**Introduction:**

**Types of servers:**Application servers, Web servers, DB servers, mail servers and File servers.  
WebLogic, WebSphere, JBoss (red hat), Tomcat (Apache), Oracle Application server and GlassFish (Oracle) are application servers.  
Apache http server and Sun IPlanet are webservers.

WebLogic performs better compared to WebSphere. If the client is having all the IBM products then they are going for WebSphere and one more reason to go for WebSphere is MQ server.

For Dev purpose also we need to pay for WebSphere. No need to pay for WebLogic in Dev. If you see any issue in production and want support then only we will pay for WebLogic.

WebLogic was started in 1995 with the name tengah and renamed to WebLogic in 1997.

Three people started a company called BEA with their initials and the initial version is 3.0.When they came up with 9.0 versions, oracle has acquired BEA systems. From 2009 onwards, it is called as oracle WebLogic server and version is 9.0.

**Note:** In 11g, ‘g’ means grid (Grid is some feature in Oracle DB) and in 12c, ‘c’ means cloud.  
**Note:** If you are using 12c WebLogic then SOA should also be of same version and not 11g.

Application Server architecture:

Static

Web container

Dynamic

EJB Container

Webserver Architecture:

Static

Web container

Dynamic

WebLogic uses JVM, so we should have some java version installed.

WebLogic supports all OS’s like both windows and Non-windows. To install WebLogic in any OS, we need 10 GB free space, 1GHz processor and 1 GB RAM.

In oracle site you will find 2 types of installers for WebLogic.   
Generic Installer: It can be used for both 32 and 64 bit. It will have only WebLogic for all the OS’s. It is a .jar file.  
OS-specific Package Installer: It can be used for both 32 and 64 bit. Along with WebLogic, you will find other supported s/w’s like JDK’s, DB drivers etc. Its extension is .etc for windows and .bin for others.

If your machine is 64-bit then you can install either 32 or 64 bit.  
If your machine is 32-bit then you can install only 32 bit.

**Note:** While installing WebLogic on any OS, you will see 3 options and they are GUI, Console and silent. Silent mode is Non-interactive and the other 2 are interactive.  
In case of silent mode we will pass all the information in a file as an input and it will be installed.

In real time we will use console mode only for installation.

**Note:** Silent mode will be used to install entire FMW which takes 72 hours to install which contains all the software’s like WebLogic, SOA, and IDM etc.

**Note:** In one system you can install WebLogic ‘N’ no.of times but should be in different folders.

To uninstall weblogic just go to /utils/uninstall/uninstall.exe. It only uninstalls the files and to uninstall completely we should delete Middleware home folder.

**Deployment:**.war (Web Archive): Web applications like simple java files, .properties, html and css files which won’t talk to any DB. It will just have jsp’s and html’s. It will not have any transactional programs.

.ear (Enterprise Archive): It will have .war along with some other complex programs like talking to DB. (.war + .jar=.ear)

.jar (Java Archive): Only java programs and .properties files. No html, css etc.

.rar (Resource Adapter Archive): xml’s etc.

**Note:** It is not mandatory to have a MS in a domain. We can only have AS also. Normally the applications will be deployed on MS. If we don’t have MS then it will be deployed on AS.

Note: Admin server is mandatory in a domain.

We can deploy an application in below ways:  
-->Using Admin console  
-->Using command line  
-->WLST  
-->Auto deploy (Only for Dev mode)

*Question:* What we need to do, if we forgot the Admin server password?  
*Answer:*   
**Note:** If you are deploying .jar and .rar file then you should choose ‘deploy this application as a library’ option in console.  
**Note:** If you are deploying on MS then you should stop the MS and then deploy and start the MS.

**Note:** Suppose you have deployed application and you want to test that application then use the Admin URL like <http://localhost:7001/console>. Here instead of console you should provide the context root of the application means http://localhost:7001/ShoppingCart.  
Or else just go to deployments in console and click on the ShoppingCart or expand the ShoppingCart and click on the URL’s you see.

**Application lifecycle states**:   
New: When you deploy an application on shutdown MS then the status will be New.  
Prepared: MS is running but if you stop the application then it will come to prepared state.  
Active: If the application is running then it will be active state.  
Installed: After deploying the application, go to targets and unselect the target then you will see installed state.  
Failed:   
Retired:   
Admin:

**Note:** If the server is overloaded then it will be in overloaded state in the console. Click on the servers in console and at the end of the console you will see all the states.

**Deploying using command line:**   
**Deployer.class** is a compiled java program and deploy (), undeploy() and redeploy() are the methods in this program.  
You will find Deployer.class in weblogic.jar and the location of weblogic.jar is wlserver\_10.3/server/lib.  
**Hot** deployment: Deploying something on running servers is called hot deployment.  
**Cold** deployment: Deploying something on stopped servers is called Cold deployment.

**Note:** OS related commands are internal commands and software (here WebLogic) related commands (startWeblogic.cmd) are external commands.

Run setWLSEnv.cmd to set the environment and then run the below java program to deploy etc...

Java weblogic.deployer –adminURL –username –pwd –deploy –source –targets –name

**Note:** Deployer is under weblogic.rar

**Note:** While deploying source is mandatory and name is optional, but while undeploying name is mandatory and source is optional.

To set the path temporarily in **Linux**export **PATH**= <Java home>  
If you close the window and open again then the settings will be lost and you need to export again. In this case the other commands like ls, clear will not work because the PATH is set to java home and other settings will be lost.  
If you want both commands to work then   
export PATH=$PATH :< Java home> but this is also temporary.  
To set the path permanently, add an entry in .profile file under /etc folder.  
In Linux also to deploy an application we will use   
Java weblogic.deployer –adminURL –username –pwd –deploy –source –targets –name

But it gives the error here because the system is not able to find weblogic.jar file. It is because we have set the PATH and not CLASSPATH.   
So use below command  
java –cp /opt/../../lib/weblogic.jar weblogic.deployer –adminURL –username –pwd –deploy –source –targets –name

**Auto deploy:   
Note:** When you click on an application in a console which is deployed using autodeploy then you will see only 4 tabs. But for other way deployments you will see more than 4 tabs.  
Auto deploy is only for dev mode (dev mode is something which you will select while creating domain).You will see a folder called autodeploy under each domain folder (domains/Devdomain/autodeploy).

Copy the .war file whichever you want to deploy to autodeploy folder. Autodeploy will deploy the application on Admin server only and there is no way to deploy on MS’s.

To undeploy, we just need to remove from autodeploy folder. There is no other way to undeploy.

We do this kind of deployments for small servers and that too for POC purpose.

In prod mode also you will have autodeploy folder but you can’t do autodeploy in prod mode.

When you deploy using wlserver\_10.3/common/bin/wlst.sh script also, it internally calls Deployer.class only.

After giving wlst.sh, we should connect to any one of the domains first. We connect to a Admin server in a domain as each domain will have only one AS. After connecting, all the linux commands will work but you should append them with (). ‘ls’ should be written as ls(). Ls() command displays all the MBeans and serverconfig is also an MBean.

**Deploy** (‘Renaming application’,’/home/Sandeep/benefits.war)—Here we are not mentioning the server, so it will be deployed on AS.

**Note:** If you want to deploy an application on all the MS’s then we can do it only by using cluster or through Admin console only.

**Note:** It is not recommended to change the **targets** from console. For example you deployed something on ‘S1’ but when you change the targets from S1 to S2 (or) if you select both the servers then it will be deployed on S2 also. But this is not a better way to change the targets.

**Note:** Update button in the console is introduced from WebLogic 9 onwards. Update means redeploy only. We do redeploy from command prompt. We can redeploy or update an application ‘n’ no.of times if there is a change in the code.

**Side by Side deployment:**It can be done only through the command prompt means   
java weblogic.deployer --- -----  
Apart from other options you will mention **-appversion** also. Using Side by Side deployment, we can deploy the same application twice either with the same name or with different name. Initially both these 2 versions will be active state. The first deployed application will serve all the old requests and after completing all of them it goes to retired state and the new requests will come to the another application.

In case of update or redeploy, we are updating the existing one with new one. It means we will have only one application at any point of time in console. But using Side by Side deployment we can have 2 versions of same application at any point of time and not more than 2.

**Note:** If you use –appversion while deploying the application for the first time then only you can deploy it for 2nd time also. Otherwise you can’t use **Side by Side deployment.**

**Note:** Normally we can deploy an application only once in a server. If you try to deploy the same application with a different name then it will be failed, because the context root will be same even though the application name is changed. Context root can only be changed by application code developer.

**Deployment Mode or staging Mode or Messaging Mode:   
Stage:** In stage mode, the Administration Server copies the deployment files from their original location (may be /upload folder) on the Administration Server machine to the staging directories (/stage folder on each target MS) of each target server. For example, if you deploy a Java EE Application to three servers in a cluster using stage mode, the Administration Server copies the deployment files to directories on each of the three server machines. Each server then deploys the Java EE Application using its local copy of the archive files.  
This is the default staging mode for Managed Servers.  
**Note:** Stage mode ensures that each server has a local copy of the deployment files on hand, even if a network outage makes the Administration Server unreachable. If you are deploying very large applications to multiple servers or to a cluster, the time required to copy files to target servers can be considerable. Consider Nostage mode for the applications which are big in size.

**NoStage:** In nostage mode, the Administration Server does not copy the archive files from their source location. Instead, each target server must access the archive files from a single source directory for deployment. The staging directory of target servers is ignored for nostage deployments.

For example, if you deploy a Java EE Application to three servers in a cluster, each server must be able to access the same application archive files (from a shared or network-mounted directory) to deploy the application.

In nostage mode, the Web application container automatically detects changes to JSPs and servlets. Nostage also allows you to later update only parts of an application by updating those parts in one file system location and then redeploying.  
**Note:** Deploying very large applications in nostage mode saves time during deployment because no files are copied.  
This is the default staging mode for the Administration Server.  
**Note:** The source for the deployment files in nostage mode is the path provided by the user at deployment time (as opposed to stage mode, where the source is the path in each server’s staging directory). However, even in nostage mode, WebLogic Server copies out parts of the deployment to temporary directories. This enables users to update entire archived deployments or parts of archived deployments.

**External\_stage:** External\_stage mode is similar to stage mode, in that target servers deploy using local copies of the deployment files. However, the Administration Server does not automatically copy the deployment files to targeted servers in external\_stage mode; instead, you must copy the files to the staging directory of each target server before deployment. You can perform the copy manually or use automated scripts.  
External\_stage mode is the least common deployment staging mode. It is generally used only in environments that are managed by third-party tools that automate the required copying of files. You may also choose to use external\_stage mode when you are deploying very large applications to multiple machines and you do not have a shared file system (and cannot use nostage mode). Using external\_stage in this scenario decreases the deployment time because files are not copied during deployment.

You can use the -noversion option to turn off the requirement that deployment files be on the Administration Server, but the -noversion option causes versioning information to be ignored so you cannot use the -noversion option with versioned applications.

**Deployment Order:** It is not useful in case of pure java war files. It will be useful for SOA deployments. If one application depends on another application then we will mention the deployment order while deploying applications.

**Note:** Whatever the deployments we do, we will never stop the Admin server. If you stop Admin then you can’t deploy anything. It is always better to stop the MS before deploying anything on that particular MS. You can deploy an application without stopping MS but it is not recommended.

--If you have Generic WebLogic installer then first you need to install JDK then run the Generic installer.  
--In case of package installer, no need to install JDK. You can directly run the package installer.  
--If your system is 32-bit then it allows only 4 GB of RAM.  
--If your system is 64-bit then it allows ‘N’ GB of RAM.  
--32 bit WebLogic software can be installed both in 32-bit and 64-bit windows whereas 64-bit WebLogic can only be installed on 64-bit windows.  
--**OEL** (Oracle Enterprise Linux) is the Linux system from Oracle.  
--12c only supports JDK 1.7 (recommended). You can also use the old versions like 1.6 and all but not recommended. Now we have java 8 means JDK 1.8 but 12c won’t support it.  
--Recommended for 11g is 1.6 and for 10g is 1.5 and for 9 it is 1.4 and for 8 it is 1.3

--Admin server will be used only for admin purpose only. MS’s will be used to execute the application code.  
--Sun JDK is also called as Hot Spot JDK.  
--We can use Demo certificate in Dev mode but not in prod mode.  
--In Dev mode Lock & Edit will be disabled by default but in prod mode it will be enabled.  
--When you start the Admin server in Dev mode then no need to provide user name and password. These details will be picked from boot.properties file. But in prod mode we should provide user name and password. If you delete the boot.properties file then for Dev mode also you should provide username and password while starting the server. You will see this file under security folder.

**WebLogic server life cycle and server states**

**Creating domain:**Using Console mode  
./config.sh –mode=console (Linux)  
./config.cmd –mode=console (Windows)  
In real time we use console mode only and not GUI because the pop-ups will be disabled in real time.

If you create Admin and MS at the same time while creating domain then for Admin server 2 ports will be created. One is for http and one is for https.

We have a folder called pending under Domain folder in Linux. When you click on ‘Activate changes’, then the changes will be reflected in config.xml file or will be merged with original config.xml file.

**Note:** If a server is already in running mode and if you are trying to start the same server again then it will throw the error saying “Unable to get file lock”, because the config.xml is already locked when the server started for first time.

**Note:** Total ports available in a system are 65536.

If you get some output for the below 2 commands then it means the port is already is being used.

Netstat ano | grep ‘portNo’  
netstat anp | grep ‘portNo’

You can add or create a new MS in different ways. Using configuration wizard (config.cmd or .sh) or directly from console by clicking ‘New’ or else you can add an entry in **config.xml** but in this case you need to restart the servers.   
We have other method to create a domain i.e. Restoration (Pack or Unpack)

**Note:** If you make any changes in config.xml then you must restart Admin server.

In HC method, we can create the MS on one machine and Admin on another machine. In this type of situations only we will provide Admin URL to start the MS’s.

**Note:** In the command prompt we will use ‘t3’ protocol instead of ‘http’.

*Question:*What is http tunneling?

**Log Files:**Servername.out is the log which will have all the server related messages. Access.log and servername.log files will be created automatically. Access.log is not of much use.  
If you are starting the servers from console or script then only servername.out file will be created. Each MS and Admin server will have separate log files.

Remote log file on startup: By default it is checked in Dev mode and not in prod mode.

*Question:* What is domain?  
*Answer:* Group of WebLogic resources like servers, Data sources, application deployments etc...

*Question:* Why do we need to create a domain?

**Node manager:**   
We will have only one NM for all the domains.

We have 2 types of NM’s  
Java-based NM: It is the default one which will be installed along with WebLogic and it works with both windows and Non-windows. It is more secure.  
  
Script-based NM: It works only in Non-windows. You need to write a script (Perl or python) to create script-based NM.  
**Note:** Up to WebLogic 8.1, the default port no for NM is 1555 and from ‘9’ onwards it is 5556. You can change the port no which is available in nodemanager.properties file, if you want in both the versions.

**Note:** From 12c onwards, you can create a NM for each domain. This is called Domain specific NM’s.

**Cluster:**   
Cluster is nothing but a group of Managed Servers (WebLogic Server instances). All these MS’s can be from same machine or different physical machine. But all of them should belong to same domain.  
If the MS’s are on different hosts then the hosts should have same OS, same hardware like 32 bit etc. and also same WebLogic version.  
Use of cluster is high availability, load balancing, Fail over and Scalability (Scalability means you can add or remove the servers to or from cluster).

There are two methods of clustering:  vertical scaling and horizontal scaling.  
**Horizontal clustering:** Horizontal scaling is more reliable than vertical scaling, because in HC multiple physical machines are involved in the cluster environment, as compared to only one physical machine in VC.

**Vertical clustering:** However, consists of multiple Java application servers on a single physical machine. With vertical scaling, the physical machine's processing power, CPU usage, and JVM heap memory configurations are the main factors in deciding how many server instances should be run on one physical machine.

**Note:** In a cluster if a server stops while processing a request then the other server will process the request from where it has failed by using something called session ID’s. This is called Fail-over situation.

**Note:** Whether it is HC or VC, all the MS’s should belong to the same domain.  
**Note:** Admin server is also a WebLogic Server instances but it can’t be part of cluster.

**Note:** If we have AS in one physical machine and MS’s in other physical machine and you are grouping the MS’s then also it is VC only. Because AS can’t be part of a cluster.

**Note:** We should have at least 2 MS’s in a cluster and max is N.

**Note:** We can use the same port No for any purpose if the IP address is different.

**Note:** Even though the admin server is shut down, you can test the applications which are deployed on MS’s, but you can’t deploy the applications and also can’t login to admin console.  
To test the application you will give <http://hostname:MS-PortNo/Application-ContextRoot>

**Note:** If we have a webserver then whatever the requests you sent, first it will go to webserver (In real time we will have DNS above the webserver and above DNS we may have network load balancer. The setting of DNS and N/W load balancer will be taken care by other team and not by us. We are responsible to install webserver and below that) and then webserver will route the request to the MS whichever is free.  
If we have a webserver then to test the application we will give   
<http://hostname:WebServer-PortNo/Application-ContextRoot>  
by default the webserver port no is 80. If the port no is 80 then it is not mandatory to mention it in above URL. If it is other than 80 then we need to mention.

**Note:** If we get an error message like “Address already in use, JVM\_Bind” then it means that the same port no is being used by other process.

**Data source:**Why we need a data source when the application developers are mentioning everything like DB name and other details to connect to the DB using JDBC or other functions in the application code itself.   
 Because each DB we are trying to connect will have a limited number of connections. You can buy more connections by paying the money. For example consider we have 50 DB connections but we are getting lot of requests from different applications to connect to the DB and do some operation like select, update etc. At any point of time the DB can serve only 50 requests and the remaining requests will wait for their turn. But if we have lot of requests then the DB server will be crashed.

To control the number of requests, we have a feature called connection pool in data source where you can mention the number of connections. If you mention 50 in connection pool then only 50 requests can go to the DB.

One more reason to use the data sources is, for example you are changing the DB means instead of Oracle you want to use DB2 then in the application code you need to make the necessary changes to connect to DB2 database and then deploy the code. In this if we have data source instead of making the changes in the code we will make the changes in data source itself.

Data source is like a pipe means to establish the connection between your application and the DB and connection pool in the data source is like size of the pipe means how many requests can go to the DB at a point of time.

**JNDI** is the one where we will have all the DB details. Using this JNDI only, application program will connect to the DB.  
Either we will create the JNDI and asks the programmers to use it or else he will use some name and asks us to create a JNDI with the same name.  
**Note:** Before mentioning the number of connections in connection pool we should contact the DB admin to know maximum no.of connections allowed on that DB server.

In Dev mode, the default connections will be like initial capacity is ‘1’, minimum capacity ‘1’ and maximum capacity ‘15’.

In Prod mode, the default connections will be like initial capacity is ‘5’, minimum capacity ‘5’ and maximum capacity ‘25’.  
If you mention a value other than ‘0’ for initial capacity, then that many connections will be established between the DB server and the application when we start the WebLogic server. If the DB server is down then connection will not be established and the warnings will be thrown. Because of these warnings the Admin server will come into Admin state and not running state. So it is better to keep initial capacity value as ‘0’.

Min.capacity means it will try to open the no.of connections mentioned for this apart from initial capacity.  
**Note:** Whenever you start a server, it will try to load the applications which are deployed on it and also the connections into the memory.  
**Note:** If you see a server in Admin state means there is some deployment issue or some warnings are there while starting the server. Using console you can resume the server from Admin state.  
**Note:** Initial capacity can be less than or equal to max.capacity. Mostly in real time we keep the initial capacity as ‘0’.

Up to WebLogic 10g, we have 2 types of data sources and they are 1) Generic or simple data source and 2) Multi data source. From 11g onwards we have one more data source i.e. Grid link data source.

**Note:** Grid link DS works only with Oracle DB which is having RAC.  
  
Up to WebLogic 10g, by default we get a DB called PointBase DB. In 11g we have a DB called Derby. In 12c TopLink (may be) is the DB. All are file based (may be).  
**Note:** You can point Derby only to Admin server and not to MS’s. If we have a full version of Derby then you can point it to any server. But the default Derby which we get with WebLogic is a basic version. By default the port no for Derby is 1527 and for Oracle it is 1521. You can change these values.

In WebLogic 11g, the Derby server will not be started automatically. You may see some errors related to Derby when you start the WebLogic server. You need to start Derby using startDerby.cmd (username and pwd for Derby is test only). In 12c, Derby will be started automatically.

Up to WebLogic 8, there were separate links for creating data source and connection pool in the console and also the data related to both of these used to be saved in config.xml along with all the other configuration information. But from WebLogic 9 onwards, only the data source location details will be maintained in config.xml along with other configuration information. Other details of data sources will be mentioned in other files.

While creating the data source we will select the DB driver also. We have 4 types of drivers and they are type 1 to type 4 and in each type we will have XA and Non-XA. Type 4 is the one which will be used for java applications (Whatever the DB you choose you will select type 4 only). The purpose of these drivers is to open/close the connection to DB. The drivers will tell the data source and the data source will tell connection pool.

In real time mostly we use Multi data source only. We can use Multi DS with RAC also apart from normal DB environment (means 2 or more grid DS’s).  
For creating Multi DS, first we need to create Generic DS’s then we need to create Multi DS and add the Generic DS’s to it. The application programmers will use the JNDI of the Multi DS in their code to use the DS’s.

**Note:** The purpose of using multi DS is, if one of the DS is crashed because of lot of requests then the other in the same group can handle them (Multi DS is like a cluster).

**Note:** The DB’s to which these DS’s are connected can be of same DB type or different (one can be oracle and other can be DB2). But both of them should have same data tables.

**Note:** For example one application has done the insert operation on one of the DB but these records are not available on other DB. To maintain the same data in both the DB’s, at some regular intervals the data will be copied from one DB to another DB and vice versa. If we won’t copy the data then if one application issues select operation then the query will be failed.

**Note:** If we don’t have same data in the DB’s then using multi DS is of no use. So we should manually copy the data.

Instead of manually copying the data from one DB to other, we can use RAC. RAC will have minimum of 2 servers and maximum can be 32. But RAC is very costly to buy and only oracle DB is having this feature. If it is really required then the client will go for RAC otherwise not. Using multi DS (multiple Generic DS’s) also we can connect to RAC. And using Grid Link data source also we can connect. Grid Link is only one DS and not collection of DS’s and it takes care of all the server instances inside RAC.

**Doubt:** When we can use multi DS for RAC then what is the use of Grid Link DS whose purpose is to serve only RAC environment.  
**Doubt:** What is FAN (Failure at Network?)

**Patch:**Patch is nothing but a fix for a problem.  
We have 2 types of patches: 1) **Service Pack** and 2) **Maintenance Pack**  
In 10.3.1.1, 10.3.2.2 and 10.3.3.3, all the first 1, 2, or 3’s are called MP and the second 1, 2 and 3’s are called SP. MP may have lot of bug fixes and SP might have single bug fix.

**Critical Patch Updates** (CPU) is a collection of security fixes for Oracle products. Oracle *strongly* recommends that customers remain on actively-supported versions and apply Critical Patch Update fixes *without* delay.

To download the patches you should login to either My Oracle Support (**MOS**) (support.oracle.com). It is also called as **Oracle Metalink**. You can login to this site with your Gmail or other account also but you can’t create SR’s (Service Requests). You should login with your company id if your company as the valid license.  
If you see any issue or bug in your WebLogic server then you can raise an SR. If it is a valid bug then they will take up your SR and assign a bug number for your issue. They may take 3 months of time to release next patch which will have the solution for your bug along with other bugs.

Or else while installing the WebLogic if you select ‘want to receive regular updates’ button then when you run the **bsu.cmd** (It comes by default with 11g), in the **pop-up** itself it will show all the newly available patches.  
**Note:** We should apply the patches on the WebLogic folder i.e. weblogic\_10.3, then the patch will be applied to all the domains under this server. We can have same version of WebLogic server installed more than once on a single physical machine (One IP address). For each installation we will have a separate WebLogic folder.  
For each installation the folder structure will be (or whatever the folders you choose) like below. The versions of the WebLogic may be same or different.  
D:/Middleware/WebLogic/servers/wlserver\_10.3/Utils/bsu.cmd  
D: /Middleware2/ WebLogic/servers/wlserver\_10.3/Utils /bsu.cmd  
.  
.  
Etc...

When you run the bsu.cmd command from any one of the folders, it will show all the WebLogic server installations on the left hand side of the **pop-up**. When you select the particular installation, it will show the patches available to apply on that server (If you select ‘want to receive regular updates’ while server installation then only you will see the available patches) in ‘**Get patches**’ tab. Select the patch which you want to apply and click on download. Now go to ‘**Manage patches**’ tab and there you will see the downloaded patch and select it and click on apply and the patch will be applied to that server.

**Note:** Whether you are applying the patch or removing the patch, both the times it will check for any conflicts with other patches in both GUI and command line modes.

**Note:** You can apply the patches on running servers and not running servers also.  
 **Note:** For example we have some applications and all of them are running on only one WebLogic server in different domains. We have some issue in one of the application and you wanted to apply a patch to fix that issue. The patch will be applied on all the domains and not on any particular domain. Because of this patch the troubled application may work fine but the other applications may face some issue. In this case it is better to have multiple installation of WebLogic server and keep the applications on different WebLogic server installations instead of same installation.

**Apply patch through command line:**In real time mostly we use command mode to apply the patches.  
First download the patch from Meta link and then create a directory called cache\_dir under D:\Bea\_Home\utils\bsu\ and copy the patch to **cache\_dir** folder (Patch is available has a zip file in Meta link site and when you extract it you will find one .jar file and one patch-catalog.xml file which tells where exactly you have to apply the patches along with readme.txt file. But in 12c when you extract the zip file you will find a folder and under that folder you will see all these files).  
Now run the below commands to see the list of options of bsu.cmd and also to apply/remove the patch etc.   
**bsu.cmd** –view (It will display the list of options for bsu.cmd)  
**bsu.cmd** –view –status=applied –prod\_dir= D:\Bea\_Home\wlserver\_10.3 (It will display if any patches are applied to this server)  
bsu.cmd –install –patchlist=<name of the jar file> -prod\_dir=<Directory of the server> (This command will apply the patch. If the directory of the patch is other than cache\_dir then you should mention the path of the patch using –patch\_download\_dir)

**Note:** After applying the patch, if the server won’t start or if the issue still not resolved then remove the patch. To remove patch, instead of –install give –remove in above command.

**Note:** In 11g, **bsu** is the default one and in 12c **opatch** is the default one to install the patches. You can use opatch in 11g and bsu in 12c but you need to download them explicitly.

**Note:** Not only WebLogic patches, for applying patches for any Oracle products we will use **opatch** and **bsu** only.

*Question:* When the server comes to Admin state even though there are no deployments exists?  
*Answer:*

**Notes on 12c:**--In Non-windows, you can’t install 12c with root (\) user  
--While installing, Installer will pre-check for the CPU, RAM etc…  
--In 12c, you don’t see typical and custom mode installation. By default all will be installed. You can’t customize the installation.  
-- Will have only 4 steps while installing 12c  
--In 12c, JDK will be in middleware/Oracle\_Home/JDK, but in 11g it will be under middleware only.

**Opatch:   
Note**: We don’t have GUI mode to apply the patch in 12c.  
First download the patch from MIS site and paste it in some directory. When you extract this zip file you will see one **Bug number**. While applying Opatch you will give this bug number only.

Opatch apply <Number> -jdk %JAVA\_HOME%

Here the %JAVA\_HOME% is D:\middleware\oracle\_common\JDK and for other installation you will have separate JDK. We will have only one version of java for 12c that’s why we are mentioning %JAVA\_HOME% for Opatch.

After applying the patch, if you give “Opatch lsinventory”, it will not only display the main patch number or bug number but it also displays all the bugs under this.

To roll back the patch, issue the below command

Opatch rollback –id <PatchNumber>

**Note:** To apply SOA patches, instead of %JAVA\_HOME% we will give SOA home directory.

**Upgrade:** In the same version if you want to upgrade to other subversion like from 10.3 to 10.4 or 10.6 then it is called upgrade.  
**Migration:** 11g to 12c or 10g to 12c is called migration, means one version to other version.   
In 12c, we have **reconfig.cmd** to migrate a domain or NM from previous versions to 12c. There will not be any folder created in 12c but the domain or NM which is in 11g will be migrated to 12c.

In the same way we have **upgrade.cmd** in 11g which is used to migrate the domains in previous versions to 11g.

**Note:** Before migrating the domain, everything should be shutdown including DB which is recommended.  
**Note:** If you want to take a backup of the domain before migrating then use **Pack** command.  
**Note:** If the 11g domain is in DEV mode then 12c domain will also be created in DEV mode only. The mode should be same while migrating.  
**Note:** In 12c we have dynamic cluster.

**Weblogic Security**Authentication: Authenticating the users to login  
Authorization: After logging in, authorizing the users to access something.

**LDAP:** Below are the different LDAP systems from different vendors  
--Oracle IDM  
--Microsoft AD (Active Directory) (Mostly we use this one but it works only on windows)  
--CA Site minder (We use this for Non-windows)  
--IBM TDS  
--Open LDAP

Each LDAP system will have separate DB and will be maintained by separate teams.

**Note:** What is the purpose of **Radio button**?

*Question:*what is **coherence server** or **coherence cluster?**

**JMS:**JMS will be used only with java applications, because JMS doesn’t convert the messages (Sender sends the messages and these messages will be stored in JMS in byte format. Then these messages will be sent to receiver and receiver converts the messages from byte format to Normal format).

JMS comes by default with WebLogic. To configure **IBM MQ** in WebLogic, under Queue and Topic you will see foreign server. Select this option and give the MQ server details.

The other messaging systems like IBM MQ or MS MQ or TIBCO can be used with any applications because these systems convert the messages to xml format and xml is the universal language which can be used with any application like java, .NET, mainframe etc…

**Note:** In 12c, we have a cluster concept for JMS servers also.

**Note:** Persistent store is something which you will see while creating JMS Server. We have 2 types of persistent store. One is DB and another one is File. In real time, mostly we use File store as persistent store. If the file store is corrupted then the server will not restarted properly. In this case either rename the file store (If you want to take a backup of the messages) or else delete the file store. When you restart the server successfully, the file store will be created automatically.

**Thread Dump  
Note:** Each thread can handle only one request at any point of time. After serving the request the thread will be closed and a new thread will be created (Why a new thread will be created. Once the thread is free, why can’t we use the same thread for other requests? It is because; if we use the same thread then it will be difficult to find which thread is causing the issue when we generate thread dump.).

Thread dump is a snap shot of all the threads created by JVM at a certain point of time. When you run a command to generate a thread dump then it will generate the dump of the threads at that point of time.  
**Note:** Admin server will have its own JVM and in the same way each of the MS’s will have their own JVM.

On windows we will use below commands to generate thread dump  
Jstack –l <PID> (For Sun JDK)   
Jrcmd print\_threads <PID> (For JRockIt)

On UNIX systems below command will be used to generate thread dump  
kill -3 <PID> (It only generates the thread dump but not kills the process)

For JRockIT, the thread dump looks like below  
======FULL THREAD DUMP=====

=====END OF THREAD DUMP=====

**Note:** PID is the process id of the server. For each server we will generate the thread dump if it is having any issue.  
*Question:* Where exactly the thread dump file will be created?  
*Answer:* There is no new file will be created. The thread dump will be appended to the .out file. If you have not started the server with NM itself then you will not have any .out file. In this case thread dump goes to command line only.

*Question:* Difference between server hang and server crash?  
*Answer:* Server hangs means Process ID will be there but the server will not respond.  
Server crash means, the server will be shut down and it will not have any process ID.

Thread states:   
Running—healthy  
Idle—healthy  
blocked—unhealthy  
stuck

**Need to write more on this and also check the doc provided by Venu on Thread dump**

**Note:** If we have some issue with the server and want to resolve that issue, if it is mandatory to restart the server then we will restart the server. All the threads will be released, when we restart the server. We should take minimum of 3 thread dumps before restarting the server with a gap of 20 seconds. The time will depend on the applications. If the applications won’t get any new requests in 20 seconds then there is no point in taking the thread dump for every 20 seconds. After taking the thread dumps with a gap of time, we will compare these dumps and decide whether to restart the server or not. If a minimum of 2 threads are continuously in Blocked state in all the 3 dumps then we can think that there is some issue.

**Note:** If we have raised any SR with Oracle then they will ask at least 10 thread dumps.

**Note:** In thread dumps you can see which session is causing the issue. Go to DB and generate the ASH (**Active Session History**) and here you will see the session which is in thread dump. Now you can kill the session.

**Note:** **ulimit** is configured in a file on UNIX systems. Check that file.

**Socket Muxer threads**:   
2 to 5 threads will be treated as Muxer (Multiplexer) threads by default by WebLogic. Muxer threads will take the requests and gives them to idle threads.

**Note:** **Samurai tool** shows the Muxer threads as Blocked threads which is incorrect. In real time we use Samurai tool to analyze the thread dumps. We just import the .out file in to this tool and based on the color we can say which color is of which type.

Oracle **TDA** (**Thread Dump Analyzer**) is another tool to analyze the thread dumps.

**Note:** If you see “Open lock Chains” in log files, it means the threads are locked but they can be released at a later point of time.

**Take 2 docs from Venu for thread dumps**

**What is thread pool and thread queue**

**Heap dump or Core dump or RAM**Heap dump is nothing but a snapshot of all objects created by JVM at a given point of time.  
If we have any memory issues then we generate heap dump.

**Note:** If you have seen any <channel conflicts> error in log files, it means port no is same.

*Question:* Can we deploy applications on web server?  
*Answer:* We can deploy only web applications (.war) on web server.

**Note:** Click on each server->monitoring->threads: It will show the threads  
server->monitoring->performance.

**Note:** JVM or core or server. All are same.  
The RAM/Heap memory will be divided into 2 parts  
Heap Area for Sun JDK  
Min Heap – 256 MB or 256m  
Max Heap – 512

Heap Area for JRockIT  
Min Heap – 512 MB   
Max Heap – 512  
  
PermGen Area  
Min – 48 MB  
Max= (Min.HeapArea)/2—PermGen should not be more than this.

PermGen is the space which is used to store server internal libraries, DB drivers and the .class files which are generated from JSP files.

PermGen will only be cleared when the server shuts down.  
  
**Note:** If you have any memory issues while starting the server then it is the issue with PermGen. We should increase it by going to setDomainEnv.cmd and increase the value of PermSize.

Heap Area PermGen Area

A heap is a general term used for any memory that is allocated dynamically.  
Heap Area is divided into 2 parts and they are Young Gen and old Gen.   
Old Gen is used to store old objects and Young Gen is for Young objects.

Young Gen Old Gen

Young and old Gen will be like below. Sometimes young will takes more space and sometimes old.

If you have any memory issues after starting the server then it is a heap area issue.

Run the below commands to generate heap dump for both on windows and on Linux.  
  
**Jmap.exe** –dump:format=b,file=D:\NewFolder\abc.hprof PID (Here PID is server PID and hprof means heap profile)  
  
**jrcmd**—It is for JRockIT

**Note:** **JVisualVM** is another tool which generates heap dump and in this tool if you click on thread it will also generates thread dump. But in real time we won’t use it.  
**Memory Analyzer** (tool from Eclipse) is the tool which is used to analyze the heap dump.

Garbage Collection:   
GC is used to clear the unused memory.   
By default GC goes only to old Gen. It is called partial GC. When you run GC manually then only it goes to both Young and old Gen. It is called Full GC.  
GC (Garbage Collection) never goes to PermGen.

GC uses 3 **algorithms** and they are CMS, parallel and serial. By default it will use CMS (Concurrent Mark and Sweep). CMS means first it will mark the unused objects and then will release them. CMS and parallel works mostly in the same way.

Either in setDomainEnv.cmd or in console->server->configuration->serverstart->Arguments will mention the arguments for GC.

Heapdumponoutofmemory—when you mention this option then the heap dump will be taken before the server gets shutdown when we get out of memory errors.  
**Note:** Whatever it is, before generating the heap dump the GC will run and will delete the unused objects.  
**Note:** It is better not to run GC manually more no.of times, because it may delete currently used or running objects.

**Note:** max size of the heap dump is RAM size. Normally the size of the heap dump will be 2 GB. Sometimes it may be less also if no objects are created.

**Ticketing tools:**Client or users will come to the Global Support Desk and these people will raise the tickets and assign it to the particular team.  
  
For all the below type of requests we will have a priority. For p1, we will have 4hrs and for p2 is 8hrs, p3 is 24hrs and p4 is 72hrs. The time will depend on the clients. Based on this time also we can decide the team size.  
SR (Service Request): will be raised to pass the info like list of servers etc...  
INC (Incident): Something which is unexpected like server down etc...  
PR (): If the same issue happens again and again for the same server then we raise PR. For example if a server is getting restarted frequently with out of memory exception then we raise PR. (For any environment)  
CR (Change Request): Any kind of configuration changes like adding servers, clusters, deployments, changing port No, applying patches, certificates etc… then we will raise CR. Whenever you raise a CR, you should have backup plan if the change won’t work properly.

After taking the approval from all the related/dependent teams, change management team (CMT) will approve your change. Before approving the change along with other changes (if any), CMT team will arrange below meetings.  
**CTR** (Change Technical Review)   
**CAB** (Change Authorization Board)—these people will review the approvals you got.

**Note:** If the client is a UK or US client then we won’t implement any changes in December and it is called Change Freeze time.

L1—team will do basic Admin activities  
L2—some advanced admin activities  
L3—If L2 team won’t solve the problem then it comes to L3 team. More than 7 yrs. of experience people will be placed here.  
**Note:** If L3 team also won’t solve then Oracle will be involved.   
L4—Architects  
**Note:** In some companies you won’t see these groups at all.

**Silent Mode:**It is used only for WebLogic installation and we can’t create a domain using silent mode. For creating domain you can use either GUI or console mode.  
  
wlserver\_10.3.exe –mode=silent –silent\_xml=filename.xml –log\_file=somefilename  
Here filename.xml is the one where we will mention server and other details required for installation.  
You can get the sample filename.xml from net. Use the –log\_file, if you want to redirect the output to a file.

**Upgrade:** In the same version if you want to upgrade to other subversion like from 10.3 to 10.4 or 10.6 then it is called upgrade.  
**Migration:** 11g to 12c or 10g to 12c is called migration, means one version other version.   
In 12c, we have **reconfig.cmd** to migrate a domain or NM from previous versions to 12c. There will not be any folder created in 12c but the domain or NM which is in 11g will be migrated to 12c.

In the same way we have **upgrade.cmd** in 11g which is used to migrate the domains in previous versions to 11g.

*Question*: What is 2-phase deployment?  
*Answer:* In development we will have only one-way deployment. But when you deploy the application in production, it comes to pending state first and when you click on Activate changes it comes to active state. This is called 2-phase deployment.

**Note:** If any deployment issue is there then the server comes to **admin state** in **production mode**. Here the server is telling you that there are some issues while starting the server. But in **development mode** the server will comes to Running state even though some warnings are there. (Deployment means apart from applications we will have JDBC etc…)

**Enable Admin port:** Click on Domain->General->Enable Admin port. The port no is 9002 by default. For any running server including Admin server if you click on ‘**Force suspend now**’ then they will come to Admin state. If a server is in Admin state then no one can test the applications deployed on that server. In this case ‘Enable Admin port’ and use this port to test the applications. If you want to enable the admin port then you should have valid certificates installed (demo or self-signed certificates can’t be used) and use the <https://hostname:9002/ApplicationName> whichever you want to test.

**--‘ulimit –a**’ will display the no.of processes that particular user can create.By default it is 4096. In log files if you see the error ‘Out of Memory: unable to create **Native Thread**’ then in this case it is better to increase the no.of processes (related to ulimit command only). Check for other solutions also.

Installation issues:--No enough space  
--No proper permissions (may be folder permission or permission to execute .exe file)  
--trying to install 64-bit in 32-bit OS

Types of domains:  
--Development mode domain  
--Production mode domain (We can’t use Demo certificates)

**Note:** If config.xml is corrupted then enable ‘Archive configuration’

*Question:* Is Admin server mandatory in a domain?  
*Answer:* Yes. AS is mandatory but not MS’s. If AS is running then only we can make the changes like deployments etc. Because while deploying we will mention AS URL only. Note that we can deploy applications on AS also.

Issues while starting a server:  
--We will see **port conflict**, out of memory, deployment issue or authentication issue  
--If there is no disk space then **server can’t create a new log file** and server will not start.

*Question:* Why we connect to an Admin server after issuing **wlst.sh**?  
*Answer:* We will have only one Admin server per domain. Connecting to the Admin server is nothing but connecting to the domain itself.

*Question:* What are the advantages of **side by side deployment**?  
*Answer:* Down time is not required and at any point of time we can have only 2 versions of application. If you want to deploy 3rd version then it will fail. You should delete one of the first 2 versions then only 3rd will be deployed.

*Question:* Deployment modes or **Messaging modes**?  
*Answer:* Stage, No stage and external stage.

Deployment issues  
--may be issues in application code  
--trying to deploy the same application again  
--memory issue (As we know that we can deploy the applications on a server which is shut down. You will not see any memory issues as the server is shut down. But when you start the server you will see memory issue.)

Data source issues  
--May be DB down or JNDI issue etc…

*Question:* When you will be not able to access the deployed applications?  
*Answer:* If the server is down or the application itself down.

*Question:* If the application is not accessible all of a sudden then what all you will check?  
*Answer:* First we will hit web server by issuing <http://localhost>:80. Then issue <http://localhost:80/applicationName>. In this case it goes to web server first and from there it goes to cluster and if we are getting the response then issue <http://localhost:7011/applicationName>, it goes from web server to application server.

--**Top** is the command to see the CPU utilization in Non-windows. Top’s equivalent is **Task manager**. If a process is using more CPU then we may kill it.

--Use ‘**Uptime**’ in Non-windows. Sometimes it will show 3200% because if we have 32 CPU’s on that OS then it will show this much (32\*100=3200)

**25th July:**Q: Use of Node manager?  
A: Main use is, to Auto restart the failed servers (If you want to disable Auto restart then go to /Nodemanager/startup.properties) and Remote Administration.  
  
--Script based NM is not secured and it works only on Non-windows. Java based NM is secured and works both on windows and Non-windows.

--Machine is mandatory to use NM

NM

NMA

AS

M1

S1

M2

S2

--We will see some small issues while starting NM, like if in machine you have mentioned plain and in NM you mention SSL then it is a mismatch. Actually it should match.  
--Based on the business criticality we decide whether to use VC or HC.  
  
--With the use of Unicast and Multicast, cluster will know which server is running and which is not.  
In case of Unicast, one main server will be there and all other MS’s will send messages to the main server and main server will send to the **stub**.  
In case of Multi-cast, each server will send the messages to the stub.  
Unicast and Multi-cast are the **Messaging modes**.

--Plug-in is the one which manages load in load balancer.

--In real time, mostly we have only one application in each server.

--XA  
Global transaction, 2-phase commit  
Non-XA  
Local, 1-phase commit  
below is the place where we will mention Two-way SSL  
Server->protocol->Channel->Next->Next->Two-way SSL

--To **renew a SSL certificate**, just copy the certificate to trust store.  
You can copy the new certificate before the expiration of existing certificate also. When the existing certificate expires then the new certificate will automatically be used.

--In console we have **start up and shutdown classes.** The application team will give us the jar file which contains the classes to be loaded either at startup or at shutdown. Venu said that he has faced some memory issues because of these classes (which were useful for some application) and he has removed those classes from console.

--We might have **temp, cache and stage folders**. Mostly we delete temp and cache folders and sometimes MS’s data folder. We will not delete AS’s data folder because it contains usernames and passwords.

**Differences between 11g and 12c:**--12c only supports JDK 1.7 (recommended). You can also use the old versions like 1.6 and all but not recommended.  
-- Recommended for 11g is 1.6.  
--Up to WebLogic 10g, we have 2 types of data sources and they are 1) Generic or simple data source and 2) Multi data source. From 11g onwards we have one more data source i.e. Grid link data source.|  
-- Up to WebLogic 10g, by default we get a DB called PointBase DB. In 11g we have a DB called Derby. In 12c TopLink (maybe) is the DB. All are file based (maybe).  
--In WebLogic 11g, the Derby server will not be started automatically. You may see some errors related to Derby when you start the WebLogic server. You need to start Derby using startDerby.cmd (username and pwd for Derby is test only). In 12c, Derby will be started automatically.  
--In 11g, **bsu** is the default one and in 12c **opatch** is the default one to install the patches. You can use opatch in 11g and bsu in 12c but you need to download them explicitly.  
-- While installing WebLogic 12c you will see only one version of java and that is Sun JDK. In earlier versions you will see 2 options and they are **Sun JDK** and **JRockIT**.  
-- In 12c, we have a cluster concept for JMS servers also.  
--In 12c, we have **reconfig.cmd** to migrate a domain or NM from previous versions to 12c. There will not be any folder created in 12c but the domain or NM which is in 11g will be migrated to 12c.  
--In the same way we have **upgrade.cmd** in 11g which is used to migrate the domains in previous versions to 11g.

**Differences between 8 and 9:**  
-- Up to WebLogic 8, there were separate links for creating data source and connection pool in the console and also the data related to both of these used to be saved in config.xml along with all the other configuration information. But from WebLogic 9 onwards, only the data source location details will be maintained in config.xml along with other configuration information. Other details of data sources will be mentioned in other files.  
--Up to WebLogic 8, MS always depends on Admin server but from WebLogic 9 onwards a new concept called MS Independence (MSI) mode is introduced. It is not mandatory that Admin server should be up and running to start MS in WebLogic 9. But you should have started MS at least once after creating it and this time Admin server should be in running state. Even though you start the MS in independence mode, you can’t deploy any new applications on MS because the Admin server is down. But you can test existing deployed applications.