Intel® Deployment Manager 3.3

by VERITAS OpForce™

User's Guide

Intel Disclaimer

This *Intel Deployement Manager 3.3 by VERITAS OpForce User's Guide* as well as the software described in it is furnished under license and may only be used or copied in accordance with the terms of the license. The information in this manual is furnished for informational use only, is subject to change without notice, and should not be construed as a commitment by Intel Corporation. Intel Corporation assumes no responsibility or liability for any errors or inaccuracies that may appear in this document or any software that may be provided in association with this document.

Except as permitted by such license, no part of this document may be reproduced, stored in a retrieval system, or transmitted in any form or by any means without the express written consent of Intel Corporation. Intel, Xeon, and Itanium are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Disclaimer

The information contained in this publication is subject to change without notice. Neither Intel nor VERITAS Software Corporation makes any warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. VERITAS Software Corporation shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this manual.

VERITAS Copyright

VERITAS Software Corporation
350 Ellis Street
Mountain View, CA 94043
USA
Phone 650–527–8000 Fax 650–527–2908
http://www.veritas.com

Third-Party Copyrights

Other names and brands may be claimed as the property of others.

Bean Scripting Framework

This product includes software developed by the Apache Software Foundation. (http://www.apache.org) Copyright (c) 2002 The Apache Software Foundation. All rights reserved. "Apache Software License" on page 291

BeanShell

This product includes a program that is subject to the Sun Public License. (http://www.sun.com/developers/spl.html) Copyright 1994-2003 Sun Microsystems, Inc. All rights reserved. "BeanShell License" on page 291

Castor

This product includes software developed by Intalio, Inc. (http://www.exolab.org) Copyright 2000-2002 (c) Intalio, Inc. All rights reserved. "Intalio Software License" on page 296

Cygwin

Cygwin is a UNIX environment, developed by Red Hat, Inc., for Windows. Copyright © 2000, 2001,2002 Red Hat, Inc. http://www.cygwin.com/ "Cygwin License" on page 296

Data Encryption Standard (DES) Copyright

Copyright (c) 1990 Dennis Ferguson. All rights reserved.

Commercial use is permitted only if products which are derived from or include this software are made available for purchase and/or use in Canada. Otherwise, redistribution and use in source and binary forms are permitted.

Copyright 1985, 1986, 1987, 1988, 1990 by the Massachusetts Institute of Technology. All rights reserved.

Export of this software from the United States of America may require a specific license from the United States Government. It is the responsibility of any person or organization contemplating export to obtain such a license before exporting. WITHIN THAT CONSTRAINT, permission to use, copy, modify, and distribute this software and its documentation for any purpose and without fee is hereby granted, provided that the above copyright notice appear in all copies and that both that copyright notice and this permission notice appear in supporting documentation, and that the name of M.I.T. not be used in advertising or publicity pertaining to distribution of the software without specific, written prior permission. M.I.T. makes no representations about the suitability of this software for any purpose. It is provided "as is" without express or implied warranty.

Editline

This product includes software developed by Simmule Turner and Rich Salz. Copyright 1992,1993 Simmule Turner and Rich Salz. All rights reserved. "Editline Software License" on page 304

EFI Application Toolkit

Copyright (c) 1998-2000 Intel Corporation. (http://www.intel.com/technology/efi/agreetoolkit.htm) See "EFI Application Toolkit Copyright" on page 296

Encode HanExtra

Copyright © 2002, 2003 by Autrijus Tang (autrijus@autrijus.org). This program is free software licensed under the terms of the Artistic license.

GNU Standard C++ Library

GNU Standard C++ Library is distributed under GPL with the 'runtime exception'.
(http://gcc.gnu.org/onlinedocs/libstdc++/17_intro/license.html) (http://gcc.gnu.org/onlinedocs/libstdc++/17_intro/COPYING) Copyright (c) 2000 Free Software Foundation, Inc. "GNU Standard C++ Library General Public License" on page 297

JAXP

Sun Microsystems, Inc. - Binary Code License Agreement and Java Interface Classes Java API for XML Processing (JAXP), version 1.1 Supplemental License Terms. (http://java.sun.com/xml/jaxp) "JAVA API FOR XML PROCESSING (JAXP), VERSION 1.1" on page 306

JDK

Sun Microsystems, Inc. - Binary Code License Agreement and Sun Java SDK 1.4.2 Supplemental License Terms.
(http://java.sun.com/j2se/1.4.2/j2sdk-1_4_1_02-license.html) "JAVATM 2 SOFTWARE DEVELOPMENT KIT (J2SDK), STANDARD EDITION, VERSION 1.4.2_X" on page 307

JDOM

This product includes software developed by the JDOM project. (http://www.jdom.org) Copyright (c) 2001 Brett McLaughlin & Jason Hunter. All rights reserved. "JDOM License" on page 309

JRE

Sun Microsystems, Inc. - Binary Code License Agreement and Sun JRE version 1.4.2 Supplemental License Terms. (http:/ /java.sun.com /getjava /license.html) "JAVATM 2 RUNTIME ENVIRONMENT (J2RE), STANDARD EDITION, VERSION 1.4.2_X" on page 308

Kernel.org

Kernel.org software is distributed under GNU General Public License. (http:/ /www.kernel.org) Copyright (c) 2000 Free Software Foundation, Inc. "Kernel.org License" on page 297

Ncurses

Copyright © 2000, 2001, 2002, 2003 Free Software Foundation, Inc. Permission is granted to copy, distribute, and /or modify this document under the terms of the GNU Free Documentation License, Version 1.1 or any later version published by the Free Software Foundation; with no Invariant Sections, with no Front-Cover Texts, and with no Back-Cover Texts. (http:/ /www.gnu.org /directory /libs /ncurses.html) "GNU GENERAL PUBLIC LICENSE Version 2, June 1991" on page 301

Net-SNMP

Net-SNMP Copyright 1989, 1991, 1992 by Carnegie Mellon University Derivative Work - 1996, 1998-2000 Copyright 1996, 1998-2000 The Regents of the University of California. All Rights Reserved. Copyright (c) 2001-2002, Networks Associates Technology, Inc. All rights reserved. Portions of this code are copyright (c) 2001-2002, Cambridge Broadband.

PostgreSQL Database Management System

PostgreSQL Database Management System may be used, if selected by the user. (<http:/ /www.postgresql.org>) Portions Copyright (c) 1996-2002, The PostgreSQL Global Development Group Portions Copyright (c) 1994, The Regents of the University of California. "PostgreSQL Software License" on page 310

PXE SDK for Linux

Contains Preboot Execution Environment technology licensed from Intel Corporation. Copyright © 1996-1999, Intel Corporation. All rights reserved. (http:/ /www.intel.com) PXE Software Copyright © 1997-2000, Intel Corporation. "Preboot Execution Environment (PXE) Software License" on page 310

SOAP

This product includes software developed by the Apache Software Foundation. (http:/ /www.apache.org) Copyright (c) 2002 The Apache Software Foundation. All rights reserved. "Apache Software License" on page 291

Tomcat

This product includes software developed by the Apache Software Foundation. (http:/ /www.apache.org) Copyright (c) 2002 The Apache Software Foundation. All rights reserved. "Apache Software License" on page 291

TridiaVNC

TridiaVNC is distributed under GNU General Public License. (http:tridiavnc.com) Copyright (c) 2000-2002 Tridia Corporation. All rights reserved. "GNU GENERAL PUBLIC LICENSE Version 2, June 1991" on page 301

Xerces

This product includes software developed by the Apache Software Foundation. (http:/ /www.apache.org) Copyright © 1999-2000, The Apache Software Foundation. All rights reserved. "Apache Software License" on page 291

VNC

Copyright © 1999 AT&T Laboratories Cambridge. All Rights Reserved. "GNU GENERAL PUBLIC LICENSE Version 2, June 1991" on page 301

Westhawk's Java SNMP Stack

Copyright (C) 2001 by Westhawk Ltd (http:/ /www.westhawk.co.uk) "Westhawk's Java SNMP Stack" on page 310

zlib

Copyright (C) 1995-2002 Jean-loup Gailly and Mark Adler. (http:/ /www.gzip.org) "zlib Software License" on page 311

Contents

Preface

Intel® Deployment Manager 3.3 by VERITAS OpForce™ automates administrative tasks involved with controlling, provisioning, and updating Intel Server Compute Blades within sophisticated, heterogeneous IT environments. These tasks include discovering and recording hardware and software inventory, tracking software licenses, provisioning blade servers, setting up networks and storage, and setting up administrative and user access.

Typically, each of these IT administration activities is handled by different functional groups within an organization. As each task is completed, there is a hand-off to another group who then starts the next set of activities, and so on, until servers, networking, storage, operating systems, and applications are ready for use. Intel Deployment Manager closes the gaps in the way computers, networks, and storage are managed today and helps you eliminate the manual hand-off between groups within the enterprise by automating and streamlining IT administration activities.

About this Document

This guide describes the initial setup of Intel Deployment Manager, the user interface, and procedures for performing IT management tasks on Intel Server Chassis and Intel Server Compute Blades. It is intended for use by system engineers, system administrators, network engineers, network administrators, data center operators, and IT and system engineers.

What's in this Guide?

This guide includes the following chapters and appendixes:

- "Introducing Intel Deployment Manager" describes features and illustrates a simple IT environment.
- "VERITAS OpForce Enterprise Edition" describes the VERITAS OpForce Enterprise Edition and tells you how to upgrade from Intel Deployment Manager.
- "Getting Started" shows you how to log in, and describes the administration console.
- "Using Table and Icon Views" shows you how to display and manipulate tables in the administration console Display Area, and describes the icon view that is available from the **Provisioned Resources** page.
- "Understanding Workflows" provides overviews of major workflows that you can accomplish using Intel Deployment Manager, including the sequence in which tasks are performed, and their dependencies. Each workflow has cross-references to specific procedures.
- "Working with Workspaces" describes actions that can be performed using workspaces. A *workspace* is a collection of all the resources that you can use.
- "Users, Roles, and Access Controls" describes how to set users, roles, and access controls to control user permissions and capabilities to access various resources.
- "Managing System Storage" describes how to add system storage to store snapshots and software.
- "Organizing Networks" describes how to set up networks to be managed by Intel Deployment Manager.
- "Preparing and Discovering Computing Devices" provides information about preparing machines for discovery, and describes automatic hardware and OS (Operating System) discovery processes.
- "Configuring Computing Devices" provides procedures for configuring computing devices that are managed by Intel Deployment Manager.
- "Managing Computing Devices" describes the tasks that you can perform using the **Computing Devices** page maintenance, power, and console menus.
- "Working with Snapshots" describes how to save snapshots, and provides information about the various actions that you can perform with snapshots.
- "Tracking Licenses" describes how to track application and operating system licenses on provisioned managed servers.
- "Provisioning Servers" provides an overview of the provisioning process, and discusses several methods you can use to provision servers.

- “Using Server Templates” shows you how to create server templates to make the provisioning process quicker and easier.
- “Managing Intel® Server Chassis and Server Compute Blades” shows you how to prepare Intel Server Chassis and Intel Server Compute Blades for management by Intel Deployment Manager, and provides management procedures for the blade servers.
- “Working With Software” describes how to create and deploy applications, patches, scripts, utilities, and other types of software on blades.
- “Working with OpScripts and Jobs” describes how you can add OpScripts, and add and use jobs using client-side scripts on blades.
- “Managing Events” describes the types of events logged, and how to view events in the custom log.
- The “Glossary” provides definitions of terminology used by Intel Deployment Manager.
- “Server and Server Asset States” provides tables that list system states for servers and server assets.
- “Constraints and the Constraints Editor” describes constraints, how to use the constraints editor, and how you can apply constraints to snapshots and computing devices.
- “Network Topologies” provides information about various network configurations.

Related Documents

The Intel Deployment Manager documentation set includes this guide and the:

- *Intel Deployment Manager 3.3 by VERITAS OpForce Release Notes*
- *Intel Deployment Manager 3.3 by VERITAS OpForce Installation Guide for Red Hat Linux*
- *Intel Deployment Manager 3.3 by VERITAS OpForce Installation Guide for Windows*
- *Online Help*

Conventions

Document Conventions

Typeface	Usage	Examples
`monospace`	Computer output, files, directories, account names, roles, permissions, events, event types, workspaces, software elements such as command options, function names, and parameters	Read tunables from the `/etc/vx/tunefstab` file. You must have the `Full Control` permission.
`monospace` (**`bold`**)	User input	# **`mount - vxfs /h/filesys`**
italic	New terms, book titles, emphasis, variables	See the *User's Guide* for details. The variable *vxfs_ninode* determines the value of...
Palatino (bold)	GUI objects (buttons, labels, categories, field names, selections)	Click **OK**. Enter the IP address in the **IP Address** field.
*	Indicates required field	**IP Address ***
$	Bourne / Korn / Bash shell prompt	
#	Superuser prompt (all shells)	
`[ ]`	In a command synopsis, brackets indicate an optional argument	`ls [ -a ]`
`|`	In a command synopsis, a vertical bar separates mutually exclusive arguments	`mount [ suid | nosuid ]`
blue text	Indicates an active hypertext link. In PDF and HTML files, click links to move to the specified location	http: / / www.veritas.com

VERITAS OpForce Enterprise Edition

This chapter describes the VERITAS OpForce Enterprise Edition and tells you how to upgrade to it from Intel Deployment Manager 3.3.

Enterprise Edition

Intel Deployment Manager 3.3 by VERITAS OpForce focuses on the management of Intel Server Chassis and Intel Server Compute Blades. Other OpForce features are available only in the Enterprise Edition.

VERITAS OpForce Enterprise Edition features include:

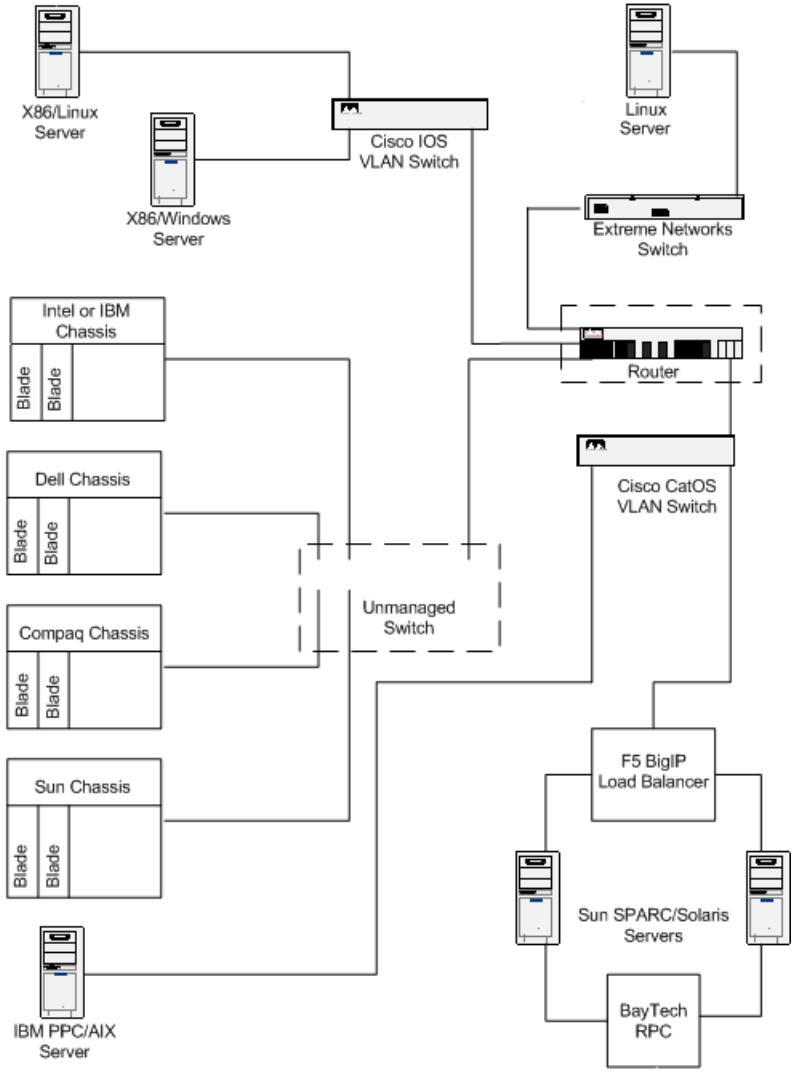
- The VERITAS OpForce server can be installed and run on a Microsoft Windows, Linux, or Sun® Solaris™ platform.
- Support for Intel, IBM®, Dell, HP / Compaq, and Sun server chassis management.
- Support for Intel, IBM, Dell®, HP / Compaq, and Sun blade management.
- Support for manual addition of managed and unmanaged assets.
- Support for VLAN Switches: including Cisco IOS®, CatOS, and Extreme Networks switch management.
- Support for F5 BigIP® load balancer management.
- Support for BayTech™ Remote Power Controller management (remote power management through the Intel Server Chassis *is* supported in Intel Deployment Manager 3.3 by VERITAS OpForce).
- Discovery of servers through PXE, RARP or bootp requests.
- Live server discovery: can be used instead of auto-discovery to discover computing devices and add assets. Live server discovery is useful if a server does not have a PXE enabled NIC, or for production machines you don’t want to reboot.
- Server Comparisons: You can detect differences in files on a provisioned server, or between two or more provisioned servers. You can also define a server comparison template for comparisons that you want to run more than once.

- Software Discovery: You can discover software applications on your servers including: packages on Solaris systems, RPMs on Linux systems, and software in the registry of Windows systems.
- Software Configuration: You can configure software by creating a software model that describes the software in detail and provides information about prerequisites, configuration files, parameters, and commands to run when configuring the software.
- Remote OS Installation: You can perform a fresh, remote install of an operating system onto a managed asset. Supported operating systems include Microsoft Windows 2000, 2003, and Red Hat Linux AS 3.0.
- Integrated Sysprep enables you to prepare a Windows 2000 or 2003 snapshot for automated deployment, even to machines with dissimilar hardware.
- NIC Teams: NIC teams enable a server to treat multiple physical NIC connections to a switch as a single logical link. Linux Bonding and BASP NIC teams are supported.
- Support for AIX 5.1 and 5.2 snapshots, including the ability to save images containing VxVM and LVM.
- Custom Fields: Custom fields are user-defined fields (attributes) that can be associated with computing devices, provisioned resources, or snapshots. A custom field can be displayed in the table view for these resources.
- Custom Actions: After creating custom OpScripts for computing devices, network devices, or provisioned servers you can quickly access and run them from the menu bar.
- Print View: you can convert table data into an HTML or Excel format (assuming Excel is installed on your system), and send it to a printer.
- SNMP traps: SNoMan (SNMP Notification Manager) can detect SNMP traps from your network devices and map them to user-defined events. When an event occurs, you can be notified. You can also trigger a job.
- Remote Desktop Access (access to the KVM on Intel Server Chassis is supported)
- The ability to add, import, and export scripts (Intel Deployment Manager 3.3 by VERITAS OpForce comes with two canned scripts
- Provisioning SDK / API

Note Routers are not managed by OpForce. You can choose whether or not to manage a switch. The devices enclosed in dotted-line boxes in the figure are not managed by the OpForce Enterprise Edition.

In addition to Intel Server Chassis and Intel Server Compute Blades, the VERITAS OpForce Enterprise Edition can manage most of the devices shown in the following figure, which illustrates a sample data center infrastructure managed by OpForce.

OpForce Enterprise Edition Managed Data Center



Upgrading to the Enterprise Edition

Contact VERITAS Software Corporation for information about upgrading to the VERITAS OpForce Enterprise Edition, or go to:

http://www.veritas.com/products/category/ProductDetail.jhtml?productId=itautomation

You will receive one or two product keys when upgrading to VERITAS OpForce Enterprise Edition.

▼ **Enter the product key(s) as follows**

1. In the Navigation Area, click **Settings**.
2. Click the **Product Keys** tab.
3. Click **Action** > **Add**, and enter the product key.
4. Repeat these steps if you have received two product keys (the keys can be entered in any order).
5. Log out, then log back in to OpForce.

Introducing Intel Deployment Manager 1

Intel Deployment Manager 3.3 by VERITAS OpForce automates administrative tasks involved with controlling, provisioning, and updating Intel Server Compute Blades within sophisticated, heterogeneous IT environments. You can optimize your existing IT infrastructure by using Intel Deployment Manager to implement on-demand computing services.

This chapter includes the following topics:

Introduction

Intel Deployment Manager enables you to optimize your existing IT infrastructure by implementing on-demand computing services. With Intel Deployment Manager, you can easily perform any of the following operations:

- Automatically discover your Intel Server Compute Blades in the bare-metal (no OS) or deployed state.
- Create server pools that can be shared among multiple network segments.
- Pre-provision a chassis slot so that a blade inserted into the slot is automatically provisioned and put into production.
- Rapidly deploy and provision servers in a multi-network environment.
- Create system snapshots that contain a complete software environment (OS, applications, data, network settings and personalization information) for your servers.
- Manage your server snapshots with built-in powerful cataloging and profiling capabilities.
- Remotely access and manage your Intel Server Chassis and Intel Server Compute Blades.
- Deploy applications, OS patches, and quickly upgrade BIOS, Firmware, and Hardware RAID configurations on multiple systems.
- Standardize on an integrated provisioning tool for your Linux and Windows environments.

Intel Deployment Manager 3.3 by VERITAS OpForce provides you with a hardware inventory of your chassis and blade, reports, and detailed event logs. Moreover, its role-based administration capabilities allow you to divide and compartmentalize server management tasks among your system administrators.

Intel Deployment Manager server runs on Linux and Windows platforms, and supports deployment, provisioning and configuration of Intel Server Compute Blades.

Features

The following is a more detailed list of Intel Deployment Manager features:

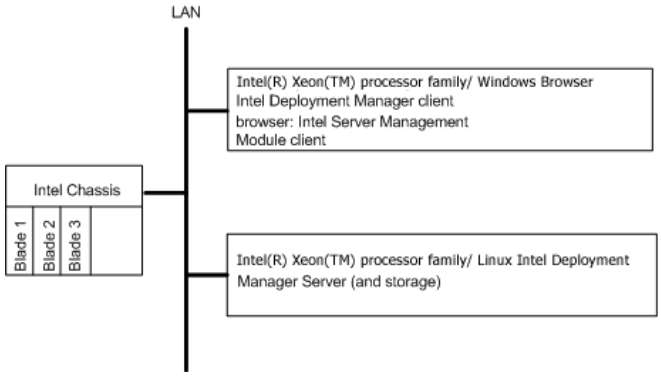
- Discovery
  - bare-metal server discovery
  - Intel Server Chassis and Intel Server Compute Blade discovery
  - support for Intel® Xeon™ processor family, and Intel Xeon processor family with 64-bit extension - Intel Server Compute Blades
- Inventory
  - inventory of resources
  - servers, chassis
  - server snapshots, application packages, software licenses
  - IP addresses
- Provision
  - image-based one-to-many server snapshot provisioning
  - pre-provisioning policies that define operations to be performed automatically on servers
  - server templates that provide a set of default values to be used when provisioning a server
  - full and incremental server snapshots with file-mode or sector-mode imaging
  - intelligent snapshots with partition maps, volume information, network configuration and license tracking
  - “point-in-time” saves of snapshots for replication or recovery
  - proportional partitioning to support images on dissimilar disks, or disks with different geometries
  - integration with Windows Sysprep for dissimilar hardware deployment
  - snapshot management with tree or table-based views
  - pre-boot configuration – RAID, BIOS, Firmware updates
  - post-boot configuration – application, patch installs, and updates
  - network configuration – multiple NICs, IP addresses, subnets

- Access
  - integrated, multi-platform, secure remote access to servers (desktop, console, shell)
  - Telnet, FTP, HTTP access
  - server pre-boot environment access
- Management
  - package-based one-to-many software deployment for applications, patches and updates
  - event management for troubleshooting and auditing
  - customizable reporting of managed resources
  - filtering and sorting inventory for selective group operations
- Interfaces
  - point-and-click graphical web interface
  - privilege and context-sensitive operational interfaces
  - use of a command generator to build scripts
- Security
  - directory-based authentication with industry-standard LDAP servers
  - role-based administration with users, user roles, and granular access control lists
  - grouping of users and resources for authorization levels
  - HTTPS / SSL encryption for GUI access
  - TCP / SEAL encryption for API access
  - authenticated and encrypted communication between management server, agents and network elements
- Platforms
  - The Intel Deployment Manager server can run on Microsoft® Windows® 2000 Server with SP3, Microsoft Windows 2000 Advanced Server, Microsoft Windows 2003 Enterprise Edition, Red Hat® Advanced Server 2.1, or Red Hat Enterprise Linux® 3.0 AS.
  - OSes supported on managed servers include: Microsoft Windows 2000 Advanced Server with SP3, Microsoft Windows 2003 Enterprise Edition, Red Hat Enterprise Linux® 3.0 AS Update 3, SuSE® Linux Enterprise Server 9

Describing the Intel Deployment Manager Environment

In a simple, single-network configuration, the Intel Deployment Manager 3.3 by VERITAS OpForce environment might look something like that shown in the following figure. In this case, the Intel Deployment Manager server is on the same LAN as the Intel Server Chassis and Intel Server Compute Blades it manages.

Simple Intel Deployment Manager Implementation



The Intel Deployment Manager server can run on Intel Xeon processor family Linux or Windows systems. See the Prerequisites chapter in the *Intel Deployment Manager 3.3 by VERITAS OpForce Installation Guide for Red Hat Linux* or the *Intel Deployment Manager 3.3 by VERITAS OpForce Installation Guide for Windows* for detailed information about supported platforms and releases.

Optionally, you can set up one or more separate system storage servers to store snapshots and software. You make system storage available to Intel Deployment Manager through an NFS or a Windows share.

Getting Started 2

This chapter describes how to get started using Intel Deployment Manager. It refers to installation guides, tells you how to obtain a product key, and talks about how to log in to the administration console (user interface). The administration console is described in detail.

This chapter includes the following topics:

Installing Intel Deployment Manager

Select an appropriate system on which to install the Intel Deployment Manager 3.3 by VERITAS OpForce server. The Intel Deployment Manager server should be on the same LAN as the Intel Server Chassis and Intel Server Compute Blades.

- If you want to install Intel Deployment Manager on a Windows system see the *Intel Deployment Manager 3.3 by VERITAS OpForce Installation Guide for Windows* for system requirements and the installation procedure.
- If you want to install Intel Deployment Manager on a Red Hat Linux system see the *Intel Deployment Manager 3.3 by VERITAS OpForce Installation Guide for Red Hat Linux* for system requirements and the installation procedure.

Logging into Intel Deployment Manager

Use any one of the following web browsers to display the administration console:

- Internet Explorer 6.0 (or higher)
- Netscape 6.2.3 (or higher)
- Mozilla 1.0 (or higher)

▼ To log in to the administration console

1. Bring up your browser and enter the URL for the Intel Deployment Manager server. Use the fully qualified domain name or IP address. For example:
   `http://`*`yourserver.yourcompany`***`.com`**.

   If you reassigned the web server port numbers when you installed Intel Deployment Manager, you need to specify the port number in the URL. For example:
   `http://`*`yourserver.yourcompany`***`.com:8080`**

   If you have not yet licensed Intel Deployment Manager, the **Product Key** page is displayed. To get a product key, see "Getting a Product Key for Intel Deployment Manager" on page 9.

   You can enter a product key and click **Add and Done**. Then, the login screen is displayed.

Tip When adding product keys, you can avoid typos by copying and pasting the provided key string.

2. If a security certificate is presented, accept the certificate for an HTTPS secure session.

3. Log in with a valid user ID and password. The user ID and password are case-sensitive.

   - If you are using default authentication you can log in as `Administrator`. The default password is `sysadmin`. You should change the `Administrator` password right away to maintain security.
   - If you are using LDAP (Lightweight Directory Access Protocol) server for authentication, enter your LDAP user ID and password. Contact your LDAP administrator if you do not have an LDAP user ID and password.

4. The administration console is displayed upon successful login.

Note The session timeout value is one hour by default.

Working With Product Keys

A product key is required to log in to Intel Deployment Manager for the first time.

You can add or delete product keys from the **Product Keys** page. The **Product Key Summary** page provides information about enabled features and current product key use.

Getting a Product Key for Intel Deployment Manager

To get a product key to manage chassis and any number of blades, go to http://www.veritas.com/opforcelicense and provide your contact information. A product key will be e-mailed to you within a few minutes.

Adding Product Keys

▼ **To add a product key**

1. In the Navigation Area, click **Settings** > **Product Keys**.
2. Click the **Product Keys** tab.
3. Click **Actions** > **Add**.
4. Enter a product key.
5. Click **OK**.

Deleting Product Keys

▼ **To delete a product key**

1. In the Navigation Area, click **Settings** > **Product Keys**.
2. Click the **Product Keys** tab.
3. Select the product key you want to delete.
4. Click **Actions** > **Delete**. The verification dialog is displayed.
5. Click **Yes** to delete this license.

Understanding the Product Key Summary Page

The **Product Key Summary** page displays information about your product key(s). The following types of information are provided:

- Enabled Features
- Current Usage Information

Enabled Features

This section lists all the features that are enabled for your product key:

Note By default, all features are enabled except for **Server Load Balancer Management and Network Provisioning**. These features are *not* applicable for Intel Deployment Manager.

- **Core License**—enables use of core Intel Deployment Manager features
- **Server Provisioning**—enables you to provision servers
- **Backup and Recovery**—enables backup and recovery operations
- **Server Personalization**—enables you to personalize servers with a unique host name and network configuration
- **Server Pre-boot Configuration**—enables you to deploy configuration software.
- **Code Deployment**—enables you to deploy software
- **Remote Desktop Access**—enables secure remote access to assets
- **Remote Power Management**—enables power cycling of assets
- **Network provisioning**—enables networks to be provisioned—*not* applicable for Intel Deployment Manager
- **Server Load Balancer Management**—enables support for load balancers—*not* applicable for Intel Deployment Manager
- **Scripting and Scheduling**—enables OpScripts and jobs

Current Usage Information

This section includes information about your environment:

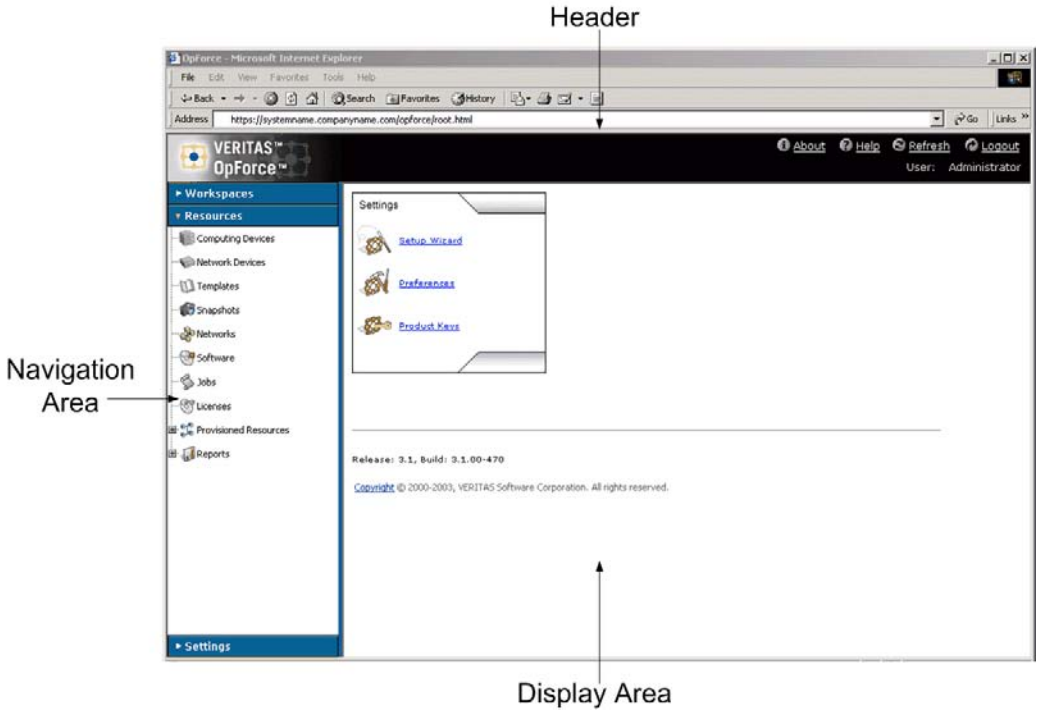
Note Non-IA CPUs are not applicable for Intel Deployment Manager.

- **Maximum Number of IA CPUs**—The value, **Unlimited**, indicates that an unlimited number of CPUs can be managed by Intel Deployment Manager.
- **Total Number of IA CPUs**—displays the total number of Intel architecture CPUs currently being managed.

Understanding the Administration Console

The administration console is divided into a Header, Navigation Area, and Display Area. In the following figure, the administration console is shown with the **Resources** section open.

Intel Deployment Manager Administration Console



Header

The Header portion of the administration console displays the Intel Deployment Manager logo and the **About**, **Help**, **Refresh**, and **Logout** buttons. Click the Intel Deployment Manager logo to go to the main page. The user name that you used when logging in is displayed in the header.

Navigation Area

The Navigation Area displays three sections: **Workspaces**, **Resources**, and **Settings**. All Intel Deployment Manager containers and their respective pages are divided among these categories.

Workspaces defines the various work areas that can be used by different users. Every workspace contains all of the resources assigned to it.

Resources consists of all the resources that are managed by Intel Deployment Manager (Computing Devices, Network Devices, and so on). Your ability to access these resources depends on your permissions. You can assign or unassign resources to workspaces using the **Actions** menu on the menu bar of most resource pages.

Settings enables you to customize Intel Deployment Manager according to your requirements. You can:

- add or delete users and storage
- define user roles and access controls
- load and unload drivers
- set preferences
- view the custom log, task manager, and the Intel Deployment Manager product keys

Display Area

When you click a container in the Navigation Area, the Display Area displays category tabs across the top of the area, such as **Discovered**, **Accepted**, **Server Assets**, **Blade Server Assets**, and so on. The tabs shown depend on the page you are on.

The menu bar, positioned immediately below the category tabs, displays various menus like **Actions**, **Table**, and others depending on the category you are working with. Selecting an option from one of these menus displays a dialog that enables you to perform an operation, like adding or editing a resource.

Setting Up Intel Deployment Manager

You should perform the following tasks when initially setting up Intel Deployment Manager.

1. Add a Network. See "Adding Networks *" on page 29.

2. Add storage for snapshots, software, and OpScripts. See "Adding Storage *" on page 29.

3. Define ranges of IP addresses for use by Intel Deployment Manager during provisioning of blades. See "Adding IP Lists" on page 29.

4. Prepare the chassis and blades for discovery. See "Preparing an Intel Server Chassis" on page 188.

5. Accept the chassis. See "Accepting Discovered Chassis" on page 28.

6. Accept the blades, and power-cycle them so that Intel Deployment Manager can start managing them. (See "Accepting and Power Cycling Intel Server Compute Blades" on page 196.) When you refresh the page, you'll see the server status change to UPDATE DONE.

7. Take full snapshots of your blades or import existing snapshots. For Windows-based blades see "Preparing to Save a Full Snapshot for a Windows System" on page 131.

8. Provision servers using your snapshots. See "Provisioning Servers" on page 163.

Optionally, you can perform other setup tasks like the following:

- Set up workspaces. See "Working with Workspaces" on page 41.
- Set up users, user roles and access controls. See "Users, Roles, and Access Controls" on page 45.
- Set up OpScripts and jobs. See "Working with OpScripts and Jobs" on page 221.
- Set up server templates for use in provisioning. See "Using Server Templates" on page 151.
- Set up provisioning policies. See "Pre-Provisioning" on page 177.

For specific information about working with chassis and blades, see "Managing Intel® Server Chassis and Server Compute Blades" on page 187.

Using Table and Icon Views 3

In the Display Area of the administration console you can view in table format detailed information about items under any tab. In addition to the table view, from the **Provisioned Resources** page, you can select an icon view that graphically represents servers on a selected network.

This chapter includes the following topics:

Using the Table View

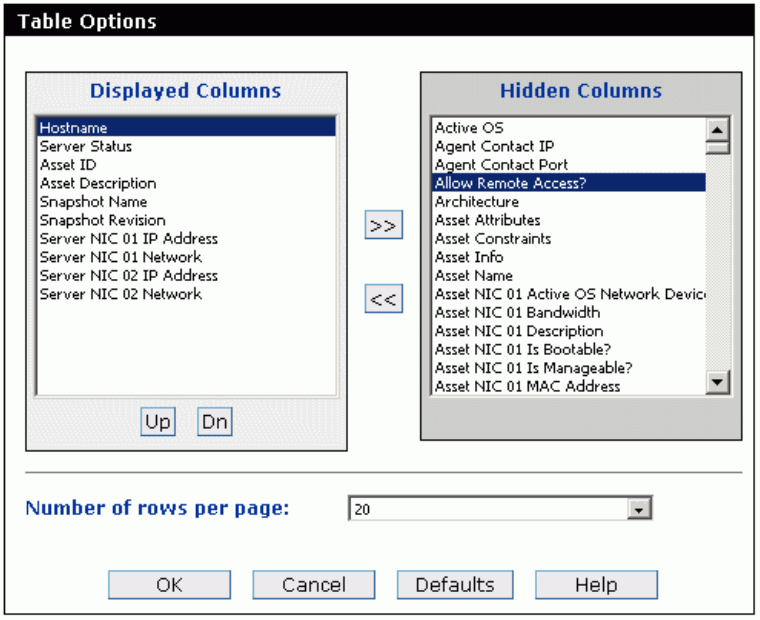
In the Display Area of the administration console you can view in table format detailed information about items under any tab.

The tables can be customized according to your requirements. You can select the columns you want to display, and determine the sort order. On Windows systems, you can export the table data into a format that can be used with other applications.You can also set an auto-refresh interval so that tables are updated at specified intervals.

Setting Table Options

Use the **Table Options** dialog to set the number of columns and the order of the columns displayed in a table, or to display or hide table columns. You can also set the number of rows to display per screen page. The changes you make to the table are preserved for your login user ID after you log out.

Table Options



▼ To set table options

1. Click **Table** > **Options**.
2. In **Displayed Columns** or **Hidden Columns**, select the column header you want to move.
3. Click the arrow (>>) or (<<) that points in the direction you want to move the column header. The selected column header now appears in the other column, **Displayed Columns** or **Hidden Columns**.
4. If you want to change the position of a displayed field:

- In **Displayed Columns**, select the column header you want to position. Click the **Up** or **Dn** buttons to move the column header to the desired position.
- Select a column header from the **Hidden Columns** section and move it to the **Displayed Columns** section. Notice that the newly displayed column header moves to the bottom of the list in the **Displayed Columns** section. If necessary, click the **Up** or **Dn** buttons to move the column header to the desired position.

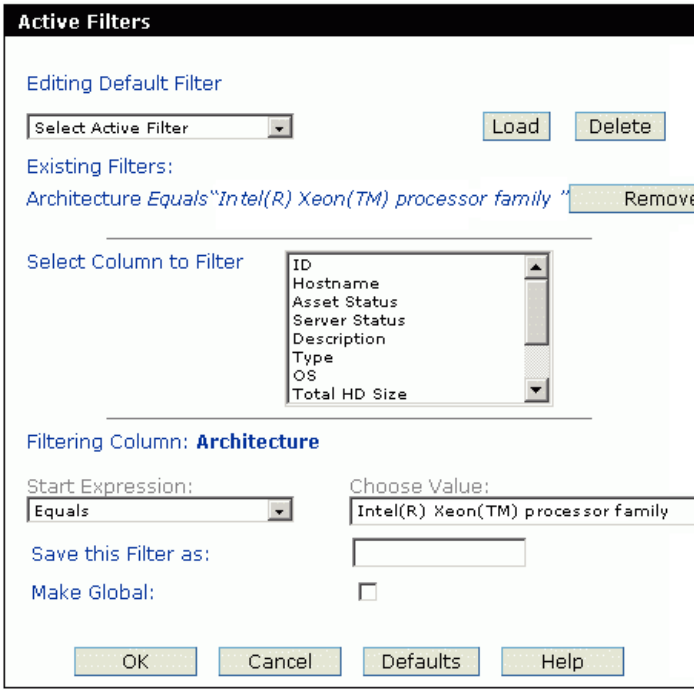
5. Select the number of rows to display per screen page. The default is 20.
6. Click **OK**. The change is reflected in the table, and stays that way until you update the columns again.

Note You can click the **Defaults** button to restore **Table Options** to default settings.

Using Active Filters

When any table (for example, a table for computing devices, snapshots, users, or servers) is displayed in the Display Area, you can apply *active filters*. Use the **Active Filters** dialog to define and save a particular set of filter values.

Active Filters



▼ To set or edit an active filter

1. If you want to edit an existing active filter, select it, and click **Load**. If no column headers are displayed, no existing active filters are available.

1. If no column headers are displayed, no existing active filters are available.

2. Select a column (attribute) to filter. The name of the selected column is displayed in the **Filtering Column** field.

3. Select an operator to use when filtering. Note that the operators available depend upon the column being filtered.

4. If the operator you selected in the **Start Expression** field has pre-defined values, you can select a value.

   For example, if you selected the **Architecture** column to filter, and an **Equals** start expression, you could choose **Intel Xeon processor family** from the **Enter Value** field. Then, your active filter would apply to all systems with an architecture equal to Intel Xeon processor family.

5. Repeat Steps step 2, step 3, and step 4 for each column you want to filter. All of the columns you have selected for filtering are listed under **Existing Filters**.

6. If you want to save this active filter for future use, enter a unique name for the filter in the **Save this filter as** field. For example, **`Intel Xeon processor family filter`**.

7. Click the **Make Global** check box if you want this active filter to be available to all users. Leave it unchecked if you want it to be available only for your user login ID.

8. Click **OK**.

9. Scroll horizontally so you can view the far right side of the current table. Select the active filter you want to use. The table data is then filtered based on the active filter you selected.

Sorting Table Columns

All tables can be sorted and customized. Tables are sorted by columns, by default, in ascending order. You can sort the order of the rows in the table based on any of the column headers.

▼ To sort table columns

1. Click an underlined column header in the table.
2. To sort data in descending order, click the underlined column header a second time. The sort preference is preserved when you exit from the table.

In the following figure, the up arrow (next to **Name**) shows that the table is sorted in the ascending order.

Sorting Tables

	Name ▲ [1]	Description	LDAP Group
☐	Developers	All Developers	
☐	Everyone	Every authorized user	
☐	Lab 160298	Lab 160298	
☐	Lab Managers	Lab Managers	
☐	Production 1	Production Level 1	
☐	Production 2	Production Level 2	
☐	Production 3	Production Level 3	
☐	Sysadmin	Elemental system administrators	%server.role.sysadmingroup%
☐	Tester	All Testers	
☐	Test Managers	Test Managers	
☐	Unix Systems Admin	Unix Systems Admins	

Sorting Multiple Table Columns

Multiple sorting enables you to sort data in a table based on more than one column. To sort by a second or subsequent column, press the CTRL key and click the column header.

Exporting Table Data

Note This procedure apples only for users of Internet Explorer on Windows systems. If Microsoft Excel is not installed on your desktop, the CSV extension is not available.

The **Export** option allows you to export table data into CSV (Comma Separated Values) to be used with applications such as Excel.

To save the exported data in a readable format, you need to associate an application with the selected file type.

▼ To export table data on your Windows system

1. Click **Start** > **Settings** > **Control Panel**.
2. Double-click **Folder Options**.
3. Click the **File Type** tab.
4. Scroll down the list to extension **CSV**. Click **Advanced**.
5. Click **New** to add the CSV extension. The **Edit File Type** window is displayed.
6. Check the **Confirm open after download** option.
7. Click **New**. The **New Action Window** is displayed.
8. In the **Action** field, type `open`.
9. Click **Browse** to select an application to associate with this action.
10. Select **notepad.exe**.
11. Click **OK**. Close all windows.

Setting the Table View Auto-Refresh Interval

By default, information displayed in a table is not updated until you perform a manual refresh. The status of a resource can change before you manually refresh the table.

You can set a auto-refresh value to automatically update some tables at regular intervals. Tables in the **Computing Devices**, **Network Devices**, **Snapshots**, and **Provisioned Resources** pages can be automatically refreshed.

Caution The default session timeout value is one hour. If you set a Auto-Refresh interval that is less than 1 hour, your session will never time out—this could pose a security risk. If you set a very low value, you might find it difficult to work effectively. For example, you might select several resources on a page and, before you can initiate an operation, the screen refreshes and wipes out your checkmarks.

▼ **To set or change the table view auto-refresh interval**

1. In the Navigation Area, click **Settings** >**Preferences** > **Auto-refresh Tables (User specific setting) In seconds, 0 disables auto-refresh**.
2. Click the link that displays the current value for the option.
3. Enter a value. This is the number of seconds between each refresh. Setting this value to 0 (default) disables auto-refresh.
4. Click **Submit**.

Note Setting a table view auto-refresh interval has no effect on the tree view for snapshots.

Using the Icon View

The icon view displays a graphical representation of servers on a provisioned network. You can use the icon view to display information about, or perform operations on servers. The icon view is only available for networks under **Provisioned Resources**.

Displaying an Icon View

▼ To display an icon view

1. In the Navigation Area, click **Resources.**

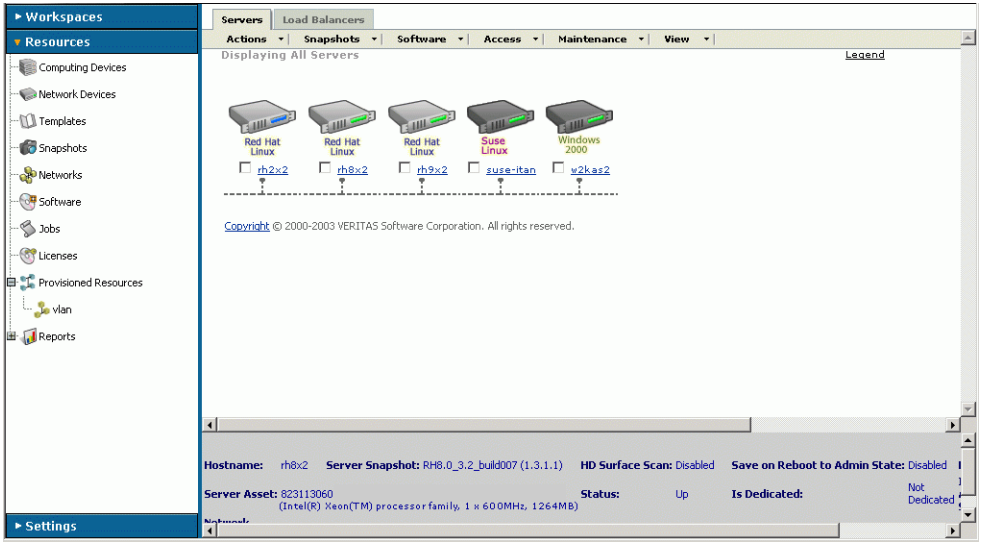
   Networks and IP lists are listed immediately under **Provisioned Resources**.

2. Click on the name of the network or IP list that contains the servers you want to view.

3. The icon view is the default view for provisioned resources, but if the last view your user ID used was the table view, a table view is displayed. Click **View** > **Icon** to switch back to the icon view. The icon view is displayed.

4. Optionally, click the **Legend** link to view the meaning of icon colors and symbols (light grey = autoselected servers, dark grey = dedicated servers, and so on).

Icon View



From the icon view you can:

- Place your mouse cursor over a server icon to display information about the server in the bottom portion of the screen.
- Click on a server icon to pop-up a status window for the server.
- Click on the underlined host name of a server to display an **Edit Server** dialog you can use to update information for the server. See “Editing Servers” on page 171.
- Click to select the check box for one or more servers, and then perform any operation from the menu bar for the selected server(s). For example, you can select a server and click **Access** > **Open** to open a terminal window to the selected server. Or, you could select several servers and then click **Maintenance** > **Reboot** to reboot them all simultaneously. See “Managing Computing Devices” on page 107 for information about operations you can perform from the menu bar.

Understanding Workflows 4

A *workflow* is the operational aspect of a business process. In this chapter you are given a set of workflows for performing basic tasks using Intel Deployment Manager. The purpose of each workflow is described, dependencies are noted, and references to specific procedures are given.

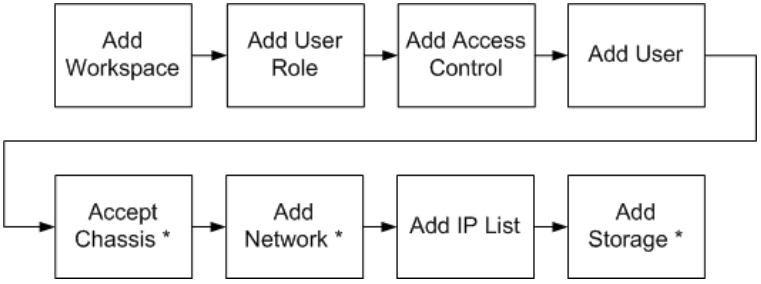
This chapter includes the following workflows:

Performing Initial Configuration (Using the Setup Wizard)

Intel Deployment Manager must be configured before it can be used effectively. Use the Setup Wizard to guide you through initial configuration tasks—just fill in the blanks as each dialog is displayed. You can click on a dialog's **Help** button to get information about fields, or click **Skip Step** to bypass tasks you don't want to perform now.

You must always add a network and add system storage before using Intel Deployment Manager, but other tasks are optional depending upon your environment and organizational requirements. The Setup Wizard presents tasks in the order in which they should be performed.

Setup Wizard Workflow



Note All Setup Wizard tasks can be performed individually as standalone procedures without using the wizard. Conversely, you can run the Setup Wizard at any time to perform any of these tasks.

Assumptions

- Your Intel Deployment Manager server is fully installed and connected to a network, and you are ready to start configuring it.
- You are logged in to the Intel Deployment Manager server from the administration console as `Administrator`, or as a user that has administrative permissions.
- Before starting the Setup Wizard, you set up a shared folder for system storage. (After clicking **Setup Wizard**, follow the online instructions presented in "Welcome to the Intel Deployment Manager Wizard".)

Adding Workspaces

A *workspace* is a collection of all the resources that can be assigned to a particular group of users. (See “Understanding Workspaces” on page 42.) All resources are automatically assigned to the `Everything` workspace, but you can add other workspaces as well. You can add workspaces as part of your initial configuration, or at any time afterward.

Before adding workspaces you should analyze your organizational needs and determine the workspaces you need. For example, you might want to add one workspace that contains all Linux-related resources, and another that contains all Windows-related resources. Or, you might want to allocate workspaces based on resources associated with functional groups like sales, marketing, finance, and so on.

Workspaces, user roles, and access controls are inter-related—if you add workspaces, you probably want to add user roles and access controls as well. Workspaces are tied to user roles using access controls. If you want to add a `Linux` workspace, you might also add a user role called `Linux Administrators` that has full control over everything in the `Linux` workspace. You could also have a user role called `Linux Users` that has permissions to use, but not update configurations for resources in the `Linux` workspace.

User roles can be assigned to one or more workspaces to control the resources to which they have access. IP lists can be assigned to workspaces to control user access to particular parts of the network.

See “Adding Workspaces” on page 43 for the standalone procedure and field descriptions.

Adding User Roles

A *user role* represents an arbitrary group of users who have a need to perform a similar set of tasks. See “Understanding Role-based Administration” on page 46.

Every Intel Deployment Manager user belongs to one or more user roles. The system comes with two pre-defined roles: `Sysadmin` and `Everyone`. The `Sysadmin` role has all privileges, but the `Everyone` role, which all users belong to by default, has no privileges. You can add additional user roles as part of your initial configuration, or at any time afterward.

Workspaces, user roles, and access controls are inter-related—if you add user roles, you probably want to add workspaces and access controls as well. Before adding user roles you should analyze your organizational needs and determine the user roles you need. If you want to define new user roles it is preferable to add them before adding users so that, when you add new users, you can assign them to roles.

See “Adding/Editing User Roles” on page 54 for the standalone procedure and field descriptions.

Adding Access Controls

An *access control* maps a specific user role to a workspace. If you have added workspaces and user roles, you need to add access controls to tie them together. Each access control has a set of permissions which identify the operations the user role is allowed to perform on the resources included in the workspace. Access controls for `Sysadmin` and the `Everything` workspace are defined so that users in the `Sysadmin` role can access all resources. See "Understanding Role-based Administration" on page 46 for information about access controls and their relationship to workspaces and user roles.

See "Adding/Editing Access Controls" on page 58 for the standalone procedure and field descriptions.

Adding Users

Intel Deployment Manager provides a pre-defined, all-powerful user called `Administrator` that is mapped to the `Sysadmin` user role. This is useful for system administration, but of course, you don't want to grant total control to every user on the system. You can add other users at any time. When you add a user, you should assign that user to a user role.

See "Adding/Editing Users" on page 49 for the standalone procedure and field descriptions.

Accepting Discovered Chassis

This task applies only to Intel Server Chassis. After preparing a chassis for discovery (see "Preparing an Intel Server Chassis" on page 188) the chassis is automatically discovered, but you must accept it before it can be managed.

Tip If you do not see your prepared chassis in the **Displaying All Discovered Chassis Assets** table shortly after beginning this task, refresh the display to update the table.

▼ To accept the chassis

1. Select the chassis you want to accept.
2. Click **Actions** > **Accept**.

See "Accepting a Discovered Intel Server Chassis" on page 192 for a standalone procedure.

Adding Networks *

Intel Deployment Manager operates within the network it manages. Resources you want to manage must be connected to the network that Intel Deployment Manager manages. See “Network Topologies” on page 273 for information about network configurations that can be used.

You must add a network before you can use Intel Deployment Manager—this is a basic requirement. You add a network by entering the network settings of the network you want to manage in the **Add Network** dialog. The network you need to add is the one that includes Intel Deployment Manager and the resources you want to manage.

See “Adding/Editing Networks” on page 62 for the standalone procedure and field descriptions.

Adding IP Lists

An *IP list* is a list containing a range of network addresses in an Intel Deployment Manager-managed network. Using IP lists, you can divide a network into multiple sections that can be assigned to different workspaces. Then, users can only manage systems in the section of the network (in the workspace) for which they have access permissions.

See “Adding/Editing IP Lists” on page 67 for the standalone procedure and field descriptions.

Adding Storage *

You must add system storage in order to have a place to store snapshots, software, and OpScripts—this is another basic requirement. Storage can be allocated on the Intel Deployment Manager server, but it is usually set up on one or more storage servers. You can add additional system storage at any time.

Intel Deployment Manager supports access to storage devices via NFS (Linux) and CIFS (Linux, Windows).

You should analyze your environment to determine your storage requirements. A full snapshot might require 1GB of system storage, but snapshots from very large systems can require many times this amount.

See “Adding System Storage” on page 72 for the standalone procedure and field descriptions.

Next Step

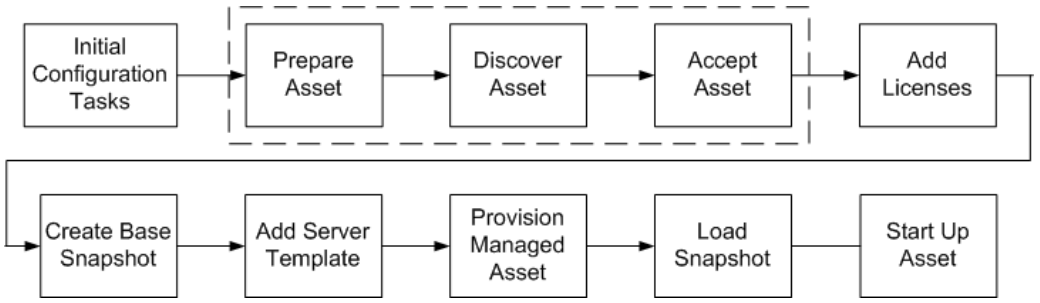
After performing initial configuration you can begin using Intel Deployment Manager. If this is a new installation, you can prepare for discovery the systems you want to manage, or create a full snapshot.

Preparing, Discovering, and Accepting Computing Devices

The Intel Deployment Manager term *Computing Devices* refers to servers, server assets, blade assets, and chassis assets that can be managed by Intel Deployment Manager. A *server* or *blade server* is an object that represents a server—it is a server asset with a snapshot, personalization information (host name and network configuration), and optional software. A *server asset* or *blade server asset* is a physical machine (hardware) that runs an ActiveOS.

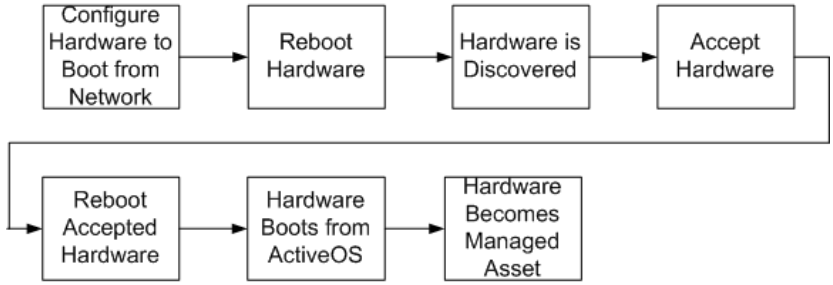
In order for Intel Deployment Manager to discover a machine it must be configured to boot from the network. Properly configured machines are automatically discovered after they are rebooted. After they are discovered, you must accept them before they can be managed by Intel Deployment Manager. Preparing, discovering, and accepting a computing device are all part of the overall *auto-discovery* process.

Preparing, Discovering, and Accepting Workflow



The steps that occur as part of the preparing, discovering, and accepting workflow are illustrated in the following figure.

Steps for Preparing, Discovering, and Accepting Workflow



Information about how to perform these steps is provided in the following sections.

Assumptions

- You have successfully performed initial setup tasks (minimum requirements: adding a network and adding storage).
- Your computing device has at least one NIC, and is connected to a network managed by Intel Deployment Manager.

Preparing the Hardware

Preparing a machine for discovery always involves setting the system to boot from the network before booting from a hard disk. Other preparation tasks are specific to the architecture of the asset being prepared.

For Intel Server Compute Blades see “Preparing an Intel Server Chassis” on page 188.

Discovering Machines

After preparing a machines for discovery, reboot it. The machine is then automatically discovered by Intel Deployment Manager. You can click **Resources** > **Computing Devices**, and then click the **Discovered** tab to view discovered machines.

The *ActiveOS* is used by Intel Deployment Manager to perform automatic OS and hardware attribute detection.

Accepting Server Assets

You must accept the asset before it can be managed. See “Accepting Computing Devices” on page 103. For Intel Server Compute Blades, also see “Accepting and Power Cycling Intel Server Compute Blades” on page 196.

Reboot your Intel Xeon processor family machine after accepting it. After rebooting, the machine begins running ActiveOS, and is then a server asset (or blade asset). Intel Server Compute Blades are based on Intel Xeon processor family architecture, so references to Intel Xeon processor family in this guide can be viewed as references to your blade servers.

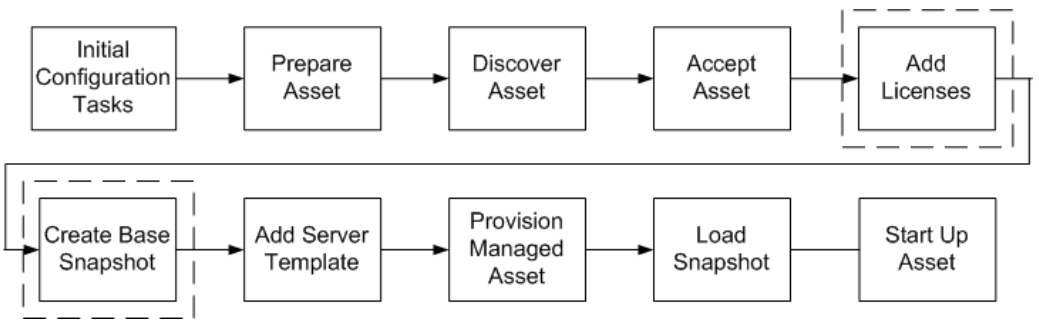
Next Step

You can create a full snapshot for an accepted server asset, or you can provision the server asset and load a snapshot onto it.

Saving a Full Snapshot from a Server Asset

A *snapshot* is an Intel Deployment Manager package that includes an OS, possibly other software and data, and a configuration. A *full snapshot* is a complete image of the server asset’s hard disks. You might take a full snapshot for disaster recovery purposes, to use in the provisioning of servers, or just to provide a base so you can later take incremental snapshots. A snapshot can optionally have an associated set of licenses for the OS and application software included in the image.

Saving a Full Snapshot Workflow



Assumptions

- An OS is installed and configured on the local hard disks of a server asset from which you want to take a full snapshot (other software is optional).
- If you want to take a full snapshot of a MicroSoft Windows server asset that uses an NTFS filesystem, you installed an ITAP agent on one of the Windows local hard disk partitions. (This is not necessary for Linux systems.)

Adding Licenses

Intel Deployment Manager provides the means to track application and operating system licenses for provisioned managed servers—every snapshot can have an associated set of licenses. Adding licenses is optional. After you add a license, you might associate it with one or more snapshots. See “Understanding Licenses” on page 146.

To add licenses, follow the procedure given in “Adding/Editing Licenses” on page 147.

Creating a Full Snapshot

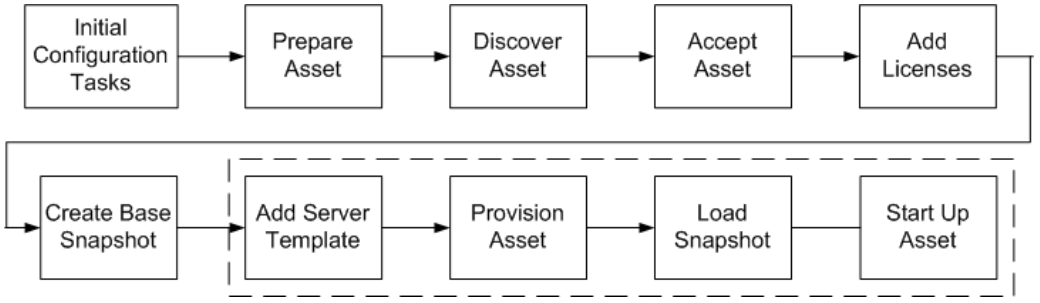
To create a full snapshot, follow the procedure given in “Saving Snapshots for Computing Devices” on page 123. Note that you can only create a full snapshot from a server asset, and you can only initiate a snapshot save for a server asset or blade asset (which is another type of server asset) from the **Computing Devices** (not **Provisioned Resources**) page.

Provisioning Servers

The *provisioning* process involves creating a server object, placing a snapshot onto a physical server asset, *personalizing* the configuration (host name and network), and booting the server asset from its local hard disk into a server with an OS and the personalized configuration. A *server object* is an object for which you have defined a snapshot, included personalization information, possibly included hardware attribute information (beyond that already included in the snapshot), and for dedicated assets, includes assignment of physical machines. You might perform provisioning to change a bare-metal machine (a machine that has no OS) into a functioning server, or to re-purpose an existing server. Servers provisioned by Intel Deployment Manager can be fully managed by Intel Deployment Manager.

We normally think of provisioning as an overall process, but note that the term provisioning can also refer to a specific portion of the process, as shown in the Provision Asset task shown in the following figure. This provisioning task corresponds to the part of the process that creates a server object.

Server Provisioning Workflow



Assumptions

- You have already performed required initial configuration tasks.
- A disk-based OS is already fully installed on the assets you want to provision.
- You have already performed the earlier tasks in the workflow, such as creating a full snapshot, and preparing, discovering, and accepting the assets to be provisioned.

Adding Server Templates

A s*erver template* provides a set of default values to be used when creating a virtual server—this makes the provisioning process much faster and easier. Using a server template is optional, but its recommended if you intend to provision assets frequently. See “Adding / Editing Server Templates” on page 153.

Given an existing template, you can add a similar template and customize it to meet different requirements. You can maintain as many server templates as you need. See “Adding a Similar Server Template” on page 159.

Provisioning Assets

You can provision servers manually, from a server template, or by running an OpScript. You can also set pre-provisioning policies that define provisioning operations that should be performed automatically on a server when specified conditions are met.

You can use any one of several methods to provision servers.

- Provision Managed Servers—Use this procedure to manually provision one or more servers that are already managed. This procedure is for provisioned resources only. The server allocation can be dedicated (you specifically select the servers to be allocated) or auto-selected. Auto-selected allocation supports use of a parameterized host name. See “Provisioning Managed Servers” on page 164
- Provision Servers using a Server Template—A server template provides a set of default values to use when provisioning servers. There are several different ways to provision servers using a template. You can initiate provisioning of server assets using a template from the **Computing Devices** page, provision servers from the **Provisioning Resources** page, or provision either server assets or servers using an OpScript. Each of these methods is discussed in “Provisioning Servers Using a Server Template” on page 172.
- Pre-provisioning— Pre-provisioning enables you to specify rules and actions so that, if an auto-discovered server meets specified conditions, the actions are automatically performed. For example, you can designate a particular chassis slot to automatically accept, provision, load a snapshot, and start up a blade after it is installed in a slot and discovered. See “Pre-Provisioning” on page 177.

Loading Snapshots

Load a snapshot as shown in “Loading a Snapshot” on page 183. If you intend to start up the server immediately after the snapshot is loaded, you can combine the load and startup operations as shown in “Loading a Snapshot and Starting Up a Server” on page 185.

When a snapshot is loaded the server is personalized with the host name and network configuration that you defined when you created the virtual server.

The server automatically reboots if you load and startup a server (Windows servers automatically reboot multiple times).

Starting up Assets

Any time after a snapshot is loaded, you can start up a server as shown in “Starting Up a Server” on page 184.

Next Step

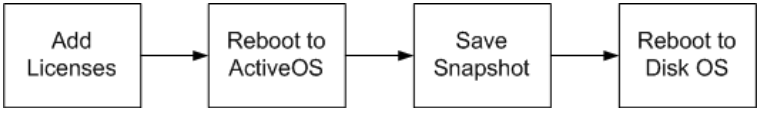
After a server is up and running, you can use it as a production server. You can also deploy software on it, or take a full or incremental snapshot from it.

Saving Snapshots from Managed Servers

You can save full or incremental snapshots from managed servers. A *full snapshot* is a complete image of a managed server's hard drive(s) including the OS, and all software and data on the system. Any full snapshot can be used to provision a bare-metal machine. You must have a full snapshot that provides a base before you can save an incremental snapshot. An *incremental snapshot* contains only the changes (delta) since the last full or incremental snapshot. See "Understanding Snapshots" on page 120.

The steps that occur as part of the saving full or incremental snapshot workflow are illustrated in the following figure.

Steps for Saving Full or Incremental Snapshots



Information about how to perform these steps is provided in the following sections.

Adding Licenses

Intel Deployment Manager provides the means to track application and operating system licenses for provisioned managed servers—every snapshot can have an associated set of licenses. See "Understanding Licenses" on page 146.

Adding licenses is optional. After you add a license, you can associate it with one or more snapshots. For incremental snapshots, you only need to add licenses for new licensed software that was added since the last full or incremental snapshot.

To add licenses, follow the procedure given in "Adding / Editing Licenses" on page 147.

Saving a Full or Incremental Snapshot

You can save a full or an incremental snapshot at any time. When you save an incremental snapshot, Intel Deployment Manager boots the server from the network into ActiveOS. When the snapshot save is completed, it is rebooted back to its original state.

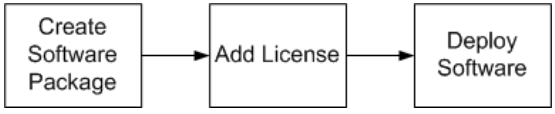
To save a full or incremental snapshot, follow the procedure given in "Saving Snapshots for Provisioned Resources" on page 126. Note that you can only initiate a snapshot save for a provisioned server from the **Provisioned Resources** (not **Computing Devices**) page.

Creating and Deploying Software Packages

Software and files packaged in ZIP format can be loaded into Intel Deployment Manager and deployed to managed server assets and servers. The software can be an application, patch, script, utility or other category of software. Additionally, several software types are supported: bootable software, configuration software, and control software. See "Understanding Software Categories", and "Understanding Software Types" on page 202.

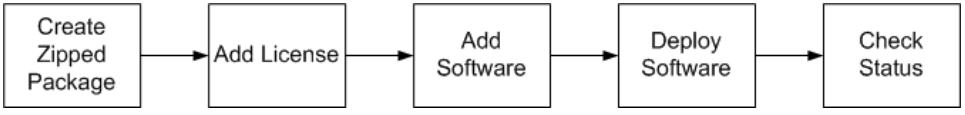
This workflow assumes you are creating and deploying configuration software to a server. This is the most common case. The server(s) you deploy software to can be provisioned at any time before the deployment.

Creating and Deploying Software Workflow



The steps that occur as part of the creating and deploying software workflow are illustrated in the following figure.

Steps for Creating and Deploying Software



Assumptions

- You have configuration software you want to distribute to target servers.
- The server(s) you want to deploy software to are managed.

1. Create a ZIP package that contains the software you want to deploy.
   - Because the software installs silently (without prompts) the package should include an answer or response file if the software requires input as it installs. The answer file provides answers to any questions that the software asks. For example, it might agree to license terms and supply the name of the directory where the software is to be installed.
   - Optionally, provide a script to set up the environment for, and run the software on the target system(s). For example, you might create a shell script on a Linux system that makes a directory, copies various files into the new directory, changes the directory's permissions, and executes a program.
   - Use WinZip, pkzip, or some other ZIP utility to zip the package.
2. Optionally, add a license for the software you want to deploy. See "Adding/Editing Licenses" on page 147.
3. Add the software package to Intel Deployment Manager. See "Adding/Editing Software" on page 206. The software package is loaded into system storage. When adding the software, key fields are:
   - **Command Line**—Enter a command to launch the software or script on the target server(s); for example, *./installme*. You must enter the path of the executable relative to the current directory. The current directory is set to the directory where the uploaded .zip file was unzipped.

     On Linux and UNIX systems, the command line is executed by exec(). On Windows systems, the command line is executed by CreateProcess(). Therefore, a command line with multiple commands must be grouped and launched by a shell. For example, on a Linux system: **`/bin/sh/ -c ".installme; .configureme"`**.
   - **Return Value**—Define a value to indicate successful completion.
   - **Causes Reboot**—Be sure to select this option if the software you are deploying can reboot the system.
4. Select the server(s) you want to deploy software to from the **Provisioned Resources** or **Computing Devices** page, and deploy the software package. See "Deploying Software on Servers" on page 215. Note that, after the software is installed (or fails to install successfully), the software package is automatically deleted from the target server(s).

   If you want to deploy the software and then take a snapshot of the server(s) you just deployed to, see "Using Deploy and Save" on page 217.

Caution By default, the software package deploys to a subdirectory on the target server under the directory where the ITAP agent is installed. Verify that there is sufficient disk space available before deploying software. For provisioned servers, the Linux ITAP agent is automatically installed in `/.dd`.

5. Display the status and check for errors:
   - If you are using an icon view from the **Provisioned Resources** page, click on the icon of a target server to popup a status window.
   - If you are using the table view, click the **Status** field of a target server to popup a status window.

Creating and Deploying Bootable or Control Software

Bootable and control software can only be run on server assets.

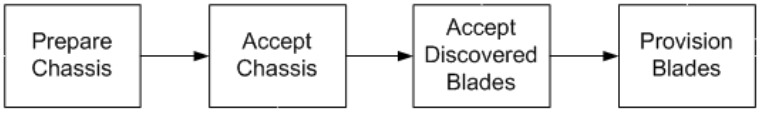
To create and deploy bootable software you can use the same workflow as for configuration software, but create the bootable software and supply a command line as shown in “Creating Bootable Software for Intel Xeon Processor Family Machines” on page 203.

Control software is a script or software that runs under an ActiveOS. The software must be compatible with the ActiveOS running on the target system. Your control software should be written as a Bourne shell (`sh`) script because this is the only type compatible with all versions of ActiveOS Software compiled under a Linux kernel should also work with the Linux ActiveOS.

Setting Up New Intel Server Chassis and Intel Server Compute Blades

This workflow tells you how to set up a new chassis and blades, from preparing the chassis to provisioning the blades. Note that this workflow incorporates other workflows, and that some of the steps are optional depending upon how you want to manage your chassis and blades.

Workflow for Setting Up an Intel Chassis and Blades



Assumptions

- You want to prepare chassis and blades for management by Intel Deployment Manager.
- You want to load a snapshot on one or more new blades.

▼ To set up the chassis and blades

1. Prepare the chassis. See "Preparing an Intel Server Chassis" on page 188.
2. Accept the chassis. See "Accepting a Discovered Intel Server Chassis" on page 192.
3. Accept discovered blades. See "Accepting and Power Cycling Intel Server Compute Blades" on page 196.
4. If you have a suitable, existing snapshot you can use it to provision one or more of the blades as discussed in the workflow "Provisioning Servers" on page 33.

Working with Workspaces 5

You can add workspaces to provide a means of grouping related resources for ease of use, and to restrict or grant access to sets of resources for specific user roles

This chapter includes the following topics:

Understanding Workspaces

A *Workspace* is a collection of all the resources that can be assigned to a particular group of users. You use workspaces to:

- provide a means of grouping related resources for ease of use
- restrict or grant access to particular sets of resources for specific user roles

Resources include computing devices, network devices, templates, snapshots, networks, software, jobs, licenses, and provisioned resources. Each workspace has certain resources allocated to it.

When you click the **Workspaces** tab, several icons are displayed under it in the Navigation Area. The icons are:

- **Display** (the icon on the left that represents a window)—Enables you to select the workspaces that you want to display, depending on whether or not you have permission to access to the selected workspaces.
- **View All** (the icon in the center that represents an eye)—Displays all the workspaces you can access.
- **Add** (the icon on the right that has a "+" sign on it)—This icon enables you to add a workspace. When you log into Intel Deployment Manager for the first time, you see two pre-defined workspaces: `Everything` and `Discovered`.

Workspaces, user roles, and access controls are inter-related—if you add workspaces, you probably want to add user roles and access controls as well. Workspaces are tied to user roles using access controls.

Everything Workspace

The `Everything` workspace contains all of the resources managed by Intel Deployment Manager. Every resource that is added to Intel Deployment Manager is included in the `Everything` workspace. All resources always belong to the `Everything` workspace: you cannot remove resources from this workspace. Assigning a resource to `Everything` has no effect because it already belongs to this workspace (If you assign a resource to `Everything`, the resource does *not* become available to all the workspaces).

Discovered Workspace

All of the assets that Intel Deployment Manager discovers are listed in the `Discovered` workspace. The newly discovered assets remain listed here until they get accepted or get allocated to another workspace.

Managing Workspaces

Perform the following operations from the administration console.

Adding Workspaces

▼ **To add a workspace**

1. In the Navigation Area, click **Workspaces**.
2. In the Navigation Area, click the **Add** icon.
3. The **Add Workspace** dialog is displayed. Enter the following information.

Add Workspace Fields

Field	Description
Name	Enter a unique name for the workspace. Maximum 32 characters. Most alphanumeric and special characters are allowed, but avoid greater than (>), less than (<), ampersand (&), single quotes (‘), and double quotes (“).
Description	Enter a description for the workspace. This description is used for your own reference.
Comments	You can enter any comments about the workspace. These comments are also used for your own reference.

4. Click **OK** to add the workspace.

Editing Workspaces

▼ **To edit the information for an existing workspace**

1. In the Navigation Area, click **Workspaces**, then click on the workspace you want to edit.
2. Click **Actions** > **Edit**.
3. Update the **Name**, **Description**, or **Comment** fields as desired.
4. Click **OK**.

Deleting Workspaces

Before deleting a workspace, you must first shut down all computing devices within the workspace. You can delete multiple workspaces at the same time using the **Delete Workspaces** dialog. When you delete a workspace, all of the resources allocated to that workspace are deleted.

▼ To delete a workspace

1. In the Navigation Area, click **Workspaces**.
2. The **Displaying all Workspaces** table appears. Select the check box(es) for the workspaces you want to delete.
3. Click **Actions** > **Delete**. The verification dialog is displayed.
4. Click **Yes** to delete the workspace(s).

Users, Roles, and Access Controls 6

A user role represents an arbitrary group of users who have a need to perform a similar set of tasks. You can add a user at any time, and associate that user with a user role. An access control maps a specific user role to a workspace.

This chapter includes the following topics:

Understanding Role-based Administration

A *user role* represents an arbitrary group of users who have a need to perform a similar set of tasks. Every user belongs to one or more roles. Each role contains fields to map itself to LDAP groups, containers, or organizational units. Roles also contain a small set of resource-independent permissions (capabilities).

A *workspace* represents an arbitrary grouping of resources. Any object that can be managed by Intel Deployment Manager, including a snapshot, is considered to be a resource. Resources can belong to one or more workspaces. Workspaces can contain resources of different types.

Roles are linked to workspaces with an *access control*. An access control maps a specific role to a workspace. When a role is mapped to a workspace by an access control, all users in that role are allowed to view all the resources in the workspace. Each access control also has a set of permissions which identify the operations the user is allowed to perform on the resources included in the workspace.

Intel Deployment Manager contains some predefined roles, workspaces, and access controls. There is a special, unmodifiable workspace called `Everything` to which every resource belongs. There are two unmodifiable roles: `Everyone` and `Sysadmin`. Every user belongs to the user role `Everyone`. The `Sysadmin` user role has all system permissions enabled. Additionally, the LDAP group that this `Sysadmin` role maps to is determined by a property in the elemental platform named `server.role.sysadmingroup`. The default value of this property is Intel Deployment Manager `Admins`. This `Sysadmin` role has an unmodifiable access control that maps it to the `Everything` workspace with all resource permissions.

To support access to auto-discovered assets, all such assets are automatically placed in a special workspace called `Discovered`. Therefore, if you want to assign a role to permit access to newly discovered assets, you would allow this role to access the workspace named `Discovered`. Also, the owner of the discovered assets is set to a special user named `System`.

A user is allowed access to a resource if and only if:

- The user is the owner of that resource. The user that created the resource is always its owner.
- There is an access control between user role *R* and workspace *W* where the user belongs to role *R* and the resource belongs to workspace *W*.

The owner of a resource and the workspaces it belongs to are displayed in the owner and workspaces columns for that resource. You might need to use the table options page for the resource's table to display these columns.

For a user to perform an action on a resource, that user must possess the proper permissions for the resource. With the exception of permissions to create resources, all permissions for a resource are determined from the permissions of all access controls that

apply for the user and the resource. If there are multiple applicable access controls, the user has permission to perform the action if any access control has the permissions to allow the user to perform that action. *Create permissions* are associated with the user's roles. If a user belongs to multiple roles, the create permissions for that user are the union of the create permissions for the user's roles. The permissions for roles and access controls are described in detail later in this chapter.

Understanding Permissions

Permissions identify the operations a user can perform on resources within Intel Deployment Manager. Permissions are classified into two categories: *system permissions* and *resource permissions*. System permissions are independent of resources and are associated directly with roles. Resource permissions are mapped to access controls and indicate the operations allowed for the resources in the workspaces of those access controls.

System Permissions

A user role can be assigned the following set of system permissions. If a user belongs to multiple roles, the user's permissions are a union of the all permissions assigned to the user's roles.

- `View Custom Logs`—Allows the user to view the custom log (events) table.
- `View Reports`—Allows the user to view reports.
- `View System Settings`—Allows the user to view any Intel Deployment Manager preferences, settings, and drivers.
- `Modify System Settings`—Allows the user to modify any Intel Deployment Manager preferences, settings, system storage, licenses, and drivers. The user is also allowed to delete events from the custom log.
- `Modify Roles`—Allows the user to modify users and roles.
- `Modify constraints`—Allows the user to modify the constraints for snapshots, computing devices, and software. The resource must also have the *`Full Control`* permission to perform this operation.
- `View Pre-Provision Policies`—Allows the user to view pre-provision policies.
- `Modify Pre-Provision Policies`—Allows the user to define or modify pre-provision policies.

Some system permissions are called *create permissions* since they control the creation of resources. There is a create permission for each kind of resource. If a user role has a create permission, users assigned to that role can create resources of that type. For example, if a role has the `Create workspaces` permission, users in that role can create workspace objects. See the table in step 7 on page 56 for a list of create permissions.

Resource Permissions

Resource permissions are further divided into two categories: *common resource permissions* and *operation permissions*. Common resource permissions are a small set of permissions common across all resources. Operation permissions apply only to particular commands, or a set of commands for a resource.

Common resource permissions can be applied to resources such as workspaces, snapshots, network device configurations, computing devices, licenses, software, OpScripts, jobs, tasks, networks, and system storage.

For each of these resource categories, you can specify one of the following common resource permission levels. These resource permission levels are listed starting with the most powerful and ending with the least powerful.

- `Full Control`—Allows you to view, delete, edit, assign, and perform operations on this resource. Assigning a resource refers to assigning this resource to a workspace. If a user owns a resource, it is assumed that the user has `Full Control` permission for that resource.
- `Use`—Allows you to view and use this resource. You are not allowed to edit the configuration. The resource permission also allows a user to assign this resource to other resources. For example, you can assign a server asset to a server only if you have the `Use` permission for both the server and the computing device. However, you cannot assign resources to workspaces.
- `Read Only`—Allows you to only view a resource. No operations can be performed on this resource with the exception of viewing its data.

Operation permissions control the actions you can perform on resources. If you are not assigned a particular operation permission, you cannot perform the operation associated with the permission, irrespective of the common resource permission value for that resource. See the list in step 7 on page 59 for a list of operation permissions.

Managing Users

A *User* represents a single user of Intel Deployment Manager. You can add users, display all users, and view information about each user. You can also change user passwords.

We recommend that general non-administrative users not use the `Administrator` login, except in emergencies. This is because the `Administrator` login has the maximum permissions, making the administrator the most powerful, and potentially the most dangerous user in Intel Deployment Manager. To maintain security, you should change the `Administrator` password whenever you perform a fresh installation or upgrade of Intel Deployment Manager.

Note The **Users** tab is visible from the administration console only if **Default Authentication** was selected during installation. This tab is not displayed if Intel Deployment Manager is using directory services for user authentication.

Adding/Editing Users

▼ To add or edit a user

1. In the Navigation Area, click **Settings** > **User Management.**
2. Click the **Users** tab. The **Displaying All Users** table is displayed.

   If you are editing a user, select the user you want to edit. Do not select any users if you are adding a new user.
3. Click **Actions** > **Add** or **Edit**. The **Add User** or **Edit User** dialog displays the following sections:
   - **Basic**
   - **Roles** (Displayed only when editing a user).

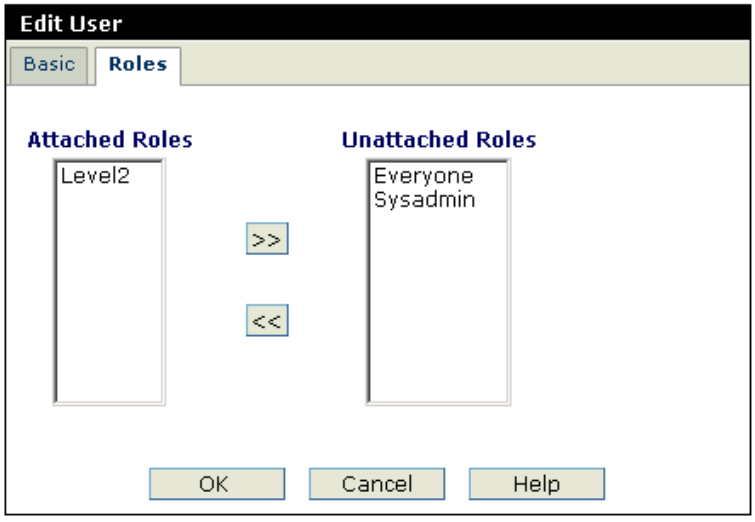
4. Add or edit the following information in the **Basic** section:

Add/Edit User: Basic Section

Field Name	Description
User ID *	Enter a unique user ID. The user ID identifies the user. Maximum 32 characters. Use only letters, numbers, and underscores. Note: You cannot edit the user ID.
Password *	Enter a password for the user. The password can contain from 6 to 32 characters, alphabetic, numeric or underscores.
R**etype Password** *	Re-enter the password.
User role *	Select a user role to associate with this user.
Full Name	Enter the user's full name.
E-mail	Enter the user's e-mail address.
Organization	Enter the user's organization.
Title	Enter the user's title.
Phone (Work)	Enter the user's work phone number.
Phone (Home)	Enter the user's home phone number.
Phone (Fax)	Enter the user's fax number.
Phone (Pager)	Enter the user's pager number.
Phone (Cell)	Enter the user's cell phone number.

5. In the **Roles** section all of the roles that you added are displayed. The **Roles** section of the **Edit User** dialog enables you to attach or unattach roles to a specific user using the >> or << buttons. The user ID you logged in as must have the permission to attach or unattach roles. Click **OK** to save changes. You can click **Cancel** if you do not want to save your changes.

Edit User > Roles



Deleting Users

▼ To delete a user

1. In the Navigation Area, click **Settings** > **User Management**.
2. Click the **Users** tab. The **Displaying All Users** table is displayed.
3. Select the user(s) you want to delete.
4. Click **Actions** > **Delete**. The user ID(s) you selected for deletion are displayed for verification.
5. Click **Yes** to delete the selected user ID(s).

Changing Passwords

A password is required to log in to Intel Deployment Manager. Depending on permissions, a user might be able to change another user's password.

▼ To change your password or a user's password

1. In the Navigation Area, click **Settings** > **User Management**.
2. Click the **Users** tab. The **Displaying All Users** table is displayed.
3. Select a user by clicking the check box associated with that user.
4. Click **Actions** > **Change Password**. The user ID is displayed in the **Enter a New Password** dialog. To change the password:
   - **New Password**—Enter the new password. The password must be between 6 and 32 characters in length. It can contain alphabetic, numberic or underscore characters.
   - **Retype New Password**—Retype the new password. The password must exactly match the password you entered in the **New Password** field.
5. Click **OK** to change the password. You can click **Cancel** to dismiss the dialog without changing the password, or click **Reset** to clear the password fields.

Managing User Roles

A *user role* represents an arbitrary group of users who have a need to perform a similar set of tasks. Every user belongs to one or more roles. Each role contains fields to map itself to LDAP groups, containers, or organizational units. Roles also contain a small set of resource-independent permissions (capabilities).

When using any directory services, you must have the following information available:

- IP addresses of directory servers
- Organizational units on the target directory trees

If you are using LDAP server for user authentication:

- Add a group of users to your LDAP server for Intel Deployment Manager.
- Assign users to this group. Any LDAP user that belongs to this group can log in as an Intel Deployment Manager administrator using the correct user name and password.

  When creating non-administrative users for Intel Deployment Manager, create a group in the LDAP server and map it to a role in the Intel Deployment Manager server.
- Contact your LDAP administrator to create users on the LDAP server for Intel Deployment Manager and to assign them to the groups.

Adding/Editing User Roles

Note All user roles, except `Sysadmin`, can be edited.

You can add or edit user roles depending on your user permissions.

▼ To add or edit a user role

1. In the Navigation Area, click **Settings** > **User Management**.
2. Click the **User Roles** tab. The **Displaying All User Roles** table is displayed.
3. If you want to edit a user role, select the role you want to edit.
4. Click **Actions** > **Add** or **Edit**. The **Add User Role** or **Edit User Role** dialog displays the following sections:
   - **Basic**
   - **System Permissions**
   - **Create Permissions**
   - **Users** (This tab is displayed only if LDAP authentication is *not* used for login).
5. In the **Basic** section, you can enter the following information:

Add/Edit User Roles: Basic Section

Field Name	Description
Name *	Enter a unique name for this user role. Most alphanumeric and special characters are allowed, but avoid greater than (>), less than (<), ampersand (&), single quotes (‘), and double quotes (“).
Description	Enter a description for this user role.
LDAP Group	If you are using an LDAP directory server to authenticate the user logins, enter the name of the LDAP Group for this user role. Otherwise, leave it blank. The LDAP group is assigned by, and obtained from your LDAP administrator.
Comments	Enter any comments related to this user role. For example, you might enter the permissions that are granted to this user role.

6. The **System Permissions** section enables you to assign the following set of permissions. If a user belongs to multiple roles, the permissions of that user is a union of the permissions of all of the user's roles.

Add/Edit User Roles: System Permissions Section

Field Name	Description
Select All	If checked all of the following permissions are selected.
View Custom Logs	If checked, the user can view the custom logs table.
View Reports	If checked, the user can view reports.
View System Settings	If checked, the user can view any system preferences, settings, and drivers.
Modify System Settings	If checked, the user can modify any Intel Deployment Manager system preferences, settings, drivers, Intel Deployment Manager licenses, and storage. The user is also allowed to delete events from the custom log.
Modify Roles	If checked, the user is allowed to add, edit, and delete users and roles within Intel Deployment Manager.
Modify Constraints	If checked, the user is allowed to add, edit, and delete constraints applicable to snapshots and computing devices.
View Pre-Provision Policies	If checked, the user can view pre-provision policies.
Modify Pre-Provision Policies	If checked, the user can define or modify pre-provision policies.

Note System permissions are resource-independent and are associated directly with user roles. Resource permissions are mapped to access controls and indicate the operations allowed for the resources in the workspaces of those access controls.

7. The **Create Permissions** section enables you to assign the following set of permissions that allow a user to add specific resources to Intel Deployment Manager:

Add/Edit User Roles: Create Permissions Section

Field Name	Description
Select All	If checked all the following permissions are selected.
Create workspaces	If checked, the user can add a new workspace.
Create Snapshots	If checked, the user can add snapshots.
Create Servers	If checked, the user can provision servers.
Create Computing Devices	If checked, the user can add and accept computing devices.
Create Licenses	If checked, the user can add licenses.
Create Software	If checked, the user can add software.
Create Jobs	If checked, the user can add jobs.
Create Networks	If checked, the user can add networks.

8. In the **Users** section, you can attach or unattach users to the user role. For example, you can select a user from the **Unattached Users** column, and click the << button to attach the selected user to the user role.

9. Click **OK**.

Deleting User Roles

Only a user with `Administrator` permissions can delete user roles. The roles, `Everyone` and `Sysadmin` cannot be deleted.

If you delete a user role from Intel Deployment Manager and you are using LDAP, it does *not* delete the LDAP group that the user role was associated with. For example, a user role `test_user_role` is associated with an LDAP group named Intel Deployment Manager `users`. An Intel Deployment Manager administrator deletes `test_user_role`. The LDAP group Intel Deployment Manager `users` is not affected when `test_user_role is deleted`.

▼ To delete a user role

1. In the Navigation Area, click **Settings** > **User Management**.
2. Click the **User Roles** tab. The **Displaying All User Roles** table is displayed.
3. Select the user role(s) you want to delete.
4. Click **Actions** > **Delete**.
5. Click **Yes** to delete the role(s).

Managing Access Controls

An *access control* maps a specific user role to a workspace. When a user role is mapped to a workspace by an access control, all users in that role are allowed to view all of the resources in the workspace. Each access control has a set of permissions which identify the operations the user is allowed to perform on the resources included in the workspace.

Adding/Editing Access Controls

▼ To add or edit an access control

1. In the Navigation Area, click **Settings** > **User Management**.
2. Click the **Access Controls** tab.
3. The **Displaying All Access Controls** table is displayed.

   If you are editing an access control, select the control you want to edit. Do not select an access control if you are adding a new one.
4. Click **Actions** >**Add** or **Edit**. The **Add Access Control** or **Edit Access Control** dialog displays the following sections:
   - **Basic**
   - **Resource Permissions**
   - **Operation Permissions**
5. Enter the following information in the **Basic** section:

Add/Edit Access Control: Basic Section

Field Name	Description
User Role *	Select a user role to associate with this access control.
Workspace *	Select a workspace to associate with this access control.
Description	Enter a description for this access control.
Comments	You can enter any comments. For example, you might describe the permissions that are granted to this access control.

6. In the **Resource Permissions** section, you can assign the permissions that a specific access control can have when accessing the **Resources** section in the Navigation Area. The levels of permissions are:

Add/Edit Access Control: Resource Permissions Section

Field	Description
`Full Control`	Allows you to view, delete, edit, assign, and perform operations on this resource. Assigning a resource refers to assigning this resource to a workspace. If a user owns a resource, the system assumes that the user has `Full Control` permission for that resource.
`Use`	Allows you to view and use this resource. You cannot edit the resource's configuration. This permission also allows you to assign this resource to other resources. For example, you can assign a server asset to a server only if you have the `Use` permission for both the server and the asset. However, you cannot assign resources to workspaces.
`View Only`	Only allows you to view a resource. No operations can be performed on the resource.

7. In the **Operation Permissions** section, you select the permissions a specific access control can have when performing a particular operation. The operation permissions are:

Add/Edit Access Control: Operation Permissions Section

Field	Description
`Power Cycle Servers`	Allows you to power on, off, or cycle any computing device that you can access. The computing device must also have the `Use` or `Full Control` permission to perform one of the power operations.
`Open Terminal`	Allows you to have remote access to any server. The server must also have `Use` or `Full Control` permission to perform this operation.
`Open Console`	Allows you to open a console on any server. The server must also have `Use` or `Full Control` permission to perform this operation.
`Save (Computing Devices) Snapshot`	Allows you to save a snapshot for a computing device. The computing device must also have the `Use` or `Full Control` permission to perform the save.
`Save (Provisioned Resources) Snapshot`	Allows you to save a full custom snapshot for a server. The server must also have the `Use` or `Full Control` permission to perform the save.

8. Click **OK**.

Deleting Access Controls

▼ **To delete an access control**

1. In the Navigation Area, click **Settings** > **User Management**.
2. Click the **Access Controls** tab.
3. Select the access control(s) you want to delete.
4. Click **Actions** > **Delete**. The verification dialog is displayed.
5. Click **Yes** if you want to delete the access control.

Organizing Networks 7

You can use Intel Deployment Manager to manage networks and IP lists.

This chapter includes the following topics:

See "Network Topologies" on page 273 for information about network configurations that can be used with Intel Deployment Manager.

Managing Networks

A network entity in Intel Deployment Manager represents a subnet in a TCP/IP network. The network entity contains fields specifying the network address, the subnet mask, excluded IP addresses, default gateway, and associated DNS servers.

You must specify a network address for every managed computing device so that Intel Deployment Manager can communicate with the device.

Adding/Editing Networks

Adding a network enables you to define a network topology that can be used to provision computing and network devices. IP addresses in the network can be assigned to discovered assets unless they are reserved by defining an IP exclusion range.

▼ To add or edit a network

1. In the Navigation Area, click **Resources** > **Networks**. The **Displaying All Networks** table is displayed.
2. Click the **Networks** tab (if it is not already open).
3. If you want to edit a network, select the network.
4. Click **Actions** > **Add** or **Edit**. The **Add Network** or **Edit Network** dialog is displayed.

5. Enter the following information:

Add/Edit Network Fields

Field Name	Description
Name *	Enter a unique name for the new network that you are creating. Maximum 32 characters. Do not use any special characters.
Network Address *	Enter the base IP address of the network.
Subnet Mask *	Enter the subnet mask. The subnet mask determines the subnet to which an IP address belongs.
Default Gateway *	Enter the IP address of the default gateway that this network should use.
Domain Name Servers *	Enter the IP address for your DNS server(s). Separate multiple IP addresses with spaces or commas. For example, `10.1.48.30,10.1.62.12`. By default, the Domain Name Servers assigned to this network are used.
Accept Discovered Asset	Check this check box if this network should be used when automatically determining a network when accepting a discovered server asset.
New IP Exclusion Range	Optionally, add a new IP exclusion range, and click **Add Range**. For example, `192.168.7.185 : 192.168.7.189`. Machines must have unique IP addresses in order to avoid address conflicts. Adding an IP exclusion range blocks IP address in the specified range from being assigned to other machines. In effect, you are telling Intel Deployment Manager that the addresses in the IP exclusion range are reserved or already used, and are not available for assignment. You should always exclude the IP addresses of the Intel Deployment Manager server, and DNS and gateway servers.
IP Exclusion Ranges	Enables you to block a range of IP addresses. When blocked, IP addresses in the specified range cannot be used when assigning IP addresses to newly discovered assets. To delete a range, select it and click **Delete Range**. For example, if 10.1.1.1 is a gateway, 10.1.1.2 is a DNS server, and 10.1.1.3 is your Intel Deployment Manager server, you can exclude these addresses from being assigned to any other asset by putting them in the IP exclusion range 10.1.1.1 : 10.1.1.3. **Note**: An IP address range excluded from a network is automatically excluded from its IP list. But the IP addresses excluded from an IP list are not excluded from its network.
Assigned to Workspace	Select the workspace to which this network should belong.

6. Click **OK** to add the network.

Assigning a Network to a Workspace

▼ **To assign a network to a workspace**

1. In the Navigation Area, click **Resources** > **Networks**.
2. Select the network you want to assign.
3. Click **Actions** > **Assign**.
4. Select the workspace to which you want to assign the network.
5. Click **OK**.

Unassigning Networks

▼ **To unassign a network from a workspace**

1. In the Navigation Area, click **Resources** > **Networks**.
2. Select the network you want to unassign.
3. Click **Actions** > **Uassign**.
4. Select the workspace from which you want to unassign the network.
5. Click **OK**.

Deleting Networks

You cannot delete a network if:

- servers are using the network
- there are server assets in the FREE state

If you try to delete a network that has one or more servers, the following error message is displayed:

```
Network is used.
```

▼ To delete a network

1. In the Navigation Area, click **Resources** > **Networks**.
2. Select the network you want to delete.
3. Click **Actions** > **Delete**.
4. Click **YES** to delete the network.

Managing IP Lists

An IP list contains a range of network addresses in an Intel Deployment Manager-managed network. Using IP Lists, you can divide a network into multiple sections that can be assigned to different workspaces. Then, users can only manage systems in the section of the network (in the workspace) for which they have access permissions.

The **IP Lists** table displays the IP address ranges that are free or in use by IP lists for each network. You can use this table to determine which IP address ranges are available for a network when creating a new IP list. You can also track which IP addresses are currently in use.

Displaying IP Lists

The **Displaying All IP Lists** table displays the IP address ranges that are free or in use by IP lists for each network. You can use this table to determine which IP address ranges are available for a network when creating a new IP list. You can also use this table to track which IP addresses are currently in use.

▼ To display all IP lists

1. In the Navigation Area, click **Resources** > **Networks**.
2. Click the **IP Lists** tab.

▼ To display a particular IP list

Click **Resources** > **Provisioned Resources** > *NetworkName* > *IPListName* (where *NetworkName* is the name of the network, and *IPListName* is the name assigned to the IP list).

or

If you assign an IP list to a workspace, it can be displayed by clicking **Workspaces** > **Provisioned Resources** > *NetworkName* > *IPListname.*

Adding/Editing IP Lists

When you add an IP list, you are allocating IP addresses from a network.

▼ To add or edit an IP list

1. In the Navigation Area, click **Resources** > **Networks**.
2. Click the **IP Lists** tab.
3. If you want to edit an IP list, select the IP list that needs to be edited.
4. Click **Actions** > **Add** or **Edit**.
5. The **Add IP List** or **Edit IP List** dialog appears. Enter the following information:

Add/Edit IP List Fields

Field Name	Description
Name *	Enter a unique name for this IP list. Most alphanumeric and special characters are allowed, but avoid greater than (>), less than (<), ampersand (&), single quotes (‘), and double quotes (“).
Network	Select the network from which you want to allocate a section of IP addresses.
Number of IP Addresses *	Enter the number of IP addresses. You must provide a value in this field, *or* in the **Start and End IP Addresses field**.
Start and End IP Addresses*	You can enter the first and last IP addresses of the range. These fields should be within the range of the network listed in the field above. You must provide a value in this field, *or* in the **Number of IP Addresses** field.
Default Gateway	Enter the IP address of the default gateway for this IP list.
Domain Name Servers	Enter the IP address for your DNS server(s). Separate multiple IP addresses with spaces or commas. For example, `10.1.48.30,10.1.62.12`.

Add/Edit IP List Fields

Field Name	Description
New IP Exclusion Range	Optionally, add a new IP exclusion range, and click **Add Range**. For example, `192.168.7.185 : 192.168.7.189`. Machines must have unique IP addresses in order to avoid address conflicts. Adding an IP exclusion range blocks IP address in the specified range from being assigned to other machines. In effect, you are telling Intel Deployment Manager that the addresses in the IP exclusion range are reserved or already used, and are not available for assignment. You should always exclude the IP addresses of the Intel Deployment Manager server, and DNS and gateway servers.
IP Exclusion Ranges	Enables you to block a range of IP addresses. When blocked, IP addresses in the specified range cannot be used when assigning IP addresses to newly discovered assets. To delete a range, select it and click **Delete Range**. For example, if 10.1.1.1 is a gateway, 10.1.1.2 is a DNS server, and 10.1.1.3 is your Intel Deployment Manager server, you can exclude these addresses from being assigned to any other asset by putting them in the IP exclusion range 10.1.1.1 : 10.1.1.3. **Note**: An IP address range excluded from a network is automatically excluded from its IP list. But the IP addresses excluded from an IP list are not excluded from its network.
Assigned to Workspace	Select the workspace to which this IP list should belong.

6. Click **OK** to add this IP list to provisioned resources.

The IP list name is displayed by clicking **Resources** > **Provisioned Resources** > *NetworkName* > *IPListName* (where *NetworkName* is the name of your network, and *IPListName* is the name you assigned to the IP list). It can be used to provision servers. If you assign this IP list to a workspace, it is displayed by clicking **Workspaces** > **Provisioned Resources** > *NetworkName* > *IPListname* too.

Assigning IP Lists

▼ **To assign an IP list to a workspace**

1. In the Navigation Area, click **Resources** > **Networks**.
2. Click the **IP Lists** tab.
3. Select the IP list(s) you want to assign.
4. Click **Actions** > **Assign**.
5. Select the workspace to which you want to assign the IP list.
6. Click **OK**.

Unassigning IP Lists

▼ **To unassign an IP list from a workspace**

1. In the Navigation Area, click **Resources** > **Networks**.
2. Click the **IP Lists** tab.
3. Select the IP list(s) you want to unassign.
4. Click **Actions** > **Unassign**.
5. Select the workspace from which you want to unassign the IP list.
6. Click **OK.**

Deleting IP Lists

You can delete an IP list that contains servers. All of the servers in the IP list are moved to the IP list's network.

▼ **To delete an IP list.**

1. In the Navigation Area, click **Resources** > **Networks**.
2. Click the **IP Lists** tab.
3. Select the IP list(s) you want to delete.
4. Click **Actions** > **Delete**.
5. Click **OK**.

Managing System Storage 8

System storage is physical disk space allocated for use by Intel Deployment Manager. You add system storage to control where snapshots, software, and OpScripts are stored. You need to have sufficient system storage before you can perform basic tasks like taking snapshots or provisioning.

This chapter includes the following topics:

Adding System Storage

You register system storage with Intel Deployment Manager by adding it. When you add system storage you describe the storage characteristics, and how Intel Deployment Manager can access the storage. You can add additional system storage at any time.

System storage can be on the Intel Deployment Manager server, but is usually on one or more other servers accessible over the network. Make sure that all of your machines can communicate with the storage server(s).

Intel Deployment Manager supports access to storage devices via NFS, CIFS, or Samba, but not all protocols are supported on each platform.

- If your Intel Deployment Manager server is running on a Windows system your storage server can run Linux with Samba, or Windows with CIFS. Running a Linux storage server with NFS is not supported.
- If your Intel Deployment Manager server is running on a Linux system, your storage server can run Linux with Samba or NFS, or Windows with CIFS.

After adding system storage, Intel Deployment Manager gathers information about the storage and begins to track it.

▼ To add system storage

1. In the Navigation Area, click **Settings** > **System Storage**.
2. Click **Actions** > **Add**.
3. In the **Add System Storage** dialog, select a storage driver.

4. Enter the following information:

Add System Storage Fields

Field Name	Description
Storage Host IP Address *	Enter the IP address of the server where the system storage is located. For example, 192.168.7.50.
Storage Directory *	Enter the path of the storage directory. For example, if a shared storage directory was created on a Windows C: drive named `test_storage1`, enter **`test_storage1`**.
Mount Protocol *	Select the protocol used to mount the storage device. For Windows, select **CIFS/SMB**. For Linux systems, use **NFS**.
Storage User Domain (For CIFS / SMB shared directories only)	Enter the Windows user domain. For example, **`DOMAIN`**.
Storage User ID (For CIFS / SMB shared directories only)	Enter the Windows user ID. For example, **`Administrator`**.
Storage User Password (For CIFS / SMB shared directories only)	Enter the password for the user ID. Passwords are case-sensitive. The storage user domain, user ID, and user password are displayed if the CIFS protocol is specified.
Find and import Snapshots, Software, OpScripts	Check this check box if you want to automatically find and import existing snapshots, software, and OpScripts.

5. Click **OK**. When you click **OK**, the system storage is automatically allocated and made available for use.

Editing System Storage

If you need to modify any information for existing system storage, you can edit it.

▼ To edit system storage

1. In the Navigation Area, click **Settings** > **System Storage**.
2. Select the storage location that you want to edit.
3. Click **Actions** > **Edit**.
4. The **Edit System Storage** dialog is displayed. Enter the following:

Edit System Storage Fields

Field Name	Description
Storage Host IP Address *	Enter the IP address of the server where the storage directory is located. For example, 192.168.7.50.
Storage Host Directory *	Enter the path of the storage directory. For example, if a shared storage directory was created on your C: drive named test_storage1, enter **`test_storage1`**.
Storage Description *	DIsplay only field. By default, the system populates this field with the storage device IP address and directory pathname.
Mount Protocol	Display only field. Shows the protocol that was used to mount the selected storage.
Storage Domain * (For CIFS / SMB shared directories only)	Enter the Windows user domain. For example, **`DOMAIN`**.
Storage User ID (For CIFS / SMB shared directories only)	Enter the Windows user ID. For example, **`Administrator`**.
Change Storage Password (For CIFS / SMB shared directories only)	Check this check box if you want to change the storage password. Then, enter the new password in the **New Storage Password** field.
New Storage Password (For CIFS / SMB shared directories only)	Enter a new password for the user ID. Passwords are case-sensitive.

5. Click **OK**.

Deleting System Storage

You cannot delete system storage if there is any snapshot, software, or OpScript using it.

▼ **To delete system storage**

1. In the Navigation Area, click **Settings** > **System Storage**.
2. In the **Displaying All Storage Devices** table, select the storage device you want to delete.
3. Click **Actions** > **Delete**.
4. Click **Yes** to delete the storage.

Preparing and Discovering Computing Devices 9

Computing devices include Intel Server Chassis, Intel Server Compute Blades, and non-blade server assets. Intel Deployment Manager can automatically discover the operating system and hardware attributes of computing devices.

This chapter includes the following topics:

Preparing Computing Devices for Discovery

Use the following instructions to prepare Intel Xeon processor family Intel Server Compute Blades for discovery by Intel Deployment Manager.

Preparing an Intel Xeon Processor Family Computing Device

▼ **To prepare an Intel Xeon processor family computing device (including Intel Xeon processor family with 64-bit extension) for discovery**

1. Verify that your Intel Xeon processor family Intel Server Compute Blades has at least one bootable NIC (supports PXE 2.0 build 72 or later with Intel boot agent).

2. Set the BIOS boot order on the server so that it boots from the network *before* booting from the hard disk. You can set the first boot devices to be floppy or CD drives, but be sure that there is no floppy diskette or bootable CD left in these drives. For example, you can set the boot order as floppy drive first, CD drive second, network third, and hard drive fourth.

3. For Intel Xeon processor family models that have a BIOS setting for OS (HP / Compaq), set the BIOS setting for OS as `Other`. This enables you to use either Windows or Linux snapshots.

4. Connect the bootable NIC on your server to a switch or hub that is on the same network as the Intel Deployment Manager server.

Discovering Computing Devices

Intel Deployment Manager can automatically discover the operating system and hardware attributes of computing devices.

Automatically Detecting Hardware Attributes

Automatic hardware attribute detection is the process which automatically detects various hardware attributes of a system, such as system-type, CPU-speed, and memory size.

Automatic hardware attribute detection occurs at several points. It is invoked just after accepting an auto-discovered computing device, when you update a computing device, or after an update package is executed. For automatic hardware attribute detection to succeed, the system must be running ActiveOS. If the system is not currently running ActiveOS, it waits for a fixed period of time for the system to boot into ActiveOS. If the ActiveOS does not come up after this period of time, the hardware attribute detection fails.

Some of these attributes are detected by examining the system's SMBIOS (if present). Others are detected by examining the NICs and hard-disks attached to the system. When the system determines its hardware attributes, they are stored in the computing device object.

Many hardware attributes has a corresponding constraint. The following table presents hardware attributes and their corresponding constraints. Attributes that do not have corresponding constraints are marked with N/A (Not Applicable).

Hardware Constraints and Descriptions

Hardware Attribute	Description	Constraints
Memory	Motherboard memory size	MBmem
Architecture	Hardware Architecture	MBArch
Chip Set	Motherboard chip set	MBcset
CPU Count	CPU count	MBcpuCount
CPU Speed	CPU speed in megahertz	MBavgCpuMHz
Manufacturer	Name of manufacturer	SYmanufacturer
Model	Model Name	SYmodel
Serial Number	Serial Number	SYSerialNo
NIC Count	Number of NICs	CNcount
NIC Type	Type of NIC	N/A

Hardware Constraints and Descriptions

Hardware Attribute	Description	Constraints
NIC Speed	Speed / bandwidth of NIC	N / A
NIC Duplex	Duplex of NIC / connection	N / A
NIC MAC Address	Hardware / MAC address of NIC	N / A
Hard Disk Count	Number of hard disks	HDcount
Hard Disk Device Name	Hardware device name (IDE or SCSI)	HDtype
Hard Disk Size	Hard disk size	HDsize
BIOS Manufacturer	Name of BIOS Manufacturer	N / A
BIOS Version	BIOS version	MBbiosVer (from “BIOS” firmware in the Server Asset)
BIOS Release Date	BIOS release date	N / A
Chassis Slot	System slot	SYslot
Chassis Serial Number	System chassis number	EXchassisId
Chassis Model	System chassis model	N / A
Chassis IP Address	System chassis IP address	N / A
Chassis Manufacturer	System chassis manufacturer	N / A
Rack Number	Rack serial number	N / A

Automatically Detecting an Operating System

The *Automatic OS detection* process detects the OS for a computing device. The OS is detected by examining the partitions of the hard disks attached to the system.

Automatic OS selection is performed by examining various files contained on the disk partitions. The data collected from this operation is sent to the Intel Deployment Manager back-end, which then determines which OS best matches this data.

Automatic OS detection is performed whenever you choose **Automatically Detect OS** in the **Save Snapshot** dialog. The OS of the saved snapshot is set to the OS that is on the computing device's disks.

Configuring Computing Devices 10

Computing devices include Intel Server Chassis, Intel Server Compute Blades, and non-blade server assets. In this chapter you are given procedures for manually adding, editing, and managing computing devices.

This chapter includes the following topics:

Using the Computing Devices Page

The **Computing Devices** page provides a view of all managed assets, to the extent permitted by a user's assigned permissions. The assets are divided into tabs, including:

- **Accepted**—These are all computing devices that have been discovered and accepted.
- **Blade Assets**—These are all entities that represent a physical blade server that can be provisioned.
- **Chassis Assets**—This tab lists all entities that represent a physical chassis that contains blade assets.
- **Server Assets**—These are entities that represent a physical server that can be provisioned.
- **Discovered**—This tab lists all the computing devices that have been auto-discovered, but have not yet been accepted. After discovering a computing device, you must accept it before you can perform other operations on it like provisioning, or loading or saving a snapshot.
- **Rejected**—This tab lists all the computing devices that have been auto-discovered, but which you have rejected for management by Intel Deployment Manager.
- **Summary**—This tab provides a list of used and free assets by architecture and status.

From the **Accepted** tab you can work on all blade assets, chassis assets, and server assets that have been accepted. You can also work on blade assets, chassis assets, and server assets from their respective tabs. The **Accepted**, **Blade Assets**, and **Server Assets** tabs display the following menus (the **Chassis Assets** tab presents a subset of these menus):

- **Actions**—enables you to edit, add mananaged or unmanaged servers, provision servers, initiate discovery, assign, unassign, or delete computing devices.
- **Snapshots**—enables you to save a snapshot of the selected computing device.
- **Software**—enables you to deploy selected software on selected computing devices.
- **Maintenance**—enables you to reboot, reset, reconnect, rediscover asset attributes, enter, or exit maintenance mode, and reboot to ADMIN State.
- **Power**—enables you to power on, off, or cycle selected computing devices.
- **Console**—enables you to access the system console of a selected computing device remotely. This is only supported on some devices.
- **Access**—enables you to access the selected computing device remotely.

Editing Managed Server Assets

▼ **To manually edit a managed server asset**

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Click the **Server Assets** tab.
3. Select the server asset you want to edit, and click **Actions** > **Edit**.

Tip Alternatively, click a server asset **ID** to display the **Edit Server Asset** dialog for a blade.

4. The **Edit Server Asset** dialog displays the following sections:
   - **Basic**
   - **Network**
   - **System**
   - **Console**
   - **NIC**
   - **Hard Disk**
   - **RPC**
   - **Firmware**
   - **Constraints**
5. The **Basic** section includes information that is collected for your records. Entries in this section are optional. You can go directly to the **Network** section.

Note Model name, serial number, and description are automatically detected from the device when it is added or when an update is performed, if the device supports these attributes. This data is used by Intel Deployment Manager when logging information about the device.

In the **Basic** section, you can enter the following information:

Edit Managed Server Asset: Basic Section

Field Name	Description
ID (Edit mode only)	Displays the unique ID that was assigned by Intel Deployment Manager to identify this asset when it was added.
Description	You can provide a description for the server asset. For example, **`server2`**. Maximum 64 characters.
Manufacturer	Displays the manufacturer discovered by Intel Deployment Manager. You can override this information by entering new data (not recommended). Maximum 64 characters.
Model Name	Displays the model name discovered by Intel Deployment Manager. You can override this information by entering new data (not recommended). Maximum 64 characters.
Serial Number	Displays the serial number discovered by Intel Deployment Manager. You can override this information by entering new data (not recommended). Maximum 64 characters.
GUID	Displays the GUID discovered by Intel Deployment Manager. A Globally Unique Identifier, or GUID, is a pseudo-random number used in software applications. Each generated GUID is supposed to be unique.
Miscellaneous Info.	You can enter any information you want in this field and in the **Additional Info** fields. Maximum 256 characters per field.
Additional Info #1	You can enter additional information here.
Additional Info #2	You can enter additional information here.
Additional Info #3	You can enter additional information here.
Additional Info #4	You can enter additional information here.
Additional Info #5	You can enter additional information here.
Comment	You can enter any comment about this asset.
Location	Enter the rack or cage number of this server asset.
Licenses	Select the applicable license(s) from those you registered earlier. When software for which you have added a license is loaded on a server and the server status is UP, one license is counted as being in use

Edit Managed Server Asset: Basic Section

Field Name	Description
Assigned to Workspace	Select the workspace to which you want to assign this server. All of the workspaces your user login has access to are displayed.

6. The **Network** section includes information about your network. Some entries in this section are required. Enter the following information:

Edit Managed Server Asset: Network Section

Field Name	Description
Maintenance Mode	Checking this check box turns enables maintenance mode. The asset must be in an IDLE state to go into maintenance mode. No Intel Deployment Manager operations can be performed on a server that is in maintenance mode—you cannot provision a server, save a snapshot, or perform other tasks that require network access while in maintenance mode.
Architecture *	Select the appropriate architecture for this asset.
Management Protocol *	Select a management protocol. The management protocol is an ActiveOS or a management device driver, deployed to run on a managed server, or to remotely manage a device. This entry depends on the architecture selected for the asset.
Boot Domain *	Select the boot domain. A boot domain is a collection of servers, within a distributed Intel Deployment Manager installation, running a boot daemon that can only boot a subset of machines within that network.
IP Address	Enter the IP address for this managed server asset. If you leave this field blank, Intel Deployment Manager looks for an IP address in the network that can be assigned to this server asset.
Network *	Select the network to which this asset belongs.
Subnet Mask	Enter the subnet mask for this managed server asset.
Gateway	You can enter the default gateway that this asset should use.
Next Server IP	Enter the IP address for the next server. If you want to use another TFTP server besides the one installed by default (on the same server as Intel Deployment Manager), then you must specify its IP address.

7. In the **System** section, you can enter the following information:.

Edit Managed Server Asset: System Section

Field Name	Description
Allocatable	Check this check box if you are preparing to save a snapshot.
Number of CPUs	Enter the number of CPUs supported by this asset. The default is 1.
Procesor Family	Select the processor type.
Supports 64-bit addressing	Check this check box if the processor supports 64-bit addressing. For example, if the asset uses a processor from the Intel Xeon processor family with 64-bit extension, you should check this check box.
Average CPU Clock Rate	Enter the average CPU clock rate in MHz.
Motherboard Chip set	Enter the chipset number of the motherboard.
Amount of Memory (RAM)	Enter the amount of memory in MB.

8. Enter the following information in the **Console** section:

Edit Managed Server Asset: Console Section

Field Name	Description
Protocol	Select the protocol used to access the system console of this computing device. For server and blade assets the default console protocol is OpShell.
IP Address	Enter the IP address that you want to use to access the system console of this computing device. By default, the IP address entered in the **Network** section is used.
Port	Enter the port that should be used to contact the console.
Initial Window Size	Select the size for the console browser window. For example, **800x600**.

9. Enter the following information in the **NIC** section (some fields in this section are required:

Edit Managed Server Asset: NIC Section

Field Name	Description
ActiveOS Network Device (displayed in edit mode only)	Display only field. Displays the ActiveOS device name of the NIC asset as discovered by Intel Deployment Manager.
MAC Address *	Enter the MAC Address of the server asset in the form: **AF**:04:0F:1E:17:69
NIC is Manageable	Check this check box if you want this NIC to be managed by Intel Deployment Manager.
NIC is Bootable	If the server asset is set to boot from this NIC, then check this check box. Otherwise select the bootable check box for the server asset. Also, you can have one NIC checked as bootable.
Description	Enter a description for the NIC. Maximum 64 characters.
Comments	Enter any extra comments for the NIC.

If you want to add an additional NIC, click **Add NIC**.

10. Enter the following information in the **HD** (Hard Disks) section (some fields in this section are required):

Edit Managed Server Asset: Hard Disks Section

Field Name	Description
Interface *	Select either **IDE** or **SCSI**.
Device Path *	Enter the location of the hard disk. For example, **`/dev/hda/`**
Saveable	Check this check box if the disk image should be saved when a snapshot is saved. An asset can have multiple hard disks, and you can save or not save one or more of them.
Size *	Enter the size of the disk in gigabytes.
Hardware Raid Level	If you are using a RAID array, select the RAID Level. This information is *not* automatically discovered.
Misc. RAID Controller Info	Enter any information that you would like to record regarding the Hardware RAID controller.

If you want to add another hard disk, click **Add Hard Disk**.

11. Enter the following information in the **Firmware** section:

Edit Managed Server Asset: Firmware Section

Field Name	Description
Name *	Enter a name for the firmware.
Manufacturer	Enter the name of the manufacturer for the firmware.
Version	Enter the version of the firmware.
Release Date	Enter a release date in *mm/dd/yy* format.
Misc. BIOS Info.	Enter any information that you would like to record regarding the firmware.

12. In the **Constraints** section you can edit the constraints applicable to this blade asset. See “Constraints and the Constraints Editor” on page 257

13. Click **OK**.

Editing Blade Assets

A *Blade Server* is a computer built on a thin, modular circuit board that must be plugged into a blade chassis. A *Blade Asset* is an entity that represents a bare-metal physical blade server that can be provisioned. Intel Deployment Manager can automatically discover, track, provision, manage, and re-purpose blade assets.

Blade assets can be managed or unmanaged (typically, for maintenance purposes). Every blade asset must identify the chassis that contains it, as well as its slot in the chassis. You manage blade assets by using the options listed in the **Actions** drop-down menu.

Editing Managed Blade Server Assets

▼ **To edit a managed blade server asset**

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Click the **Blade Assets** tab. The **Displaying All Blade Server Assets** table is displayed.
3. Select the blade server asset you want to edit.
4. Click **Actions** > **Edit**.

Tip Alternatively, click a blade server asset **ID** to display the **Edit Blade Server Asset** dialog for a blade.

5. The **Edit Blade Server Asset** dialog displays the following sections:
   - **Basic**
   - **Network** *
   - **System**
   - **Console**
   - **NIC** *
   - **Hard Disk** *
   - **RPC**
   - **Firmware**
   - **Constraints**

6. The **Basic** section includes information that is collected for your records. Entries in this section are optional. You can go directly to the **Network** section.

Note Model name, serial number, and description are automatically detected from the device when it is added or when an update is performed, if the device supports these attributes. This data is used by Intel Deployment Manager when logging information about the device.

7. In the **Basic** section, you can enter the following information:

Edit Managed Blade Server Asset: Basic Section

Field Name	Description
ID	Displays the unique ID that was assigned by Intel Deployment Manager to identify this asset when it was added.
Description	Enter a description for the blade asset. Maximum 80 characters.
Manufacturer	Displays the manufacturer discovered by Intel Deployment Manager. You can override this information by entering new data (not recommended). Maximum 64 characters.
Model Name	Displays the model name discovered by Intel Deployment Manager. You can override this information by entering new data (not recommended). Maximum 64 characters.
Serial Number	Displays the serial number discovered by Intel Deployment Manager. You can override this information by entering new data (not recommended). Maximum 64 characters.
GUID	Displays the GUID discovered by Intel Deployment Manager. A Globally Unique Identifier, or GUID, is a pseudo-random number used in software applications. Each generated GUID is supposed to be unique.
Miscellaneous Info.	You can enter any information you want in this field and in the **Additional Info** fields. Maximum 256 characters per field.
Additional Info #1	You can enter additional information here.
Additional Info #2	You can enter additional information here.
Additional Info #3	You can enter additional information here.
Additional Info #4	You can enter additional information here.
Additional Info #5	You can enter additional information here.

Edit Managed Blade Server Asset: Basic Section

Field Name	Description
Comment	Enter any comments for this blade asset.
Chassis	Select the chassis where this blade server asset is located.
Slot	Enter the chassis slot number that the blade is in.
Licenses	Select the applicable license(s) from those you registered earlier. When software for which you have added a license is loaded on a server and the server status is UP, one license is counted as being in use

8. The **Network** section includes information that is used by Intel Deployment Manager when managing this computing device. This section is required. Enter the following information:

Edit Managed Blade Server Asset: Network Section

Field Name	Description
Maintenance mode	Maintenance mode enables you to block-out one or many computing devices for any changes that you might need to make to its physical hardware. For a computing device to enter the maintenance mode it must be in the IDLE state.
Architecture *	Select the appropriate architecture for this asset.
Management Protocol*	Select the management protocol. The management protocol is an ActiveOS or a management device driver, deployed to run on an Intel Deployment Manager managed server, or to remotely manage a device. This entry depends on the architecture selected for the asset.
Boot Domain *	Select the boot domain if you installed a boot daemon on a different server on a different part of the network (for example, on a different subnet) than the Intel Deployment Manager server.
IP Address	Enter the IP address for this managed blade asset. If you leave this field blank, Intel Deployment Manager looks through the network for an IP address that can be assigned to this server asset.
Network *	Select the network to which this blade server asset belongs.
Subnet Mask	Enter the subnet mask for this asset. If this field is left blank, the subnet mask of the network is used.
Gateway	Enter the default gateway this asset should use. If this field is left blank, the default gateway for the network is used.

Edit Managed Blade Server Asset: Network Section

Field Name	Description
Next Server IP	Enter the IP address for the next server. In case you want to use another TFTP server besides the one installed by default (on the same server as Intel Deployment Manager), then you must specify another IP address.

9. In the **System** section, enter the following:

Edit Managed Blade Server Asset: System Section

Field Name	Description
Allocatable	Check this check box if you want this blade server asset to be available for server provisioning.
Number of CPUs	Enter the number of CPUs physically supported by this blade server asset. The default is 1.
Procesor Family	Select the processor type.
Supports 64-bit addressing	Check this check box if the processor supports 64-bit addressing. For example, if the asset uses a processor from the Intel Xeon processor family with 64-bit extension, you should check this check box.
Average CPU Clock Rate	Enter the average CPU rate in MHz.
Motherboard Chip Set	Enter the chip set number of the motherboard.
Amount of Memory (RAM)	Enter the amount of memory in MB.

10. Enter the following information in the **Console** section:

Edit Managed Blade Server Asset: Console Section

Field Name	Description
Protocol	Select the protocol used to access this blade server asset's system console.
IP Address	Enter the IP address used to access this blade server asset's system console. If left blank, the address entered in the **Network** section is used.
Port	Enter the port that should be used to contact the console.
Initial Window Size	Select the size for the browser window. For example, **800x600**.

11. Enter the following information in the **NIC** section:

Edit Managed Blade Server Asset: NIC Section

Field Name	Description
ActiveOS Network Device (displayed in edit mode only)	Display only field. Displays the ActiveOS device name of the NIC as discovered by Intel Deployment Manager.
MAC Address *	Enter the MAC Address of the blade server asset in the form: **AF:04:0F:1E:17:69**
NIC is Manageable	Check this check box if you want this NIC to be managed by Intel Deployment Manager.
NIC is Bootable	If the blade asset is set to boot from this NIC, then check this check box. Otherwise, select the **Bootable** check box for the blade server asset. Also, you can have one NIC checked as bootable.
Switch Settings	Enter any extra settings for this switch. The extra switch options are switch dependent and passed directly to the switch for execution.
Description	Enter a description for the NIC. Maximum 64 characters.
Comments	Enter any additional comments for the NIC.

If you want to add another NIC, click **Add NIC**. The Add NIC dialog displays. Enter the required information.

12. Enter the following information in the **HD** (Hard Disks) section:

Edit Managed Blade Asset: Hard Disks Section

Field Name	Description
Interface *	Select either IDE or SCSI.
Device Path *	Enter the location of the hard disk; for example, **`/dev/hda/`**
Saveable	Check this check box if the disk image should be saved when a snapshot is saved. An asset can have multiple hard disks, and you can save or not save one or more of them.
Size *	Displays the size of the disk in gigabytes. You can enter a new value, if applicable.
Hardware Raid Level	If you are using a RAID array, select the RAID Level. This information is *not* automatically discovered.
Misc. RAID Controller Info	Enter any information that you would like to record regarding the Hardware RAID controller.

If you want to add more hard disks, click **Add Hard Disk**. The **Add Hard Disk** dialog is displayed. Enter the required information.

13. Enter the following information in the **Firmware** section:

Edit Managed Blade Asset: Firmware Section

Field Name	Description
Name *	Enter a name for the firmware system.
Manufacturer	Enter the name of the manufacturer for the firmware.
Version	Enter the version for the firmware.
Release Date	Enter a release date in the *mm/dd/yy* format.
Misc. BIOS Info.	Enter any information that you would like to record regarding the firmware.

14. In the **Constraints** section you can edit the constraints applicable to this blade asset. See “Constraints and the Constraints Editor” on page 257.

15. Click **OK** to finish editing the blade server asset. A unique ID is assigned to identify the asset in the Intel Deployment Manager system (the ID appears in the table when the asset is displayed).

Editing Chassis Assets

For information about adding chassis see "Preparing an Intel Server Chassis" on page 188.

▼ **To edit a chassis asset**

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Click the **Chassis Assets** tab.
3. If you want to edit a blade chassis, select it in the **Displaying All Blade Chassis** table.
4. Click **Actions** > **Edit**.
5. The **Edit Blade Chassis Asset** dialog displays the following sections:
   - **Basic**
   - **Network**
   - **Console**
6. The **Basic** section includes information that is collected for your records. This section is optional. You can go directly to the **Network** section. In the **Basic** section, you can enter the following information:

Edit Chassis Asset: Basic Section

Field Name	Description
ID (Edit mode only)	Displays the unique ID that was assigned by Intel Deployment Manager to identify this asset when it was added.
Number of Slots	Enter the number of slots in the chassis.
Description	Enter a description of the chassis asset. Maximum 80 characters.
Manufacturer	Displays the manufacturer discovered by Intel Deployment Manager. You can override this information by entering new data (not recommended). Maximum 64 characters.

Edit Chassis Asset: Basic Section

Field Name	Description
Model Name	Displays the model name discovered by Intel Deployment Manager. You can override this information by entering new data (not recommended). Maximum 64 characters.
Serial Number	Displays the serial number discovered by Intel Deployment Manager. You can override this information by entering new data (not recommended). Maximum 64 characters.
GUID	Displays the GUID discovered by Intel Deployment Manager. A Globally Unique Identifier, or GUID, is a pseudo-random number used in software applications. Each generated GUID is supposed to be unique.
Miscellaneous Info.	You can enter any information you want in this field and in the **Additional Info** fields. Maximum 256 characters per field.
Additional Info #1	You can enter additional information here.
Additional Info #2	You can enter additional information here.
Additional Info #3	You can enter additional information here.
Additional Info #4	You can enter additional information here.
Additional Info #5	You can enter additional information here.
Comment	You can enter any comments specific to this asset.
Location	Enter the rack number of this chassis asset.
Assigned to Workspace	Select the workspace to which you want to assign this chassis.

7. The **Network** section includes information that is used by Intel Deployment Manager when managing this computing device. This section is required. Enter the following information:

Edit Chassis Asset: Network Section

Field Name	Description
Maintenance Mode	Checking this check box turns enables maintenance mode. The asset must be in an IDLE state to go into maintenance mode. You cannot provision a server, save a snapshot, or perform other tasks that require network access while in maintenance mode.
Architecture *	Select the appropriate architecture for this asset.
Management Protocol *	The management protocol is an ActiveOS or a management device driver, deployed to run on an Intel Deployment Manager managed server, or to remotely manage a device. This entry depends on the architecture selected for the asset.
Boot Domain *	Select the boot domain if you installed a boot daemon on a different server on a different part of the network (for example, on a different subnet) than the Intel Deployment Manager server. If you select **Any**, then Intel Deployment Manager selects whichever boot domain is available.
IP Address *	Enter the IP address for this chassis asset.
Port	Enter the IP port for the chassis asset.
User Login ID	Enter the user login ID for this chassis asset.
User Password	Enter the password for the chassis asset's user login ID.
Network *	Select the network to which this blade chassis asset belongs.

8. Enter the following information in the **Console** section:

Edit Chassis Asset: Console Section

Field Name	Description
Protocol	Select the protocol used to access the console of this chassis. • **Chassis KVM**—Opens a browser to a particular blade. • **CMM**—Opens a browser to a chassis management interface.
IP Address	Enter the IP address that you want to use to access the system console of this chassis. By default, Intel Deployment Manager uses the IP address you entered in the **Network** section.
Port	Enter the port that should be used to contact the console.
Initial Window Size	Select the size for the console browser window. For example, **800x600**.

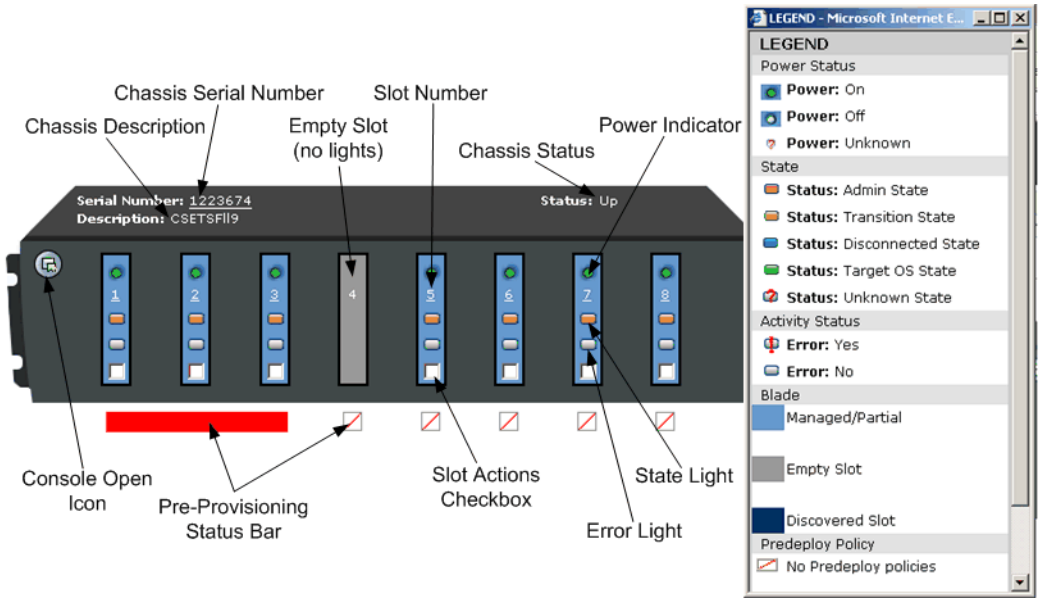
9. Click **OK** to finish editing this chassis asset.

Using the Graphical Chassis Slot View

The graphical chassis slot view provides a graphic representation of a chassis and its slots. It shows the chassis serial number, a description of the chassis, and the chassis status. LEDs indicate whether there is power to a slot, display the state of each blade, and indicate error conditions.

You can select any slot to perform additional actions upon it, or select a logical slot number to edit the blade chassis asset. You can also tell at a glance whether a pre-provisioning policy has been assigned to a slot, and you can add or edit a policy. Click **Legend** to see what various colors mean.

Graphical Chassis Slot View



The graphical view provides the following indicators and controls:

- Open Console Icon—Click the **Open Console** icon in the upper-left to access the Intel Server Chassis Management Module.
- Slot Number—Place your mouse cursor over the slot number of any used slot to view information about the Intel Server Compute Blade in that slot (Serial Number, Architecture, Snapshot, Operating System, Description, and Type).

  Click the slot number to edit the information for the Intel Server Compute Blade. Moving your cursor over, or clicking on the slot number of an unused slot has no effect.
- Power Indicator—This LED is lit when there is power to the Intel Server Compute Blade.
- State Light: The State Light indicates that a blade is in one of the following states (click the online **Legend** for the meaning of colors):
  - ADMIN STATE
  - TRANSITIONAL STATE
  - DISCONNECTED
  - TARGET OS
  - UNKNOWN

- Error: Yes, or No. The **Server Asset Status** popup window displays a message if the server asset has a bad health status and is not busy. If the status of the server is in the OK state, it only displays the bad health status message. If the status of the server is in one of the failed statuses, both the failed status and the bad health status message are displayed.
- Slot Actions: Click the Slot Actions *check box* at the bottom of a slot to select it. Then, you can select any menu from the Menu Bar and initiate an action for the selected slot(s).

  For some actions you can only select one slot at a time, but for others, you can operate on multiple slots at the same time. For example, you might select one slot and click **Actions** > **Edit** to edit a server asset, or select multiple slots and click **Maintenance** > **Reboot** to reboot multiple servers
- Pre-provisioning Status Bar:

  A bar with a diagonal line through it indicates that no pre-provisioning policy is assigned to that slot. If you want to assign a policy to a slot, click the bar to add the new policy.

  A colored bar (without a diagonal line) indicates that a pre-provisioning policy is assigned to the slot. Different colors represent different policies. You can click the bar under a slot to edit a pre-provision policy. If multiple, adjacent slot

  See “Adding or Editing a Pre-Provisioning Policy” on page 177.

Displaying the Graphical Chassis Slot View

▼ To display a graphical view of a chassis asset

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Click the **Chassis Assets** tab.
3. Select one or more chassis assets. If you select multiple chassis, a graphical view is displayed for each one.
4. Click **Show** > **Graphical Slot View**.

Configuring Computing Devices

The following procedures can be performed upon any computing device: server assets, blade assets, or chassis assets.

Accepting Computing Devices

In order to manage a computing device you must first accept it. Only assets in a DISCOVERED state can be accepted.

▼ To accept a computing device

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Click the **Discovered** tab. All discovered computing devices are displayed.
3. Select the computing device(s) you want to accept.
4. Click **Actions** > **Accept**. The verification dialog is displayed.
5. Click **Yes** to accept the selected computing device(s).
6. After you accept a computing device, it needs to be manually rebooted (or, for an Intel Server Compute Blade, manually rebooted or power-cycled from the UI) to load the ActiveOS and continue the auto-discovery process.

   After the reboot is completed, in the **Displaying All Computing Devices** table, the **Asset Status** of the accepted computing device displays UPDATE DONE. You should see `Elemental` displayed on the console of the computing device. This indicates that ActiveOS is running.

Assigning a Computing Device to a Workspace

Assigning a computing device to a workspace makes the device available to that workspace for users.

▼ To assign a computing device to a workspace

1. In the Navigation Area, click **Resources** > **Computing Device**s.
2. Select one of the following categories: **Blade Assets**, **Chassis Assets**, or **Server Assets**.
3. Select the asset you want to assign to a workspace.
4. Click **Actions** > **Assign**.
5. In the **Assign to Workspace** dialog, select the workspace for this computing device.
6. Click **OK**.

Unassigning a Computing Device

You can unassign a resource from a specific workspace if it is no longer to be used by that workspace.

▼ To unassign a resource from a workspace

1. In the Navigation Area, click **Resources** > **Computing Device**s.
2. Click a tab: **Blade Assets**, **Chassis Assets**, or **Server Assets**.
3. Select the asset you want to unassign from a workspace.
4. Click **Actions** > **Unassign**.
5. In the **Unassign from Workspace** dialog, select the workspace from which this resource should be unassigned.
6. Click **OK**.

Deleting Computing Devices

Deleting a computing device removes the device from the Intel Deployment Manager records. It might halt the device if it is running ActiveOS.

▼ To delete a computing device

1. Click **Resources** > **Computing Devices**.
2. Click the **Accepted** tab**.**
3. Select the computing device you want to delete.
4. Click **Actions** > **Delete**.
5. The verification dialog is displayed. The computing device is displayed in the following format: *<description> : <id> (<model>, <serial-number>).* For example,

   Adisys2 : 35299908 (Intel*(R)* Xeon(TM) processor family, 1 x 600MHz, 13660MB)

   For partial assets the computing devices are displayed as

   <mac-addr> : <description> : <id> (<model>, <serial-number>).
6. Click **Yes** to delete the selected device.

Managing Computing Devices 11

From the **Computing Devices** page, you can use the **Maintenance**, **Power**, and **Console** menus available from the **Accepted**, **Blade Assets**, **Chassis Assets**, and **Server Assets** tabs, to assist you in managing selected computing devices. The **Maintenance** menu is also available from the **Provisioned Resources** page.

This chapter includes the following topics:

Using the Maintenance Menu

The **Maintenance** menu supports the following operations:

- **Enter Maintenance Mode** (from the **Computing Devices** page only)
- **Exit Maintenance Mode** (from the **Computing Devices** page only)
- **Reset** (from the **Computing Devices** page only)
- **Rediscover Asset Attributes** (from the **Computing Devices** page only)
- **Reconnect**
- **Abort Operation(s)**
- **Reboot**
- **Reboot to Admin State**

Note All operations listed in the **Maintenance** menu perform the same function for both computing devices and provisioned resources. The following examples are shown as performed from the **Computing Devices** page.

Entering Maintenance Mode

Maintenance mode enables you to block external access so that you can make changes to hardware. The asset must be in an IDLE state to go into maintenance mode.

▼ To enter maintenance mode

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Select a category. You can enter an asset into maintenance mode from the **Accepted**, **Blade Assets**, **Chassis Assets**, or the **Server Assets** category.
3. Select the asset(s) you want to put into maintenance mode.
4. Click **Maintenance** > **Enter Maintenance Mode**.
5. You are asked to verify that you want to enter maintenance mode for the selected asset(s). Click **Yes** to enter maintenance mode.

Exiting Maintenance Mode

▼ **To exit maintenance mode**

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Select a tab. You can exit maintenance mode for an asset from **Accepted**, **Blade Assets**, **Chassis Assets**, or **Server Assets**.
3. Select the asset(s) you want to take out of maintenance mode.
4. Click **Maintenance** > **Exit Maintenance Mode**.
5. You are asked to verify that you want to exit maintenance mode for the selected asset(s). Click **Yes** to exit maintenance mode for the selected asset(s).

Resetting

The `Reset` command resets the state of an asset so that it boots into ActiveOS upon the next reboot. If the asset is already running ActiveOS, the reset command causes a reboot of the ActiveOS. The reset command has no effect on server assets that are busy or that are are booting from a disk.

▼ **To reset a FREE computing device**

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Click a tab. You can reset an asset from **Accepted**, **Blade Assets**, **Chassis Assets**, or **Server Assets**.
3. Select the asset(s) to reset.
4. Click **Maintenance** > **Reset**. You are asked to verify that you want to reset the selected asset(s).
5. Click **Yes** to reset the asset.

Reconnecting

If an asset is in the DISCONNECTED state, you can attempt to reconnect it. Reconnect attempts occur automatically every 15 minutes (by default), but you can manually force an immediate reconnect attempt. If the disconnected asset can be contacted, it is put back into the UP state.

▼ **To manually reconnect an asset**

1. In the Navigation Area, click **Resources** > **Computing Devices** or **Provisioned Resources**.
2. Click a tab.
   - From the **Computing Devices** page, you can reconnect an asset from the **Accepted**, **Blade Assets**, **Chassis Assets**, or the **Server Assets** category.
   - From the **Provisioned Resources** page, click the **Servers** tab.
3. Select the machine(s) that you want to reconnect.
4. Click **Maintenance** > **Reconnect**. You are asked to verify that you want to reset the selected machine(s).
5. Click **Yes** to reconnect the asset.

Rediscovering Asset Attributes

Hardware architecture attributes can be updated on an unallocated computing device running ActiveOS. Its attributes can be rediscovered. This is accomplished using auto-discovery.

For example, assume that a provisioned server has a certain hardware attribute defined initially in a server asset. Then, the server asset hardware is changed or removed. This modification is not known to the provisioned server that maintains the initial (in this case outdated) information until the asset attributes are rediscovered.

It is recommended that you *not* execute `Rediscover Asset Attributes` on a server asset used by a provisioned server because the updated hardware architecture does not propagate to the provisioned server.

Note Make sure there are no other network boot servers (such as bootp servers) that can respond to network boot requests from managed server assets in Intel Deployment Manager managed networks.

▼ **To rediscover the asset attributes of an asset**

1. You must reboot to ADMIN STATE before rediscovering asset attributes. See “Rebooting to Admin State” on page 113.
2. In the Navigation Area, click **Resources** > **Computing Devices**.
3. Select a tab. You can rediscover the asset attributes of an asset from **Accepted**, **Blade Assets**, **Chassis Assets**, or **Server Assets**.
4. Select the asset whose hardware attributes you want to rediscover.
5. Click **Maintenance** > **Rediscover Asset Attributes**. You are asked to verify that you want to rediscover the asset attributes.
6. Click **Yes** to rediscover the asset attributes.

Aborting Operation(s)

Most operations can be aborted, if necessary, after they begin running on an asset. For example, you might begin loading a snapshot and then realize that you meant to load a different snapshot. You can abort the load operation, and then start a new load with the correct snapshot.

You can abort operations like loading or saving a snapshot, or rediscovering asset attributes, but some operations cannot be aborted. For example, Intel Deployment Manager cannot manage an asset while it rebooting, so you cannot abort a reboot operation.

Tip Aborts are asynchronous. For example, if you abort while saving a snapshot with **autodetectOS** on, and then start another snapshot save with **autodetectOS** on, the auto-detection of the OS may fail for the second snapshot save. This is because the first abort did not finish before the second was begun. The solution is simply to wait a minute or so before starting the second snapshot save.

▼ **To abort one or more operations**

1. In the Navigation Area, click **Resources** > **Computing Devices** or **Provisioned Resources**.
2. Click a tab.
   - From the **Computing Devices** page, you can abort an operation from the **Accepted**, **Blade Assets**, **Chassis Assets**, or the **Server Assets** category.
   - From the **Provisioned Resources** page, click the **Servers** tab.
3. Select the asset(s) that is running an operation that you want to abort. Check the status fields of the asset to determine the operation in progress.
4. Click **Maintenance** > **Abort Operation(s)**.

Rebooting

When you reboot a computing device, it reboots from the disk OS. To reboot a computing device, it should be:

- managed
- in the UP state

▼ **To reboot a machine**

1. In the Navigation Area, click **Resources** > **Computing Devices** or **Provisioned Resources**.
2. Click a tab.
   - From the **Computing Devices** page, you can reboot an asset from the **Accepted**, **Blade Assets**, **Chassis Assets**, or the **Server Assets** category.
   - From the **Provisioned Resources** page, click the **Servers** tab.
3. Select the device(s) you want to reboot.
4. Click **Maintenance** > **Reboot**. The verification dialog is displayed.
5. Click **Yes** to reboot.

Rebooting to Admin State

When a managed server is running the ActiveOS, it is said to be in the ADMIN STATE. An ActiveOS is a system software process deployed and run automatically on an Intel Deployment Manager-managed server. The ActiveOS enables remote operation and management of servers that are registered as assets and set to the MANAGED state.

When making any hardware changes, including hot swapping peripherals such as storage and PCI cards, reboot your system into the ADMIN STATE and rediscover the asset attributes to ensure that the changes are detected by Intel Deployment Manager.

If you hot swap a peripheral, there is a discrepancy between the hardware inventory and the actual hardware on the system, but this is non-critical. Just remember to rediscover asset attributes the next time you boot the system to ADMIN STATE. See “Rediscovering Asset Attributes” on page 110.

▼ To reboot to ADMIN STATE

1. In the Navigation Area, click **Resources** > **Computing Devices** or **Provisioned Resources**.
2. Click a tab.
   - From the **Computing Devices** page, you can reboot a device to the ADMIN STATE from the **Accepted**, **Blade Assets**, **Chassis Assets**, or the **Server Assets** category.
   - From the **Provisioned Resources** page, click the **Servers** tab.
3. Select the machine(s) you want reboot to ADMIN STATE.
4. Click **Maintenance** > **Reboot to Admin State**.The verification dialog is displayed.
5. Click **Yes** to reboot to ADMIN STATE.

Performing Remote Power Management

Blades are power managed through the chassis they are in. Enter the chassis asset access information: IP address, user name, and password. When Intel Deployment Manager can access the chassis, the blades can be power managed.

Powering On

Blades are power managed through the chassis they are in. Enter the chassis asset access information: IP address, user name, and password. When Intel Deployment Manager can access the chassis, the blades can be power managed.

▼ To power on computing devices

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Select a tab. You can power on a computing device from **Accepted**, **Blade Assets**, **Chassis Assets**, or **Server Assets**.
3. Select the computing device(s) to power on.
4. Click **Power** > **On**. You are asked to verify that you want to turn on the power for the selected computing devices.
5. Click **Yes** to power on the computing devices.

The displayed status can be: `Power On Queued, Powering On, or Power On Done`.

Powering Off

▼ To power off a computing device

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Select a tab. You can power off a computing device from **Accepted**, **Blade Assets**, **Chassis Assets**, or **Server Assets**.
3. Select the computing device that you want to remotely power off.
4. Click **Power** > **Off**. You are asked to verify that you want to turn off the power for the selected computing device.

5. Click **Yes** to power off the computing device.

The displayed status can be: `Power Off Queued, Powering Off, or Power Off Done`.

Power Cycling

▼ To power cycle a computing device

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Select a tab. You can power cycle a computing device from **Accepted**, **Blade Assets**, **Chassis Assets**, or **Server Assets**.
3. Select the computing device that you want to remotely power cycle.
4. Click **Power** > **Cycle**. You are asked to verify that you want to power cycle the selected computing device.
5. Click **Yes** to power cycle the computing device.

The displayed status can be: `Power Cycle Queued, Power Cycling, or Power Cycle Done`.

Accessing Remote Consoles

The C**onsole** menu offers the **Open** and**Open with** operations, which enable you to access the console of a selected chassis.

Using Console Open

By default, **Console** > **Open** defaults to the selected chassis Intel Chassis Management Module.

▼ **To open a chassis Intel Chassis Management Module**

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Select the chassis whose Intel Chassis Management Module you want to access.
3. Click **Console** > **Open**. An **Enter Network Password** dialog is displayed.
4. Enter the UserID and Password for the Intel Chassis Management Module. The Management Module interface is displayed.

Using Console Open With

For an Intel Server Chassis you can open a console to the Intel Chassis Management Module, or to the chassis KVM.

▼ **To configure information about a chassis console so that you can select access to Intel Chassis Management Module or to the chassis KVM**

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Select the chassis associated with the console you want to access.
3. Click **Console** > **Open with**. The **Open Console** dialog is displayed.

4. Enter the following:

Console Open With Fields

Field Name	Description
Console Protocol	Select a protocol for accessing the chassis. For Intel Server Chassis the default console protocol is **Chassis KVM**.
Console IP	Enter the IP Address of the chassis whose system console you want to access.
Console Port	Enter the port that should be used to contact the console.
Console Initial Size	Select the size for the console's browser window.

5. Click **OK**.

Working with Snapshots 12

A *snapshot* is an Intel Deployment Manager package that includes an operating system (OS), other (optional) software and data, and a configuration registered for use on specific server types. You can take full or incremental snapshots.

This chapter includes the following topics:

Understanding Snapshots

A snapshot is a package that includes an operating system (OS), software and data (optional), and a configuration—it is an image of a machines hard disk(s). Snapshots are useful for:

- performing instant and full disaster recovery
- accelerating server provisioning and deployment
- rapid replication and deployment of software environments
- migration of software environments across hardware models
- reduction in cost and time spent to install, configure, and re-purpose servers

You can save full or incremental snapshots:

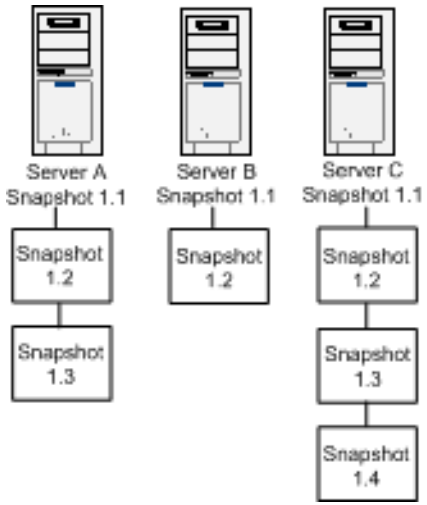
- A *full snapshot* is a complete image of a server or server asset's hard drive(s). You can take a full snapshot from any server or server asset managed by Intel Deployment Manager. A full snapshot is required before you can perform incremental snapshots.
- An *incremental snapshot* can only be taken from a server that is provisioned with a full snapshot, or a live discovered server that had a full snapshot. contains the changes (delta) since the last full or incremental snapshot. An incremental snapshot uses less disk space than a full snapshot and, depending on selected options, can be saved significantly faster than a full snapshot.

You can save snapshots manually, or periodically using a script. You can schedule jobs to save incremental snapshots from single or multiple targets. The full snapshot on which an increment is based is found automatically from the image of the current target machine.When you save a full snapshot:

- A new snapshot tree is created, with the full (base) snapshot saved as the root node.
- Each snapshot tree has only one full snapshot, namely the root node. All other nodes in the tree are increments only.
- When saving a full snapshot, you can save the snapshot in the file mode or sector mode. Incremental snapshots that you take later use the same mode.

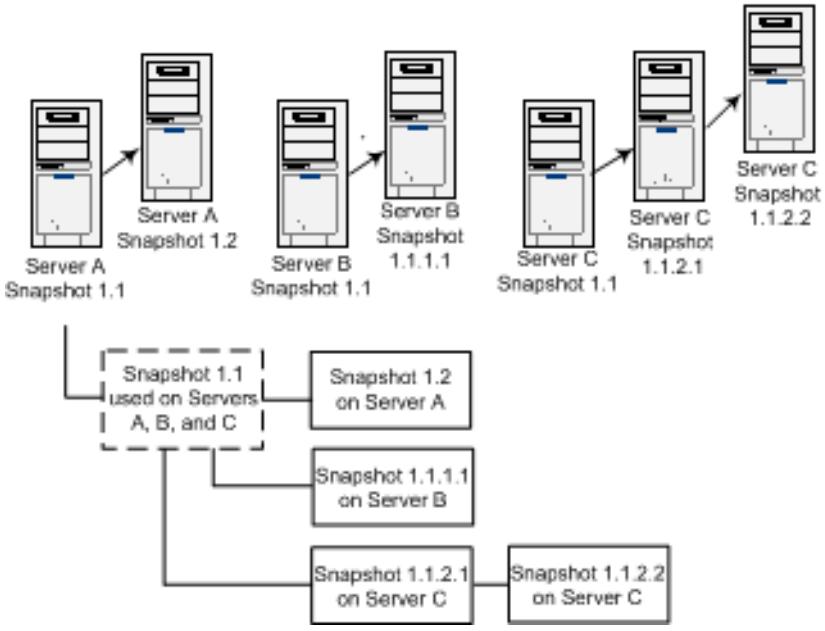
- You can save full backup snapshots periodically and save their increments for multiple targets. This can be achieved in either of the following ways:
  - Save a full snapshot of each target (which goes into its own tree at the root node), and have incremental snapshots go into each tree. Each target has its own snapshot tree. When you perform a full snapshot save for each of the server snapshots, that snapshot is saved at the root node (identified by the dotted box). All consecutive increments of these snapshots are then saved after the root node. The Figure “Full Snapshot Save (Case 1)” describes a scenario where you save a snapshot used on three different servers. For example, Snapshot 1.1 is used on Server A, Server B, and Server C. A full save is performed for the snapshot on each server. So when you perform a full save of the snapshot, as used on three different servers, they are saved as three separate root nodes. The following increments of the snapshot (Snapshot 1.2, Snapshot 1.3, and so on) are then saved in the trunk of the root nodes.

Full Snapshot Save (Case 1)



  - You can also deploy a snapshot (full or incremental) on multiple target servers and then save increments relative to the root node. For example (see the Figure “Full Snapshot Save (Case 2)”), Snapshot 1.1 is used on Server A, Server B, and Server C. No full save is performed for any of the servers, only incremental saves relative to the deployed image are saved (shown in the gray-dotted box). The first increment saved for Snapshot 1.1 (on server A) is saved as Snapshot 1.2 on the same trunk. The increment saved (on server B) for Snapshot 1.1 is saved as Snapshot 1.1.1.1 in the next branch. The following increments saved (on Server C) for Snapshot 1.1 are saved as Snapshot 1.1.2.1 and Snapshot 1.1.2.2 in the next branch.

Full Snapshot Save (Case 2)



You cannot save incremental snapshots if:

- the number of partitions have changed.
- a partition has changed type (for example, from NTFS to ext3)
- it is a snapshot for a network device—incremental snapshots are not supported for network devices

You can always take a full snapshot, and then take incremental snapshots relative to the new base.

After taking a full or incremental snapshot, the system you took the snapshot from must be rebooted to the ADMIN STATE. See “Rebooting to Admin State” on page 113.

Saving Snapshots

You can save snapshots for computing devices or provisioned servers. You must install an ITAP agent before saving a snapshot for the first time on a Microsoft Windows system

Saving Snapshots for Computing Devices

You can only initiate a snapshot save for a server asset or blade asset from the **Computing Devices** (*not* **Provisioned Resources**) page. Note also that you can only save a full snapshot from a server asset.

Before you save a snapshot of a server asset or blade asset ensure that:

- You have sufficient system storage to store the snapshot.
- The selected asset is not in use by a provisioned resource that is in the UP state. Also, no software should be running on the provisioned resource.
- Optionally, you have added applicable licenses for the software on the snapshot.

Note Before saving a full snapshot of a Windows system with an NTFS filesystem, make sure you have the ITAP agent, `itap_agent.exe`, installed on it. See "Preparing to Save a Full Snapshot for a Windows System" on page 131.

▼ To save a snapshot of a server asset or blade asset

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Click the **Accepted**, **Blade Assets**, or **Server Assets** tab.
3. Select the computing device for which you want to save a snapshot.
4. Click **Snapshots** > **Save**. The **Save Snapshot** dialog is divided into the following sections:
   - **Basic**
   - **Constraints**
   - **Advanced**

5. Enter the following information in the **Basic** section:

Save Snapshot for Computing Device: Basic Section

Field Name	Description
Asset	Displays the asset ID.
Name *	Enter a unique name for the snapshot. Maximum 64 characters. Most alphanumeric and special characters are allowed, but avoid greater than (>), less than (<), ampersand (&), single quotes (‘), and double quotes (“).
OS *	Select the operating system.
Licenses	Select any license(s) associated with software that you want to track. When this snapshot is loaded onto a server and the status of the server is UP, the selected license(s) are in use.
Storage *	Select a system storage option. The message, `No System Storage available`, displays if you did not add any system storage.
Comments	Enter any comments about the selected server snapshot.

6. In the **Constraints** section, you can edit the applicable constraints for the snapshot. See “Constraints and the Constraints Editor” on page 257.

7. In the **Advanced** section, enter the following information:

Save Snapshot for Computing Device: Advanced Section

Field Name	Description
Custom save reboot Software	Normally, Intel Deployment Manager reboots your server asset into an ADMIN STATE (running ActiveOS) before saving a snapshot. *Custom save reboot software* is software that can be used, instead of Intel Deployment Manager, to cause the re-boot. Sysprep is an example of software that can cause a reboot on a Microsoft Windows system. To select the custom save reboot software, click the magnifying glass icon icon to display a list of available software packages, and select a package. Then click **Use Selected Item**. Software displays in the **Displaying List of Available Software** table because it was identified in the **Causes Reboot** field of the **Add Software** or **Edit Software** dialog.
Save Reboot Software Arguments	Enter any arguments or switches to be used with the custom save reboot software.

Save Snapshot for Computing Device: Advanced Section

Field Name	Description
Sector by Sector Save	Select the method you want to use to save the snapshot: ◆ **None**—(default) Select this option to save in file mode. ◆ **Partition Sector Save**—Select this option if the snapshot should be saved by storing sectors for each partition on each disk. Snapshots saved in partition mode provide a complete backup of primary and logical partitions on the disks, excluding swap partitions and extended partitions (if they exist). Note that even sectors not used by the file system are saved using this option. ◆ **Disk Sector Save**— Select this option if the snapshot should be saved by storing sectors for each disk. Snapshots saved in this mode provide a complete backup of the disks, covering both partitioned and un-partitioned disk space. Such snapshots can not be used on server assets with different disk sizes or disk geometries. Note that even sectors not used by the file system are saved using this option.
Filesystem Check	**Filesystem Check**—Filesystem Check enables disks to be scanned for errors by running fsck on disk partitions before saving the snapshot. Select one of the following options. ◆ **None**—Does not check and repair the filesystem. An error is returned if a filesystem check was needed. ◆ **Automatic**—Runs a filesystem check and repairs errors only if the filesystem was not unmounted cleanly. ◆ **Force**—Always runs a filesystem check, and repairs errors if necessary.
Ignore FS Check Errors	If you check this check box, any errors detected during a file system check are ignored, and the partition with errors is *not* saved. This option is unchecked by default.
Compression of Snapshot	Check (default) this check box to compress the snapshot as it is saved. Compressing a snapshot takes additional time, but saves space by reducing the size of the snapshot.

8. Click **Save**.

9. In the **Displaying All Server Assets** table, click on **Asset Status** for the selected device to check the progress of the save.

10. When the asset status is `Save Done`, you can click **Resources** > **Snapshots**, then click the **Servers** tab to view the snapshot you just saved.

Saving Snapshots for Provisioned Resources

You can only initiate a snapshot save for a server from the **Provisioned Resources** (not **Computing Devices**) page. Before you save a snapshot of a server ensure that:

- You have sufficient system storage to store the snapshot.
- The selected server is not in use by a provisioned resource that is in the UP state. Also, no software should be running on the provisioned resource.
- Optionally, you have added applicable licenses for the software on the snapshot.

▼ To save a snapshot of a server

1. In the Navigation Area, click **Resources** > **Provisioned Resources**.
2. Click the **Servers** tab.
3. Select the server for which you want to save a snapshot.
4. Click **Snapshots** > **Save**. The **Save Snapshot** dialog is divided into the following sections:
   - **Basic**
   - **Constraints**
   - **Advanced**

5. Enter the following information in the **Basic** section:

Save Snapshot for Provisioned Resource: Basic Section

Field Name	Description
Hostname	Displays the host name of the server for which the snapshot is to be saved.
Incremental Snapshot	Displays **Yes** if this is an incremental snapshot, or **No** if it is a full snapshot.
New Snapshot Name *	Enter a name for the new snapshot. Maximum 64 characters. Most alphanumeric and special characters are allowed, but avoid greater than (>), less than (<), ampersand (&), single quotes (‘), and double quotes (“).
OS *	Displays the operating system used by the server.
Licenses	Select any license(s) associated with software that you want to track. When this snapshot is loaded onto a server and the status of the server is UP, the selected license(s) are in use.
Storage *	Select a system storage option. The message, `No System Storage available`, displays if you did not add any system storage.
Comments	Enter comments about the selected server snapshot.

6. In the **Constraints** section, you can edit the applicable constraints for the snapshot. See “Constraints and the Constraints Editor” on page 257.

7. In the **Advanced** section, enter the following information:

Save Snapshot for Provisioned Resource: Advanced Section

Field Name	Description
Custom save reboot Software	Normally, Intel Deployment Manager reboots your server asset into an ADMIN STATE (running ActiveOS) before saving a snapshot. Custom save reboot software is software that can be used, instead of Intel Deployment Manager, to cause the re-boot. Sysprep is an example of software that can cause a reboot on a Windows system. To select the custom save reboot software, click the magnifying glass icon icon to display a list of available software packages, and select a package. Then click **Use Selected Item**. Software displays in the **Displaying List of Available Software** table because it was identified in the **Causes Reboot** field of the **Add Software** or **Edit Software** dialog.
Save Reboot Software Arguments	Enter any arguments or switches to be used with the custom save reboot software.
Sector by Sector Save	Select the method you want to use to save the snapshot: ◆ **None**—(default) Select this option to save in file mode. ◆ **Partition Sector Save**—Select this option if the snapshot should be saved by storing sectors for each partition on each disk. Snapshots saved in partition mode provide a complete backup of primary and logical partitions on the disks, excluding swap partitions and extended partitions (if they exist). Note that even sectors not used by the file system are saved using this option. ◆ **Disk Sector Save**— Select this option if the snapshot should be saved by storing sectors for each disk. Snapshots saved in this mode provide a complete backup of the disks, covering both partitioned and un-partitioned disk space. Such snapshots can not be used on server assets with different disk sizes or disk geometries. Note that even sectors not used by the file system are saved using this option.
Filesystem Check *	**Filesystem Check**—Filesystem Check enables disks to be scanned for errors by running fsck on disk partitions before saving the snapshot. Select one of the following options. ◆ **None**—Does not check and repair the filesystem. An error is returned if a filesystem check was needed. ◆ **Automatic**—Runs a filesystem check and repairs errors only if the filesystem was not unmounted cleanly. ◆ **Force**—Always runs a filesystem check, and repairs errors if necessary.

Save Snapshot for Provisioned Resource: Advanced Section

Field Name	Description
Ignore FS Check Errors	If you check this check box, any errors detected during a file system check are ignored, and the partition with errors is *not* saved. This option is unchecked by default.
Compression of Snapshot	Check (default) this check box to compress the snapshot as it is saved. Compressing a snapshot takes additional time, but saves space by reducing the size of the snapshot.
Date Based Save	Select this option if you want to perform a date-based save. The date-based stamp is used to determine whether or not a file has changed. This option is only displayed if the **Incremental Snapshot** flag is set. Date-based saves work with Linux (EXT2/EXT3) file systems. NTFS file systems such as Windows 2000 or XP do not support this feature. ◆ When the flag is *not* selected, only the file contents are used to determine if anything changed. ◆ When the flag is selected, a quick check is made to determine whether or not the time stamp of a file has changed. If the time stamp has not changed, it is assumed that the file has not changed. If the time stamp has changed then the contents of the file are considered. Setting the date-based stamp flag results in a faster incremental backup, but with a slight risk that a backup could be incomplete because of an incorrect time stamp.

8. Click **Save and Reboot**.

Caution If you click **Save** the server is rebooted. The server is first booted to ActiveOS, then the snapshot is saved, and finally, the server is rebooted from its disk OS. Click **Cancel** if you do not want to reboot.

9. In the **Displaying All Servers** table, click on **Server Status** for the selected device to check the progress of the save.

10. When the asset status is `Save Done`, you can click **Resources** > **Snapshots**, then click the **Servers** tab to view the snapshot you just saved.

Enable Save on Reboot to Admin State

The **Enable Save on Reboot to Admin State** option ensures that any changes made to a server are saved when it is rebooted to ADMIN STATE.

If this option is enabled, a snapshot is saved for the selected server whenever it is rebooted to ADMIN STATE.

- If the server is provisioned with an incremental snapshot, an incremental snapshot is saved.
- If the server was provisioned with a full snapshot, a full snapshot is saved.

The snapshot can only be used by the selected server, and is automatically updated every time the server is rebooted to ADMIN STATE.

The current setting for this attribute can be displayed under the **Save on Reboot to** ADMIN STATE column in the **Displaying All Servers** table if you choose to display that column in the table.

▼ To enable save on reboot to Admin State

1. In the Navigation Area, click **Resources** > **Provisioned Resources**.
2. Click the **Servers** tab (if it is not already open)
3. Select the server for which you want to enable save on reboot.
4. Click **Snapshots** > **Enable Save on Reboot to Admin State**.
5. Enter information for the following fields:

Enable Save on Reboot to Admin State Fields

Field Name	Description
Hostname	Displays the host name of the server that you are enabling the save for.
Snapshot Name	Enter a name for the snapshot. Maximum 32 characters. Most alphanumeric and special characters are allowed, but avoid greater than (>), less than (<), ampersand (&), single quotes (‘), and double quotes (“).
Storage *	Select the system storage where this snapshot should be stored. The message, `No System Storage available`, displays if you did not add any system storage.

6. Click **Save Changes**.

If you want to disable the save on reboot feature for a particular server, click **Snapshots** > **Disable Save on Reboot to Admin State**.

You can also load a snapshot (see “Loading a Snapshot” on page 183), or load a snapshot and startup the server (see “Loading a Snapshot and Starting Up a Server” on page 185) from the Snapshot menu on the Provisioned Resources page.

Preparing to Save a Full Snapshot for a Windows System

If you want to take a full snapshot of a MicroSoft Windows server asset that uses an NTFS filesystem, you must first install an Intel Deployment Manager ITAP agent on one of the Windows local hard disk partitions.

The ITAP agent enables communication between Intel Deployment Manager and the Windows System, performs hardware attribute detection, supports network personalization, and enables the deployment of software packages and the execution of scripts.

Note You only need to install the ITAP agent if you need to take a full snapshot of a Windows system that is not already managed by Intel Deployment Manager.

▼ To install the ITAP agent

1. Download the Windows `itap_agent.exe` from your Intel Deployment Manager server at http://<Intel Deployment Manager *Server IP Address*>/downloads.
2. Review the readme carefully.
3. Install the ITAP agent using the `-a` option. For example:

   ```
   itap_agent.exe -a -dir c:\itap
   ```

▼ To verify installation of the ITAP agent

1. Check Windows Services to verify that the service named `Device Daemon Monitor` is running, or you can check status by running:

   ```
   itap_agent.exe -s
   ```

2. Verify that the ITAP agent installation directory is created on your Windows system drive (for example, `c:\win_itap`).

Note If the computing device supports the `Stand by` and `Hibernate` features available in the **Power Options** within **Control Panel**; do not enable these two features because Intel Deployment Manager cannot remotely wake up the server if it is in `Stand by` or `Hibernate` mode.

Now, you are ready to save the full snapshot. See "Saving Snapshots for Computing Devices" on page 123.

Managing Snapshots

The **Snapshots** container in the Navigation Area displays a view of all the snapshots that are imported or saved under the Servers tab. The **Servers** tab provides the following menus:

- **Actions**—allows you to edit, HTTP import, HTTP export, server side import, server side export, assign, unassign, and delete snapshots
- **View**—allows you to display a tree and/or table view

Editing Snapshots

▼ **To edit a snapshot**

1. In the Navigation Area, click **Resources** > **Snapshots**.
2. Select the snapshot you want to edit.
3. Click **Actions** > **Edit**.
4. The **Edit Snapshot** dialog displays the following sections:
   - **Basic**
   - **Constraints**

5. When in edit mode most fields in the **Basic** section are display only.

Edit Snapshot: Basic Section

Field Name	Description
Snapshot Name *	You can edit the snapshot name. Maximum 64 characters. Most alphanumeric and special characters are allowed, but avoid greater than (>), less than (<), ampersand (&), single quotes (‘), and double quotes ("). Renaming the snapshot, if in use, updates the snapshot name in the provisioned servers entries.
HW Architecture	The HW architecture that the selected snapshot supports is displayed. For example, **Intel Xeon processor family**.
Revision No.	The revision number of the snapshot is displayed.
HW Manufacturer	The HW manufacturer name of the machine from which this snapshot was saved is displayed.
HW Model	The HW model from which this snapshot was saved is displayed.
OS *	Select the OS from. If the correct OS is not selected, the snapshot will not be loaded successfully onto the computing device.
Incremental Snapshot	When editing a snapshot this is a display only field. You already set the option when saving the snapshot. This flag displays as a check box if you are trying to save a snapshot for a server in the Up state. You see **No** in this field if you are trying to save a snapshot for a server in the DOWN state.
Compression of Snapshot	When editing a snapshot this is a display-only field. You already set the option when saving the snapshot. This option is checked by default. Compressing a snapshot causes the save to take longer, but reduces the size of the snapshot.

Edit Snapshot: Basic Section

Field Name	Description
Date Based Snapshot	When editing a snapshot this is a display-only field. You already set the option when saving the snapshot. The date-based stamp is used to determine whether or not a file has changed. This option is only displayed if the **Incremental Snapshot** flag is set. Date-based saves work with Linux (EXT2 / EXT3) file systems. NTFS file systems such as Windows 2000 or XP do not support this feature. ◆ When the flag is *not* selected, only the file contents are used to determine if anything changed. ◆ When the flag is selected, a quick check is made to determine whether or not the time stamp of a file has changed. If the time stamp has not changed, it is assumed that the file has not changed. If the time stamp has changed then the contents of the file are considered. Setting the date-based stamp flag results in a faster incremental backup, but with a slight risk that a backup could be incomplete because of an incorrect time stamp.
Partition Sector Save	When editing a snapshot this is a display-only field. You already set the option when saving the snapshot. Select this option if the snapshot should be saved by storing sectors for each partition on each disk. Snapshots saved in this mode provide a complete backup of primary and logical partitions on the disks, excluding swap partitions and the extended partition (if any exist). Note that even sectors not used by the filesystem are saved using this option.
Disk Sector Save	When editing a snapshot this is a display-only field. If checked (**Yes**) the snapshot is saved by storing sectors for each disk. Snapshots saved in this mode provide a complete backup of the disks, covering both partitioned and un-partitioned disk space. Such snapshots can not be used on server assets with different disk sizes or disk geometries. Note that even sectors not used by the file system are saved using this option.
Licenses	Select any license(s) associated with software that you want to track. When this snapshot is loaded onto a server and the status of the server is UP, the selected license(s) are in use.
Storage	Displays the storage you assigned for saving this snapshot.
Directory	Displays the directory in which the snapshot is saved.
Creator	Displays the login of the user who imported the snapshot.
Creation Time	Displays the time when this snapshot was imported.

Edit Snapshot: Basic Section

Field Name	Description
Assigned to Workspace	Displays the workspace to which this snapshot is assigned.
Comments	Enter any comments for this snapshot.

6. In the **Constraints** section you can assign constraints to the selected snapshot. Select the required parameters by checking the check boxes. See "Constraints and the Constraints Editor" on page 257.

7. Click **OK** to save the changes to this snapshot, or you can click **Cancel** to exit this dialog without modifying the snapshot.

Applying Licenses to Snapshots

Before you apply licenses to snapshots, you must add the licenses. See "Adding/Editing Licenses" on page 147.

▼ To associate a license with a server snapshot

1. In the Navigation Area, click **Resources** > **Snapshots**.

2. Click a tab: **Servers**, or **Switches**.

3. Select the snapshot to which you want to associate a License.

4. Click **Actions** > **Edit**. The **Edit Snapshot** dialog is displayed.

5. In the **Basic** section, the **Licenses** field enables you to associate licenses with the selected snapshot.

6. Click **OK** after you have made the necessary changes.

Performing an HTTP Import

Snapshots are imported in `.epk` format. The `.epk` file format checks for the validity of the file being imported. If the original `.epk` file was altered and you attempt to import it, the attempt will fail. Because of browser restrictions, the size of the snapshot that you import must be less than 2 GB.

You need to import a snapshot to load it on a computing device. Only snapshots exported by Intel Deployment Manager can be imported. Before importing a snapshot, verify that you have sufficient system storage.

▼ To perform an HTTP import

1. In the Navigation Area, click **Resources** > **Snapshots**.
2. Click **Actions** > **HTTP Import**. The **HTTP Import** dialog is displayed.
3. Enter a name for the snapshot. For example, **`http1_Snapshot.epk`**. Maximum 64 characters.
4. Select the **Storage** for this snapshot.
5. Select the workspace to which you want to assign this snapshot.
6. Enter any comments regarding this snapshot, and click **Continue**.
7. The **HTTP Import...** window appears. Click **Browse** to select the snapshot you want to import, and click **OK**.

Caution Do not close the **HTTP Import** window until the import has completed. If you do, the import will stop.

Performing a Server Side Import

A server-side import provides faster transfer than an HTTP import. Only snapshots exported by Intel Deployment Manager can be imported. There is no restriction on the size of the snapshot you can import.

▼ To perform a server-side import

1. Mount or map the source directory (the directory that contains the snapshot you want to import) to the Intel Deployment Manager server.
2. In the Navigation Area, click **Resources** > **Snapshots**.
3. Click a tab: **Servers** or **Switches**.
4. Click **Actions** > **Import Snapshot (Server Side)**. The **Server Side Import** dialog is displayed.
5. Enter the location of the snapshot in the **Source Snapshot Location** field. Use the full path name of the mounted or mapped directory.
6. Enter the name of the snapshot that you are importing. The file name must have a `.epk` extension. For example, (Linux) `/tmp/MySnapshot.epk`, or (Windows) *`systemdrive`*`:\temp\MySnapshot.epk`.
7. Select system storage for this snapshot.
8. Select the workspace to which you want to assign the snapshot.
9. Enter any comments regarding this snapshot in the **Comment** field.
10. Click **OK**.

Performing an HTTP Export

An HTTP export performs an export of the selected snapshot from the Intel Deployment Manager server to a local machine. Because of browser restrictions, the size of the snapshot that you export must be less than 2 GB.

Note When a snapshot is exported it does *not* get removed from the Intel Deployment Manager server.

▼ To perform an HTTP export

1. In the Navigation Area, click **Resources** > **Snapshots**.
2. Select the snapshot you want to export.
3. Click **Actions** > **HTTP Export**. The **File Download** dialog displays.
4. Choose whether to open the file from its current location, or save the file to your disk.
5. Click **OK**.

Performing a Server Side Export

A server-side export provides faster transfer than an HTTP export. There is no restriction on the size of the snapshot you can export using a server-side export.

▼ To perform a server-side export

1. Mount or map the destination directory (the directory you are exporting to) to the Intel Deployment Manager server.
2. In the Navigation Area, click **Resources** > **Snapshots**.
3. Select the snapshot you want to export.
4. Click **Actions** > **Server Side Export**.
5. Enter the **Destination File Name** for the snapshot. Include the full path name of the mounted or mapped directory. The path you enter must be an existing directory or mount point on the Intel Deployment Manager server. The file name must have a `.epk` extension.
6. Click **Continue**.

Displaying Import/Export Status

You can determine when a transfer has completed by checking the **Import/Export Substatus** field in the snapshots table. This field is hidden by default, so you may have to display it first. See “Setting Table Options” on page 16 for information about displaying and positioning a table column.

When you run an import or export, click the administration console **Refresh** button until you see a a message in the **Import/Export Substatus** field that indicates the transfer has completed. For example, `Export Done`.

Using Search Storage

You can use the **Search Storage** action if you want to search for a snapshot, software, or OpScript on a system storage device. Each subdirectory in the system storage is checked. If files for a snapshot, software, or OpScript are found on the filesystem, but no database entry exists, Intel Deployment Manager creates the database entries and associates them with the files.

Using **Search Repository** you can quickly import packages from a previous installation of Intel Deployment Manager, or from a set of files copied from a different Intel Deployment Manager server. But, note that sharing of package files between multiple Intel Deployment Manager servers is not supported.

▼ To search storage

1. In the Navigation Area, click **Resources** > **Snapshots.**
2. Click **Actions** > **Search Storage**. The **Search Storage** dialog is displayed.
3. Select a system storage device to search.
4. Click **OK**. A list of all server snapshots is displayed.

Assigning Snapshots to a Workspace

▼ **To assign a snapshot to a workspace**

1. In the Navigation Area, click **Resources** > **Snapshots**.
2. Select the asset you want to assign to a workspace. You can assign computing devices from **Blade Assets**, **Chassis Assets**, or **Server Assets**.
3. Click **Actions** > **Assign**.
4. Select a workspace.
5. Click **OK**.

Unassigning Snapshots

You can unassign a snapshot from a workspace if it is no longer to be used by that workspace.

▼ **To unassign a snapshot**

1. In the Navigation Area, click **Resources** > **Snapshots**.
2. Select the snapshot(s) you want to unassign.
3. Click **Actions** > **Unassign**. The **Unassign from Workspaces** dialog is displayed.
4. Select the workspace from which the snapshot should be unassigned.
5. Click **OK**.

Deleting Snapshots

Deleting a snapshot deletes the snapshot and its associated incremental snapshots.

A snapshot cannot be deleted if it was used to create an existing server or a server template. The server or server template has to be deleted before you can delete the snapshot. This restriction does not apply for snapshots of network devices.

▼ To delete snapshot(s)

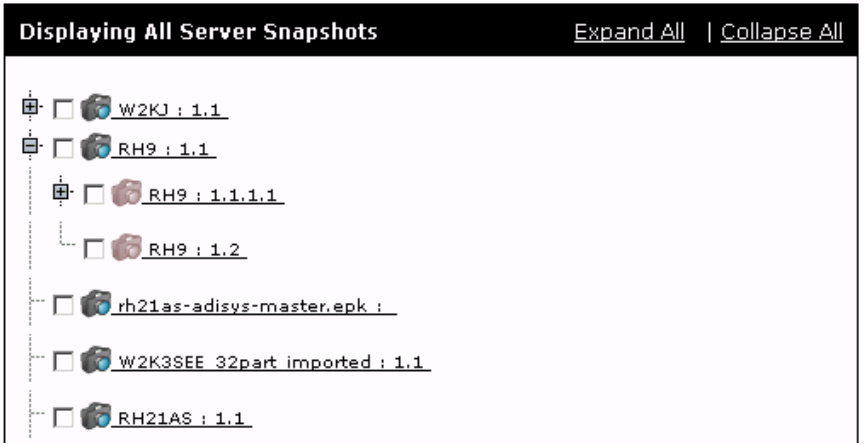
1. In the Navigation Area, click **Resources** > **Snapshots**.
2. Click a tab: **Servers**, **Load Balancers**, or **Switches**.
3. Select the snapshot(s) you want to delete.
4. Click **Actions** > **Delete**. The names of snapshots to be deleted are displayed for verification.
5. Click **Yes** to delete these snapshot(s).

Displaying Tree/Table Views for Snapshots

You can view all available snapshots in a tree or a table view. The **View** menu lists the tree and table options.

The tree view for snapshots displays the saved snapshots in a tree format, as shown in the following figure. You can click a plus (**+**) sign to view incremental backups that have been saved for a snapshot. The **Expand All** and **Collapse All** options enable you to view or hide all of the incremental snapshots.

Tree View



The table view for snapshots displays the saved snapshots in a table format, as shown in the following figure. You may need to scroll horizontally to see all of the columns in the table.

Table View

Name [1]	Revision	Import/Export Status	Architecture
NT4	1.1	Loaded	x86
nt4-adisys-master.epk		Loaded	x86
RH21AS	1.1	Loaded	x86
rh21as-adisys-master.epk		Loaded	x86
rh80-full-adisys-master.epk		Loaded	x86
RH9	1.1	Loaded	x86
RH9	1.2	Loaded	x86
RH9	1.1.1.1	Loaded	x86
RH9	1.1.1.2	Loaded	x86

Tracking Licenses 13

Intel Deployment Manager provides the means to track application and operating system licenses for provisioned, managed servers.

This chapter includes the following topics:

Understanding Licenses

Intel Deployment Manager provides the means to track application and operating system licenses for provisioned managed, servers that are in the UP state.

All license entities have a unique name, a count of managed servers in the UP state that are using a license, and a maximum allowed value for that count. Any snapshot can have an associated set of licenses. When a managed server using a snapshot is started up, the counter for each of the snapshot's licenses increases. If any of the license counters exceed their maximum allowed value, the start up command is aborted. When a managed server is shut down, all of the counters for the server's snapshot licenses are decremented. Therefore, by using Intel Deployment Manager's license entities, you can keep track of, and limit the number of simultaneous running copies of applications or operating systems.

Licenses can also be used to limit server assets that can run server snapshots using a particular license. This limitation is useful if your set of application or operating system licenses are applicable to only a subset of the server assets. You can limit server assets that can be run by unchecking the **Any Server Asset Can Use This license** option for a license. When unchecked, a server asset can be assigned to a server if that server's snapshot has an unchecked license, and if it has that license in its list of licenses. By default, **Any Server Asset Can Use This License** is unchecked for new licenses.

When you provision a server, the constraint check during a start up verifies that your Intel Deployment Manager server has a license. If not, you need to add a license before you can start up.

Managing Licenses

Perform the following operations from the administration console.

Adding/Editing Licenses

▼ To add a license

1. In the Navigation Area, click **Resources** > **Licenses**. The **Displaying All Licenses** table appears.
2. If you want to edit a license, select the license you want to edit.
3. Click **Actions** > **Add** or **Edit**.
4. The **Add License** or **Edit License** dialog is displayed. Enter the following information:

Add/Edit License Fields

Field Name	Description
License Name *	Enter the license name. Maximum 32 characters. Most alphanumeric and special characters are allowed, but avoid greater than (>), less than (<), ampersand (&), single quotes (‘), and double quotes (“).
Description *	Enter a license description. Maximum 64 characters.
Type *	Select a license type (**OS** or **Application**).
Maximum Number of Licenses *	Enter the maximum number of licenses that can be run simultaneously. Maximum value for this field is 9999999.
Current Number of Licenses Used	Enter the current number of licenses used or leave it at 0 (zero) initially. Maximum value for this field is 9999999.
Any Server Asset can use this License	Click the check box if the license can be used by any server managed by Intel Deployment Manager. Some licenses are intended for use on specific servers. Those licenses cannot be used on other servers. A server always consumes licenses while it is not in the DOWN state. If startup fails on a server, but it goes into the DISCONNECTED state, the licenses are still held for that server.
Expiration Date	Enter the date that the license expires. Maximum 16 characters. License expiration is not tracked; this is only for information purposes.

Add/Edit License Fields

Field Name	Description
Assigned to Workspace	Select the workspace to which this license should be assigned. This license is applicable only to those server assets that are assigned to the same workspace.

5. Click **OK** to add the license.

Assigning Licenses to a Workspace

▼ **To assign a license to a workspace**

1. In the Navigation Area, click **Resources** > **Licenses**.
2. Select the license you want to assign.
3. Click **Actions** > **Assign**.
4. Select the workspace to which you want to assign the license.
5. Click **OK**.

Unassigning Licenses

▼ **To unassign a license from a workspace**

1. In the Navigation Area, click **Resources** > **Licenses**.
2. Select the license you want to unassign.
3. Click **Actions** > **Unassign**.
4. Select the workspace from which you want to unassign the license.
5. Click **OK**.

Deleting Licenses

▼ To delete a license

1. In the Navigation Area, click **Resources** > **Licenses**. The **Displaying All Licenses** table is displayed.
2. Select the license(s) you want to delete.
3. Click **Actions** > **Delete**. The verification dialog is displayed.
4. Click **Yes** to delete the license(s).

Using Server Templates 14

A server template provides a set of default values to be used when provisioning servers—this makes the provisioning process much faster and easier. Server templates are especially useful if you provision assets frequently.

This chapter includes the following topics:

Understanding Server Templates

A *server template* provides a set of default values to be used when provisioning one or more servers. Instead of repeatedly, manually entering the same information every time you provision a server, you can enter the information once into a server template, and then assign the template to the server(s) you want to provision.

Given an existing template, you can add a similar template and customize it as needed—you can maintain as many different server templates as you need. For an environment with various provisioning requirements, you might build a library of templates, each customized for a particular configuration.

To use a template to provision servers, see “Provisioning Servers Using a Server Template” on page 172.

Managing Server templates

Perform the following operations from the administration console.

Displaying Server Templates

▼ **To display existing server templates**

1. In the Navigation Area, click **Resources** > **Templates.**
2. Click the **Servers** tab.

   All existing server templates are displayed in the **Displaying All Server Templates Table**.

Adding/Editing Server Templates

When you add a server template you need to specify all of the values to be used when provisioning a server. This process is very similar to provisioning a server (see “Provisioning Managed Servers” on page 164).

Adding and editing server templates are similar tasks. Editing involves updating field entries for an existing template.

▼ To add or edit a server template

1. In the Navigation Area, click **Resources** > **Templates**
2. Click the **Servers** tab.
3. Click **Actions** > **Add** or **Edit**. The following sections are displayed:
   - **Basic**
   - **Management Options**
   - **Access**
   - **Constraints**
   - **NIC**

4. In the **Add Server Template** or **Edit Server Template** dialog, enter the following information under the **Basic** tab:

Add/Edit Server Template: Basic Section

Field	Description
Template Name *	Enter a unique name for the server template. Most alphanumeric and special characters are allowed, but avoid greater than (>), less than (<), ampersand (&), single quotes (‘), and double quotes (“).
Operating System	Select the operating system for this server template. Selecting an operating system acts as a filter when listing the snapshots.
Server Snapshot *	Click the magnifying glass icon to select a server snapshot, and click **Use Selected item**. All available server snapshots that are compatible with the selected OS are listed.
Hostname *	Enter the host name(s) of the server(s) you want to provision. Only alphanumeric characters, hyphens, and *%num%* or *%let%* variables are allowed in a host name. Multiple names should be separated by spaces. If you leave this field blank, Intel Deployment Manager assigns names. To control how host names are assigned to servers, you can use *%num%* or *%let%* variables in the host name you enter. The variable %num% is replaced with a number that makes the host name unique. For example, `redhat`*%num%* is translated into the host name `redhat2` when the server is provisioned, where `2` was chosen by Intel Deployment Manager. The variable *%let%* is similar to *%num%* except that a letter is substituted instead of a number.
Assigned to Workspace	Select the workspace to which you want to assign this server template.

5. In the **Management Options** section, enter the following information:

Add/Edit Server Template: Management Options Section

Field	Description
DNS Domain Name	Enter the name of the DNS domain for your server(s) in the format, ***YourDomain*.com**.
Domain Name Servers	Enter the IP address for your DNS server(s). Separate multiple IP addresses with spaces or commas. For example, **10.1.48.30,10.1.62.12**.
Default Gateway	Enter the IP address of the default gateway that the server(s) should use. If one of the server's NICs specifies a default gateway, this field is ignored. If this field is left blank, Intel Deployment Manager determines the default gateway for the server from the default gateway of the network for the first NIC that does not use DHCP.
Miscellaneous Information	You can enter any information you want in this field.
Enable Personalization	Check this check box to set up the networking information and the host name for the server. If you do not check this option, after loading the snapshot, Intel Deployment Manager cannot manage the server.
Install Remote Access Software	Check this check box if you want Intel Deployment Manager to be able to access this server from a remote location. Leave this check box unchecked (default) if you use different remote access software (such as X windows) on the server(s), as this may cause a conflict. Also leave this check box unchecked if your security policy restricts remote access to servers.
Hard Disk Surface Scan on Start Up	Check this check box if you want to perform a surface scan of the hard disk(s) when the snapshot is loaded on the server. A surface scan ensures that the disks are in good condition before loading the snapshot. This improves reliability, but increases the amount of time it takes to start up the server.
Resize Partition	Check this check box to proportionally resize snapshot partitions, if necessary, when loading the snapshot. Any unpartitioned space between disk partitions is removed. Logical volumes can also be resized. Checking this option enables you to save and restore the used portion of a partition, thereby allowing the target disk on which the snapshot to be restored to have a smaller size.

Add/Edit Server Template: Management Options Section

Field	Description
Resize Last Partition	Check this check box if you want to extend or reduce the last resizable partition to the last sector of the hard drive when loading the snapshot (if the last partition is saved in sector-by-sector mode, or if the last partition is a swap partition, the last partition cannot be resized). This enables you to load a snapshot on a server that has a larger or smaller disk size by extending or reducing the size of the last partition to use the remaining disk space. You may want to use this option if you are loading a snapshot from a smaller disk to a larger disk. For example, if you load a snapshot to a 24 GB target disk from a source disk that has three 4 GB partitions, the first two partitions on the target disk will be 4 GB each, and the last partition will be resized to 16 GB. Do *not* use this option when loading a snapshot to a target disk that is smaller than the source disk. FAT12 / 16 partitions can be resized, but the size is bounded by the file system. **Note** If both **Last Partition** and **Resize Partition** are selected, the **Resize Partition** option takes precedence.
Cross Network Checks	If checked (default), Intel Deployment Manager verifies that the server asset can be automatically reconnected to the provisioning network, if necessary. The automatic re-connection can be done only if one of the following is true: ◆ the server asset and the server network are the same ◆ the VLAN domain of the networks is the same, and a VLAN switch is being used.
Wait for Startup	◆ If checked (default), the server waits for the ITAP agent to start up before being placed in the UP state. If the ITAP agent does not contact the elemental server after a period of time, the server is placed in the UP state. ◆ If unchecked, the server is placed in the DISCONNECTED state when the start up command is issued—It does not wait for the ITAP server to start up. After the ITAP server starts up, the server is placed in the UP state.
IP Address Checks	◆ If checked (default), Intel Deployment Manager validates IP addresses associated with this server in order to avoid address conflicts. ◆ If unchecked, Intel Deployment Manager does not check or allocate the IP addresses associated with the server. This option supports the Load Server Snapshot operation by allowing you to specify an IP address for a server snapshot without requiring it to be a valid IP address.

6. In the **Access** section, enter the following information:

Add/Edit Server Template: Access Section

Field	Description
Protocol	Select a protocol for accessing the terminal of the selected server asset. The options listed are remote access applications that are supported on the target computing device. Note that Hyper Text Transfer Protocol (HTTP) requires that a web server be installed on the target server asset.
IP Address	Enter the IP address of the computing device whose system console you want to access. If this field is left blank the default IP address is used to remotely access the server using the chosen protocol.
Port	Enter the port that is dedicated to the remote access application on the target computing device. The default ports associated with terminal protocols are: ◆ Telnet Protocol—23 ◆ Hyper Text Transfer Protocol—80 ◆ Anonymous File Transfer Protocol—21 If this field is left blank the default IP address is used to remotely access the server using the chosen protocol.
Initial Window Size	Select the size for the browser window. For example, **800x600**.
Access via Console when in ActiveOS	Check this option if you want to access the server asset when it is in the ADMIN STATE.

7. In the **Constraints** section, you can edit the applicable constraints using the constraints editor. See “Constraints and the Constraints Editor” on page 257 You can enter the constraints that you want for the selected template. An error is displayed if incompatible constraints are assigned.

8. In the **NIC** section, enter the following information:

Add/Edit Server Template: NIC Section

Field	Description
Network Name	Select the network to which this NIC is attached. ◆ If you select **Default Network**, the network for the NIC is not specified until you provision a server using the template. ◆ If you provision a server using a network displayed under the **Provisioned Resource** section, the network name for the NIC is set to that network. ◆ If you provision a server from the **Provisioned Resources** page or from the **Computing Devices** page, the provisioning dialog asks you to specify a network to use for the default network.
IP Allocation Scheme	Select one of the following: ◆ **External DHCP**—If you select this option the NIC uses an external DHCP server to determine its IP address. ◆ **Static**—This option causes an IP address to be immediately allocated for this NIC. If you do not enter an IP address, a free IP address is allocated for you. ◆ **Dynamic**—This radio button appears if a network is selected from the **Network Name** field. It causes allocation of the IP address to be delayed until the server is started up. If you **Reboot to Admin State** a server with **Dynamic** set, the IP address is freed when the server is down. You are not allowed to enter an IP address, subnet mask, or default gateway for external DHCP and dynamic allocation schemes. ◆ **No IP Personalization**—The server is not personalized with an IP address. **Note** If the IP address is not personalized, the server uses the IP address from the snapshot that was used to provision it. If this NIC is used to boot from the network, this can result in an address conflict with another system that is using the same IP address.
Subnet Mask	Enter the subnet mask for this network. If this field is left blank, the related management network's subnet mask is used.
Default Gateway	Enter the IP address for the default gateway. If this field is left blank, the server's default gateway is determined from the server's **Default Gateway** field.
NIC Asset ID	Select the ID of the server asset's NIC.

Add/Edit Server Template: NIC Section

Field	Description
Network Device Name	Enter the network device name (`eth0`, and so on). If you enter **Auto** then a name is automatically assigned for the network device. For most servers, **Auto** is the default. For example, you can enter `eth0` for Linux-based systems.
Domain Name Servers	Enter the IP address for your DNS server(s). Separate multiple IP addresses with spaces or commas. For example, **`10.1.48.30,10.1.62.12`**.

If you want to provision a server that has more than one NIC, click A**dd More NIC**.

9. Click **OK**.

Adding a Similar Server Template

You can add an existing template that is similar to a template that you want to create, and then customize it as needed. This enables you to quickly develop new templates to satisfy specific requirements by only having to change a few field entries in each new template.

▼ To add a similar template

1. Identify an existing template that has entries similar to those required for the new template you need. Perhaps you have an existing template that already has all but two of the entries you require, so after adding this similar template, you will only have to edit a few fields to create the new template.
2. In the Navigation Area, click **Resources** > **Templates**.
3. Click the **Servers** tab.
4. Select a server template that is similar to the one you want to add.
5. Click **Actions** > **Add Similar**.

Edit the new template to customize it as discussed in “Adding/Editing Server Templates” on page 153.

Deleting a Server Template

Given various provisioning requirements, you might maintain a number of different server templates, each customized for a particular configuration. As provisioning requirements for your environment change, some of these server templates might become obsolete. If a server template is not in use, and you do not expect to use it again, you can delete it.

You cannot delete a server template if it is in use by a server.

▼ To delete a template

1. In the Navigation Area, click **Resources** > **Templates.**
2. Click the **Servers** tab.
3. Select the template(s) to be deleted.
4. Click **Actions** > **Delete**.

Assigning a Server Template

▼ To assign a server template to a workspace

1. In the Navigation Area, click **Resources** > **Templates.**
2. Click the **Servers** tab.
3. Click **Actions** > **Assign**.
4. Select the template you want to assign to a workspace.
5. Select the workspace to which you want to assign the template.
6. Click **OK**.

Unassigning a Server Template

▼ **To unassign a server template from a workspace**

1. In the Navigation Area, click **Resources** > **Templates**.
2. Click the **Servers** tab.
3. Select the template you want to unassign.
4. Click **Actions** > **Unassign**.
5. Select the workspace from which you want to unassign the template.
6. Click **OK**.

Provisioning Servers 15

The *Provisioning* process involves creating a server object, placing a snapshot onto a physical server asset, *personalizing* the configuration (host name and network), and booting the server asset from its local hard disk into a server with an OS and the personalized configuration. You might perform provisioning to change a bare-metal machine (a machine that has no OS) into a functioning server, or to re-purpose an existing server.

You can provision servers manually, from a server template, or by using an OpScript. You can also set pre-provisioning policies that define operations to be performed automatically on a server when specified conditions are met.

Before provisioning servers, you must first discover and accept the servers. It is also useful to create an IP list.

This chapter includes the following topics:

Provisioning Managed Servers

You can use this procedure to manually provision one or more servers. The server allocation can be dedicated (you select the servers to be allocated) or auto-selected. Auto-selected allocation supports use of a parameterized host name.

▼ To provision one or more servers

1. In the Navigation Area, click **Provisioned Resources**

   Networks and IP lists that you previously added are listed immediately under **Provisioned Resources**.

2. Click the name of a network. The servers in this network are listed in the **Displaying All Servers** table.

3. Click **Actions** > **Provision Managed Server(s)**. The **Provision Managed Server(s)** dialog displays the following sections:

   - **Basic**
   - **Management Options**
   - **Access**
   - **Constraints**
   - **NIC 1**

4. In the **Basic** section enter the following information:

Provision Managed Servers: Basic Section

Field Name	Description
Server Allocation	Select **Dedicated** if you want to allocate selected computing devices, or select Autoselected if you want Intel Deployment Manager to automatically select and allocate available computing devices upon startup. Selecting the type of server allocation determines what other options are available.
Allocate From Workspace (displayed only if **Autoselected** is checked)	Select a workspace. Your selection identifies the workspace that server assets should be allocated from when a server is started up. Server assets are allocated only from the selected workspace. If you select **Any workspace**, any server asset accessible by the user can be allocated, regardless of the workspace it is in.
Quantity (displayed only if **Autoselected** is selected)	Enter the number of servers you want to provision.
Computing Device * (displayed only if **Dedicated** is selected)	Select a computing device from the displayed assets and click **Add**. Only FREE (available) assets are displayed in the list. **Note** The **Any Server Asset** option enables you to provision more than one server using the **Quantity** field. When you select the **Any Computing Device**, Intel Deployment Manager chooses any FREE computing device that matches the snapshot requirements (constraints).
Operating System	Select the operating system for this server. The selected operating system acts as a filter when listing the snapshots.
Server Snapshot *	Select a server snapshot. All available server snapshots that meet the filtering criteria defined in other fields are listed.
Hostname(s)	Enter the host name(s) of the server(s) you want to provision. Only alphanumeric characters, hyphens, and *%num%* or *%let%* variables are allowed in a host name. Multiple names should be separated by spaces. If you leave this field blank, Intel Deployment Manager assigns names. To control how host names are assigned to servers, you can use *%num%* or *%let%* variables in the host name you enter. The variable %num% is replaced with a number that makes the host name unique. For example, `redhat`*%num%* is translated into the host name `redhat2` when the server is provisioned, where `2` was chosen by Intel Deployment Manager. The variable *%let%* is similar to *%num%* except that a letter is substituted instead of a number.

Provision Managed Servers: Basic Section

Field Name	Description
Assigned to Workspace	Select a workspace.

5. In the **Management Options** section, enter the following information:

Provision Managed Servers: Management Options Section

Field Name	Description
DNS Domain Name	Enter the name of the DNS domain for your server(s) in the format, ***YourDomain.com***.
Domain Name Servers	Enter the IP address for your DNS server(s). Separate multiple IP addresses with spaces or commas. For example, **10.1.48.30,10.1.62.12**.
Default Save Snapshot Name	Enter the default name you want to assign to the snapshot when it is saved. If left blank, the server's current snapshot name is used as the default save snapshot name. Most alphanumeric and special characters are allowed, but avoid greater than (>), less than (<), ampersand (&), single quotes (‘), and double quotes (“).
Default Gateway	Enter the IP address of the default gateway that the server(s) should use. If one of the server’s NICs specifies a default gateway, this field is ignored. If this field is left blank, Intel Deployment Manager determines the default gateway for the server from the default gateway of the network for the first NIC that does not use DHCP.
Agent Contact IP	Enter the IP address used to communicate with the ITAP agent running on the server. The agent contact IP address is mapped to the IP address assigned to the bootable NIC of the provisioned server. If this field is left blank, an IP address is automatically assigned.
Agent Contact Port	Enter the ITAP agent contact port. If the ITAP agent listening port is port-forwarded on the network, enter the port forwarding port number. On the provisioned server, the ITAP agent listens at port 10001.
Miscellaneous Information	You can enter any information you want in this field.
Enable Personalization	Check this check box to set up the networking information and the host name for the server. If you do not check this check box, after loading the snapshot, Intel Deployment Manager cannot manage the server.

Provision Managed Servers: Management Options Section

Field Name	Description
Install Remote Access Software	Check this check box if you want Intel Deployment Manager to able to access this server from a remote location. Leave this check box unchecked (default) if you use different remote access software (such as X windows) on the server(s), as this may cause a conflict. Also leave this check box unchecked if your security policy restricts remote access to servers.
Hard Disk Surface Scan on Start Up	Check this check box if you want to perform a surface scan of the hard disk(s) when the snapshot is loaded on the server. A surface scan ensures that the disks are in good condition before loading the snapshot. This improves reliability, but increases the amount of time it takes to start up the server.
Resize Partition	Check this check box to proportionally resize snapshot partitions, if necessary, when loading the snapshot. Any unpartitioned space between disk partitions is removed. Checking this option enables you to save and restore the used portion of a partition, thereby allowing the target disk on which the snapshot to be restored to have a smaller size.
Resize Last Partition	Check this check box if you want to extend or reduce the last resizable partition to the last sector of the hard drive when loading the snapshot (if the last partition is saved in sector-by-sector mode, or if the last partition is a swap partition, the last partition cannot be resized). This enables you to load a snapshot on a server that has a larger or smaller disk size by extending or reducing the size of the last partition to use the remaining disk space. You may want to use this option if you are loading a snapshot from a smaller disk to a larger disk. For example, if you load a snapshot to a 24 GB target disk from a source disk that has three 4 GB partitions, the first two partitions on the target disk will be 4 GB each, and the last partition will be resized to 16 GB. Do *not* use this option when loading a snapshot to a target disk that is smaller than the source disk. FAT12/16 partitions can be resized, but the size is bounded by the file system. **Note** If both **Last Partition** and **Resize Partition** are selected, the **Resize Partition** option takes precedence.

Provision Managed Servers: Management Options Section

Field Name	Description
Cross Network Checks	If checked (default), Intel Deployment Manager verifies that the server asset can be automatically reconnected to the provisioning network, if necessary. The automatic re-connection can be done only if one of the following is true: ◆ the server asset and the server network are the same ◆ the VLAN domain of the networks is the same, and a VLAN switch is being used.
Wait for Startup	◆ If checked (default), the server waits for the ITAP agent to start up before being placed in the UP state. If the ITAP agent does not contact the elemental server after a period of time, the server is placed in the UP state. ◆ If unchecked, the server is placed in the DISCONNECTED state when the start up command is issued—It does not wait for the ITAP server to start up. After the ITAP server starts up, the server is placed in the UP state.
IP Address Checks	◆ If checked (default), Intel Deployment Manager validates IP addresses associated with this server in order to avoid address conflicts. ◆ If unchecked, Intel Deployment Manager does not check or allocate the IP addresses associated with the server. This option supports the Load Server Snapshot operation by allowing you to specify an IP address for a server snapshot without requiring it to be a valid IP address.

6. In the **Access** section, enter the following information:

Provision Managed Servers: Access Section

Field Name	Description
Protocol	Select a protocol for accessing the terminal of the selected server asset. The options listed are remote access applications like Telnet hat are supported on the target computing device. Note that Hyper Text Transfer Protocol requires that a web server be installed on the target server asset.
IP Address	Enter the IP address of the computing device whose system console you want to access.
Port	Enter the port that is dedicated to the remote access application on the target computing device. The default ports associated with terminal protocols are: ◆ secure VNC Access—5900 ◆ Telnet Protocol—23 ◆ VNC HTTP Protocol—5900 ◆ Hypertext Transfer Protocol—80 ◆ Anonymous File Transfer Protocol—21
Initial Window Size	Select the size for the browser window. For example, **800x600**.
Access via Console when in ActiveOS	Check this check box if you want to be able to access the server asset when it is in the ADMIN STATE.

7. In the **Constraints** section, you can edit the applicable constraints using the constraints editor. See “Constraints and the Constraints Editor” on page 257. You can enter the constraints that you want for the selected server. An error is displayed if incompatible constraints are assigned to this server.

8. In the **NIC** section, enter the following information:

Provision Managed Servers: NIC Section

Field Name	Description
Network Name	Displays the network to which this NIC is attached.
IP Allocation Scheme	Select one of the following: ◆ **External DHCP**— If you select this option the NIC uses an external DHCP server to determine its IP address. ◆ **Static**—This option causes an IP address to be immediately allocated for this NIC. If you do not enter an IP address, a free IP address is allocated for you. ◆ **Dynamic**—This radio button is displayed if a network is selected from the **Network Name** field. It causes allocation of the IP address to be delayed until the server is started up. ◆ If you **Reboot to Admin State** a server with **Dynamic** set, the IP address is freed when the server is down. You are not allowed to enter an IP address, subnet mask, or default gateway for external DHCP and dynamic allocation schemes. ◆ **No IP Personalization**- The server is not personalized with an IP address. **Note** If the IP address is not personalized the server uses the IP address from the snapshot that was used to provision it. If this NIC is used to boot from the network, this can result in an address conflict with another system using the same IP address.
IP Address	Enter the IP address(es) for your server(s). If this field is left blank, an IP address is allocated for you.
Subnet Mask	Enter the subnet mask for this network. If this field is left blank, the related management network's subnet mask is used.
Default Gateway	Enter the IP address for the default gateway. If this field is left blank, the server's default gateway is determined from the server's **Default Gateway** field.
NIC Asset ID	Select the asset ID of the NIC. For example, **Boot NIC** if this is the NIC used to boot from the network.
Network Device Name	Enter the network device name. The value, **Auto** (default) automatically assigns a name for the network device. For example, you could enter `eth0` for most Linux-based systems.

Provision Managed Servers: NIC Section

Field Name	Description
Domain Name Servers	Enter the IP address for your DNS server(s). Separate multiple IP addresses with spaces or commas. For example, `10.1.48.30,10.1.62.12`. If you leave this field blank, the domain name servers assigned in the **Management Options** section are used.

9. Click **Done** to provision the server(s).

Editing Servers

You can edit the information you provided when provisioning servers from **Provisioned Resources**. When in edit mode you can also view software logs and application licenses.

1. In the Navigation Area, click **Resources** > **Provisioned Resources**.

2. Click the **Servers** tab.

3. Select the server that you want to edit.

4. Click **Actions** > **Edit**.

   Fields for most sections are the same as when provisioning managed servers (see "Provisioning Managed Servers" on page 164), but the **Hostname(s)** field in the **Basic** section is required when editing. Additional tabs, **Software Logs** and **Licenses,** appear when in edit mode. You can modify field entries as desired.

5. Under the **Software Logs** tab, you can view a software log that contains entries for all of the software run on the server.

6. Under the **Licenses** tab, you can view a list of all application software licenses that are in use by this server. These are the licenses you added for applications on the snapshot used to provision the asset.

7. Click **OK** to finish editing the asset.

Provisioning Servers Using a Server Template

A *server template* provides a set of default values to use when provisioning one or more servers. You can create as many server templates as you need, each one tailored to satisfy specific provisioning requirements. Given a server template, you can quickly provision servers. See “Adding/Editing Server Templates” on page 153 for information about creating server templates.

There are several different ways to provision servers using a template. You can initiate provisioning using a template from the **Computing Devices** page, the **Provisioning Resources** page, or from an OpScript.

Provisioning from Computing Devices

Provisioning from the **Computing Devices** page assumes that you are provisioning dedicated (as opposed to auto-selected) server assets. You select the server assets to be provisioned.

Note that, after provisioning, you cannot load a snapshot or start up a server from the **Computing Devices** page because the server assets become servers as part of the process—load and startup operations must be done from the **Provisioned Resources** page.

▼ To provision one or more servers from the Computing Devices page using a server template

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Click the **Server Assets**, **Blade Assets** or **Accepted** tab.
3. Select one or more assets to be provisioned.

   An asset can only be provisioned if it is in the FREE state.
4. Click **Actions** > **Provision Server(s)**.

5. In the **Provision Managed Server(s) Using a Template** dialog provide information for the following fields:

Provision Managed Server(s) Using a Template Fields

Field Name	Description
Server Template	Select a server template.
Assigned to Workspace	Select the workspace to which you want to assign this asset.

6. Click **OK**.

Provisioning From Provisioned Resources

This procedure is functionally similar to that given in "Provisioning Managed Servers" on page 164. The main difference is that you use a server template to supply default values, so you only have to fill in a few fields when you begin provisioning.

▼ To provision one or more servers from the Provisioned Resources page using a server template

1. In the Navigation Area, click **Resources** > **Provisioned Resources**.

2. Click **Actions** > **Provision Managed Server(s) Using a Template**.

3. In the **Provision Managed Server(s) Using a Template** dialog provide information for the following fields:

Provision Managed Server(s) Using a Template Fields

Field Name	Description
Server Template	Select a server template.
Server Allocation	Select either **Dedicated** or **Autoselected**. ◆ **Dedicated**—If you check **Dedicated** you are choosing to allocate server assets immediately. You can then select a server asset. ◆ **Autoselected**—If you check **Autoselected** server assets are automatically selected upon startup. Enter the number of servers to provision in the **Quantity** field.
Allocate From Workspace (displayed only if **Autoselected** is checked)	Select a workspace. Your selection identifies the workspace that server assets should be allocated from when a server is started up—server assets are allocated only from the selected workspace. If you select **Any workspace**, any server asset accessible by the user can be allocated, regardless of the workspace it is in.
Computing Device * (displayed only if **Dedicated** is selected)	Select a computing device from the displayed assets and click **Add**. Only FREE (available) assets are displayed in the list.
Assigned to Workspace	Select the workspace to which you want to assign this server.

4. Click **OK**.

Provisioning Servers from an OpScript

You can use an OpScript to provision either a managed server asset, or a server. You should be familiar with OpScripts before trying these procedures.

In order to provision server assets or servers from an OpScript using a template you must first add a server template. See “Adding/Editing Server Templates” on page 153.

Provisioning Managed Server Assets from an OpScript

▼ To provision one or more managed server assets from an OpScript using a server template

1. Add an OpScript as shown in **“Adding/Editing OpScripts**” on page 222. To add an OpScript to provision managed server assets use the following settings:

   a. Set **Command Target** to **Managed Server Asset**.

      In this case you are provisioning a dedicated server asset, one that is already managed. This is similar to provisioning an asset from the **Computing Devices** page.

   b. Click the **Add** button associated with the **Command Sequence** field, and select **Provision Server** from the **Command Name** drop-down options list.

   c. Optionally, you can add other commands.

   d. Select a server template.

   e. Click **OK**.

2. After adding the new OpScript, select it from the **Displaying All OpScripts** table.

3. Click **Actions** > **Run.**

4. Select a server asset from the pop-up list.

5. Click **OK**.

Provisioning Servers from an OpScript

▼ To provision one or more servers from an OpScript using a server template

1. Add an OpScript as shown in **"Adding/Editing OpScripts**" on page 222. To add an OpScript to provision managed server assets use the following settings:

   a. Set **Command Target** to **Server**.

   b. Check the check box when the **Provision a New Server** check box is displayed.

   c. Select a server template.

   d. From the **Server Assigned to Workspace** field, select a workspace to which to assign the server.

   e. Click **OK**.

2. After adding the new OpScript, select it from the **Displaying All OpScripts** table.

3. Click **Actions** > **Run.**

4. Select a server asset from the pop-up list.

5. Click **OK**.

Pre-Provisioning

Pre-provisioning enables you to specify rules and actions so that, if an auto-discovered server meets the specified conditions, the actions are automatically performed. For example, you could designate a particular chassis slot to automatically accept, provision, load a snapshot, and start up a blade after it is installed in a slot and discovered. You can add or edit pre-provisioning policies from the **Settings** page, or from the graphical chassis slot view.

The graphical chassis slot view also displays whether a pre-provisioning policy has been assigned to a chassis slot. You can assign colors to pre-provisioning policies so that you can tell at a glance which polices are assigned to slots. You can even add or edit pre-provisioning policies from the graphical slot view. See "Using the Graphical Chassis Slot View" on page 100.

Adding or Editing a Pre-Provisioning Policy

▼ To add or edit a pre-provisioning policy

1. Access a pre-provisioning dialog.

   a. In the Navigation Area, click **Settings** > **Provisioning Policies**. A table of pre-provisioning policies is displayed.

   b. If you want to edit a pre-provision policy, select it.

   c. Click **Actions** > **Add** or **Edit**.

   Tip Alternately, you can access a pre-provisioning dialog by displaying a graphical view of chassis slots as shown in "Displaying the Graphical Chassis Slot View" on page 102. Then click the pre-provisioning bar for a slot. (This is the box immediately below the slot.)

2. The **Add Pre-Provision Policy** or **Edit Pre-Provision Policy** dialog displays the following sections:

   - **Basic**
   - **Rules**
   - **Constraints**
   - **Operations**

3. In the **Basic** section, enter:

Add/Edit Pre-provision Policy: Basic Section

Field Name	Description
Name *	Enter a unique name for this pre-provisioning policy. For example, **`provision_web_server_blades`**. Most alphanumeric and special characters are allowed, but avoid greater than (>), less than (<), ampersand (&), single quotes (‘), and double quotes (“).
Description	Enter a text description of this pre-provision policy. For example: **`accepts blade, provisions using autodetectOS server template, and starts up the server`**.
Comments	You can enter any additional comments here.

4. In the **Rules** section, enter:

Add/Edit Pre-provision Policy: Rules Section

Field Name	Description
Policy Priority	Select a priority. The default is **8 (normal)**. If a server meets the conditions of two or more pre-provisioning policies, the higher priority policy is applied.
All Computing Devices	Check this box if you want the pre-provisioning policy to apply to all computing devices that are discovered after creating this policy. **Caution:** Use this option with caution. This check box affects *all* computing devices.
Serial Number	Check the **Any** box (default) if *any* serial number is acceptable. If you want to enter a specific serial number, uncheck this box and enter the serial number in the **Specific** field. Entering a specific serial number means that the pre-provisioning policy applies only to the server with that serial number.
GUID	Check the **Any** (default) box if *any* GUID is acceptable. If you want to enter a specific GUID, uncheck this box and enter the GUID in the **Specific** field. Entering a specific GUID means that the pre-provisioning policy applies only to the server with that GUID. A Globally Unique Identifier, or GUID, is a pseudo-random number used in software applications. Each generated GUID is supposed to be unique for each server.

Add/Edit Pre-provision Policy: Rules Section

Field Name	Description
MAC Address	Check the **Any** box (default) if any MAC address is acceptable. If you want to enter a specific MAC address that corresponds to one of the server's NICs, uncheck this box and enter the MAC address. Entering a specific MAC address means that the pre-provisioning policy applies only to the server with the NIC that has that address.
Must Be a Blade Server	You must check this box for Intel Server Compute Blades. If you check this box, the following fields are displayed: ◆ **Chassis**—Select a chassis. If no specific chassis is listed you must select **Any Chassis**. ◆ **Slots**—Enter a comma-separated list of slot numbers for which this pre-provisioning policy applies. You can also use a dash to specify a range. For example: **7**, **11**, **14-17**, **24**. ◆ **Color**—You can enter a hexidecimal number corresponding to the color you want to associate with this pre-provisioning policy, or click on the box to the right of the **Color** field to select a color from the pallette. The color you select is displayed in the pre-provison bar under a slot in the graphical chassis slot view. If adjacent slots use the same pre-provisioning policy, the boxes appear together as a single bar of the assigned color. Different colors represent differrent pre-provisioning policies. See "Using the Graphical Chassis Slot View" on page 100.
Disable	Check this box if you want to temporarily disable this pre-provisioning policy. The policy cannot be applied to any servers until the **Disable** check box is unchecked.

5. In the **Constraints** section, you can optionally define constraints to further identify the server for which the pre-provisioning policy applies. For example:

   a. Click the **Constraints** tab.

   b. Click **Add Row(s).**

   c. Select a parameter, operator, and value.

   d. Click **Done with this Row(s)**.

      You can add additional constraints by adding more rows.

   See "Comparing Constraints" on page 261 for further information about constraints.

6. In the **Operations** section, enter:

Add/Edit Pre-provision Policy: Operations Section

Field Name	Description
Accept on Discovery	Check this check box to automatically accept a server when it is auto-discovered. Leave it unchecked if you want to manually accept discovered servers. You must power-cycle the server after it is accepted.
Power Cycle After Accept	Check this check box to automatically power-cycle a server after it is accepted.
Provision a Server	Check this check box if you want to automatically provision a server when it is auto-discovered.
Provision Using Template	If you want to provision a server using a pre-provisioning policy, you *must* specify a server template. Click on the magnifying glass icon to display a list of available server templates. Check the server template you want to use, and click **Use selected item**. See "Using Server Templates" on page 151 for information about templates. The **AutodetectOS** template is a special system-defined template. If you select **AutodetectOS**, Intel Deployment Manager scans the server's hard disks to determine if an OS is already installed. If it is, the server's snapshot is changed to a dummy snapshot for the detected OS.
Assign to Workspace	Select a workspace.
Additional Operations	Select additional operations. For example, you might select to automatically load a snapshot and start up a server after it is provisioned, or run an OpScript that copies additional software to the server.
Delete After Completion	If you check this option, this pre-provisioning policy is deleted after the servers it is applied to go into a MANAGED state.

7. Click OK.

Deleting a Pre-Provisioning Policy

▼ **To delete a pre-provisioning policy**

1. In the Navigation Area, click **Settings** > **Provisioning Policies** (the **Pre-Provision** tab is displayed by default).
2. Select the pre-provisioning policy or policies to be deleted.
3. Click **Actions** > **Delete**. The verification dialog is displayed.
4. Click **Yes** to delete the selected policy or policies.

Loading Snapshots and Starting Up Servers

After provisioning, you need to select a compatible snapshot, load the snapshot onto the target server, and start it up to put it into production.

- If you want to load a snapshot, but not start up the server yet, see “Loading a Snapshot”.
- If you want to use the snapshot that is currently loaded and just start up the server, see “Starting Up a Server”.
- If you want to both load a snapshot and then start up the server, see “Loading a Snapshot and Starting Up a Server”.

Selecting a Compatible Snapshot

Each snapshot contains information about the processor type of the machine from which it was saved. If you want to load a snapshot, the processor type associated with the snapshot must be compatible with the processor type of the target machine onto which you want to load it.

The following table shows compatibility between snapshot and target system processor types (left column = snapshot processor type, top row = target system processor type):

Snapshot versus Target System Processor Compatibility

	Xeon (32-bit)	**Xeon (64-bit)**
Xeon (32-bit)	**X**	**X** (with edit)
Xeon (64-bit)		**X**

To further clarity the compatibility issue, note that:

- A snapshot can be loaded on a machine if the processor type of the machine from which the snapshot was taken matches that of the target machine. For example, a snapshot taken from an Intel Server Compute Blade that uses a processor from the Intel Xeon processor family can be loaded onto a different Compute Blade that also uses a processor from the Intel Xeon processor family.
- A snapshot taken from a machine that uses a processor from the Intel Xeon processor family can be loaded onto a Compute Blade that uses a processor from the Intel Xeon processor family with 64-bit extension, but you must first edit the snapshot and change the constraints to allow the load.
- A snapshot taken from a machine that uses a processor from the Intel Xeon processor family with 64-bit extension *cannot* be loaded onto a machine that does not use a processor with the 64-bit extension.

Loading a Snapshot

A load operation copies a snapshot to the hard disk and personalizes it with a host name and network configuration without rebooting. When you load a server snapshot you are copying the snapshot to the selected server. You can only load a snapshot onto a server that is in the ADMIN STATE.

▼ To load a snapshot

1. In the Navigation Area, click **Resources** > **Provisioned Resources**.
2. Click the **Servers** tab.
3. Select the server(s) on which you want to load the snapshot.
4. Click **Actions** > **Load**. The **Load Server Snapshot** dialog is displayed.
5. If you want to skip personalization for the selected server(s) check the **Skip Personalization** option. For example, you might check this option because you don't want to change the configuration of a snapshot intended for use in disaster recovery.

Caution Checking this option could cause a server to go into a DISCONNECTED state because it is not personalized.

6. Click **Yes** to load the snapshot.

Starting Up a Server

You must have one or more dedicated server assets assigned to the provisioned server in order to use the **Startup** option.

▼ To start up a server

1. In the Navigation Area, click **Resources** > **Provisioned Resources**. The **Displaying All Servers** table is displayed.
2. Select the server(s) you want to start up.
3. Click **Snapshot** > **Startup**. The **Startup Server(s)** dialog is displayed.
4. Select from the following options:

Startup Server(s) Fields

Field Name	Description
Skip snapshot load and start	This option skips loading the snapshot and boots up directly using the copy of the snapshot on the hard disk.
Skip Personalization	This option is unchecked by default. Check this option if you do not want to personalize the server when starting it up. Checking this option overrides any personalization information that you previously entered—it also overrides the **Enable Personalization** check box. You might check this option because you don't want to change the configuration of a snapshot intended for use in disaster recovery. **Caution:** Checking this option might cause a server to go into a DISCONNECTED state because it is not personalized.

5. Click **Yes** to start up the server.

Loading a Snapshot and Starting Up a Server

This option loads the specified snapshot and starts up the server. You must have dedicated server assets assigned to the provisioned server in order to start up the server.

▼ To load a snapshot and start up a server

1. In the Navigation Area, click **Resources** > **Provisioned Resources**. The **Displaying All Servers** table is displayed.
2. Select the server(s) you want to start up.
3. Click **Snapshot** > **Load and Startup**.
4. The **Startup Server(s)** dialog is displayed.
5. Select a startup mode:

Startup Server(s) Fields

Field Name	Description
Auto Load Server Snapshot	This option is always applicable. It causes a snapshot load only if it is necessary. If the server was just created, or if you changed the snapshot name or the computing device for the server, the snapshot is forcibly loaded.
Overwrite Server Disk OS	This option causes the server's disk OS to be overwritten with the OS on the snapshot.
Reboot to Disk OS	This option skips loading the snapshot, and boots up directly using the copy of the snapshot already on the hard disk. This option is provided only if a provisioned server has assets assigned to it.
Skip Personalization	This option is unchecked by default. Check this option if you do not want to personalize the server when starting it up. Checking this option overrides any personalization information that you previously entered—it also overrides the **Enable Personalization** check box. You might check this option because you don't want to change the configuration of a snapshot intended for use in disaster recovery. **Caution:** Checking this option might cause a server to go into a DISCONNECTED state because it is not personalized.

6. Click **Yes** to load the snapshot and start up the server.

Managing Intel® Server Chassis and Server Compute Blades 16

In this chapter you learn how to use Intel Deployment Manager to manage Intel Server Chassis and Intel Server Compute Blades.

This chapter includes the following topics:

Preparing an Intel Server Chassis

Preparing an Intel Server Chassis for management by Intel Deployment Manager involves:

- setting up switch and chassis hardware
- configuring the chassis using the Intel Server Management Module
- accepting the chassis
- installing the Intel Server Compute Blades
- setting boot order for each blade

▼ To prepare a chassis to be managed

1. Prepare and connect switch ports (you need an external switch with at least three free ports):

   a. Verify that there is a 4-port gigabit switch properly installed on the back of the chassis.

   b. Connect one port on the 4-port gigabit switch to the external switch.

      Intel Deployment Manager only supports management of a chassis over a Ethernet switch using CAT5. Fiberchannel switches are not supported.

   c. Connect the chassis management port to the external switch.

2. Power on the chassis.

3. Make sure that the chassis and the Intel Deployment Manager server are on the same LAN:

   a. Login to the Intel Server Management Module on a client connected to the chassis.

   Note The Intel Server Management Module is used to configure blade servers and other modules in the chassis. It communicates with multiple blade servers to provide functions like blade server power-on requests, error and event reporting, and requests for storage, I/O devices, and ports.

   b. Select **MM Control** > **Network Interfaces**.

   c. Under **External Network Interface (eth0)** > **Static IP Configuration**, verify that the interface is on the same LAN as Intel Deployment Manager server.

d. Save the configuration.

e. If you made any changes, select **MM Control** > **Restart MM**. When MM is restarted you are automatically logged out of your browser session.

4. After a couple of minutes, the chassis is automatically discovered. If you want to initiate immediate discovery, you can use the procedure given in "Forcing Chassis Discovery" on page 192.

If the chassis is successfully discovered it is listed under both **Discovered** and **Chassis Assets** tabs.

Note To successfully update chassis on an Intel Deployment Manager server running Red Hat Linux 9.0, ssl must be installed on the server. For example, you might install `openssl096b-0.9.6b-3.i386.rpm`.

5. You must accept discovered chassis before they can be managed. See "Accepting a Discovered Intel Server Chassis" on page 192.

6. Plug blades into the chassis.

Note Running Intel Deployment Manager on a blade is not supported.

7. For each blade, set the boot order so that it boots from the network before attempting to boot from a hard disk. From the Intel Server Management Module:

a. Select **Blade tasks** > **Configuration**.

b. Set the boot order. For example: `CD`, `Floppy`, `Network`, `HD0`.

This boot order causes the blade to attempt to boot from the network (Network-PXE) before trying to boot from a hard disk. Booting from the network enables an ActiveOS to be loaded and run in memory on the blade.

c. Save the configuration.

Repeat this step for each blade in the chassis.

Preparing Chassis that have Newer Firmware

Chassis running newer firmware (73E and higher) have been certified for use with Intel Deployment Manager, but additional settings are required to enable to discovery and management of the chassis and blades. Use the Intel Server Management Module to set the following:

- In the Simple Network Management Protocol (SNMP) section:
  - The SNMPv1 agent should be **enabled**.
  - For the `private` community, `set` access, enter the IP address of the Intel Deployment Manager server.
- In the TCP Command Mode Protocol section: set **Command Mode** to **enabled** (which is the default).
- In the Service Location Protocol (SLP) section: set **Address type** to **Broadcast** (the default is **Multicast**). If this value is not set to **Broadcast** chassis cannot be discovered.

Managing Intel Server Chassis

Intel Deployment Manager uses the Service Location Protocol (SLP), described in RFC 2608, to discover chassis. You can change the default interval between automatic discovery attempts, or manually initiate chassis discovery.

You can also display a graphical representation of a chassis and its slots. The *graphical chassis slot view* provides information about the chassis and the blades it contains. You can perform various actions from the graphical interface. See “Displaying the Graphical Chassis Slot View” on page 102.

Enabling Discovery of Chassis Attributes

Automatic discovery (and rediscovery) of chassis attributes is enabled by default. When enabled, Intel Deployment Manager checks chassis at regular intervals, and updates information if any chassis attributes have changed.

We generally recommend that the discovery of chassis attributes feature be enabled, but there may be times when you may want to temporarily disable it. For example, if you want to delete a chassis and the blades it contains, you must first delete the blades. You might delete all of the blades, but before you can delete the chassis, the chassis attributes are rediscovered and automatically added. In this case, you should disable discovery until you have successfully deleted the chassis, then re-enable it.

▼ **To enable discovery of chassis attributes**

1. In the Navigation Area, click **Settings** >**Preferences**.
2. In the **Enable discovery of chassis attributes** field:
   - Click **True** to toggle the value to **False**. When the value is **False**, auto-discovery is disabled.
   - Click **False** to toggle the value to **True**. When the value is **True**, auto-discovery of chassis attributes is enabled.
3. Click **Submit**.

Changing the Default Discovery Interval

By default, a broadcast is sent out every two minutes to automatically discover chassis. Note that you must enable chassis discovery (as shown in "Enabling Discovery of Chassis Attributes" on page 190) and set a broadcast interval greater than 0 for chassis attributes to be discovered.

▼ **To change the broadcast interval**

1. In the Navigation Area, click **Settings** >**Preferences**.
2. Click on the value shown for the global setting, **Set SLP Poll Interval for Blade Chassis Discovery In Seconds, 0 disables polling**. The **Change Preferences** dialog is displayed.
3. Enter the new value.

   This is the number of seconds between broadcasts. The default is 120 seconds. Setting this value to zero disables auto-discovery.
4. Click **Submit**.

Forcing Chassis Discovery

▼ **To immediately initiate discovery of a chassis**

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Click the **Discovered** tab.
3. Click **Actions** > **Initiate Active SLP Discovery**.

Discovered chassis appear under the **Discovered** tab. After discovering a chassis, you must accept it before it can be managed by Intel Deployment Manager.

Accepting a Discovered Intel Server Chassis

You must accept a discovered chassis before it can be managed.

▼ **To accept a chassis**

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Under the **Discovered** tab, select the chassis you want to accept.
3. Click **Actions** >**Accept**.

After you accept the chassis, as Intel Deployment Manager attempts to learn about the blades it contains, the status of the chassis is shown as UPDATING. When this process is complete, the chassis appears under the **Chassis Assets** tab with a status of UPDATE DONE.

Troubleshooting if Chassis Discovery Fails

If you try to accept a discovered chassis and the attempt fails with a `Bad Password` or `Login` error, you can fix the problem by manually enabling SNMP on the chassis.

▼ **To enable SNMP on the chassis**

1. Login to the Intel Server Management Module on the chassis.
2. Select **MM Control** > **Network Protocols**.
3. Select **SNMP Agent: enabled**.
4. Add **Community Name** = `private`, and **Host Name or IP Address** as the name or IP address of the Intel Deployment Manager server.

5. Save the configuration.

6. Select **MM Control** > **Restart MM.**

Changing the IP Address of a Chassis

If you change the IP address of a discovered or accepted chassis using the Intel Server Management Module you should manually update Intel Deployment Manager to ensure the information is correct.

Changing the IP Address of an Accepted Chassis

If you use the Intel Server Management Module interface to change the IP address of a chassis, you also need to manually update the information Intel Deployment Manager has.

▼ **To change the IP address of an accepted chassis**

1. In the Navigation Area, click **Resources** > **Computing Devices**.

2. Click the **Accepted** tab.

3. Select the chassis.

4. Click **Actions** > **Edit**.

5. Click the **Network** tab.

6. Update the IP address.

7. Click **OK.**

8. Click the **Discovered** tab.

9. If a duplicate chassis appears (the same chassis as the one already accepted, but with a different IP address), delete it.

Changing the IP Address of a Discovered Chassis

After changing the IP address of a chassis using the Intel Server Management Module interface, you should delete any previously discovered chassis that have the old IP address.

▼ **To delete a previously discovered chassis**

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Click the **Discovered** tab.
3. Delete the chassis that has the old IP address. To identify the correct IP address to delete:
   a. Click **Table** > **Table Options.**
   b. Select **Management IP Address** from **Hidden Columns**, and click the **<<** button to move it to **Displayed Columns.**
   c. Click **OK**. You can now see the Management IP Address in the **Displaying all Discovered Server Assets** table.

Rejecting a Discovered Chassis

When you reject a discovered chassis it is removed from the **Discovered** list. Intel Deployment Manager does not manage, and does not normally attempt to rediscover rejected chassis.

Note Intel Deployment Manager uses the IP address of the chassis to determine its identity. Changing the IP address can cause it to be rediscovered.

▼ **To reject a chassis**

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Under the **Discovered** or **Chassis Assets** tab, select the chassis you want to reject.
3. Click **Actions** >**Reject**.

Deleting a Network or Chassis if there are Rejected Assets

If you get an error message when attempting to delete a network or a chassis, and you have rejected blades or chassis, try deleting the rejected assets.

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Click the **Rejected** tab.
3. Select all of the rejected blades and chassis.
4. Click **Actions** > **Delete**.
5. Retry the delete network or delete chassis operation.

Deleting an Accepted Intel Server Chassis

If you do not want Intel Deployment Manager to manage a chassis and its blades anymore, delete the blades, then delete the chassis. You cannot delete a chassis without first deleting its blades. You should disable auto-discovery of chassis attributes before trying this procedure; then re-enable it after you have successfully deleted the blades and chassis (see “Enabling Discovery of Chassis Attributes” on page 190).

1. In the Navigation Area, click **Computing Devices** > **Accepted**.
2. Select all blades that belong to the chassis.
3. Click **Actions** > **Delete**.
4. Select the chassis to be deleted.
5. Click **Actions** > **Delete**.

Managing Intel Server Compute Blades

In addition to the table view, you can display a graphical chassis slot view to obtain information about a chassis and the blades it contains. You can perform various actions on the blades from the graphical interface. See “Displaying the Graphical Chassis Slot View” on page 102.

Note that you can pre-provision a chassis slot so that, if a discovered blade meets conditions you define, specified actions are automatically performed. For example, you could designate a particular chassis slot to automatically accept, provision, and start up a blade after it is installed in a slot and discovered. See “Pre-Provisioning” on page 177. General information about provisioning is given as a workflow in “Provisioning Servers” on page 33, and in the chapter “Provisioning Servers” on page 163.

Discovering Intel Server Compute Blades

Before discovering blades you must prepare them for discovery. See “Preparing Computing Devices for Discovery” on page 78.

Properly prepared blades are automatically discovered after they are inserted into a chassis slot. By default, blade discovery occurs every five minutes.

Accepting and Power Cycling Intel Server Compute Blades

Blades in the DISCOVERED state can be accepted using the same procedure used for other types of computing devices. See “Accepting Computing Devices” on page 103. After accepting a blade you can power cycle the asset to put it into the maintenance mode.

▼ **To power cycle the asset**

1. In the Navigation Area, click **Resouces** > **Computing Devices**.
2. Click the **Accepted** tab.
3. Click **Power** > **Cycle**.

No Bootable NICs Error

An attempt to accept blades without first defining a network results in assets with the status: `Server asset XXXXXXX:YYYYYYY:Intel Blade Server has no bootable NICs.`

▼ **To resolve this problem**

1. Add a network. See "Adding/Editing Networks" on page 62.
2. Delete the blades that are in the error state. See "Deleting Computing Devices" on page 105.
3. Rediscover attributes for the chassis. See "Forcing Chassis Discovery" on page 192.

Adding an Intel Server Compute Blade

Whenever a blade is added to a chassis, Intel Deployment Manager needs to rediscover the chassis attributes.

▼ **After adding a blade to a chassis, perform the following procedure to get an *accepted* chassis to discover the blade**

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Click the **Accepted** tab.
3. Select the chassis into which the blade was inserted.
4. Click **Maintenance** > **Rediscover Asset Attributes**.

After the asset attributes are rediscovered, you can click the **Chassis Assets** tab and select **View Slots** for the chassis to verify that the new blade was discovered.

Moving an Intel Server Compute Blade

Whenever a blade is moved from one slot to another on a chassis, Intel Deployment Manager needs to rediscover the chassis attributes.

▼ **After moving a blade from one slot to another on an *accepted* chassis, perform the following procedure to update the slot information for all of the blades**

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Click the **Chassis Assets** tab.
3. Select the chassis to which the blade was moved.
4. Click **Maintenance** > **Rediscover Asset Attributes**.

After the slot information is updated, you can click the **Chassis Assets** tab and select **View Slots** for the chassis to verify that the new blade was discovered.

Removing an Intel Server Compute Blade

Whenever a blade is removed from a chassis, Intel Deployment Manager needs to rediscover the chassis attributes.

Intel Deployment Manager does not automatically remove a blade from its database. If you want to permanently remove a blade from an accepted chassis, you must also manually delete the blade. If the blade was provisioned before being removed, you must remove any provisioned resources that are using the blade.

▼ **To remove the blade**

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Click the **Chassis Assets** tab.
3. Select **View Slots** on the chassis from which the blade was removed.
4. Select the blade that was removed.
5. Click **Actions** > **Delete**.

Deploying OpScripts on One Intel Server Compute Blade

▼ **To deploy OpScripts on one blade**

1. In the Navigation Area, click **Resources** > **Jobs**.
2. Click the **OpScripts** tab.
3. Select an OpScript (**Startup** or **Shutdown**) by clicking on its check box.
4. Click **Actions** > **Run**.
5. On the **Run OpScript** page, click the magnifying glass icon next to **Script Parameter—Server**. A popup window appears with a list of managed servers.
6. Select the server on which you want to run the OpScript.
7. Click **Use Selected Item**.

Deploying OpScripts on Multiple Intel Server Compute Blades

▼ **To deploy OpScripts on multiple blades**

1. In the Navigation Area, click **Resources** > **Jobs**.
2. Click the **Jobs** tab.
3. Click **Actions** > **Add**.
4. On the **Add Job** page, click on the magnifying glass icon next to **Script Name**. A popup window appears with a list of OpScripts.
5. Select the OpScript which you want to schedule as a job.
6. Click **Use Selected Item**.
7. Enter a Job Name and Description.

8. For **Execution Criteria**, select an option:
   - Use **On Scheduled Dates** if you want the job to be executed based on time and date.
   - Use **On Occurrence of Listed Event Category** if you want the job to be executed based on a specified event.
9. For **Wild Card Parameter** select **Script Parameter—Server** and choose a **Wild Card Parameter Value**.
10. Click **OK**.

Working With Software 17

You can use Intel Deployment Manager to create and deploy applications, patches, scripts, utilities, and other types of software.

This chapter includes the following topics:

Understanding Software Categories

Intel Deployment Manager supports the following categories of software:

- **Applications** are programs issued by software vendors.
- **Patches** are updates issued by software vendors or manufacturers.
- **Scripts** are customized software written in a scripting language.
- **Utilities** are applicable at the firmware, BIOS, and hardware layer. Using utilities software, you can accomplish: BIOS and firmware updates, or a utility update for a new network card type.
- The **Others** category includes activities such as deploying customer data. For example, adding content to your database.

Understanding Software Types

Intel Deployment Manager supports the following software types:

- Bootable Software—Bootable software can be directly booted by a target server asset. Uses of bootable software include remote booting of OS kernels, floppy disk images, and other software that is directly loaded into memory and executed without a pre-existing, running operating system. This type of software is useful for BIOS upgrades, Hardware RAID configurations, and other low-level device operations that require a clean, real-mode DOS or EFI environment.
- Configuration Software—Configuration software is run on servers that Intel Deployment Manager has deployed a snapshot to, and started up. Configuration software typically includes scripts or binaries compatible with the managed server's running OS.
- Control Software—Control software is run on server assets running ActiveOS (that is, servers in the DOWN state, or server assets that are not assigned to a server or which are assigned to down servers). Control software can be a script or binary compatible with the server asset's ActiveOS. We recommend that your control software be written as a Bourne shell (`sh`) script because it is the only type compatible with all versions of ActiveOS.

Creating Bootable Software for Intel Xeon Processor Family Machines

Bootable software contains an OS image that can be loaded and directly booted on managed servers. For bootable software that uses 1.44 MB MS-DOS floppy images, the image must be created manually.

The following procedure describes how to create bootable software for a DOS-based operating system. You can use any Intel Xeon processor family PC with a floppy drive.

▼ To create bootable software for a DOS-based operating system

1. Start with a DOS-based operating system such as DOS 6.22. You could also use a UNIX or Linux utility that creates a bootable DOS-formatted disk.

   If you are using DOS from the command line:

   a. Insert a floppy disk into the floppy diskette drive.

   b. Type: **`format a: /q /s`**

      The **`/q`** is for quick format. The **`/s`** is to transfer system files to the floppy, making it bootable.

   c. When you see a `format complete, System transferred` message the disk is ready. Go to step 2.

   If you are using a GUI-based OS such as Windows then:

   a. Insert a floppy disk into the floppy diskette drive.

   b. Use the instructions provided with your system to create a Startup Disk.

   Windows formats the disk and transfers system files as well as additional tools. The additional files can be deleted if space is limited. A complete list of required files is listed in step 6.

2. Copy your files to the root level of a bootable floppy disk (`a:\`).

3. Copy `wrapinit.com` and `wrapdone.com` to the root level of the same bootable floppy disk. These files are located in *`install_directory`*`\opbootapps` (Windows) or *`install_directory`*`/tools/opbootapps` (Linux) on the Intel Deployment Manager server. They manage communication with the Intel Deployment Manager server and reboot the server on termination.

Note Alternatively, you can use `wrapapp.com` instead of `wrapinit.com` and `wrapdone.com`. The `wrapapp.com` program is easier to use, but requires 64MB of real mode memory while your program is running.

4. Create or edit a file called `autoexec.bat`. This is the first file that executes if the PC is booted from the floppy disk.

```
Line 1: @echo off
Line 2: wrapinit.com
Line 3: test.bat
Line 4: wrapdone.com 0
```

   `test.bat` is your program (which can also be an `.exe` or `.com` file).

5. Test your program. Run `test.bat` and verify that it works before imaging the floppy. Also, create a `config.sys` file, if necessary.

6. The final directory contents on the floppy disk are:

   - `command.com` (this is the command interpreter)
   - `*.sys` files (required system files)
   - `autoexec.bat`
   - `test.bat`
   - `wrapinit.com`
   - `wrapdone.com`

7. Create a binary image of the bootable floppy.

   - For Windows, use a Windows-based shareware utility such as WinImage. You can download it from: `http://www.winimage.com/download.htm`
   - For UNIX/Linux, use the `dd` command. Insert the bootable floppy into the drive of a PC running Linux and enter: **`dd if=/dev/fd0 of=/tmp/test.bin`**

8. Zip the file `test.bin` into `test.zip`

9. In the Navigation Area, click **Resources** > **Software**.

   a. Click **Actions > Add**.

   b. In the **Command Line** field, enter: **`dos=%FILE%\test.bin;%ARGS%`**

   c. When prompted, browse for `test.zip` and upload it to the system. No booter was specified so the default booter, OpBoot, is used by default. The command line conveys to OpBoot the name of the floppy disk image. It also tells OpBoot to pass the arguments entered when the software was run. OpBoot returns the execution status back to the Intel Deployment Manager server.

Using Bootable Software Command Lines

The identifier *file-path* is a file path relative to the TFTP server directory. The identifier *booter-argument-string* is an argument string that is passed to the boot mechanism. If a DHCP server is the boot mechanism, the argument string will represent a DHCP vendor option. The following variables can be used in the identifiers:

- *%FILE%*—The directory string within which the bootable software was placed in the TFTP server directory replaces this token.
- *%ARGS%*—Replaced by the additional arguments entered when the software was executed. They are directly passed to the booter.

Using Configuration and Control Software Command Lines

For configuration and control (ActiveOS) software, the command line is executed when the files are placed on the target machine. The current directory is set to the directory that represents the ZIP root directory. As with bootable software, additional arguments entered when the software was executed are specified in the command line by *%ARGS%*.

Managing Software

Perform the following operations from the administration console.

Adding/Editing Software

After creating a software package, you can add it. Before you add software, verify that the software package is in `.zip` format, and you have enough system storage to store it.

▼ To add/edit software

1. In the Navigation Area, click **Resources** > **Software**.
2. Click a tab: **Applications**, **Utilities**, **Scripts**, **Patches**, or **Others**.
3. Click **Actions** > **Add** or **Edit**.
4. Select the software from the **Displaying All Software** table. The **Add Software** or **Edit Software** dialog displays the following sections:
   - **Basic**
   - **Constraints**
5. Enter the following information in the **Basic** section:

Add/Edit Software: Basic Section

Field Name	Description
Name *	Enter a unique name for this software. Maximum 32 characters. Most alphanumeric and special characters are allowed, but avoid greater than (>), less than (<), ampersand (&), single quotes (‘), and double quotes (“).
Description	Enter a description for the software. The description is a short summary for the software. Maximum 128 characters.
Category	The category of software you selected is displayed.
Version	You can enter the version of the software for your records. Maximum 64 characters.
Type *	Select the type of software: **Bootable Software**, **Configuration Software**, or **ActiveOS Software**.

Add/Edit Software: Basic Section

Field Name	Description
Architecture *	Select the processor architecture of the server on which this software will be executed.
OS *	Select the OS that the server executing this software is using. This selection is necessary for configuration software and control software. Bootable software does not require a running operating system.
Command Line *	Enter the commands to be run when software is executed. All command lines support the following macros: ◆ *%FILE%* is the root directory of the `.zip` file. ◆ *%ARGS%* are additional arguments. For bootable software, the command line specifies the bootable file within the `.zip` package, and how to load the software. By default, the OpBoot loader is used to load bootable software. For example: `dos=`*%FILE%*`/floppy-144MB.img;`*%ARGS%* tells the OpBoot loader to load an MS-DOS floppy image. The image is named floppy-144MB.img in the root directory of the `.zip` file. Any additional arguments specified while running the software are also passed to the code. `booter=`*%FILE%*`/boot-loader;arg={dos=`*%FILE%*`floppy-144MB.img;%` tells the system to use “boot-loader” as the boot loader instead of the OpBoot loader. Everything within the "arg={...}" is passed to the boot loader through a DHCP option-135 as a string. For configuration software or control software, enter the command to execute a script or program on the target server(s). You must enter the path of the executable relative to the current directory. The current directory is set to the directory where the uploaded `.zip` file was unzipped. The syntax for Linux is: *./myscript.sh/arguments* or on Windows, *myscript.bat arguments*. On Linux and UNIX systems, the command line is executed by exec(). On Windows systems, the command line is executed by CreateProcess(). Therefore, a command line with multiple commands must be grouped and launched by a shell. ◆ Linux: **`/bin/sh -c "./myscript.sh; echo foo >> /tmp/log.txt"`** ◆ Windows: **`"myscript.bat & echo foo >> c:\log.txt"`** In both examples, the OS-native script is executed, followed by a command separator, ending with a text file written to the target's disk.
Licenses	Select the license(s) you want to add or remove, and click the **Add** button to add, or the **Remove** button to remove the selected licenses.

Add/Edit Software: Basic Section

Field Name	Description
Architecture *	Select the processor architecture of the server on which this software will be executed.
OS *	Select the OS that the server executing this software is using. This selection is necessary for configuration software and control software. Bootable software does not require a running operating system.
Command Line *	Enter the commands to be run when software is executed. All command lines support the following macros: • *%FILE%* is the root directory of the `.zip` file. • *%ARGS%* are additional arguments. For bootable software, the command line specifies the bootable file within the `.zip` package, and how to load the software. By default, the OpBoot loader is used to load bootable software. For example: `dos=`*%FILE%*`/floppy-144MB.img;`*%ARGS%* tells the OpBoot loader to load an MS-DOS floppy image. The image is named floppy-144MB.img in the root directory of the `.zip` file. Any additional arguments specified while running the software are also passed to the code. `booter=`*%FILE%*`/boot-loader;arg={dos=`*%FILE%*`floppy-144MB.img;%` tells the system to use “boot-loader” as the boot loader instead of the OpBoot loader. Everything within the "arg={...}" is passed to the boot loader through a DHCP option-135 as a string. For configuration software or control software, enter the command to execute a script or program on the target server(s). You must enter the path of the executable relative to the current directory. The current directory is set to the directory where the uploaded `.zip` file was unzipped. The syntax for Linux is: *./myscript.sh / arguments* or on Windows, *myscript.bat arguments*. On Linux and UNIX systems, the command line is executed by exec(). On Windows systems, the command line is executed by CreateProcess(). Therefore, a command line with multiple commands must be grouped and launched by a shell. • Linux: **`/bin/sh -c "./myscript.sh; echo foo >> /tmp/log.txt"`** • Windows: **`"myscript.bat & echo foo >> c:\log.txt"`** In both examples, the OS-native script is executed, followed by a command separator, ending with a text file written to the target's disk.
Licenses	Select the license(s) you want to add or remove, and click the **Add** button to add, or the **Remove** button to remove the selected licenses.

Add/Edit Software: Basic Section

Field Name	Description
Architecture *	Select the processor architecture of the server on which this software will be executed.
OS *	Select the OS that the server executing this software is using. This selection is necessary for configuration software and control software. Bootable software does not require a running operating system.
Command Line *	Enter the commands to be run when software is executed. All command lines support the following macros: ◆ *%FILE%* is the root directory of the `.zip` file. ◆ *%ARGS%* are additional arguments. For bootable software, the command line specifies the bootable file within the `.zip` package, and how to load the software. By default, the OpBoot loader is used to load bootable software. For example: `dos=`*%FILE%*`/floppy-144MB.img;`*%ARGS%* tells the OpBoot loader to load an MS-DOS floppy image. The image is named floppy-144MB.img in the root directory of the `.zip` file. Any additional arguments specified while running the software are also passed to the code. `booter=`*%FILE%*`/boot-loader;arg={dos=`*%FILE%*`floppy-144MB.img;%` tells the system to use “boot-loader” as the boot loader instead of the OpBoot loader. Everything within the "arg={...}" is passed to the boot loader through a DHCP option-135 as a string. For configuration software or control software, enter the command to execute a script or program on the target server(s). You must enter the path of the executable relative to the current directory. The current directory is set to the directory where the uploaded `.zip` file was unzipped. The syntax for Linux is: *./myscript.sh/arguments* or on Windows, *myscript.bat arguments*. On Linux and UNIX systems, the command line is executed by exec(). On Windows systems, the command line is executed by CreateProcess(). Therefore, a command line with multiple commands must be grouped and launched by a shell. ◆ Linux: **`/bin/sh -c "./myscript.sh; echo foo >> /tmp/log.txt"`** ◆ Windows: **`"myscript.bat & echo foo >> c:\log.txt"`** In both examples, the OS-native script is executed, followed by a command separator, ending with a text file written to the target's disk.
Licenses	Select the license(s) you want to add or remove, and click the **Add** button to add, or the **Remove** button to remove the selected licenses.

Add/Edit Software: Basic Section

Field Name	Description
Architecture *	Select the processor architecture of the server on which this software will be executed.
OS *	Select the OS that the server executing this software is using. This selection is necessary for configuration software and control software. Bootable software does not require a running operating system.
Command Line *	Enter the commands to be run when software is executed. All command lines support the following macros: ◆ *%FILE%* is the root directory of the `.zip` file. ◆ *%ARGS%* are additional arguments. For bootable software, the command line specifies the bootable file within the `.zip` package, and how to load the software. By default, the OpBoot loader is used to load bootable software. For example: `dos=`*%FILE%*`/floppy-144MB.img;`*%ARGS%* tells the OpBoot loader to load an MS-DOS floppy image. The image is named floppy-144MB.img in the root directory of the `.zip` file. Any additional arguments specified while running the software are also passed to the code. `booter=`*%FILE%*`/boot-loader;arg={dos=`*%FILE%*`floppy-144MB.img;%` tells the system to use “boot-loader” as the boot loader instead of the OpBoot loader. Everything within the "arg={...}" is passed to the boot loader through a DHCP option-135 as a string. For configuration software or control software, enter the command to execute a script or program on the target server(s). You must enter the path of the executable relative to the current directory. The current directory is set to the directory where the uploaded `.zip` file was unzipped. The syntax for Linux is: *./myscript.sh / arguments* or on Windows, *myscript.bat arguments*. On Linux and UNIX systems, the command line is executed by exec(). On Windows systems, the command line is executed by CreateProcess(). Therefore, a command line with multiple commands must be grouped and launched by a shell. ◆ Linux: **`/bin/sh -c "./myscript.sh; echo foo >> /tmp/log.txt"`** ◆ Windows: **`"myscript.bat & echo foo >> c:\log.txt"`** In both examples, the OS-native script is executed, followed by a command separator, ending with a text file written to the target's disk.
Licenses	Select the license(s) you want to add or remove, and click the **Add** button to add, or the **Remove** button to remove the selected licenses.

Add/Edit Software: Basic Section

Field Name	Description
Storage *	Select the system storage where the software package is located.
Return Values	Enter the return values that indicate successful execution of the software. On Linux the return value variable is *$?*. On Windows the return value variable is *%errorlevel%*. The format for return values is: *num*[..*num*][,*num*[..*num*]]. For example, if you enter **0** it means that 0 (and only 0) indicates success. If you enter return values as **1..5,101..105** this indicates that 1 through 5 and 101 through 105 represent successful return values. If this field is left blank, *any* return value is considered successful.
Causes Reboot	Check this check box if the software you are adding reboots the target system during the software installation process.
Proxy Data	Check this check box if you want the Intel Deployment Manager server to copy the desired software from system storage and send it to the target machine for execution—this is necessary if the target machine cannot communicate with the system storage. If unchecked, the target machine attempts to communicate directly with the system storage and get the desired software.
Assign to Workspace	Select a workspace.
Comments	Enter any comments for this software.

6. Click the **Constraints** tab if you want to edit the constraints for this software.

7. Click **OK** to add the software.

Deleting Software

▼ To delete software

1. In the Navigation Area, click **Resources** > **Software**.
2. Click a tab: **Applications**, **Utilities**, **Scripts**, **Patches**, **Others**.
3. Select the software that you want to delete.
4. Click **Actions** > **Delete**.
5. Click **Yes** to delete the selected software.

Importing Software

Only software added by Intel Deployment Manager can be imported. If you exported software to a different system storage location you can import it back to Intel Deployment Manager (see “Exporting Software” on page 213). You can also import software from a different Intel Deployment Manager server.

▼ To import software

1. In the Navigation Area, click **Resources** > **Software**.
2. Click a tab: **Applications**, **Utilities**, **Scripts**, **Patches**, **Others**.
3. Click **Actions** > **Import**. The **Import Software** dialog is displayed.
4. Enter a name for the software that you want to import. This may be the existing name, or you can specify a new name as long as it is unique. Maximum 32 characters.
5. Optionally, enter a description of this software.
6. Select system storage to store the imported software.
7. Select a workspace.
8. Click **Continue**.
9. In the popup **Import Software** dialog, browse to select the file to be imported. Only a zipped file (`.zip` or `.epk`) can be imported.

10. Click **OK**.

Exporting Software

You can export software that you have added to Intel Deployment Manager from its system storage location to a different storage location. When software is exported, Intel Deployment Manager repackages the software as a <*packagename*>.epk file. After exporting the software, you can import the software to any Intel Deployment Manager server, at any time.

▼ To export software

1. In the Navigation Area, Click **Resources** > **Software**.
2. Click a tab: **Applications**, **Utilities**, **Scripts**, **Patches**, **Others**.
3. Select the software that you want to export.
4. Select **Actions** > **Export**.
5. The **File Download** dialog is displayed. You can save the file on your machine or open it from its current location. All software files are in the zipped (.zip or .epk) format.

Using Search Storage

You can use the **Search Storage** action if you want to search for a pre-installed package (snapshot, software, or OpScript) on a system storage device. Each subdirectory in the system storage is checked. If files for a snapshot, software, or OpScript are found on the filesystem, but no database entry exists, Intel Deployment Manager creates the database entries and associates them with the files.

Using Search Storage you can quickly import packages from a previous installation of Intel Deployment Manager, or from a set of files copied over from a different Intel Deployment Manager server. Note that sharing of package files between multiple Intel Deployment Manager servers is not supported.

▼ **To search storage**

1. In the Navigation Area, click **Resources** > **Software.**
2. Click a tab. All of the tabs under **Software** support **Search Storage**.
3. Click **Actions** > **Search Storage**. The **Search Storage** dialog is displayed.
4. Select a storage device to search.
5. Click **OK**. A list of all software applications is displayed.

Reporting all Software

You can view all of the software on your Intel Deployment Manager server from the **Reports** section by clicking **Resources** > **Reports** > **All Software**.

Deploying Software on Computing Devices

Deploying software is the process of putting a software package on a target computing device. The software package can be used for tasks like updating firmware, configuring Hardware RAID devices, or installing applications on a server.

Before deploying software you must create a zipped software package, and add the software package (see "Adding/Editing Software" on page 206).

Caution By default, the software package deploys to a subdirectory under the directory where the ITAP agent is installed. Verify that there is sufficient disk space available before deploying software. For provisioned servers, the Linux ITAP agent is automatically installed in `/.dd`.

▼ **To deploy software to a computing device**

1. In the Navigation Area, click **Resources** > **Computing Devices**.
2. Click a tab. You can deploy software from the **Accepted**, **Blade Assets**, or **Server Assets** tab.
3. Select the computing device to which you want to deploy the software package.
4. Click **Software** > **Deploy**.

5. In the **Deploy Software** dialog, enter the following information:

Deploy Software Fields

Field Name	Description
Assets	The computing device(s) onto which you want to deploy software are displayed.
Software *	Click the magnifying glass icon to display a list of available software packages stored on the Intel Deployment Manager server. Only the software packages loaded in Intel Deployment Manager are displayed (what you see depends on your user permissions).
Additional Arguments	Add any additional arguments you want to pass to the software. If *%ARGS%* was specified in the **Command Line** field when you added or edited the software package, you can enter space-separated values for any additional arguments you want to pass to the software.
Do not monitor progress	If you check this check box, the progress of this software is not monitored. Instead, the software is started and the system immediately returns.
Post Action	Select an action to be performed after receiving a return value. If the return value indicates that software execution was successful the post action is performed. Currently, there is only one post action, **Re-discover Asset Attributes**. This action updates the information for the asset.

6. Click **Deploy**.

Deploying Software on Servers

Deploying software is the process of putting a software package on a target computing device. The software package can be used for tasks like updating firmware, configuring RAID devices, or installing applications on a server.

Before deploying software you must create a zipped software package, and add the software package (see "Adding/Editing Software" on page 206).

Caution By default, the software package deploys to a subdirectory under the directory where the ITAP agent is installed. Verify that there is sufficient disk space available before deploying software. For provisioned servers, the Linux ITAP agent is automatically installed in `/.dd`.

▼ To deploy software on a server

1. In the Navigation Area, click **Resources** > **Provisioned Resources**.
2. Click the **Servers** tab (if it is not already open).
3. Select the server on which you want to deploy software.
4. Click **Software** > **Deploy**.
5. In the **Deploy Software on Servers** dialog, enter the following:

Deploy Software Fields

Field Name	Description
Hostname	Displays the host name of the server to which you want to deploy software.
Software *	Click the **Select a software** icon to view the available software, and select the software you want to deploy.
Additional Arguments	You can add any additional arguments you want to pass to the software. If *%ARGS%* was specified in the **Command Line** field when you added or edited the software package, you can enter space-separated values for any additional arguments you want to pass to the software.
Do not Monitor Progress	If you check this check box, the progress of this software is not monitored.
Post Action	Select an action to be performed after receiving a return value. If the return value indicates that software execution was successful the post action is performed. Currently, there is only one post action; **Re-discover Asset Attributes**. This action updates the information for the asset.

6. Click **Deploy**.

Using Deploy and Save

Use this option to deploy a software package and then save a snapshot immediately following the deployment.

▼ To deploy software and then save a snapshot

1. In the Navigation Area, click **Resources** > **Provisioned Resources**.
2. Select the asset on which you want to deploy software and save a snapshot.
3. Click the **Software** tab.
4. Select **Deploy and Save**. The **Deploy and Save** dialog displays the following sections:
   - **Basic**
   - **Constraints**
   - **Advanced**
5. In the **Basic** section enter the following information:

Deploy and Save: Basic Section

Field Name	Description
Hostname	Displays the host name of the server onto which you are going to deploy the software package, and from which you will save a snapshot.
Pre-save Software	Click the **Pre-save Software** icon. Pre-save software is executed on the server before the `Save Server Snapshot` operation takes place. `Sysprep` is an example of pre-save software on a Windows system. Intel Deployment Manager runs the pre-save software and transfers the files to the server asset. The arguments can be defined in such a way that Sysprep is executed at a specified time.
Pre-save Software Arguments	Enter the arguments you want this software to use when executed on the server. If you specify arguments for this software, the **Command Line** field of this software entity must contain the variable string *%ARGS%* at its end. This string is replaced with the pre-save software arguments when the software is executed.

Deploy and Save: Basic Section

Field Name	Description
Hostname	Displays the host name of the server onto which you are going to deploy the software package, and from which you will save a snapshot.
Incremental Snapshot	This flag displays as a check box if you are trying to save a snapshot for a server in the UP state. You see **No** in this field if you are trying to save a snapshot for a server in the DOWN state.
New Snapshot Name	By default, the **Default Save Snapshot Name** (in the **Provision Managed Servers** > **Management Options** section) that you entered when provisioning this resource is displayed.
OS *	The OS is displayed. The OS field represents the category or family of an operating system. For example, Red Hat Linux 7.x, or Windows.
Licenses	Select any licenses that apply. When the snapshot is loaded onto a server and the status of the server is UP, the selected license(s) are in use.
Storage	Select system storage to store this snapshot.
Comment	Enter any comments regarding this snapshot.

6. In the **Constraints** section enter the parameter, operator, and value. You can enter the constraints that you want applied to the selected server. An error is displayed if incompatible constraints are assigned. See “Comparing Constraints” on page 261.

7. Enter the following information in the **Advanced** section:

Deploy and Save: Advanced Section

Field Name	Description
Custom Save Reboot Software	You can specify that you want to reboot the server with custom software when saving. Any software which had the **Causes Reboot** option checked qualifies for this selection. If the software you select cannot cause reboot then the operation fails. Software is assigned to the **Causes Reboot** functionality in the **Add Software** and / or **Edit Software** dialogs.
Save Reboot Software Arguments	Enter additional arguments to pass to the custom reboot software.
Sector by Sector Save	Check this option to save this snapshot using raw sector blocks. With this option the snapshot saves each sector of the server’s hard disk.

Deploy and Save: Advanced Section

Field Name	Description
Filesystem Check *	**Filesystem Check**—This option enables disks to be scanned for errors by running `fsck` on disk partitions before saving the snapshot. Select one of the following options. ◆ **None**—(default) Do not check and repair the filesystem. An error is returned if a filesystem check was needed. ◆ **Automatic**—Run a filesystem check and repair errors only if the filesystem was not unmounted cleanly. ◆ **Force**—Always run a filesystem check, and repair errors if necessary.
Ignore FS Check Errors	If you check this check box, any errors detected during a file system check are ignored, and the partition with errors is *not* saved. This option is unchecked by default.
Compression of Snapshot	Check (default) this check box to compress the snapshot as it is saved. Compressing a snapshot takes additional time, but saves space by reducing the size of the snapshot.
Date Based Save	Select this option if you want to perform a date-based save. The date-based stamp is used to determine whether or not a file has changed. This option is only displayed if the **Incremental Snapshot** flag is set. Date-based saves work with Linux (EXT2/EXT3) file systems. NTFS file systems such as Windows 2000 or XP do not support this feature. ◆ When the flag is *not* selected, only the file contents are used to determine if anything changed. ◆ When the flag is selected, a quick check is made to determine whether or not the time stamp of a file has changed. If the time stamp has not changed, it is assumed that the file has not changed. If the time stamp has changed then the contents of the file are considered. Setting the date-based stamp flag results in a faster incremental backup, but with a slight risk that a backup could be incomplete because of an incorrect time stamp.

8. Click **Save and Reboot**.

Working with OpScripts and Jobs 18

Intel Deployment Manager supports server-side scripting using *OpScripts*. An OpScript enables you to select and run a sequence of Intel Deployment Manager commands.

Note Intel Deployment Manager comes with two canned scripts: start multiple servers, and stop multiple servers. You can modify these scripts or add new ones to perform any operation.

Jobs enable you to schedule an OpScript to run at specified times, or to be triggered by particular system or user events. You can view the execution status of jobs and OpScripts from the Job Monitor.

This chapter includes the following topics:

Working with OpScripts

OpScripts provide several benefits. OpScripts use the JAPI interface, so all commands can be performed from OpScripts. Native Java and BeanShell OpScripts fully leverage the power and robustness of the Java language and JDK.

Adding/Editing OpScripts

▼ To add or edit an OpScript

1. In the Navigation Area, click **Resources** > **`Jobs`**. By default, the **OpScripts** tab is opened, and the **Displaying All OpScripts** table is displayed.

2. To edit an OpScript, select the OpScript you want to edit.

3. Click **Actions** > **Add** or **Edit**. The **Add OpScript** or **Edit OpScript** dialog displays the following sections:
   - **Basic**
   - **Commands**

4. Enter the following information in the **Basic** section:

Add/Edit OpScript: Basic Section

Field Name	Description
Name *	Enter a name for the OpScript. For example, **`Server AssetPowerOn`**. Most alphanumeric and special characters are allowed, but avoid greater than (>), less than (<), ampersand (&), single quotes (‘), and double quotes (“).
Description	Enter a description for the OpScript. For example, **`used for Powering On a Server Asset`**.
Storage *	Select system storage for this OpScript. When editing, this is a display-only field.
Directory (displays only when editing)	Displays the directory location of the OpScript.

Add/Edit OpScript: Basic Section

Field Name	Description
Command Target	Select a command target. The target on which this OpScript should run is a: ◆ **Server** ◆ **Managed Computing Device**
Provision a New Server	Check this check box if you are adding an OpScript to provision servers (**Command Target** = **Server**). If you check this check box, the following additional fields appear: ◆ **Server template *** —Select a server template for use when provisioning. ◆ **Allocate Server Asset From Workspace** (displayed only if **Autoselected** is checked)—Select a workspace. Your selection identifies the workspace that server assets should be allocated from when a server is started up—server assets are allocated only from the selected workspace. If you select **Any workspace**, any server asset accessible by the user can be allocated, regardless of the workspace it is in. ◆ **Server Assigned to Workspace**—Select a workspace that the server should be assigned to after it is provisioned. ◆ **Default Provisioned Server Network ***—Select a network for the NIC specified in the server template to use as the default network. This field is displayed if you seleced **Default Network** in the **Network Name** field when you added or edited the server template. When the script is run an available server asset is auto-selected and the new server is provisioned.

Add/Edit OpScript: Basic Section

Field Name	Description
Command Sequence *	Click **Add** to select commands. Select the commands in the sequence in which they should be executed. You must enter a logical sequence for the commands to execute properly. If you select **Server** in the **Command Target** field, the following commands are displayed: ◆ **Reboot to Admin State**—See "Rebooting to Admin State" on page 113 for information about fields. ◆ **Reboot** —See "Rebooting" on page 112 for information about fields. ◆ **Deploy Software**—See "Deploying Software on Computing Devices" on page 214 or "Deploying Software on Servers" on page 215 for information about fields. ◆ **Save Snapshot**—See "Saving Snapshots for Computing Devices" on page 123 or "Saving Snapshots for Provisioned Resources" on page 126 for information about fields. ◆ **Delete**—Select whether to stop or continue execution on error. If you select **Managed Computing Device** in the **Command Target** field, the following commands are displayed: ◆ **Power On—**See **"Powering On**" on page 114 for information about this command. ◆ **Power Off—**See "Powering Off" on page 114for information about commands. ◆ **Power Cycle—**See "Power Cycling" on page 115for information about commands. ◆ **Provision Server—**See the chapter "Provisioning Servers" on page 163 for information about provisioning. ◆ **Load and Startup—**See "Loading a Snapshot and Starting Up a Server" on page 185 for information about fields. ◆ **Reboot to Admin State—**See "Rebooting to Admin State**" on page 113** for information about fields. ◆ **Reboot—**See "Rebooting**" on page 112** for information about fields. ◆ **Deploy Software—** See "Deploying Software on Computing Devices" on page 214 or "Deploying Software on Servers" on page 215 for information about fields. ◆ **Save Snapshot—**See "Saving Snapshots for Computing Devices" on page 123 or "Saving Snapshots for Provisioned Resources" on page 126 for information about fields. ◆ **Delete—**Select whether to stop or continue execution on error.

Add/Edit OpScript: Basic Section

Field Name	Description
Beanshell Code * (displays in Edit mode only)	Displays the Beanshell code for the selected commands.
OpScript Assigned to Workspace	Select the workspace to which you want to assign this OpScript.
Comments	Enter any relevant comments about the OpScript.
Created by	This field displays the user login of the user who is adding this OpScript.

5. The **Commands** section enables you to add arguments and specify actions to the OpScript. The fields displayed depend on the command that the OpScript is going to execute.

   For example, if you run an OpScript to deploy software, on execution, the OpScript asks you to enter a name for the software. The following fields are displayed when you select **Deploy Software**.The following fields are displayed when you choose **Deploy Software**.

Add/Edit OpScript: Commands Section

Field Name	Description
Choose Software	Click the magnifying glass icon to display a list of software. Select a software package, and click **Use selected item**.
Additional Arguments	Enter any additional arguments you want to pass to the software.
Do not monitor progress	Check this check box if you do not want to monitor progress when deploying the software.
On Error	You can stop or continue execution of the software if an error occurs. This field is displayed for all commands.
Wait for Completion	Check this option if you want Intel Deployment Manager to wait for this command to finish before starting the next command or exiting the OpScript. This field is displayed for all commands.

6. Click **OK**.

Deleting OpScripts

▼ To delete an OpScript

1. In the Navigation Area, click **Resources** > **Jobs**. By default, the **OpScripts** tab is opened, and the **Displaying All OpScripts** table is displayed.
2. Select the OpScript you want to delete.
3. Click **Actions** > **Delete**. The verification dialog is displayed.
4. Click **Yes** to delete the OpScript.

Running OpScripts

▼ To run an OpScript

1. In the Navigation Area, click **Resources** > **Jobs**. By default, the **OpScripts** tab is opened, and the **Displaying All OpScripts** table is displayed.
2. From the **Displaying All OpScripts** table, select the OpScript you want to run.
3. Click **Actions** > **Run**. The **Run OpScript** dialog displays the name, description, and language of the OpScript.
4. Select the server on which you want to run the OpScript.
5. Click **OK** to run the OpScript.

Creating and Scheduling Jobs

You can create a *job* to schedule an OpScript to run at particular times, or to be triggered by particular system or user events. The execution status of jobs and OpScripts can be monitored on the **Job Monitor** page.

Adding/Editing Jobs

▼ To add or edit a job

1. In the Navigation Area, click **Resources** > **Jobs**.
2. Click the **Jobs** tab.
3. If you want to edit a job, select it.
4. Click **Actions** > **Add** or **Edit**. The **Add Job** or **Edit Job** dialog displays the following sections:
   - **Basic**
   - **Parameters** (Displayed only for jobs with parameters).
   - **Recurrence**
5. Enter the following information in the **Basic** section:

Add/Edit Job: Basic Section

Field	Description
Script Name	Click the magnifying glass icon to display a list of available scripts. Select the OpScript you want, and then click **Use Selected Item**.
Job Name *	Enter a name for this job. Maximum 64 characters. Most alphanumeric and special characters are allowed, but avoid greater than (>), less than (<), ampersand (&), single quotes (‘), and double quotes (“).
Job Description	Enter a description of this job.

Add/Edit Job: Basic Section

Field	Description
Execution Criteria *	Select the execution criteria for this OpScript: • **On Scheduled Dates** • **On occurrence of listed event category** If you select **On occurrence of listed event category**, and select a **Wild Card Parameter,** an additional radio button labeled **The resource that triggererd the event** is displayed under **Wild Card Parameter Values**.
Start Date * (displays only if you select **On Scheduled Date** as the Execution Criteria)	Click the calendar icon to select the time and date this job should be executed.
Triggering event categories * (displays only if you select **On occurrence of listed event** category)	Click **Add** to select an event from the list of event categories. This job is executed when the selected event occurs.
Skip Time (displays only if you select **On Scheduled Date** as the **Execution Criteria**)	Enter a skip time in milliseconds. The skip time determines the maximum amount of time after the scheduled start time that a job can start. For example, if the job is scheduled to run at 12:00 and the skip time is set to `600000` (10 minutes), the job can start as late as 12:10, but not later. The default is 15 minutes.
Delete after Triggering	Check this check box if you want the job to be deleted from the database after it starts.
Priority	Select a priority for the job. Priorities range from 5 (low) to 11 (high), with 8 as the default (normal) priority. If multiple jobs are scheduled to run at the same time, the highest priority job runs first. If all jobs have the same priority, the order in which they are executed is undetermined; any one of them might run first. Changing priority has no effect on a job that is already running.

Add/Edit Job: Basic Section

Field	Description
Wild Card Parameter (displays only if you select an OpScript with one or more parameters)	The wild card parameter contains a list of all of the selected OpScript's parameters. If an OpScript parameter is chosen as the wildcard parameter, the value of that parameter is not filled in until the OpScript's job is triggered. The value for this parameter is determined by whether the job is time based or event based. If it is event based, the value of that parameter is set to the resource that generated an event. For example, if you created a job using an event with a parameter of type `server`, and if the job was set up to be triggered by a `Server is Up` event, when the job is triggered, the value of the wildcard parameter will be set to that server. For time-based jobs, the job will start *n* copies of the OpScript each copy of which will have its wildcard parameter set to one of the *n* resources that are the same type as the wildcard parameter and which are accessible by the user.
Wild Card Parameter Values (displays only if you select a **Wild Card Parameter**)	You can select a wild card parameter value in order to further constrain the selection of targets for your command script. Select one of the following: ◆ **The resource that triggererd the event**—Select this option if you want to target the resource that triggered the event that executed the job. If you select this option you must also select the workspace location of the resource; for example, **All Workspaces** or **Discovered**. If you have defined one or more active filters an additional options list is displayed from which you can select an active filter to further constrain your selection of targets. ◆ **All Accessible Resources**—Select this option if you want to target all resources, constrained by workspace and possibly active filter. If you select this option you must also select the workspace location of the resources; for example, **All Workspaces** or **Discovered**. If you have defined one or more active filters an additional options list is displayed from which you can select an active filter to further constrain your selection of targets. ◆ **Selected values**—If you select this option a selection box is displayed with a list of values for the selected wild card parameter. Use the **Add** and **Remove** buttons to add or remove values.
OpScript Assigned to Workspace	Select the workspace to which the OpScript should be assigned.
Comments	Enter any comments.
Created By	This field displays the user login of the user who added this job.

6. In the **Parameters** section, select the server on which you want to run this OpScript. You need to provide values for all of the OpScript parameters, except for the wildcard parameter. When the job is triggered, the parameters of the job's OpScript will be set to these values.

7. Enter the following information in the **Recurrence** section:

Add/Edit Job: Recurrence Section

Field Name	Description
Recurrence	Select an interval for running this job: **None**, **Daily**, **Weekly**, **Monthly**.
Range of Recurrence	Click the calendar icon, select a date, and click **OK** to set the start date. Repeat this procedure to set an end date. You can select **No end date** if this is a recurring job.

8. Click **OK**.

Using the Job Monitor

The **Job Monitor** page displays the status of all OpScripts that are running, or that have completed their run. Each instance of an OpScript that is running, or that ran previously, is called a *Task*. Each task displays the OpScript, and the status of that task. If the OpScript was started by a triggered job, the job is also displayed. If the OpScript failed, the error that caused the failure is displayed in the **Status** field.

▼ To display the Job Monitor

1. In the Navigation Area, click **Resources** > **Jobs**.

2. Click the **Job Monitor** tab.

Deleting Jobs

Only completed jobs can be deleted.

▼ **To delete a job**

1. In the Navigation Area, click **Resources** > **Jobs**.
2. Click the **Jobs** tab. The **Displaying All Jobs** table is displayed.
3. Select the job(s) that you want to delete.
4. Click **Actions** > **Delete**. The verification dialog is displayed.
5. Click **Yes** to delete the job.

Using the Task Manager

A *task* is a long-running operation performed by Intel Deployment Manager. It might be a running OpScript, or a server operation like startup, shutdown, save or deploy software. Intel Deployment Manager tasks are analogous to operating system processes.

The Task Manager provides a list of all of the tasks on the system and gives you information about the status of each task. You can use it to determine if a task is QUEUED, SLEEPING, RUNNING, COMPLETED, or COMPLETED WITH ERRORS. Whereas the Job Monitor gives you information only about jobs, the Task Manager provides information about all tasks, including jobs.

Use the Task Manager to change the priority of a task, to abort a running task, or to delete a completed task.

Aborting Tasks

The abort operation enables you to kill a running task.

▼ **To abort a task**

1. In the Navigation Area, click **Settings** > **Task Manager**.
2. Select one task.
3. Click **Action** > **Abort**. The verification dialog is displayed.
4. Click **Yes** to abort the task.

Changing Task Priority

▼ **To change the priority of a task**

1. In the Navigation Area, click **Settings** > **Task Manager**.
2. Select one task.
3. Click **Action** > **Change Priority**.
4. Enter a new priority in the **New Priority** field. Priorities range from 1 to 64, with 64 being the highest priority. You can only change a priority by +/- three points; for example, if the current priority is 23, you can raise it to 26, or lower it to 20.
5. Click **OK**.

Deleting Tasks

When a task completes, you can use the Task Manager to determine whether or not it completed successfully. Completed tasks remain in the table until you delete them. The delete operation enables you to delete a completed task from the **Displaying All Tasks** table. You can only delete completed tasks.

▼ **To delete a task**

1. In the Navigation Area, click **Settings** > **Task Manager**.
2. Select one task.
3. Click **Action** > **Delete**. The verification dialog is displayed.
4. Click **Yes** to delete the task.

Managing Events 19

All operations that change the state of resources generate events. Events are logged in the *Custom Log*.

This chapter includes the following topics:

Viewing the Custom Log

The Custom Log tracks all events. You can display all events for a specified range of dates.

▼ To view a list of events

1. In the Navigation Area, click **Settings** > **Custom Log**.
2. The **Custom Log** dialog is displayed. Enter the required fields:

Custom Log Fields

Field	Description
From Date	Select the day, month and year.
To Date	Select a day, month, and year.

3. Click **OK** when finished entering data. You might click **Export** if you want to export the data to a Microsoft Excel spreadsheet.

Deleting Custom Log Entries

Old Custom Log entries are *not* automatically deleted. You should delete old entries periodically to speed up searches and let DB maintenance reclaim unused space. You can delete any events accessible by your login user ID.

▼ To delete events

1. In the Navigation Area, click **Settings** > **Custom Log**.
2. Click **Delete**
3. In the **Delete Custom Log Entries** dialog, select a date. Entries before this date are to be deleted.
4. Click **Delete** to delete the entries.

Understanding Intel Deployment Manager Events

An Intel Deployment Manager *event* is a record that identifies an operation performed on a managed resource at a particular time. All operations that change the state of managed resources generate events, so events can be used to track changes. You can use events to track which users or workspaces are using resources, and you can use this information to generate billing information. Events are also generated for any errors that occur in Intel Deployment Manager.

Understanding Event Kinds

All events have an associated *kind*. The **Kind** field identifies the change of status that occurred to the object or operation identified by the **Type** field. The event *type*, describes the type of resource or entity that an event is associated with. Most **Kind** field values are one of the duration kinds—such events are known as duration events. A *duration event* indicates that some important change has occurred concerning the lifetime of a managed resource or entity.

Duration events can be of the kind: `Start`, `End`, or `Change`. `Start` events indicate the start of a resource or entity lifetime. `End` events indicate the end of the lifetime. `Change` events indicate that a live resource or entity has changed in some significant way.

In addition to duration events, you can also encounter events of kind `Activity`, `Activity Done`, `Point`, `Info`, and `Error`. The `Activity` event indicates that the user has just begun an operation. The `Activity Done` event indicates that a long-running user-initiated operation has completed. The `Point` event indicates that some important operation occurred at the given point in time, however, there is no duration or resource lifetime associated with that event. `Info` events provide informational messages and can usually be ignored. `Error` events are generated whenever any user command or long running operation generates an error: they can also be generated if server assets or other resources are currently in an invalid or inaccessible state.

The following table lists event kind fields:

Event Kind Fields

Field Name	Description
Time	The time the event occurred.
User ID	The user ID of the user that performed the operation recorded by this event. If this field is the empty string (“”), then the event was not performed by a user. That is, the Intel Deployment Manager system performed the operation.
Serial Number	The serial number of the server asset that this event is associated with. If this event cannot be associated with a server asset, or if that server asset does not have a serial number, this field is the empty string (“”).
Event ID	A numerical resource ID used to uniquely identify the operation or object that this event is based on. This resource ID is unique only for a given set of values for the field event-type. This field can be used to determine duration events associated with a particular resource to compute resource lifetimes for billing purposes.
Type	The type of event. The type identifies what type of resource or operation that this event is associated with. The possible types of events are described in the section “Understanding Event Types” on page 238.
Kind	The **Kind** field identifies the change of status that occurred to the object or operation identified by the **Type** field. See “Understanding Event Kinds” on page 235.
Message	A short, user-friendly message describing this event. If the event is an error, the error message is displayed here.
Attributes	A string of optional attributes associated with the event. The meaning of the attributes varies depending on the value of the event's **Type** field. The attributes string consists of a semicolon (;) separated list of *var=value* expressions, where *var* is a variable name and *value* is the value of the variable.
Resource ID	An ID that identifies the event’s resource. This resource ID is unique for a given owning workspace and event type. This ID is set to 0 if this is an user activity event.
Activity ID	An ID that identifies the user operation that created this event.

Event Kind Fields

Field Name	Description
Message Parameter #1	The first parameter in a list of parameters associated with this event's message. The meaning of these parameters depend upon the value of the **Message** field. The purpose of these parameters is to allow you to generate an internationalized message with detailed information, where the format of the message is determined from the value of the event-message field, and the parameters are instantiated from the values given in this parameter list.
Message Parameter #2	The second parameter in a list of parameters associated with the event's message. See the description for Message Parameter #1.
Message Parameter #3	The third parameter in a list of parameters associated with the event's message. See the description for Message Parameter #1.
Message Parameter #4	The fourth parameter in a list of parameters associated with the event's message. See the description for Message Parameter #1.
Message Parameter #5	The fifth parameter in a list of parameters associated with the event's message. See the description for Message Parameter #1.

Understanding Event Types

All events have an *event type*, which identifies the type of resource or operation that the event is associated with. Except where noted, all event types use only the duration event kinds, or the error event kind. The event type can take one of the following values.

Event Types

Event Type	Description
`User`	This event corresponds to the creation, modification, or deletion of an user object. The **Serial Number** fields are always empty. The user being created, deleted, or modified is specified as the value of the *userId* variable in the **Attributes** field. The **User** field of the event contains the user ID of the user performing the operation, *not* the user that the operation is being performed on.
`Workspace`	This event corresponds to adding, editing, or deleting a workspace
`Snapshot`	This event corresponds to the creation, modification, or deletion of a snapshot object. The **User** fields contain the name of the owner of the snapshot. The **Serial Number** field is always empty. The **Attributes** field of the snapshot contains the following variables: ◆ *type*—type of server asset that this snapshot runs on. ◆ *arch*—CPU architecture that this snapshot runs on. ◆ *baseOs*—operating system that this snapshot is based on. ◆ *isGateway*—this variable is deprecated and should always be set to false ◆ *isLoadable*—true if this snapshot can be loaded onto a server ◆ *scope*—scope in which this snapshot can be used ◆ *configName*—name of the snapshot ◆ *licenses*—semicolon separated list of licenses required by this snapshot
`Server`	This event corresponds to the creation, modification, or deletion of a server object. The **Serial Number** field is always empty. The **Attributes** field of the server equals the **Attributes** field of the snapshot associated with the server. The **Attributes** field is empty if the server has no associated snapshot.

Event Types

Event Type	Description
`ServerUse`	This event corresponds to a change of status for a server object, typically due to a start-up, shut-down, reboot, or save command. The lifetime of a `ServerUse` duration begins when a server is changed from a DOWN state to a non-down state, and the lifetime ends when a server is changed from a non-down state to a DOWN state. The **Serial Number** field contains the serial number of the asset that is assigned to the server. The **Attributes** field contains all variables and values of the snapshot associated with the server. In addition to the snapshot variables, the **Attributes** field contains the following variables: ◆ *serverId*—ID of the server whose state was changed. This ID equals the **Event-id** field of the corresponding server event(s). ◆ *hostName*—Host name of the server. ◆ *status*—Status of the server. ◆ *imageLoad*—If the server is being started up, this variable indicates the load command for server's snapshot. Otherwise, this variable is not displayed. ◆ *assetId*—ID number of the server asset assigned to this server. If this server does not have a server asset, this variable is not displayed
`Network`	This event corresponds to the creation, modification, or deletion of a network. The **Serial Number** field is always empty. The **Attributes** field of the network contains the following variables: ◆ *startIp*—Start IP address of the network. ◆ *endIp*—End IP address of the network. ◆ *mask*—IP address mask of the network.
`Storage Device`	This event corresponds to the creation, modification, or deletion of a storage device. The **Serial Number** fields are always blank. The **Attributes** field contains the following variables: ◆ *location*—The name of the storage device's driver. ◆ *slocitem*—A string key uniquely identifying the storage device.
`Session`	This event corresponds to a user logging in or out of the Intel Deployment Manager system. The **Serial Number** fields are always empty. The **Attributes** field is always empty.
`Terminal`	This event corresponds to the opening of a remote terminal on one of the servers managed by Intel Deployment Manager. All `terminal` events are of the kind `Point`. The **Serial Number** field contains the serial number of the server's assigned server asset. The **Attributes** field of this event equals the contents of the **Attributes** field for the `ServerUse` event for this server.

Event Types

Event Type	Description
`Server Asset`	This event corresponds to the creation, modification, or deletion of a server asset. The **Serial Number** field equals the server asset's serial number. The **ID** field equals the server asset's ID. The **Attributes** field contains the following variables: ◆ *arch*—The architecture of the server asset. ◆ *serverId*—ID of the server associated with the server asset being updated. This ID equals the **Event Id** field of the corresponding server event(s). If the server asset is not assigned to a server, this variable is not present. ◆ *hostName*—Host name of the server associated with the server asset being updated. If the server asset is not assigned to a server, this variable is not present.
`Power`	This event corresponds to a remote power-on, power-off, or power-cycle command. All `Power` events are of kind `Point`. The **Serial Number** field equals the server asset's serial number. The **ID** field equals the ID of the asset being powered on / off / cycled. The **Attributes** field contains the following variables: ◆ *assetId*—ID of server asset being powered on / off / cycled. ◆ *command*—Identifies the power command: `on` for power-on, `off` for power-off, or *cycle* for power-cycle. ◆ *serverId*—ID of the server associated with the server asset being powered on / off / cycled. This ID equals the **Event ID** field of the corresponding server event(s). If the server asset is not assigned to a server, this variable is not present. ◆ *hostName*—Host name of the server associated with the server asset being powered on / off / cycled. If the server asset is not assigned to a server, this variable is not present.
`Delete Events`	This event corresponds to the deletion of a set of events from the system. The **Serial Number** fields are always empty. The **Atributes** field contains the following variables: ◆ *time*—The time up to which events were deleted. This time is a numerical value representing the number of milliseconds since midnight January 1, 1970. ◆ *timeStr*—The time to which events were deleted. Unlike the *time* variable, however, the value of this field is printed out in a user-readable form.
`System Storage`	This event corresponds to the creation, deletion or modification of system storage. The **Serial Number** fields are always blank. The **ID** field equals the ID of the system storage. The **Attributes** field contains the following variables: ◆ *type*—The type of system storage. ◆ *location*—The storage device location for the system storage. ◆ *size*—The allocated size of the system storage.

Event Types

Event Type	Description
Software	This event corresponds to the creation, deletion or modification of software entities. The **Serial Number** fields are always blank. The **Attributes** field contains the following variables: ◆ *name*—The name of the software. ◆ *type*—The software's type.
Software Use	This event indicates that someone is executing a software entity object on a server asset object. The **Serial Number** field equals the server asset's serial number. The **Attributes** field contains the following variables: ◆ *arch*—The architecture of the server asset. ◆ *codeid*—The ID number of the software being run. ◆ *name*—The name of the software being run. ◆ *type*—The type of the software being run. ◆ *returnCode*—If the software run has completed, this is set to the return code for that software. Otherwise, this variable is not present.
None	This event is not associated with a resource. This type is typically used for error events. The **Serial Number**, and **Attributes** fields are always blank.
Licenses	This event corresponds to the creation, modification, or deletion of a license entity. The **Serial Number** fields are always empty. The **Atributes** field of the network contains the following variables: ◆ *name*—Name of the license entity. ◆ *type*—Type of the license entity. ◆ *maxNumLicenses*—Maximum number of licenses allocatable for this license entity. ◆ *usableOnAnyServer*—True if licenses from this license entity can be allocated using any server asset in the system.
Activity	This event corresponds to the initiation of a user operation (activity) on the system, where this user operation can change the state of elements within this system. If the activity has an associated server asset, the **Serial Number** field contains the serial number of the asset. The **Attributes** field is always blank.
IP List	This event corresponds to the creation, modification, or deletion of an IP list. The **Serial Number** field is always empty. The **Atributes** field contains the following variables: ◆ *name*: Name of the IP list ◆ *startIP*: Start IP address of the IP list ◆ *endIP*: End IP address of the IP list ◆ *provisioningNetwork*: Network that this IP list is allocated from

Event Types

Event Type	Description
`Unmanaged Server`	This event corresponds to the creation, modification, or deletion of a unmanaged server object. The **Serial Number** field contains the serial number of the unmanaged server asset associated with this unmanaged server. The **Atributes** field contains the following variables: ◆ *hostName*: Host name of the unmanaged server ◆ *assetId*: ID of unmanaged server asset associated with the unmanaged server
`Unmanaged Server Asset`	This event corresponds to the creation, modification, or deletion of a unmanaged server object. The **Serial Number** field contains the serial number of the unmanaged server asset. The **Atributes** field contains the following variables: ◆ *serverId*: ID number of the unmanaged server associated with this unmanaged server asset. If this unmanaged server asset is not assigned to a unmanaged server, this variable is not provided. ◆ *hostName*: Host name of the unmanaged server associated with this unmanaged server asset. If this unmanaged server asset is not assigned to a unmanaged server, this variable is not provided.
`Chassis`	This event corresponds to the creation, modification, or deletion of a chassis server asset object. The **Serial Number** field contains the serial number of the chassis.The **Atributes** field contains the following variables: ◆ *manip*: IP address used by Intel Deployment Manager to communicate with this chassis ◆ *manport*: Port used by Intel Deployment Manager to communicate with this chassis
`NIC`	This event corresponds to the creation, modification, or deletion of a server NIC. The **Serial Number** field contains the serial number of the server asset associated with this NIC's server, if the NIC's server has an assigned server asset. The **Atributes** field contains the following variables: ◆ *serverId*: ID of the server that this NIC is attached to ◆ *hostName*: Host name of the server that this NIC is attached to ◆ *nicId*: ID of this NIC for the NIC's server assetId: ID of the NIC's server's assigned server asset. If there is no such asset, this variable is not provided. ◆ *subnetName*: Name of the network or IP list to which this NIC is attached. ◆ *ipAddr*: IP address of the NIC. If the NIC has no IP address, this variable is not provided.

Event Types

Event Type	Description
Package	This event corresponds to the creation, modification, or deletion of a storage package. Storage packages are data structures used by Intel Deployment Manager to represent the set of files and directories used by snapshots, software, or OpScripts. The **Serial Number** field is always blank. The **Atributes** field contains the following values: ◆ *managedStorageId*: ID of the system storage that holds this package ◆ *location*: Storage driver used to manage the package's storage ◆ *resourceDesc*: Location of the package's storage directory.
System	This event reports some important change to Intel Deployment Manager itself, such as the startup of Intel Deployment Manager. The **Serial Number** field is always blank. The **Atributes** field is always blank.
Alarm	This event is generated whenever Intel Deployment Manager attempts to trigger one of its time-base jobs. The **Serial Number** and **Atributes** fields are always empty.
OpScript	This event corresponds to the creation, modification, or deletion of an OpScript. The **Serial Number** field is always blank. The **Atributes** field contains the following values: ◆ *language*: Language OpScript is written in: 1=Java, 2=BeanShell ◆ *packageId*: ID of storage package for OpScript ◆ *parameters*: Comma-separated list of the OpScript's parameter names ◆ *scriptFile*: Name of class for a native OpScript.
User Role	This event corresponds to the creation, modification, or deletion of a user role. The **Serial Number** and **Atributes** field are always blank.
Job	This event corresponds to the creation, modification, or deletion of a job. The **Serial Number** field is always blank. The **Atributes** field contains the following values: ◆ *eventMessageIds*: Comma separated list of the event IDs that trigger this event ◆ *wilcardParamName*: Name of the job's wildcard parameter. This variable is not provided if the job has no wildcard parameter
Network element	This event corresponds to the creation, modification, or deletion of a network element. The **Serial Number** field contains the serial number of the network element. The **Atributes** field contains the following values: ◆ *manip*: IP address used by Intel Deployment Manager to communicate with this network element ◆ *manport*: Port used by Intel Deployment Manager to communicate with this network element ◆ *isswitch*: Equals 'y' if the network element is a switch

Event Types

Event Type	Description
`OpScript Output`	This event contains one line of output from a running OpScript. If the output was written to standard error, the event kind is `error`. Otherwise, the event kind is `info`. The **Serial Number** and **Atributes** fields are always blank.
`Access Control`	This event corresponds to the creation, modification, or deletion of an access control. The **Serial Number** field is always blank. The **Atributes** field contains the following values: ◆ *resPerms*: An integer representing a bit vector of the resource permissions for this access control ◆ *opPerms*: An integer representing a bit vector of the operation permissions for this access control
`User Event`	This event corresponds the a user-defined event created by some external process. The contents of this event are determined by this external process.
Intel Deployment Manager `License Key`	This event corresponds to the creation or deletion of an Intel Deployment Manager product key. The **Serial Number** and **Atributes** fields are always blank.

Generating Lifetimes from Events

For billing generation and other purposes, you might want to recreate resource lifetimes from their associated events. All events associated with a given lifetime are guaranteed to have the same **Type** and **Event-id** fields. However, fetching all events with the same type, and event-ids is not sufficient, since events associated with several non-overlapping lifetimes are returned. This is because the system can reuse the event-id when a lifetime is finished. Therefore, when generating lifetimes from events, you must also carefully examine the **Time** and **Kind** fields of events to avoid mistaking several non-overlapping lifetimes as one long lifetime.

Server and Server Asset States A

All servers and server assets have states assigned to them to identify their status.

This appendix includes the following topics:

Understanding Server States

Every server has **Asset Status** (hardware-oriented) and **Server Status** (software oriented) fields that display server states. You can click on the state shown in a status field to pop up a window that contains additional information about the state.

If either the **Asset Status** or **Server Status** field is not displayed, you can display it by moving it to a displayed column as shown in "Setting Table Options" on page 16.

The server states are categorized as follows:

- Intermediate steps during the Startup-Server request.
- Intermediate steps in the Shutdown-Server, Reboot-Server, Save-Config requests. Steps are listed in the order in which they are performed.
- Server State Errors. By convention, these names start with `Failed`.

The following table lists the server states.

Server States

State ID	Server Status	Description
0	OK	The server is in a valid state. The system has no information on the status of the previous `startup`, `shutdown`, `reboot`, or `save`. Newly created servers start off with the OK state.
Intermediate Steps during the Startup-Server Request		
2001	BEGINNING START UP	The server has just begin a startup command. This value only occurs for servers with a STARTINGUP status.
2002	GETTING LICENSE	The server is reserving licenses required by its snapshot. This value only occurs for servers with a STARTINGUP status. If the server has a dedicated server asset, it will also wait in this state until it can contact the default controller OS of this server asset.
2003	RESERVING SERVER	The server is allocating a free server asset for its use. This value only occurs for servers with a STARTINGUP status.
2004	Mounting hard disks	The system is mounting disks on the server in preparation of the load of the server's Snapshot. This value only occurs for servers with a STARTINGUP status.

Server States

State ID	Server Status	Description
2005	PREPARING TO LOAD SNAPSHOT	The system is waiting for the server to initiate the load of the server's snapshot. This value only occurs for servers with a STARTINGUP status.
2006	LOADING SNAPSHOT	The system is loading the server's snapshot onto the server's hard disks. The progress of this load will be stored in the **status-info** field as a percentage. The progress of this load will be displayed in the progress bar of the server's status window as a percentage. The throughput and the percentage complete for the current partition will also be displayed.
2007	BOOTING	The system is booting the server. This value only occurs for servers with a STARTINGUP status.
Intermediate Steps in the Shutdown-Server, Reboot-Server, Save-Config Requests		
2020	BEGINNING SHUT DOWN	The system has just begun a `shutdown`, `reboot`, or `save` command on the server.
2021	PREPARING TO SAVE SNAPSHOT	The system is preparing the server to save a snapshot for a `save` or `shutdown` command.
2022	PREPARING TO SHUT DOWN	The system is preparing the server to perform a `shutdown`.
2023	SAVING SNAPSHOT	The system is saving the server's snapshot. The progress of this save will be stored in the **status-info** field as a percentage. The progress of this save will be displayed in the progress bar of the server's status window as a percentage. The throughput and the percentage complete for the current partition will also be displayed.
2024	REBOOTING	The system is rebooting the server back into the UP state.
2099	DONE	The system has successfully completed the last `startup`, `shutdown`, `reboot`, or `save` operation for this server.
Intermediate Steps in the Execute Software Request		
2040	RUNNING SOFTWARE	This server is running software
2041	UPLOADING SOFTWARE	This server is uploading a software package.

Server States

State ID	Server Status	Description
2098	SOFTWARE DOWN	The system has successfully finished running software.
Servers State Errors		
2100	FAILED: UNKNOWN ERROR	The last command performed on this server failed for some unknown reason. This can be treated as a variant of an internal error. The message of the error will be displayed via the `more details` link in the status window
2101	FAILED: INTERNAL ERROR	The last command performed on this server failed due to an internal error. Such errors might be due to bugs in the Intel Deployment Manager software. The message of the error will be displayed via the `more details` link in the status window.
2102	FAILED: NO AVAILABLE SERVER ASSETS	The last startup command performed on this server failed because there are no more available server assets.
2103	FAILED: NETWORK ERROR	The last command performed on this server failed due to a communication error. Such errors are usually due to a hardware failure of one of Intel Deployment Manager's servers.
2104	FAILED: NO AVAILABLE LICENSES	The last start up command performed on this server failed due a lack of available licenses required by the server's snapshot.
2105	FAILED: TOO MANY SERVERS ARE UP	The user's account has too many servers in the UP state to allow any more servers to be started up.
2106	FAILED: COULDN'T START SNAPSHOT LOAD	The last start-up command performed on this server failed because the system could not contact the server to initiate the load of the snapshot.
2107	FAILED: BOOT FAILED	The last command performed on this server failed because the system could not contact the server to initiate a `boot` or `reboot` operation.
2108	FAILED: COULDN'T CONTACT SERVER AFTER BOOT	The last command performed on this server timed out while waiting for the server to `boot` or `reboot`. This usually indicates a failure in the setup of the server's snapshot.

Server States

State ID	Server Status	Description
2109	FAILED: SNAPSHOT ALREADY EXISTS	The last `save` or `shutdown` command performed on this server failed because it attempted to save a snapshot with the same name as an existing snapshot.
2110	FAILED: COULDN'T START SNAPSHOT SAVE	The last `save` or `shutdown` command performed on this server failed because the system could not contact the server to initiate the save of the snapshot.
2111	FAILED: IP ADDRESS IS RESERVED	The last start-up command performed on this server failed because this server has an IP address that is being used by another active server in the same managed network.
2112	FAILED: NO AVAILABLE IP ADDRESSES	The last start-up command performed on this server failed because there are no more available IP addresses to assign to this server.
2113	FAILED: ILLEGAL IP ADDRESS	The last start-up command performed on this server failed because it had an illegal IP address.
2114	FAILED: SYSTEM IS TOO BUSY	The last command performed on this server failed because the system is too busy to initiate another command.
2115	SCRUBBED	The last command performed on this server was aborted by the Intel Deployment Manager system's consistency checker.
2116	FAILED: SERVER ASSET IS NOT POWERED ON	The last command performed on this server failed since the server asset for this server is not in a POWERED ON state.
2117	FAILED: SERVER PERSONALIZATION FAILED	An error occurred while trying to personalize the server. Personalization is Intel Deployment Manager's process of injecting IP addresses and other server specific information into a server's snapshot after it is loaded on a server.
2118	FAILED: CANNOT CONTACT SERVER	The Intel Deployment Manager server was unable to contact the server. Make sure that the server is running the ActiveOS software. If it is not, try power cycling the server to get it to load and execute the ActiveOS software.

Server States

State ID	Server Status	Description
2119	FAILED: CANNOT ROUTE COMMUNICATIONS TO SERVER	The Intel Deployment Manager server was unable to start up the server because it could not determine a communications route to the server. This is usually due to the fact that the server is not connected to an IP list or a network with a uplink.
2120	FAILED: NO OS DETECTED	The Intel Deployment Manager server was unable to perform the save because it could not determine what OS the custom snapshot should be.
2121	FAILED: IO ERROR	While saving or loading a snapshot, an I/O error occurred.
2122	FAILED: CHECKSUM MISMATCH	The checksum failed while loading a snapshot.
2123	FAILED: CHECKSUM GENERATION	Generating a checksum failed while saving a snapshot.
2124	FAILED: SAVE ERROR	Saving a snapshot failed.
2125	FAILED: LOAD ERROR	Loading a snapshot failed.
2126	FAILED: MBR RESTORE	Restoring the MBR (Master Boot Record) failed while loading a snapshot.
2127	FAILED: STORAGE MOUNT	Failed to access the storage of a snapshot while loading or saving.
2128	FAILED: MBR FIXUP	An error occurred while the MBR (Master Boot Record) was being fixed up after imaging.
2129	FAILED: RESTORE BOOT SECTOR	Failed to restore the bootsector while loading a snapshot.
2130	FAILED: SAVE BOOT SECTOR	An error occurred while saving the bootsector.
2131	FAILED: FILE SYSTEM CREATE	An error occurred while creating a filesystem while loading a snapshot.
2132	FAILED: FILE SYSTEM LABELING	Failed to restore a filesystem label while loading a snapshot.
2133	FAILED: READ ONLY FILESYSTEM	Attempted to save a snapshot on a display only filesystem.

Server States

State ID	Server Status	Description
2134	FAILED: OUT OF DISK SPACE	A disk full error occurred while saving a snapshot.
2135	FAILED: PARTITION TABLE SAVE	Failed to save the partition table of the disk.
2136	FAILED: PARTITION TABLE RESTORE	Failed to restore the partition table from the snapshot to the disk.
2137	FAILED: COULDN'T RUN SOFTWARE	An error occurred while trying to execute the software. This can be due to a bad command line argument, a corrupted software file, or a crash during execution of the software.
2138	FAILED: SOFTWARE RETURNED AN ERROR	The given software has completed execution, but returned an error return code. The return code is displayed in the server's status window.
2139	FAILED: SERVER COULD NOT BE CONTACTED AFTER PERSONALIZATION	After loading a snapshot, the server can not accessed over the network.
2140	FAILED: FILE SYSTEM CHECK	Filesystem needs to be checked with `fsck` or `chkdsk`.
2141	FAILED: OUT OF MEMORY	The system being imaged ran out of memory.
2142	FAILED: DISK TOO SMALL	The hard disk was too small to accept the snapshot being loaded.
2143	FAILED: REBOOT	Attempt to reboot the server failed.
2144	FAILED: CUSTOM REBOOT	Attempt to reboot using a custom reboot, specified by the user, failed.

Understanding Server Asset States

Every server asset has **Asset Status** (hardware-oriented) and **Server Status** (software oriented) fields that display server asset states. You can click on the state shown in a status field to pop up a window that contains additional information about the state.

If either the **Asset Status** or **Server Status** field is not displayed, you can display it by moving it to a displayed column as shown in "Setting Table Options" on page 16.

Server asset states are categorized as follows:

- Intermediate Steps for the Save Snapshot request.
- Intermediate Steps for the Deploy Software request.
- Last Save or Software execution Errors.

The following table lists the server asset states.

Server Asset States

State ID	Server Asset States	Description
0	OK	The server asset is in a valid state. No knowledge is known about the status of the last operation. Newly created server assets are assigned this value.
Intermediate Steps for the Save Snapshot Request		
3001	STARTING SAVE	A `Save Snapshot` command is just starting on this server asset.
3002	SAVING SNAPSHOT	The progress of this save is displayed in the progress bar of the server's status window as a percentage. The throughput and the percentage complete for the current partition is also displayed.
3003	FINISHING SAVE	A `Save Snapshot` command has just finished saving the server asset's hard disk contents into a snapshot and the system is now performing some final bookkeeping to commit this save.
3099	SAVE DONE	The last command performed on this server asset was successful.
Intermediate Steps for the Execute Software Request		
3020	RUNNING SOFTWARE	Software is currently running on this server asset.

Server Asset States

State ID	Server Asset States	Description
3021	UPLOADING SOFTWARE	Software is currently being uploaded to this server asset.
3098	SOFTWARE DONE	The last execution of software on this server asset was successful.
3040	UPDATING	The Intel Deployment Manager system is currently updating the server asset.
3097	UPDATE DONE	The update of the server asset is done.
Last Save or Software execution Errors		
3100	FAILED: UNKNOWN ERROR	The last `Save Snapshot` command failed for some reason unknown by the Intel Deployment Manager system. This error can be treated as another kind of internal error. The message of the error is displayed via the `more details` link in the server asset status window
3101	FAILED: INTERNAL ERROR	The last `Save Snapshot` command failed due to some internal error in the Intel Deployment Manager system. This error might indicate there is a bug in the Intel Deployment Manager system software. The message of the error is displayed via the `more details` link in the server asset status window
3103	FAILED: NETWORK ERROR	The last `Save Snapshot` command failed due to some a networking failure. This error usually indicates a hardware failure.
3104	FAILED: SNAPSHOT ALREADY EXISTS	The last `Save Snapshot` command failed due to the fact that the system already has a snapshot with the same name.
3105	FAILED: COULDN'T START SAVE	The last `Save Snapshot` command failed due to the fact that the Intel Deployment Manager server was not able to communicate with the agent running on the server asset. This usually indicates that something is wrong with the server asset.
3106	FAILED: STARTUP FAILED	The system could not start up the server asset.
3107	SCRUBBED	The Intel Deployment Manager system's consistency checker found this server in an inconsistent state and has reset it to a consistent state.

Server Asset States

State ID	Server Asset States	Description
3108	NOT RESPONDING	The Intel Deployment Manager system is unable to communicate with the server asset.
3109	FAILED: COULDN'T CONTACT SERVER ASSET	The Intel Deployment Manager server was unable to contact the server asset. Make sure that the server asset is running the ActiveOS software. If it is not, try power cycling the server asset to get it to load and execute the ActiveOS software.
3110	FAILED: NO OS DETECTED	The Intel Deployment Manager server was unable to perform the `save` command because it could not automatically determine the base OS of the snapshot to save.
3111	FAILED: COULDN'T RUN SOFTWARE	An error occurred while trying to execute the software. This can be due to a bad command line argument, a corrupted software file, or a crash during execution of the software.
3112	FAILED: SOFTWARE RETURNED AN ERROR	The software has completed execution but returned an error return code. The return code is displayed in the server asset's status window.
3113	FAILED: IO ERROR	While saving or loading a snapshot, an I/O error occurred.
3114	FAILED: CHECK SUM MISMATCH	The checksum failed while loading a snapshot.
3115	FAILED: CHECK SUM GENERATION	Generating a checksum failed while saving a snapshot.
3116	FAILED: SAVE ERROR	Saving a snapshot failed.
3117	FAILED: LOAD ERROR	Loading a snapshot failed.
3118	FAILED: MBR RESTORE	Restoring the MBR (Master Boot Record) failed while loading a snapshot.
3119	FAILED: STORAGE MOUNT	Failed to access the storage of a snapshot while loading or saving.
3120	FAILED: MBR FIXUP	An error occurred while the MBR (Master Boot Record) was being fixed up after imaging.
3121	FAILED: RESTORE BOOT SECTOR	Failed to restore the bootsector while loading a configuation.

Server Asset States

State ID	Server Asset States	Description
3122	FAILED: SAVE BOOT SECTOR	An error occurred while saving the bootsector.
3123	FAILED: FILESYSTEM CREATE	An error occurred while creating a filesystem while loading a snapshot.
3124	FAILED: FILESYSTEM LABELING	Failed to restore a filesystem label while loading a configuation.
3125	FAILED: READ ONLY FILE SYSTEM	Attempted to save a snapshot on a read-only filesystem.
3126	FAILED: OUT OF DISK SPACE	A disk full error occurred while saving a snapshot.
3127	FAILED: PARTITION TABLE SAVE	Failed to save the partition table of the disk.
3128	FAILED: PARTITION TABLE RESTORE	Failed to restore the partition table from the snapshot to the disk.
3129	FAILED: FILE SYSTEM CHECK	Filesystem needs to be checked with `fsck` or `chkdsk`.
3130	FAILED: OUT OF MEMORY	The system being imaged ran of memory.
3131	FAILED: DISK TOO SMALL	The hard disk was too small to accept the snapshot being loaded.
3132	FAILED: REBOOT	Attempt to reboot the server failed.
3133	FAILED: CUSTOM REBOOT	Attempt to reboot using a custom reboot, specified by the user, failed.

Constraints and the Constraints Editor B

Constraints are a set of rules or expressions that describe where and how you can use an object. Constraints provide the means for you to systematically define the notable attributes of a snapshot, hardware, or any other platform-specific data, and the rules that these attributes should follow. Intel Deployment Manager can then perform a verification process using the attribute and the user defined rule set.

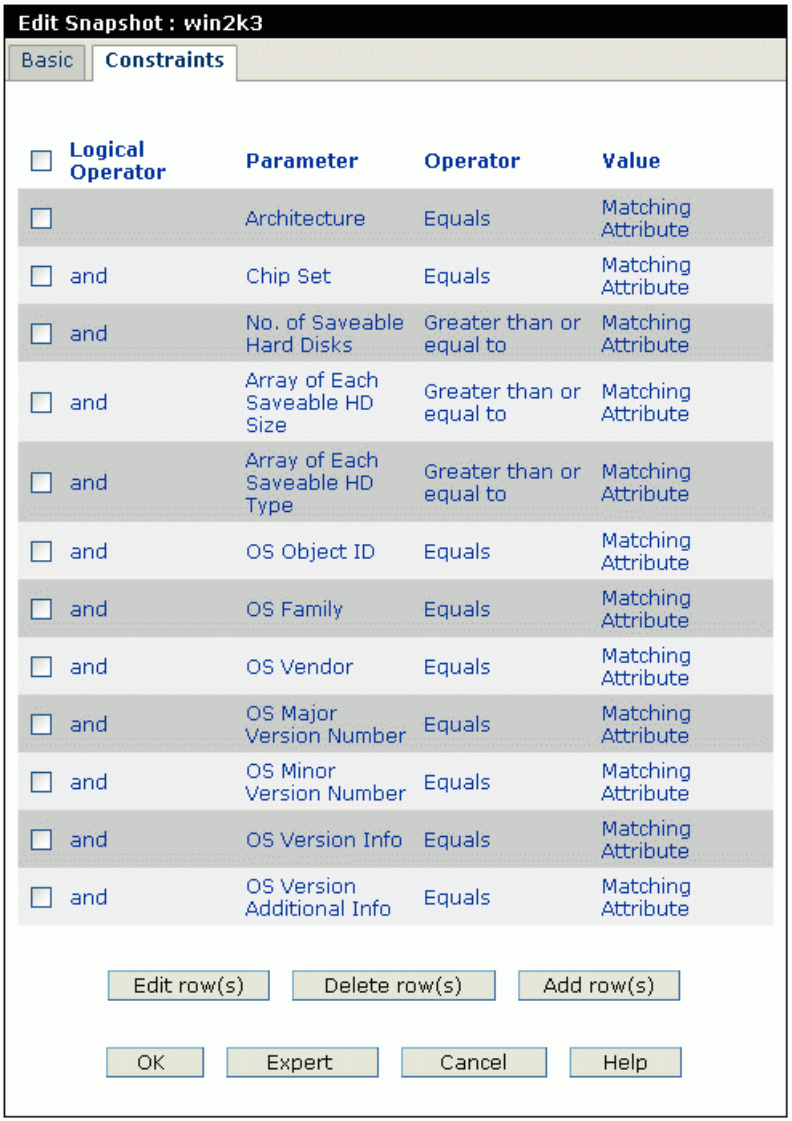
After clicking the **Constraints** tab, you can use the Constraints Editor, or you can click the **Expert** button to directly edit constraint rules.

This Appendix includes the following topics:

Using the Constraints Editor

Clicking the Constraints tab in any dialog displays the Constraints Editor. The Constraints Editor enables you to edit the constraints assigned to servers, computing devices, software, and snapshots. The editor displays each constraint on a separate line.

Constraints for the Edit Snapshot Dialog



You can define constraints for each parameter when you add or edit a row. Fields are as follows:

Constraint Editor Fields

Field	Description
Check Box	The check box is used to add or delete a row. For example, if the check box of a row is selected and you click the **Add row(s)** button, a blank row is inserted below the selected row. If no row is selected and the **Add row(s)** button is clicked, a blank row is added below the last row. Multiple rows can be inserted simultaneously.
Logical Operator	The options for the logical operator are **and** and **or**. For example: ◆ Constraint A **and** Constraint B—both should be satisfied. ◆ Constraint A **or** Constraint B—any one of the constraints should be satisfied. ◆ Constraint A **and** Constraint B **or** Constraint C **and** Constraint D—one of the constraints from C and D should be satisfied and one of the constraints from A and B should be satisfied.
Parameter	The parameter column lists all constraint key words. The key words are limited on the basis of an object. For example, for software, only those constraints relevant to software are shown. If you select the user-defined option, a text box replaces the drop-down list.
Operator	This column displays the following operators: **Equals**, **Not equal to, Greater than**, **Greater than or equal to**, **Less than**, **Less than or Equal to**.
Value	The value column has the following options: matching attribute, single value, set of values, and array of values. ◆ **Matching Attribute**—A matching attribute indicates that the selected attribute matches completely. If a selected snapshot has a configuration for two hard disks then the machine on which it is loaded should also have two hard disks. ◆ **Single Value**—A single value indicates that the selected attribute has a single hardcoded value which becomes a constraint. ◆ **Set of Values**—A set of values is an *unordered* collection of strings or numbers. Because each set is unordered, set (1, 2, 3) equals (3, 2, 1) and (2, 1, 3), etc. ◆ **Array of Values**—An array of values is an *ordered* collection of strings or numbers. Unlike a set of values, array (1, 2, 3) does not equal (3, 2, 1) or (2, 1, 3). Each is a different array. If you select **Single Value**, **Set of Values**, or **Array of Values**, a text box is displayed where you can enter the required values needed to complete that specific row of constraints.

When you click **OK**, the dialog is refreshed and displays the edited constraints. You cannot click **OK** if a row is not completely edited.

To directly edit a constraint string, click on the **Expert** button.

Understanding Constraint Rules

Intel Deployment Manager lets you create and manage snapshots and other hardware-dependent data. In order to simplify cross-platform verification processes, a constraint system is defined. Attributes for an object are created when that object is added to the system. When creating a computing device object; for example, you enter attributes such as the machine architecture, CPU speed, and hard disk size.

A *constraint rule* is a rule written as a character string that describes how attributes describe the object. For example, a snapshot can have the attribute `128 MB of memory`, because it was saved from a machine with 128 MB of memory. A rule might say that this snapshot is allowed to run on machines, not only those with exactly 128 MB of memory, but also on machines with more memory.

When loading a snapshot onto a computing device or executing software on a server OS, the constraint rules of the two are compared. If the comparison finds that the two are compatible, the operation is allowed to proceed. If, for instance, you request to load a snapshot on a computing device, the constraint rule for the snapshot is verified against the constraint rule of the computing device. Only on a positive match does the system actually place the snapshot onto the computing device's hard disk.

When entering information for any of the constraint-protected objects (for example, computing devices), included is the constraint rule itself. Often, you might want to write the rule so that it is dependent on other attributes of the object, such as the architecture or chipset of a server system. Without some forethought, you could create a constraint rule where some of the values might have to be entered multiple times, once for the attribute and once for the constraint rule. In order to simplify this process, constraint rule templates can be used. Using a specific syntax, you can write a constraint rule template that refers to other attributes of the object. These values will be substituted when the rule is actually being used in a comparison. This allows you to enter a generic, custom constraint rule without having to enter or update certain values in multiple locations.

Comparing Constraints

Constraints can be applied to servers, computing devices, software, and snapshots. When any of these objects are used, the constraints for the object are compared against the constraints of any other object involved in the operation. If any of the constraints fail to match, an error is generated and the operation is aborted.

A constraint rule is attached to each object and defines how an object's attributes relate to the object itself. To compare if two objects are compatible, the constraint rule for each object is compared to the other. If there exists any values for each attribute that satisfy both constraint rules, the constraint rules are said to match. So, for example, comparing two constraint rules A=B and A<C, a comparison would result in a match if (and only if) the inequality B < C were true.

Integers follow the normal comparisons. Strings are not lexicographically ordered (hereby referred to as *unordered* strings). This means that given two unequal unordered strings, comparisons dealing with less-than or greater-than operators are defined to be false. This behavior is different than how strings are defined in many other languages, such as C, Java, and Perl.

Sets are unordered and each element is unique (that is, it behaves as if it occurs only once in the set). Because sets are unordered, set (1, 2, 3) equals (3, 2, 1) and (2, 1, 3), and so on.... The comparison operators specify set operations, such as subset, strict subnet, and others (See “Describing Grammar and Semantics for Constraints” on page 266).

Arrays, on the other hand, are ordered and each element might not be unique. Unlike a set, array (1, 2, 3) does not equal (3, 2, 1) or (2, 1, 3). Each is a different array.The operators do individual comparisons on each array element at each corresponding array index. If the arrays are of different lengths, shorter arrays are defined to be less-than arrays with more items. In either case, each array element is compared up to the length of the shorter array.

The constraint matches only if the operations on each element and the array length (shorter arrays are defined to be less-than longer ones) are verified.

Examples

In order to provide a clearer view of how constraint rules are created and written, the following table describes some of the constraints used in the examples.

Objects and Constraint Rules

Object Type and Name	Constraint Rule applicable to Object
Server Asset A (SA-A)	MBArch=%MBArch%&HDcount=1
Server Asset B (SA-B)	MBArch=sun4u&HDcount=2
Server w/ OS A (S-A)	MBArch=%MBArch%&HDcount=1&OSBase=%OSBase%
Server w/ OS B (S-B)	MBArch=sun4u&HDcount=2&OSBase=redhat6
Software B (MC-B)	MBArch=dlx
Snapshot A (IM-A)	MBArch=%MBArch%&HDcount>=2

Note the use of template variables, such as in SA-A, S-A, and IM-A, which use the *MBArch* variable name as the right-hand side of the MBArch attribute. These are simplified versions of the actual default constraint rules that are used in the system.

Loading a Snapshot on a computing device

When loading a snapshot on a computing device, the computing device's rule and the snapshot's rule are compared. Note that the comparison does not happen when loading, but rather, when provisioning. Executing software (such as a bootable software) on a computing device goes through identical verification steps for the constraints.

- IM-A onto SA-A. This will never match because the attribute HDcount will never match.
- IM-B onto SA-A. The HDcount matches, and MBArch will match only if SA-A's architecture field has set it to an Intel Xeon processor family compatible.
- IM-A on SA-B. This will match if the snapshot was saved from a sun4u machine. Remember that the constraint variables for a snapshot come from the computing device from which the snapshot was saved.
- IM-B on SA-B. This will never match because the MBArch fields will never match.

Executing Software on a Server with an Operating System

The following is an example of running OS dependent software, such as ActiveOS software, on a provisioned target server.

- MC-A on S-A. This will match if the computing device is Intel Xeon processor family compatible and the OS matches. S-A gets its OS from the snapshot that is loaded onto it. MC-A gets its OS from an attribute.
- MC-B on S-A. This will run if the computing device is of type dlx.
- MC-A on S-B, MC-B on S-B. This will never run because the architecture (*MBArch*) fields mismatch.

Additional Examples

The following are some additional examples of constraints. The types of the constraint attributes are denoted in the name of the attribute.

Additional Examples

Constraint 1	Constraint 2	Matches	Description
INT_A=1	INT_A=1	true	1 equals 1
INT_A=2	INT_A=1	false	2 does not equal 1
INT_A=1& INT_B=2	INT_A=1& INT_B=3	false	2 does not equal 3
INT_A=1	INT_A<3	true	1 is less-than 3
STR_A=abc	STR_A=abc	true	Strings "abc" match
STR_A=abc	STR_A<=cde	false	Strings are unordered.
SET_A={1,2}	SET_A<={0,1,2,3}	true	{1,2} is a subset of {0,1,2,3}
SET_A={1,2}	SET_A<={2,3}	false	{1,2} is not a subset of {2,3}
ARR_A={1,2}	ARR_A<={2,3}	true	Each element satisfies the <= condition and the arrays are of the same length
ARR_A={1}	ARR_A<={2,3}	true	Each element satisfies the <= condition. The first array is shorter, satisfying the less-than condition.

Intel Deployment Manager supports a strong set of constraint bases analyses for verification purposes. Custom constraints and constraint rules can be created and tailored by you to your own specific needs. In addition, the language supports the use of template strings that reduce the cost of maintenance and the possibility for human error.

Understanding the Constraints Language

The constraint rule string is written as a series of attributes, values, and operations on those attributes and values. The basic structure of the rule can be broken down into two parts. First, a constraint is defined by creating a relation using an attribute, value, and an attribute-op triplet (for example, A=B signifies attribute A, value B, and an attribute-op of equals). Second, constraints are then glued together by a constraint-op (for example, A=B&C=D signifies two constraints A=B and C=D combined with an & constraint-op).

The language is strongly typed. Attributes have a type associated with them that determines how each operation executes. The type system supports integer and string primitives, sets, and arrays, as well as operations on those sets and arrays. The following table describes the properties that can be set in the file `elemental.properties`

Constraint Properties

Property Name	Description
`constraints.types`	Attribute type definition string. String that declares the types for all valid attributes.
`constraints.contexts.all`	Attribute list string. A list of attributes that the constraint system recognizes. There is normally one entry for each type definition.
`server.constraints.server.default`	The default constraint rule to use for computing devices.
`server.constraints.vserver.default`	The default constraint rule to use for servers.
`server.constraints.dmexec.defaults`	The default constraint rule to use for configuration software.
`server.constraints.mcode.default`	The default constraint rule to use for bootable software.
`server.constraints.bmexec.default`	The default constraint rule to use for ActiveOS software.
`server.constraints.image.default`	The default constraint rule to use when saving new snapshots.

The section “Describing Grammar and Semantics for Constraints” on page 266, describes the grammar for the constraint rule string, as well as some basic semantic and type information.

Using Constraint Template Variables

In order to simplify the creation of generic rules, constraint templates are supported. By specifying a constraint value (right-hand side) that corresponds to a constraint variable name (signified by surrounding % symbols), the value of the attribute name is inserted from the variable. The variable is usually set to an attribute of the object, such as the architecture of a computing device, or some other property that can be inferred, such as the number of hard disks. Each variable corresponds to a field or multiple fields in the computing device, snapshot, or software.

Note that template variables are not true variables in the sense that they can be assigned, or that they can be a part of a larger expression. There are restrictions on their use that prevent you from creating complex expressions. (See “Describing Grammar and Semantics for Constraints” on page 266). They are limited to the right-hand side of the constraint expression and cannot be part of a more complex right-hand side expression.

For snapshots, the template variables are extracted from the computing device from which the snapshot was saved. This behavior allows you to create constraint rules based off of that original computing device.

Describing Grammar and Semantics for Constraints

The following sections describe grammar and semantics for constraint rules and template variable definitions.

Describing Grammar for Constraint Rules

```
constraint-rule <- constraint
                 | constraint constraint-op constraint-rule

constraint-op   <- '&' | '+'

constraint      <- attribute attribute-op value

attribute       <- id

attribute-op    <- '=' | '<' | '<=' | '>' | '>=' | '!='

value           <- single-value
                 | set-value
                 | templated-value
                 | any-value
                 | array-value

single-value    <- id

set-value       <- '{' set-value-list '}'

set-value-list  <- single-value
                 | single-value ',' set-value-list

array-value     <- set value
templated-value <- '%' id '%'

any-value       <- '%' '*' '%'

id              <- [A-Za-z0-9_]*
                 | '\'' (id | [ ])* '\''
```

Using Template Variable Definitions

The following sections list the computing device variables, snapshot variables, provisioned server variables, and software variables.

Server Asset Variables

Name	Description
MBcpuCount	CPU count
MBavgCpuMhz	CPU speed
MBArch	architecture
MBcpuFamily	Processor Family
MBcpu64Bits	supports 64-bit addressing
MBcset	chip set
MBmem	memory size
MBbiosVer	BIOS version
MBbiosInfo	BIOS Info
MBraidInfo	Hardware RAID Info
HDsize	array of each hard disk size
HDtype	array of each hard disk type
HDcount	number of hard disks
HDdev	array of each device paths
HDraidLevel	array of each HD RAID level
HDIcount	number of savable hard disks
HDIsize	array of each savable HD size
HDItype	array of each savable HD type
HDIdev	array of each savable HD device paths
HDIraidLevel	array of each savable HD RAID level
CNcount	number of network connections
SYslot	slot
SYlocation	location
SYmanufacturer	manufacturer

Server Asset Variables

Name	Description
SYmodel	model name
SYmodelNo	model number
SYserialNo	serial number
SYlicenses	licenses associated with the server asset as a set of strings
SYuuid	GUID
EXchassisId	serial number of chassis

Snapshot Variables

Name	Description
IMlicenses	licenses associated with an snapshot as a set of strings
OSBase	operating system name
OSFamily	operating system family
OSVendor	operating system vendor
OSMajorVer	operating system major version number
OSMinorVer	operating system minor version number
OSVerType	OS version information
OSVerSubType	OS version, additional information
OSKrnMajorVer	operating system kernel major version number
OSKrnMinorVer	operating system kernel minor version number
OSKrnMicroVer	operating system kernel micro version number

Provisioned Server Variables

Name	Description
SEhostName	host name of the server
SEinfo	server info field

Software Variables

Name	Description
MBArch	architecture
OSBase	operating system name
OSFamily	operating system family
OSVendor	operating system vendor
OSMajorVer	operating system major version number
OSMinorVer	operating system minor version number
OSVerType	OS version information
OSVerSubType	OS version, additional information
OSKrnMajorVer	operating system kernel major version number
OSKrnMinorVer	operating system kernel minor version number
OSKrnMicroVer	operating system kernel micro version number
OSBase	operating system name

Using Operators

The following table lists the operators used in constraints, in order of precedence.

Operators used in Constraints

Operator	Description
=	Equality or set equality
<	Less than or strict subset
<=	Less than equal or subset
>	Greater than or strict superset
>=	Greater than equal or superset
!=	Not equal or set inequality
&	Intersection
+	Union

Grammar for Attribute Type Definitions

```
Type-definition       <- attrib-with-type-list

attrib-with-type-list <- attribute-with-type
                       | attribute-with-type';'
                         attribute-with-type-list

attribute-with-type   <- attribute-name ':' attribute-type

attribute-type        <- [0-9]*

attribute-name        <- id
```

Type of Attributes

The following table lists the attribute names, values, and descriptions.

Attributes, Values, and Description

Name	Value	Description
Integer	1	Positive numerical value
String	2	Unordered string
Set of Integers	31	Set of unique numerical values
Set of Strings	32	Set of unique strings of characters
Array of Integers	41	Array of numerical values
Array of Strings	42	Array of strings of characters

Grammar for the Attribute List String

```
attribute-list <- attribute-name
                | attribute-name ',' attribute-list

attribute-name <- id
```

Attribute Type Definition Strings with Default Values

The following is a list of attribute type definition strings (attribute list strings) with default values in the format, *attribute* : *default_value.*

Attribute Type Definition Strings with Default Values

CNcount:1	EXchassisId:2	EXmisc:32	EXpool:32
HDcount:1	HDtotal:1	HDsize:41	HDtype:42
IMlicenses:32	MBArch:2	MBavgCpuMhz:1	MBbiosInfo:2
MBbiosVer:2	MBcpuCount:1	MBcset:2	MBmem:1
MBraidInfo:2	OSBase:32	OSFamily:2	OSKrnMajorVer:1
OSKrnMinorVer:1	OSKrnMicroVer:2	OSMajorVer:1	OSMinorVer:1
OSVendor:2	SNdomain:32	SNglobal:32	SNlocal:32
SNsig:32	SYlicenses:32	SYlocation:2	SYmanufacturer:2
SYmodel:2	SYmodelNo:1	SYserialNo:2	SYslot:2

Network Topologies C

This Appendix provides information about various network topologies that can be used with Intel Deployment Manager.

Topics in this Appendix include:

Things to Consider

There are special considerations when using Intel Deployment Manager on a network with an external DHCP server or routers. You can run only one TFTP server on the Intel Deployment Manager server, but you can configure multiple HTTP and HTTPS servers on the same machine as long as you assign non-default ports during installation.

Intel Deployment Manager and DHCP

DHCP is a UDP-based broadcast protocol. When a machine sends out a DHCP request it is usually sending out a request for an IP address. When a machine sends out a PXE request (which is based on DHCP) it is sending out a request for an operating system to load. The Intel Deployment Manager server has a built-in DHCP server that is used to satisfy PXE and DHCP requests.

When a server's NIC broadcasts a DHCP request, every machine that can be reached by the broadcast receives a DHCP packet from the DHCP server. If the request reaches more than one DHCP server, the NIC making the DHCP request might get more than one response (one from the Intel Deployment Manager server and others from the other DHCP servers). If the packet from the Intel Deployment Manager server arrives first then everything works properly. If the external DHCP server's packet arrives first the machine is configured by the external DHCP server. There is no way to know which packet will arrive first; this is *random* and *unpredictable*. This is why we generally recommend that Intel Deployment Manager not be on the same broadcast network as an external DHCP server.

In fact, it is possible to use an external DHCP server on the same broadcast network as Intel Deployment Manager, but additional configuration is required. For example, you might be able to configure your DHCP server to ignore DHCP requests from the MAC addresses of Intel Deployment Manager-managed assets. You could then use the DHCP server to boot other machines in the same network. The additional configuration work required varies depending upon your goals and the kind of DHCP server you are using. See "Using Intel Deployment Manager With an External DHCP Server" on page 278.

DHCP is broadcast; it is not routable because there is no destination address. You can get around this restriction if your routers support application level proxies known as DHCP Relays. These relays can be configured to send broadcast DHCP packets to and from other networks. See "Using Intel Deployment Manager Over a Routed Network" on page 277 for more information about routed networks and DHCP relays.

DHCP typically has a short timeout, so fast response times are critical for DHCP to work properly. These timeouts are typically not configurable because they are burned into the ROMS of the PXE NICs.

Intel Deployment Manager and TFTP

TFTP (Trivial File Transfer Protocol) is a point-to-point UDP-based file transfer protocol. When using TFTP, only the specified server address is contacted, so running more than one TFTP server on the same broadcast network as Intel Deployment Manager is allowed. But, because TFTP uses a well known port number, you cannot have more than one TFTP server running on the Intel Deployment Manager server.

Intel Deployment Manager runs a `tftp` daemon because PXE requires it in order to load bootable images over the network. The `tftp` daemon runs as part of the Intel Deployment Manager process so you do not see it when you do a `ps` on a Linux or UNIX system, or when using the Windows Task Manager. The port TFTP uses *cannot* be configured because the clients (PXE NICs) have this port number burned into their ROMs.

Intel Deployment Manager and HTTP

HTTP and HTTPS are TCP-based point-to-point protocols. Intel Deployment Manager uses HTTP and HTTPS to serve its user interface (the administration console) to a web browser and to execute client programs that are written in JXML.

You can have other HTTP / HTTPS servers on the same broadcast network as Intel Deployment Manager. HTTP / HTTPS uses well known ports, but these ports can be configured since your web browser can use a different URL to communicate with Intel Deployment Manager. For example, if you specified port 6798 for HTTP, and 6799 for HTTPS, then the URLs for communicating with Intel Deployment Manager are:

- http://*yourserver.yourcompany*.com:6798
- https://*yourserver.yourcompany*.com:6799

If you use different ports for each web server, you can have more than one web server running on the Intel Deployment Manager server.

Intel Deployment Manager and Firewalls

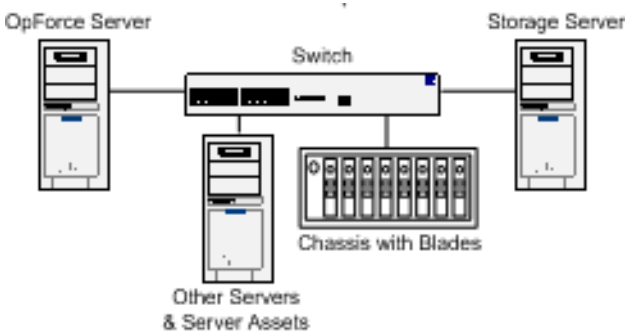
You can access the Intel Deployment Manager administration console from beyond a firewall, but the Intel Deployment Manager server, its storage server(s), and the machines Intel Deployment Manager manages must all be on the same side of the firewall.

Caution Placing a firewall between your Intel Deployment Manager server and the storage server(s) causes severe performance degradation! This configuration is not supported.

Using Intel Deployment Manager Over a Simple Network

In a simple network topology, everything is on the same network (subnet). This topology works well with Intel Deployment Manager—there are no special configuration issues.

Simple Network

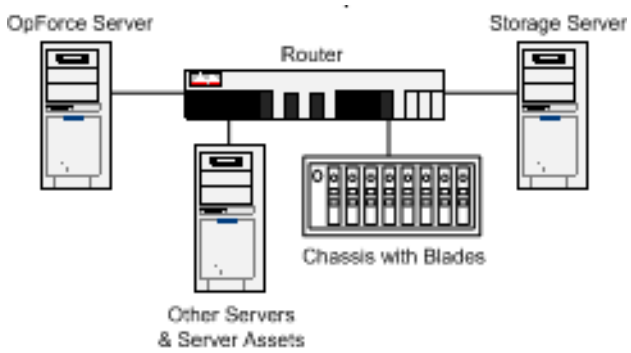


You might use this topology in a Quality Assurance laboratory, or a computer vendor / integrator might use this topology when replicating a particular system configuration on new machines.

Using Intel Deployment Manager Over a Routed Network

Normally, you would not want to place a router between the Intel Deployment Manager server and its storage server(s) or the assets it manages because DHCP and PXE broadcasts are not routable. However, a router can be used if it supports DHCP relay, or if you set up a DHCP relay server, so that broadcasts can pass to other networks. Procedures for setting up a DHCP relay vary depending upon the type of router or OS you are using. See the documentation for your equipment.

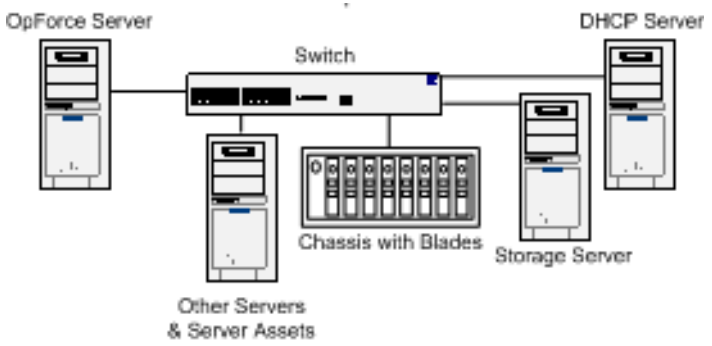
Routed Intel Deployment Manager Network



Using Intel Deployment Manager With an External DHCP Server

You can run Intel Deployment Manager on a broadcast network with an external DHCP server, but additional configuration work is required. The tasks required vary depending upon your goals and the kind of DHCP server you are using.

Network with External DHCP Server



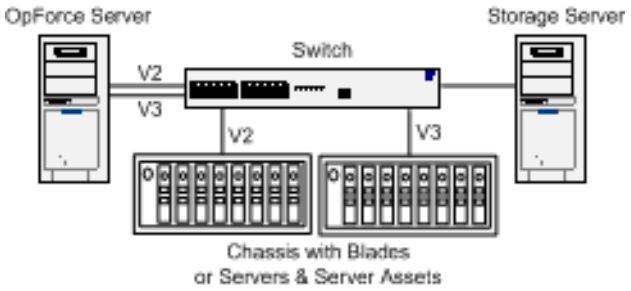
In order to successfully run Intel Deployment Manager on a network that has an external DHCP server:

- The boot NIC for each managed asset must get a static IP address from Intel Deployment Manager. Other NICs can have IP addresses assigned by DHCP.
- The DHCP server must be configured to ignore the MAC addresses of boot NICs on managed assets.
- The DHCP server must be configured to ignore DHCP packets from managed servers.

Using Intel Deployment Manager Over a Simple VLAN Network

You can set up multiple VLANs as long as Intel Deployment Manager can receive DHCP requests over each VLAN.

Simple VLAN Network



Using Intel Deployment Manager Over a WAN

Using Intel Deployment Manager over a WAN is not supported. Attempting to use Intel Deployment Manager over a WAN introduces unacceptable latency for broadcast protocols. It also results in poor response times from the storage server(s).

Glossary D

Access Control

An *access control* maps a specific user role to a workspace. When a user role is mapped to a workspace by an access control, all users in that role are allowed to view all of the resources in the workspace. Each access control has a set of permissions which identify the operations the user is allowed to perform on the resources included in the workspace. See “User Role” and “Workspace”.

ActiveOS

The *ActiveOS* is a self-contained “memory resident” OS used by Intel Deployment Manager to perform automatic hardware attribute and OS detection, as well as snapshot deployment and capture. For Intel Xeon processor family-based Compute Blades the ActiveOS is a Linux-based OS.

The ActiveOS collects various hardware attributes from the server, such as system-type, CPU-speed, motherboard, chipset, number and size of hard disks, memory, and network card parameters. Some of these attributes are determined by examining the system’s SMBIOS. Others are determined by examining the NICs and hard disks attached to the system.

Administrator

`Administrator` is privileged user ID that has complete control of all the resources managed by Intel Deployment Manager. The `Administrator` account has full privileges to perform system-wide operations including user and account management, resource management, and event management.

Admin State

A server is considered to be in an ADMIN STATE if it is under the control of ActiveOS. Valid operations for servers in the ADMIN STATE include: Save Snapshot, Provision Server, Startup, Load Server, Load & Startup, Deploy Software, Reboot to Admin State, Reset, Rediscover Asset Attributes, Reconnect, and Open a Console. When you display the asset, the **Server Status** shows ADMIN STATE. See “State”.

Bare Metal Server Discovery

Bare Metal Server Discovery refers to the discovery of new Intel Server Compute Blades that do not have an available operating system.

Blade Server Asset

A *blade server asset* is a server asset that is associated with a chassis and a slot number that is either detected by Intel Deployment Manager or entered by a user. See “Server Asset”.

Computing Device

The term *Computing Device* refers to servers, server assets, blade assets, and chassis assets that can be managed by Intel Deployment Manager.

Constraints

Constraints are a set of rules or expressions that describe where and how you can use an object. Detailed information about constraints is provided in “Constraints and the Constraints Editor” on page 257.

DNS

See “Domain Name Service”.

DNS Address

A *DNS address* is the network address of a domain name server—usually, an IP address. See also “Domain Name Server”.

Domain Name

A *domain name* is an internet address in alphabetic form. Domain names must have at least 2 parts: the part on the left which names the organization, and the part on the right which identifies the highest subdomain, such as the country or the type of organization (com, edu, and so on). For example, *yourcompany*.net.

Domain Name Service

A *domain name service* is a network service that resolves domain names into IP addresses. Such a service is usually software that is running on a designated DNS (domain name server) on the network at a known IP address. See also “Domain Name Server”.

Domain Name Server

A *domain name server* is a server, at a known IP address, that runs a Domain Name Service for the network. See “Domain Name Service”.

Event

An *event* is a record that identifies an operation that was performed on a resource at a particular time. All operations that change the state of managed resources generate events, so events can be used to track changes.

File Mode

When a snapshot saves partition data in *file mode*, the saved data includes only data that is used by the filesystem on that partition. Instead of saving data at the sector level, file mode saves data at the file or directory level. Partition data saved in file mode represents an archived bundle of all files and directories in the filesystem. Unless you specifically save a snapshot in sector-by-sector mode, partition data is saved in file mode.

Firewall

A *firewall* consists of software, or a specially programmed computer system, that stands between an organization's network and the internet. Firewalls can be used to detect, prevent, or mitigate certain types of network attack. A firewall is a security measure used to prevent unauthorized users or programs from accessing internal networks.

Free State

A server asset is in a FREE state when it is available to be allocated or dedicated to a provisioned server, or to a server which is getting provisioned. When you display the asset, the **Server Status** shows FREE. See “State”.

FQDN (Fully Qualified Domain Name)

An *FQDN* is the full name of a system, consisting of its local host name and its domain name, including a top-level domain For example, www.*yourcompanyname*.com is a fully qualified domain name.

Full Snapshot

A *full snapshot* is a complete image of a server asset’s hard drive(s)—any full snapshot can serve as a base. A full snapshot that provides a base is required before you can perform incremental snapshots. See “Incremental Snapshot”.

Gateway Server

A *gateway server* connects and routes network traffic across two different networks, at Layer 3 of the Open Systems Interconnect (OSI) model. An example is a Gateway Server that connects to the Internet at one end and a private network of servers at the other end.

Incremental Snapshot

An *incremental snapshot* contains the changes since the last full or incremental snapshot of a server asset. A full snapshot that provides a reference or base is required before you can perform incremental snapshots. See “Full Snapshot”.

Internet Protocol (IP) Address

The *IP address* is a unique string of numbers that identifies a computer on the internet. It is a 32-bit numeric network address consisting of four parts with 8 bits in each part. An IP address usually is four numbers (in the range 0 to 255) that are separated by dots. For example, an IP address usually is shown in the form: 192.168.1.12.

IP Lists

An *IP list* is a list containing a range of network addresses in an Intel Deployment Manager-managed network. Using IP Lists, you can divide a network into multiple sections that can be assigned to different workspaces.

ITAP

ITAP is the IT Abstraction Protocol. The ITAP protocol is a proprietary TCP/IP protocol used by Intel Deployment Manager to update, access, and control target servers.

ITAP Agent

An Intel Deployment Manager *ITAP Agent* can be installed manually, or can be installed on a server by Intel Deployment Manager after it is provisioned and started up. The ITAP agent enables the following operations: establishes communication with the Intel Deployment Manager server, deploys software package(s), allows remote access, executes scripts, performs hardware attribute detection, and performs network personalization.

LDAP (Lightweight Directory Access Protocol)

LDAP defines a protocol for updating and searching directories running over TCP/IP. An LDAP directory entry is a set of attributes with a name, called a distinguished name (DN). Each of the attributes has a type and one or more values. Types are strings like “cn” for common name. The values depend on the type. LDAP directory entities are arranged in a hierarchy.

License

Intel Deployment Manager provides the means to track application and operating system *licenses* for provisioned managed servers. Any snapshot can have an associated set of licenses.

Network

A *network* in Intel Deployment Manager represents a subnet in a TCP/IP network. The network entity contains fields specifying the network's public or private IP address range, the subnet mask, and default gateway for the network.

Network Boot Protocols

Network Boot Protocols enable a server asset to obtain network personalization information and a boot image over the network. Intel Deployment Manager supports the Preboot eXecution Environment (PXE) DHCP

.Network Interface Card (NIC)

A NIC is an adapter installed in a computer to provide a physical connection to a network. It might also be referred to as a network card, or Ethernet card.

Open Systems Interconnect (OSI) Model

A standard defined by the International Standards Organization for network entities. The model specifies a stack of network protocol layers. In particular, the model specifies Layer 1 as the physical layer for handling network cables or other connection media, Layer 2 as the data link layer for handling point-to-point communication at the bit level, and Layer 3 as the network layer for handling routing and related functions.

OpScript

Intel Deployment Manager supports server-side scripting using *OpScripts*. You can build an OpScript by specifying a sequence of Intel Deployment Manager commands, or you can write Java code directly.

Personalization

Personalization is the process of giving a unique host name and network identity to each Intel Server Compute Blade that is being provisioned.

Preboot Execution Environment

Intel Deployment Manager uses *Preboot Execution Environment* (PXE) technology from Intel, or equivalent technology from other vendors (available in most off-the-shelf NICs), to enable booting from the network.

Provisioning

Provisioning is the process of specifying a snapshot and personalization information for a server asset so that a server can be provisioned (supplied with an OS, possibly application software and data, a host name, and a network identity) for its intended use. The term can also refer to a specific step in the provisioning process; the point when the virtual server is defined.

PXE

See “Preboot Execution Environment”.

RBA (Role Based Administration)

When a role is mapped to a workspace by an access control, all users in that role are allowed to view all of the resources in the workspace. Each access control also has a set of permissions which identify the operations the user is allowed to perform. For detailed information about RBA, see “Understanding Role-based Administration” on page 46.

Remote Power Controller (RPC)

Remote power controllers are power switches and power strips used to remotely power-on, power-off, or power cycle servers. Telnet is used to connect with the servers.

Resource

Resources include all of the things that can be managed by Intel Deployment Manager, including computing devices, network devices, templates, snapshots, networks, software, jobs, licenses, and provisioned resources. Each workspace has certain resources allocated to it.

Role

See “User Role” on page 289.

Root Node

The *root node* of a snapshot tree is where the full (base) snapshot is saved. All other snapshots in a tree are incremental snapshots. See “Incremental Snapshot”.

Router

A *router* is a device which connects two networks, and forwards packets between the networks. The forwarding decision is based on network layer information and routing tables.

RPC

See “Remote Power Controller (RPC)”.

Sector-by-sector mode

When a snapshot save is performed in *sector-by-sector mode*, all sectors of the partition are saved. Even if only a small portion of the partition is used by the filesystem, the entire partition is saved sector-by-sector.

Server

In Intel Deployment Manager, the term *server* refers to an object that represents a server. It is a server asset with a snapshot, personalization information (host name and network configuration), and optional software. See “Server Asset”.

Server Asset

A *server asset* is a physical machine (hardware). Server assets managed by Intel Deployment Manager run an ActiveOS. If a server asset is associated with a chassis, and a slot number is either detected by or entered by a user, the server asset is considered to be a blade server asset. See “Blade Server Asset”.

Server Object

A *server object* is an object for which you have defined a snapshot, included personalization information, possibly included hardware attribute information (beyond that already included in the snapshot), and for dedicated assets, includes assignment of physical machines.

Server Template

A *server template* provides a set of default values to be used when provisioning one or more servers. Instead of repeatedly, manually entering the same information every time you provision a server, you enter the information once into a server template, and then assign the template to the server(s) you want to provision. See “Provisioning”.

Smart sector mode

When a snapshot save is performed in *smart sector mode*, only the portion of the partition used by the filesystem is included. Compared to file mode, which also saves only the used portion of a partition, smart sector mode saves partition data at the sector level. For example, at the time of the save operation, if the filesystem on a partition uses sectors 1 .. 40 and 2000 .. 2160, only sectors 1 .. 40 and 2000 .. 2160 are saved.

Snapshot

A *snapshot* is a comprehensive software package containing:

- The software environment that is deployed on a server asset. This includes the OS, applications, and the data stored on the blade server's hard disks.
- A set of meta-information (attributes) needed to successfully and reliably deploy a snapshot on a server asset. Attributes include: hard disk type, number of hard disks, size of each hard disk, hard disk partition map, master boot record, BIOS, firmware information, and Hardware RAID settings.

See "Full Snapshot" and "Incremental Snapshot".

Snapshot Tree

Each snapshot is given a node position in the *snapshot tree* which indicates which node is the base of the delta. The root node, which is a full snapshot, is the only node without a parent. See "Root Node".

Software

Software includes both source code written by humans and executable machine code produced by assemblers or compilers. It can be viewed as a delivery mechanism for a variety of updates for the server stack. Intel Deployment Manager supports the creation and deployment of the following categories of Software: Applications, Patches, Scripts, Utilities, and Others.

Software Deployment

The process of running code or a script to deploy software on a server.

State

Every server and server asset has an asset status (hardware-oriented) and a server status (software oriented) that displays server asset states. List of server and server asset states are provided in "Server and Server Asset States" on page 245.

Storage

See "System Storage".

System Storage

System storage is physical disk space allocated for use by Intel Deployment Manager. It is a storage area where snapshots, software, and OpScripts can be saved. System storage can be local to the Intel Deployment Manager server, or can be located on one or more storage server(s) on the network. You must add system storage before most Intel Deployment Manager operations can be performed.

Subnet Mask

A *subnet mask* is a bit mask used to identify which bits in an IP address correspond to the network address and subnet portions of the address.

Switch

A *switch* is a hardware device that filters and forwards packets between LAN segments. Switches operate at the data link layer (layer 2) of the OSI Reference Model and therefore support any packet protocol.

Task

A t*ask* is a long-running operation performed by Intel Deployment Manager. It might be a running OpScript, or a server operation like startup, shutdown, save or deploy software.

Task Manager

The Task Manager provides a list of all of the tasks on the system and gives you information about the status of each task. You can use it to determine if a task is queued, sleeping, running, completed, or completed with errors.

Template

A *template* is a pattern that provides a set of default values. Intel Deployment Manager supports the use of server templates for provisioning. See “Server Template”.

Up State

A server is said to be in the UP state when it is running a disk-based operating system, and communication is established between it and the Intel Deployment Manager server. When you display the server, the **Server Status** is UP. See “State”.

User Role

A *user role* represents an arbitrary group of users who have a need to perform a similar set of tasks. Every Intel Deployment Manager user belongs to one or more user roles.

Workspace

A *workspace* is a collection of all the resources that can be assigned to a particular group of users. You can assign a resource to a workspace, as well as control user access to any workspace.

Index

W