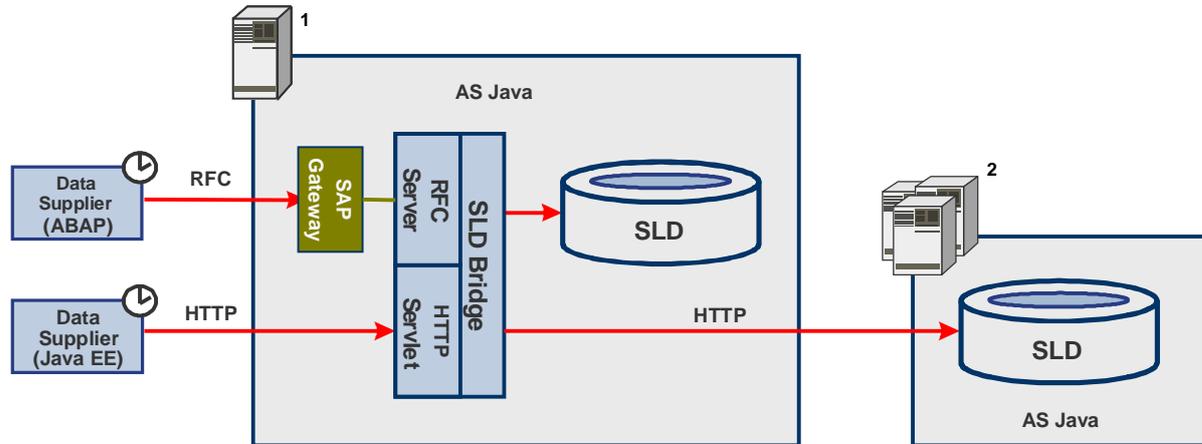
- The ABAP-based data supplier uses an RFC connection to send data to the SLD bridge by means of a gateway and an RFC server.

  As of SAP NetWeaver 7.1, usage type AS Java that comprises SLD also contains an SAP Gateway.

- The Java EE based data supplier uses HTTP to send data to the SLD bridge by means of an HTTP servlet.

The SLD bridge receives the reported data and converts it into the format that is defined in the Web-Based Enterprise Management (WBEM) standard (see `http://www.dmtf.org/standards/wbem`). The SLD bridge then stores the data in the SLD server.

If you have installed multiple SLD servers (for example, for security purposes), the SLD bridge can be configured to forward data to each of these servers. In the figure above, one SLD server has been installed locally (indicated by 1 in the figure) and multiple SLD servers have been installed remotely (indicated by 2 in the figure).

For more information, see Configuring the SLD Bridge [page 58].

ABAP Data Collection Programs

The SLD ABAP data supplier collects and sends data about the current AS ABAP system to the SLD. To collect this data, the ABAP data supplier uses data collection programs.

You can modify data collection programs only for AS ABAP. For AS Java it is a fixed list.

The following table shows the type of data that is provided by the ABAP data collection programs that are available:

Data Collection Program	System Element for Which Data Is Provided
_SLD_APPL_SERV	Application servers/instances
_SLD_BCSYS	SAP system
_SLD_CLIENT	Clients
_SLD_COMPSYS	Hosts
_SLD_DBSYS	Database
_SLD_GWSRV	RFC gateway
_SLD_HTTPSERV	HTTP port
_SLD_INSTSC	Installed software components
_SLD_INSTSP	Installed support packages
_SLD_IPSERV	Network services
_SLD_LIVECHACHE	liveCache
_SLD_MSGSERV	SAP message server
_SLD_NWSYS	Compound System
_SLD_RFC	RFC destinations (deactivated by default) If you want to create additional information about RFC destinations in the SLD, you can activate this data collection program.
_SLD_ASSOC	CIM associations between generated objects

Do not deactivate any of the data collection programs without consulting SAP.

You can set a time interval for collecting data about the current system. More information: Setting Up the SLD Data Supplier for AS ABAP Based Systems [page 12]

Configuring Systems to Connect to the SLD

Use

You configure systems that are installed in your system landscape to periodically register themselves at the SLD.

Procedure

Depending on the type of system, there are the following ways to send system data to the SLD.

Connecting AS ABAP Systems to the SLD

An ABAP-based SLD data supplier uses an RFC connection to send data about an AS ABAP system to the SLD bridge by means of a gateway.

> The ABAP-based SLD data supplier (transaction RZ70) is available for all SAP systems as of SAP Basis release 4.0B.

More information: Connecting AS ABAP Systems to the SLD [page 11].

Connecting AS Java Systems to the SLD

A Java EE based SLD data supplier uses HTTP to send data about an AS Java system to the SLD.

> The Java EE based SLD data supplier is available for all AS Java systems as of SAP NetWeaver release 6.40.

More information: Connecting AS Java Systems to the SLD [page 18]

Connecting Other Systems to the SLD

Systems that are neither ABAP nor Java EE based use the `sldreg` executable to send data to the SLD.

More information: Connecting Other SAP Systems to the SLD [page 20]

Connecting AS ABAP Systems to the SLD

Use

The following figure shows how AS ABAP systems send system data to the SLD:
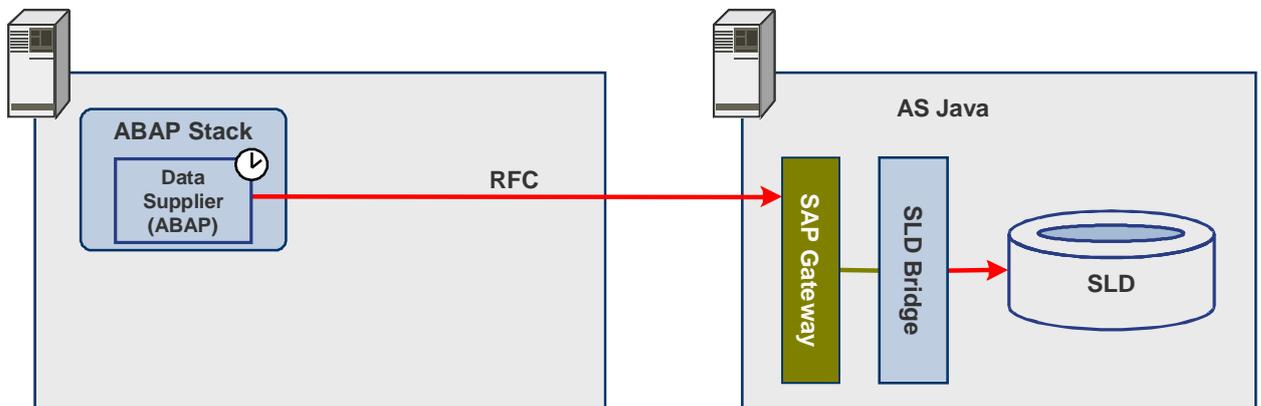
The data supplier periodically sends data to the SLD bridge via the SAP Gateway by using an RFC connection.

More information:

Architecture Overview of the SLD Data Suppliers [page 8]

ABAP Data Collection Programs [page 10]

Procedure

Configuring the SLD Data Supplier

If your AS ABAP system is running on a release lower than 6.30 and contains the data supplier function, you have to perform an initial configuration of the SLD data supplier.

More information: Setting Up the SLD Data Supplier for AS ABAP Based Systems [page 12]

Using SLD API on ABAP-Based Systems

Some components, such as SAP NetWeaver PI and the SAP Solution Manager, use the SLD Application Programming Interface for ABAP-based systems (SLD ABAP API) to read data from and write data to the SLD.

More information: Using SLD API on ABAP-Based Systems [page 13]

Setting Up the SLD Data Supplier for AS ABAP Based Systems

Use

As of SAP Web AS ABAP Release 6.30, the initial configuration of the data supplier is performed automatically during installation. No further configuration is required unless you want to change the default settings.

If your AS ABAP system is running on release lower than 6.30 and contains the data supplier function, you have to perform an initial configuration of the SLD data supplier.

We recommend that you use the following default settings:

- Use an RFC connection with an automatic RFC destination to send data directly to the SLD gateway. As of SAP NetWeaver 7.1, usage type AS Java that comprises SLD also contains an SAP Gateway.

- Use a standard set of predefined data collection programs. More information, ABAP Data Collection Programs [page 10].

- Schedule the data collection programs to be executed in a batch job once every 720 minutes (12 hours).

Prerequisites

The SLD and the SLD bridge (the receiving thread of the SLD which runs on the AS Java system) are running.

Procedure

Configure the RFC Connection

1. Call transaction RZ70.

2. In the *SLD Bridge: Gateway Information* group box, enter the following data:

   You can find the necessary data in AS Java, on the following page:
   http://<host>:<port>/sld → *Administration* → *Details* → *Data Suppliers*

   a. In the *Host* field, enter the name of the gateway host.

   b. In the *Service* field, enter the name of the gateway service.

      Make sure that the entries for the gateway host and service are exactly the same as in the SLD Bridge. Avoid the usage of one gateway by more than one SLD bridge.

Scheduled Data Collection

3. Enter the required amount of time in field *Period of Minutes*. It will determine how often the data gathering transfer will take place.

4. Choose *Schedule Background Job* button.

   By default, the *Schedule background job indicator* is selected. If you deselect it, the data about the system will be sent to the SLD only if you trigger the data collection programs manually.

5. In the *Period in Minutes* field, enter a time interval in minutes for collecting data.

   The default setting is 720 minutes (12 hours). This means that the data about the current system will be updated in the SLD twice a day.

6. Save these settings by choosing *Schedule Background Job*.

7. Choose 🕐 *Start SLD Data Collection Now*.

   A separate screen appears with a list of the executed programs and you are informed whether the initial transfer of data by RFC has been successful. For more information, see SAP Note 584654.

Single Data Collection

If you want to perform the data collection process once, you do not need to schedule it. Choose the *Start SLD Data Collection Now* button.

Result

You have set the data supplier to send data about the current system directly to the SLD by using an RFC connection. A scheduled batch job updates the data about the system in the SLD twice a day.

Using SLD API for AS ABAP Based Systems

Use

Some components, such as SAP NetWeaver PI and the SAP Solution Manager, use the SLD Application Programming Interface for ABAP-based systems (SLD ABAP API) to read data from and write data to the SLD.

The SLD ABAP API is delivered in the SLD_API package of the AS ABAP system release 6.20 and higher.

If you run applications that read data from and write data to the SLD and the connection between the SLD ABAP API and the SLD is not configured, you have to do it manually.

> For more information about configuring the connection to an ABAP system, which is part of the SAP NetWeaver PI, see Connecting to the System Landscape Directory [external].

Both RFC and HTTP protocols are used for connecting the SLD ABAP API and the SLD. You have to configure both the RFC channel from ABAP to Java and the HTTP connection parameters. The following figure shows the configuration of the SLD ABAP API in detail:
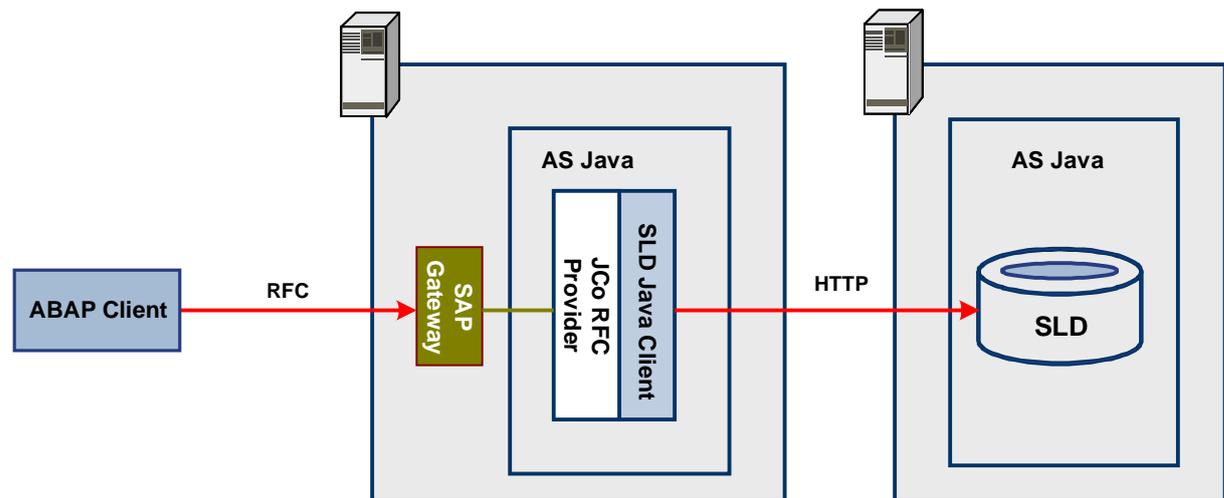
We recommend that you use the same AS Java and a local gateway for the whole infrastructure.

> As of SAP NetWeaver 7.1, usage type AS Java that comprises SLD also contains an SAP Gateway.

In this case, only the RFC communication is remote. If you are running a system with both usage types - AS ABAP and AS Java, the corresponding AS Java can be used alternatively as an RFC provider. Then only the HTTP communication is remote.

Procedure

Creating an RFC Destination

More information: Creating an RFC Destination for the SLD ABAP API on the Java EE Side [page 15]

More information: Creating an RFC Destination for the SLD ABAP API on the ABAP Side [page 16]

Creating an HTTP Destination

More information: Creating an HTTP Destination for the SLD ABAP API on the ABAP Side [page 17]

Creating an RFC Destination for the SLD ABAP API on the AS Java Side

Use

If you want to configure an ABAP application to access data in the SLD using the SLD ABAP API, you have to create a Jco RFC destination on the AS Java system that hosts the SLD.

More information: SAP Java Connector [external]

Procedure

1. In your web browser enter the URL of SAP NetWeaver Administrator using the following pattern:

   ```
   http://<host>:<port>/nwa
   ```

2. Choose *Configuration Management → Infrastructure → Jco RFC Provider*.

3. Choose *Create*.

4. Enter the connection and transport parameters as required.

Field	Description
Program ID	We recommend that you use the following pattern: **SLDAPI_<SAPSID>**, where SAPSID is the system identifier of the AS Java system.
Gateway host *Gateway service*	If your AS Java is an add-on installation to an ABAP system, use the gateway of the ABAP system. You can look up the gateway parameters of an ABAP system using transaction SMGW. If your AS Java is a standalone installation, we recommend using a gateway on the Central Services Instance host. As of SAP NetWeaver 7.1, usage type AS Java that comprises SLD also contains an SAP RFC Gateway.
Server Count (1..20)	Enter **3**.

5. Choose *Next*.

6. Enter logon data to the ABAP repository. You can either use an existing RFC destination from *SAP NetWeaver Administrator → Configuration Management → Security → Destinations* or you specify the ABAP repository explicitly. The destination should point preferably to an AS ABAP close to SLD. If SLD is running on an AS Java standalone installation, the calling AS ABAP can be used instead. We recommend to create and use an RFC destination instead of explicitly specifying the connect parameters here. In both cases, the following parameters are required

Field	Description
Application Server Host *System Number* *Client* *Language* *User* *Password*	Enter the logon data for the AS ABAP system. You must specify an existing user with appropriate authorizations for RFC. If you are running a system with both usage types AS ABAP and AS Java, use the logon data for this system. If your AS Java is a standalone installation, use the logon data for the calling ABAP system.

7. Choose *Next*.

8. Leave the default values in the next steps.

9. Create the Jco RFC destination by choosing *Finish*.

Result

The Jco RFC destination creates a number of "Server Count" RFC server threads on your AS Java. These RFC servers register at the RFC Gateway specified. You may configure multiple ABAP systems to connect to these RFC servers. In this case, the Server Count should be increased when TIME_OUT exceptions are observed.

Creating an RFC Destination for the SLD ABAP API on the AS ABAP Side

Use

If you want to configure an ABAP application to access data in the SLD through the SLD ABAP API, you have to create an RFC destination on the ABAP system side.

Prerequisites

An RFC destination for the SLD ABAP API is created on the AS Java side.

Procedure

1. Call transaction SM59.

2. Select *TCP/IP connections* and choose *Create*.

3. In the *RFC Destination* field, enter `SAPSLDAPI`.

4. On the *Technical Settings* tab page, select the *Registered Server Program* indicator.

5. In the *Program ID* field, enter the same program ID as the one you entered when you created an RFC destination for the SLD ABAP API on the AS Java side. We recommend that you use the following pattern: `SLDAPI_<SAPSID>`, where SAPSID is the system identifier of the AS Java system.

6. Choose *MDMP & Unicode* and select the *Unicode* indicator.

7. Choose *Save*.

8. Test the RFC destination by choosing *Connection Test*.

    If the display shows the response times, you have configured the RFC destination correctly.

More information

Creating an RFC Destination for the SLD ABAP API on the AS Java Side [page 15]

Creating an HTTP Destination for the SLD ABAP API on the AS ABAP Side

Use

If you want to configure an ABAP application to access data in the SLD through the ABAP API, you have to configure the HTTP connection parameters on the ABAP system side.

Prerequisites

- An RFC destination for the SLD ABAP API has been created on the AS Java side.

  More information: Creating an RFC Destination for the SLD ABAP API on the AS Java Side [page 15]

- An RFC destination for the SLD ABAP API has been created on the AS ABAP side.

  More information: Creating an RFC Destination for the SLD ABAP API on the AS ABAP Side [page 16]

Procedure

1. Call transaction SLDAPICUST.

2. Switch to CHANGE mode by choosing *Display <-> Change*.

3. Choose *Append Row*.

4. Enter the HTTP connection parameters in the following fields:

Field	Description
Alias Name	Enter a string that uniquely identifies the HTTP connection record in the table.
Host Name *Port*	Enter the host name and HTTP port number of the SLD.
User	Enter the user name of the user who logs on to the SLD. The user has to be assigned to the LcrInstanceWriterLD role.

5. Select the *Prim.* indicator.

   The table may have several HTTP connection records that point to different SLD servers. Only one of them can be selected as primary. The primary HTTP connection is the default connection for all applications that use the SLD ABAP API.

6. Choose the *Password* button.

7. Enter a password and choose *Continue*.

8. Choose *Save*.

9. If you want to check the status of the HTTP connection, select a row and choose the *Test* button.

Connecting AS Java Systems to the SLD

Use

Each AS Java system has a built-in mechanism for reporting data to the central SLD that you can configure. In addition, other applications that run on the AS Java system can find out where the central SLD is located and use this information to connect. Bear in mind that in special cases, such as for testing or development, the destination SLD to which other application data is reported may be different from the one to which landscape data of your AS Java system is reported.

> You must configure the SLD target even if the central SLD is running on the local AS Java system.

In addition, SLD provides every Java application with the option to generate an SLD client for accessing an SLD server.

Configuring the SLD Data Supplier for AS Java Systems

More information:

Configuring the SLD Client

More information:

More information

SLD Data Supplier Service Properties [external]

Setting Up the SLD Data Supplier for AS Java Based Systems

Use

An initial configuration of the data supplier for AS Java based systems is automatically performed by the SAP installation tool during the AS Java system installation. However, if you want to change the default settings, or if you have installed the AS Java in a different way, you can use the SAP NetWeaver Administrator to configure the data supplier manually.

Prerequisites

*   The SLD server is running.

*   You are logged on to the SAP NetWeaver Administrator (`http://<host>:<port>/nwa`) as an AS Java administrator.

Procedure

1. Choose *Configuration Management* → *Security Management* → *Destinations*

2. Create a destination of type **HTTP** and with name **SLD_DataSupplier**.

   If such a destination already exists you can check if its value suits you and use it.

   a. Choose *Create* (or *Edit* if it already exists).

   b. Enter the required data.

   c. You can use *Ping Destination* to check the correctness of the entered parameters.

3. When ready, choose *Save*.

Result

You can now collect and send data to the SLD Server. To do so:

1. Choose *Configuration Management → Infrastructure Management → SLD Data Supplier Configuration.*

2. Choose *Collect and Send Data*.

Configuring the SLD Client

Use

The SLD service provides every Java application with the option to generate an SLD client for accessing an SLD server. The SLD client is common for all applications that are using it and therefore plays the role of a standard SLD. Changing the data for the SLD client may have an effect on many applications. We recommend that you consider it carefully.

Procedure

3. To change the default SLD in SAP NetWeaver Administrator, go to *Configuration Management → Security Management → Destinations*.

4. Change the connect parameters in HTTP Destination "SLD_Client"

   Both the sending and the receiving AS Java must be configured for Secure Socket Layer (SSL) connections before you can use HTTPS.

   More information: <u>Configuring the Use of SSL on the AS Java [external]</u>

Information About the SLD Data Supplier Service in SAP NetWeaver Administrator

Procedure

1. In SAP NetWeaver Administrator navigate to *Configuration Management → Infrastructure Management → SLD Data Supplier Configuration*. You can find there information about the status of the last data transfer.

2. Navigate to the destination maintenance to check the connect parameters in HTTP Destination SLD_DataSupplier.

   On the *Information* tab page, the following fields are displayed:

Field	Description
Used URL [http host:port]	Connection data (except the SLD AS Java user password) of the last transfer.
Send Time	Indicates the date and time of the last data transfer.
Automatic Send	Indicates whether the displayed transfer attempt was a manual attempt (`Manual`) or an automatic one (`Automatic (Time Scheduled)`).
Success	Indicates for each send attempt whether the data has been transferred to the SLD correctly (`true`) or not (`false`).
Next Automatic Send time	Indicates when the next automatic transfer from a node of the AS Java cluster will start.

More information: SLD Data Supplier Service Properties [external]

Connecting Other SAP Systems to the SLD

Use

The `sldreg` executable is used to transfer data to the SLD for a system that is neither AS ABAP nor AS Java based. When triggered by other SAP systems, the `sldreg` sends data in an XML format using a predefined document type definition (DTD). It uses an HTTP connection, as shown in the figure below.
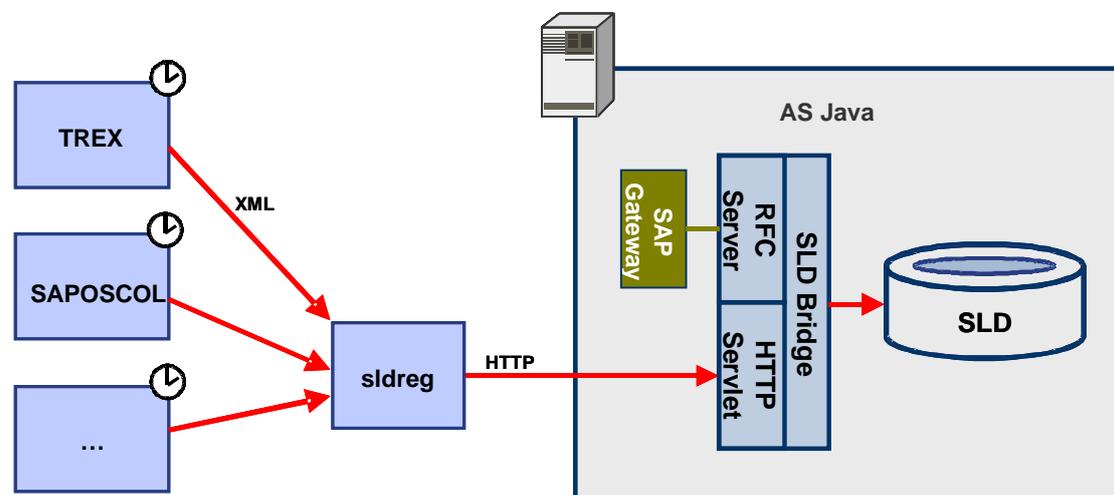
For example, Search and Classification (TREX) and SAPOSCOL generate such an XML and trigger the transfer of data to SLD via `sldreg`.

The `sldreg` executable and the required libraries, which are linked dynamically, are installed with your AS ABAP or AS Java based system.

Prerequisites

- The SLD server is running.

- You have a user assigned to the DataSupplierLD role. The user must be specific for this computer and must not be used on another computer.

- You have logged on to the operating system as a <sid>adm user.

Procedure

From the command line, change the directory to **/usr/sap/<SAPSID>/SYS/global**.

Configuring sldreg

1. To check the path to the program, enter sldreg.

2. Press *Enter*. The sldreg command line parameters are printed.

3. Enter **sldreg -configure slddest.cfg –usekeyfile**

4. You are prompted to enter HTTP user, password, host, and port information.

   The program creates the *slddest.cfg* and *slddest.cfg.key* files.

   Make sure that users other than <sapsid>adm do not have read and write permissions for the *slddest.cfg.key* file.

   Using this global directory, all generic data suppliers of the current SAP NetWeaver system can send data to the same SLD with one central configuration.

Transferring Data to SLD

Test your configuration by sending detailed data about your computer to SLD:

1. Enter **saposcol –b**.

   A short XML document is printed.

2. Enter **saposcol –b | sldreg -connectfile slddest.cfg -stdin - oldtransferdtd**

   The **oldtransferdtdoption** is required for compatibility reasons in release 7.0 of sldreg and is specific for SAPOSCOL only.

The program output is as follows:

```
HTTP response: Success. HTTP status code: 200

End SLD Registration

Data send status: true
```

You can see your computer as an SAP_ComputerSystem instance in SLD. In addition, details such as the sizes of the physical and virtual memory are reported.

Working with SLD

Use

SLD is the central information repository for your system landscape. It contains information about technical systems, landscapes, business systems, products, and software components in your system landscape. You can view and manage this information in the SLD.

Prerequisites

You have a user assigned to a particular SLD security role. For more information about SLD security roles: Configuring SLD Security Roles [page 53]

Procedure

In your Web browser, enter the URL of the SLD using the following pattern:

http://<host>:<port>/sld.

Working with Technical Systems

More information: Working with Technical Systems [page 22]

Working with Landscapes

More information: Working with Landscapes [page 28]

Working with Business Systems

More information: Working with Business Systems [page 33]

Working with Products and Software Components

More information: Working with Products and Software Components [page 38]

Software Catalog [page 38]

Products and Software Components [page 39]

Working with Names and Name Prefixes

More information: Working with Names and Name Prefixes [page 45]

Working with Data on CIM Level

More information: Working with Data on CIM Level [page 50]

Working with Technical Systems

Use

Technical systems are application systems that are installed in your system landscape. If configured, they automatically register information in the SLD about most of the elements that they contain by using the SLD data supplier. If your system is not able to register data in your SLD, then you have to register these missing technical system elements manually. Also, you might have to register manually installed products on a technical system if they are not reported to SLD automatically.

More information: Technical Systems [page 23]

Procedure

Creating New AS ABAP Technical Systems

AS ABAP technical systems are normally registered in the SLD server automatically. If automatic registration is not possible, you have to create the new AS ABAP technical systems manually.

For more information: Creating New AS ABAP Technical Systems [page 24]

Creating New AS Java Technical Systems

AS Java technical systems are normally registered in the SLD server automatically. If automatic registration is not possible, you have to create the new AS Java technical systems manually.

For more information: Creating New AS Java Technical Systems [page 25]

Creating New Standalone Technical Systems

Standalone technical systems are not registered in the SLD server automatically. You have to register them manually.

For more information: Creating New Standalone Technical Systems [page 26]

Creating new Third-Party Technical Systems

Third-party technical systems are typically not automatically registered in the SLD server. You normally have to create them manually.

For more information: Creating New Third-Party Technical Systems [page 27]

Editing Technical Systems

If required you can edit parameters of already registered technical systems.

For more information: Editing Technical Systems [page 27]

Technical Systems

Definition

Technical systems are application systems that are installed in your system landscape. An example of an application system is an SAP Customer Relationship Management (CRM) server. The following technical systems are included in the SLD:

- AS ABAP

   AS ABAP technical systems, ranging from any SAP system to an SAP SCM server or CRM server, are based on the same Basis Component (BC), which includes the ABAP interpreter, ABAP Dictionary, ABAP Workbench, and so on. A dedicated database, which stores most of the data of the BC and the application components, is associated with the system.

- AS Java

   AS Java systems consist of one or more instances and each instance is usually installed on a separate host. An instance itself consists of cluster nodes that can communicate with each other.

- Standalone Java

   Standalone Java technical systems are standalone Java applications that are installed on a computer system from an administration perspective. They can also be installed as an operating system service or as a daemon.

- Third-Party

   Third-party technical systems contain third-party software components and products.

   Third-party systems are normally not automatically registered in the SLD. Therefore, you normally have to register them manually by creating:

   - Third-party products [page 41]

   - Third-party software components [page 42]

   - Third-party technical system [page 27]

   - Third-party business system [page 34]

- TREX

Search and Classification (TREX) is the central SAP NetWeaver search engine that provides underlying search technology and functions for Enterprise Knowledge Management.

- Process Integration (XPI)

   Process Integration technical systems are the infrastructure systems (such as integration servers and adapter engines) of SAP NetWeaver Process Integration.

- Computer Systems

   Computer systems (hosts) are described in SLD with a few selected attributes only. Computer systems are identified by a host name written in lower case without network domain. If virtual host names or aliases are defined, identical hardware may appear multiple times in the SLD.

Use

AS ABAP and AS Java technical systems automatically register information in the SLD about most of the elements that they contain if you have configured the SLD data supplier accordingly. Not all technical system elements are registered automatically in the SLD by the reporting system – if required, you have to register these missing elements manually. You also have to register products manually installed on a technical system if you require this information, because a technical system has no information about which products are installed on it

If automatic registration of AS ABAP and AS Java technical systems is not possible, you have to register corresponding systems manually. You normally also have to register manually standalone and third-party technical systems.

Systems that comprise both AS ABAP and AS Java (AS ABAP + AS Java systems) appear in SLD both as an AS ABAP and as an AS Java system.

More information:

Creating New AS ABAP Technical Systems

Use

AS ABAP technical systems are registered in the SLD server automatically. If automatic registration is not possible, you have to create the new AS ABAP technical systems manually.

Prerequisites

The SLD data supplier is configured accordingly.

You have a user assigned to the `LcrInstanceWriterLD` role.

Procedure

1. Choose *Home → Technical Systems*.

2. Choose *New Technical System*.

3. Select the *AS ABAP* radio button. The appropriate sequence of steps is loaded.

4. Choose *Next*.

5. Enter system details. All fields marked with asterisk are compulsory.

6. Choose *Next*.

7. Enter message server and central application server details.

8. Choose *Next*.

9. If you want to define additional application servers, choose *Add Application Server*, enter the host name and instance number.

    💡

    You can skip this step if you do not need an additional application server.

10. Choose *Next*.

11. To add at least one ABAP client, choose *Add New Client*, enter client number and client logical name.

    💡

    The ABAP client number is required, it must be a three-digit number. The client logical name is optional.

12. Choose *Next*.

13. Select installed software products from the *Installed Products* by selecting the *Installed* indicator. Choose *Filter On* and use the text filter to search for the right product.

    All software components that are part of the selected software products appear in the *Installed Software Components* list.

14. Select installed software components from the *Installed Software Components* by selecting the *Installed* indicator.

15. Choose *Finish*.

Result

You have manually created an AS ABAP technical system.

For more information how to view details about the technical systems, go to .

⬇ Creating New AS Java Technical Systems

Use

AS Java technical systems are registered in the SLD server automatically. If automatic registration is not possible, you have to create the new AS Java technical systems manually.

Prerequisites

The SLD data supplier is configured accordingly.

You have a user assigned to the `LcrInstanceWriterLD` role.

Procedure

1. Choose *Home* → *Technical Systems*.

2. Choose *New Technical System*. The *Technical System Wizard* appears.

3. Select the *AS Java* radio button. The appropriate sequence of steps is loaded.

4. Choose *Next* to go to the *General* step.

5. If the AS Java technical system is part of a dual-stack system, from the *AS ABAP* dropdown box, select the AS ABAP technical system for the AS Java. If the AS Java technical system is a standalone AS Java system, select *None (Standalone AS Java)*.

6. Enter system details.

7. Choose *Next* to go to the *Instances* step.

8. If you want to define new instances, choose *Add New Instance* and enter instance details.

9. If you want to define new server nodes:

   a. Select an instance from the *Instance List*.

   b. In the *Server Nodes* choose *Add New Server Node* and enter server node details.

10. If you want to define new ports

    a. Select an instance from the *Instance List*.

    b. In the *Ports* choose *Add New Port* and enter port details.

11. Choose *Next* to go to the *Ports* step.

12. If you want to define central instance ports, for each port, enter a port number and select a protocol.

13. Choose *Next* to go to the *Installed Software* step.

14. Select installed software products from the *Installed Products* list by selecting the *Installed* indicator.

    All software components that are part of the selected software products appear in *Installed Software Components*.

15. Select installed software components from the *Installed Software Components* by selecting the *Installed* indicator.

16. Choose *Finish*.

Result

You have manually created an AS Java technical system in your SLD.

More information how to view and change its details: <u>Editing Technical Systems [page 27]</u>

Creating New Standalone Technical Systems

Use

Normally standalone technical systems are not registered in the SLD server automatically so you have to register them manually.

Prerequisites

You have a user assigned to the `LcrInstanceWriterLD` role.

Procedure

1. Choose *Home → Technical Systems*.

2. Choose *New Technical System*.

3. Select the *Standalone* radio button

4. Choose *Next*.

5. Enter system details.

6. Choose *Finish*.

Result

You have manually created a standalone technical system.

More information how to view and change its details: Editing Technical Systems [page 27]

Creating New Third-Party Technical Systems

Use

Third-party technical systems are typically not automatically registered in the SLD server, you have to create them manually. This procedure describes how to register an installed third-party system.

> Unlike SAP products, third-party products are not necessarily included in the SLD component description. Therefore, you have to create the third-party product before you can create an installed system of this third-party product in the SLD.

Prerequisites

You have a user assigned to the `LcrInstanceWriterLD` role.

Procedure

1. Choose *Home → Technical Systems*.

2. Choose *New Technical System*.

3. Select the *Third-Party* radio button.

4. Choose *Next* to go to the *General* step.

5. Enter system details.

6. Choose *Next* to go to the *Installed Software* step.

7. Select installed software products by selecting the *Installed* indicator from the *Installed Products* list.

   All software components that are part of the selected software products appear in the *Installed Software Components* list.

8. Select installed software components from the *Installed Software Components* list by selecting the *Installed* indicator.

9. Choose *Finish*.

Result

You have manually created a third-party technical system in your SLD.

More information how to view and change its details: Editing Technical Systems [page 27]

Editing Technical Systems

Use

Check what systems are registered in the SLD and tune their parameters according to your requirements.

Procedure

1. Choose *Home → Technical Systems.*

2. Select a system. Its details are shown in *Java System Details* list.

3. Select a tab from the *Java System Details* list and enter the required information.

Working with Landscapes

Use

A landscape represents a logical, complex unit that consists of multiple, distributed components. Some of these components may be systems; others may be services, installed products, or other managed elements.

More information: Landscapes [page 28]

Procedure

Creating and Removing Landscapes

For more information: Creating and Removing Landscapes [page 29]

Adding and Removing Systems

For more information: Adding and Removing Systems [page 30]

Creating and Removing Hosted Systems

For more information: Creating and Removing Hosted Systems [page 31]

Adding and Removing Sub-Landscapes

For more information: Adding and Removing Sub-Landscapes [page 32]

Landscapes

Definition

A landscape represents a logical, complex unit that consists of multiple, distributed components. Some of these components may be systems; others may be services, installed products, or other managed elements.

SLD offers the following types of landscapes:

*   *Administration*

    A system landscape that is created for administration purposes, such as for common monitoring.

*   *General*

    A system landscape that can contain all types of application systems.

*   *NWDI Systems*

    A system landscape that contains servers of the NWDI. The NWDI consists of a local client – the Integrated Development Environment (IDE) and some logical servers for processes like source management, build tasks, transport, and name reservation. All servers work together and require a common user base together with the IDE.

*   *Scenarios*

A system landscape that is created for a specific business scenario. Scenario landscapes are used in the Software Lifecycle Manager of SAP NetWeaver.

- *Transports*

    A system landscape that is created for transport purposes. For example, all SAP AS systems that are connected by an ABAP transport path distributing the ABAP development.

- *Web Services*

    A system landscape that is created to collect systems offering Web services and their clients. The hierarchy of these landscapes defines the Web service that is most appropriate for a given client.

More information

Working with Landscapes [page 28]

Creating and Removing Landscapes

Prerequisites

You have a user assigned to the LcrInstanceWriterLD role.

Creating Landscapes

4. Choose *Home → Landscapes → New Landscape*.

5. From the *Landscape Type* dropdown box, select the landscape type you want to create.

6. Enter a landscape name.

7. Choose *Create*.

Removing Landscapes

1. Choose *Home → Landscapes*.

2. From the *Landscape Type* dropdown box, select the landscape type of the landscape that you want to remove.

3. From *Landscapes* table, select the landscapes you want to remove by holding down the CTRL key and clicking the landscapes with the primary mouse button.

4. Choose *Remove*.

    The landscapes that you have selected are removed but the systems that are part of these landscapes are not removed.

    More Information:

    Technical Systems [page 23]

    Business Systems [page 33]

    Creating and Removing Hosted Systems [page 31]

Adding and Removing Systems

Use

You can add the following systems to the following system landscapes:

System	System Landscape
Technical	Administration, General and Transports
Business	General
Hosted	General and NWDI systems

Prerequisites

- You have a user assigned to the `LcrInstanceWriterLD` role.

- If you want to add a technical system to a landscape, the technical system has an instance in the SLD. For more information about creating a technical system, see Technical Systems [page 23].

- If you want to add a new hosted system to a landscape, you have created it. For more information, see Creating and Removing Hosted Systems [page 31].

Procedure

1. Choose *Home → Landscapes*.

2. From the *Landscape Type* dropdown box, select the landscape type to which you want to add or from which you want to remove a system.

3. From the table with landscapes, select the landscape to which you want to add or from which you want to remove a system.

4. Choose the *Systems* tab.

Adding Systems to Landscapes

1. Choose *Add System to Landscape*.

2. From the *Technical System Type* or *System Type* dropdown box depending on the landscape type, select the system type you want to add to the landscape.

3. Select the systems you want to add by holding down the CTRL key and clicking the systems with the primary mouse button.

4. Choose *OK*.

Removing Systems from Landscapes

1. Select the systems you want to remove by holding down the CTRL key and clicking the systems with the primary mouse button.

2. Choose *Remove System from Landscape*.

    The systems that you have selected are removed only from the landscape.

More information:

Technical Systems [page 23].

Business Systems [page 33].

Creating and Removing Hosted Systems [page 31].

Creating and Removing Hosted Systems

Use

Hosted systems are applications that run on a technical system. Hosted systems have the same features as technical systems, that is, specific software must be installed on them. For example, an AS Java application that is deployed on an AS Java system can be a hosted system. Applications are classified as systems depending on their independence, complexity, and manageability (that is, depending on whether they are started, stopped, administered, monitored separately, and so on). You can create the following types of hosted systems:

- System Landscape Directory

  Maintains information about all installable and installed elements of your system landscape.

  Each SLD automatically creates a hosted system entry for itself in the central SLD. For more information, see Configuring the SLD Client [page 19].

- Change Management System

  It is part of the SAP NetWeaver Development Infrastructure and centrally controls the transport and deployment of software components at design time and manages design time configurations.

- Component Build Service

  It is part of the SAP NetWeaver Development Infrastructure and compiles different sources and creates build results like built software delivery files or archives.

- Design Time Repository

  It is part of the SAP NetWeaver Development Infrastructure and a source control system that is based on the Web-based distributed authoring and versioning standard.

Prerequisites

You have a user assigned to the `LcrInstanceWriterLD` role.

Procedure

1. Choose *Home → Landscapes*.
2. From the *Landscape Type* dropdown box, select *NWDI Systems*.
3. From the table with landscapes, select the necessary landscape.
4. Choose the *Systems* tab.
5. Choose *Add System to Landscape*.

Creating Hosted Systems

1. Choose *New Hosted System*.
2. Specify necessary details.
3. Choose *Create*.

   The system and appears in the list of existing hosted systems.

4. To add the system to the landscape, select the system and choose *OK*.

Removing Hosted Systems

⚠️

The system is removed not only from the landscape, but also from the SLD. If you want to use it, you have to create it again.

1. From the *Hosted System Type* dropdown box, select the hosted system type you want to remove.

2. Select the hosted systems you want to remove by holding down the CTRL key and clicking the systems with the primary mouse button.

3. Choose *Remove Hosted System.*

Adding and Removing Sub-Landscapes

Use

You can add a sub-landscape to a landscape only if they are of the same type. You cannot add the same sub-landscape to more than one landscape.

Prerequisites

You have a user assigned to the LcrInstanceWriterLD role.

Procedure

1. Choose *Home → Landscapes.*

2. From the *Landscape Type* dropdown box, select the landscape type to which you want to add or from which you want to remove a sub-landscape.

3. From the table with landscapes, select the landscape to which you want to add or from which you want to remove a sub-landscape.

4. Choose the *Sub-Landscapes* tab.

Adding Sub-Landscapes to Landscapes

1. Choose *Add Sub-Landscape.*

2. Choose *New Landscape.*

3. Enter a landscape name and choose *Create.*

4. From the table with landscapes, select the landscape you want to add and choose *OK.*

Removing Sub-Landscapes from Landscapes

1. Select the landscapes you want to remove by holding down the CTRL key and clicking the landscapes with the primary mouse button.

2. Choose *Remove Sub-Landscape.*

   💡

   The sub-landscapes that you have selected are removed only from the landscape. For more information about removing landscapes from the SLD, see .

   💡

   The sub-landscapes that you have selected are removed but the systems that are part of these sub-landscapes are not removed.

   More information:

Working with Business Systems

Use

Business systems are logical systems, which function as senders or receivers within SAP NetWeaver Process Integration.

Business systems can be SAP systems or third-party systems.

More information: Business Systems [page 33]

Procedure

Creating and Removing Business Systems

More information: Creating and Removing Business Systems [page 34]

Managing Business System Details

More information: Managing Business System Details [page 35]

Creating and Removing Business System Groups

More information: Creating and Removing Business System Groups [page 37]

Business Systems

Definition

Business systems are logical systems, which function as senders or receivers within SAP NetWeaver Process Integration. Business systems can be SAP systems or third-party systems.

Depending on the associated technical system, the following types of business systems are defined in the SLD:

- AS ABAP

  A business system, which is associated with an AS ABAP.

  A business system, which is associated with a client of an AS ABAP, can play the role either of an application system or an integration server. An integration server is the central component of SAP NetWeaver Process Integration that provides a framework for the exchange of information among various internal and external entities of a company. The integration server facilitates the interaction between the different operating systems and applications across internal and external networks.

- AS Java

  A business system, which is associated with an AS Java.

- Standalone

  A business system, which is associated with a standalone Java application.

- Third-party

  A business system, which is associated with a third-party technical system.

Third-party systems are not automatically registered in the SLD. Therefore, you have to create them manually by creating:

- Third-party products [page 41]

- Third-party software components [page 42]

- Third-party technical system [page 27]

- Third-party business system [page 34]

- Business Partner

  A business system, which is a sender or receiver within SAP NetWeaver Process Integration and which is not associated with a technical system. Business partners are used for backward compatibility with SAP NetWeaver Process Integration release 2.0.

Creating and Removing Business Systems

Use

You can create all five types of business systems in the SLD by going through the steps of a business system wizard.

Bear in mind that some of the steps in the procedure are optional because the steps in the wizard vary depending on the technical system with which you want to associate the business system.

Prerequisite

You have a user assigned to the LcrInstanceWriterLD role.

Procedure

Choose *Home → Business Systems*.

Creating Business Systems

1. Choose *New Business System*.

   The *System Type* screen appears.

2. Select the type of business system you want to create, depending on the type of technical system with which you want to associate it, and choose *Next*.

   The *Technical System* screen appears.

3. If you want to create a business system different from a business partner, from the *System* dropdown box, select the technical system with which you want to associate the business system.

   If you want to create a business system that is associated with AS ABAP, from the *Client* dropdown box, select an ABAP client. Optionally, if you want to distribute central Web service configuration data to the back-end system, from the *URL* dropdown box, select a URL from the ones that are available, or in the adjacent field, enter the URL of the selected technical system using the following pattern `http://<host>:<port>/`.

4. If you want to create a business partner, in the *Logical System Name* field, enter the primary URL of the business partner.

5. Choose *Next*.

   The *Name* screen appears.

6. Enter a name for the business system and choose *Next*.

   We recommend that you use the internal naming convention
   `<namespace>_<name>` for business systems. For `<namespace>`, enter the
   URL or the namespace of the Integration Repository. Use only alphanumeric
   characters or use underscore (_) to replace non-alphanumeric characters. For
   `<name>`, enter the role of the system. A valid example of a name is
   `SAP_COM_SRM_DEVELOPMENT`.

   If you are creating a business system that is associated either with an AS ABAP or a
   third-party technical system, the *Installed Products* screen appears.

7. If there are products that are not installed on the system, on the left, deselect them by
   deselecting the *Installed* indicator and choose *Next*.

   If you do not see any products, you have to add them in the technical systems
   [page 23] views.

   The *Integration* screen appears.

8. If you are creating a business system that is associated with AS ABAP, from the
   Business System Role dropdown box, select the role of the business system.

   If the business system plays a role of an application system, from the *Related
   Integration Server* dropdown box, you can also select an integration server.

   If the business system plays a role of an integration server, also specify a
   pipeline URL.

9. If you are creating a business system that is not associated with AS ABAP, from the
   Related Integration Server dropdown box, you can select an integration server.

10. Choose *Finish*.

Removing Business Systems

1. From the table with business systems, select the systems you want to remove by
   holding down the `CTRL` key and clicking the systems with the primary mouse button.

2. Choose *Remove*.

   If you remove a business system that is an integration server, all business
   systems that are associated with this integration server will lose their group
   assignments.

Managing Business System Details

Use

After you create a business system, you can change the following details about the business
system:

- Business system role

- Related integration server

- Integration server pipeline URL

- Group

- Associated technical system

- Logical system name

- Installed products.

In addition, you can define a transport target for a business system. A transport target is a business system to which the source business system transports content. By means of transport targets, SAP NetWeaver Process Integration content is transported between different groups of business systems.

Prerequisites

You have a user assigned to the `LcrInstanceWriterLD` role.

Procedure

1. Choose *Home → Business Systems.*

2. From the table with business systems, select the business system whose details you want to change.

If You Want To	Then
Change the general settings of the business system	1. Choose the *General* tab. 2. Change the details you want. 3. Choose *Save.*
Change the following business system details: • Role • Related Integration Server • Group • Technical System • Logical System Name	1. Choose the *Integration* tab. 2. Change the details you want. For a business system, which is associated with an AS ABAP technical system, the logical system name is identical with the ABAP client name and you can change it only in the technical systems [page 23] views. 3. Choose *Save.*

Add or change a transport target	1. Choose the *Transport* tab. 2. Choose *Add/Change Target*. 3. From the *Group* dropdown box, select the group to which you want to add a transport target or whose transport target you want to change. 4. From the *Target* dropdown box, select the business system you want. 💡 You can define only one target system for a business system group. You can define one business system as the transport target for multiple systems from different groups. 5. Choose *OK*.
Change the details about installed products	1. Choose the *Installed Software* tab. 💡 If you do not see any products, you have to add them in the technical systems [page 23] views. 2. Select the *Installed* indicator next to the software products that are installed and deselect it next to the software products that are not installed. On the right, all software components that are part of the selected software products appear. 3. Choose *Save*.

⬇ Creating and Removing Business System Groups

Use

You can create a group of all business systems that are associated with one and the same integration server. The integration server acts as a central system for the business system group.

Prerequisites

You have a user assigned to the LcrInstanceWriterLD role.

Procedure

1. Choose *Home → Business Systems*.
2. From the *Group* dropdown box, select *Edit Groups*.

Creating Business Systems Groups

1. Choose *New Group*.

2. Enter a name and optionally a location for the business system group.

3. If you want to assign an integration server to the business system group, from the *Integration Server* dropdown box, select an integration server.

All business systems that are associated with the integration server you selected are automatically assigned to the business system group.

4. Choose *Create*.

Removing Business System Groups

Select the indicator next to the group(s) you want to remove and choose *Remove*.

Working with Products and Software Components

Use

The SLD contains a Software Catalog of all installable SAP products and software components. The Software Catalog includes information about support packages and dependencies between the products and software components. This information is the basis for the description of the system landscape.

More information:

Software Catalog [page 38]

Products and Software Components [page 39]

Procedure

Viewing the Software Catalog

More information: Viewing the Software Catalog [page 40]

Creating and Removing Third-Party Products

More information: Creating and Removing Third-Party Products [page 41]

Creating and Removing Third-Party Software Components

More information: Creating and Removing Third-Party Software Components [page 42]

Defining and Removing Software Dependencies

More information: Defining and Removing Software Dependencies [page 43]

Software Catalog

Definition

The SLD contains a software catalog of all installable SAP products and software components. The software catalog includes information about available support packages and existing dependencies between the products and software components. This information is the basis for the description of the system landscape.

In addition, third-party products and software components can be implemented in a system landscape. Third-party products can be standalone applications or additional modules for

SAP applications. In the SLD, you normally have to create third-party products and software components manually.

More information:

Products and Software Components

The classes of the SAP schema, the software components and their versions, and the products and their versions are the basis for the component information.

This section describes the connection between these terms as shown in the following figure:
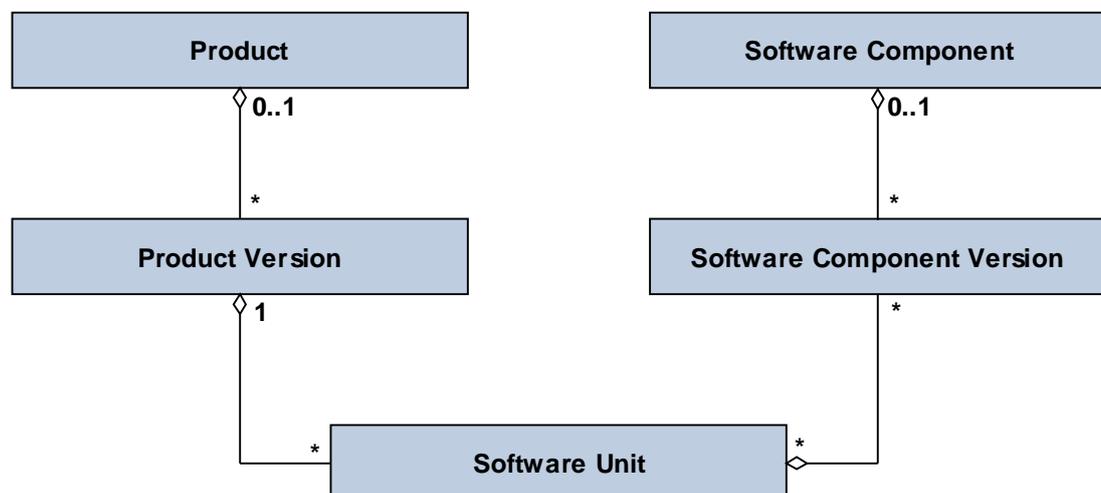
- Product

  A collection of all versions of a product. A product is a unit that is delivered, visible to the customer, and is installable and renewable. In an SAP environment, a product corresponds to an SAP technical component.

- Product version

  A particular version of a product.

- Software component

  A collection of all versions of a software component. Software components represent the reusable modules of a product. You can upgrade them or install patches on them.

- Software component version

  A particular version of a software component.

- Software unit

  A particular aspect of the product functions that is always installed as a bundle. It is a logical link between the product version and the software component version.

As shown in the figure above, a product version consists of multiple software units. These units are implemented by various software components (or versions).

The product SAP R/3 Enterprise, which represents an SAP technical component, has, among other things, a product version SAP R/3 Enterprise Core 4.7. The software unit that represents the functions of this product version has the same description as the product version. Various software components (or versions) fill the functions of this software unit. These software components (or versions) are, among others, SAP Basis 6.20, SAP HR 4.70, and SAP Enterprise Retail 1.10.

The data about SAP software components and SAP products is maintained by SAP. You cannot change this data. You can add current data to your SLD by using the data import [page 60] function.

Viewing the Software Catalog

Use

You can use the Software Catalog views in the SLD to view data about SAP and third-party registered products and their versions, as well as SAP and third-party registered software components and their versions. The data about SAP products and software components is provided by SAP. You cannot change this data.

Procedure

1. Choose *Home* → *Products*.

   The *Software Catalog* screen appears.

Viewing Products

1. From the *Software Type* dropdown box, select *Products*.

   A table with product versions appears.

2. Select the product with required product version whose details you want to view. Choose one of the following tabs:

   ○ *General*

      Displays general details about the product version.

   ○ *Software Components*

      Displays the software units of the product version.

   ○ *Support Package Stacks*

      Displays the support package stacks that are recommended for the product version.

   ○ *Installed Systems*

      Displays the systems, on which the product version is installed.

Viewing Software Components

1. To view details about a software component, from the *Software Type* dropdown box, select *Software Components*.

   A table with software component versions appears.

2. Select the software component with the required version whose details you want to view. Choose one of the following tabs:

- *General*

  Displays general details about the software component version.

- *Products*

  Displays the products, which contain the software component version.

  If you choose the name of a product, the *Product* tab page is displayed.

- *Dependencies*

  Displays the dependencies between the software component versions in different contexts: `BuildTime`, `InstallationTime`, and `MetaDataRequest`.

- *Support Packages*

  Displays the support package stacks that are recommended for the software component version.

- *Release Compatibility*

  Displays compatible software component versions, which can replace the software component version.

- *Release History*

  Displays the predecessor and successor software component versions of the software component version.

- *Installed Systems*

  Displays the systems, on which the software component version is installed.

Creating and Removing Third-Party Products

Use

You can use this procedure to create third-party products. As part of this procedure, you can also add software units and software component to the products (at the same time or later).

More information: Creating and Removing Third-Party Software Components [page 42])

You can create different versions of a registered third-party product. You have to specify exactly the same vendor and name for the product, but you have to specify version numbers accordingly.

Prerequisites

You have a user assigned to the `LcrInstanceWriterCR` role.

Procedure

Choose *Home → Products*.

Creating Third-Party Products

1. From the *Software Type* dropdown box, select *Products*.

2. Choose *New*.

3. Choose *Create a new product and verion*.

4. Choose *Next*.

5. Enter the required information about the third-party product.

   Enter the vendor as a URL.

6. Choose *Next*.

7. Enter a name for a software unit for the third-party product.

8. Choose *Next*.

9. Enter name and version for a software component you want to add to the third-party product.

10. Choose *Finish*.

   Instances for the `Product` and `Product Version` CIM classes, as well as the association instances are created.

   Instances for the `Software Unit` CIM class are created.

   Instances for the `Software Component` and `Software Component Version` CIM classes, as well as the association instances are created.

Removing Third-Party Products

1. From the table with product versions, select the product you want to remove.

2. Choose *Remove*.

Creating and Removing Third-Party Software Components

Use

You can use this procedure to:

- create a third-party software component and add it to a third-party product that you have created earlier.

  More inform|ation: Creating and Removing Third-Party Products [page 41]

- remove third-party components

  You can create different versions of a registered third-party software component. You have to specify exactly the same vendor and name for the software component, but you have to specify version numbers accordingly.

Prerequisites

You have a user assigned to the LcrInstanceWriterCR role.

Procedure

Creating Third-Party Software Components

1. *Choose Home → Software Components*.

2. Choose *New*.

3. Choose whether to:

    ○ *Create a new version for an existing software component*

    ○ *Create a new software component and version*

4. Choose *Next*.

5. From the *Product Version* dropdown box, select the third-party product to which you want to define a third-party software component.

6. From the *Unit* dropdown box, select a software unit of the third-party product.

    Alternatively, to create a new software unit, choose *Create New Unit*, enter the required data, and then choose *Create*.

    The software unit is automatically selected in the *Unit* dropdown box.

7. In the corresponding fields, enter the vendor, name, and version of the software component.

8. From the *Production State* dropdown box, select the production state you want.

9. Choose *Finish*.

    Instances for the `Software Component` and `Software Component Version` CIM classes, as well as the association instances are created.

Removing Third-Party Software Components

1. *Choose Home → Software Components*.

2. From the table with software component versions, select the software component you want to remove.

3. Choose *Remove*.

Defining and Removing Software Dependencies

Use

The software catalog in the SLD contains information about dependencies between SAP software component versions. However, you may have to define dependencies between third-party software component versions and SAP component versions.

You can define dependencies between your software component version and each other software component version in the following contexts:

- InstallationTime

  The context of an installation process (first installation or an upgrade to a newer version). The requirements of this context list the external resources that are needed for the installation.

    Software component C1 requires software components C3 and C4 to be installed. The installation and basic usage of C1 does not work without C3 and C4 being installed first. On the other hand, the installation of C1 does not require any other software component to be installed first, except the ones that are required implicitly through C3 and C4.

- BuildTime

  The context of a build process, such as the compilation of sources and the packing of archives. The requirements of this context list the external resources that are needed for the build.

For example, software component C1 requires software components C2 and C3 to be built. The build of C1 does not work without building C2 or C3 first. On the other hand, the build of C1 does not require any other software component, except the ones that are required implicitly through C2 and C3.

- MetaDataRequest

  The context of a process which requires external metadata in an installed system.

  An example is the content of the SAP NetWeaver Process Integration repository which contains the definitions of services and interfaces. The SAP NetWeaver Process Integration content is used by the software components that implement the services and interfaces (in ABAP or Java).

You can also remove software components.

Prerequisite

You have a user assigned to the `LcrInstanceWriterCR` role.

Procedure

1. Choose *Home → Software Components*.

2. From the *Software Type* dropdown box, select *Software Components*.

3. From the table with software component versions, select the software component version for which you want to define or remove the dependencies.

4. Choose the *Dependencies* tab.

5. From the *Context* dropdown box, select the context for which you want to define or remove the dependencies.

    Prerequisite and dependent software component versions that are defined in the context you have selected are displayed.

Defining Dependencies Between Software Component Versions

1. Choose *Define Prerequisite Software Component Versions*.

2. From the *Context* dropdown box, select the context for which you want to define the dependencies.

3. From the table with software component versions, select the software component versions that are prerequisites for the software component version that you have selected, in the context that you have selected.

4. Choose *Define As Prerequisite Software Component*.

    Prerequisite and dependent software component versions that are defined in the context you have selected are displayed.

Removing Dependencies Between Software Component Versions

1. From the table with prerequisite software component versions, select the prerequisite software component versions you want to remove.

2. Choose *Remove*.

Working with Names and Name Prefixes

Use

Software development, which is globally distributed, requires a conflict-free method of creating names for software objects. Without such a method, different development teams can accidentally use the same name for software objects that have different purposes. If the software objects are then merged in the same runtime environment, this can even disable applications due to a simple naming conflict.

More information:

Name Reservation [page 45]

Object Name Categories [page 46]

Procedure

Defining Name Prefixes

More information: Defining Name Prefixes [page 48]

Reserving Names

More information: Reserving Names [page 49]

Removing Name Prefixes and Names

More information: Removing Name Prefixes and Names [page 50]

Name Reservation

Definition

Software development, which is globally distributed, requires a conflict-free method of creating names for software objects. Without such a method, different development teams can accidentally use the same name for software objects that have different purposes. If the software objects are then merged in the same runtime environment, this can even disable applications due to a simple naming conflict.

To avoid naming conflicts, SLD provides a name reservation service, also known as a name server, which allows you to reserve names that are globally unique. The underlying principle is the namespace concept. For more information about the namespace concept, see SAP Service Marketplace at `http://service.sap.com/namespaces`. A namespace is defined by a name prefix, which is the leading part of a development object name. For certain object name categories [page 46], a namespace can contain a number of subnamespaces.

Use

You can reserve a namespace prefix for your development on SAP Service Marketplace. This namespace prefix spans a unique namespace for each SAP customer. To use this reserved namespace prefix in a local NWDI system, you have to enter the prefix in the name server of the SLD. You can then use this namespace prefix in the SLD as a prefix for names that are reserved. Thus these names are globally unique. No check is implemented to guarantee that a prefix is really reserved on SAP Service Marketplace. It is the responsibility of the administrator to enter in the SLD only namespace prefixes that have been reserved at SAP.

Additionally, some namespaces use your Internet domain names to be globally unique. The Internet domain name is used in Java package names, for example, `com.sap.sld`. The Internet domain name is unique, and so is the package name that is reserved in the SLD.

Object names must adhere to the following rules:

- The name corresponds to the syntax of the object name category.

- The leading parts of a name are exactly the same as the namespace definition (name prefix).

- The closing asterisk (*) in the name prefix represents all names that belong to this namespace. For example, `testcomp` belongs to `test*`.

If you define new development objects within NWDI, the SAP NetWeaver Developer Studio provides you with all the namespaces that have already been defined for the relevant object name category. You can then choose an appropriate name prefix and complete the naming of your objects. Subsequently, the NWDI reserves this name for you on the name server.

More information:

Object Name Categories

The object name categories for the name prefixes are implemented as . The global uniqueness of prefixes is guaranteed by using Internet domains or by using a prefix that has been reserved at SAP.

You reserve a prefix globally on SAP Service Marketplace at `http://service.sap.com/namespaces` (see SAP Note 105132). The syntax for a prefix that is reserved globally on SAP Service Marketplace is a string of three to eight characters, which are enclosed between slashes (/).

This section describes all object name categories that are available and their syntax rules. The syntax rules are important when you define a new .

Object Name Categories and Syntax Rules for Non-ABAP Objects

Object Name Category	Usage	Prefix Naming Convention	Syntax Rules
Application Context Root	Names of Java EE applications. Defines the first path component in the URLs of the application.	`x`	Hierarchical name in lowercase as a first name component separated by a slash (/). For example, `x/....`
DB Field Name	Field names of any database (DB) tables. Developers of different DB tables can reuse field names. Name prefixes can be separated according to vendor in this category.	`X_*`	The prefix must have a trailing asterisk (*) to allow more than one single name. In front of the asterisk (*), you have to insert an underscore (_) to separate the name prefixes. Any string of uppercase letters and digits may be inserted in front of the underscore (_) and the asterisk (*). The minimum length is 1. For example, `X_*`. Do not insert an underscore (_) in the prefix string as this will be ambiguous.

DB Object Name	Names of DB tables and DB indexes.	X_*	See above.
DB Pool Name	Names of DB pool aliases. These aliases are used by Java EE applications to refer to a DB connection, including the physical DB, the DB schema, and access rights.	X/P_*	The prefix must have a trailing asterisk (*) to allow more than one single name. In front of the asterisk (*), you have to insert an underscore (_) to separate the name prefixes. Any string of uppercase letters, digits, and a slash (/) may be inserted in front of the underscore (_) and the asterisk (*). The minimum length is 1. For example, X/P_*. Do not insert an underscore (_) in the prefix string as this will be ambiguous.
Design Time Package	Names of Java packages.	<rid>	Reverse Internet domain (rid). For example, sap.com is the normal Internet domain and the reverse is com.sap.
Development Component Name	Names of the smallest reusable software bundles (development components).	<id>	Internet domain (id). Hierarchical name separated by a slash (/). The file path in the Design Time Repository (DTR) is preceded by a vendor path that belongs to the development component name. The vendor path at SAP is sap.com. For example, sap.com/*.
Installation Unit Name	Names that identify any type of installation unit (executable, dynamically linked library, archive, VB module, and so on). The type is designed mainly for file system elements but not necessarily restricted to them. To avoid name reservation conflicts, do not assign the same	X_*	See above.

	name to elements of different types.		
Software Component Name	Names of the smallest shipped software bundles. These bundles contain many development components.	`<id>`	Software components that have sources in the DTR have to follow the same naming rules as for development components. For example, `sap.com/*`.

Defining Name Prefixes for Development

Prerequisites

You have a user assigned to the `LcrAdministrator` role.

Procedure

1. Choose *Home → Name Reservation*.

2. To enable the SLD as a name server for NWDI and if all of the following is true:

   ○   AS Java has reported data about it in the SLD

   ○   SLD has reported data about itself in the SLD

   choose *Enable this SLD as Name Server for NWDI*. For more information, see Setting Up Development Landscapes [external].

3. Choose *New Name Prefix*.

4. If you want to define a single name prefix for a certain category, select the *Single* indicator.

5. If you want to define name prefixes for the most commonly used categories, select the *Bundle* indicator.

6. If you have selected to define a single name prefix, from the *Name Category* dropdown box, select the name category you want (see Object Name Categories [page 46]).

7. In the *Name Prefix* field, enter a prefix that follows the syntax rules.

8. From the *Purpose* dropdown box, select the purpose you want.

   By default, the *Used by developer* option is selected. Such a standard prefix is used in the SAP NetWeaver Developer Studio that is connected to this name server. However, you can use the following options:

   • *Used in generator*

   A name prefix that is reserved for generator programs. The prefix is locked for use in the SAP NetWeaver Developer Studio.

   • *Used everywhere*

   A local name prefix for prototyping. All name servers are allowed to allocate names in this namespace. Therefore, name conflicts can occur in this namespace.

- *Used by other name server*

  For example, a name prefix for Business One development. No names are reserved on this name server; only name prefixes have been created.

  - *Used nowhere*

  A locked name prefix that is reserved for future use.

9. Optionally, in the *Owner* field, enter the e-mail address of the owner of the namespace and, in the *Description* field, enter a description of the namespace.

10. Choose *Define*.

⬇ Reserving Names

Use

You can use this function if you want to reserve a single name on a particular name server. This task is usually performed by the SAP NetWeaver Developer Studio. However, for developments without an SAP NetWeaver Developer Studio, you can use the *Name Reservation* function of the SLD.

Prerequisites

You have a user assigned to the `LcrAdministrator` role.

Procedure

1. Choose *Home → Name Reservation*.

2. If you wan to enable the SLD as a name server for NWDI and if all of the following is true:

   - AS Java has reported data about it in the SLD

   - SLD has reported data about itself in the SLD

   choose *Enable this SLD as Name Server for NWDI*.

   More information: <u>Setting Up Development Landscapes [external]</u>

3. Choose the *Names* tab.

4. Choose *New Name*.

5. From the *Name Category* dropdown box, select the object name category for which you want to reserve a name.

   More information: <u>Object Name Categories [page 46]</u>

6. From the *Name Prefix* dropdown box, select the prefix for which you want to reserve a name.

7. In the *Name* field, enter the name you want to reserve following the syntax rules.

8. In the *Caption* field, enter a caption of the name.

9. Optionally, in the *Owner* field, enter the e-mail address of the owner of the namespace and, in the *Description* field, enter a description of the namespace.

10. Choose *Reserve*.

Removing Name Prefixes and Names

Use

> You can remove name prefixes and names only if they have not been used in a product that is released.

Prerequisites

You have a user assigned to the `LcrAdministrator` role.

Procedure

1. Choose *Home → Name Reservation*.
2. Choose either the *Name Prefixes* or the *Names* tab.
3. From the *Name Category* dropdown box, select the object name category you want.
4. Select the name prefix or name you want to remove.
5. Choose *Remove*.

Working with Data on CIM Level

Use

You can work directly with CIM instance and CIM classes if you prefer the more direct approach.

More information: Common Information Model [page 8]

Procedure

Working with CIM Instances

1. Choose *Home → CIM Instances*.
2. Choose a subset and a class.
3. Modify them as necessary.

Working with CIM Classes

1. Choose Home → *CIM Classes*.
2. Choose a type of items on the left.
3. Choose an item.

Administrating the SLD

Use

The initially configured SLD is ready to provide its services to the systems that are connected to it. To better suit the SLD functionality to the needs of the system, you might need to:

- Change the initial SLD configuration

- Regularly perform tasks that are vital for the consistent state of your SLD and the information it contains

Changing the SLD Configuration

If you need to alter the SLD configuration after an update or a patch, use the SLD Administration page.

More information:

Performing Regular Administrative Tasks

During the SLD functioning, you need to ensure that everything runs smoothly. The only way is by regularly checking the system.

More Information:

Changing the SLD Configuration

Use

The SLD configuration is already included in the standard AS Java installation and configuration procedure. However, in some cases, the SLD configuration has to be changed. This section describes where to set the essential parameters to set up or reconfigure an SLD.

For information about general SLD configuration topics, see the *Planning Guide - System Landscape Directory* at **http://www.sdn.sap.com/irj/sdn/nw-sld**.

Procedure

Specify the Groups and Users That Can Operate in the SLD

More information:

Specify Where to Persist the SLD Information and Who Can Alter It

More information:

Fine Tuning the SLD Server

More information:

Initial Import of the SAP CIM model or the CR Content

More information:

Configuring the Channel for Receiving SLD Reports

To receive data that is automatically reported and sent by the SLD data suppliers running on individual systems, you have to configure and start the SLD bridge. The SLD bridge converts the system data sent by the SLD data suppliers to the SLD server into a CIM-compliant format.

More information:

Checking Your Steps

More information:

Post-Installation Checklist

Purpose

Use the table below as a checklist for the configuration steps you have to perform after you have installed AS Java.

All necessary configuration phases on the SLD server side are listed in the table. Use the links to the general descriptions of the actions and to any additional information to help you perform the actions. This prevents you missing important information.

Process Flow

Activities on the SLD Server Side

✔	Activity
☐	**Configuration of SLD security – Assigning SLD security roles and actions to users and user groups:** 1. Open the SLD home page `http://<host>:<port>/sld` in a Web browser. 2. Navigate to *Administration → Settings to apply the standard SLD role mapping*. More information: Configuring SLD Security Roles [page 53]
☐	**Configuring the server parameters – Starting the SLD server:** If you have already performed the initial configuration of SLD as part of the AS Java system installation, ignore the steps below. 1. Open the SLD home page `http://<host>:<port>/sld` in a web browser. 2. Choose *Administration → Settings → Server Settings*. 3. Enter a name for the *Object Server*. Preferably use a prefix that has been reserved on SAP Service Marketplace as an *Object Server* name. 4. Start the SLD server. More information: Configuring Server Parameters [page 55]
☐	**Performing initial data import:** If you have already performed the initial configuration of SLD as part of the AS Java system installation, ignore the steps below. 1. On the SLD home page, choose *Administration → Import*. 2. Press *Import CR Content*. More information: Performing Initial Data Import [page 57]

	Configuring the SLD Bridge:
☐	💡 If you have already performed the initial configuration of SLD as part of the AS Java system installation, ignore the steps below. 💡 You need to change these settings only if the RFC server configuration needs to be changed. To check the current settings on the SLD home page, choose *Administration → Details → Data Suppliers*. 1. On the SLD home page, choose *Administration → Settings* and from the *Section* field choose *datasupplier*. 2. Maintain the gateway host and a name of the gateway service. The SAP NetWeaver AS Java standalone includes a gateway on the Central Services Instance which is used if no gateway is configured explicitly. More information: Configuring the SLD Bridge [page 58]

Configuring SLD Security Roles

Use

The SLD functions are protected from unauthorized access. There are several AS Java security roles and User Management Engine (UME) actions that are assigned to different SLD functions. Before you can use SLD, you have to map these roles and actions to individual users or user groups.

The following table lists the SLD security roles together with their recommended SLD user group and UME role.

AS Java security Role	UME Action	User Group/UME Role
LcrUser	`LcrUser`	SAP_SLD_GUEST
LcrSupport	`LcrSupport`	SAP_SLD_SUPPORT
DataSupplierLD	No corresponding UME action.	SAP_SLD_DATA_SUPPLIER
LcrContentSync	`LcrContentSync`	SAP_SLD_CONTENT_SYNC
LcrInstanceWriterNR	`LcrInstanceWriterNR`	SAP_SLD_DEVELOPER
LcrInstanceWriterLD	`LcrInstanceWriterLD`	SAP_SLD_CONFIGURATOR
LcrInstanceWriterCR	`LcrInstanceWriterCR`	SAP_SLD_ORGANIZER
LcrInstanceWriterAll	`LcrInstanceWriterAll`	SAP_SLD_ORGANIZER
LcrClassWriter	`LcrClassWriter`	SAP_SLD_ADMINISTRATOR
LcrAdministrator	`LcrAdministrator`	SAP_SLD_ADMINISTRATOR

💡

We recommend that you use user groups and map them to the appropriate UME roles and actions instead of assigning them to individual users. Users that belong to a particular group receive all permissions that are granted to the group.

We recommend that you use the following user groups that correspond to the following identically named UME roles:

UME Role / User Group	Permissions
SAP_SLD_GUEST	Read access to SLD data
SAP_SLD_SUPPORT	Read-only access to all SLD data and UI, including the Administration area (used for SAP support)
SAP_SLD_CONFIGURATOR	Create, modify and delete CIM instances of the Landscape Description and Name Reservation subsets (includes read permissions).
SAP_SLD_CONTENT_SYNC	Synchronize SLD content changes with other SLD CIM namespaces. Includes read access to the SLD UI.
SAP_SLD_DATA_SUPPLIER	Create, modify and delete CIM instances of the Landscape Description subset as a data supplier without access to the SLD UI.
SAP_SLD_DEVELOPER	Create, modify and delete CIM instances of the Name Reservation subset (includes read permissions).
SAP_SLD_ORGANIZER	Create, modify and delete all types of CIM instances (includes read permissions).
SAP_SLD_ADMINISTRATOR	Administrative tasks (includes all other roles)

If the UME is used with an ABAP-based system as the back-end user storage, these groups already exist. ABAP user roles appear as user groups on the AS Java side. SAP NetWeaver Application Server ABAP (AS ABAP) contains these default user roles.

If you are allowed to create user groups as a local AS Java administrator, the SLD user groups are created by the standard SLD configuration described below.

If your LDAP user store is configured in a way that no user groups can be created by the local UME, you have to create the user groups listed above first.

If you want to set up SLD security for test purposes, you can simply use an AS Java administrative user which also has administrative permissions for SLD by default.

Creating Standard SLD Security Roles - Automatically

1. Enter the URL of the application using the following pattern:
   `http://<host>:<port>/sld` (details are described below).

   Log on as an administrator for AS Java and an SLD administrator.

2. Choose *Administration → Settings* and choose *Perform Role Mapping*.

   The system tries to create or complete the user group list and the mappings described above. Depending on the settings of your user store, the group creation may fail. In this case, you have to create the groups manually and press the button again.

3. Assign users to the groups as needed.

Creating SLD Security Roles - Manually

1. In your Web browser, enter the URL of the Identity Management using the following pattern: `http://<host>:<port>/useradmin`.

2. Create user groups and UME roles of your choice and assign each UME role to the appropriate user group or groups as well as to the UME action or actions.

> Additionally, assign the GUEST UME role to each of these user groups.

3. Assign users to the user groups.

For more information about managing security roles, see Managing Users, Groups, and Roles [external].

Configuring Server Parameters

Use

Use this procedure to configure the SLD server parameters.

> If you have already performed the initial configuration of SLD as part of the AS Java system installation, use this procedure only to adapt the configuration.

Prerequisites

You have a user assigned to the `LcrAdministrator` role.

Procedure

1. Choose *Administration* → *Settings*.

2. From the *Section* dropdown box, select *Server Settings*.

3. Enter the name of the object server.

    > Make sure that the object server name is unique within your system landscape or even globally. The value of the parameter must be equal to an ABAP namespace (without the enclosing slashes) that is reserved on SAP Service Marketplace at `http://service.sap.com/namespaces`. If you do not have an ABAP namespace, enter the name of the host where your SLD is running.
    >
    > If SLD acts as:
    >
    > - A landscape directory in your system landscape, the name of the host where your SLD is running is sufficient.
    >
    > - A name server for the SAP NetWeaver Development Infrastructure, the name must be reserved on SAP Service Marketplace. For more information, see SAP Notes 105132 and 710315.

4. Specify a *Working Directory* for the application configuration.

5. Specify writing authorization:

    a. *None* – there are no write restrictions (normal mode).

    b. *Single User* – only the current user has write access (exclusive write mode).

    c. *Read-only* – no user have write access (read-only access).

6. Choose *Save*.

Fine Tuning the SLD Server

Use

You can fine tune the SLD server by changing the server parameters. The configuration of the server parameters is called a system profile. When you start the SLD server for the first time, the system profile is automatically uploaded. This system profile contains the default server parameters. However, you can:

- Change the system profile online in the SLD UI

- Download the system profile, change it offline, and upload it back to the SLD server.

    The default system profile is located in the `sldprofile.xml` file in the following directory (on Microsoft Windows):
    `<Drive>:\usr\sap\<SID>\SYS\global\sld`. You can upload the configuration file to the SLD server directly as an XML or a ZIP file.

    The changes to the system profile take effect only after you restart the SLD server.

Prerequisites

You have a user assigned to the `LcrAdministrator` role.

Procedure

1. Choose *Administration* → *Settings*.

Changing a System Profile

To change the system profile online:

1. From the *Section* dropdown menu choose *Server Settings*.

2. Change the server parameters and choose *Save*.

3. If you want to discard the changes you have made, choose *Reset to Defaults*.

Downloading a System Profile

To change the system profile offline:

1. Choose *Download* → *Download Profile*.

2. Choose *Save*, enter a file name and browse to the directory where you want to save the configuration file.

    Now you can change the server parameters offline in the configuration file.

Uploading a System Profile

1. Choose *Upload*.

2. Browse to the configuration file. You can select *Overwrite* if you want to replace the existing configuration.

3. Choose *Upload*.

Adding a New Profile Parameter

You can introduce new profile parameters to extend the profile with parameters described in a SAP note.

1. Choose *Add Parameter*.

2. Specify values.

3. Choose *Save*.

Performing Initial Data Import

Use

The SLD server implements the Common Information Model (CIM) of the Distributed Management Task Force (see **www.dmtf.org**). The SAP CIM model and SAP component information (`CR_Content.zip`) are automatically imported when the SLD is started for the first time during installation. Use this procedure if you want to import a different CIM model or to import the CR content files after the installation.

The CIM model and CR content files are located in the `<Drive>:\usr\sap\<SID>\SYS\global\sld\model` directory.

> Since the `CR_Content.zip` file contains all SAP components that are available, the content of this file increases over time to include information about new components, such as new releases and support packages. You have to update the content in the SLD from time to time. You can download the most up-to-date files on SAP Service Marketplace at **http://service.sap.com**. For more information, see SAP Note 669669.

> If you have already performed the initial configuration of SLD as part of the AS Java system installation, use this procedure only to adapt the configuration.

Prerequisites

- The CIM model and CR content files must be either XML files or compressed XML files in ZIP format.

Procedure

1. Choose *Administration* → *Import*.

   The *Import* screen appears.

2. To import the following file (Microsoft Windows):
   `<Drive>:\usr\sap\<SID>\SYS\global\sld\model\CR_Content.zip`, choose *Import CR Content*, and then choose *Continue Import*.

3. To import a different file, choose *Browse*, navigate to the file you want to import, and then choose *Import Selected File*.

   > Objects that already exist in the system are automatically overwritten.

   > If the import has been interrupted due to insufficient memory (for instance, the Java VM runs out of memory), you have to increase the heap size of the Java VM and restart the AS Java. To have the complete data, you have to import the `CR_Content.zip` file again. You can ignore the warning about a non-fitting content update.

4. If you do not want to import the content into the current namespace, you can change the namespace.

Result

After you have triggered the import, the *Administration* screen appears. The status bar shows the status of the import.

Configuring the SLD Bridge

Use

To receive data that is automatically reported and sent by the SLD data suppliers running on individual systems, you have to configure the SLD bridge. The SLD bridge converts the system data sent by the SLD data suppliers to the SLD server into a CIM-compliant format.

The data between the SLD data suppliers of ABAP-based systems and the SLD bridge is exchanged by means of an RFC connection. Therefore, you have to configure a gateway service.

We recommend that you use a local gateway.

The configuration of the SLD bridge can also be performed during installation. Use this procedure if you want to change the configuration.

If you have already performed the initial configuration of SLD as part of the AS Java system installation, use this procedure only to adapt the configuration.

Procedure

1. Choose *Administration* → *Data Suppliers*.

2. If you want the SLD data supplier bridge to forward system data that is received from data suppliers to a namespace of this SLD server, check the indicator in the *Update* column for the respective namespace.

3. You can also add URLs of an external SLD (called SLD bridge forwarding):

    a. Choose *Add*.

    b. Set details.

    c. Choose *Add*.

4. Choose *Administration* → *Settings.*

5. From the *Section* dropdown box, select *Datasupplier*.

6. If you want to use a remote gateway, enter gateway host and gateway service.

    The changes to the gateway service take effect only after you restart the SLD server.

    One gateway server must be linked only to one SLD server as a data receiver. Sharing one gateway server for multiple SLD servers leads to errors.

    Make sure that you have set up correctly the SLD data suppliers in the systems that have to report system data automatically

7. If you do not want to receive data from ABAP systems in your SLD, set the parameter *StartRfcServer* to *false*.

8. Choose *Save*.

Performing Regular Administrative Tasks

Use

You must perform certain administrative tasks regularly to make sure that the SLD is working properly.

Prerequisites

You have a user assigned to the LcrAdministrator role.

Process Flow

Updating the Software Catalog

More information: Updating the Software Catalog [page 60]

Exporting and Backing Up Your SLD Information

All SLD data is stored in the Java EE database and is therefore automatically backed up during the database backup. However, from time to time, especially before installing updates, you may want to create SLD-specific data backups.

More information: Exporting and Backing Up Your SLD Information [page 61]

Synchronizing the SLD Content

If you have a landscape with several SLDs, you have to make sure that their content is correctly synchronized.

More information: Synchronizing SLD Content [page 67]

Viewing Logs

During a system process, the SLD server logs relevant information for monitoring purposes.

More information: Viewing Logs [page 73]

Cleaning Up Data

The data supplier programs report data about the systems in your landscape to the SLD. When a system no longer exists in your landscape, you have to clean up this data manually in the SLD.

More information: Cleaning Up Data [page 75]

Creating System Messages

You can create an informational system message, which will be seen by every user who logs on to the SLD.

More information: Creating System Messages [page 76]

Maintaining SLD Content at CIM Instance Level

More information: Maintaining SLD Content at CIM Instance Level [page 76]

Using Namespaces to Test Data

More information: Using Namespaces to Test Data [page 78]

Result

More information: System Landscape Directory Security Guide [external]

Updating the Software Catalog

Use

The SLD contains a Software Catalog of all installable SAP products and software components. The Software Catalog includes information about support packages and dependencies between the products and software components. You have to make sure that this content is always up-to-date.

You can find the updates of the Software Catalog on SAP Service Marketplace at `http://service.sap.com/swdc` → *SAP Software Distribution Center* → *Download* → *Support Packages and Patches* → *Entry by Application Group* → *SAP Technology Components* → *SAP CR CONTENT.*

The updates of the Software Catalog are provided as deltas to the previous version. Therefore, you do not need to download and install the complete Software Catalog every time.

The updates of the Software Catalog are provided as ZIP files. Note that an import can take a long time (up to an hour), depending on the data size. For more information, see SAP Note 669669.

Prerequisites

- You have downloaded the ZIP file(s) from SAP Service Marketplace to a directory of your choice.

Procedure

1. Choose *Administration* → *Import*.

2. If you want to perform an initial import of the content that is delivered with an installation or an upgrade, select the *From Server* radio button.

3. If you want to import content that you have downloaded on SAP Service Marketplace or if you want to import a backup, choose *Browse* and browse to the ZIP file you want to import.

4. Choose *Import*.

   The CIM standard is based on the object-oriented approach. When you import instances from multiple files, make sure that you import them in the correct sequence since this has an effect on the inheritance hierarchy.

   At this step the system checks the type of the import, and whether the import fits into the current content.

   - If the import does not fit into the current content, a warning is displayed. Choose *Import Anyway* only if you are sure that you have selected the correct data.

   - If the import type and content is correct, the information about the import is displayed. To continue with the import, choose *Continue Import*.

Handling Import Errors

The import ignores a certain number of errors related to conflicts between the objects to be imported and the given SLD content. The profile parameter to define this limit is `ImportMaxErrorCount` with a default value of 100. You can set the value -1 for an infinite

limit or the value 0 for the zero tolerance level. Because the import is performed in chunks, the displayed error count at the end may be somewhat higher than the limit. But this does not matter, the import aborts at the end of the chunk where the limit is reached and performs a rollback for this chunk.

All non content related errors, for example an SQL runtime exception, cause an immediate abort of the import.

An aborted import has to be restarted until it finishes (with or without errors). No other data must be imported until then. A restart offers the options:

- *Continue Import* – use it as the default option, because repeating successfully processed operations could lead to artificial errors.

- *Restart From Beginning* - it should be selected carefully in the case you need to repeat the object import of content related errors. This is reasonable only if the conflicting objects in the SLD have been moved or removed in the meanwhile.

If the import has been finished before (with or without errors), you are warned not to repeat it at all. Repeating successfully applied changes could result in artificial errors. Thus, the *Import Anyway* option is not reasonable here, as almost in all cases.

> For safety reasons, a test import should be considered for non standard imports. Please refer to note 1045423 for the details.

Result

The import starts and the status of the import is displayed at the top of the screen.

Exporting and Backing Up Your SLD Information

Use

All SLD data is stored in the AS Java database and is therefore automatically backed up during the database backup. However, there are scenarios where it is useful to export SLD-specific data. SLD provides different mechanisms for these purposes. You can:

- produce unversioned backups

- perform versioned exports of specific export lines

- selectively export individual objects stored in SLD.

The following table gives an overview of typical scenarios. More details are explained in subsequent chapters below.

Scenario or Use Case	Proposed Procedure
Backup of all SLD data (before installing updates or patches, before updating the Software Catalog, etc.)	Unversioned backup of *All CIM Instances*.
SLD system copy: Setup of a new SLD system with initial content from an existing system.	Import an unversioned backup of *All CIM Instances* from the source system into the new system.
Synchronization of all content: Regular transports of all changes between different SLD systems.	Versioned exports of *All CIM Instances*. Alternative: SLD Content Synchronization [page 68]

Synchronization of partial content: Regular transports of content changes between different SLD systems for parts of SLD data.	Versioned exports of required export line, for example Landscape Directory, to transport system data only. Alternative: Automatic forwarding of system registrations to other SLDs.
Selective copy of SLD data	Example: Export a business system from one SLD and import it into another one.

Export Lines of SLD Data

Depending on the subset of SLD data you want to export or back up, you can select from the following export lines defined in SLD:

- *All CIM Instances* - This line encompasses all data stored in SLD (see scenarios 1, 2, and 3).

- *Landscape Directory* is used for landscape data only. This includes technical systems, business systems, etc. (See scenario *Synchronization of partial content*)

- *Component Repository* - This line includes all Component Repository data (software components, products, etc.). This means it includes both SAP Component Repository content as well as any of your own products and software components you may have defined.

- *Name Reservation* is used for name reservation data only.

In general, different export lines are independent of each other. However there are some things to consider when working with different lines:

- Data exported or backed up using the *Landscape Directory* line depends on Component Repository content present at the time of export. Therefore you should take care when importing such data into another SLD, as the target SLD should contain the same or a newer version of the Component Repository content to avoid import errors due to missing component repository objects.

- Because exports of *All CIM Instances* contain all data, this line is incompatible with all other lines. This means that you should not import data of subset *All CIM Instances* into a target system that already contains imports of other data (or vice versa). During the import checks you will get an appropriate warning and should cancel the import.

- In particular, SAP Component Repository content must not be imported into target systems of export lines that already include this content. This means that any system you use for versioned imports of *All CIM Instances* or *Component Repository* must not be updated directly with SAP Component Repository content.

Unversioned Backups

Unversioned backups contain a snapshot of the current state of the SLD. You can use them to copy an SLD system or to restore it after a crash. Backups themselves are not stored in the database; they are saved by the user after the backup is finished. In contrast to versioned exports they cannot be used to transport changes regularly.

Various options for backups are available:

- *Back Up All Instances*: Produce a snapshot of all data currently stored in SLD. This is the most commonly used option for backups (see scenario 1).

- *Back Up Model*: Back up the SLD data model. Together with Back Up All Instances this can be used to import a snapshot of both data model and instance data into a new system.

- *Back Up Instances by Export Line*: You can use the same export lines that are available for versioned exports to produce an unversioned backup of data belonging to

each export line. This option is used rarely, but may be useful if you want to back up data of a specific line only, for example Landscape Directory.

- *Back Up Instances by Class*: You can back up instances of selected classes of the SLD data model. This option is used rarely.

To ensure consistency, the backup namespace is locked against concurrent changes while a backup is running.

More information on how to perform a backup: Producing Unversioned Backups

Versioned Exports

Versioned exports are used to transport updates of data from one SLD to another regularly. They contain information about additions, modifications, and deletions. Exports are directly related to a specific export line and provide a history of content versions for the selected export line. The history of every line starts with an initial full export containing all data for this line at the time of export. Later incremental exports only contain changes, including additions and removals that occurred since the last export.

Next shows a typical scenario where all changes of a source SLD are transported to a target SLD (see scenario 3 above). In this case SAP CR content must be imported into the source SLD only, not into the target SLD. Initially the target SLD should not contain any SAP CR content or the same version of SAP CR content that is delivered with the initial full export. Otherwise there will be content mismatches that will lead to import errors.
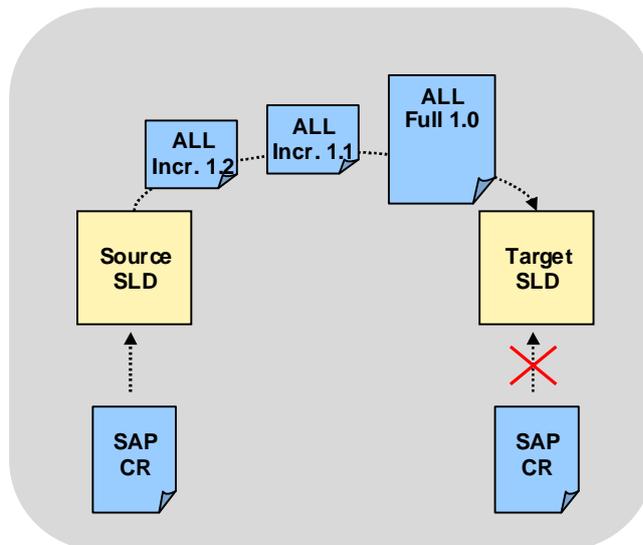
An import into a target system or namespace always starts with a full import. Afterwards updates are applied by incremental imports in the same order as they were produced. To facilitate transports into a new system without having to import a long history of versioned exports, new full exports can be produced at any point during the version history. A full export of this type is always accompanied by a corresponding incremental export containing the same end state. The initial full export is the only export that has no related incremental export.

> ⚠️
>
> You never import more than one full export of a given export line into a target system. Once the full import is finished, only incremental imports are used. Later full exports are only used for new systems that have not yet received any previous data of the corresponding export line.

All versioned exports are stored in the database and can be downloaded whenever necessary at a later time by choosing *Administration → Export → Administrate Incremental Exports.*

To simplify the import of multiple incremental exports it is possible to aggregate them into a single file. An aggregate export contains multiple consecutive incremental export versions (incremental aggregate). The first version in an aggregate export may also be a full export (full aggregate).

For example, consider an export system where you have performed an initial full export 1.0 and incremental exports 1.1 through 1.5. Now you can produce an aggregate export containing versions 1.0 through 1.5. This export can be used as an initial import into a new system because it starts with a full export. After the import the new system will contain version 1.5. Alternatively, this aggregate can be used to update any target system to version 1.5 that contains any previous version 1.0 through 1.4.

Note that aggregate exports starting with a full export are only supported by version 7.10 and above. Imports into older SLD releases are able to process incremental aggregate exports only.

Updates of the Software Catalog are created as exports from an SAP master system. The files published on SAP Service Marketplace are full exports providing initial content for setting up new systems and incremental aggregate exports containing updates.

To produce incremental exports, information from the SLD change log is used. Therefore change log entries must not be deleted when they may still be required for future incremental exports. You will be warned about this situation when you try to remove required change log entries using the SLD UI.

You should never produce a full versioned export when you do not intend to start regular transports to another system. Use an unversioned backup instead. Otherwise you are forced to keep any change log entries starting from the time of the full export.

To ensure consistency, the export namespace is locked against concurrent changes while an export is running.

More information on how to perform a versioned export: Producing Versioned Exports

Selective Partial Exports

At different locations in the SLD Web user interface you can export individual objects.

You can use the individual export option on the Landscape and Software Catalog pages for selective distribution of technical or business systems and products or software components.

These exports typically contain associations to objects not being exported themselves (called "external references"). For example, a business system export does not include the underlying technical system. Thus external references cannot be imported if referenced objects are missing in the target system. You may repair this loss of data by transporting or creating missing objects and repeating the import. In the example mentioned before, you have to import the technical system first if it does not exist in the target SLD yet.

⚠️

Note that manually creating the same software component or the same PI business system in another SLD will create a different GUID. As a result, these objects are not identical even though their names are the same. Hence, use the individual export option to distribute this data. Do not enter it manually in multiple systems.

Synchronizing Content of SLD Server Instances

Versioned exports can be used to synchronize content of different SLD instances.

In the most typical scenario export line *All Instances* is used for this purpose (see figure above). Because this export line contains all data, the export system becomes the exclusive supplier for following systems. Therefore SAP Component Repository content must only be imported into the export system. It is distributed from there along with any other data.

An example of this is the use of separate production and non-production SLD instances. Typically you will need to transport changes from the non-production SLD to the production SLD, not vice versa. As a result all systems are displayed in the production SLD, whereas the non-production SLD does not contain data about productive systems.

If you want to include an additional SLD instance in the transport route that already contains Component Repository content independently of the exclusive supplier, you must first update this SLD instance to the same Component Repository content version as the one installed on the supplier system.

Note that the synchronization of all data of different SLD systems can also be achieved using SLD Content Synchronization without the need for regular manual exports and imports. In contrast to versioned exports, content synchronization also supports bidirectional synchronization.

There are other ways of synchronizing SLD instances, but these are limited to specific applications. For example, the SAP Component Repository content can be updated separately in different systems, while only Landscape Directory data is distributed by means of exports and imports. This is a useful configuration if no customer development is performed. Alternatively, you may want to distribute landscape data by forwarding reporting systems to secondary SLD instances automatically. This can be configured on the Data Supplier pages.

Result

You can import both exports and backups into another SLD system.

More information:

For more information about import error handling, see SAP Note 1045423.

Producing Unversioned Backups

Procedure

Choose *Administration → Export → Backup*.

Backing Up All CIM Instances

1. Choose *All Instances*.

2. Choose *Backup*.

The status of the backup is displayed. When the backup is complete, the *Download* screen appears.

3. Use the primary mouse button to click the name of the archive.

4. Choose *Save* to save the file to a directory of your choice.

Backing Up the CIM Data Model

1. Choose *Model*.

2. Choose *Backup*.

3. Save the file to a directory of your choice.

Backing Up Instances by Export Line

1. Choose *Instances by Export Line*.

2. Choose *Backup*.

3. Save the file to a directory of your choice.

Backing Up Instances of Selected Classes

1. Choose *Instances by Class*.

You can choose *Include subclasses* and *Include references* in the list.

2. From the list, select the entries you want by holding down the CTRL key and selecting the indicators.

3. Choose *Backup*.

The status of the backup is displayed. When the backup is complete, the *Download* screen appears.

4. Click with the primary mouse button over the name of the archive.

5. Choose *Save* to save the file to a directory of your choice.

Producing Versioned Exports

Procedure

1. Choose *Administration* → *Export*.

   The Export screen appears. The namespace that will be exported is displayed in the header bar (if you work with multiple namespaces).

2. You have the following choices:

   o *Incremental Export* – perform full or incremental exports for an export line of your choice.

   o *Administrate Incremental Exports* – view the history of previous exports, download individual exports, or aggregate multiple exports so that you can easily import them into another system.

Performing a Full or Incremental Export

1. Choose the *Incremental Export* tab.

2. From the *Export Line* dropdown box, select the export line of your choice.

3. Choose the type of export you want to perform: *Incremental* or *Full*.

If this is the first export for the selected export line, you have to perform a full export. For all following exports, you can choose whether to perform an incremental or a full export.

4. Choose *Export*.

The status of the export is displayed. When the export is complete, it is listed in the Result section. Click on the file name link to download the result.

The system exports the data of the export line you have selected. This export is also saved to the database and you can use it later. You can import the data into other systems of your system landscape, for example.

More information: Updating the Software Catalog

Administrating Incremental Exports

1. To manage your exports, choose *Administrate Incremental Exports*.

A list of your previous exports is displayed. You can aggregate multiple exports and import them together as a single file into another system.

You must import exports with versioning into the target system in the same sequence in which they were produced.

2. Choose an option from the *Export Line* dropdown box.

You can also use a filter to make the list more specific.

3. To aggregate several exports into one:

a. Select the required exports. It is sufficient to mark the first and last version of the exports you want to include in your aggregate.

b. Choose *Aggregate*.

c. Download the resulting aggregate file.

4. To remove an unnecessary incremental export:

a. Select the corresponding export.

b. Choose *Remove*.

Only remove exports that are not required again. You must not remove exports that have already been imported into another system. Otherwise, the history of this export is lost, which will result in import errors the next time you perform an import.

Synchronizing SLD Content

Use

In your system landscape, you might face the requirement to run more than one SLD system and to transport (or synchronize) content between these different SLD systems. With SLD content synchronization you can define the required transport topology and perform automated synchronization of all content changes without regular manual operations.

More information: SLD Content Synchronization [page 68]

Procedure

Adding a new content synchronization connection

More information:

Maintaining content synchronization connections

More information:

SLD Content Synchronization

Use

In your system landscape, you might face the requirement to run more than one SLD system and to transport (or synchronize) content between these different SLD systems. In the past, this could only be done by

- SLD bridge forwarding: You can configure an SLD to transport data received from SLD data suppliers automatically to other SLD systems – manually entered data can not be transported automatically with SLD bridge forwarding.

- Incremental exports and imports (a process with which you can transport all SLD data, but you have to perform it manually).

With SLD content synchronization, it is now possible to define the required transport topology and perform automated synchronization of all content changes without regular manual operations.

The following figure shows an example for a global or widely distributed IT landscape. More than a single central SLD may be required because it is expected that an SLD is available locally but that it provides a more than a local view. This and other examples can be found in the *Planning Guide – System Landscape Directory* at
http://www.sdn.sap.com/irj/sdn/nw-sld.
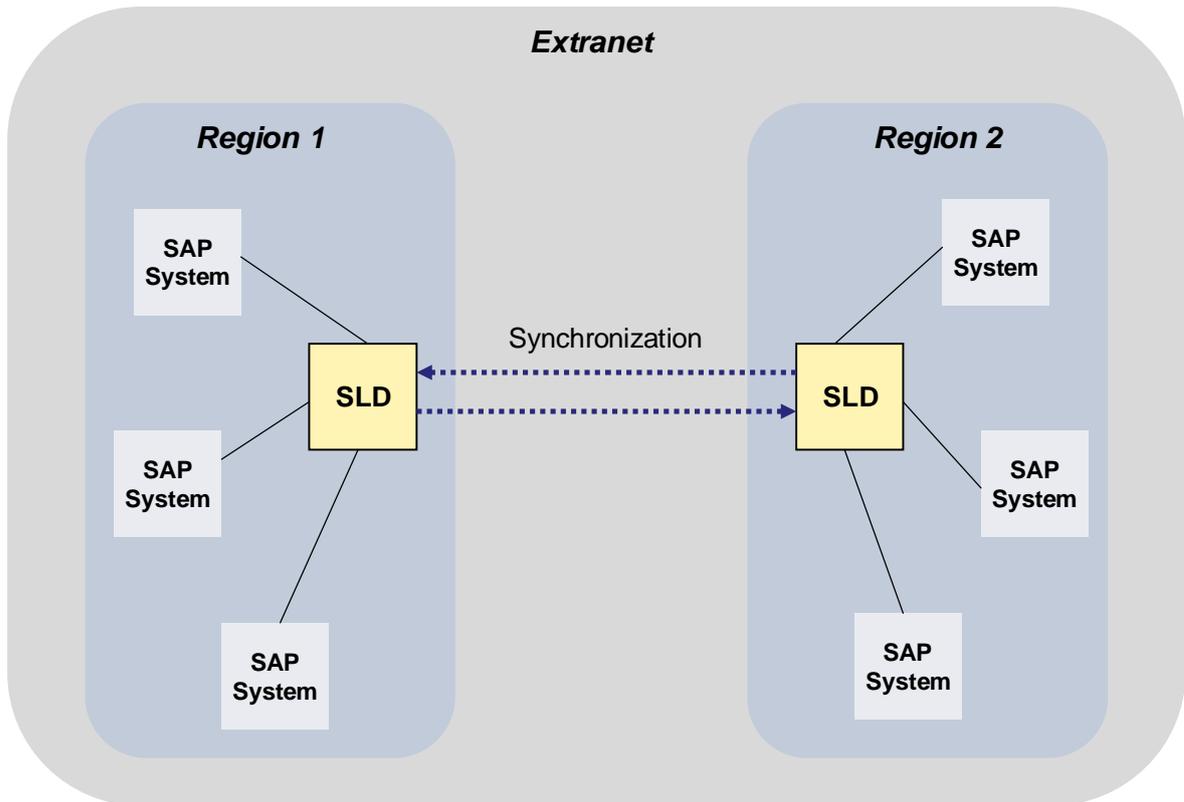
Content synchronization is configured on the level of CIM namespaces. This means changes in one CIM namespace are transported to another CIM namespace. Both namespaces can reside in the same SLD system (local synchronization), but will typically reside in two different SLD systems (remote synchronization). For simplicity reasons, this documentation only describes remote synchronization between different SLD systems. However, all considerations equally apply for local synchronization.

Properties and Features

SLD content synchronization has the following properties and features:

- Content synchronization can be configured for unidirectional or bidirectional use. Unidirectional synchronization is used for transport topologies where changes are only transported from the source to the target system. Bidirectional synchronization is employed when changes need to be transported mutually between systems.

- Changes are synchronized using a "publish and subscribe" mechanism. Listener systems can either periodically poll for changes or they can be notified of changes, or both.

- Changes are recorded synchronously in the SLD system where they are applied (in the SLD change log).

- The content synchronization process is performed asynchronously. Thus, participating SLD systems can be unreachable temporarily (for example, due to network or system downtime) without disrupting the overall synchronization process.

- Reconciliation of conflicts (due to conflicting parallel changes) is done automatically, no user interaction is required. For this purpose, you need to assign a rank to each participating SLD namespace during configuration. This rank defines the outcome of reconciliation: Changes of a higher ranking namespace override conflicting changes of a lower ranking namespace. This implies that a change in a lower ranking namespace is an "optimistic write".

- Content synchronization is only possible for the whole content of participating namespaces; filtering of namespace content to be synchronized is not supported.

- All communication related to content synchronization between SLD systems is based on HTTP or HTTPS. This simplifies configuration of firewalls, for example.

- The SLD content synchronization is fully supported by AS Java 7.1 and above. Additionally, an SLD of version 7.0 SP12 and above can be used as source system for unidirectional synchronization only. Systems with version 7.0 do not support the notification of listener systems.

Usage Rules

To ensure correct synchronization, certain rules must be adhered to when defining a content synchronization topology:

- Each SLD system participating in a landscape must have a unique object server name to correctly identify the origin of changes. However, this is not specific to content synchronization, but is recommended for any SLD system.

- Each namespace included in a synchronization landscape should have a unique rank. Otherwise conflicts may not be resolved correctly because none of the conflicting changes has a higher rank, leading to an unsynchronized differing end state in different namespaces.

- In a complex landscape (with more than two synchronized namespaces or systems) the synchronization topology should adhere to the unique path principle. This means that a change in an origin system A must only have one transport path to any listener system B. Otherwise subsequent changes in A could travel over different paths to and be processed out of order by B, leading to an erroneous end state in B. The following figure shows an example for a violation of the unique path principle, where changes from A can travel to B either directly or via C. The bidirectional synchronization between A and B does not violate the unique path principle for the opposite direction from B to A.
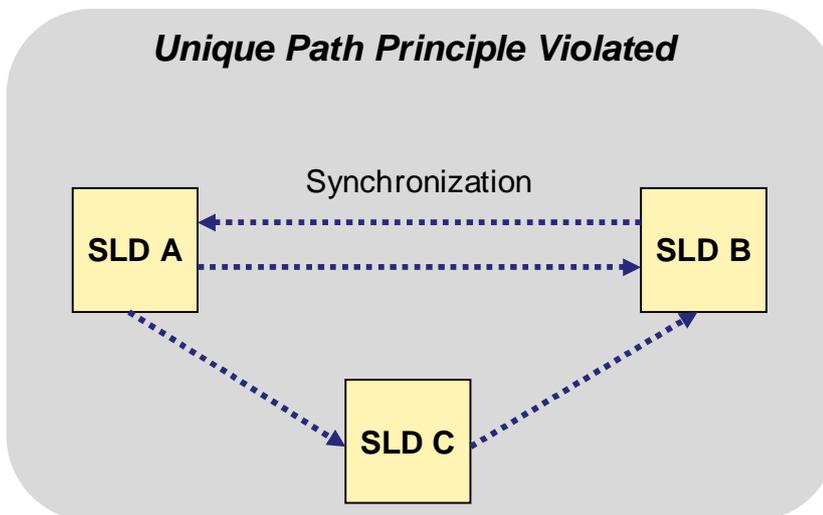
Once your content synchronization topology is defined and active, model or content updates only need to be applied to one specific SLD from where they will be forwarded to other participating SLD systems. Select the SLD system to update according to the following rules:

- If possible, updates should be applied to the SLD with highest rank. This ensures updates are accepted by all other SLD systems and avoids unnecessary overhead during propagation of these changes.

- For a unidirectional synchronization updates must be applied at the source of the unidirectional path to reach all participating systems. This is required for CIM model or

SAP CR content updates (see SAP note 669669). However, if updates are not meant to reach all participating SLD systems on purpose, they may of course be applied to a system anywhere in a unidirectional path.

Typical Scenarios

Various reasons can lead to the use of multiple SLD systems and requirements for content synchronization between them. These can be legal constraints, company rules, availability considerations, network constraints (such as firewalls) or the possibility to provide different views of SLD data (for example, to hide productive systems from the DEV landscape). As a result, some typical use cases for content synchronization are:

- Transport topology for a landscape like DEV – TEST – PROD.

- Managing complex landscapes distributed over multiple computer centers and/or optimized for decentralized use.

  More information: *Planning Guide – System Landscape Directory* at **http://www.sdn.sap.com/irj/sdn/nw-sld**.

- SLD migration:

  A new SLD system can be installed in parallel to an old one. Using content synchronization and host name virtualization (for example, by DNS aliasing), continuous availability can be reached for all software life-cycle tasks (such as applying patches, performing upgrades and hardware relocation).

- Namespace copy

  A second namespace with the same content can easily be set up by configuring an appropriate (most likely unidirectional) synchronization. This has some advantages over using backup or full and incremental exports. Other than a backup or an export, content synchronization does not lock the source namespace against concurrent changes. Later changes in the source namespace are easily transported to the copied namespace as long as desired until the synchronization is removed or deactivated.

Adding a New Content Synchronization Connection

Use

During this step, all existing data is synchronized with the source namespace.

Prerequisite

Both source and target namespace contain the same CIM model version.

This is only a requirement when setting up a new content synchronization. Later model updates in any participating SLD will automatically be transported according to your content synchronization topology.

Procedure

1. Choose *Administration → Content Synchronization*.

   If you want to set up a unidirectional synchronization, perform these steps on the target SLD. For a bi-directional synchronization these steps can be performed on any of both SLD systems.

2. Choose *Add* to start the wizard for adding a new content synchronization connection. You will be asked to provide information in the following steps:

a. General

Specify if you want to set up a local or remote synchronization, and if the synchronization should be uni- or bi-directional.

If you are setting up a local synchronization, then you do not need to specify source SLD.

Choose *Next*.

b. Source SLD

Specify all necessary information for obtaining changes from the source SLD (URL, user and password). The specified user must have role `LcrContentSync` (member of group `SAP_SLD_CONTENT_SYNC`).

Choose *Next*.

c. Source Namespace

Select the source namespace for changes from the list of available namespaces of the source system.

Also define the rank of the source namespace if it is still undefined (-1) or change it if desired.

Choose *Next*.

d. Local SLD

Specify all necessary information about the local SLD (URL, user and password). The specified user must also have role `LcrContentSync` (member of group `SAP_SLD_CONTENT_SYNC`). This information is required for two purposes:

- Upon changes, the source SLD will send a notification using this data.

- For a bi-directional synchronization, the source SLD will use this information for the opposite direction.

Also define the rank of the local target namespace if it is still undefined (-1) or change it if desired.

Choose *Next*.

e. Summary

A summary of all configuration data is shown. You can either choose Back to apply changes if necessary or choose Finish to add and activate this new configuration.

Result

The new synchronization just configured will be started immediately. (By default any new synchronization is active and uses notifications.) As shown by the status information in the user interface, a full sync will be performed first.

Once this step is finished, content synchronization will continue in incremental mode, synchronizing any changes that occur in the source namespace automatically.

Maintaining Content Synchronization Connections

Prerequisites

SLD content synchronization is already created.

Procedure

Open *Administration → Content Synchronization*.

Activating a Synchronization or Notification

1. Check *Active* for the *Sync Subscriptions* you want to activate.

   Synchronization of changes will continued from the point in time it was last deactivated. No new full sync is required.

2. Check *Active* for the *Sync Notifications* you want to activate.

   From now on any changes in the local namespace will again sent a notification to the target system.

Temporarily Deactivating a Synchronization or Notification

1. Uncheck *Active* for the *Sync Subscriptions* you want to deactivate.

   If you deactivate a subscription for an extended period of time, you may also want to deactivate the corresponding notification in the source system to avoid unnecessary network traffic due to irrelevant notifications.

2. Uncheck *Active* for the *Sync Notifications* you want to deactivate.

   If the corresponding subscription in the target system is still active, this means that the target system will no longer be notified about changes and only uses periodic polling to retrieve changes.

Repeating a Complete Content Synchronization

This will repeat the synchronization as if it was newly created. This may be necessary in case of unrecoverable errors but should be avoided otherwise because pending changes not processed so far will be discarded.

This step is time consuming.

To repeat a complete Content Synchronization, in the *Action* column choose *Full Sync* for the *Sync Subscription* you want to repeat.

Removing a Synchronization or Notification Permanently

1. Select any *Sync Subscription* or *Sync Notification* you want to remove.

2. Choose *Remove*.

Changing Ranks

When changing ranks make sure the new values also fulfill the requirement for uniqueness. This requirement may be violated temporarily while making multiple rank changes if all subscriptions in SLD systems of the affected synchronization landscape are deactivated.

1. Change the value of any *Local Rank* or *Source Rank* you want to adapt.

2. Choose *Update Ranks*.

Viewing Logs

Use

The SLD server logs relevant information for monitoring purposes. The SLD writes data to standard logs, which you can view with any SAP log viewer. You can also view the SLD logs in the SLD UI.

You have to view the logs regularly to check whether any errors have occurred. SAP support can also use the logs to analyze problems.

Procedure

Choose *Administration → Log*.

Setting the Log Level

1. Choose the *Settings* tab.

2. From the *Current Log Level* dropdown box, select the log level you want.

3. Choose *Set Permanently*.

   The log level you have selected is set permanently and globally for the cluster.

   Bear in mind that setting the log level to DEBUG or FINE may impact system performance because a lot of information is written to the file system. To improve system performance, set the log level to INFORMATION.

4. Choose the *Display* tab.

5. Choose *Refresh*.

   The log data you have selected is displayed. The newest log record is displayed at the top.

   To refresh the log data automatically, choose *Auto-Refresh On*.

Filtering and Viewing the SLD Server Log

1. Choose the *Display* tab.

2. To filter the SLD server log by log level, from the *Level* dropdown box, select the log level you want.

3. To filter the SLD server log and view only log data, which contains a certain string, in the *Filter* field, enter the string you want.

4. Choose *Go*.

Deleting the SLD Server Log

1. Choose the *Display* tab.

2. Choose *Clear Log*.

   The log data is deleted permanently.

Downloading the SLD Server Log from the Server

1. Choose the *Settings* tab.

2. To download the SLD server log from the server to view it locally or archive it, choose *Download Log*.

3. Save the file to a directory of your choice.

Downloading the Log Properties File from the Server

1. Choose the *Settings* tab.

2. To download the log properties file from the server to process it locally, choose *Download Log Settings*.

3. Save the file to a directory of your choice.

Transferring the Log Properties File to the Server

1. Choose the *Settings* tab.

2. To transfer the log properties file back to the server after you have modified it locally, choose *Browse* and browse to the file you want to upload.

3. Choose *Upload*.

   The file is transferred to the server.

   ⚠

   The changes to the configuration take effect only after you restart the SLD server.

4. Choose *Administration*.

5. Choose *Stop Server* and then choose *Start Server*.

Setting Filters for Logs of Different Components

1. Choose the *Component Filter* tab.

2. Use the filter to display the required components.

3. Change the value in the *Filter Level* column for every component you need.

   💡

   All sub-components have the same filter level as their parent component.

4. Continue your work with the logs.

⬇ Cleaning Up Data

Use

The data supplier programs report data about the systems in your landscape to the SLD. If a system no longer exists in your landscape, the data about the system remains in the SLD. You have to clean up the data manually in the SLD.

Procedure

1. Choose *Administration* → *Automatically Updated Data*.

2. Specify a *Data Supplier Type*.

3. Enter a date and optionally a time before which you want to display the systems that reported data to the SLD.

4. Set a *Filter* if you need.

5. Choose *Go*.

   A list of systems is displayed along with the date on which they last reported data to the SLD.

   💡

   If a particular date is too far in the past (see Scheduling the Data Collection Programs [external]), this is a good indication that the system no longer exists in the system landscape. The SLD has no way of knowing this, so you must manually remove the system from the SLD.

6. Select the system(s) you want to remove from the SLD.

⚠️

> Bear in mind that you remove not only the system but also all its associations to other systems, for example to business systems.

7. If you want to remove the selected system(s) and all their instances, choose *Remove*.

8. If you want to export the selected system(s) and all their instances:

   a. Choose *Export*.

   b. Choose *Download Exported Data*.

   c. Save the file to a directory of your choice.

Creating System Messages

Use

You can create an informational system message, which will be seen by every user who logs on to the SLD.

Procedure

1. Choose *Administration* → *System Message*.

2. In the *Message* field, enter the text of the system message.

3. Optionally, in the *Expires* field, enter a date when the message expires.

4. Choose *Save*.

Maintaining SLD Content at CIM Instance Level

Use

Rarely, you may have to use the *CIM Instances* function in the SLD, which allows you to maintain the content on a low-level by allowing you to work directly at CIM instance level.

💡

> Maintenance interfaces exist for most SAP applications that use the SLD. Use these specific interfaces for maintaining data to ensure that the applications run smoothly.

⚠️

> Bear in mind that modifying or removing CIM instances in the *CIM Instances* function may have undesired effects, for example, if you do not maintain the associations to other instances. Use the *CIM Instaces* function only for CIM classes with which you are familiar and only after consulting with SAP support.

Procedure

1. Choose *Home* → *CIM Instances*.

2. From the *Subset* dropdown box, select the subset of the data you want to display.

3. From the *Class* dropdown box, select the class whose instances you want to view and process.

> The number in the parentheses after the class name indicates the number of instances.

A table with all instances of the selected CIM class appears.

4. To filter the instance data and view only instance data that contains a certain string in the *Filter* field, enter the string you want and choose *Refresh*.

A table with all instances that meet the filter criteria appears.

Creating an Instance of the Selected CIM Class

1. Choose *New <Class_Name>*.

2. Enter the instance properties and choose *Save*.

> You must enter all key properties. The key properties are displayed in boldface.

Removing Multiple Instances

1. In the table with all instances, select the corresponding indicator in front of the instance(s) you want to remove.

2. Choose *Remove*.

Viewing and Modifying Instance Properties

1. To view instance properties, in the table with all instances, choose the corresponding link in the *<Class_Name>* column.

2. Change the instance properties you want and choose *Save*.

Creating a Copy of an Instance

1. On the instance detail screen, choose the *General* tab.

2. Choose *Copy*.

A separate detail screen for the new instance appears. All properties on this screen are identical to properties of the original instance.

3. Change the instance key properties you want, as well as any other properties you want.

4. Choose *Save*.

> The CIM concept does not allow you to create completely identical instances. Creating a copy means only that the data of the original instance is copied as a template for the new instance. The new instance must have at least one key property that is different from the key properties of the original instance.

Renaming an Instance

1. On the instance detail screen choose the *General* tab.

2. Choose *Move*.

3. Change the instance key properties you want, as well as any other properties you want.

4. Choose *Save*.

The renamed instance is saved. The renamed instance replaces the original instance. All associations to this instance are updated accordingly.

Viewing the Associations of an Instance

1. On the instance detail screen, choose *Associations*.

Creating an Association of the Selected Instance

1. Choose *New Association*.

2. From the *Association Type* dropdown box, select the association type you want.

    All instances that are suitable for the association type you selected are displayed.

3. Select the corresponding indicator in front of the instance(s) you want to associate with the selected instance and choose *Associate*.

Removing Associations of an Instance

1. Select the corresponding indicator(s) in front of the association(s) you want to remove.

2. Choose *Remove*.

    The associated instances are removed. The instances that are connected with the original instance by these associations remain.

Using Namespaces to Test Data

Use

The component and system landscape description contain current information about your system landscape. Using SLD simulations, you can also plan a future system landscape. We recommend that you test simulations in a different area to the current system landscape.

The namespace concept of the SLD allows you to create various namespaces as logical areas. The `sld/active` namespace mirrors the real system landscape. When you create a new namespace, it is initially empty. To create a duplicate of a namespace for test purposes, back up [external] the content of the source namespace and import [page 60] it to the target namespace.

Procedure

Choose *Administration* → *Namespaces*.

Creating Namespaces

1. Choose *Add*.

    A dialog box appears.

2. In the *Namespace* field, enter a name of the new namespace.

3. Choose *Add*.

Switching Between Namespaces

1. From the table with namespaces, select the namespace to which you want to switch.

    The system switches to the namespace you have selected.

    You can switch between namespaces on all other screens on which a namespace switch is relevant. The current namespace is displayed at the top of each screen.

⚠

The namespace could be reset to `sld/active` after some time. Check for the correct namespace before each critical operation such as delete.

Deleting Namespaces

1. From the *Namespaces* column, select the namespace you want to delete.

   The system switches to the namespace you have selected.

2. Choose *Remove <namespace>*.

   ⚠

   The system deletes the namespace, including all its data.

Deleting Namespace Instances

⚠

The system deletes permanently all instances of the namespace you selected. Make a backup copy of the instances, before you delete any instances permanently. For more information see Backing Up CIM Instances [external].

1. From the *Namespaces* column, select the namespace whose instances you want to delete.

   The system switches to the namespace you have selected.

2. Choose *Remove Instances*.

Using Exclusive Write Access

Use

Some use cases require that you restrict write access to your SLD. For example, during a system migration you may want to ensure that no data in the old SLD is changed until the migration to the new SLD is finished in order to avoid losing such changes.

For such purposes SLD provides features for exclusive write access. Write access will be restricted whereas read access is possible as usual. You have two options:

- Grant exclusive write access for a single user

  This may be necessary if you yourself need to make changes to the SLD content, but you need to prohibit that anyone else performs any changes.

- Deny write access for all users

  This option is appropriate if no one should be allowed to change any data.

Exclusive write access can be configured by any SLD administrative user. Any setting you select is persisted and will therefore still be active after an SLD restart.

Result

1. Log on to the SLD UI with an administrative user.

2. Choose *Administration → Settings.*

3. Choose section *Server Settings*.

4. Select an appropriate value from the dropdown list of parameter *Write Protection*. The following values are available:

- ○ *None*: This setting represent normal operation of SLD. Write access is not restricted. Each user can perform write operations according to his user permissions.

- ○ *Single User*: Exclusive write access is granted to the user that enables this setting. Write permissions for any other users are disabled.

- ○ *Read-Only*: Write permissions for all users are disabled.

5. Save your changes.

Additional Online Information about SLD

For more information about SLD, see SAP Service Marketplace at:

URL	Title
`service.sap.com/notes`	SAP Note 105132
	SAP Note 710315
	SAP Note 669669
	SAP Note 935245
`http://www.sdn.sap.com/irj/sdn/nw-sld`	*Planning Guide – System Landscape Directory*
	Post Installation Guide – SLD of SAP NetWeaver 7.10