

**Oracle Communications® ASAP™ Cartridge 1.0
GA Release for Ericsson INS 2.1 Cartridge**

Ericsson INS 2.1 Cartridge Guide

Second Edition
July 2008

ORACLE®

Copyright and Trademark Information

Copyright © 1992, 2008, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this software or related documentation is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, duplication, disclosure, modification, and adaptation shall be subject to the restrictions and license terms set forth in the applicable Government contract, and, to the extent applicable by the terms of the Government contract, the additional rights set forth in FAR 52.227-19, Commercial Computer Software License (December 2007). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

This software is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications which may create a risk of personal injury. If you use this software in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of this software. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software in dangerous applications.

This software and documentation may provide access to or information on content, products and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third party content, products and services. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third party content, products or services.

Contents

Cartridge Overview	1
Cartridge content	2
Prerequisites	2
About this Guide	2
Services, features, and options	3
Hardware and software requirements	3
Network element (NE) interface	4
ASAP version	4
Connecting to the NE	4
Related Documentation	4
Installing and Testing the Cartridge	5
Downloading the cartridge	5
Starting ASAP	6
Installing the cartridge	7
Uninstalling the cartridge	7
Testing the cartridge installation	8
Configuring loopback and live mode parameters	8
Testing the installation	10
Atomic Service Description Layer (ASDL) Commands	11
Cross-connect services	13
A_ERIC-INS_2-1_ACTION_MO	13
A_ERIC-INS_2-1_CANCELGET_MO	14
A_ERIC-INS_2-1_CREATE_MO	14
A_ERIC-INS_2-1_CREATE_MO-RB	15
A_ERIC-INS_2-1_DELETE_MO	16
A_ERIC-INS_2-1_GETNEXT_MO	17
A_ERIC-INS_2-1_GET_MO	18
A_ERIC-INS_2-1_GET_MO-WITH-FILTER	19
A_ERIC-INS_2-1_SET_MO	20
User exit types	20
Understanding user exit type XML files	21
userExitType.xml	22
Service Definition	33
Common Service Description Layer (CSDL) commands	35
C_ERIC-INS_2-1_ACTION_MO	35
C_ERIC-INS_2-1_CANCELGET_MO	36
C_ERIC-INS_2-1_CREATE_MO	36
C_ERIC-INS_2-1_DELETE_MO	37
C_ERIC-INS_2-1_GETNEXT_MO	38
C_ERIC-INS_2-1_GET_MO	38
C_ERIC-INS_2-1_GET_MO-WITH-FILTER	39
C_ERIC-INS_2-1_SET_MO	40
Configuring ASAP to Support Additional NE Instances	41

Extracting source files	43
Loading a new XML file	44

Cartridge Overview

ASAP cartridges are discrete software components that are developed for the ASAP product. An ASAP cartridge offers specific domain behavior on top of the core ASAP software, and provides the configuration that supports a set of services on a network element (NE).

An ASAP cartridge is not a stand-alone component, but operates in conjunction with the ASAP core product. ASAP cartridges offer the following benefits:

- ◆ **Reduced Time to Market**—time to market of new services is reduced through simplified development, implementation, and extension of cartridges on customer sites.
- ◆ **Extendable**—cartridges can be extended to include additional services and components that deliver business value, without requiring changes to the original cartridge.
- ◆ **Simplified Effort**—the effort and technical knowledge that is required to perform customizations is reduced.
- ◆ **Ease of Installation**—cartridges can be installed into an ASAP environment without interfering with the existing install base.

An ASAP cartridge can be used to configure ASAP to provision the following:

- ◆ NEs from a specific vendor, such as Nortel or Lucent.
- ◆ Technologies, such as Asynchronous Transfer Mode (ATM) and Frame Relay switches, or Internet Protocol (IP) routers.
- ◆ Services that are supported on the NE, such as ATM, IP Virtual Private Networks (VPN), Wireless, or Optical.



Cartridges are designed for a specific technology, software load, and service.

An ASAP cartridge supports a particular set of services on an NE. These services are independent of customer-specific service definitions. Extensions to the cartridge can be performed by Professional Services or systems integrators to support customer-specific requirements.

For more information on extending a cartridge, refer to the *ASAP Cartridge Development Guide for Service Activation*.

Cartridge content

An ASAP cartridge contains the following:

- ◆ An interface to the NE.
- ◆ A set of scripts, such as State Tables or Java methods.
- ◆ A set of atomic actions in the form of Atomic Service Description Layer (ASDL) commands.
- ◆ A set of Common Service Description Layer (CSDL) commands that form meaningful services.
- ◆ Sample work orders.
- ◆ Installation scripts.

Prerequisites

This guide is a cartridge reference for use by anyone who is responsible for the adaptation and integration of ASAP-based solutions (for example, managers, designers, programmers, and testers). It assumes that readers possess the following skills:

- ◆ A knowledge of ASAP programming concepts
- ◆ A good working knowledge of the UNIX operating system
- ◆ A thorough understanding of service and network provisioning
- ◆ Familiarity with telecommunications

About this Guide

This guide provides a detailed description of the Ericsson INS 2.1 cartridge. It contains overview and technical information to assist in extending and integrating the cartridge into a customer environment.

This guide includes references to ASAP as it pertains to this cartridge. This guide is not a complete ASAP reference guide.

For additional information when using this cartridge, refer to the following supporting documentation:

- ◆ **ASAP documentation set**—for detailed information on the ASAP core product.
- ◆ **ASAP Cartridge Development Guide for Service Activation**—for information on how to extend a cartridge.

The Ericsson INS 2.1 cartridge provides the ASAP service configuration and network element (NE) interface to activate the Managed object services on Intelligent Network Server (INS) NEs.

Services, features, and options

Services, features, and options:

Table 1: Supported services

Service	Description
CREATE	This service creates a uniquely identified Management Object (MO), as specified in the attributes. Existing attributes default value will be created with the value, if it is specified. Other optional attributes which are not specified, will be left undefined.
SET	This service modifies the attributes specified for a uniquely identified MO.
GET	This service retrieves data for a uniquely identified MO.
GET [with FILTER(s)]	This service GET with a filter expression makes it possible to retrieve several instances in one request.
GETNEXT	This service retrieves the next object, whenever matches with the filter expression in a GET [with FILTER(s)] operation.
CANCELGET	This service cancels the fetching of multiple responses and can be applied on GET requests after having retrieved the first response.
DELETE	This service deletes a uniquely identified MO.
ACTION	This service performs an action on a uniquely identified MO.

Hardware and software requirements

The following sections contain the high-level software and hardware environment requirements to activate Managed object services using this cartridge, including:

- ◆ Network element (NE) interface
- ◆ ASAP version
- ◆ Operating environment

Network element (NE) interface

This cartridge is designed to operate with the following:

- ◆ Connection Protocol = telnet
- ◆ Vendor = ERIC
- ◆ Technology = INS
- ◆ Software Load = 2-1

The following settings are used for development and unit testing purpose only. User may require fine-tuning some or all of these settings during production phase.

- ◆ Maximum number of connections = 5
- ◆ Maximum number of devices in resource pool = 5
- ◆ Drop timeout = 1
- ◆ Spawn threshold = 3
- ◆ Kill threshold = 1

A sample Network Element is included in this cartridge for testing purpose:

- ◆ Sample NE name = INS1

ASAP version

ASAP requirements are ASAP 4.7.1 eFix5 or above, Sun Solaris 2.9 and Oracle 9i.

For more information on the operating environment of this ASAP version, refer to the ASAP 4.7.1 Release Record.

Connecting to the NE

This cartridge communicates with telnet protocol.

Related Documentation

- ◆ Ericsson INS 2.1 Interface Description Based on CAI.pdf
- ◆ CustomNet Managed Objects API.doc
- ◆ Ericsson INS 2.1 SDK CAI Driver User's Guide.pdf

Installing and Testing the Cartridge

This chapter describes the following procedures related to installing and testing the cartridge:

- ◆ [Downloading the cartridge](#)
- ◆ [Installing the cartridge](#)
- ◆ [Uninstalling the cartridge](#)
- ◆ [Testing the cartridge installation](#)

Downloading the cartridge

Before you can install the cartridge, you must use the internet to download the cartridge's TAR file from Oracle's Customer Portal.

Use the following instructions to download, then unTAR the TAR file.

To download the TAR file

1. Login to Oracle MetaLink internet home page (<http://www.metalink.oracle.com>).
2. Download the cartridge patch to your workstation.

To unTAR the TAR file

1. On your workstation, create a repository directory—the naming of which is your choice.

```
mkdir <repository_dir>
```

2. Untar EricssonIN_R1_1_0.b<#>.tar.

```
tar xvf EricssonIN_R1_1_0.b<#>.tar
```

3. Copy the resulting /ERIC_INS_2-1 directory and its contents to the repository directory.

```
cp -rf /ERIC_INS_2-1 <repository_dir>
```

The directory structure in the repository directory should look like the following illustration. (this illustration describes the minimum required structure; you can enhance this directory structure with additional directories based on your requirements and deliverables).

```
<repository_directory>
ERIC_INS_2-1
  /README
  /installCartridge
  /uninstallCartridge
```

```
/eric_ins_2_1.sar
```

Starting ASAP

Before installing the cartridge, ensure that ASAP is running.

To start ASAP

1. To start ASAP, execute the following script:

```
start_asap_sys
```

2. Ensure the ASAP Daemon (DAM_\$ENV_ID) is running by checking the ASAP status using the ASAP script “status”.
3. Check whether the WebLogic instance for this ASAP environment is running. If not, start the WebLogic instance.

The *ASAP Administration Guide* contains more information on starting ASAP, the ASAP Daemon, and WebLogic.

Installing the cartridge

Run the installation script `installCartridge` to install the cartridge. You will find this script under `/ERIC_INS_2-1`. The script executes the following tasks:

- ◆ Configures the Ericsson INS 2.1-specific NE using the SACT.
- ◆ Deploys the Ericsson INS 2.1 cartridge service model (only if the Ericsson INS 2.1 service model is not yet deployed) using the Service Activation Deployment Tool (SADT).
- ◆ Copies the Ericsson INS 2.1-specific jar files and the cpp library file to the ASAP environment.
- ◆ Loads the sample work orders to the SRP database.

For information on the SACT and the SADT, refer to the *ASAP Administration Guide*.

To install the cartridge

1. Run the `installCartridge` script from `/ERIC_INS_2-1`. At the prompt, type:

```
installCartridge eric_ins_2_1.sar
```

2. The script prompts you for the values of the following WebLogic login parameters:

- ◆ WebLogic Hostname
- ◆ WebLogic HTTP Port
- ◆ WebLogic Login User ID
- ◆ WebLogic Login Password

The script loads the NEP-NE configuration and the CSDL-ASDL configuration to the SARM database, and loads sample work orders to the SRP database. The script also copies the cartridge-specific jar files and cpp library file to the ASAP environment.

3. Restart ASAP to upload the cartridge configuration into ASAP.

Uninstalling the cartridge

Run the uninstallation script `uninstallCartridge` to uninstall the Ericsson INS 2.1 cartridge. This script is located under `ERIC_INS_2-1`. The script executes the following tasks:

- ◆ Unconfigures Ericsson INS 2.1-specific NEs using the SACT.
- ◆ Undeploys the Ericsson INS 2.1 cartridge service model (only if the Ericsson INS 2.1 service model is already deployed) using the Service Activation Deployment Tool (SADT).
- ◆ Removes the Ericsson INS 2.1-specific jar files and cpp library file from the ASAP environment.

For more information on the SACT and the SADT, refer to the *ASAP Administration Guide*.

To uninstall the cartridge

1. Run the `uninstallCartridge` script from `/ERIC_INS_2-1`. At the prompt, type

```
uninstallCartridge eric_ins_2_1.<timestamp>.sar
```

2. The script prompts you for the values of the following parameters:

- ◆ WebLogic Hostname
- ◆ WebLogic HTTP Port
- ◆ WebLogic Login User ID
- ◆ WebLogic Login Password

The script unloads the NEP-NE configuration and CSDL-ASDL configuration from SARM database. It also removes the cartridge specific jar files and cpp library file from the ASAP environment.

Testing the cartridge installation

To test this cartridge installation, you need to know about the network element (NE), services, and basic ASAP configuration. You may need to perform adjustments to provision a service for a specific NE, network, or connectivity configuration.

You can test the cartridge installation using one of the following methods:

- ◆ **Loopback mode**—does not actually connect to or send commands to the NE.
- ◆ **Live mode**—connects to and sends commands to a live NE.

Configuring loopback and live mode parameters

Set the following variables to test the cartridge in loopback or live testing modes.

Loopback mode

Set the following parameter to test the cartridge in loopback mode.

Table 2: Loopback Mode Parameter Settings

Configuration Variable	Parameter Settings	Location
LOOPBACK_ON	1 - loopback on; 0 - loopback off	ASAP.cfg

Live mode

Set the following parameter to test the cartridge in live mode.

Communication parameters

The communication parameters can be defined in Studio 2.0 and load to SARM database table `tbl_comm_param`. These parameters are being used to connection class connect method for connection to the device. The table consists of the following fields:

- ◆ Label - the label of the communication parameter.
- ◆ Value - the value of the communication parameter.
- ◆ Description - a thorough and accurate description of the communication parameter

Table 3: - Communication Parameters

Label	Value	Description
HOST_NAME		Machine name for the host NE.
HOST_IPADDR		Network IP address for the host NE.
PORT	3300	Port for the host NE.
HOST_USERID	tester1	Login user ID.
HOST_PASSWORD	sfdh312	Login password
OPEN_TIMEOUT	10	Connection establishment timeout (in seconds).
READ_TIMEOUT	10	Timeout for the telnet read functions (in seconds).
PROMPT	Enter command:	Provision prompt expected in the telnet session.
PROVIDER_NAME		Provider name within DN.
APPLICATION_NAME	JasManObj	Application name within DN.
NODE_NAME	JAMBALA	Node name within DN.

Testing the installation

The following procedure describes the steps required to test the cartridge installation in loopback mode. We recommend that you perform the initial cartridge installation test in loopback mode.

To test in loopback mode

1. Stop ASAP by typing the following command at the UNIX prompt:

```
stop_asap_sys
```

2. Ensure loop back mode is on. See [“Loopback mode” on page 8](#) for a description of how to set the loop back parameter to “On”.
3. Start ASAP by typing:

```
start_asap_sys
```

4. Send the sample work orders through the SRP Emulator by typing:

```
run_suite $SRP <ctrl_password> <suite name>
```

You can locate the suite names in /ERIC_INS_2-1/sample_wo by typing:

```
grep SUITE * | grep -v END
```

A list of all available suites appears.

For more information on the SRP Emulator, refer to the *ASAP Administration Guide*.

5. Verify the status of the sample work orders by typing:

```
asap_utils l
```

All successful work orders return the 104 state.

To view the sample work orders provided with this cartridge, refer to the Ericsson INS 2.1 cartridge source.

Atomic Service Description Layer (ASDL) Commands

ASDL commands represent a set of atomic actions that ASAP can perform on a network element (NE). ASAP can combine ASDLs to create meaningful services (CSDLs) within a cartridge.

This chapter presents detailed information on the ASDL parameters that we provide with this cartridge. The following table lists and describes the type of parameter information that is included.

Table 4: ASDL parameter information

Item	Description
Parameter Name	Identifies the parameter that is configured for the stated service.
Description	Describes the parameter.
Range	Describes or lists the range of values that can be used to satisfy this parameter.
Default Value	Configures a default value for the parameter so that it is not mandatory for the upstream system to provide a value.

Table 4: ASDL parameter information

Item	Description
Type	<p>Indicates one of the following parameter types:</p> <ul style="list-style-type: none"> ◆ S—Scalar, specifies the parameter label transmitted on the ASDL command. Scalar parameters are conventional name-value pair parameters. ◆ C—Compound, specifies the base name of the compound parameter transmitted on the ASDL command. A compound parameter contains structures or arrays of information that are represented by a particular structure name or compound parameter name. Each compound parameter can contain a large number of elements. If you use compound parameters, you only require a single entry in the ASAP translation tables to call the compound parameter and all its associated parameter elements. ◆ I—Indexed, identifies a parameter that contains a sequential numerical index value to tell the SARM that it should execute the same operation (for example, an ASDL command) for all occurrences of that index. Consequently, if there are several options on a particular CSDL command (OPT1, OPT2, OPT3, etc.), you can specify the OPT parameter as an indexed parameter. When you specify the OPT parameter as an indexed parameter, the SARM generates several occurrences of that same ASDL command and each command has a different value for the option being transmitted to the NEP. <p>For more information on parameter types, refer to the <i>ASAP Developer's Reference</i>.</p>
Class	<p>Indicates one of the following parameter classifications:</p> <ul style="list-style-type: none"> ◆ R—Required scalar parameter ◆ O—Optional scalar parameter ◆ C—Required compound parameter ◆ N—Optional compound parameter ◆ M—Mandatory indexed parameter ◆ I—Optional indexed parameter ◆ S—Parameter count

For a detailed description of the Required and Optional parameter classifications, refer to the *ASAP Administration Guide*.

Cross-connect services

This cartridge provides the following ASDL commands:

- ◆ A_ERIC-INS_2-1_ACTION_MO
- ◆ A_ERIC-INS_2-1_CANCELGET_MO
- ◆ A_ERIC-INS_2-1_CREATE_MO
- ◆ A_ERIC-INS_2-1_CREATE_MO-RB
- ◆ A_ERIC-INS_2-1_DELETE_MO
- ◆ A_ERIC-INS_2-1_DELETE_MO-RB
- ◆ A_ERIC-INS_2-1_GETNEXT_MO
- ◆ A_ERIC-INS_2-1_GET_MO
- ◆ A_ERIC-INS_2-1_GET_MO-WITH-FILTER
- ◆ A_ERIC-INS_2-1_SET_MO

A_ERIC-INS_2-1_ACTION_MO

Performs an action on a uniquely identified MO. It is implemented by the Java method `com.metasolv.cartridge.oss.eric_ins_2_1.prov.EricINSProvisioning.actionMO`.

Table 5: A_ERIC-INS_2-1_ACTION_MO

Parameter Name	Description	Range	Default Value	Type	Class
MCLI	Logical NE name.			S	R
GEN_DN_PARAMS	DN related parameters.			C	O
GEN_PARAMS_LIST	CAI command attribute list.			C	O
ACTION	Action type.			S	O

MML Command

```
ACTION:MO_JAS:DN, "<DN>":ACTIONTYPE, <ACTION_TYPE>: <ATTRIBUTES>;
```

Where:

- ◆ <DN>
- ◆ <ACTION_TYPE>
- ◆ <ATTRIBUTES>

are defined within the methods.

Output Parameters

If a network action is not successful, return ERIC_INS_ACTION_ERR_CODE as CSDL parameter to the SARM table TBL_SRQ_PARM.

A_ERIC-INS_2-1_CANCELGET_MO

Cancels the fetching of multiple responses. Can only be applied on GET requests and after having retrieved the first response. It is implemented by the Java method `com.metasolv.cartridge.oss.eric_ins_2_1.prov.EricINSProvisioning.cancelGetMO`.

Table 6: A_ERIC-INS_2-1_CANCELGET_MO

Parameter Name	Description	Range	Default Value	Type	Class
MCLI	Logical NE name.			S	R

MML Command

```
CANCELGET:MO_JAS;
```

Output Parameters

If a network action is not successful, return ERIC_INS_GET_CANCEL_ERR_CODE as CSDL parameter to the SARM table TBL_SRQ_PARM.

A_ERIC-INS_2-1_CREATE_MO

Creates a uniquely identified Management Object (MO), using the attributes as specified. It is implemented by the Java method `com.metasolv.cartridge.oss.eric_ins_2_1.prov.EricINSProvisioning.createMO`.

Table 7: A_ERIC-INS_2-1_CREATE_MO

Parameter Name	Description	Range	Default Value	Type	Class
MCLI	Logical NE name.			S	R
GEN_DN_PARAMS	DN related parameters.			C	O
GEN_PARAMS_LIST	CAI command attribute list.			C	O

MML Command

```
CREATE:MO_JAS:DN,"<DN>":<ATTRIBUTES>;
```

Where

- ◆ <DN> and
- ◆ <ATTRIBUTES>

are defined within the methods.

Output Parameters

If a network action is not successful, return ERIC_INS_CREATE_ERR_CODE as CSDL parameter to the SARM table TBL_SRQ_PARM.

A_ERIC-INS_2-1_CREATE_MO-RB

Rollback ASDL to re-create a uniquely identified Management Object (MO), using the attributes as specified. It is implemented by the Java method `com.metasolv.cartridge.oss.eric_ins_2_1.prov.EricINSProvisioning.rollbackCreateMO`.

Table 8: A_ERIC-INS_2-1_CREATE_MO-RB

Parameter Name	Description	Range	Default Value	Type	Class
MCLI	Logical NE name.			S	R
GEN_DN_PARAMS	DN related parameters.			C	O
OLD_PARAMS_LIST	CAI command attribute list used for rollback only.			C	O

MML Command

```
CREATE:MO_JAS:DN,"<DN>":<ATTRIBUTES>;
```

Where:

- ◆ <DN> and
- ◆ <ATTRIBUTES>

are defined within the methods.

Output Parameters

In case of rollback failure, return ERIC_INS_RB_ERR_INFO information parameter for the error message.

A_ERIC-INS_2-1_DELETE_MO

Deletes a uniquely identified MO. It is implemented by the Java method `com.metasolv.cartridge.oss.eric_ins_2_1.prov.EricINSProvisioning.deleteMO`.

Table 9: A_ERIC-INS_2-1_DELETE_MO

Parameter Name	Description	Range	Default Value	Type	Class
MCLI	Logical NE name.			S	R
GEN_DN_PARAMS	DN related parameters.			C	O
GEN_PARAMS_LIST	CAI command attribute list: <ul style="list-style-type: none"> ◆ This parameter is used by the GET command to get the current attributes before deleting the specific DN. ◆ The current attributes are stored in ASAP as rollback parameters, after the DELETE command successfully executed. ◆ If this parameter is not used, GET All attributes are assumed. 			C	O

MML Command

```
DELETE:MO_JAS:DN,"<DN>" ;
```

Where:

- ◆ <DN>

as defined in the method.

Output Parameters

If a network action is not successful, return ERIC_INS_DELETE_ERR_CODE as CSDL parameter to the SARM table TBL_SRQ_PARM.

A_ERIC-INS_2-1_GETNEXT_MO

Receives the next object, which was matching the filter expression in a GETNEXT operation. It is implemented by the Java method `com.metasolv.cartridge.oss.eric_ins_2_1.prov.EricINSProvisioning.getNextMO`.

Table 10: A_ERIC-INS_2-1_GETNEXT_MO

Parameter Name	Description	Range	Default Value	Type	Class
MCLI	Logical NE name.			S	R

MML Command

```
GETNEXT:MO_JAS;
```

Output Parameters

Returns a compound parameter OLD_PARMS_LIST following the same format as GEN_PARMS_LIST.

If a network action is not successful, return ERIC_INS_GETNEXT_ERR_CODE as CSDL parameter to the SARM table TBL_SRQ_PARM.

A_ERIC-INS_2-1_GET_MO

Retrieves data for a uniquely identified MO. It is implemented by the Java method `com.metasolv.cartridge.oss.eric_ins_2_1.prov.EricINSProvisioning.getMO`.

Table 11: A_ERIC-INS_2-1_GET_MO

Parameter Name	Description	Range	Default Value	Type	Class
MCLI	Logical NE name.			S	R
GEN_DN_PARAMS	DN related parameters.			C	O
GEN_PARAMS_LIST	CAI command attribute list.			C	O

MML Command

```
GET:MO_JAS:DN, "<DN>":<ATTRIBUTES>;
```

Where:

- ◆ <DN> and
- ◆ <ATTRIBUTES>

are defined within the methods.

Output Parameters

Returns a compound parameter OLD_PARAMS_LIST following the same format as GEN_PARAMS_LIST.

If a network action is not successful, return ERIC_INS_GET_ERR_CODE as CSDL parameter to the SARM table TBL_SRQ_PARM.

A_ERIC-INS_2-1_GET_MO-WITH-FILTER

GETs with a filter expression makes it possible to retrieve several instances in one request. It is implemented by the Java method `com.metasolv.cartridge.oss.eric_ins_2_1.prov.EricINSProvisioning.getFilterMO`.

Table 12: A_ERIC-INS_2-1_GET_MO-WITH-FILTER

Parameter Name	Description	Range	Default Value	Type	Class
MCLI	Logical NE name.			S	R
GEN_DN_PARMs	DN related parameters.			C	O
GEN_PARMs_LIST	CAI command attribute list.			C	O
SCOPE_LEVEL	Scope of get request.	0=base, 1=single level, 2=whole subtree		S	O
N_ITEMS	Number of items the search can be retrieved.			S	O
TIME_LIMIT	Number of seconds allowed for the search.			S	O

MML Command

```
GET:MO_JAS:DN, "<DN>":<ATTRIBUTES>;
```

Where:

- ◆ <DN> and
- ◆ <ATTRIBUTES>

are defined within the methods.

Output Parameters

Returns a compound parameter OLD_PARMs_LIST following the same format as GEN_PARMs_LIST.

If a network action is not successful, return ERIC_INS_GET_ERR_CODE as CSDL parameter to the SARM table TBL_SRQ_PARM.

A_ERIC-INS_2-1_SET_MO

Modifies the attributes specified for a uniquely identified MO. It is implemented by the Java method `com.metasolv.cartridge.oss.eric_ins_2_1.prov.EricINSProvisioning.setMO`.

Table 13: A_ERIC-INS_2-1_SET_MO

Parameter Name	Description	Range	Default Value	Type	Class
MCLI	Logical NE name.			S	R
GEN_DN_PARMs	DN related parameters.			C	O
GEN_PARMs_LIST	CAI command attribute list.			C	O

MML Command

```
SET:MO_JAS:DN, "<DN>":<ATTRIBUTES>;
```

Where:

- ◆ <DN> and
- ◆ <ATTRIBUTES>

are defined within the methods.

Output Parameters

If the network action is not successful, return `ERIC_INS_SET_ERR_CODE` as CSDL parameter to the SARM table `TBL_SRQ_PARM`.

User exit types

User exit types allow cartridge developers and systems administrators to map ASDL exit codes to one of the predefined base exit types. Base exit types determine the product behavior. Cartridges map return codes and status values from a network element to a user defined exit type.

Regular expressions (regex) are used to perform pattern searches on responses from network elements. The pattern is stored in "tbl_user_err" in the SARM database. The user exit type contains a regex pattern that is applied at runtime.

Regular expressions enable users to associate a series of responses to a specific base type. For example, a regular expression "6." can identify a pattern where any response with the character "6" followed by any number of characters will translate to base type of FAIL.

Regular expressions can also allow very specific searches within a response from a network element. Regular expressions are typically compiled before being executed. Compilation produces a binary version of the expression and ensures that the syntax of the regular expression is correct. This compilation occurs using SACT\SADT when user exit types are deployed into ASAP. If the syntax is deemed to be incorrect during compilation, SADT displays an error message and the deployment of the user exit type will fail.

For more information on pattern matching, refer to the *ASAP Developer Reference* and the *ASAP Administration Guide*.

Understanding user exit type XML files

```
<userDefinedExitType>
  <neDescriptor>
    <softwareLoad>DYNAMIC_SL</softwareLoad>
    <technology>DYNAMIC_VENDOR-DYNAMIC_TECH</technology>
  </neDescriptor>
  <searchPattern>SUCCESS.</searchPattern>1
  <userType>U_SUCCEED</userType>2
  <baseType>SUCCEED</baseType>3
  <description>The ASDL provisioning was successful</description>
</userDefinedExitType>
<userDefinedExitType>
  <searchPattern>90.</searchPattern>
  <userType>U_FAIL</userType>
  <baseType>FAIL</baseType>
  <description>The ASDL failed - fail the current order
    and stop processing.</description>
</userDefinedExitType>
<userDefinedExitType>
  <searchPattern>101-110[201-215]</searchPattern>4
  <userType>U_SOFT_FAIL</userType>
  <baseType>SOFT_FAIL</baseType>
  <description>The ASDL has encountered a soft failure. Processing will
    continue.</description>
</userDefinedExitType>
<userDefinedExitType>
  <searchPattern>801-850</searchPattern>5
  <userType>U_MINOR_ERROR</userType>
  <baseType>SOFT_FAIL</baseType>
```

1. Pattern searches accommodate situations in which responses from the device contain small variants that represent the same meaning. The user type contains an associated search pattern that is applied at runtime. Using regular expressions, you can default a series of responses. For example a regular expression "90." can specify a pattern where any response with the character "90" followed by any number of characters will translate to base type of FAIL.
2. The user type that the search pattern maps to.
3. The base type that maps to the user type.
4. 101 to 110 and 201 to 215 will translate to a base type of SOFT_FAIL

```

        <description>The ASDL has encountered a soft failure. Processing will
            continue.</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <userType>U_DELAYED_FAIL</userType>
        <baseType>DELAYED_FAIL</baseType>
        <description>The ASDL has failed during provisioning.</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>BCS36</softwareLoad>
            <technology>NORTEL_DMS</technology>
            <neVendor>Nortel</neVendor>
        </neDescriptor>
        <searchPattern>*.</searchPattern>
        <userType>U_MAINTAIN</userType>
        <baseType>MAINTENANCE</baseType>
        <description>The ASDL will Wait until the NE comes out of
            Maintenance Mode</description>
    </userDefinedExitType>

```

The previous code sample shows some typical search pattern examples. Some additional examples follow:

- ◆ `^\.*\b(one|two|three)\b.*$` = matches a complete line of text that contains any of the words "one", "two" or "three"
- ◆ `^(?=.*?\bone\b)(?=.*?\btwo\b)(?=.*?\bthree\b).*$` matches a complete line of text that contains all of the words "one", "two" and "three"
- ◆ `"[^\r\n]*"` matches a single-line string that does not allow the quote character to appear inside the string.
- ◆ `\b\d{1,3}\.\d{1,3}\.\d{1,3}\.\d{1,3}\b` matches any IP address.

For more information on search patterns, refer to <http://java.sun.com/j2se/1.4.2/docs/api/java/util/regex/Pattern.html>.

For more information on user exit types, refer to chapter 3 of the *ASAP Developer Reference*.

userExitType.xml

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- -Sample XML file generated by XML Spy v4.3 U (http://www.xmlspy.com) by
Nortel Networks (Nortel Networks) -->
<serviceModel xmlns="http://www.metasolv.com/ServiceActivation/2003/
ServiceModel" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.metasolv.com/ServiceActivation/2003/
ServiceModel
C:\OGIRLAN_DATA\CART_SOURCE\Asap71Schemas\xsd\ServiceModel.xsd">

```

5. 801-850 will translate to a base type of SOFT_FAIL. Note that the user type differs from the previous range.

```
<userDefinedExitType>
  <neDescriptor>
    <softwareLoad>2-1</softwareLoad>
    <technology>INS</technology>
    <neVendor>ERIC</neVendor>
  </neDescriptor>
  <searchPattern>NO_MATCH</searchPattern>
  <userType>NO_MATCH</userType>
  <baseType>FAIL</baseType>
  <description>No match found</description>
</userDefinedExitType>
<userDefinedExitType>
  <neDescriptor>
    <softwareLoad>2-1</softwareLoad>
    <technology>INS</technology>
    <neVendor>ERIC</neVendor>
  </neDescriptor>
  <searchPattern>PROV_CART_EXCEPTION</searchPattern>
  <userType>PROV_CART_EXCEPTION</userType>
  <baseType>FAIL</baseType>
  <description>ProvCartridgeException caught</description>
</userDefinedExitType>
<userDefinedExitType>
  <neDescriptor>
    <softwareLoad>2-1</softwareLoad>
    <technology>INS</technology>
    <neVendor>ERIC</neVendor>
  </neDescriptor>
  <searchPattern>VALIDATE_EXCEPTION</searchPattern>
  <userType>VALIDATE_EXCEPTION</userType>
  <baseType>FAIL</baseType>
  <description>ValidateException caught</description>
</userDefinedExitType>
<userDefinedExitType>
  <neDescriptor>
    <softwareLoad>2-1</softwareLoad>
    <technology>INS</technology>
    <neVendor>ERIC</neVendor>
  </neDescriptor>
  <searchPattern>0</searchPattern>
  <userType>CAI_SUCCEED</userType>
  <baseType>SUCCEED</baseType>
  <description>CAI command succeed</description>
</userDefinedExitType>
<userDefinedExitType>
  <neDescriptor>
    <softwareLoad>2-1</softwareLoad>
    <technology>INS</technology>
    <neVendor>ERIC</neVendor>
  </neDescriptor>
  <searchPattern>2001</searchPattern>
  <userType>DATABASE_ERROR</userType>
```

```

        <baseType>FAIL</baseType>
        <description>If the database is down or does not respond in actual CAI
node.</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>2-1</softwareLoad>
            <technology>INS</technology>
            <neVendor>ERIC</neVendor>
        </neDescriptor>
        <searchPattern>2002</searchPattern>
        <userType>INTERNAL_APP_ERROR</userType>
        <baseType>FAIL</baseType>
        <description>Fatal application error, due to bad configuration or
misuse of the system.</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>2-1</softwareLoad>
            <technology>INS</technology>
            <neVendor>ERIC</neVendor>
        </neDescriptor>
        <searchPattern>3001</searchPattern>
        <userType>INVALID_COMMAND</userType>
        <baseType>FAIL</baseType>
        <description>Syntax error</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>2-1</softwareLoad>
            <technology>INS</technology>
            <neVendor>ERIC</neVendor>
        </neDescriptor>
        <searchPattern>3002</searchPattern>
        <userType>OP_NOT_SUPPORTED</userType>
        <baseType>FAIL</baseType>
        <description>An operation that is not supported by the actual MO.</
description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>2-1</softwareLoad>
            <technology>INS</technology>
            <neVendor>ERIC</neVendor>
        </neDescriptor>
        <searchPattern>3006</searchPattern>
        <userType>INVALID_PASSWORD</userType>
        <baseType>FAIL</baseType>
        <description>Invalid password.</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>

```

```

        <softwareLoad>2-1</softwareLoad>
        <technology>INS</technology>
        <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>3007</searchPattern>
    <userType>NOT_LOGGED_IN</userType>
    <baseType>FAIL</baseType>
    <description>Attempt to send MO request before logged into the CAI
node.</description>
</userDefinedExitType>
<userDefinedExitType>
    <neDescriptor>
        <softwareLoad>2-1</softwareLoad>
        <technology>INS</technology>
        <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>3008</searchPattern>
    <userType>INVALID_CMD_SEQ</userType>
    <baseType>FAIL</baseType>
    <description>Invalid CAI command sequence.</description>
</userDefinedExitType>
<userDefinedExitType>
    <neDescriptor>
        <softwareLoad>2-1</softwareLoad>
        <technology>INS</technology>
        <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>9331</searchPattern>
    <userType>ALIAS_DEREF_PROBLEM</userType>
    <baseType>FAIL</baseType>
    <description>An alias was encountered in a situation where it was not
allowed or where access was denied.</description>
</userDefinedExitType>
<userDefinedExitType>
    <neDescriptor>
        <softwareLoad>2-1</softwareLoad>
        <technology>INS</technology>
        <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>9332</searchPattern>
    <userType>ALIAS_PROBLEM</userType>
    <baseType>FAIL</baseType>
    <description>An alias has been dereferenced which names no object.</
description>
</userDefinedExitType>
<userDefinedExitType>
    <neDescriptor>
        <softwareLoad>2-1</softwareLoad>
        <technology>INS</technology>
        <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>9333</searchPattern>

```

```

        <userType>ALREADY_EXIST</userType>
        <baseType>FAIL</baseType>
        <description>An attempted addEntry or modifyDN operation names an
entry that already exists.</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>2-1</softwareLoad>
            <technology>INS</technology>
            <neVendor>ERIC</neVendor>
        </neDescriptor>
        <searchPattern>9335</searchPattern>
        <userType>BUSY</userType>
        <baseType>FAIL</baseType>
        <description>The Directory, or some part of it, is presently too busy
to perform the requested operation, but may be able to do so after a short
while.</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>2-1</softwareLoad>
            <technology>INS</technology>
            <neVendor>ERIC</neVendor>
        </neDescriptor>
        <searchPattern>9336</searchPattern>
        <userType>COMPARE_FALSE</userType>
        <baseType>FAIL</baseType>
        <description>Compare False.</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>2-1</softwareLoad>
            <technology>INS</technology>
            <neVendor>ERIC</neVendor>
        </neDescriptor>
        <searchPattern>9337</searchPattern>
        <userType>COMPARE_TRUE</userType>
        <baseType>FAIL</baseType>
        <description>Compare True.</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>2-1</softwareLoad>
            <technology>INS</technology>
            <neVendor>ERIC</neVendor>
        </neDescriptor>
        <searchPattern>9338</searchPattern>
        <userType>CONSTRAINT_VIOLATION</userType>
        <baseType>FAIL</baseType>
        <description>An attribute value supplied in the argument of an
operation does not conform to the constraints imposed by ITU-T Rec. X.501,
ISO/IEC 9594-2 or by the attribute definition.</description>

```

```

</userDefinedExitType>
<userDefinedExitType>
  <neDescriptor>
    <softwareLoad>2-1</softwareLoad>
    <technology>INS</technology>
    <neVendor>ERIC</neVendor>
  </neDescriptor>
  <searchPattern>9342</searchPattern>
  <userType>INAPPROPRIATE_AUTH</userType>
  <baseType>FAIL</baseType>
  <description>The level of security associated with the requestor's
credentials is inconsistent with the level of protection requested.</
description>
</userDefinedExitType>
<userDefinedExitType>
  <neDescriptor>
    <softwareLoad>2-1</softwareLoad>
    <technology>INS</technology>
    <neVendor>ERIC</neVendor>
  </neDescriptor>
  <searchPattern>9343</searchPattern>
  <userType>INAPPROPRIATE_MATCH</userType>
  <baseType>FAIL</baseType>
  <description>An attempt was made to use a matching rule not defined for
the attribute type concerned.</description>
</userDefinedExitType>
<userDefinedExitType>
  <neDescriptor>
    <softwareLoad>2-1</softwareLoad>
    <technology>INS</technology>
    <neVendor>ERIC</neVendor>
  </neDescriptor>
  <searchPattern>9344</searchPattern>
  <userType>INSUFFICIENT_ACCESS</userType>
  <baseType>FAIL</baseType>
  <description>The requestor does not have the right to carry out the
requested operation.</description>
</userDefinedExitType>
<userDefinedExitType>
  <neDescriptor>
    <softwareLoad>2-1</softwareLoad>
    <technology>INS</technology>
    <neVendor>ERIC</neVendor>
  </neDescriptor>
  <searchPattern>9346</searchPattern>
  <userType>INVALID_CREDENTIALS</userType>
  <baseType>FAIL</baseType>
  <description>The supplied credentials were invalid.</description>
</userDefinedExitType>
<userDefinedExitType>
  <neDescriptor>
    <softwareLoad>2-1</softwareLoad>

```

```

        <technology>INS</technology>
        <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>9347</searchPattern>
    <userType>INVALID_DN_SYNTAX</userType>
    <baseType>FAIL</baseType>
    <description>Invalid DN syntax.</description>
</userDefinedExitType>
<userDefinedExitType>
    <neDescriptor>
        <softwareLoad>2-1</softwareLoad>
        <technology>INS</technology>
        <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>9348</searchPattern>
    <userType>INVALID_SYNTAX</userType>
    <baseType>FAIL</baseType>
    <description>A purported attribute value, specified as an argument of
the operation, does not conform to the attribute syntax of the attribute
type.</description>
</userDefinedExitType>
<userDefinedExitType>
    <neDescriptor>
        <softwareLoad>2-1</softwareLoad>
        <technology>INS</technology>
        <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>9351</searchPattern>
    <userType>INTERNAL_LOOP</userType>
    <baseType>FAIL</baseType>
    <description>The Directory is unable to accomplish this request due to
an internal loop.</description>
</userDefinedExitType>
<userDefinedExitType>
    <neDescriptor>
        <softwareLoad>2-1</softwareLoad>
        <technology>INS</technology>
        <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>9352</searchPattern>
    <userType>NAMING_VIOLATION</userType>
    <baseType>FAIL</baseType>
    <description>The attempted addition or modification would violate the
structure rules of the DIT as defined in the Directory schema and ITU-T Rec.
X.501, ISO/IEC 9594-2.</description>
</userDefinedExitType>
<userDefinedExitType>
    <neDescriptor>
        <softwareLoad>2-1</softwareLoad>
        <technology>INS</technology>
        <neVendor>ERIC</neVendor>
    </neDescriptor>

```



```

    <searchPattern>9354</searchPattern>
    <userType>NO_OBJECT_MODS</userType>
    <baseType>FAIL</baseType>
    <description>An operation attempted to modify the structural object
class of an enary.</description>
  </userDefinedExitType>
  <userDefinedExitType>
    <neDescriptor>
      <softwareLoad>2-1</softwareLoad>
      <technology>INS</technology>
      <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>9355</searchPattern>
    <userType>NO_SUCH_ATT_OR_VAL</userType>
    <baseType>FAIL</baseType>
    <description>The named enary lacks one of the attributes or attribute
values specified as an argument of the operation.</description>
  </userDefinedExitType>
  <userDefinedExitType>
    <neDescriptor>
      <softwareLoad>2-1</softwareLoad>
      <technology>INS</technology>
      <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>9356</searchPattern>
    <userType>NO_SUCH_OBJECT</userType>
    <baseType>FAIL</baseType>
    <description>The name supplied does not match the name of any object.</
description>
  </userDefinedExitType>
  <userDefinedExitType>
    <neDescriptor>
      <softwareLoad>2-1</softwareLoad>
      <technology>INS</technology>
      <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>9357</searchPattern>
    <userType>NOT_ALLOW_NON_LEAF</userType>
    <baseType>FAIL</baseType>
    <description>The attempted operation is only allowed on leaf enaries
of the DIT.</description>
  </userDefinedExitType>
  <userDefinedExitType>
    <neDescriptor>
      <softwareLoad>2-1</softwareLoad>
      <technology>INS</technology>
      <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>9358</searchPattern>
    <userType>NOT_ALLOW_RDN</userType>
    <baseType>FAIL</baseType>

```

```

        <description>The attempted operation would affect the RDN (e.g.
removal of an attribute, which is a part of the RDN).</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>2-1</softwareLoad>
            <technology>INS</technology>
            <neVendor>ERIC</neVendor>
        </neDescriptor>
        <searchPattern>9359</searchPattern>
        <userType>OBJ_CLASS_VIOLATION</userType>
        <baseType>FAIL</baseType>
        <description>The attempted update would produce an enary inconsistent
with the rules for enary content.</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>2-1</softwareLoad>
            <technology>INS</technology>
            <neVendor>ERIC</neVendor>
        </neDescriptor>
        <searchPattern>9360</searchPattern>
        <userType>OPERATIONS_ERROR</userType>
        <baseType>FAIL</baseType>
        <description>Operations error.</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>2-1</softwareLoad>
            <technology>INS</technology>
            <neVendor>ERIC</neVendor>
        </neDescriptor>
        <searchPattern>9361</searchPattern>
        <userType>OTHER</userType>
        <baseType>FAIL</baseType>
        <description>Unknown error.</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>2-1</softwareLoad>
            <technology>INS</technology>
            <neVendor>ERIC</neVendor>
        </neDescriptor>
        <searchPattern>9364</searchPattern>
        <userType>PROTOCOL_ERROR</userType>
        <baseType>FAIL</baseType>
        <description>Invalid filter expression on search, or DN on add, modify
or delete.</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>2-1</softwareLoad>

```

```

        <technology>INS</technology>
        <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>9367</searchPattern>
    <userType>SIZE_LIMIT_EXCEEDED</userType>
    <baseType>FAIL</baseType>
    <description>Either the server or the client specified limit on number
of search results was exceeded.</description>
</userDefinedExitType>
<userDefinedExitType>
    <neDescriptor>
        <softwareLoad>2-1</softwareLoad>
        <technology>INS</technology>
        <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>9368</searchPattern>
    <userType>STRONG_AUTH_NOT_SUPP</userType>
    <baseType>FAIL</baseType>
    <description>The server does not support the requested authentication
method.</description>
</userDefinedExitType>
<userDefinedExitType>
    <neDescriptor>
        <softwareLoad>2-1</softwareLoad>
        <technology>INS</technology>
        <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>9369</searchPattern>
    <userType>STRONG_AUTH_REQUIRED</userType>
    <baseType>FAIL</baseType>
    <description>The server requires an authentication method stronger
than unencrypted user name and password.</description>
</userDefinedExitType>
<userDefinedExitType>
    <neDescriptor>
        <softwareLoad>2-1</softwareLoad>
        <technology>INS</technology>
        <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>9370</searchPattern>
    <userType>TIME_LIMIT_EXCEEDED</userType>
    <baseType>FAIL</baseType>
    <description>The Directory has reached the limit of time set by the
user in a service conarol.</description>
</userDefinedExitType>
<userDefinedExitType>
    <neDescriptor>
        <softwareLoad>2-1</softwareLoad>
        <technology>INS</technology>
        <neVendor>ERIC</neVendor>
    </neDescriptor>
    <searchPattern>9372</searchPattern>

```

```

        <userType>TYPE_OR_VALUE_EXISTS</userType>
        <baseType>FAIL</baseType>
        <description>An attempt was made to add an attribute that already
        existed in the enary, or a value that already existed in the attribute.</
        description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>2-1</softwareLoad>
            <technology>INS</technology>
            <neVendor>ERIC</neVendor>
        </neDescriptor>
        <searchPattern>9373</searchPattern>
        <userType>UNAVAILABLE</userType>
        <baseType>FAIL</baseType>
        <description>The Directory, or some part of it, is currently
        unavailable.</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>2-1</softwareLoad>
            <technology>INS</technology>
            <neVendor>ERIC</neVendor>
        </neDescriptor>
        <searchPattern>9374</searchPattern>
        <userType>UNDEFINED_TYPE</userType>
        <baseType>FAIL</baseType>
        <description>An undefined attribute type was provided as an argument
        to the operation. This error may occur only in relation to addEnary or
        ModifyEnary operations.</description>
    </userDefinedExitType>
    <userDefinedExitType>
        <neDescriptor>
            <softwareLoad>2-1</softwareLoad>
            <technology>INS</technology>
            <neVendor>ERIC</neVendor>
        </neDescriptor>
        <searchPattern>9375</searchPattern>
        <userType>UNWILLING_TO_PERFORM</userType>
        <baseType>FAIL</baseType>
        <description>The Directory, or some part of it, is not prepared to
        execute this request.</description>
    </userDefinedExitType>
</serviceModel>

```

Service Definition

The BroadWorksAS cartridge contains a set of CSDLs that map to one or more ASDL commands. You can also create additional CSDLs that map to existing and newly-created ASDLs. An upstream system can assemble any of these CSDL commands onto a work order for provisioning.

This chapter presents detailed information on the CSDL parameters that we provide in this cartridge. The following table lists and describes the type of parameter information that is included.

Table 14: ASDL parameter information

Item	Description
Parameter Name	Identifies the parameter that is configured for the stated service.
Description	Describes the parameter.
Range	Describes or lists the range of values that can be used to satisfy this parameter.
Default Value	Configures a default value for the parameter and it is not mandatory for the upstream system to provide a value.

Table 14: ASDL parameter information

Item	Description
Type	<p>Indicates one of the following parameter types:</p> <ul style="list-style-type: none"> ◆ S—Scalar, specifies the parameter label transmitted on the ASDL command. Scalar parameters are conventional name-value pair parameters. ◆ C—Compound, specifies the base name of the compound parameter transmitted on the ASDL command. A compound parameter contains structures or arrays of information that are represented by a particular structure name or compound parameter name. Each compound parameter can contain a large number of elements. If you use compound parameters, you only require a single entry in the ASAP translation tables to call the compound parameter and all its associated parameter elements. ◆ I—Indexed, identifies a parameter that contains a sequential numerical index value to tell the SARM that it should execute the same operation (for example, an ASDL command) for all occurrences of that index. Consequently, if there are several options on a particular CSDL command (OPT1, OPT2, OPT3, etc.), you can specify the OPT parameter as an indexed parameter. When you specify the OPT parameter as an indexed parameter, the SARM generates several occurrences of that same ASDL command and each command has a different value for the option being transmitted to the NEP. <p>For more information on parameter types, refer to the <i>ASAP Developer Reference</i>.</p>
Class	<p>Indicates one of the following parameter classifications:</p> <ul style="list-style-type: none"> ◆ R—Required scalar parameter ◆ O—Optional scalar parameter ◆ C—Required compound parameter ◆ N—Optional compound parameter ◆ M—Mandatory indexed parameter ◆ I—Optional indexed parameter ◆ S—Parameter count

For a detailed description of the Required and Optional parameter classifications, refer to the *ASAP Administration Guide*.

Common Service Description Layer (CSDL) commands

This cartridge provides the following CSDL commands:

- ◆ C_ERIC-INS_2-1_ACTION_MO
- ◆ C_ERIC-INS_2-1_CANCELGET_MO
- ◆ C_ERIC-INS_2-1_CREATE_MO
- ◆ C_ERIC-INS_2-1_DELETE_MO
- ◆ C_ERIC-INS_2-1_GETNEXT_MO
- ◆ C_ERIC-INS_2-1_GET_MO
- ◆ C_ERIC-INS_2-1_GET_MO-WITH-FILTER
- ◆ C_ERIC-INS_2-1_SET_MO

C_ERIC-INS_2-1_ACTION_MO

Performs an action on a uniquely identified MO.

Table 15: C_ERIC-INS_2-1_ACTION_MO

Parameter Name	Description	Range	Default Value	Type	Class
ACTION	Action type.			S	O
GEN_DN_PARMS	DN related parameters.			C	O
GEN_PARMS_LIST	CAI command attribute list.			C	O
NE_ID_ERIC-INS_2-1	Logical NE name.			S	R

Mapping to ASDLs

The following table illustrates the CSDL to ASDL mapping for this service.

Table 16: CSDL to ASDL Mapping

CSDLs	ASDLs
C_ERIC-INS_2-1_ACTION_MO	A_ERIC-INS_2-1_ACTION_MO

C_ERIC-INS_2-1_CANCELGET_MO

Cancels the fetching of multiple responses. Can only be applied on GET requests and after having retrieved the first response.

Table 17: C_ERIC-INS_2-1_CANCELGET_MO

Parameter Name	Description	Range	Default Value	Type	Class
NE_ID_ERIC-INS_2-1	Logical NE name.			S	R

Mapping to ASDLs

The following table illustrates the CSDL to ASDL mapping for this service.

Table 18: CSDL to ASDL Mapping

CSDLs	ASDLs
C_ERIC-INS_2-1_CANCELGET_MO	A_ERIC-INS_2-1_CANCELGET_MO

C_ERIC-INS_2-1_CREATE_MO

Creates a uniquely identified Management Object (MO), using the attributes as specified.

Table 19: C_BROADSOFT-BAS_11-0_ADD_DEVICES_SP

Parameter Name	Description	Range	Default Value	Type	Class
GEN_DN_PARAMS	DN related parameters.			C	O
GEN_PARAMS_LIST	CAI command attribute list.			C	O
NE_ID_ERIC-INS_2-1	Logical NE name.			S	R

Mapping to ASDLs

The following table illustrates the CSDL to ASDL mapping for this service.

Table 20: CSDL to ASDL Mapping

CSDLs	ASDLs
C_ERIC-INS_2-1_CREATE_MO	A_ERIC-INS_2-1_CREATE_MO

C_ERIC-INS_2-1_DELETE_MO

Deletes a uniquely identified MO.

Table 21: C_ERIC-INS_2-1_DELETE_MO

Parameter Name	Description	Range	Default Value	Type	Class
GEN_DN_PARMS	DN related parameters.			C	O
GEN_PARMS_LIST	CAI command attribute list: <ul style="list-style-type: none"> ◆ This parameter is used by the GET command to get the current attributes before deleting the specific DN. ◆ The current attributes are stored in ASAP as rollback parameters, after the DELETE command is successfully executed. ◆ If this parameter is not used, GET All attributes are assumed. 			C	O
NE_ID_ERIC-INS_2-1	Logical NE name.			S	R

Mapping to ASDLs

The following table illustrates the CSDL to ASDL mapping for this service.

Table 22: CSDL to ASDL Mapping

CSDLs	ASDLs
C_ERIC-INS_2-1_DELETE_MO	A_ERIC-INS_2-1_DELETE_MO

C_ERIC-INS_2-1_GETNEXT_MO

Receives the next object, which was matching the filter expression in a GETNEXT operation.

Table 23: C_ERIC-INS_2-1_GETNEXT_MO

Parameter Name	Description	Range	Default Value	Type	Class
NE_ID_ERIC-INS_2-1	Logical NE name.			S	R

Mapping to ASDLs

The following table illustrates the CSDL to ASDL mapping for this service.

Table 24: CSDL to ASDL Mapping

CSDLs	ASDLs
C_ERIC-INS_2-1_GETNEXT_MO	A_ERIC-INS_2-1_GETNEXT_MO

C_ERIC-INS_2-1_GET_MO

Retrieves data for a uniquely identified MO.

Table 25: C_ERIC-INS_2-1_GET_MO

Parameter Name	Description	Range	Default Value	Type	Class
GEN_DN_PARAMS	DN related parameters.			C	O
GEN_PARAMS_LIST	CAI command attribute list.			C	O
NE_ID_ERIC-INS_2-1	Logical NE name.			S	R

Mapping to ASDLs

The following table illustrates the CSDL to ASDL mapping for this service.

Table 26: CSDL to ASDL Mapping

CSDLs	ASDLs
C_ERIC-INS_2-1_GET_MO	A_ERIC-INS_2-1_GET_MO

C_ERIC-INS_2-1_GET_MO-WITH-FILTER

GET with a filter expression makes it possible to retrieve several instances in one request.

Table 27: C_ERIC-INS_2-1_GET_MO-WITH-FILTER

Parameter Name	Description	Range	Default Value	Type	Class
GEN_DN_PARMS	DN related parameters.			C	O
GEN_PARMS_LIST	CAI command attribute list.			C	O
NE_ID_ERIC-INS_2-1	Logical NE name.			S	R
N_ITEMS	Number of items the search can be retrieved.			S	O
SCOPE_LEVEL	Scope of get request.	0=base, 1=single level, 2=whole subtree		S	O
TIME_LIMIT	Number of seconds allowed for the search.			S	O

Mapping to ASDLs

The following table illustrates the CSDL to ASDL mapping for this service.

Table 28: CSDL to ASDL Mapping

CSDLs	ASDLs
C_ERIC-INS_2-1_GET_MO-WITH-FILTER	A_ERIC-INS_2-1_GET_MO-WITH-FILTER

C_ERIC-INS_2-1_SET_MO

Modifies the attributes specified for a uniquely identified MO.

Table 29: C_ERIC-INS_2-1_SET_MO

Parameter Name	Description	Range	Default Value	Type	Class
GEN_DN_PARMs	DN related parameters.			C	O
GEN_PARMs_LIST	CAI command attribute list.			C	O
NE_ID_ERIC-INS_2-1	Logical NE name.			S	R

Mapping to ASDLs

The following table illustrates the CSDL to ASDL mapping for this service.

Table 30: CSDL to ASDL Mapping

CSDLs	ASDLs
C_ERIC-INS_2-1_SET_MO	A_ERIC-INS_2-1_SET_MO

Configuring ASAP to Support Additional NE Instances

You can configure ASAP to support the INS1 - NEP configuration using the Service Activation Configuration Tool (SACT). Refer to the *ASAP Administration Guide* for more information.

Below is an example of the Activation.Configuration.XML file for the Ericsson INS 2.1 cartridge.

```
<?xml version="1.0" encoding="UTF-8"?>
<activationConfig xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns:in="http://www.metasolv.com/ServiceActivation/2003/SAMActivationConfig" xmlns="http://www.metasolv.com/ServiceActivation/2003/ActivationConfig" xmlns:fo="http://www.w3.org/1999/XSL/Format">
  <connectionPool name="INS_POOL">
    <device name="INS1_conn_1">
      <environment>MY_ASAP_SYS</environment>
      <lineType>TELNET_CONNECTION</lineType>
    </device>
    <device name="INS1_conn_2">
      <environment>MY_ASAP_SYS</environment>
      <lineType>TELNET_CONNECTION</lineType>
    </device>
    <device name="INS1_conn_3">
      <environment>MY_ASAP_SYS</environment>
      <lineType>TELNET_CONNECTION</lineType>
    </device>
    <device name="INS1_conn_4">
      <environment>MY_ASAP_SYS</environment>
      <lineType>TELNET_CONNECTION</lineType>
    </device>
    <device name="INS1_conn_5">
      <environment>MY_ASAP_SYS</environment>
      <lineType>TELNET_CONNECTION</lineType>
    </device>
  </connectionPool>
  <element name="INS1">
    <vendor>ERIC</vendor>
    <technology>INS</technology>
    <softwareLoad>2-1</softwareLoad>
    <nepServerName>$NEP</nepServerName>
    <primaryPool>INS_POOL</primaryPool>
    <maximumConnections>5</maximumConnections>
    <dropTimeout>1</dropTimeout>
  </element>
</activationConfig>
```

```

<spawnThreshold>3</spawnThreshold>
<killThreshold>1</killThreshold>
<routingElement name="INS1"/>
<communicationParameter>
  <label>HOST_NAME</label>
  <value>
    <value>beaver</value>
  </value>
  <description>Machine name for the host NE.</description>
  <lineType>TELNET_CONNECTION</lineType>
</communicationParameter>
<communicationParameter>
  <label>HOST_IPADDR</label>
  <value>
    <value>10.9.1.199</value>
  </value>
  <description>Network IP address for the host NE.</descrip-
tion>
  <lineType>TELNET_CONNECTION</lineType>
</communicationParameter>
<communicationParameter>
  <label>PORT</label>
  <value>
    <value>3300</value>
  </value>
  <description>Telnet service port.</description>
  <lineType>TELNET_CONNECTION</lineType>
</communicationParameter>
<communicationParameter>
  <label>HOST_USERID</label>
  <value>
    <value>asap</value>
  </value>
  <description>Login user name.</description>
  <lineType>TELNET_CONNECTION</lineType>
</communicationParameter>
<communicationParameter>
  <label>HOST_PASSWORD</label>
  <value>
    <value>asap123</value>
  </value>
  <description>Login password.</description>
  <lineType>TELNET_CONNECTION</lineType>
</communicationParameter>
<communicationParameter>
  <label>OPEN_TIMEOUT</label>
  <value>
    <value>10</value>
  </value>
  <description>Connection establishment timeout (in seconds).</
description>
  <lineType>TELNET_CONNECTION</lineType>
</communicationParameter>
<communicationParameter>
  <label>READ_TIMEOUT</label>
  <value>
    <value>10</value>

```

```

        </value>
        <description>Timeout for the telnet read functions (in seconds).</description>
        <lineType>TELNET_CONNECTION</lineType>
    </communicationParameter>
    <communicationParameter>
        <label>PROMPT</label>
        <value>
            <value>Enter command:</value>
        </value>
        <description>The prompt expected in the telnet session.</description>
    </communicationParameter>
    <communicationParameter>
        <label>PROVIDER_NAME</label>
        <value>
            <value>CustomNet</value>
        </value>
        <description>Provider Name within DN</description>
    </communicationParameter>
    <communicationParameter>
        <label>APPLICATION_NAME</label>
        <value>
            <value>JAS</value>
        </value>
        <description>Application Name within DN</description>
    </communicationParameter>
    <communicationParameter>
        <label>NODE_NAME</label>
        <value>
            <value>JAMBALA</value>
        </value>
        <description>Node Name within DN</description>
    </communicationParameter>
    <description>Sample NE for Ericsson INS 2.1</description>
</element>
</activationConfig>

```

Extracting source files

Before you can access an XML file to modify it, you must extract it from the sar file. Use the following procedure to extract source files from the sar file.

To extract source files

1. If necessary, create a repository directory under /ERIC_INS_2-1, copy the .sar file to the new directory and un-jar the sar file.
2. After you un-jar the sar file, you can access the XML files.

Loading a new XML file

When you finish modifying an XML, you must create a new sar file, then restart the cartridge using the new file.

Follow the instructions in [“Testing the cartridge installation” on page 8](#) for directions on how to load a new XML file.