HP Client Automation

for the AIX, HP-UX, Linux, Solaris, and Windows® operating systems

Release Version: 7.80

Essentials Guide

Manufacturing Part Number: None

Document Release Date: November 2009

Software Release Date: November 2009

Legal Notices

Warranty

The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

The information contained herein is subject to change without notice.

Restricted Rights Legend

Confidential computer software. Valid license from HP required for possession, use or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

Copyright Notices

© Copyright 1993-2009 Hewlett-Packard Development Company, L.P.

Trademark Notices

The Apache Software License, Version 1.1
This product includes software developed by the Apache Software Foundation (http://www.apache.org//)
Copyright © 1999-2001 The Apache Software Foundation. All rights reserved.

Linux is a registered trademark of Linus Torvalds.

Microsoft®, Windows®, and Windows® XP are U.S. registered trademarks of Microsoft Corporation.

OpenLDAP is a registered trademark of the OpenLDAP Foundation.

PREBOOT EXECUTION ENVIRONMENT (PXE) SERVER
Copyright © 1996-1999 Intel Corporation.

TFTP SERVER
Copyright © 1983, 1993
The Regents of the University of California.

OpenLDAP
Copyright 1999-2001 The OpenLDAP Foundation, Redwood City, California, USA.
Portions Copyright © 1992-1996 Regents of the University of Michigan.

OpenSSL License
Copyright © 1998-2001 The OpenSSLProject.

Original SSLeay License
Copyright © 1995-1998 Eric Young (eay@cryptsoft.com)

DHTML Calendar
Copyright Mihai Bazon, 2002, 2003

Documentation Updates

The title page of this document contains the following identifying information:

- Software Version number, which indicates the software version.
  - The number before the period identifies the major release number.
  - The first number after the period identifies the minor release number.
  - The second number after the period represents the minor-minor release number.
- Document Release Date, which changes each time the document is updated.
- Software Release Date, which indicates the release date of this version of the software.

To check for recent updates or to verify that you are using the most recent edition, visit the following URL:

http://h20230.www2.hp.com/selfsolve/manuals

This site requires that you register for an HP Passport and sign-in. To register for an HP Passport ID, go to:

http://h20229.www2.hp.com/passport-registration.html

Or click the **New users - please register** link on the HP Passport login page.

You will also receive updated or new editions if you subscribe to the appropriate product support service. Contact your HP sales representative for details.

Table 1 Documentation changes

Chapter	Version	Changes
All	7.80	Updated product version number and release date for 7.80.

Support

You can visit the HP Software support web site at:

www.hp.com/go/hpsoftwaresupport

This web site provides contact information and details about the products, services, and support that HP Software offers.

HP Software online software support provides customer self-solve capabilities. It provides a fast and efficient way to access interactive technical support tools needed to manage your business. As a valued support customer, you can benefit by using the support site to:

- Search for knowledge documents of interest
- Submit and track support cases and enhancement requests
- Download software patches
- Manage support contracts
- Look up HP support contacts
- Review information about available services
- Enter into discussions with other software customers
- Research and register for software training

Most of the support areas require that you register as an HP Passport user and sign in. Many also require an active support contract. To find more information about support access levels, go to the following URL:

http://h20230.www2.hp.com/new_access_levels.jsp

To register for an HP Passport ID, go to the following URL:

http://h20229.www2.hp.com/passport-registration.html

Contents

1 Introduction

At the end of this chapter, you will:

- Know the scope and content of this book.
- Be familiar with terminology associated with HP Client Automation (HPCA).

About Client Automation Technology

Client Automation technology provides high levels of adaptability, flexibility, and automation. Adaptability comes from the embedded intelligence of platform-independent object-oriented technology. Flexibility is provided by the media-independence of Client Automation technology that enables content to be easily revised and customized. And Client Automation solutions automate digital asset management across virtually any kind of network. The following bullets describe each of these distinct capabilities that are essential to Client Automation technology.

- **The Embedded Intelligence of Object-Oriented Technology**
  Object-oriented technology transforms software and content from file-based media into self-aware, platform-independent, intelligent objects that automatically assess the environment into which they are deployed, and personalize, install, update, and repair themselves accordingly. In other words, as intelligent objects, they know what they need for a particular device or user, where to get it, when they need to change, how to change themselves, and how to repair themselves.
- **Revisable Packaging for Revisable Content**
  Client Automation technology enables revision and customization of software and content at any midstream point in the publisher-to-subscriber deployment process. Because Client Automation technology transforms software and content into objects, these objects can be easily modified midstream—subtracted from, added to, or reconfigured—simply by packaging them with other objects or new configuration information. Revisable packaging allows value-added service providers and IT administrators to customize standard published software offerings for the specific needs of their users without having to unpack and repackage the software.
- **Self-Managing Infrastructure**
  The object-oriented intelligence of Client Automation technology incorporates a self-managing infrastructure. This capability begins with network-independence. Client Automation technology supports any deployment environment, whether client/server, local, wide or virtual area network, intranet, extranet, or the Internet. Furthermore, Client Automation supports whatever distribution media make sense for the target audience and the provider (which might be a software publisher, application service provider (ASP), Internet service provider (ISP), provider of enterprise application integration services, e-business integrator, e-commerce component provider, or in-house IT administrator).

In the Internet age in which software is fundamental to the ability of businesses to compete, change is a constant state, and audience diversity has grown beyond the capacity of older technologies to manage. Client Automation technology provides the necessary automation, adaptability, and flexibility to solve the software management challenge.

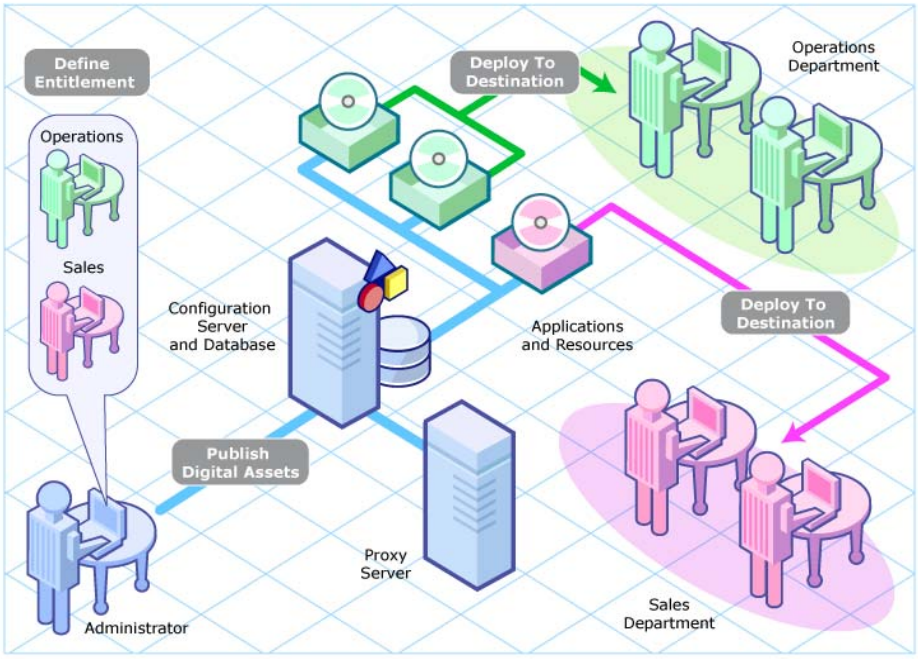
Why Use Client Automation?

Client Automation manages the distribution of data based on your **desired state**. The desired state is the condition of a device defined by configuration parameters set in the Configuration Server Database (CSDB).

At a minimum, the desired state includes the following five elements:

- **Users**
  The identity of the devices or subscribers being managed.
- **Applications**
  The software that is being managed.
- **Application Files**
  The components that make up the applications.
- **Deployment Source**
  The Client Automation product on which the application components are stored (such as Proxy Server or Configuration Server) and from which they are distributed.
- **Deployment Destinations**
  The devices (such as desktop computers, PDAs, and laptops) to which the application and its files are distributed.

Figure 1 Elements in the desired state



Use Client Automation to manage all of these components. You publish **packages** of data, determine entitlement policy, and define how the packages are deployed.

Terminology

The following terms are used frequently in this publication.

administrator

The person who uses the Client Automation Administrator to configure and maintain the Client Automation environment.

Administrator

A set of tools (Agent Explorer, Packager, Publisher, CSDB Editor, Screen Painter, and the AMP Editor) that you use to manage the Client Automation environment.

agent

The agent software (such as the Application Self-service Manager, Application Manager, and Inventory Manager) that runs on the managed device and communicates with the Configuration Server.

agent computer

A computer (workstation or server) that has the Client Automation agent software installed on it. It may also be referred to as a device.

agent connect

The process by which a managed device communicates with the Configuration Server.

Configuration Server

In conjunction with the CSDB, a server that stores, manages, and distributes application package information, and manages policy relationships and information about managed devices. This server is the only product that is mandatory in a Client Automation environment; without it, the infrastructure will not function.

Configuration Server Database (CSDB)

An object-oriented database that stores all the information needed to manage assets on a device, including the software and/or data that Client Automation distributes, the policies that determine which users are entitled to which packages, and security and access rules for administrators. It has a hierarchical structure containing four levels: files, domains, classes, and instances.

desired state

The condition of a device as defined by the configuration parameters you set in the CSDB. These parameters include software, operating system, and policy.

device

A piece of hardware, such as a computer or ATM, that may be either a managed device or a target device.

managed device

A computer, ATM, or other piece of hardware, that is managed by Client Automation.

package

A unit of software or data that can be published to the CSDB.

policy

A designation of the services to which a user, an agent computer, or a managed device is entitled.

resolution

The process by which the object attribute values on a managed device are replaced with those that are required to achieve its desired state.

service

A group of related packages, methods, or behaviors organized into manageable units.

target device

A workstation or server on which you want to install, replace, or update software.

user

The person who uses managed applications on a managed device.

About this Guide

The purpose of this guide is to describe essential Client Automation concepts and the benefits of the Client Automation components. For information about the installation and configuration of Client Automation products, go to the HP Software support web site or the appropriate Client Automation publication.

IMPORTANT NOTE:

With the introduction of Client Automation, version 7.20**,** HP has simplified and streamlined the installation, configuration, and use of our product by introducing two new server components: the *Core and the Satellite*. These components provide an end-to-end experience that encompasses all of our product capabilities.

The new **Core** and **Satellite** (see the *HPCA Core and Satellite Getting Started and Concepts Guide* in the `Documentation` directory of the HPCA media) are available to new customers who use **Windows Servers** as their primary infrastructure platforms or existing customers who are migrating from a version 7.20 Core and Satellite implementation.

Existing customers, and new customers who require **UNIX** infrastructure support, should consult the *HPCA Configuration Server, Portal, and Enterprise Manager Getting Started Guide* for information on alternative methods for installing, configuring, and using HP's **Client Automation** infrastructure.

The remaining chapters of this book contain the following topics:

- Chapter 2, Products and Benefits
  discusses Client Automation products, their benefits, and their place in relation to other Client Automation components.
- Chapter 3, Essential Processes
  describes some of the essential Client Automation processes. It helps you gain an understanding of how Client Automation works.

Client Automation Publications

While this book provides an overview of Client Automation products and processes, it is merely a starting point. You can find more information on all of the Client Automation products on the HP Software support web site and the Client Automation DVD media. These publications describe how to prepare data for management on your enterprise's computers. See Appendix A, Publications on page 85.

Be sure to periodically check the HP Software support web site for new and updated publications.

2 Products and Benefits

At the end of this chapter, you will:

- Know the Client Automation family of products.
- Understand the benefits of each product.

Overview

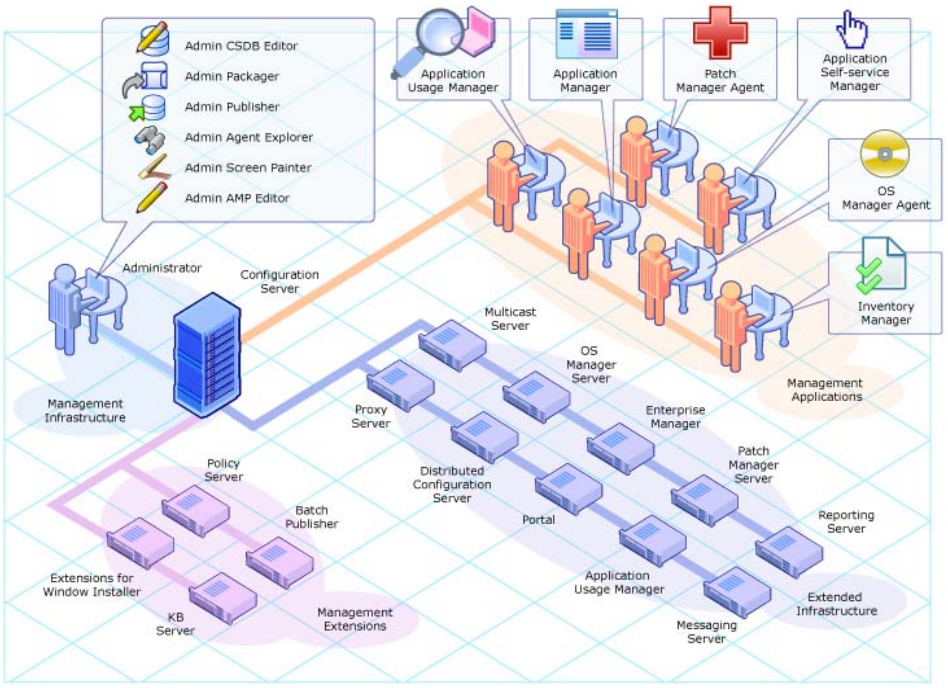
This section contains an overview of the HP Client Automation products and components. Client Automation products are divided into four categories.

- Management applications (agents)
- Management infrastructure
- Extended infrastructure
- Management extensions

Components are shared among multiple products.

Figure 2 below shows the four Client Automation product categories: and their products.

Figure 2 Infrastructure overview



Essential Functions

Table 2 below summarizes the essential functions for each product. For more information on each product, see the product's section in this chapter and the associated publications.

Table 2 Products and essential functions

Use	To
Management applications	
Application Management Profiles Agent	Deploy complex client applications using Application Management Profiles. Manage Windows Terminal Server and Citrix applications.
Application Manager Agent	Deploy mandatory packages to unattended devices. Supports Thin Clients to manage Windows Terminal Server and Citrix applications.
Application Self-service Manager Agent	Allows users to decide when to install the packages to which they are entitled.
Inventory Manager Agent	Track and report on hardware and software on agent devices. Used with the Inventory Manager Server. The Inventory Manager has agent and server components.
OS Manager Agent	Provision and manage operating systems on agent devices. The OS Manager has agent and server components.
Patch Manager Agent	Deploy and analyze vendor's security patches and bulletins. Used with the Patch Manager Server. The Patch Manager has agent and server components.
Management infrastructure	
Administrator	Configure and maintain your Client Automation environment.
Configuration Server	Configure and maintain the desired state for your enterprise devices and agent computers.
Extended infrastructure	
Application Usage Manager	Assess patterns of application usage in your environment.

Use	To
Distributed Configuration Server	Replicate part or all of your CSDB across a network of Configuration Servers.
Enterprise Manager	Uses Web services to set policy that interacts with your directory service and your Configuration Server.
Portal	Stores information about the target devices in your environment.
Multicast Server	Simultaneously sends one data stream to multiple agents.
OS Manager	Provision and manage operating systems on Client Automation agent devices. The OS Manager has agent and server components.
Patch Manager	Deploy and analyze vendor's security patches and bulletins. Used with the Patch Manager Server. The Patch Manager has agent and server components.
Proxy Server	Use cache management over HTTP and TCP/IP to store and transmit application data dynamically, freeing resources on the Configuration Server.
Management extensions	
Batch Publisher	Create fully automated, unattended updates to application packages.
Configuration analyzer	State files keep a detailed history of the resources needed by applications to run successfully.
Extensions for WI	Publish and manage Windows Installer applications.
Knowledge Base Server	Monitor and collect data from a user-defined directory and store them in the Knowledge Base.
Policy Server	Use directory services to implement policies.

Infrastructure Example

The following example shows how Client Automation products and components work together.

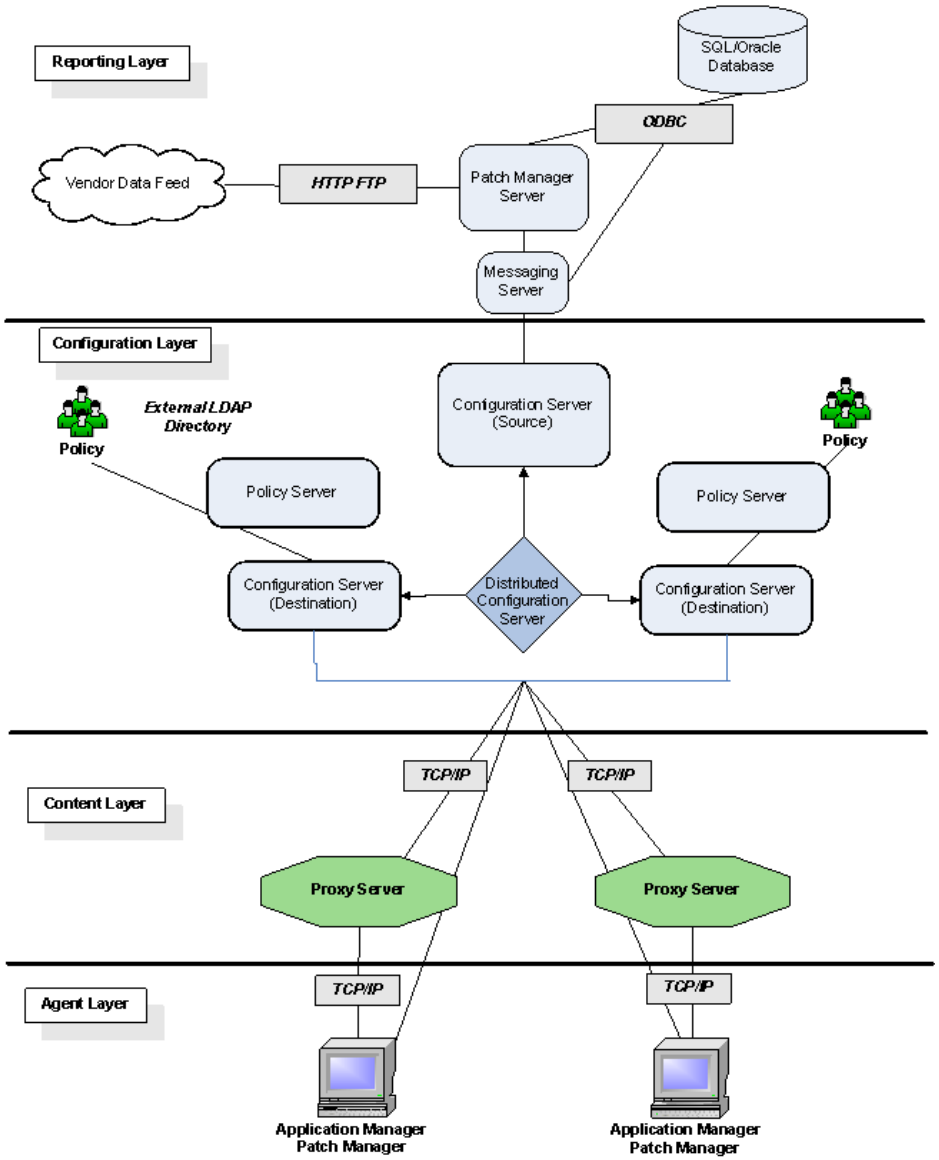
If you want to set up a Patch Manager environment to:

- Manage mandatory data (Application Manager).
- Analyze and manage security vulnerabilities (Patch Manager).

- Create policy using your existing external LDAP directory services (Policy Server).
- Place servers with your data in network locations that are strategic to your target devices (Proxy Server).
- Distribute the management of the devices across your enterprise (Configuration Server and Distributed Configuration Server).

You could combine the functions of the Client Automation products that are mentioned in Table 2 on page 23. A diagram of your network might be similar to Figure 3 on page 26.

Figure 3 A Patch Manager environment



Management Applications

Management applications are agent-based applications that can be installed on target devices in your enterprise. They communicate with the servers to enable you to manage the discovery, deployment, configuration, repair, update, and removal of data on devices such as servers, desktops, mobile devices such as laptops and handhelds, and specialty devices such as ATMs, point-of-sale, and Internet kiosks.

When a management application is installed on a target device, the device becomes a managed device.

There are several Client Automation agents available for communicating with the Configuration Server. The Configuration Server stores the configuration parameters and links policies to your managed devices. This server is discussed in detail in Configuration Server on page 34. You can install more than one agent on a device to combine features.

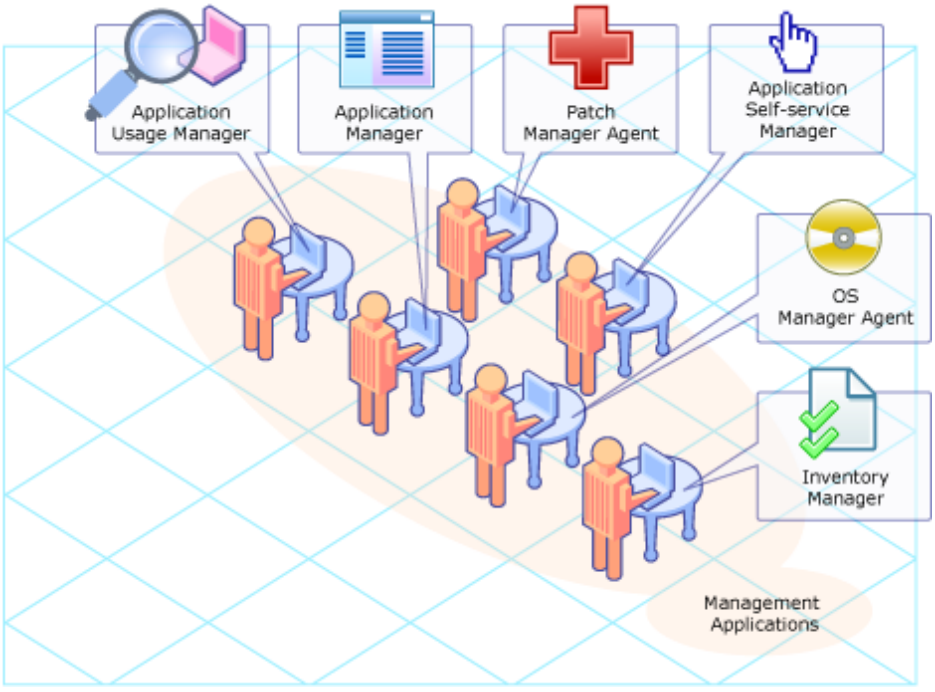
Table 3 below describes the essential functions of each of the Client Automation agents.

Table 3 Client Automation agent essential functions

Agent	Use
Application Manager	Deploys mandatory services to unattended devices. Supports deployment of complex applications using Application Management Profiles. Supports management of thin clients and Windows Terminal Server and Citrix applications.
Application Self-service Manager	Allows users to install the services to which they are entitled.
Inventory Manager	Tracks and reports hardware and software on managed devices.
OS Manager agent	Enables your managed devices to work properly with OS Manager for operating system deployment.
Patch Manager agent	Deploys and analyzes vendor's security patches and bulletins.

Not all management applications are available for all operating systems and architectures. Consult the HPCA 7.20 Release Notes on the HP support web site or your HP representative for the most current information on platform availability.

Figure 4 Management applications (agents)



Application Manager

The HP Client Automation Application Manager (Application Manager) allows you to deploy mandatory data to devices without user intervention. Administrators control deployments, updates, repairs, and removals through policy-based entitlements.

When the Application Manager is installed on the managed device, the administrator can:

- Deploy mandatory (required) data to unattended devices.

- Install, remove, verify, repair, and update data on a schedule or immediately.
- Control application versions.
- Control deployment of complex applications using Application Management Profiles.
- Manage thin clients and terminal server applications.

Refer to the *HP Client Automation Application Manager and Application Self-service Manager Installation and Configuration Guide* (*Application Manager and Application Self-service Manager Guid*e).

See the following topics for more information on Application Management Profiles and Terminal Server Applications.

Application Management Profiles

Application Management Profiles is a set of tools that enables the deployment and management of complex software products that are typically required in a Client Automation-managed environment, such as Microsoft Office 2007, Symantec Antivirus, Citrix Presentation Agent, among others. These products often employ their own instrumentation or repositories for deployment and management of their implementation.

Application Management Profiles provides tools to:

- Analyze and parameterize configuration control data for targeted products.
- Specify values in the form of a model to be used at deployment time.
- Articulate in the model the prerequisites and constraints that qualify the ability to deploy the products.
- Publish the control information and required ancillary tools such as utility programs to the Client Automation infrastructure for deployment.
- Deploy and configure the software to targeted devices.
- Interact with the target environment before and after installation to enhance management of the deployment.
- Report on the success of the deployment.
- Allow for load balancing for Apache Servers.

Refer to the *HP Client Automation Application Management Profiles User Guide (AMPs Guide).*

Terminal Server Support

Windows Terminal Services is a thin-client server. The processing of one or more applications is located on a centralized server rather than a user's desktop. Only screen, mouse, and keyboard information is passed between the agent and the server.

Terminal Server Support provides the ability to install and manage applications in a Windows Terminal Server environment, for applications to be run by Windows Terminal Server agents, and for applications to be run locally on the Windows Terminal server.

As part of the application deployment process, the Terminal Server support embedded in the Application Manager Agent automatically manages the install and execute modes in which a Windows Terminal Server can install and maintain applications. Before installing or updating an application, the Application Manager Agent queries the Windows Terminal Server for active sessions, prompts users to log off, and, if needed, disconnects user sessions. After completion of the installation, the Windows Terminal Server is brought back online to accept incoming user sessions.

Refer to the *HP Configuration Management Solutions for Servers Windows Terminal Server and Citrix Support Guide* (*Windows Terminal Server and Citrix Support Guide*).

Application Self-service Manager

Users can install, remove, and update optional applications that the administrator makes available to them. The HP Client Automation Application Self-service Manager (Application Self-service Manager):

- Enables self-service software and content management for users within the extended enterprise.
- Provides users with an interface to install, remove, verify, and update their *own* elective software and content. The administrator decides which software and content the user is entitled to.

Refer to the *Application Manager and Application Self-Service Manager Guide*.

Inventory Manager

This management application, used in conjunction with the HP Client Automation Messaging Server (Messaging Server) and HP Client Automation Reporting Server (Reporting Server), allows you to collect hardware and software information for reporting.

The HP Client Automation Inventory Manager (Inventory Manager):

- Automatically gathers information about software and hardware configurations and consolidates the results into web-based reports.
- Increases manageability of enterprise data by maintaining current inventory information collected across LAN, Internet, and dial-up links, and across a wide array of heterogeneous devices and operating systems.

Refer to the *HP Client Automation Inventory Manager Installation and Configuration Guide (Inventory Manager Guide)*.

Patch Manager

This management application is used with the HP Client Automation Patch Manager (Patch Manager) Server as described on page 43. Patch Manager automatically discovers, analyzes, and deploys software patches for Windows, Linux, and UNIX platforms. The IT administrator controls the patch lifecycle, which includes acquisition, testing, conflict analysis, vulnerability assessments, deployment, and ongoing management, through policy-based entitlements.

The Patch Manager agent:

- Gathers information about security patches installed on the managed device.
- Manages the deployment of patches.
- Monitors the continued security vulnerability compliance of managed devices.

Refer to the *HP Client Automation Patch Manager Installation and Configuration Guide (Patch Manager Guide)*.

OS Manager

The HP Client Automation OS Manager (OS Manager) is comprised of both an agent and a server. These two pieces allow you to provision operating systems on agent devices.

The OS Manager:

- Deploys operating systems based on policy assignments.
- Can prompt the user to choose an operating system based on a set of criteria.

Refer to the *HP Client Automation Enterprise OS Manager System Administrator Guide (OS Manager Guide)*.

Application Usage Manager

The HP Configuration Management Application Usage Manager (Application Usage Manager) uses parameters that you set in the CSDB to collect data on the patterns of application usage on your managed devices. These data are reported back to the Application Usage Manager Server for reporting and analysis.

For more information, refer to the *HP Client Automation Application Usage Manager User Guide (Application Usage Manager Guide)*.

Infrastructure

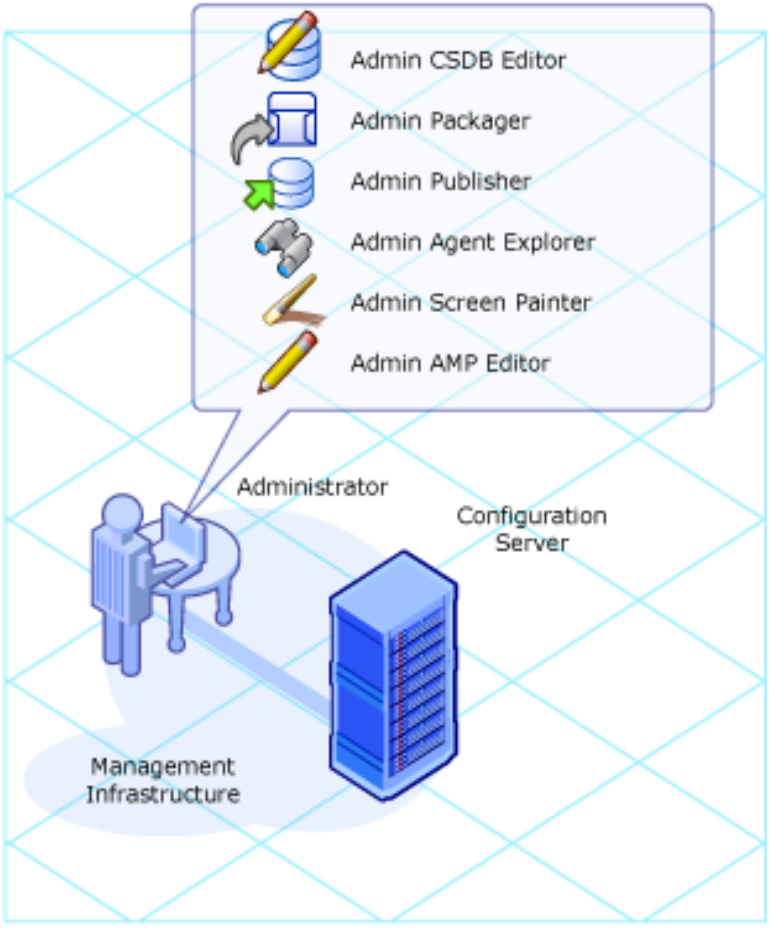
Use the Client Automation infrastructure to maintain desired state information, store data packages, automate software management activities, and administer your environment. The HP Client Automation Configuration Server (Configuration Server) and HP Client Automation Configuration Server Database (Configuration Server Database, CSDB) are the core of your management infrastructure. Consult the HP Software support web site or your HP representative for the most current information on platform availability.

Table 4 on page 33 describes the essential functions of the infrastructure components.

Table 4 Infrastructure essential functions

Infrastructure product	Use
Administrator	Contains tools that you use to configure and maintain your environment.
Configuration Server	Configures and maintains desired state information for your devices.
Configuration Server Database	Stores the desired state configuration in a hierarchical structure. The CSDB resides on the Configuration Server.

Figure 5 Client Automation infrastructure



Configuration Server

The Configuration Server resides on a single server or multiple servers. Applications and information about the users and managed devices are stored in the CSDB on the Configuration Server. The Configuration Server distributes application packages based on policies that are established by the administrator. Refer to the *HP Client Automation Configuration Server User Guide (Configuration Server Guide)*.

The Configuration Server:

- Dynamically generates the desired states based on situation-specific data that create a software environment that automatically adapts to changes.
- Synchronizes distributed objects, such as application components, packages, computer configurations, and policy relationships across the network.
- Maintains enterprise policies in the CSDB. When a managed device connects to the Configuration Server, current policy is automatically transmitted to and updated on the managed device.
- Contacts devices to have them initiate requests to the Configuration Server according to a schedule, upon notification from an administrator, or when invoked by the user. Managed devices do not poll over the network, conserving network bandwidth.

To synchronize multiple Configuration Servers, use the HP Client Automation Distributed Configuration Server (Distributed Configuration Server). See Distributed Configuration Server on page 40 and refer to the *HP Client Automation Distributed Configuration Server Installation and Configuration Guide (Distributed Configuration Server Guide)*.

Configuration Server Database (CSDB)

Administrators maintain an enterprise's policies in the CSDB. A policy defines the services to which users and managed devices are entitled. More experienced users may use the CSDB Editor of the HP Client Automation Administrator (Administrator) for advanced tasks.

The CSDB is stored on the Configuration Server. It includes the following information:

- The data that Client Automation distributes.
- The policies showing which managed devices and users are assigned to which packages.
- Security and access rules for Client Automation administrators.

For more information on the structure of the CSDB, see Configuration Server Database on page 60.

Administrator

The HP Client Automation Administrator (Administrator) tools provide centralized control of Client Automation objects and policy. Administrators use these tools to manage the CSDB, prepare applications for management,

view Client Automation agent objects, and customize their environments. Refer to the *HP Client Automation Administrator Guide (Administrator Guide)*. The Administrator includes the following tools and capabilities:

Agent Explorer

The HP Client Automation Administrator Agent Explorer (Agent Explorer) provides a user interface to view and edit agent objects on managed devices, as well as diagnose issues by viewing error objects. Agent objects represent the current state of the managed device.

AMP Editor

The HP Client Automation Application Management Profiles Editor (AMP Editor) is a tool you use to create and modify application profiles. These profiles allow you to deploy and manage complex software products (such as Microsoft Office 2007, Symantec Antivirus, and Citrix Presentation Agent) that are typically required on devices in a Client Automation environment. Refer to the *HP Client Automation Application Management Profiles Guide (AMP Guide).*

CSDB Editor

The HP Client Automation Administrator CSDB Editor (CSDB Editor) gives an experienced administrator a user interface with which to configure policy and application services stored in the CSDB, although most users will use the Enterprise Manager to administer their environments. In addition, the CSDB Editor allows administrators to perform the following tasks:

- Modify application packages after the initial publishing process.
- Establish reuse of application components between application services.
- Define application service prerequisites.
- Define policy for application entitlements.
- Control deployment of application versions.
- Centralize control for unattended application service updates, install, and repair.

Packager

Before data can be deployed across an enterprise, it must be organized into a unit that can be distributed. This unit is called a **package**. The Packager provides a user interface for packaging components for distribution. After a package is created, it is published to the Configuration Server Database (CSDB).

The Packager has one publishing mode, Installation Monitor Mode. In Installation Monitor Mode, the Packager determines what to package by scanning the computer before and after the software has been installed. It *differences* the before and after scans to determine what changes were made to the computer. These differences make up the package that you publish to the CSDB.

Publisher

The HP Client Automation Administrator Publisher (Publisher) is an administrative tool that you use to package and publish Windows Installer applications, hardware configurations, and operating system images to your CSDB. It also allows you to publish files in batch mode. As of the CM 5.10 release, it also includes a packaging process called Component Select.

Install the Publisher to the computer where your administrator will have access to the necessary files for publishing.

In contrast to the Packager, where before and after installation scans are used to identify the contents of a package, when you use the Publisher, you select the individual components (such as files, directories, registry entries, and links) that make up the package.

Screen Painter

Use the HP Client Automation Administrator Screen Painter (Screen Painter) to create and design custom dialog boxes.

Extended Infrastructure

Use the Client Automation extended infrastructure to scale software management services across your enterprise. This gives you the ability to manage devices across multiple network segments. The extended infrastructure can be divided into two categories: products and components.

Components are shared among two or more products. The software for components is provided as needed with the products.

Products

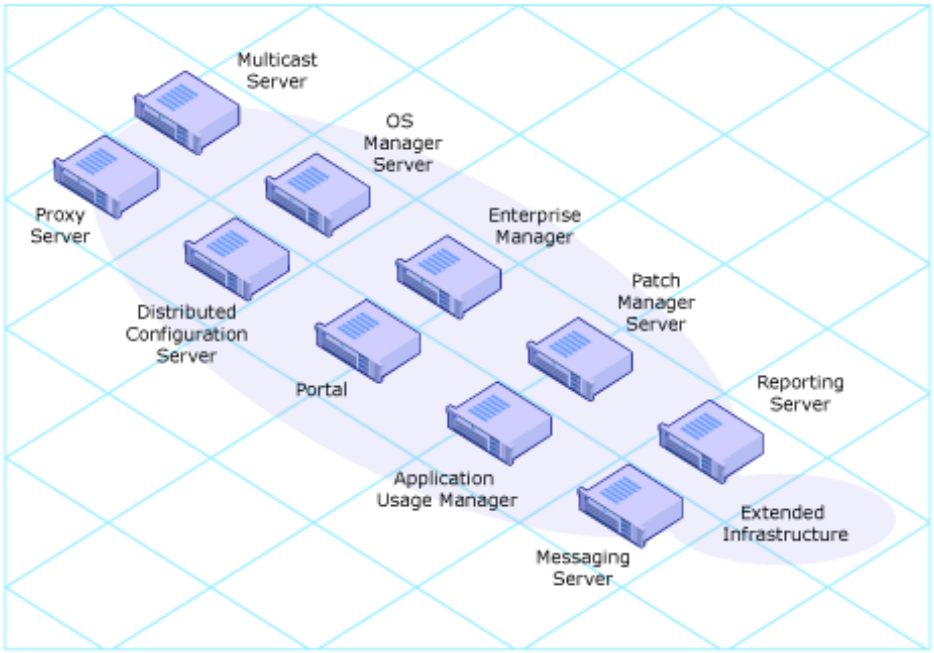
Client Automation extended infrastructure provides a complete analysis of your management solution through distributed administrative capabilities, vulnerability assessments, and monitoring of application usage patterns.

Table 5 below describes the essential functions of the extended infrastructure components.

Table 5 Extended infrastructure essential functions

Extended infrastructure product	Use
Application Usage Manager	Assesses patterns of application usage in your environment.
Distributed Configuration Server	Replicates part or all of your databases across a network of Configuration Servers.
Enterprise Manager	Web-based, agent management tool that allows you to quickly and easily manage software, patches, and inventory for devices in your environment.
Multicast Server	Simultaneously sends the same resources in one data stream to multiple devices.
OS Manager	Provisions and manages operating systems on target devices.
Patch Manager	Deploys and analyzes vendors' security patches and bulletins. Used with the Patch Manager Server. The Patch Manager has agent and server components.
Portal	Stores information about the target devices in your environment.
Proxy Server	Uses cache management over HTTP or TCP/IP to store and transmit application data dynamically, freeing resources on the Configuration Server.

Figure 6 Extended infrastructure



Application Usage Manager

You can use the HP Client Automation Application Usage Manager (Application Usage Manager) to assess patterns of application usage in your environment. This allows you to facilitate adherence to license agreements, re-provision licenses if needed, and monitor user productivity.

The Application Usage Manager monitors the use of every application on all of your devices. This enables you to:

- Enforce corporate standards by identifying non-standard software and software versions in use.
- Implement license tracking, which gives you the ability to purchase and maintain only those licenses that are needed.
- Enable OS migration support by prioritizing software distribution based on actual usage.
- Use reporting to view the actual use of application resources.

Refer to the *HP Client Automation Application Usage Manager User Guide (Application Usage Manager Guide)*.

Distributed Configuration Server

The size of your enterprise may require more than one Configuration Server. This could mean that different data are stored in each CSDB and must be shared with other, remote CSDBs. The Distributed Configuration Server allows multiple Configuration Servers to share information about policies and managed data.

The Distributed Configuration Server facilitates the information sharing by allowing an administrator to configure and run database-to-database **synchronizations**. A synchronization must be configured for two Configuration Servers: a source and a destination. The destination domain is always a replica of the source domain. See Configuration Server Database on page 60.

The benefits of the Distributed Configuration Server are:

- Administrators can automatically synchronize distributed CSDBs. This allows managed applications and policy information to be shared across the enterprise.
- Individual Configuration Servers do not need to share a common network protocol or operating system.

Refer to the *HP Client Automation Distributed Configuration Server Installation and Configuration Guide (Distributed Configuration Server Guide)*.

Enterprise Manager

The HP Client Automation Enterprise Manager (Enterprise Manager) is a web-based, agent management tool that allows you to manage software, patches, and inventory for devices in your environment.

To use the Enterprise Manager, you need the following Client Automation products installed in your environment, and you need to determine which will be used by Enterprise Manager.

- Configuration Server
- Reporting Server
- Portal

You do not need the Reporting Server to use the Enterprise Manager. However, if one is not identified or installed, you will not be able to access any of the Reporting Server features in your Enterprise Manager.

Beginning with version 7.20, HP Client Automation offers a vulnerability management solution that enables you to detect security vulnerabilities on managed client devices in your enterprise and quickly assess the severity and scope of the related risk. You can then take steps to remediate these vulnerabilities.

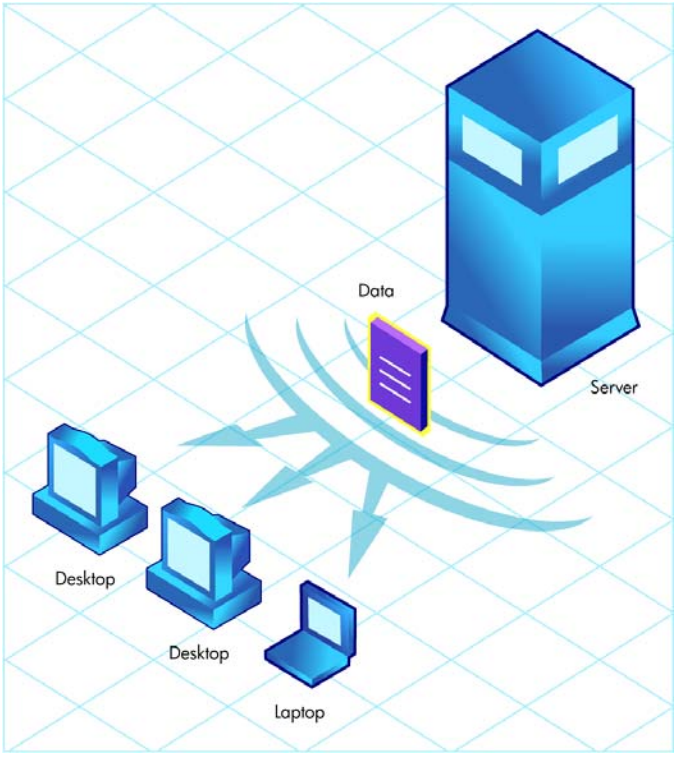
The Enterprise Manager provides a Vulnerability Management dashboard that shows the security vulnerability status of your enterprise at a glance. It also provides a Patch Management dashboard to help you quickly assess patch policy compliance across the enterprise and an HPCA Operations dashboard to show you the number and type of operations HPCA has performed over time.

Multicast Server

The HP Client Automation Multicast Server (Multicast Server) reduces the number of transmissions necessary and maximizes the use of network bandwidth. This enhances and simplifies data-transmission technology. Multicasting allows the transmission of the same data stream to many targets, simultaneously. This method of information transmission differs from the usual method of transmission, where a server has to transmit the same information to each of its targets individually.

Most multicast utilities are designed to provide the simplest delivery of a payload that has been statically composed of all files and components for all possible recipients. In this model, every receiver is forced to take all resources bundled into that payload. The Multicast Server allows the collection of the sets of resources that are needed by only those receivers that are eligible to participate in a specific multicast transmission. Only data required by managed devices are sent, and these devices retrieve only data that they have requested.

Figure 7 Multicast transmissions



The Multicast Server provides the following benefits:

- Sends only data that are required by the devices.
- Maximizes the utilization of network bandwidth by transmitting a single data-stream to multiple target devices.
- Saves resources of the server by not having to set up separate agent connect sessions and then repeatedly transmit data to each device individually.

The Configuration Server takes part in the multicast process during the agent connect, and as the repository for the database files. The Multicast Server accesses the resources needed for transmission during the multicast from the CSDB. The CSDB also contains the information required to:

- Determine whether a managed device is eligible.
- Assign the managed device to a multicast group.

- Inform the managed device of the files it needs in order to match the desired state.

Refer to the *HP Client Automation Multicast Server Installation and Configuration Guide (Multicast Guide)*.

OS Manager

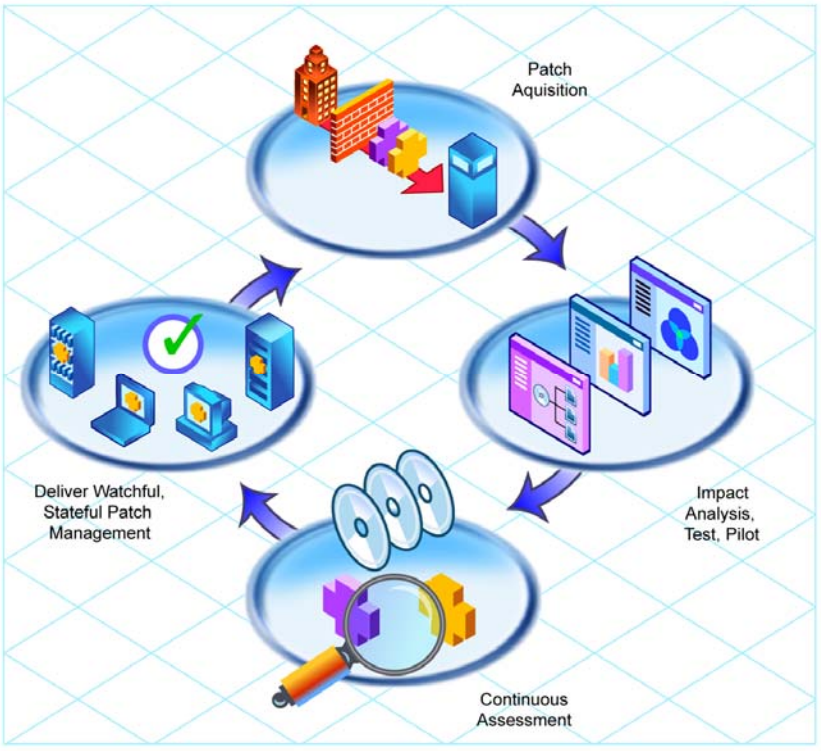
The HP Client Automation OS Manager uses policy-driven, real-time, state-based management so you can configure and deploy operating systems (OSs). Use the OS Manager to install or replace operating systems on a device and maintain the device according to policy. Benefits of the OS Manager include:

- Reduces the learning curve for administrators because it is a fully integrated component of Client Automation.
- Automated policy-based management that improves speed and reliability of operating system deployment.
- Desired state automation that maintains operating systems in the right configuration and increases service levels.
- Reduced IT costs by simplifying and streamlining the OS management process across multiple platforms.

Patch Manager

HP Client Automation Patch Manager (Patch Manager) eliminates known software vulnerabilities by automating the patch management process including acquisition, impact analysis, pilot testing, discovery, assessment, deployment, maintenance, and compliance assurance. This ensures that managed devices are always correctly configured. Use the Patch Manager to configure acquisition tools to collect security patches from a vendor's web-based security patch repository, as well as perform impact analysis and pilot testing to identify affected applications and devices.

Figure 8 Patch Management life cycle



Features of the Patch Manager include:

- An acquisition tool that can be configured to enable programmatic collection of new security patches directly from a vendor's web-based depository for security patches.
- Ability to perform impact analysis to identify affected applications and devices.
- Automatic and continuous discovery of devices on the network, software products that are installed on each device, the collected security patches that are already applied to each software product, and identification of software products that the device executes.
- Policy-based management capabilities that directly interface with a variety of existing policy sources such as Active Directory, LDAP, and SQL databases.

- Monitoring devices and checking policy to see if they are in compliance. If they are not in compliance, devices are automatically updated with the appropriate patches.

Portal

The HP Client Automation Portal (Portal) is web-based and gives you the ability to manage your environment regardless of location or computing platform. Administrative tasks can be distributed to administrators in remote locations based on roles and policies. Some of the tasks allow you to deploy agents, detect the status of installed services, manage the CSDB, and track the completion status of all Portal tasks.

The Portal cannot always perform tasks remotely; therefore, the Portal agent, which must be installed on a remote managed device, performs these tasks on behalf of the Portal.

Some of the functions the Portal can perform are:

- Discover and view devices on your network.
- Set policy and configure the desired state.
- Remotely start and stop services.
- Remotely install some of the products, such as the Client Automation agents, the Portal agent, and the Proxy Server.
- Notify a group of devices to perform an action such as install software or audit services.

Refer to the *HP Client Automation Portal Installation and Configuration Guide (Portal Guide)*.

Proxy Server

The HP Client Automation Proxy Server (Proxy Server) allows data to be locally available to managed devices. Managed devices can receive data over the Local Area Network (LAN) instead of across a Wide Area Network (WAN). Proxy Servers increase scalability while dramatically reducing traffic over the network. When data are cached on the Proxy Server, the demand placed on the Configuration Server is decreased, allowing it to allocate more resources to other tasks.

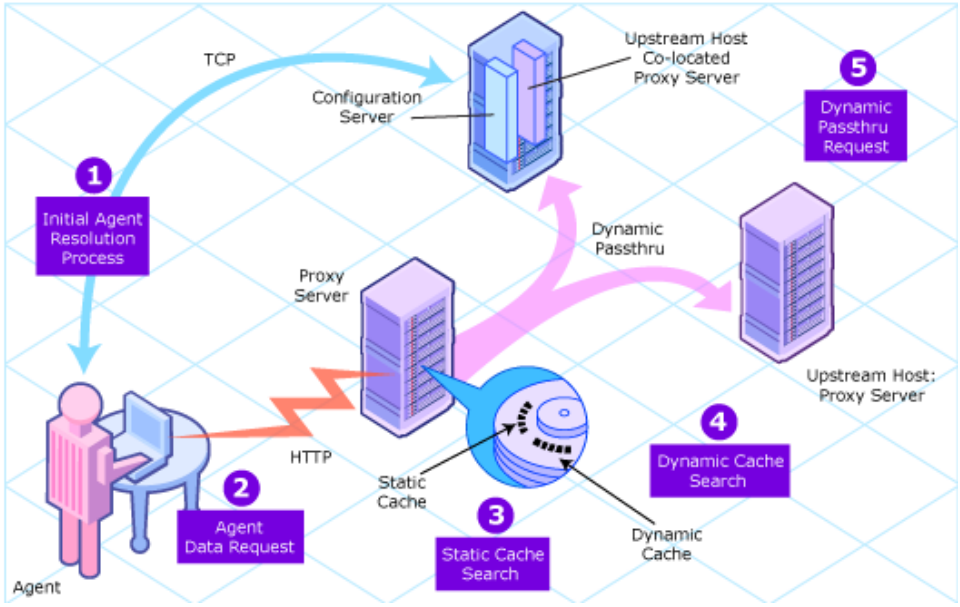
Place Proxy Servers at strategic points in your network to increase the efficiency at which data are transferred. The connection between users and the Proxy Server may be more efficient than the users' connections to the

Configuration Server. The factors that determine the efficiency of a connection between a server and a device include hardware capability, network bandwidth, workload on the servers, network traffic patterns, and the volume of software to be distributed.

The Proxy Server is not a generic proxy, but rather a proxy specifically designed to manage and distribute Client Automation data.

The Proxy Server can be installed and software can be preloaded using the Portal. If the Integration Server is not already installed, the Proxy Server installation will install it.

Figure 9 Proxy Server caching



The Proxy Server is the primary repository for data that is to be deployed. After the managed device determines the resources needed for its desired state, it can request the resources from the Proxy Server. The Proxy Server provides the following benefits:

- The ability to choose between having requests made using either HTTP (recommended) or TCP/IP.
- The ability to service multiple, concurrent requests from either protocol source.

- The ability to have data automatically loaded onto the Proxy Server for distribution when the first request comes in from a managed device. This occurs if the data do not already exist on the Proxy Server.
- The ability to automatically send requests to the Configuration Server for processing if the Proxy Server cannot handle the request.

Refer to the *HP Client Automation Proxy Server Installation and Configuration Guide (Proxy Server Guide)*.

Shared Components

Some products share components to consolidate communications and facilitate data flow. Shared components include the Integration Server, Messaging Server, and the Reporting Server.

Table 6 Shared components essential functions

Component	Use
Integration Server	Some products that use the Integration Server are loaded from a single Windows service called HPCA Integration Server; other products require a dedicated Windows service that is not shared by another Client Automation component. Dedicated service name examples include the HPCA Portal and HPCA Patch Manager Server. The Integration Server is used by the Portal, Proxy Server, Policy Server, and Patch Manager.
Messaging Server	The Messaging Server routes data from agent objects to the appropriate infrastructure server or reporting database. The Messaging Server is used by all Client Automation agents that report objects, as well as the Portal, Reporting Server, and Vulnerability Management.
Reporting Server	The web-based Reporting Server allows you to use data from SQL and Oracle databases for reporting. The Reporting Server is used by Patch Manager, Application Usage Manager, Inventory Manager, Vulnerability Management, and Application Management Profiles.

Integration Server

Many of the extended infrastructure products (namely, the Portal, Proxy Server, Policy Server, and Patch Manager Server) use the Integration Server. The Integration Server does not have its own installation. It is loaded if the product needs it and if it is not already installed in the installation directory. Each product is composed of modules that reside in the Integration Server `modules` directory.

For some products, the components use the same core Integration Server files and run under the same process. For other products, such as the Portal and Patch Manager, the components require a dedicated instance of the Integration Server and run under independent processes.

Benefits of the Integration Server are:

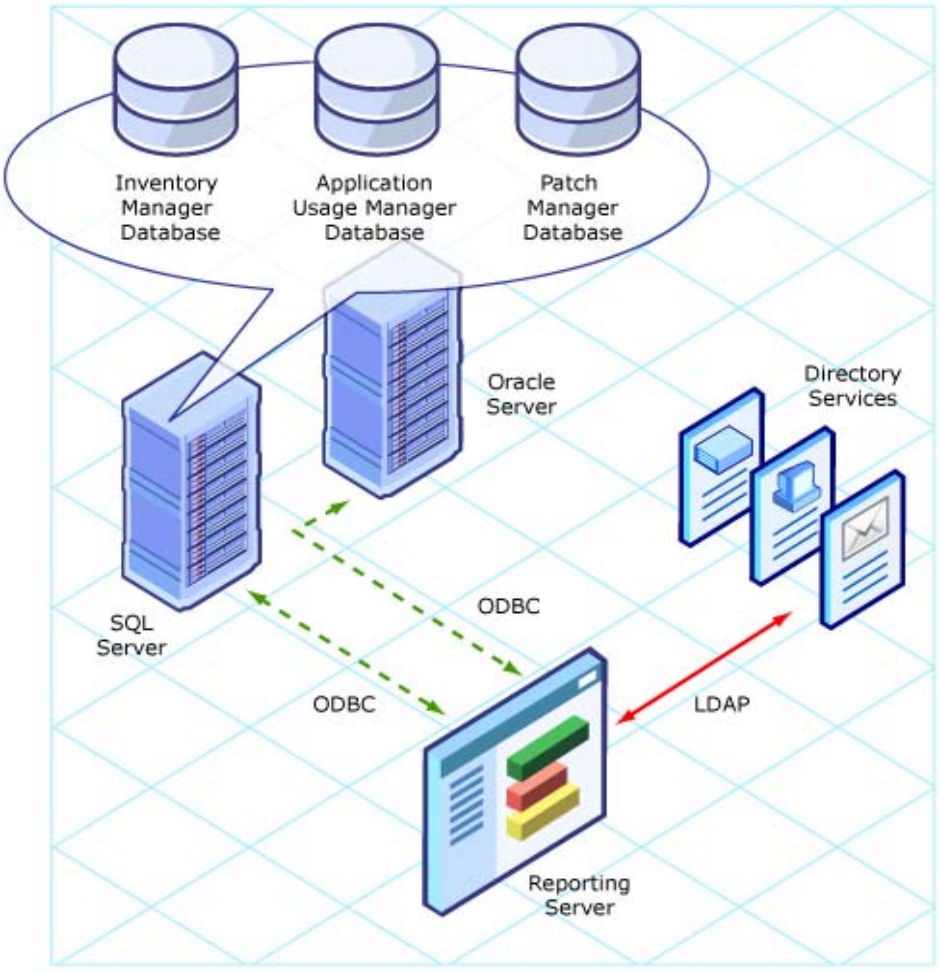
- The products that can use a common Integration Server for Windows may be loaded from a single Windows service.
- When the Integration Server starts, it scans its configuration file and attempts to load all the products marked for loading.
- Each product loaded from the Integration Server is separately licensed.
- The Integration Server provides Web services that may be shared by all loaded modules. This integration provides increased performance, efficiency, and maintenance ease.

Reporting Server

The HP Client Automation Reporting Server (Reporting Server) allows you to use SQL data for reporting. As part of the extended infrastructure, the web-based Reporting Server allows you to query the data in the combined Patch Manager, Inventory Manager, Application Usage Manager, and Application Management Profiles databases, and create detailed reports. You may also mount an existing LDAP directory, which allows you to filter your data using your LDAP directory levels.

Figure 10 on page 49 illustrates a sample Reporting Server environment.

Figure 10 Sample Reporting Environment



The Reporting Server provides the following additional value to the infrastructure:

- Connections to SQL databases: The Reporting Server can access any SQL database, such as those for Inventory Manager, Patch Manager, Application Management Profiles, and Application Usage Manager. However, all SQL databases accessed by the Reporting Server must exist on a single SQL or Oracle Server.

- Connections to LDAP directory (optional): The Reporting Server supports optional access to an LDAP directory in your enterprise. Access to an LDAP directory allows you to filter report data according to the directory entries.

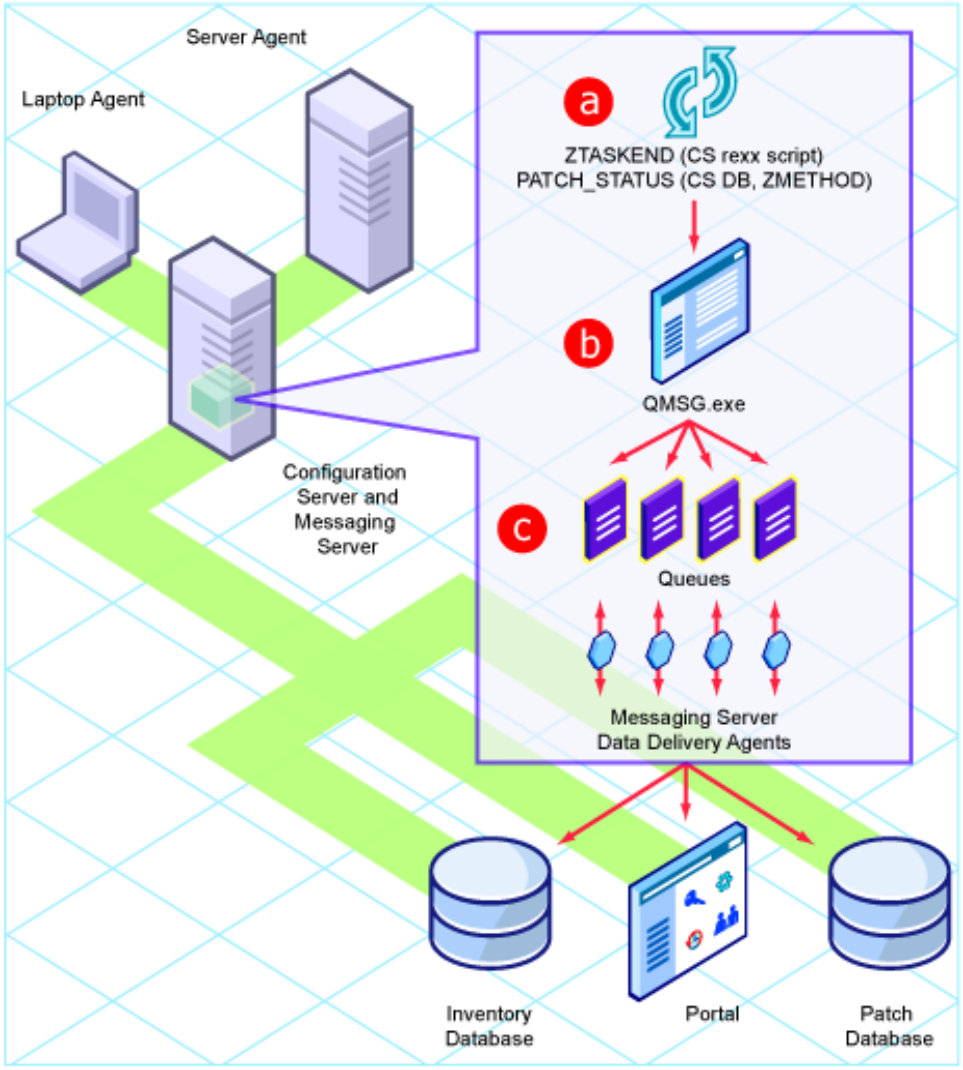
Refer to the *HP Client Automation Reporting Server Installation and Configuration Guide (Reporting Server Guide)*.

Messaging Server

The HP Client Automation Messaging Server (Messaging Server) is a service that continually monitors pre-defined locations on a server and routes data to external destinations. The Messaging Server provides retry, rerouting, and failover capabilities to ensure that all data are transferred efficiently and reliably.

The Messaging Server receives data from managed devices and delivers it to the appropriate ODBC database, the Portal directory, or another server. For example, if Patch Manager data are being transferred, the Messaging Server will post the patch data to the appropriate SQL database.

Figure 11 Messaging Server routes data



The Messaging Server uses **Data Delivery Agents** (DDAs) to handle the routing of agent objects for patch management, inventory management, application management profiles, and application usage management to the appropriate reporting database. Data Delivery Agents use HTTPS or HTTP to post data directly to an ODBC database, or to forward data securely to another directory or server (such as another Messaging Server that is used to post the data to a local database).

The Messaging Server runs on all Windows and UNIX platforms that are supported by the Configuration Server. The Messaging Server can:

- Route a single message to multiple destinations.
- Automatically retry a delivery.
- Re-route messages to a new host after several unsuccessful delivery attempts.

Refer to the *HP Client Automation Messaging Server Installation and Configuration Guide (Messaging Server Guide)*.

Management Extensions

Management extensions provide integration and extended enterprise functionality. They allow Client Automation to interface with other technologies such as **Lightweight Directory Access Protocol** (LDAP) and **Secure Sockets Layer** (SSL). Similar to the extended infrastructure, management extensions can be divided into common components, such as the Knowledge Base Server, and products, including Extensions for Windows Installer, Policy Server, and the Batch Publisher.

Figure 12 Management extensions

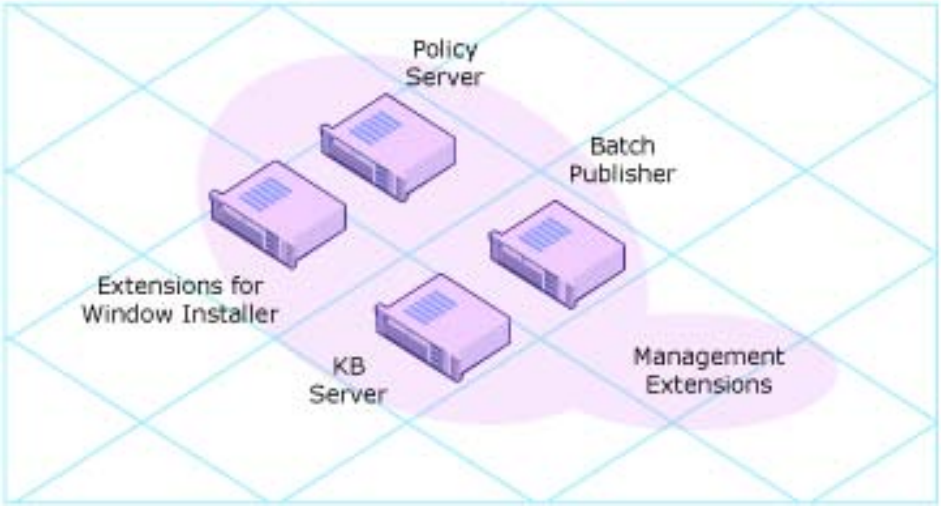


Table 7 Management extensions essential functions

Management extensions	Use
Batch Publisher	Creates fully automated, unattended updates to application packages.
Extensions for Windows Installer	Publishes and manages Windows Installer applications.
Knowledge Base Server	Populated with data in the form of state files. State files consist of data that represent the current state of an application.
Policy Server	Uses external directory services to implement policy.

Adapter for SSL

This stand-alone product has been retired from the Client Automation suite of products. SSL functionality is now built into each of the Client Automation products.

The *HP Client Automation Adapter for SSL Installation and Configuration Guide* has been replaced by the *HP Client Automation SSL Implementation Guide (SSL Guide)*, which documents the optional **Certificate Generation Utility**. The Certificate Generation Utility is:

- Provided *as-is* and *free of charge*.
- *Not* a supported Client Automation product.
- Used at *your own discretion*; HP Technical Support will *not* address any issues regarding its use or functionality.

Batch Publisher

The HP Client Automation Batch Publisher (Batch Publisher) is a command line driven publishing tool that can easily be integrated with third-party Client Automation and packaging products for fully automated, unattended updates to application packages. The Batch Publisher provides an alternative to the Component Selection Mode of the Administrator Publisher. It offers an automated, repeatable command-line process; while the Administrator Publisher must be monitored from start to finish.

The Batch Publisher identifies a set of files and components, and publishes them in a controlled, automated, repeatable manner, to the CSDB where they are stored as objects. Its focus is distributing updates to content, data, and applications rather than initial application packaging. Digital content, such as file sets, graphics, price lists, and interest rates, are types of managed lists that might require an automated update process that Batch Publisher can provide.

The Batch Publisher can:

- Scan for files on multiple drives or file systems.
- Scan and publish files from any mapped drive or file system.
- Be configured to limit the subdirectories that are scanned.
- Include or exclude files at the file level.
- Select files by type.

The Batch Publisher can also accommodate frequent patching of internal applications, as well as publish build versions, and output from HP legacy (PVCS or ClearCase) adapters. Its capacity to revise content material is reliable, and can execute continuously, at designated times, and in pre-determined intervals. It can be easily run from within any script or code capable of calling a command prompt. Refer to the *HP Client Automation Batch Publisher Installation and Configuration Guide (Batch Publisher Guide)*.

Configuration Analyzer

The HP Client Automation Configuration Analyzer (Configuration Analyzer) administrator console simplifies your view of application management. Backed by a database, imported state files keep a detailed history of the resources that are needed by an application to successfully run. The console can identify conflicts between two or more applications. With this historical and complete set of information, you can determine the impact on your environment of:

- Deploying a new application.
- Upgrading an existing application.
- Adding or modifying modules, registry keys, and data files.

The Configuration Analyzer performs integration and management analysis functions. Administrators can profile applications, initiate application comparisons or views, analyze applications, populate the Knowledge Base Server (KB Server), and establish and manage KB permissions.

To analyze data with the Configuration Analyzer, data must be in the form of state files. State files can be generated by different Client Automation products, including the Packager for WI, Application Usage Manager, and Patch Manager. Refer to the product guides for more information about how to create state files with each product. For more information on the Configuration Analyzer refer to the *Configuration Analyzer Guide*.

Extensions for WI

The HP Client Automation Extensions for Windows Installer (Extensions for WI) is a management system that automates and simplifies the enterprise application integration process. Use this product to build, test, maintain, deploy, and troubleshoot Windows Installer applications and installation packages.

The **Packager for WI** gives you complete control over the resource gathering, analysis, and creation of a Windows Installer package. Typically you can package an application in less than an hour. These interfaces allow you to choose the session granularity that you need:

- Use the Packager Menu to run a typical, unified packaging session or to use one of the custom options for creating a modified package.
- Use the Packager Process Menu to access a comprehensive flowchart of all package creation components that are part of the Extensions for WI suite.

The underlying components of the Packager for WI include a set of wizards and an editor. These can be launched individually or can be automatically launched from one of the menu options.

- **Configuration Analyzer**
  View, store, and compare application data. See Configuration Analyzer on page 54 and refer to the *Configuration Analyzer Guide*.
- **Install Wizard**
  Publish packages to a non- Client Automation distribution point, or modify packages using transforms. For publishing Windows Installer packages to the CSDB, refer to the *Administrator Guide*.
- **Knowledge Base Server**
  Populates the Knowledge Base with data in the form of state files.
- **Library Wizard**
  Create and modify library files.

- **MSI Editor - MSIEdit**
  Allows you direct access to the Windows Installer database tables through an easy to use interface.
- **Package Wizard**
  Build and modify Windows Installer packages.
- **State Wizard**
  Create and modify state files.

Refer to the *HP Configuration Management Extensions for Windows Installer Getting Started Guide* (*Extensions for WI GSG*) and the *HP Configuration Management Extensions for Windows Installer User Guide (Extensions for WI Guide).*

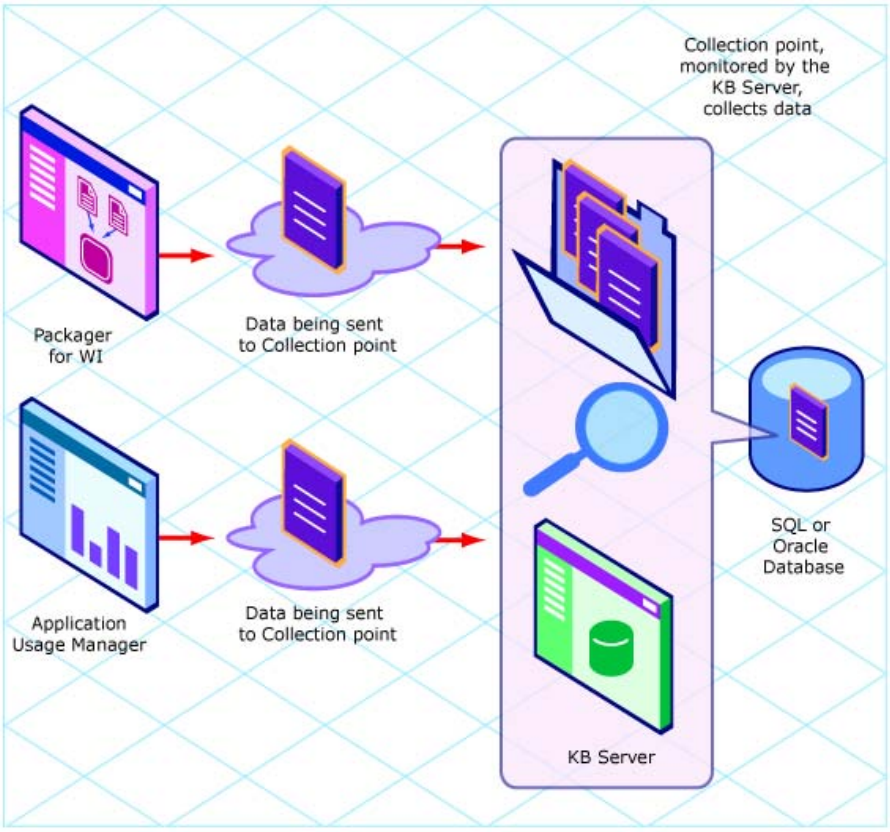
Knowledge Base Server

The HP Client Automation Knowledge Base (KB) database may be either a SQL Server or an Oracle database configured in your environment. The KB is populated with data in the form of state files. State files consist of data that represent the current state of an application. These data are acquired by the Knowledge Base Server from a continuously monitored user-specified directory referred to as a collection point. When data are detected in this collection point, they are automatically transferred to the KB. The collection point is populated by one or more products including the Application Usage Manager and Packager for WI. From here, the Configuration Analyzer can analyze the data. Application usage can be viewed with the Reporting Server.

The KB Server is capable of importing several types of state files including:

- Configuration Server service/package component extracts.
- State files built by the Extensions for WI components.
- Application Usage Manager collection files.
- State files built by the Patch Manager.

Figure 13 KB Server process



Refer to the *HP Client Automation Knowledge Base Server Installation and Configuration Guide (KB Server Guide)*.

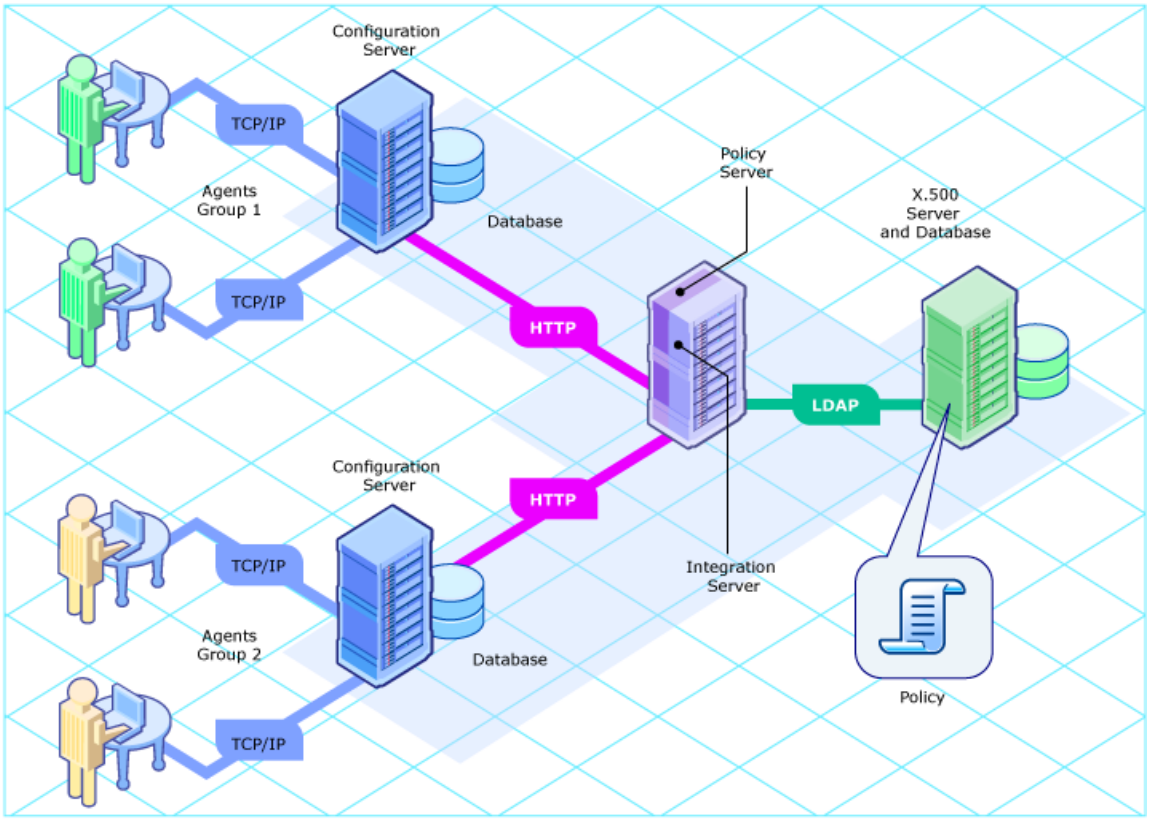
Policy Server

The HP Client Automation Policy Server (Policy Server) is a web server used to integrate the CSDB with existing, external **Lightweight Directory Access Protocol** (LDAP) directory servers and SQL databases in your environment. This integration enables single source points of control for user authentication, access policies, and user entitlement that already exist in your LDAP directory servers. Supported LDAP directory servers include

Microsoft Active Directory, Novell NDS, and other vendors' LDAP servers, as well as Oracle, Sybase, and Microsoft SQL-based databases.

Once the Policy Server is configured to provide access to your external LDAP directory servers, use the Enterprise Manager or the Portal to set policy by mapping services to the users in the external directory tree. The underlying connections made in the CSDB to an LDAP directory are used to determine which services should be distributed and managed for users.

Figure 14 Policy Server integrates with LDAP and SQL



For more information, refer to the HP Software support web site and the *HP Client Automation Policy Server Installation and Configuration Guide (Policy Server Guide).*

3 Essential Processes

At the end of this chapter, you will:

- Be familiar with the structure of the Configuration Server Database (CSDB) and Client Automation agent objects.
- Know the dynamics of a service.
- Be familiar with the packaging process.
- Understand the Client Automation agent connect process.
- Understand the resolution process.
- Understand the inventory collection process.
- Know the basics of Proxy Server processing.
- Know how the Patch Manager acquires security patches.

In order to understand the discussions of essential Client Automation processes in this chapter, you must have an understanding of the CSDB and Client Automation agent objects. You should also be familiar with the terms described in the section Terminology, starting on page 14.

Configuration Server Database

The HP Client Automation Configuration Server Database (CSDB), which is stored on the Configuration Server, records your enterprise's desired state model. This model contains the data that are distributed, policies that define the services to which users and devices are entitled, and security and access rules for administrators. Refer to the *HP Client Automation Configuration Server Database Reference Guide (CSDB Reference Guide)*.

The database is hierarchically structured as follows:

- **Files** are used to group similar domains. The PRIMARY File is used to define and maintain the desired state.
- **Domains** are logical file partitions that are used to group similar classes. The POLICY Domain contains the classes that are used to create users and groups.
- **Classes** are templates that contain the attributes that are used to create an instance. A class represents a category of the desired state. The USER Class of the POLICY Domain defines users of managed applications. It defines all of the attributes that are needed to identify the managed device.
- **Instances** are actual occurrences of classes. The attributes of a class instance contain data that describe a specific entity of that class. For example, a USER instance contains the information that is needed to identify target devices or users.
- **Attributes** are data elements of a class. The class contains the definition (that is, the name, data type, description, and length) for each attribute that belongs to the class. Each class instance that is created from the class contains a value for each of the attributes that are defined in the class. For example, the NAME attribute of a USER class contains the name of the user.

Default Domains

When you install the Configuration Server, LICENSE and PRIMARY are the only two files available. As you use Client Automation, your CSDB will change. Some of the management infrastructure products add more domains. For example, Patch Manager adds the PATCHMGR Domain, and Application Usage Manager adds the USAGE Domain. OS Manager uses the OS and MACHINE Domains.

- The LICENSE File is used for Configuration Server processing. This file is only for HP use.
- The PRIMARY File is where you will find most information regarding software management. Within the PRIMARY File, there are eight default domains.
  - Use the ADMIN Domain to define administrative rights and rules for connecting classes.
  - Use the APPMGMT Domain to manage Application Management Profiles.
  - Use the AUDIT Domain to configure tasks that will inventory managed devices.
  - Use the CLIENT Domain to configure Client Operations Profiles. This includes defining which Configuration Servers and Proxy Servers the managed devices can use.
  - Use the POLICY Domain to create users and groups, and to assign users to groups.
  - Use the PRDMAINT Domain to store packages for self-maintenance.
  - The SOFTWARE Domain contains information about the software that is managed and the methods used to deploy the software.
  - The SYSTEM Domain contains administrative and process control definitions.

The PROFILE File will appear after the first Client Automation agent has registered with the Configuration Server. This file contains information that is collected from managed devices. This information is used to connect to target devices to deploy software managed by Client Automation, and to see the configuration of the managed device.

The NOTIFY File appears after the first attempted Notify and contains information about attempts by the Notify function to update, remove, or email subscribers.

In this chapter, you will find the different parts of the CSDB defined as described in Table 8 below.

Table 8 Configuration Server Database usage

Element	Style	Example
Files	All uppercase	PRIMARY
Domains	All uppercase	PRIMARY.SOFTWARE May also be referred to as the SOFTWARE Domain in the PRIMARY File.
Classes	All uppercase	PRIMARY.SOFTWARE.ZSERVICE May also be referred to as the ZSERVICE Class in the SOFTWARE Domain in the PRIMARY File.

Refer to the *CSDB Reference Guide*.

Agent Objects

When a device connects to the Configuration Server, information is exchanged between the Client Automation agent and the Configuration Server. This exchange is called **resolution**. During resolution, the Client Automation agent checks the status of services and updates the Configuration Server with information from objects stored on the device. The resolution process will be described in Resolution Process on page 72.

Client Automation agent objects are stored in a directory called `IDMLIB` on the managed device. After installing the Client Automation agent and connecting to the Configuration Server, you can use Client Automation agent objects to determine:

- What is the hardware configuration of the managed device?
- Was the service successfully installed?
- When was the service installed?
- What is the managed device's name, and who was the last user to log on?
- What are the possible data sources for this managed device?

While there are multiple Client Automation agent objects on a managed device, there is a core group of objects that supply information and the status of the current agent connect. Table 9 on page 63 includes information about

when these objects are created and updated, as well as a brief summary of what the object includes. There are other objects created during the agent connect, only some are noted here. Refer to the *HP Client Automation Application Manager and Application Self-service Manager Installation and Configuration Guide* (*Application Manager and Application Self-service Manager Guide*).

Table 9 Agent objects

Object Name	Description
PREFACE	PREFACE is sent to the Configuration Server at every phase of an agent connect. It contains parameters used for the current connect such as the type of connect, the user name, and if the list of applications is being updated.
ZCONFIG	ZCONFIG is created at the start of the agent connect process. It contains basic hardware information for the managed device such as processor, operating system, and drives.
APPEVENT	APPEVENT reports on the status of application events, such as installation, verify, repair, and removal.
ZMASTER	ZMASTER is sent to the Configuration Server at the beginning of the agent connect. It includes information that is used to identify the managed device to implement policy, such as user identification, operating system, and computer name.

Service Dynamics

There are groups of data and applications (packages) that you want to control in your enterprise. A **package** is a data set that is published as an individual entity. It includes the files, desktop shortcuts, and registry entries that you need for an application, such as Microsoft Word. After you identify these packages, you categorize them into services. A **service** allows you to organize a group of related packages, methods, or behaviors into manageable units. One package could be a manageable unit.

For example, you may want to manage an anti-virus application. Usually, you will need the anti-virus software and a data file that holds virus information. Since you would have no use for the data file without the

software, you might group these two packages into one service. If you want to manage them separately, you would associate each of these packages with a separate service.

If you are creating a service in order to manage an application, the service might evolve as follows:

1. Use the Packager or Publisher to create a package and publish it to the CSDB.
2. Use the CSDB Editor to create and edit the properties of a service associated with packages.
3. Use the CSDB Editor to set **policy** for a user or group and, thereby, create your desired state. A policy defines to which applications subscribers and agent computers are entitled.
4. The Client Automation agent and the Configuration Server use the agent connect process and the resolution process to create the desired state. See Agent Connect on page on page 67 and Resolution Process on page 72.
5. The Client Automation agent completes the updates, removals, installations, and verifications that are needed to achieve its desired state.

Packaging versus Publishing

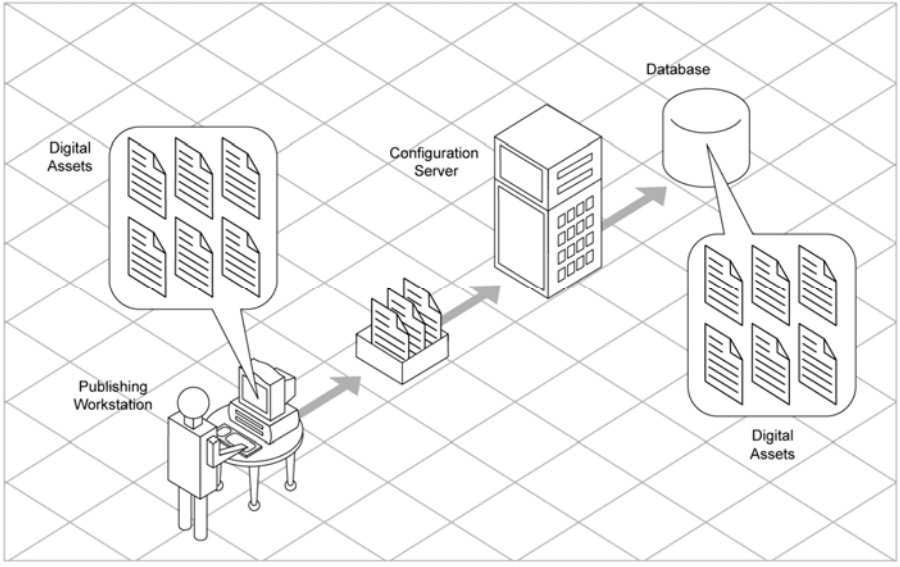
It is important to understand the distinction between these two Client Automation processes before you create a package and publish it into your Client Automation environment.

- **Packaging**
  is the process of identifying the components of the software that you want to manage, and organizing them into **packages**. Packages contain the files, shortcuts, links, and/or registry entries that make up the software. The software that you distribute can vary greatly—from a single data file, such as a company telephone list, to an entire application suite, such as Microsoft Office 2000.
- **Publishing**
  is the process of importing a package and its imbedded information into the CSDB. A package must be published before its content can be distributed and deployed into your environment.

Packaging

The two primary ways of creating packages are Installation Monitor Mode and Component Select Mode. Several factors influence which mode you will use: the complexity of the application, your knowledge of the application's structure, and whether to use the application's native installation capabilities.

Figure 15 Packaging data

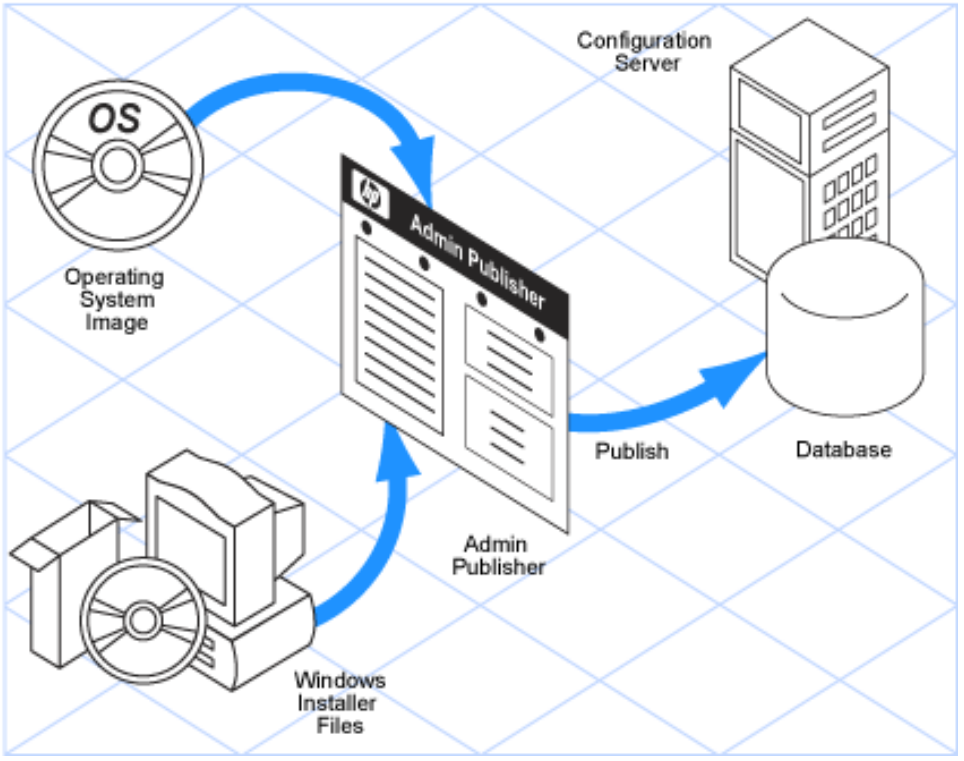


- **Installation Monitor Mode is a feature of the Packager.**
  In Installation Monitor Mode, the Administrator Packager determines what to package by scanning the computer before and after installing the software. It **differences** the before and after scans to determine what changes were made to the computer during the installation of the software. These differences make up the package that you promote to the CSDB. We recommend this mode for packaging when you do not know all of the components that make up the application.
- **Component Selection Mode is a feature of the Publisher.**
  In this mode, you select the individual components (such as files, directories, registry entries, and links) that make up the application. We recommend this mode for packaging simple data and applications where you can easily identify all the components in the package.

Publishing

After you create a package, you **publish** it to the CSDB.

Figure 16 Publishing process



The package is copied to the CSDB and several instances are created:

- An Application Packages (PACKAGE) instance that represents the promoted package.
- One File Resources (FILE) instance for each file in the package.
- One Desktop (DESKTOP) instance for each program group, link, and shortcut in the package.
- One Path (PATH) instance for each unique path to one or more components on the computer where the software is installed.
- One Registry Resources (REGISTRY) instance for each hive in the package.

Each instance described above is stored in one of the default classes, such as PACKAGE or FILE, in the SOFTWARE Domain. You can also add your own classes, such as a DLL class, to the CSDB.

After publishing the package, use the CSDB Editor to create a service and assign policies. Refer to the Entitlement chapter in the *Application Manager and Application Self-service Manager Guide* for more information.

Agent Connect

The purpose of the agent connect is to ensure that the device matches its desired state. The desired state embodies the data and entitlements for each device. A model representing the desired state for each device is stored in the CSDB.

The agent connect is initiated when a Client Automation agent object is sent to the Configuration Server. Typically, this is the ZMASTER object, which contains information about the managed device, such as its identity and IP address.

The ZMASTER object can be sent to the Configuration Server as a result of one of the following events:

- A scheduled Timer event.
  Timers are usually associated with a service. Use the Application Manager or Inventory Manager to trigger timers periodically or randomly within a specified time period.
- A Notify sent by the Configuration Server to the device.
  A Notify is a message that is sent to the managed device. Use the Application Manager or Inventory Manager to tell the device to start an agent connect.
- A refresh of the Service List on the managed device in the Application Self-service Manager.
  To manage services, the administrator first entitles the services. Then, the user uses the Service List to manage the installation, verification, removal, repair, and update of a service.

The agent connect process involves the following three stages.

1. **Tree Differencing** to download the new objects from the Configuration Server, create the **difference objects** (delta objects), and identify any data that needs to be retrieved.
2. **Data Transfer** in which data is downloaded to a temporary location.

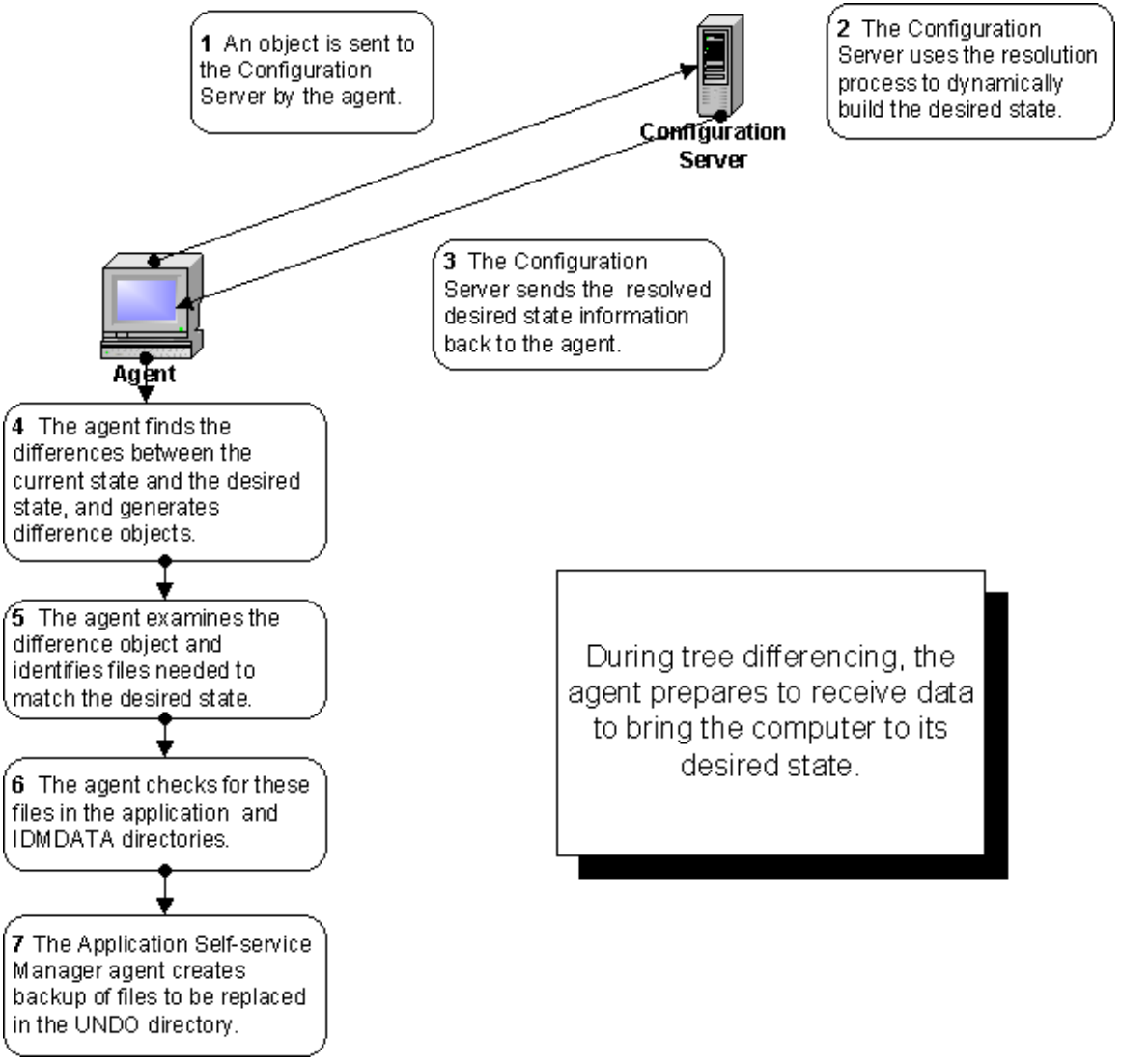
3 **State Machine** processing to install files from the temporary location to the live location, and create the new desired state objects to manage services.

Figure 17 Three phases of agent connect process

Tree Differencing

During the Tree Differencing phase of the agent connect the managed device identifies which files it needs to bring the device to the desired state. The agent differences the data between the configuration information on the device and the Configuration Server. First, the agent sends the ZMASTER object to the Configuration Server. Then, the Configuration Server builds the desired state based on the parameters designated for the user in the ZMASTER object. The desired state is sent back to the device as a new object, where the agent synchronizes the old and new objects between the server and the device. See Figure 18 on page 69.

Figure 18 Agent completes Tree Differencing process



Tree differencing works by using a reference list, which is like an object dictionary; it stores the different names for a particular class. The reference list is updated whenever a change is detected in the branches or leaves of the tree. The differencing algorithm relies on a name algorithm to generate predictable names for the difference object, the downloaded object, and the branch object.

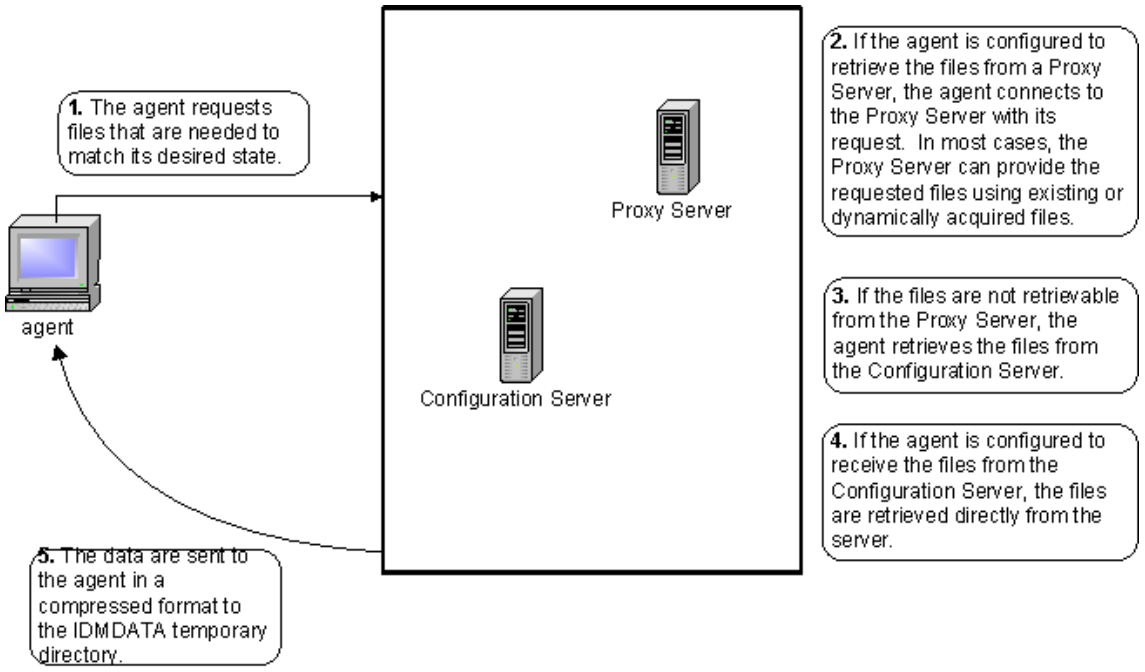
After generating the difference object, the agent determines if it needs to get any data files or install applications to bring the device to its desired state. The agent requests and downloads these files from the Configuration Server, or Proxy Server, during the Data Transfer stage of the agent connect.

Data Transfer

The Data Transfer phase of the agent connect begins when the agent sends a request for these files to the Configuration Server. If the agent is configured to retrieve files from the Proxy Server, the agent checks those servers for the files it needs. If the files are present, the agent downloads them. If any of the files cannot be retrieved from the Proxy Server, the agent retrieves the files from the Configuration Server. If the agent is not configured to use the Proxy Server, the files are retrieved directly from the Configuration Server.

The Configuration Server or the Proxy Server sends the data to the device in a compressed form and copies it to a pre-defined directory, `IDMDATA`, on the device. The `IDMDATA` directory is used as a temporary storage location on the managed device for these compressed files. After the files are decompressed and installed on the managed device, the compressed files are erased automatically if configured.

Figure 19 Agent receives application data

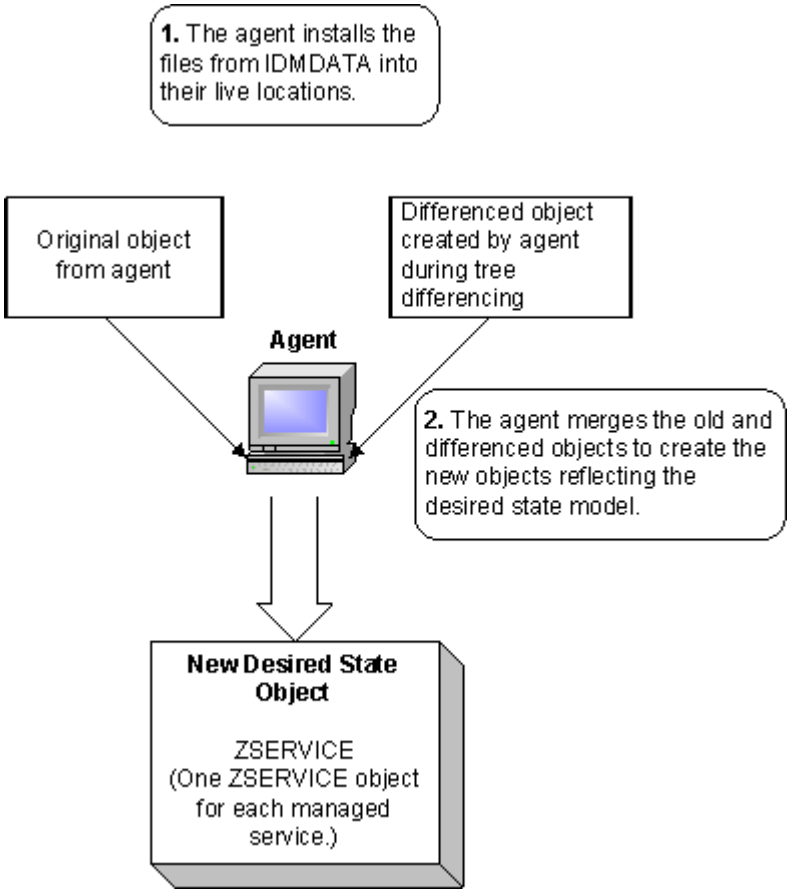


If you have multiple Configuration Servers, Proxy Servers, or if you want to store files for managing applications on a local CD-ROM, you may want to reconfigure the agent before connecting to the Configuration Server. Use Client Operations Profiles to prioritize and set criteria from where the managed devices should obtain their data. Refer to the *Application Manager and Application Self-service Manager Guide*.

State Machine

After the agent downloads the files (during the Data Transfer phase) that are needed to bring the managed device to its desired state, the agent installs the files from the `IDMDATA` directory. The agent erases the compressed files after they are installed on the managed device. Then, the agent merges the original object from the agent with the differenced object that was created during Tree Differencing.

Figure 20 State machine processing



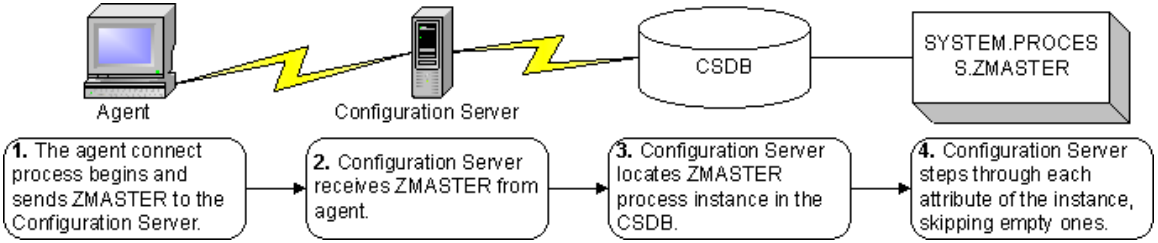
At the end of the agent connect the managed device's configuration should match its desired state in the CSDB.

Resolution Process

The Configuration Server uses the **resolution process** to accomplish a unit of work in response to a service request. The unit of work is defined by the contents of the CSDB and parameters included in the request. In other words, what the Client Automation infrastructure does depends on what information is stored in its CSDB and what information accompanies the request for Client Automation to perform some action. For example, the

agent connect submits service requests to the Configuration Server, and the Configuration Server performs a resolution in response to each request.

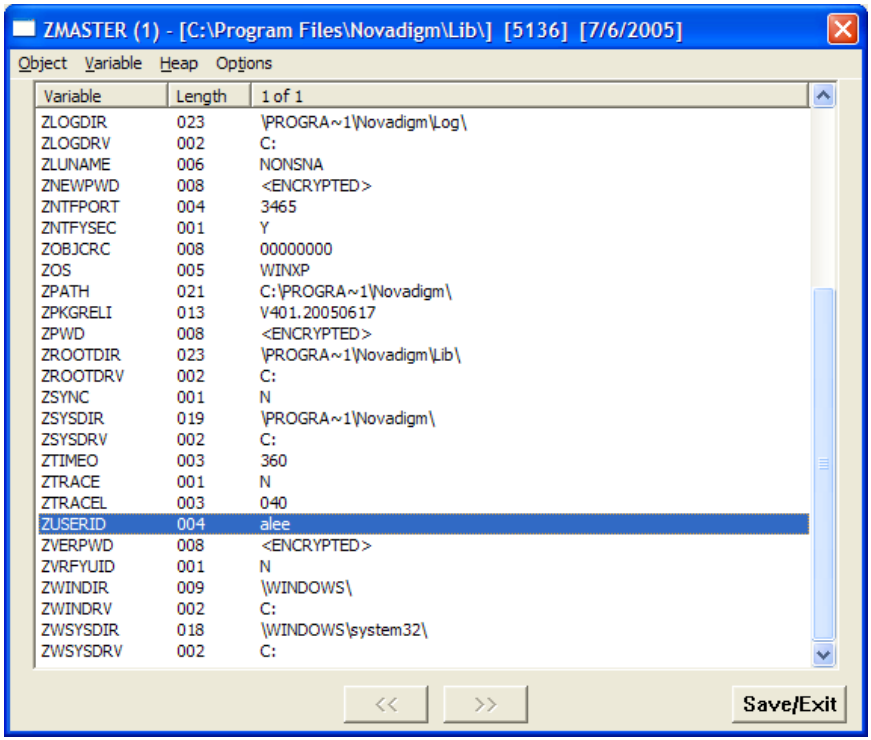
Figure 21 Configuration Server performs resolution



The ZMASTER object is sent to the Configuration Server during the agent connect. The ZMASTER object contains information about the agent computer that is needed to run Client Automation, such as the identity of the subscriber and the IP address of the agent computer.

The Configuration Server stores the ZMASTER object in **global memory**. Global memory is a temporary storage area in the Configuration Server. The Configuration Server maintains global memory's contents for the duration of the resolution process.

Figure 22 Subscriber's local ZMASTER object



After storing ZMASTER in global memory, the Configuration Server finds the Process instance for the ZMASTER. This is the **process entry point**. Its location is SYSTEM.PROCESS.ZMASTER.

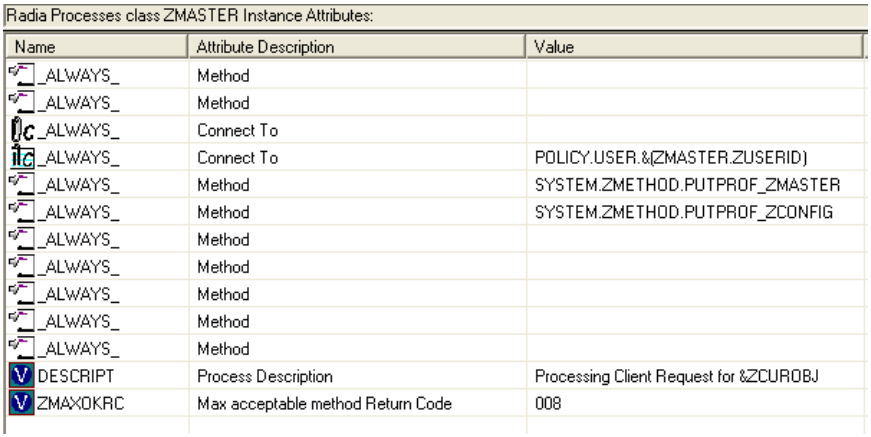
The Configuration Server reads each attribute of SYSTEM.PROCESS.ZMASTER. Based on an attribute's value, the Configuration Server may:

- Set variable values.
- Evaluate an expression.
- Execute a method.
- Connect to other instances.

If there is a connection to another instance, the Configuration Server processes the connected instance. Then the resolution process resumes in the referring instance at the next attribute after the connection attribute. For example, in Figure 23 on page 75, the first connection instance links to POLICY.USER.&(ZMASTER.ZUSERID). After processing this connection

instance, the resolution process will return to PRIMARY.SYSTEM .PROCESS.ZMASTER, and will process the next attribute that is a connection instance to SYSTEM.ZMETHOD.PUTPROF_ZMASTER.

Figure 23 PRIMARY.SYSTEM.PROCESS.ZMASTER instance

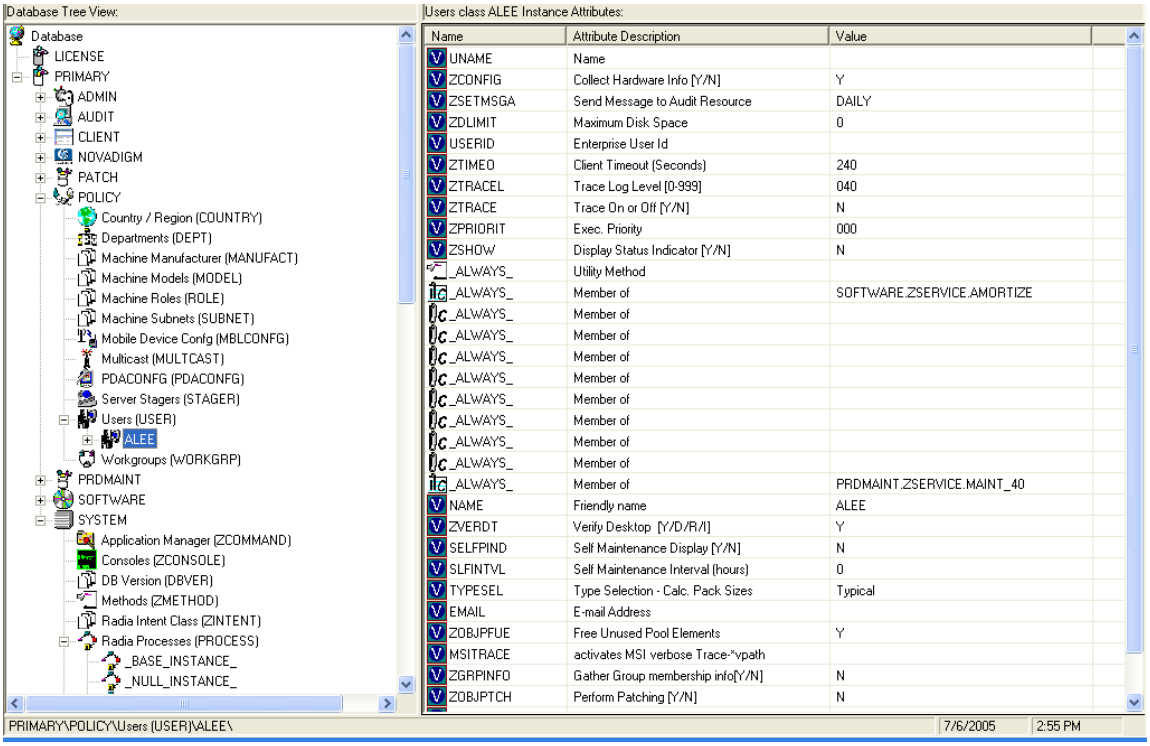


Radia Processes class ZMASTER Instance Attributes:

Name	Attribute Description	Value
ALWAYS	Method	
ALWAYS	Method	
ALWAYS	Connect To	
ALWAYS	Connect To	POLICY.USER.&(ZMASTER.ZUSERID)
ALWAYS	Method	SYSTEM.ZMETHOD.PUTPROF_ZMASTER
ALWAYS	Method	SYSTEM.ZMETHOD.PUTPROF_ZCONFIG
ALWAYS	Method	
ALWAYS	Method	
ALWAYS	Method	
ALWAYS	Method	
ALWAYS	Method	
DESCRIPT	Process Description	Processing Client Request for &ZCUROBJ
ZMAXOKRC	Max acceptable method Return Code	008

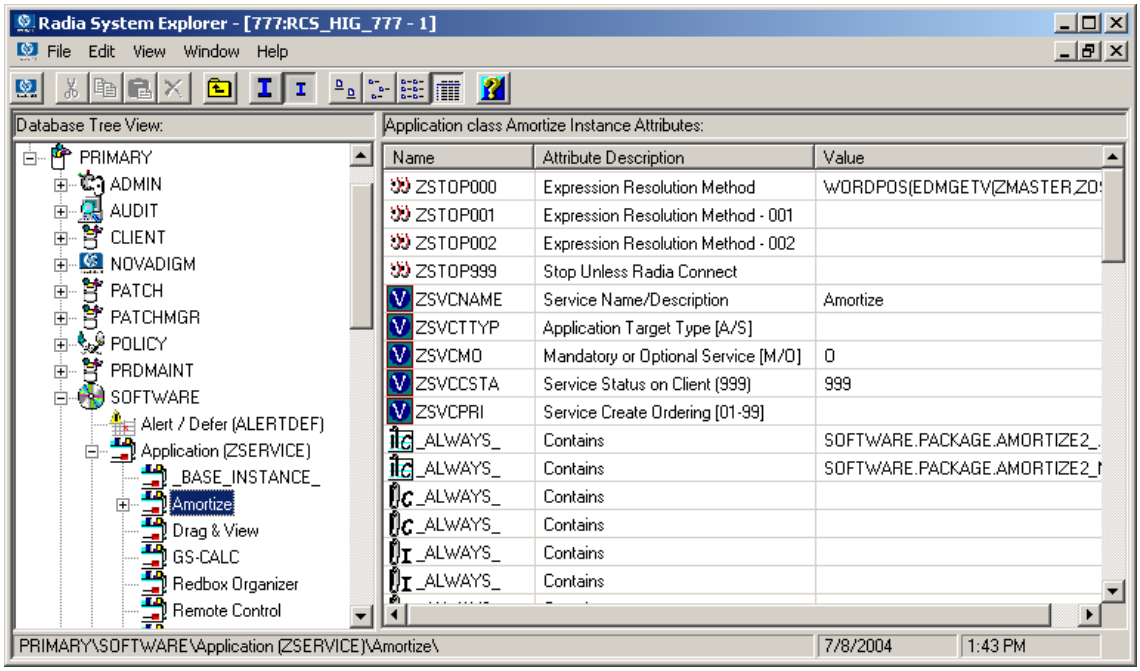
During resolution the Configuration Server performs **symbolic substitution** to set values and to connect to other instances. For example, in SYSTEM.PROCESS.ZMASTER there is a connection to POLICY.USER .&(ZMASTER.ZUSERID). The Configuration Server substitutes the value of the ZUSERID from the ZMASTER object that is in global memory. In Figure 22 on page 74, the value of ZUSERID is **alee**. Therefore, the resolution process will connect to POLICY.USER.ALEE, and resolve that instance.

Figure 24 POLICY.USER.ALEE instance



In Figure 24 above, after setting a number of variables, the first connection attribute is to SOFTWARE.ZSERVICE.AMORTIZE. In your implementation, the POLICY instance may connect to a workgroup that connects to a service.

Figure 25 ZSERVICE.AMORTIZE instance

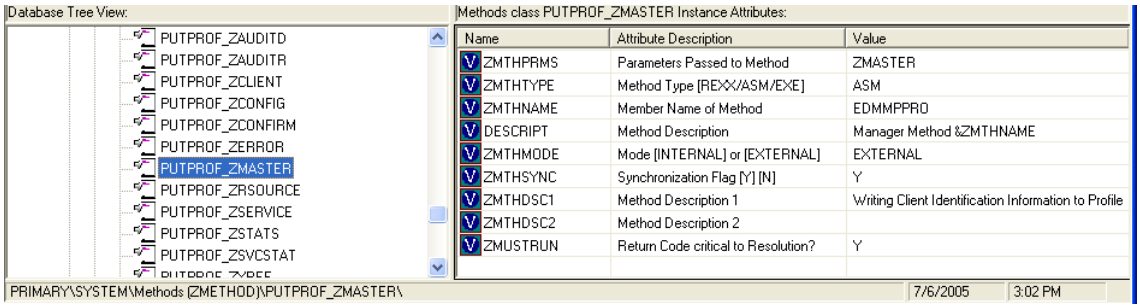


A Service instance links to packages. Figure 25 above begins with a ZSTOP expression variable. An expression variable contains statements that, if evaluated to “true,” stop the resolution of the current instance. An expression allows alternative paths to be taken during resolution based on variable data. In this case, the expression checks to be sure that the operating system of the agent computer is authorized for the Amortize software.

If the agent computer’s operating system is Windows 2000, Windows 2003, Windows XP, or Windows Vista, the resolution process continues with this instance, connecting the file instances, registry entries, path instances, and shortcuts. If the agent has an operating system other than one of the four that were previously mentioned, resolution returns to SOFTWARE .ZSERVICE.AMORTIZE, and to the next connection instance.

Eventually, the resolution process will return to the User instance, finish resolving it, and return to the process entry point, SYSTEM.ZPROCESS .ZMASTER. In Figure 23 on page 75, the next attribute connects to the PUTPROF_ZMASTER method. A **method** is a program that performs functions based on specified parameters.

Figure 26 ZMETHOD.PUTPROF_ZMASTER instance



The Configuration Server executes the EDMMPPRO method, passing ZMASTER as a parameter. This causes the contents of the ZMASTER object in global memory to be written to the PROFILE File of the CSDB.

After processing all attributes in the SYSTEM.PROCESS.ZMASTER instance, resolution terminates.

Inventory Collection

The following example of inventory collection shows how the Messaging Server and Reporting Server work together. The Inventory Manager Agent discovers configuration information on the managed device and reports it to the Messaging Server. The Messaging Server posts the information to a SQL database, and then you use the Reporting Server to view the results. **Web-based Enterprise Management** (WBEM) enables the collection of information such as the amount of RAM in a computer, hard disk capacity, process type, and operating system versions from computers, routers, switches, and other networked devices.

Windows Management Instrumentation (WMI) is the Microsoft implementation of WBEM for Microsoft Windows platforms.

1. The agent connects to the Configuration Server and sends agent objects to it. Some objects are always sent, others are sent only as a result of an inventory audit service being performed. The following information may be sent:
   - The APPEVENT object that describes the most recent service events.

— The ZCONFIG object that contains information about the device's hardware configuration.

— If a WBEM audit is performed and the agent is a WBEM consumer, WBEM objects will be sent.

2 The Messaging Server posts the objects to the appropriate ODBC data source. All inventory-related and audit objects are posted to the inventory database.

3 The Reporting Server accesses the data and allows you to view the inventory reports.

The Reporting Server allows you to view reports for Patch Manager, Application Usage Manager, and Application Management Profiles.

Proxy Server Processing

If the Client Automation agent has been configured to use a Proxy Server, it will attempt to retrieve files from the Proxy Server when resolving a service. The logical flow for a typical Client Automation agent request to a Proxy Server is as follows (assuming all components are enabled):

1 The agent sends a request to the Proxy Server.

2 The Proxy Server protocol front end receives the request.

3 This request is validated and passed to the main cache manager.

4 The local static cache is searched and, if the file is found, the request is satisfied.

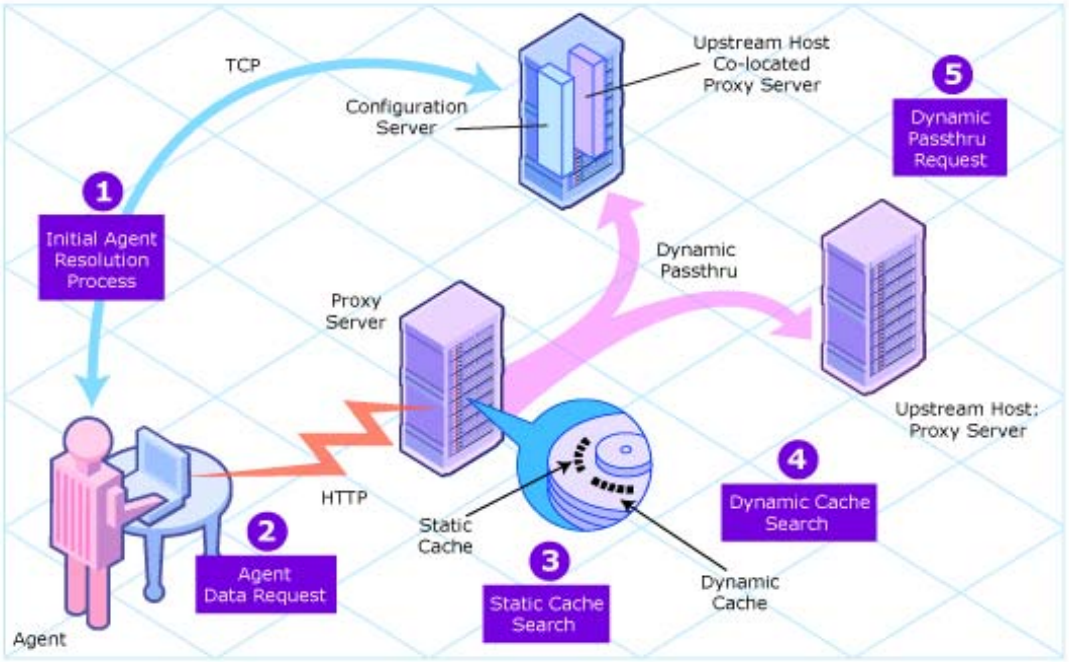
5 If the file is not in the static cache, the dynamic cache is searched.

— If the file is found, the request is satisfied.

— If the file does not reside in the dynamic cache, Dynamic PassThru requests the file from the defined upstream host (typically, the Configuration Server). The original request is satisfied, and the file is stored in the dynamic cache for future requests.

Figure 27 on page 80 is a simple overview of the process described above.

Figure 27 Proxy Server process



Static and Dynamic Cache

The **static cache** is managed by the Preloader component of the Proxy Server. This component manages this cache by connecting to the assigned Configuration Server, similar to a Client Automation agent. The static cache is typically **preloaded** during off-peak hours so that the required resources are available when requested. This is the primary cache location for the Proxy Server. For performance efficiency, this cache should be preloaded with all resources that are to be distributed by the Proxy Server.

The **dynamic cache** is populated on demand by the Dynamic PassThru component of the Proxy Server. When a requested resource is not found in the primary (static) cache, the dynamic cache is searched and populated if needed. This cache is viewed as a safety net for requests that fall through the static cache search. The Dynamic PassThru component also manages this cache and removes files that have not been requested in a specified number of days. You can define the maximum number of days this cache is defined in the Proxy Server configuration file.

Preloader

The Preloader component maintains the static cache by interacting with its defined Configuration Server. Required resources are placed into the cache, while resources no longer included in the Proxy Server model are removed from the cache. The Proxy Server's model is created according to the POLICY Domain on the Configuration Server.

Dynamic PassThru

When a Client Automation agent request is received for a resource that does not exist locally, the Proxy Server can request the resource from an upstream host such as the Configuration Server or another Proxy Server. The resource is then returned to the requesting Client Automation agent, as well as stored locally in the dynamic cache for subsequent requests.

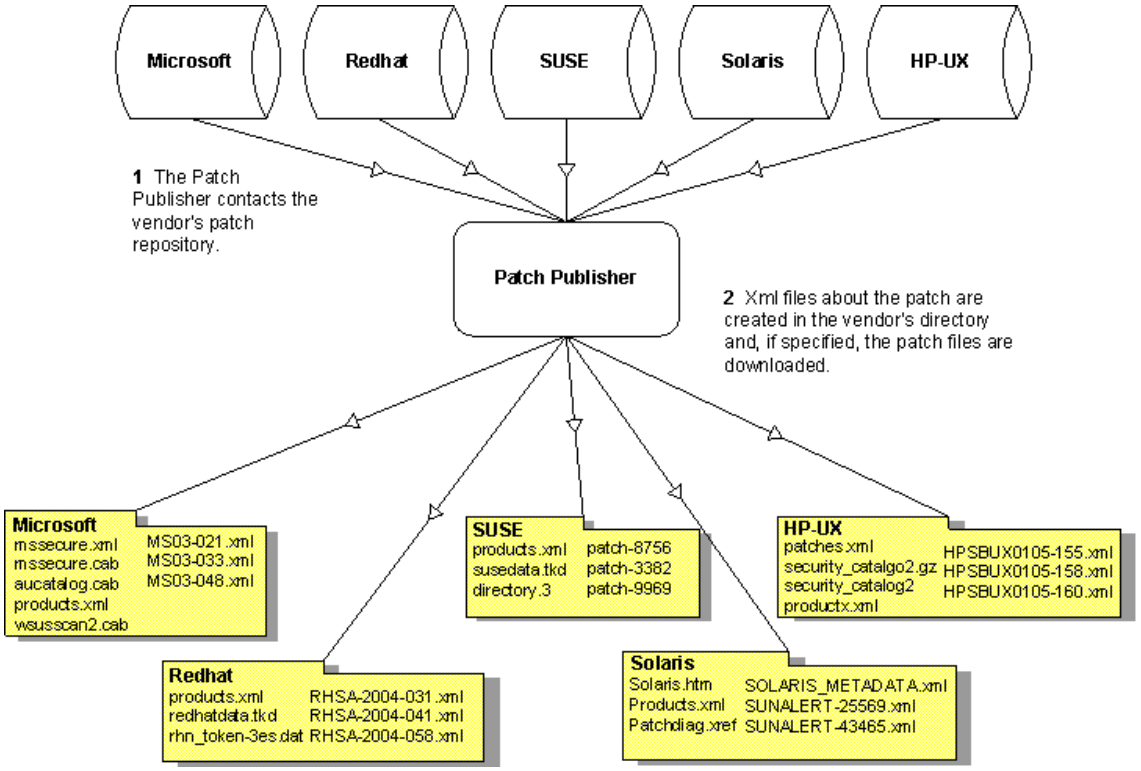
Patch Manager Acquisition

The Patch Manager acquires security patches and synchronizes the patch's information in the CSDB on the Configuration Server with the Patch database on a SQL or Oracle server. During the acquisition, the following happens:

- The vendor's web site is contacted to prepare for the acquisition of bulletins.
- Information about the security bulletins and service packs and (optionally) the actual patch files is downloaded. The information downloaded contains detailed data about each patch, such as supercedence, reboot requirements, and probe information.
- An XML file is created for each security bulletin that is acquired and is put in the vendor's folder in the Integration Server directory. These files are called **patch descriptor files**.
- The PATCHMGR Domain is populated with this information.
- Services are created in the PATCHMGR Domain for each of the acquired bulletins.
- The PATCHMGR Domain is synchronized with the SQL database you created.

If you have already performed an acquisition, only instances that are different are updated.

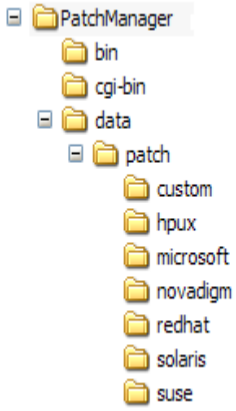
Figure 28 Vendor's patch repository is contacted



Patch Descriptor Files

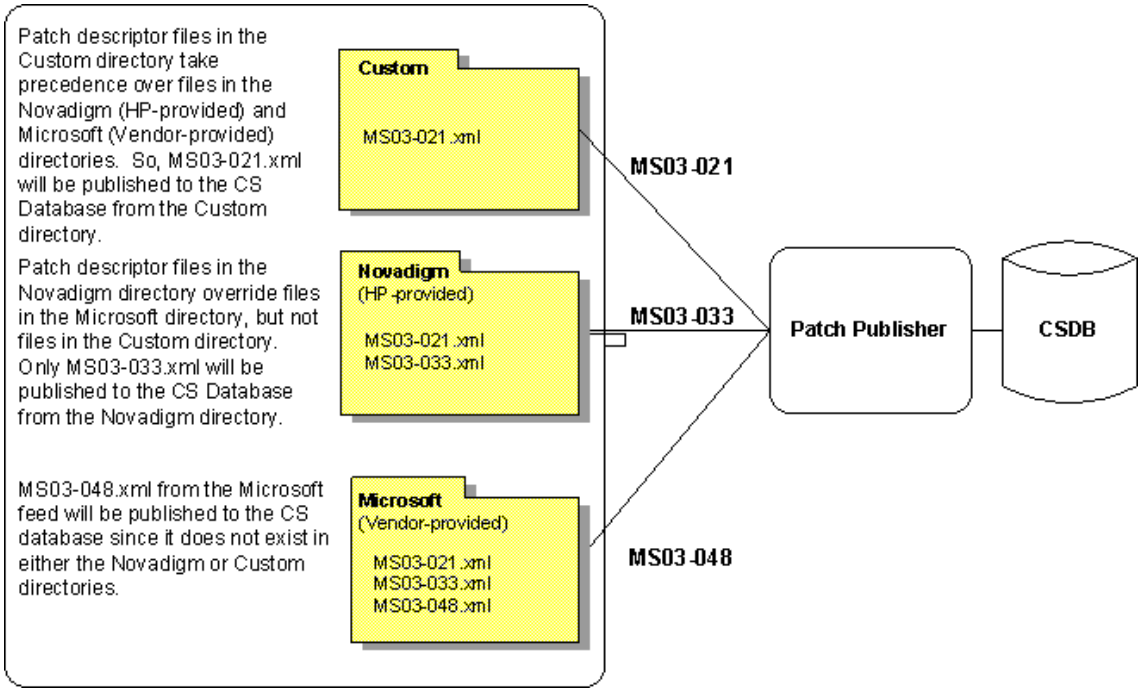
When security patches are acquired, an XML file (patch descriptor file) that contains information about the patch is created and placed in the vendor's directory. The vendor directories are located by default in `\\Hewlett-Packard\CM\IntegrationServer\Data\Patch`. For example, patch descriptor files for Microsoft bulletins would be in `\\Hewlett-Packard\CM\IntegrationServer\Data\Patch\Microsoft`. The security bulletin number is the file name with an XML extension. If the bulletin is identified as MS03-051, the patch descriptor will be named `MS03-051.xml`. If you also acquired the actual files associated with the bulletin, a folder is created with the name of the bulletin that contains the patch files.

Figure 29 Acquired patch descriptor file directory structure



You might need to change some of the information acquired from the vendor before the patch can be managed. Therefore, there are two other subdirectories in `\\Hewlett-Packard\CM\IntegrationServer\Data\Patch`. HP provides some additional patch descriptor files that are located in the `novadigm` subdirectory. The HP-provided patch descriptor files override patch descriptor files in the `microsoft` directory. You can also create or modify your own custom patch descriptors that will override the HP-provided files and those in the `microsoft` directory. Use a text editor to make the changes, name the file *exactly* as it is named in the vendor's directory, and place these XML files in the `custom` subdirectory.

Figure 30 Security patch descriptor files in custom override



A Publications

HP has a large Client Automation library. The following table helps you find more information about Client Automation products. Check the HP Software support web site for new publications and updates to current publications.

Table 10 Client Automation products and publications

Learn	Read
Management Applications	
Application Manager	• Application Manager and Application Self-service Manager Guide • Application Management Profiles Guide • Management Applications Messages and Codes Guide • Windows Terminal Server and Citrix Support Installation and Configuration Guide • REXX Programming Guide
Application Self-service Manager	• Application Manager and Application Self-service Manager Guide • Management Applications Messages and Codes Guide • REXX Programming Guide
Application Usage Manager	• Application Usage Manager User Guide • Knowledge Base Server Guide • Messaging Server Guide • Reporting Server Guide
Inventory Manager	• Inventory Manager Guide • Messaging Server Guide • Management Applications Messages and Codes Guide • Reporting Server Guide

Learn	Read
OS Manager	• OS Manager Guide
Patch Manager	• Patch Manager Guide • Messaging Server Guide • Reporting Server Guide
Management Infrastructure	
Administrator	
• Agent Explorer	• Administrator Guide
• AMP Editor	• Application Management Profiles Guide
• CSDB Editor	• Administrator Guide
• Packager	• Administrator Guide
• Publisher	• Administrator Guide
• Screen Painter	• Administrator Guide
Configuration Server	• Configuration Server Guide • Configuration Server Messages Guide • Getting Started Guide • REXX Programming Guide • Database Reference Guide
Extended Infrastructure	
Application Usage Manager	• Application Usage Manager Guide • Knowledge Base Server Guide
Distributed Configuration Server	• Distributed Configuration Server Guide
Enterprise Manager	• Getting Started Guide • Enterprise Manager Guide

Learn	Read
Multicast Server	• Multicast Server Guide
OS Manager	• OS Manager Guide
Patch Manager	• Patch Manager Guide • Messaging Server Guide • Reporting Server Guide
Portal	• Getting Started Guide • Portal Guide
Proxy Server	• Proxy Server Guide
Management Extensions	
Batch Publisher	• Batch Publisher Guide
Extensions for Windows Installer	• Extensions for Windows Installer Getting Started Guide • Extensions for Windows Installer User Guide • Configuration Analyzer Guide • Knowledge Base Server Guide
Policy Server	• Policy Server Guide • Enterprise Manager Guide • Portal Guide

Index

F

G

H

I

K

L

M

N

O

P

R

S

T

U

V

W

Z