



# OCAP Architecture Guide

# Please Read

---

## Important

Please read this entire guide. If this guide provides installation or operation instructions, give particular attention to all safety statements included in this guide.

# Notices

---

## Trademark Acknowledgments

Cisco and the Cisco logo are trademarks or registered trademarks of Cisco and/or its affiliates in the U.S. and other countries. To view a list of cisco trademarks, go to this URL: [www.cisco.com/go/trademarks](http://www.cisco.com/go/trademarks).

CableCARD, OCAP, OpenCable, CableLabs and DOCSIS are trademarks or registered trademarks of Cable Television Laboratories, Inc.

Other third party trademarks mentioned are the property of their respective owners.

The use of the word partner does not imply a partnership relationship between Cisco and any other company. (1110R)

## Publication Disclaimer

Cisco Systems, Inc. assumes no responsibility for errors or omissions that may appear in this publication. We reserve the right to change this publication at any time without notice. This document is not to be construed as conferring by implication, estoppel, or otherwise any license or right under any copyright or patent, whether or not the use of any information in this document employs an invention claimed in any existing **or** later issued patent.

## Copyright

© 2006, 2012 Cisco and/or its affiliates. All rights reserved. Printed in the United States of America.

Information in this publication is subject to change without notice. No part of this publication may be reproduced or transmitted in any form, by photocopy, microfilm, xerography, or any other means, or incorporated into any information retrieval system, electronic or mechanical, for any purpose, without the express permission of Cisco Systems, Inc.

# Contents

---

## Preface

About This Guide.....	v
-----------------------	---

## Chapter 1 Introducing OCAP

Overview .....	1-1
What is OCAP?.....	1-2
OCAP Applications .....	1-3
OCAP Middleware .....	1-5
Signal and Delivery Process .....	1-6

## Chapter 2 OCAP System Implementation

Overview .....	2-1
SR 4.0 System Support.....	2-2
Hardware Components of an OCAP System.....	2-4
Software Components of an OCAP System .....	2-6

## Chapter 3 Customer Information ..... 3-1

# Preface

## About This Guide

---

### Introduction

This guide provides a high-level description of the OpenCable™ Application Platform (OCAP™) service offering. This guide also provides descriptions and illustrations of how the OCAP service works with the different elements of the Digital Broadband Delivery System (DBDS).

### Purpose

The purpose of this guide is to provide the following information:

- The basic concepts of OCAP
- A general idea about the three parts of the OCAP software
- An understanding of the OCAP application environment
- The types of applications and services supported by OCAP
- How Cisco plans to implement OCAP within the cable industry
- An understanding of how different hardware and software elements of the DBDS work within an OCAP system

### Audience

This guide is written for the following audiences:

- System operator marketing managers who are responsible for determining how service offerings should be packaged
- DBDS system administrators
- Digital Network Control System (DNCS) operators
- Cisco Services engineers
- Call-center personnel
- System operators who are responsible for maintaining the OCAP service
- Developers designing applications and services for the OCAP system
- Other parties interested in a general knowledge of OCAP

### Scope

This guide provides only high-level descriptive information that applies to OCAP. For more detailed descriptions, contact Cisco Services.

## About This Guide, Continued

---

### Related Publications

Refer to the following publications for more information:

- *System Release 4.0 Release Notes*, (part number 4001703, expected publish date: February 2006)
- *OCAP Installation and Upgrade Instructions*, (part number 732578, published January 2006)
- *TSBroadcaster User's Guide Cisco's OCAP™ Object Carousel Solution*, (part number 4011043, published January 2006)
- *Recommendations for Data Carousel Rate Management Technical Bulletin*, (part number 716377, published June 2005)
- *OpenCable Application Platform Specification*, OC-SP-OCAP10-I16-050803 (Available from CableLabs)
- *OpenCable Common Download Specification*, OC-SP-CD-IF-I08-040831 (Available from CableLabs)

### For More Information

This reference table includes a list of World Wide Web (WWW) links to technical publications containing useful information for digital systems. The links are listed alphabetically by company or feature name.

Company or Feature	Link
CableLabs	<a href="http://www.cablelabs.com">www.cablelabs.com</a>
SCTE	<a href="http://www.scte.org">www.scte.org</a>

### Document Version

This is the second release of this guide.

# Chapter 1

## Introducing OCAP

### Overview

---

#### Introduction

The OCAP software uses a common method of delivering applications and data through the cable television network system.

This chapter provides an overview of the OCAP software, including the basic concepts of the OCAP applications, middleware, and signal and delivery process.

#### In This Chapter

Topic	See Page
What is OCAP?	1-2
OCAP Applications	1-3
OCAP Middleware	1-5
Signal and Delivery Process	1-6

# What is OCAP?

---

## Introducing OpenCable

In recent years, the cable industry has worked toward developing a software platform standard to run interactive applications over the cable television network system. In response to the 1996 Telecommunications Act, CableLabs® and cable service providers created the OpenCable Initiative to set standards for retail cable deployment. One of these standards, OpenCable Application Platform (OCAP), was developed to serve as a common software standard for the cable industry.

OCAP is based on the European Digital Video Broadcasting (DVB) Multimedia Home Platform (MHP) specification and was created as a Java-based middleware and application delivery standard. OCAP assists the industry with the delivery of digital signaling and interactive applications to cable subscribers. Software developers can use OCAP to create applications and other programs for deployment over host devices, such as set-top boxes and televisions. The delivery of applications and data to hosts is defined by the OCAP specification.

## Three Parts of OCAP

The OCAP software uses a common method of delivering applications and data through the cable television network system. The OCAP software consists of three parts:

- OCAP Applications
- OCAP Middleware
- Signal and Delivery Process

One way to understand how these three parts work together is to understand how the pieces are dependant upon one another in the overall OCAP design. The OCAP applications are at the top of the OCAP software (stack). The applications are designed to run on any platform that supports OCAP-compliant middleware. The OCAP middleware is a software layer built on top of an operating system (OS) interface and software drivers. OCAP uses a common method of delivering applications and data through the cable television network system (signal and delivery process).

The following drawing shows the layers of the OCAP software stack.

OCAP Application
OCAP Middleware
OS
Drivers

# OCAP Applications

---

## Overview

OCAP applications are written using Java and the standard OCAP Application Programming Interfaces (APIs) in a way that they can be run on any platform that supports OCAP-compliant middleware.

## Basic OCAP Application Concepts

An OCAP application consists of compiled Java code. Applications may have different levels of priority and privileges.

OCAP applications may be bound or unbound to a channel service. Bound or unbound relates to whether the application is associated with a channel service (for example, a specific television program). Bound applications are loaded only when the program is tuned. Unbound applications are loadable regardless of which program is tuned.

In addition to all of the OCAP standard interfaces, a privileged application can use the Cisco Point-of-Deployment handler (PODHandler) APIs for network-specific functions outside of OCAP standards.

## Bound Applications

A bound application is an application that is bound to program content. The application is terminated when the user tunes to another channel service or when the program ends. Bound applications are typically signaled and delivered on the same data transport stream as the video. The life of the application is typically tied to a television program.

An application that lets you participate in interactive voting, such as voting for a contestant on a television show, is an example of a bound application.

## Unbound Applications

An unbound application is an application that is not bound to program content. Unbound applications are signaled out-of-band (OOB) and either carried using a Digital Storage Media – Command and Control (DSM-CC) object carousel for delivery on a QAM modulator (QAM) transport stream or on a QPSK or CMTS path; or the applications are carried through DOCSIS TFTP.

Examples of unbound applications are an electronic program guide navigator or an instant messaging application that lets viewers send instant messages to each other while watching television shows.

# OCAP Applications, Continued

---

## Applications Security

CableLabs is considered the trusted security source that verifies the OCAP middleware. CableLabs is responsible for supplying the core verification root certificate. This certificate is used to authenticate OCAP applications that have a security certificate. Certain privileges are only available for authenticated OCAP applications.

## Monitor Application Functionality

The monitor application is an OCAP unbound application using standard OCAP interfaces. Much like the resident application in set-tops, the monitor application negotiates security requirements for premium applications, manages application life cycles, and referees when multiple applications request the same resource at the same time.

The monitor application is an optional OCAP component with the highest level of security. It ultimately determines which applications may run in the OCAP system.

Like all unbound applications, the monitor application is signaled using the Extended Application Information Table (XAiT). The monitor application is distinguished in the priority and permissions given to it by the XAIT. These permissions allow the application to “call” OCAP functions that influence the operation of other applications.

The cable service provider or the application vendor may sign a certificate to an application. Based on the presence of a signature, the application receives or is denied security privileges, subject to additional restrictions placed by the monitor application. The OCAP object carousel is responsible for carrying the application signatures.

**Note:** For more information, refer to **Section 6** of the *OpenCable Security Specification*, OC-SP-SEC-I05-040831 (Available from CableLabs).

# OCAP Middleware

---

## Overview

The OCAP middleware includes a Java Virtual Machine (JVM) and the required OCAP Java APIs. The OCAP middleware also includes support for the CableCARD™ module.

Cisco has created the Axiom™ OCAP Middleware product. The OCAP middleware is a software layer built on top of the PowerTV® Operating System (OS). The middleware includes a Java Virtual Machine and implements the standardized OCAP APIs.

**Note:** Other OCAP systems may utilize a different OS and provide the same level of OCAP compatibility.

# Signal and Delivery Process

---

## Overview

The signaling and delivery process sends files (applications or data) over a cable network. The applications are then launched by the OCAP implementation (middleware) and delivered by the OCAP object carousel.

Different signaling methods are used for bound and unbound applications. The Applications Information Table (AIT) method is used for signaling bound applications. Unbound applications use the XAIT for signaling.

Bound applications are usually delivered through a QAM. Unbound applications may be delivered through a QAM or through DOCSIS TFTP. Another carousel may be delivered through the OOB channel, but this is usually for data delivery.

## AIT Method

The AIT is carried in the same QAM inband transport stream as the content to which it is bound. The AIT is defined in the Program Map Table (PMT) of the associated data content.

Because bound applications are associated with specific content, the AIT is transmitted on the same transport stream as the content, which should also include the object carousel (but is not required). This is similar to how MHP is implemented.

## XAIT Method

The XAIT is carried over the OOB channel with the service information data. The unbound application is delivered in either the IB or OOB stream, or the unbound application is carried through DOCSIS TFTP.

**Note:** Due to the low bit rate of the OOB stream, Cisco does not recommend delivery of unbound applications through the OOB stream.

# Chapter 2

## OCAP System Implementation

### Overview

---

#### Introduction

Cable systems use a combination of hardware and software components to send OCAP services to OpenCable hosts. The selection of these components depends upon what kind of system you are using to implement the OCAP services. Cisco provides support for the OCAP middleware standard with System Release 4.0 (SR 4.0).

This chapter provides an overview of the OCAP features supported by SR 4.0 and the hardware and software components used in the implementation of a Cisco OCAP system.

#### In This Chapter

This chapter contains the following topics.

Topic	See Page
SR 4.0 System Support	2-2
Hardware Components of an OCAP System	2-4
Software Components of an OCAP System	2-6

# SR 4.0 System Support

---

## Overview

With SR 4.0, cable service providers can choose to add OCAP applications and OpenCable network components. Customers with SR 4.0 must purchase a license to add the object carousel support for the OCAP platform to their system.

## Object Carousel Output Types

The OCAP object carousel has the following three output types:

- Output over Ethernet to the DNCS for the XAIT kit (XAIT Delivery)
- Output over ASI (or Gigabit Ethernet) to the QAM modulator for the OCAP object carousel that is transmitted inband
- Output over Ethernet to a QPSK modulator for OOB OCAP object carousel (or over DOCSIS® Set-Top Gateway [DSG])

### XAIT Delivery

The object carousel builds an XAIT kit. The XAIT kit consists of a zip file. The zip file contains two files: a binary file of the XAIT and an XML file containing the binary file name and configuration information. The object carousel interface delivers the XAIT kit to the DNCS anytime an unbound OCAP application is added or deleted.

### OpenCable Common Download

Cisco designed OCAP to support the OpenCable Common Download method for firmware download. The host firmware data may be downloaded through the Common Download data carousel. The DNCS OCAP Common Download Manager (ocdlManager) application provides the common download function for the DNCS. Signaling is done using the SCTE 28 System Control resource (same as the *OpenCable Common Download Specification*).

The ocdlManager uses a Web User Interface (WUI) that provides the ability to load and playout a Common Download Code Version Table (CDVT) file set.

The CableCARD module implements common download signaling to use the OOB forward data channel (FDC), either QPSK or DSG. This implementation for upgrading an OpenCable-compatible host meets the requirements of the *OpenCable Common Download Specification*.

### DOCSIS Set-Top Gateway

The DOCSIS Set-Top Gateway (DSG) data delivery method is derived from a CableLabs specification that describes how the Cable Modem Termination System (CMTS) forwards broadcast OOB data to OpenCable hosts.

**Note:** For more information, refer to the *DOCSIS® Set-top Gateway (DSG) Interface Specification*, CM-SP-DSG-I03-041124 (Available from CableLabs).

DSG allows cable operators to move the CableCARD module's QPSK OOB downstream signaling to the DOCSIS channel. The headend uses the OOB signal to send encryption and authorization signals down to the CableCARD module. Specifically, the DSG OOB signal transmits from the headend to a cable modem termination system (CMTS). From the CMTS, the OOB signal is transmitted to an OpenCable host containing an embedded DOCSIS cable modem. The host forwards the OOB data to the CableCARD module.

With DSG, OOB data flows from the DNCS to one or multiple IP addresses. Cable operators are responsible for defining the MAC addresses. The hosts dynamically learn the MAC addresses through a downstream channel detector (DCD) message.



## Hardware Components of an OCAP System, Continued

---

### DNCS

The DNCS allows you to authorize and deauthorize hosts to receive specific services, to modify services, and to configure and control other components in the DBDS. A direct ASI card in the DNCS delivers BFS traffic directly to a QAM device or multiplexer. The Common Download data carousel utilizes an additional ASI card to deliver the carousel directly to a QAM device or multiplexer.

### OCAP Object Carousel

The OCAP object carousel transports data modules and application server processes to the host. In the OCAP system environment, the OCAP object carousel takes the place of the BFS BIG device.

You may have multiple OCAP object carousels installed in your system for delivery of application and data files to OCAP hosts.

**Note:** The BFS is still required for the PowerKEY CableCARD module.

### QPSK Modulator/Demodulator

The QPSK modulator works with the QPSK demodulator and the host to provide forward signaling and a reverse communications path for interactive video and data services. The QPSK modulator and demodulator convert digital bit streams to RF format and RF signals to digital bits, respectively.

### QAM / MQAM / GOAM / GoQAM Modulator

These devices use QAM techniques to modulate a digital signal onto a hybrid fiber coax (HFC) network to deliver voice, video, and data to a host.

### PowerKEY CableCARD Module or Virtual CableCARD Module

The PowerKEY CableCARD module, or POD module, uses the PCMCIA card design and inserts into a slot on a host device (either a set-top or cable-ready television) to provide conditional access to secure digital content.

The CableCARD module routes the XAIT table to the OCAP host so that the monitor application can be launched from the OCAP host.

### OCAP Host

Cisco provides two OCAP host platform environments using either the Explorer® 8300HD™ Set-Top with a PowerKEY virtual POD or an OCAP-compliant host device with a PowerKEY CableCARD module installed.

# Software Components of an OCAP System

---

## Overview

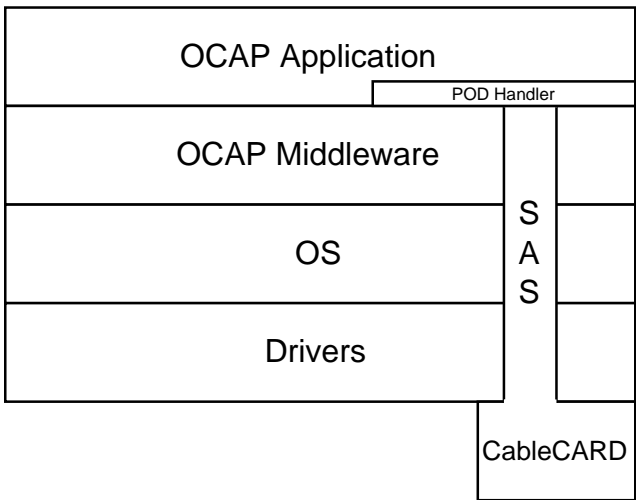
This section describes the software components of an OCAP system. The software components are arranged in the following order:

- OCAP Application
- OCAP Middleware
- OS
- Drivers

The POD handler runs as a part of the monitor application to interface for the application layer. The Specific Application Support (SAS) resource connects the OCAP application layer to the CableCARD module.

## OCAP Software Components Diagram

The following illustration shows the OCAP system software stack, including the POD handler and SAS connection.



# Chapter 3

## Customer Information

### Overview

---

#### If You Have Questions

If you have technical questions, call Cisco Services for assistance. Follow the menu options to speak with a service engineer.

Access your company's extranet site to view or order additional technical publications. For accessing instructions, contact the representative who handles your account. Check your extranet site often as the information is updated frequently



Cisco Systems, Inc.  
5030 Sugarloaf Parkway, Box 465447  
Lawrenceville, GA 30042

678 277-1120  
800 722-2009  
[www.cisco.com](http://www.cisco.com)

This document includes various trademarks of Cisco Systems, Inc. Please see the Notices section of this document for a list of the Cisco Systems, Inc. trademarks used in this document.

Product and service availability are subject to change without notice.

© 2006, 2012 Cisco and/or its affiliates. All rights reserved.

September 2012 Printed in USA

Part Number 738184 Rev B