



Node Manager

Objectives

After completing this lesson, you should be able to:

- Configure WebLogic Server machines
- Set up and configure Node Manager
- Start managed servers through Node Manager

Node Manager

Node Manager is a WebLogic Server utility that:

- Can remotely start and stop the administration server or managed servers
- Can monitor the health of an instance of WebLogic Server and automatically restart it if it fails
- Runs as a separate process on the same machines as instances of WebLogic Server
- Is available as either a Java-based or a script-based process (for UNIX or Linux)
- Can be set up as an operating system service to start automatically when a system is rebooted

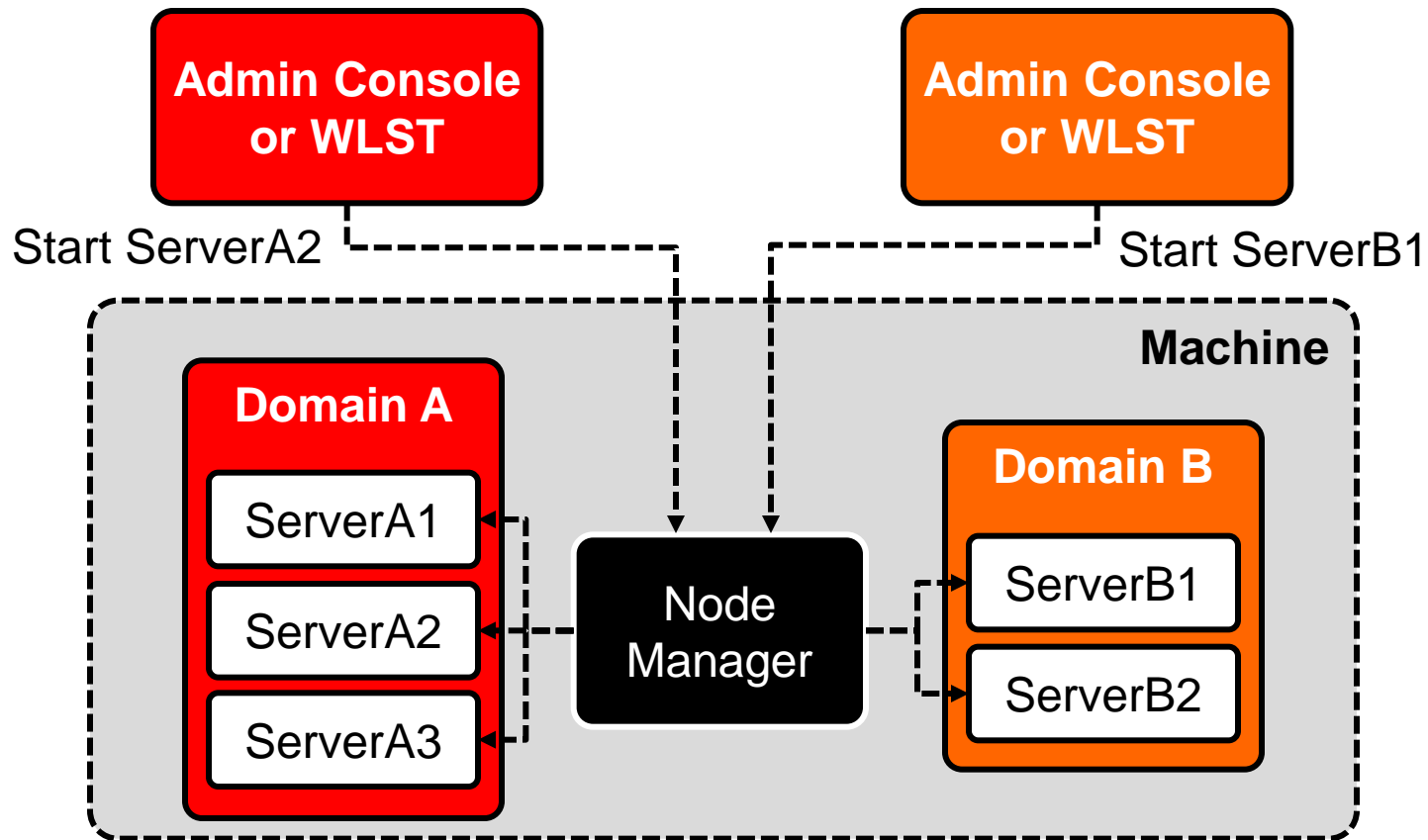
Two Types of Node Manager

- Java based
 - Runs within its own Java Virtual Machine
 - Runs on any operating system that runs WebLogic Server
 - Can use secure SSL communication, if desired
- Script based
 - Runs in its own process
 - Available only for UNIX or Linux systems
 - Uses Remote Shell (RSH) or Secure Shell (SSH)

Either type should be set up to start automatically when the computer is booted (as a UNIX daemon or a Windows service).

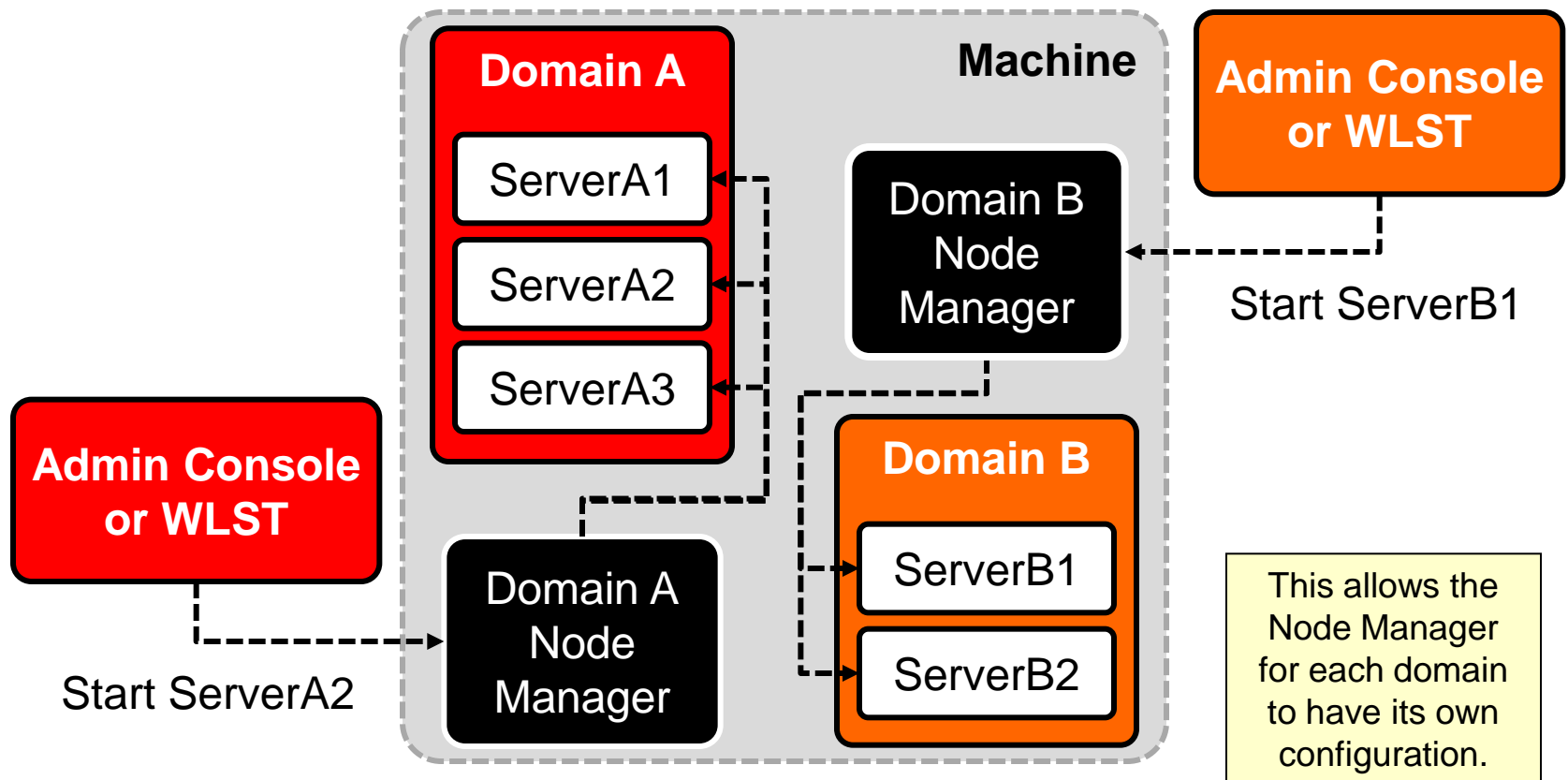
Node Manager Architecture: Per Machine

A single Node Manager can control servers from multiple domains.

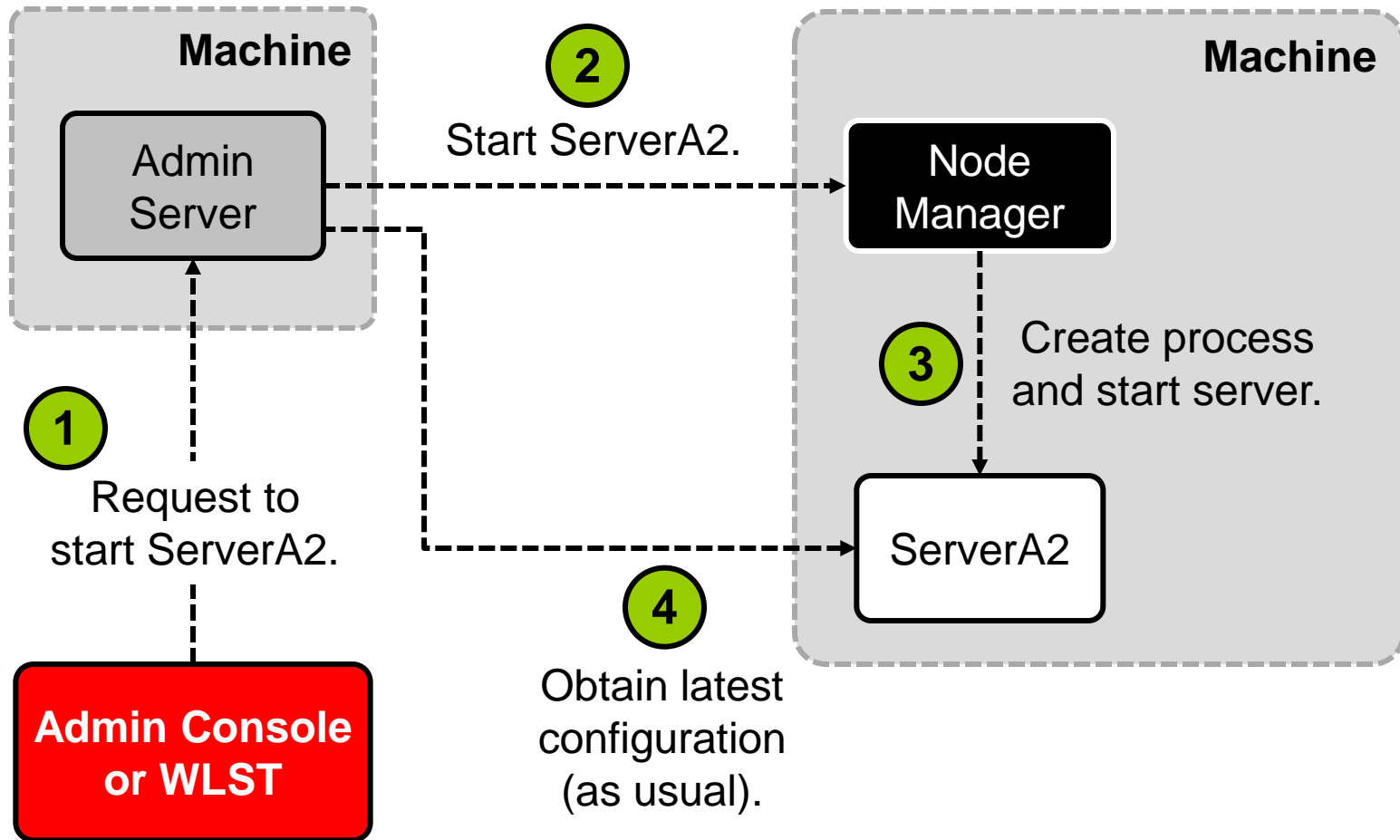


Node Manager Architecture: Per Domain

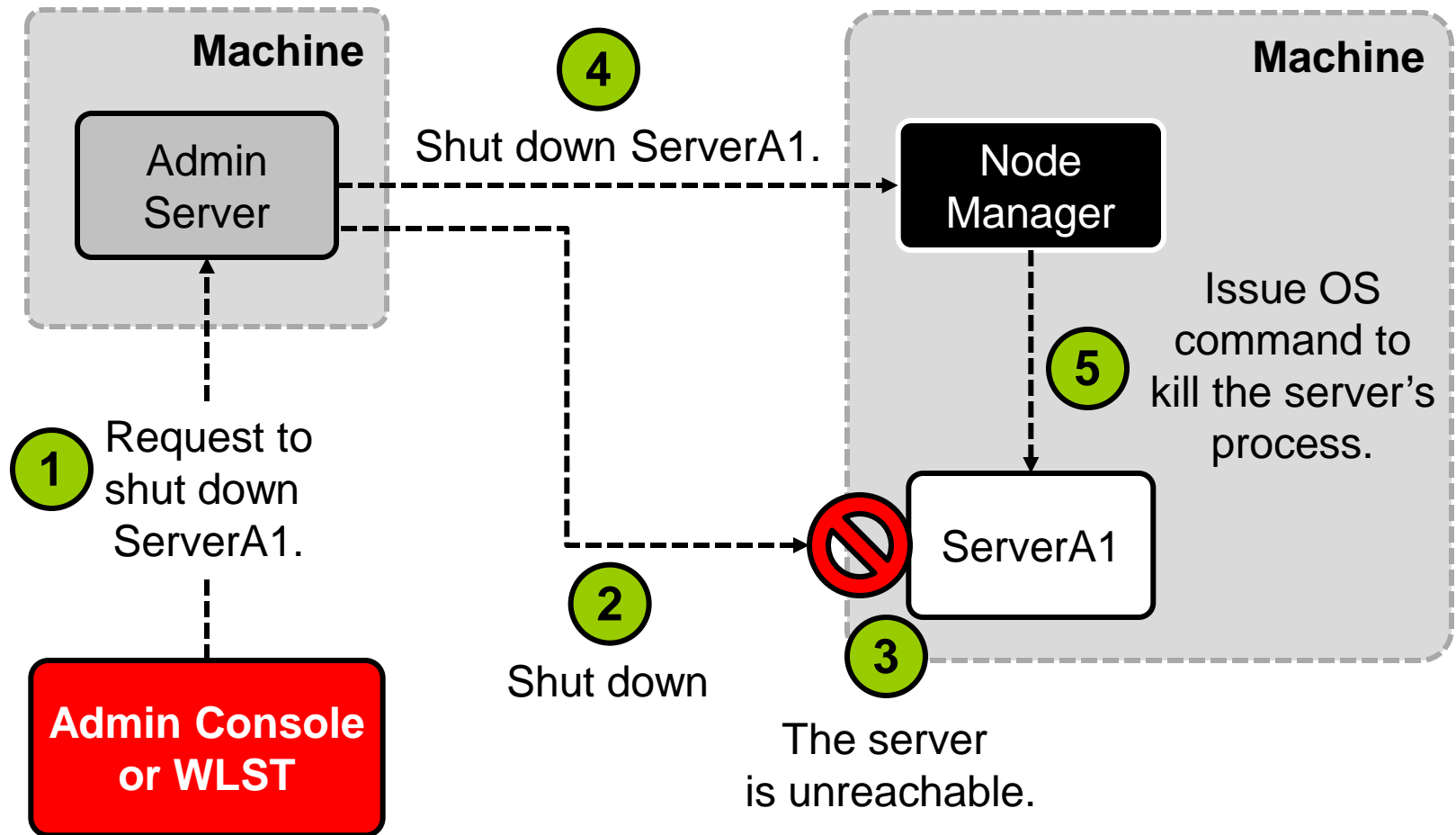
The default is to have one Java-based Node Manager per domain.



How Node Manager Starts a Managed Server



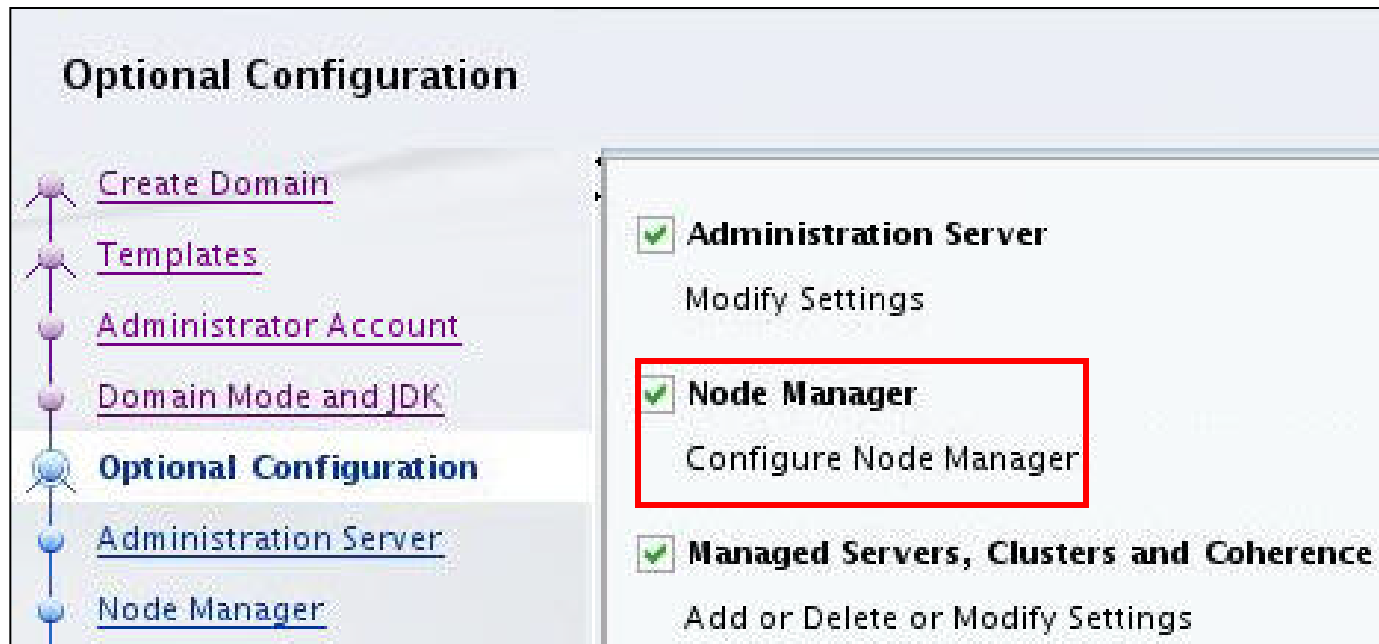
How Node Manager Can Help Shut Down a Managed Server



Configuration Wizard and Node Manager

When creating a domain with the Configuration Wizard, there is an option for Node Manager.

- You can select the **Node Manager** check box on the Optional Configuration page of a wizard.



Configuration Wizard and Node Manager

The screenshot shows the 'Node Manager' configuration window in the Oracle Fusion Middleware Configuration Wizard. The left sidebar contains a list of steps: 'Create Domain', 'Templates', 'Administrator Account', 'Domain Mode and JDK', 'Optional Components', 'Administration Server', 'Node Manager' (highlighted), 'Coherence Clusters', 'Machines', 'Configuration Summary', 'Configuration Progress', and 'End Of Configuration'. The main area is titled 'Node Manager' and features the Oracle Fusion Middleware logo. It includes three radio buttons for 'Node Manager Type': 'Per Domain' (selected), 'Custom Location', and 'Manual Node Manager Setup'. A text box for 'Node Manager Home' contains the path 'part1/wlsadmin/nodemanager' with a 'Browse' button. Below this are fields for 'Node Manager Credentials': 'Username' (nmadmin), 'Password', and 'Confirm Password'. Annotations with yellow callouts provide additional information: 'The default' points to the 'Optional Components' step; 'To change the Node Manager home location' points to the 'Node Manager Home' text box; 'Per Domain uses the default location. If Custom Location selected, enter the location here.' points to the 'Custom Location' radio button; and 'Credentials used by a client (admin server or WLST) to access Node Manager' points to the password fields.

Node Manager

ORACLE
FUSION MIDDLEWARE

[Create Domain](#)

[Templates](#)

[Administrator Account](#)

[Domain Mode and JDK](#)

[Optional Components](#)

[Administration Server](#)

Node Manager

[Coherence Clusters](#)

[Machines](#)

[Configuration Summary](#)

[Configuration Progress](#)

[End Of Configuration](#)

Node Manager Type

☒ Per Domain

☐ Custom Location

☐ Manual Node Manager Setup

Node Manager Home:

Node Manager Credentials

Username:

Password:

Confirm Password:

The default

To change the Node Manager home location

Per Domain uses the default location. If Custom Location selected, enter the location here.

Credentials used by a client (admin server or WLST) to access Node Manager

Configuring the Java-Based Node Manager

1. Define machines in the domain to represent each computer.
2. Under each machine configuration:
 - A. Assign servers to the machine.
 - B. Set the Java Node Manager type to Plain or SSL.
 - C. Set the listen address to the Node Manager computer's IP address or its host or DNS name. Select a listen port.
3. In each managed server configuration, set the server start parameters and the monitoring and restart parameters used by Node Manager. (If not set by you, these parameters have default values.)

Note that some of the configuration can be done when creating the domain.

Configuring Server Start and Health Monitoring Parameters

Settings for server1

Configuration Protocols Logging

Health Monitoring **Server Start** Web S

Save

Java Home:

Java Vendor:

BEA Home:

Root Directory:

Settings for server1

Configuration Protocols Logging De

Health Monitoring Server Start Web S

Save

Health Check Interval:

☐ Auto Kill If Failed

☒ Auto Restart

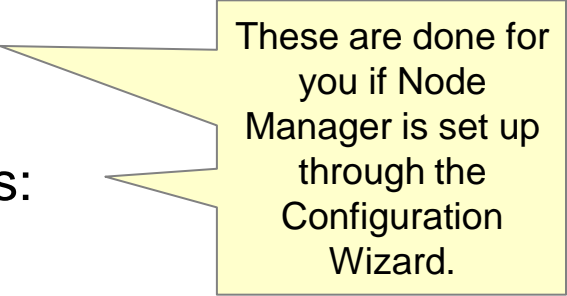
Restart Interval:

Max Restarts Within Interval:

Restart Delay Seconds:

Configuring the Java-Based Node Manager

4. Install WebLogic Server on the computers where you plan to run Node Manager (and managed servers).
5. *Create a directory for Node Manager on each computer:
 - A. Copy the `startNodeManager.sh` script to the directory from `<WL_HOME>/server/bin`.
 - B. Edit the script and set `NODEMGR_HOME` to the current path.
 - C. Create and edit a text file named `nodemanager.properties`.
 - D. Set (at least) the following properties:
 - `ListenAddress=value`
 - `ListenPort=value`



These are done for you if Node Manager is set up through the Configuration Wizard.

* The default Node Manager home is the `nodemanager` directory under the domain directory. This is the location used when Node Manager is configured during domain creation by using the Configuration Wizard and the “Per Domain” Node Manager type is selected.

Configuring the Java-Based Node Manager

6. On each computer, enroll Node Manager with the domain.
7. Start Node Manager by running `startNodeManager.sh`.
8. *Set up a UNIX daemon to start Node Manager automatically on computer startup.

* Not required, but recommended

Other Node Manager Properties

The `nodemanager.properties` file has many properties:

Property	Description
<code>AuthenticationEnabled</code>	Node Manager authenticates the admin server against the credentials defined in the domain.
<code>StartScriptEnabled</code>	Start servers by using a script.
<code>StopScriptEnabled</code>	Stop servers by using a script.
<code>StartScriptName</code>	Name of the script used to start servers
<code>StopScriptName</code>	Name of the script used to stop servers
<code>CrashRecoveryEnabled</code>	Automatically restart failed servers after machine restart.
<code>StateCheckInterval</code>	Time in between checks of a server's state

Node Manager Files

Under the Node Manager home directory:

- **`nodemanager.properties`**: Is used to set Node Manager attributes
- **`nodemanager.domains`**: Stores the domains with which this Node Manager communicates
- **`nodemanager.log`**: Is used by a running Node Manager to log information about its work

Under the domain's `config/nodemanager` directory:

- **`nm_password.properties`**: Stores the encrypted Node Manager username/password that the Node Manager client uses to authenticate itself to Node Manager

Node Manager Files

Under the domain's `security` directory:

- **`SerializedSystemIni.dat`**: Is used by Node Manager for encryption and decryption

Under the domain's directory called `servers/servername/data/nodemanager`:

- **`boot.properties`**: Is created by Node Manager to hold a server's encrypted credentials
- **`startup.properties`**: Keeps track of server start and health monitoring options
- **`servername.lck`, `servername.pid`, `servername.state`**: Are files used internally by Node Manager to track a server it started

Node Manager Files

Under the domain's directory called
`servers/servername/logs`:

- ***servername.out***: Is the log file for a server started by Node Manager that contains `stdout` and `stderr` messages generated by that server
 - You cannot set up “rotation” for this file, as you can with a server log file, which can be an issue. An easy solution:
 - For servers that are started by Node Manager, set the standard out “severity level” to `Critical` or higher. Then very few messages go into this file.

Enrolling Node Manager with a Domain

Enrolling Node Manager with a domain copies files required by Node Manager to the remote computer (or updates existing files). Then the Node Manager running on this machine can accept requests from that domain. To enroll with a domain:

1. Start WLST on the computer where you plan to run Node Manager.
2. Run the `connect()` command to connect to the administration server of the domain:

```
connect('username', 'pw', 't3://host:port')
```

3. Run the `nmEnroll()` command:

```
nmEnroll('domain_home', 'node_mgr_home')
```

Domain location

Node Manager home

When Not to Use `nmEnroll()`

- `nmEnroll()` is for registering Node Manager with a “WebLogic Server only” domain on a remote machine. It can be part of the process of preparing that machine for running the managed servers of the domain.
 - You should *not* use `nmEnroll()` if your domain uses other FMW components, because it is insufficient to ready a machine to run the managed servers in such a domain.
 - The `nmEnroll()` command does not transfer the required FMW component-specific files to the new machine.
- Using `nmEnroll()` is not needed if you have copied the domain to the new machine by using the pack and unpack utilities.

Reminder: Pack

1. On the machine where the domain files reside (and the administration server runs), use the `pack.sh` script with the `-managed=true` option:

```
cd <MW_HOME>/oracle_common/common/bin
./pack.sh -domain=domain_path/domain_name
          -template=name.jar
          -template_name=somename
          -managed=true
```

2. Move the archive file to the machine where you plan to run Node Manager and the managed servers (and the products are already installed).

Reminder: Unpack

3. Before running the `unpack.sh` script, ensure that the directory exists in which to place the domain: *new_path*
4. Run the `unpack.sh` script:

```
cd <MW_HOME>/oracle_common/common/bin
./unpack.sh -domain=new_path/domain_name
            -template=name.jar
            -app_dir=new_path/applications
            -nodemanager_home=nodemgr_home_dir
```

- The `-app_dir` argument is optional. It specifies the application directory (used by some FMW components).
- The `-nodemanager_home` argument is also optional. Use this to place Node Manager somewhere other than the default location: *<domain>/nodemanager*.

Controlling Servers Through Node Manager

After the configuration is set and Node Manager is running on the remote machines, servers can be started by using the WebLogic Server administration console or WLST.


Summary of Servers

Configuration **Control**

Customize this table

Servers (Filtered - More Columns Exist)

Start Resume Suspend ▾ Shutdown ▾ Restart SSL

<input type="checkbox"/>	Server 	Machine	State
<input type="checkbox"/>	AdminServer(admin)	machine1	RUNNING
<input type="checkbox"/>	server1	machine1	RUNNING
<input checked="" type="checkbox"/>	server2	machine2	SHUTDOWN



Node Manager: Best Practices

- Do not place the Node Manager home under the WebLogic Server installation directories.
 - The default location for the per-domain Java-based Node Manager is under the domain directories.
- Use the Java-based Node Manager.
 - It is portable and has the latest features.
- Use the “per domain” Node Manager.
 - The default with the Java-based Node Manager
 - Allows each domain to have a different Node Manager configuration
- Use Node Manager to start the administration server, too.
 - If it fails, Node Manager can automatically restart it.

Node Manager: Best Practices

- Use a start script to start servers.
 - Use the script to set WebLogic Server parameters or prepare resources before the server starts.
 - The default for `StartScriptEnabled` is `true`.
 - The default `StartScriptName` is `startWebLogic.sh`.
- In production, use SSL between the admin server and the Java-based Node Manager.
 - This is one-way SSL.
 - Set up SSL by obtaining and configuring Identity and Trust keystores on the admin server.
 - Configure the admin server to listen on a specific IP, not “all local addresses.”
 - Set `SecureListener=true` in `nodemanager.properties`.

Quiz

To configure a Node Manager by using the admin console, what resource is selected before you click its **Node Manager** tab?

- a. A managed server
- b. The administration server
- c. A machine
- d. A cluster

Quiz

When you shut down a managed server by using the admin console, Node Manager is involved:

- a. Each time
- b. If the managed server cannot be reached by the admin server
- c. Never

Summary

In this lesson, you should have learned how to:

- Configure WebLogic Server machines
- Set up and configure Node Manager
- Start managed servers through Node Manager

Practice 9-1 Overview: Configuring and Using Node Manager

This practice covers the following topics:

- Configuring the Java-based Node Manager
- Starting Node Manager
- Starting servers through Node Manager by using the administration console