

# CA Server Automation

## Rapid Server Imaging Server Administration Guide

**12.6**



This documentation, which includes embedded help systems and electronically distributed materials, (hereinafter referred to as the "Documentation") is for your informational purposes only and is subject to change or withdrawal by CA at any time.

This Documentation may not be copied, transferred, reproduced, disclosed, modified or duplicated, in whole or in part, without the prior written consent of CA. This Documentation is confidential and proprietary information of CA and may not be disclosed by you or used for any purpose other than as may be permitted in (i) a separate agreement between you and CA governing your use of the CA software to which the Documentation relates; or (ii) a separate confidentiality agreement between you and CA.

Notwithstanding the foregoing, if you are a licensed user of the software product(s) addressed in the Documentation, you may print or otherwise make available a reasonable number of copies of the Documentation for internal use by you and your employees in connection with that software, provided that all CA copyright notices and legends are affixed to each reproduced copy.

The right to print or otherwise make available copies of the Documentation is limited to the period during which the applicable license for such software remains in full force and effect. Should the license terminate for any reason, it is your responsibility to certify in writing to CA that all copies and partial copies of the Documentation have been returned to CA or destroyed.

TO THE EXTENT PERMITTED BY APPLICABLE LAW, CA PROVIDES THIS DOCUMENTATION "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. IN NO EVENT WILL CA BE LIABLE TO YOU OR ANY THIRD PARTY FOR ANY LOSS OR DAMAGE, DIRECT OR INDIRECT, FROM THE USE OF THIS DOCUMENTATION, INCLUDING WITHOUT LIMITATION, LOST PROFITS, LOST INVESTMENT, BUSINESS INTERRUPTION, GOODWILL, OR LOST DATA, EVEN IF CA IS EXPRESSLY ADVISED IN ADVANCE OF THE POSSIBILITY OF SUCH LOSS OR DAMAGE.

The use of any software product referenced in the Documentation is governed by the applicable license agreement and such license agreement is not modified in any way by the terms of this notice.

The manufacturer of this Documentation is CA.

Provided with "Restricted Rights." Use, duplication or disclosure by the United States Government is subject to the restrictions set forth in FAR Sections 12.212, 52.227-14, and 52.227-19(c)(1) - (2) and DFARS Section 252.227-7014(b)(3), as applicable, or their successors.

Copyright © 2011 CA. All rights reserved. All trademarks, trade names, service marks, and logos referenced herein belong to their respective companies.

# Contact CA Technologies

## Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information you need for your Home Office, Small Business, and Enterprise CA Technologies products. At <http://ca.com/support>, you can access the following:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

## Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to [techpubs@ca.com](mailto:techpubs@ca.com).

If you would like to provide feedback about CA Technologies product documentation, complete our short customer survey, which is available on the CA Support website at <http://ca.com/docs>.

## CA Technologies Product References

The following terms are used interchangeably and refer to the same things in the documentation:

- *Rapid Server Imaging (RSI) server* and *DynaCenter server*
- *RSI agent* and *DynaCenter Provisioning Agent (DPAD)*
- *RSI* and *Management Workstation (MWS)*

# Contents

---

<b>Preface.....</b>	<b>9</b>
Overview .....	9
About This Guide.....	9
Audience .....	9
What's New in this Guide .....	9
DynaCenter Version 4.1.....	9
DynaCenter Cloud Version 2.1 .....	10
Supplemental Documentation .....	10
Need Help?.....	10
Working with the Command Line.....	10
CLI Authentication .....	11
CLI Help.....	11
<b>Chapter 1: Working with Microsoft Windows Images</b>	
Background .....	13
Windows Driver Collection .....	13
Windows Driversets.....	13
Create the Windows Driver Collection .....	14
Import from vendor media .....	14
Import from a target server.....	14
Import from a user-prepared directory .....	14
Create a Driverset .....	16
Driverset naming conventions.....	16
<b>Chapter 2: Working with Hypervisors and Virtual Machines</b>	
Vendor Considerations.....	19
VMware ESX.....	19
Microsoft Hyper-V .....	20
Register a Hypervisor Controller .....	21
Register VMs and VM Networks.....	21
Add Additional VMs to an Existing Hypervisor Controller .....	22
<b>Chapter 3: Expanding Your DynaCenter Environment</b>	
Add a network that is associated with an interface on the MWS .....	23
Task 2: Register the depot .....	23
Add a network that is not associated with an interface on the MWS .....	24
Task 1: Set up a DHCP server (optional) .....	24
Task 2: Create firewall tunnels.....	27
Task 3: Register the external network.....	27
Task 4: Register the depot .....	27
Configure Multiple Static VLANs.....	28
Task 1: Register the VLAN.....	28
Add Local Depot Storage .....	28
Add External Depot Storage.....	29
Task 1: Confirm storage server requirements .....	29

Task 2: Set up the storage server .....	29
Task 3: Register the depot .....	33
Task 4: Edit the oem.ini file .....	33

## Chapter 4: Customizing Images

Overview .....	35
Section A: Understand Profiles .....	37
Types of profiles .....	37
Capture profiles .....	37
Deploy profiles.....	37
Storage Layers.....	37
Linux storage layers .....	37
Windows storage layers .....	38
Windows Filesystems.....	39
Section B: Create a Profile .....	41
Create a Profile .....	41
Sample Capture Profile Statements.....	41
Storage capture profile statements .....	42
Sample Deploy Profile Statements .....	43
Network deploy profile statements.....	43
Storage deploy profile statements.....	45
Miscellaneous deploy profile statements .....	51
Sample Complex Deploy Profile.....	51
Test a profile .....	52
Procedure.....	53
Section C: Create a Software Component .....	55
Set Permissions for Linux/Unix Components.....	55
Create a Component .....	55
Windows Component Example .....	56
Linux Component Example.....	57

## Chapter 5: General Maintenance

Manage the DynaCenter License .....	59
Examine your existing license .....	59
Request a new license.....	60
Install a new license .....	60
Backup and Restore the DynaCenter Database and Repositories .....	60
Back up the database.....	60
Restore the database .....	61
Back up the repositories .....	61
Restore the repositories .....	62

## Chapter 6: Advanced Configurations

DynaCenter Provisioning Agent Daemon (DPAD) .....	65
Install DPAD on Target Servers.....	65
Remove a DPAD from a Target Server .....	66
DynaCenter Logging.....	67
Log levels.....	67

Log files .....	69
Log file backups.....	70
<b>Chapter 7: Troubleshooting</b>	
Simultaneous Operations Fail to Complete .....	71
"Windows was improperly shutdown" Popup on Newly Provisioned Windows System .....	71
IBM Windows Provision is Missing Drivers .....	72
Windows Devices Show Exclamation Points or Question Marks After Deploy .....	72
VM Does Not PXE Boot After Agent Image Assigned .....	73
Cannot Communicate with Cloud Server After Capturing its Image .....	74
Physical NICs not Functioning as Bonded NICs After Image is Deployed .....	75
"Failed dependencies" Error When Installing DPAD on Linux 64-bit System.....	75
"Temporary failure in name resolution" Error When Deploying an Image .....	76
Target Server Storage Does Not Match Source Image/Profile Specified Storage .....	76
"SSHException: Error reading SSH protocol banner" Error When Registering a Hypervisor	77
<b>Appendix A: Support Matrices</b>	
Environmental Support List .....	79
Hypervisor Environments .....	79
Operating system matrix .....	80
Provisioning matrix .....	80
Cloud Environments .....	80
Vendor and operating system matrix.....	80
DynaCenter and cloud API matrix .....	80
Unified Computing System Environments.....	81
Supported environment.....	81
<b>Glossary .....</b>	<b>83</b>
<b>Index .....</b>	<b>87</b>



## Overview

The Racemi DynaCenter-OEM provides an API that third-party software can integrate with to perform image-based provisioning. Image-based provisioning is possible through the server and image management capabilities of DynaCenter.

DynaCenter captures server images and networking information and stores the configurations in a depot. Prior to provisioning target servers, image configurations can be customized according to the environment. For example, new binaries can be added to captured images using software components and new networking information can be added to captured images when provisioning to dissimilar hardware. After images are captured and modified, DynaCenter can deploy the images and provision new servers.

DynaCenter supports a wide range of Operating Systems and hardware. Review Appendix A: Support Matrices, which starts on page 79, for compatible Operating Systems, virtualization technologies, and hardware platforms.

## About This Guide

This guide provides procedures for customizing your DynaCenter environment, and maintaining and operating DynaCenter. After DynaCenter is installed, the CA Server Automation user interface and Command Line Interface (CLI), are available to prepare for DynaCenter-OEM deployments and image-based provisioning.

For the latest version of this Guide, contact CA Support.

## Audience

This document is intended for system and network administrators who perform administrative and operational tasks related to capturing and deploying server images to servers in datacenters and in the cloud that are managed with DynaCenter.

## What's New in this Guide

### DynaCenter Version 4.1

This Guide was updated for the 4.1 release and includes the following new or revised information:

- Revised information about using profiles for capture and deploy operations and added new sample profile statements. See Chapter 4: Customizing Images, which starts on page 35.

- Added information about setting the ipxe priority code when using multiple DHCP servers. See Add a network that is not associated with an interface on the MWS on page 24.
- Updated the process for adding a VLAN to your environment. See Configure Multiple Static VLANs on page 28.
- Updated the `dccmd register hypervisor` procedure to include the new `--power-change-timeout` option. See Register a Hypervisor Controller on page 21.
- Added information about registering the depot used by the servers that reside on a newly added internal network. See Add a network that is associated with an interface on the MWS on page 23.
- Revised information about DynaCenter log files. See DynaCenter Logging on page 67.
- Updated the support matrices. See Appendix A: Support Matrices, which starts on page 79.

## DynaCenter Cloud Version 2.1

This Guide was updated for the 2.1 release of the Cloud API and includes the following new or revised information:

- Added information about the new `dccmd cloud list vendors` command.
- Added information about using profiles with capture and deploy operations in cloud environments.

## Supplemental Documentation

- *Rapid Server Imaging Server Installation Guide*
- *Rapid Server Imaging Server Release Notes*
- *Rapid Server Imaging for AppLogic Installation and User Guide*

## Need Help?

Contact CA Support.

## Working with the Command Line

This guide provides procedures for working in your DynaCenter environment using the Command Line Interface (CLI). This section provides information that can help you as you work with the CLI.

**Note:** The CA Server Automation user interface provides equivalent operations to the CLI. See the CA Server Automation Reference Guide, RSI CLI section, for command and parameter details.

## CLI Authentication

Certain commands issued from the CLI require authentication. If authorization is required, the command will prompt for a username and password. You can avoid passing the username and password as clear text on the command line by creating an authenticated session.

### Set up an authenticated session

At any point when you are using the CLI, you can begin an authenticated session to eliminate the need to authenticate each command as you issue it.

1. On the MWS, type the following command:

```
racelogin --use-auth-cache
```

The system will prompt you for the username and password to use for this session.

2. On the MWS, run the DynaCenter CLI commands you need.

DynaCenter will execute the commands without requesting your username and password.

3. On the MWS, type the following command:

```
racelogout
```

DynaCenter ends the authenticated session and log in security is restored to the system.

**Important:** Logging out of Linux does not end the authenticated session.

## CLI Help

The dccmd program drives DynaCenter from the command line.

- To see a list of supported dccmd commands, on the MWS type:

```
dccmd --help
```

- To see help for a specific command, on the MWS type:

```
dccmd <command_name> --help
```



# Chapter 1: Working with Microsoft Windows Images

---

DynaCenter is not distributed with any Microsoft Windows drivers pre-loaded. If you are working with Microsoft Windows images, you must create a Windows Driver Collection or create appropriate Driversets. The driver collection is more convenient to use, but a Driverset gives you more control.

Images, driver collections, and driversets can be created using the CA Server Automation user interface, and the `dpmrsi` command line interface. See the CA Server Automation bookshelf for documentation.

## Background

After you deploy an image to a server, the server uses that image when it boots. To boot correctly, the image must contain the hardware drivers necessary for the hardware components of the server. The drivers needed to make UNIX and Linux images work across different hardware are usually included in the image you capture. Microsoft Windows images, however, do not include all of the supplemental drivers necessary to allow the DynaCenter Image Mobility feature to work; therefore you must collect the drivers manually and add them to the reusable Driver Collection or Driverset components.

## Windows Driver Collection

The Windows Driver Collection is a set of supplemental operating system device drivers and their required installation files for multiple hardware configurations. It is the central repository for supplemental drivers for all servers in the datacenter and is the default source of drivers when you deploy Windows images. When deploying an image to a server, DynaCenter copies drivers from the collection for the appropriate version of Windows. In most cases, you will want to use the Driver Collection as the most convenient way to manage drivers and take advantage of the DynaCenter Image Mobility feature.

## Windows Driversets

In rare cases, mixing drivers from different vendors in the Driver Collection can lead to conflicts and images might fail to recognize some hardware devices after being deployed. In these situations, you should use a more specific Driverset to override the files coming from the Driver Collection. A Driverset allows you to control exactly which drivers are added to an image while it is being deployed. Each Driverset is intended for use with the server from which it was captured and any other server with an identical hardware configuration, or subset thereof. That means that you must create a separate Driverset for each hardware configuration for which you do not want to use the Driver Collection.

## Create the Windows Driver Collection

The most common ways to create the Windows Driver Collection are:

- By importing driver files directly from the vendor's installation media.
- By importing driver files from a target server that is already running Windows.
- By importing from a user-prepared directory.

### Import from vendor media

The most common way to create a Driver Collection is directly from vendor media.

- On the MWS, mount the CD-ROM. From the CA Server Automation command window, type and instruct DynaCenter to read the media:

```
dpmrsi collect_drivers -media_path /media/cdrom  
<credentials>
```

### Import from a target server

You can create a Driver Collection by pulling the drivers from a server that is configured with all of the necessary drivers.

- From the MWS, type:

```
dpmrsi collect_drivers -media_path <server_id>  
<credentials>
```

Where:

Arguments	server_id	The server ID used to register the server with the MWS.
-----------	-----------	---

### Import from a user-prepared directory

You can add drivers to the Driver Collection by copying all of the driver files that you want to add to the collection into a manually created directory. After you copy all of the files into the directory, you collect the drivers from the directory.

#### Create the required directory structure

Before DynaCenter can collect any driver files, you must create the necessary directory structure for the files.

- On the MWS, create the following directory structure anywhere on the system:

Directory	Description
.../.../winDrv/	Directory used to indicate a driver file collection
.../.../winDrv/w2kx-y/	Directory that contains all PnP format driver files to be collected
.../.../winDrv/w2kx-y/ <driver_distribution_name>	(Optional) Directory used to separate driver distributions to avoid file name collisions

Where:

'x' is 3 or 8 for Windows Server 2003 or Windows Server 2008 respectively.

'y' is 32 or 64 to indicate the bitness of the drivers.

**Example:**

To make a collection of ATI drivers for both Windows 2003, 32-bit and Windows 2008, 64-bit systems, use the following commands to create the directory structure:

```
mkdir /mydrivers/winDrv
mkdir /mydrivers/winDrv/w2k3-32
mkdir /mydrivers/winDrv/w2k8-64
mkdir /mydrivers/winDrv/w2k3-32/ATI
mkdir /mydrivers/winDrv/w2k8-64/ATI
```

**Copy PnP format driver files into the file structure**

**Note:** As driver distributions vary, it might be necessary to perform some additional steps before you can copy the driver files into the structure. For example, you might have to extract the driver files.

- Copy the driver files from the source location into the appropriate user-prepared directory on the MWS.

**Example:** If the source files are on the MWS, use the copy command:

```
cp -r \
<path_to_source_driver_distribution_directory>/* \
<path_to_destination_driver_distribution_directory>
```

For example, to copy ATI drivers for both Windows 2003, 32-bit and Windows 2008, 64-bit systems, use the following commands:

```
cp -r /source_drivers/32/ATI/* \
/mydrivers/winDrv/w2k3-32/ATI
cp -r /source_drivers/64/ATI/* \
/mydrivers/winDrv/w2k8-64/ATI
```

**Collect the driver files**

To collect the driver files:

- In the CA Server Automation command window, type:

```
dpmrsi collect_drivers -media_<path <path_to_/winDrv_
directory> <credentials>>
```

**Capture Tip**

**Important:** Use this method with caution—if you do not associate all of the necessary driver files with the vendorid, your deployed server might not function correctly.

If the user-prepared directory structure contains **all** of the drivers needed for a particular type of deployment, you can ensure that DynaCenter deploys only these drivers to a server during a deployment operation.

When you collect the drivers, specify a value for the `vendorid` argument. You can then specify that `vendorid` value in a profile and reference the profile when you use the `dpmrsi deploy image` command to deploy an image. DynaCenter will only deploy the driver files associated with the specified `vendorid` and no extraneous drivers will be installed on the system.

## Create a Driverset

Because Driversets are based on a specific hardware configuration, they are always captured from a server that is running an image with the necessary drivers already installed.

After you manually install a working image on the server, you can initiate a capture of the supplemental drivers available on that server. The server reboots to run the agent image and captures all of the drivers. The server then reboots again to run the image you installed. At this point, the Driverset contains the drivers and can be used to provision other servers with the same hardware configuration.

- In the CA Server Automation command window, type the following command:  
`dpmrsi capture_driverset -name <driverset_name> [-desc <driverset_description>] -serverid <target server_id> | -macaddr <target MAC address> <CREDENTIALS>`

Where:

Options	desc	Helpful comment about the Driverset
	name	User-specified name for the Driverset
Arguments	server_id	The server ID used to registerHostname or IP address of the server with the MWS

## Driverset naming conventions

Consider the following as you create and name a Driverset:

- Names can contain alphanumeric characters, spaces, tabs, and the following special characters:
  - period (.)
  - hyphen (-)
  - equal sign (=)
  - plus sign (+)
  - underscore (\_)
- Names must end with an alphanumeric character or an underscore (\_)
- Include the bitness of the Driverset in the name. As the bitness of a Driverset must correspond to the bitness of the image that is being deployed, indicating

the bitness in the Driverset name will make it easier to deploy the correct Driverset with an image.



# Chapter 2: Working with Hypervisors and Virtual Machines

---

**Important:** This Chapter refers to virtual machines in non-cloud environments.

DynaCenter can perform capture and deploy operations on pre-configured virtual machines, but it does not make new virtual machines or modify the hardware settings of existing virtual machines. Before DynaCenter can perform operations on a virtual machine, you must register the hypervisors and the virtual machines. For the most part, DynaCenter does not distinguish between physical and virtual servers. There are two situations however, where DynaCenter treats virtual servers differently from physical servers:

- Registering servers
- Assigning network IPs to virtual machines

Hypervisor registration is managed in the CA Server Automation user interface, and the dpmrsi command line interface. Registered hypervisors allow DynaCenter to scan them for all virtual machines and register them. See the CA Server Automation bookshelf for documentation.

## Vendor Considerations

This section outlines specific considerations when working with supported virtual machine vendors.

### VMware ESX

This section outlines specific considerations when working with VMware ESX.

#### General

- Virtual machines must have at least 768MB of RAM for deploy operations to complete successfully.
- Each virtual machine must have the first virtual NIC configured to be the 'boot' NIC. Additional NICs can be configured as necessary.

#### Windows images

- When capturing and deploying Windows images, DynaCenter supports virtual machines configured with paravirtualized devices such as the vmxnet2/vmxnet3 NIC drivers.
- During deploy operations, DynaCenter will install the VMware guest tools, which provide all the necessary drivers for the paravirtualized devices, to the target image.
- The target VM might reboot several times as part of the deploy process before the VM is ready for use.

- Windows images might require re-activation after deploy.

### Linux images

- Agent images for deploying Linux images do not contain the necessary drivers for using paravirtualized network or storage devices.
- DynaCenter currently does not install the VMware guest tools during the deploy of Linux images.
- The recommended virtual machine configuration for Linux virtual machines is:
  - Network devices: e1000
  - Storage: LSI Logic Parallel SCSI
  - Guest OS Type: Match OS as closely as possible (for example, RedHat Enterprise Linux 5 (64-bit))

### Solaris images

- Agent images for deploying Solaris x86 images do not contain the necessary drivers for using paravirtualized network or storage devices.
- DynaCenter does not currently install the VMware guest tools during the deploy of Solaris images.
- The recommended VM configuration for Solaris VMs is:
  - Network devices: e1000
  - Storage: LSI Logic Parallel SCSI
  - Guest OS Type: Sun Solaris 10 64-bit

## Microsoft Hyper-V

This section outlines specific considerations when working with Microsoft Hyper-V.

### General

- Virtual machines must have at least 768MB of RAM for deploy operations to complete successfully.
- Each virtual machine must have one "Legacy" NIC as the boot NIC, and one or more paravirtualized NICs for normal operation.

**Note:** The Microsoft Hyper-V guest tools do not contain a legacy NIC driver for Windows Server 2003 64-bit. This means that after a Windows Server 2003 64-bit image is deployed to Hyper-V, the deployed image will not be able to communicate with DynaCenter to signal that the deploy is complete.

**Workaround:** Log on to the server to see if the image deployed successfully. If the image deployed successfully, the server will be operational and you will be able to perform deploy and offline capture operations

### Windows images

- During the deploy of Windows images, DynaCenter installs the Hyper-V guest tools into the target image so that it has all the necessary drivers to use the virtual network and storage devices.

- The target virtual machine might reboot several times as part of the deploy process before the VM is ready for use.
- Windows images might require re-activation after the deploy operation completes.

## Register a Hypervisor Controller

A hypervisor controller is the system that manages a group of virtual machines. In a DynaCenter environment that has numerous virtual machines, you should register each hypervisor controller so that DynaCenter has power management control of the virtual machines that run on that controller. DynaCenter needs power management control because, while DynaCenter is capturing images from or deploying images to virtual machines, it might need to boot the target server.

1. In the CA Server Automation command window, type:

```
dpmrsi register_hypervisor -id <Hypervisor Hostname> -user
<Hypervisor Username> -password <Hypervisor Password>
-hypervisor_type <Hypervisor Type HyperV or VMware>
<CREDENTIALS>
```

**Tip:** If the MWS is configured with multiple networks, the network information has to be supplied as part of running any provisioning tasks such as offline capture or deploy.

2. Confirm that the hypervisor registered successfully:

```
dpmrsi list_hypervisors <CREDENTIALS>
```

## Register VMs and VM Networks

If all of the networks that are available to a hypervisor are not available to the virtual machines that run on that hypervisor, you must register each virtual machine and specify which networks each VM has access to.

1. From the MWS command line, type:

```
dccmd register server [--arch=<architecture>] \
[--networks=<network>[,<networks=<network>]] \
[--interface=<níc_id>] [--name=<server_name>] \
[--shutdown-timeout=<timeout>] <MAC>
```

Where:

Options	--arch	Architecture of the server
	--networks	Comma-separated list of names or CIDR formatted addresses for each network the server can see
	--interface	ID (NIC number) for the managed boot interface. This value is 1 or greater.

	--name	ID for the server in the driving application. This value will become the server_id and will override the name assigned by DynaCenter.
	--shutdown-timeout	Number of seconds this server has to shutdown before a timeout situation occurs. The default is 600 seconds.
Arguments	MAC	MAC address of the server being registered, formatted as 6 pairs of hex digits separated by colons <b>Example:</b> 00:1e:c2:00:d9:e0

2. Confirm that the server registered successfully:

```
dccmd show server <server_name> --format=full
```

## Add Additional VMs to an Existing Hypervisor Controller

To add new VMs to an existing hypervisor controller you must reregister the hypervisor controller and let discovery find the VMs. See the register hypervisor command for details.

# Chapter 3: Expanding Your DynaCenter Environment

---

As your environment changes, you might need to manage servers on additional networks or add additional depots for storing captured images.

## Add a network that is associated with an interface on the MWS

Before DynaCenter can manage servers on additional networks, you must register each network so that the relationship between the managed servers and the networked resources used to capture and deploy images can be resolved.

If you have servers that are part of a client network associated with an interface on the MWS (internal network) and you want to manage those servers, you must register the network. You can define many managed networks with different IP ranges to manage servers on different subnet segments.

**Example:** You add a new NIC to the MWS because your company is expanding and you want to manage servers associated with that NIC.

To register an internal network, complete the following tasks:

1. In the CA Server Automation command window, type:

```
dpmrsi register_boot_network [-sc <SC_URL>] -name <network name> -interface <Interface Id> [-desc <network description>] [-gateway <gateway IP address>] [-address <IP address or range>] <CREDENTIALS>
```

2. Confirm that the network registered successfully:

```
dpmrsi list_networks [-sc <SC_URL>] <CREDENTIALS>
```

## Task 2: Register the depot

**Note:** This task is only required if the depot is on a WebDAV server that is not located on the MWS.

Before the servers on an internal network can connect to a depot that is external to the MWS, you must associate the network where the servers reside with the depot.

1. In the CA Server Automation command window, type the following command to register the depot:

```
dpmrsi register_depot [-sc <SC_URL>] -depot <depot name> -depotURL <depot URL> -networks <depot Location> [-desc <depot description>] [-access-identity <identity>] [-access-secret <secret>] <CREDENTIALS>
```

2. Confirm that the depot registered successfully:

```
dpmrsi show_depot [-sc <SC_URL>] -depot <depot name>  
<CREDENTIALS>
```

## Add a network that is not associated with an interface on the MWS

Before DynaCenter can manage servers on additional networks, you must register each network so that the relationship between the managed servers and the networked resources used to capture and deploy images can be resolved.

If you have servers on a network that is not part of a local network but that is routable to the MWS (an external network) and you want to manage those servers, you must register the network. You can define many managed networks with different IP ranges to manage servers on different subnet segments.

To register an external network, you must complete the following tasks:

Task	Description	Reference
1	Set up a DHCP server	Task 1: Set up a DHCP server (optional) on page 24
2	Create the necessary firewall tunnels on your site	Task 2: Create firewall tunnels on page 27
3	Register the external network with DynaCenter	Task 3: Register the external network on page 27
4	Register the depot used by the server	Task 4: Register the depot on page 27

### Task 1: Set up a DHCP server (optional)

**Note:** If you assign static IP addresses to managed servers on an external network, you do not need to complete this task.

When you manage servers on an external network and you dynamically assign IP addresses to those servers, the DynaCenter DHCP server cannot manage the PXE boot requests for those servers because the servers are not on the same subnet as the MWS where DynaCenter is installed.

In this situation you must configure an external DHCP server on **each** external network where target servers reside. You can set up an external ISC DHCP server or an external Windows DHCP server.

**Important:** It is strongly recommended that only one DHCP server is configured to answer PXE/DHCP requests on each subnet. When multiple DHCP servers are configured to answer PXE/DHCP requests on the same subnet, the agent image boot process can be negatively affected.

#### Set up an ISC DHCP server

1. If you are using the internal DHCP server that comes with DynaCenter to manage PXE boot requests and to assign addresses, go to Step 6.

2. If you are using an external DHCP server to manage all PXE boot requests and to assign all addresses, go to Step 3.

3. On the MWS, turn off the internal dhcp server:

```
service rdhcpd stop
```

4. On the MWS, confirm that the rdhcpd service is stopped:

```
service rdhcpd status
```

5. On the MWS, disable the internal dhcp server so that it does not automatically start:

```
chkconfig rdhcpd off
```

6. In the dhcpcd.conf file on the external DHCP server, configure the `next-server` parameter to point to the IP address of the MWS on the boot network.

7. In the dhcpcd.conf file on the external DHCP server, change the `filename` parameter to `/tftpboot/racemiboot.kpxe`.

8. Ensure that TFTP, HTTP, and HTTPS traffic can pass through all routers and or firewalls between the external network and the MWS.

9. If you have more than one DHCP server configured to answer PXE/DHCP requests (not recommended), in the dhcpcd.conf file on the external DHCP server, add the following lines:

```
option space ipxe;
option ipxe-encap-opts code 175 = encapsulate ipxe;
option ipxe.priority code 1 = signed integer 8;

option ipxe.priority 1;
```

Setting the `ipxe.priority` setting tells the agent image boot loader to prioritize DHCP responses from this DHCP server; this can mitigate agent image boot issues when there are multiple DHCP servers on the same subnet.

### Example

If the external network is 105.23.0.0/16, and the IP of the MWS is 10.4.1.1, the relevant section of dhcpcd.conf file would look like this:

```
subnet 105.23.0.0 netmask 255.255.0.0 {
    # Set next-server to the IP address of the DynaCenter MWS
    next-server 10.4.1.1;
    filename      "/tftpboot/racemiboot.kpxe";
```

### Set up a Windows DHCP server

1. If you are using the internal DHCP server that comes with DynaCenter to manage PXE boot requests and to assign addresses, go to Step 6.
2. If you are using an external DHCP server to manage all PXE boot requests and to assign all addresses, go to Step 3.
3. On the MWS, turn off the internal dhcp server:

```
service rdhcpd stop
```

4. On the MWS, confirm that the rdhcpd service is stopped:

```
service rdhcpd status
```

5. On the MWS, disable the internal dhcp server so that it does not automatically start:

```
chkconfig rdhcpd off
```

6. On the Windows DHCP server, open a command window.

7. Type netsh and press **Enter**.

8. Type dhcp server and press **Enter**.

9. Type add optiondef 60 PXEClient STRING 0 comment=option added for PXE support and then press **Enter**.

10. Type set optionvalue 60 STRING PXEClient and press **Enter**.

11. Close the command window.

12. Click **Start**→**Administrative Tools**→**DHCP**.

13. Expand the node for the DHCP server.

14. Right-click on the **Scope Options** folder for the MWS boot scope, and then click **Configure Options** in the context menu.

15. Select **066 Boot Server Host Name**, and then type the IP address of the MWS in the **String value** field.

16. Select **067 Bootfile Name**, and then type /tftpboot/racemiboot.kpxe in the **String value** field.

17. Click **Apply**.

18. Right-click the DHCP server node, click **All Tasks**→**Restart**.

19. Click **OK**.

## Task 2: Create firewall tunnels

If the external network you want to register is outside of your corporate firewall, you must open a port on your firewall to allow DynaCenter to communicate with the servers on the external network.

**Example:** Open port 4102 on the firewall and point it to port 443 on the MWS.

If your depot is being served from a server other than the MWS, you must also open a port on your firewall and direct traffic from that port to port 443 on the server where the depot is located.

**Example:** Open port 4103 on the firewall and point it to port 443 on the server where the depot is located.

**Note:** If the tunnels already exist, confirm that they are operational.

To confirm that DynaCenter can communicate with resources outside of the firewall:

- On the MWS, run the following command to confirm that DynaCenter can communicate with the server:

```
ping <hostname>
```

## Task 3: Register the external network

1. InOn the CA Server Automation command window, type:

```
dpmrsi register_ext_ external network [-sc <SC_URL>] -name  
<network name> -url <RSI_URL> [-desc <network  
description>] <CREDENTIALS>
```

2. Confirm that the network registered successfully:

```
dpmrsi list_networks [-sc <SC_URL>] <CREDENTIALS>
```

## Task 4: Register the depot

Before the servers on an external network can connect to the depot where captured images are stored, you must associate the network where the servers reside with the depot.

1. In the CA Server Automation command window,, type the following command to register the depot:

```
dpmrsi register_depot [-sc <SC_URL>] -depot <depot name> -  
depotURL <depot URL> -networks <depot Location> [-desc  
<depot description>] [-access-identity <identity>] [-  
access-secret <secret>] <CREDENTIALS>
```

2. Confirm that the depot registered successfully:

```
dpmrsi show_depot [-sc <SC_URL>] -depot <depot name>  
<CREDENTIALS>
```

## Configure Multiple Static VLANs

You can configure DynaCenter to accommodate multiple static VLANs to provision in environments where the servers are on multiple VLANs. This configuration includes creating a boot network, which appropriately assigns addresses to an image as it is moved from server to server with potentially different VLANs.

**Note:** The VLANs are not managed by DynaCenter and must be created and configured in advance.

To register a VLAN, complete the following tasks:

Task	Description	Reference
1	Register the VLAN with DynaCenter	Task 1: Register the VLAN on page 28
2	Register the depot used by the server <b>Note:</b> This task is only required if the depot is on a WebDAV server that is not located on the MWS.	

### Task 1: Register the VLAN

1. Type the following command window, type:

```
dpmrsi register_boot_ network [-sc <SC_URL>] [-locale  
<locale>] -name <network name> -interface <Interface Id>  
[-desc <network description>][-gateway <gateway IP  
address>][-address <IP address or range>]<CREDENTIALS>
```

2. Confirm that the network registered successfully:

```
dpmrsi list_networks <CREDENTIALS>
```

## Add Local Depot Storage

After the initial installation of DynaCenter, you might want to add an additional local depot for storing captured images.

1. Create the directory that is to be used as depot storage.
2. In the CA Server Automation command window,, type the following command to register the depot:

```
dpmrsi register_depot [-sc <SC_URL>] -depot <depot name> -  
depotURL <depot URL> -networks <depot Location> [-desc  
<depot description>] [-access-identity <identity>] [-  
access-secret <secret>] <CREDENTIALS>
```

3. Confirm that the depot registered successfully:

```
dpmrsi show_depot [-sc <SC_URL>] -depot <depot name>  
<CREDENTIALS>
```

## Add External Depot Storage

After the initial installation of DynaCenter, you might want to add an external depot for storing captured images.

The following table outlines the process for adding network attached storage for the captured image depot:

Task	Description	Reference
1	Confirm the storage server meets storage requirements	Task 1: Confirm storage server requirements on page 29
2	Set up the storage server	Task 2: Set up the storage server using the following option: <ul style="list-style-type: none"><li>▪ Add storage server with NFS/CIFS support (NAS without DAV support) on page 31</li></ul>
3	Register the depot	Task 3: Register the depot on page 33
4	Edit the oem.ini file	Task 4: Edit the oem.ini file on page 33

### Task 1: Confirm storage server requirements

Storage on an external device must meet the following criteria:

- Be a properly configured device that is routable from the MWS and the managed server(s) that will use it.
- Have the correct time set or have NTP configured.

### Task 2: Set up the storage server

When you use external storage for the captured image depot (recommended), you have the following options:

- Using a storage server that runs the Apache web server and is capable of supporting DAV (Distributed Authoring and Versioning)—Recommended
- Using a storage server that does not have DAV support

#### Add a storage server with Apache and DAV support

1. Log into the storage server as the root user.
2. Locate or create the directory that is to be used as depot storage.
3. Set the directory and file ownership to the httpd user (usually apache:apache).

**Example:** On a Red Hat system type, `chown apache:apache /path/to/depot`

4. Set the directory permissions to read/write for any user or group that will use the httpd process.

**Example:** Type, `chmod 755 /path/to/depot`

5. On the storage server, create a new file in the conf.d directory of the webserver with the following settings:

**Tip:** You can name the file anything, but a name such as depot.conf will be descriptive of the file function.

```
Alias </my_depot_alias/>  </path/to/depot/>
<Directory </path/to/depot>>
  DAV On
  AuthType Basic
  AuthName "DAV Restricted"
  AuthUserFile /path/to/password/file/<password_file>
  <LimitExcept GET HEAD OPTIONS>
    Require user <username>
  </LimitExcept>
</Directory>
```

Where:

Value	Description
my_depot_alias	A user-friendly alias for the depot <b>Note:</b> This can be the path to the depot if a user-friendly name is not important to you.
path/to/depot	The location of the depot on the storage server
path/to/password/file	The location of the apache password file where the user's password can be verified
password_file	The apache password file where the user's password can be verified
username	The username of the user authorized to access the depot directory

### Example

On a Red Hat system, create a new file called depot.conf in /etc/httpd/conf.d with the following settings:

```
Alias /my_depot/ /repo/images/
<Directory /repo/images>
  DAV On
  AuthType Basic
  AuthName "DAV Restricted"
  AuthUserFile /etc/httpd/repo.pwd
  <LimitExcept GET HEAD OPTIONS>
    Require user someuser
  </LimitExcept>
</Directory>
```

6. Type the following command to create or update the file used to store the authentication information for users authorized to access the depot:

```
htpasswd -bc /path/to/password/file/<password_file>
<username> <password>
```

**Example:** htpasswd -bc /etc/httpd/repo.pwd someuser  
somepassword

7. As a root user, type the following command to restart the apache server:

```
service httpd restart
```

### Add storage server with NFS/CIFS support (NAS without DAV support) .

When the external storage server supports NFS/CIFS and cannot be DAV enabled, the MWS can function as the DAV frontend to the storage server.

1. Log into the NAS as the root user.
2. Locate or create the directory that is to be used as depot storage.
3. Set the directory permissions to read/write for any user or group that will use the httpd process.
4. On the NAS, add the following line to the /etc/exports file:

```
/<mount point> <hostname or IP address of NAS> \
(rw, sync, hide, insecure, no_root_squash, secure_locks)
```

5. On the NAS, run the following command to reload the list of exported file systems:

```
exportfs -a
```

6. On the MWS, type the following command to create a mount point:

```
mkdir /depot
```

7. On the MWS, update /etc/fstab to include the following:

```
<hostname or IP of nas>:<path/to/depot/on/nas> \
/depot nfs defaults 0 0
```

8. On the MWS, type the following command to mount the depot:

```
mount -a
```

9. On the MWS, run the following command to confirm that the MWS server can access the share:

```
showmount -e [hostname or IP address of NAS]
```

The output should show the share for the captured image depot.

**Example:**

```
showmount -e <IP address of NAS>
Export list for <IP address of NAS>:
/depot <IP address of NAS server>
```

10. On the MWS, create a new file in /etc/httpd/conf.d with the following configuration:

**Tip:** You can name the file anything, but a name such as `depot.conf` will be descriptive of the file function.

```
Alias </my_depot_alias/>  </path/to/depot/>
<Directory </path/to/depot>>
    DAV On
    AuthType Basic
    AuthName "DAV Restricted"
    AuthUserFile /etc/httpd/<password_file>
    <LimitExcept GET HEAD OPTIONS>
        Require user <username>
    </LimitExcept>
</Directory>
```

Where:

Value	Description
<code>my_depot_alias</code>	A user-friendly alias for the depot <b>Note:</b> This can be the path to the depot if a user-friendly name is not important to you.
<code>path/to/depot</code>	The location of the depot on the storage server
<code>password_file</code>	The apache password file where the user's password can be verified
<code>username</code>	The username of the user authorized to access the depot directory

### Example

```
Alias /my_depot/ /repo/images/
<Directory /repo/images>
    DAV On
    AuthType Basic
    AuthName "DAV Restricted"
    AuthUserFile /opt/race/etc/http/auth/depot.pwd
    <LimitExcept GET HEAD OPTIONS>
        Require user someuser
    </LimitExcept>
</Directory>
```

11. On the MWS, type the following command to create or update the file used to store the authentication information for users authorized to access the depot:

```
htpasswd -bc /opt/race/etc/http/auth/repo.pwd someuser
somepassword
```

12. On the MWS, as a root user, type the following command to restart the apache server:

```
service httpd restart
```

### Task 3: Register the depot

To ensure that the new depot is visible to Dynacenter and any servers DynaCenter will manage, you must register the depot.

1. In the CA Server Automation command window,, type the following command to register the depot:

```
dpmrsi register_depot [-sc <SC_URL>] -depot <depot name> -  
depotURL <depot URL> -networks <depot Location> [-desc  
<depot description>] [-access-identity <identity>] [-  
access-secret <secret>] <CREDENTIALS>
```

2. Confirm that the depot registered successfully:

```
dpmrsi show_depot [-sc <SC_URL>] -depot <depot name>  
<CREDENTIALS>
```

### Task 4: Edit the oem.ini file

If you want the newly added depot to be the default depot, edit the oem.ini file. The oem.ini file defines the location of the depot so that DynaCenter can locate the depot when it performs operations.

1. On the MWS, open the following file:

```
/opt/race/share/conf/oem.ini
```

2. Edit the depot section of the oem.ini file as follows:

```
[depot]  
uri = depot://<depot_name_on_storage_server>
```

**Example:** uri = depot://mystorage



# Chapter 4: Customizing Images

---

## Overview

An image is a collection of files that holds the information gathered from the disk(s) on the system that was captured, the information gathered about the system that was captured, and the characteristics of the image (filesystem type, original partition size, etc.).

Sometimes, capturing or deploying an exact replica of a system is not appropriate. In these situations, you can use a profile to customize how an image is captured or deployed. Using a profile to customize a capture or deploy operation you can:

- Specify which storage on the server should be captured.
- Specify how the storage on the target server should be configured.
- Replace default values in an image with custom values such as the hostname, or the network addresses.
- Add supplemental files, scripts, and patches to an image to satisfy a unique deployment scenario by creating a component.
- Control exactly which drivers are deployed with an image (Windows images only).

This chapter contains the following sections, which provide information about using profiles to customize how an image is captured or deployed:

- The different types of profiles and how DynaCenter refers to storage and filesystems on a server so that you can create profiles that achieve the results you expect.  
**Reference:** See Section A: Understand Profiles, which starts on page 37.
- How to create a profile and sample profile statements.  
**Reference:** See Section B: Create a Profile, which starts on page 41.
- How to create a software component with examples.  
**Reference:** See Section C: Create a Software Component, which starts on page 55.



---

## Section A: Understand Profiles

---

A profile contains a list of statements that specify how the capture or deploy of an image should be customized. When you use a profile, the profile settings override the metadata associated with the image. You can create and store as many profiles as you need and then reference a specific profile when you deploy or capture an image.

This section provides some basic concepts about profiles so that you can build profiles that achieve your desired goals.

### Types of profiles

There are two types of profiles—profiles that control the behavior of the image capture operation and profiles that control the behavior of the image deploy operation.

#### Capture profiles

By default, DynaCenter captures the entire image for the operating system installed on the server; however, capture profiles allow you to specify exactly which data on a server should be part of the image captured from that server. By controlling which storage is captured, you can save space (smaller images) and capture images that can be used in more deployment operations. For example, if Disk 1 contains the operating system and Disk 2 contains a database, you might want the captured image to only contain the operating system disk. A capture profile allows you to specify that only Disk 1 should be captured.

#### Deploy profiles

Deploy profiles minimize the number of images you need to store in the depot by allowing you to customize a golden image for each unique deployment. For example, you can store a generic file server image and then, using a deploy profile, customize the hostname and IP of the target server based on where the image is deployed in the network.

**Important:** When you deploy an image, you must, at a minimum, use a profile that specifies unique networking information to avoid creating a server with a duplicate hostname and/or a duplicate IP address.

### Storage Layers

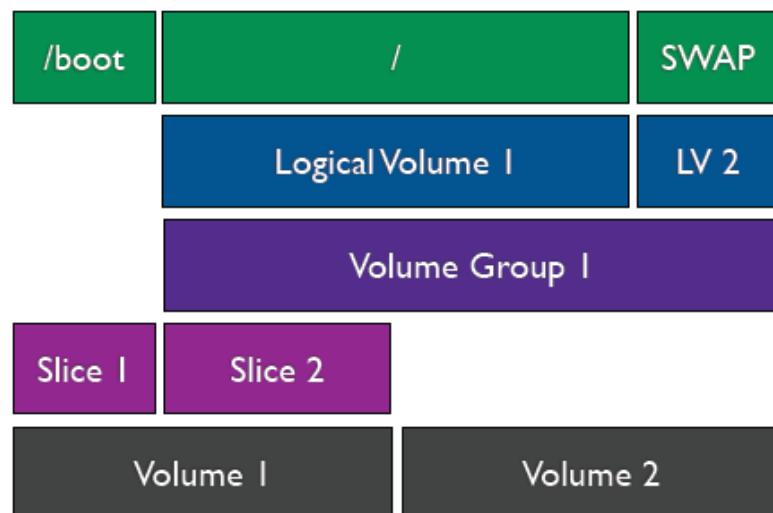
Before you can effectively use profiles to configure storage when you deploy an image to a server, you must understand how DynaCenter interprets server storage.

#### Linux storage layers

In Linux-like environments, DynaCenter uses five layers of storage to define how a filesystem is laid down on a server; the following table describes these layers:

Storage Layer	Description
Volumes	Disks or LUNS on the server
Slices	Partitions on the disk. A slice can be part of a disk or it can consume the whole disk. Slices are constrained by the characteristics of the physical disk.
Volume Groups	Collections of Logical Volumes, physical volumes (disks), and slices (partitions)
Logical Volumes	Partitions within the Volume Group
Filesystems	Collections of data that sit on a slice or a Logical Volume

The following diagram illustrates how these layers might be laid out on a Linux-based server:



When you deploy an image, you can specify how you want the storage laid out on the target server by defining, in a profile, the storage layers from the bottom up.

## Windows storage layers

In Windows environments, DynaCenter uses three layers of storage to define how a filesystem is laid down on a server; the following table describes these layers:

Storage Layer	Description
Volumes	Disks or LUNS on the server
Slices	Partitions on the disk. A slice can be part of a disk or it can consume the whole disk. Slices are constrained by the characteristics of the physical disk.
Filesystems	Collections of data <b>Note:</b> All filesystems must reside on their own slice.

The following diagram illustrates how these layers might be laid out on a Windows server:



When you deploy an image, you can specify how you want the storage laid out on the target server by defining, in a profile, the storage layers from the bottom up.

## Windows Filesystems

Before you can effectively use profiles to define which storage to capture from a Windows server you must understand how DynaCenter interprets the Windows filesystem layout.

The following table provides examples of how to identify a specific Windows filesystem given a particular disk layout:

Disk Layout	Mountpoint	Refers to
Disk 1 partition 1 partition 2 Disk 2 partition 1	E	Disk 2, partition 1
Disk 1 partition 1 Disk 2 partition 1 partition 2	E	Disk 2, partition 2
Disk 1 partition 1 Disk 2 partition 1 partition 2	D	Disk 2, partition 1



---

## Section B: Create a Profile

---

Profiles are represented differently depending on whether you are using the command line or web services. This section describes how to create profiles that are used by the command line tools. These profiles are defined in a text file using a simple descriptive language.

**Tip:** If your environment has a set of standard capture and deploy scenarios, you can create a profile for each scenario and make those profiles available to your end-users through the driving application.

### Create a Profile

You create a profile by listing, in a text file, all of the statements that will customize the image. After you create a profile, you apply the profile to a capture or deploy operation using the `--profile <profile_location>` option in either the `dpmrssi capture server` command or the `dpmrssi deploy` command. The statements defined in the profile override the information in the image to create a custom image. Any user defined profile options will supercede the profile statements.

You can combine any number of statements in a profile to create a simple or a very complex customization. The statements in a profile might include any or all of the following:

- Any custom storage settings.  
**References:** Storage capture profile statements, which start on page 42 and Storage deploy profile statements, which start on page 45.
- Any custom network settings.  
**References:** Network deploy profile statements, which start on page 43.
- Any drivers to deploy with an image (Windows images only).  
**Reference:** Miscellaneous deploy profile statements, which starts on page 51.
- Any supplemental components, such as additional applications, patches, or scripts to deploy with an image.  
**Reference:** Miscellaneous deploy profile statements, which starts on page 51.

### Sample Capture Profile Statements

This section provides samples of profile statements commonly used for capture operations.

**Note:** You can add descriptive comments, preceded with `#`, to explain the purpose of the statement.

## Storage capture profile statements

This section provides samples of profile statements commonly used to define which storage should be captured in an image.

### Capture all filesystems except a specific filesystem (Linux/UNIX)

By default, DynaCenter captures all of the filesystems on a server. If you do not want to capture a particular filesystem, you can exclude that filesystem from the capture operation.

```
# Capture all filesystems on the server except /home  
fs /home --exclude
```

**Note:** If your system has only the /, /boot, /home, and /opt filesystems, this capture profile would have the same result as Capture a subset of filesystems (Linux/UNIX) on page 42.

### Capture all filesystems/disks/drives except a specific filesystem/disk/drive (Windows)

By default, DynaCenter captures all of the filesystems on a server. If you do not want to capture a particular filesystem, you can exclude that filesystem from the capture operation.

```
# Capture all filesystems on the server except E:\  
fs E --exclude
```

**Important:** You must capture the drive where the operating system is installed. The profile statement `fs /` uses the `%SYSTEMROOT%` environment variable to capture the drive where the operating system files are located.

**Note:** If your system has only the C:\, D:\, E:\, and F:\ filesystems, this capture profile would have the same result as Capture a subset of filesystems/disks/drives (Windows) on page 43.

### Capture a subset of filesystems (Linux/UNIX)

By default, DynaCenter captures all of the filesystems on a server. If you want to capture only a subset of filesystems, you must name the specific filesystems you want to capture. When you name a specific filesystem to capture, DynaCenter only captures the filesystem that is specifically named; DynaCenter does not capture any filesystems mounted under the named filesystem.

```
# Capture the operating system, the kernel, and "applications"  
fs /  
fs /boot  
fs /opt
```

**Note:** Many UNIX variants put applications in the /opt hierarchy. This sample assumes that /opt is a separate filesystem.

**Note:** If your system has only the /, /boot, /home, and /opt filesystems, this capture profile would have the same result as Capture all filesystems except a specific filesystem on page 42.

### **Capture a subset of filesystems/disks/drives (Windows)**

By default, DynaCenter captures all of the filesystems on a server. If you want to capture only a subset of filesystems, you must name the specific filesystems you want to capture.

```
# Capture the operating system, D:\, and F:\ (but not E:\)  
fs /  
fs D  
fs F
```

**Important:** You must capture the drive where the operating system is installed. The profile statement `fs /` uses the `%SYSTEMROOT%` environment variable to capture the drive where the operating system files are located.

**Note:** If your system has only the C:\, D:\, E:\, and F:\ filesystems, this capture profile would have the same result as Capture all filesystems/disks/drives except a specific filesystem/disk/drive (Windows) on page 42.

## **Sample Deploy Profile Statements**

This section provides samples of profile statements commonly used for deploy operations.

**Note:** You can add descriptive comments, preceded with `#`, to explain the purpose of the statement.

### **Network deploy profile statements**

This section provides samples of profile statements commonly used to define the network settings to be used for a server when you deploy an image.

#### **Deploy an image with a new hostname**

```
# Deploy an image with a unique hostname  
hostname <name>
```

Where `<name>` is the fully qualified hostname that should be given to the target server.

**Important:** Ensure that each deployed image has a unique hostname by editing the hostname entry in the profile before you deploy an image.

**Note:** For Windows images, DynaCenter only supports hostnames that use ASCII characters. If the server where the image was captured from had a hostname that contained non-ASCII characters, use a deploy profile to specify a hostname that only has ASCII characters.

**Note:** For Linux and Solaris images, DynaCenter profiles support unicode strings for the hostname; however, because a-z, 0-9, dot, and dash are the only valid characters for a hostname, hostnames specified in unicode are encoded using IDNA. For example, the hostname 忍者.com would be converted to xn--c6t203e.com when the image is deployed to the target server.

### Deploy an image with new DNS settings

```
# Configure DNS settings for the image

dns [--domain=<name> | --search=<name>] \
[--nameserver=<ip_address>]
```

Where --domain is the domain name for the server, --search identifies a list that can be searched to identify the hostname, and --nameserver is the IP address of the server used for DNS queries. You can specify multiple --search values and --nameserver values by repeating the switch. If you specify multiple nameservers, they will be queried in the order they are listed.

### Deploy an image with controlled routing

```
# Control the routing of network traffic into and out of the
# server

route [--default | --destination=<addr> --netmask=<mask>] \
[--gateway=<addr>]
```

Where --default specifies that this is the default route, --destination --netmask identifies the IPv4 network subnet address of the destination, and --gateway is the IP address of the gateway associated with this server.

**Note:** You can omit the --netmask option if you use CIDR notation for the --destination option. For example, --destination 10.5/16.

### Deploy a two NIC capture to a one NIC target server

```
# Deploy a two NIC capture to a one NIC target server and
# specify a static network address for the image

interface 1

ip --interface 1 --address=192.168.1.3 --netmask=255.255.255.0
```

**Note:** You can specify the --address using CIDR notation; in this example, --address=192.168.1.3/24.

### Deploy a one NIC capture to a two NIC target server

```
# Deploy a one NIC capture to a two NIC target server and
# specify the interface should respond to requests by DHCP

interface 1

ip --interface 1 --dhcp

interface 2

ip --interface 2 --dhcp
```

**Note:** The interface number is the sequence number of the interface as it appears to the operating system, starting with "1". The number must be unique within a single profile.

### Storage deploy profile statements

This section provides samples of profile statements commonly used to define the storage configuration for a server when you deploy an image to that server.

#### Deploy a two disk capture to a two disk target server (Linux/UNIX)

**Note:** This sample illustrates how to place two filesystems on the same partition. If you wanted to deploy each filesystem to a dedicated slice, you could eliminate the use of volume groups and logical volumes shown in this sample.

```
# Deploy a two disk capture to a two disk Linux target server

# Create two disks on the target server

volume 1
volume 2

# Create two slices to fill the first disk; allow each slice
# to expand in size to consume available space on the volume;
# specify the minimum size of each slice in mebibyte (MiB)

slice 1 --volume 1 --expand --size=300
slice 2 --volume 1 --expand --size=30000

# Create a volume group to contain one logical volume for root
# and one logical volume for swap

vg MyVolGroup1 --slice 2

lv RootLV --vg MyVolGroup1 --size 5000
lv SwapLV --vg MyVolGroup1 --size 1300

# Create three filesystems—a boot filesystem on the first
# slice, a root filesystem in the root logical volume and a
# swap filesystem in the swap logical volume; specify the type
```

```

# of filesystem; specify which slice/logical volume will hold
# the filesystems

fs / --type ext3 --lv RootLV

fs /boot --type ext3 --slice 1

fs /swap --type swap --lv SwapLV

```

**Important:** The order of the slice statements determines the position of the slice in the partition table; the integers 1 and 2 do not determine the position of the slice, and each slice statement must have a unique integer. The order of the fs statements determines the order of the filesystem entries in the fstab file.

**Reference:** See Deploy an image to five partitions (Linux) on page 49 for information about how DynaCenter handles deploy operations when your profile specifies that you want more than four slices in one volume.

### Deploy a two disk capture to a two disk target server (Windows)

**Note:** In Windows environments all filesystems **must** reside on their own slice.

```

# Deploy a two disk capture to a two disk Windows target
# server

# Create two disks on the target server

volume 1
volume 2

# Create one slice to fill each disk; allow each slice to
# expand in size to consume available space on the volume;
# specify the minimum size of each slice in mebibyte (MiB)

slice 1 --volume 1 --expand --size=30000

slice 2 --volume 2 --expand --size=30000

# Create a filesystem on each slice; specify the type of
# filesystem; specify which slice will hold the filesystem

fs / --type ntfs --slice 1

fs E --type ntfs --slice 2

```

**Important:** The order of the slice statements determines the position of the slice in the partition table; the integers 1 and 2 do not determine the position of the slice, and each slice statement must have a unique integer. The order of the fs statements determines the order of the filesystem entries in the fstab file.

**Reference:** See Deploy an image to five partitions (Windows) on page 50 for information about how DynaCenter handles deploy operations when your profile specifies that you want more than four slices in one volume.

### Deploy a two disk capture to a one disk target server (Linux/UNIX)

```
# Deploy a two disk capture to a one disk Linux target
# server

# Create one disk on the target server

volume 1

# Create two slices; allow one slice to expand in size to
# consume available space on the volume; specify the minimum
# size of the slices in MiB

slice 1 --volume 1 --size=300

slice 2 --volume 1 --expand --size=5000

# Create a filesystem on each slice; specify the type of
# filesystem; specify the slice that will hold the filesystem

fs / --type ext3 --slice 2

fs /boot --type ext3 --slice 1
```

**Important:** The order of the slice statements determines the position of the slice in the partition table; the integers 1 and 2 do not determine the position of the slice, and each slice statement must have a unique integer. The order of the fs statements determines the order of the filesystem entries in the fstab file.

### Deploy a two disk capture to a one disk target server (Windows)

**Note:** In Windows environments all filesystems **must** reside on their own slice.

```
# Deploy a two disk capture to a one disk Windows target
# server

# Create one disk on the target server

volume 1

# Create two slices; allow one slice to expand in size to
# consume available space on the volume; specify the minimum
# size of each slice in MiB

slice 1 --volume 1 --size=5000

slice 2 --volume 1 --expand --size=30000

# Create a filesystem on each slice; specify the type of
# filesystem; specify the slice that will hold the filesystem

fs / --type ntfs --slice 1

fs E --type ntfs --slice 2
```

## Deploy a two partition capture to a three partition target server (Linux/UNIX)

```
# Deploy a two partition capture to the first and third
# partitions of a three partition target server

# Create one disk on the target server

volume 1

# Create three slices on the disk; specify the minimum size of
# each slice in MiB

slice 1 --volume 1 --size=25000

slice 2 --volume 1 --size=5000

slice 3 --volume 1 --size=10000

# Create a filesystem on two slices; specify the type of
# filesystem; specify which slice will hold the filesystem

fs / --type ext3 --slice 1

fs /boot --type ext3 --slice 3
```

**Important:** The order of the slice statements determines the position of the slice in the partition table; the integers 1, 2, and 3 do not determine the position of the slice, and each slice statement must have a unique integer. The order of the fs statements determines the order of the filesystem entries in the fstab file.

## Deploy a two partition capture to a three partition target server (Windows)

```
# Deploy a two partition capture to the first and third
# partitions of a three partition target server

# Create one disk on the target server

volume 1

# Create three slices on the disk; specify the minimum size of
# each slice in MiB

slice 1 --volume 1 --size=25000

slice 2 --volume 1 --size=5000

slice 3 --volume 1 --size=10000

# Create a filesystem on two slices; specify the type of
# filesystem; specify which slice will hold the filesystem

fs / --type ntfs --slice 1
```

```
fs E --type ntfs --slice 3
```

**Important:** The order of the slice statements determines the position of the slice in the partition table; the integers 1, 2, and 3 do not determine the position of the slice, and each slice statement must have a unique integer.

**Reference:** See Deploy an image to five partitions (Windows) on page 50 for information about how DynaCenter handles deploy operations when your profile specifies that you want more than four slices in one volume.

### Deploy an image to five partitions (Linux)

```
# Deploy an image to five partitions

# Create a disk on the target server

volume 1

# Create one slice for each partition; allow each slice to
# expand in size to consume available space on the volume;
# specify the minimum size of each slice in mebibyte (MiB)

slice 1 --volume 1 --expand --size=300
slice 2 --volume 1 --expand --size=30000
slice 3 --volume 1 --expand --size=30000
slice 4 --volume 1 --expand --size=30000
slice 5 --volume 1 --expand --size=30000

# Create a filesystem on each slice; specify the type of
# filesystem; specify which slice will hold the filesystem

fs / --type ext3 --slice 2
fs /boot --type ext3 --slice 1
fs swap --type swap --slice 3
fs /usr --type ext3 --slice 4
fs /home --type ext3 --slice 5
```

**Important:** The order of the slice statements determines the position of the slice in the partition table; the integers 1, 2, and 3 do not determine the position of the slice, and each slice statement must have a unique integer. The order of the fs statements determines the order of the filesystem entries in the fstab file.

**Note:** Linux systems only support four partitions per disk so, when you deploy this image, DynaCenter will place slice four and above on an extended partition that is divided into logical partitions. The deployed image will be laid out as follows:

Disk Layout	Mountpoint
Primary Partition 1	/boot
Primary Partition 2	/
Primary Partition 3	swap
Extended Partition 4	
Logical Partition 5	/usr
Logical Partition 6	/home

### Deploy an image to five partitions (Windows)

```
# Deploy an image to five partitions

# Create a disk on the target server

volume 1

# Create one slice for each partition; allow each slice to
# expand in size to consume available space on the volume;
# specify the minimum size of each slice in mebibyte (MiB)

slice 1 --volume 1 --expand --size=30000

slice 2 --volume 1 --expand --size=30000

slice 3 --volume 1 --expand --size=30000

slice 4 --volume 1 --expand --size=30000

slice 5 --volume 1 --expand --size=30000

# Create a filesystem on each slice; specify the type of
# filesystem; specify which slice will hold the filesystem

fs / --type ntfs --slice 1

fs E --type ntfs --slice 2

fs F --type ntfs --slice 3

fs G --type ntfs --slice 4

fs H --type ntfs --slice 5
```

**Note:** Microsoft Windows systems only support four partitions per disk so, when you deploy this image, DynaCenter will place slice four and above on an extended partition that is divided into logical partitions. The deployed image will be laid out as follows:

Disk Layout	Mountpoint
Primary Partition 1	/
Primary Partition 2	E

Primary Partition 3	F
Extended Partition 4	
Logical Partition 5	G
Logical Partition 6	H

## Miscellaneous deploy profile statements

This section provides samples of other profile statements commonly used for deploy operations.

### Deploy a Driveset with an image

#Deploy a custom Driveset with an image

driveset <name>

Where <name> is the name of the Driveset that contains the Windows device drivers to be deployed with an image.

**Reference:** Chapter 1: Working with Microsoft Windows Images, which starts on page 13.

### Deploy vendor-specific drivers from the Driver Collection

# Deploy only vendor-specific drivers from the driver collection

vendorid <vendor-id>

Where <vendor-id> is the name of the vendor assigned to drivers in the Windows Driver Collection.

**Reference:** Chapter 1: Working with Microsoft Windows Images, which starts on page 13.

### Deploy a component with an image

#Deploy a software component with an image

component <name>

Where <name> is the name of the component to be deployed with an image. More than one component can be specified in the profile; however, a profile can only contain components with the same OSType as the image.

**Reference:** Section C: Create a Software Component, which starts on page 49.

## Sample Complex Deploy Profile

This section provides a sample deploy profile to illustrate how multiple profile statements are joined to create a complex customization of an image.

```
# Sample Complex Deploy Profile
# Define networking information:
```

```

# Specify the hostname

hostname DELL-r200

# Create two interfaces:
# - first interface should use the DHCP server to obtain
#   IP address
# - second interface should use a specific IP address
#   and netmask
# - control the default routing of network traffic into
#   and out of the server

interface 1
ip --interface 1 --dhcp
interface 2
ip --interface 2 --address 192.168.10.171 --netmask
255.255.128.0
route --gateway 192.168.1.1 --default

# Define storage:
# - one disk
# - two slices/partitions
# - one volume group on second slice/partition
# - two logical volumes called root and swap in volume
#   group
# - three filesystems -- boot filesystem on first
#   slice/partition, root filesystem in root logical
#   volume and swap filesystem in swap logical volume

volume 1
slice 1 --volume 1 --size 300
slice 2 --volume 1 --size 7000
vg VG1 --slice 2
lv RootLV --vg VG1 --size 5000
lv SwapLV --vg VG1 --size 1300
fs / --type ext3 --lv RootLV
fs /boot --type ext3 --slice 1
fs /swap --type swap --lv SwapLV

# Include SSH certificate component

component RootSSHCertificate

```

## Test a profile

Profiles contain complex, related data but have only a loose structure. This means that it is possible to create profiles that do not make sense. You can test a profile to ensure the following:

- That the profile is not empty.

- That there is only one Driverset referenced in the profile.
- That a Driverset or any components referenced in the profile exist.
- That each profile statement is syntactically and semantically valid.

## Procedure

- On the MWS, type one of the following:

- To test a capture profile:

```
dccmd test captureprofile <profile_url>
```

- To test a deploy profile:

```
dccmd test deployprofile <profile_url>
```

Where:

Arguments	profile_url	Profile to validate
-----------	-------------	---------------------

DynaCenter uses standard output to list any errors with the profile.



---

## Section C: Create a Software Component

---

A software component is used to manipulate the content and behavior of an image by applying supplemental files, scripts, patches, etc. to the image during a deploy operation. A component can contain one or more files or binaries. All the necessary files and binaries can be contained in one component or they can be divided between several smaller components to provide flexibility.

### Set Permissions for Linux/Unix Components

When you install a component as part of a deployment it is important that the ownership and permissions for the files that make up the component are set appropriately. If you set the ownership and permissions appropriately when you create the component, you will not have to customize them on each target server where the component is installed. Use the chmod and chown commands to set directory and file permissions.

### Create a Component

When you create a component you create a source tree that contains all the directories and files for the component. All directories and files must have the appropriate permissions and ownership.

The following instructions are for creating one component with one file; examples later in this section show how to create more components with different files.

1. Download the file for the component.
2. Create a directory for the file.
3. Upload the file to the directory.
4. Install the file as a component:

```
rinstallsnapshot --description=<component_description> \
--no-bootable --os-type=<ostype> --prefix=<directory> \
--vendor=<vendor> <name> <source> <repository>
```

Where:

Options	--description	Helpful comment about the component.
	--no-bootable	Makes the component not bootable by itself.
	--os-type	Operating system of the component and the image it will be applied to. <b>Note:</b> Each component must have the same os-type as the image that will use it or the provision will fail. The <code>dcmd list ostypes</code> command lists valid operating system types. The format supports wildcards;

		however, it is always best to select the most specific value possible.
	--prefix	Directory where all the files for the component will be installed. Whatever is given as the source is assumed to be the top of the desired hierarchy. If no --prefix is given, only a root directory exists.
	--vendor	Vendor of the component <b>Note:</b> You can specify any custom value here. Any non-default value ensures that your component will be migrated if you ever upgrade DynaCenter.
Arguments	name	Descriptive name for the software component
	source	Source of the files for the component
	repository	Repository that will hold the component

5. Verify the component was installed and review its configuration:

```
rshowsnapshot <component_name>
```

6. Verify the executable file was created for the component:

```
rcommandprompt --component=<component_name>
ls <prefix>
exit
```

7. Add the component to the image profile using the component statement.

**Reference:** See Deploy a component with an image on page 51.

## Windows Component Example

This example creates two Windows components for Windows 2003 images.

1. Download the files for the components WinSCP and TightVNC for Windows 2003.
2. Create /root/install on the MWS with the following subdirectories: "W2k3\_WinSCP" and "W2k3\_VNC".
3. Upload the files to their respective subdirectories in /root/install.
4. Install the WinSCP component:

```
rinstallsnapshot --description="WinSCP for Windows 2003" --
no-bootable --os-type=Microsoft_Windows- \
"*_*_*-*" --prefix=/install/W2k3_WinSCP --vendor=WinSCP \
Win2K3_WinSCP /root/install/W2k3_WinSCP default
```

5. Verify the WinSCP component was installed and review its configuration:

```
rshowsnapshot Win2K3_WinSCP
```

6. Verify the executable file was created for the component:

```
rcommandprompt --component=Win2K3_WinSCP  
ls root/install/W2k3_WinSCP  
exit
```

7. Install the TightVNC component:

```
rinstallsnapshot --description="Tight VNC for Windows 2003" --no-bootable --os-type=Microsoft_Windows-*_*-* --prefix=/install/W2k3_VNC --vendor=TightVNC Win2K3_VNC /root/install/W2k3_VNC \ default
```

8. Verify the TightVNC component was installed and review its configuration:

```
rshowsnapshot Win2K3_VNC
```

9. Verify the executable file was created for the component:

```
rcommandprompt --component=Win2K3_VNC  
ls root/install/W2k3_VNC  
exit
```

10. Add the components to the image profile and include the profile when you deploy the image.

After the image is deployed, C:\install\W2k3\_WinSCP and C:\install\W2k3\_VNC will have been created.

## Linux Component Example

This example creates an RHEL4 component for RHEL 4 32-bit images.

1. Download htop for RHEL 4 32-bit.
2. On the MWS, create /tmp/rpm\_install/RHEL4\_32bit\_htop.
3. Upload the file to the /tmp/rpm\_install/RHEL4\_32bit\_htop directory.
4. Use the chmod and chown commands to set the permissions and ownership of the directories.
5. Create the RHEL4 component:

```
rinstallsnapshot --description="htop for RHEL4 32-bit" \ --no-bootable --os-type=RedHat_Linux-EL_4_*-i686 \ --prefix=/tmp/install/RHEL4_32bit_htop --vendor=htop \ RHEL4_32bit_htop /tmp/rpm_install/RHEL4_32bit_htop \ default
```

The component file list will include entries for /tmp/, tmp/rpm\_install, and tmp/rpm\_install/RHEL4\_htop, each of which will have its permissions and ownership set for provisioning depending on what is given as the source.

**Note:** If no --prefix is specified, only a root directory exists.

6. Verify the component was installed and review the configuration:

```
rshowsnapshot RHEL4_32bit_htop
```

7. Add component statements to the profile and include the profile when you deploy the image.

# Chapter 5: General Maintenance

---

Periodically complete general maintenance procedures to ensure that the DynaCenter license remains up to date and the database is securely backed up or restored.

## Manage the DynaCenter License

DynaCenter is licensed via a cryptographically signed license with an expiration date. If your license expires, you must request a new license if you want to continue performing DynaCenter operations. This section describes how to examine your existing license, then request and install a new one.

### Examine your existing license

Check your existing license to see when it expires so that you can request a new license before DynaCenter operations are interrupted.

**Note:** All license operations are managed from the Linux shell via command-line tools.

1. Log in to the Management Workstation.
2. At the shell prompt, type `rshowlicense`, and then provide the DynaCenter administrative username and password when prompted.

You should see output like:

```
DynaCenter License Information
Issuer: Racemi
Issued To: Data Center
License Type: demo
Hostname: mws.yourcompany.com
Database Instance: demouser:mws.yourcompany.com:127.0.0.1
Issue Date: 2011-03-23 12:00:00.00
Expiration Date: 2011-04-22 00:00:00.00
Grace Period: 0 days
Component(s): DynaCenter
```

Your license might include different components and it might have a different license type; it will include a different expiration date.

Notice that this license has an expiration date with no grace period. This means that after the expiration date has passed, DynaCenter will refuse any operational instructions until a new license is provided. Your license might have a grace period, during which DynaCenter will still support normal operation, but will warn you that the license is expiring. After the license has expired, DynaCenter will not perform licensed operations.

## Request a new license

Before the expiration date of the license expires, request a new license from CA Technical Support so that you can continue performing DynaCenter licensed operations.

## Install a new license

After the license renewal is approved, you will receive an email with your new license. Review the content of the email to ensure that the information is correct before you install the license.

**Note:** The license update process should take less than five minutes.

1. Save the license file, dynacenterLicense.xml, that is attached to the email and place it on the MWS in the /opt/race/share/conf directory.
2. Log into the MWS as the root user.
3. Type the following command to stop all DynaCenter services:

```
dcctl stop
```

This will not disrupt any of your managed servers, but the system will not respond to new server boot requests while the license upgrade is in process.

4. Type the following command to load the new license:

```
/opt/race/bin/rimportlicensefile \
/opt/race/share/conf/dynacenterLicense.xml
```

5. Type the following command to start all DynaCenter services:

```
dcctl start
```

You should be back on line with the new license installed.

## Backup and Restore the DynaCenter Database and Repositories

Periodically back up the DynaCenter database and image repositories. The best practice is to back up your database and your image repositories at the same time.

### Back up the database

DynaCenter includes a command line tool that simplifies the database backup process. Running the tool will temporarily shut down DynaCenter services on the MWS, so do not run it while systems are trying to boot or while the MWS is committing changes to the database.

1. Open a command terminal on the MWS.
2. Use the following command to shut down the DynaCenter daemons and back up the database non-interactively:

```
/opt/race/share/conf/dbbacker.sh -y
```

The backup location is:

```
/opt/race/share/conf/dynacenter_database.${datestamp}.sql
```

**Note:** Run `dbbacker.sh -h` to obtain a full list of arguments and options for the script.

## Restore the database

**Note:** If the DynaCenter database does not already exist, create one using the `make_db` script:

```
/opt/race/share/conf/make_db
```

1. Open a command terminal on the MWS.
2. To restore the database, run:

```
psql dynacenter < backupfile.sql
```

The `backupfile.sql` file is the file created by `dbbacker.sh`.

## Back up the repositories

Your image repositories are the core of your DynaCenter installation, and they should be backed up in conjunction with the database. DynaCenter does not ship with a tool to backup repositories because storage configurations typically differ widely across deployments.

The default locations for DynaCenter repositories are:

- `/repo/I` , which stores metadata for DynaCenter agents and images
- `/repo/R` , which stores DynaCenter components and templates
- `/repo/images` , which stores captured images (the depot)

**Note:** While the default location for the depot is on the MWS, Racemi recommends using external storage for your depot. See the *Rapid Server Imaging Server DynaCenter Installation Guide* for more information.

Whenever you perform a database backup, you should create an archive of at least the `/repo/I` and `/repo/R` repositories using tar, cpio, or some similar tool and then secure the archives on durable media along with the database backup file that they correspond to. You should perform a repository backup when DynaCenter processes are under minimal load to avoid file contention.

**Note:** Do not use the Secure Copy Protocol (SCP) for copying files on a network as it does not retain any symbolic links in the files. You can use SCP to copy tar archives as tarring preserves the symbolic links.

## Back up metadata, component, and template repositories

To archive metadata for agents and images and components and templates stored on the MWS:

1. Log in to the MWS as root.
2. From the MWS command line, type:

```
/opt/race/bin/dcctl stop
```

3. Execute your preferred archive command.

**Example:**

```
cd /
tar cvzf 20100711metadata_arch.tgz repo/I
tar cvzf 20100711component_arch.tgz repo/R
```

4. From the MWS command line, type:

```
/opt/race/bin/dcctl start
```

### Back up the captured image repository (Depot)

**Note:** If you are not using external storage for your captured image depot, use the procedure in Back up metadata, component, and template on page 61 to archive your depot.

To archive the captured image repository (the depot) stored on external storage:

1. Log in to the external storage as root.
2. Execute your preferred archive command.

**Example:**

```
cd /
tar cvzf 20100711captured_images_arch.tgz repo/images
```

## Restore the repositories

Use the procedures in this section to restore the DynaCenter repositories.

### Restoring the metadata, component, and template repositories

1. Log in to the MWS as root.
2. From the MWS command line, type:  
`/opt/race/bin/dcctl stop`
3. Untar the archived file.

**Example:**

```
cd /
tar xvzf 20100711metadata_arch.tgz
tar xvzf 20100711component_arch.tgz
```

4. From the MWS command line, type:  
`/opt/race/bin/dcctl start`

### Restoring the captured image repository (depot)

**Note:** If you are not using external storage for your captured image depot, use the procedure in Restoring the metadata, component, and template repositories on page 62 to restore your depot.

To restore the captured image repository to external storage:

1. Log in to the external storage as root.

2. Untar the archived file.

**Example:**

```
cd /
tar xvzf 20100711captured_images_arch.tgz
```



# Chapter 6: Advanced Configurations

---

This chapter describes advanced DynaCenter configurations that might be needed if the default DynaCenter configuration is not appropriate for your environment.

## DynaCenter Provisioning Agent Daemon (DPAD)

The DynaCenter Provisioning Agent Daemon (DPAD) is a management service that communicates with the DynaCenter Management Workstation (MWS) to manage target servers. The DPAD is installed as part of an Agent Image, so in most situations there is no need to manually install a DPAD. When you no longer want DynaCenter to manage a server, you can remove the DPAD from the server.

### Install DPAD on Target Servers

The DynaCenter Provisioning Agent (DPAD) allows DynaCenter to recognize and collect information about a server. Before you can capture a server image, you must install a DPAD on the server. Use the procedure in this section to manually install a DPAD. For easy installation of dpad agents, install the optional product, CA ITCM, from the product media. See the *CA Server Automation Administration Guide*.

The DPAD installation packages are located in the following directory on the MWS:  
opt/race/mws/software/packages/agent.

The available DPAD packages are:

- **Windows:** dpsetup.exe
- **Linux:** racemi-blademgmt-linux-\*.rpm
- **Solaris:** racemi-blademgmt-solaris-xxx-<version>.pkg

Where xxx is either i86pc or sun4u.

### Procedure

1. Copy the appropriate package from the MWS to the target server.

#### Example:

On the target server, type the following command:

```
scp user@hostname_of_DynaCenter_Appliance:\  
/opt/race/mws/software/packages/agent\  
/racemi-blademgmt-linux-*.rpm \  
/tmp
```

**Note:** The hostname of the MWS must be DNS resolvable.

2. From the directory on the target server where you placed the package, install the package using the appropriate command:

- **Windows non-interactive:** Open the command prompt as an administrator and then type the following command:  
`dpsetup.exe /S /mwsurl=<"your MWS URL">`  
**Example:** `dpsetup.exe /S /mwsurl="https://10.54.1.1"`  
The DPAD is installed on the server; go to Step 6.
- **Windows interactive:** Double-click the `dpsetup.exe` executable file and respond to the installer wizard.  
The DPAD is installed on the server; go to Step 6.
- **Linux:** Type `rpm -Uvh racemi-blademgmt-linux-* .rpm`
- **Solaris:** Type `pkgadd -d racemi-blademgmt-solaris-xxx-<version>.pkg`  
Where `xxx` is either `i86pc` or `sun4u`.

3. On non-Windows systems, open the `dpad.ini` file with an editor such as `vi` from the following location on the target server:  
`/opt/race/etc/dpad.ini`
4. Edit the `mws_url` line to point to the MWS that manages the server where you installed the DPAD.  
**Example 1:** `mws_url = https://10.4.1.1`  
**Example 2:** `mws_url = https://<your_company>.com:4433`  
**Note:** Do not edit any other settings in the `dpad.ini` file.
5. Type the following command to restart the DPAD:  
`service dpad.rc restart`
6. Confirm that the DPAD installed correctly by checking that the server registers with DynaCenter:  
`dccmd show server <server_name>`  
The substatus should be Running.

## Remove a DPAD from a Target Server

When you no longer want DynaCenter to manage a server, you can remove the DPAD from the server. If the server was provisioned by DynaCenter, you must manually remove the DPAD as it will not be part of the native OS software packaging/database system.

### Windows

If the system **was** provisioned by DynaCenter:

1. Type `cmd /c net stop dpad`
2. Type `cmd /c sc delete dpad`
3. Type `rmdir /Q /S "c:\Program Files\Dynamic Center"`

If the system **was not** provisioned by DynaCenter:

1. Depending on the version of Windows you are running, go to **Programs→Uninstall a program**.
2. Right-click the DynaCenter dpad and click **Uninstall**.

## Linux

If the system **was** provisioned by DynaCenter:

1. Type `/etc/init.d/dpad.rc stop > /dev/null`
2. Type `/bin/rm -rf /etc/rc.d/rc6.d/K98dpad \ /etc/rc.d/rc3.d/S25zdpad /etc/rc.d/rc0.d/K98dpad \ /etc/rc.d/rc5.d/S25zdpad /etc/rc.d/rc1.d/K98dpad \ /etc/rc.d/rc4.d/S25zdpad /opt/race \ /var/log/dynacenter /dpad.ini \ /etc/init.d/dpad.rc /.race_network /.raceinfo \ /.raceip /.raceslotid /.race_services`

If the system **was not** provisioned by DynaCenter:

- Type `rpm -e racemi-servermgmt-linux-*`

## Solaris

If the system **was** provisioned by DynaCenter:

1. Type `/etc/init.d/dpad.rc stop > /dev/null`
2. Type `/bin/rm -rf /etc/rc2.d/S73zdpad \ /etc/rc2.d/S73racemi_staticroutes /etc/rc0.d/K98dpad \ /etc/rc1.d/K98dpad /etc/rcS.d/K98dpad /opt/race \ /var/log/dynacenter /dpad.ini /etc/init.d/dpad.rc \ /.race_network /.raceinfo /.raceip /.raceslotid \ /.race_services`

If the system **was not** provisioned by DynaCenter:

- Type `pkgrm RACEdpad`

## DynaCenter Logging

DynaCenter logs all of its operations to files on the Management Workstation (MWS).

The default logging configuration should meet your needs; however, if you need to change the logging behavior, use the information in this section to reconfigure the settings.

### Log levels

The log level specifies the severity and threshold of a log message. The log levels listed below are in order from the most severe to the least severe; lower log levels include the higher ones, for example, the level **INFO** includes all levels up to **SYSMESG**.

Log Level	Description	Log Volume
SYSMESG	System messages, usually with information about server startup and shutdown	Lowest
CRITICAL	Messages that show critical or fatal errors	
ERROR	Error messages about failed operations, for example, the cause of a failed capture or provision	
WARNING	Messages about errors other than those that caused an operation to fail or abort	
USER	High-level messages that show the progress of a current operation	
INFO	Specific messages that show the progress of a current operation	
DEBUG	Verbose messages used by Support to troubleshoot issues	Highest

### Configure log levels

There are various loggers that send their log messages to the DynaCenter Logging Daemon (DLAD). The loggers determine which messages are sent to the DLAD based on the threshold specified in the logclient.ini file. By default, the loggers log all messages at level DEBUG and higher so that all messages are sent to the DLAD. The DLAD determines which messages are written to the log files based on the threshold specified in the logserver.ini file.

Configure logging in the following ways:

- To specify which log messages the loggers send to the DLAD, edit the log level in the logclient.ini configuration file.
 

**Important:** Racemi does not recommend changing the default setting unless the default DEBUG log level is so verbose that it is causing performance issues.

**Reference:** See Configure the logging level for a logger on page 68.
- To specify which log messages the DLAD writes to the log files, edit the log level in the logserver.ini configuration file.
 

**Reference:** See Configure the logging level for the DLAD on page 69.

### Configure the logging level for a logger

**⚠ Caution:** Changing the logger logging level from the default DEBUG setting can result in the loss of important information.

1. On the MWS, open the logclient.ini file from the following location with an editor such as vi:  
`/opt/race/share/conf/logclient.ini`
2. Locate the section for the logger that you want to change the log level for.

**Example:** To change the log level for the task logger, locate the [logger\_task] section.

3. Change the `level=` line to reflect the new log level.

**Example:** To change the task logger level from DEBUG to USER, edit the line `level=DEBUG` to `level=USER`.

4. Save and close the file.
5. Type the following command to restart DynaCenter services:  
`dcctl restart`

### Configure the logging level for the DLAD

1. On the MWS, open the `logserver.ini` file from the following location with an editor such as vi:

`/opt/race/share/conf/logserver.ini`

2. Locate the section for the logger that you want to change the log level for.

**Example:** To change the log level for the task logger, locate the [logger\_task] section.

3. Change the `level=` line to reflect the new log level.

**Example:** To change the task logger level from INFO to USER, edit the line `level=INFO` to `level=USER`.

4. Save and close the file.
5. Type the following command to restart DynaCenter services:  
`dcctl restart`

## Log files

DynaCenter generates several log files that you might need to reference. The log files are written in `/var/log/dynacenter`.

### Task log

DynaCenter writes logs that are related to tasks (such as capture and provision tasks) to `/var/log/dynacenter/task.log`. Each task-related log message contains the task ID of the operation.

Command line programs that start tasks print the task ID to stdout when run with the `--no-wait` or `-v` options.

### Other log files

DynaCenter creates several other log files in `/var/log/dynacenter` that you might need to reference for troubleshooting purposes:

Log File	Description
<code>configure.log</code>	Contains information related to the DynaCenter installation process

daemon.log	Contains messages from the various daemons about non-task-related processes
dc-install.log	Contains information related to the DynaCenter installation process
install.log	Contains information related to the DynaCenter installation process
handler.log	Failsafe log if there is a problem with the log system
Pyro_log	Contains debugging information related to the internal communication mechanism
userlogs	Directory that contains logs from user-run command line programs such as dccmd

## Log file backups

DynaCenter automatically backs up log files in one of the following ways:

- install.log—this log file is backed up every time you install DynaCenter. A timestamp is appended to the file name of the older log file to prevent older files from being overwritten.
- All other log files (except Pyro\_log)—these log files are rotated when they reach 100MB. When the file reaches 100MB, it is renamed to <name>.log.1 and any existing log files of the same type are incremented by one. DynaCenter stores a maximum of ten backups, so, when the log files are rotated, <name>.log.10 is deleted and <name>.log.9 becomes <name>.log.10.

# Chapter 7: Troubleshooting

---

This chapter identifies issues you might encounter as you use DynaCenter.

## Simultaneous Operations Fail to Complete

### Problem

When you initiate simultaneous operations from a DynaCenter instance that is running on a virtual machine, the operations fail to complete successfully.

### Background

In a production environment the MWS (the system where DynaCenter is installed) should be a dedicated, physical system that at least meets the minimum specifications listed in the *Rapid Server Imaging Server Installation Guide*.

In a non-production environment, however, you might have decided to use a virtual machine for the MWS and it is possible that insufficient resources have been allocated to the MWS VM.

### Solution

Before you call CA Support for assistance, try the following:

- Allocate additional resources to the MWS VM and retry the DynaCenter operations.
- Temporarily shut down all of the other VMs that are running on the same server and retry the DynaCenter operations.

If the operations succeed after the additional resources are available to the MWS VM, consider the following:

- Permanently allocate additional resources to the MWS VM.
- Use a dedicated, physical system for the MWS.

If the operations do not succeed after the additional resources are available to the MWS VM, call CA Support.

## "Windows was improperly shutdown" Popup on Newly Provisioned Windows System

### Problem

You provision a Windows system from an image that was obtained through a live capture and you see a "Windows was improperly shutdown" popup message when you log in to the provisioned system for the first time.

### Background

This occurs because the registry settings saved at the time of the live capture reflect a running system. When the image is deployed on a new disk and the system is started, Windows interprets the startup as an indication that an improper shutdown occurred.

### Solution

You can safely ignore the popup and allow the system to start.

## IBM Windows Provision is Missing Drivers

You deploy a Windows image to an IBM server using drivers collected from the IBM Server Guide but a number of devices show exclamation points or question marks after the provision completes.

### Background

The IBM Server Guide media includes drivers that are not distributed in Windows PnP format (that is, they are inside a Windows .exe file). As such they are not currently recognized as driver files.

### Solution

If this occurs, do one of the following:

- Collect the missing IBM drivers directly from a server where they are installed.
- Manually extract the PnP driver files from the .exe files in the Server Guide and then create a file structure to use as input for the `dpmrsi collect_drivers` command or use a Driverset.

## Windows Devices Show Exclamation Points or Question Marks After Deploy

### Problem

You deploy a Windows image but a number of devices show exclamation points or question marks after the provision completes.

### Solution

If this occurs and the Windows Driver Collection does not provide the desired results, try the following:

If...	Then...
the wrong driver was installed	<p>redeploy the image using a vendor identifier to specify which drivers should be selected from the Windows Driver Collection.</p> <p><b>Note:</b> Use the <code>vendorid &lt;vendor-id&gt;</code> statement in the deploy profile to specify a specific vendor.</p> <p><b>Reference:</b> See Deploy vendor-specific drivers</p>

	from the Driver Collection on page 51. If that does not resolve the issue, capture a Driverset with the appropriate drivers.
no driver was installed	collect the drivers from a server that has the same devices and then redeploy the image. If that does not resolve the issue, capture a Driverset with the appropriate drivers.

## VM Does Not PXE Boot After Agent Image Assigned

### Problem

You assign an agent image to a virtual machine but the server never reboots into the agent image.

### Background

When you register a hypervisor you might register it with one or more networks. All of the virtual machines that run on the hypervisor will be registered with the same networks as the hypervisor. When an agent image is assigned to the virtual machine, the list of available networks is sorted and the first network in the list is chosen when an IP address is allocated to the agent image. Unfortunately, because all of the networks that are available to a hypervisor might not be available to a specific VM that runs on that hypervisor, the IP that is assigned to the VM might not be a valid IP for that VM.

### Solution

You must re-register the VM, specifying an appropriate network, and then assign an agent image.

1. From the MWS command line, type:

```
dccmd register server [--arch=<architecture>] \
[--networks=<network>[,<networks=<network>]] \
[--interface=<níc_id>] [--name=<server_name>] \
[--shutdown-timeout=<timeout>] <MAC>
```

Where:

Options	--arch	Architecture of the server
	--networks	Comma-separated list of names or CIDR formatted addresses for each network the server can see
	--interface	ID (NIC number) for the managed boot interface. This value is 1 or greater.
	--name	ID for the server in the driving application. This value will become the server_id and will override the name assigned by DynaCenter.
	--shutdown-	Number of seconds this server has to shutdown

	timeout	before a timeout situation occurs. The default is 600 seconds.
Arguments	MAC	MAC address of the server being registered, formatted as 6 pairs of hex digits separated by colons <b>Example:</b> 00:1e:c2:00:d9:e0

2. Confirm that the server registered successfully:

```
dccmd show server <server_name> --format=full
```

3. From the MWS command line, type:

```
dccmd assign agent <server_id> <ostype>
```

Where:

Arguments	server_id	Hostname or IP address of the server that is to receive the image
	ostype	String that describes the operating system to be deployed to the server. This value is used to pick the appropriate agent image, but the agent might not use the same operating system.

### Example

To register the server named v215 and assign an agent image:

1. Type `dccmd register server --arch=sun4u --name=v215 00:14:4f:c3:e0:4c.`

Where `sun4u` is the architecture of the server, `v215` is the server name, and `00:14:4f:c3:e0:4c` is the MAC address.

2. Type `dccmd assign agent IntelServer 'Microsoft_Windows-*'.`

Where `IntelServer` is the server ID and 'Microsoft\_Windows-\*' is the operating system type of the server.

## Cannot Communicate with Cloud Server After Capturing its Image

### Problem

Prior to initiating an offline capture of a cloud server, you could communicate with the server; after the capture operation, you cannot communicate with the server.

### Background

DynaCenter can only capture images of live cloud servers; offline capture is not supported in cloud environments. When you stop and then restart a cloud server, a new MAC address and IP address is assigned to the server. DynaCenter will be unable to communicate with the server due to the change in the network settings.

### **Solution**

To reestablish communication with the cloud server you must reregister the server with DynaCenter. To reregister the server, you will need the new MAC address that was assigned when the server was restarted.

## **Physical NICs not Functioning as Bonded NICs After Image is Deployed**

### **Problem**

After deploying an image that contained teamed (bonded) NICs, the physical NICs do not function as bonded NICs.

**Example:** When a physical NIC breaks or its cable is unplugged, traffic is not moved to another NIC in the bond.

### **Background**

When DynaCenter captures an image of a server with teamed NICs, it captures the addressing information of the bond but not the individual IP addresses of the bonded NICs or the bond type configured for the NICs. Because DynaCenter does not have the information needed to configure the bond on the system that the image is being deployed to, no bond exists between the NICs after the image is deployed.

### **Solution**

After you deploy an image that contains teamed NICs, you must use the configuration mechanism appropriate for the operating system to configure the bond between the interfaces on the system.

## **“Failed dependencies” Error When Installing DPAD on Linux 64-bit System**

### **Problem**

You attempted to install a DPAD agent on a 64-bit Linux system but the installation failed with an “error: Failed dependencies” message.

### **Background**

The DPAD agent package requires the 32-bit libraries that are contained in the glibc.i686 portion of the glibc package. These libraries are sometimes excluded from a 64-bit installation—particularly in cloud vendor environments.

### **Solution**

Install the required dependencies:

1. Type the following command to install the glibc.i686 package:  
`yum -y install glibc.i686`
2. Reinstall the DPAD package.

## "Temporary failure in name resolution" Error When Deploying an Image

### Problem

You attempted to deploy an image to a target server but the deploy operation failed with a runtime error that states "Failed to open dav://your\_depot.com/depot\_path/depot\_name: Temporary failure in name resolution" message.

### Background

When you register your image depot, you can specify the depot URL, which can be the hostname or the IP address of the server where the depot is located. If you used the hostname of the server when you registered the depot, DynaCenter must be able to resolve the hostname when it needs to connect to the depot.

### Solution

Do one of the following:

- Ensure that the agent image that is running on the target server can connect to your name server.
- Use the `<networkId>=<depotURL>` option in the `dcmd register depot` command to specify the IP address of the network that contains the managed server and the URL of the depot when accessed from that network.

## Target Server Storage Does Not Match Source Image/Profile Specified Storage

### Problem

You deployed an image to a target server, but the storage on the deployed server does not match the storage on the source image or the storage specified in the deploy profile.

- The source image has more than four slices in one volume
- You specified that you want more than four slices in one volume in the deploy profile

### Background

On Windows and Linux systems the total data storage space of a disk can be divided into at most four primary partitions, or alternatively three primary partitions and an extended partition. If your source image has more than four partitions/slices in one disk/volume or, if you specified that you want more than four partitions/slices in one disk/volume in the deploy profile, DynaCenter will place partition/slice four and above on an extended partition that is divided into logical partitions.

## "SSHException: Error reading SSH protocol banner" Error When Registering a Hypervisor

### Problem

You attempted to register a hypervisor with DynaCenter but the operation did not complete and you see a runtime error that states "Error in task SSHException: Error reading SSH protocol banner".

### Background

If the hypervisor has insufficient resources allocated to complete operations in a timely manner, the register hypervisor operation can timeout while waiting for the secure connection to be established.

### Solution

Before you call CA support for assistance, try the following:

- Allocate additional resources to the hypervisor and retry the DynaCenter operation.
- Wait for the hypervisor to complete other pending operations and then retry the DynaCenter operation.



## Appendix A: Support Matrices

### Environmental Support List

Operating System	Microsoft Windows			Linux			Solaris	
Operating System Variant	2003 Server	2008	Red Hat	SUSE	CentOS	SPARC	x86	
Supported Versions / Releases	SP2	R2	Enterprise 4.x & 5.x	SLES 10.x & 11.x	4.x & 5.x	10u6	10.x	
Migrate into Public Clouds	No	Yes	No	Yes	Yes	N/A	N/A	
Supported Public Clouds	N/A			Amazon EC2 (limited regions), GoGrid, Rackspace, Terremark			N/A	
Virtualization Support	VMware ESX 3.x, 4.x, and ESXi 4.1 u1; Citrix Xen 5.x, Hyper-V			VMware ESX 3.x, 4.x and ESXi 4.1 u1			N/A	
Supported Logical Volume Managers	Windows native, excluding GPT and Dynamic Disks			Linux LVM			Solaris Native	
P2P Capture/Deploy	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Cross-vendor Capture/Deploy (same processor family)	Yes	Yes	Yes	Yes	Yes	Yes	N/A	Yes
P2V Capture/Deploy	Yes	Yes	Yes	Yes	Yes	Yes	N/A	N/A
V2P Capture/Deploy	Yes	Yes	Yes	Yes	Yes	Yes	N/A	N/A
V2V Capture/Deploy	Yes	Yes	Yes	Yes	Yes	Yes	N/A	N/A
Heterogeneous V2V	Yes	Yes	No	No	No	No	N/A	N/A
Supported Storage (for managed servers)	DAS and SAN			DAS and SAN			DAS and SAN	

#### Supported server families:

- HP Proliant Racked servers (DLxxx)
- HP Proliant Tower servers (MLxxx)
- HP Proliant Blade servers
- Sun x86 Racked servers
- Sun SPARC Racked servers
- IBM X series
- IBM
- IBM BladeCenter PPC servers
- Dell PowerEdge Racked servers
- Cisco UCS B200 Blade servers (UCS 5108 Chassis)

### Hypervisor Environments

DynaCenter can provision images both between hypervisors of the same type and between hypervisors of different types.

## Operating system matrix

DynaCenter supports the following hypervisors and operating systems:

Hypervisor	Operating Systems					
	SLES 10.x, 11.x	CentOS 4.x	CentOS 5.x	RHEL 4.x, 5.x	Windows 2003	Windows 2008 R2
ESX 3.5	Yes	No*	Yes	Yes	Yes	Yes
ESX 4.x	Yes	Yes	Yes	Yes	Yes	Yes
ESXi 4.1u1^	Yes	Yes	Yes	Yes	Yes	Yes
Hyper-V	No	No	No	No	Yes	Yes

\*Vendor support issue

^ESXi servers registered with a Virtual Center server

## Provisioning matrix

DynaCenter supports provisioning between the following hypervisors:

Source	Target		
	ESX	Xen	Hyper-V
ESX	Yes	Yes	Yes*
Hyper-V	Yes*	Yes*	Yes*

\*Windows operating systems only

## Cloud Environments

### Vendor and operating system matrix

DynaCenter supports capture and deploy operations for the following operating systems in AppLogic Grid:

Cloud	Operating Systems				
	RHEL 4.x	RHEL 5.x	CentOS 4.x	CentOS 5.x	Windows 2008 R2
Amazon	Yes	Yes	Yes	Yes	No
AppLogic	No	5.4 and later	No	5.4 and later	Yes

### DynaCenter and cloud API matrix

DynaCenter releases are updated independently from cloud API releases and AppLogic Module releases. The following matrix lists which versions of DynaCenter support certain cloud API versions and AppLogic Module versions:

DynaCenter Version	Cloud API Version	AppLogic Module Version
4.0.0-x	2.0.0-x	N/A
4.1.0-x	2.1.0-x	1.0.0-x

## Unified Computing System Environments

DynaCenter has limited support for Cisco Unified Computing System (UCS) blade servers.

### Supported environment

DynaCenter was tested on a UCS environment with the following configuration:

- B200 M1 Blade Server mounted in a UCS 5108 Blade Server Chassis
- ixgbe – Intel 10Gb PCI Express NIC Driver
- Dual Serial Attached SCSI (SAS) drives in a striped and mirrored RAID configuration as well as non-RAID configurations



# Glossary

---

## Agent Image

While DynaCenter is capturing or deploying images, it may boot the server into an image that contains special support software. This image is called an “agent image”. Agent images are created in the repository and managed automatically by DynaCenter. They use network addresses allocated from the boot network for the boot group of the server. You can use an agent image to deploy a captured image to a server with no software installed on it.

**Reference:** Bare-metal Provisioning in the *DynaCenter OEM Integration Guide*.

## Base Component

A captured Operating System that is installed on the MWS as a software component and is used to create an agent image.

## Capture

The process of scanning an image already installed on a server and saving the data into an archive format.

## Captured Image

An archived form of a standard image. A captured image is identified with a name assigned by DynaCenter.

## Chassis

A chassis is a representation of a collection of servers. Chassis have a type that corresponds to a driver for controlling the servers on the chassis. Common types of chassis are VMware and Virtual (for standalone servers).

## Deploy

The process of moving a captured image from the depot to a target server for provisioning.

## Deployed Image

An image that has already been provisioned onto a server.

## Depot

Storage container for captured images. The depot must be routable to the MWS and to servers being managed by DynaCenter. There can be more than one depot.

## Directly Attached Network

Any local subnet attached to DynaCenter.

## Driving Application

The third-party vendor software that controls DynaCenter.

## Driverset

A collection of files captured from a server that is composed of supplemental operating system device drivers and their required installation files specifically tailored to a server's hardware configuration. A Driverset is intended for use with the server from which it was captured and any other server with an identical hardware configuration, or subset thereof.

**Reference:** Managing Windows Drivers in the *DynaCenter OEM Integration Guide*.

## External Network

Any network behind one or more routers or firewalls.

## Image

A complete software image (OS, applications, data) that can be booted and run on a server.

## Image Scaling

During image capture, the process whereby DynaCenter examines each filesystem to determine how much space is allocated to and used by the files and then computes the filesystem size to hold original contents while increasing the size based on consumption and availability.

## Managed Server

A server that is managed by DynaCenter for the purposes of capturing and deploying images.

**Note:** The terms managed server and target server are used interchangeably.

## Master Image

Part of an image template, the master image includes the files which will eventually be copied into the captured image. The master image is used by configure scripts in software components to apply global changes so those changes can propagate efficiently without having to be recomputed each time.

## MWS

The Management Work Station (MWS) runs DynaCenter and in some configurations serves as a storage location for the depot.

## OSType

DynaCenter uses an ostype string to identify the operating system name, version, and platform. These strings follow a specific syntax with well-known values in certain positions to make parsing easier. The syntax is rigid and the full specifier includes many details. Partner-specific scripts might be needed to convert another notion of operating system and hardware platform to an ostype before it is passed to DynaCenter.

**Reference:** `rinstallsnapshot --help` for more specific details on the ostype string.

## Profile

Settings that can be used to specify the storage to be captured for a server or that can replace the default configuration already in the image for deployment.

## Repository

Storage location known to DynaCenter that holds metadata for images, files that are part of software components, and other data necessary for provisioning and manipulating images.

## Server

A real or virtual server capable of running software images.

## Server Group

A server group is a container for servers used to control access through permissions and compatibility checks. When using DynaCenter in most OEM configurations, only one server group is needed, however more can be created.

## Software Component

A software component is used to manipulate the content and behavior of an image by applying supplemental files, scripts, patches, etc. to the image during deployment. A component can contain one or more files or binaries. One component can be created with all the needed files and binaries or several smaller components can be created to manipulate the image(s).

## Storage Server

Any device that contains storage. Storage servers will typically be devices that are large enough to store many captured images. The MWS functions as storage server that contains at least a repository of software components.

## Target Server

A server that is managed by DynaCenter for the purposes of capturing and deploying images.

**Note:** The terms target server and managed server are used interchangeably.

## Template

A DynaCenter template controls how images are created using software components and default profile information.

## Windows Driver Collection

A collection of supplemental operating system device drivers and their required installation files. The Windows Driver Collection is the central repository for supplemental driver files for all servers in the datacenter and is the default driver source for provisioning of Windows images.



# Index

---

## A

authentication  
  CLI, 11  
  depot, 30, 32

## B

back up  
  database, 60  
  depot, 62  
  repositories, 61  
backupfile.sql, 61  
bonded NICs, 75

## C

capture  
  profile, 37  
CLI  
  authentication, 11  
  help, 11  
  working with, 10  
cloud  
  register depot prerequisite, 27  
command  
  dbbacker.sh, 60  
  dccmd assign agent, 74  
  dccmd capture server, 41  
  dccmd collect drivers, 14, 15  
  dccmd deploy image, 16, 41  
  dccmd --help, 11  
  dccmd list ostypes, 55  
  dccmd register server, 21, 73  
  dccmd show network, 23  
  dccmd show server, 22, 66, 74  
  dccmd test captureprofile, 53  
  dccmd test deployprofile, 53  
  dcctl restart, 69  
  dcctl start, 60, 62  
  dcctl stop, 60, 61, 62  
  psql dynacenter, 61  
  racelogin --use-auth-cache, 11

racelogout, 11  
rcommandprompt, 56  
rimportlicensefile, 60  
rinstallsnapshot, 55  
rshowlicense, 59  
rshowsnapshot, 56  
command line interface. *See* CLI  
component  
  example, Linux, 57  
  example, Windows, 56  
conf.d, 31

## D

database  
  backing up, 60  
  backup location, 60  
  restoring, 61  
DAV  
  for storage server, 29, 31  
dbbacker.sh, 60  
dccmd assign agent, 74  
dccmd capture server, 41  
dccmd collect drivers, 14, 15  
dccmd command  
  help for, 11  
dccmd commands  
  list of, 11  
dccmd deploy image, 16, 41  
dccmd --help, 11  
dccmd list ostypes, 55  
dccmd register server, 21, 73  
dccmd show network, 23  
dccmd show server, 22, 66, 74  
dccmd test captureprofile, 53  
dccmd test deployprofile, 53  
dcctl restart, 69  
dcctl start, 60, 62  
dcctl stop, 60, 61, 62  
deploy  
  profile, 37  
depot  
  add additional, 28, 29

- authentication, 30, 32
- register location, 28, 29
- restore, 62
- DHCP server
  - on external network, 24
- dhcpd.conf, 25
- DPAD
  - failed installation, 75
  - Windows non-interactive install, 66
  - Windows silent install, 66
- DPAD package location, 65
- dpad.ini, 66
- driver collection, 13
  - add from file structure, 14
  - add from media, 14
  - add from server, 14
- driverset, 13
  - create, 16
  - naming conventions, 16
- DynaCenter
  - overview, 9
- dynacenterLicense.xml, 60

## E

- external network
  - firewall configuration, 27
  - ISC DHCP server, 24
  - Windows DHCP server, 25

## F

- file location
  - conf.d, 31
  - DPAD packages, 65
  - dpad.ini, 66
  - dynacenterLicense.xml, 60
  - logclient.ini, 68
  - logserver.ini, 69
  - make\_db, 61
  - oem.ini, 33

## G

- glibc.i686, 75

- help
  - CLI, 11
  - customer support, 10
  - dccmd command, 11
- hypervisor
  - register VM, example, 74
- hypervisor controller
  - definition, 21
  - register, 21, 22

## I

- ISC DHCP server
  - external network, 24

## L

- license, 59
  - examine existing, 59
  - install new, 60
  - renew, 60
- list
  - dccmd commands, 11
- log
  - file location, 69
- log files
  - backup, 70
  - other, 69
  - task log, 69
- log levels, 67
  - configure, 68
- logclient.ini, 68
- logserver.ini, 69

## M

- make\_db, 61

## N

- network
  - add, 23
  - add external, 24

## O

- oem.ini, 33

## P

profile  
  capture  
    exclude filesystem Linux, 42  
    exclude filesystem Windows, 42  
    subset of filesystems Linux, 42  
    subset of filesystems Windows, 43  
  capture type, 37  
  complex example, 51  
  components of, 41  
  create, 41  
  deploy  
    disk (more to fewer) Linux, 47  
    disk (more to fewer) Windows, 47  
  driverset, 51  
  five partitions Linux, 49  
  five partitions Windows, 50  
  NICs (fewer to more), 45  
  NICs (more to fewer), 44  
  partition (fewer to more) Linux, 48  
  partition (fewer to more) Windows, 48  
  software component, 51  
  specify DNS settings, 44  
  specify hostname, 43  
  specify routing, 44  
  two disk to two disk Linux, 45  
  two disk to two disk Windows, 46  
  vendor drivers, 51  
  deploy type, 37  
  purpose of, 35  
  sample  
    capture statements, 41  
    deploy statements, 43  
    network deploy statements, 43  
    storage capture statements, 42  
    storage deploy statements, 45  
  test, 52  
  types of, 37  
psql dynacenter, 61

## R

racelogin --use-auth-cache, 11  
racelogout, 11  
Racemi Customer Support, 10  
rcommandprompt, 56

## register

  depot location, 28, 29  
  hypervisor controller, 21, 22  
  VM, 21

## repositories

  /repo/I, 61  
  /repo/images, 61  
  /repo/R, 61  
  backing up, 61  
  backing up depot, 62  
  restoring, 62  
  restoring depot, 62

## restore

  database, 61  
  depot, 62  
  repositories, 62  
rimportlicensefile, 60  
rinstallsnapshot, 55  
rshowlicense, 59  
rshowsnapshot, 56

## S

software component  
  example, Linux, 57  
  example, Windows, 56  
storage server  
  DAV, 29  
  NFS/CIFS, 31  
  with Apache/DAV support, 29  
  without Apache/DAV support, 31  
Support, 10

## T

task log, 69  
  location, 69  
teamed NICs, 75  
troubleshooting  
  bonded NICs, 75  
  deployed image storage, 76  
  DPAD failed to install, 75  
  exclamation points, 72  
  glibc.i686 dependency, 75  
  improper shutdown message, 71  
  lost communication, 74  
  missing drivers, 72

name resolution error, 76  
operations fail, 71  
SSH protocol banner error, 77  
teamed NICs, 75  
VM not booting, 73

register, 21  
vendor considerations, 19  
VMware ESX considerations, 19  
VLANs  
multiple static, 28  
VM. *See* virtual machine

## V

vendors  
virtual machine, 19  
virtual machines  
Microsoft Hyper-V considerations, 20

## W

Windows DHCP server  
external network, 25  
Windows drivers, 13