



CISCO CONFIDENTIAL

CHAPTER 29

Using Per-Product Services

Per-Product Services are additional, value-added services that individual products can choose to include based on their particular requirements. Installations of Per-Product Services are normally not shared among the applications installed on a single server.

The following topics discuss the CWCS Per-Product Services and the components you need to include to use them as part of your applications:

- [Understanding Per-Product Services](#)
- [About the Per-Product Services Components](#)

Understanding Per-Product Services

The Per-Product Services discussed in this section are always installed and run by individual applications. They form a set of advanced services that add value to applications by either meeting requirements for advanced functions in CiscoWorks applications, or allowing for variation in how these features are implemented. [Table 29-1](#) summarizes these feature requirements and the corresponding Per-Product Services designed to meet them.

Note that some Per-Product components (such as UII) are also used in the CWCS-R distribution. This is done to support certain Shared Services. Note that these CWCS-R-distributed Per-Product components are for CWCS use only. Applications may *not* share Per-Product components installed with CWCS-R. Application teams must install their own copies of Per-Product components with their applications, and they may not modify the source of or substitute products intended to perform the same functions as any Per-Product component distributed with CWCS-R.

As noted in the table, many Per-Product components are provided as part of the CWCS-SRC distribution. Application teams may modify CWCS-SRC-distributed Per-Product components for their own requirements, and install them with their applications. Note that it is a requirement of CWCS-SRC distribution that application teams who modify a CWCS-SRC component's source are responsible for providing their own support on these components. Application teams must also provide fully commented copies of their modified source to the CWCS Team.

Per-Product Services often represent the future direction of Common Services. Where possible, you should plan your application to adopt them as standard. For example: the CiscoWorks Home Page, which provides GUI access to applications, has been adopted as the standard GUI, entirely replacing the old CMF Desktop.

CISCO CONFIDENTIAL**Table 29-1 Per-Product Services: Feature Requirements and Services Map**

Category	Requirement	Description	Type	Per-Product Service
Graphic User Interface	User Interface Support	Support a web-based launch point that aggregates all applications.	CWCS-SRC	The launch point is provided by the CiscoWorks Home Page (see the “About the CiscoWorks Home Page Component” section on page 6-5), which in turn requires the User Interface Infrastructure (UII). See the “About the User Interface Infrastructure” section on page 29-7.
	Device Selector	Let users see and select devices and groups of objects	CWCS-SRC	Part of the UII; comes in several different forms. See the “About the User Interface Infrastructure” section on page 29-7.
	Graphing Service	Allow GUI display of graphs from user data.	CWCS-SRC	A Graphing Framework is provided as part of the UII. See the “About the User Interface Infrastructure” section on page 29-7.
	Group Management	Let users or administrators create and maintain groups of objects (e.g., devices or user IDs).	CWCS-SRC	See the “About the Object Grouping Service (OGS) Components” section on page 29-4.
Operations	Device Package Update	Support delivery of incremental device packages.	CWCS-R	Provided by PSU with VDS. See the “About the Package Support Updater (PSU) Components” section on page 29-5
	Incremental Device Support	Permit incremental device support.	CWCS-SRC	Provided by the CIDS package. See the “About the Common Incremental Device Support (CIDS) Component” section on page 29-5
	IPC/RPC Support	Provide support for Inter-Process Communications (IPC), In-Process Calls, and Remote Procedure Calls (RPC)	CWCS-SRC	CSTM provides full IPC/RPC support. See the “About the Common Services Transport Mechanism (CSTM) Components” section on page 29-4.
	Licensing	Provide flexible management of licenses, including multiple keys.	CWCS-R	See the “About CWCS Licensing” section on page 29-6.
	SOAP Support	Support SOAP encoding.	CWCS-SRC	CSTM uses SOAP internally and exposes northbound APIs in a SOAP-based format. See “About the Common Services Transport Mechanism (CSTM) Components” section on page 29-4
	Virtual Download Service	Download and install incremental device packages.	CWCS-R	Provided by PSU with VDS. See the “About the Package Support Updater (PSU) Components” section on page 29-5

CISCO CONFIDENTIAL**Table 29-1 Per-Product Services: Feature Requirements and Services Map (continued)**

Category	Requirement	Description	Type	Per-Product Service
Utilities	DOM Parser	Provide Document Object Model parser support for Core, MC and Kilner applications	CWCS-R	JDOM 1.0 Beta 8 is supported. See the "" section on page 29-8.
	Email Support	Provide utilities that support email generation	CWCS-R	Utilities provided include JavaMail 1.2 and Blat for Windows NT. Solaris email is used on Solaris platforms. See the "" section on page 29-8.
	JRE Support	Support multiple versions of the Java Runtime Environment.	CWCS-R	JRE versions 1.2.2, 1.3.1, and 1.4 are supported. See the "" section on page 29-8.
	Logging Service	Provide utilities that support creation and management of application logs.	CWCS-R	Utilities provided include Log4J 1.1.3, coreLog4c, CoreLogger, CMF logging, and JGL 3.1. See the "" section on page 29-8.
	SNMP Library	Support SNMP calls from Java applications.	CWCS-SRC	The latest version of the SNMPOnJava library supports SNMPv3. See the "" section on page 29-8.
	XML Parser	Support a variety of XML parsers.	CWCS-R	Libraries provided include standard Xerces Java 1.4.3 and 1.4.4 library, JAXP, Crimson, Xerces C++ 1.5.1, and IBM4J. See the "" section on page 29-8.
	XSLT Support	Support a variety of XML Stylesheet processors.	CWCS-R	Libraries provided include LotusXSL 0.16.3, Xalan Java 2.2.d9, Xalan C++ 1.5.1, and Xalan 2.4.1. See the "" section on page 29-8.

About the Per-Product Services Components

The following topics provide basic information for each of the Per-Product Service components:

- [About the Object Grouping Service \(OGS\) Components](#)
- [About the Common Services Transport Mechanism \(CSTM\) Components](#)
- [About the Package Support Updater \(PSU\) Components](#)
- [About the Common Incremental Device Support \(CIDS\) Component](#)
- [About CWCS Licensing](#)
- [About the User Interface Infrastructure](#)

Each topic includes:

- Basic information about the component's purpose and features.
- Usage guidelines. Where these guidelines require a separate chapter in this *Guide*, a pointer to that chapter is supplied.
- Pointers to Java, C++, Perl or other references and code samples supplied as part of the SDK.
- The names of the packages on which the component depends.
- Where they exist: The names of utilities that help you troubleshoot or use the component.

For the same information for Shared Services, see [Chapter 6, "Using Shared Services"](#).

CISCO CONFIDENTIAL**About the Object Grouping Service (OGS) Components**

The Object Grouping Service provides a generic framework for creating, managing, and sharing hierarchical groups of objects. These objects can be any type of data object, from network devices to user IDs, capable of being grouped based on shared attributes. The OGS framework provides tools to define groups useful to your application, and to supply them in predefined form with your application.

For guidelines to follow when including OGS with your application, see [Chapter 30, “Using Object Grouping Services”](#). OGS is functionally dependent on the packages shown in [Table 29-2](#).

Table 29-2 OGS Package Dependencies

Package Name	Description
CSCOapch/apache	Core Apache Web Server with SSL
CSCOdb/db	Common Services Database (includes diskwatcher, DB wrappers)
CSCOess/ess	Event Services Software (includes Tibco event bus)
CSCOgrid/grid	Grid
CSCOjaws/jaws	JSCAPE JavaAWT for widgetd
CSCOjcht/jchart	Jchart Java Class
CSCOjpwr/jpwr	JSCAPE Power Search Classes
CSCOjre2/jre2	CMF JRE 1.2.2
CSCOlog4j/log4j	Log4j Logging Framework
CSCOmmd/dmgt	Daemon Manager (Process Manager)
CSCOperl/perl	Perl Support
CSCOswing2/swng2	Java Swing2 Package
CSCOTmct/tomcat	Tomcat Servlet Engine
CSCOweb/web	Web Services: Apache, OpenSSL, ModSSL
CSCOxln/xalan	Apache Xalan XSLT Processor
CSCOxrcs/xerces	Apache Xerces XML Parser
CSCOxsl/lotusxsl	Lotus XSL Engine Classes
CTM	Common Services Transport Mechanism
GS	Graphing Service
OGS	Object Grouping Service
SVC	NT Services (includes TFTP, RSH/RCP, CRM Logger, Blat mail for NT)
UII	User Interface Infrastructure

About the Common Services Transport Mechanism (CSTM) Components

CSTM (formerly Common Transport Mechanism, or CTM) is a simple, platform-agnostic method for performing all types of inter-program communications. It does not require that sender or receiver applications implement proprietary standards like CORBA or DCOM, use the same object model, have full knowledge of message contexts, or avoid encoding metadata. Instead, it abstracts these

CISCO CONFIDENTIAL

communications, provides a single API for dealing with all of them, and bases the API on well-known non-proprietary communications and data standards, including XML, Serialized Java Objects, HTTP, sockets, and standard binary and SOAP encoding.

For guidelines to follow when including CSTM with your application, see [Chapter 31, “Using the Common Services Transport Mechanism”](#). CSTM is functionally dependent on the packages shown in [Table 29-3](#),

Table 29-3 *CSTM Package Dependencies*

Package Name	Description
CSCOjre2/jre2	CMF JRE 1.2.2
CSCOjrun/jrun	JRun Servlet Engine
CSCOlg4j/log4j	Log4j Logging Framework
CSCOmd/dmgt	Daemon Manager (Process Manager)
CSCOperl/perl	Perl Support
CSCOweb/web	Web Services: Apache, OpenSSL, ModSSL
CSCOxln/xalan	Apache Xalan XSLT Processor
CSCOxrcs/xerces	Apache Xerces XML Parser
CTM	Common Services Transport Mechanism

About the Package Support Updater (PSU) Components

The PSU (and associated Virtual Download Service, or VDS) allows your application to:

- Check for software and device support updates.
- Download these updates and related dependent packages to the application’s server file system.
- Install them.

For guidelines to follow when including PSU/VDS with your application, see [Chapter 32, “Using Package Support Updater”](#). PSU is functionally dependent on the packages shown in [Table 29-4](#).

Table 29-4 *PSU Package Dependencies*

Package Name	Description
CSCOweb/web	Web Services: Apache, OpenSSL, ModSSL
PSU/VDS	Package Support Updater/Virtual Download Service

About the Common Incremental Device Support (CIDS) Component

Common Incremental Device Support (CIDS) provides a method of adding device information to any network management application without requiring re-installation of the entire product. CIDS provides an application layer that encapsulates all device information and update mechanisms, permitting applications that include a CIDS layer to be incrementally updatable for new instances of existing devices, as well as new types of devices.

For guidelines to follow when including CIDS with your application, see [Chapter 33, “Using Common Incremental Device Support”](#). CIDS is functionally dependent on the packages shown in [Table 29-5](#).

CISCO CONFIDENTIAL**Table 29-5 CIDS Package Dependencies**

Package Name	Description
CIDS	Common Incremental Device Support
CSCOjava/java	Core JRE 1.3.1 JARs
CSCOjgl/jgl	ObjectSpace JGL Classes
CSCOlg4j/log4j	Log4j Logging Framework
CSCOsnmp/snmp	Java SNMP APIs
CSCOweb/web	Web Services: Apache, OpenSSL, ModSSL

About CWCS Licensing

The CWCS Licensing API allows your application to:

- Install and update licenses
- Create, manage, retrieve and restore license information
- Access FLEXlm license utilities.

The API supports:

- Evaluation-onlylicensing
- Full-purchase licenses.
- Resource-limited licenses, which specify (for example) the number of devices an application can manage.
- Feature licenses, which grant the right to use features within an application.
- Temporary use of a PIN to validate use of a feature.

The API does not explicitly support other kinds of license models, such as node-locked licenses, floating licenses, and counted licenses. However, the API provides access to the FLEXlm toolkit for applications that must implement a licensing model with these kinds of special requirements.

The API provides:

- A repository to store PINs and PAKs.
- An API to install licenses and to retrieve license information, including PIN/PAK.
- A license administration GUI.
- The FLEXlm license toolkit.
- A license backup and restore function.

For guidelines to follow when including the Licensing API with your application, see [Chapter 34, “Using the Licensing APIs”](#). The Licensing API is functionally dependent on the packages shown in [Table 29-6](#).

Table 29-6 Licensing API Dependencies

Package Name	Description
CSCOapch/apache	Core Apache Web Server with SSL
CSCOchlp/chlp	Core Help Files

CISCO CONFIDENTIAL**Table 29-6** Licensing API Dependencies (continued)

Package Name	Description
CSCOcore/core	Core Modules
CSCOcsdb/ccsdb	Core Database
CSCOess/ess	Event Services Software (includes Tibco event bus)
CSCOjava/java	Core JRE 1.3.1 JARs
CSCOjre2/jre2	CMF JRE 1.2.2
CSCOjrun/jrun	JRun Servlet Engine
CSCOmd/dmgt	Daemon Manager (Process Manager)
CSCOperl/perl	Perl Support
CSCOsjre/sunjre	Core JRE 1.3.1 libraries (.so, .font, etc.)
CSCOmct/tomcat	Tomcat Servlet Engine
CSCOweb/web	Web Services: Apache, OpenSSL, ModSSL
CSCOxln/xalan	Apache Xalan XSLT Processor
CSCOxrcs/xerces	Apache Xerces XML Parser
SNMPv3 support	SNMPv3 Support for AuthNoPriv Mode
SVC	NT Services (includes TFTP, RSH/RCP, CRM Logger, Blat mail for NT)

About the User Interface Infrastructure

The User Interface Infrastructure (UII) is a web-application framework for creating application GUIs that conform to Cisco's User Experience Initiative (UE) guidelines (available at <http://picasso>). Based on the Apache/Jakarta STRUTS framework, the UII provides:

- A complete set of tools for creating displays, dialogs, tables, lists, wizards, buttons and icons.
- A Reporting Framework that allows you to create and display all types of reports as web pages.
- A Graph Framework that allows you to create and display line, pie, bar, stacked bar, and area graphs.

The UII is not documented in this *Guide*. Guidelines for using these tools and implementing a UII-compliant GUI are provided in the *SDK Developer's Guide for UI Infrastructure* (release 6.1 or later). This document is available

- As a PDF file within the SDK WAR file, or within EDCS (Release 6.1 of the Guide is available as EDCS-275335). You can use the PDF to print a hardcopy version.
- As HTML-based online help. The HTML version is available from the User Experience web site at <http://picasso>. Select **Technology > UII Releases**.

The UII is functionally dependent on the packages shown in [Table 29-7](#). For a working model of the UII, see <http://uii.cisco.com>.

Table 29-7 UII Package Dependencies

Package Name	Description
CSCOgrid/grid	Grid
CSCOhelpDM/cdone	CMF Help Files

CISCO CONFIDENTIAL**Table 29-7** *UII Package Dependencies (continued)*

Package Name	Description
CSCOjawt/jawt	JSCAPE JavaAWT for widgetd
CSCOjcht/jchart	Jchart Java Class
CSCOjpwr/jpwr	JSCAPE Power Search Classes
CSCOjrun/jrun	JRun Servlet Engine
CSCOmd/dmgt	Daemon Manager (Process Manager)
CSCOperl/perl	Perl Support
CSCOswng2/swng2	Java Swing2 Package
CSCOmct/tomcat	Tomcat Servlet Engine
CSCOweb/web	Web Services: Apache, OpenSSL, ModSSL
CSCOxln/xalan	Apache Xalan XSLT Processor
CSCOxml4j/xml4j	IBM XML4J Parser
CSCOxrcs/xerces	Apache Xerces XML Parser
CSCOxsl/lotusxsl	Lotus XSL Engine Classes
GS	Graphing Service
UII	User Interface Infrastructure