# CA Server Automation

## Rapid Server Imaging Server for AppLogic Installation and User Guide

12.6

# 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.

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

## Chapter 9: Customizing Images



## Chapter 10: General Maintenance

# Preface

## About This Guide

This guide provides the information and procedures needed to install DynaCenter in your CA AppLogic environment. It also provides the procedures for maintaining and operating DynaCenter and its components after the installation.

While Racemi provides a robust programming interface for integrating applications with DynaCenter's image and server management capabilities, this document describes performing operations from the Command Line Interface that is available immediately after DynaCenter is installed.

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

## Audience

This document is intended for system and network administrators who perform installation, administrative, and operational tasks related to capturing images for provision to appliances in CA AppLogic environments using DynaCenter.

## Supplemental documentation

- *Rapid Server Imaging Server Administration Guide*
- *Rapid Server Imaging Server Release Notes*
- *Rapid Server Imaging Server Installation Guide*

## Need Help?

Contact CA Support with questions or comments.

## Working with the Command Line

This guide provides procedures for customizing 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 DynaCenter Appliance, 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 DynaCenter Appliance, run the DynaCenter CLI commands you need.

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

3. On the DynaCenter Appliance, type the following command:

   ```
   racelogout
   ```

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

## CLI Help

The dccmd program drives DynaCenter from the command line.

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

  ```
  dccmd –help
  ```

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

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

# Chapter 1: Introduction

## Overview

DynaCenter provides next-generation server imaging and rapid provisioning technology to capture server images from physical or virtual systems and deploy them across any compatible resource without the requirement to use identical hardware.

DynaCenter for AppLogic extends DynaCenter capabilities to support image-based provisioning to CA AppLogic 2.9.9 environments.

### Image-Based Provisioning

Server provisioning is the act of preparing a server for use by installing an operating system (OS), all required applications, and any necessary data.

Image-based provisioning is faster, more flexible, and significantly less error-prone than manual or script-based provisioning. The content of the disks on a server (the "image") is copied and stored in an image library (Depot) and from there the image can be written to the disks of the same server or to a different server. The image includes the operating system, applications (optional), and data. The provisioned server boots the same OS and has the same applications installed as the original server had when the image was captured.

## Architecture

DynaCenter automates the migration of existing applications from physical, virtual, or cloud servers into an AppLogic Grid. Images are captured from the source server and stored in the image depot. Those images can then be deployed from the image depot on to Singleton appliances in the AppLogic Grid as needed.

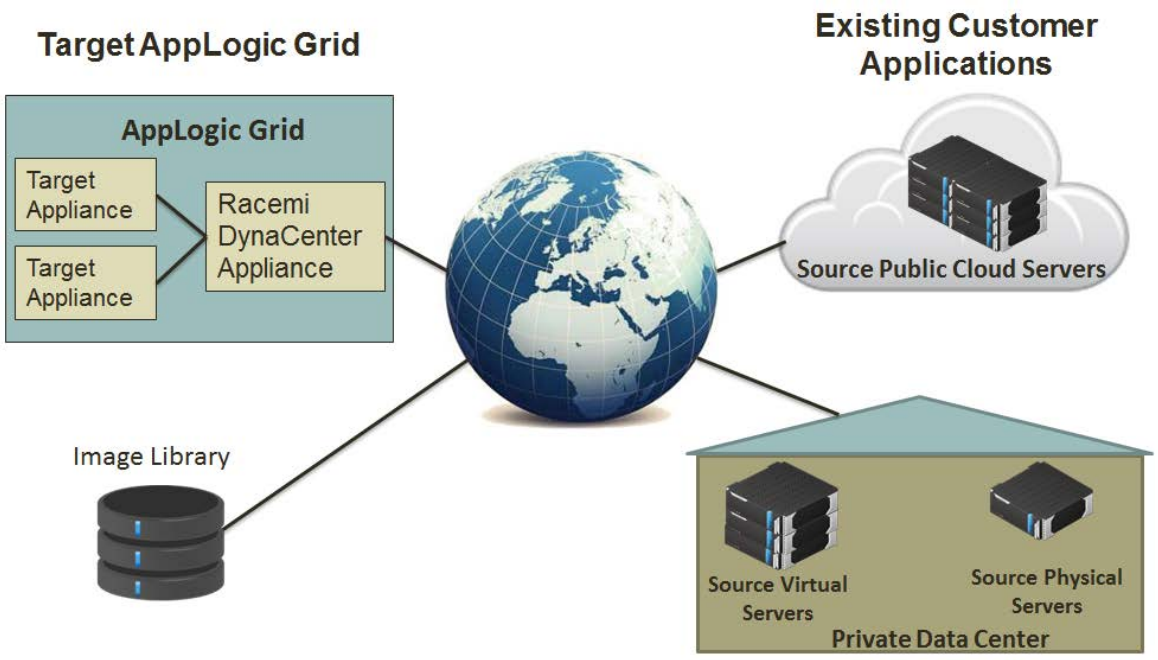
### Image Depot Storage

DynaCenter supports a flexible storage architecture that allows you to do any of the following:

- Use a storage server in the AppLogic Grid
- Use existing NAS or SAN within your datacenter
- Use public cloud storage
- Use portable storage in a remote site

  This option can resolve issues with network bandwidth when migrating large numbers of servers from a remote site to an AppLogic Grid.

## Architecture Diagram



# Part I: Installation

# Chapter 2: Before You Install DynaCenter for AppLogic

## Overview

This chapter describes the decisions and preparations you should make before you begin to set up your DynaCenter environment on your AppLogic grid.

**Important:** DynaCenter must be installed on an already installed and configured AppLogic grid. If your AppLogic grid is not installed and properly configured, you must complete that task before you attempt to install DynaCenter.

## Installation Checklist

Use the following checklist to ensure that your environment is installed and set up correctly.

| ✓ | Description | Reference |
|---|---|---|
| ☐ | Determine where you will store captured images | Storage Options on page 15 |
| ☐ | Determine the space you will need to store captured images | Storage Sizing Guidelines on page 16 |
| ☐ | Create a CentOS server on the AppLogic Grid | Create a DynaCenter Appliance on the AppLogic Grid, on page 16 |
| ☐ | Determine which DynaCenter installation option is best for your environment | DynaCenter Installation Options on page 21 |
| ☐ | Install DynaCenter (choose one installation option) | Basic installation, which starts on page 23 |
| | | Advanced Installation with External Storage for Depot (Recommended), which starts on page 27 |
| ☐ | Install the DynaCenter Cloud API | Install the DynaCenter Cloud API, which starts on page 31 |
| ☐ | Install the DynaCenter for AppLogic Module | Install the DynaCenter AppLogic Module, which starts on page 33 |

## Storage Options

DynaCenter needs a storage location for the server images you will capture. By default the captured image depot is located on the root filesystem of the server where DynaCenter is installed (DynaCenter Basic Installation); however, you should consider using an alternate storage location to ensure that there is enough storage

space to accommodate all of your data (DynaCenter Advanced Installation). The advanced installation option is the recommended installation method.

Before you install DynaCenter, understand which installation option is best suited to your environment.

**Reference:** DynaCenter Installation Options on page 21 explains the various installation options.

## Storage Sizing Guidelines

The amount of space you need to allocate for storage depends entirely on how many images you will need to store and how large those images will be. DynaCenter automatically compresses system images before it stores them, so you do not need a one-to-one data to storage ratio; however, increasing disk space upfront is inexpensive relative to running out of storage later. You can prepare a storage environment that is guaranteed to accommodate all your data with plenty of space by calculating the total disk usage of all the systems you will protect with DynaCenter. Or, if it makes sense for your environment, you might calculate your average data usage per system, and then use a storage device with N x A available space - where “N” is the number of systems you will need to capture and “A” is the average data requirement. Of course storage space can be added later if you need to capture additional systems.

## Create a DynaCenter Appliance on the AppLogic Grid

Before you can install your DynaCenter environment, you must create a DynaCenter Appliance. The DynaCenter Appliance is a simple appliance (server) in the AppLogic grid.

| Task | Description | Reference |
|---|---|---|
| 1 | Create the DynaCenter Appliance | Task 1: Create the DynaCenter Appliance on page 17 |
| 2 | Install the required packages on the operating system | Task 2: Install required packages on page 18 |
| 3 | Update the operating system | Task 3: Update the operating system on page 19 |
| 4 | Edit the /etc/hosts.template file | Task 4: Edit the /etc/hosts.template file on page 19 |
| 5 | Confirm that namespace servers were added to the /etc/resolv.conf file | Task 5: Confirm that the /etc/resolv.conf file updated on page 19 |

## Task 1: Create the DynaCenter Appliance

Use the following specifications to create the DynaCenter Appliance:

**Note:** For specific instructions on how to create an appliance, see your AppLogic documentation.

| Element | Sub-Item | Description | Value |
|---|---|---|---|
| Template | | Partially configured application used as a template for provisioning application instances | VDS64_CentOS54_r1 |
| Instance name | | Name for the appliance | User-specified value |
| Properties | hostname | Hostname to assign | User-specified value |
| | primary_ip | Primary IP address of the external interface of the appliance | User-specified value |
| | netmask | Network mask of the external interface | User-specified value |
| | gateway | Default IP network gateway for the external interface | User-specified value |
| | dns1 | Primary nameserver | User-specified value |
| | dns2 | Secondary nameserver | User-specified value |
| | user | User account to be created on provision of the application | User-specified value |
| | user_pw | Password of the user account to be created | User-specified value |
| | root_pw | Password of the root account | User-specified value |
| Resources | CPU | CPU defined as a fraction of a CPU | 2 |
| | Memory | Memory defined in gigabytes | 4G (minimum) |
| | Bandwidth | Bandwidth defined in megabits per second | 1000M<br>**Note:** This is a suggestion; throttle this setting to best suit your environment |
| | Storage | Size of the disk volume; | One of the following: |

| | | | |
|---|---|---|---|
| | | specify the volume name and the desired size | ▪ **Local image depot:** VDS64_CENTOS54.boot.size =80G<br>▪ **External image depot:** VDS64_CENTOS54.boot.size =15G |

**Example**

To provision from the AppLogic command line:

```
app provision VDS64_CentOS54_r1\
dynacenterapp hostname=dynacenterapp \
primary_ip=172.16.120.15 \
netmask=255.255.255.0 gateway=172.16.120.1 \
dns1=172.16.240.1 dns2=172.16.250.1 \
user=admin user_pw=dynacenter \
root_pw=dynacenter \
cpu.dflt=2 mem.dflt=4G bw.dflt=1000M \
VDS64_CENTOS54.boot.size=15G
```

## Task 2: Install required packages

When you provision the DynaCenter Appliance the template does not contain all of the packages that DynaCenter requires. You must install those additional packages before you install the DynaCenter software on the appliance.

1. As the root user, access the appliance over SSH using any suitable SSH client package.
2. Run `yum -y install <package>` for the following packages:
   - dhcp
   - expect
   - gmp
   - mod_ssl
   - perl-Crypt-SSLeay
   - popt.i386
   - postgresql-server
   - samba
   - tftp-server
   - webserver

### Task 3: Update the operating system and installed packages

The operating system and any installed packages need to be updated with the latest patches and security fixes.

1. If not already connected to the appliance, access the appliance as the root user over SSH using any suitable SSH client package.
2. Run `yum update` to update the appliance.

### Task 4: Edit the /etc/hosts.template file

1. If not already connected to the appliance, access the appliance as the root user over SSH using any suitable SSH client package.
2. Open the `/etc/hosts.template` file in a text editor such as vi.
3. Add the appliance hostname to the 127.0.0.1 line in the file.

   **Example:**

   ```
   127.0.0.1 dynacenterapp
   ```

4. Save your changes to the file.
5. Reboot the appliance.

### Task 5: Confirm that the /etc/resolv.conf file updated

When you created the DynaCenter Appliance you specified the nameservers the Appliance should use and the /etc/resolv.conf file should have been updated to reflect this.

1. If not already connected to the appliance, access the appliance as the root user over SSH using any suitable SSH client package.
2. Open the `/etc/resolv.conf` file in a text editor such as vi.
3. Were the nameservers added to the file?
   - If yes, go to Step 4.
   - If no, add the nameservers to the file and save the file.

     **Example:**

     ```
     nameserver 172.16.240.1
     nameserver 172.16.250.1
     ```

4. Close the file.

# Chapter 3: Install the DynaCenter Software

DynaCenter is a server imaging and rapid provisioning technology that captures server images from physical or virtual systems and deploys those images to any compatible resource without the limitation of requiring identical hardware. This chapter describes how to install the DynaCenter software on the DynaCenter Appliance.

## DynaCenter Installation Options

You can install DynaCenter using either of the following methods:

- Basic installation
- Advanced installation with external storage for depot (recommended)

Use the information in this section to help you decided which installation method is best for your environment.

### Basic installation

The basic installation places all DynaCenter components on the root filesystem on the DynaCenter Appliance.

#### When to use

Use this installation method for non-production environments or for environments where only a small number of servers will be managed.

**Reference:** See Section A: Basic Installation, which starts on page 23.

### Advanced installation

The advanced installation places all components except the captured image depot on the DynaCenter Appliance. The captured image depot must be set up on a separate storage server (recommended).

#### When to use

Use this installation method for production environments when an external storage server is available. This is the recommended installation method.

**Reference:** See Section B: Advanced Installation with External Storage for Depot (Recommended), which starts on page 27.

# Section A: Basic Installation

DynaCenter needs a storage location for the server images you will capture. The basic installation places the captured image depot on the root filesystem of the DynaCenter Appliance.

**Note:** This is not the recommended installation configuration for production environments.

## Install DynaCenter

1. Access the DynaCenter Appliance as the root user over SSH using any suitable SSH client package.
2. Mount or copy the DynaCenter installation media to a directory on the DynaCenter Appliance.
3. Change to the directory on the DynaCenter Appliance where you copied the installation media.
4. Type the following command:

   ```
   ./dc-install
   ```

   The installation program displays the following screen:

   ```
   DynaCenter Package Installation
   -------------------------------
   Welcome to the DynaCenter Package Manager. This program
   will guide you through the installation/upgrade process.
   ```

5. Choose and confirm the operating systems that you want DynaCenter to support.

   Each operating system that will be captured or provisioned needs a corresponding agent image. Agent images for Red Hat Linux, SUSE Linux, and Windows are provided by default; agent images for other operating systems are optional.

   **Note:** Select N (no) for the optional operating systems as DynaCenter for AppLogic does not currently support Solaris.

   ```
   Client Operating System Support
   -------------------------------
   DynaCenter supports capture and deploy of a variety of
   operating systems.  Support for Windows, RHEL, and SLES are
   included by default.  Select the additional operating
   systems that should be supported by this installation.
   Install support for Solaris-sun4u? (N) [Y, N]
   Install support for Solaris-i86pc? (N) [Y, N]
   No additional operating systems have been selected
   Continue with installation? (Y) [Y, N]
   ```

6. Review and confirm the list of RPM packages that will be installed or upgraded. These packages include DynaCenter binaries as well as other files that DynaCenter needs to operate.

```
Searching for installed packages.
The following packages will be upgraded/installed:
vmware-viperl-4.1.0-254719.x86_64.rpm
racemi-dynacenter-4.1.0-0.rpm
racemi-mws-4.1.0-0.rpm
racemi-help-4.1.0-0.rpm
racemi-comp-solaris-i86pc-3.2.0-0.rpm
racemi-python-2.7.1-12.rpm
ntfsprogs-2.0.0-1.i386.rpm
racemi-comp-solaris-sun4u-4.1.0-0.rpm
.
.
Continue with installation? (Y) [Y, N]
```

7. Choose and confirm the network interface for the management network.

   The management network is used for communication between the DynaCenter daemons. For a standalone DynaCenter Appliance, the interface for the Management Address should be on localhost (loopback).

```
Management Address
------------------
The MWS requires an address for accessing its management
services as well as remote access on the network. For
security, it is recommended, however not required, that
this address be on a network that is not exposed to managed
clients. The following list of addresses are configured on
this system and available for use.
0. 127.0.0.1 on loopback
1. 172.16.120.15 on eth0
Select management address (0) [0, 1, 2] 0
Selected: 127.0.0.1 on loopback
Accept and continue? (Y) [Y, N]
```

   **Note:** Do not set the management network to an IP that is in the range of the AppLogic public IP addresses.

8. Choose and confirm the client networks that contain servers that will be managed by the DynaCenter Appliance.

   A Client Network is a network interface card (NIC) or VLAN configured on the DynaCenter Appliance that is connected to the same network(s) as the managed servers. Managed servers are the systems that DynaCenter will be capturing and provisioning. The DynaCenter installer will scan the operating system for its network configuration and present each detected NIC or VLAN as an option in the "Client Networks" section. Note that more than one network can be specified during installation.

```
Client Networks
---------------
The MWS must be configured to manage servers on each of the
networks configured on its interfaces. For security, it is
recommended, however not required, that the interface used
for management is not included in the set of client
networks. The following list of networks are configured on
this system and available for use.
0. 172.16.120.0/24 on eth0
Select client network(s) (0) [0] 0
Selected: 172.16.120.0/24 on eth0
Accept and continue? (Y) [Y, N]
```

9. Specify and confirm the Agent Image Addresses on the client network that will be used by agent images during capture, provision, and driver capture operations.

   Agent Image Addresses are dedicated IP addresses on the Client Network that DynaCenter assigns to agent images during capture, provision, and driver capture operations. The number of IP addresses specified should be equal to or greater than the number of concurrent operations that will be performed.

   **Example:** If the user needs to run five captures and three provisions simultaneously, there should be at least eight IP addresses available in the Agent Image Address space.

```
Agent Image Addresses
---------------------
The MWS uses diskless agent images for offline capture and
deploy of managed servers. These agent images need an
address pool on each client network. By default, agent
images will use the first unused address available on a
given client network, however specifying a limited set of
addresses minimizes the potential for collision with other
servers which might become active while an agent image is
in use. Addresses may be specified as a comma or space
separated list of addresses or address ranges, such as:
10.10.20.2, 10.10.20.50-10.10.20.100, 10.10.20.200
Enter addresses for network 172.16.120.0/24
Current [172.16.120.0-172.16.120.255] 172.16.120.100-
172.16.120.105
Using addresses: 172.16.120.100-172.16.120.150
Accept and continue? (Y) [Y, N]
```

10. The installation begins.

    An ‘Installation complete’ message is displayed when the installation completes.

# Section B: Advanced Installation with External Storage for Depot (Recommended)

DynaCenter needs a storage location for the server images you will capture. This section describes how to install DynaCenter when you want to use an external storage server for the image depot. This is the recommended DynaCenter installation option.

The following table outlines the process for installing DynaCenter when using an external storage server for the captured image depot:

| Task | Description | Reference |
|---|---|---|
| 1 | Install DynaCenter | Task 1: Install DynaCenter on page 27 |
| 2 | Set up the storage server | Task 2: Set up the Storage Server using the following: Storage Server with Apache and DAV support on page 27 |
| 3 | Register the depot | Task 3: Register the Depot on page 29 |
| 4 | Edit the oem.ini file | Task 4: Edit the oem.ini file on page 30 |

## Task 1: Install DynaCenter

Install DynaCenter using the instructions provided in Section A: Basic Installation, which starts on page 23.

## Task 2: Set up the Storage Server

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

- Using a storage server that runs the Apache web server and is capable of supporting DAV (Distributed Authoring and Versioning)—Recommended

### Storage Server with Apache and DAV support

Storage on an external device must meet the following criteria:

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

#### Configure the storage server

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 (normally 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<br>**Note:** 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
   ```

## Task 3: Register the Depot

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

1. On the DynaCenter Appliance, type the following command to register the depot with the credentials you previously specified or your S3 credentials:

   ```
   dccmd register depot \
   [--access-identity=<id_or_user_name>] \
   [--access-secret=<secret_key>] \
   <depotId> <depotURL> \
   <networkId>=<depotURL> [<networkId>=<depotURL>...]
   ```

   Where:

| | | |
|---|---|---|
| Options | `--access-identity` | An authorized identity to access this storage server (for example, the username, the account, etc.).<br>**Note:** If you use this option, you must also define `--access-secret`.<br>**WebDAV storage:** This is a username of a user authorized to access the server. |
| | `--access-secret` | The password, key, or certificate associated with the authorized identity.<br>**WebDAV storage:** This is the password associated with the username specified for `--access-identity`. |
| Arguments | `depotId` | The name of the depot. |
| | `depotURL` | The URL of the network storage. It can be one of the following:<br>▪ **WebDAV storage:** dav://server- |

| | | |
|---|---|---|
| | | address/path |
| | networkId=<br>depotURL | Network that contains managed servers and the URL of the depot when accessed from that network |

**Example:**

```
dccmd register depot mystorage \
--access-identity=someuser --access-secret=somepassword \
dav://my_storage_server_hostname_or_IP/my_depot \
172.16.120.0/24=dav://172.16.120.3/repo/images
```

2. Confirm that the depot registered successfully:

   ```
   dccmd show depot <depotId> --format=full>
   ```

   **Example:**

   ```
   dccmd show depot mystorage --format=full
   ```

**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.

## Task 4: Edit the oem.ini file

The oem.ini file specifies the location of the depot. If you do not want to have to specify the depot location each time you perform an operation, you must edit the oem.ini file to point to the depot location on the storage server.

1. On the DynaCenter Appliance, 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>
   ```

# Chapter 4: Install the DynaCenter Cloud API

The DynaCenter Cloud API integrates a driving application with the cloud management capabilities of DynaCenter, which allows you to deploy images to appliances in an AppLogic Grid.

## Install the Cloud API

The Cloud API is delivered on a separate DVD.

1. Access the DynaCenter Appliance as the root user over SSH using any suitable SSH client package.
2. Type the following command to create an authenticated DynaCenter session:

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

   The system will prompt you for the username and password to use for this session.
3. Mount or copy the DynaCenter Cloud installation media to a directory on the DynaCenter Appliance.
4. Change to the directory on the DynaCenter Appliance where you copied the installation media.
5. Type the following command:

   ```
   ./cloud-install
   ```

   An “Installation successful.” message indicates that the installation has completed.
6. If you do not plan to install the AppLogic module at this time, type the following command to end the authenticated session:

   ```
   racelogout
   ```

# Chapter 5: Install the DynaCenter AppLogic Module

The DynaCenter AppLogic Module integrates the cloud management capabilities of DynaCenter with CA AppLogic 2.9.9, which allows you to deploy images to appliances in an AppLogic Grid.

## Prerequisite

The DynaCenter Appliance uses a public and private key pair to authenticate communication with the AppLogic Grid Controller. You must create a public/private key pair and then copy the public key to the AppLogic Grid Controller so that it can communicate with the DynaCenter Appliance.

1. On the DynaCenter Appliance, navigate to the `/root/.ssh` directory.
2. Type the following command to create the key pair:

   `ssh-keygen -v -f <key_name>`

   **Example:** `ssh-keygen -v -f applogic_key`

   **Note:** Do not provide a passphrase for the key. If you provide a passphrase here, you will be prompted for the passphrase before each DynaCenter operation is executed.

   **Tip:** Store the key pair in a safe location in case you need it in the future. For example, `/home/your_user/.ssh`.
3. Open the public key file in a text editor and copy the content of the file.
4. From the shell prompt, log in to the AppLogic Grid Controller.
5. Type the following command to put the public key on the AppLogic Grid Controller:

   `user set <username> sshkey="<content_of_public_key>"`

   Where `<username>` is a registered AppLogic Grid user or an AppLogic Grid user account created for use with DynaCenter and `<content_of_public_key>` is the pasted content of your public key file.

   **Example:** `user set jsmith@mycompany.com sshkey="ssh-rsa AAAAB3NzaC1yc2EAAAABJQAAAIEAjImKG3fwQT1VLOQcEGz1K3h/KnYEtOf L4SPCkyNG6EFGxNzWMYktapijMEyQ5wovGkTcsDt9C555J4dN9Dv65M7T2h ePOEEItBbY5uygvD7LPx8c8GQhytpR7oywQ30GI4xdIVNQg3QgolW3nnzVl pz3YsrGRYInWus4+W4PTEc=="`

   Placing the public key on the AppLogic Grid Controller authorizes the DynaCenter Appliance to pass commands to the Grid Controller.

## Install the AppLogic Module

The DynaCenter AppLogic Module provides the capability to manage AppLogic resources and is delivered on a separate DVD.

1. Access the DynaCenter Appliance as the root user over SSH using any suitable SSH client package.
2. If you do not already have an authenticated DynaCenter session running, type the following command to create an authenticated session:

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

   The system will prompt you for the username and password to use for this session.
3. Mount or copy the DynaCenter AppLogic Module installation media to a directory on the DynaCenter Appliance.
4. Change to the directory on the DynaCenter Appliance where you copied the installation media.
5. Type the following command:

   ```
   ./cloud-applogic-install \
   file:///root/.ssh/<name_of_private_key_file>
   ```

   **Example:** `./cloud-applogic-install \`
   `file:///root/.ssh/applogic_key`

   **Note:** The three forward slash characters are necessary for this command to work.

   An “Installation successful.” message indicates that the installation has completed.
6. Type the following command to end the authenticated session:

   ```
   racelogout
   ```

# Chapter 6: Configure DynaCenter

You can customize certain default DynaCenter settings that control the behavior for specific API commands, task options, and logging levels. This chapter describes DynaCenter files that you might want to customize for your environment.

## OEM Configuration File

The OEM configuration file defines the default behavior for specific API command and task options. This section describes how to edit the file to set the values that are appropriate for your DynaCenter environment.

### OEM Configuration File Sections and Options

The sections in the OEM configuration file relate to a specific OEM API command or task:

- chassis
- server_group
- depot
- repository
- image
- error_handling
- deploy
- agent
- capture

Each section specifies the values that define the default behavior for the OEM API command or task. Certain values should not be edited, but they are listed here for completeness. If any of the editable values listed below are not appropriate for your environment, edit the file. The file is located on the DynaCenter Appliance in `/opt/race/share/conf/oem.ini`.

#### chassis

A chassis is a representation of a physical collection of servers. The chassis option affects the defaults used when target servers are registered through the OEM API commands.

| Value | Description | Default |
|---|---|---|
| `id` | The name of the chassis to use for newly created servers. Target servers can belong to any chassis, but servers that are manually registered are added to this chassis. | OEM |

### server_group

A server group is used to organize servers and is used to control which combinations of software and hardware resources can communicate.

| Value | Description | Default |
|---|---|---|
| `name` | The name of the server group for managed servers. All servers are added to this group during provisioning.<br>**Note:** Do not edit this option. | OEM |

### depot

The depot options describe where and how captured images are stored on network attached storage devices.

| Value | Description | Default |
|---|---|---|
| `uri` | The URI for the captured images depot, for example, depot://images. The value is used as a default if no value is given on the command line.<br>**Important:** If your depot is not in the default location (the root directory on the DynaCenter Appliance), you must edit this setting. | |

### repository

The repository options control where meta-data data is written in the DynaCenter repositories.

| Value | Description | Default |
|---|---|---|
| `drivers` | The name of the repository where Driversets should be saved.<br>**Note:** Do not edit this option. | default |
| `image` | The name of the repository where captured and deployed images should be saved by default.<br>**Note:** Do not edit this option. | instance |
| `template` | The name of the repository where templates for captured and deployed images should be saved.<br>**Note:** Do not edit this option. | default |

### image

The image options provide defaults for image handling operations.

| Value | Description | Default |
|---|---|---|
| `remove_empty_template` | A Boolean value that controls whether an empty template is removed after the last deployed image that is using it is removed. | True |

### error_handling

The error handling section of the configuration file controls how some classes of errors are handled by default.

| Value | Description | Default |
|---|---|---|
| `ignore_missing` | Delete/remove operations will not produce an error message if the specified item has already been deleted or does not exist. | True |
| `overwrite_existing` | Overwrite existing captured image if one with the same name exists. | False |
| `remove_failed_captures` | When true, partial images are removed from the depot if the full capture fails for any reason. When false, images are only removed if the capture is canceled. | True |

CA Server Automation prechecks for existing image and doesn't allow overwriting exisiting image.

### deploy

The deploy options set the defaults for the deploy operation.

| Value | Description | Default |
|---|---|---|
| `scale_images` | A Boolean value that globally controls image scaling. When true, captured images are resized as needed to fit onto servers and consume all available space. | True |
| `remove_agent` | A Boolean value that controls whether the DPAD is removed after the image has been deployed. When true, the DPAD agent is removed from the target server. | False |

### agent

The agent options control the management of agent images.

| Value | Description | Default |
|---|---|---|
| `retain_images` | The number of agent images to cache in the depot so that they are available for deploy and capture operations. If all of the cached images are in use at the same time, DynaCenter will create new agent images as needed, but that increases the amount of time needed to complete the operation. | 5 |
| `heartbeat_interval` | The frequency (in seconds) with which the DPAD communicates with DynaCenter.<br>**Tip:** When working with slower networks or cloud | 30 |

| | | |
|---|---|---|
| | networks, consider increasing the heartbeat interval to avoid missed heartbeats during capture and deploy operations.<br>**Note:** A longer heartbeat interval means that commands from DynaCenter will take longer to reach the DPAD. | |
| `pm_heartbeat_interval` | The frequency (in seconds) with which the agent image communicates with DynaCenter.<br>**Tip:** When working with slower networks or cloud networks, consider increasing the heartbeat interval to avoid missed heartbeats during capture and deploy operations.<br>**Note:** A longer heartbeat interval means that commands from DynaCenter will take longer to reach the DPAD. | 10 |
| `mws_url` | The url base used to access the DynaCenter Appliance.<br>**Note:** Do not edit this option. | |
| `register_url` | The communication path used by the DPAD to communicate with DynaCenter when the DPAD first starts.<br>**Note:** Do not edit this option. | |
| `heartbeat_url` | The communication path used by the DPAD to let DynaCenter know that it is running and to check if there are any pending tasks.<br>**Note:** Do not edit this option. | |
| `time_url` | The communication path used by the DPAD to get the current time from DynaCenter.<br>**Note:** Do not edit this option. | |
| `taskstatus_url` | The communication path used by the DPAD to send information to DynaCenter about the status of any tasks the DPAD is performing.<br>**Note:** Do not edit this option. | |
| `log_url` | The communication path used by the DPAD to send log messages to DynaCenter.<br>**Note:** Do not edit this option. | |
| `event_url` | The communication path used by the DPAD to send information to DynaCenter about any actions or tasks the DPAD is performing.<br>**Note:** Do not edit this option. | |
| `request_url` | The communication path used by the DPAD to request the next provisioning task. | |

| | | |
|---|---|---|
| | **Note:** Do not edit this option. | |
| `command_ url` | The communication path used by the DPAD to send information about the status of the provisioning task that is currently running.<br>**Note:** Do not edit this option. | |
| `set_clock` | A Boolean value that determines whether the DPAD asks DynaCenter for the current time.<br>When set to True, the DPAD changes the system time on the server when the daemon starts. | False |

### capture

The capture option sets the defaults for the deploy operation.

| Value | Description | Default |
|---|---|---|
| `compress_image` | A Boolean value that globally controls image compression. When true, captured images are compressed. | True |

## Sample OEM Configuration File

This sample configuration file represents the default configuration of DynaCenter.

```
[chassis]
id = OEM

[server_group]
name = OEM

[depot]
uri = depot://image

[repository]
drivers = default
image = instance
template = default

[image]
remove_empty_template = True

[error_handling]
ignore_missing = True
overwrite_existing = False
remove_failed_captures = True

[deploy]
scale_images = True
remove_agent = False
```

```
[agent]
retain_images = 5
heartbeat_interval = 30
pm_heartbeat_interval = 10
mws_url = https://localhost
register_url = %(mws_url)s/agent/register/
heartbeat_url = %(mws_url)s/agent/heartbeat
time_url = %(mws_url)s/agent/get_current_time
taskstatus_url = %(mws_url)s/agent/taskstatus
log_url = %(mws_url)s/agent/log
event_url = %(mws_url)s/agent/event
command_url = %(mws_url)s/agent/mark_command
request_url = %(mws_url)s/agent/next_request
set_clock = False

[capture]
compress_image = True
```

### Retaining customizations

When you upgrade DynaCenter, you do not lose any customizations you have made to the oem.ini file. During an upgrade, DynaCenter rewrites the oem.ini file with your custom settings and any new configuration settings introduced as part of the upgrade.

## Configure Logging

DynaCenter logs all of its operations to files on the DynaCenter Appliance. 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 41.
- To specify which log messages the DLAD writes to the log files, edit the log level in the logserver.ini configuration file.

7. **Reference:** See Configure the logging level for the DLAD on page 41.

### 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 DynaCenter Appliance, 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 DynaCenter Appliance, 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 |
|---|---|
| configure.log | 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.

## Retaining customizations

When you upgrade DynaCenter, you do not lose the customizations you have made to the logserver.ini or logclient.ini files as these files are not overwritten during the upgrade process. Instead of overwriting the files and erasing your custom settings, DynaCenter creates new versions of the files by appending '.rpmnew' to the original file name. You can then compare your file with the new file created as part of the upgrade and copy any new options in the new file to your original file.

# Part II: Working with the DynaCenter Appliance

# Chapter 7: Capture Server Images

The capture operation uses the content of a disk to create a server image and then places that image in the image depot so that it can be deployed to an AppLogic appliance. During the capture operation, you can use a capture profile to define which storage on the server should be captured as part of the image.

**Note:** DynaCenter does not currently support capturing an AppLogic appliance image.

## Operating System Support for Capture Operations

DynaCenter supports the capture of servers for the following operating systems that can be deployed to an AppLogic Grid:

- RedHat 5.4 and later 32-bit
- RedHat 5.4 and later 64-bit
- CentOS 5.4 and later 32-bit
- CentOS 5.4 and later 64-bit
- Windows 2008 Release 2

## Prerequisites to Capture Operations

Before you can capture a server image there are prerequisite tasks that must be completed. This section identifies those tasks.

**Note:** Some of these tasks might already have been completed as part of other operations.

| ✓ | Description | Reference |
|---|---|---|
| ☐ | Create the necessary firewall tunnels on your site | Create firewall tunnels on page 48.<br>**Tip:** If the firewall tunnels have already been created, you can skip this task. Use the `ping <hostname>` command to test whether the tunnels are already open. |
| ☐ | Register the network used by the server you want to capture | Register the server network on page 48.<br>**Tip:** If this network is already registered with DynaCenter, you can skip this task. Check the list of networks in the output from the `dccmd list networks` command to confirm whether the network has already been registered. |
| ☐ | Register the depot where the | Register the depot on page 49. |

| | | |
|---|---|---|
| | server image will be stored | |
| ☐ | Determine the MAC address of the source server | Determine the MAC address of the server to be captured on page 51.<br>**Tip:** If this server is already registered with DynaCenter, you can skip this task. Check the output from the `dccmd list servers` command to confirm whether the server has already been registered. |
| ☐ | Configure the source server to meet AppLogic requirements | Configure the source server (Windows 2008 only) on page 51. |

## Create firewall tunnels

Before DynaCenter can communicate with servers outside of the AppLogic Grid, you must open a port on your firewall and direct traffic on that port to port 443 on the DynaCenter Appliance.

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

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 Firewall tunnels already exist, confirm that they are operational.

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

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

  `ping <hostname>`

## Register the server network

Before DynaCenter can communicate with the server, you must tell DynaCenter about the network where the server resides.

**Note:** If the network is already registered, you do not need to perform this task. To see if the network is already registered, run the `dccmd list networks` command on the DynaCenter Appliance and check the output for the network where the server resides. If the `dccmd list networks` command does not provide enough detail, use the `dccmd show network <network name>` command.

### Procedure

1. On the DynaCenter Appliance, type the following command:

   ```
   dccmd register external network <network name> \
   <MWS URL base>
   ```

   Where:

| | | |
|---|---|---|
| Arguments | `network name` | Unique name for the network you are registering<br>**Example:** server_network |
| | `MWS URL base` | URL used to access the DynaCenter Appliance from outside the local area network<br>**Example:** https://<host>, where `host` is the externally addressable name or IP address of the DynaCenter Appliance. |

   **Example:**

   ```
   dccmd register external network server_network \
   https://192.168.10.20:4102
   ```

2. Confirm that the network registered successfully:

   ```
   dccmd show network <network name> --format=full
   ```

   **Example:**

   ```
   dccmd show network server_network --format=full
   ```

**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.

## Register the depot

Before a server can connect to the depot where captured images are stored, you must associate the network where the server resides with the depot.

**Note:** You do not have to complete this task if the depot is located on the DynaCenter Appliance (default installation option).

### Procedure

1. On the DynaCenter Appliance, type the following command to register the depot:

   ```
   dccmd register depot \
   [--access-identity=<id_or_user_name>] \
   [--access-secret=<secret_key>] \
   <depotId> <depotURL> \
   <networkId>=<depotURL> [<networkId>=<depotURL>...]
   ```

   Where:

| | | |
|---|---|---|
| Options | `--access-identity` | Authorized identity to access this storage server (for example, the |

| | | |
|---|---|---|
| | | username, the account, etc.)<br>**Note:** If you use this option, you must also define `--access-secret`.<br>**WebDAV storage:** This is a username of a user authorized to access the server. |
| | `--access-secret` | Password, key, or certificate associated with the authorized identity<br>**WebDAV storage:** This is the password associated with the username specified for `--access-identity`. |
| Arguments | `depotId` | Name of the depot |
| | `depotURL` | URL of the network storage; it can be one of the following:<br>▪ **WebDAV storage:** dav://server-address/path |
| | `networkId= depotURL` | Network that contains servers you want to capture and the URL of the depot when accessed from that network |

**Example:**

```
dccmd register depot mystorage \
dav://mystorage/repo/images \
--access-identity=someuser \
--access-secret=somepassword \
server_network=dav://192.168.127.253:4102/repo/images
```

2. Confirm that the depot registered successfully:

   ```
   dccmd show depot <depotId> --format=full
   ```

**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.

## Determine the MAC address of the server to be captured

Before you can capture a server, the server must be registered with DynaCenter. To register the server you must know the MAC address of the server.

**Note:** If the server is already registered, you do not need to perform this task. To see if the server is already registered, run the `dccmd list servers` command on the DynaCenter Appliance and check the output for the server name.

### Procedure

1. Open the command line interface on the server.
2. In the command line interface, use a command such as `ifconfig` or `ipconfig` to determine the MAC address.

   The standard format for MAC addresses is six groups of two hexadecimal digits separated by colons (:). You can find the MAC address in the HWaddr field.

   **Example:** 00:1d:c2:00:d7:e0

   **Note:** If there is more than one interface listed in the output, use the MAC address for the interface that is the external IP address.
3. Record the MAC address.

## Configure the source server (Windows 2008 only)

AppLogic requires the Windows Server 2008 operating system be configured in a specific way when part of an AppLogic Grid. If you are capturing a Windows Server 2008 image to deploy into your AppLogic Grid, you should review the AppLogic requirements before you capture the image and deploy that image to the Grid.

**Reference:** The AppLogic documentation for Windows Server 2008 located at (http://doc.3tera.com/AppLogic29/RefWindowsInstallWin08.html) describes how to create a Windows Server 2008 Base Server Class; the configuration requirements for Windows 2008 are contained within these instructions as outlined in the following table:

| Requirement | Noted in |
|---|---|
| Set the screen resolution | Step 4 |
| Set the screen saver timeout | Step 4 |
| Disable the page file | Step 4 |
| Disable power management hibernation support | Step 4 |
| Disable IE enhanced security | Step 4 |
| Clean up any leftover pagefile | Step 5 |
| Install high priority and optional Windows updates | Step 7 |
| Disable the Windows Firewall | Step 10 |
| Disable automatic updates | Step 11 |
| Disable password complexity requirements | Step 12 |

| | |
|---|---|
| Set the Administrator password to never expire | Step 13 |
| Disable the shutdown event tracker | Step 14 |
| Set the disk device timeout | Step 15 |
| Disable TCP checksum offload | Step 16 |

## Capture a Server

The capture operation captures the image of a server so that it can be moved or cloned to additional systems. This section lists and describes the tasks that you must perform before you can capture an image from a server to deploy to an AppLogic appliance.

**Important:** See Prerequisites to Capture Operations on page 47 before you perform this procedure.

| Task | Description | Reference |
|---|---|---|
| 1 | Register the server | Task 1: Register the server on page 52.<br>**Tip:** Check the output from the `dccmd list servers` command to confirm whether the server has already been registered. |
| 2 | Install a DPAD on the server | Task 2: Install a DPAD on the server on page 52. |
| 3 | Capture the server image | Task 3: Capture the server on page 54. |
| 4 | Remove the DPAD from the server (optional) | Task 4: Remove the DPAD from the server (optional) on page 56. |

### Task 1: Register the server

You can only capture a server that is registered with DynaCenter.

**Note:** If the server is already registered, you do not need to perform this task. To see if the server is already registered, run the `dccmd list servers` command on the MWS and check the output for the server name.

#### Procedure

1. From the command line of the DynaCenter Appliance, run the following command:

   ```
   dccmd register server [--networks=<network>] \
   [--name=<server_name>] <MAC>
   ```

   Where:

| | | |
|---|---|---|
| Options | `--networks` | `network name` specified when you registered the network where the server is located<br>**Tip:** To view a list of registered networks, run |

| | | |
|---|---|---|
| | | `dccmd list networks`. |
| | `--name` | Name for this server in the list of registered servers |
| Arguments | `MAC` | MAC address of the server being registered<br>**Example:** 00:1d:c2:00:d8:b0<br>**Reference:** Determine the MAC address of the server to be captured on page 51. |

**Example:**

```
dccmd register server \
--networks=server_network --name=my_server \
00:1d:c2:00:d8:b0
```

2. Confirm that the server registered successfully:

   `dccmd show server <server_name>`

   The substatus should be Ready.

## Task 2: Install a DPAD on the server

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.

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`

**Note:** If the DPAD is already installed, you do not need to perform this task. To see if the DPAD is already installed, run the `dccmd ping server <server_name>` command from the DynaCenter Appliance. If the DPAD is installed, the ping response will be:

```
DynaCenter agent in server <server_name> is alive and well.
```

### Procedure

1. Copy the appropriate package from the DynaCenter Appliance 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 DynaCenter Appliance 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`

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 DynaCenter Appliance 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. On the DynaCenter Appliance, run the following command to confirm that the DPAD installed correctly:

   `dccmd show server <server_name>`

   The substatus should be Running.

## Task 3: Capture the server

1. From the command line of the DynaCenter Appliance, run the following command:

```
dccmd capture server [--depot=<depot_url>] \
[--description=<image_description>] \
[--name=<image_name>] \
[--offline|--live] \
[--overwrite-existing|--no-overwrite-existing] \
[--profile=<profile_url>] \
<server_id> <ostype>\
```

Where:

| | | |
|---|---|---|
| Options | `--depot` | Location of the depot where the image will be stored |
| | `--description` | Helpful comment about the image |
| | `--name` | Alternate image name for the captured image. The default is to generate a name that uses information found in the image. |
| | `--offline\|--live` | Capture mode<br>**Note:** The server will be rebooted during an offline capture. |
| | `--overwrite-existing\|no-overwrite-existing` | Image overwrite mode if an image with the same name already exists |
| | `--profile` | Profile to use when capturing an image. The value is either a URL that points to a document accessible from the DynaCenter Appliance or a string that contains all of the profile data.<br>**Example:** `--profile=file://localhost/profiles/my_capture_profile.txt`<br>**Reference:** See Chapter 9: Customizing Images, which starts on page 65. |
| Arguments | `server_id` | Name of the registered server to be captured |
| | `ostype` | String that describes the operating system that is to be captured<br>**Tip:** To view a list of valid os types, run `dccmd list ostypes`. |

**Example:**

```
dccmd capture server \
--–depot=dav://mystorage/repo/images \
--description=’my_ server’ \
--live \
Dell-r200 \
‘RedHat_Linux-EL_5_*-x86_64’
```

A new captured image is created in the image depot.

2. Confirm that the image was captured:

   ```
   dccmd list images –captured
   ```

**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.

## Task 4: Remove the DPAD from the server (optional)

When you no longer want DynaCenter to communicate with the server you are capturing, you can remove the DPAD from the server.

### Windows

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

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

# Chapter 8: Deploy Images

The deploy operation takes a captured image from the image depot and provisions it to an AppLogic appliance. You can use a profile to modify the storage configuration of the image during the deploy operation.

## Operating System Support for Capture Operations

DynaCenter supports the capture of servers for the following operating systems that can be deployed to an AppLogic Grid:

- RedHat 5.4 and later 32-bit
- RedHat 5.4 and later 64-bit
- CentOS 5.4 and later 32-bit
- CentOS 5.4 and later 64-bit
- Windows 2008 Release 2

## Prerequisites

Before you can deploy an image to an appliance there are prerequisite tasks that must be completed. This section identifies those tasks.

### Build AppLogic templates

Before you can deploy an image to an AppLogic appliance, you must have a template that has the same operating system and the same bitness as the image you want to deploy. AppLogic provides templates for CentOS 5.4 32- and 64-bit; if you are deploying to any other operating system, you must create an appropriate template.

**Reference:** See the AppLogic documentation for information about how to create a template.

### Build AppLogic Windows filer applications (Windows only)

If you plan to deploy an image to an appliance that is running Windows 2008, you must have a Windows filer template application for that version of the operating system.

The Windows filer template application allows AppLogic to resize volumes on the appliance that you are deploying an image to.

**Reference:** See the AppLogic documentation for information about how to create a Windows filer application.

### Source images

Before you can deploy an image to an AppLogic appliance, you must have images in the depot.

**Reference:** See Chapter 7: Capture Server Images, which starts on page 47 for information about capturing source images.

The source image contains the networking and storage information from the server that it was captured from. Some of this information must be modified when you deploy the image to another system. During the deploy operation, an image is modified in the following ways:

- DynaCenter removes the networking information because AppLogic manages the network information for an appliance.

  **Note:** Any network information contained in your source image or specified in a deploy profile is ignored when the image is deployed. The deployed image uses the network information that was assigned to the target appliance when you created the appliance.

- DynaCenter disables the "Passwords must meet complexity requirements" setting on the target server. AppLogic requires certain configuration settings on Windows Server 2008 systems and, because the password complexity check prevents a newly deployed image from registering with the AppLogic Grid, DynaCenter makes this configuration change.

  **Note:** While disabling password complexity requirements might be seen as a security risk, this is an AppLogic requirement.

  **Reference:** See Task 5: Configure the Windows operating system (Windows 2008 only) on page 62.

- You can modify the storage configuration using a profile.

  **Reference:** See Sample Storage Deploy Profile Statements on page 73.

## Deploy an Image to an Appliance where no Image is Deployed

This section describes how to deploy an image to an appliance that does not already have an image deployed to it.

### Process Overview

| Task | Description | Reference |
|---|---|---|
| 1 | Create an appliance on the AppLogic Grid | Task 1: Create an AppLogic appliance to receive the image on page 59 |
| 2 | Deploy an image to the appliance | Task 2: Deploy an image to the AppLogic appliance on page 60 |
| 3 | Re-enable ssh root access on Linux appliances (optional) | Task 3: (Optional) Enable SSH root access on page 61 |
| 4 | Install the TurboGate drivers | Task 4: Install TurboGate |

| | | |
|---|---|---|
| | (Windows only) | paravirtualized drivers (Windows only) on page 62 |
| 5 | Configure the Windows operating system (Windows only) | Task 5: Configure the Windows operating system (Windows 2008 only) on page 62 |

## Task 1: Create an AppLogic appliance to receive the image

In an AppLogic environment, you must create an appliance before DynaCenter can deploy an image. When you provision the appliance, choose a template that has the same operating system and the same bitness as the image you want to deploy to the appliance and allocate sufficient resources for the image.

**Note:** For specific instructions on how to create an appliance, see your AppLogic documentation.

### Procedure

1. To provision an appliance from the AppLogic command line, type the following command:

```
app provision <template-name> <appliance-name> \
user=<some-user-name> user_pw=<some-user-password>
root_pw=<root-user-password> \
primary_ip=<external-interface-ip> \
netmask=<external-interface-mask> \
gateway=<external-interface-default-gateway> \
hostname=<new-hostname> \
dns1=<primary-nameserver> dns2=<secondary-nameserver> \
cpu.dflt=<default-cpu> mem.dflt=<default-memory> \
<volume-name>.boot.size=<volume-size>
```

**Note:** To provision Windows appliances, use `admin_pw=some_admin_password` instead of `root_pw=<root-user-password>`.

**Linux Example**

To provision an appliance for a 64-bit CentOS 5.4 image from the AppLogic command line:

```
app provision VDS64_CentOS54_r1\
my_centos_system \
user=admin user_pw=some_password \
root_pw=some_root_password \
primary_ip=172.16.120.15 \
netmask=255.255.255.0 gateway=172.16.120.1 \
hostname=centos_54_64 \
dns1=172.16.240.1 dns2=172.16.250.1 \
cpu.dflt=1 mem.dflt=2G\
VDS64_CENTOS54.boot.size=50G
```

**Windows Example**

To provision an appliance for a Windows 2008 image from the AppLogic command line:

```
app provision VDS_Win08E\
my_windows_system \
user=admin user_pw=some_password \
admin_pw=some_admin_password \
primary_ip=172.16.120.25 \
netmask=255.255.255.0 gateway=172.16.120.1 \
hostname=win_8 \
dns1=172.16.240.1 dns2=172.16.250.1 \
cpu.dflt=1 mem.dflt=4G \
VDS_Win08E.boot.size=100G
```

**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.

2. Type `app list` to verify that the appliance was provisioned.

## Task 2: Deploy an image to the AppLogic appliance

After you have created an appliance on the AppLogic grid, you can deploy any image to the appliance if the image has the same operating system and the same bitness as the appliance.

### Procedure

1. From the command line of the DynaCenter Appliance, run the following command:

```
dccmd cloud deploy image \
[--depot=<depot_url>] \
[--profile=<profile_url>] \
<image_id> <domain_id> <server_id> <network_id>
```

Where:

| | | |
|---|---|---|
| Options | `--depot` | Location of the depot that contains the captured image to be deployed |
| | `--profile` | Profile to use when deploying this image. The value is either a URL that points to a document accessible from the DynaCenter Appliance or a string that contains all of the profile data.<br>**Example:** `--profile=file://localhost/profiles/my_deploy_profile.txt`<br>**Reference:** See Chapter 9: Customizing Images, which starts on page 65. |
| Arguments | `image_id` | Name of the captured image to deploy to the |

| | | |
|---|---|---|
| | | appliance |
| | `domain_id` | DynaCenter domain that should be assigned to the new server<br>**Note:** In AppLogic environments, this is always `applogic-domain`. |
| | `server_id` | Name of the appliance<br>**Note:** This is the `<appliance-name>` argument you specified when you created the appliance. |
| | `network_id` | `network name` specified when you registered the external network<br>**Note:** In AppLogic environments, this is always `applogic-network`. |

**Linux Example:**

```
dccmd cloud deploy image \
--depot=dav://mystorage/repo/images \
--profile=file://localhost/profiles/email_cluster.conf
dell_r200_centos54_64 \
applogic-domain \
my_centos_system \
applogic-network
```

**Windows Example:**

```
dccmd cloud deploy image \
--depot=dav://mystorage/repo/images \
--profile=file://localhost/profiles/ email_cluster.conf
dell_r200_w2k3_32 \
applogic-domain \
my_windows_system \
applogic-network
```

**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.

## Task 3: (Optional) Enable SSH root access (Linux only)

When DynaCenter deploys an image to your Linux appliance, it injects the AppLogic Appliance Kit, which disables SSH access for the root user. If you require SSH root access to the appliance, you must enable it after the deploy operation completes.

1. From the AppLogic Grid Shell, login to the deployed appliance.
2. Open the `sshd_conf` file with an editor such as vi from the following location on the appliance:

   `/etc/ssh/sshd_conf`

3. Change the PermitRootLogin setting to "yes".
4. Save your changes to the file.
5. Type the following command to restart the secure shell daemon:

   ```
   /etc/init.d/sshd restart
   ```

### Task 4: Install TurboGate paravirtualized drivers (Windows only)

The TurboGate paravirtualized drivers provide improved performance for fully virtualized Windows systems.

1. Log in to the appliance as an Administrator.
2. On the appliance, open the TurboGate folder that DynaCenter placed on the Desktop.
3. Double-click the install_turbogate.bat file.

   The TurboGate paravirtualized drivers are installed.
4. From the AppLogic Grid Shell, type the following command to restart the appliance:

   ```
   app restart <appliance_name>
   ```

### Task 5: Configure the Windows operating system (Windows 2008 only)

AppLogic requires that the Windows Server 2008 operating system be configured in a specific way when part of an AppLogic Grid. If you are deploying a Windows Server 2008 image into your AppLogic Grid, you should review the AppLogic requirements.

**Note:** If the server that the image was captured from was configured as required by AppLogic, you do not have to perform this task.

**Reference:** The AppLogic documentation for Windows Server 2008 located at http://doc.3tera.com/AppLogic29/RefWindowsInstallWin08.html describes how to create a Windows Server 2008 Base Server Class; the configuration requirements for Windows 2008 are contained within these instructions as outlined in the following table:

| Requirement | Noted in |
|---|---|
| Set the screen resolution | Step 4 |
| Set the screen saver timeout | Step 4 |
| Disable the page file | Step 4 |
| Disable power management hibernation support | Step 4 |
| Disable IE enhanced security | Step 4 |
| Clean up any leftover pagefile | Step 5 |
| Install high priority and optional Windows updates | Step 7 |
| Disable the Windows Firewall | Step 10 |
| Disable automatic updates | Step 11 |

| | |
|---|---|
| Disable password complexity requirements<br>**Note:** DynaCenter automatically disables this setting when it deploys an image. | Step 12 |
| Set the Administrator password to never expire | Step 13 |
| Disable the shutdown event tracker | Step 14 |
| Set the disk device timeout | Step 15 |
| Disable TCP checksum offload | Step 16 |

## Deploy an Image to an Appliance where an Image is Already Deployed

This section describes how to deploy an image to an appliance that has an image from a previous deploy operation still deployed.

### Process Overview

| Task | Description | Reference |
|---|---|---|
| 1 | Delete the existing appliance from the AppLogic Grid | Task 1: Delete the existing appliance on page 63 |
| 2 | Remove the appliance from the DynaCenter database | Task 2: Remove the appliance from the DynaCenter database on page 63 |
| 3 | Create a new appliance and deploy the new image | Deploy an Image to an Appliance where no Image is Deployed on page 58 |

### Task 1: Delete the existing appliance

⚠ **Caution:** When you delete an appliance, all information such as configuration settings and any user data is lost. Before you delete the appliance, backup or record any information or data that you do not want to lose.

- Type the following command from the AppLogic command line to delete the appliance:

  ```
  app destroy <appliance_name>
  ```

### Task 2: Remove the appliance from the DynaCenter database

- From the command line of the DynaCenter Appliance, run the following command:

  ```
  dccmd remove server <server_id>
  ```

  Where:

| | | |
|---|---|---|
| Arguments | `server_id` | Name of the appliance<br>**Note:** This is the `<appliance-name>` argument you specified when you created the appliance. |

**Example:**

```
dccmd remove server my_centos_system
```

### Task 3: Create a new appliance and deploy the new image

**Reference:** See Deploy an Image to an Appliance where no Image is Deployed on page 58.

# Chapter 9: 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.
- Add supplemental files, scripts, and patches to an image to satisfy a unique deployment scenario by creating a component.

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 67.
- How to create a profile and sample profile statements.

  **Reference:** See Section B: Create a Profile, which starts on page 71.
- How to create a software component with examples.

  **Reference:** See Section C: Create a Software Component, which starts on page 82.

# 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 storage layout on the target appliance based on the requirements of each appliance instance.

## 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.

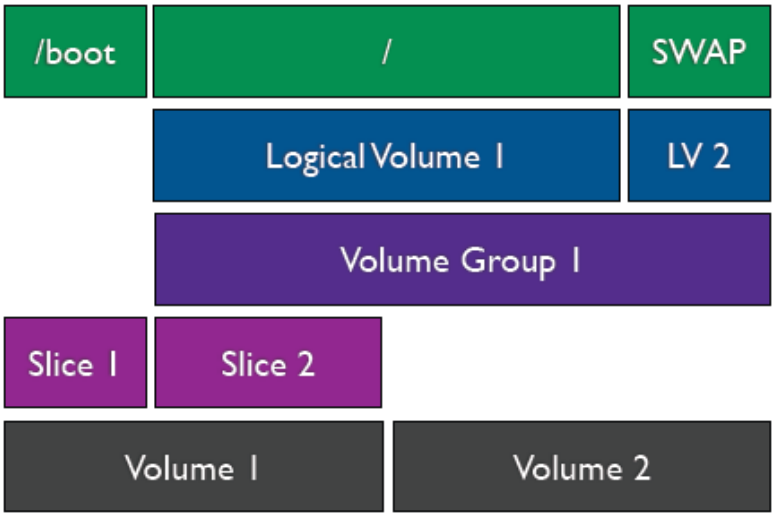
### Linux storage layers

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:



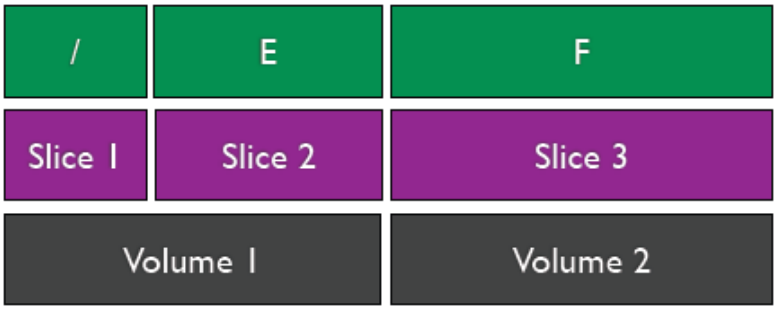

When you deploy an image, you can specify how you want the storage laid out on the target server by defining, in a profile, the storage layers from the bottom up.

## 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<br>**Note:** 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<br>partition 1<br>partition 2<br>Disk 2<br>partition 1 | E | Disk 2, partition 1 |
| Disk 1<br>partition 1<br>Disk 2<br>partition 1<br>partition 2 | E | Disk 2, partition 2 |
| Disk 1<br>partition 1<br>Disk 2<br>partition 1<br>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.

**Reference:** The *DynaCenter OEM Integration Guide* explains how to create profiles for use with web services and explains how to convert a profile text file for use with web services.

**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` option in either the dpmrsi capture command or the dpmrsi 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:** Sample Storage Capture Profile Statements, which start on page 72 and Sample Storage Deploy Profile Statements, which start on page 73.

- Any supplemental components, such as additional applications, patches, or scripts to deploy with an image.

  **Reference:** Miscellaneous deploy profile statements, which starts on page 79.

## Sample Storage Capture Profile Statements

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

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

### 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 72.

### 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 73.

### 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 72.

### 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 72.

## Sample Storage Deploy Profile Statements

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

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

### 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 77 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 78 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 78 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 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 82.

## 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.

**Note:** Testing a profile does not ensure that the profile is correct; it only tests that the syntax is valid.

### Procedure

- On the DynaCenter Appliance, 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.<br>**Note:** Each component must have the same os-type as the image that will use it or the provision will fail. The `dccmd list ostypes` 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<br>**Note:** 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 79.

## Windows Component Example

This example creates two Windows components for Windows 2008 images.

1. Download the files for the components WinSCP and TightVNC for Windows 2008.
2. Create `/root/install` on the DynaCenter Appliance with the following subdirectories: “W2k8_WinSCP” and “W2k8_VNC”.
3. Upload the files to their respective subdirectories in `/root/install`.
4. Install the WinSCP component:

   ```
   rinstallsnapshot --description="WinSCP for Windows \ 2008"
   --no-bootable --os-type=Microsoft_Windows- \
   ”*_*_*- *” --prefix=/install/W2k8_WinSCP --vendor=WinSCP \
   Win2K8_WinSCP /root/install/W2k8_WinSCP default
   ```

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

   ```
   rshowsnapshot Win2k8_WinSCP
   ```

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

```
rcommandprompt --component=Win2K8_WinSCP
ls root/install/W2k8_WinSCP
exit
```

7. Install the TightVNC component:

```
rinstallsnapshot --description="Tight VNC for Windows \
2008" --no-bootable --os-type=Microsoft_Windows- \ “*_*_*-
*” -- prefix=/install/W2k8_VNC \
--vendor=TightVNC Win2K8_VNC /root/install/W2k8_VNC \
default
```

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

```
rshowsnapshot Win2K8_VNC
```

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

```
rcommandprompt --component=Win2K8_VNC
ls root/install/W2k8_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\W2k8_WinSCP and C:\install\W2k8_VNC will have been created.

## Linux Component Example

This example creates an RHEL 5 component for RHEL 5 32-bit images.

1. Download htop for RHEL 5 32-bit.
2. On the DynaCenter Appliance, create /tmp/rpm_install/RHEL5_32bit_htop.
3. Upload the file to the /tmp/rpm_install/RHEL5_32bit_htop directory.
4. Set the permissions and ownership of the directories using the chmod and chown commands.
5. Create the RHEL5 component:

```
rinstallsnapshot --description="htop for RHEL5 32-bit" \
--no-bootable --os-type=RedHat_Linux-EL_5_*-i686 \
--prefix=/tmp/install/RHEL5_32bit_htop --vendor=htop \
RHEL5_32bit_htop /tmp/rpm_install/RHEL5_32bit_htop \
default
```

   The component file list will include entries for `/tmp/`, `tmp/rpm_install`, and `tmp/rpm_install/RHEL5_htop`, each of which will have its permissions and ownership tailored 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 RHEL5_32bit_htop
```

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

# Chapter 10: 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 DynaCenter Appliance.
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, request a new license from CA Technical Support.

## 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 DynaCenter Appliance in the `/opt/race/share/conf` directory.
2. Log into the DynaCenter Appliance 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 DynaCenter Appliance, so do not run it while systems are trying to boot or while the DynaCenter Appliance is committing changes to the database.

1. Open a command terminal on the DynaCenter Appliance.
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 DynaCenter Appliance.
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 DynaCenter Appliance, Racemi recommends using external storage for your depot. See DynaCenter Installation Options on page 21 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 DynaCenter Appliance:

1. Log in to the DynaCenter Appliance as root.
2. From the DynaCenter Appliance 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 DynaCenter Appliance 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 90 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 DynaCenter Appliance as root.
2. From the DynaCenter Appliance 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 DynaCenter Appliance 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 91 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
   ```

# 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.

### Appliance

In an AppLogic environment, a self-contained virtual environment that provides a particular function inside an application. The DynaCenter environment requires a simple appliance. See also simple appliance.

### 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.

### 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 DynaCenter Appliance 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.

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

### DynaCenter Appliance

A simple appliance where the DynaCenter software is installed and from which DynaCenter operations are executed.

### 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. See also DynaCenter Appliance.

### 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).

### Simple appliance

In an AppLogic environment, a single virtual environment. It is comprised of a virtual machine, virtual volumes, and virtual network interfaces. It runs its own copy of an operating system (e.g., CentOS) and appliance-specific software. See also appliance.

### Storage Server

Any device that contains storage. Storage servers will typically be devices that are large enough to store many captured images. The DynaCenter Appliance functions as a 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.

# Index

## M



## O



## P



## R



## S

## T