ORACLE

Managing WebLogic Server Lifecycle

Starting WLS with Domain Supplied Scripts

- $DOMAIN_HOME/startWebLogic.sh
  - Starts the domain admin server
- Starting Managed Servers
  - $DOMAIN_HOME/bin/startManagedWebLogic.sh managedServerName http://adminserverdomainname:7001

Stopping WLS Servers via Scripts

- $DOMAIN_HOME/bin/stopWebLogic optionalUser optionalPassword t3://host:port
  - If no arguments are supplied then t3://localhost:7001 is used
  - Uses WSLT do shutdown WLS

Creating a Boot Identity File

- Create a file called `boot.properties` in the *`DOMAIN_NAME`*`/servers/<server_name>/security` directory that contains two lines:
  - `username=`*`username`*
  - `password=`*`password`*
- The first time you start the server, the server reads the Boot Identity file and overwrites it with an encrypted version of the username and password.
- Thereafter, the server remembers the credentials for subsequent startup cycles.

Starting WLS Servers as a Windows Service

- There are the scripts named *installSvc.cmd* and *uninstallSvc.cmd* located in %MIDDLEWARE_HOME%/wlserver_10.3/server/bin
- You should set a number of environment variables before calling these scripts
- Boot.properties file should exists for each servers that is going to be started as a Windows service

Setting up a Windows Service to start WLS server - example

```
@echo off
SETLOCAL
set DOMAIN_NAME=FoundationLab
set USERDOMAIN_HOME=c:\wls103\user_projects\domains\FoundationLab
set SERVER_NAME=AdminServer
set PRODUCTION_MODE=true
set JAVA_OPTIONS=
    Dweblogic.Stdout="c:\wls103\user_projects\domains\FoundationLab\stdout.txt" -
    Dweblogic.Stderr="c:\wls103\user_projects\domains\FoundationLab\stderr.txt"
set MEM_ARGS=-Xms128m -Xmx256m
call "c:\wls103\wlserver_10.3\server\bin\installSvc.cmd"
ENDLOCAL
```

Node Manager

- Is a process that runs on a machine and it not associated with a domain
  - Required to start WLS Servers from Console
  - WLST can also call node manager to start/stop servers
- Two Flavors
  - Java Flavor
  - Shell Scripts

Starting/Stopping Node Manager

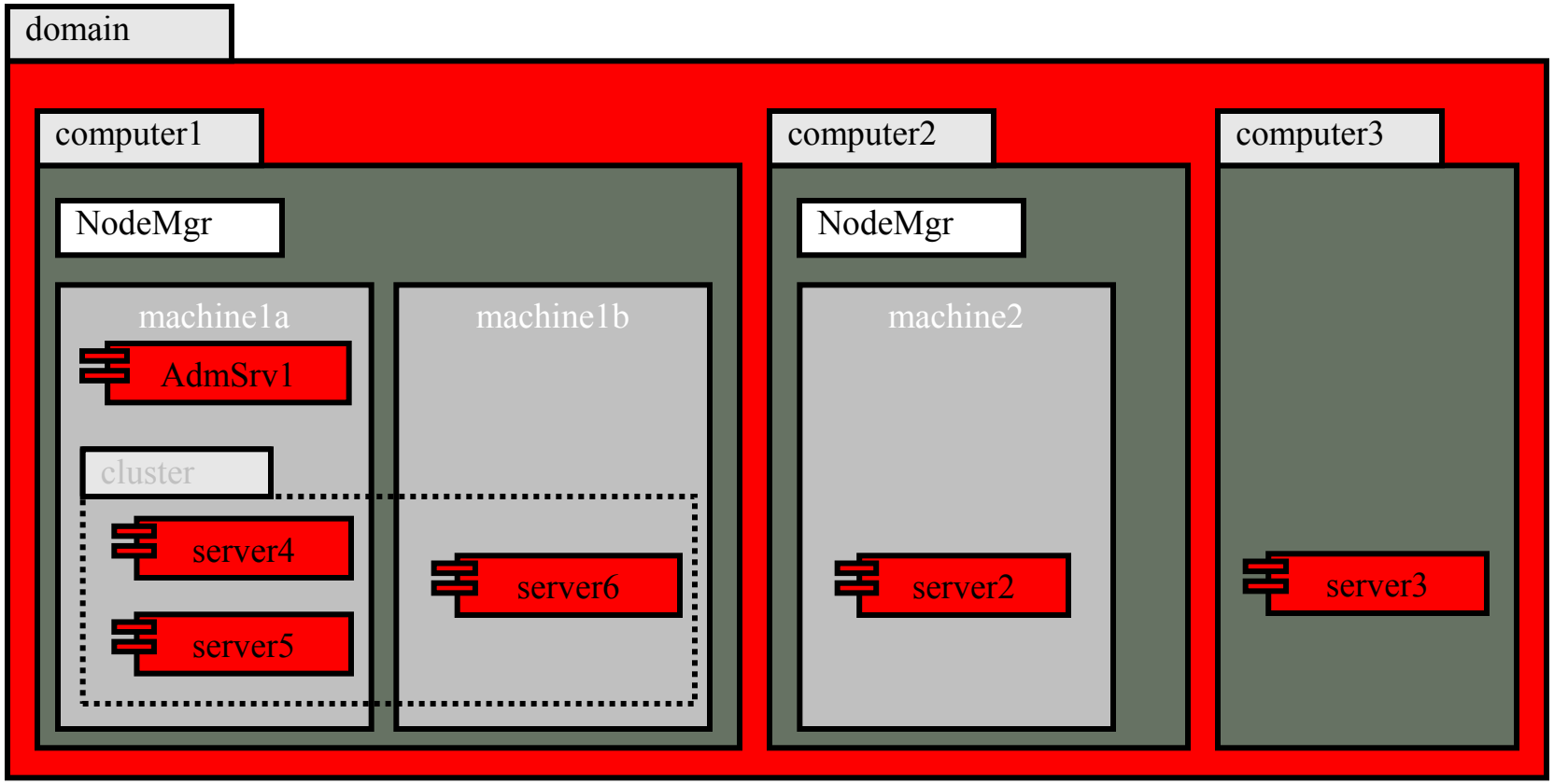
- Starting Node Manager
  - $WL_HOME/server/bin/startNodeManager.sh(cmd)
- Stopping Node Manager
  - Nothing supplied (kill -9 in UNIX for example)

- You can install/uninstall NodeManager as a Windows Service using scripts named *installNodeMgrSvc.cmd* and *uninstallNodeMgrSvc.cmd* located in %MIDDLEWARE_HOME%/wlserver_10.3/server/bin

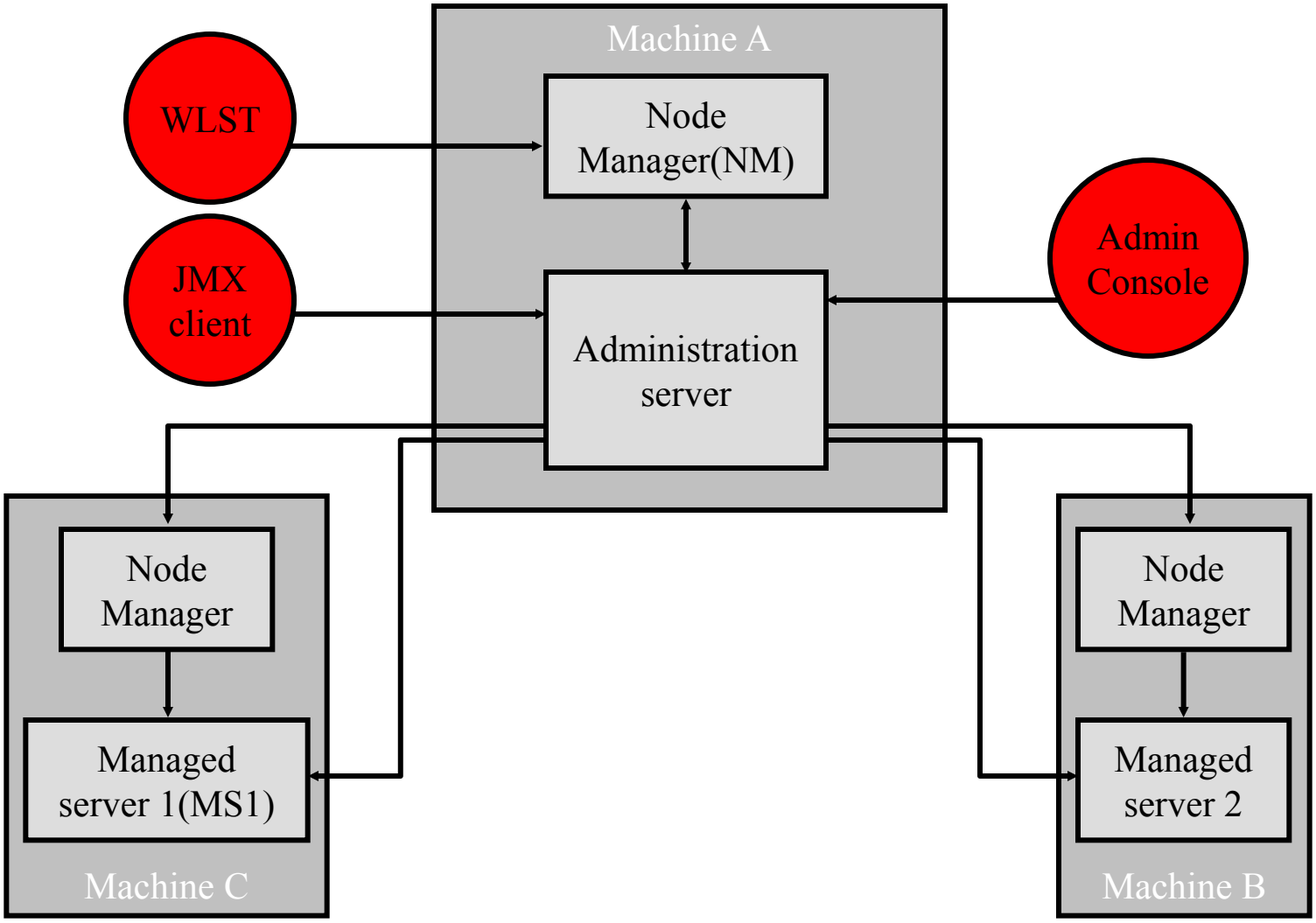
What is a Machine?

- Logical representation of physical host
- Includes configuration of Node Manager
  - Listen Ports
- Ensures State is replicated to JVM's on separate machines

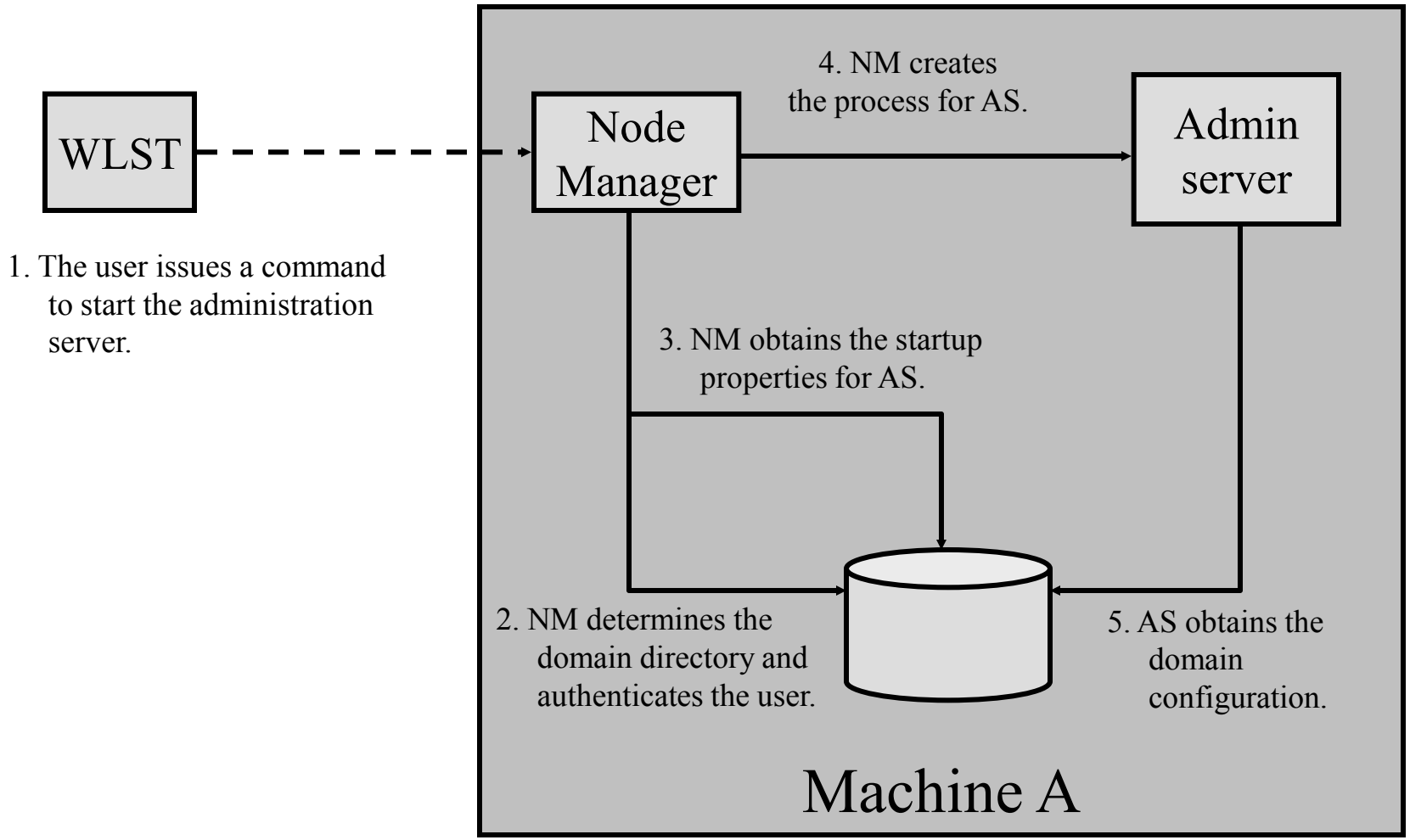
Relationship of Machines to Other Components



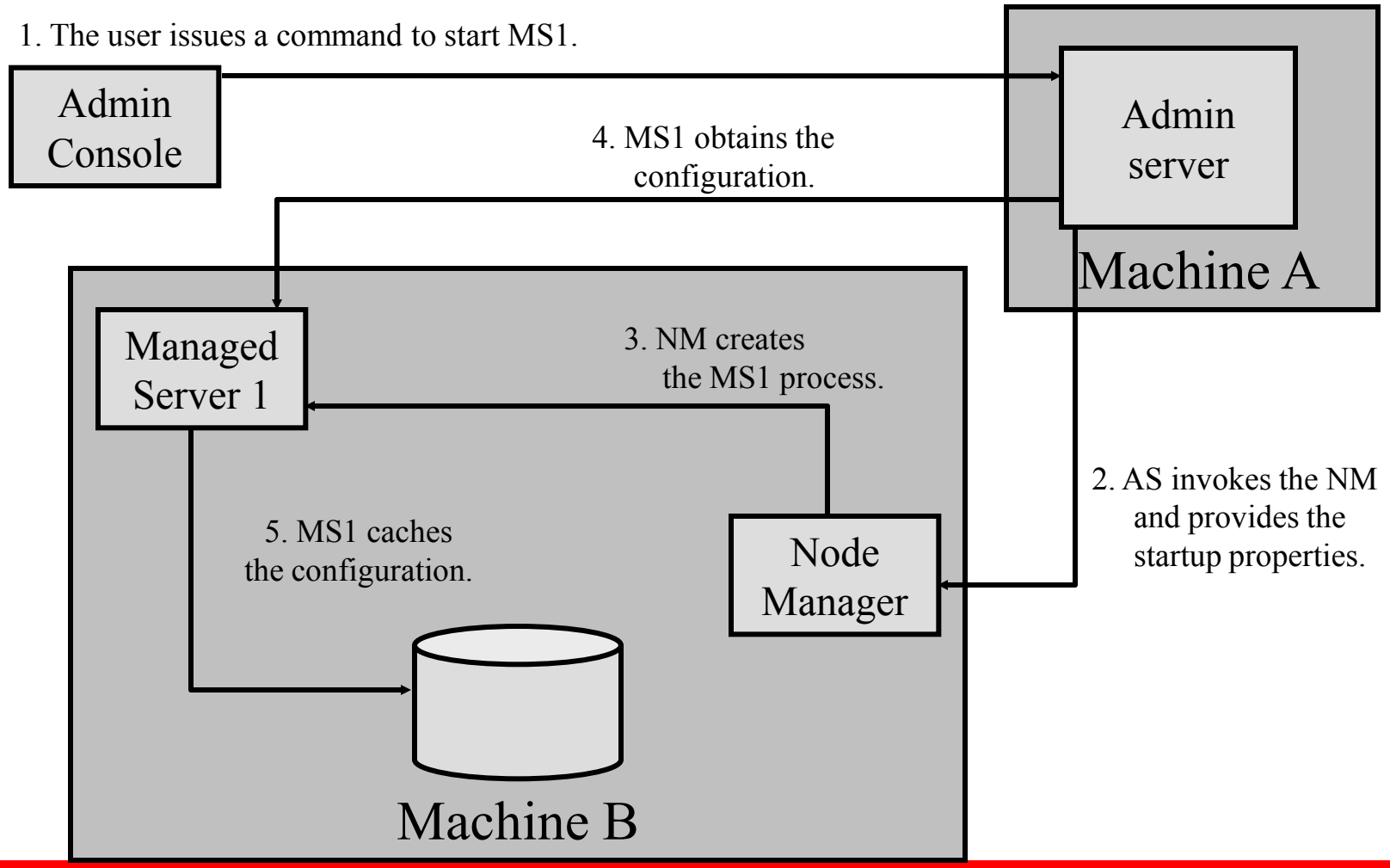
Node Manager Architecture



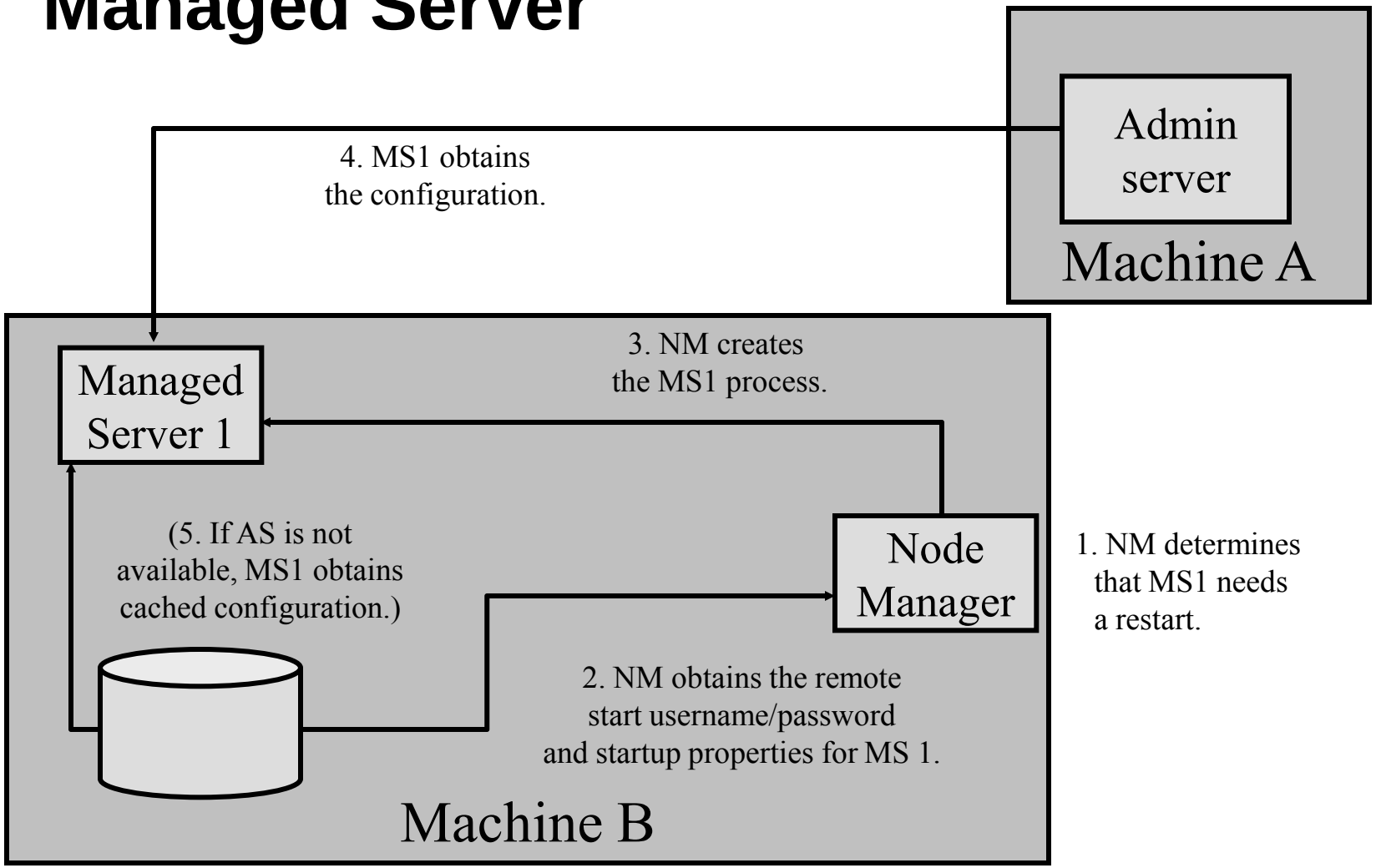
How a Node Manager Starts an Administration Server



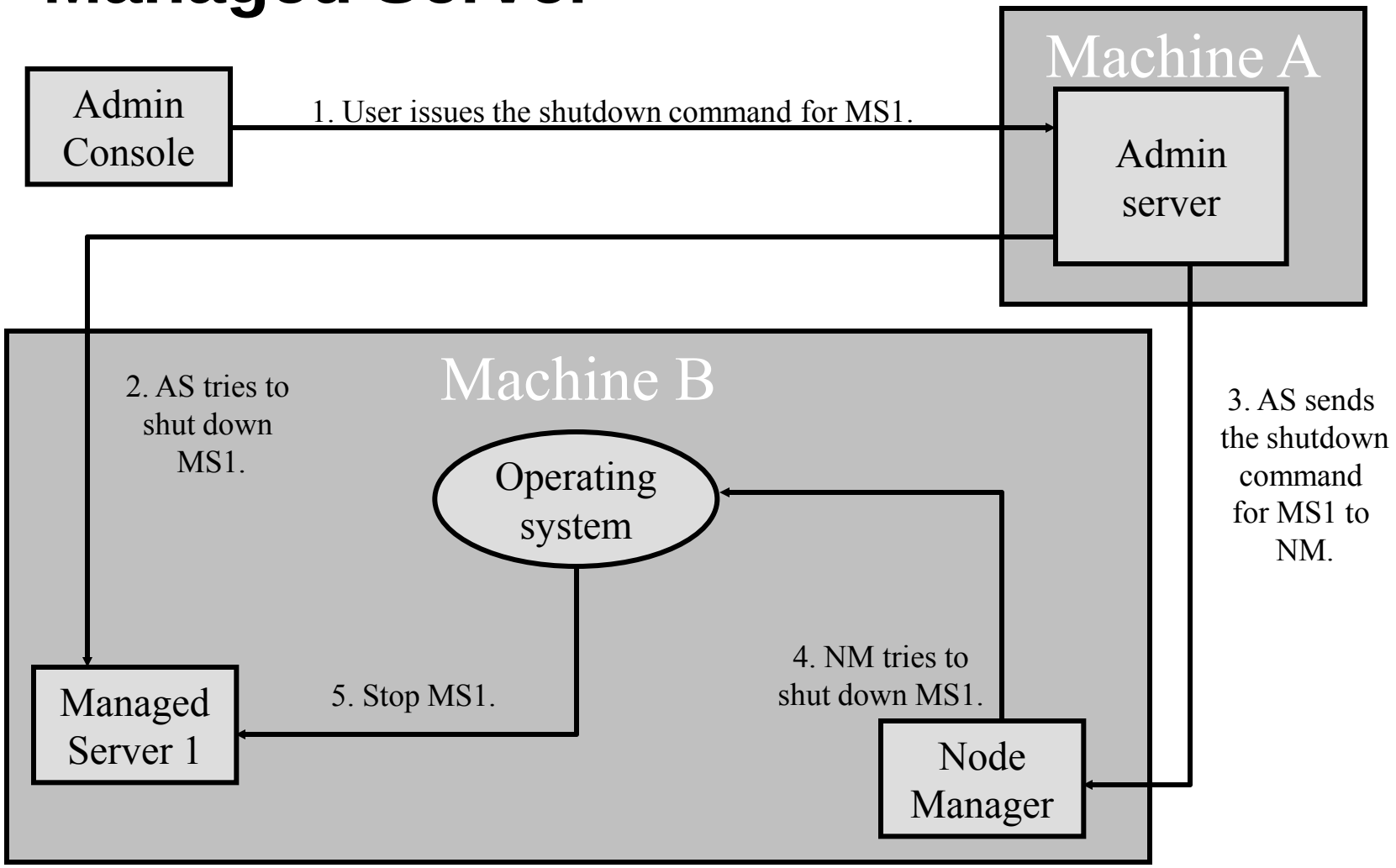
How a Node Manager Starts a Managed Server



How a Node Manager Restarts a Managed Server



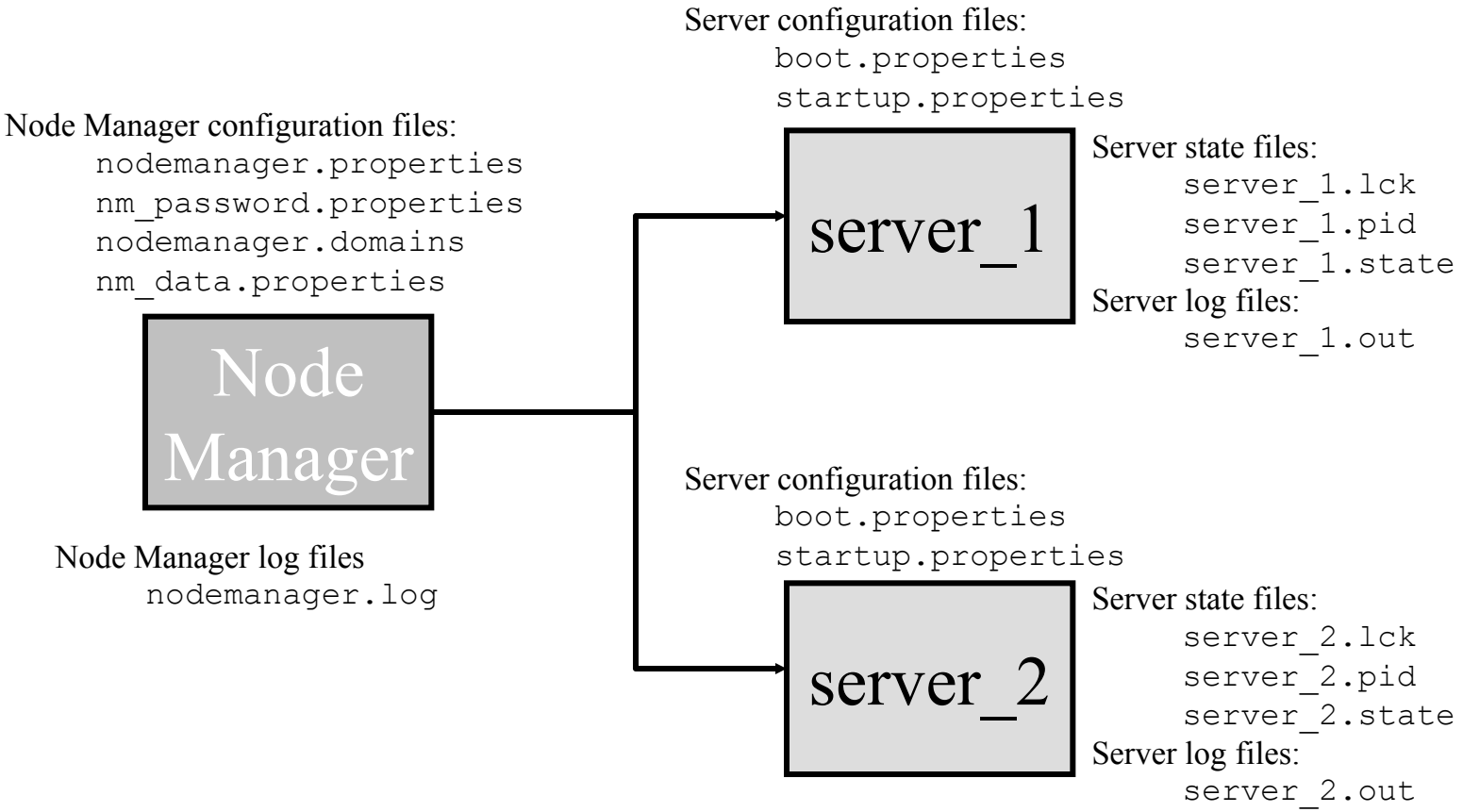
How a Node Manager Shuts Down a Managed Server



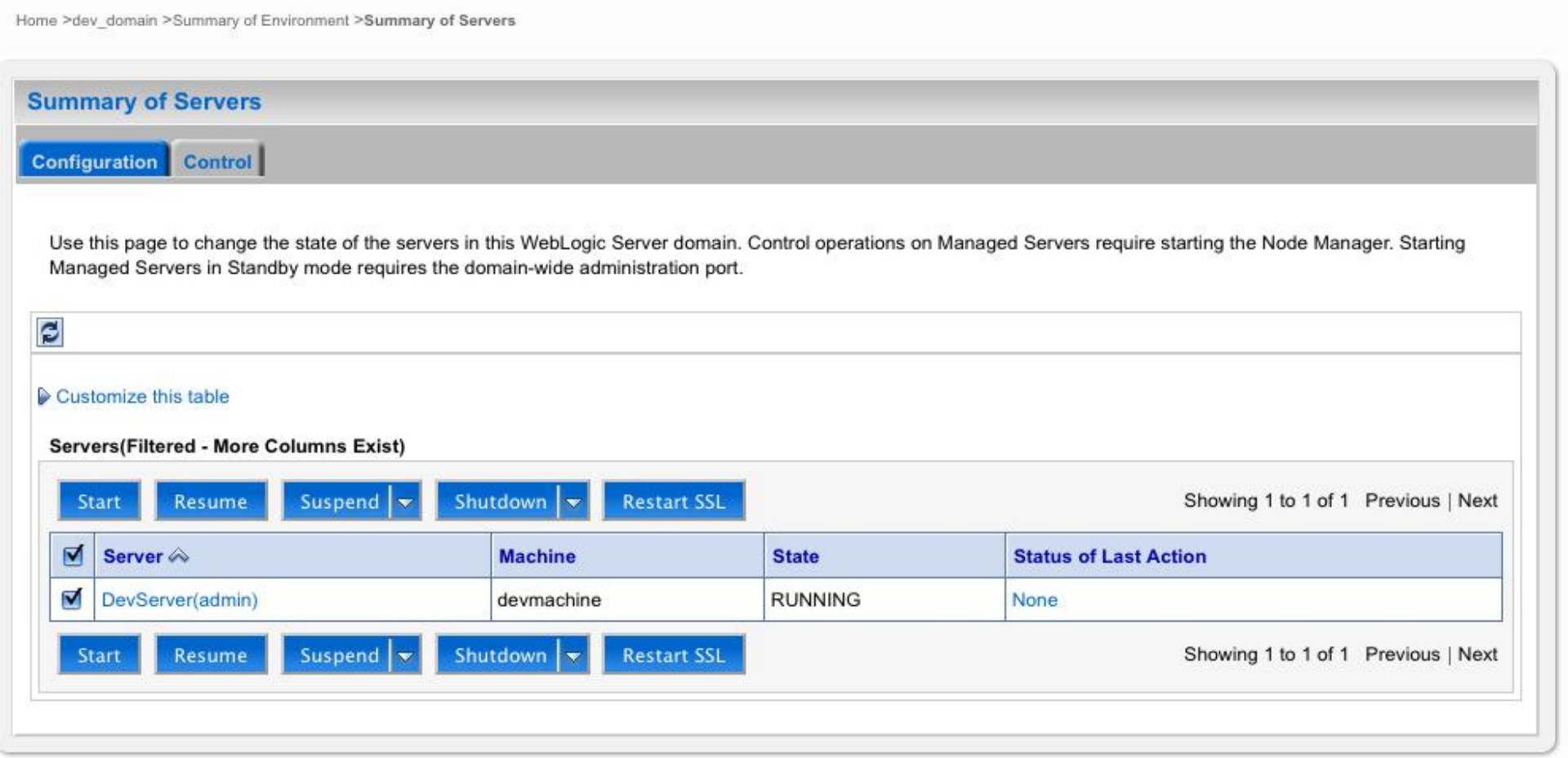
Node Manager Configuration and Log Files

- Two sets of files:
  - The Node Manager `config` files, located in *DOMAIN_HOME*`/servers/server_name/data/nodemanager`
  - The Node Manager `log` files, located in *DOMAIN_HOME*`/servers/server_name/logs` and `<WL_HOME>/common/nodemanager`

Node Manager Configuration and Log Files



Starting/Stopping Servers with WLS Admin Console



Starting/Stopping Servers with WLS Admin Console Requirements

- Admin Server has to be running
  - Hosts the Console
- Node Manager must be running to start servers from the console

Starting/Stopping With WLST Without Node Manager

- Starting

```
wls:offline/>startServer('AdminServer','mydomain','t3://localhost:70
   01','weblogic','weblogic','c:/bea/user_projects/domains/mydomai
   n','true','60000','false')
```

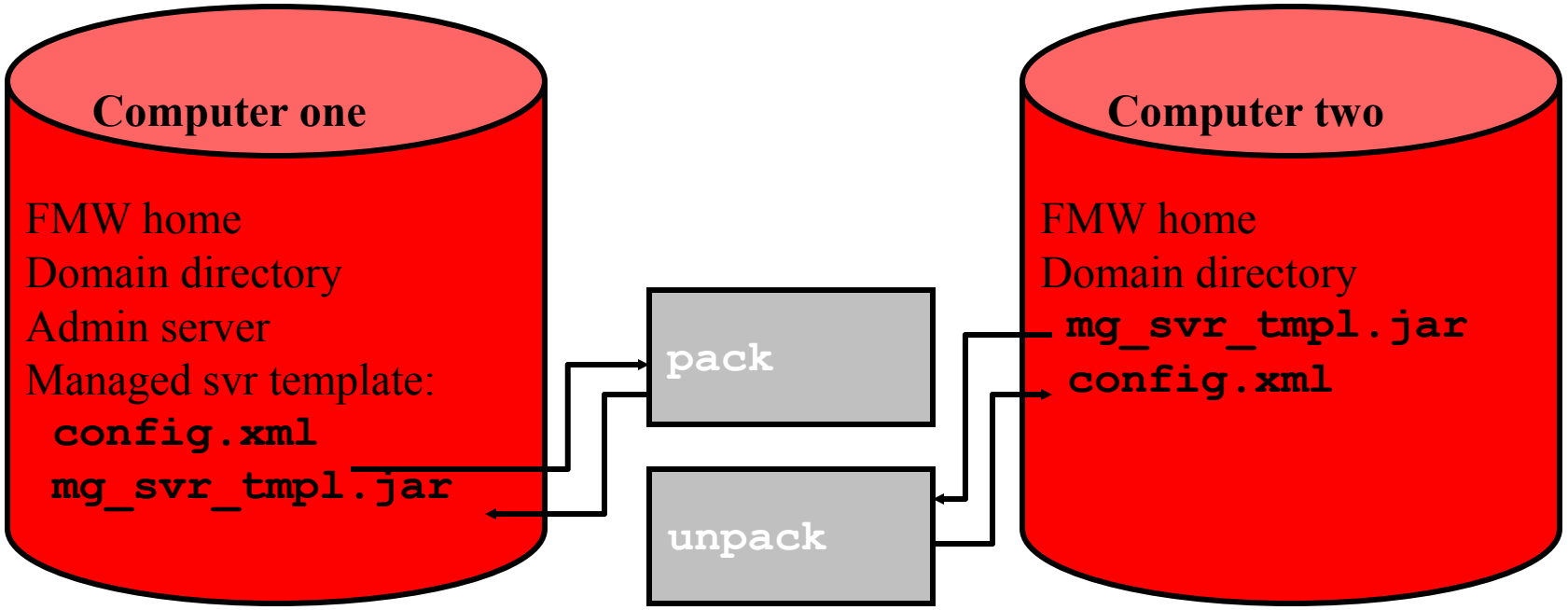
- Stopping

```
wls:/offline> connect('weblogic','weblogic','t3://localhost:7001')
….connected
wls:/mydomain/serverConfig> shutdown('AdminServer','
   Server','false',1000, block='false')
```

Creating a Managed Server on a Remote Computer

1. Install WLS on both computers.
2. Create a managed server using the Administration Console.
3. Create a managed server template using `pack`.
4. Create a managed server on a remote computer using `unpack`.
5. Start the remote managed server.

Creating a Managed Server on a Remote Computer



pack and unpack: Examples

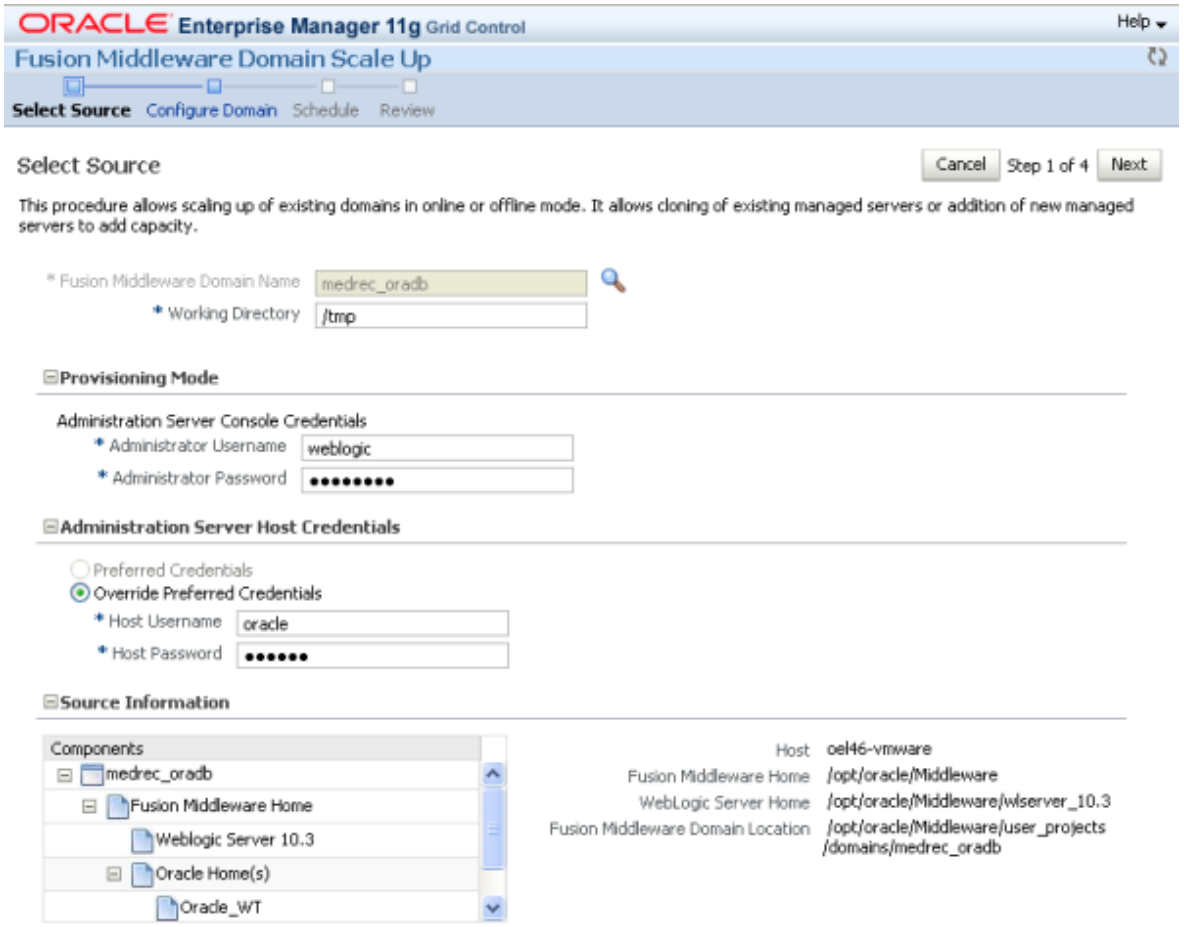
- On computer one (administration server):

```
[oracle@wls-sysadm]$ cd $WL_HOME/common/bin
[oracle@wls-sysadm]$ pack -managed=true
    -domain=/u01/app/oracle/user_projects/domains/mydomain
    -template=/home/oracle/work/mydomain_managed.jar
    -template_name="My Managed Server Domain"
```

- On computer two (remote managed server):

```
[oracle@wls-mgdsvr2]$ cd $WL_HOME/common/bin
[oracle@wls-mgdsvr2]$ unpack
    -domain=/u01/app/oracle/user_projects/domains/mydomain
    -template=/home/oracle/work/mydomain_managed.jar
```

Domain scale-up using GridControl



http://bit.ly/WLS11gDomainScaleup