

**Oracle ASAP™ Cartridge 1.0 for
Microsoft IPTV**

Microsoft IPTV

Second Edition
August 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

1. Cartridge Overview	1
Cartridge content	2
Prerequisites	2
About this guide	2
Services	3
Principal Management	3
Service Assurance	3
Hardware and software requirements	4
Network element (NE) interface	4
ASAP version	4
Connecting to the NE	4
Security and Authentication	4
2. Installing and Testing the Cartridge	9
Starting ASAP	9
Downloading the cartridge	10
Installing the cartridge	11
Configuring the communication parameters	11
Uninstalling the cartridge	14
Testing the cartridge installation	15
Configuring loopback and live mode parameters	15
Modifying microsoft_iptv_1_0_ne_config	15
Testing the installation	18
Logging Cartridge Activities	19
3. Atomic Service Description Layer (ASDL) Commands	21
IPTV Service Pack	22
A_MSFT-IPTV_1-0_CLR_PARENTAL-LOCK-PIN	23
A_MSFT-IPTV_1-0_DEL_DEVICE	23
A_MSFT-IPTV_1-0_MOD_CREDIT-LIMIT	24
A_MSFT-IPTV_1-0_MOD_DEV-RIGHTS-TO-PKGS	24
A_MSFT-IPTV_1-0_MOD_POSTAL-CODE	26
A_MSFT-IPTV_1-0_QRY_PACKAGE	27
A_MSFT-IPTV_1-0_QRY_PKGS-FOR-DEVICE	27
User defined exit types (tbl_user_err)	28
4. Service Definition	33
Common Service Description Layer (CSDL) commands	35
C_MSFT-IPTV_1-0_CLR_PARENTAL-LOCK-PIN	35
C_MSFT-IPTV_1-0_DEL_DEVICE	36
C_MSFT-IPTV_1-0_MOD_CREDIT-LIMIT	36
C_MSFT-IPTV_1-0_MOD_DEV-RIGHTS-TO-PKGS	37
C_MSFT-IPTV_1-0_MOD_DEVICE	38
C_MSFT-IPTV_1-0_MOD_POSTAL-CODE	40

C_MSFT-IPTV_1-0_QRY_PACKAGE	41
C_MSFT-IPTV_1-0_QRY_PKGS-FOR-DEVICE	42
5. Configuring ASAP to Support Additional NE Instances	43
Extracting source files	46
Loading a new XML file	47

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. Professional Services or systems integrators can perform extensions to the cartridge 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

System integrators such as managers, designers, programmers, and testers who are responsible for the adaptation and integration of ASAP-based solutions should use this manual as a reference. 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 Microsoft IPTV cartridge. It contains overview and technical information to assist with extending and integrating the cartridge into a customer environment.

The scope of this guide includes ASAP as it pertains to this cartridge. It is not a complete ASAP reference guide.

For additional ASAP 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 Microsoft IPTV cartridge provides the ASAP service configuration and network element (NE) interface to activate a subset of the Principal Management and Service Assurance services on Microsoft IPTV web server NEs.

Services

The ASAP Cartridge for Microsoft IPTV provides the following services:

- ◆ Principal Management (web services for managing Set-Top box devices).
- ◆ Service Assurance (web services for querying objects from the system).

Principal Management

Microsoft IPTV supports configuration and management of customer devices. Devices can have various properties managed, such as credit limit, parental-lock password and postal codes. Devices also have programming packages configured on them that control what the customer can watch on television.

The ASAP Cartridge for Microsoft IPTV supports the following actions and services.

Table 1:

Service	Description
Update Device Rights to Packages	Grants rights to a device for a list of packages.
Update Credit Limit	Updates the credit limit for a device.
Update Postal Code	Updates the postal code for a device.
Clear Parental Lock PIN	Disables a device's parental control PIN. A user reactivates their PIN through the STB UI.
Delete Device	Removes a device from the system.

Service Assurance

Microsoft IPTV supports retrieval of object properties from the system. This includes objects such as devices and packages.

The ASAP Cartridge for Microsoft IPTV supports the following actions and services.

Table 2:

Service	Description
Query Packages for Device	Retrieves a list of zero or more package names for which the device has rights.

Table 2:

Service	Description
Query Package	Returns the properties of the package object in question, describing the service collections in the package.

Hardware and software requirements

The following sections contain the high-level software and hardware environment requirements for provisioning Principal Management and Service Assurance services using this cartridge, including:

- ◆ Network element (NE) interface — Microsoft IPTV web server
- ◆ ASAP version — 4.7 and above

Network element (NE) interface

This cartridge operates with the Microsoft IPTV web server, running software load 1.0 via the SOAP/XML protocol over HTTP/HTTPS.

ASAP version

This cartridge was developed and tested using ASAP 4.7.

For more information on the operating environment of this ASAP version, refer to the ASAP 4.7 Release Record.

Connecting to the NE

The ASAP Cartridge for Microsoft IPTV interfaces with the Microsoft IPTV server to send requests and receive responses.

Security and Authentication

The following sections deal with security and authentication issues.

HTTPS Certificates

The communication layer for transmitting the SOAP messages could be either HTTP or HTTPS, depending on how the server hosting the web services was configured. A flag, `HTTPS`, exists in `tbl_comm_param` that you can set to `TRUE` if you want to use HTTPS. Set it to `FALSE` for plain HTTP. You must also modify the `PORT` variable in the `config.xml` file on

the IPTV server machine you are connecting to so that it agrees with the type of communication layer you are using (value = 80 for HTTP and 443 for HTTPS).

Where HTTPS (based on SSL) is specified, you must provide a public-key certificate file that is used to encrypt any data sent to the IPTV server. You must load this certificate into a Java keystore so that the ASAP cartridge can load it at run-time. The certificate should be a standard X.509 certificate, base64 or DER format (for example, `microsoft_iptv_x509.cer`).

Once connected via HTTPS/SSL, all transmissions between the client and server are secure. This includes a username and password that the client can send for authentication.

Java Keystore Management

The utility used to manage the Java keystore is called `keytool`. It is delivered with the J2SE 1.4 JRE, and installed on the ASAP server. To use it, login to the ASAP box as the standard ASAP user, then follow the instructions below. A keystore has two passwords `keypass` and `storepass`, which are used to secure the Java keystore. At delivery, the default values for each password are `asapPass`.

To import a certificate into a Java keystore:

```
keytool -import -v -trustcacerts -alias microsoft -file
Microsoft_IPTV_base64.cer -keystore Microsoft_IPTV_WS.keystore
```



If the specified keystore does not already exist, it will be created for you. You will be promoted to select a password at this time.

To list certificates in a Java keystore:

```
keytool -list -v -keystore Microsoft_IPTV_WS.keystore
```

To delete a certificate from a Java keystore:

```
keytool -delete -v -alias microsoft -keystore Microsoft_IPTV_WS.keystore
```

To replace the HTTPS certificate in the keystore

1. Using the `asap` login ID, log-in to the ASAP server (Note, the user ID you login with might be different, depending on the server configuration).
2. Change directory to the ASAP config directory (or the keystore location, if it is not in the default location).

```
cd $ASAP_BASE/config
```

3. Use the following command to remove the existing certificate.

```
keytool -delete -v -alias microsoft -keystore <keystore>
```

4. Load the new certificate. To do this, type the following at the prompt:

```
keytool -import -v -trustcacerts -alias microsoft -file <cert> -
keystore <keystore>
```

- Verify the new certificate is loaded by issuing the following command.

```
keytool -list -v -keystore <keystore>
```

Where,

<keystore> is the name of the keystore file (the default name is Microsoft_IPTV_WS.keystore)

<cert> is the name of the certificate file you are loading into the keystore

Username and Password Authentication

In addition, you can supply a username and password for validation on the web server. If you want to encrypt them in the database, store them in `tbl_classB_secu`. Otherwise, you can put them in `tbl_comm_param` with the `USERNAME` and `PASSWORD` names. The cartridge determines whether or not to use the encrypted credentials or the ones in `tbl_comm_param` based on the flag `USE_SECURE_DATA` (values are either: `TRUE` or `FALSE`; where `TRUE` = use encrypted credentials) in `tbl_comm_param`.

You can store the entry in a text file with the format shown below, and then load it into the database using `asap_security_tool`, which is described below.

Format

NAME:VALUE:CLASS:S_CACHE:DESCRIPTION

Example:

MSFTIPTV:user123***password123:1:0:Login username and password for IPTV

99.225.225.99:user123***password123:1:0:Login userid and password for Cisco CCM

Table 3: Secure Data

NAME:	VALUE:	CLASS:	S_CACHE:	DESCRIPTION
MSFTIPTV	user123***password123	1	0	Login user name and password for IPTV



The Username and Password must be separated by 3 asterisks (***)

ASAP Security Tool Script

To load the above file into `tbl_classB_secu` use the `asap_security_tool` script. The syntax of the script is:

```
asap_security_tool -u $CTRL_USER -p $CTRL_PASSWORD -r <filename>
```

To change the secure username and password

1. Using the `asap` login ID, log-in to the ASAP server (Note, the user ID you login with might be different, depending on the server configuration).
2. Create a text file to hold the new username and password (in the [Step 4](#) example, below, the new file is called `secu.txt`).
3. Create the following entry in the new file:

```
MSFTIPTV:<user>***<pass>:1:0:IPTV login username/password
```

Where,

<user> is username to login to IPTV

<pass> is the password to login to IPTV



The required delimiter between <user> and <pass> is three asterisks (***)

4. Load the file into the secure ASAP database table, using the following command:

```
asap_security_tool -u $CTRL_USER -p $CTRL_PASSWORD -r secu.txt
```

5. Restart ASAP, by stopping it, then restarting it; as follows:

```
stop_asap_sys -d
```

```
start_asap_sys -d
```


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](#)

Starting ASAP

Before downloading 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 System Configuration and Management Guide* contains more information on starting ASAP, the ASAP Daemon, and WebLogic.

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 MicrosoftIPTV_R1_0.bx.tar.

```
tar xvf MicrosoftIPTV_R1_0.bx.tar
```

3. Copy the resulting /MICROSOFT_IPTV_1_0_IPTV_1_0 directory and its contents to the repository directory.

```
cp -rf /MICROSOFT_IPTV_1_0_IPTV_1_0 <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>
  MICROSOFT_IPTV_1_0_IPTV_1_0
    /README
    /installCartridge
    /uninstallCartridge
    /MICROSOFT_IPTV_1_0_IPTV_1_0.sar
```

Installing the cartridge

Run the installation script `installCartridge` to install the cartridge. You will find this script under `/MICROSOFT_IPTV_1_0_IPTV_1_0`. The script executes the following tasks:

- ◆ Configures the Microsoft IPTV-specific NE using the SACT.
- ◆ Deploys the Microsoft IPTV cartridge service model (only if the Microsoft IPTV service model is not yet deployed) using the Service Activation Deployment Tool (SADT).
- ◆ Copies the Microsoft IPTV-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 System Configuration and Management Guide*.

To install the cartridge

1. Run the `installCartridge` script from `/MICROSOFT_IPTV_1_0_IPTV_1_0`. At the prompt, type:

```
installCartridge MICROSOFT_IPTV_1_0_IPTV_1_0
```

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.

Configuring the communication parameters

This static database table contains the communication parameters required to communicate with various external systems. It consists of the following fields:

- ◆ `dev_type` — the protocol used to connect to the device.
- ◆ `host` — the host NE to which the configured parameter value applies.
- ◆ `device` — the device for which this parameter value applies.
- ◆ `param_label` — the label of the communication parameter.
- ◆ `param_value` — the value of the communication parameter.
- ◆ `param_desc` — a thorough and accurate description of the communication parameter

Use the following table as a guide to configuring the communication parameter table in your installation. Ensure you substitute your own, site-specific parameter values for the sample ones provided, below.

Table 4: tbl_comm_param

TYPE	HOST	DEVICE	PARAM_LABEL	PARAM_VALUE	PARAM_DESC
G	MSFT-IPTV-HOST	COMMON_DEVICE_CFG	HOST_IPADDR	142.183.190.228	The host name or IP Address of the remote NE
G	MSFT-IPTV-HOST	COMMON_DEVICE_CFG	PORT	80	Port number to connect on remote NE host
G	MSFT-IPTV-HOST	COMMON_DEVICE_CFG	READ_TIMEOUT	5000	The read timeout in milliseconds
G	MSFT-IPTV-HOST	COMMON_DEVICE_CFG	USER_ERROR_TYPES_FILE	/config/Microsoft_IPTV_1_0_UserExitTypes.cfg	The User Exit types file. This file is relative to ASAP_BASE directory
G	MSFT-IPTV-HOST	COMMON_DEVICE_CFG	NE_DIALOG_LOGGING	TRUE	Flag to turn On or Off SRQ logging
G	MSFT-IPTV-HOST	COMMON_DEVICE_CFG	HTTPS	TRUE	Flag to turn On or Off HTTPS security
G	MSFT-IPTV-HOST	COMMON_DEVICE_CFG	USE_SECURE_DATA	TRUE	Flag to turn On or Off use of secure data
G	MSFT-IPTV-HOST	COMMON_DEVICE_CFG	USERNAME	Username	String containing username for authentication with web service

Table 4: tbl_comm_param

TYPE	HOST	DEVICE	PARAM_LABEL	PARAM_VALUE	PARAM_DESC
G	MSFT-IPTV-HOST	COMMON_DEVICE_CFG	PASSWORD	Password	String containing password for authentication with web service
G	MSFT-IPTV-HOST	COMMON_DEVICE_CFG	PM_SOAP_FILE	PrincipalManagementWS/PrincipalManagementWS.asmx	The SOAP URL to add to the HOST_IPADDR and PORT to connect to the remote PM web service
G	MSFT-IPTV-HOST	COMMON_DEVICE_CFG	SA_SOAP_FILE	ServiceAssuranceWS/ServiceAssuranceWS.asmx	The SOAP URL to add to the HOST_IPADDR and PORT to connect to the remote SA web service
G	MSFT-IPTV-HOST	COMMON_DEVICE_CFG	USE_PROXY_SERVER	FALSE	Flag to turn On or Off use of a Web Proxy to communicate with the SOAP service through
G	MSFT-IPTV-HOST	COMMON_DEVICE_CFG	PROXY_HOST	Proxy.domain.com	The hostname of a web proxy server required to access the web service host via HTTP/HTTPS

Table 4: tbl_comm_param

TYPE	HOST	DEVICE	PARAM_LABEL	PARAM_VALUE	PARAM_DESC
G	MSFT-IPTV-HOST	COMMON_DEVICE_CFG	PROXY_PORT	80	The port number of a web proxy server required to access the web service host via HTTP/HTTPS
G	MSFT-IPTV-HOST	COMMON_DEVICE_CFG	CERTIFICATE_FILE	/config/Microsoft_IPTV_WS.keystore	SSL trusted certificate file to use for HTTPS. This file is relative to ASAP_BASE directory

Uninstalling the cartridge

Run the uninstallation script `uninstallCartridge` to uninstall the Microsoft IPTV cartridge. This script is located under `MICROSOFT_IPTV_1_0_IPTV_1_0`. The script executes the following tasks:

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

For more information on the SACT and the SADT, refer to the *ASAP System Configuration and Management Guide*.

To uninstall the cartridge

1. Run the `uninstallCartridge` script from `/MICROSOFT_IPTV_1_0_IPTV_1_0`. At the prompt, type

```
uninstallCartridge MICROSOFT_IPTV_1_0_IPTV_1_0.
```

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

The following sections tell you which variables you must configure in to use the loopback and live testing modes.

Loopback mode

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

Table 5: Loopback Mode Parameter Settings

Configuration Variable	Parameter Settings	Location
LOOPBACK_ON	1 (default setting)	ASAP.cfg

Live mode

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

Table 6: Live Mode Parameter Settings

Configuration Variable	Parameter Settings	Location
LOOPBACK_ON	0	ASAP.cfg

Modifying microsoft_ip_tv_1_0_ne_config

Use the following procedure to modify microsoft_ip_tv_1_0_ne_config.

To modify microsoft iptv_1_0_ne_config

1. Create a new source directory under /MICROSOFT_IPTV_1_0_IPTV_1_0. You can give this directory any appropriate, meaningful name you want to.

```
mkdir <new_source_directory>
```

2. Copy MICROSOFT_IPTV_1_0_IPTV_1_0.sar to this new source directory.

```
cp MICROSOFT_IPTV_1_0_IPTV_1_0.sar ./<new_source_directory>
```

3. Change directory to <new_source_directory>.

```
cd <new_source_directory>
```

4. Un-jar MICROSOFT_IPTV_1_0_IPTV_1_0.sar This extracts the contents of the sar file (see [Figure 1](#) on page 17 for an example of the resulting file structure).

```
jar xvf MICROSOFT_IPTV_1_0_IPTV_1_0.sar
```

5. Edit <new_source_directory>/MICROSOFT_IPTV_1_0_IPTV_1_0/common/application_config/microsoft iptv_1_0_ne_config in with the appropriate changes.

6. Create a new sar file at the <new_source_directory> level.

```
CreateSar $PWD
```

7. Uninstall the cartridge using MICROSOFT_IPTV_1_0_IPTV_1_0.sar in /MICROSOFT_IPTV_1_0_IPTV_1_0 (That is, use the original sar file that you copied in [Step 2](#) above—see [“Uninstalling the cartridge”](#) on page 14 for uninstallation instructions).

8. After you uninstall the cartridge, rename the sar file in /MICROSOFT_IPTV_1_0_IPTV_1_0 so you have a backup copy of it.

9. Copy the new sar file from <new_source_directory> to /MICROSOFT_IPTV_1_0_IPTV_1_0.

10. Reinstall the cartridge (see [“Installing the cartridge”](#) on page 11 for installation instructions).

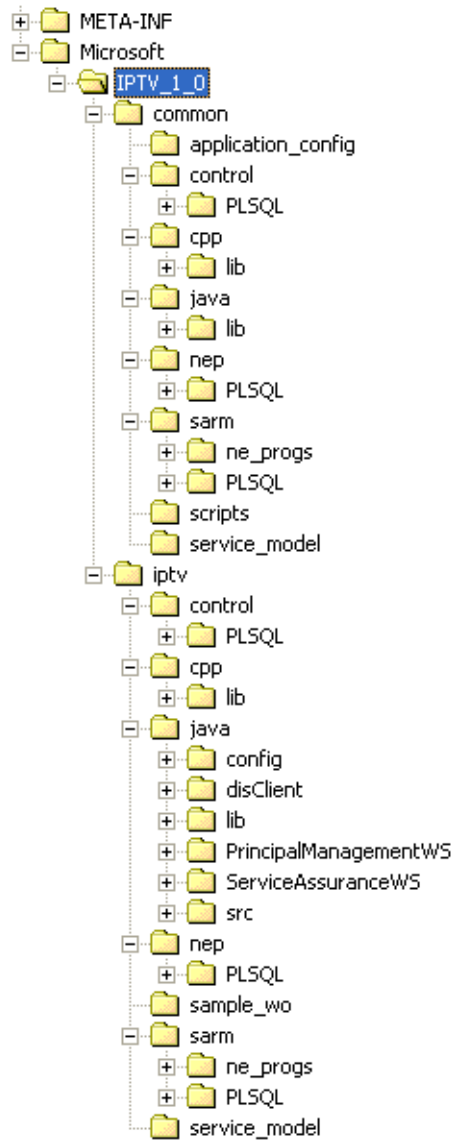


Figure 1: File Structure of the Un-Jared .sar File

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 15 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 `/MICROSOFT_IPTV_1_0_IPTV_1_0/sample_wo` by typing:

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

A list of all available suites appears.

To see the sample work orders, refer to [Viewing the sample work orders](#), below.

For more information on the SRP Emulator, refer to the *ASAP System Configuration and Management 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 Microsoft IPTV cartridge source.

Viewing the sample work orders

You find the sample work orders under the sample_wo directory in the sar file. The following procedure describes how to view the sample work orders.

To view the sample work orders

1. If necessary, create a repository directory under /MICROSOFT_IPTV_1_0_IPTV_1_0, copy the sar file to the new directory and un-jar the sar file, as described by [Step 1](#) through [Step 4](#) in “[Modifying microsoft_iptv_1_0_ne_config](#)” on page 15.
2. Locate and view the sample work order files under /MICROSOFT_IPTV_1_0_IPTV_1_0/NE_Technology/service pack/sample_wo.

Logging Cartridge Activities

The cartridge uses different logging mechanisms to record events and diagnostic information, which you can trace. The logging mechanisms are as follows:

- ◆ com.mslv.activation.server.Diagnosis -
 - ◆ used to write to the local file system for debugging events
 - ◆ usually in \$ASAP_HOME/DATA/logs/<date>/*.diag
- ◆ com.mslv.activation.jinterpreter.Log
 - ◆ used to write to the persistent storage for event tracking,
 - ◆ writes to the db, SRQ_LOG table.)
 - ◆ will be used to store Microsoft IPTV communication events

Details that are logged to SRQ_LOG include:

- ◆ ASDL parameters
- ◆ SOAP request
- ◆ SOAP response
- ◆ Exception messages and exit types

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 7: 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 7: 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 System Configuration and Management Guide*.

IPTV Service Pack

This cartridge provides the following ASDL commands:

- ◆ A_MSFT-IPTV_1-0_CLR_PARENTAL-LOCK-PIN
- ◆ A_MSFT-IPTV_1-0_DEL_DEVICE
- ◆ A_MSFT-IPTV_1-0_MOD_CREDIT-LIMIT
- ◆ A_MSFT-IPTV_1-0_MOD_DEV-RIGHTS-TO-PKGS
- ◆ A_MSFT-IPTV_1-0_MOD_POSTAL-CODE
- ◆ A_MSFT-IPTV_1-0_QRY_PACKAGE
- ◆ A_MSFT-IPTV_1-0_QRY_PKGS-FOR-DEVICE

A_MSFT-IPTV_1-0_CLR_PARENTAL-LOCK-PIN

Disables a device's parental lock PIN. It is implemented by the Java method `com.metasolv.cartridge.oss.microsoft_iptv_1_0.prov.Provisioning.disablePin`.

Table 8: A_MSFT-IPTV_1-0_CLR_PARENTAL-LOCK-PIN

Parameter Name	Description	Range	Default Value	Type	Class
MCLI	The host NE identifier.			S	R
DEVICE_ID	The device ID.			S	R

A_MSFT-IPTV_1-0_DEL_DEVICE

Deletes a device. It is implemented by the Java method `com.metasolv.cartridge.oss.microsoft_iptv_1_0.prov.Provisioning.deleteDevice`.

Table 9: A_MSFT-IPTV_1-0_DEL_DEVICE

Parameter Name	Description	Range	Default Value	Type	Class
MCLI	The host NE identifier.			S	R
DEVICE_ID	The device ID.			S	R

A_MSFT-IPTV_1-0_MOD_CREDIT-LIMIT

Updates the credit limit for a device. It is implemented by the Java method `com.metasolv.cartridge.oss.microsoft_iptv_1_0.prov.Provisioning.updateCreditLimit`.

Table 10: A_MSFT-IPTV_1-0_MOD_CREDIT-LIMIT

Parameter Name	Description	Range	Default Value	Type	Class
MCLI	The host NE identifier.			S	R
DEVICE_ID	The device ID.			S	R
CREDIT_LIMIT	The credit limit.			S	R

A_MSFT-IPTV_1-0_MOD_DEV-RIGHTS-TO-PKGS

Grants rights to a device for a list of packages. It is implemented by the Java method `com.metasolv.cartridge.oss.microsoft_iptv_1_0.prov.Provisioning.updateDeviceRightsTo Packages`.

Table 11: A_MSFT-IPTV_1-0_MOD_DEV-RIGHTS-TO-PKGS

Parameter Name	Description	Range	Default Value	Type	Class
MCLI	The host NE identifier.			S	R
DEVICE_ID	The device ID.			S	R
PACKAGE	<p>The programming package and associated right type.</p> <p>This optional compound parameter contains both the PACKAGE_NAME and LICENSE_RIGHT to be granted to the device in question. See “Compound Parameter Description” below, for more details and examples. PACKAGE_NAME and LICENSE_RIGHT are expected as string values.</p>	Play Pause Record		N	O

Compound Parameter Description

The `PACKAGE` compound parameter contains the programming package-names the command is adding to the device, and indicates the kind of licensing rights-type the command is granting for each package.

When working with this parameter, keep the following in mind:

- ◆ Omitting this parameter instructs the cartridge to delete rights for all packages from the device.
- ◆ There must be an equal number of package names and license rights, if the number is unequal the ASDL fails.
- ◆ The list of packages/rights passed to this ASDL overwrites whatever packages/rights were previously configured on this device.
- ◆ The index always start with 1 and increments by 1, for example: 1, 2, 3, 4, ..., N.

Example 1 - CSDL Work Order parameters: remove all package rights from device:

```
NE_ID_MSFT-IPTV    msiptv10
DEVICE_ID         toronto1234
```

Example 2 - CSDL Work Order parameters: two packages granted to the device:

```
NE_ID_MSFT-IPTV    msiptv10
DEVICE_ID         toronto1234
PACKAGE[1].NAME    CHANNEL_1
PACKAGE[1].RIGHT   Play
PACKAGE[2].NAME    CHANNEL_3
PACKAGE[2].RIGHT   Record
```

Table 12: Package Compound Parameter Subparameter Descriptions

PARAM LABEL	PARAM TYPE	ELEMENT ATTRIBUTE TYPE	DESCRIPTION
PACKAGE[N].NAME	N	String (Where NAME is a string. Note: there must be a corresponding package with the same name as .NAME already added to the IPTV system.)	Name of the programming package to be granted access to for the device in question.

Table 12: Package Compound Parameter Subparameter Descriptions (Continued)

PARAM LABEL	PARAM TYPE	ELEMENT ATTRIBUTE TYPE	DESCRIPTION
PACKAGE[N].RIGHT	N	String	<p>License Right Type to be enabled for each programming package that is to be added to the customer device.</p> <p>Choices include: “Play”, “Pause”, “Record”.</p> <p>Each of the three right types enables support for the previous right type. For example, a right type of “Pause” results in right types of “Play” and “Pause” being enabled for that package on the device. Similarly, a right type of “Record” results in right types of “Play”, “Pause” and “Record” being enabled for that package.</p>

A_MSFT-IPTV_1-0_MOD_POSTAL-CODE

Updates the postal code for a device. It is implemented by the Java method `com.metasolv.cartridge.oss.microsoft_ip_tv_1_0.prov.Provisioning.updatePostalCode`.

Table 13: A_MSFT-IPTV_1-0_MOD_POSTAL-CODE

Parameter Name	Description	Range	Default Value	Type	Class
MCLI	The host NE identifier.			S	R
DEVICE_ID	The device ID.			S	R
POSTAL_CODE	The postal code.			S	R

A_MSFT-IPTV_1-0_QRY_PACKAGE

Retrieves package information. It is implemented by the Java method `com.metasolv.cartridge.oss.microsoft_iptv_1_0.prov.Provisioning.readPackage`.

Table 14: A_MSFT-IPTV_1-0_QRY_PACKAGE

Parameter Name	Description	Range	Default Value	Type	Class
MCLI	The host NE identifier.			S	R
PACKAGE_NAME	The package name.			S	R

A_MSFT-IPTV_1-0_QRY_PKGS-FOR-DEVICE

Retrieves an array of zero or more package names for which the device has rights. It is implemented by the Java method `com.metasolv.cartridge.oss.microsoft_iptv_1_0.prov.Provisioning.readPackagesForDevice`.

Table 15: A_MSFT-IPTV_1-0_QRY_PKGS-FOR-DEVICE

Parameter Name	Description	Range	Default Value	Type	Class
MCLI	The host NE identifier.			S	R
DEVICE_ID	The device ID.			S	R

User defined exit types (tbl_user_err)

You can customize the cartridge's error handling behavior by updating the following file:

`user_err.sql`

From this file, you can add new user exit types or remap any existing user-type exit code (USER_TYPE) to a different base exit type (BASE_TYPE). For example, if on a given NE you want a particular ASDL to return a SOFT_FAIL instead of a FAIL (a hard fail) when the NE detects a specific error, you change the BASE_TYPE associated with the appropriate USER_TYPE from FAIL to SOFT_FAIL (the ASDL state table or Java method's exit type is mapped to the USER_TYPE).



If you add new user exit types to `user_err.sql`, you should add corresponding "removal" statements to the `"undeploy_user_err.sql"` too.

Before

USER_TYPE	BASE_TYPE	DESCRIPTION
MSIPTV10_DEV_NOTFND	FAIL	The specified device was not found

After

USER_TYPE	BASE_TYPE	DESCRIPTION
MSIPTV10_DEV_NOTFND	SOFT_FAIL	The specified device was not found

`tbl_user_err` consists of the following fields:

- ◆ `user_type`: User-defined ASDL exit type.
- ◆ `base_type`: The base ASDL exit type where this user specified ASDL exit type maps to. The base types include:
 - ◆ SUCCEED – successful ASDL execution.
 - ◆ FAIL – hard error.
 - ◆ RETRY – ASDL failed, but retries.
 - ◆ MAINTENANCE – ASDL detected NE in maintenance mode.
 - ◆ SOFT_FAIL – a soft error occurred, but processing continues.
 - ◆ DELAYED_FAIL – ASDL failed. Processing continues, but the order fails.
 - ◆ STOP – stops the ASDL from processing.
- ◆ `description`: Description of the user exit type.

The following table lists the contents of tbl_user_err.


 If this file is removed or its path is incorrect, the user exit type will default to MSFTIPTV_NO_MATCH and is mapped to ASAP exit type FAIL.

Table 16: tbl_usr_err

USER_TYPE	BASE_TYPE	Description
MSIPTV10_SUCCESS	SUCCEED	None.
MSIPTV10_GENERERROR	FAIL	GENERAL_ERROR.
MSIPTV10_DEV_NOTFND	FAIL	The specified device was not found.
MSIPTV10_DEV_DELETED	SOFT_FAIL	DeleteDevice: The specified device was not found.
MSIPTV10_EXTID_INV	FAIL	The external ID is invalid.
MSIPTV10_PKG_NULL	FAIL	The package name cannot be null.
MSIPTV10_PKG_NOTFND	FAIL	The specified package was not found.
MSIPTV10_RT_INVALID	FAIL	The right is invalid.
MSIPTV10_UNEQUALNUMS	FAIL	There must be the same number of packages as requestedRights.
MSIPTV10_PC_INVALID	FAIL	The postal code is not correctly formed.
MSIPTV10_PC_NULL	FAIL	The postal code cannot be null.
MSIPTV10_PC_NOTFND	FAIL	The specified postal code was not found.
MSIPTV10_PC_6CHARS	FAIL	The postal code must contain 6 characters.
MSIPTV10_CL_POSVALUE	FAIL	Credit limit must be a positive value.
MSIPTV10_SERV_FAIL	FAIL	Service failure.
MSIPTV10_SERV_FAIL	FAIL	Technical service failure.
MSIPTV10_BILL_PEND	FAIL	The device cannot be deleted because it has associated billing events.
MSIPTV10_RT_INVALID	FAIL	The requestedRights is invalid.
MSIPTV10_PKG_NULL	FAIL	The packageNames cannot be null.

The following tables show how the Microsoft errors map to each of the user exit types. These will help you understand what the possible errors are, and which provisioning methods they come from. The tables are split logically, one each for the different web services, as follows:

- ◆ Microsoft Principal Management Exceptions (Table 17) — errors from supported principal management API's.
- ◆ Microsoft Service Assurance Exceptions (Table 18) — errors from supported service assurance APIs.

Table 17: Microsoft Principal Management Exceptions

API METHOD NAME	EXCEPTION MESSAGE	USER EXIT TYPE
UpdateDeviceRightsToPackages	The packageNames cannot be null	MSIPTV10_PKG_NULL
	The package name cannot be null	MSIPTV10_PKG_NULL
	The specified package was not found	MSIPTV10_PKG_NOTFND
	The specified device was not found	MSIPTV10_DEV_NOTFND
	The external ID is invalid	MSIPTV10_EXTID_INV
	The requestedRights is invalid	MSIPTV10_RT_INVALID
	There must be the same number of packages as requestedRights	MSIPTV10_UNEQUALNUMS
	Service failure	MSIPTV10_SERV_FAIL
UpdatePostalCode	The specified device was not found	MSIPTV10_DEV_NOTFND
	The external ID is invalid	MSIPTV10_EXTID_INV
	The postal code is not correctly formed	MSIPTV10_PC_INVALID
	The postal code cannot be null	MSIPTV10_PC_NULL
	The specified postal code was not found	MSIPTV10_PC_NOTFND
	The postal code must contain 6 characters	MSIPTV10_PC_6CHARS
	Service failure	MSIPTV10_SERV_FAIL

Table 17: Microsoft Principal Management Exceptions

API METHOD NAME	EXCEPTION MESSAGE	USER EXIT TYPE
UpdateCreditLimit	The specified device was not found	MSIPTV10_DEV_NOTFND
	The external ID is invalid	MSIPTV10_EXTID_INV
	Credit limit must be a positive value	MSIPTV10_CL_POSVALUE
	Service failure	MSIPTV10_SERV_FAIL
DisablePin	The specified device was not found	MSIPTV10_DEV_NOTFND
	The external ID is invalid	MSIPTV10_EXTID_INV
	Service failure	MSIPTV10_SERV_FAIL
DeleteDevice	DeleteDevice: The specified device was not found	MSIPTV10_DEV_NOTFND
	The external ID is invalid	MSIPTV10_EXTID_INV
	The device cannot be deleted because it has associated billing events	MSIPTV10_BILL_PEND
	Service failure	MSIPTV10_SERV_FAIL
ReadPackagesForDevice	External ID is invalid	MSIPTV10_EXTID_INV
	Device was not found	MSIPTV10_DEV_NOTFND
	Service failure	MSIPTV10_SERV_FAIL

Table 18: Microsoft Service Assurance Exceptions

API METHOD NAME	EXCEPTION MESSAGE	USER EXIT TYPE
ReadPackagesForDevice	External ID is invalid	MSIPTV10_EXTID_INV
	External Id cannot be null	MSIPTV10_EXTID_NULL
	Device was not found	MSIPTV10_DEV_NOTFND
	Service Failure	MSIPTV10_SERV_FAIL

Table 18: Microsoft Service Assurance Exceptions

API METHOD NAME	EXCEPTION MESSAGE	USER EXIT TYPE
ReadPackage	Package name is invalid	MSIPTV10_PKG_INVALID
	Package not found	MSIPTV10_PKG_NOTFND
	Package name cannot be null	MSIPTV10_PKG_NULL
	Service Failure	MSIPTV10_SERV_FAIL

Service Definition

The Microsoft IPTV 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 19: 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 19: 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 System Configuration and Management Guide*.

Common Service Description Layer (CSDL) commands

This cartridge provides the following CSDL Commands:

- ◆ C_MSFT-IPTV_1-0_CLR_PARENTAL-LOCK-PIN
- ◆ C_MSFT-IPTV_1-0_DEL_DEVICE
- ◆ C_MSFT-IPTV_1-0_MOD_CREDIT-LIMIT
- ◆ C_MSFT-IPTV_1-0_MOD_DEV-RIGHTS-TO-PKGS
- ◆ C_MSFT-IPTV_1-0_MOD_DEVICE
- ◆ C_MSFT-IPTV_1-0_MOD_POSTAL-CODE
- ◆ C_MSFT-IPTV_1-0_QRY_PACKAGE
- ◆ C_MSFT-IPTV_1-0_QRY_PKGS-FOR-DEVICE

C_MSFT-IPTV_1-0_CLR_PARENTAL-LOCK-PIN

Disables a devices parental lock PIN.

Table 20: C_MSFT-IPTV_1-0_CLR_PARENTAL-LOCK-PIN

Parameter Name	Description	Range	Default Value	Type	Class
DEVICE_ID	The device ID.			S	R
NE_ID_MSFT-IPTV	The host NE identifier.			S	R

Mapping to ASDLs

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

Table 21: CSDL to ASDL Mapping

CSDL	ASDL
C_MSFT-IPTV_1-0_CLR_PARENTAL-LOCK-PIN	A_MSFT-IPTV_1-0_CLR_PARENTAL-LOCK-PIN

C_MSFT-IPTV_1-0_DEL_DEVICE

Delete a device.

Table 22: C_MSFT-IPTV_1-0_DEL_DEVICE

Parameter Name	Description	Range	Default Value	Type	Class
DEVICE_ID	The device ID.			S	R
NE_ID_MSFT-IPTV	The host NE identifier.			S	R

Mapping to ASDLs

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

Table 23: CSDL to ASDL Mapping

CSDL	ASDL
C_MSFT-IPTV_1-0_DEL_DEVICE	A_MSFT-IPTV_1-0_DEL_DEVICE

C_MSFT-IPTV_1-0_MOD_CREDIT-LIMIT

Changes the credit limit for a device.

Table 24: C_MSFT-IPTV_1-0_MOD_CREDIT-LIMIT

Parameter Name	Description	Range	Default Value	Type	Class
CREDIT_LIMIT	The credit limit.			S	R
DEVICE_ID	The device ID.			S	R
NE_ID_MSFT-IPTV	The host NE identifier.			S	R

Mapping to ASDLs

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

Table 25: CSDL to ASDL Mapping

CSDL	ASDL
C_MSFT-IPTV_1-0_MOD_CREDIT-LIMIT	A_MSFT-IPTV_1-0_MOD_CREDIT-LIMIT

C_MSFT-IPTV_1-0_MOD_DEV-RIGHTS-TO-PKGS

Grants rights to a device for a list of packages.

Table 26: C_MSFT-IPTV_1-0_MOD_DEV-RIGHTS-TO-PKGS

Parameter Name	Description	Range	Default Value	Type	Class
DEVICE_ID	The device ID.			S	R
NE_ID_MSFT-IPTV	The host NE identifier.			S	R
PACKAGE	<p>The programming package and associated right type.</p> <p>This optional compound parameter contains both the PACKAGE_NAME and LICENSE_RIGHT to be granted to the device in question. See “Compound Parameter Description” below, for more details and examples. PACKAGE_NAME and LICENSE_RIGHT are expected as string values.</p>	Play Pause Record		N	O

Mapping to ASDLs

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

Table 27: CSDL to ASDL Mapping

CSDL	ASDL
C_MSFT-IPTV_1-0_MOD_DEV-RIGHTS-TO-PKGS	A_MSFT-IPTV_1-0_MOD_DEV-RIGHTS-TO-PKGS

C_MSFT-IPTV_1-0_MOD_DEVICE

Deletes a device.

Table 28: C_MSFT-IPTV_1-0_MOD_DEVICE

Parameter Name	Description	Range	Default Value	Type	Class
CREDIT_LIMIT	The credit limit.			S	R
DEVICE_ID	The device ID.			S	R
NE_ID_MSFT-IPTV	The host NE identifier.			S	R
PACKAGE	The programming package and associated right type. This optional compound parameter contains both the PACKAGE_NAME and LICENSE_RIGHT to be granted to the device in question. See “Compound Parameter Description” below, for more details and examples. PACKAGE_NAME and LICENSE_RIGHT are expected as string values.	Play Pause Record		N	O
POSTAL_CODE	The postal code.			S	R

Mapping to ASDLs

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

Table 29: CSDL to ASDL Mapping

CSDL	ASDL
C_MSFT-IPTV_1-0_MOD_DEVICE	A_MSFT-IPTV_1-0_MOD_DEV-RIGHTS-TO-PKGS
	A_MSFT-IPTV_1-0_MOD_CREDIT-LIMIT
	A_MSFT-IPTV_1-0_MOD_POSTAL-CODE

Compound Parameter Description

The `PACKAGE` compound parameter contains the programming package-names the command is adding to the device, and indicates the kind of licensing right-types the command is granting.

When working with this parameter, keep the following in mind:

- ◆ Omitting this parameter instructs the cartridge to delete rights for all packages from the device.
- ◆ There must be an equal number of package names and license rights, if the number is unequal the ASDL fails.
- ◆ The list of packages/rights passed to this ASDL overwrites whatever packages/rights were previously configured on this device.
- ◆ The index always start with 1 and increments by 1, for example: 1, 2, 3, 4, ..., N.

Example 1 - CSDL Work Order parameters: remove all package rights from device:

```
NE_ID_MSFT-IPTV    msiptv10
DEVICE_ID         toronto1234
```

Example 2 - CSDL Work Order parameters: two packages granted to the device:

```
NE_ID_MSFT-IPTV    msiptv10
DEVICE_ID         toronto1234
PACKAGE[1].NAME    CHANNEL_1
PACKAGE[1].RIGHT   Play
PACKAGE[2].NAME    CHANNEL_3
```

PACKAGE[2].RIGHT Record

Table 30: Package Compound Parameter Subparameter Descriptions

PARAM LABEL	PARAM TYPE	ELEMENT ATTRIBUTE TYPE	DESCRIPTION
PACKAGE[N].NAME	N	String (Where NAME is a string. Note: there must be a corresponding package with the same name as .NAME already added to the IPTV system.)	Name of the programming package to be granted access to for the device in question.
PACKAGE[N].RIGHT	N	String	<p>License Right Type to be enabled for each programming package that is to be added to the customer device.</p> <p>Choices include: “Play”, “Pause”, “Record”.</p> <p>Each of the three right types enables support for the previous right type. For example, a right type of “Pause” results in right types of “Play” and “Pause” being enabled for that package on the device. Similarly, a right type of “Record” results in right types of “Play”, “Pause” and “Record” being enabled for that package.</p>

C_MSFT-IPTV_1-0_MOD_POSTAL-CODE

Changes the postal code for a device.

Table 31: C_MSFT-IPTV_1-0_MOD_POSTAL-CODE

Parameter Name	Description	Range	Default Value	Type	Class
DEVICE_ID	The device ID.			S	R
NE_ID_MSFT-IPTV	The host NE identifier.			S	R

Table 31: C_MSFT-IPTV_1-0_MOD_POSTAL-CODE

Parameter Name	Description	Range	Default Value	Type	Class
POSTAL_CODE	The postal code.			S	R

Mapping to ASDLs

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

Table 32: CSDL to ASDL Mapping

CSDL	ASDL
C_MSFT-IPTV_1-0_MOD_POSTAL-CODE	A_MSFT-IPTV_1-0_MOD_POSTAL-CODE

C_MSFT-IPTV_1-0_QRY_PACKAGE

Retrieves package information.

Table 33: C_MSFT-IPTV_1-0_QRY_PACKAGE

Parameter Name	Description	Range	Default Value	Type	Class
NE_ID_MSFT-IPTV	The host NE identifier.			S	R
PACKAGE_NAME	The package name.			S	R

Mapping to ASDLs

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

Table 34: CSDL to ASDL Mapping

CSDL	ASDL
C_MSFT-IPTV_1-0_QRY_PACKAGE	A_MSFT-IPTV_1-0_QRY_PACKAGE

C_MSFT-IPTV_1-0_QRY_PKGS-FOR-DEVICE

Retrieves an array of zero or more package names for which the device has rights.

Table 35: C_MSFT-IPTV_1-0_QRY_PKGS-FOR-DEVICE

Parameter Name	Description	Range	Default Value	Type	Class
DEVICE_ID	The device ID.			S	R
NE_ID_MSFT-IPTV	The host NE identifier.			S	R

Mapping to ASDLs

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

Table 36: CSDL to ASDL Mapping

CSDL	ASDL
C_MSFT-IPTV_1-0_QRY_PKGS-FOR-DEVICE	A_MSFT-IPTV_1-0_QRY_PKGS-FOR-DEVICE

Configuring ASAP to Support Additional NE Instances

You can configure ASAP to support the Microsoft IPTV web server - NEP configuration using the Service Activation Configuration Tool (SACT). Refer to the *ASAP System Configuration and Management Guide* for more information.

Below is an example of the Activation.Configuration.XML file for the Microsoft IPTV cartridge.

```
<activationConfig xmlns="http://www.metasolv.com/ServiceActivation/2003/
ActivationConfig" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.metasolv.com/ServiceActivation/2003/
ActivationConfig
ActivationConfig.xsd">
  <connectionPool name="IPTVPOOL">
    <device name="MSFT-IPTV-HOST">
      <environment>MY_ASAP_SYS</environment>
      <lineType>GENERIC_MESSAGE_BASED_CONNECTION</lineType>
    </device>
  </connectionPool>
  <element name="MSFT-IPTV-HOST">
    <technology>MSFT-IPTV</technology>
    <softwareLoad>1-0</softwareLoad>
    <nepServerName>$NEP</nepServerName>
    <primaryPool>IPTVPOOL</primaryPool>
    <maximumConnections>1</maximumConnections>
    <dropTimeout>2</dropTimeout>
    <spawnThreshold>0</spawnThreshold>
    <killThreshold>0</killThreshold>
    <routingElement name="MSFT-IPTV-HOST">
      <atomicService/>
    </routingElement>
    <communicationParameter>
      <label>HOST_IPADDR</label>
      <value>
        <value>142.183.190.228</value>
      </value>
      <description>The host name or IP Address of the remote NE</
description>
      <deviceName>COMMON_DEVICE_CFG</deviceName>
```

```

        <lineType>GENERIC_MESSAGE_BASED_CONNECTION</lineType>
    </communicationParameter>
    <communicationParameter>
        <label>PORT</label>
        <value>
            <value>443</value>
        </value>
        <description>Port number to connect on remote NE host</description>
        <deviceName>COMMON_DEVICE_CFG</deviceName>
        <lineType>GENERIC_MESSAGE_BASED_CONNECTION</lineType>
    </communicationParameter>
    <communicationParameter>
        <label>READ_TIMEOUT</label>
        <value>
            <value>60000</value>
        </value>
        <description>The read timeout in milliseconds</description>
        <deviceName>COMMON_DEVICE_CFG</deviceName>
        <lineType>GENERIC_MESSAGE_BASED_CONNECTION</lineType>
    </communicationParameter>
    <communicationParameter>
        <label>USER_ERROR_TYPES_FILE</label>
        <value>
            <value>/config/Microsoft_IPTV_1_0_UserExitTypes.cfg</value>
        </value>
        <description>The User Exit types file. This file is relative to
ASAP_BASE directory</description>
        <deviceName>COMMON_DEVICE_CFG</deviceName>
        <lineType>GENERIC_MESSAGE_BASED_CONNECTION</lineType>
    </communicationParameter>
    <communicationParameter>
        <label>PM_SOAP_FILE</label>
        <value>
            <value>PrincipalManagementWS/PrincipalManagementWS.asmx</value>
        </value>
        <description>The remote NE Principal management service file path,
added to the IP address and port</description>
        <deviceName>COMMON_DEVICE_CFG</deviceName>
        <lineType>GENERIC_MESSAGE_BASED_CONNECTION</lineType>
    </communicationParameter>
    <communicationParameter>
        <label>SA_SOAP_FILE</label>
        <value>
            <value>ServiceAssuranceWS/ServiceAssuranceWS.asmx</value>
        </value>
        <description>The remote NE Service Assurance service file path,
added to the IP address and port</description>
        <deviceName>COMMON_DEVICE_CFG</deviceName>
        <lineType>GENERIC_MESSAGE_BASED_CONNECTION</lineType>
    </communicationParameter>
    <communicationParameter>
        <label>NE_DIALOG_LOGGING</label>

```

```

    <value>
      <value>TRUE</value>
    </value>
    <description>Flag to turn On or Off logging of API commands and
responses</description>
    <deviceName>COMMON_DEVICE_CFG</deviceName>
    <lineType>GENERIC_MESSAGE_BASED_CONNECTION</lineType>
  </communicationParameter>
  <communicationParameter>
    <label>HTTPS</label>
    <value>
      <value>TRUE</value>
    </value>
    <description>Flag to turn On or Off HTTPS transfer. If Off is
selected, standard HTTP will be used</description>
    <deviceName>COMMON_DEVICE_CFG</deviceName>
    <lineType>GENERIC_MESSAGE_BASED_CONNECTION</lineType>
  </communicationParameter>
  <communicationParameter>
    <label>USE_SECURE_DATA</label>
    <value>
      <value>TRUE</value>
    </value>
    <description>Flag indicating wether Secure Login Username and
Password Required</description>
    <deviceName>COMMON_DEVICE_CFG</deviceName>
    <lineType>GENERIC_MESSAGE_BASED_CONNECTION</lineType>
  </communicationParameter>
  <communicationParameter>
    <label>USERNAME</label>
    <value>
      <value>Administrator</value>
    </value>
    <description>Insecure username field for authentication with web
service</description>
    <deviceName>COMMON_DEVICE_CFG</deviceName>
    <lineType>GENERIC_MESSAGE_BASED_CONNECTION</lineType>
  </communicationParameter>
  <communicationParameter>
    <label>PASSWORD</label>
    <value>
      <value>M1cro$oft</value>
    </value>
    <description>Insecure password field for authentication with web
service</description>
    <deviceName>COMMON_DEVICE_CFG</deviceName>
    <lineType>GENERIC_MESSAGE_BASED_CONNECTION</lineType>
  </communicationParameter>
  <communicationParameter>
    <label>CERTIFICATE_FILE</label>
    <value>
      <value>/config/Microsoft_IPTV_WS.keystore</value>

```

```
        </value>
        <description>The certificate file to be used with HTTPS</
description>
        <deviceName>COMMON_DEVICE_CFG</deviceName>
        <lineType>GENERIC_MESSAGE_BASED_CONNECTION</lineType>
    </communicationParameter>
    <communicationParameter>
        <label>USE_PROXY_SERVER</label>
        <value>
            <value>FALSE</value>
        </value>
        <description>Flag to turn On of Off use of a proxy web server for
HTTP/HTTPS communication</description>
        <deviceName>COMMON_DEVICE_CFG</deviceName>
        <lineType>GENERIC_MESSAGE_BASED_CONNECTION</lineType>
    </communicationParameter>
    <communicationParameter>
        <label>PROXY_HOST</label>
        <value>
            <value>fastweb.bell.ca</value>
        </value>
        <description>The host name or IP address of the proxy web server
(if one is to be used)</description>
        <deviceName>COMMON_DEVICE_CFG</deviceName>
        <lineType>GENERIC_MESSAGE_BASED_CONNECTION</lineType>
    </communicationParameter>
    <communicationParameter>
        <label>PROXY_PORT</label>
        <value>
            <value>80</value>
        </value>
        <description>Port number of proxy host (if one is to be used)</
description>
        <deviceName>COMMON_DEVICE_CFG</deviceName>
        <lineType>GENERIC_MESSAGE_BASED_CONNECTION</lineType>
    </communicationParameter>
</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 `/MICROSOFT_IPTV_1_0_IPTV_1_0`, copy the .sar file to the new directory and un-jar the sar file, as described by [Step 1](#) through [Step 4](#) in “[Modifying microsoft iptv 1_0_ne_config](#)” on page 15.
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 reinstall the cartridge using the new file.

Follow the instructions in [“Modifying microsoft_iptv_1_0_ne_config”](#) on page 15 for directions on how to load a new XML file.

