

PDB User Guide

Parameter Database

USER GUIDE

Copyright

© Ericsson AB 2012 – 2016. All rights reserved. No part of this document may be reproduced in any form without the written permission of the copyright owner.

Disclaimer

The contents of this document are subject to revision without notice due to continued progress in methodology, design, and manufacturing. Ericsson shall have no liability for any error or damage of any kind resulting from the use of this document.

Trademark List

Ericsson

Ericsson is the trademark or registered trademark of Telefonaktiebolaget LM Ericsson. All other product or service names mentioned in this manual are trademarks of their respective companies.



Contents

1	About This Guide	1
1.1	Intended Audience	1
1.2	How This Guide is Organized	1
1.3	Conventions Used in This Guide	2
1.4	Prerequisites	3
1.5	Comments About the Documentation	4
2	PDB Overview	4
2.1	PDB GUI	5
2.2	PDB CLI	8
3	Authentication	8
3.1	Access Control Lists	9
4	Managing Applications	12
4.1	Adding a New Application	13
4.2	Modifying an Application	14
5	Managing Nodes	16
5.1	Adding Node Information	16
5.2	Modifying Node Information	25
6	Working with Solution Baselines	29
6.1	The Solution Baselines Workspace	29
6.2	Adding Baseline Information	33
6.3	Working with Baseline Revisions	36
6.4	Modifying Baseline Information	39
7	Working with Schemas	40
7.1	The Schemas Workspace	41
7.2	Importing a Schema	45
7.3	Working with Schema Revisions	49
7.4	Modifying Schema Properties	53
7.5	Exporting Schematron Rules	54
7.6	Deleting a Schema	55
7.7	Viewing Schema Elements	55



7.8	Comparing Two Schemas	63
8	Working with Node Configurations	67
8.1	The Configurations Workspace	68
8.2	Importing a Node Configuration	74
8.3	Creating a New Node Configuration	77
8.4	Creating a Delta Configuration	86
8.5	Working with Configuration Revisions	92
8.6	Modifying Configuration Properties	98
8.7	Migrating a Configuration to Another Schema	99
8.8	Validating a Configuration	102
8.9	Exporting a Node Configuration	104
8.10	Deleting a Node Configuration	114
9	Working with Configuration Data	115
9.1	The Configuration Browser	115
9.2	Working With Parameter Groups	129
9.3	Working with Parameters	134
10	Comparing Two Configurations	146
10.1	The Configuration Comparison Workspace	147
10.2	Performing a Comparison	152
10.3	Creating a Delta Configuration From Comparison Results	158
11	Working with Multi-Solution Configurations	159
11.1	The Multi-Solution Configurations Workspace	160
11.2	Creating a New Multi-Solution Configuration	164
11.3	Working with Multi-Solution Configuration Revisions	165
11.4	Working with Multi-Solution Configurations	169
11.5	Exporting a Multi-Solution Configuration	170
12	Working with Configuration Sets	171
12.1	Adding a New Configuration Set	172
12.2	Modifying a Configuration Set	173
12.3	Linking Node Configurations to a Configuration Set	175
13	Working with Site-Specific Lists	177
13.1	The Site-Specific Lists Workspace	178
13.2	Site-Specific Lists	182
13.3	Global Variables	190



13.4	Site-Specific Variables	193
14	Transferring PDB Data	201
14.1	The Data Transfer Workspace	202
14.2	Exporting PDB Data	206
14.3	Importing PDB Data	207
Appendix		
15	Export Criteria	210
15.1	LDIF Export Criteria	210
15.2	PVL Export Criteria	211
16	Configuration Validation Errors	212
16.1	Messages Applicable to All Configuration Formats	213
16.2	Messages Applicable to the LDIF Format	215
16.3	Messages Applicable to the NETCONF Format	216
16.4	Messages Applicable to the MPVL Format	216
17	MIM Validation Errors	218
17.1	Errors and Warnings Applicable to All CM MIM Formats	218
17.2	Errors and Warnings Applicable to the IS CM MIM Format	223
17.3	Errors and Warnings Applicable to the MP_DTD Format	225
18	Configuration Management Variables	227





1 About This Guide

The PDB User Guide describes the functionality offered by the Parameter Database (PDB) application. This guide provides an overview of the PDB interfaces and includes procedures and information required to interact with the system.

1.1 Intended Audience

This document is intended for PDB end-users.

Personnel working on Ericsson products or systems must have the training and competence required to perform their work correctly.

1.1.1 Prerequisite Knowledge

Users of this document should have knowledge and experience of the following:

- Node configuration files (LDIF, NETCONF, and so on)
- MIM files
- Linux

1.2 How This Guide is Organized

This document is organized into the following major sections:

Table 1 Document Organization

Section	Description
About This Guide	Introduces the guide. It describes the structure of the guide, conventions used, and all related documentation.
PDB Overview	Provides an overview of PDB.
Authentication	Provides instructions for logging in to the PDB GUI.
Managing Applications	Provides tasks for managing applications.
Managing Nodes	Provides tasks for managing nodes.
Working with Solution Baselines	Provides tasks for working with solution baselines.



Section	Description
Working with Schemas	Provides tasks for working with configuration schemas.
Node Configurations	Provides tasks for working with node configurations.
Working with Configuration Sets	Provides tasks for working with configuration sets.
Working with Site-Specific Lists	Provides tasks for working with site-specific lists.
Transferring PDB Data	Provides tasks for transferring data between PDB servers.
Appendix	Provides reference material on configuration export criteria, validation errors, and CM variables.

1.3 Conventions Used in This Guide

Table 2 provides a list of typographic conventions that may be encountered in this document:

Table 2 Typographic Conventions

Convention	Description	Example
Code Examples	Code examples	<code>stat char* months[] = {"Jan", "Feb"}</code>
Command Variables	You need to supply the values within the <code><></code>	<code><home_directory></code>
Document and File Names	References to document titles or sections in a document and file names	For more information, refer to the System Administrator Guide. Check the local runlog files (xxx.runlog and xxa.runlog) in the /var/log/xxx directory.
GUI Objects	GUI objects, such as menus, fields, and buttons, dialog boxes, and options	On the File menu, click Exit .
Key Combinations	Key combinations	Press Ctrl+X to delete the selected value. ⁽¹⁾



Convention	Description	Example
Output Information	Text displayed by the system	System awaiting input
Parameter/Configuration Values	Parameter values (numbers, true/false, yes/no, and so on)	To use this feature, the parameter must be set to true
System Elements	Command and parameter names, program names, path names, URLs, and directory names	The files are located in E:\Test The files are located in /etc/opt/ericsson/bin. ⁽²⁾
User Input	In this document when you are required to input content, the input content is displayed using this bold mono-spaced font. The content must be added exactly as shown.	cd \$HOME
Line Break	The arrow symbol (⇒) can be used when an inappropriate line break has been made. An inappropriate line break occurs when the code lines are too long to fit on the page, and there is no appropriate place for a line break.	cd /opt/msmw-cds-⇒ cyp-<version> (3)

(1) The plus sign (+) indicates that you must press the keys simultaneously.

(2) The use of the forward slash (/) is for Linux and UNIX systems; Windows systems use the backslash (\).

(3) The use of the ⇒ symbol (character entity ⇒) at the end of a line has a meaning to the human reader, but if copied and pasted from a CPI document to a command line interpreter the symbol must be cut from the code.

1.4 Prerequisites

- PDB has been successfully installed and configured.

For installation, upgrade, and rollback procedures, refer to the Parameter Database (PDB) Installation Instructions, 1/1531-CXP 902 0212.

- A web browser with JAVA support is available on the client machine.



Microsoft® Internet Explorer® version 7, or higher, is recommended.

1.5 Comments About the Documentation

Ericsson encourages you to provide feedback, comments, or suggestions so that we can improve the documentation to better meet your needs. With your comments, provide the following:

- Document title
- Document number and revision
- Page number

Please send your comments to your local Ericsson Support.

2 PDB Overview

PDB is the placeholder for configuration information of nodes, baselines and solutions. It contains all of the necessary information to successfully configure IMS nodes within a network environment.

PDB includes a GUI and a CLI that allow you to work directly with configuration data. Some tasks that can be performed using PDB include:

- Provisioning information related to baselines, solutions, nodes, node schemas, node configurations and configuration sets
- Importing schema data from different nodes and associate it to several node revisions
- Importing configuration data in several formats and from different nodes and associate it to several node revisions
- Working with node configurations, add/modify/delete configuration elements
- Defining conditions for exporting configuration elements based on site-specific parameters
- Creating delta configurations based on existing ones, which can be as well manipulated as any other regular configuration
- Exporting configuration data in several formats and for different nodes
- Comparing configurations
- Validating configurations characteristics like cardinality and value formatting



- Viewing information about configuration elements, such as descriptions, cardinality, value patterns and so on
- Provisioning lists of site-specific parameters, thus allowing a configuration to be independent of site-specific information

2.1 PDB GUI

PDB provides a Graphical User Interface (GUI) to manage the parameter database. The GUI is the primary way for users to interact with the system.

After logging in as described in Section 3 on page 8, the PDB home page is displayed. The home page shows the PDB welcome message or system notifications. See Figure 1.

System notifications are messages drafted by a PDB system administrator to communicate important information to users.

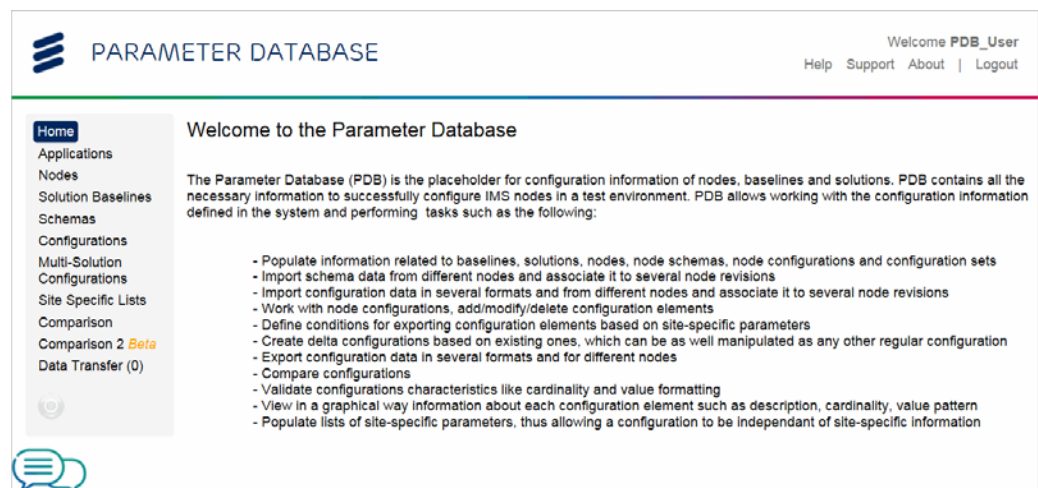


Figure 1 The PDB Home Page

PDB includes a link for **IDEAS FEEDBACK** on each page. Use  to provide ideas on new features or feedback for improvements.

All PDB functionality that is available in the GUI is accessible from the menu options that are always visible on the left side of the window.

The PDB menu provides direct access to the following interfaces:

Table 3 PDB Interfaces

Interface	Function
Home	Shows the welcome message.



Interface	Function
Applications	Allows users to create and manage applications. For more information see Section 4 on page 12.
Nodes	Allows users to create and manage definitions for node-revision pairs. For more information see Section 5 on page 16.
Solution Baselines	Allows users to create and manage IMS solution baselines. For more information see Section 6 on page 29.
Schemas	Allows the user to create and manage templates. For more information see Section 7 on page 40.
Configurations	Allows users to work with node configurations. For more information see Section 8 on page 67.
Multi-Solution Configurations	Allows users to work with multi-solution PVL configurations. For more information see Section 11 on page 159.
Configuration Sets	Allows users to create and manage node configuration sets. For more information see Section 12 on page 171.
Site Specific Lists	Allows users to manage the site-specific parameters in a node configuration. For more information see Section 13 on page 177.
Comparison	Allows users to compare two node configurations. For more information see Section 10 on page 146.
Data Transfer	Allows users to perform a PDB data transfer. For more information see Section 14 on page 201.

Help and Support

The PDB GUI includes several links to help and support located in the upper-right corner of the window, under the welcome message.

The following links are available:







Table 4 PDB Help and Support Links

Help	Connects to the Customer Product Information (CPI).
Support	Connects to the official PDB Support and Maintenance web page.
About	Provides the following information about PDB: <ul style="list-style-type: none"> • Product Number • Version • Server Host Name

PDB Activity Indicator

The PDB GUI includes an activity indicator located at the bottom of the menu options on the left side of the window. The indicator transitions through various colors to reflect the state of PDB requests.







Table 5 PDB Activity Indicators



Color	Indicator	Description
Grey		The connection to the PDB server is alive and no requests are pending for this account.
Blue		The connection to the PDB server is alive and a request is pending for this account.
Orange		The connection to the PDB server is alive and a request is pending for this account, but the expected response from the PDB server is delayed.
Red		The connection to the PDB server has been lost.

2.1.1

Using Tables

PDB data is presented in tables. Each table uses pagination to divide large data

sets. Navigate these tables using the controls (     ) at the bottom of the frame.

The data within each table is sorted in ascending or descending order. The sorting criteria is indicated by the up () or down () arrows in the column headings. You can change the sorting criteria within a table by clicking on the available headings.



2.2 PDB CLI

PDB includes a simple Command-Line Interface (CLI) client that can be used to execute common PDB tasks from a local machine. Two versions of the CLI client are available, one for Linux and one for Windows.

Refer to the PDB Command Line Interface (CLI) Reference, 1/1540-CXP 902 0212 for a complete description of each command.

3 Authentication

Connections to the PDB GUI are encrypted. Before accessing the PDB GUI, you must first log in and authenticate yourself. Logging in to PDB requires a valid user account. For more information on PDB user accounts, refer to the PDB System Administration Guide, 2/1543-CXP 902 0212.

Network connectivity is required for PDB user authentication. If a connection to the authentication server cannot be established, users are denied access to the PDB GUI.

To log in to the PDB GUI:

1. In your web browser, connect to the PDB GUI.

`https://<PDB_IP_ADDRESS>:8181/pdb`

The **Parameter Database Login** window opens. See Figure 2.



ERICSSON

PARAMETER DATABASE

Help Support

Use your Corporate ID and your LAN password to log in

Username

Password

Login

Version 2.34.0-R34A01

Server CAMOVCCLI248135.mo.ca.am.ericsson.se

Figure 2 Parameter Database Login

2. Enter the valid PDB user name and password.
3. Click **Login**.

The PDB GUI is displayed in your browser window.

3.1 Access Control Lists

PDB supports an Access Control List (ACL) for each configuration, schema, and site-specific list. An ACL sets access rights per user, granting permission to read or modify the associated object. ACLs act cumulatively with the role permissions granted by a system administrator, meaning that a given user would need both the necessary role membership and correct ACL permissions to work with specific configurations, schemas, or site-specific lists.

When a configuration, schema, or site-specific list is first added to PDB, the creating user is designated as the object owner and is granted unrestricted access. These owners or a system administrator are then able to grant access to other users by adding them to the ACL. Without ACL permissions, PDB users will not see the associated object when they log into PDB and will be unable to work with it.

Note: ACL permissions extend to all revisions of the same configuration or schema.

The following permissions are granted through the ACL:



READ	READ permission makes the governed object visible to the selected user. While READ users cannot make direct modifications to the object, they are able to perform indirect operations that do not modify the controlled data. READ permissions are granted automatically when a user is added to the ACL.
WRITE	WRITE permission authorizes the selected user to work with the controlled data and make modifications to it.
GRANT	GRANT permission authorizes the selected user to grant ACL permissions to other users.

Note: Users belonging to the `pdb_administrators` group always have READ, WRITE, and GRANT permissions on every object. Adding or removing an administrator from the ACL does not change their access rights.

Each configuration, schema, and site-specific list has its own ACL that is accessed through the associated workspace.

- For more information on working with node configurations , refer to Section 8 on page 67.
- For more information on working with multi-solution configurations, refer to Section 11.1 on page 160.
- For more information on working with configuration schemas, refer to Section 7.1 on page 41.
- For more information on working with site-specific lists, refer to Section 13 on page 177.

Working with Access Control Lists involves modifying the set permissions.

3.1.1 Modifying ACL Permissions

Authorized users can modify an ACL to grant or remove permissions.

Modifying an Access Control List requires GRANT privileges. Users with READ or WRITE permissions are able to view the ACL, but cannot make changes.








To modify an ACL:

1. In the corresponding workspace, right-click a configuration, schema, or ACL and select **Permissions**.

The ACL **Permissions** dialog box opens. See Figure 3.

☐ Allow anyone to read.

☐ Allow anyone to write.

User ▲	READ	WRITE	GRANT	
ceajmiz	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
ceajofr	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
efegasa	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
efergal	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	
efrapau	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	
egapasz	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	
ejoecar	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	

Add Users:

Separate usernames with spaces.


Figure 3 Permissions Dialog Box

The following table describes the elements of the **Permissions** dialog box:

Table 6 Permissions Dialog Box Elements

Element	Description	Notes
Allow anyone to read.	Grants READ permission to all PDB users.	
Allow anyone to write.	Grants WRITE permission to all PDB users.	Without public READ permissions, the configuration or schema will remain invisible to unauthorized users.



Element	Description	Notes
Access Control List (Table)	<p>Lists users that are already on the ACL. This table includes the following fields:</p> <ul style="list-style-type: none">• User• READ• WRITE• GRANT <p>READ, WRITE and GRANT privileges are set for each user with the corresponding check boxes.</p>	<p>Click  to remove a user from the ACL.</p>
Add Users	<p>Use this field to add PDB users to the ACL.</p>	<p>All ACL users must match the corresponding PDB user name.</p> <p>Multiple users can be added at the same time by separating the user names with spaces.</p>

2. Modify the ACL permissions as required. See Table 6.

3. Click **Apply**.

The permission changes are committed.

Note: Affected users may have to refresh the PDB GUI to see the changes.

4 Managing Applications

In PDB, applications, such as HSS, CSCF, PGM and MGW, are used to classify nodes by identifying the major tasks associated with it.

Working with applications involves the following activities:

- Adding a New Application
- Modifying an Application



4.1 Adding a New Application

New applications can be defined in PDB.

To add a new application:

1. In the PDB GUI, select **Applications** from the menu options on the left.

The **Applications** table is displayed. See Figure 4.

Applications

Name ▲	Description
CSCF	CSCF application
CUSTOM	CUSTOM application
HSS	HSS application
M-MGw	M-MGw application
MGC	MGC application
MGW	MGW application
MMAS	MMAS application
MTAS	MTAS application
PGM	PGM application
SBG	SBG application

Figure 4 Applications Table

The following table describes the different elements forming the **Applications** interface:

Table 7 Applications Interface Elements

Element	Description
	Updates the table with the latest information from the PDB database.
	Adds a new application to the table in edit mode.
	Opens the selected application in edit mode where it can be modified.
	Removes the selected application.



Element	Description
Name	The name of the application. Mandatory.
Description	A short description of the application. Optional.

2. Click **New**.

An empty application is added to the top of the table in edit mode. See Figure 5.

Applications

Name ▲	Description
<input type="text"/>	<input type="text"/>
CSCF	CSCF application
CUSTOM	CUSTOM application
HSS	HSS application
M-MGw	M-MGw application
MGC	MGC application
MGW	MGW application
MMAS	MMAS application
MTAS	MTAS application
PGM	PGM application
SBG	SBG application

Figure 5 Adding a New Application

3. Enter the required information. See Table 7.
4. Click **Apply**.

The new application is added to the database.

4.2 Modifying an Application

The properties of existing applications can be modified.

To modify an application:



1. In the PDB GUI, select **Applications** from the menu options on the left.

The **Applications** table is displayed. See Figure 6.

Applications	
<input type="button" value="Refresh"/> <input type="button" value="New"/> <input type="button" value="Edit"/> <input type="button" value="Delete"/>	
Name ▲	Description
CSCF	CSCF application
CUSTOM	CUSTOM application
HSS	HSS application
M-MGw	M-MGw application
MGC	MGC application
MGW	MGW application
MMAS	MMAS application
MTAS	MTAS application
PGM	PGM application
SBG	SBG application

Figure 6 Applications Table

The following table describes the different elements forming the **Applications** interface:

Table 8 Applications Interface Elements

Element	Description
<input type="button" value="Refresh"/>	Updates the table with the latest information from the PDB database.
<input type="button" value="New"/>	Adds a new application to the table in edit mode.
<input type="button" value="Edit"/>	Opens the selected application in edit mode where it can be modified.
<input type="button" value="Delete"/>	Removes the selected application.
Name	The name of the application.
Description	A short description of the application.

2. Select an application to modify from the table.



The **Edit** button becomes available.

3. Click **Edit**.

The selected application is opened in edit mode.

4. Update the node information as required. See Table 8.
5. Click **Apply**.

The updated node definition is saved to the database.

Note: To remove an application from PDB:

- Select an application to remove from the **Applications** table.

The **Delete** button becomes available.

- Click **Delete**.

A confirmation dialog box opens.

- Click **OK**.

An application that has been assigned to one or more nodes cannot be deleted until all node associations have been removed.

5 Managing Nodes

PDB handles the configuration for Ericsson IMS nodes (such as HSS, CSCF, and MTAS). Working with nodes involves the following activities:

- Adding Node Information
- Modifying Node Information

5.1 Adding Node Information

Adding node information to the PDB database involves the following tasks:

- Defining a New Node
- Defining a New Node Revision
- Associating a Schema with Node Revisions



— Associating a Configuration with Node Revisions

5.1.1 Defining a New Node

New nodes can be defined in PDB.

To add a new node:

1. In the PDB GUI, select **Nodes** from the menu options on the left.

The **Nodes** table is displayed. See Figure 7.

<input type="button" value="Refresh"/>	<input type="button" value="New"/>	<input type="button" value="Edit"/>	<input type="button" value="Delete"/>			
Name ▲	Application	Product Number	Description	Platform	Default Export Type	
CSCF 11B	CSCF	CXP 902 0350/1		TSP	LDIF	
HSS 11A	HSS	CXP 901 1865/6	Includes SLF 5.0	TSP	LDIF	
M-MGw R6	M-MGw	FGC 101 0559		CPP	PVL	
MGC 6.1 FD1	MGC	CXP 901 2687/1		IS	NETCONF	
MGW 2.0 FD1 IP4.2	MGW	CXR 101 0356		IS	NETCONF	
MTAS 11A	MTAS	CXP 901 1263/5		TSP	LDIF	
MTAS 11B	MTAS	CXP 901 1263/5		TSP	LDIF	
PGM 5.0	PGM	CXP 904 0082/2		OMP	NETCONF	
SBG 4.0	SBG	HSD 101 96/1		IS	NETCONF	


Figure 7 Nodes Table

The following table describes the different elements forming the **Nodes** interface:

Table 9 Nodes Interface Elements

Element	Description	Notes
<input type="button" value="Refresh"/>	Updates the table with the latest information from the PDB database.	
<input type="button" value="New"/>	Adds a new node definition to the table in edit mode.	
<input type="button" value="Edit"/>	Opens the selected node definition in edit mode where it can be modified.	



Element	Description	Notes
	Removes the selected node definition.	
Name	The name of the node.	
Application	The IMS application running on the node.	Mandatory. Available applications are provisioned in the Applications interface. For more information, see Section 4 on page 12.
Product Number	The product number associated with this node.	Mandatory.
Description	A short description of the node.	Optional.
Platform	The node platform. PDB supports the following platforms: <ul style="list-style-type: none">• CBA• CPP• EAS• IS• OMP• TSP	Mandatory.
Default Export Type	The default configuration format used by the node. PDB supports the following formats: <ul style="list-style-type: none">• LDIF• NETCONF• PED (EAS)• PVL	Mandatory.

2. Click **New**.

An empty node is added to the top of the table in edit mode. See Figure 8.



Nodes

Apply Cancel

Name ▲	Application	Product Number	Description	Platform	Default Export Type
<input type="text"/>	CUSTOM ▼	<input type="text"/>	<input type="text"/>	CBA ▼	CSV (report) ▼
CSCF 11B	CSCF	CXP 902 0350/1		TSP	LDIF
HSS 11A	HSS	CXP 901 1865/6	Includes SLF 5.0	TSP	LDIF
M-MGw R6	M-MGw	FGC 101 0559		CPP	PVL
MGC 6.1 FD1	MGC	CXP 901 2687/1		IS	NETCONF
MGW 2.0 FD1 IP4.2	MGW	CXR 101 0356		IS	NETCONF
MTAS 11A	MTAS	CXP 901 1263/5		TSP	LDIF
MTAS 11B	MTAS	CXP 901 1263/5		TSP	LDIF
PGM 5.0	PGM	CXP 904 0082/2		OMP	NETCONF
SBG 4.0	SBG	HSD 101 96/1		IS	NETCONF

Figure 8 Adding a New Node

- Enter the required information. See Table 9.
- Click **Apply**.

The new node is added to the database. New definitions are positioned in the table based on the sorting criteria.

5.1.2 Defining a New Node Revision

New node revisions can be defined in PDB.

To add a new node revision:

- In the PDB GUI, select **Nodes** from the menu options on the left.

The **Nodes** table is displayed.

- Select the node to which to you want to add a new revision.

The **Node Revisions** table is displayed. See Figure 9.



Node Revisions	
Refresh	New Edit Delete
Revision State ▲	Description
R6C	LSV154
R6D	LSV155
R6E	LSV156

Figure 9 Node Revisions Table

The following table describes the different elements forming the **Node Revision** interface:

Table 10 Node Revision Interface Elements

Element	Description
Refresh	Updates the table with the latest information from the PDB database.
New	Adds a new node revision to the table in edit mode.
Edit	Opens the selected node revision in edit mode where it can be modified.
Delete	Removes the selected node revision.
Revision State	The revision number associated with this node revision. Mandatory.
Description	A short description of the node revision. Optional

3. Click **New**.

An empty node revision is added to the top of the table in edit mode. See Figure 10.



Node Revisions

Apply
Cancel

Revision State ▲	Description
<input type="text"/>	<input type="text"/>
R6C	LSV154
R6D	LSV155
R6E	LSV156

Figure 10 Adding a New Node Revision

- Enter the required information. See Table 10.
- Click **Apply**.

The new node revision is added to the database. New revisions are positioned in the table based on the sorting criteria.

5.1.3

Associating a Schema with Node Revisions

While schemas are associated with a node during the import process, a given schema may not support all revisions of the selected node. Therefore, PDB node editors can directly associate a schema with the supported node revisions.

Note: Associating a schema with node revisions requires a user account with READ permission on the schema's ACL.

To associate a schema with node revisions:

- In the PDB GUI, select **Nodes** from the menu options on the left.

The **Nodes** table is displayed.

- Select a node from the table.

The **Node Revisions** table is displayed.

- Select a node revision to associate with a schema.

The **Schemas** table is displayed. See Figure 11.





Schemas			
<div>Refresh Add/Remove</div>			
	Name ▲	Document Number	Revision
<input checked="" type="checkbox"/>	MTAS_CM_MIM 	1/155 19-CXP 901 1263/5	PA1

Figure 11 Schemas Table

The following table describes the elements of the **Schemas** table:

Table 11 Schemas Table Elements

Element	Description
<div>Add/Remove</div>	Opens the list of available schemas in edit mode where specific revisions can be added or removed.
Name	The name of the schema. Click  to open the schema in the Schemas workspace.
Document Number	The document number associated with the schema.
Revision	The revision level of the schema.

4. Click **Add/Remove**.

A list of schema revisions that are associated with the selected node is displayed in edit mode. See Figure 12.

Note: Schema revisions are associated with a node during the schema import process. For more information on importing schemas, refer to Section 7.2 on page 45.



Schemas

Apply
Cancel

	Name ▲	Document Number	Revision
<input type="checkbox"/>	MTAS_11A_CM_MIM 	1/155 19-CXP 901 1263/5	PA2
<input checked="" type="checkbox"/>	MTAS_CM_MIM 	1/155 19-CXP 901 1263/5	PA1

Figure 12 Linking a Schema to Node Revisions

- Select the check boxes next to the schema revisions that are compatible with the node revision.
- Click **Apply**.

The selected schema revisions are linked to the node revision.

5.1.4

Associating a Configuration with Node Revisions

Node configurations are associated with a schema during the import process. While a schema may have already been associated with one or more node revisions, a specific configuration using that schema may only be suitable for a subset of the revisions supported by the configuration schema. Therefore, PDB node editors can directly associate a node configuration with the specific node revisions that are supported.

Note: Associating a configuration with node revisions requires a user account with READ permission on the configuration's ACL.

To associate a configuration with node revisions:

- In the PDB GUI, select **Nodes** from the menu options on the left.

The **Nodes** table is displayed.

- Select a node from the table.

The **Node Revisions** table is displayed.

- Select a node revision to associate with a configuration.

The **Configurations** table is displayed. See Figure 13.





Configurations			
<div>Refresh Add/Remove</div>			
	Name	Document Number	Revision
<input checked="" type="checkbox"/>	/MTAS_11A_R1D_LSV118_LDIF 	4/190 73-CXP 901 1263/5	PA2

Figure 13 Configurations Table

The following table describes the elements of the **Configurations** table:

Table 12 Configurations Table Elements

Element	Description
<div>Add/Remove</div>	Opens the list of available configurations in edit mode where specific revisions can be added or removed.
Name	The name of the configuration. Click  to open the configuration in the Configurations workspace.
Document Number	The document number associated with the configuration.
Revision	The revision level of the configuration.

4. Click **Add/Remove**.

A list of configuration revisions is displayed in edit mode. See Figure 14.

Note: The list of available configurations is derived from the schemas that have already been associated with the node revision. If the selected node revision has not been associated with a schema, this list will be empty. For more information on associating a schema with node revisions, refer to Section 5.1.3 on page 21.



Configurations

Apply Cancel



	Name	Document Number	Revision
<input checked="" type="checkbox"/>	/MTAS_11A_R1D_LSV118_LDIF 	4/190 73-CXP 901 1263/5	PA2
<input type="checkbox"/>	/MTAS_11A_R1D_LSV118_LDIF 	4/190 73-CXP 901 1263/5	PA1

Figure 14 Linking a Configuration with Node Revisions

5. Select the check boxes next to the configuration revisions that are compatible with the node revision.
6. Click **Apply**.

The selected configuration revisions are linked to the node revision.

5.2 Modifying Node Information

Modifying node information in the PDB database involves the following tasks:

- Modifying a Node
- Modifying a Node Revision

5.2.1 Modifying a Node

The properties of existing node definitions can be modified.

To modify a node:

1. In the PDB GUI, select **Nodes** from the menu options on the left.

The **Nodes** table is displayed. See Figure 15.



Nodes					
<div>Refresh New Edit Delete</div>					
Name ▲	Application	Product Number	Description	Platform	Default Export Type
CSCF 11B	CSCF	CXP 902 0350/1		TSP	LDIF
HSS 11A	HSS	CXP 901 1865/6	Includes SLF 5.0	TSP	LDIF
M-MGw R6	M-MGw	FGC 101 0559		CPP	PVL
MGC 6.1 FD1	MGC	CXP 901 2687/1		IS	NETCONF
MGW 2.0 FD1 IP4.2	MGW	CXR 101 0356		IS	NETCONF
MTAS 11A	MTAS	CXP 901 1263/5		TSP	LDIF
MTAS 11B	MTAS	CXP 901 1263/5		TSP	LDIF
PGM 5.0	PGM	CXP 904 0082/2		OMP	NETCONF
SBG 4.0	SBG	HSD 101 96/1		IS	NETCONF

Figure 15 Nodes Table

The following table describes the different elements forming the **Nodes** interface:

Table 13 Nodes Interface Elements

Element	Description
<div>Refresh</div>	Updates the table with the latest information from the PDB database.
<div>New</div>	Adds a new node definition to the table in edit mode.
<div>Edit</div>	Opens the selected node definition in edit mode where it can be modified.
<div>Delete</div>	Removes the selected node definition.
Name	The name of the node.
Application	The IMS application running on the node. Note: Available applications are provisioned in the Applications interface. For more information, see Section 4 on page 12.
Product Number	The product number associated with this node.
Description	A short description of the node.



Element	Description
Platform	The node platform.
Default Export Type	The default configuration format used by the node.

2. Select a node to modify from the table.

The **Edit** button becomes available.

3. Click **Edit**.

The selected node is opened in edit mode.

4. Update the node information as required. See Table 13.

5. Click **Apply**.

The updated node definition is saved to the database.

Note: To remove a node definition from PDB:

- Select a node to remove from the **Nodes** table.

The **Delete** button becomes available.

- Click **Delete**.

A confirmation dialog box opens.

- Click **OK**.

When a node is removed from the database, all of the associated node revisions are automatically removed.

5.2.2 Modifying a Node Revision

The properties of existing node revisions can be modified.

To modify a node revision:

1. In the PDB GUI, select **Nodes** from the menu options on the left.

The **Nodes** table is displayed.

2. Select a node from the table.

The **Node Revisions** table is displayed.

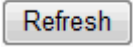





Node Revisions	
Refresh	New Edit Delete
Revision State ▲	Description
R6C	LSV154
R6D	LSV155
R6E	LSV156

Figure 16 Node Revisions Table

The following table describes the different elements forming the **Node Revision** interface:

Table 14 Node Revision Interface Elements

Element	Description
	Updates the table with the latest information from the PDB database.
	Adds a new node revision to the table in edit mode.
	Opens the selected node revision in edit mode where it can be modified.
	Removes the selected node revision.
Revision State	The revision number associated with this node revision.
Description	A short description of the node revision.

3. Select a node revision to modify from the table.

The **Edit** button becomes available.

4. Click **Edit**.

The selected node revision is opened in edit mode.

5. Update the node revision information as required. See Table 14.

6. Click **Apply**.

The updated node revision is saved to the database.



Note: To remove a node revision from PDB:

- Select a node to remove from the **Node Revisions** table.

The **Delete** button becomes available.

- Click **Delete**.

A confirmation dialog box opens.

- Click **OK**.

6 Working with Solution Baselines

Solution baselines are a controlled collection of node configurations. In PDB, only baseline editors or administrators can add or modify solution baselines. Working with solution baselines involves the following activities:

— Adding Baseline Information

— Modifying Baseline Information

Solution baselines stored in PDB are revision controlled. PDB validates revision levels following Ericsson's standard rules for document handling. A given revision will precede or supersede other revisions of the same solution baseline; this relationship opens a number of operations when working with a revised solution baseline. These operations include:

— Changing the Document State of a Baseline

— Creating a New Baseline Revision

6.1 The Solution Baselines Workspace

The **Solution Baselines** workspace allows you to carry out tasks with solution baselines.

To access the **Solution Baselines** workspace, select **Solution Baselines** from the menu options on the left. See Figure 17.



Solution baselines

Name Revision

Document Number User

Document States ☒ PREL ☒ FROZ ☒ FREE ☒ WIDR

Name ▲	Action	Document Number	Revision	Document State	Description
VoLTE (Native) Parameter Value List BL 16A.14	<input type="button" value="Open"/> ▼	PDB-INTERNAL-YBUXCBPDFGNKN	PA1	FREE	This is a base VoLTE Solution configuration set. Prior to IMS 16.1 release.
VoLTE (Native) Parameter Value List BL 16A.15	<input type="button" value="Open"/> ▼	PDB-INTERNAL-BSHVHGMFZRULZZ	PA1	FROZ	This is a VoLTE Parameter Value List for Solution Base line 16A.15 (Internal delivery). This is the base for IMS 16.1 Release.
VoLTE Parameter Value List BL N549	<input type="button" value="Open"/> ▼	PDB-INTERNAL-BTWCRJYVDTJZGB	PA1	FROZ	This is a VoLTE Parameter Value List for Solution Base line N549 (Internal delivery). This corresponds to IMS 16.1 Release.
VoLTE Parameter Value List BL N551	<input type="button" value="Open"/> ▼	PDB-INTERNAL-BXMZGXVOWHRMHJ	PA1	PREL	This is a VoLTE Parameter Value List for Solution Base line N551 (Internal delivery). This corresponds to IMS 16.1 Release.

Figure 17

The workspace is divided into two principal areas as follows:

Search

The search options, located at the top of the page, allow you to filter the solution baselines that are displayed in the **Solution Baselines** table. For more information on performing a search, refer to Section 6.1.1 on page 31.

Table

The **Solution Baselines** table is the centerpiece of the **Solution Baselines** workspace. This table displays solution baselines that match the selected search criteria and allows you to perform a number of baseline-specific operations using a context menu.

The following table describes the elements of the **Solution Baselines** workspace.

Table 15 Solution Baselines Workspace Elements

Element	Description
<input type="button" value="New"/>	Creates a new solution baseline.
<input type="button" value="Show Details"/>	Expands the Solution Baselines table to show the User and Last Modification columns.
Name	The name of the solution baseline.



Element	Description
Action	Provides an Open button for quick access to the configurations in the solution baseline. The drop-down arrow opens a context menu for the baseline.
Document Number	The document number associated with the solution baseline.
Revision	The revision of the solution baseline.
Document State	Displays the document state of the solution baseline. The following document states are available: <ul style="list-style-type: none"> • PREL - Preliminary • FROZ - Frozen • FREE - Approved • WIDR - Withdrawn For more information on document states, refer to Section 6.3.1 on page 36.
Description	An optional description of the solution baseline.
User	The last user to modify baseline information. ⁽¹⁾
Last Modification	A timestamp marking the last change to the baseline information. ⁽¹⁾

(1) This column is normally hidden. Click **Show Details** to display this information.

Each row in the **Solution Baselines** table is selectable. Right-clicking a row opens a context menu where operations specific to the selected baseline can be executed.

6.1.1 Searching for Solution Baselines

The **Solution Baselines** table can be filtered by performing a search. A number of search criteria are available to help you find specific baselines. PDB reports partial matches on search strings. Use double quotes <" "> to restrict the search to exact matches.

Searches are performed using the search workspace at the top of the **Solution Baselines** page. See Figure 18.

Solution baselines

Name

Revision

Document Number

User

Document States
☒ PREL
☒ FROZ
☒ FREE
☒ WIDR

Search

Clear

Figure 18 Baselines Search

The following table describes the available search criteria.

Table 16 Solution Baselines Search Elements

Element	Description
Name	Filters the table for baselines that match the selected name.
Revision	Filters the table for baselines that have the selected revision.
Document Number	Filters the table for baselines that have the selected with document number.
User	Filters the table for baseline revisions that were created by the specified user.
Document States	Filters the table for baselines with the selected document states.

To search the **Solution Baselines** table:

1. In the **Solution Baselines** workspace, set your search criteria.
2. Click **Search**.

The **Solution Baselines** table is populated with baselines that match the selected criteria.

Search results are retained as you navigate through the web portal. To reset the **Solution Baselines** table to the default display, click **Clear** then click **Search**.

Each solution baseline has a URL. This link provides an external reference to the specific baseline. Following a direct link connects you to the PDB server. After logging in, PDB automatically loads the **Solution Baselines** workspace and shows the linked baseline.

URLs are automatically generated by PDB. To access a URL, right-click a baseline and select **Copy URL**.



6.2 Adding Baseline Information

Adding baseline information to the PDB database involves the following tasks:

- Creating a New Solution Baseline
- Adding Node Configurations to a Solution Baseline

6.2.1 Creating a New Solution Baseline

New baselines can be created in PDB.

To create a new baseline:

1. In the **Solution Baselines** workspace, click **New**.

The **New** dialog box opens. See Figure 19.

The 'New' dialog box is a light gray window with a title bar labeled 'New'. It contains four labeled input fields arranged vertically. The first is 'Baseline Name' with a text box. The second is 'Document Number' with a text box and a small lightbulb icon to its right. The third is 'Description' with a larger text area and a vertical scroll bar on its right. The fourth is 'Revision' with a text box containing the text 'PA1'. At the bottom center of the dialog are two buttons: 'Create' and 'Cancel'.

Figure 19 New Dialog Box

The following table describes the different elements forming the **New** dialog box:

Table 17 New Dialog Box Elements

Element	Description	Notes
Baseline Name	The name of the solution baseline.	Mandatory.



Table 17 New Dialog Box Elements

Element	Description	Notes
Document Number	The document number of the solution baseline.	Optional. If left blank, PDB will automatically generate an internal document number for the new baseline. This number can be modified by changing the baseline properties as described in Section 6.4.1 on page 39.
Description	A short description of the new solution baseline.	Optional.
Revision	The revision of the new solution baseline.	Mandatory. New baselines are automatically assigned a PA1 revision. This selection can be changed by updating the Revision field.

2. Enter the required information. See Table 17.
3. Click **Create**.

The new baseline is added to PDB. The **Solution Baselines** is automatically filtered to display the new baseline.

6.2.2 Adding Node Configurations to a Solution Baseline

Node configurations are added to a solution baseline in the **Configurations** workspace. For more information on the **Configurations** workspace, refer to Section 8.1 on page 68.

To add a node configuration to a solution baseline:

1. In the **Configurations** workspace, click **Add to Solution Baseline**.

The **Add a Configuration** dialog box opens. See Figure 20.



Add a configuration

Solution baseline

The 'Node' and 'Node Revision' fields act as a filter for configuration names.
You may use them or simply start typing the name of the configuration in the 'Configuration' field.
The use of the filter is optional.

Node

Node Revision

Configuration

Figure 20 Add a Configuration Dialog Box

The following table describes the different elements forming the **Add a Configuration** dialog box:

Table 18 Add a Configuration Dialog Box Elements

Element	Description	Notes
Solution Baseline	The name of the solution baseline.	<p>Mandatory.</p> <p>Only baselines in a PREL state can be modified. For more information on document states, refer to Section 6.3.1 on page 36.</p> <p>This field uses auto-complete functionality. Typing part of the baseline name displays a list of matching solution baselines. Use the down-arrow on your keyboard to display the complete list.</p>
Node (List)	Filters the available node configurations for configurations that are associated with the selected node.	Optional.



Table 18 Add a Configuration Dialog Box Elements

Element	Description	Notes
Node Revision (List)	Filters the available node configurations for configurations that are associated with the selected node revision.	Optional.
Configuration	The name of the node configuration to add to the selected solution baseline.	Mandatory. This field uses auto-complete functionality. Typing part of the configuration name displays a list of matching node configurations. Use the down-arrow on your keyboard to display the complete list.

2. Enter the required information. See Table 18.

3. Click **Add**.

The selected node configuration is added to the specified solution baseline.

6.3 Working with Baseline Revisions

Solution baselines stored in PDB are revision controlled. A given revision will precede or supersede other revisions of the same solution baseline, allowing for a number of revision-specific operations.

These operations include:

- Changing the Document State of a Baseline
- Creating a New Baseline Revision

By default, only the latest revision of each baseline is displayed in the **Solution Baselines** table. Older revisions are accessible through the revision history or by refining the search criteria. For more information on performing a search, refer to Section 6.1.1 on page 31.

To view all revisions of a selected baseline, right-click to open the context menu and select **Revision > Show Revision History**.

A solution baseline must be in a non-preliminary state (FROZ, FREE, WIDR) before a new revision can be created.



6.3.1 Document States

PDB uses a system of document states to provide information on the completeness, quality, and approval status of a particular solution baseline. The document state is indicated by a status code that is part of the baseline metadata. PDB uses the following document states for solution baselines:

PREL	Preliminary. Used to designate an unlocked baseline that is still under development.
	Note: Baselines in the PREL state cannot serve as a basis for new revisions.
FROZ	Frozen. Used to designate a frozen baseline where the content has been locked to prevent further changes. This is the basic state for baseline revisions that are stored in PDB but are not approved for release.
FREE	Approved. Used to designate solution baselines that are approved for release. Approved baselines are locked to prevent any changes to the released content.
WIDR	Withdrawn. Used to designate solution baselines that are no longer intended for use. Withdrawn baselines are still accessible in PDB, but are locked to prevent further changes.

6.3.1.1 Changing the Document State

PDB allows you to change the document state of solution baselines.

Solution baselines must follow the following sequence of document states:

PREL > FROZ > FREE > WIDR

Note: Only PDB System Administrators can set a baseline to a previous state.

New baselines start in the PREL state where the baseline information can be modified and updated. The document state is changed as the baseline progresses through its lifecycle.

To change the document state of a solution baseline:

1. In the **Solution Baselines** workspace, right-click a baseline and select **Revision > Set to <STATE>**.

Note: If the baseline is not visible, perform a search as outlined in Section 6.1.1 on page 31.

When moving from PREL to FROZ, the **Freeze** dialog box opens. See Figure 21.

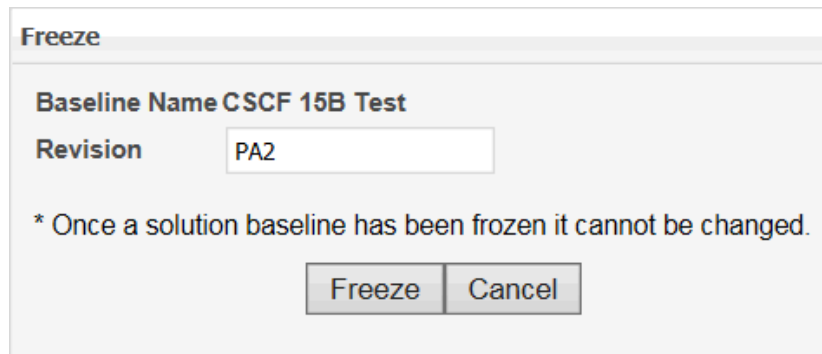


Figure 21 Freeze Dialog Box

The Freeze dialog box allows you to modify the revision level of the baseline. Here preliminary revisions can be set to a solid state before freezing.

Stop!

Only PDB System Administrators can unfreeze a baseline. Do not continue if additional changes are required to the baseline information.

2. Click **Freeze**.

The selected baseline is locked and the document state is set to FROZ.

For all other state transitions a confirmation dialog box is displayed.

Click **OK** to change the document state.

6.3.2 Creating A New Baseline Revision

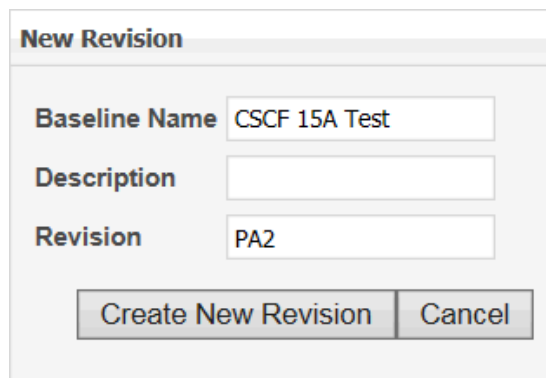
A locked solution baseline can be updated by creating a new revision. This process creates an unlocked copy of the original baseline at a new revision level. By default, the new revision retains the same name and description as the original baseline. These fields can be modified, if necessary.

To create a new revision of a locked baseline:

1. In the **Solution Baselines** workspace, right-click a baseline in state FROZ, FREE, or WIDR and select **Revision > Create New Revision**.

Note: If the baseline is not displayed, perform a search as outlined in .

The **New Revision** dialog box opens with the next legal revision displayed. See Figure 22.

The dialog box is titled "New Revision". It contains three input fields: "Baseline Name" with the text "CSCF 15A Test", "Description" which is empty, and "Revision" with the text "PA2". At the bottom, there are two buttons: "Create New Revision" and "Cancel".

Figure 22 New Revision Dialog Box

2. Verify that the proposed name, description, and revision are correct and update as necessary.

3. Click **Create New Revision**.

A new revision is created.

6.4 Modifying Baseline Information

Modifying baseline information in the PDB database involves the following tasks:

- Modifying a Solution Baseline
- Deleting a Solution Baseline

6.4.1 Modifying a Solution Baseline

While in PREL state, the following baseline properties can be modified:

- Names
- Document numbers

If no document number is specified, PDB automatically generates an internal number.

Note: Document numbers are validated using Ericsson's standard rules for registration notation.

Subsequent revisions of an existing baseline are locked to the document number of their predecessor.

- Descriptions

If the solution baseline is in a locked state, only the **Description** field can be modified.

To edit baseline properties:



1. In the **Solution Baselines** workspace, right-click a baseline and select **Edit**.

Note: If the baseline is not displayed, perform a search as outlined in Section 6.1.1 on page 31.

The **Edit** dialog box opens.

2. Update the baseline properties as required.
3. Click **Apply**.

The updated properties are saved.

6.4.2 Deleting a Solution Baseline

Solution Baselines in the PREL state can be deleted. Deleting a solution baseline does not remove any associated node configurations.

Note: Only PDB System Administrators can remove locked solution baselines, including older revisions of a current baseline.

To delete a solution baseline:

1. In the **Solution Baselines** workspace, right-click a baseline schema and select **Delete**.

Note: If the baseline is not displayed, perform a search as outlined in Section 6.1.1 on page 31.

A confirmation dialog box opens.

Caution!

Deleting a baseline permanently removes it from PDB.

2. Click **OK**.

The solution baseline is deleted.

7 Working with Schemas

Configuration schemas are templates containing all the possible classes, attributes, relationships, and constraints that are part of the configuration of a



specific node revision. PDB binds all configurations to the constraints laid out in a configuration schema. This relationship ensures that the elements contained within a configuration are understood by the supported nodes.

Working with configuration schemas can involve the following activities:

- Importing a Schema
- Modifying Schema Properties
- Exporting Schematron Rules
- Deleting a Schema
- Viewing Schema Elements
- Comparing Two Schemas

PDB keeps an Access Control List (ACL) for each schema. All non-administrative users require ACL permissions to view or modify the associated schema. For more information on ACLs, refer to Section 3.1 on page 9.

Schemas stored in PDB are revision controlled. PDB validates revision levels following Ericsson's standard rules for document handling. A given revision will precede or supersede other revisions of the same configuration schema; this relationship opens a number of operations when working with a revised schema. These operations include the following:

- Changing the Document State of a Configuration Schema
- Importing a New Schema Revision

7.1 The Schemas Workspace

The **Schemas** workspace allows you to carry out tasks with configuration schemas.

To access the **Schemas** workspace, select **Schemas** from the menu options on the left. See Figure 23.



Schemas

Node Node Revision

Document Number User Name Revision ☒ Latest Only

Document States

☒ PREL ☒ FROZ

☒ FREE ☒ WIDR

Name	Action	Document Number	Revision	Document State	Node	Node Revision	IVL
CSCF_CM_MIM	Open	1/155 19-CXP 902 0350/1	A	FROZ	CSCF 11B	R5B	/CSCF_11B_R4B_LDIF_NoEmptyValues[A]
HSS_CM_MIM	Open	1/155 19-CXP 901 1865/6	A	FROZ	HSS 11A	R7A	NONE
MTAS_11A_CM_MIM	Open	1/155 19-CXP 901 1263/5	PA2	PREL	MTAS 11A		NONE
MTAS_11B_CM_MIM	Open	8/155 19-CXP 901 1263/5	PA4	FROZ	MTAS 11B	R6D	/MPVL_MTAS_11B_Fixed[PA4]

Figure 23 The Schemas Workspace

The workspace is divided into two principal areas as follows:

Search

The search options, located at the top of the page, allow you to filter the schemas that are displayed in the **Schemas** table. For more information on performing a search, refer to Section 7.1.1 on page 43.

Schemas Table

The **Schemas** table is the centerpiece of the **Schemas** workspace. This table displays configuration schemas that match the selected search criteria and allows you to perform a number of schema-specific operations using a context menu.

The following table describes the elements of the **Schemas** workspace.

Table 19 Schemas Workspace Elements

Element	Description
<input type="button" value="Import..."/>	Imports a ZIP or TAR file containing Schema data. PDB automatically validates the syntax of the schema files and examines the relationship between configuration elements. A report is generated if problems are found.
<input type="button" value="Show Details"/>	Expands the Schemas table to show the User and Import Date columns.
Name	The name of the schema.



Element	Description
Action	Provides an Open button for quick access to schema data. The drop-down arrow opens a context menu for the schema.
Document Number	The document number associated with the schema.
Revision	The revision of the schema.
Document State	<p>Displays the document state of the schema. The following document states are available:</p> <ul style="list-style-type: none"> • PREL - Preliminary • FROZ - Frozen • FREE - Approved • WIDR - Withdrawn <p>For more information on document states, refer to Section 7.3.1 on page 50.</p>
Node	The node associated with the schema.
Node Revision	A comma-separated list of node revisions that have been associated with the schema. For more information on associating a schema with node revisions, refer to Section 5.1.3 on page 21.
IVL	<p>The name of the Initial Value List (IVL) associated with the schema.</p> <p>Click 🔗 to open the configuration in the Configurations workspace.</p>
User	The user who imported the schema. ⁽¹⁾
Import Date	A timestamp marking when a schema was imported. ⁽¹⁾

(1) This column is normally hidden. Click **Show Details** to display this information.

Each configuration schema listed in the **Schemas** table is selectable. Right-clicking on a schema opens a context menu where operations specific to the selected schema can be executed.



7.1.1 Searching for Schemas

The **Schemas** table can be filtered by performing a search. A number of search criteria are available to help you find specific schemas. PDB reports partial matches on search strings. Use double quotes ">" to restrict the search to exact matches.

Searches are performed using the search workspace at the top of the **Schemas** page. See Figure 24.

Schemas

Node Node Revision

Document Number User Name Revision ☒ Latest Only

Document States

☒ PREL ☒ FROZ

☒ FREE ☒ WIDR

Figure 24 Schemas Search

The following table describes the available search criteria.

Table 20 Schema Search Elements

Element	Description	Notes
Node (List)	Filters the table for schemas that are associated with the selected node.	
Node Revision (List)	Filters the table for schemas that are associated with the selected node revision.	The Node Revision list is only populated after a selection has been made from the Node list.
Document Number	Filters the table for schemas with document numbers that match the selected criteria.	
User	Filters the table for schemas that were imported by the specified user.	
Name	Filters the table for schemas with names matching the selected criteria. All entries are case sensitive.	
Revision	Filters the table for schemas with revisions that match the selected criteria.	Latest Only must be deselected to perform a search using this field.



Table 20 Schema Search Elements

Element	Description	Notes
Latest Only	Includes only the latest schema revisions that match the other search criteria.	<p>If this option is selected, search results are restricted to the latest revision.</p> <p>Older revisions can be accessed by deselecting this option or by using revision history. To view all revisions of a selected schema, right-click to open the context menu and select Revision > Show Revision History.</p>
Document States	Filters the table for schemas with the selected document states.	

To search the **Schemas** table:

1. In the **Schemas** workspace, set your search criteria.
2. Click **Search**.

The **Schemas** table is populated with schemas that match the selected criteria.

Note: PDB automatically stores up to 10 consecutive searches. Use the navigation buttons to move between each search.

Search results are retained as you navigate through the web portal. To reset the **Schemas** table to the default display, click **Clear** then click **Search**.

Each configuration schema has a URL. This link provides an external reference to the specific schema. Following a direct link connects you to the PDB server. After logging in, PDB automatically loads the **Schemas** workspace and shows the linked schema.

URLs are automatically generated by PDB. To access a URL and other properties, right-click a schema and select **Properties**.

7.2 Importing a Schema

A schema is created by importing Management Information Modelling (MIM) files to PDB. These MIM files are produced and delivered by the different Node Development Organizations (NDOs) for each software release.

Schemas are composed of MIMs and support files such as the index and model files that are part of the IS_CM_MIM format.



Once a configuration schema has been imported, a list of the imported files is available from the **Schema Details** dialog box. To access **Schema Details**, right-click a schema and select **Properties**.

PDB supports the following MIM formats:

— IMS_CM_MIM

- 3.0
- 2.0
- 1.0
- 0.1

For more information on the IMS CM MIM format, refer to IMS CM MIM Description for TSP Nodes, 17/1550-HSC 113 06 Uen.

— IS_CM_MIM

For more information on IS CM MIM format, refer to Management Information Modeling in IS, 25/155 19-AZE 101 01/1 Uen E.

— MP_DTD

- B
- D
- E
- E1
- F
- G

For more information on the MP_DTD format, refer to the MOM DTD, 006 91-APR 901 950 Uen.

— TSP_MIM

For more information on TSP MIM format, refer to TSP MIM Specification Reference Manual, 1/1553-CXA 110 3387 Uen.

Do!

All necessary schema files (such as: MIM, index, model, and so on) must be packaged in a ZIP or TAR archive before they can be imported to PDB.

An archive directory structure is not required by PDB to import schemas; however, if directories are present inside the file, then PDB will recursively scan for schema files within the directories as needed.

Schemas are an essential part of node configurations and must be imported before working with configuration data.

To import a configuration schema:

- 1. In the **Schemas** workspace, click **Import**.

The **Import** dialog box opens. See Figure 25.

Import ...

Node

M-MGw R6

Schema Name

Document Number

Description

Revision

PA1

Revision Comment

Input File

Browse...

Uploaded File

Import

Cancel

Figure 25 Import Dialog Box

The following table describes the elements of the **Import** dialog box:



Table 21 Import Dialog Box Elements

Element	Description	Notes
Node (List)	A list of nodes defined in PDB. Used to associate the new schema with an existing node.	Mandatory. For more information on working with nodes in PDB, refer to Section 5 on page 16.
Schema Name	The name of the schema.	Mandatory.
Document Number	The document number of the schema.	Optional. If left blank, PDB will automatically generate an internal document number for the new schema. This number can be modified by editing the schema properties as described in Section 7.4 on page 53.
Description	A short description of the new configuration.	Optional.
Revision	The revision of the schema.	Mandatory. During the same session, PDB will automatically populate this field with the revision set in the previous import operation. The next expected revision level is suggested to the right of the Revision field.
Input File	The path and file name of the schema archive.	Mandatory.

2. Enter the required information. See Table 17.

Stop!

PM MIM files are not supported by PDB.

3. Click **Import**.

PDB performs the following operations when importing a new configuration schema.



- a PDB imports the data contained within the selected archive and generates a new configuration schema.
- b PDB performs a validation check during the import process. This check verifies the syntax of the schema files and examines the relationship between configuration elements.

Note: Structural errors interfere with PDB's ability to parse the incoming files and will cause the import process to fail.

- c PDB automatically generates an import validation report when problems are found. Errors and warnings that were found during validation are reported for each affected schema file. This report can be downloaded from the **Report Available** dialog box that appears at the end of the import process.
- d PDB stores the schema archive and the import validation report. These files can be downloaded from the properties dialog box for the new schema.

The following example presents a sample import validation report.

HSS_11A_R6A_MIM_CM_001.ZIP Validation Report

FILE VALIDATION REPORT
Filename: sda_cm_mim.xml

```
WARNING Attribute 'HSS-AdministrativeState' is already defined in class 'HSS-Application'. Skipping it
WARNING Attribute 'HSS-InstallationType' is already defined in class 'HSS-Application'. Skipping it
WARNING Attribute 'HSS-IsDataCacheUsed' is already defined in class 'HSS-Application'. Skipping it
WARNING Attribute 'HSS-TransparentDataLicense' is already defined in class 'HSS-License'. Skipping it
WARNING Attribute 'HSS-PsCsDataRequestLicense' is already defined in class 'HSS-License'. Skipping it
WARNING Attribute 'HSS-GbaLicense' is already defined in class 'HSS-License'. Skipping it
WARNING Attribute 'HSS-ExtDbConfigLogActive' is already defined in class 'HSS-ExtDbConfig'. Skipping it
WARNING Attribute 'HSS-ExtDbConfigUrrList' is already defined in class 'HSS-ExtDbConfig'. Skipping it
WARNING Attribute 'HSS-ExtDbMaxConcurrentClients' is already defined in class 'HSS-ExtDbConfig'. Skipping it
WARNING Attribute 'HSS-ExtDbConfigOrigVipList' is already defined in class 'HSS-ExtDbConfig'. Skipping it
WARNING Attribute 'HSS-ExtDbConfigRootOnList' is already defined in class 'HSS-ExtDbConfig'. Skipping it
```

Example 1 Sample Schema Validation Report

For a description of the messages presented in schema validation reports, refer to MIM Validation Errors in the Appendix.

- 4. If required, grant permission for other users to work with the new schema by modifying the ACL. For more information on granting ACL permissions, refer to Section 3.1.1 on page 10.

7.3 Working with Schema Revisions

Schemas stored in PDB are revision controlled. A given revision will precede or supersede other revisions of the same configuration schema, allowing for a number of revision-specific operations.

These operations include:



- Changing the Document State of a Configuration Schema
- Importing a New Schema Revision

By default, only the latest revision of each schema is displayed in the **Schemas** table. Older revisions are accessible through the revision history or by refining the search criteria. For more information on performing a search, refer to Section 7.1.1 on page 43.

To view all revisions of a selected schema, right-click to open the context menu and select **Revision > Show Revision History**.

PDB requires all predecessors to be in a non-preliminary state (FROZ, FREE, WIDR) before subsequent revisions can be created.

7.3.1 Document States

PDB uses a system of document states to provide information on the completeness, quality, and approval status of a particular schema revision. The document state is indicated by a status code that is part of the schema metadata. PDB uses the following document states for node configurations:

PREL	Preliminary. Used to designate an unlocked schema that is still under development. Note: Although schema data cannot be edited in PDB, schemas in the PREL state cannot serve as a basis for new revisions.
FROZ	Frozen. Used to designate a frozen schema where the content has been locked to prevent further changes. This is the basic state for schema revisions that are stored in PDB but are not approved for release.
FREE	Approved. Used to designate schemas that are approved for release. Approved schemas are locked to prevent any changes to the released content.
WIDR	Withdrawn. Used to designate schemas that are no longer intended for use. Withdrawn schemas are still accessible in PDB, but are locked to prevent further changes.

7.3.1.1 Changing the Document State

PDB allows you to change the document state of configuration schemas that you have permission to modify. For more information on ACL permissions, refer to Section 3.1 on page 9.

Schema revisions must follow the following sequence of document states:

PREL > FROZ > FREE > WIDR



Note: Only PDB System Administrators can set a schema to a previous state.

New schemas, or schema revisions start in the PREL state where the associated metadata can be modified and updated. The document state is changed as the schema progresses through its lifecycle.

To change the document state of a schema revision:

1. In the **Schemas** workspace, right-click a schema and select **Revision > Set to <STATE>**.

Note: If the schema is not visible, perform a search as outlined in Section 7.1.1 on page 43.

When moving from PREL to FROZ, the **Freeze** dialog box opens. See Figure 26.

The screenshot shows a dialog box titled "Freeze". It contains the following fields and controls:

- Schema Name:** MTAS_11A_CM_MIM
- Revision:** PA2
- Update Revision Comment:** A large text area with a vertical scrollbar.
- Footer:** A note stating "* Once a schema has been frozen it cannot be changed." and two buttons: "Freeze" and "Cancel".

Figure 26 Freeze Dialog Box

The Freeze dialog box allows you to modify the revision level of the schema. Here preliminary revisions can be set to a solid state before freezing.

Stop!

Only PDB System Administrators can unfreeze a schema. Do not continue if additional changes are required to the schema metadata.

2. Click **Freeze**.

The selected schema is locked and the document state is set to FROZ.

For all other state transitions a confirmation dialog box is displayed.

Click **OK** to change the document state.

7.3.2 Importing a New Schema Revision

A revised set of files can be imported to create a new schema revision.

Although a new schema revision supersedes previous revisions of the same configuration schema it does not automatically change any relationships between the former schema revision and node configurations. In order to base existing node configurations on a new schema revision, you must trigger the schema migration process from the **Configurations** workspace. For more information on migrating configurations to another schema, refer to Section 8.7 on page 99.

To import a new schema revision:

1. In the **Schemas** workspace, right-click a non-preliminary schema and select **Revision > Import New Revision**.

The **Import New Revision** dialog box opens with the next legal revision displayed. See Figure 27.

Figure 27 Import New Revision Dialog Box

2. Verify that the proposed revision is correct and update as necessary.
3. Enter the path and file name of the schema archive.
4. Click **Import**.

PDB performs the following operations when importing a new configuration schema.

- a PDB imports the data contained within the selected archive and generates a new configuration schema revision.
- b PDB performs a validation check during the import process. For more information on the validation check, refer to Section 7.2 on page 45.
- c PDB automatically generates an import validation report when problems are found. Errors and warnings that were found during validation are reported for each affected schema file. This report can be downloaded from the **Report Available** dialog box that appears at the end of the import process.



- d PDB stores the schema archive and the import validation report. These files can be downloaded from the properties dialog box for the new schema.

7.4 Modifying Schema Properties

While in PREL state, PDB allows you to modify the following schema properties:

- Names
- Document numbers

If no document number is specified, PDB automatically generates an internal number.

Note: Document numbers are validated using Ericsson's standard rules for registration notation.

Subsequent revisions of an existing schema are locked to the document number of their predecessor.

- Descriptions
- Initial Value List Associations

For more information on associating an Initial Value List (IVL) with a schema, refer to Section 7.4.1 on page 53.

If the schema is in a locked state, only the **Description** and **Revision Comment** fields can be modified.

To edit schema properties:

1. In the **Schemas** workspace, right-click a schema in PREL state and select **Edit**.

Note: If the schema is not displayed, perform a search as outlined in Section 7.1.1 on page 43.

The **Edit** dialog box opens.

2. Update the schema properties as required.
3. Click **Apply**.

The updated properties are saved.

7.4.1 Associating an Initial Value List with a Schema

An Initial Value List (IVL) represents the configuration of an LDAP node after maiden installation. When exporting a node configuration in LDIF format, PDB will use IVL data associated with the configuration schema to produce



configuration files that do not collide with the configuration values that are assumed to already exist in the real node.

Node configurations stored in PDB must be tagged as IVL before they can be associated with a schema. For more information on tagging configurations as IVL, refer to Section 8.6 on page 97.

To associate an with a configuration schema:

1. In the **Schemas** workspace, right-click a schema in PREL state and select **Edit**.

Note: If the schema is not displayed, perform a search as outlined in Section 7.1.1 on page 43.

The **Edit** dialog box opens.

2. Under **Initial Value List**, Select an IVL configuration to associate with the schema.
3. Click **Apply**.

The updated properties are saved.

7.5 Exporting Schematron Rules

Schematron rules are used in some NETCONF-based configuration schemas to provide an additional layer of validation. These rules constrain patterns in the configuration XML, ensuring, for example, that related parameter values correspond with each other.

When a configuration schema stored in PDB contains schematron rules, these rules can be exported to an XML file.

To export schematron rules:

1. In the **Schemas** workspace, right-click a schema containing schematron rules and select **Export Schematron**.

Note: If the schema is not displayed, perform a search as outlined in Section 7.1.1 on page 43.

The **Export** dialog box opens.

2. Click **Download Schematron**.

An XML file containing the schematron rules is downloaded. By default, the downloaded XML file uses the following naming convention:

```
<schema_name>[<revision>]_schematron.xml
```

Where:



<schema_name Is the name of the configuration schema that the rules are exported from.

<revision> Is the revision of the selected schema in PDB.

7.6 Deleting a Schema

Authorized users can remove schemas from PDB as needed.

Note: PDB cannot remove older schema revisions or any schema that has been associated with node revisions. In these cases, you must ensure that the configuration schema is no longer being referenced within PDB before attempting to delete it.

To remove a schema from PDB:

1. In the **Schemas** workspace, right-click a schema and select **Delete**.

Note: If the schema is not displayed, perform a search as outlined in Section 7.1.1 on page 43.

A confirmation dialog box opens.

Caution!

Deleting a schema permanently removes it from PDB.

2. Click **OK**.

The configuration schema is deleted.

7.7 Viewing Schema Elements

Configuration schemas are composed of parameters and parameter groups that are used to define a valid structure for the associated configurations. Where a specific element exists in the schema, a corresponding element can exist in a configuration using that schema. Each element (parameter or parameter group) is defined with cardinality and value constraints that ensure a configuration will be compatible with the destination node.

To facilitate working with this information, PDB provides a dedicated browser for schema elements. The schema browser is described in Section 7.7.1 on page 55.

7.7.1 The Schema Browser

The schema browser allows users to view the structure of schema elements in PDB.



Schema elements can be displayed in two possible formats:

- TREE view (Default)
- TABLE view

In the TREE view, PDB displays schema elements as a cascading tree of parameter groups, each group containing one or more parameters.

See Figure 28.

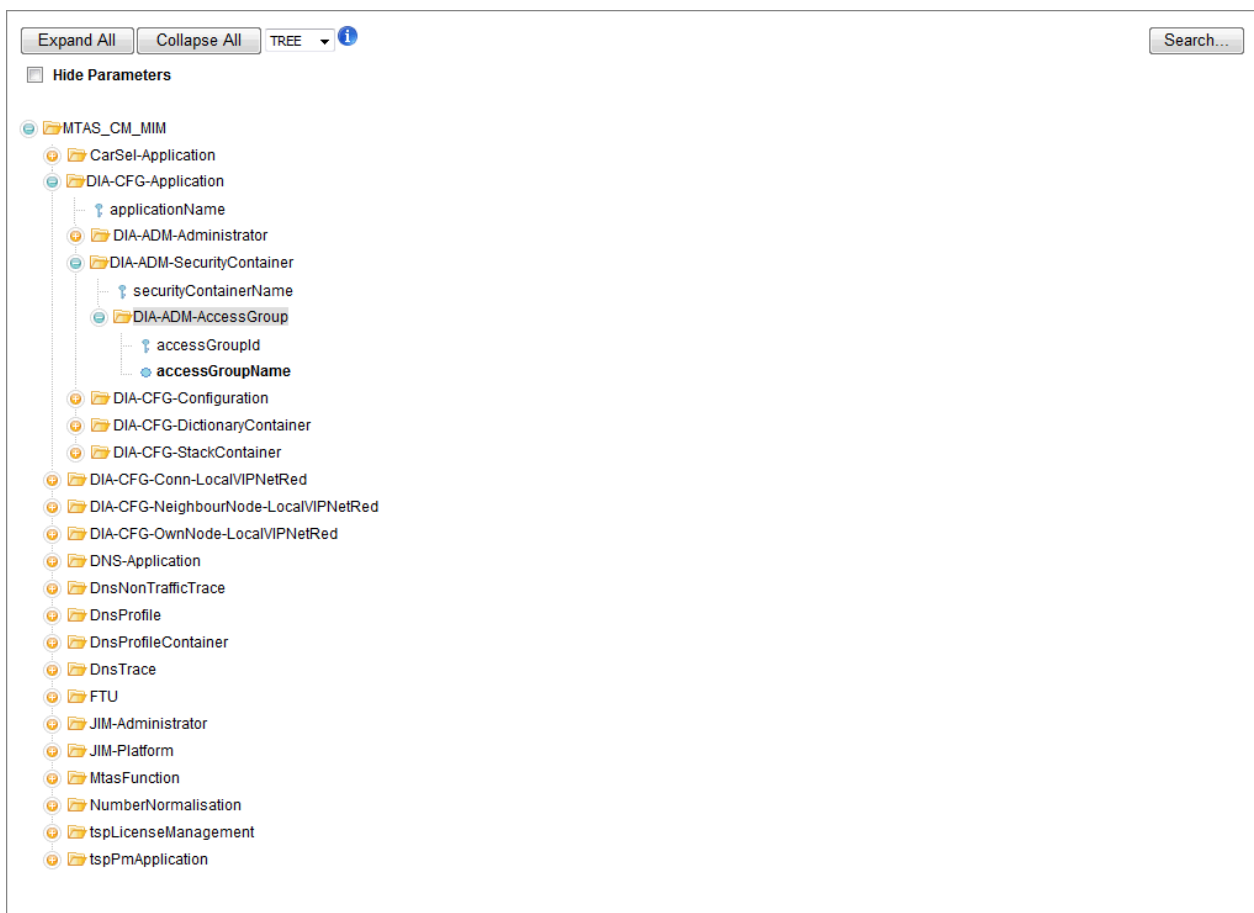


Figure 28 Schemas - Tree View

In the TABLE view, PDB displays schema elements as a table of nested parameter groups, each containing one or more parameters. The table includes the following information, where applicable:

- MOC

The Managed Object Class (MOC) shows the path of the configuration element in the configuration tree.



Note: Only the last section of the MOC is shown in table view. The full path of the schema element is available as a tooltip.

- Name
- Description

See Figure 29.

Expand All	Collapse All	TABLE	?	Search...
MOC	Name	Description		
MTAS_CM_MIM				
CarSel-Application	CarSel-Application	The Carrier Select Managed Object (MO) represents the top level of the Carrier Select application		
DIA-CFG-Application	DIA-CFG-Application			
	applicationName	Application key attribute used in the Managed object instance		
DIA-ADM-Administrator	DIA-ADM-Administrator			
DIA-ADM-SecurityContainer	DIA-ADM-SecurityContainer			
	securityContainerName			
DIA-ADM-AccessGroup	DIA-ADM-AccessGroup			
	accessGroupId			
	accessGroupName			
DIA-CFG-Configuration	DIA-CFG-Configuration			
DIA-CFG-DictionaryContainer	DIA-CFG-DictionaryContainer			
DIA-CFG-StackContainer	DIA-CFG-StackContainer			
DIA-CFG-Conn-LocalVIPNetRed	DIA-CFG-Conn-LocalVIPNetRed			
DIA-CFG-NeighbourNode-LocalVIPNetRed	DIA-CFG-NeighbourNode-LocalVIPNetRed			

Figure 29

To switch between TREE and TABLE views, make a selection from the **Format** drop-down list at the top of the page. Selected elements are not affected by switching between views.

To browse a schema in PDB:

- In the **Schemas** workspace, select a schema to browse and click **Open**.

Note: If the schema is not displayed, perform a search as outlined in Section 7.1.1 on page 43.


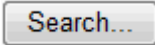
Elements of the selected schema are displayed in the schema browser.



To facilitate working with large schemas, the schema browser includes a search functionality that allows you to filter the schema for specific elements. For more information on the search functionality, refer to Section 7.7.1.1 on page 58.

The following table describes the elements of the schema browser:

Table 22 Schema Browser Elements

Element	Description
	Displays information about the selected schema element.
Format (List)	Sets the display format of the schema. TREE: Displays schema elements as a cascading tree of parameter groups, each containing one or more parameters. TABLE: Displays schema elements as a table of nested parameter groups, each containing one or more parameters.
Hide Parameters	TREE view only. Hides parameters in the element tree, allowing you to quickly navigate the parameter groups in the schema
	Perform a search of the current schema.

7.7.1.1 Searching Configuration Schemas

The schema browser includes a search functionality that locates specific elements using a number of different criteria.

The search field accepts keywords to narrow down the search results. The syntax for using keywords is as follows:

<keyword1>:value1 <keyword2>:value2 <keyword3>:value3 ...

The following table describes the available keywords.



Table 23 Search Keywords

Keyword	Description
type: <type>	Searches for schema elements that match the selected type. Valid types include: <ul style="list-style-type: none"> • p • pg
name: <string>	Searches for parameter and parameter group names that contain the input string. Searching on element names is the default search operation and is performed when no search criteria are specified.
description: <string>	Searches for parameter and parameter group descriptions that contain the input string.
category: <category>	Searches for parameters and parameter groups matching the selected category. Valid categories include: <ul style="list-style-type: none"> • Internal • Operator Configurable • Site Specific • Solution Integration



Keyword	Description
iskey: <boolean>	<p>Filters parameters for primary keys.</p> <ul style="list-style-type: none">• true - Filters out parameters that are not primary keys.• false - Filters out parameters that are primary keys. <p>Note: Because primary keys are a type of parameter, parameter groups will not be included in the search results when iskey is set to true.</p>
status: <status>	<p>Searches for parameters and parameter groups matching the selected status. Valid status types include:</p> <ul style="list-style-type: none">• current• deprecated• obsolete
sdn: <string>	<p>Searches for Schema Distinguished Names (SDN) that contain the input string.</p>



Keyword	Description
readonly: <boolean>	<p>Filters parameters based on the <code>readonly</code> attribute.</p> <ul style="list-style-type: none"> • <code>true</code> - Filters out parameters that are not read-only. • <code>false</code> - Filters out parameters that are read-only. <p>Note: Because parameter groups do not have a <code>readonly</code> attribute, they are included in all searches using this keyword unless they are removed by other search criteria.</p>
restricted: <boolean>	<p>Filters parameters based on the <code>restricted</code> attribute.</p> <ul style="list-style-type: none"> • <code>true</code> - Filters out parameters that are not restricted. • <code>false</code> - Filters out parameters that are restricted. <p>Note: Because parameter groups do not have a <code>restricted</code> attribute, they are included in all searches using this keyword unless they are removed by other search criteria.</p>

To perform a search:

1. In the schema browser, click **Search**.

The **Search** dialog box is displayed. See Figure 30.

Figure 30 Search Box

2. Enter your search query.

For a basic search, type the name of the schema element you are searching for.

Note: Search strings are not case-sensitive. All searches return a partial match unless the string is surrounded by double quotes.



For a keyword search, compose a search string using one or more valid keywords. See Table 23 for a description of the available search options.

Note: Different keywords can be combined to refine the search pattern.

Some sample search queries, include:

- **name: <string> type:P**

This query would search for parameters with a name matching the input string.

- **name: <string> type:PG status:deprecated**

This query would search for parameter groups with a name matching the input string that are currently set as deprecated.

- **description: <string>**

This query would search for parameters or parameter groups that contain a description matching the input string.

- **sdn:Root[RootPrimaryKey=0],A1[A1PrimaryKey=0] type:P**

This query would search for parameters located at the specified location in the schema.

3. Click **Search**.

The search hits are listed in the table at the bottom of the **Search** dialog box. See Figure 31.

Search results are listed with the following information, where applicable:

- SDN
- Parameter Name

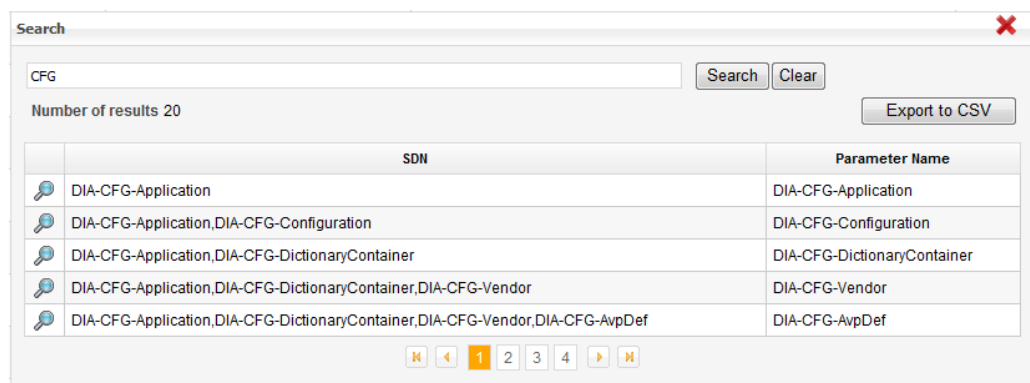



Figure 31 Search Results



4. Click the magnifying glass () next to one of the search results to open the schema at that element's location.

The search is complete.

Search results are available in CSV format. Click **Export to CSV** to download the CSV file.

7.8 Comparing Two Schemas

Using PDB you can compare two configuration schemas to identify certain differences between them. Both parameters and parameter groups are analyzed in a comparison and the following differences can be highlighted:

Table 24 Schema Comparison Properties

Property	Description
Missing	The element is missing from the schema.
Default Values	Differences in the default value of the schema elements.
Category	Differences in the category of the schema elements.
Format Description	Differences in the format description of the schema elements.
Name Case Differences	Differences in the letter case of the element names.
Status	Differences in the status of the schema elements.
Read Only	Differences in the read/write status of the schema elements.
Constraints	Differences in the value constraints of the schema elements.
Primary Key	Differences in the primary key designation of the schema elements.
Cardinality	Differences in the cardinality of the schema elements.
Description	Differences in the description of the schema elements.

The schema comparison is useful for identifying these differences between revisions of the same schema.

When performing a comparison, PDB distinguishes between a Base Schema and a Target Schema. The Base Schema serves as the starting point for a comparison. It



is contrasted with a Target Schema that may be a previous revision of the Original Schema or a separate schema that has been imported to PDB.

Shortcuts in the **Schemas** context menu (under **Compare > Compare with Previous** and **Compare > Compare with Another**) allow you to generate comparison reports.

Comparison results are output to a table in the **Schema Comparison** dialog box. This table lists the differences that have been identified between the two schemas. The schema comparison can be downloaded as a Comma Separated Values (CSV) file.

To compare two schemas:

1. In the **Schemas** workspace, right-click a schema and select one of the following:

- **Compare > Compare with Previous**

Selecting **Compare with Previous** opens the **Schema Comparison** dialog box with the previous revision of the base schema selected for comparison.

- **Compare > Compare with Another**

Selecting **Compare with Another** opens the **Schema Comparison** dialog box to select the Target Schema. See Figure 32.

Schema Comparison

Search

Node Node Revision

Document Number User Name Revision ☐ Latest Only

Document States

☒ PREL ☒ FROZ

☒ FREE ☒ WIDR

Select Target Schema

Base Schema TSP New Format[PA2]

Target Schema

Report the following types of differences:

☒ Missing ☒ Name Case Differences ☒ Primary Key

☒ Default Values ☒ Status ☒ Cardinality

☒ Category ☒ Read Only ☒ Description

☒ Format Description ☒ Constraints

Figure 32 Schema Comparison Dialog Box

2. If required, update the search filters to find the target schema. For more information on the search criteria, refer to Section 7.1.1 on page 43.
3. Select the target schema and the properties to compare. For more information on the comparison properties, see Table 24

4. Click **Compare**

PDB performs the selected comparison and displays the results. See Figure 33.

The schema comparison is complete.

Schema Comparison

Base Schema TSP New Format[PA2]
Target Schema TSP New Format[PA1]

CloseExport to CSV

Number of differences 12

Type	Name (2)		Is Key (1)		Default Values (1)		Status (1)		Cardinality (1)		Category (1)		Is Read Only (1)		Description (2)		Format Description (1)		Constraints (1)		MOC	SON	
	Base	Target	Base	Target	Base	Target	Base	Target	Base	Target	Base	Target	Base	Target	Base	Target	Base	Target	Base	Target		Base	Target
A	RootPrimaryKey	RootPrimaryKey	true	true	1	0	current	current			Site Specific	Site Specific	true	true	This is the primary key of Root	This is the primary key of Root	Case sensitive string	Case sensitive string	None	None	Root	Root	Root
A	[not present]	Param1																			Root	[not present]	Root
A	Param2	Param2	false	false	=	=	deprecated	current	0-1	0-2147483647	Internal	Operator Configurable	true	false	This is a description of Param2	This is the description of Param2	Only digits string, length of string from 1 to 11	Only digits string, length of string from 1 to 10	Pattern: ([0-9]{1,11}) Minimum length: 1 characters Maximum length: 11 characters	Pattern: ([0-9]{1,10}) Minimum length: 1 characters Maximum length: 10 characters	Root	Root	Root
A	Param1	Param1	true	false	0	0	current	current		0-2147483647	Operator Configurable	Operator Configurable	false	false	This is the description of Param1	This is the description of Param1	Values 0, 1 or 2	Values 0, 1 or 2	Enumeration (0 1 2)	Enumeration (0 1 2)	A1	Root.A1	Root.A1
A	Param1	Param1	false	false	0	0	current	current	0-2147483647	0-2147483647	Operator Configurable	Operator Configurable	false	false	Hover over cell or export to CSV for full text. This is the description of Param1	This is the description of Param1	Values 0, 1 or 2	Values 0, 1 or 2	Enumeration (0 1 2)	Enumeration (0 1 2)	A2	Root.A1.A2	Root.A1.A2
A	param1	Param1	false	false	0	0	current	current	0-2147483647	0-2147483647	Operator Configurable	Operator Configurable	false	false	This is the description of Param1	This is the description of Param1	Values 0, 1 or 2	Values 0, 1 or 2	Enumeration (0 1 2)	Enumeration (0 1 2)	D3	Root.D1.D2.D3	Root.D1.D2.D3

Figure 33 Schema Comparison Report

The schema comparison report is color coded as follows:

— Red highlights missing elements.

For example: [not present]

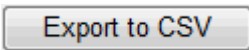
— Orange highlights value differences.

For example: param1

Different shades of the same color are used to distinguish separate rows and do not mark additional information.

The following table describes the different elements of a **Schema Comparison**:

Table 25 Schema Comparison Elements

Element	Description
	Exports the Schema Comparison Report to a CSV file that must be downloaded from the PDB server.
Type	The classification of the schema element, Class (C) or Attribute (A).



Element	Description
Name	The name of the schema element. Note: A value of [not present] means the element is missing from the schema.
Is Key	Boolean value marking whether or not the schema element is a primary key.
Default Values	The default values for the element as defined in the schema.
Status	The status of the element as defined in the schema. Possible values include: <ul style="list-style-type: none">• current• deprecated• obsolete
Category	The category of the schema element. Possible values include: <ul style="list-style-type: none">• Internal• Operator Configurable• Site Specific• Solution Integration
Is Read Only	Boolean value marking whether or not the schema element is read-only.
Description	The description of the element as defined in the schema. ⁽¹⁾
Format Description	The format description of the element as defined in the schema. ⁽¹⁾
Constraints	The value constraints for the element as defined in the schema.
MOC	The Managed Object Class (MOC) that the element belongs to.
SDN	The Schema Distinguished Name of the element.

(1) To constrain the size of the comparison report, descriptions longer than 50 characters are automatically hidden. Hidden descriptions are identified with descriptive text. To see the full text, hover over a hidden description or export the report to CSV.



Note: Values in the table headings mark the number of differences captured in that column.

8 Working with Node Configurations

A node configuration is an organized collection of parameter instances and values that define configuration settings for a specific node. In PDB, node configurations can be created from scratch or imported from configuration files.

PDB can import configuration data in the following file formats:

- LDIF
- NETCONF
- Parameter Value List (PVL)

Note: PDB supports multi-solution configurations. For more information on working with multi-solution configurations, refer to Section 11 on page 159.

PVL formatted files are typically generated by node development organizations while LDIF and NETCONF formatted files are typically extracted from service nodes in the field. For more information on importing node configurations, refer to Section 8.2 on page 74.

Node configurations must be based on a configuration schema. The schema acts as a template and constrains the content of the associated configurations. The use of schemas ensures the validity of configurations by preventing the entry of invalid data. For more information on configuration schemas, refer to Section 7 on page 40.

Once a node configuration has been created in PDB, you can work with it in a number of ways including:

- Managing Configuration Data
- Creating a Delta Configuration
- Cloning a Configuration
- Merging Node Configurations
- Migrating a Configuration to Another Schema
- Comparing Two Configurations



- Validating a Configuration
- Exporting a Node Configuration

PDB keeps an Access Control List (ACL) for each node configuration. All non-administrative users require ACL permissions to view or modify the associated configuration. For more information on ACLs, refer to Section 3.1 on page 9.

All node configurations stored in PDB are revision controlled. PDB validates revision levels following Ericsson's standard rules for document handling. A given revision will precede or supersede other revisions of the same configuration; this relationship opens a number of operations when working with a revised node configuration. These operations include:

- Comparing Different Revision Levels
- Changing the Document State
- Creating a New Revision
- Importing a New Revision
- Rebasing a Delta Configuration

Node configurations can be associated with one or more node revisions that have been defined in PDB. This association ultimately helps to link the configuration to a baseline revision. For more information on associating a configuration with node revisions, refer to Section 5.1.4 on page 23.

8.1 The Configurations Workspace

The **Configurations** workspace allows you to carry out tasks with node configurations.

To access the **Configurations** workspace, select **Configurations** from the menu options on the left. See Figure 34.



Configurations

Node

All

Node Revision

All

Solution baseline

All

Document Number

User

Name

Revision

☒ Latest Only

Document States

☒ PREL

☒ FROZ

☐ IVL

☐ IFN

☒ FREE

☒ WIDR

☐ MPVL

☐ DPVL

<<

Search

Clear

>>

Add to Solution Baseline

Refresh

Import...

Create Empty...

Merge...

Show Details

Name	Action	Document Number	Revision	Schema	Tags	Document State	Node	Node Revision
/CSCF_11B_R4B_LDIF_NoEmptyValues	Open	2/190 73-CXP 902 0350/1	A	CSCF_CM_MIM[A]	IVL MPVL	FROZ	CSCF 11B	R5B
/CSCF_11B_R4G_LDIF	Open	1/190 73-CXP 902 0350/1	A	CSCF_CM_MIM[A]	IFN	FREE	CSCF 11B	
/HSS_11A_R9B_LDIF/HSS_11A_R9B_NoEmptyValues	Open	3/190 73-CXP 901 1865/6	PA1	HSS_CM_MIM[A]	DPVL	PREL	HSS 11A	
/MPVL_MTAS_11B_Converged	Open	19/190 73-CXP 901 1263/5	PA2	MTAS_11B_CM_MIM[PA4]	MPVL	FROZ	MTAS 11B	R6D
/MPVL_MTAS_11B_Fixed	Open	17/190 73-CXP 901 1263/5	PA4	MTAS_11B_CM_MIM[PA4]	IVL MPVL	FROZ	MTAS 11B	
/MPVL_MTAS_11B_Mobile	Open	18/190 73-CXP 901 1263/5	PA2	MTAS_11B_CM_MIM[PA4]	MPVL	FROZ	MTAS 11B	R6D
/MTAS_11A_R1D_LSV118_LDIF	Open	4/190 73-CXP 901 1263/5	PA2	MTAS_CM_MIM[PA1]		PREL	MTAS 11A	R4A LSV134

Figure 34 The Configurations Workspace

The workspace is divided into two principal areas as follows:

Search

The search options, located at the top of the page, allow you to filter the configurations that are displayed in the **Configurations** table. For more information on performing a search, refer to Section 8.1.1 on page 72.

Configurations Table

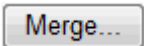
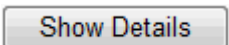

The **Configurations** table is the centerpiece of the **Configurations** workspace. This table displays node configurations that match the selected search criteria and allows you to perform a number of configuration-specific operations using a context menu.

The following table describes the elements of the **Configurations** workspace.

Table 26 Configurations Workspace Elements

Element	Description
<div>Add to Solution Baseline</div>	Adds a node configuration to a solution baseline. For more information on adding configurations to a baseline, refer to Section 6.2.2 on page 34.
<div>Import...</div>	Imports node configuration data. PDB automatically validates the imported configuration against the associated schema and generates a report.
<div>Create Empty...</div>	Generates an empty configuration from a selected schema.



Element	Description
	Merges the contents of two or more configurations.
	Expands the Configurations table to show the User and Last Modification columns.
Name	<p>The name of the configuration.</p> <p>Note: PDB uses a path-based naming convention for configurations. In this format, a configuration name is appended to the name of any parent configurations in the following structure:</p> <pre>/<root configuration>/<configuration 1>/...</pre>
Action	Provides an Open button for quick access to configuration data. The drop-down arrow opens a context menu for the configuration.
Document Number	The document number associated with the node configuration.
Revision	The revision of the configuration.
Schema	<p>The configuration schema.</p> <p>Click  to open the configuration schema in the Schemas workspace.</p>



Element	Description
Tags	<p>Displays tags associated with each node configuration. PDB supports configuration tagging to categorize and add prominence to certain node configurations. The following tags are available:</p> <ul style="list-style-type: none"> • IVL - The IVL tag marks the configuration as an Initial Value List. • IFN - The IFN tag marks the configuration as Imported From Node. • MPVL - The MPVL tag marks the configuration as a Master Parameter Value List. • DPVL - This DPVL tag marks the configuration as a Delta Parameter Value List. <p>For more information on tagging node configurations, refer to Section 8.6 on page 97.</p>
Document State	<p>Displays the document state of the configuration. The following document states are available:</p> <ul style="list-style-type: none"> • PREL - Preliminary • FROZ - Frozen • FREE - Approved • WIDR - Withdrawn <p>For more information on document states, refer to Section 8.5.1 on page 93.</p>
Node	The node associated with the configuration schema.
Node Revision	A comma-separated list of node revisions that have been associated with the configuration. For more information on associating a configuration with node revisions, refer to Section 5.1.4 on page 23.



Element	Description
User	The user who created the configuration in PDB. ⁽¹⁾
Last Modification	A timestamp marking when a configuration was last modified. ⁽¹⁾

(1) This column is normally hidden. Click **Show Details** to display this information.

Each configuration listed in the **Configurations** table is selectable. Right-clicking a configuration opens a context menu where operations specific to the selected configuration can be executed.

8.1.1 Searching for Node Configurations

The **Configurations** table is empty by default and is populated with node configurations by performing a search. A number of search criteria are available to help you find specific configurations. PDB reports partial matches on search strings. Use double quotes "<" ">" to restrict the search to exact matches.

Searches are performed using the search workspace at the top of the **Configurations** page. See Figure 35.

Figure 35 Configurations Search

The following table describes the available search criteria.

Table 27 Configuration Search Elements

Element	Description	Notes
Node (List)	Filters the table for configurations that are associated with the selected node.	
Node Revision (List)	Filters the table for configurations that are associated with the selected node revision.	The Node Revision list is only populated after a selection has been made from the Node list.
Solution Baseline (List)	Filters the table for configurations that are part of the selected solution baseline.	



Table 27 Configuration Search Elements

Element	Description	Notes
Document Number	Filters the table for configurations with document numbers that match the selected criteria.	
User	Filters the table for configurations that were created by the specified user.	
Name	Filters the table for configurations with names that match the selected criteria. All entries are case sensitive.	
Revision	Filters the table for configurations with revisions that match the selected criteria.	Latest Only must be deselected to perform a search using this field.
Latest Only	Includes only the latest configuration revisions that match the other search criteria.	If this option is selected, search results are restricted to the latest revision. Older revisions can be accessed by deselecting this option or by using revision history. To view all revisions of a selected configuration, right-click to open the context menu and select Revision > Show Revision History .
Document States	Filters the table for configurations with the selected document states.	
Tags	Filters the table for configurations with the selected tags.	

All search filters are additive. Using multiple filters will narrow the search results.

To filter the **Configurations** table:

1. In the **Configurations** workspace, set your search criteria.
2. Click **Search**.

The **Configurations** table is populated with configurations that match the selected criteria.



Note: PDB automatically stores up to 10 consecutive searches. Use the navigation buttons to move between each search.

Search results are retained as you navigate through the web portal. To reset the **Configurations** table to the default display, click **Clear** then click **Search**.

Each node configuration has a URL. This link provides an external reference to the specific configuration. Following a direct link connects you to the PDB server. After logging in, PDB automatically loads the **Configurations** workspace and shows the linked configuration.

URLs are automatically generated by PDB. To access a URL and other properties, right-click a configuration and select **Properties**.

8.2 Importing a Node Configuration

Most of the configuration data included in PDB has been imported from configuration files.

PDB can parse configuration data in the following formats:

CPP_CSV	The working configuration of a Connectivity Packet Platform (CPP)-based node can be exported in comma-separated values (CSV) format for use in PDB.
LDIF	The working configuration of a TSP-based node can be exported in the LDAP Data Interchange Format (LDIF) for use in PDB.
NETCONF	The working configuration of an IS or CBA-based node can be exported in the NETCONF format for use in PDB.
PVL	The Parameter Value List (PVL) format used by Node Development Organizations (NDOs) to deliver official configurations. For more information on the PVL format, refer to the Parameter List Template Description, EAB/FTI-08:0686 Uen.

Configuration files can be packaged in a ZIP or TAR archive, or imported directly from LDIF or XML files.

Note: An archive directory structure is not required by PDB to import configurations; however, if directories are present inside the file, then PDB will recursively scan for configuration files within the directories as needed.

To import a node configuration:

1. In the **Configurations** workspace, click **Import**.



The **Import** dialog box opens. See Figure 36.

Import

Import Type

PVL

Import From

FIXED

☐ Use Delta

Schema

CSCF_CM_MIM[A]

☒ Latest Schema Revisions Only

Configuration Name

Document Number

Description

Revision

PA1

Revision Comment

IVL

☐

IFN

☐

MPVL

☐

Input File

Browse...

Uploaded File

Import

Cancel

Figure 36 Import Dialog Box

The following table describes the elements of the **Import** dialog box:

Table 28 Import Dialog Box Elements

Element	Description	Notes
Import Type (List)	The configuration format.	Mandatory.
Import From (List)	For PVL input files only. The parameter value set to import.	Mandatory. Select Use Delta to give precedence to delta values over the chosen parameter value set.



Table 28 Import Dialog Box Elements

Element	Description	Notes
Schema (List)	<p>The schema to use as a template for the incoming configuration data.</p> <p>The list shows both the schema name and revision (in square brackets).</p> <p>By default, the list of schemas is filtered to display only the latest revisions. Clear Latest Schemas Only to display the complete list.</p>	<p>Mandatory.</p> <p>An appropriate schema must exist in PDB before it can be associated with a node configuration. For more information on importing schemas, refer to Section 7.2 on page 45.</p>
Configuration Name	The name of the configuration.	Mandatory.
Document Number	The document number of the configuration.	<p>Optional.</p> <p>If left blank, PDB will automatically generate an internal document number for the new configuration. This number can be modified by editing the configuration properties as described in Section 8.6 on page 97.</p>
Description	A short description of the configuration.	Optional.
Revision	The revision of the configuration.	<p>Mandatory.</p> <p>During the same session, PDB will automatically populate this field with the revision set in the previous import operation.</p> <p>The next expected revision level is suggested to the right of the Revision field.</p>
Revision Comment	Comments on this particular revision of the configuration.	Optional.
IVL	Marks the new configuration as an Initial Value List (IVL).	Optional.
IFN	Marks the new configuration as Imported From Node (IFN).	Optional.



Table 28 Import Dialog Box Elements

Element	Description	Notes
MPVL	Marks the new configuration as a Master Parameter Value List (MPVL).	Optional.
Input File	The path and file name of the configuration archive.	Mandatory.

2. Enter the required information. See Table 28.
3. Click **Import**.

PDB performs the following operations when importing a new configuration.

- a PDB imports the data contained within the selected archive and generates a new configuration.
 - b PDB automatically performs a validation check during the import process. This check verifies the syntax of the configuration files and validates the configuration against the selected schema.
 - c PDB generates an import validation report that can be downloaded from the **Report Available** dialog box that appears at the end of the import process. For a description of the messages presented in import validation reports, refer to Configuration Validation Errors in the Appendix.
 - d PDB stores the configuration archive and the import validation report. These files can be downloaded from the properties dialog box for the new configuration.
4. If required, grant permission for other users to work with the new configuration by modifying the ACL. For more information on granting ACL permissions, refer to Section 3.1.1 on page 10.

8.3 Creating a New Node Configuration

PDB allows you to create node configurations without importing new data. These configurations are either built from scratch or based on one or more existing configurations.

The following tasks will create a configuration without importing new files:

- Creating an Empty Configuration
- Cloning a Configuration
- Merging Node Configurations

Creating new revisions of an existing configuration is part of revision handling and is fully discussed in Section 8.5 on page 92.



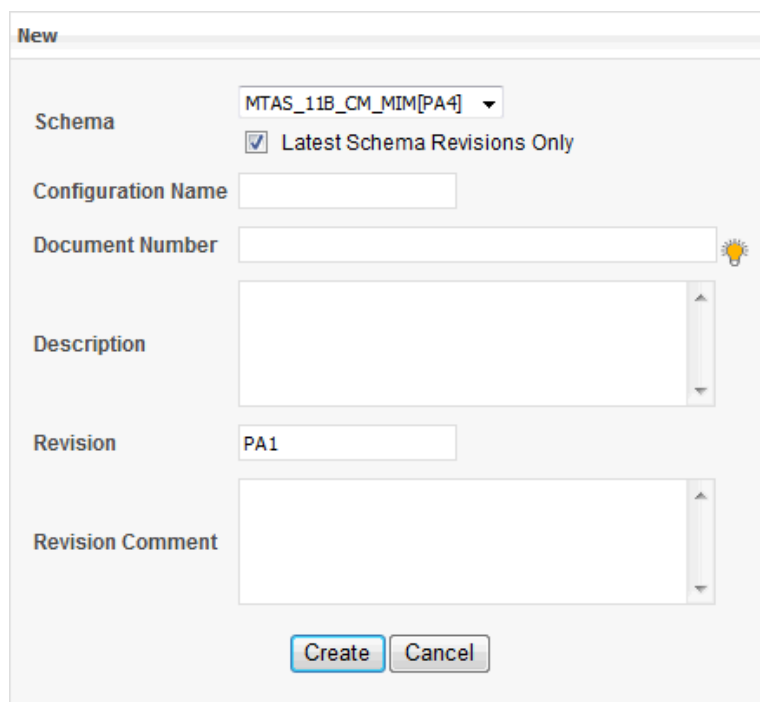
8.3.1 Creating an Empty Configuration

Empty configurations do not contain any configuration elements when they are first created and must be built from scratch. Data must be manually added to the configuration using the configuration browser. For more information on working with configuration data, refer to Section 9 on page 114.

To create an empty configuration:

1. In the **Configurations** workspace, click **Create Empty**.

The **New** dialog box opens. See Figure 37.



The 'New' dialog box is a light gray window with a title bar. It contains several input fields and a checkbox. The 'Schema' field is a dropdown menu showing 'MTAS_11B_CM_MIM[PA4]'. Below it is a checkbox labeled 'Latest Schema Revisions Only' which is checked. The 'Configuration Name' field is a text box. The 'Document Number' field is a text box with a lightbulb icon to its right. The 'Description' field is a large text area. The 'Revision' field is a text box showing 'PA1'. The 'Revision Comment' field is a large text area. At the bottom are 'Create' and 'Cancel' buttons.

Figure 37 New Dialog Box

The following table describes the elements of the **New** dialog box:



Table 29 New Dialog Box Elements

Element	Description	Notes
Schema (List)	<p>The schema to use as a template for the configuration data.</p> <p>The drop-down list shows both the schema name and revision (in square brackets).</p> <p>By default, the list of schemas is filtered to display only the latest revisions. Clear Latest Schemas Only to display the complete list.</p>	<p>Mandatory.</p> <p>An appropriate schema must exist in PDB before it can be associated with a node configuration. For more information on importing schemas, refer to Section 7.2 on page 45.</p>
Configuration Name	The name of the configuration.	Mandatory.
Document Number	The document number of the configuration.	<p>Optional.</p> <p>If left blank, PDB will automatically generate an internal document number for the new configuration. This number can be modified by editing the configuration properties as described in Section 8.6 on page 97.</p>
Description	A short description of the configuration.	Optional.
Revision	The revision of the configuration.	Mandatory.
Revision Comment	Comments on this particular revision of the configuration.	Optional.

2. Enter the required information. See Table 29.
3. Click **Create**.

The empty configuration is created.
4. If required, grant permission for other users to work with the new configuration by modifying the ACL. For more information on granting ACL permissions, refer to Section 3.1.1 on page 10.

8.3.2 Cloning a Configuration

Cloning creates a copy of an existing configuration with a new name and revision. A cloned configuration uses the same schema as the original and contains the same configuration data.



In addition, cloning a configuration can optionally replicate any child configurations that hang from the selected parent.

To clone a configuration:

1. In the **Configurations** workspace, right-click a configuration and select **Clone**.

Note: If the configuration is not displayed, perform a search as outlined in Section 8.1.1 on page 72.

The **Clone** dialog box opens. See Figure 38.

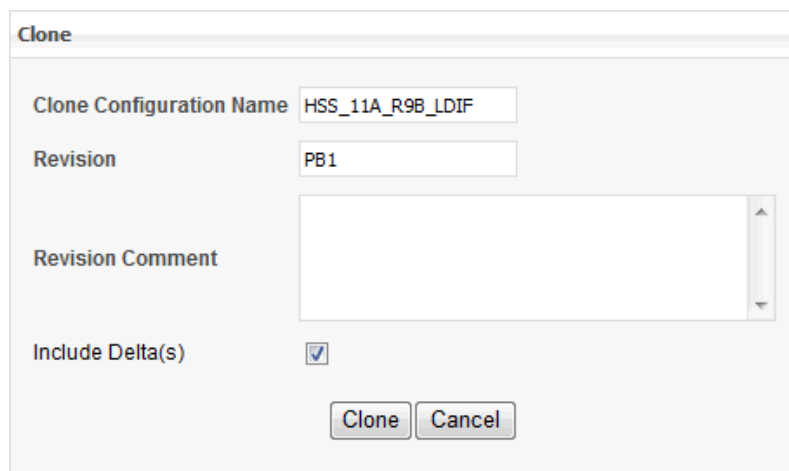
The image shows a 'Clone' dialog box with a title bar. It contains four input fields: 'Clone Configuration Name' with the text 'HSS_11A_R9B_LDIF', 'Revision' with 'PB1', and 'Revision Comment' which is an empty text area. Below these is a checkbox labeled 'Include Delta(s)' which is checked. At the bottom are 'Clone' and 'Cancel' buttons.

Figure 38 Clone Dialog Box

The following table describes the elements of the **Clone** dialog box:

Table 30 Clone Dialog Box Elements

Element	Description	Notes
Clone Configuration Name	The name of the clone configuration.	Mandatory. The clone operation does not automatically rename child configurations, since the full path of newly cloned configurations differs from the original.
Revision	The revision of the clone configuration.	Mandatory.



Table 30 Clone Dialog Box Elements

Element	Description	Notes
Revision Comment	Comments on this particular revision of the clone configuration.	Optional.
Include Delta(s)	When selected, PDB will also clone any delta configurations that hang from the original configuration.	Optional. Enabled by default.

2. Enter the required information. See Table 30.

3. Click **Clone**.

Clones of the selected configuration and the associated deltas (if selected) are created.

Note: PDB automatically generates an internal document number for the new configuration(s). This number can be modified by editing the configuration properties as described in Section 8.6 on page 97.

4. If required, grant permission for other users to work with the new configuration(s) by modifying the ACL. For more information on granting ACL permissions, refer to Section 3.1.1 on page 10.

8.3.3 Merging Node Configurations

PDB allows you to merge the contents of two or more node configurations into a single configuration. This new configuration can be a root configuration or a delta configuration.

When merging elements into a root configuration, the new configuration must be based on one of the available schemas. This schema is selected during the merge process.

When merging elements into a delta configuration, the new delta is restricted to the same schema as its parent. Any configuration can be selected as a parent for the new delta, including another delta configuration.

Regardless of the resulting configuration (root or delta), if any of the merged elements are invalid under the new schema, they are dropped from the merged configuration.

Up to 10 configurations can be merged at the same time. PDB uses the order of the selected configurations to resolve merge conflicts with configurations located at the top of the list being given more precedence.

PDB performs two types of merge operations:

— Trivial Merges



Trivial merges are automatically resolved by PDB and include nonconflicting merges of different parameter names and values. Merging parameters groups is always considered a trivial merge since parameter groups with different primary keys are treated as separate entities.

— Non-Trivial Merges

Non-trivial merges occur when one or more configuration parameters have the same name but different values. By default, these merges are resolved based on the precedence of the merged configurations, with PDB taking the values of the top-most configuration that is in conflict. You can modify these default values during the merge procedure.

To merge configurations:

1. In the **Configurations** workspace, ensure that the configurations you want to merge are displayed.

Note: If any of the configurations are not visible, perform a search as outlined in Section 8.1.1 on page 72.

2. Click **Merge**.

The **Merge** dialog box opens. See Figure 39.



Merge

Select the configurations you wish to merge. Default resolution of non-trivial merges will be based on the order of selected configurations. You will have the opportunity to modify non-trivial merge cases in the following step.

Available

/CSCF_11B_R4B_LDIF_NoEmptyValues[A]

/CSCF_11B_R4G_LDIF[A]

/MPVL_MTAS_11B_Converged[PA2]

/MPVL_MTAS_11B_Fixed[PA4]

/MPVL_MTAS_11B_Mobile[PA2]

/MTAS_11A_R1D_LSV118_LDIF[PA2]

Selected

/HSS_11A_R9B_LDIF/HSS_11A_R9B_NoEmptyValues[PA1]

/HSS_11A_R9B_LDIF[PB1]

The available list matches the displayed contents of the Configurations table. Updating the search filter will change the selectable options.

The merge will be stored at the location and under the name of your choice.

Target Location

/

Schema

HSS_CM_MIM[A]

Latest Schema Revisions Only

Name

tmp

Revision

PA1

Revision Comment

Next >

Cancel

Figure 39 Merge Dialog Box

The following table describes the elements of the **Merge** dialog box:

Note: All items are mandatory.

Table 31 Merge Dialog Box Elements

Element	Description	Notes
Available (List)	<div>A multi-select list of configurations that can be merged.</div> <div>Use the horizontal arrows to move configurations to and from the Selected list.</div>	This list is only populated with configurations that are visible in the Configurations table at the time of the merge.

2/1553-CXP 902 0212 Uen AC | 2016-07-15

83



Table 31 Merge Dialog Box Elements

Element	Description	Notes
Selected (List)	<p>A multi-select list of configurations that have been selected for merge.</p> <p>Use the vertical arrows to adjust the order of the selected configurations.</p> <p>Configurations located higher on this list are given more precedence in resolving non-trivial merge conflicts.</p>	Mandatory.
Target Location (List)	<p>Specifies a target location in the PDB configuration structure for the merged configuration.</p> <p>Selecting / creates a new root configuration.</p> <p>Selecting an existing configuration creates a new delta configuration hanging from the chosen parent.</p>	Mandatory.
Schema (List)	<p>The schema that will be applied to the merged configuration.</p> <p>By default, the list of schemas is filtered to display only the latest revisions. Clear Latest Schemas Only to display the complete list.</p>	Schemas can only be selected for root configurations.
Name	The name of the merged configuration.	Mandatory.
Revision	<p>The revision of the merged configuration.</p> <p>Note: Revisions can only be set for root configurations.</p>	Mandatory.
Revision Comment	Comments on this particular revision of the configuration. Optional.	Optional.

3. Enter the required information. See Table 31.

4. Click **Merge**.



The selected configurations are merged using the default resolution for non-trivial merge conflicts. When the merge is complete, the **Merge Report** dialog box opens. See Figure 40 for a sample summary.

Merge Report

Configurations Merged

- /HSS_11A_R9B_LDIF/HSS_11A_R9B_NoEmptyValues[PA1]
- /HSS_11A_R9B_LDIF[PB1]

The merge is ready. Non-trivial merges have been resolved using the default resolution. You can modify those cases in the table below.

[Download Report](#)

Non Trivial Merge Resolution Table

☒ Use Default Resolution ☐ Manually Select Value

Parameter Name	MOI	Possible Values
RAD-LengthSize	RAD-Application[applicationName=RADIUS],RAD-AppInstance[RAD-ApplicationName=HSS_SM],RAD-Vendor[RAD-VendorId=HSS_SM:10415]	[1]
RAD-TypeSize	RAD-Application[applicationName=RADIUS],RAD-AppInstance[RAD-ApplicationName=HSS_SM],RAD-Vendor[RAD-VendorId=HSS_SM:10415]	[1]

[Finish](#) [Cancel](#)

Figure 40 Sample Merge Report Summary

The Merge Report dialog box provides a summary of the merge and allows you to modify the non-trivial merges.

The **Non Trivial Merge Resolution Table** lists all of the non-trivial merges that were performed during the selected operation. If required, you can modify the default resolution for each parameter by clicking **Manually Select Value** and setting a value for each non-trivial merge.

Note: If the table is paginated, updates on each page are saved automatically.

5. Click **Download Report** to download the merge report.

The merge report is a space-separated text file containing a complete list of all actions that were performed to merge the selected configurations. Inside the report, merge activities are recorded in the following format:

<Merge Type> <Element Type> <Action> <MOI>

Example 2 shows an extract of a sample merge report.



```

# Merged
#
# Configuration: /HSS_11A_R9B_LDIF/HSS_11A_R9B_NoEmptyValues[PA1]
# Schema: HSS_CM_MIM
#
# Configuration: /HSS_11A_R9B_LDIF[PB1]
# Schema: HSS_CM_MIM
#
# Target
#
# Configuration: /HSS_11A_R9B_MergedValues[PA1]
# Schema: HSS_CM_MIM
#
...
TRIVIAL PG ADD RAD-Application[applicationName=RADIUS],RAD-AppInstance[RAD-ApplicationName=HSS_SM],
RAD-Vendor[RAD-VendorId=HSS_SM:10415],RAD-Vsa[RAD-VsaId=HSS_SM:10415:6]
TRIVIAL P MERGE RAD-Application[applicationName=RADIUS],RAD-AppInstance[RAD-ApplicationName=HSS_SM],
RAD-Vendor[RAD-VendorId=HSS_SM:10415],RAD-Vsa[RAD-VsaId=HSS_SM:10415:6]:RAD-VsaFormat [[address]]
TRIVIAL P MERGE RAD-Application[applicationName=RADIUS],RAD-AppInstance[RAD-ApplicationName=HSS_SM],
RAD-Vendor[RAD-VendorId=HSS_SM:10415],RAD-Vsa[RAD-VsaId=HSS_SM:10415:6]:RAD-VsaName [[3GPP-SGSN-Address]]
NON TRIVIAL P MERGE RAD-Application[applicationName=RADIUS],RAD-AppInstance[RAD-ApplicationName=HSS_SM],
RAD-Vendor[RAD-VendorId=HSS_SM:10415]:RAD-LengthSize [[2], [1]]
NON TRIVIAL P MERGE RAD-Application[applicationName=RADIUS],RAD-AppInstance[RAD-ApplicationName=HSS_SM],
RAD-Vendor[RAD-VendorId=HSS_SM:10415]:RAD-ContentSize [[2], [1]]
TRIVIAL P MERGE RAD-Application[applicationName=RADIUS],RAD-AppInstance[RAD-ApplicationName=HSS_SM],
RAD-Vendor[RAD-VendorId=HSS_SM:10415]:RAD-VendorName [[3GPP]]
TRIVIAL PG ADD RAD-Application[applicationName=RADIUS],RAD-AppInstance[RAD-ApplicationName=HSS_SM],
RAD-Vendor[RAD-VendorId=HSS_SM:5535]
...

```

Example 2 Sample Merge Report

6. Click **Done with Merge**.

The Merge is complete.

Note: PDB automatically generates an internal document number for the new configuration. This number can be modified by editing the configuration properties as described in Section 8.6 on page 97.

- If required, grant permission for other users to work with the new configuration by modifying the ACL. For more information on granting ACL permissions, refer to Section 3.1.1 on page 10.

8.4 Creating a Delta Configuration

Delta configurations build upon and modify a parent configuration. The delta inherits all of its configuration data from the parent, except those elements that are locally overridden.

Delta configurations store only the changes that are made to the inherited data, including additions, deletions, and modifications. These updates supersede any corresponding elements that were inherited from the parent.

Any configuration can be used as the parent of a delta configuration, including another delta configuration.



Caution!

Delta configurations are bound to parent configurations. Any change to the parent is automatically propagated to the delta unless it has been specifically modified to contain different information.

PDB allows you to attach an existing delta configuration to a different parent configuration. This process is called Change Parent and is fully described in Section 8.4.1 on page 88.

To create a delta configuration:

- 1. In the **Configurations** workspace, right-click a parent configuration for the new delta and select **Create Delta**.
Note: If the configuration is not displayed, perform a search as outlined in Section 8.1.1 on page 72.

The **Delta** dialog box opens. See Figure 41.

The Delta dialog box is a light gray window with a title bar labeled 'Delta'. It contains several input fields: 'Configuration Name' (a single-line text box), 'Document Number' (a single-line text box with a yellow lightbulb icon to its right), 'Description' (a multi-line text area), 'Revision' (a single-line text box containing the text 'PA1'), and 'Revision Comment' (a multi-line text area). At the bottom of the dialog are two buttons: 'Create' and 'Cancel'.

Figure 41 Delta Dialog Box

The following table describes the elements of the **Delta** dialog box:

Table 32 Delta Dialog Box Elements

Element	Description	Notes
Configuration Name	The name of the configuration.	Mandatory.



Table 32 Delta Dialog Box Elements

Element	Description	Notes
Document Number	The document number of the configuration.	Optional. If left blank, PDB will automatically generate an internal document number for the new configuration. This number can be modified by editing the configuration properties as described in Section 8.6 on page 97.
Description	A short description of the configuration.	Optional.
Revision	The revision of the configuration. Mandatory.	Mandatory.
Revision Comment	Comments on this particular revision of the configuration.	Optional.

2. Enter the required information. See Table 32.
3. Click **Create**.

The delta configuration is created.
4. If required, grant permission for other users to work with the new configuration by modifying the ACL. For more information on granting ACL permissions, refer to Section 3.1.1 on page 10.

8.4.1 Changing the Parent of a Delta Configuration

PDB can perform a **Change Parent** operation to attach an existing delta configuration to a different parent configuration. This new parent can be a newer revision of the same parent configuration or another configuration in PDB.

Change Parent maps configuration elements (in the delta) onto the new parent. If the structure of the parent configuration is different, this mapping will identify the reference points that correspond to the original configuration structure, where possible. For example, if a delta configuration adds a parameter to an inherited parameter group, **Change Parent** will attempt to locate the corresponding parameter group in the new parent configuration and update the delta references accordingly. For more information on Delta configurations, refer to Section 8.4 on page 86.

Note: It is possible that the delta configuration will reference configuration elements that are no longer included in the parent. These elements are automatically removed during the **Change Parent** process. Any configuration elements that are removed in this way are documented in the Change Parent Configuration report.



8.4.1.1 Change to a New Revision of the Same Parent

To change the parent of a delta configuration to a new revision of the same configuration:

1. In the **Configurations** workspace, right-click a delta configuration and select **Change Parent > Same Parent Different Revision**.

Note: If the delta configuration is not displayed, perform a search as outlined in Section 8.1.1 on page 72.

The **Change Parent** dialog box opens. See Figure 42.

The screenshot shows a dialog box titled "Change Parent". It contains two labels: "Configuration" and "New Parent Revision". The "Configuration" field displays the path "/HSS_11A_R9B_LDIF/HSS_11A_R9B_Lab2[PA1]". The "New Parent Revision" field displays the path "/HSS_11A_R9B_LDIF[PB1]" followed by a dropdown arrow. At the bottom right of the dialog box are two buttons: "Next >" and "Cancel".

Figure 42 Change Parent Dialog Box

2. Select a new parent revision from the list.
3. Click **Next**.

PDB analyzes the proposed change and updates the **Change Parent** dialog box with a summary of the configuration impacts. See Figure 43 for a sample summary.

Note: No changes are made to the configuration during this phase.

If changing the parent configuration will force PDB to discard configuration items, a Change Parent Configuration report is generated to highlight the impacted configuration elements. In addition, any validation issues with the updated configuration are captured in a validation report.

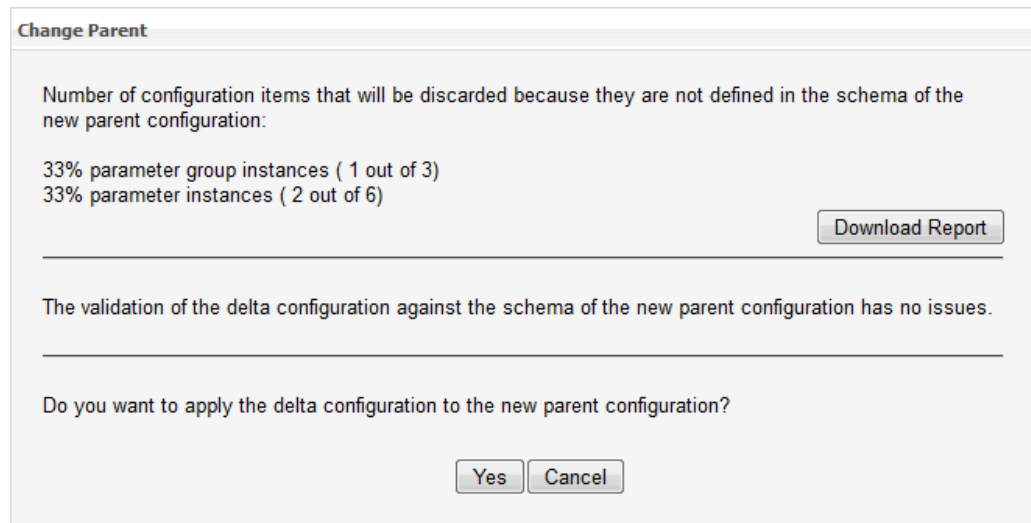


Figure 43 Sample Change Parent Summary

4. If required, click **Download Report** to download the report files.

The following example illustrates a sample Change Parent Configuration report. For a sample validation report, refer to Section 8.8 on page 101.

REPORT - CHANGE PARENT CONFIGURATION

SOURCE CONFIGURATION : /HSS_11A_R9B_LDIF/HSS_11A_R9B_Lab2[PA1]
NEW PARENT CONFIGURATION : /HSS_11A_R9B_LDIF[PB1]

TARGET : /HSS_11A_R9B_LDIF/HSS_11A_R9B_Lab2[PA1]

33% parameter group instances (1 out of 3) will not be moved or copied
33% parameter instances (2 out of 6) will not be moved or copied

> Discarded PARAMETER-GROUP instances - Not defined in the schema of the new parent configuration:
FTU[applicationName=FTU],tspFtuKeyReference[tspFtuKeyReferenceKey=0]

> Discarded PARAMETER instances - Not defined in the schema of the new parent configuration:
FTU[applicationName=FTU],tspFtuKeyReference[tspFtuKeyReferenceKey=0],tspFtuKey=0
FTU[applicationName=FTU],tspFtuKeyReference[tspFtuKeyReferenceKey=0],tspFtuKey=2

Example 3 Sample Change Parent Configuration Report

5. Click **Yes** to perform the change parent operation.

8.4.1.2 Change to a Different Parent

To change the parent of a delta configuration to a different configuration:

1. In the **Configurations** workspace, right-click a delta configuration and select **Change Parent > Different Parent**.

Note: If the delta configuration is not displayed, perform a search as outlined in Section 8.1.1 on page 72.

The **Change Parent** dialog box opens. See Figure 44.



The dialog box is titled "Change Parent". It contains the following fields and controls:

- Delta Configuration:** A text field containing the path "/MPVL CSCF 13A Converged/Lab2[PA1]".
- Node:** Two dropdown menus, both currently set to "All".
- New Parent Configuration:** A text input field with a lightbulb icon on the right, indicating auto-complete functionality.
- Keep a copy of the original delta configuration:** A checkbox that is currently unchecked.
- Buttons:** "Next >" and "Cancel" buttons at the bottom right.

Figure 44 Change Parent Dialog Box

- Using the Node and Revision lists as filters, select a new parent configuration.

Note: The **New Parent Configuration** field uses auto-complete functionality. Typing part of a name displays matching configurations for the selected node revision. Use the down-arrow on your keyboard to display all of the available configurations.

- If required, select **Keep a copy of the original delta configuration** to preserve the original delta configuration.

By default, **Change Parent** will modify the selected delta configuration to hang from a new parent without preserving the original configuration data. **Keep a copy of the original delta configuration** preserves the original data by creating a clone of the delta configuration that will hang from the new parent, leaving the selected delta configuration intact.

- Click **Next**.

PDB analyzes the proposed change and updates the **Change Parent** dialog box with a summary of the configuration impacts. See Figure 45 for a sample summary.

Note: No changes are made to the configuration during this phase.

If changing the parent configuration will force PDB to discard configuration items, a Change Parent Configuration report is generated to highlight the impacted configuration elements. In addition, any validation issues with the updated configuration are captured in a validation report.



Change Parent

Number of configuration items that will be discarded because they are not defined in the schema of the new parent configuration:

66% parameter group instances (2 out of 3)
40% parameter instances (4 out of 10)

Download Report

The validation of the delta configuration against the schema of the new parent configuration has issues.

Download Report

Do you want to apply the delta configuration to the new parent configuration?

YesCancel

Figure 45 Sample Change Parent Summary

5. If required, click **Download Report** to download the report files.

The following example illustrates a sample Change Parent Configuration report. For a sample validation report, refer to Section 8.8 on page 101.

REPORT - CHANGE PARENT CONFIGURATION

SOURCE CONFIGURATION : /MPVL CSCF 13A Converged/Lab2[PA1]
NEW PARENT CONFIGURATION : /MPVL CSCF 14A Converged[F]

TARGET : /MPVL CSCF 14A Converged/Lab2[PA1]

66% parameter group instances (2 out of 3) will not be moved or copied
40% parameter instances (4 out of 10) will not be moved or copied

> Discarded PARAMETER-GROUP instances - Not defined in the schema of the new parent configuration:
DIA-CFG-NeighbourNode-LocalVIPNetRed[localVIPNetRedNodeId=1]
DIA-CFG-OwnNode-LocalVIPNetRed[localVIPNetRedStackId=1]

> Discarded PARAMETER instances - Not defined in the schema of the new parent configuration:
DIA-CFG-NeighbourNode-LocalVIPNetRed[localVIPNetRedNodeId=1],localVIPNetRedNodeId=1
DIA-CFG-NeighbourNode-LocalVIPNetRed[localVIPNetRedNodeId=1],connectFailureQuarantineTime=30
DIA-CFG-OwnNode-LocalVIPNetRed[localVIPNetRedStackId=1],localVIPNetRedStackId=1
DIA-CFG-OwnNode-LocalVIPNetRed[localVIPNetRedStackId=1],zone2TcpAddressList=127.0.0.1

Example 4 Sample Change Parent Configuration Report

6. Click **Yes** to perform the change parent operation.

8.5 Working with Configuration Revisions

Node configurations stored in PDB are revision controlled. A given revision will precede or supersede other revisions of the same configuration, allowing for a number of revision-specific operations.

These operations include:

- Changing the Document State



- Creating a New Revision
- Importing a New Revision

By default, only the latest revision of each configuration is displayed in the **Configurations** table. Older revisions are accessible through the revision history or by refining the search criteria. For more information on performing a search, refer to Section 8.1.1 on page 72.

To view all revisions of a selected configuration, right-click to open the context menu and select **Revision > Show Revision History**.

PDB requires all predecessors to be in a locked state (FROZ, FREE, WIDR) before subsequent revisions can be created.

8.5.1 Document States

PDB uses a system of document states to provide information on the completeness, quality, and approval status of a particular configuration revision. The document state is indicated by a status code that is part of the configuration metadata. PDB uses the following document states for node configurations:

PREL	Preliminary. Used to designate an unlocked configuration that is open for modification. Configurations in the PREL state cannot serve as a basis for new revisions.
FROZ	Frozen. Used to designate a frozen configuration where the content has been locked to prevent further changes. This is the basic state for configuration revisions that are stored in PDB but are not approved for release.
FREE	Approved. Used to designate configurations that are approved for release. Approved configurations are locked to prevent any changes to the released content.
WIDR	Withdrawn. Used to designate configurations that are no longer intended for use. Withdrawn configurations are still accessible in PDB, but are locked to prevent further changes.

8.5.1.1 Changing the Document State

PDB allows you to change the document state of node configurations that you have permission to modify. For more information on ACL permissions, refer to Section 3.1 on page 9.

Configuration revisions must follow the following sequence of document states:

PREL > FROZ > FREE > WIDR



Note: Only PDB System Administrators can set a configuration to a previous state.

New configurations, or configuration revisions start in the PREL state where they can be modified and updated. The document state is changed as the configuration progresses through its lifecycle.

To change the document state of a configuration revision:

1. In the **Configurations** workspace, right-click a configuration and select **Revision > Set to <STATE>**.

Note: If the configuration is not visible, perform a search as outlined in Section 8.1.1 on page 72.

When moving from PREL to FROZ, the **Freeze** dialog box opens. See Figure 46.

Note: Before a configuration can be frozen, the associated schema and IVL, if used, must also be frozen.

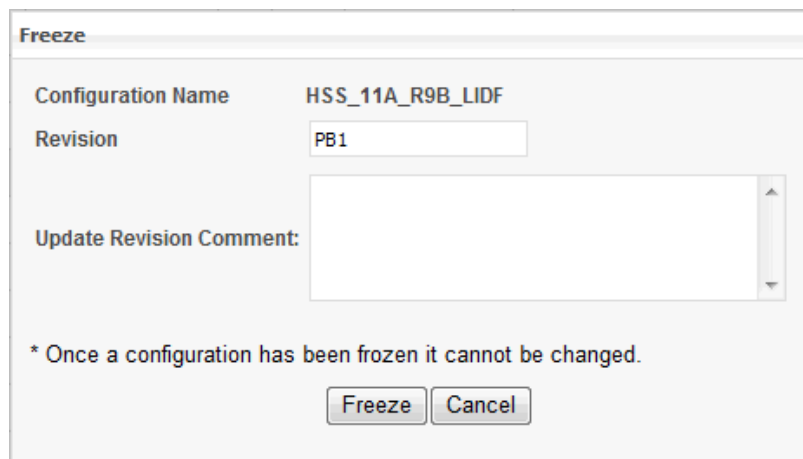
The image shows a 'Freeze' dialog box with a title bar. Inside, there are two labels: 'Configuration Name' and 'Revision'. The 'Configuration Name' field contains the text 'HSS_11A_R9B_LIDF'. The 'Revision' field contains the text 'PB1'. Below these fields is a larger text area labeled 'Update Revision Comment:'. At the bottom of the dialog, there is a line of text: '* Once a configuration has been frozen it cannot be changed.' and two buttons labeled 'Freeze' and 'Cancel'.

Figure 46 Freeze Dialog Box

The Freeze dialog box allows you to modify the revision level of the configuration. Here preliminary revisions can be set to a solid state before freezing.

Stop!

Only PDB System Administrators can unfreeze a configuration. Do not continue if additional changes are required.

2. Click **Freeze**.

The selected configuration is locked and the document state is set to FROZ.



For all other state transitions a confirmation dialog box is displayed.

Click **OK** to change the document state.

8.5.2 Creating A New Revision

You can update a locked configuration by creating a new revision. This process creates a clone of the locked configuration at a new revision level. The new revision retains the name and tags (IVL, IFN, MPVL, DPVL) associated with the original configuration and resets the document state to PREL, allowing you to modify the data.

To create a new revision of a frozen configuration:

1. In the **Configurations** workspace, right-click a locked configuration and select **Revision > Create New Revision**.

Note: If the configuration is not displayed, perform a search as outlined in Section 8.1.1 on page 72.

The **New Revision** dialog box opens with the next legal revision displayed. See Figure 47.

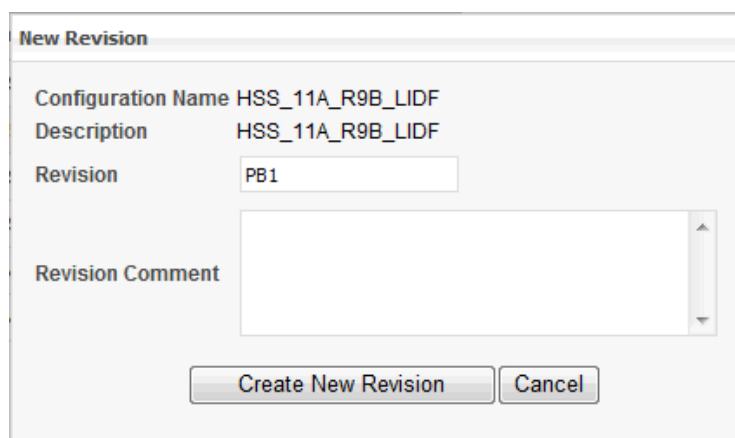
The image shows a 'New Revision' dialog box. It has a title bar with the text 'New Revision'. Inside the dialog, there are four labeled fields: 'Configuration Name' with the value 'HSS_11A_R9B_LIDF', 'Description' with the value 'HSS_11A_R9B_LIDF', 'Revision' with the value 'PB1', and 'Revision Comment' which is an empty text area. At the bottom of the dialog, there are two buttons: 'Create New Revision' and 'Cancel'.

Figure 47 New Revision Dialog Box

2. Verify that the proposed revision is correct and update as necessary.
3. Click **Create New Revision**.

A new revision is created.

8.5.3 Importing a New Revision

A revised set of files can be imported to create a new revision of a locked configuration. The new revision retains the name and tags (IVL, IFN, MPVL, DPVL) associated with the original configuration and resets the document state to PREL, allowing you to modify the data.



Configuration files can be packaged in a ZIP or TAR archive, or imported directly from LDIF or XML files.

Note: An archive directory structure is not required by PDB to import configurations; however, if directories are present inside the file, then PDB will recursively scan for configuration files within the directories as needed.

To import a new revision of a configuration:

1. In the **Configurations** workspace, right-click a locked configuration and select **Revision > Import New Revision**.

The **Import New Revision** dialog box opens with the next legal revision displayed. See Figure 48.

The screenshot shows the 'Import New Revision' dialog box. It contains the following fields and controls:

- Configuration Name:** HSS_11A_R9B_LIDF
- Description:** (empty text area)
- Revision:** PB3 (text input)
- Revision Comment:** (empty text area)
- Schema:** HSS_CM_MIM[A] (dropdown menu)
- Import Type:** PVL (dropdown menu)
- Import From:** FIXED (dropdown menu)
- Use Delta:** (checkbox, currently unchecked)
- Input File:** (empty text input)
- Browse...:** (button next to Input File)
- Uploaded File:** (empty text input)
- Buttons:** 'Import Revision' and 'Cancel' at the bottom.

Figure 48 Import New Revision Dialog Box

The following table describes the elements of the **Import New Revision** dialog box:

Table 33 Import New Revision Dialog Box Elements

Element	Description	Notes
Revision	The new revision level of the configuration.	Mandatory PDB automatically proposes the new legal revision level.



Table 33 Import New Revision Dialog Box Elements

Element	Description	Notes
Revision Comment	Comments on this particular revision of the configuration.	Optional.
Schema (List)	The schema revision to use as a template for the incoming configuration data. The drop-down list shows both the schema name and revision (in square brackets).	Mandatory.
Import Type (List)	The configuration format. Possible formats include: <ul style="list-style-type: none"> • LDIF • NETCONF • PVL 	Mandatory. For more information on the PVL format, refer to the Parameter List Template Description, EAB/FTI-08:0686 Uen.
Import From (List)	For PVL input files only. The PVL parameter value set to import.	Mandatory. Select Use Delta to give precedence to delta values over the chosen parameter value set.
Input File	The path and file name of the configuration archive.	Mandatory.

2. Enter the required information. See Table 33.
3. Click **Import**.

PDB performs the following operations when importing a new configuration.

- a PDB imports the data contained within the selected archive and generates a new configuration revision.
- b PDB automatically performs a validation check during the import process. This check verifies the syntax of the configuration files and validates the configuration against the selected schema.
- c PDB generates an import validation report that can be downloaded from the **Report Available** dialog box that appears at the end of the import process. For a description of the messages presented in import validation reports, refer to Configuration Validation Errors in the Appendix.
- d PDB stores the configuration archive and the import validation report. These files can be downloaded from the properties dialog box for the new configuration.



8.6 Modifying Configuration Properties

While in PREL state, PDB allows you to modify the following configuration properties:

- Name

- Document number

If no document number is specified, PDB automatically generates an internal number.

Note: Document numbers are validated using Ericsson's standard rules for registration notation.

Subsequent revisions of an existing node configuration are locked to the document number of their predecessor.

- Description

Note: Subsequent revisions of an existing node configuration are locked to the description of their predecessor.

- Revision Comment

- Tag as an Initial Value List (IVL)

Use the IVL tag to mark configurations that can act as an initial value list. For more information on the use of IVLs, refer to Section 8.9 on page 104.

Tagging a node configuration as IVL allows you to associate it with a configuration schema as an IVL. For more information on associating an IVL with a configuration schemas, refer to Section 7.4.1 on page 53.

- Tag as Imported From Node (IFN)

Use the IFN tag to mark configurations that have been imported from a node.

- Tag as Master Parameter Value List (MPVL)

Use the MPVL tag to identify the official Master Parameter Value List for a specific node revision.

- Tag as Delta Parameter Value List (DPVL)

Use the DPVL tag to mark delta configurations. For more information on delta configurations, refer to Section 8.4 on page 86.

If the configuration is in a locked state, only the **Description** and **Revision Comment** fields can be modified.



Most configuration properties, including the tags, are visible in the **Configurations** table. To view all properties associated with a specific configuration, open the context menu for that configuration and select **Properties**.

To edit node configuration properties:

1. In the **Configurations** workspace, right-click an unlocked configuration and select **Edit**.

Note: If the configuration is not displayed, perform a search as outlined in Section 8.1.1 on page 72.

The **Edit** dialog box opens.

2. Update the configuration properties as required.
3. Click **Apply**.

The updated properties are saved.

8.7 Migrating a Configuration to Another Schema

While in PREL state, PDB allows you to migrate a root configuration from one schema to another. This schema update can be performed between versions of the same schema or between different schemas.

Note: Because schemas outline the valid elements for a configuration, updating from one schema to another applies the requirements of the new schema to the existing data. This transition between schemas can invalidate parts of the configuration.

Before performing an update, PDB analyzes the proposed operation and generates a schema update report that documents all of the configuration elements that are structurally invalid under the new schema. Any parameters or parameter groups that do not fit within the new schema will be automatically removed during the migration process. The schema update report provides a detailed list of the configuration elements that must be removed during the procedure.

In addition to generating a schema update report, PDB performs a validation check on the remaining configuration elements to ensure that they are in line with the new schema definition. For more information on schema validation, refer to Section 8.8 on page 101.

A schema update can only be applied to root configurations. Any child configurations are moved between the selected schemas simultaneously.

Note: The migration process requires a target schema. Ensure that the target schema exists in PDB before beginning this process. For more information on working with schemas, refer to Section 7 on page 40.

To migrate a configuration:



1. In the **Configurations** workspace, right-click a root configuration in PREL state and select **Update Schema**. You cannot update the schema of a delta configuration since delta configurations must follow the schema used by the parent configuration.

Note: If the configuration is not displayed, perform a search as outlined in Section 8.1.1 on page 72.

The **Schema Update** dialog box opens. See Figure 49.

Note: Migrating a configuration changes the data structure. To preserve a copy of the original configuration tree, clone it prior to executing the migration procedure. For more information on cloning a configuration, refer to Section 8.3.2 on page 79.

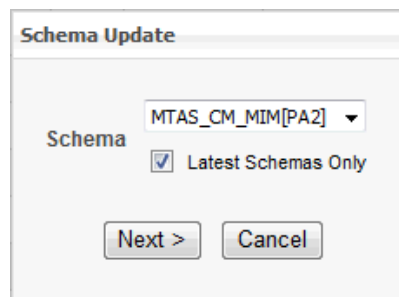


Figure 49 Schema Update Dialog Box

2. Select a target schema for the migration.

Note: By default, the list of schemas is filtered to display only the latest revisions. Clear **Latest Schemas Only** to display the complete list.

3. Click **Next**.

PDB analyzes the proposed migration, updating the **Schema Update** dialog box with a summary of the configuration impacts and links to the schema update report and the validation report. See Figure 50 for a sample summary.

Note: No changes are made to the configuration(s) during this phase.

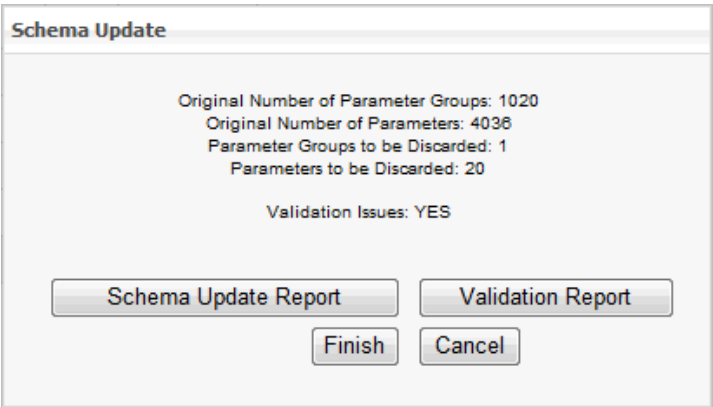


Figure 50 Sample Migration Summary

- 4. Click **Schema Update Report** or **Validation Report** to download the reports.

The reports are pipe-separated text files that identify the impacted configuration elements. The following example illustrates a sample migration report. For a sample validation report, refer to Section 8.8 on page 101.

Schema Update Report
Original Number of Parameter Groups: 1020
Original Number of Parameters: 4036
Parameter Groups to be Discarded: 1
Parameters to be Discarded: 20
Validation Issues: YES
Schema Update Report
Validation Report
Finish
Cancel

Example 5 Sample Schema Update Report

- 5. Click **Finish** to migrate the configuration.



8.8 Validating a Configuration

PDB can validate node configurations to ensure they conform to their respective schema definitions. A validation check is performed automatically as part of the following operations, but should be triggered manually, as needed:

- Importing Configurations
- Migrating Configurations to Another Schema
- Modifying Configuration Data (PDB validates changes to the configuration data on-the-fly)
- Rebasing Delta Configurations

Triggering a manual validation check through the PDB GUI is strongly recommended after any operation that impacts the configuration data and prior to export.

PDB performs the following checks during schema validation:

Cardinality	PDB verifies that the number of instances defined for a given element within a node configuration is allowed by the schema.
--------------------	---

Value Constraints	PDB verifies that the parameter values defined within a node configuration follow the type constraints and value patterns that are allowed for that element as defined by the schema.
--------------------------	---

Issues discovered during the validation check are output to a plain text file that can be downloaded from the PDB server. The report displays the following messages:

[CARDINALITY]	These warning messages appear when the number of instances defined for a given element is not permitted by the schema.
----------------------	--

[VALUE]	These warning messages appear when the value format of a given parameter does not follow the rules defined in the schema.
----------------	---

[UNUSED]	These info messages appear when an element is defined in the schema but has not been instantiated in the node configuration.
-----------------	--

By default, these messages are excluded from the validation report. To include information about unused elements when triggering a validation from the PDB GUI, select **Report unused parameters and parameter groups** during validation.



[NO_ACTION]	These info messages appear when the value of a given parameter cannot be verified because it contains site-specific variables.
[VAR_USAGE]	These info messages appear when the name of a parameter value variable in the node configuration is not found in the suggested list of Global Variables. Usage of suggested Global Variables is highly recommended. For more information, refer to Section 13.3 on page 190.

To validate a configuration:

1. In the **Configurations** workspace, right-click a configuration and select **Validate**.

Note: If the configuration is not displayed, perform a search as outlined in Section 8.1.1 on page 72.

The Validation dialog box opens. See Figure 51.

The screenshot shows a dialog box titled "Validation". Inside, there is a section labeled "Site Specific List" with a text input field below it. Below the input field are two checkboxes: "Show All Site Specific Lists" and "Report unused parameters and parameter groups". At the bottom right of the dialog are two buttons: "Validate" and "Cancel".

Figure 51 Validation Dialog Box

2. If required, select an available site-specific list to resolve parameter value variables in the selected node configuration.

Note: The **Site Specific List** field uses auto-complete functionality. Typing part of a name displays matching site-specific lists. Use the down-arrow on your keyboard to display all of the available lists.

By default, PDB only shows site-specific lists that are associated with this configuration. Select **Show All Site-Specific Lists** to see all available lists.

3. Click **Validate**.

The node configuration is validated against the associated schema.

If validation issues are found, the **Report Available** dialog box opens.

4. Click **Download Report** to save the Validation report.



Example 6 presents a sample validation report.

```
CONFIGURATION VALIDATION REPORT
Configuration: /CSCF_11B_R4G_LDIF[PB1]
Schema: CSCF_CM_MIM[A]

WARNING CARDINALITY | PG | SigCompAdvertisedState | Actual cardinality: 0 | Schema constraints: 1-1
WARNING CARDINALITY | PG | SigCompNonTrafficTrace | Actual cardinality: 0 | Schema constraints: 1-1
WARNING CARDINALITY | PG | SigCompProfileContainer | Actual cardinality: 0 | Schema constraints: 1-1
WARNING CARDINALITY | PG | SigCompTrace | Actual cardinality: 0 | Schema constraints: 1-1
WARNING CARDINALITY | PG | SigCompProfile | Actual cardinality: 0 | Schema constraints: 1-1
WARNING VALUE | P | tspPmApplication[applicationName=tspPM],tspPmMeasReaderNr=0 |
Data type: NUMERIC BIG_DECIMAL | Format: Greater or Equal to: 1 / Less or Equal to: 5000
WARNING VALUE | P | tspPmApplication[applicationName=tspPM],tspPmMonitorNr=0 |
Data type: NUMERIC BIG_DECIMAL | Format: Greater or Equal to: 1 / Less or Equal to: 500
WARNING VALUE | P | DNS-Application[applicationName=DNS],DnsServerEntry=10.80.52.34:53 |
Data type: STRING | Format: Pattern: ([0-9]{1,3}\.){4}:([0-9])+
INFO UNUSED | PG | tspPmApplication[applicationName=tspPM],tspPmMonitorGroup
INFO UNUSED | PG | tspLicenseManagement[applicationName=tspLicenseManagement],
tspLicenseServerConnection
INFO UNUSED | PG | NumberNormalisation[applicationName=NumberNormalisation],
NumNormProfile[NumNormProfile=0],NumNormNsnData
INFO UNUSED | PG | NumberNormalisation[applicationName=NumberNormalisation],
NumNormProfile[NumNormProfile=0],NumNormOsnData
INFO UNUSED | PG | NumberNormalisation[applicationName=NumberNormalisation],
NumNormProfile[NumNormProfile=0],NumNormDenormalizationSubstitutionRule
...
...
...
```

Example 6 Sample Validation Report

8.9 Exporting a Node Configuration

PDB exports node configurations as ZIP or TAR files. With the exception of configurations exported in the PVL format, all exported configurations contain the necessary logic to actually configure the target node.

PDB supports the use of Initial Value Lists (IVLs) that can be associated with configuration schemas. IVLs represent the configuration of an LDAP node after maiden installation. When exporting a node configuration in LDIF format, PDB will use IVL data associated with the configuration schema to produce configuration files that do not collide with the configuration values that are assumed to already exist in the real node. For more information on linking an IVL to a schema, refer to Section 7.4.1 on page 53.

Before exporting a configuration in EAS, LDIF, or NETCONF format, PDB must resolve all of the site-specific variables that have been defined in the configuration data. These variables can be resolved manually or through the use of a site-specific list.

- For more information on site-specific variables, refer to Section 9.3.4 on page 144.
- For more information on site-specific lists, refer to Section 13 on page 177.

In addition to any site-specific variables present in a node configuration, PDB must resolve values for several mandatory Configuration Management (CM) variables that are used to set up the configuration tool itself. For more information on CM variables refer to Section 18 on page 227.



Exporting node configurations can be triggered through web services or using the GUI or CLI.

8.9.1 Web Services Export

When a web service export is initiated, a web client requests a specific configuration and provides the required site-specific values to PDB. After processing the request, PDB returns a configuration ZIP file to the web client for use in configuring the node.

In the IMSREF System Tool, the RIOT component is configured as a PDB web client.

Note: Alignment between the site-specific parameters defined in PDB and the parameter mapping within RIOT is required to perform a successful web services export. For more information on the RIOT parameter mapping, refer to Mapping Parameters in the RIOT Provisioning and Configuration section of the IMSREF Reusable Inter-Operability Testing (RIOT) System Administration Guide, 1/1543-CXP 902 0096.

8.9.2 Exporting a Configuration from the PDB GUI

PDB allows you to export node configurations through the GUI or the CLI. This section describes the export procedure using the GUI. For information on exporting node configurations using the PDB CLI, refer to [export-configuration](#) in the PDB Command Line Interface (CLI) Reference, 1/1540-CXP 902 0212.

Before a configuration can be exported in LDIF, NETCONF or PED (EAS) format, all of the required site-specific values must be populated. Values for individual site-specific parameters can be input during the export procedure or resolved using a site-specific list. For more information on site-specific lists, refer to Section 13 on page 177.

Note: Exporting a configuration in PVL format does not require any site-specific information. This format is used to preserve configuration data along with any parameter value variables.

Using PDB, you can export the following configuration types:

- Full Configurations

Exporting a full configuration includes the entire configuration tree. All parameter groups and parameters associated with the selected configuration, either directly or through inheritance, are included. Full configurations can be used to completely replace any configuration data currently installed on the selected node.

- Delta Configurations

Exporting a configuration delta includes only the parameter groups and parameters that are local to the delta configuration. Any data inherited



from parent configurations are not included. While delta configurations are not structurally valid on their own, they can be used, for example, to directly upgrade a node from one configuration to another by capturing the differences between them or to set a specific feature on/off. For more information on generating a delta configuration from comparison results, refer to Section 10.3 on page 158.

To manually export a node configuration:

1. In the **Configurations** workspace, right-click a configuration and select **Export**.

Note: If the configuration is not displayed, perform a search as outlined in Section 8.1.1 on page 72.

The **Export** dialog box opens.

The following table describes the elements of the **Export** dialog box:

Table 34 Export Dialog Box Elements

Element	Description	Notes
Format (List)	Sets the format of the exported configuration. Possible formats include: <ul style="list-style-type: none">• CSV (report)• LDIF• NETCONF• PED (EAS)• PVL	<p>The default configuration format is specified by the PDB node definition. For more information on managing nodes in PDB, see Section 5 on page 16</p> <p>PVL data cannot be used to configure nodes directly. For more information on the PVL format, refer to the Parameter List Template Description, EAB/FTI-08:0686 Uen.</p> <p>Configurations exported in the PVL format will include the configuration name, document number, and revision as XML comments.</p> <p>For more information on the configuration report, refer to Section 8.9.2.1 on page 110.</p>



Table 34 Export Dialog Box Elements

Element	Description	Notes
Transaction (List)	<p>For NETCONF export format only.</p> <p>Customizes the NETCONF configuration bundle to deploy the node configuration in one or more transactions. PDB supports the following transaction types:</p> <ul style="list-style-type: none"> • Single transaction • Multiple transactions, split at parameter group level • Multiple transactions, split at parameter level 	<p>Single transaction produces one file with the full configuration.</p> <p>Multiple transactions, split at parameter group level produces one snippet file for each parameter group instance.</p> <p>Multiple transactions, split at parameter level produces one snippet file for each parameter instance.</p> <p>When exporting for multiple transactions, PDB adheres to the following rules:</p> <ul style="list-style-type: none"> • Read-only parameters are not included. • Parameters with multiple values in the same class instance are always part of the same snippet. • Primary key parameters are always pushed with each class instance. • Parameter struct members are always processed together.



Table 34 Export Dialog Box Elements

Element	Description	Notes
Use NETCONF Subsystem (List)	<p>For NETCONF export format only.</p> <p>Configures direct access the NETCONF subsystem when pushing the configuration bundle over an SSH connection.</p>	<p>By default, NETCONF configuration bundles exported from PDB use the NETCONF subsystem.</p> <p>Systems that use Common Operation and Maintenance (COM) 6.0+ can be configured to enforce the use of the NETCONF subsystem. If subsystem access is not enforced, configuration bundles that interface with the NETCONF subsystem will fail to push. In these situations, the configuration bundle can be exported without using the NETCONF subsystem.</p>
Include System-Created Classes (List)	<p>For NETCONF export format only when exporting in multiple transactions.</p> <p>Specifies if system-created classes are included in the configuration snippet files.</p>	<p>By default, NETCONF configuration bundles exported from PDB include system-created classes. These classes are instantiated by the Managed Element (ME) and cannot be created or deleted over the Northbound Interface.</p> <p>When exporting a NETCONF configuration in multiple transactions, system-created classes can be excluded from the configuration snippet files, if required by the node.</p>
Export To (List)	<p>For PVL export format only.</p> <p>Specifies which PVL parameter value set to export. Possible values include:</p> <ul style="list-style-type: none">• CONVERGED• DEFAULT_VALUE• FIXED• MOBILE• STANDALONE	



Table 34 Export Dialog Box Elements

Element	Description	Notes
Export Type (Option Buttons)	<p>For delta configurations only.</p> <ul style="list-style-type: none"> • Select Entire Configuration to export the full configuration tree, including inherited data. • Select Delta Only to export only the data that is local to the delta configuration. 	
Site-Specific Variable Resolution (Option Buttons)	<p>Select how parameter value variables in the node configuration will be resolved.</p> <ul style="list-style-type: none"> • Do not resolve variables Only valid for CSV (report) and PVL formats. • Resolve variables using a Site Specific List • Resolve variables manually 	
Archive Type (Option Buttons)	<p>Select an archive type for the exported configuration. The following archive types are available:</p> <ul style="list-style-type: none"> • ZIP • TAR • TAR_AIT 	<p>PDB defaults to the ZIP format.</p> <p>TAR_AIT produces an Automatic Installation Tool (AIT) compliant TAR file that contains additional information intended for use by AIT. This format is only available when exporting CBA-based node configurations in NETCONF format.</p>



Table 34 Export Dialog Box Elements

Element	Description	Notes
Site Specific List	<p>Only displayed when Resolve variables using a Site Specific List is selected.</p> <p>Select a site specific list to resolve parameter value variables in the node configuration.</p>	<p>The Site Specific List field uses auto-complete functionality. Typing part of a name displays matching site-specific lists. Use the down-arrow on your keyboard to display all of the available lists. For more information on site-specific lists, see Section 13 on page 177.</p> <p>By default, PDB only shows site-specific lists that are associated with this configuration. Select Show All Site-Specific Lists to see all available lists.</p>
Custom Values (Table)	<p>Only displayed when Resolve variables manually is selected.</p> <p>Lists the parameter value variables in the selected configuration and allows you to input values manually.</p>	<p>All nodes have parameter value variables that are required by the PDB configuration script. These values must be supplied before a configuration can be exported.</p>

2. Enter the required information. See Table 34.

3. Click **Export**.

When exporting a node configuration, PDB follows a set of business rules that govern the format and structure of the configuration data. For more information on the PDB export criteria, refer to Section 15 on page 210.

The configuration is exported in the selected format. When PDB has finished processing the data, a link to the exported configuration is displayed.

4. Click **Download Configuration** to save the configuration archive.

8.9.2.1 Exporting a Configuration Report

Using PDB, you can export configuration data to a CSV file. This format can be post-processed to produce readable reports that detail the configuration. The exported CSV file combines information from the node configuration and the configuration schema to provide context and a better understanding of each parameter and parameter group.



A CSV report includes the following information:

- Node Name
- Configuration Name
- Configuration Document Number
- Configuration Revision
- Element Type
- MOC
- MOI
- Element Name
- Configured Value
- Default Value
- Parameter Type
- Element Description

Note: From the configuration schema.

- Comment

To generate a configuration report, export a configuration as outlined in Section 8.9.2 on page 105 and select **CSV (report)** as the export format.

8.9.3 Deploying and Executing a Node Configuration

Configurations exported in EAS, LDIF, or NETCONF format can be used to configure nodes directly.

Node configurations are exported in a compressed TAR or ZIP format. Each archive contains the `run_configure` script and the node configuration data.

Note: When deploying a configuration using a NETCONF format, Windows users must run the `run_configure.bat` script instead of the `run_configure.sh`.

Prerequisites

`run_configure` requires a password. The script will prompt for the password if it has not been specified using the `<application>_USER_DN_LDAP_PWD` or `<application>_NC_PASSWORD` CM variables. For more information on CM variables, see Section 18 on page 227.

To configure a node using a configuration bundle exported from PDB:

1. Deploy the configuration ZIP or TAR file where applicable.
2. Extract the archive within an empty transition directory.
3. If required, set permissions for `run_configure.sh` so that it can be executed in Linux.

```
chmod 777 run_configure.sh
```

4. Run the script.

During execution, `run_configure` produces a log file called `log.<pid>` in the local directory that contains all of the information displayed in the console.

The following example shows one possible command sequence for executing the configuration script in Linux.

```

setenv LD_LIBRARY_PATH /usr/local/lib
if [ $? != 0 ]; then
    echo "LD_LIBRARY_PATH not set"
fi

echo "LD_LIBRARY_PATH is set to: $LD_LIBRARY_PATH"

setenv LD_LIBRARY_PATH /usr/local/lib
if [ $? != 0 ]; then
    echo "LD_LIBRARY_PATH not set"
fi

echo "LD_LIBRARY_PATH is set to: $LD_LIBRARY_PATH"

setenv LD_LIBRARY_PATH /usr/local/lib
if [ $? != 0 ]; then
    echo "LD_LIBRARY_PATH not set"
fi

echo "LD_LIBRARY_PATH is set to: $LD_LIBRARY_PATH"

```

Example 7 Sample Execution of the Configuration Script.

8.9.4 Working With Dependencies on LDAP Configurations

In addition to the dependencies defined at configuration schema level, dependencies can also be associated with LDAP configurations to guide provisioning. When a configuration element has dependencies, the value of a second configuration element must be set to a specific value before or after configuring the original element.

For example, parameter A cannot be defined unless parameter B is set to 0. After parameter A has been set, parameter B must be set back to 1.

PDB uses the **ellsh** tool to configure nodes using LDAP. **ellsh** handles the following dependency scenarios:

pre	Parameter B is set before parameter A.
post	Parameter B is set after having set parameter A.
toggle	Parameter A is set to a value and its only toggled back to the original value at the end of the configuration process.

For more information on the syntax of LDAP dependencies, refer to the *ellsh 1.2 User Guide*, 198 17-CXP 902 0245.

PDB is pre-provisioned with several common dependencies (such as: DIAMETER stack dependencies, administrative states of some IMS nodes, and so on.):



however, the pre-provisioned list does not cover all of the possible dependencies. PDB administrators can add new dependency statements or override existing ones by making modifications to the dependencies file.

Note: PDB end users must contact an administrator to add, modify, or delete parameter dependencies.

To modify the dependencies file:

1. Use SSH to log in to the PDB server as glassfish.
2. Navigate to the file location:

```
cd /usr/local/glassfish/domains/domain1/config/pdb/scripts/
```

3. Make a backup copy of the dependencies file:

```
cp dependencies <backup file name>
```

4. Open the dependencies file for editing:

```
vi dependencies
```

Example 8 shows sample statements from the dependencies file.

```
...
pre "STARTING THE GLASSFISH SERVER"
{
  "STARTING THE GLASSFISH SERVER"
}

post "STARTING THE GLASSFISH SERVER"
{
  "STARTING THE GLASSFISH SERVER"
}

toggle "STARTING THE GLASSFISH SERVER"
{
  "STARTING THE GLASSFISH SERVER"
}

...
pre "STOPPING THE GLASSFISH SERVER"
{
  "STOPPING THE GLASSFISH SERVER"
}

post "STOPPING THE GLASSFISH SERVER"
{
  "STOPPING THE GLASSFISH SERVER"
}

toggle "STOPPING THE GLASSFISH SERVER"
{
  "STOPPING THE GLASSFISH SERVER"
}

...
```

Example 8 Sample dependencies File

Note: Each pre/post entry represents one line in the dependencies file.

5. Update the pre, post, and toggle operations as required.



For more information on the syntax of LDAP dependencies, refer to the *ellsh 1.2 User Guide*, 198 17-CXP 902 0245.

6. Exit *vi*, saving your changes:

```
:wq!
```

When PDB exports a configuration in LDIF format, it looks for the dependencies file to include in the configuration bundle under `/usr/local/glassfish/domains/domain1/config/pdb/scripts/`. If the dependencies file exists, PDB uses it to produce the configuration ZIP file. If the file does not exist, PDB uses the built-in dependencies file.

8.10 Deleting a Node Configuration

Authorized users can remove node configurations from PDB as needed.

Note: PDB cannot remove configurations in a locked state, parent configurations, a configuration that has been associated with node revisions, or any configuration that is part of a configuration set. In these cases, you must ensure that the configuration is no longer being referenced within PDB before attempting to delete it.

To delete a configuration:

1. In the **Configurations** workspace, right-click a configuration and select **Delete**.

Note: If the configuration is not displayed, perform a search as outlined in Section 8.1.1 on page 72.

A confirmation dialog box opens.

Caution!

Deleting a configuration permanently removes it from PDB.

2. Click **OK**.

The configuration is deleted.



9 Working with Configuration Data

Node configurations contain a collection of parameters and parameter groups that are used to configure the software on a node. These parameters and parameter groups (collectively called configuration data) are structured in accordance with a configuration schema that defines all of the legitimate classes and attributes for a specific node revision. For more information on configuration schemas, refer to Section 7 on page 40.

PDB provides a dedicated workspace for configuration data. The configuration browser is described in Section 9.1 on page 115.

Managing configuration data involves making modifications to parameters and parameter groups, such as:

- Adding a Parameter Group
- Commenting a Parameter Group
- Deleting a Parameter Group
- Adding a Parameter
- Modifying a Parameter
- Deleting a Parameter

PDB supports the following node configurations:

Root Configurations

Root configurations are self-contained and do not inherit any configuration data from any other source.

Delta Configurations

Delta configurations build upon and modify a parent configuration. The delta inherits all of its configuration data from the parent, except those elements that are locally overridden.

Delta configurations store only the changes that are made to the inherited data, including additions, deletions, and modifications. These updates supersede any corresponding elements that were inherited from the parent.

9.1 The Configuration Browser

The configuration browser is the primary way for users to manage configuration data in PDB.



Configuration data can be displayed in two possible formats:

- TREE view (Default)
- TABLE view

In the TREE view, PDB displays configuration data as a cascading tree of parameter groups, each group containing one or more parameters. In this view, information is divided between two frames.

The left frame shows a compressed list of parameter groups along with any parameter structs in those groups. The right frame displays the parameters or struct members contained within a selected parameter group or parameter struct. Selecting a configuration element in the left frame will display its contents in the right frame.

Parameters or struct members are presented in a table with the following information:

- Name
- Value
- Comment

See Figure 52.

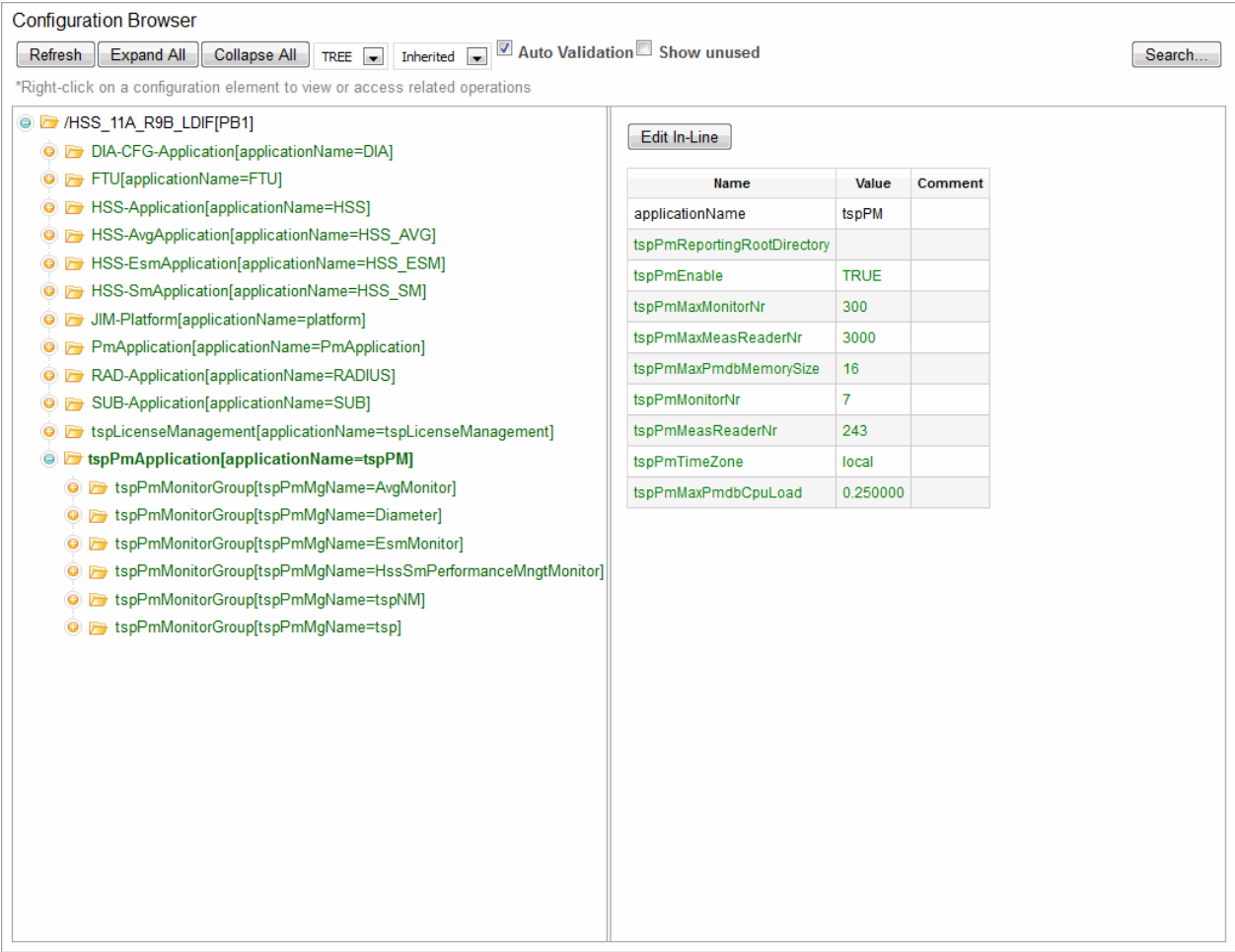


Figure 52 Node Configuration - Tree View

In the TABLE view, PDB displays configuration data as a table of nested parameter groups, each containing one or more parameters. The table includes the following information, where applicable:

— MOI

The Managed Object Instance (MOI) shows the path of the configuration element in the configuration tree.

Note: Only the last section of the MOI is shown in table view. The full path of the configuration element is available as a tooltip.

— Name

— Value

— Comment

See Figure 53.



Configuration Browser

Refresh Expand All Collapse All TABLE Inherited Auto Validation Show unused Search...

*Right-click on a configuration element to view or access related operations

	RAD-Application[applicationName=RADIUS]	RAD-Application		
	SUB-Application[applicationName=SUB]	SUB-Application		
	tspLicenseManagement[applicationName=tspLicenseManagement]	tspLicenseManagement		
	tspPmApplication[applicationName=tspPM]	tspPmApplication		
	applicationName	tspPM		
	tspPmReportingRootDirectory			
	tspPmEnable	TRUE		
	tspPmMaxMonitorNr	300		
	tspPmMaxMeasReaderNr	3000		
	tspPmMaxPmdbMemorySize	16		
	tspPmMonitorNr	7		
	tspPmMeasReaderNr	243		
	tspPmTimeZone	local		
	tspPmMaxPmdbCpuLoad	0.250000		
	tspPmMonitorGroup[tspPmMgName=AvgMonitor]	tspPmMonitorGroup		
	tspPmMonitorGroup[tspPmMgName=Diameter]	tspPmMonitorGroup		
	tspPmMonitorGroup[tspPmMgName=EsmMonitor]	tspPmMonitorGroup		
	tspPmMonitorGroup[tspPmMgName=HssSmPerformanceMngtMonitor]	tspPmMonitorGroup		
	tspPmMonitorGroup[tspPmMgName=tspNM]	tspPmMonitorGroup		
	tspPmMonitorGroup	tspPmMonitorGroup		

Figure 53 Node Configuration - Table View

To switch between TREE and TABLE views, make a selection from the **Format** drop-down list at the top of the page. Selected elements are not affected by switching between views.

When working with configuration data, colors are used to provide extra information about the configuration elements as follows:

Table 35 Color Use in Node Configurations

Color	Sample Representation	Description
Black	applicationName	Primary Keys.
Green	HSS-Application[applicationName=HSS]	Elements that are native to the current configuration.
Grey	HSS-Application[applicationName=HSS]	Elements that are inherited from a parent configuration. This data is not native to the current configuration.



Table 35 Color Use in Node Configurations

Color	Sample Representation	Description
Red	HSS-Application[applicationName=HSS]	Inherited data that has been removed from the delta configuration. Removed elements are only displayed in the Delta Only view. Note: Only applicable for delta configurations.
Purple	HSS-Application[applicationName=HSS]	Unused elements that are not instantiated in the current configuration. Note: Unused elements are only visible when Show Unused is selected.

To browse a configuration in PDB:

- In the **Configurations** workspace, select a configuration to browse and click **Open**.

Note: If the configuration is not displayed, perform a search as outlined in Section 8.1.1 on page 72.


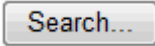
Elements of the selected configuration are displayed in the configuration browser.

To facilitate working with large configurations, the configuration browser includes a search functionality that allows you to filter the displayed list of configuration elements. For more information on the search functionality, refer to Section 9.1.1 on page 121.

The following table describes the elements of the configuration browser:



Table 36 Configuration Browser Elements

Element	Description
Format (List)	Sets the display format of the configuration. TREE: Displays configuration elements as a cascading tree of parameter groups, each containing one or more parameters. TABLE: Displays configuration elements as a table of nested parameter groups, each containing one or more parameters.
Views (List)	Sets the display criteria for the node configuration. Inherited : Shows local and inherited configuration elements. All inherited configuration elements are displayed except those that are locally overridden. Delta Only: Restricts the view to the local configuration, including any configuration elements that have been locally deleted.
Auto Validation (Check Box)	Enables or disables automatic validation of configuration changes.
Show Unused (Check Box)	Displays configuration elements that are defined in the schema but have not been instantiated in the current configuration. Shows the default values, if available.
	Opens the parameters table in edit mode.
	Perform a search of the current configuration.

The configuration browser uses contextual menus to work with parameters and parameter groups. Simply right-click a configuration element to access the available operations. These operations include:

Add Parameter Group

Adds a new parameter group at the selected location in the configuration tree.



Add Parameter	Adds a new parameter to the selected parameter group.
Add Parameter Struct	Adds a parameter struct to the selected parameter group.
Edit	Opens the selected configuration element for editing.
Edit In-Line	Opens the parameters table in edit mode.
Block All	Blocks the inheritance of all instances of the selected parameter type. Note: The block all functionality is not available for root configurations.
History	View the history of a selected parameter across configuration revisions. Tracks changes to the parameter value.
Schema Information	Displays information about the selected configuration element.

9.1.1 Searching Node Configurations

The configuration browser includes a search functionality that locates specific configuration elements using a number of different criteria.

The search field accepts keywords to narrow down the search results. The syntax for using keywords is as follows:

<keyword1>:value1 <keyword2>:value2 <keyword3>:value3 ...

The following table describes the available keywords.

Table 37 Search Keywords

Keyword	Description
name: <string>	Searches for parameter and parameter group names that contain the input string. Searching on element names is the default search operation and is performed when no search criteria are used.
value: <string>	Searches for parameter values and parameter group primary keys that contain the input string.



Keyword	Description
comment: <string>	Searches for parameter and parameter group comments that contain the input string.
description: <string>	Searches for parameter and parameter group descriptions that contain the input string.
moi: <string>	Searches for parameter group DNs that contain the input string.
status: <status>	Searches for parameters and parameter groups matching the selected status. Valid status types include: <ul style="list-style-type: none">• current• deprecated• obsolete
category: <category>	Searches for parameters and parameter groups matching the selected category. Valid categories include: <ul style="list-style-type: none">• Internal• Operator Configurable• Site Specific• Solution Integration



Keyword	Description
iskey: <boolean>	<p>Filters parameters for primary keys.</p> <ul style="list-style-type: none">• true - Filters out parameters that are not primary keys.• false - Filters out parameters that are primary keys. <p>Note: Because primary keys are a type of parameter, parameter groups will not be included in the search results when iskey is set to true.</p>
type: <type>	<p>Searches for configuration elements matching the selected type. Valid types include:</p> <ul style="list-style-type: none">• p• pg



Keyword	Description
readonly: <boolean>	<p>Filters parameters based on the readonly attribute.</p> <ul style="list-style-type: none">• true - Filters out parameters that are not read-only.• false - Filters out parameters that are read-only. <p>Note: Because parameter groups do not have a readonly attribute, they are included in all searches using this keyword unless they are removed by other search criteria.</p>
restricted: <boolean>	<p>Filters parameters based on the restricted attribute.</p> <ul style="list-style-type: none">• true - Filters out parameters that are not restricted.• false - Filters out parameters that are restricted. <p>Note: Because parameter groups do not have a restricted attribute, they are included in all searches using this keyword unless they are removed by other search criteria.</p>

To perform a search:

1. In the configuration browser, click **Search**.

The **Search** dialog box is displayed. See Figure 54.



Search

Number of results 0

		Parameter Name	Value	MOC	MOI
<input type="checkbox"/>					

Bulk Operations

Delete... Replace...

Figure 54 Search Box

2. Enter your search query.

For a basic search, type the name or value of the configuration element you are searching for.

Note: Search strings are not case-sensitive. All searches return a partial match unless the string is surrounded by double quotes.

For a keyword search, compose a search string using one or more valid keywords. See Table 37 for a description of the available search options.

Note: Different keywords can be combined to refine the search pattern.

Some sample search queries, include:

- **name: tspPmEnable type:P**

This query would search for parameters named tspPmEnable.

- **type:PG status:obsolete**

This query would search for parameter groups that are currently set as obsolete.

- **comment: for performance management**

This query would search for parameters or parameter groups that contain a for performance management comment.

- **moi:tspPmApplication[applicationName=tspPM] type:P**

This query would search for parameters located at the specified location in the configuration tree.

3. Click Search.



The search hits are listed in the table at the bottom of the **Search** dialog box. See Figure 55.

Note: If you have enabled **Show Unused** in the configuration browser, unused configuration elements will be included in the search results.

Search results are listed with the following information, where applicable:

- Parameter Name
- Value
- MOC
- MOI

The screenshot shows a 'Search' dialog box with a search input field containing 'FTU'. Below the input field, it says 'Number of results 5'. There are 'Search' and 'Clear' buttons. An 'Export to CSV' button is in the top right. The results are displayed in a table with columns: Parameter Name, Value, MOC, and MOI. There are also 'Bulk Operations' buttons: 'Delete...' and 'Replace...'.

	Parameter Name	Value	MOC	MOI
			FTU	FTU[applicationName=FTU]
	applicationName	FTU	FTU	FTU[applicationName=FTU]
			tspFtuKeyReference	FTU[applicationName=FTU],tspFtuKeyReference[tspFtuKeyReferenceKey=0]
	tspFtuKeyReferenceKey	0	tspFtuKeyReference	FTU[applicationName=FTU],tspFtuKeyReference[tspFtuKeyReferenceKey=0]
	tspFtuKey	0	tspFtuKeyReference	FTU[applicationName=FTU],tspFtuKeyReference[tspFtuKeyReferenceKey=0]

Figure 55 Search Results

4. Click the magnifying glass () next to one of the search results to open the configuration tree at that element's location.

The search is complete.

Search results are available in CSV format. Click **Export to CSV** to download the CSV file.

9.1.1.1

Bulk Operations

When working with an unlocked configuration, the configuration browser allows you to perform bulk operations on search results. Using bulk operations, you can edit or delete multiple parameter instances simultaneously.

Bulk Replacement

To perform bulk replacement of parameter values:

1. Perform a search for the required parameters as described in Section 9.1.1 on page 121.



The search results are listed.

2. Using the check boxes, select the parameters to update.

Note: PDB does not allow you to perform bulk operations on parameter groups. No check box is available for these configuration elements.

3. Click **Replace**.

The **Bulk Operations** area is displayed. See Figure 56.

Bulk Operations

Replacement Pattern: ☒ substring ☐ exact match ☐ regular expression *Replace all occurrences.*

Replacement Value: ☒ substring ☐ regular expression *Replace pattern matches with the selected value.*

Comment: *Add a comment to all modified parameters.*

Figure 56 Search Bulk Operations

The following table describes the elements of the **Bulk Operations** area:

Table 38 Add Parameter Group Dialog Box Elements

Element	Description
Replacement Pattern	Identifies the parameter value to replace. Values are matched using one of the following options: <ul style="list-style-type: none"> • substring - Matches all occurrences of the pattern. • exact match - Matches only the exact pattern. • regular expression - Matches based on the selected regex.
Replacement Value	The replacement value for pattern matches. Values are replaced using on of the following options: <ul style="list-style-type: none"> • substring - Replaces pattern matches with the selected value. • regular expression - Replaces pattern matches using regex capture groups.⁽¹⁾
Comment	Add a descriptive comment to all modified parameters. Existing comments will be replaced.

(1) Only available when a regular expression has been used for the replacement pattern.

4. Enter the required information. See Table 38.
5. If required, click **Preview** to see your changes reflected in the table of search results.

Note: The preview operation does not modify configuration data.

6. Click **Apply**.



All pattern matches are replaced with the selected value.

By default, PDB automatically performs a validation check when modifying parameters and displays the results, if applicable. The check verifies that the bulk replacement is in line with the schema definition for this configuration. For a description of the validation check, refer to Section 8.8 on page 101.

Note: PDB tracks parameter modifications across configuration revisions. Changes to the parameter value are captured in the **Parameter Value History**.

To view the history of a parameter, select it in the configuration browser and click **History**.

Bulk Delete

Caution!

Deleting local data from a configuration is a permanent operation.

Deleting inherited elements does not remove them from the parent configuration. When an inherited element is removed from a delta configuration, it is replaced with a deleted data reference that is accessible at the corresponding location in the **Delta Only** view. This data reference can be restored to bring back the inherited data.

If a configuration has inherited several instances of a given parameter type, all instances can be blocked at the same time using **Block All**.

To perform a bulk delete of configuration parameters:

1. Perform a search for the required parameters as described in Section 9.1.1 on page 121.

The search results are listed.

2. Using the check boxes, select the parameters to delete.

Note: PDB does not allow you to perform bulk operations on parameter groups. No check box is available for these configuration elements.

3. Click **Delete**.



Note: Primary key parameters cannot be deleted in this way. To remove a primary key parameter, you must delete the associated parameter group. For more information on deleting parameter groups, refer to Section 9.2.4 on page 133.

A confirmation dialog box opens.

- 4. Click **Apply**.

The selected parameters are deleted.

9.2 Working With Parameter Groups

Within the constraints of the schema, you can add or remove parameter groups and provide descriptive comments.

9.2.1 Adding Parameter Groups

The configuration browser can be used to add new parameter groups to the local configuration.

To add a new parameter group:

- 1. In the configuration browser, right-click a parent for the new parameter group and select **Add Parameter Group**.

If needed, you can perform a search as described in Section 9.1.1 on page 121.

The **Add Parameter Group** dialog box opens. See Figure 57.

Name	Value
HSS-ConfigurationContainerName	

Figure 57 Add Parameter Group Dialog Box

The following table describes the elements of the **Add Parameter Group** dialog box:



Table 39 Add Parameter Group Dialog Box Elements

Element	Description	Notes
Name (List)	A list of available parameter groups that can be added at this location. The list of names is derived from the schema definition and indicates what can be added at the selected level of the tree.	Mandatory. When adding an unused parameter group, the corresponding name is preselected and cannot be changed.
Comment	Add a descriptive comment to the new parameter group.	Optional. In the TREE view, parameter group comments are presented as tool tips.
Primary Key (Table)	Lists the primary key for the selected parameter group and allows you to define its value. If the copied parameter group has multiple primary keys, all of them will be shown.	Mandatory. This field is automatically populated with a default value for the corresponding primary key that is derived from the schema.
Schema Information	Displays information about the selected parameter group (Type) that is stored in the schema.	

2. Enter the required information. See Table 39.

3. Click **Apply**.

The parameter group is added to the local configuration.

By default, PDB automatically performs a validation check when adding parameter groups and displays the results, if applicable. The check verifies that the new addition is in line with the schema definition for this configuration. For a description of the validation check, refer to Section 8.8 on page 101.

9.2.2 Copying and Pasting a Parameter Group

The configuration browser can be used to copy a parameter group and paste it within the same configuration.

Copying a parameter group captures all configuration elements within the selected group, including subgroups and parameters.

To copy and paste a parameter group:

1. In the configuration browser, right-click a parameter group and select **Copy**.



If needed, you can perform a search as described in Section 9.1.1 on page 121.

Note: Root elements and singletons cannot be copied.

The parameter group is copied.

2. Navigate the configuration and select a location to paste the parameter group.

If needed, you can perform a search as described in Section 9.1.1 on page 121.

Note: The configuration schema must allow the copied parameter group to be inserted at the selected location.

3. Right-click the selected location and select **Paste**.

The **Copy/Paste Parameter Group** dialog box opens. See Figure 58.

Figure 58

The following table describes the elements of the **Copy/Paste Parameter Group** dialog box:

Table 40 Copy/Paste Parameter Group Dialog Box Elements

Element	Description	Notes
Name	The name of the copied parameter group.	Mandatory. When copying and pasting a parameter group, the corresponding name is preselected and cannot be changed.
Comment	Add an optional comment to the copied parameter group.	Optional. In the TREE view, parameter group comments are presented as tool tips.



Table 40 Copy/Paste Parameter Group Dialog Box Elements

Element	Description	Notes
Primary Key (Table)	Lists the primary key for the copied parameter group and allows you to define its value. If the copied parameter group has multiple primary keys, all of them will be shown.	Mandatory.
Schema Information	Displays information about the copied parameter group (Type) that is stored in the schema.	

4. Enter the required information. See Table 40.

5. Click **Apply**.

The copied parameter group is pasted under the selected location.

By default, PDB automatically performs a validation check when pasting parameter groups and displays the results, if applicable. The check verifies that the new addition is in line with the schema definition for this configuration. For a description of the validation check, refer to Section 8.8 on page 101.

9.2.3 Commenting a Parameter Group

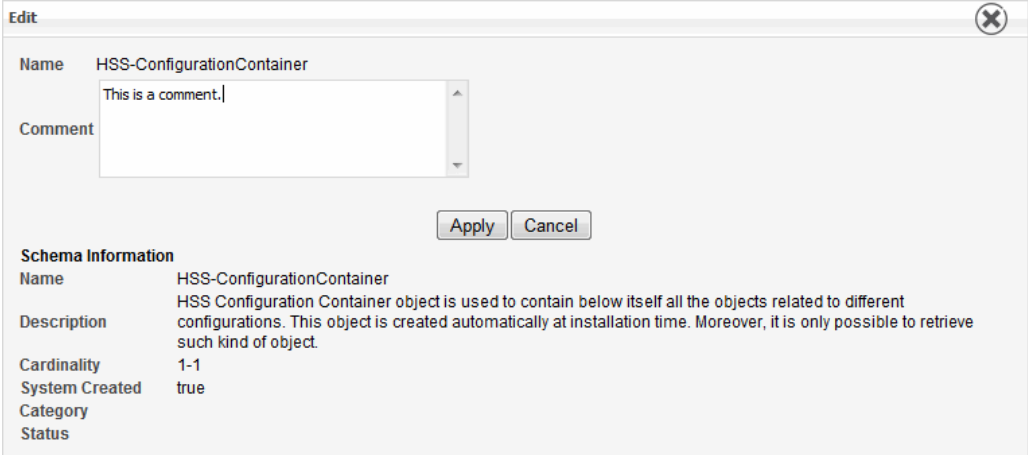
The configuration browser can be used to add, modify, or remove descriptive comments for parameter groups in the local configuration.

To comment on a parameter group:

1. In the configuration browser, right-click a parameter group to comment and select **Edit**.

If needed, you can perform a search as described in Section 9.1.1 on page 121.

The **Edit** dialog box opens. See Figure 59.

The dialog box is titled "Edit" and contains the following fields and sections:

- Name:** HSS-ConfigurationContainer
- Comment:** A text area containing "This is a comment." with a vertical scrollbar.
- Buttons:** "Apply" and "Cancel" buttons.
- Schema Information:**
 - Name:** HSS-ConfigurationContainer
 - Description:** HSS Configuration Container object is used to contain below itself all the objects related to different configurations. This object is created automatically at installation time. Moreover, it is only possible to retrieve such kind of object.
 - Cardinality:** 1-1
 - System Created:** true
 - Category:**
 - Status:**

Figure 59 Edit Dialog Box

2. Add, modify, or remove your comment in the **Comment** field.
3. Click **Apply**.

The parameter group is updated.

9.2.4

Deleting a Parameter Group

Authorized users can remove parameter groups from a configuration.

Caution!

Deleting local data from a configuration is a permanent operation.

Deleting inherited elements does not remove them from the parent configuration. When an inherited element is removed from a delta configuration, it is replaced with a deleted data reference that is accessible at the corresponding location in the Delta Only view. This data reference can be restored to bring back the inherited data.

To delete a parameter group from the local configuration:

1. In the configuration browser, right-click a parameter group to remove and select **Delete**.

If needed, you can perform a search as described in Section 9.1.1 on page 121.

A confirmation dialog box opens.



Caution!

Deleting a parameter group also removes all child parameter groups and any associated parameters.

2. Click **OK**.

The selected parameter group is removed from the local configuration.

9.3 Working with Parameters

Within the constraints of the schema, you can add, modify, or remove configuration parameters and parameter structs.

9.3.1 Adding Parameters

The configuration browser can be used to add new parameters to the local configuration. Parameters can be added using the **Add Parameter** dialog box, described here, or directly in the configuration browser using in-line editing. For more information on using in-line editing to add new parameters, refer to Section 9.3.1.1 on page 136.

To add a new parameter using the **Add Parameter** dialog box:

1. In the configuration browser, right-click a parameter group that will contain the new parameter and select **Add Parameter**.

If needed, you can perform a search as described in Section 9.1.1 on page 121.

The **Add Parameter** dialog box opens. See Figure 60.

The screenshot shows the 'Add Parameter' dialog box. On the left, there are input fields for 'Name' (containing 'HSS-ApplicationServerName'), 'Value', and 'Comment'. Below these are 'Apply' and 'Cancel' buttons. On the right, a 'Schema Information' table provides details about the parameter.

Schema Information	
Name	HSS-ApplicationServerName
Description	This attribute is the application server's SIP URI
Cardinality	0-1
Data Type	String
Default Value	
Initial Value	
Read Only	false
Restricted	false
Unit	
Format Description	Data type: SIP URI.
Value Constraints	Pattern: (sip:).*
Category	Site Specific
Status	
Related Parameter (s)	

Figure 60 Add Parameter Dialog Box



The following table describes the different elements forming the **Add Parameter** dialog box:

Table 41 Add Parameter Dialog Box Elements

Element	Description	Notes
Name (List)	A list of available parameters that can be added at this location. The list of names is derived from the schema definition and indicates what can be added at the selected level of the tree.	Mandatory. When adding an unused parameter, the corresponding name is preselected and cannot be changed.
Value	The value of the new parameter.	Mandatory. This field is automatically populated with a default value for the selected parameter that is derived from the schema. Values can be input as parameter value variables. These variables are populated with real values at export time using a site-specific list. For more information, refer to Section 9.3.4 on page 144. This field supports auto-complete for global variables. Opening a new variable with {{ triggers the auto-complete. For more information on global variables, refer to Section 13.3 on page 190.
Comment	Add a descriptive comment to the new parameter.	Optional.
Schema Information	Displays information about the selected parameter (Type) that is stored in the schema.	

2. Enter the required information. See Table 41.
3. Click **Apply**.

The parameter is added to the local configuration.

By default, PDB automatically performs a validation check when adding parameters and displays the results, if applicable. The check verifies that the

new addition is in line with the schema definition for this configuration. For a description of the validation check, refer to Section 8.8 on page 101.

If this parameter has been populated with a parameter value variable, the validation check will verify that the variable is included on the Global Variables list. PDB warns if the variable is not defined in the global list, but does not prevent the use of custom variables.

9.3.1.1 Adding Parameters Using In-Line Editing

The PDB configuration browser allows you to add parameters directly to a configuration using in-line editing. This process makes it easier to work with multiple parameters at the same time, allowing you to make additions in a single step.

Note: In-line editing can only add unused parameters to a node configuration.

To add parameters using in-line editing:

1. In the configuration browser, select the following options:
 - Enable TREE view (default) by selecting TREE from the format list.
 - Enable the display of unused configuration elements by selecting the **Show Unused** check box.

For more information on these configuration browser options, refer to Section 9.1 on page 115.

2. Navigate to a parameter group containing the unused parameters that you want to add.

If needed, you can perform a search as described in Section 9.1.1 on page 121.

The list of parameters is displayed in the right frame. See Figure 61.

Note: Unused parameters that are not instantiated in the current configuration are colored purple and are commented with Not present. Existing parameters can be modified while adding new elements. For more information on modifying parameters using in-line editing, refer to Section 9.3.3.1 on page 142.



Edit In-Line		
Name	Value	Comment
SLF-LicenseName	SLF-License	
SLF-BasicActivationEnabled	FALSE	Not present
SLF-DBManagerEnabled	FALSE	Not present
SLF-DiameterProxyEnabled	FALSE	Not present
SLF-DiameterRedirectEnabled	FALSE	Not present
SLF-LoadbalancerEnabled	FALSE	Not present
SLF-RadiusProxyEnabled	FALSE	Not present

Figure 61 Parameters Table Showing Unused Parameters

- Click **Edit In-Line** at the top of the parameters table or right-click a parameter and select **Edit In-Line**.

The parameters table opens in edit mode. See Figure 62.

<div> <div>Apply</div> <div>Cancel</div> </div>			
Name	Value	Comment	Create
SLF-LicenseName	<input type="text" value="SLF-License"/>	<input type="text"/>	
SLF-BasicActivationEnabled	<input type="text" value="FALSE"/>	<input type="text"/>	<input type="checkbox"/>
SLF-DBManagerEnabled	<input type="text" value="FALSE"/>	<input type="text"/>	<input type="checkbox"/>
SLF-DiameterProxyEnabled	<input type="text" value="FALSE"/>	<input type="text"/>	<input type="checkbox"/>
SLF-DiameterRedirectEnabled	<input type="text" value="FALSE"/>	<input type="text"/>	<input type="checkbox"/>
SLF-LoadbalancerEnabled	<input type="text" value="FALSE"/>	<input type="text"/>	<input type="checkbox"/>
SLF-RadiusProxyEnabled	<input type="text" value="FALSE"/>	<input type="text"/>	<input type="checkbox"/>

Figure 62 Unused Parameters in Edit Mode

- For each unused parameter, enter the required information and select the **Create** check box.

If an unused parameter has a default value, the default will be set as the initial value.

For more information on the parameter fields, refer to Section 9.3.1 on page 134.

Note: Unused parameters can only be added to a parameter group that is part of the node configuration. If the selected parameter group is also unused, the corresponding primary key will be automatically selected for addition.

5. Click **Apply**.

The new parameters are added to the node configuration.

9.3.2 Adding Parameter Structs

Where permitted by the configuration schema, parameter structs can be added to the local configuration. The configuration browser treats parameter structs in the same way as parameter groups. In TREE view, a parameter struct is added in the left frame, along with parameter groups, and the associated struct members are modified in the right frame.

To add a new parameter struct:

1. In the configuration browser, right-click a parameter group that will contain the new parameter struct and select **Add Parameter Struct**.

If needed, you can perform a search as described in Section 9.1.1 on page 121.

The **Add Parameter Struct** dialog box opens. See Figure 63.

The screenshot shows the 'Add Parameter Struct' dialog box. It has a title bar with a close button. The dialog is divided into several sections:

- Name:** A dropdown menu showing 'agentAddress'.
- Comment:** A text area.
- Click on Data Type for more info and description of each struct member.** A text label.
- Struct Member(s):** A table with two columns: 'Name' and 'Value'. It contains two rows: 'host' and 'port'.
- Schema Information:** A section containing various properties:
 - Name:** agentAddress
 - Description:** The listen addresses for the SNMP agent. A sequence of interfaces and ports which is used by the SNMP agent for listening on incoming SNMP requests. The typical configuration is using one port with IP address 0.0.0.0 and port 161, which means that the standard SNMP port is used on all interfaces.
 - Is Read Only:** false
 - Restricted:** false
 - Cardinality:** 0-2147483647
 - Derived Data Type Name:**
 - Default Value:**
 - Initial Value:**
 - Collection Type:** Set (isOrdered=false, isUnique=true)
 - Unit:**
 - Format Description:**
 - Value Constraints:**
 - Category:**
 - Status:** false
 - PmData:**
 - Related Parameter(s):**
 - Data Type:** HostAndPort Structure

At the bottom, there are 'Apply' and 'Cancel' buttons.

Figure 63 Add Parameter Struct Dialog Box

The following table describes the different elements forming the **Add Parameter Struct** dialog box:



Table 42 Add Parameter Struct Dialog Box Elements

Element	Description	Notes
Name (List)	A list of available parameter structs that can be added at this location. The list of names is derived from the schema definition and indicates what can be added at the selected level of the tree.	Mandatory. When adding an unused parameter struct, the corresponding name is preselected and cannot be changed.
Comment	Add a descriptive comment to the new parameter struct.	Optional.
Struct Member(s) (Table)	A table listing the name of all members in the selected parameter struct. Each struct member must be assigned a value in the corresponding Value field.	Mandatory. When available, struct members are automatically populated with a default value that is derived from the schema. Values cannot be input as parameter value variables.
Schema Information	Displays schema information for the selected parameter struct.	The Data Type can be expanded to show the schema definition for the parameter struct and all struct members.

2. Enter the required information. See Table 42.

3. Click **Apply**.

The parameter struct is added to the local configuration.

By default, PDB automatically performs a validation check when adding parameter structs and displays the results, if applicable. The check verifies that the new addition is in line with the schema definition for this configuration. For a description of the validation check, refer to Section 8.8 on page 101.

9.3.3 Modifying Parameters or Parameter Structs

The properties of an existing parameter or parameter struct can be modified. In addition to these changes, PDB also allows you to edit Primary Key parameters that are local to the configuration.

Note: Modifying a Primary Key changes the DN of the associated parameter group.

Any Primary Key parameters that are inherited from the parent configuration cannot be modified in a delta configuration.



Parameters and struct members can be modified through the **Edit** dialog box, described here, or directly in the configuration browser using in-line editing. For more information on using in-line editing to modify parameters or struct members, refer to Section 9.3.3.1 on page 142.

To update a parameter or parameter struct using the **Edit** dialog box:

1. In the configuration browser, right-click a configuration element to modify and select **Edit**.

If needed, you can perform a search as described in Section 9.1.1 on page 121.

The **Edit** dialog box opens. See Figure 64.

Schema Information	
Name	HSS-ApplicationServerName
Description	This attribute is the application server's SIP URI
Cardinality	0-1
Data Type	String
Default Value	
Initial Value	
Read Only	false
Restricted	false
Unit	
Format Description	Data type: SIP URI.
Value Constraints	Pattern: (sip:).*
Category	Site Specific
Status	
Related Parameter(s)	

Figure 64 Edit Dialog Box



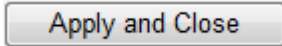

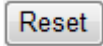
The following table describes the elements of the **Edit** dialog box:

Table 43 Edit Parameter Dialog Box Elements

Element	Description	Notes
Name	The name of the parameter or parameter struct.	
Value	Lists the current value for the selected parameter and allows you to modify it.	For parameters only. Values can be input as parameter value variables. These variables are populated with real values at export time using a site-specific list. For more information, refer to Section 9.3.4 on page 144.



Table 43 Edit Parameter Dialog Box Elements

Element	Description	Notes
Comment	Update the comment for this parameter or parameter struct.	Optional.
Struct Member(s) (Table)	Lists the name and current value of all members in a parameter struct.	For parameter structs only. Values cannot be input as parameter value variables.
Schema Information	Displays information about the selected parameter that is stored in the schema.	
	Moves to the previous parameter or parameter struct instance within the same parameter group.	Includes primary keys.
	Apply the changes and keep the Edit dialog box open.	
	Apply the changes and close the Edit dialog box.	
	Moves to the next parameter or parameter struct instance within the same parameter group.	Includes primary keys.
	Resets the configuration element to the last saved state.	Includes primary keys.

2. Update the **Value** or **Comment** fields, as required.
3. Apply your changes.

The configuration element is updated. If required you can move directly to an adjacent parameter or parameter struct instance using the navigation buttons.

By default, PDB automatically performs a validation check when modifying a configuration element and displays the results, if applicable. The check verifies that your change is in line with the schema definition for this configuration. For a description of the validation check, refer to Section 8.8 on page 101.

If this edit has populated a parameter with a parameter value variable, the validation check will verify that the variable is included on the Global Variables list. PDB warns if the variable is not defined in the global list, but does not prevent the use of custom variables.



Note: PDB tracks modifications to parameters and parameter structs across configuration revisions. Value changes are captured in the **Parameter Value History**.

To view the history of a parameter or parameter struct, select it in the configuration browser and click **History**.

9.3.3.1 Modifying Parameters and Struct Members Using In-Line Editing

The PDB configuration browser allows you to modify parameters or struct members using in-line editing. This process makes it easier to work with multiple values at the same time, allowing you to make several changes in a single step.

To modify parameters or struct members using in-line editing:

1. In the configuration browser, enable TREE view (default) by selecting TREE from the format list.

For more information on configuration browser options, refer to Section 9.1 on page 115.

2. Select a parameter group or parameter struct containing the configuration elements that you want to modify.

If needed, you can perform a search as described in Section 9.1.1 on page 121.

A list of parameters or struct members is displayed in the right frame. See Figure 65.

Note: Unused parameters that are not instantiated in the current configuration are colored purple and are commented with Not present. These parameters can be added to the configuration while modifying other elements. For more information on adding unused parameters using in-line editing, refer to Section 9.3.1.1 on page 136.



Edit In-Line		
Name	Value	Comment
applicationName	tspPM	
tspPmReportingRootDirectory		
tspPmEnable	TRUE	
tspPmMaxMonitorNr	300	
tspPmMaxMeasReaderNr	3000	
tspPmMaxPmdbMemorySize	16	
tspPmMonitorNr	7	
tspPmMeasReaderNr	243	
tspPmTimeZone	local	
tspPmMaxPmdbCpuLoad	0.250000	

Figure 65 Parameters Table

- Click **Edit In-Line** at the top of the table or right-click a configuration element and select **Edit In-Line**.

The parameters table opens in edit mode. See Figure 66.



Name	Value	Comment
applicationName	<input type="text" value="tspPM"/>	<input type="text"/>
tspPmReportingRootDirectory	<input type="text"/>	<input type="text"/>
tspPmEnable	<input type="text" value="TRUE"/>	<input type="text"/>
tspPmMaxMonitorNr	<input type="text" value="300"/>	<input type="text"/>
tspPmMaxMeasReaderNr	<input type="text" value="3000"/>	<input type="text"/>
tspPmMaxPmdbMemorySize	<input type="text" value="16"/>	<input type="text"/>
tspPmMonitorNr	<input type="text" value="7"/>	<input type="text"/>
tspPmMeasReaderNr	<input type="text" value="243"/>	<input type="text"/>
tspPmTimeZone	<input type="text" value="local"/>	<input type="text"/>
tspPmMaxPmdbCpuLoad	<input type="text" value="0.250000"/>	<input type="text"/>

Figure 66 Parameters in Edit Mode

- For each parameter or struct member, enter the required information.

For more information on the parameter fields, refer to Section 9.3.3 on page 139.

- Click **Apply**.

The updated configuration elements are added to the node configuration.

9.3.4 Site-Specific Variables

Node configurations can include a number of site-specific parameters that define parts of the target network environment, such as IP addresses, port numbers, domain names, and login credentials, that cannot be known in advance. To facilitate working with site-specific information, PDB allows you to populate these values with parameter value variables.

Site-specific variables act as a place holder for legitimate values. When exporting a node configuration in EAS, LDIF, or NETCONF format, these variables must be resolved manually or through the use of a site-specific list.

- For more information on exporting node configurations, refer to Section 8.9 on page 104.
- For more information on site-specific lists, refer to Section 13 on page 177.



Using variables to populate the site-specific parameters inside a node configuration allows it to function independently of site-specific information.

PDB identifies a parameter value variable as a string defined between double curly brackets `{{` and `}}` that uses the following supported characters:

- alpha-numeric (a,A, to z,Z and 0-9)
- underscore (_)
- dash (-)

For example, the string `{{server_abc_ip_address}}` is considered to be a site-specific variable called `server_abc_ip_address`.

PDB does not support nested site-specific variables, such as `{{somename@{{domain}}}}`.

PDB includes a global list of parameter value variables. This list allows authorized users to define a set of variables that are recommended for use in PDB. Global variables can be added to a node configuration using auto-complete functionality.

Variables are defined for site-specific parameters in the same way as regular values. For more information, refer to the previous sections.

9.3.5 Deleting Parameters or Parameter Structs

Authorized users can remove parameters or parameter structs from a configuration.

Note: Individual struct members and primary key parameters cannot be deleted in this way.

To remove a primary key parameter, you must delete the associated parameter group. For more information on deleting parameter groups, refer to Section 9.2.4 on page 133.

To remove a struct member, the entire parameter struct must be deleted.

Caution!

Deleting local data from a configuration is a permanent operation.

Deleting inherited elements does not remove them from the parent configuration. When an inherited element is removed from a delta configuration, it is replaced with a deleted data reference that is accessible at the corresponding location in the `Delta Only` view. This data reference can be restored to bring back the inherited data.



If a configuration has inherited several instances of a given parameter type, all instances can be blocked at the same time using **Block All**.

To delete a parameter or parameter struct from the local configuration:

1. In the configuration browser, right-click a parameter or parameter struct to remove and select **Delete**.

If needed, you can perform a search as described in Section 9.1.1 on page 121.

A confirmation dialog box opens.

2. Click **OK**.

The selected parameter or parameter struct is removed from the local configuration.

10 Comparing Two Configurations

Using PDB you can compare the contents of two node configurations to identify the differences between them. Both parameters and parameter groups are analyzed in a comparison and any difference in value is identified.

When performing a comparison, PDB distinguishes between an Original Configuration and a Target Configuration. The Original Configuration serves as a starting point for the comparison. Data from the Original Configuration is compared against a Target Configuration. Both the Original Configuration and the Target Configuration can be defined within PDB, or uploaded specifically for comparison purposes. A comparison of uploaded configurations is performed on the fly, without creating persistent data in PDB. Performing a comparison on the fly can be useful when comparing the configuration of two live nodes or comparing a real configuration against a reference configuration defined in PDB.

Comparison results are output to a table on screen. The Configuration Comparison Report is a customizable printout of the differences between the Original and the Target configurations. For more information on the comparison options, refer to Section 10.1.1 on page 147.

Comparison results can be used to generate a new delta configuration that hangs from the Original Configuration. This delta contains only the differences between the Original and Target configurations. When combined, data from the delta configuration and the Original Configuration is identical to the Target Configuration. For more information on generating a delta configuration from comparison results, refer to Section 10.3 on page 158.



The **Configuration Comparison Report** can be downloaded as a Comma Separated Values (CSV) file.

10.1 The Configuration Comparison Workspace

The **Configuration Comparison** workspace allows you to compare node configurations.

To access the **Configuration Comparison** workspace, select **Comparison** from the menu options on the left. See Figure 67.

Figure 67 The Configuration Comparison Workspace

The workspace is divided into three areas as follows:

Configuration Selection

Allows you to select an **Original Configuration** and a **Target Configuration** to compare.

Comparison Options

Allows you to customize the comparison report by setting a number of comparison options. For more information on comparison options, refer to Section 10.1.1 on page 147.

Manage Columns

Allows you to control what information is displayed in the comparison report table by showing or hiding individual columns. For more information on managing columns, refer to Section 10.1.1.1 on page 152.

Note: The **Comparison Options** and **Manage Columns** areas are collapsed by default. These areas can be expanded with a mouse click.

10.1.1 Customizing Comparison Results

The PDB comparison tool includes a set of options to customize the comparison report. These options help to restrict the overall number of differences identified in the report.



In addition to options that modify the report data, PDB allows you to control which columns are displayed in the comparison report. For more information on managing columns in the comparison report, see Section 10.1.1.1 on page 152.

To set the comparison options:

- 1. In the **Configuration Comparison** workspace, click **Comparison Options**.

The comparison options are displayed, see Figure 68.

Comparison Options

☐ Use Delta Only

Site Specific List

☐ Show All Site Specific Lists

☐ Fallback to IVL

☐ Fallback to Default Values

☐ Case Insensitive

☐ Ignore ReadOnly

MOC Filtering

Category

☒ Exclude ☐ Include

Name (available after first comparison)

☒ Exclude ☐ Include


Internal
Operator Configurable
Site Specific
Solution Integration


CTRL+Click to toggle selections.

MOI Mapping

MOI Mapping...

Differences to Include

☒  Show All Differences

☐  Show Differences Unique to Original Configuration


☐  Show Differences Unique to Target Configuration

Figure 68 Comparison Options

The following table describes the different comparison options:

Table 44 Comparison Options



Table 44 Comparison Options

Element	Description
<Original Configuration>	<p>Lists a number of options that allow you to filter the comparison differences that are unique to the Original Configuration.</p> <p>Options include:</p> <ul style="list-style-type: none"> • Use Delta Only - Only available when comparing a delta configuration, this option restricts the comparison to those elements that are defined in the delta configuration. When left unchecked, inherited data is included in the comparison report. • Site Specific List - Use a site-specific list to resolve any site-specific variables in the configuration. For more information on site-specific lists, refer to Section 13 on page 177. <p>This field uses auto-complete functionality. Typing part of a name displays matching site-specific lists. Use the down-arrow on your keyboard to display all of the available lists.</p> <p>Note: By default, PDB only shows site-specific lists that are associated with this configuration. Select Show All Site-Specific Lists to see all available lists.</p> <ul style="list-style-type: none"> • Fallback to IVL - When a configuration element is defined in one configuration but is absent from the other, this option attempts a match by falling back to the initial value for the absent element, if available. • Fallback to Default Values - When a configuration element is defined in one configuration but is absent from the other, this option attempts a match by falling back to the default value for the absent element, if available. <p>Note: If Fallback to IVL and Fallback to Default Values are both selected, PDB attempts to fallback to the initial value first and proceeds to fallback to the default value only when an initial value is not available for the configuration element.</p>



Table 44 Comparison Options

Element	Description
<Target Configuration>	<p>Lists a number of options that allow you to filter the comparison differences that are unique to the Target Configuration.</p> <p>These options are identical to those defined for the Original Configuration.</p>
Case Insensitive	PDB ignores the letter case of parameter values.
Ignore ReadOnly	PDB ignores differences in ReadOnly parameters.
MOC Filtering	<p>This multi-select list box allows you to filter the comparison results and include or exclude specific MOCs.</p> <p>Note: By selecting to include certain MOCs, only those MOCs will included. By selecting to exclude certain MOCs, only those MOCs will be excluded.</p>



Table 44 Comparison Options

Element	Description
Differences to Include	<p>This option allows you to configure how PDB performs the comparison.</p> <ul style="list-style-type: none"> • Show All Differences - Enabled by default, PDB examines all of the parameters and parameter groups within both configurations and identifies all differences in the comparison report. • Show Differences Unique to Original Configuration - PDB examines all of the parameters and parameter groups within the Original Configuration and compares them against the Target Configuration. Only those differences unique to the Original Configuration are identified in the comparison report. • Show Differences Unique to Target Configuration - PDB examines all of the parameters and parameter groups within the Target Configuration and compares them against the Original Configuration. Only those differences unique to the Target Configuration are identified in the comparison report.
Category	<p>This multi-select list box allows you to filter the comparison results and include or exclude specific categories of parameters and parameter groups. Available categories include:</p> <ul style="list-style-type: none"> • Internal • Operator Configurable • Site Specific • Solution Integration <p>Use Ctrl + click or Shift + click for multi-select.</p> <p>Note: By selecting to include certain categories, only those categories will included. By selecting to exclude certain categories, only those categories will be excluded.</p>

2. Select your options. See Table 44.



A comparison report can now be generated using the selected options. For more information on performing a comparison, refer to Section 10.2 on page 152.

Note: Customized options persist until you log off. If the default options are modified, a **[Modified]** flag is added to the heading at the top of the frame as a visual aid. Click **Reset Options** to restore the default settings.

10.1.1.1 Managing Columns in the Comparison Report

You can control what information is displayed in the comparison report table by showing or hiding individual columns.

Note: Managing columns has no impact on the data contained in the comparison report. Exporting the report in CSV format will include any hidden columns.

To show or hide columns:

1. In the **Configuration Comparison** workspace, click **Manage Columns**.

The **Manage Columns** area opens.

2. Select the individual columns that you would like to display.
3. If a comparison report has already been generated, click **Apply Now** to update the table view.

10.2 Performing a Comparison

To compare two configurations:

1. Open the **Configuration Comparison** workspace.

Note: Shortcuts in the **Configurations** context menu (under **Revision > Compare with Previous** and **Revision > Compare with Another**) allow you to generate comparison reports directly.

The **Configuration Selection** area is displayed. See Figure 69.

Figure 69 Configuration Selection

The following table describes the elements of the **Configuration Selection** area.



Table 45 Configuration Comparison Workspace Elements

Element	Description	Notes
Original Configuration		
Node (List)	Nodes defined in PDB.	Used to filter the available configurations in the Configuration field.
Node Revision (List)	Node revisions attached to the selected Node .	Used to filter the available configuration revisions in the Configuration field.
Existing Configuration	Select a node configuration stored in PDB for comparison.	<p>This field uses auto-complete functionality. Typing part of the configuration name displays matching node configurations. Use the down-arrow on your keyboard to display all of the available configurations.</p> <p>You must have the correct ACL permissions to view a configuration before it can be selected for comparison.</p> <p>The revision of the selected configuration is presented in square brackets at the end of the configuration name.</p>



Table 45 Configuration Comparison Workspace Elements

Element	Description	Notes
Upload New Configuration	<p>If selected, PDB will perform the comparison on a configuration file that does not exist in the PDB database. This file is used for comparison purposes only and does not persist in PDB (comparison on the fly).</p> <p>Uploading a configuration file requires the following information:</p> <ul style="list-style-type: none">• Upload Type - LDIF, NETCONF, PVL• Schema - List of available schemas• Input File	<p>The comparison tool supports the same file formats that are used to import node configurations (ZIP, TAR, LDIF, XML). For more information on importing configuration data, refer to Section 8.2 on page 74.</p> <p>When uploading a configuration file in PVL format, the Import From field allows you to select which PVL parameter value set to compare.</p> <p>Use Delta instructs PDB to process values from the PVL delta column in addition to the specified solution type.</p> <p>The associated configuration schema must be stored in PDB and visible with the correct ACL permissions. To facilitate the comparison, PDB will automatically pre-select the Schema based on the Original Configuration. If required, this selection can be changed.</p>
Target Configuration		
Existing Configuration	If selected, PDB will use an existing configuration as a target for the comparison.	The fields used to select a Target configuration are the same as those used for the Original.
Upload New Configuration	If selected, PDB will perform the comparison against a configuration file that does not exist in the PDB database.	The fields used to upload a Target Configuration are the same as those used for the Original.

2. Under **Original Configuration**, choose one of the following options:

- Select **Existing Configuration** to use a configuration from PDB.



- Select **Upload New Configuration** to upload a configuration for comparison purposes (comparison on the fly).

When uploading a configuration file, PDB automatically performs a validation check to ensure that the configuration is in line with the selected schema. If the check identifies errors or warnings, a Validation report is generated and can be downloaded from the **Configuration Comparison Report**.

3. Enter the required information. See Table 45.

4. Select a **Target Configuration**.

Note: The fields used to select a Target configuration are the same as those used for the Original.

5. If required, set comparison options to customize the comparison report. See Section 10.1.1 on page 147.

6. Click **Compare**.

The Configuration Comparison Report is displayed. See Figure 70.

Configuration Comparison Report										
<div>Export to CSV Create Delta... Extract Variables...</div>										
Number of differences: 11										
Type	Name	HSS_11A_R00_L00(A)			HSS_11A_R00_L00(B)			Category	MOC	MOR
		Default	IVL	Value	Default	IVL	Value			
P	tsfPuKey			0			[not present]	Internal	tsfPuKeyReference	FTU(applicationName=FTU)tsfPuKeyReference [tsfPuKeyReferenceKey=0]
PG	tsfPuKeyReference			[present]			[not present]	Internal	tsfPuKeyReference	FTU(applicationName=FTU)tsfPuKeyReference [tsfPuKeyReferenceKey=0]
P	tsfPuKey			[not present]			1	Internal	tsfPuKeyReference	FTU(applicationName=FTU)tsfPuKeyReference [tsfPuKeyReferenceKey=1]
PG	tsfPuKeyReference			[not present]			[present]	Internal	tsfPuKeyReference	FTU(applicationName=FTU)tsfPuKeyReference [tsfPuKeyReferenceKey=1]
P	HSS-InstallationType			Monolithic			Front-End	Internal	HSS-Application	HSS-Application(applicationName=HSS)
P	HSS-NodeIdentity			HSS-ty-ics-se			HSS_TEST-ty-ics-se	Site Specific	HSS-Application	HSS-Application(applicationName=HSS)
P	HSS-RoamingBehaviour	REJECT		REJECT	REJECT		ALLOW	Solution Integration	HSS-RoamingConfiguration	HSS-Application(applicationName=HSS)HSS-ConfigurationContainer [HSS-ConfigurationContainerName=HSS-ConfigurationContainer]HSS-RoamingConfigurationHSS-RoamingConfigurationHSS-RoamingConfiguration
P	HSS-SessionMngtDiameter			HSS-SM-ty-ics-se			HSS_TEST-SM-ty-ics-se	Site Specific	HSS-SsoConfiguration	HSS-Application(applicationName=HSS)HSS-ConfigurationContainer [HSS-ConfigurationContainerName=HSS-ConfigurationContainer]HSS-SsoConfigurationHSS-SsoConfigurationHSS-SsoConfiguration
P	HSS-MaxNumberOfApplicationsLicense	0		100	0		10	Site Specific	HSS-License	HSS-Application(applicationName=HSS)HSS-License[HSS-LicenseName=HSS-License]
P	HSS-MaxNumberOfPublicIPs	10000		2000000	10000		100000	Site Specific	HSS-License	HSS-Application(applicationName=HSS)HSS-License[HSS-LicenseName=HSS-License]
P	tsfPmAsPmdoCpuLoad	1.0		1.000000	1.0		0.250000	Internal	tsfPmApplication	tsfPmApplication(applicationName=tsfPM)

Figure 70 Configuration Comparison Report Table

The following table describes the different elements forming the **Configuration Comparison Report** table:

Table 46 Configuration Comparison Report Table Elements

Element	Description	Notes
Export to CSV	Exports the Configuration Comparison Report to a CSV file that must be downloaded from the PDB server.	



Table 46 Configuration Comparison Report Table Elements

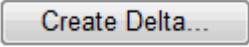
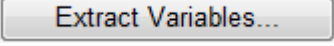
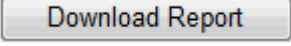
Element	Description	Notes
	Creates a delta configuration from the result of the comparison. For more information, refer to Section 10.3 on page 158.	
	Extracts variables from the Original Configuration and matches them with corresponding values from the Target Configuration. These variable-value pairs can be added to an existing site-specific list.	<p>The extraction of variable-value pairs is best effort.</p> <p>When adding variable-value pairs to a site-specific list, preexisting variable-value pairs are not modified.</p> <p>Variables that could not be mapped to a corresponding value will not be added to the site-specific list.</p>
	Click to download the Validation report for an uploaded configuration file.	
Type	The classification of the configuration element, Parameter Group (PG) or Parameter (P).	
Name	The name of the configuration element.	



Table 46 Configuration Comparison Report Table Elements

Element	Description	Notes
<Original Configuration>	<p>Lists the properties of the configuration element inside the Original Configuration.</p> <p>Properties include:</p> <ul style="list-style-type: none"> • Default - The default value of the parameter instance as defined in the schema of the original configuration, if available. • IVL - The initial value of the parameter instance as defined in the IVL associated with the schema of the original configuration, if available. • Value - The set value of the parameter instance. 	A value of not present means the configuration item is missing from the original configuration.
<Target Configuration>	<p>Lists the properties of the configuration element inside the Target Configuration.</p> <p>Properties include:</p> <ul style="list-style-type: none"> • Default - The default value of the parameter instance as defined in the schema of the target configuration, if available. • IVL - The initial value of the parameter instance as defined in the IVL associated with the schema of the target configuration, if available. • Value - The set value of the parameter instance. 	A value of not present means the configuration item is missing from the target configuration.
Category	The category of the configuration element as defined in the associated schema, if available.	



Table 46 Configuration Comparison Report Table Elements

Element	Description	Notes
MOC	The Managed Object Class (MOC) that the configuration element belongs to.	
MOI	The Managed Object Instance (MOI) shows the full path of the configuration element in the configuration tree.	

The configuration comparison is complete. You can use **Manage Columns** to update the table view.

10.3 Creating a Delta Configuration From Comparison Results

PDB allows you to generate a delta configuration from the results of a configuration comparison. This delta hangs from the Original Configuration and contains information that can transform the configuration data of the Original Configuration into the Target Configuration. In order to generate a delta configuration from comparison results, the Original configuration must be stored in PDB and the Original and Target configurations must use the same schema revision.

Delta configurations that are generated from a comparison result are populated with the configuration changes that make the Target Configuration different from the Original Configuration. The delta inherits all of the parameter groups and parameters from the Original Configuration, which becomes the parent of the new delta. All of the additions, deletions, and modifications necessary to transform the Original Configuration into the Target Configuration are included in the delta. For more information on delta configurations, refer to Section 8.4 on page 86.

To generate a delta configuration from comparison results:

1. Generate a Configuration Comparison Report by following the steps outlined in Section 10.2 on page 152.

Note: Both the Original and the Target configuration must use the same schema revision.

2. Click **Create Delta**.

The **Delta** dialog box opens.

3. Enter the required information.

If a delta configuration already hangs from the **Original Configuration**, you have the option to use the comparison results to create new revision of the existing delta. The current revision must be in locked state before a new



revision can be created. For more information on document states, refer to Section 8.5.1 on page 93.

4. Click **Create**.

The delta configuration is added to PDB and can be viewed in the **Configurations** interface.

11 Working with Multi-Solution Configurations

A multi-solution configuration is a collection of configuration data that contains up to four sets of parameter values. Each set of values is represented by a distinct solution type within the multi-solution configuration.

Solution types are a separate version of the configuration data that can be extracted as a stand-alone configuration. Having multiple values for a given parameter instance allows the multi-solution configuration to accommodate the different networking environments where the node can operate.

The following solution types are currently supported:

- IMS fixed
- IMS mobile
- Converged
- Node standalone

For more information on the PVL format, refer to the Parameter List Template Description, EAB/FTI-08:0686 Uen.

Each multi-solution configuration within PDB is composed of a collection of discrete MPVL configurations that share the same schema. These MPVLs are treated as separate entities until export time when they are bundled together. PDB uses the Initial Value List (IVL) associated with the common schema to set default values for the multi-solution configuration. These values are left blank if no IVL is present.

PDB allows you to work with multi-solution configurations in the following ways:

- Creating a New Multi-Solution Configuration
- Modifying a Multi-Solution Configuration
- Exporting a Multi-Solution Configuration



PDB keeps an Access Control List (ACL) for each multi-solution configuration. All non-administrative users require ACL permissions to view or modify the associated content. For more information on ACLs, refer to Section 3.1 on page 9.

All multi-solution configurations stored in PDB are revision controlled. PDB validates revision levels following Ericsson's standard rules for document handling. A given revision will precede or supersede other revisions of the same multi-solution configuration; this relationship opens a number of operations when working with a revised multi-solution configuration. These operations include:

- Changing the Document State of a Multi-Solution Configuration
- Creating a New Revision of a Multi-Solution Configuration

11.1 The Multi-Solution Configurations Workspace

The **Multi-Solution Configurations** workspace allows you to carry out tasks with multi-solution configurations.

To access the **Multi-Solution Configurations** workspace, select **Multi-Solution Configurations** from the menu options on the left. See Figure 71.

Multi-Solution Configurations

Document Number User Name Revision ☒ Latest Only

Document States

☒ PREL ☒ FROZ
☒ FREE ☒ WIDR

<< Search Clear >>

Refresh New Show Details

Name	Document Number	Revision	Document State
MTAS_11B_R6D_FMC	21/190 73-CXP 901 1263/5	PA1	PREL

Figure 71 The Multi-Solution Configurations Workspace

The workspace is divided into two principal areas as follows:

Search

The search options, located at the top of the page, allow you to filter the multi-solution configurations that are displayed in the **Multi-Solution Configurations** table. For more information on performing a search, refer to Section 11.1.1 on page 162.


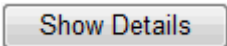


Multi-Solution Configurations Table

The **Multi-Solution Configurations** table is the centerpiece of the Multi-Solution Configurations workspace. This table displays the multi-solution configurations that are stored in PDB and allows you to perform a number of configuration-specific operations using a context menu.

The following table describes the elements of the **Multi-Solution Configurations** workspace.

Table 47 Multi-Solution Configurations Workspace Elements

Element	Description
	Opens the Create dialog box where you can define a new multi-solution configuration.
	Expands the Multi-Solution Configurations table to show the User and Last Modification columns. Once open, these columns can be hidden by clicking Hide Details .
Name	The name of the multi-solution configuration.
Document Number	The document number associated with the multi-solution configuration.
Revision	The revision of the multi-solution configuration.
Document State	<p>Displays the document state of the multi-solution configuration. The following document states are available:</p> <ul style="list-style-type: none"> • PREL - Preliminary • FROZ - Frozen • FREE - Approved • WIDR - Withdrawn <p>For more information on document states, refer to Section 11.3.1 on page 166.</p>



Element	Description
User	The user who created the multi-solution configuration in PDB. ⁽¹⁾
Last Modification	A timestamp marking when the multi-solution configuration was last modified. ⁽¹⁾

(1) This column is normally hidden. Click **Show Details** to display this information.

Each multi-solution configuration listed in the **Multi-Solution Configurations** table is selectable. Right-clicking a multi-solution configuration opens a context menu where operations specific to the selected bundle can be executed.

11.1.1 Searching for Multi-Solution Configurations

By default, the **Multi-Solution Configurations** table includes only the latest revisions of the available configuration bundles. A number of search criteria are available to help you find specific multi-solution configurations that may not be displayed. PDB reports partial matches on search strings. Use double quotes "<" ">" to restrict the search to exact matches.

Searches are performed using the search workspace at the top of the **Multi-Solution Configurations** page. See Figure 72.

Multi-Solution Configurations

Document Number User Name Revision ☒ Latest Only

Document States

☒ PREL ☒ FROZ

☒ FREE ☒ WIDR

<< Search Clear >>

Figure 72 Multi-Solution Configurations Search

The following table describes the available search criteria.

Table 48 Multi-Solution Configuration Search Elements

Element	Description	Notes
Document Number	Filters the table for multi-solution configurations with document numbers that match the selected criteria.	
User	Filters the table for multi-solution configurations that were created by the specified user.	



Table 48 Multi-Solution Configuration Search Elements

Element	Description	Notes
Name	Filters the table for multi-solution configurations with names that match the selected criteria. All entries are case-sensitive.	
Revision	Filters the table for multi-solution configurations with revisions that match the selected criteria.	Latest Only must be deselected to perform a search using this field.
Latest Only	Includes only the latest multi-solution configuration revisions that match the other search criteria.	If this option is selected, search results are restricted to the latest revision.
Document States	Filters the table for multi-solution configurations with the selected document states.	

All search filters are additive. Using multiple filters will narrow the search results.

To filter the **Multi-Solution Configurations** table:

1. In the **Multi-Solution Configurations** workspace, set your search criteria.
2. Click **Search**.

The **Multi-Solution Configurations** table is populated with configurations that match the selected criteria.

Note: PDB automatically stores up to 10 consecutive searches. Use the navigation buttons to move between each search.

Search results are retained as you navigate through the web portal. To reset the **Multi-Solution Configurations** table to the default display, click **Clear** then click **Search**.

Each multi-solution configuration has a URL. This link provides an external reference to the specific multi-solution configuration. Following a direct link connects you to the PDB server. After logging in, PDB automatically loads the **Multi-Solution Configurations** workspace and shows the linked multi-solution configuration.

URLs are automatically generated by PDB. To access a URL and other properties, right-click a multi-solution configuration and select **Properties**.

11.2 Creating a New Multi-Solution Configuration

Multi-solution configurations can be created in PDB.

Each multi-solution configuration is composed of up to four pointers to separate MPVL configurations, one per solution type. These MPVL configurations are independent of the multi-solution configuration and must be added to PDB separately before starting this procedure. For more information on adding configurations to PDB, refer to Section 8 on page 67.

To create a multi-solution configuration:

1. In the **Multi-Solution Configurations** workspace, click **New**.

The **Create** dialog box opens. See Figure 73.

Figure 73 Create Dialog Box

The following table describes the elements of the **Create** dialog box:

Table 49 Create Dialog Box Elements

Element	Description	Notes
Name	The name of the multi-solution configuration.	Mandatory.
Document Number	The document number associated with the multi-solution configuration.	Mandatory. Document numbers are validated using Ericsson's standard rules for registration notation.



Element	Description	Notes
Description	A short description of the multi-solution configuration.	Optional.
Revision	The initial revision of the multi-solution configuration.	Mandatory.
Revision Comment	Comments on this particular revision of the multi-solution configuration.	Optional.
Schema (List)	<p>The schema associated with the MPVL configurations that will be added to the multi-solution configuration.</p> <p>The list shows both the schema name and revision (in square brackets).</p>	<p>Mandatory.</p> <p>By default, the list of schemas is filtered to display only the latest revisions. Clear Latest Schema Revisions Only to display the complete list.</p>
<ul style="list-style-type: none"> • Fixed (List) • Mobile (List) • Standalone (List) • Converged (List) 	<p>Specifies an MPVL configuration to use as input for each solution type.</p> <p>Each list shows both the configuration name and revision (in square brackets).</p>	<p>Optional.</p> <p>By default, the list of configurations for each solution type is filtered to display only the latest revisions. Clear Latest Configuration Revisions Only to display the complete list.</p>

2. Enter the required information. See Table 49.

3. Click **Create**.

The multi-solution configuration is created.

4. If required, grant permission for other users to work with the new configuration by modifying the ACL. For more information on granting ACL permissions, refer to Section 3.1.1 on page 10.

11.3 Working with Multi-Solution Configuration Revisions

Multi-solution configurations stored in PDB are revision controlled. A given revision precedes or supersedes other revisions of the same configuration bundle, allowing for a number of revision-specific operations.



These operations include:

- Changing the Document State of a Multi-Solution Configuration
- Creating a New Revision of a Multi-Solution Configuration

By default, only the latest revision of each multi-solution configuration is displayed in the **Multi-Solution Configurations** table. Additional revisions can be displayed by refining the search criteria. For more information on performing a search, refer to Section 11.1.1 on page 162.

PDB requires all predecessors to be in a locked state (FROZ, FREE, WIDR) before subsequent revisions can be created.

11.3.1 Document States

PDB uses a system of document states to provide information on the completeness, quality, and approval status of a particular multi-solution configuration revision. The document state is indicated by a status code that is part of the multi-solution configuration metadata. PDB uses the following document states for multi-solution configuration:

PREL	Preliminary. Used to designate an unlocked multi-solution configuration that is open for modification. Multi-solution configurations in the PREL state cannot serve as a basis for new revisions.
FROZ	Frozen. Used to designate a frozen multi-solution configuration where the content has been locked to prevent further changes. This is the basic state for multi-solution configuration revisions that are stored in PDB but are not approved for release.
FREE	Approved. Used to designate multi-solution configuration that are approved for release. Approved multi-solution configuration are locked to prevent any changes to the released content.
WIDR	Withdrawn. Used to designate multi-solution configuration that are no longer intended for use. Withdrawn multi-solution configuration are still accessible in PDB, but are locked to prevent further changes.

11.3.1.1 Changing the Document State of a Multi-Solution Configuration

PDB allows you to change the document state of multi-solution configurations that you have permission to modify. For more information on ACL permissions, refer to Section 3.1 on page 9.



Multi-solution configuration revisions must follow the following sequence of document states:

PREL > FROZ > FREE > WIDR

Note: Only PDB System Administrators can set a multi-solution configuration to a previous state.

New multi-solution configuration, or multi-solution configuration revisions start in the PREL state where they can be modified and updated. The document state is changed as the configuration progresses through its lifecycle.

To change the document state of a multi-solution configuration revision:

1. In the **Multi-Solution Configurations** workspace, right-click a configuration bundle and select **Revision > Set to <STATE>**.

Note: If the required multi-solution configuration is not visible, perform a search as outlined in Section 11.1.1 on page 162.

When moving from PREL to FROZ, the **Freeze** dialog box opens. See Figure 74.

Note: Before freezing a multi-solution configuration bundle, all constituent configurations (MPVL, IVL) must also be frozen. For more information on freezing node configurations, refer to Section 8.5.1.1 on page 93.

The screenshot shows a 'Freeze' dialog box with the following fields and controls:

- Configuration Name:** MTAS_11B_R6D_FMC
- Revision:** PA1
- Update Revision Comment:** A text area with a vertical scrollbar.
- Footer:** A note stating '*Once a configuration has been frozen it cannot be changed.' and two buttons: 'Freeze' and 'Cancel'.

Figure 74 Freeze Dialog Box

The Freeze dialog box allows you to modify the revision level of the multi-solution configuration. Here preliminary revisions can be set to a solid state before freezing.

Stop!

Only PDB System Administrators can unfreeze a multi-solution configuration. Do not continue if additional changes are required.



2. Click **Freeze**.

The selected multi-solution configuration is locked and the document state is set to FROZ.

For all other state transitions a confirmation dialog box is displayed.

Click **OK** to change the document state.

11.3.2 Creating A New Revision of a Multi-Solution Configuration

You can update a locked multi-solution configuration by creating a new revision. This process creates a clone of the locked configuration bundle at a new revision level. The new revision retains the name of the original multi-solution configuration and resets the document state to PREL, allowing you to make modifications.

To create a new revision of a locked multi-solution configuration:

1. In the **Multi-Solution Configurations** workspace, right-click a locked configuration bundle and select **Revision > Create New Revision**.

Note: If the required multi-solution configuration is not visible, perform a search as outlined in Section 11.1.1 on page 162.

The **New Revision** dialog box opens with the next revision displayed. See Figure 75.

The 'New Revision' dialog box is shown with the following fields and options:

- Name:** MTAS_11B_R6D_FMC
- Document Number:** 21/190 73-CXP 901 1263
- Description:** MTAS 11B Bundle
- Revision:** PA2
- Revision Comment:** (empty text area)
- Schema:** MTAS_11B_CM_MIM[PA4]
- ☐ Latest Schema Revisions Only
- ☐ Latest Configuration Revisions Only
- Fixed:** /MPVL_MTAS_11B_Fixed[PA3]
- Mobile:** /MPVL_MTAS_11B_Mobile[PA2]
- Standalone:** [Select]
- Converged:** /MPVL_MTAS_11B_Converged[PA2]
- *Only MPVL configurations are shown
- Create** and **Cancel** buttons at the bottom.

Figure 75 New Revision Dialog Box

Note: Fields identifying the multi-solution configuration, including Document Number and Description, are locked and inherited from the previous revision.



2. Verify that the proposed revision is correct and update the list of constituent configurations as required.

3. Click **Create**.

A new revision is created.

11.4 Working with Multi-Solution Configurations

PDB allows you to work with multi-solution configurations in the following ways:

- Modifying a Multi-Solution Configuration
- Deleting a Multi-Solution Configuration

11.4.1 Modifying a Multi-Solution Configuration

PDB allows you to modify most of the properties associated with a multi-solution configuration in PREL state. If the multi-solution configuration is in a locked state, only the **Description** and **Revision Comment** fields can be modified.

Many properties, including the constituent MPVL configurations and various metadata, are not visible in the **Multi-Solution Configurations** table. To view a read-only printout of all properties associated with a specific configuration bundle, open the context menu for that multi-solution configuration and select **Properties**.

To edit a multi-solution configuration:

1. In the **Multi-Solution Configurations** workspace, right-click an unlocked configuration bundle and select **Edit**.

Note: If the required multi-solution configuration is not visible, perform a search as outlined in Section 11.1.1 on page 162.

The **Multi-Solution Configurations** dialog box opens in edit mode.

2. Update the properties as required.

Note: Document numbers are validated using Ericsson's standard rules for registration notation.

Subsequent revisions of an existing multi-solution configuration are locked to the description and document number of their predecessor.

3. Click **Apply**.

The updated properties are saved.



11.4.2 Deleting a Multi-Solution Configuration

Authorized users can remove multi-solution configurations from PDB as needed. PDB does not allow you to remove multi-solution configurations in a locked state or parents of subsequent revisions.

Caution!

Deleting a multi-solution configuration permanently removes it from PDB.

To delete a multi-solution configuration:

1. In the **Multi-Solution Configurations** workspace, right-click a multi-solution configuration and select **Delete**.

Note: If the configuration is not displayed, perform a search as outlined in Section 11.1.1 on page 162.

A confirmation dialog box opens.

2. Click **OK**.

The multi-solution configuration is deleted.

11.5 Exporting a Multi-Solution Configuration

PDB allows you to export multi-solution configurations through the GUI. These configurations are exported in PVL format and packaged in ZIP or TAR files.

Note: Exporting a multi-solution configuration in PVL format does not require any site-specific information. This format is used to preserve configuration data along with any parameter value variables.

For the purpose of exporting multi-solution configurations, PDB treats each package as a collection of up to four MPVL configurations that share the same schema.

To export a multi-solution configuration:

1. In the **Multi-Solution Configurations** workspace, right-click a configuration bundle and select **Export**.

Note: If the required multi-solution configuration is not visible, perform a search as outlined in Section 11.1.1 on page 162.

The **Export Multi-Solution Configuration** dialog box opens. See Figure 76.



The dialog box titled "Export Multi-Solution Configuration" displays the following information:

Name	MTAS_11B_R6D_FMC
Document Number	21/190 73-CXP 901 1263/5
Revision	PA1

Fixed	/MPVL_MTAS_11B_Fixed[PA3]
Mobile	/MPVL_MTAS_11B_Mobile[PA2]
Standalone	[Not selected]
Converged	/MPVL_MTAS_11B_Converged[PA2]

☒ ZIP ☐ TAR

Export PVL Cancel

Figure 76 Export Multi-Solution Configuration Dialog Box

The **Export Multi-Solution Configuration** dialog box displays the selected properties of the configuration bundle that will be exported:

2. Select whether to export the multi-solution configuration in ZIP or TAR format.
3. Click **Export PVL**.

When exporting a multi-solution configuration, PDB follows a set of business rules that govern the format and structure of the configuration data. For more information on the PDB export criteria, refer to Section 15 on page 210.

The selected configurations are exported as a multi-solution PVL. When PDB has finished processing the data, a link to the exported configuration is displayed.

4. Click **Download Configuration** to save the configuration archive.

12 Working with Configuration Sets

A configuration set is a collection of node configurations. In PDB, working with configuration sets involves the following tasks:

- Adding a New Configuration Set



- Modifying a Configuration Set
- Linking Node Configurations to a Configuration Set

12.1 Adding a New Configuration Set

New configuration sets can be defined in PDB.

To add a new configuration set:

1. In the PDB GUI, select **Configuration Sets** from the menu options on the left.

The **Configuration Sets** table is displayed. See Figure 77.

Configuration Sets

Refresh

New

Edit

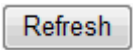



Delete

Name ▲	Description
ConfigSet 00	This is an empty configuration set for testing purposes
IMS Fixed	This configuration set is used to test web services
MMTEL 2.0	Configuration set for MMTEL 2.0

Figure 77 Configuration Sets Table

The following table describes the different elements forming the **Configuration Sets** interface:

Table 50 Configuration Sets Interface Elements

Element	Description
	Updates the table with the latest information from the PDB database.
	Adds a new configuration set to the table in edit mode.
	Opens the selected configuration set in edit mode where it can be modified.
	Removes the selected configuration set.



Element	Description
Name	The name of the configuration set. Mandatory.
Description	A short description of the configuration set. Optional.

2. Click **New**.

An empty configuration set is added to the top of the table in edit mode. See Figure 78.

Configuration Sets

Apply

Cancel

Name ▲	Description
<input type="text"/>	<input type="text"/>
ConfigSet 00	This is an empty configuration set for testing purposes
IMS Fixed	This configuration set is used to test web services
MMTEL 2.0	Configuration set for MMTEL 2.0

Figure 78 Adding a New Configuration Set

3. Enter the required information. See Table 50.

4. Click **Apply**.

12.2

Modifying a Configuration Set

The properties of existing configuration sets can be modified.

To modify a configuration set:

1. In the PDB GUI, select **Configuration Sets** from the menu options on the left.

The **Configuration Sets** table is displayed. See Figure 80.

Configuration Sets	
<input type="button" value="Refresh"/> <input type="button" value="New"/> <input type="button" value="Edit"/> <input type="button" value="Delete"/>	
Name ▲	Description
ConfigSet 00	This is an empty configuration set for testing purposes
IMS Fixed	This configuration set is used to test web services
MMTEL 2.0	Configuration set for MMTEL 2.0

Figure 79 Configuration Sets Table

The following table describes the different elements forming the **Configuration Sets** interface:

Table 51 Configuration Sets Interface Elements

Element	Description
<input type="button" value="Refresh"/>	Updates the table with the latest information from the PDB database.
<input type="button" value="New"/>	Adds a new configuration set to the table in edit mode.
<input type="button" value="Edit"/>	Opens the selected configuration set in edit mode where it can be modified.
<input type="button" value="Delete"/>	Removes the selected configuration set.
Name	A descriptive name for the configuration set.
Description	A short description of the configuration set.

2. Select a configuration set to modify from the table.

The **Edit** button becomes available.

3. Click **Edit**.

The selected configuration set is opened in edit mode.

4. Update the configuration set information as required. See Table 51.
5. Click **Apply**.



The updated configuration set is saved to the database.

Note: To remove a configuration set from PDB:

- Select a configuration set to remove from the **Configuration Sets** table.

The **Delete** button becomes available.

- Click **Delete**.

A confirmation dialog box opens.

- Click **OK**.

When a configuration set is removed from the database, all of the links to the associated node configurations are automatically removed as well.

12.3 Linking Node Configurations to a Configuration Set

Note: Only one configuration of the same node configuration format is allowed per configuration set.

To link node configurations to a configuration set:

1. In the PDB GUI, select **Configuration Sets** from the menu options on the left.

The **Configuration Sets** table is displayed. See Figure 80.

Configuration Sets	
<input type="button" value="Refresh"/> <input type="button" value="New"/> <input type="button" value="Edit"/> <input type="button" value="Delete"/>	
Name ▲	Description
ConfigSet 00	This is an empty configuration set for testing purposes
IMS Fixed	This configuration set is used to test web services
MMTEL 2.0	Configuration set for MMTEL 2.0

Figure 80 Configuration Sets Table

1. Select a configuration set from the table.

The **Configurations** table is displayed. See Figure 81.

Note: The **Configurations** table shows the node configurations that are currently linked to the selected configuration set.



Configurations			
<div>Refresh Add/Remove</div>			
	Name ▲	Node	Description
<input checked="" type="checkbox"/>	/HSS 4.1 Full	HSS 4.1	pvl_HSS_4.1_ownformat.zip
<input checked="" type="checkbox"/>	/MGW 1.3.5 Full	MGW 1.3.5	pvl_MGW_1.3.5_ownformat.zip

Figure 81 Configurations Table

The following table describes the different elements forming the **Configurations** interface:

Table 52 Configurations Interface Elements

Element	Description
<div>Refresh</div>	Updates the table with the latest information from the PDB database.
<div>Add/Remove</div>	Opens the list of available node revisions in edit mode where revisions can be added or removed.
Name	The name of the configuration.
Node	The node associated with the configuration.
Parent	The parent configuration. If any.
Description	A short description for this configuration.

2. Click **Add/Remove**.

A list of available node configurations is displayed in edit mode.



Configurations

	Name ▲	Node	Description
<input type="checkbox"/>	/Clone HSS 5.0	HSS 5.0	pvl_HSS_5.0_ownformat.zip
<input type="checkbox"/>	/Empty HSS 5.0	HSS 5.0	For testing
<input checked="" type="checkbox"/>	/HSS 4.1 Full	HSS 4.1	pvl_HSS_4.1_ownformat.zip
<input type="checkbox"/>	/HSS 4.1 Full/Delta HSS 4.1	HSS 4.1	1st Level Delta
<input type="checkbox"/>	/HSS 5.0 Full	HSS 5.0	pvl_HSS_5.0_ownformat.zip
<input checked="" type="checkbox"/>	/MGW 1.3.5 Full	MGW 1.3.5	pvl_MGW_1.3.5_ownformat.zip
<input type="checkbox"/>	/MGW 1.3.5 Full/Delta MGW 1.3.5	MGW 1.3.5	1st Level Delta

Figure 82 Linking Node Configurations to a Configuration Set

3. Select the check boxes next to the node configurations that are part of the selected configuration set.
4. Click **Apply**.

The selected node configurations are now linked to the configuration set.

13 Working with Site-Specific Lists

A site-specific list is a collection of site-specific variables and associated values that can be applied to a node configuration when exporting it in EAS, LDIF or NETCONF format.

Note: Exporting a configuration in PVL format does not require any site-specific information. This format is used to preserve configuration data along with any parameter value variables.

PDB can automatically add site-specific variables from a node configuration to a site-specific list. For more information on site-specific variables, refer to Section 9.3.4 on page 144. Once populated with variables, values specific to a target network environment can be added to the site-specific list. Each list then becomes an adaptation of a configuration's site-specific information, allowing PDB to resolve site-specific data at export time.

Working with site-specific lists can involve the following activities:



- Adding a new site-specific list
- Modifying a site-specific list
- Adding site-specific variables from a node configuration
- Reporting site-specific parameter usage
- Importing and exporting site-specific lists

PDB keeps an Access Control List (ACL) for each site-specific list. All non-administrative users require ACL permissions to view or modify the associated site-specific list. For more information on ACLs, refer to Section 3.1 on page 9.

13.1 The Site-Specific Lists Workspace

The **Site-Specific Lists** workspace allows you to carry out tasks with site-specific lists.

To access the **Site-Specific Lists** workspace, select **Site Specific Lists** from the menu options on the left. See Figure 83.

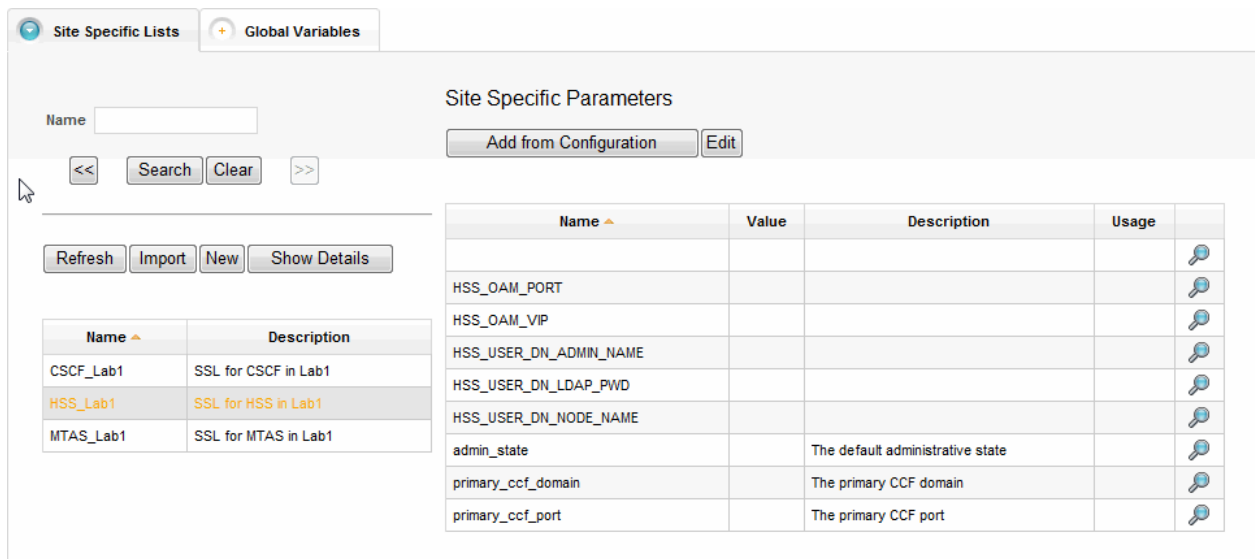


Figure 83 The Site-Specific Lists Workspace

The workspace is divided into two principal areas as follows:

Site Specific Lists

Contains the **Site Specific Lists** table. This table displays site-specific lists and allows you to perform a number of list-specific operations using a context menu.



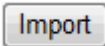

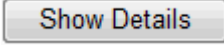
Selecting a site-specific list displays the **Site Specific Parameters** table. This table allows you to populate the list parameter value variables and their associated values.

Global Variables

Contains the **Global Variables** table. This table displays a managed list of parameter value variables that are recommended for use in node configurations.

The following tables describe the elements of the **Site-Specific Lists** workspace.

Table 53 Site-Specific Lists Workspace Elements

Element	Description
	Generates a new site-specific list using data imported from a variables file.
	Adds a new site-specific list to the table in edit mode.
	Expands the Site Specific Lists table to show the User and Last Modification columns.
Name	A descriptive name for the site-specific list. Mandatory.
Description	A short description of the site-specific list. Optional.
User	The user who created the site-specific list. ⁽¹⁾
Last Modification	A timestamp marking when a site-specific list was last modified. ⁽¹⁾

(1) This column is normally hidden. Click **Show Details** to display this information.

Each site-specific list in the **Site Specific Lists** table is selectable. Right-clicking a list opens a context menu where a number of list-specific operations can be executed. These operations include:

Edit	Opens the site-specific list for editing. Allows you to modify the name and description. For more information on modifying a site-specific list, refer to Section 13.2.4 on page 186.
Delete	Deletes the site-specific list.
Export	Exports the site-specific list to a variables file. For more information on exporting a site-specific list, refer to Section 13.2.6 on page 188.
Duplicate	Copies the contents of the site-specific list to a new list. For more information on duplicating a site-specific list, refer to Section 13.2.3 on page 185.

**Configurations**

Allows you to associate the site-specific list with node configurations. For more information on associations, refer to Section 13.2.5 on page 186.

Add to Data Transfer

Adds the site-specific list to the data transfer export list. For more information on data transfer, refer to Section 14 on page 201.

Permissions

Allows you to view and set ACL permissions for the site-specific list. For more information on ACLs, refer to Section 3.1 on page 9.

Properties

Allows you to view the properties of the site-specific list.

User accounts require permissions to create, duplicate, modify and delete site-specific lists.

Table 54 Site Specific Parameters Table

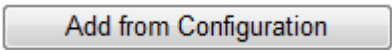



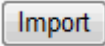


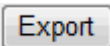
Element	Description
	Adds all variables defined in a selected node configuration to the site-specific list. This operation does not modify the selected configuration, it only retrieves the variable names.
	Opens the Site Specific Parameters table in edit mode where you can modify the list of variables.
Name	The name of the parameter value variable. Mandatory.
Value	The value for the associated parameter value variable. Optional.
Description	If the parameter value variable is present on the Global Variables list, this column will display the associated description, if available.
Usage	After reporting the usage of a site-specific variable within a configuration, this column presents a comma-separated list of parameter names that include the variable.
	Reports the usage of the site-specific variable in available node configurations. For more information, refer to Section 13.4.2 on page 195.



Table 55 Global Variables Table

Element	Description
	Adds a new parameter value variable to the table in edit mode.
	Updates the Global Variables list with data imported from a variables file. You have the option of appending the imported content or overwriting the existing list.
	Opens the selected parameter value variable in edit mode.
	Deletes the selected parameter value variable.
	Exports the Global Variables list to a variables file.
Name	The name of the parameter value variable. Mandatory.
Description	A short description of the parameter value variable. Optional.

13.1.1 Searching for Site-Specific Lists

The **Site Specific Lists** table is searchable by name. Searches are performed using the search workspace at the top of the **Site-Specific Lists** page.

Searching by name filters the table for lists that match the selected criteria. All entries are case sensitive.

PDB reports partial matches on search strings. Use double quotes "<" ">" to restrict the search to exact matches.

To search the **Site Specific Lists** table:

1. In the **Site Specific Lists** workspace, set your search criteria.
2. Click **Search**.

The **Site Specific Lists** table is populated with lists that match the selected criteria.

Note: PDB automatically stores up to 10 consecutive searches. Use the navigation buttons to move between each search.

Search results are retained as you navigate through the web portal. To reset the **Site Specific Lists** table to the default display, click **Clear** then click **Search**.



PDB allows you to link to site-specific lists with URLs. Using a direct link connects you to the PDB server. After logging in, PDB automatically loads the **Site-Specific Lists** workspace with the linked item displayed.

A URL is automatically generated by PDB. To access the URL and other properties, right-click a site-specific list and select **Properties**.

13.2 Site-Specific Lists

In PDB, site-specific lists act as containers for parameter value variables.

Working with site-specific lists involves the following tasks:

- Importing a site-specific list
- Creating a new site-specific list
- Duplicating a site-specific list
- Modifying a site-specific list
- Associating a Site-Specific List with Node Configurations
- Exporting a site-specific list

13.2.1 Importing a Site-Specific List

A site-specific list can be imported to PDB from a CSV file. This file represents each parameter value variable in the following format:

```
<name>,<value>,<description>,<usage>
```

To ensure the proper format, the `create-variables-file` CLI command can be used to generate a template file containing all of the parameter variables required by a node configuration in PDB.

For more information on this and other CLI commands, refer to the PDB Command Line Interface (CLI) Reference, [1/1540-CXP 902 0212](#).

Example 9 shows a basic CSV file that only contains the mandatory parameter value variables for TSP nodes that use the LDIF configuration format:



```

...
...
...
...
...
...
...
...

```

Example 9 Sample Variables CSV File for LDIF Configurations

Example 10 shows a basic CSV file that only contains the mandatory parameter value variables for any nodes that use the NETCONF configuration format:

```

...
...
...
...
...

```

Example 10 Sample Variables CSV File for NETCONF Configurations

To import a site-specific list from a variables file:

1. In the **Site-Specific Lists** workspace, click **Import**.

The **Import** dialog box opens. See Figure 84.

Figure 84 Import Dialog Box

The following table describes the elements of the **Import** dialog box:

Table 56 Import Dialog Box Elements

Element	Description
Name	The name of the site-specific list. Mandatory.



Element	Description
Description	A short description of the site-specific list. Optional.
Input File	The path and file name of the variables file. Mandatory.

2. Enter the required information. See Table 56.

3. Click **Import**.

PDB imports the data contained within the selected variables file and generates a new site-specific list.

4. If required, grant permission for other users to work with the new site-specific list by modifying the ACL. For more information on granting ACL permissions, refer to Section 3.1.1 on page 10.

13.2.2 Creating a New Site-Specific List

New site-specific lists can be defined in PDB.

To add a new site-specific list:

1. In the **Site-Specific Lists** workspace, click **New**.

An empty site-specific list is added to the top of the table in edit mode. See Figure 85.

The screenshot shows a window titled "Site Specific Lists". At the top are "Apply" and "Cancel" buttons. Below them is a table with two columns: "Name" and "Description". The first row of the table is highlighted with an orange border, indicating it is the active row for editing. This row contains empty text boxes for both the Name and Description fields. Below this row are three existing entries: "CSCF_Lab1" with description "SSL for CSCF in Lab1", "HSS_Lab1" with description "SSL for HSS in Lab1", and "MTAS_Lab1" with description "SSL for MTAS in Lab1".

Name	Description
CSCF_Lab1	SSL for CSCF in Lab1
HSS_Lab1	SSL for HSS in Lab1
MTAS_Lab1	SSL for MTAS in Lab1

Figure 85 Adding a New Site-Specific List

2. Enter the required information. See Table 53.

3. Click **Apply**.



The new site-specific list is added to the database. New lists are positioned in the table based on the sorting criteria.

Note: You must add parameters to the site specific list before it can be used. New site-specific lists must be populated with parameters. For more information, refer to Section 13.4 on page 193.

4. If required, grant permission for other users to work with the new site-specific list by modifying the ACL. For more information on granting ACL permissions, refer to Section 3.1.1 on page 10.

13.2.3 Duplicating a Site-Specific List

Site-specific lists that are stored in PDB can be duplicated. Duplication creates a copy of the selected site-specific list with a new name and description. The copied list contains all of the original site-specific parameters and the assigned values; however, it is treated as a new site-specific list and does not inherit any configurations or permissions that were assigned to the original list.

To duplicate a site-specific list:

1. In the **Site-Specific Lists** workspace, right-click a list and select **Duplicate**.

The **Duplicate Site Specific List** dialog box opens. See Figure 85.

Duplicate Site Specific List

Creating a duplicate for: C SCF_Lab1
Please make sure you provide a different name for your duplicate.

Name

Description

Figure 86 Duplicate Site Specific List Dialog Box

2. Enter a new name for the duplicated list and an optional description.

Note: Duplicated lists must have a unique name in PDB. Empty or duplicated names will generate a warning message.

3. Click **Duplicate**.



The duplicated site-specific list is added to the database. Duplicated lists are automatically searched for by name and will be displayed in the table on their own.

4. If required, grant permission for other users to work with the new site-specific list by modifying the ACL. For more information on granting ACL permissions, refer to Section 3.1.1 on page 10.

13.2.4 Modifying a Site-Specific List

The properties of existing site-specific lists can be modified.

To modify a site-specific list:

1. In the **Site-Specific Lists** workspace, right-click a list and select **Edit**.

The selected site-specific list is opened in edit mode.

2. Update the properties of the site-specific list as required. See Table 53.
3. Click **Apply**.

The updated site-specific list is saved to the database.

Note: To remove a site-specific list from PDB:

- In the **Site Specific Lists** table, right-click a site-specific list to remove and select **Delete**.

A confirmation dialog box opens.

- Click **OK**.

When a site-specific list is removed from the database, the list contents are also removed.

13.2.5 Associating a Site-Specific List with Node Configurations

A site-specific list can be associated with the node configurations to which it applies. When PDB operations make use of site-specific lists, these associations identify which lists are compatible with the selected configuration.

To associate site-specific list with node configurations:

1. In the **Site-Specific Lists** workspace, right-click a list and select **Configurations**.

The Configurations dialog box opens. See Figure 87



Configurations

Configuration ▲	User	Timestamp	
/MPVL_MTAS_11B_Fixed[PA4]	PDB_User	2011-08-02 19:39:09.0	

Add Configuration:

Figure 87 Configurations Dialog Box

The following table describes the elements of the **Configuration** dialog box:

Table 57 Import Dialog Box Elements

Element	Description
Configuration	The name of the associated configuration. Note: The revision of the configuration is presented in square brackets at the end of the configuration name.
User	The PDB user who associated the configuration.
Timestamp	Marks when the associated configuration was added.
	Click to remove the associated configuration from the list.



Element	Description
Add Configuration	Associates configurations with the site-specific list. Note: This field uses auto-complete functionality. Typing part of the name or revision level displays a list of configurations that match the criteria. Use the down-arrow on your keyboard to display the complete list.

1. Associate configurations by selecting them with **Add Configuration**.
2. Click **Apply**.

The selected associations are saved.

13.2.6 Exporting a Site-Specific List

Site-specific lists that are stored in PDB can be exported to CSV or Testing and Test Control Notation (TTCN) formatted files.

CSV files are compatible with Microsoft Excel and the PDB CLI.

TTCN files are used in Titansim test bundles.

CSV Format

When exporting a site-specific list in CSV format, PDB creates an output file with a ".csv" file extension to store the exported data. These files have the following structure, one variable per line:

<name>,<value>,<description>,<usage>

Example 11 shows a CSV output file with sample data.

```
NAME,VALUE,DESCRIPTION,USAGE
NAME,VALUE,DESCRIPTION,USAGE
NAME,VALUE,DESCRIPTION,USAGE
NAME,VALUE,DESCRIPTION,USAGE
NAME,VALUE,DESCRIPTION,USAGE
NAME,VALUE,DESCRIPTION,USAGE
NAME,VALUE,DESCRIPTION,USAGE
NAME,VALUE,DESCRIPTION,USAGE
```

Example 11 Sample CSV Output File

This file carries the same format that is required by CLI commands, such as:

- compare-configuration-file
- convert-configuration



— export-configuration

— validate-mpvl

For more information on these and other CLI commands, refer to the PDB Command Line Interface (CLI) Reference, 1/1540-CXP 902 0212.

Note: PDB exports all data displayed in the **Site Specific Parameters** table. In order to capture usage information, you must first generate a report on the variable usage in a specific configuration. For more information on reporting site-specific variable usage, refer to Section 13.4.2 on page 195.

TTCN Format

When exporting a site-specific list in TTCN format, PDB creates an output file with a ".cfg" file extension to store the exported data. These files have the following structure, one variable per line:

```
<name> := "<value>"
```

Example 12 shows a TTCN variables file with sample data.

```

//
//
//
//
//
//
//
//

```

Example 12 Sample TTCN Variables File

To export a site-specific list:

1. In the **Site-Specific Lists** workspace, right-click a list and select **Export**.

The **Export Site Specific List** dialog box opens. See Figure 88.

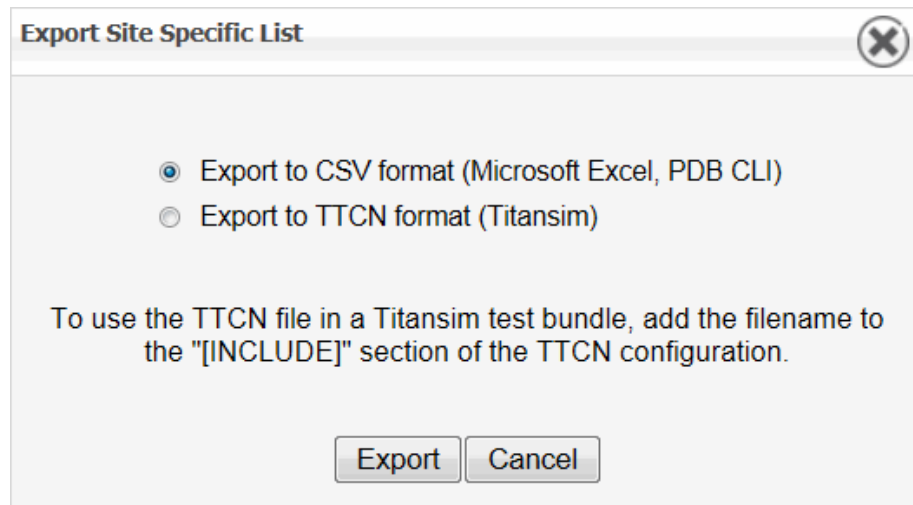


Figure 88 Export Site Specific List Dialog Box

2. Select the export format.
3. Click **Export**.

The parameter value variables are exported. When PDB has finished processing the data, a link to the output file is displayed.

13.3 Global Variables

PDB includes a global list of parameter value variables. This list allows authorized users to define a set of variables that are recommended for use in PDB. Once a variable has been added to the list, editors can add it to a node configuration using auto-complete functionality. For more information on adding parameter value variables to a node configuration, refer to Section 9.3.4 on page 144.

Working with global variables involves the following tasks:

- Creating New Global Variables
- Modifying Global Variables
- Importing Global Variables
- Exporting Global Variables

13.3.1 Creating New Global Variables

Authorized users can define global variables directly in PDB. All global variables are limited to the following characters:

- alpha-numeric (a,A, to z,Z and 0-9)
- underscore (_)



— dash (-)

To add a new global variable:

1. On the **Global Variables** tab in **Site-Specific Lists** workspace, click **New**.

An empty global variable is added to the top of the table in edit mode. See Figure 89.

Name ▲	Description
<input type="text"/>	<input type="text"/>
admin_state	The default administrative state
primary_ccf_domain	The primary CCF domain
primary_ccf_port	The primary CCF port

Figure 89 Adding a New Global Variable

2. Enter the required information. See Table 55.
3. Click **Apply**.

The new global variable is added to the database. New variables are positioned in the table based on the sorting criteria.

13.3.2 Modifying Global Variables

The properties of existing global variables can be modified.

To modify a global variable:

1. On the **Global Variables** tab in **Site-Specific Lists** workspace, select a global variable to modify and click **Edit**.

The selected variable is opened in edit mode.

2. Update the properties of the global variable as required. See Table 55.
3. Click **Apply**.

The updated global variable is saved to the database.

Note: To remove a global variable from PDB:

- On the **Global Variables** tab in **Site-Specific Lists** workspace, select a global variable to remove and click **Delete**.

A confirmation dialog box opens.

- Click **OK**.

Removing a global variable from PDB does not remove any instances of that variable from node configurations or site-specific lists.

13.3.3 Importing Global Variables

Global Variables can be imported from a properties file. This plain-text file uses a `<variable_name>=<value>` format to store information. Example 13 shows a sample variable properties file.

```

2015-07-16 16:30:04.000
primary_cst_kusto/the primary cst kusto
primary_cst_part/the primary cst part
state/state the default administration of

```

Example 13 Sample Global Variables File

To import global variables from a file:

1. On the **Global Variables** tab in **Site-Specific Lists** workspace, click **Import**.

The **Import** dialog box opens. See Figure 90.

Import

☐ Overwrite Existing Variables

Input File

Browse...

Uploaded File

Import

Cancel

Figure 90 Import Dialog Box

2. Select an input file.
3. If required, select **Overwrite Existing Variables** to overwrite matching variables already on the Global Variables list.

Note: If this option is not selected, any matching variables will cause the import operation to fail. A list of conflicting variables is available in the File Validation Report that can be downloaded from PDB after an import failure.

4. Click **Import**.



PDB imports the data contained within the selected properties file and updates the Global Variables list.

13.3.4 Exporting Global Variables

The PDB Global Variables list can be exported to a plain-text properties file for archival purposes or transfer to another PDB.

To export the Global Variables list to a properties file:

- On the **Global Variables** tab in **Site-Specific Lists** workspace, click **Export**.

The Global Variables list is exported to a properties file. When PDB has finished processing the data, a link to the exported file is displayed.

13.4 Site-Specific Variables

A site-specific list can be populated with site-specific variables and the associated values.

Populating a site-specific list involves the following tasks:

- Adding site-specific variables from a node configuration
- Adding site-specific variables manually
- Modifying site-specific variables
- Deleting site-specific variables

13.4.1 Adding Site-Specific Variables from a Node Configuration

A site-specific list is not bound to any configuration or configuration set. A particular configuration can use any site-specific list at export time, provided that the list contains all of the site-specific variables that are defined in the configuration. Consequently, it is possible for the configurations of two different IMS nodes to use the same site-specific list.

PDB can add all site-specific variables defined in a given configuration to a site-specific list. This action will also create site-specific variables derived from the conditions on parameters and parameter groups. In addition, PDB will add the mandatory CM variables for the configuration tool based on the selected configuration format. For more information on CM variables, refer to Section 18 on page 227.

When working with delta configurations, PDB has an option to include variables used by deleted parameters in a delta configuration. Adding variables from deleted parameters to a site-specific list allows you to resolve those variables when exporting a configuration in delta-only mode.

Note: This procedure can be repeated multiple times to append variables from multiple configurations. With each repeat, only the missing variables are added to the list.

Variable-value pairs from a configuration comparison can be added to a site-specific list from the Comparison report. For more information on extracting variables from a comparison report, refer to Table 46.

To add site-specific variables from a node configuration:

1. In the **Site-Specific Lists** workspace, select a list to work with.

The **Site Specific Parameters** table is displayed. See Figure 91.

Note: The **Site Specific Parameters** table shows the variables that are currently included in the list.

Site Specific Parameters				
Add from Configuration		Edit		
Name ▲	Value	Description	Usage	

Figure 91 Site-Specific Parameters Table

2. Click **Add from Configuration**.

The **Configurations** dialog box opens. See Figure 92.

Configurations	
Configuration	<input type="text"/>
Format	PVL ▼
<div>Add from Configuration</div> <div>Cancel</div>	

Figure 92 The Configurations Dialog Box

3. Select a node configuration in the **Configuration** field.

Note: This field uses auto-complete functionality. Typing part of the name or revision level displays matching node configurations. Use the down-arrow on your keyboard to display all of the available configurations.

The revision of the selected configuration is presented in square brackets at the end of the configuration name.



4. If required, select **Include variables used by deleted parameters**.

Note: This option is only available when importing variables from a delta configuration.

5. If required, select a different configuration format.

PDB uses the selected format to add the correct CM variables to the site-specific list. For more information on CM variables, refer to Section 18 on page 227. The default configuration format is specified by the PDB node definition and does not need to change in most cases.

Note: The PVL format does not add CM variables to the site-specific list.

6. Click **Add from Configuration**.

Site-specific variables from the selected node configuration are added to the list.

Note: Variables imported from a node configuration do not include values. To add site-specific values, refer to Section 13.4.4 on page 199.

13.4.2

Reporting Site-Specific Variable Usage

PDB can report the usage of site-specific variables in available node configurations. These reports include detailed information about the MOIs where variables are being used.

To report on the usage of a site-specific variable in a node configuration:

1. In the **Site-Specific Lists** workspace, select a list to report on.

The **Site Specific Parameters** table is displayed. See Figure 91.

Note: The **Site Specific Parameters** table shows the variables that are currently included in the list.



Site Specific Parameters				
<button>Add from Configuration</button>		<button>Edit</button>		
Name ▲	Value	Description	Usage	
HSS_OAM_PORT			CM VARIABLE	
HSS_OAM_VIP			CM VARIABLE	
HSS_USER_DN_ADMIN_NAME			CM VARIABLE	
HSS_USER_DN_LDAP_PWD			CM VARIABLE	
HSS_USER_DN_NODE_NAME			CM VARIABLE	
admin_state		The default administrative state		
primary_ccf_domain		The primary CCF domain		
primary_ccf_port		The primary CCF port		

Figure 93 Site-Specific Parameters Table

- Click next to the variable that you would like to report on.

The **Usage** dialog box opens. See Figure 94.

Usage

Configuration

Variable Name admin_state

Parameter Table

Parameter Group	Parameter Name	Value

Schema Information

Name

Description

Cardinality

Data Type

Default Value

Read Only

Restricted

Unit

Format Description

Value Constraints

Category

Status

Related Parameter(s)

Figure 94 Usage Dialog Box

- Select a node configuration to search in the **Configuration** field.



Note: This field uses auto-complete functionality. Typing part of the name or revision level displays matching node configurations. Use the down-arrow on your keyboard to display all of the available configurations.

The revision of the selected configuration is presented in square brackets at the end of the configuration name.

4. Click **Find**.

PDB searches the selected configuration for instances of the variable. Once the search is complete, the **Parameter** table is populated with instances of the site-specific variable that are found in the selected configuration.

Table entries can be selected to populate the schema information for the selected parameter instance.

Note: Reporting the usage of a specific variable automatically populates the **Usage** column in the **Site Specific Parameters** table with a comma-separated list of parameter names that include the variable. The column is populated for all variables that are found in the node configuration.

13.4.3

Adding Site-Specific Variables Manually

Variables can be added to a site-specific list manually. To automate this procedure by importing the site-specific variables from a node configuration, refer to Section 13.4.1 on page 193.

To add variables to a site-specific list:

1. In the **Site-Specific Lists** workspace, select a list to work with.

The **Site Specific Parameters** table is displayed. See Figure 95.

Note: The **Site Specific Parameters** table shows the variables that are currently included in the list.

Site Specific Parameters				
Add from Configuration		Edit		
Name ▲	Value	Description	Usage	

Figure 95 Site-Specific Parameters Table

2. Click **Edit**.

The **Site Specific Parameters** table switches to edit mode. See Figure 96.



Site Specific Parameters

Add

Delete

Apply




Cancel

	Name ▲	Value	Description	Usage	
--	--------	-------	-------------	-------	--

Figure 96 Site-Specific Parameters Table (Edit Mode)

The following table describes the different elements forming the **Site Specific Parameters** table in edit mode:

Table 58 Site Specific Parameters Interface Elements

Element	Description
	Adds a new site-specific parameter to the list.
	Removes the selected site-specific variables from the list.
Name	The name of the site-specific parameter. Mandatory.
Value	The value for the associated site-specific parameter. Mandatory.
Description	If the parameter value variable is present on the Global Variables list, this column will display the associated description, if available.
Usage	After reporting the usage of a site-specific variable within a configuration, this column presents a comma-separated list of parameter names that include the variable.
	Reports the usage of the site-specific parameter in available node configurations. For more information, refer to Section 13.4.2 on page 195.

3. Click **Add**.

An empty parameter is added to the top of the table. See Figure 97.



Site Specific Parameters

	Name ▲	Value	Description	Usage	
<input type="checkbox"/>	<input type="text"/>	<input type="text"/>			

Figure 97 Adding a New Site-Specific Parameter

- Enter the required information. See Table 58.

Note: Site-specific variables are not case sensitive.

- Repeat steps 4 and 5 to add more variables.
- Click **Apply**.

The new variables are added to the database.

13.4.4 Modifying Site-Specific Variables

The properties of existing site-specific variables can be modified in PDB.

To modify a site-specific variable:

- In the **Site-Specific Lists** workspace, select a list to work with.

The **Site Specific Parameters** table is displayed. See Figure 98.

Note: The **Site Specific Parameters** table shows the variables that are currently included in the list.



Site Specific Parameters				
<div>Add from Configuration Edit</div>				
Name ▲	Value	Description	Usage	
HSS_OAM_PORT			CM VARIABLE	
HSS_OAM_VIP			CM VARIABLE	
HSS_USER_DN_ADMIN_NAME			CM VARIABLE	
HSS_USER_DN_LDAP_PWD			CM VARIABLE	
HSS_USER_DN_NODE_NAME			CM VARIABLE	
admin_state		The default administrative state		
primary_ccf_domain		The primary CCF domain		
primary_ccf_port		The primary CCF port		

Figure 98 Site-Specific Parameters Table

2. Click **Edit**.

The **Site Specific Parameters** table switches to edit mode. See Figure 99.

Site Specific Parameters					
<div>Add Delete Apply Cancel</div>					
	Name ▲	Value	Description	Usage	
<input type="checkbox"/>					
<input type="checkbox"/>	HSS_OAM_PORT			CM VARIABLE	
<input type="checkbox"/>	HSS_OAM_VIP			CM VARIABLE	
<input type="checkbox"/>	HSS_USER_DN_ADMIN_NAME			CM VARIABLE	
<input type="checkbox"/>	HSS_USER_DN_LDAP_PWD			CM VARIABLE	
<input type="checkbox"/>	HSS_USER_DN_NODE_NAME			CM VARIABLE	
<input type="checkbox"/>	admin_state		The default administrative state		
<input type="checkbox"/>	primary_ccf_domain		The primary CCF domain		
<input type="checkbox"/>	primary_ccf_port		The primary CCF port		

Figure 99 The Site-Specific Parameters Table (Edit Mode)



Table 58 describes the different elements forming the **Site Specific Parameters** table in edit mode.

3. Update the site-specific information as required.

Note: Site-specific variables are not case sensitive.

4. Click **Apply**.

The updated variables are saved to the database.

13.4.5 Deleting Site-Specific Variables

Site-specific variables can be removed from a site-specific list as needed.

To remove variables from a site-specific list:

1. In the **Site-Specific Lists** workspace, select a list to work with.

The **Site Specific Parameters** table is displayed.

Note: The **Site Specific Parameters** table shows the variables that are currently included in the list.

2. Click **Edit**.

The **Site Specific Parameters** table switches to edit mode.

3. Click the check boxes next to the variables you want to remove.

4. Click **Delete**.

A confirmation dialog box opens.

5. Click **OK**.

6. Click **Apply**.

14 Transferring PDB Data

Data transfer allows you to move information from one PDB server to another. In addition to moving objects between servers, a data transfer preserves object metadata such as the name, document number, revision and so on. After information has been moved to the target PDB server, the transferred objects must be integrated with the existing PDB data structure. Data transfer allows you to make certain modifications to the object metadata that will allow PDB to correctly integrate the new information.



Data transfer allows you to move the following information:

- Configuration Schemas
- Node Configurations
- Site-Specific Lists

14.1 The Data Transfer Workspace

The **Data Transfer** workspace allows you to perform import and export operations on PDB data.

To access the **Data Transfer** workspace, select **Data Transfer** from the menu options on the left.

The workspace is divided into **Export** and **Import** tabs.

Export Tab

The **Export** tab allows you to export PDB data to a data transfer bundle. Configurations and schemas are added to the export list by selecting **Add to Data Transfer** from the context menu in their respective workspaces. See Figure 100.

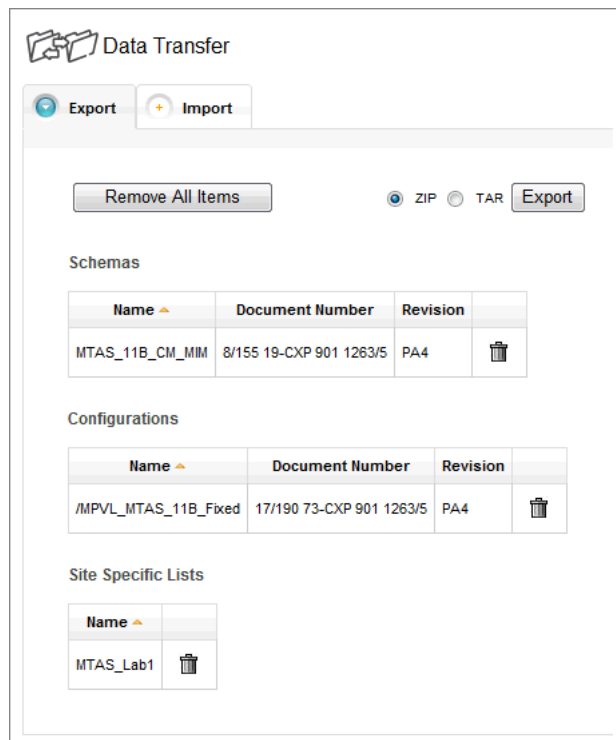
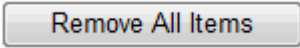



Figure 100 Data Transfer Export Tab




The following table describes the elements of the **Export** tab in the **Data Transfer** workspace.

Table 59 Data Transfer Export Elements

Element	Description
	Clears the export list of all items.
Archive Type (Option Buttons)	Select whether to export the data transfer bundle in ZIP or TAR format.
Name	<p>The name of the configuration, schema, or site-specific list.</p> <p>Note: PDB uses a path-based naming convention for configurations and schemas. In this format, a configuration or schema name is appended to the name of any predecessors in the following structure:</p> <pre>/ <root object> / <object 1> / ...</pre>
Document Number	The document number associated with the configuration or schema.
Revision	The revision of the configuration or schema.
	Click to remove the associated object from the export list.

Import Tab

The **Import** tab allows you to import a data transfer bundle containing PDB data into the PDB server. Individual objects are represented by their metadata. The metadata can be edited to successfully integrate the new information with the current data structure. See Figure 101.


Data Transfer

Export
Import

Step One - Upload File

Input File Browse... Upload Clear

datatransfer_bundle-20110829_162011.zip

Step Two - Review Metadata

Schemas

☒

☒ Import as New Schema
☐ Import as New Revision

Edit

Node

Name

Document Number Revision

Document State

Description

Revision Comment

Initial Value List ☒ Attach an Initial Value List

Name

Document Number

Configurations

☒

☐ Import as New Configuration
☒ Import as New Revision

Edit

Name

Document Number Revision

Document State

Description

Revision Comment

Predecessor

Schema

Flags ☒ MPVL ☒ DPVL ☐ IFN ☐ IVL ☒

Site Specific Lists

☒

Edit

Name

Description

Step Three - Import Data

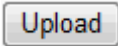

Import

Figure 101 Data Transfer Import Tab




The following table describes the elements of the **Import** tab in the **Data Transfer** workspace.

Table 60 Data Transfer Import Elements

Element	Description
	Uploads the selected data transfer bundle to PDB. Note: Click Upload again to reset the import operation.
	Clears the import list of all items.
Objects included in the uploaded data transfer package are displayed as metadata before they are imported to PDB.	
Schema Metadata	The following metadata are displayed, if applicable: <ul style="list-style-type: none"> • Node • Name • Document Number • Revision • Description • Revision Comment • Initial Value List
Configuration Metadata	The following metadata are displayed, if applicable: <ul style="list-style-type: none"> • Name • Document Number • Revision • Description • Revision Comment • Predecessor • Schema • Parent • Flags



Element	Description
Site-Specific List Metadata	The following metadata are displayed, if applicable: <ul style="list-style-type: none">• Name• Description
Import Type (Option Buttons)	For schemas and configurations, select whether to import as a new object or a new revision of an existing object.
	Opens an Object's metadata for editing. The metadata may require modification to successfully integrate the object into the current data structure.

14.2 Exporting PDB Data

During a data transfer, PDB data is exported to a data transfer bundle. Data transfer bundles are ZIP or TAR archives that contain PDB objects and associated metadata. To transfer data, an exported bundle must be uploaded to another PDB server. For more information on importing a data transfer bundle, refer to Section 14.3 on page 207.

Note: Data transfer objects are stored using a proprietary format that is unique to PDB. Configurations exported using data transfer cannot be used to configure nodes directly.

Data transfer allows you to export the following information:

- Configuration Schemas
- Node Configurations
- Site-Specific Lists

To export objects to a data transfer bundle:

1. In the corresponding workspace, right-click an object and select **Add to Data Transfer**.

You must have ACL permission to view objects in order to add them to a data transfer. For more information on ACLs, refer to Section 3.1 on page 9..

Note: Adding a schema with an IVL to a data transfer adds both the schema and the IVL configuration.



The object is added to the data transfer export list. The total number of objects added to a data transfer is listed in parenthesis beside the **Data Transfer** menu item on the left.

Repeat this step until all of the required objects have been added to the data transfer.

Note: Data transfer items are stored on a per session basis. Logging out from PDB resets the export list.

2. Open the **Data Transfer** workspace and select the **Export** tab to review transfer list, making modifications where necessary.
3. When the list of objects is complete, select an archive type and click **Export**.

The selected objects are exported to a data transfer bundle. When PDB has finished processing the data, a link to the bundle is displayed.

4. Click **Download Transfer File** to save the archive.

14.3 Importing PDB Data

Data transfer bundles must be imported to PDB in order to add their contents to the database. When importing a data transfer bundle, objects must fit within existing PDB data structure. Because data transfer objects are stored with their original metadata, making objects fit often requires changes to their metadata. Data transfer allows you to make these changes before importing the objects. This ensures that all objects are correctly positioned within PDB.

PDB strictly controls the relationships between configurations or schemas. When introducing these objects during a data transfer, the object metadata must satisfy PDB requirements or the import operation will fail.

Here are some general rules and guidelines to be aware of:

- Node configurations in PDB require schemas. If the original schema for a given configuration is not included in the data transfer package, you can select a schema on the PDB server or a different schema within the data transfer package. You may specify a different revision of the original schema. Compatibility between the configuration and the selected schema will be verified during the import process.
- All configurations and schemas are revision controlled. When introducing an object as a new revision:
 - The predecessor must be frozen.
 - The successor must have a higher revision than the predecessor.
 - Descriptions are tied to the first predecessor and will not be imported. If a description is needed, it can be input as a **Revision Comment**.



- Delta configurations inherit data from parent configurations. If the original parent of a given delta configuration is not included in the data transfer package, you can select a new parent from the PDB server or the data transfer package. Both the parent and the delta configuration must use the same schema revision.

To import objects in a data transfer bundle:

1. In the **Data Transfer** workspace, select the **Import** tab.

The **Import** opens with **Step One - Upload File**.

2. Select and input file and click **Upload**.

Metadata for each object in the data transfer bundle is displayed under **Step Two - Review Metadata**.

You can remove objects from the import by clearing the check box next to each entry.

Note: Each object that is imported to PDB must fit within the existing data structure. You must review the object metadata and ensure that the object it will fit within PDB and make modifications where necessary.

Associations between node configurations and site-specific lists are retained on a best-effort basis. To preserve these associations, the specified configurations must be present in the target PDB or the data transfer package.

3. If necessary, edit the object metadata by clicking the associated **Edit** button.

The object metadata opens in edit mode.

4. Modify the metadata as needed, then click **Apply**.

Note: The following fields use auto-complete functionality. Typing part of a name, revision or document number displays a list of objects that match the criteria. Use the down-arrow on your keyboard to display the complete list.

- Initial Value List
- Node
- Parent
- Predecessor
- Schema

5. Click **Import** under **Step Three - Import Data**.

PDB performs the following operations when importing data transfer objects.



- a PDB validates the object metadata and ensures the incoming objects will fit within the existing PDB data structure.

Note: Any metadata errors will cause the import process to fail.

- b For configurations and schemas, PDB performs a validation check on the incoming data.
 - For more information on schema validation, refer to Section 7.2 on page 45.
 - For more information on configuration validation, refer to Section 8.8 on page 101.

Note: Validation errors will cause the import process to fail. This can happen when an incompatible schema is selected for an incoming configuration.

- c PDB generates a data transfer import report that can be downloaded from the **Report Available** dialog box that appears at the end of the import process.

The report contains:

- An operation summary
- The results of the metadata validation.
- Validation reports for each configuration and schema included in the data transfer.

Appendix

15 Export Criteria

When exporting a node configuration, PDB follows a set of business rules that govern the format and structure of the configuration data. Export criteria are specific to the selected format and describe how configuration elements are represented within the format structure.

Specific rules exist for the following configuration formats:

- LDIF
- PVL

15.1 LDIF Export Criteria

The following list shows the general export criteria for LDIF configurations:

- The default change type is ADD.
- Parameter Groups are exported as LDIF entries.
- Parameters are exported as LDIF attributes.
- Parameter Groups or Parameters with status deprecated or obsolete are not exported.
- Parameters set to `readonly` are not exported.
- Parameters set with a default value are not exported.
- If the configuration schema is associated with an IVL, any parameters set with a value matching the IVL are not exported.

The following list shows specific export criteria for special cases:

- Parameter Groups that are system created with a default primary key value are not exported unless they include parameters that are exported (see other rules). Parameter groups matching this criteria are exported as MODIFY operations.



- Parameter Groups with a primary key value matching an IVL entry are not exported unless they include parameters that are exported (see other rules). Parameter groups matching this criteria are exported as MODIFY operations.
- Parameter Groups that are system created with a readonly primary key are not exported unless they include parameters that are exported (see other rules). Parameter groups matching this criteria are exported as MODIFY operations.
- When exporting a delta configuration in LDIF format as Delta Only, inherited configuration elements that are deleted in the delta configuration will not be included in the configuration file.

Sorting Criteria

Parameter Groups and Parameters are sorted to avoid conflicts when LDIF files are used to configure nodes.

The following relationships are used to sort the Parameter Groups and Parameters within LDIF configuration files:

- Parent-Child Relationships (class-class). Parent entities encapsulate child entities.
- Parameter interdependencies.

15.2 PVL Export Criteria

The representation of configuration elements within a PVL configuration file is guided by the requirements outlined in the Parameter List Template Description, EAB/FTI-08:0686 Uen.

The following list shows the export criteria for root or delta configurations exported in PVL format as an Entire Configuration:

- The PVL format supports multi-solution configurations of type `ims_fixed`, `ims_mobile`, `converged`, and `node_standalone`.
 - When exporting a single configuration in PVL format, only the user selected PVL solution type is populated with configuration data. Other solution types are populated with `[empty]`.
 - When exporting a multi-solution configuration, all of the user selected PVL solution types are populated with configuration data. Any solution type not included in the configuration bundle is populated with `[empty]`.
- The `defaultValue` field is populated with initial values from an associated IVL. Without initial values, the `defaultValue` field is populated with `[empty]`.
- The `paraCR` field is populated with `[empty]`.



- The `comment` field is populated with `[empty]` .
- The `ID` field is an enumeration of the row number within the PVL file.

The following list shows the specific export criteria for delta configurations exported in PVL format as Delta Only:

- The user selected PVL solution type (`ims_fixed`, `ims_mobile`, `converged`, `node_standalone`) is populated with inherited data from the parent configuration.
- The `delta` field is populated with the delta value.
- If an inherited configuration element is deleted in the delta configuration, the corresponding `delta` field is populated with `[Not Used]`.
- If a delta configuration includes new configuration elements that are not present in the parent configuration, the new elements are added to the user selected PVL solution type with `[empty]` values and the corresponding `delta` fields are populated with the delta values.

16 Configuration Validation Errors

PDB automatically validates configuration files during the import process. This check verifies the syntax of the configuration files and validates the node configuration against the selected schema.

Problems encountered during import validation generate informational messages that are presented in an import validation report that is available at the end of the import process. For more information on importing a node configuration, refer to Section 8.2 on page 74.

All Import validation messages have one of the following severity levels:

ERROR	An error indicates a serious problem with the node configuration. Any error will the configuration import process to fail.
WARNING	Warnings indicate that a problem was identified during the import process that impacts the configuration data. Warnings should be analyzed and corrected if necessary.
INFO	Info messages are user information and have no consequences on data integrity



Messages found in the import validation report are applicable to the following configuration formats:

- All Configuration Formats
- LDIF
- NETCONF
- MPVL

Messages are described in the following sections.

16.1 Messages Applicable to All Configuration Formats

The following table describes the import validation error messages that are applicable to all configuration formats.

Table 61 Errors Applicable to All Configuration Formats

Error Message	Description
<ul style="list-style-type: none"> Parameter Group Instance <PGI_DN> already exists for entry <DN_or_MOI>. Import aborted. Parameter Group Instance <PGI_DN> already exists. Import aborted. 	<p>A parameter group instance that PDB is trying to import is duplicated in the configuration file. PDB has aborted the import process.</p> <ul style="list-style-type: none"> For MPVL configurations, the error message will include a MOI for the affected instance. For LDIF configurations, the error message will include a DN for the affected instance. For NETCONF, the error message will include no entry.
<ul style="list-style-type: none"> Parameter instance <PI_Name> with value <PI_Value> already exists for entry <DN_or_MOI>. Import aborted. Parameter instance <PI_Name> with value <PI_Value> already exists for parameter group instance <PGI_DN>. Import aborted. 	<p>A parameter instance with the specified value that PDB is trying to import is duplicated in the configuration file. PDB has aborted the import process.</p> <ul style="list-style-type: none"> For MPVL configurations, the error message will include a MOI for the affected instance. For LDIF configurations, the error message will include a DN for the affected instance. For NETCONF, the error message will include no entry.

The following table describes the import validation warning messages that are applicable to all configuration formats.



Table 62 Warnings Applicable to All Configuration Formats

Warning Message	Description
<ul style="list-style-type: none">• Root parameter group <PG_Name> was not found in the schema. Ignoring entry with path <DN_or_MOI>.• Root parameter group <PG_Name> not found. Parameter group ignored.	<p>A root parameter group that PDB was trying to import was not found in the schema definition. The parameter group and all child elements have been ignored.</p> <ul style="list-style-type: none">• For MPVL configurations, the error message will include a MOI for the affected instance.• For LDIF configurations, the error message will include a DN for the affected instance.• For NETCONF, the error message will include no entry.
<ul style="list-style-type: none">• Parameter group <Child_PG_Name> was not found in the schema under parameter group <Parent_PG_Name>. Ignoring entry with path <DN_or_MOI>.• Parameter group <Child_PG_Name> was not found in the schema under parameter group <Parent_PG_Name>. Parameter group ignored.	<p>A child parameter group that PDB was trying to import was not found in the schema definition. The parameter group has been ignored.</p> <ul style="list-style-type: none">• For MPVL configurations, the error message will include a MOI for the affected instance.• For LDIF configurations, the error message will include a DN for the affected instance.• For NETCONF, the error message will include no entry.



Table 62 Warnings Applicable to All Configuration Formats

Warning Message	Description
<ul style="list-style-type: none"> Primary key parameter <Parameter_Name> was not found in the schema under parameter group <PG_Name>. Ignoring entry with path <DN_or_MOI>. Parameter <Parameter_Name> was not found in the schema under parameter group <PG_Name> not found. Parameter ignored. Parameter <Parameter_Name> with value <Parameter_Value> under parameter group <PG_Name> was not found. Parameter ignored. 	<p>A parameter that PDB was trying to import was not found in the schema definition. The parameter has been ignored.</p> <ul style="list-style-type: none"> For MPVL configurations, the error message will include a MOI for the affected instance. For LDIF configurations, the error message will include a DN for the affected instance. For NETCONF, the error message will include no entry.
The parent of parameter group instance <PGI_Name> was not found in the schema. Ignoring entry with path <DN_or_MOI>.	<p>The parent parameter group instance of <PGI_Name> was not found in the node configuration being imported and it has been ignored along with all child elements. The configuration file could be missing a line or have a typo.</p> <ul style="list-style-type: none"> For MPVL configurations, the error message will include a MOI for the affected instance. For LDIF configurations, the error message will include a DN for the affected instance. <p>This warning message does not apply to NETCONF configurations.</p>

16.2 Messages Applicable to the LDIF Format

The following table describes the import validation error messages that are applicable to the LDIF configuration format.

Table 63 Errors Applicable to the LDIF Format

Error Message	Description
No ldif file found.	The selected ZIP or TAR file does not contain LDIF files (files with the .ldif extension).
<Java_I/O_Exception>	Problem accessing the input file, such as a disk I/O issue.

The following table describes the input validation warning messages that are applicable to the LDIF configuration format.



Table 64 Warnings Applicable to the LDIF Format

Warning Message	Description
ObjectClass missing for entry applicationName=<app_name>,nodeName=<node_name>, ignoring it.	An object class is missing in the configuration data. The affected parameter group has been ignored.
Duplicate entry for DN: <DN> ignoring duplicate entry.	The specified DN entry is duplicated in the configuration file being imported. The duplicate entry has been ignored by PDB.

16.3 Messages Applicable to the NETCONF Format

The following table describes the import validation error messages that are applicable to the NETCONF configuration format.

Table 65 Errors Applicable to the NETCONF Format

Error Message	Description
No netconf file found.	The selected ZIP or TAR file does not contain a NETCONF file (a file with the .xml extension).
<XML_Parser_Exception>	There was a problem with the XML file being imported. Problems can vary depending on the issue.
<Java_I/O_Exception>	Problem accessing the input file, such as a disk I/O issue.
Should only have one netconf file.	The selected ZIP or TAR file contains multiple NETCONF files (files with the .xml extension).
Missing primary key for parameter group: <PG>.	A primary key was missing for the specified parameter group in the imported configuration.

16.4 Messages Applicable to the MPVL Format

The following table describes the import validation error messages that are applicable to the MPVL configuration format.



Table 66 Errors Applicable to the MPVL Format

Error Message	Description
Failed to parse PVL file '<file_name>' <message>.	There was a problem with the PVL file being imported. Problems can vary depending on the issue.
Key parameter '<Parameter_Name>', with value '<Parameter_Solution_Value>' doesn't match moi key value, for moi '<MOI>'.	The value of a primary key in the MOI does not match the value in the solution column.

The following table describes the input validation warning messages that are applicable to the MPVL configuration format.

Table 67 Warnings Applicable to the MPVL Format

Warning Message	Description
MOC mismatch for parameter '<Parameter_Name>' with MOI '<MOI>', found MOC='<MOC_Found_1>' and MOC='<MOC_Found_2>', imported using MOC='<MOC_Found_1>'.	Entries were found with the same MOI but different MOCs. Both entries will be imported using the first MOC. This needs attention from the user.
could not parse MOI for entry with MOI '<MOI>', ignoring it. Using '<Format_Being_Used>' MOI format.	The specified entry has an invalid MOI that cannot be parsed. <Format_Being_Used> can be TSP or IS. The entry cannot be imported and it is ignored.
could not import parameter '<Parameter_Name>' for entry with moi '<MOI>'.	Default message when an entry cannot be imported. The code could not determine the cause. The entry cannot be imported and it is ignored.
Parameter group '<MOC>' not found for moi '<MOI>'.	The imported configuration contains an entry (with moi: <MOI>) that makes reference to a parameter group (<MOC>) that does not exist in the schema (MIM). The entry cannot be imported and it is ignored.

The following table describes the input validation informational messages that are applicable to the MPVL configuration format.



Table 68 Infos Applicable to the MPVL Format

Info Message	Description
Ignoring parameter '<Parameter_Name>' for entry with moi '<MOI>' it has [empty] or [not used] value.	The entry with <MOI> and <Parameter_Name> was not imported because the solution value is [empty] or [not used].
Importing zero-length string for parameter <Parameter_Name> with moi <MOI>.	The entry with <MOI> and <Parameter_Name> was imported as a zero length string (that is "").
Missing primary key definition for parameter group '<MOC>', creating it with values from MOI <MOI>.	There is no entry defining the primary key for the specified parameter group instance. A value for the primary key is inferred from the MOI and created by PDB.

17 MIM Validation Errors

MIM validation generates errors and warnings that are applicable to the following MIM formats:

- All CM MIM Formats
- IS CM MIM Format
- MP_DTD Format

These messages appear in the validation report that is generated by PDB after importing a schema or using the standalone CLI MIM validator.

For more information on importing a configuration schema, refer to Section 7.2 on page 45.

For more information on the standalone MIM validator, refer to `validate-mim` in the PDB Command Line Interface (CLI) Reference, 1/1540-CXP 902 0212.

17.1 Errors and Warnings Applicable to All CM MIM Formats

The following table describes the MIM validation error messages that are applicable to all CM MIM formats.



Table 69 Errors Applicable to All CM MIM Formats

Error Message	Description
No primary key found for class <className>.	<p>All classes of the given format must have at least one attribute flagged as primary key. The mentioned class does not comply with the rule.</p> <p>Possible causes are:</p> <p>The primary key attribute is not defined at all.</p> <p>The primary key attribute is defined but it is missing the flag that marks it as primary key:</p> <ul style="list-style-type: none"> • <keyType>primary</keyType> in TSP CM MIM • <key/> in IS CM MIM
Multiple primary keys were found in class <className>. Only one primary key is allowed for this MIM format.	<p>The given MIM format only allows one primary key attribute per class. The mentioned class does not comply with this rule.</p> <p>Possible cause is:</p> <p>There is more than one attribute with the primary key flag:</p> <ul style="list-style-type: none"> • <keyType>primary</keyType> in TSP CM MIM • <key/> in IS CM MIM
Unknown constraint type <constraintType> found for attribute <attributeName> of class <className>.	The data type specified for the given attribute is not defined for the given MIM format.
Invalid constraint for attribute <attributeName>. For input string:"<message with further details>".	The data type specified for the given attribute is known and valid but constraints within the datatype are not valid.
Invalid range values for attribute <attributeName>, min = <minValue>, max = <maxValue>.	The value of the given parameter is outside of the allowed range of values.



Table 69 Errors Applicable to All CM MIM Formats

Error Message	Description
Invalid cardinality in relationship <relationship> between parent class <parentClass> and child class <childClass>, cardinality = (<parentCardinality>-<childCardinality>).	The cardinality defined in the given class-class relationship is invalid.
Invalid cardinality in relationship <relationship> from attribute <fromClassName>.<fromAttributeName> to attribute <toClassName>.<toAttributeName>, cardinality = (<fromCardinality>-<toCardinality>), only 0 and 1 are allowed.	In an attribute-attribute relationship, the cardinality of the FROM can only take a value of 0 or 1.
ERROR in relationship '<relationshipName>', the parent class 'className' in mim 'mimName' was not found.	<p>A relationship is making reference to a parent class that is not found among the MIM files being imported.</p> <p>Possible causes are:</p> <ul style="list-style-type: none">• A mismatch between the name of the parent class and the class in the relationship (typo)• The parent class does not exist
Parent class <parentClass> referenced by child class <childClass> not found.	<p>A child class is making reference to a parent class that is not found among the MIM files being imported.</p> <p>Possible causes are:</p> <ul style="list-style-type: none">• A mismatch between the name of the parent class and the class in the relationship (typo)• The parent class does not exist
Child class <childClass> referenced in relationship <relationshipName> not found.	<p>A relationship is making reference to a child class that is not found among the MIM files being imported.</p> <p>Possible causes are:</p> <ul style="list-style-type: none">• A mismatch between the name of the child class and the class in the relationship (typo)• The child class does not exist



Table 69 Errors Applicable to All CM MIM Formats

Error Message	Description
Attribute <attributeName> defined in relationship <relationshipName> not found in class <className>.	<p>A relationship is making reference to an attribute that is not found in the specified class.</p> <p>Possible causes are:</p> <ul style="list-style-type: none"> • A mismatch between the name of the attribute defined in the class and the attribute defined in the relationship (typo) • The attribute does not exist
Circular relationship found for attribute <attributeName>.	<p>Circular relationships are not allowed. A simple example of a circular relationship is as follows:</p> $A \rightarrow B, B \rightarrow C, C \rightarrow A$

The following table describes the MIM validation warning messages that are applicable to all CM MIM formats.

Table 70 Warnings Applicable to All CM MIM Formats

Warning Message	Description
Unknown data type <dataType> for attribute <attributeName>. Defaulting to STRING without constraints.	There is no data type defined for the given attribute, so the STRING data type is automatically assigned.
Unknown data type <dataType> for attribute <attributeName> of class <className>.	There is a data type defined for the given attribute, but the data type is not defined in any of the MIM files being validated.
Relation <relationshipName> is of the type class-attribute and it is not supported.	The relationship between a class and an attribute is not supported by PDB. PDB will disregard the relationship and continue importing.
Relationship <relationshipName> is of the type <type> and it is not supported.	The relationship of a given type is not supported by PDB. PDB will disregard the relationship and continue importing.



Table 70 Warnings Applicable to All CM MIM Formats

Warning Message	Description
Attribute <attributeName> already defined in class <className>. Skipping it.	<p>The attribute has been already defined for that given class. PDB will disregard subsequent definitions of the attribute and continue importing.</p> <p>Possible causes are:</p> <ul style="list-style-type: none">• The attribute is defined more than once within the same MIM file• The class and its attributes are defined in more than one MIM file
Invalid category <category> for class <className>. Category not imported.	<p>The category defined for a given class is not valid. For example, only the following categories are allowed in a TSP CM MIM:</p> <ul style="list-style-type: none">• Internal• Operator Configurable• Site Specific• Solution Integration <p>PDB will disregard the category for that given attribute and continue importing.</p>
Invalid status <status> for class <className>. Status not imported.	<p>The status defined for a given class is not valid. For example, only the following statuses are allowed in a TSP CM MIM:</p> <ul style="list-style-type: none">• current• deprecated• obsolete <p>PDB will disregard the status for that given class and continue importing.</p>
Invalid status <status> for attribute <attributeName>. Status not imported.	<p>The status defined for a given attribute is not valid. For example, only the following statuses are allowed in a TSP CM MIM:</p> <ul style="list-style-type: none">• current• deprecated• obsolete <p>PDB will disregard the status for that given class and continue importing.</p>



Table 70 Warnings Applicable to All CM MIM Formats

Warning Message	Description
Duplicate relationship <relationship> found. Attribute <fromAttributeName> is already referencing attribute <toAttributeName>.	The relationship between two attributes has been already defined. PDB will disregard subsequent definitions of the relationship and continue importing the rest of the files.
Invalid default value for attribute <attributeName> of class <className>. Value <value> doesn't comply to constraint: <constraint>.	The default value of the given attribute does not comply to the constraints specified for the attribute. For example: If the parameter is an integer, the default value abc is invalid.

17.2 Errors and Warnings Applicable to the IS CM MIM Format

The following table describes the MIM validation error messages that are applicable to the IS CM MIM format.

Table 71 Errors Applicable to the IS CM MIM Format

Error Message	Description
No index file for MIM file <mimFilename>.	<p>It was not possible to find an index file making reference to the given MIM file.</p> <p>Possible causes are:</p> <ul style="list-style-type: none"> • A mismatch between the MIM filename and the filename defined on the index file (typo) • None of the index files contain the MIM filename
Top class <nameOfTopClass> not defined in MIM file <mimFilename>.	<p>Every IS_CM_MIM file must have a top class defined. While the given MIM file has a top class definition, the definition of that class itself is missing.</p> <p>Possible causes are:</p> <ul style="list-style-type: none"> • The class definition is missing • There is a typo in either the class definition or the top class definition



Table 71 Errors Applicable to the IS CM MIM Format

Error Message	Description
Model <modelName> is not defined.	<p>There is a MIM file associated with an index file that eventually makes reference to a model that it is not defined.</p> <p>Possible causes are:</p> <ul style="list-style-type: none">• Typo in the value of the name attribute of the element <model> inside the model file• Typo in the value of the model attribute of the element <mim_file> inside the index file• The model is not defined
No mount point defined for model <modelName>.	<p>The given model has no mount point defined.</p> <p>Possible cause is: The <mountPoint> element is missing inside the mode file.</p>
No model class found for mount point <mountPointName>.	<p>There is a model file that has a mount point that is not defined as a model.</p> <p>Possible causes are:</p> <ul style="list-style-type: none">• Inside the model file, the name attribute of the element <mountpoint> has a typo• The name attribute of the element <mountpoint> is pointing to a nonexistent model
Class <className> is declared Singleton yet has primary keys defined. This is not valid.	<p>It is possible to have classes which are singleton. But singleton classes, by definition, do not have primary keys defined. The given MIM file has, for that given singleton class, at least one attribute flagged as primary key.</p>



Table 71 Errors Applicable to the IS CM MIM Format

Error Message	Description
No model file found for model <modelName>.	<p>There is an index file making reference to a model that is not defined.</p> <p>Possible cause is: Inside the index file, the model attribute of the <mim_file> entry is pointing to a nonexistent model</p>
No MIM file found for model <modelName>.	<p>There is a model defined but there are no MIM files making reference to it.</p> <p>Possible causes are:</p> <ul style="list-style-type: none"> • Inside the model file, the name attribute of the <model> entry has a typo and does not match the value defined in any of the index files • The index file pointing to the model contains MIM files that are also defined in another index file. PDB associates models with the index file that was read first. This scenario can lead to no MIM files pointing to a given model.

The following table describes the MIM validation warning messages that are applicable to the IS CM MIM format.

Table 72 Warnings Applicable to the IS CM MIM Format

Warning Message	Description
MIM file <mimFilename> not found.	<p>An index file is making reference to MIM file that doesn't exist.</p> <p>Possible causes are:</p> <ul style="list-style-type: none"> • A mismatch between the MIM filename and the filename defined on the index file (typo) • The MIM file does not exist

17.3 Errors and Warnings Applicable to the MP_DTD Format

The following table describes the MIM validation error messages that are applicable to the MP_DTD format.



Table 73 Errors Applicable to the IS CM MIM Format

Error Message	Description
<mimName> must match current MIM in intra-mim relationship. Error in MIM relationship '<relationshipName>'.	The MIM name specified in an intra-mim relationship must match the current MIM name.
Relationship '<relationshipName>' would add child parameter group '<childParameterGroupName>' to '<parentParameterGroupName>' but a group with the same name was already added.	Multiple copies of a parameter group cannot be added to the same parent.
invalid data type reference, mim: <mimName>, type: <dataType>, for attribute: <attributeName>.	Undefined data types are not supported.
MIM file: <mimName1> and <mimName2> extend the same model: <modelName>.	Multiple MIMs cannot extend the same model.

The following table describes the MIM validation warning messages that are applicable to the MP_DTD format.

Table 74 Warnings Applicable to the MP_DTD Format

Warning Message	Description
The attribute '<attributeName>' should start with a lower case letter.	According to ECIM rules, attributes should start with a lower case letter.
The attribute '<attributeName>' cannot be mandatory and at the same time be readOnly.	According to ECIM rules, mandatory attributes must be read/write.
The attribute '<attributeName>' cannot be mandatory and at the same time have a default value.	According to ECIM rules, mandatory attributes cannot have a default value.



Table 74 Warnings Applicable to the MP_DTD Format

Warning Message	Description
The attribute '<attributeName>' cannot be mandatory and have multiplicity lower band equal to zero at the same time.	According to ECIM rules, mandatory attributes cannot have a multiplicity lower band equal to zero. (0-n)
Class '<className>' should start with an upper case letter.	According to ECIM rules, classes should start with an upper case letter.
Cardinality already set for class '<className>' in another relationship with a different value, keeping value of first relationship 0-1.	The cardinality of a MOC is set by the cardinality of the child element in the relationship. If a second relationship specifies different cardinality values for the MOC, the previous values are retained.
Only containment relationships are supported, ignoring relationship '<relationshipName>' of type '<relationshipType>'.	Only relationships with the specified type are supported. Unsupported relationships are ignored.

18 Configuration Management Variables

The following CM variables are used by PDB to resolve values required by the configuration tool itself. A different set of CM variables is required by each configuration format.

TSP LDAP based Nodes (for example, HSS, CSCF, MTAS)

- <application>_DN_NODE_NAME: The name of the node, used as value for the nodeName attribute (e.g. jambala)
- <application>_OAM_PORT: The port used to connect to the TSP LDAP server (for example, secure port, 7423)
- <application>_OAM_VIP: The OAM VIP IP address of the TSP node, at which the LDAP server can be reached (Ex: 172.30.39.16)



- `<application>_USER_DN_ADMIN_NAME`: The LDAP administrator name (for example, jambala)
- `<application>_USER_DN_LDAP_PWD`: The LDAP administrator password (for example, Pokemon1)

The LDAP password is required by the `run_configure.sh` script to configure the node. If this variable is left blank at export time, `run_configure` will prompt for the password.

- `<application>_USER_DN_NODE_NAME`: The name of the node, used as value for the `nodeName` attribute (e.g. jambala)

Note: CM variables for LDIF configurations are prefixed by the associated application name. For example: `HSS_DN_NODE_NAME`.

IS Netconf based nodes (for example, MGW):

- `<application>_NC_HOST`: The public IP address to which the IS node's Configuration Management Frameworkd (CMF) interface can be contacted, (CMF - IP address: 172.20.30.19)
- `<application>_NC_PORT`: The public port on which the NetConf server is running (external port of CMF, such as 830)
- `<application>_NC_USER`: The CMF standard administrator user (such as expert)
- `<application>_NC_PASSWORD`: The password of the CMF standard administrator user (password such as expert)

The NETCONF password is required by the `run_configure.sh` script to configure the node. If this variable is left blank at export time, `run_configure` will prompt for the password.

Note: CM variables for NETCONF configurations are prefixed by the associated application name. For example: `MGW_NC_HOST`.

TSP EAS based nodes

- `IO1_IP`: the public IP address of IO1 of the EAS cabinet (Ex: 172.20.30.16)
- `IO2_IP`: the public IP address of IO2 of the EAS cabinet (Ex: 172.20.30.17)
- `IO_ROOT_PWD`: the password for the IO1/IO2 root user (Ex: root)