

# DevOps Tools

Getting Hands Dirty With DevOps Tools



[Author](#)

Gouse Shaik

# Contents

Why DevOps.....	14
What is DevOps .....	14
Before learning DevOps .....	14
DevOps Lean steps to follow.....	14
Quote:.....	14
DevOps Basic Pipeline .....	14
Proposed <i>PLAN</i> of CICD.....	15
Uname .....	16
Firewall settings.....	17
Error: No suitable device found. no device found for connection.....	17
BIOS info.....	17
Virtualization enabling from BIOS in windows 2008 server r2.....	18
BIOS changes .....	18
Virtualbox .....	18
How To Change Default Location Of Virtual Box Image Files .....	18
Vagrant Storage Path VAGRANT_HOME.....	19
Vagrant up is not getting up and running due to VT-x error.....	20
Vagrant .....	21
@\$ BUILDING A VAGRANT BOX FROM START TO FINISH .....	22
Why Build A Box?.....	22
Definitions .....	22
Shout Out .....	22
Getting Prepared .....	23
Build a Box.....	23
Configure the Virtual Hardware.....	23
Install the Operating System .....	23
Set Root Password.....	24
Setup the Super User .....	24
Creating sudo user and provide the sudo priveliges.....	24
Updating The Operating System.....	25
SSH keys Install the as Vagrant Key .....	25
Install and Configure OpenSSH Server.....	25
If you are asking for password while executing SSH to server.....	26
Installing Guest Tools .....	26
Installing the Guest Additions on a GUI-less server.....	26
Package the Box .....	26
Vagrant package Clean.....	26
Command to package a box from vagrant .....	26
Test Your Box .....	27
Packer : is a immutable means which is not manual.....	27
What Have We Learned .....	27
Automated connections configurations.....	28
To download files from server[Remote] to windows[local machine].....	30
script to check files available or not:.....	30
Vagrant user login Issue.....	31
Synced folders within your Vagrantfile.....	31
Issue while booting Vagrant box .....	32
Vagrant box cleanup.....	32
Convert a Virtualbox.ova VM into a vagrant box .....	32
config.vm.boot_timeout .....	32
Upload your custom box to Vagrant Cloud.....	33
Command to randomise MAC address in Virtualbox.....	34
Oracle vagrant linux boxes.....	34

Introduction.....	34
Prerequisites .....	35
Available Boxes .....	35
Unless otherwise stated, these boxes have the following specifications.....	35
Adding, downloading and starting the Oracle Linux Vagrant Box .....	35
For example.....	35
Vagrant boxes migrated to Local machine to RDP windows .....	35
Error at start .....	36
To verify the checksum of the downloaded box:.....	37
Vagrantfile syntax linxs.....	37
Vagrantfile Cheat Sheets:.....	37
#Creating a VM .....	37
#Starting a VM.....	37
#Getting into a VM.....	38
#Stopping a VM.....	38
#Cleaning Up a VM.....	38
#About Boxes.....	38
#Saving Progress.....	38
#Tips.....	38
#Plugins.....	38
#Notes.....	38
# Get CLI help with.....	39
# List boxes and versions .....	39
# Setup a new instance .....	39
# Login.....	39
# Check global status .....	39
#Repackage a local box.....	39
#To check the ports are opened or not for vagrant boxes .....	39
# Vagrant memory.....	39
# To set virtual machine memory size use:.....	39
# To set virtual machine CPU's count use:.....	39
# or to set half of available CPU's count.....	39
# Vagrant up getting SSH auth method   authentication failure ERROR.....	40
Vagrant login: <a href="https://app.vagrantup.com/">https://app.vagrantup.com/</a> .....	41
What is the difference between Packer and Terraform.....	42
Creating Boxes with Packer : is for images.....	42
»Creating Boxes via the Vagrant Cloud Web Interface .....	42
»Creating Boxes with the API .....	43
Unix Commands .....	43
Command check the memory usage .....	43
How to display information about the CPU on Linux.....	44
Docker [For MicroServices].....	44
To install Docker, we need to follow the steps given below.....	44
Step 1 : kernal check.....	44
Step 2 : yum update.....	45
Step 3 : install certificates for docker to install packages.....	45
Step 4 .....	45
Docker and oracle : The Policy[Certified] .....	45
To install docker on ubuntu: .....	46
OS requirements for docker on ubuntu .....	46
Docker info :WARNING: No swap limit support .....	46
Steps to Intall Docker on Oracle Linux.....	47
Oracle /siebel Docker Images list.....	48
Dockerhub official images.....	48
docker siebel images: .....	48
docker oracle images: .....	48

linux core concepts for Docker.....	49
Docker Commands.....	49
Dockerfile .....	53
Docker image creation steps for pingutils.....	54
DevOpsIssues .....	55
Docker Compose.....	55
Docker compose port mapping.....	55
Difference between Virtualization and containerization architecture .....	56
Docker compose.....	56
Docker Network.....	56
!# Single-host netowking .....	57
!# Multi-host overlay.....	58
Docker swarm .....	59
To create a Docker Swarm Manager : failover and High Availability.....	59
# Docker swarm init .....	60
# docker swarm init --adverstise-addr 172.30.2.67 .....	60
# docker swarm leave .....	60
# docker info .....	61
# Docekr node ls.....	61
#docker inspect node1 .....	61
# To retrieve the join command .....	61
# docker service.....	61
#docker service inspect qt-svc.....	63
RealTime Scenario .....	64
#docker service.....	65
# Practiced Our Swarm Cluster .....	65
Adding Worker Nodes to our Swarm.....	66
Create a Service .....	66
Accessing the Service .....	67
Scaling up and Scaling down.....	67
Inspecting nodes .....	68
Draining a node .....	68
Remove the Service.....	69
Applying Rolling Updates .....	70
Docker swarm join issues: .....	70
Docker swarm   Open protocols and ports between the hosts.....	70
#Github links .....	70
multi-machine scenario .....	70
Docker Process to build push to docker hub.....	71
# docker-machine and docker datacenters are all related to Enterprise products.....	71
Docker stack .....	71
Setting Limits on Ubuntu 16.04 .....	72
Running Without Limits .....	72
Limiting Memory.....	72
Limiting CPU.....	72
Limit Number of Cores .....	72
Lock Container to Specific Cores .....	72
Limit CPU Time.....	72
Shares and Weights.....	72
How to Increase the CPU Share and Memory of a Container .....	73
GIT.....	74
Introduction to GIT on Linux – Install, Create Project, Commit Files.....	74
1. Download and Install GIT .....	74
2. Initial Configuration .....	75
3. Create a Project .....	76

4. Add and Commit files to the Project.....	76
5. Make Changes and Commit the File.....	77
6. View Status and Commit Logs .....	77
JIRA [http://vagranchostip:8080].....	78
Check prerequisites as GIT, java, sudo user settings .....	78
Installation steps for JIRA.....	78
BitBucket.....	80
Check prerequisites as GIT, java, sudo user settings .....	80
Set the display settings for GUI installation.....	81
BAMBOO .....	86
Bamboo Best Practice – System Requirements.....	86
Bitbucket to Sourcetree configuration.....	88
Atlassian CLI .....	89
How to connect Bamboo and Bitbucket Server .....	89
Installation and Configuration.....	89
Task was canceled.....	90
Get some data into your tools.....	91
Bamboo Specifying Global or Build-specific Variables .....	98
Changing Your Shell with chsh .....	100
Confluence .....	100
JAVA .....	100
To install jre package.....	100
To check for updates to installed Java packages.....	100
To install jre updates.....	100
JAVA Basics.....	100
Prerequisites for Java Programming language.....	100
Principles of JAVA: .....	102
Runtime Architecture of JAVA.....	102
How to import Projects in Spring tool .....	102
Grails [Groovy and gsp].....	102
Move on to Groovy.....	104
What is Groovy.....	104
Why Groovy .....	104
Features.....	104
Groovy VS JAVA .....	105
Direct Mode VS precompiled Mode .....	106
Getting Started with Library Management system Application using GGTS .....	106
UNIX administration Topics.....	106
How to determine if AIX uses a 32-bit kernel or 64-bit kernel.....	106
Get complete CPU architecture .....	107
Unix Login welcome message text.....	107
For Getting xclock .....	107
Oracle sqlplus tips and tricks.....	107
LineSize & PageSize .....	107
Toad suite .....	108
Oracle database installation on Oracle Linux.....	108
Host files on unix Env.....	108
Oracle Installation Prerequisites.....	108
Oracle DB post checks.....	113
Oracle DB Details:.....	114
Oracle DB Views & Important SQL's .....	114
Oracle DB Architecture .....	114
Difference between cdb and pdb in oracle 12c .....	114

About OLTP and OLAP .....	115
Oracle 11G [g-grid] , oracle Grid Computing .....	115
RAC (Real application Cluster) and ASM(Automatic storage Management) .....	116
Oracle 12C [Cloud computing].....	116
Oracle DBA Topics: .....	117
Where is DATA PROCESSED?.....	117
UNDO and REDO .....	118
Oracle instance v/s Database.....	119
Oracle Process Architecture:.....	120
Oracle Background processes:.....	120
LOG writer:.....	121
Check Point Processes:.....	121
System Monitor Process (SMON) .....	123
Process Monitor .....	123
Recoverer Process .....	124
Archiver(ARC) [LOG_ARCHIVE_MAX_PROCESSES] .....	124
1. ....	Database Memory Structure.124
Database Buffer cache [DB_BLOCK_SIZE & DB_BLOCK_BUFFERS].....	125
SHARED POOL [is a memory] [SHARED_POOL_SIZE] .....	125
Redo Log Buffer [LOG_BUFFER] .....	126
Large pool [LARGE_POOL_SIZE] .....	127
Java Pool & Streams Pool [STREAMS_POOL_SIZE] .....	127
Keep Buffer Pool [DB_KEEP_CACHE_SIZE].....	128
Recycle Buffer Pool [DB_RECYCLE_CACHE_SIZE] .....	128
nk Buffer Cache [DB_NK_CACHE_SIZE] .....	129
Program global Area .....	129
Dedicated Vs Shared Servers.....	130
Program global Area in Shared server Environment .....	131
Database Transaction Example .....	132
Oracle Database Startup.....	132
Database shutdown .....	133
Parameter File [Show parameter spfile].....	134
How to rename a datafile in the primary database and manually.....	136
How To Check Oracle Database Memory and Storage Size .....	136
--- Get the SGA footprint of a database instance:.....	136
--- Get the current PGA consumption of a database instance: .....	137
--- Get the estimated memory requirement for the unconnected sessions:.....	137
Get the maximum number of processes that an Oracle instance can handle. ....	137
Let's count the number of connected sessions:.....	137
ESTIMATED TOTAL MEMORY REQUIREMENT.....	137
2. Get the estimated storage sizing requirements of a database.....	137
-- Copy and paste each resulting commands onto a shell script and execute in primary server to get the mount point size .....	137
-- Query all parameter file destination and determine if they are of different path from the datafile mount point or ORACLE_BASE dir:138	138
-- Query the database directories for mount points .....	138
Oracle Tablespace Management.....	138
Table space [Data File= Logically and physical data] .....	138
TableSpaces Types.....	139
Default Tablespaces comes from installations:.....	139
Online and Offline TableSpace .....	139
How to Create TableSpace.....	139
How to add more space to Database .....	140

! Practiced on oraclelinux7 VM [Tablespace] .....	140
To check the space on DB [DBA_TABLESPACE_USAGE_METRICS] .....	141
[BestPractice]TableSpace Management Demo .....	142
For deleting tablespace from database .....	143
To add more datafiles for an existing tablespace ex:tbs1 .....	143
To Rename the existing Tablespace name with new name .....	143
To drop the tablespace for a particular data file .....	143
To rename the datafile , you have to do first offline the tablespace .....	144
Tablespace with different block sizes. [Used for full table scans] .....	144
Temporary Tablespace [DBA_TEMP_FILES] .....	145
To check the database default properties .....	145
To create a temporary tablespace group[DBA_TABLESPACE_GROUPS] .....	145
Locally VS Dictionary managed tablespaces .....	146
If the data is not showing properly in sql cmd .....	147
<b>UNDO Management</b> .....	148
Read consistency [oracle row will record in rollback segment] .....	148
ORA-01555 SNAPSHOT TOO OLD ERROR .....	149
Automatic UNDO Management [Preferred than manual] undotbs1 .....	149
# ! Practiced on UNDO Tablespace .....	150
Configuring Undo Retention Period [UNDO_RETENTION=1800] .....	150
<b>REDO Management</b> .....	151
# ! Practiced on REDO Tablespace .....	152
For clearing a Redo log .....	154
Redo Log Data Dictionary Views .....	154
Archive Mode / No Archive Mode .....	154
Let see if the database is archive mode or not .....	155
<b>USER Management</b> .....	155
Schema called all logical data will be stored in schema which will have an account .....	156
Default ADMINISTRATIVE ACCOUNTS .....	156
Create users .....	156
<b>QUOTA</b> .....	156
Users privileges .....	156
System privileges .....	156
Oracle PASSWORD Expiry for user .....	157
Object privilege .....	157
REVOKE SYSTEM PRIVILEGES WITH ADMIN OPTION .....	158
# ! Practiced on privileges .....	158
ROLE [roles will be used instead of set of roles and assign to user] .....	159
<b>PROFILE</b> .....	160
Profile   password Management .....	160
To know the version of oracle db .....	161
<b>ORACLE NET SERVICES</b> .....	161
Listener.ora [DB server] .....	161
Listener Control Utility .....	162
Client listner connection string .....	162
Tnsnames.ora [client machine] .....	163
Sqlnet.ora [Client Machine] .....	163
<b>DataBase Link</b> .....	163
Public database link .....	164
Private database link .....	164
# !Practiced DB Links .....	164
<b>DATA Dictionary</b> [DBA Views for tables] .....	164

□ .....	Static views	164
Diagnostic DATA.....		166
Automatic Diagnostic Repository(ADR).....		166
# !Practiced.....		168
Automatic Diagnostic Repository command interpreter(ADRCI).....		169
# !Practiced.....		170
Table Partitioning .....		171
Range partitioning .....		171
DB Shema migration Activity.....		172
Grarfana.....		172
Monitor and visualizing Data with grafana.....		172
Use Cases of Grafana: .....		173
Start the server (init.d service) .....		173
Start the server (via systemd) .....		173
Grafana Configuration Steps: .....		173
How Grafana Works. .....		173
* What is Graphite: .....		175
Why Graphite is a preferred data source? .....		175
What is STATSD? .....		175
Components of Graphite .....		176
STatsD .....		176
StatsD Diagram: .....		176
StatsD and Graphite.....		177
Grafana installation: .....		178
Install Mysql for grafana.....		178
Grafana implmentation Example Architechture.....		179
To Xmanager encrypt password .....		179
Xshell Xftp password decrypt .....		179
Oracle Linux CheatSheet.....		179
File Commands .....		179
Process Management .....		180
File Permissions .....		180
SSH .....		180
Searching.....		180
System Info .....		180
Compression .....		180
Network .....		180
Installation .....		180
Shortcuts .....		181
Linux – Run command as a background process.....		181
GZip decompress files .....		181
How to find Operfing System is 32 bit or 64 bit?.....		181
How to Find the Version of the Oracle Solaris OS That Your System Is Running .....		181
How to find the number of cpu ?.....		182
To find the number of cores per CPU: .....		182
To check the RAM size in linux. .....		183
To find only directories in Linux/Sunsolaris? .....		183
To find operating system version?.....		183
To find the trc files which are older than 30 days? .....		183
To delete the trace files older than 30 days.?.....		183
To find the req/out/log files which are more than 10Mb. .....		184
If you ever want to confirm the block size of any filesystem of any other Linux OS.....		184
To change the the timestamp (date and time of modification) of a file or directory. ....		184

User Administration.....	184
To view how long the system is running.....	185
To find out hardware details .....	185
you can also check whether job is running or not.....	186
To stop remove job use kill cmd.....	186
To check pmon is running or not.....	186
To check fnd and http processes are running or not .....	186
To check history. ....	186
You can also use fc -l cmd to check history in linux or Sunsolaris.....	186
How to check if a Port is listening for any Service?Checking webserver port no in the below example.....	186
Also you can check with lsof (list of open files).....	186
How can I find out if my processor is 64 bit or 32, under linux? .....	187
How to check all mount pints of a server ?.....	187
How to make tar? .....	187
How to untar? .....	187
Yum update issue.....	187
No Space Left.....	188
Linux Commands Cheat Sheet.....	188
1 – SYSTEM INFORMATION .....	188
2 – HARDWARE INFORMATION .....	189
3 – PERFORMANCE MONITORING AND STATISTICS .....	189
4 – USER INFORMATION AND MANAGEMENT.....	190
5 – FILE AND DIRECTORY COMMANDS.....	190
6 – PROCESS MANAGEMENT.....	191
7 – FILE PERMISSIONS.....	191
8 – NETWORKING .....	192
9 – ARCHIVES (TAR FILES) .....	192
10 – INSTALLING PACKAGES.....	192
11 – SEARCH.....	193
12 – SSH LOGINS.....	193
13 – FILE TRANSFERS .....	193
14 – DISK USAGE .....	193
15 – DIRECTORY NAVIGATION.....	193
Linux ip Command Networking Cheat Sheet .....	194
IP QUERIES .....	194
MULTICAST ADDRESSING .....	194
MODIFYING ADDRESS AND LINK PROPERTIES.....	194
ADJUSTING AND VIEWING ROUTES .....	195
ADJUSTING AND VIEWING ROUTES .....	195
MANAGING THE ARP TABLE .....	195
USEFUL NETWORKING COMMANDS (NOT NECESSARILY PROVIDED FROM IPROUTE) .....	195
COMPARING NET-TOOLS VS. IPROUTE PACKAGE COMMANDS .....	195
Linux System Administrator Projects.....	196
Scratch Your Own Itch .....	197
Configure the LAMP Stack .....	197
Deploy an Open Source Application .....	197
Configure Common Services – Client AND Server .....	197
Configure Monitoring .....	198
Create a Build System .....	198
Create a Centralized Syslog Server .....	198
System Automation.....	198
Cluster All The Things .....	198

Build a NAS .....	199
Practice Migrating Data .....	199
Create and Manage Users .....	199
Configure a Backup Server .....	199
Configure a Firewall .....	199
Learn LVM .....	199
Configure a Proxy Server .....	199
Learn Revision Control .....	199
Installing Portal BRM on RHE 5 .....	200
How to install and configure apache tomcat as service.	200
Adding the Apache Tomcat runtimes .....	202
Centralized logging system Architecture .....	202
Create filesystem /siebellogs .....	203
100G storage added by storage team .....	204
Scp commands .....	204
Create directories in sda022 .....	204
Install tomcat on /siebellogs filesystem .....	204
create a virtual directory in tomcat .....	204
Softlink or symbolic links creation:.....	204
Cgi web page to execute unix commands .....	204
CGI enabling to tomcat:.....	204
Tomcat TTILE change in Windows .....	205
Install Tomcat as a Windows Service .....	206
Other way of creating tomcat services for liferay project .....	208
Services restart script .....	209
Tafeely SIT services restart script and scheduled Task .....	209
Tomcat Security .....	209
Notice: This section is out of date with respect to recent version of Tomcat .....	209
Run as Unprivileged User .....	210
Restrict access to critical files .....	210
Use a Firewall .....	210
Remove Default Tomcat Applications .....	210
Store Passwords as Digest .....	210
Separate User Roles For Restricted Datasets .....	211
Restrict Access to Tomcat Manager Applications .....	211
Java Security Manager .....	212
Setting default timezone for the JVM of Tomcat .....	212
Cron Job .....	212
Syntax of crontab (field description) .....	212
Job Failing in Cronjob's .....	212
Python:.....	214
How to install Python on windows .....	214
PIP installation on windows .....	214
Python scripting on unix .....	214
Python Webserver .....	214
How to embed any script in HTML web page .....	215
Pip installation and virtual environment setup steps .....	215
Python pip install & Download .....	217
How to Install a Package in Python using PIP .....	217
Steps to Install a Package in Python using PIP .....	218
Uninstall a Package using PIP .....	220
Tool to install/uninstall a Python Package .....	221
Python Automa's Functions & Classes .....	223

Apache Airflow.....	223
	
Getting started with Apache Airflow.....	224
What is Airflow?.....	224
What is Dag?.....	224
Why we use Airflow?.....	225
Installation and Setup.....	225
Apache Airflow Installation on Ubuntu .....	231
# ! How to run bash script file in Airflow.....	236
Learn Airflow basics .....	237
<i>DAG</i> .....	238
Python   web2py .....	238
Classes why we need them ?.....	239
Define a class in python is so easy.....	240
Instance will be created for the class.....	240
Adding-Methods-to-Our-Class .....	241
Constructors and other special methods.....	242
Solaris - checking whether an account is locked or not.....	242
OpenSSH.....	243
How to Install and Configure OpenSSH Server In Linux .....	243
What is OpenSSH? .....	243
Why Use OpenSSH And Over Telnet Or Ftp? .....	244
What Are Some Of The OpenSSH Features?.....	244
Installation of OpenSSH in Linux.....	244
Configuration of OpenSSH .....	244
How Do I Connect to OpenSSH.....	245
Conclusion .....	246
Playing with firewall.....	246
How to Disable the Firewall for Oracle Linux or Red Hat Enterprise Linux.....	246
VirtualBox. How to set up networking so both host and guest can access internet and talk to each other.....	246
ZABBIX.....	248
ssh-passwordless-login-using-ssh-keygen-in-5-easy-steps.....	248
Solaris Admin provided steps for ssh-keygen.....	249
How to configure Password-less authentication from Windows to Linux using Putty .....	250
Jenkins.....	255
Jenkins installation on windows.....	255
Jenkins now working unfortunately .....	255
Make sure to take the backup of existing jenkins installed directory every time you setup newly.....	255
Error 1067. The process terminated unexpectedly.....	255
How to install a plugin in Jenkins manually .....	256
BUILDING PIPELINES BY LINKING JENKINS JOBS.....	256
On keeping things small.....	256
A true story : Restarting a test server .....	257
One-to-One Relationship .....	257
Über Job. the wrong good idea .....	258
A bit of optimization. Fork and Join .....	259
A last tip .....	259
Conclusion .....	259
Usefull link: .....	259
Prerequisites and steps to setup Jenkins .....	259
To change the port: .....	260
Jenkins. How to change JENKINS_HOME on Windows.....	260
Build Jenkins Job with Condition and Parameter .....	261
Jenkins Master and slave node configuration.....	264

Jenkins blocked by Internet explorer.....	267
Jenkins-linking-to-my-local-git-repository.....	267
Access local git repository in Jenkins docker container.....	267
Apache Maven.....	268
Jenkins FIX Performance Issues.....	268
Solution.....	268
Tip 1: Minimize the amount of builds on the master node .....	268
Tip 2: Do not keep too much build history .....	269
Tip 3: Clear old Jenkins data .....	269
Tip 4: Define the right heap size.....	270
Tip 5: Tune the garbage collector .....	270
Now it's your turn.....	270
Reload Configuration from Disk.....	270
Maven coordinates.....	271
Mvn install.....	271
Assignment.....	272
Apache Archiva.....	272
How to install Apache Archiva in Windows.....	272
Apache Archiva installation.....	272
1. Download Apache Archiva.....	272
2. Extract it.....	272
3. Install as Windows service .....	273
4. Start Archive .....	273
4. Archive web admin .....	273
5. Done .....	273
Post Installation.....	273
1. Archiva configuration file .....	273
2. Update port number .....	273
3. Restart Apache Archiva .....	273
Siebel Installation with OEL7 and Docker.....	273
Quick links.....	273
00:30 – OEL Install .....	273
22:25 – Setting up Virtual Box extensions .....	274
27:00 – Setting up Docker .....	274
34:04 – Setting up Git.....	274
35:45 – Building 32-bit client.....	274
42:16 – Building IP17 Base.....	274
Setting up Siebel Provisioning Server.....	275
OpenShift.....	278
01. OpenStack Introduction.....	278
Deploy and manager OpenStack with below topics.....	278
02. OpenStack Architecture.....	278
03. OpenStack Messaging Server.....	278
04. Identity Service – Keystone .....	278
05. OpenStack Clients .....	278
06. Image Service – Glance .....	278
07. Compute Service – Nova .....	279
08. Networking Service – Neutron .....	279
09. OpenStack Dashboard – Horizon .....	279
10. Block Storage – Cinder .....	279
11. Object Storage – Swift .....	279
12. Orchestration Service – Heat .....	279

13. OpenStack Telemetry – Ceilometer .....	279
14. Launching Instances.....	279
Ansible.....	279
Oracle Linux – Install Ansible.....	279
Steps 2 .....	280
Step 3 .....	280
MongoDB .....	281
Prerequisites .....	281
Database .....	281
Collection [Tables].....	281
Document [Schema/row] .....	281
Sample Document.....	281
About MongoDB Structure .....	283
MongoDB installation.....	283
To setup mongoDB server: .....	284
MongoDB Commands Cheatsheet.....	284
Access the MongoDB.....	284
Show all databases: .....	284
Create a database .....	284
Add a collection: .....	284
Show all collections .....	285
Insert a record in the collection .....	285
Display list of records of a collection .....	285
Display a list of records matching with value (s) of specific fields.....	285
Drop the collection: .....	285
Create users in the database.....	285
Show users .....	285
Login into the database with username and password.....	285
MongoDB Server Service is not starting .....	285
MongoDBCompass :.....	286
GlassFish .....	286
Install the Oracle GlassFish Server on Windows .....	286
ElasticSearch.....	291
How to Install Java 11 on CentOS 7/6 & Fedora 30-25 .....	291
Step 1 – Download Java 11 .....	291
Step 2 – Install Java 11 .....	292
Step 3 – Check Java Version .....	292
Step 4 – Setup Java Environment Variables .....	293
How to Install Elasticsearch on Fedora & CentOS .....	293
Step 1 – Prerequisites .....	294
Step 2 – Download Elasticsearch.....	294
Step 3 – Configure Elasticsearch .....	294
Step 4 – Start Elasticsearch and Test .....	295
Step 5 – Elasticsearch Uses Examples .....	296
Create New Bucket .....	296
Adding Data to Elasticsearch .....	296
Fetching Data from Elasticsearch .....	298
Searching in Elasticsearch .....	298
How to Install and Configure Elasticsearch on Linux and Windows .....	300
Introduction to Elasticsearch .....	300
Download Elasticsearch.....	301
Modify elasticsearch.yml file .....	302
Modify Elasticsearch jvm.options .....	303

Start Elasticsearch.....	304
Verify Elasticsearch Installation.....	305
Installing On Debian / Ubuntu.....	306
Groovy Installation.....	307
Gradle.....	308
Apache Ant & Apache IVY .....	311
ELK Installation Guide.....	313
Summary .....	314
<b>Beat Overview</b>	
<b>Metricbeat.....</b>	<b>314</b>
<b>Filebeat .....</b>	<b>314</b>
<b>Installation &amp; Configuration Steps .....</b>	<b>315</b>
<b>Metric beat.....</b>	<b>315</b>
<b>Installation Steps.....</b>	<b>315</b>
<b>Configuration of Metric beat .....</b>	<b>316</b>
<b>File beat.....</b>	<b>317</b>
<b>Installation Steps.....</b>	<b>317</b>
<b>Configuration of File beat .....</b>	<b>319</b>
<b>Make Build configuration.....</b>	<b>319</b>
<b>Nexus.....</b>	<b>320</b>
<b>Windows Commands.....</b>	<b>320</b>

### Why DevOps

Gives a decent package and future technology which can give hope for next 5 yrs.

- About ticket booking sites.
- Cab booking sites
- Dev code need to take as much as fast to market.
- This will comes under software development life cycle.
- When we start Dev ops, it should be parallel to development.
- We want increase releases to deliver as soon as possible.

### What is DevOps

- Dev-ops is not about tool, is all about how the code automation faster to the market.

### Before learning DevOps

- ✓ Understand the delivery pipeline like which carries water.
- ✓ Called as DevOps Developers.
- ✓ Good at Linux, shell scripting.
- ✓ Should be good at PowerShell
- ✓ DevOps is a serial not like a movie.
- ✓ Python is a secondary skill to take your shell script to next level.
- ✓ Average salary for DevOps for 10yrs =12 lacs
- ✓ Giving unnecessary features to customer, to reduce the surprises.
- ✓ Create a workflow which should happen quickly.

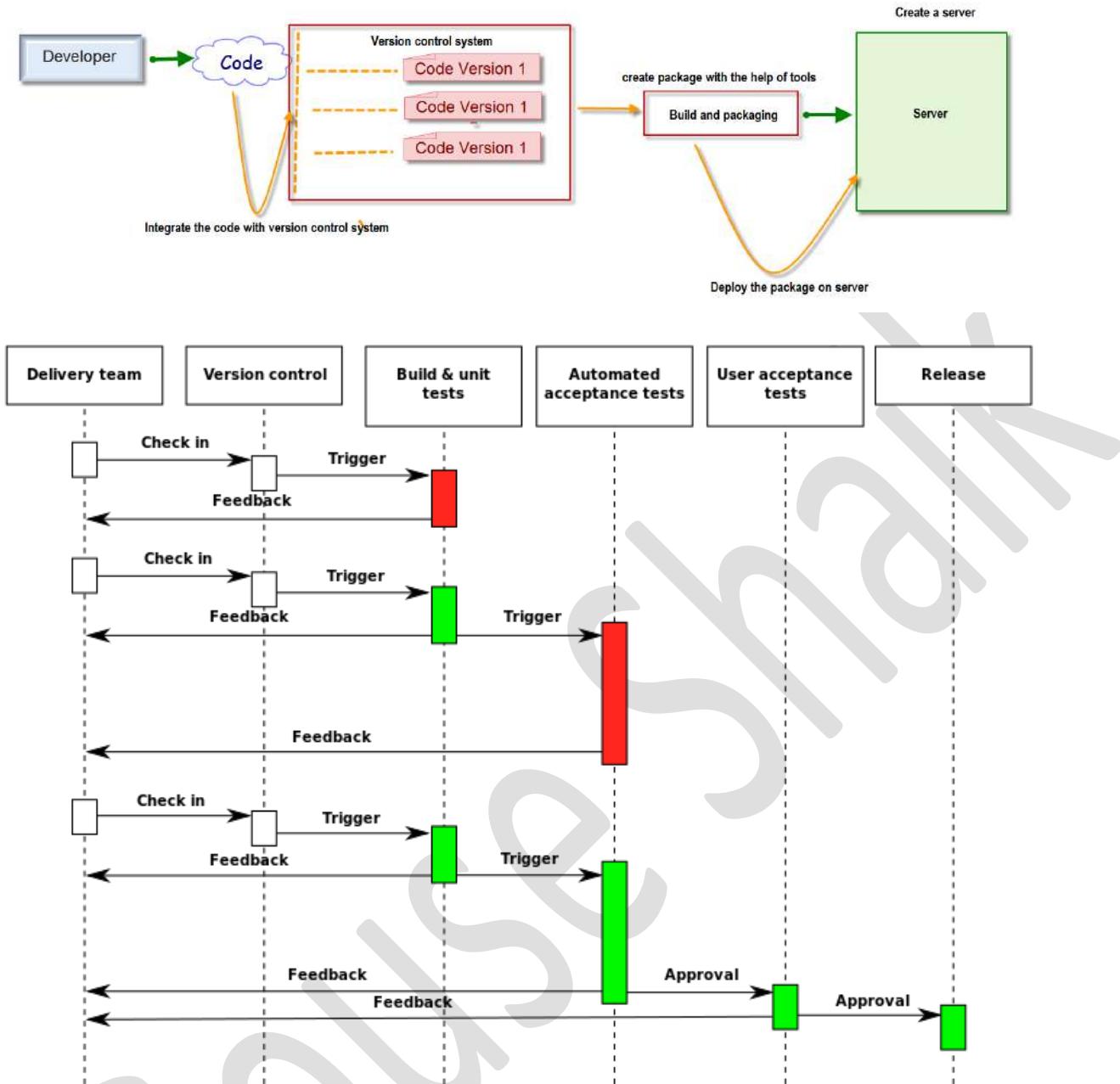
### DevOps Lean steps to follow:

- ✓ Focus on customer value
- ✓ Eliminate Waste
- ✓ Reduce cycle time
- ✓ Shared learning
- ✓ Avoid batching
- ✓ Theory of constraints
- ✓ Add value which will make the pipeline workflow works in flow.

Quote:

► IT is the factory floor of this century.

### DevOps Basic Pipeline



## Proposed *PLAN* of CICD

Below are major points should be considered with the respect to the proposed *PLAN* of CICD implementation:

- E2E Automation Process Flow for Future Enhancement “low Level Architecture Diagram”.
- E2E Automation Process Steps for each service.
- Build, Deployment and testing Automation Process Flow for the current approach.
- Docker Reference Architecture: Design Considerations and Best Practices to modernize the current traditional environment architecture.
- Configuration Management System

### 1.1 Application/Services Structure

### 1.2 Application/ Services Behavior

#### 1.2.1 Main Objects Life Cycles

##### 1.2.1.1 Release Life Cycle

##### 1.2.1.2 Build Life Cycle

- 1.2.1.3 Deployment Life Cycle
- 1.2.1.4 Code Review Life Cycle
- 1.2.1.5 Components Hierarchy
- 1.2.1.6 Branching & Merging Strategy
- 1.2.1.7 Branching Naming Convention standards
- 1.2.1.8 Cls (Services, Databases) Naming Convention standards
- 1.2.1.9 Cls Services & Databases tracking versions
- 1.2.1.10 Release & branching Scenarios and Proposed Solution for each Scenarios
- 1.2.1.11 Role of PM, Designer, Developer, QA, SCM, and Ops team.

**Also we must consider the below prerequisites in order to ensure that the entire scope will fit during the current implementation:**

**Source Code:**

1. Baseline the code which will be mainline benchmark for any future development and same will be promoted to different required environment.
2. How are you going to manage the package artifacts using Standard Repository?
3. What about Junit Plugins and monitoring and alerting engine?
4. What about the integration between Bamboo, JIRA, TestLink and so on.
5. Tag/Label Source code consistency challenges with respect to the current baseline of each component?
6. What about Standard Naming Convention for all the Tag creation activity?

**Branching:**

1. Define & implement standard method of branching strategy to fit the current delivery model
  2. Define the process to accommodate the Config , db and supporting files by versioning.
    - a. Code
    - b. configuration
      - i. SQL scripts
      - ii. MQ scripts
      - iii. Property files
      - iv. XSLT
    - c. Documentation (service design, interface ....etc)
- 2.1 Standardize service structure.
- 2.2 Adding Unit tests framework for the developed services
- 2.3 Automate the code review process.
- 2.4 Code peer review process captured and managed over integrated tool.
- 2.5 how to link code changes with corresponding configuration changes automatically during build and rollback
  - a. code tracking
  - b. configuration tracking
- 2.6 How to facilitate and automate unit tests with respect to Docker.
- 2.7 Code and configuration promotion from environment to next
- 2.8 A mechanism to monitor which code and configuration exist in which environment

## Uname

AIX - Getting OS Versions

How to check the OS version for AIX platform. There have two command usually used to check OS version. "`uname`" and "`oslevel`"

```

uname command with flag:
uname -a = "Displays all information specified with the -m, -n, -r, -s, and -v flags."
uname -p = "Displays the architecture of the system processor."
uname -r = "Displays the release number of the operating system."
uname -s = "Displays the system name. This flag is on by default."
uname -n = "Displays the name of the node. This may be a name the system is known by to a UUCP communications network."
uname -M = "Displays the system model name. If the model name attribute does not exist, a null string is displayed."
uname -v = "Displays the operating system version."
oslevel command with flag:
oslevel -l = "Lists filesets that are earlier (less) than the technology level or service pack specified by the Level parameter."
oslevel -g = "Lists filesets that are later (greater) than the technology level or service pack specified by the Level parameter."
oslevel -r = "Applies all flags to technology levels."
oslevel -q = "Lists names of known technology levels (when used with the -r flag) or service packs (when used with the -s flag) that can be specified using the -l or -g flag."
oslevel -s = "Applies all flags to service packs. The service pack level returned is in the format 6100-00-01-0748, where 6100 refers to base level 6.1.0.0; 00 refers to technology level 0; 01 refers to service pack 1; and 0748 refers to the yyww build date in year and weeks, in this case, the 48th week of the year 2007.

```

#### **Firewall settings:**

<https://oracle-base.com/articles/linux/linux-firewall>

#### **Quick Database Setup:**

If you are using the server as an Oracle database server, you will probably want to make sure the SSH and Oracle listener ports are accessible. You could lock these down to specific source IP addresses, but for a quick setup, you could just do the following, where "1521" is the port used for the listener.

```

# service iptables start
# chkconfig iptables on
# iptables -A INPUT -p tcp --dport 22 -j ACCEPT
# iptables -A INPUT -p tcp --dport 1521 -j ACCEPT
# service iptables save
# service iptables status

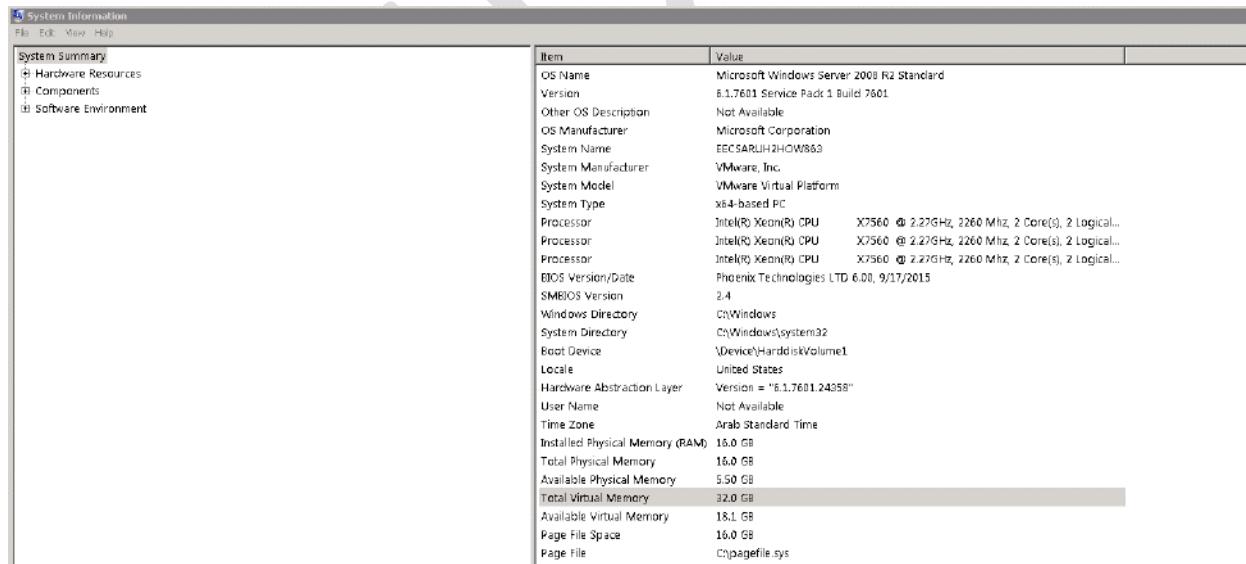
```

**Error: No suitable device found: no device found for connection**

<https://www.techlanda.com/2016/08/error-no-suitable-device-found.html>

## **BIOS info**

Command : open windows run → msinfo32



The screenshot shows the Windows System Information window with the following details:

Item	Value
OS Name	Microsoft Windows Server 2008 R2 Standard
Version	6.1.7601 Service Pack 1 Build 7601
Other OS Description	Not Available
OS Manufacturer	Microsoft Corporation
System Name	EECSARLJH2HOW863
System Manufacturer	VMware, Inc.
System Model	VMware Virtual Platform
System Type	x64-based PC
Processor	Intel(R) Xeon(R) CPU X7560 @ 2.27GHz, 2260 MHz, 2 Core(s), 2 Logical...
Processor	Intel(R) Xeon(R) CPU X7560 @ 2.27GHz, 2260 MHz, 2 Core(s), 2 Logical...
Processor	Intel(R) Xeon(R) CPU X7560 @ 2.27GHz, 2260 MHz, 2 Core(s), 2 Logical...
BIOS Version/Date	Phoenix Technologies LTD 6.00, 9/17/2015
SMBIOS Version	2.4
Windows Directory	C:\Windows
System Directory	C:\Windows\System32
Root Device	\Device\Harddisk\Volume1
Locale	United States
Hardware Abstraction Layer	Version = "6.1.7601.24358"
User Name	Not Available
Time Zone	Arab Standard Time
Installed Physical Memory (RAM)	16.0 GB
Total Physical Memory	16.0 GB
Available Physical Memory	5.50 GB
Total Virtual Memory	32.0 GB
Available Virtual Memory	18.1 GB
Page File Space	16.0 GB
Page File	C:\pagefile.sys

## Virtualization enabling from BIOS in windows 2008 server r2

### BIOS changes

You must enter the BIOS setup of the server and make sure that "Virtualization Technology" and "Execute Disable" are both set to Enabled. In most cases, the required BIOS settings can be found in these BIOS sections (actual names may differ, based upon your server's BIOS settings):

Here are some navigations of the BIOS settings on a Dell R610 server:

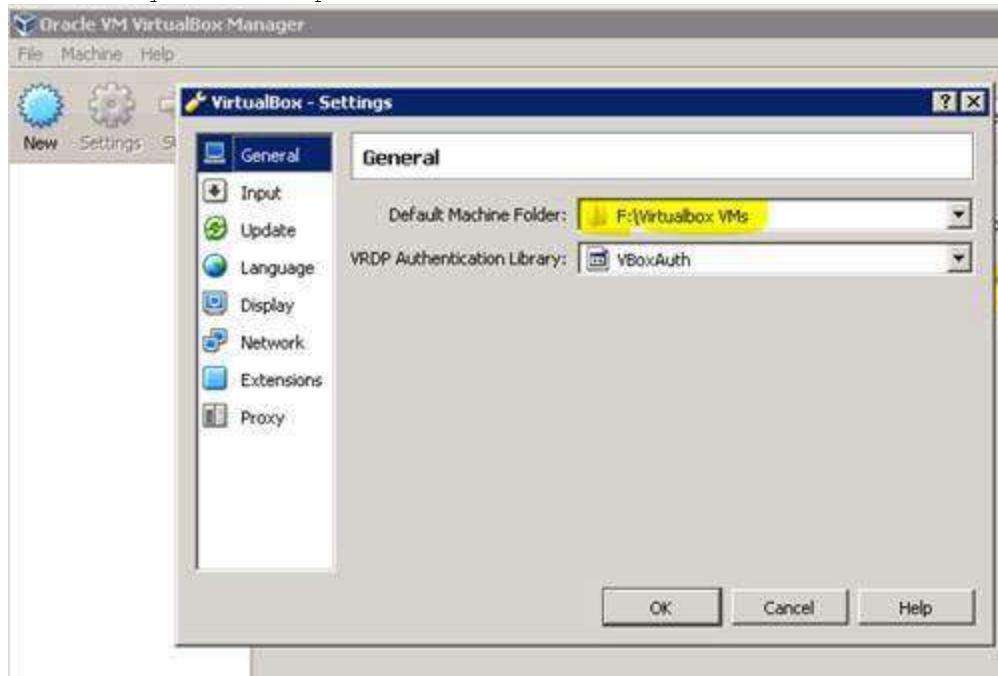
- Security > Execute Disable (set to On)
- Performance > Virtualization (set to On)
- Performance > VT for Direct I/O Access (set to On)
- Performance > Trusted Execution (set to Off)

### Virtualbox

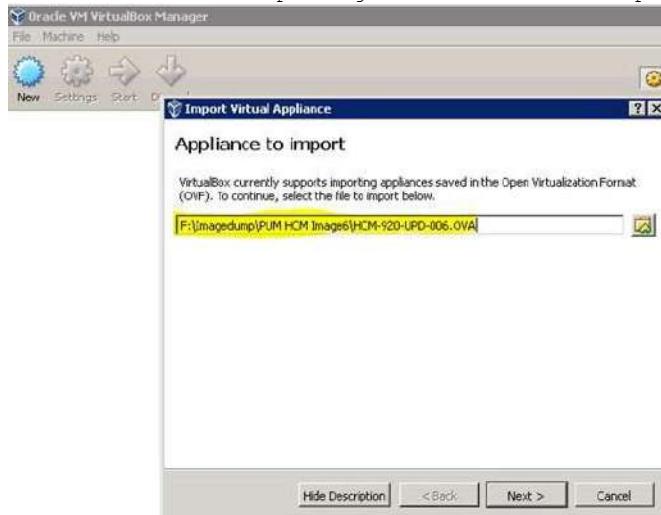
#### How To Change Default Location Of Virtual Box Image Files

By default Virtual Box files are stored in C: drive under user directory. You may want to change this default location for variety of reasons, one of the most common would be disk space constraints or security concerns. If you wish to change the default path to other drive/directory you can do so as shown below.

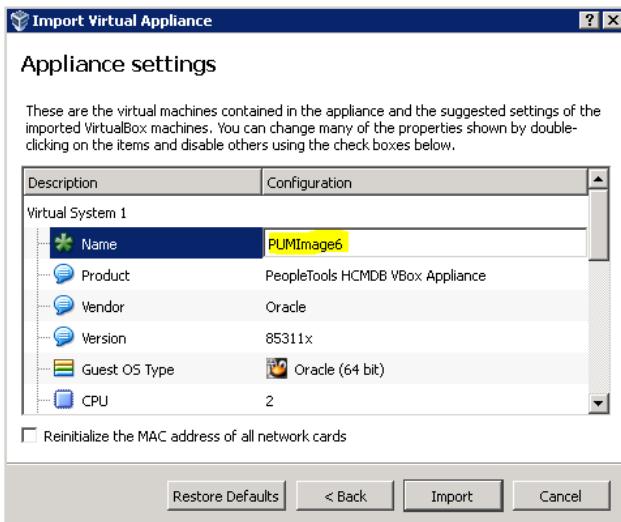
In Oracle VM Virtual Box Manager, Go to **File > Preferences**. On General Tab, Change the path for '**Default Machine Folder**' to your desired path.



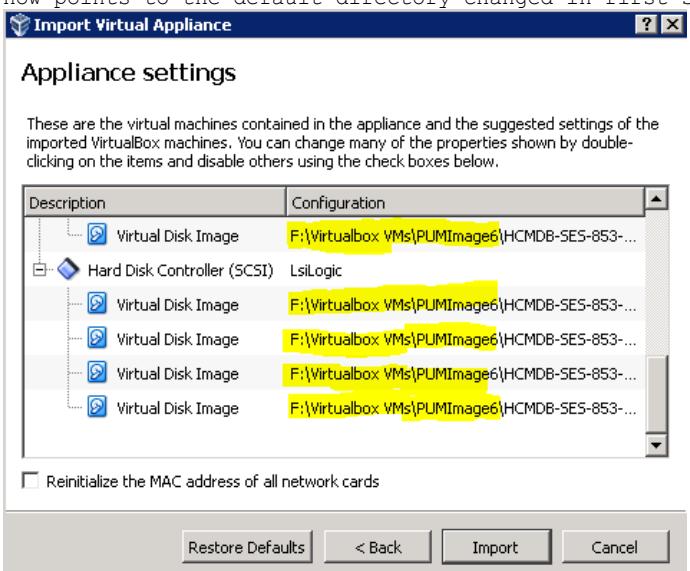
Go ahead and start importing OVA files. File > Import Appliance.



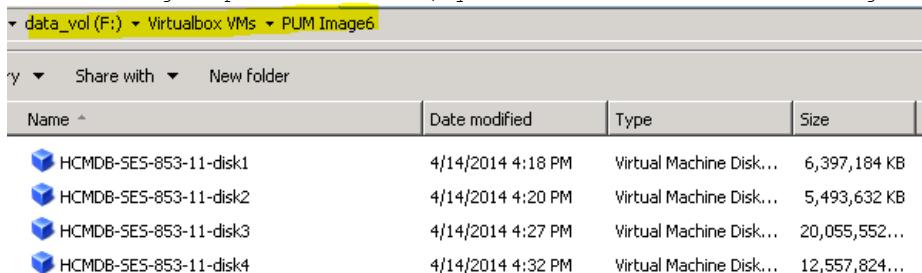
If you wish you can also change the default name of the image files. In below screenshot name of the image was change to PUMImage 6 ( PeopleSoft Image 6 ).



Once you change the name above, you can scroll below to see that directory structure format has also changed and now points to the default directory changed in first step and then the new folder name.



Once the image import is finished, you will see the Virtual Box image files in your target directory.



If you've any questions, please let us know in the comments.

## Vagrant Storage Path VAGRANT\_HOME

Change the VAGRANT\_HOME for setting the default storage path

For VirtualBox, you can change the location of what is known as the Default Machine Folder through the GUI's Preferences dialog box.

This guide, while a couple of years old, works fine and I followed it last week for how to move an existing vagrant/VirtualBox drive to a new location.

### EDIT

I have quoted the steps from the above link/guide, for posterity:

- Move `~/vagrant.d` to the external drive. I renamed it `vagrant_home` so I'd be able to see it without `ls -a`.
- Set `VAGRANT HOME` to `/path/to/drive/vagrant home` in `~/.bash profile`.
- Open the VirtualBox app, open Preferences, and set its Default Machine Folder to `/path/to/drive/VirtualBox VMs`.
- Close VirtualBox.
- Move your VirtualBox VMs folder to the drive. Reopen VirtualBox. You'll see your VMs are listed as "inaccessible". Remove them from the list.
- For each VM in your VirtualBox VMs folder on the external drive, browse to its folder in Finder and double-click the `.vbox` file to restore it to the VirtualBox Manager. (Is there an easier method than this?)
- Finally, move any existing Vagrant directories you've made with `vagrant init` (these are the directories with a `Vagrantfile` in each) to the external drive. Since these directories only store metadata you could leave them on your main drive, but it's nice to keep everything together so you could fairly easily plug the whole drive into another machine and start your VMs from there.

## Vagrant up is not getting up and running due to VT-x error

```
Stderr: VBoxManage.exe: error: The native API dll was not found
(C:\Windows\system32\WinHvPlatform.dll) (VERR_NEM_NOT_AVAILABLE).
VBoxManage.exe: error: VT-x is not available (VERR_VMX_NO_VMX)
VBoxManage.exe: error: Details: code E_FAIL (0x80004005), component Consolewrap, interface Iconsole
After installation rebot is required.
```

Solution: after installing old version of virtualbox, I didn't get the API dll was not found, but VT-X is not available

```
Stderr: VBoxManage.exe: error: VT-x is not available (VERR_VMX_NO_VMX)
VBoxManage.exe: error: Details: code E_FAIL (0x80004005), component Consolewrap, interface Iconsole
```

Waiting for solution: <https://forums.virtualbox.org/viewtopic.php?f=6&t=58820>

**Perryg wrote:** I believe this type of problem was added when they tightened/changed the way VT-x is used. You should see long mode off in the settings ( `VBoxManage showvminfo vmname` )  
You can test this by doing the following:

```
cd C:\Program Files\Oracle\VirtualBox
$ VBoxManage list vms
```

```
VBoxManage modifyvm <vmname> --longmode off
```

And see if it will boot. Remember the guest needs to be off for this to work.

```
VBoxManage modifyvm <vmname> --longmode off
```

Also worked here , my Box is Linux , and yes, I already know, this CPU doesn't have VT-x , so it is a 64-bits machine, but all vms just work in 32 bits . this issue happens on all vms: Linux , WindowsXP and Windows 7.

Thanks,

OR

Try to change setting `HardwareVirtEx` in your virtual machine config file (`<machine name>.vbox`).

It's a CPU section/ Looks like this

```
<Hardware version="2">
  <CPU count="1" hotplug="false">
    <HardwareVirtEx enabled="false"/>
    <HardwareVirtExNestedPaging enabled="true"/>
    <HardwareVirtExVPID enabled="true"/>
```

```
<HardwareVirtExUX enabled="true"/>
<PAE enabled="false"/>
<HardwareVirtExLargePages enabled="true"/>
<HardwareVirtForce enabled="false"/>
</CPU>
```

Thanks alsov.

Do I only need to set HardwareVirtEx to false or should I set all the VirtEx\* settings to false?

OB

Edit: It helps to experiment. 😊 I only needed to change the first one, HardwareVirtEx to false.

OB

If you are facing the problem in VirtualBox after the Windows 10 Founder's Update, you need to turn Core Isolation Memory Integrity back off again. The UI is bugged, here's how it can be done via registry, navigate to below and set 0:

HKEY\_LOCAL\_MACHINE\SYSTEM\CurrentControlSet\Control\DeviceGuard\Scenarios\HypervisorEnforcedCodeIntegrity\Enabled

This solves the problem if you have virtualization enabled on bios, and don't have hyper-v active. It can be edited in gui by: "Settings" → "Update & Security" → "Windows Security" → "Device Security" → "Core Isolation" → Turn it off

Error 2 :

An unexpected process (PID=0x00001CAC) has tried to lock the machine '**BOXES\_sbldb\_1554029233481\_13617**', while only the process started by LaunchVMProcess (PID=0x00001E9C) is allowed.

Result Code: E\_ACCESSDENIED (0x80070005)

Component: MachineWrap

Interface: IMachine {5047460a-265d-4538-b23e-ddba5fb84976}

**Solution 1 : VT-X enabling**

**Solution 2 :** I solved this problem by disabnling PAE/NX in the VM Settings -System - Processor. Also made sure that the number of CPU was set to one. No changes in BIOS were needed for me.

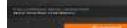
**Solution 3 :** For this you have to do the setting at bios level..

1. go to the BIOS after pressing F10 or as per your Laptop series/model.
2. Go to Configuration-> There find the Virtualization Technology in disble mode, Just enable it.
3. Save & Exit
4. Restart the machine and go to VM and try now..

It will work

Vagrant



Vagrant is a concept of virtualization.  book from stack overflow documentation

# @\$ BUILDING A VAGRANT BOX FROM START TO FINISH

September 16, 2014 | By [Tyler Bird](#)

**Note:** This blog was originally published on [tylerbird.org](http://tylerbird.org) earlier this year. It was recently updated for publication on the Engine Yard blog.

The goal of Vagrant is to make it so simple to create a local development environment, you'd never want to do it another way again. With two simple commands you can quickly setup your first vagrant environment and with a third command, be connected into your first vagrant box in under a few minutes.

```
vagrant init precise32 http://files.vagrantup.com/precise32.box  
vagrant up  
vagrant ssh # you're in!
```

This power is amazing and so useful for freelancers who work on many different projects, or companies that need to bring new members of a team up to speed as quickly as possible. With the use of Vagrant, the configuration of the development and the production environment can be mirrored as closely as possible. And errors like "works on my machine" can become a thing of the past.

If you're already a fan a Vagrant, I'm probably preaching to the choir. For more about what Vagrant can do check out the amazing documentation. Or do what I did and pick up the O'Reilly book by Vagrant's author, Mitchell Hashimoto, *Vagrant: Up and Running* and read it. It's a short but jam packed review of the details you need to get to know Vagrant, inside and out.

## Why Build A Box?

There are a bunch of amazing boxes out there available on sites like [vagrantbox.es](http://vagrantbox.es) and [vagrantcloud.com](http://vagrantcloud.com) so why would you want to build your own box?

Maybe you want to add a few extra things to your base (a runtime or two like Julia, Erlang, JVM or Python, etc.) then start this as your new "base".

Maybe you want your box to have more ram or you need your boxes to more closely mirror production and you are building a ram enriched, multiple server cluster with multiple provisioners, we get it... you've got a mountain to climb, you just need your gear.

## Definitions

What is a package.box file? When using the VirtualBox provider, it's a tarred, gzip file containing the following:

```
Vagrantfile  
box-disk.vmdk  
box.ovf  
metadata.json
```

The Vagrantfile has some information that will be merged into your Vagrantfile that is created when you run `vagrant init boxname` in a folder.

The `box-disk.vmdk` is the virtual hard disk drive.

The `box.ovf` defines the virtual hardware for the box.

The `metadata.json` tells vagrant what provider the box works with.

NOTE: These contents would be different for the VMWare provider, etc. For more about that please refer to the vagrant docs on boxes.

## Shout Out

I'm a fan of Vagrant and all things automation, they make my life a lot easier. First I'd like to give credits to my main sources which are:

- The blog article [Creating a Custom Box from Scratch](#) by Ryan Skoblenick.
- And the book *Vagrant: Up and Running* by Mitchell Hashimoto.

So why did I write this blog article, if Ryan's existed? Good question.

Well it only got me 90% of the way there. I had problems getting the Guest Tools installed and configuring the SSH User for the operating system using his article.

So my hope is that this article will improve upon that process and add a voice of sanity to those who need some help getting a box built by hand.

## Getting Prepared

If you don't already have Vagrant and VirtualBox, grab those.

Download Vagrant installer for your operating system.

Download VirtualBox installer for your operating system.

NOTE: Grab the VirtualBox Extension Pack as well. For more about the functionality it provides read [here](#).

Prior to the installation packages, Vagrant was distributed as a RubyGem. Installation packages are now the preferred way to install Vagrant, so you should uninstall the RubyGem version and follow the instructions for your platform. The RubyGem-based installation is still supported for Vagrant 1.0.x, but is deprecated and will not be supported in any future versions.

*Page 8, "Vagrant: Up and Running by Mitchell Hashimoto (O'Reilly). Copyright 2013 Mitchell Hashimoto, 978-1-449-33583-0."*

RUBY USERS: If you had Vagrant installed as a rubygem, uninstall it before you install vagrant:

```
gem uninstall vagrant
```

Run the Vagrant installer and the VirtualBox and VirtualBox Extension Pack installers, once they are installed, if you feel like a reboot would help, go ahead.

Download ubuntu server, the rest of the instructions will be for ubuntu, but if you're choosing another distro you're probably fine to adapt the commands that follow.

## Build a Box

We're going to use VirtualBox to build an ubuntu server from scratch. The reason for this is because Vagrant has native support for VirtualBox. There are more plugins out there for other providers like VMWare, Parallels or Vagrant-LXC, etc. We'll stick to VirtualBox for this guide.

When we setup our ubuntu server, it prompts us to setup a default user. We're going to name that user vagrant so it's the default user as well. This will make it the default SSH user and streamline the process.

## Configure the Virtual Hardware

Create a new Virtual Machine with the following settings:

- Name: vagrant-ubuntu64
- Type: Linux
- Version: Ubuntu64
- Memory Size: 512MB (to taste)
- New Virtual Disk: [Type: VMDK, Size: 40 GB]

Modify the hardware settings of the virtual machine for performance and because SSH needs port-forwarding enabled for the vagrant user:

- Disable audio
- Disable USB
- Ensure Network Adapter 1 is set to NAT
- Add this port-forwarding rule: [Name: SSH, Protocol: TCP, Host IP: blank, Host Port: 2222, Guest IP: blank, Guest Port: 22]

Mount the Linux Distro ISO and boot up the server.

## Install the Operating System

Setting up ubuntu is very simple. Follow the on-screen prompts. And when prompted to setup a user, set the user to vagrant and the password to vagrant. It will give you a passive-aggressive guilt trip about it being a weak sauce password. Don't let that shake you, be strong and soldier through.

Installation complete!

These are popular snaps in server environments. Select or deselect with SPACE, press ENTER to see more details of the package, publisher and versions available.

```
* microk8s           Kubernetes for workstations and appliances
nextcloud           Nextcloud Server - A safe home for all your data
kata-containers     Lightweight virtual machines that seamlessly plug into t
* docker             Docker container runtime
canonical-livepatch Canonical Livepatch Client
rocketchat-server   Group chat server for 100s, installed in seconds.
mosquitto           Eclipse Mosquitto MQTT broker
etcd                Resilient key-value store by CoreOS
* powershell         PowerShell for every system!
stress-ng           A tool to load, stress test and benchmark a computer sys
sabnzbd             SABnzbd
wormhole            get things from one computer to another, safely
aws-cli              Universal Command Line Interface for Amazon Web Services
google-cloud-sdk    Command-line interface for Google Cloud Platform product
slcli               Python based SoftLayer API Tool.
doctl               DigitalOcean command line tool
conjure-up           Package runtime for conjure-up spells
minidlna-escoand   server software with the aim of being fully compliant wi
postgresql10        PostgreSQL is a powerful, open source object-relational
heroku              CLI client for Heroku
keepalived          High availability VRRP and load-balancing for Linux
* prometheus        The Prometheus monitoring system and time series databas
juju                juju client
```

[ Done ]

10 / 11

Install complete

## Set Root Password

In order to setup the Super User, aka root user, you'll need to be able to sign in as that user. Since I asked you to make vagrant the default user while installing the Operating System in the last step, these commands should help you set the root password and then sign in as root in order to make the next configuration changes below.

```
sudo passwd root
```

This will prompt you to type the password twice, where I'd suggest the password "vagrant". Now sign in as the root user in order to Setup the Super User next.

```
su -
```

## Setup the Super User

Vagrant must be able run sudo commands without a password prompt and if you're not configuring ubuntu at this point, just make sure that requiretty is disabled for the vagrant user.

The most efficient way I've found to setup the vagrant user so it's able to use sudo without being prompted for a password is to add it to the sudoers list like this:

```
sudo visudo -f /etc/sudoers.d/vagrant
```

Anything in the /etc/sudoers.d/\* folder is included in the "sudoers" privileges when created by the root user. So that's why we created this as the root user and in that folder.

With that file open add this to the file then save it and exit.

```
# add vagrant user
vagrant ALL=(ALL) NOPASSWD:ALL
Now you can test that it works by running a simple command:
```

```
sudo pwd
```

It will return the home folder without prompting you for a password if everything is setup correctly. If you are prompted for a password, something's out of wack and things won't work right. This step was CRUCIAL for me, so please ensure this test passes for you.

## Creating sudo user and provide the sudo priveliges:

```
sudo adduser airflow
[sudo] password for tole:
Adding user `airflow' ...
Adding new group `airflow' (1001) ...
Adding new user `airflow' (1001) with group `airflow' ...
Creating home directory `/home/airflow' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
```

```

Retype new UNIX password:
passwd: password updated successfully
Changing the user information for airflow
Enter the new value, or press ENTER for the default
  Full Name []: airflow
vi /etc/sudoers
# This file MUST be edited with the 'visudo' command as root.
#
# Please consider adding local content in /etc/sudoers.d/ instead of
# directly modifying this file.
#
# See the man page for details on how to write a sudoers file.
#
Defaults        env reset
Defaults        mail_badpass
Defaults        secure_path="/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/snap/bin"

# Host alias specification
# User alias specification
# Cmnd alias specification
# User privilege specification
#root    ALL=(ALL:ALL) ALL
root ALL=(ALL) NOPASSWD:ALL
vagrant ALL=(ALL) NOPASSWD:ALL
bitbucket ALL=(ALL) NOPASSWD:ALL
# Members of the admin group may gain root privileges
gadmin ALL=(ALL) NOPASSWD:ALL
# Allow members of group sudo to execute any command
%sudo  ALL=(ALL:ALL) PASSWD:ALL
# See sudoers(5) for more information on "#include" directives:
#include /etc/sudoers.d

```

## Updating The Operating System

One of the reasons we are building this box is so we can save time by already being as up to date as the box was built. So let's get up-to-date first.

```

sudo apt-get update -y
sudo apt-get upgrade -y
Usually if there are kernel updates you'll want to reboot the server. So do that.

```

```
sudo shutdown -r now
```

## SSH keys Install the as Vagrant Key

The only way that all the vagrant commands will be able to communicate over ssh from the host machine to the guest server is if the guest server has this "insecure vagrant key" installed. It's called "insecure" because essentially everyone has this same key and anyone can hack into everyone's vagrant box if you use it.

But at the same time, we're hoping you're not running around with all your most valuable company data on your vagrant boxes, right? RIGHT? OK. Good.

```

mkdir -p /home/vagrant/.ssh
chmod 0700 /home/vagrant/.ssh
wget --no-check-certificate \
  https://raw.github.com/mitchellh/vagrant/master/keys/vagrant.pub \
  -O /home/vagrant/.ssh/authorized_keys
chmod 0600 /home/vagrant/.ssh/authorized_keys
chown -R vagrant /home/vagrant/.ssh

```

## Install and Configure OpenSSH Server

If you didn't install SSH while installing the operating system you can do it now: sudo apt-get install -y openssh-server

We need to edit the /etc/ssh/sshd\_config file:

```

sudo nano /etc/ssh/sshd_config
Find and uncomment the following line because we added the Vagrant key above to the authorized_keys file:

```

```

AuthorizedKeysFile %h/.ssh/authorized_keys
Then restart ssh:

```

```
sudo service ssh restart
```

SSH will fail to start if there are syntax errors in the /etc/ssh/sshd configuration file. The command following command will tell you if any directives are incorrect:

```
/usr/sbin/sshd -T
```

If the configuration test does not return any errors, I would suggest starting sshd in debugging mode, this will provide you with a detailed startup of the service:

```
/usr/sbin/sshd -ddd
```

## If you are asking for password while executing SSH to server

Also check the /etc/ssh/sshd config file.

### Precheck of getting solution:

PubkeyAuthentication should be set to yes

There is also the AuthorizedKeysFile directive which determines the path where the authorized keys should be located. Ensure it's commented out or on the default of %h/.ssh/authorized\_keys.

### Solution:

Issue resolved by changing the permission on destination "chmod 750 /home/siebel"

## Installing Guest Tools

Guest Tools help the operating system handle shared folders and "optimize the guest operating system for better performance and usability." 2

A compiler is required to install the Guest Tools, use this command:

```
sudo apt-get install -y gcc build-essential linux-headers-server
```

In VirtualBox browse to the Devices menu at the top, then in the drop-down list at the bottom, click on Insert Guest Additions CD Image.

This will add an ISO image to the virtual CDROM running in your server. Run these commands to mount your cdrom and then run the script. NOTE: The message about the cdrom being read-only is fine.

```
sudo mount /dev/cdrom /mnt  
cd /mnt  
sudo ./VBoxLinuxAdditions.run
```

## Installing the Guest Additions on a GUI-less server

1. Start VirtualBox.
2. Start the host in question.
3. Once the host has booted, click Devices | Insert Guest Additions CD Image.
4. Log in to your guest server.
5. Mount the CD-ROM with the command `sudo mount /dev/cdrom /media/cdrom`.
6. Change into the mounted directory with the command `cd /media/cdrom`.
7. Install the necessary dependencies with the command `sudo apt-get install -y dkms build-essential linux-headers-generic linux-headers-$(uname -r)`.
8. Change to the root user with the command `sudo su`.
9. Install the Guest Additions package with the command `./VBoxLinuxAdditions.run`.
10. Allow the installation to complete.

## Package the Box

Before you package the box you'll want to "zero out" the drive. "This fixes fragmentation issues with the underlying disk, which allows it to compress much more efficiently later." 3

```
sudo dd if=/dev/zero of=/EMPTY bs=1M  
sudo rm -f /EMPTY
```

Now we're ready to package the box. I usually make a folder to hold my boxes like so:

```
mkdir ~/code/personal/vagrant boxes  
cd ~/code/personal/vagrant boxes
```

This is the command that finally packages up the box for you as we defined above into the compressed gzip tarball file, it also generates and includes the Vagrantfile and the metadata.json file.

## Vagrant package Clean

Please do the cleanup the virtual machine you cared before packaging.

```
root@manager:~# apt-get clean  
root@manager:~# dd if=/dev/zero of=/EMPTY bs=1M  
dd: error writing '/EMPTY': No space left on device  
34596+0 records in  
34595+0 records out  
36275929088 bytes (36 GB, 34 GiB) copied, 214.505 s, 169 MB/s  
root@manager:~# rm -f /EMPTY  
root@manager:~# cat /dev/null > ~/.bash_history && history -c && exit  
exit
```

```
Before [2.78 GB]: -rw-r--r-- 1 80053806 1049089 2983807693 Mar 7 12:53 master.box  
After [1.79 GB]: -rw-r--r-- 1 80053806 1049089 1920359356 Mar 7 13:52 master.box
```

## Command to package a box from vagrant

```
$ Vagrant package --base VMS_master_1551181951081_60377 --output master.box
```

```
vagrant package --base vagrant-ubuntu64
Vagrant will then check VirtualBox for any instances of the name vagrant-ubuntu64 and attempt to ssh into them and control them.

→ vagrant package --base vagrant-ubuntu64
[vagrant-ubuntu64] Attempting graceful shutdown of VM...
[vagrant-ubuntu64] Forcing shutdown of VM...
[vagrant-ubuntu64] Clearing any previously set forwarded ports...
[vagrant-ubuntu64] Exporting VM...
[vagrant-ubuntu64] Compressing package to: /Users/tbird/code/personal/virtual_boxes/package.box
You are left with the package.box file in your ~/code/personal/vagrant_boxes folder.
```

## Test Your Box

From your same `vagrant_boxes` folder you can run these final test commands. All the heavy lifting is really done at this point. If you've screwed up something it's probably in a step up above.

You should be in `~/code/personal/vagrant_boxes/` and type:

```
vagrant box add ubuntu64 package.box
vagrant init ubuntu64
vagrant up
Connect to the server you created from start to finish!

vagrant ssh
You've won, you deserve to get your high five on.
```

## Packer : is a immutable means which is not manual

Just when you thought it was safe to do things manually at the beginning ... in comes Packer. What is Packer, you say? Well Packer automates everything we just did. /FACEPALM

Anyway... Once you have Vagrant, VirtualBox, and Packer installed you can define a `quick-start.json` file like:

```
{
  "builders": [
    {
      "type": "amazon-ebs",
      "access_key": "YOUR KEY HERE",
      "secret_key": "YOUR SECRET KEY HERE",
      "region": "us-east-1",
      "source_ami": "ami-de0d9eb7",
      "instance_type": "t1.micro",
      "ssh_username": "ubuntu",
      "ami_name": "packer-example"
    }
  ]
}
After changing the Access and Secret Key you could run the packer command with the quick-start.json file:
```

```
packer build quick-start.json
Packer will then automate the creation of that quick-start.json machine image for Amazon EC2. That can optionally include the creation of a Vagrant box.
```

What Packer aims to do is to make their images compatible with all these providers: Amazon EC2 (AMI), DigitalOcean, Docker, Google Compute Engine, OpenStack, QEMU for KVM or Xen instances, or even VirtualBox or VMWare software. (See platforms for the entire list.)

## What Have We Learned

What have we learned from this process of building boxes for Vagrant?

The process of automation is the process of taking yourself out of the equation. "Auto" is defined as self, and the suffix "ation" is defined as action. So if we think of the job of automating your development environment or production environment so it runs without your intervention, then your actions are doing it right!

And even if we want to do it by hand or use Packer, knowing how things work is a lot better than guessing how they work. And I like to experience things a few times before I start to automate those boring parts away, and then focus on new more interesting challenges.

**Deploy and run your Ruby apps on Engine Yard**  
START MY TRIAL!

**SCP command: [http://www.hypexr.org/linux\\_scp\\_help.php](http://www.hypexr.org/linux_scp_help.php)**

**StrictHostKeyChecking yes → not required**

sdes013% vi /etc/ssh/ssh\_config

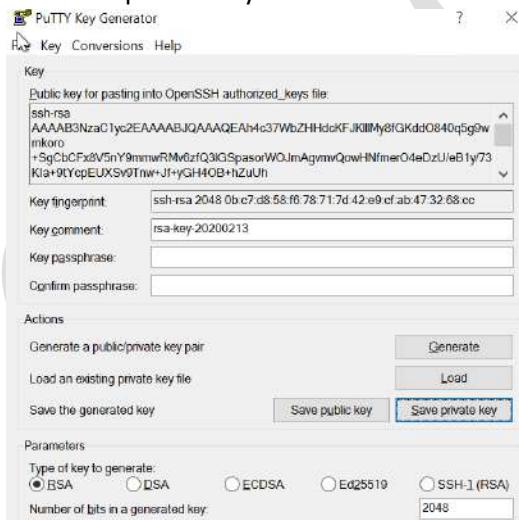
scp scp -r fm\_ee\_pynt\_pol\_allocate\_items.c billdev@eecsaruh2hos055:/export/home/billdev/gouse

password issue: give the permission 755 to home user → chmod 755 billdev

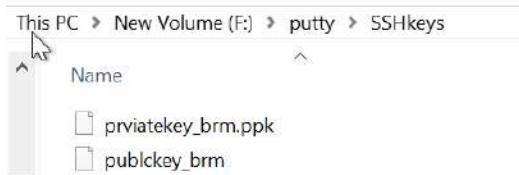
```
sdes013% vi /etc/ssh/ssh_config
sdes013% scp -r fm_ee_pynt_pol_allocate_items.c billdev@eeecsaruh2hos056:/export/home/billdev/gouse
The authenticity of host 'eeecsaruh2hos056 (10.14.11.173)' can't be established.
RSA key fingerprint is SHA256:CvECaggJrj4vYuODHj0vN/1KyY8rnN5zTIwdGu+KYIw.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/export/home/bradmin/.ssh/known_hosts).
-----
| This computer system has been provided for official use of
| Etihad Etisalat Co.- Mobily. All information including personal
| information, placed on or sent over this system is being monitored
| Uses of this system, authorized or unauthorized, constitutes consent
| to monitoring of this system. Unauthorized use will subject the user
| to disciplinary action and criminal prosecution. Evidences of any
| such unauthorized use collected during monitoring may be used for
| disciplinary, criminal or other adverse action.
| -----
Password:
sdes013% scp -r fm_ee_pynt_pol_allocate_items.c billdev@eeecsaruh2hos053:/export/home/billdev/gouse
The authenticity of host 'eeecsaruh2hos053 (10.14.11.248)' can't be established.
RSA key fingerprint is SHA256:pCvTPyDPAYRmSvm3lsyY+YbrFfZB1tu/20LZv6DlPs.
Are you sure you want to continue connecting (yes/no)? yes
Failed to add the host to the list of known hosts (/export/home/bradmin/.ssh/known_hosts).
-----
| This computer system has been provided for official use of
| Etihad Etisalat Co.- Mobily. All information including personal
| information, placed on or sent over this system is being monitored
| Uses of this system, authorized or unauthorized, constitutes consent
| to monitoring of this system. Unauthorized use will subject the user
| to disciplinary action and criminal prosecution. Evidences of any
| such unauthorized use collected during monitoring may be used for
| disciplinary, criminal or other adverse action.
| -----
Password:
```

## Automated connections configurations:

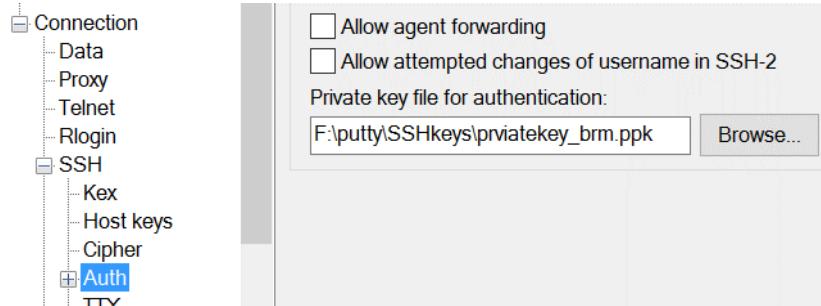
- Generate private key



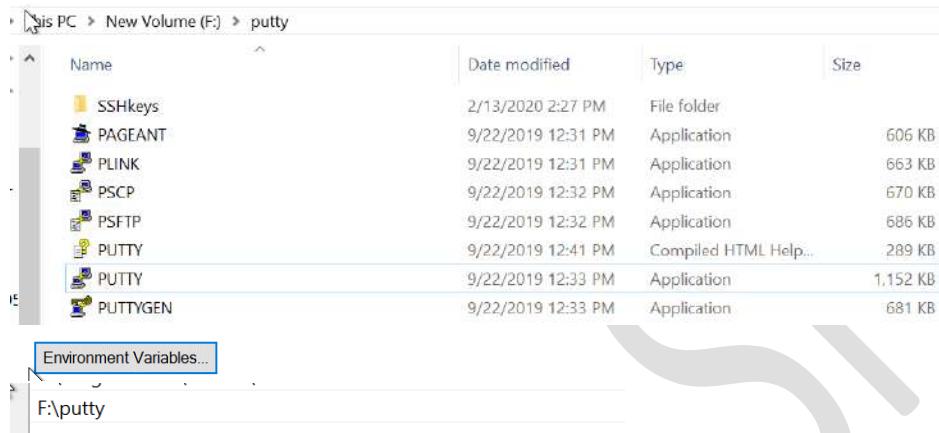
Save pulic and private keys under below path



- Configure private key on destination server authorization\_key under .ssh



- Configure same private key in putty
- Configure environment variable for the putty download directory



- Then try to download/upload a file from scp commands.

```
C:\Users\80053806>plink billdev@BRM_SIT
Using username "billdev".
-- Pre-authentication banner message from server: -----
| |
| This computer system has been provided for official use of
| Etihad Etisalat Co.- Mobily. All information including personal
| information, placed on or sent over this system is being monitored
| Uses of this system, authorized or unauthorized, constitutes consent
| to monitoring of this system. Unauthorized use will subject the user
| to disciplinary action and criminal prosecution. Evidences of any
| such unauthorized use collected during monitoring may be used for
| disciplinary, criminal or other adverse action.
| |
| End of banner message from server -----
```

Automated script for plink

<https://www.thegeekstuff.com/2017/05/putty-plink-examples/>

```
C:\>type commands.txt
hostname
service mysql stop
yum -y install httpd
service mysql start
service httpd start
crontab -l
```

you can call all commands from one file as follows.

```
C:\>plink root@192.168.101.1 -m C:\commands.txt
```

```
plink billdev@10.14.11.173 /root/bin/db-backup.sh
```

```
plink -i "C:\Downloads\devdb.ppk" root@192.168.101.1 hostname
```

## To download files from server[Remote] to windows[local machine]

```
pscp billdev@BRM_SIT:/export/home/billdev/gouse/*.* F:\PrintShop\SIT_XML
```

In your terminal, type:

```
scp your_username@remotehost.edu:foobar.txt /local/dir  
replacing the username, host, remote filename, and local directory as appropriate.
```

If you want to access EC2 (or other service that requires authenticating with a private key), use the -i option:

```
scp -i key_file.pem your_username@remotehost.edu:/remote/dir/foobar.txt /local/dir
```

From: [http://www.hypexr.org/linux\\_scp\\_help.php](http://www.hypexr.org/linux_scp_help.php)

Jenkins job:

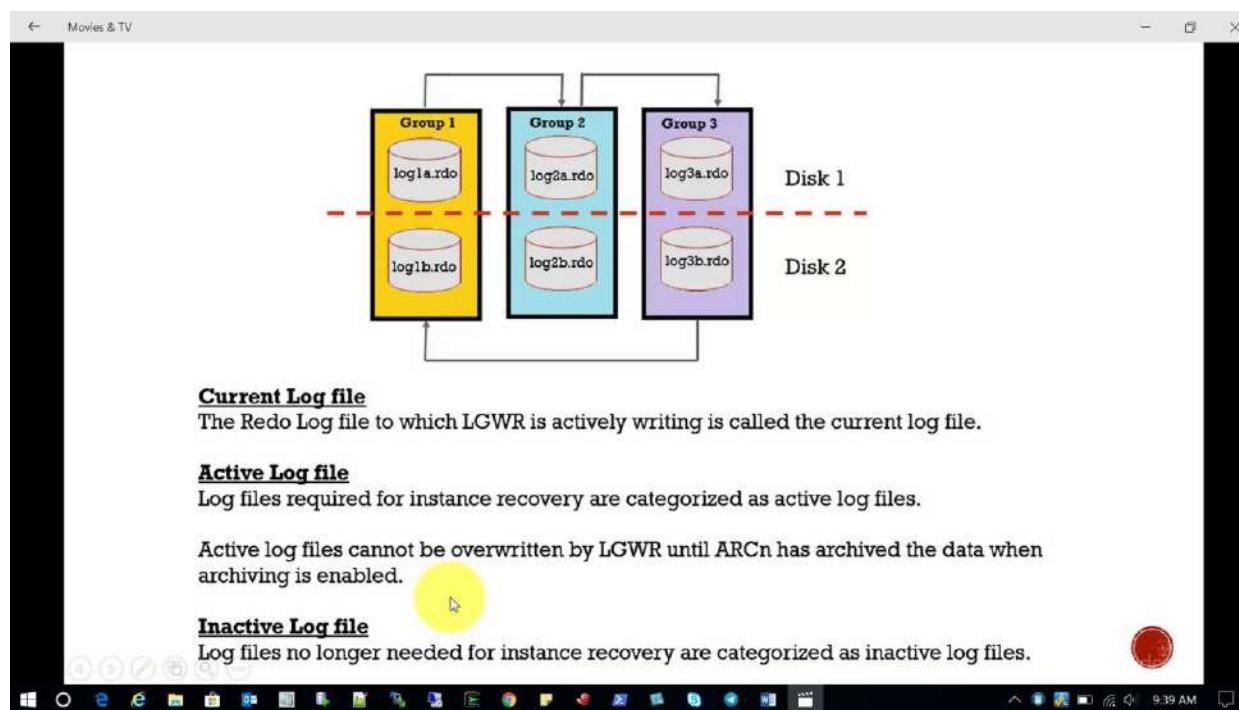
```
F:\Jenkins\workspace\XML_For_SIT>pscp -i "F:\putty\SSHkeys\prviatekey_brm.ppk"  
billdev@BRM_SIT:/export/home/billdev/gouse/Corporate/*.*  
F:\PrintShop\SIT_XML\Corporate  
ssh_init: Host does not exist
```

## script to check files available or not:

```
echo "Looking for Consumer files"  
cd /opt/common/portal/7.5/brm/apps/pin_inv/invoice_dir/Consumer  
for f in *.xml; do  
    # do something if file found  
    echo "Working on $f"  
    ls -ltr  
    chmod 777 "$f"  
    ls -ltr  
    cp "$f" /export/home/billdev/gouse/Consumer  
done  
echo "Cleaning Up all Files"  
rm -rf *.*  
  
echo "Looking for Corporate files"  
  
cd /opt/common/portal/7.5/brm/apps/pin_inv/invoice_dir/Corporate  
for f in *.xml; do  
    # do something if file found  
    echo "Working on $f"  
    ls -ltr  
    chmod 777 "$f"  
    ls -ltr  
    cp "$f" /export/home/billdev/gouse/Corporate  
done  
echo "Cleaning Up all Files"  
rm -rf *.*
```

## Vagrant user login Issue

Issue : Vagrant SSH is not working by mobaxterm when we are trying to login manually with Vagrant user.



### Current Log file

The Redo Log file to which LGWR is actively writing is called the current log file.

### Active Log file

Log files required for instance recovery are categorized as active log files.

Active log files cannot be overwritten by LGWR until ARCn has archived the data when archiving is enabled.

### Inactive Log file

Log files no longer needed for instance recovery are categorized as inactive log files.

```
Login: root
Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
```

Solution:

- ✓ You may try edit in /etc/ssh/sshd config  
PasswordAuthentication yes
  - ✓ service sshd restart
  - ✓ Reset the password for vagrant
- ```
[vagrant@localhost ~]$ vi /etc/ssh/sshd_config
[vagrant@localhost ~]$ sudo su root
[root@localhost vagrant]# ls /etc/ssh/sshd_config
/etc/ssh/sshd_config
[root@localhost vagrant]# vi /etc/ssh/sshd_config
[root@localhost vagrant]# service sshd restart
Redirecting to /bin/systemctl restart sshd.service
[root@localhost vagrant]# passwd vagrant
Changing password for user vagrant.
New password:
BAD PASSWORD: The password is shorter than 8 characters
Retype new password:
passwd: all authentication tokens updated successfully.
```

## Synced folders within your Vagrantfile

Synced folders are configured within your Vagrantfile using the config.vm.synced\_folder method.

```
Vagrant.configure("2") do |config|
  # other config here
  config.vm.synced_folder "src/", "/srv/website"
  #config.vm.synced_folder "C:\\src", "/srv/website"
```

```
end
```

Reference : [https://www.vagrantup.com/docs/synced-folders/basic\\_usage.html](https://www.vagrantup.com/docs/synced-folders/basic_usage.html)

#### Issue while booting Vagrant box

Vagrant box. For context, the command attempted was:  
mount -t vboxsf -o uid=1000,gid=1000 vagrant /vagrant  
The error output from the command was:  
: No such device

Solution : [vagrant plugin install vagrant-vbguest](#)

Then reload the vagrant box  
<https://sonnguyen.ws/vagrant-failed-mount-folders-linux-guest/>

#### Vagrant box cleanup

```
vagrant box list | cut -f 1 -d ' ' | xargs -L 1 vagrant box remove -f --all
```

#### Convert a Virtualbox.ova VM into a vagrant box

<https://www.howtogeek.com/125640/how-to-convert-virtual-machines-between-virtualbox-and-vmware/>

Sometimes distribution providers (such as [UCS](#)) only give you VirtualBox .ova files to test their software. Here is how you can easily and non-interactively import a .ova file into a .box for use with Vagrant.  
Open CMD, go to virtual box installed directory `cd C:\Program Files\Oracle\VirtualBox`. navigate to above virtual box directory and execute below command with the below .ova file to convert to vagrant box.

```
VBoxManage import C:\VMS\BRM-PS12.ova --vsys 0 --eula accept
$ VBoxManage import ./UCS-Virtualbox-Demo-Image.ova --vsys 0 --eula accept
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
Interpreting /home/crohr/dev/ucs//UCS-Virtualbox-Demo-Image.ova...
OK.
Disks: vmdisk1 53687091200      -1
http://www.vmware.com/interfaces/specifications/vmdk.html#streamOptimized          UCS-Demo-Image-virtualbox-
disk1.vmdk      -1      -1
...
0%...10%...20%...30%...40%...50%...60%...70%...80%...90%...100%
Successfully imported the appliance.
Then list your VMs to find the VM ID:
cd C:\Program Files\Oracle\VirtualBox
$ VBoxManage list vms
"UCS 4.1" {acef4c0a-35be-4640-a214-be135417f04d}
You can now package that VM as a Vagrant box:
```

Note : you have to navigate to directory where the vagrant installed or vagrantfile available. I have installed vagrant in c:\vagrant and vagrantfile initiated from [cd c:\VMS](#)  
\$ vagrant package --base acef4c0a-35be-4640-a214-be135417f04d --output UCS.box  
vagrant package --base 8dd5dc90-d09b-4ed7-8ael-cb6fdb58cda6 --output sblldb.box  
==> acef4c0a-35be-4640-a214-be135417f04d: Exporting VM...  
==> acef4c0a-35be-4640-a214-be135417f04d: Compressing package to: /home/crohr/dev/ucs/UCS.box

[cd c:\VMS](#)

```
vagrant package --base 65500f08-8489-4ac0-a373-7b0629a7a862 --output C:\VMS\BRM-PS12.box
```

And add it to the list of your local Vagrant boxes:

```
$ vagrant box add UCS.box --name UCS
$ vagrant box add C:\VMS\BRM.box --name BRMVM12
$ vagrant box add BRMVM12 BRM.box
$ vagrant box list
```

Finally, you can create a Vagrantfile to use this box:

```
Vagrant.configure("2") do |config|
  config.vm.box = "UCS"
  #
end
```

And vagrant up!

#### config.vm.boot\_timeout

issue : If the box appears to be booting properly, you may want to increase the timeout ("config.vm.boot\_timeout") value.

Solution: open you gitbash as administrator and run the below command netsh winsock reset.

```
MINGW64:/c/Users/80053806
80053806@RY-LP-80053806I MINGW64 ~
$ netsh winsock reset
Successfully reset the Winsock Catalog.
You must restart the computer in order to complete the reset.
```

Go to vagrant path and run the up command.  
# Vagrant up ubuntu

## Upload your custom box to Vagrant Cloud:

```
C:\Program Files>cd Oracle
C:\Program Files\Oracle>cd VirtualBox
C:\Program Files\Oracle\VirtualBox>VBoxManage list vms
go to Virtualbox installed directory and run the
below command to get the VMS id's

"VMS_ubuntu_1550668937603_67256" {9356d35a-38b0-449a-b2d1-ac7d09be6018}
"VMS_node1_1550670688980_15381" {c8495e61-29a7-4e0e-809f-89883e545d9f}
"VMS_node2_1550672442990_39876" {4af285cc-5b39-41c8-9010-7670da3d22ac}

C:\Program Files\Oracle\VirtualBox>cd
C:\Program Files\Oracle\VirtualBox

C:\Program Files\Oracle\VirtualBox>cd ..
C:\Program Files\Oracle>cd ..

C:\>cd VMS
go to vagrantfile where you have setup done and run the
vagrant package command.

C:\VMS>ls
ORABRM.box Vagrantfile database packer scripts siebel017

C:\VMS>vagrant package --base 9356d35a-38b0-449a-b2d1-ac7d09be6018 --output C:\VMS\ubuntu-docker-ce.box
--> 9356d35a-38b0-449a-b2d1-ac7d09be6018: Clearing any previously set forwarded ports...
--> 9356d35a-38b0-449a-b2d1-ac7d09be6018: Exporting VM...
--> 9356d35a-38b0-449a-b2d1-ac7d09be6018: Compressing package to: C:/VMS/ubuntu-docker-ce.box
```

Create a account in vagrant cloud and follow the step.

<https://app.vagrantup.com/boxes/search>



The screenshot shows two steps in the Vagrant Cloud provider configuration process. The top part is titled "Edit Provider" and shows a "Provider" dropdown menu with "virtualbox" selected. Below it is a note: "Vagrant will need to know how to use this provider." At the bottom are "Update provider" and "Delete provider" buttons. The bottom part is titled "Add Provider File" and shows an "Upload File" button with "Choose File" and "ubuntu-docker-ce.box" selected. Below it is a "Status" field showing "Uploading 10.28/742.56MB (1%)".

Refer link : <http://blog.ycshao.com/2017/09/16/how-to-upload-vagrant-box-to-vagrant-cloud/>

### Command to randomise MAC address in Virtualbox

I discovered the `auto` parameter, but it doesn't work? `VBoxManage.exe modifyvm MyVM --macaddress1 auto` runs without errors but also doesn't change the MAC, according to the Virtualbox Settings GUI, and the `.vbox` config file.

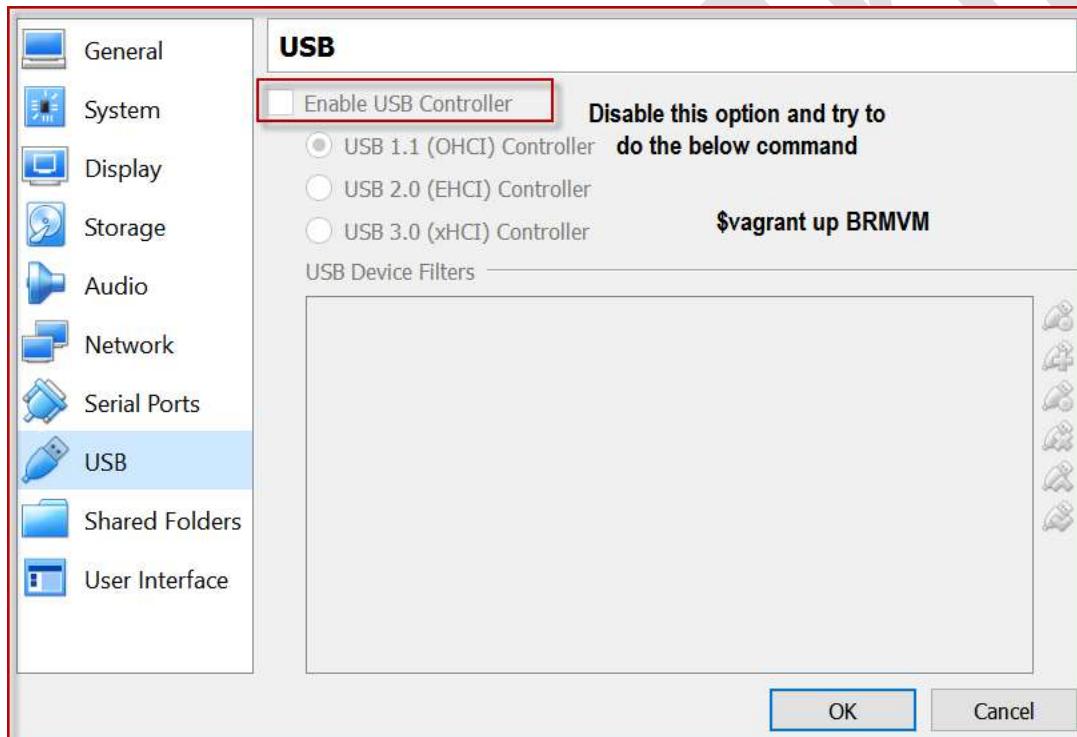
Actually, it kinda works (the GUI doesn't update instantly), but it seems it resets other settings if you run `macaddress1` and no other attributes. It might require you to re-set/specify all settings for that VM when you run `modifyvm`.

```
2005:BB06:WY-LP-8005:8006$ /cygdrive/c/VMS
$ vagrant up BRMVM
Bringing machine 'BRMVM' up with 'virtualbox' provider...
==> BRMVM: Clearing any previously set network interfaces...
==> BRMVM: Preparing network interfaces based on configuration...
BRMVM: Adapter 1: nat
BRMVM: Adapter 2: hostonly
BRMVM: Forwarding ports...
BRMVM: 22 (guest) => 2203 (host) (adapter 1)
BRMVM: Booting VM...
There was an error while executing 'VBoxManage', a CLI used by Vagrant
for controlling VirtualBox. The command and stderr is shown below.

Command: ["startvm", "240acaal-1e0b-465a-9113-8b2d69c245db", "--type", "headless"]

Stderr: VBoxManage.exe: error: Implementation of the USB 2.0 controller not found!
VBoxManage.exe: error: Because the USB 2.0 controller state is part of the saved VM state, the VM cannot be started. To fix this problem, either install the 'Oracle VM VirtualBox Extension Pack' or disable USB 2.0 support in the VM settings.
VBoxManage.exe: error: Note: This error could also mean that an incompatible version of the 'Oracle VM VirtualBox Extension Pack' is installed (VERR_NOT_FOUND)
VBoxManage.exe: error: Details: code E_FAIL (0x80004005), component ConsoleWrap, interface IConsole
```

Command: `["startvm", "240acaal-1e0b-465a-9113-8b2d69c245db", "--type", "headless"]`



## Oracle vagrant linux boxes:

<http://yum.oracle.com/boxes/oraclelinux>

## Introduction

Vagrant is a tool to manage virtual machine-based development environments. Vagrant Boxes are pre-built base images that can be imported into Vagrant as a starting point. Read more about Vagrant here. On this page Oracle publishes Vagrant boxes based on Oracle Linux.

Using Oracle Linux Vagrant boxes

## Prerequisites

You will need the following software installed on your machine. Check the minimum required versions in the Vagrant Box description.

- [Oracle VM VirtualBox](#)
- [Vagrant](#)

## Available Boxes

Unless otherwise stated, these boxes have the following specifications:

- Provider: VirtualBox
- 64 bit
- 2 vCPUs
- 2048 MB RAM
- Minimal package set installed
- 32 GB root volume
- 4 GB swap
- XFS root filesystem
- Yum configured for Oracle's Yum server - [yum.oracle.com](http://yum.oracle.com)

| Box                     | Description                                                    | URL                                                                                                                       | Size (MB) | MD5 Checksum                      |
|-------------------------|----------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------|-----------|-----------------------------------|
| <b>Oracle Linux 7.3</b> | Guest additions 5.1.14<br>Kernel: 4.1.12-61.1.28.el7uek.x86_64 | <a href="http://yum.oracle.com/boxes/oraclelinux/ol73/ol73.box">http://yum.oracle.com/boxes/oraclelinux/ol73/ol73.box</a> | 378       | e2bbb85d6d54a1221e4bb2ff07100941  |
| <b>Oracle Linux 6.8</b> | Guest additions 5.1.12<br>Kernel: 4.1.12-61.1.28.el6uek.x86_64 | <a href="http://yum.oracle.com/boxes/oraclelinux/ol68/ol68.box">http://yum.oracle.com/boxes/oraclelinux/ol68/ol68.box</a> | 390       | 1dcbaaff37e917aaa38c1c80c43058af9 |
| <b>Oracle Linux 6.9</b> | Guest additions 5.1.12<br>Kernel: 4.1.12-61.1.28.el6uek.x86_64 | <a href="http://yum.oracle.com/boxes/oraclelinux/ol69/ol69.box">http://yum.oracle.com/boxes/oraclelinux/ol69/ol69.box</a> | 373       | 3c430f27e9eb57c6dabf468ebec1354a  |

## Adding, downloading and starting the Oracle Linux Vagrant Box

Create a working directory for your development project and from within that directory, issue the following commands, substituting <name> and <url> as appropriate.

```
$ vagrant box add --name <name> <url>
$ vagrant init <name>
$ vagrant up
$ vagrant ssh
```

For example:

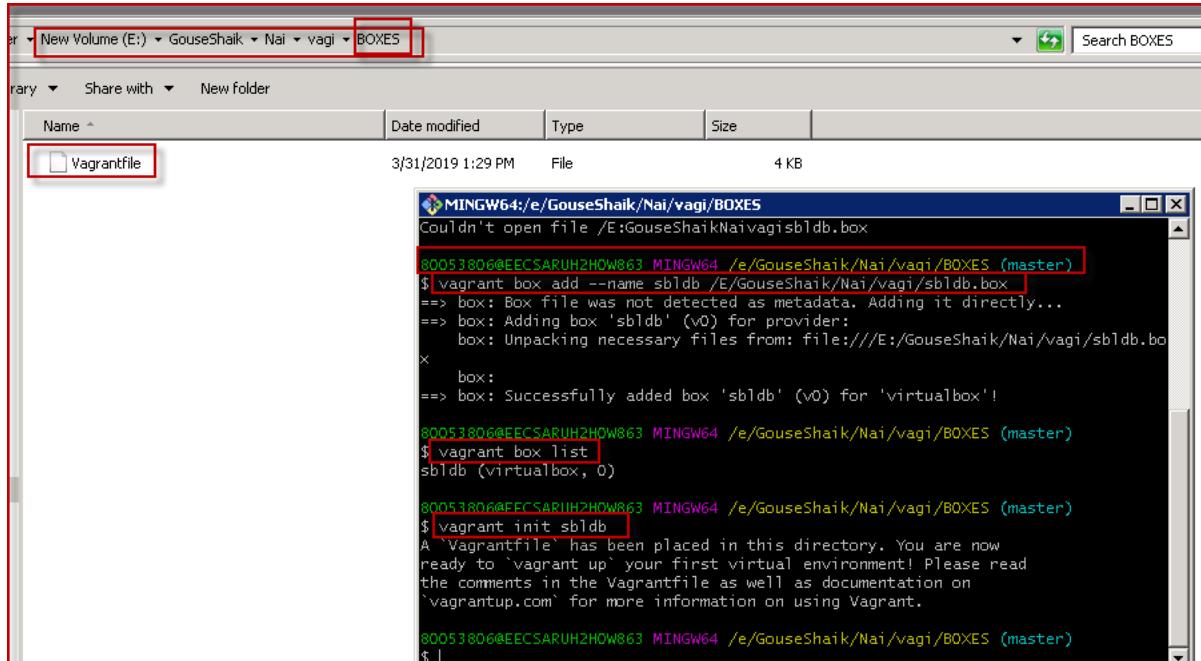
```
$ vagrant box add --name ol73 http://yum.oracle.com/boxes/oraclelinux/ol73/ol73.box
$ vagrant init ol73
$ vagrant up
$ vagrant ssh
```

## Vagrant boxes migrated to Local machine to RDP windows

1. Install virtual box : set the default location for virtual box to store the vms as in E: drive
2. Install vagrant box: set the default location to store all files in E: drive

3. Then navigate to directory you want to start the vagrant box, run below commands.

```
$ vagrant box add --name sbldb /e/GouseShaik/Nai/vagi/sbldb.box  
$ vagrant box list  
$ vagrant init sbldb
```



```
80053806@EECSARUH2HOW863 MINGW64 /e/GouseShaik/Nai/vagi/BOXES (master)  
$ vagrant box add --name sbldb /E/GouseShaik/Nai/vagi/sbldb.box  
==> box: Box file was not detected as metadata. Adding it directly...  
==> box: Adding box 'sbldb' (v0) for provider:  
box: Unpacking necessary files from: file:///E:/GouseShaik/Nai/vagi/sbldb.box  
box:  
==> box: Successfully added box 'sbldb' (v0) for 'virtualbox'!  
80053806@EECSARUH2HOW863 MINGW64 /e/GouseShaik/Nai/vagi/BOXES (master)  
$ vagrant box list  
sbldb (virtualbox, 0)  
80053806@EECSARUH2HOW863 MINGW64 /e/GouseShaik/Nai/vagi/BOXES (master)  
$ vagrant init sbldb  
A `Vagrantfile` has been placed in this directory. You are now  
ready to `vagrant up` your first virtual environment! Please read  
the comments in the Vagrantfile as well as documentation on  
`vagrantup.com` for more information on using Vagrant.
```

## Error at start

```
$ vagrant up sbldb  
$ vagrant up sbldb  
Bringing machine 'sbldb' up with 'virtualbox' provider...  
==> sbldb: Importing base box 'sbldb'...  
==> sbldb: Matching MAC address for NAT networking...  
==> sbldb: Setting the name of the VM: BOXES_sbldb_1554029233481_13617  
Vagrant is currently configured to create VirtualBox synced folders with  
the `SharedFoldersEnableSymlinksCreate` option enabled. If the Vagrant  
guest is not trusted, you may want to disable this option. For more  
information on this option, please refer to the VirtualBox manual:
```

<https://www.virtualbox.org/manual/ch04.html#sharedfolders>

This option can be disabled globally with an environment variable:

VAGRANT\_DISABLE\_VBOXSYMLINKCREATE=1

or on a per folder basis within the Vagrantfile:

```
config.vm.synced_folder '/host/path', '/guest/path', sharedFoldersEnableSymlinksCreate: false
```

```

==> sblldb: Clearing any previously set network interfaces...
==> sblldb: Preparing network interfaces based on configuration...
sblldb: Adapter 1: nat
sblldb: Adapter 2: hostonly
==> sblldb: Forwarding ports...
sblldb: 22 (guest) => 2200 (host) (adapter 1)
sblldb: 1521 (guest) => 1521 (host) (adapter 1)
==> sblldb: Booting VM...

```

There was an error while executing `VBoxManage`, a CLI used by Vagrant for controlling VirtualBox. The command and stderr is shown below.

Command: ["startvm", "3bcf2867-72d8-4a33-bdd5-e1eeb36d494d", "--type", "headless"]

```

Stderr: VBoxManage.exe: error: The native API dll was not found
(C:\Windows\system32\WinHVPlatform.dll) (VERR_NEM_NOT_AVAILABLE).
VBoxManage.exe: error: VT-x is not available (VERR_VMX_NO_VMX)
VBoxManage.exe: error: Details: code E_FAIL (0x80004005), component ConsoleWrap, interface IConsole

```

80053806@EECSARUH2HOW863 MINGW64 /e/GouseShaik/Nai/vagi/BOXES (master)

```

$ vagrant up sblldb
Bringing up 'sblldb' so with 'VirtualBox' provider...
$ vagrant up sblldb
sblldb: Matching MAC address for NAT networking...
sblldb: Setting the name of the VM: BOXES_sblldb_1554029293401_13617
Vagrant is currently configured to create VirtualBox synced folders with the 'SharedFoldersEnableSymlinksCreate' option enabled. If the Vagrant guest is not trusted, you may want to disable this option. For more information on this option, please refer to the VirtualBox manual.

https://www.virtualbox.org/manual/ch04.html#sharefolders

This option can be disabled globally with an environment variable:
$ VAGRANT_DISABLE_VRODSYMLINKCREATE=1

on a per folder basis within the Vagrantfile:
config.vm.synced_folder '/host/path', '/guest/path', SharedFoldersEnableSymlinksCreate=false
sblldb: Clearing any previously set network interfaces...
sblldb: Preparing network interfaces based on configuration...
sblldb: Adapter 1: hostonly
sblldb: Adapter 2: hostonly
sblldb: Forwarding ports...
sblldb: 22 (guest) => 2200 (host) (adapter 1)
sblldb: 1521 (guest) => 1521 (host) (adapter 1)
sblldb: Booting VM...
There was an error while executing 'VBoxManage', a CLI used by Vagrant for controlling VirtualBox. The command and stderr is shown below.

Command: ["startvm", "3bcf2867-72d8-4a33-bdd5-e1eeb36d494d", "--type", "headless"]

Stderr: VBoxManage.exe: error: The native API dll was not found (C:\Windows\system32\WinHVPlatform.dll) (VERR_NEM_NOT_AVAILABLE).
VBoxManage.exe: error: VT-x is not available (VERR_VMX_NO_VMX)
VBoxManage.exe: error: Details: code E_FAIL (0x80004005), component ConsoleWrap, interface IConsole

```

### Solution:

To verify the checksum of the downloaded box:

```
$ vagrant box add --name ol73 --checksum e2bbb85d6d54a1221e4bb2ff07100941 \
--checksum-type md5 http://yum.oracle.com/boxes/oraclelinux/ol73/ol73.box
```

### Vagrantfile syntax linxs:

```

https://gist.github.com/bvansomeren/7500194
https://www.vagrantup.com/docs/virtualbox/configuration.html
https://www.vagrantup.com/docs/cli/
https://devhints.io/vagrantfile
[Oracle vagrant boxes site link below]
http://yum.oracle.com/boxes/oraclelinux

```

### Vagrantfile Cheat Sheets:

Typing vagrant from the command line will display a list of all available commands.

Be sure that you are in the same directory as the Vagrantfile when running these commands!

#Creating a VM

```

vagrant init -- Initialize Vagrant with a Vagrantfile and ./vagrant directory, using no specified base image. Before you can do vagrant up, you'll need to specify a base image in the Vagrantfile.
vagrant init <boxpath> -- Initialize Vagrant with a specific box. To find a box, go to the public Vagrant box catalog. When you find one you like, just replace it's name with boxpath. For example, vagrant init ubuntu/trusty64.

```

#Starting a VM

```

vagrant up -- starts vagrant environment (also provisions only on the FIRST vagrant up)
vagrant resume -- resume a suspended machine (vagrant up works just fine for this as well)
vagrant provision -- forces reprovisioning of the vagrant machine
vagrant reload -- restarts vagrant machine, loads new Vagrantfile configuration
vagrant reload --provision -- restart the virtual machine and force provisioning

```

## #Getting into a VM

```
vagrant ssh -- connects to machine via SSH  
vagrant ssh <boxname> -- If you give your box a name in your Vagrantfile, you can ssh into it with boxname.  
Works from any directory.
```

## #Stopping a VM

```
vagrant halt -- stops the vagrant machine  
vagrant suspend -- suspends a virtual machine (remembers state) to save the virtual machine  
vagrant resume -- to reopen the saved virtual machine
```

## #Cleaning Up a VM

```
vagrant destroy -- stops and deletes all traces of the vagrant machine  
'vagrant destroy -f' -- same as above, without confirmation
```

## #About Boxes

```
vagrant box list -- see a list of all installed boxes on your computer  
vagrant box add <name> <url> -- download a box image to your computer  
vagrant box outdated -- check for updates vagrant box update  
vagrant boxes remove <name> -- deletes a box from the machine  
vagrant package -- packages a running virtualbox env in a reusable box
```

## #Saving Progress

```
--vagrant snapshot save [options] [vm-name] <name> -- vm-name is often default. Allows us to save so that we can  
rollback at a later time
```

## #Tips

```
vagrant -v -- get the vagrant version  
vagrant status -- outputs status of the vagrant machine  
vagrant global-status -- outputs status of all vagrant machines  
vagrant global-status --prune -- same as above, but prunes invalid entries  
vagrant provision --debug -- use the debug flag to increase the verbosity of the output  
vagrant push -- yes, vagrant can be configured to deploy code!  
vagrant up --provision | tee provision.log -- Runs vagrant up, forces provisioning and logs all output to a  
file
```

## #Plugins

```
vagrant-hostsupdater : $ vagrant plugin install vagrant-hostsupdater to update your /etc/hosts file  
automatically each time you start/stop your vagrant box.
```

## #Notes

If you are using VVV, you can enable xdebug by running vagrant ssh and then xdebug\_on from the virtual machine's CLI.  
=====

unable to ssh from mobaxter/putty of vagrant vm's guest  
error:  
Disconnected: No supported authentication methods available (server sent: publickey,gssapi-keyex,gssapi-with-mic)  
solution:  
yum install openssh-server  
Now you try to Normal login in your system.  
Then edit to using vi /etc/ssh/sshd\_config key press town now you check the PasswordAuthentication yes save and restart the service sshd.  
PasswordAuthentication no  
  
to  
  
PasswordAuthentication yes  
  
And then restart the sshd daemon.  
  
service sshd restart  
  
above worked fine, but getting access denied with vagrant user.  
  
login as: vagrant  
Access denied  
vagrant@127.0.0.1's password:  
Access denied  
vagrant@127.0.0.1's password:  
=====

```

# Get CLI help with
vagrant list-commands

# List boxes and versions
vagrant box list

# Setup a new instance
vagrant box add <name> <box path/HTTP URI>
vagrant init <name>
vagrant up
vagrant halt
vagrant box remove <name> virtualbox

vagrant box outdated      # Check for updates
vagrant box update ...    # Add newest version

# Login
vagrant ssh <name>

# Check global status
vagrant global-status
=====
General
Description      Command
List Vagrant VMs      vagrant status
Base Box Commands
Description
          Command
List local base boxes
      vagrant box list
Remove local base box
      vagrant box remove {Name}
Remove a specific version of a local base box
remove {Name} -box-version {value}
Remove local base box for a specific provider
remove {Name} --provider {value}
Export a VM as a new base box
      vagrant package --base {VM name} --output {file name}.box
Update a box from a local package
box update "./{file name}.box"
#Repackage a local box
Repackage a local box into the current directory (e.g. edistribute it)      vagrant box repack {Name}
{Provider} {Version}

#To check the ports are opened or not for vagrant boxes
Vagrant port <box name>
Example : vagrant port siebel

# Vagrant memory

config.vm.provider :virtualbox do |v|
  v.customize ["modifyvm", :id, "--memory", 2048]
end

# To set virtual machine memory size use:
config.vm.customize ["modifyvm", :id, "--memory", 2048]
# To set virtual machine CPU's count use:
config.vm.customize ["modifyvm", :id, "--cpus", 2]
# or to set half of available CPU's count:
config.vm.customize ["modifyvm", :id, "--cpus", `awk "/^processor/ {++n} END {print n}" /proc/cpuinfo 2> /dev/null || sh -c 'sysctl hw.logicalcpu 2> /dev/null || echo ": 2"' | awk '{print \$2}'`].chomp ]

config.vm.provider "virtualbox" do |vb|
  vb.memory = "2048"
  vb.cpus = "2"
end

```

## # Vagrant up getting SSH auth method | authentication failure ERROR:

```
ubuntu: SSH auth method: private key
BRM: Warning: Authentication failure. Retrying...
BRM: Warning: Authentication failure. Retrying...
ubuntu: Warning: Connection reset. Retrying...
ubuntu: Warning: Remote connection disconnect. Retrying...
ubuntu: Warning: Connection aborted. Retrying...
ubuntu: Warning: Remote connection disconnect. Retrying...
ubuntu: Warning: Connection aborted. Retrying...
ubuntu: Warning: Remote connection disconnect. Retrying...
ubuntu: Warning: Connection reset. Retrying...
ubuntu: Warning: Remote connection disconnect. Retrying...
ubuntu: Warning: Connection aborted. Retrying...
ubuntu: Warning: Remote connection disconnect. Retrying...
ubuntu: Warning: Connection aborted. Retrying...
ubuntu: Warning: Remote connection disconnect. Retrying...
ubuntu: Warning: Connection aborted. Retrying...
ubuntu: Warning: Remote connection disconnect. Retrying...
BRM: Warning: Authentication failure. Retrying...
```

### Solution1 :

Run below 2 commands and try.

```
[1] vagrant plugin list  
[2] vagrant plugin update
```

After updating the below values in vagrantfile, this didn't repeated.

```
config.ssh.username = "vagrant"  
config.ssh.password = "vagrant"
```

### Solution2 :

I solved my problem: option `config.ssh.insert\_key = false` in both Vagrantfiles.

### Solution3 :

The main thing which fixed it for me was adding the `vagrant.pub` to the `authorized\_keys` with

```
wget https://raw.githubusercontent.com/mitchellh/vagrant/master/keys/vagrant.pub -O .ssh/authorized_keys  
chmod 700 .ssh  
chmod 600 .ssh/authorized_keys  
chown -R vagrant:vagrant .ssh  
[OR]
```

```
wget --no-check-certificate  
https://raw.github.com/mitchellh/vagrant/master/keys/vagrant.pub -O  
/home/vagrant/.ssh/authorized_keys  
chmod 0700 /home/vagrant/.ssh  
chmod 0600 /home/vagrant/.ssh/authorized_keys  
chown -R vagrant /home/vagrant/.ssh
```

Exit from the guest machine and restart it ([vagrant reload](#))

### Solution4 :

```
config.ssh.username = "vagrant"  
  config.ssh.password = "vagrant"  
vagrant up BRM  
now the ssh authentication issue resolved.
```

### Solution5 :

As @rfay suggested I added `ALL=(ALL) NOPASSWD:ALL Defaults:vagrant !requiretty` to `etc/sudoers.d/README` and it worked just fine

```
***** worked *****
```

[# Warning: Remote connection disconnect. Retrying...](#)

### Solution1 :

Use `config.ssh.insert_key = false` in your Vagrantfile and then try.

Reload and check

Delete the file: C:\Users\UserName\\.vagrant.d\insecure\_private\_key

| Name                        | Date modified     | Type        | Size |
|-----------------------------|-------------------|-------------|------|
| boxes                       | 2/5/2019 1:13 PM  | File folder |      |
| data                        | 2/5/2019 8:05 PM  | File folder |      |
| gems                        | 8/31/2018 3:32 AM | File folder |      |
| rgloader                    | 8/1/2018 12:51 AM | File folder |      |
| tmp                         | 2/5/2019 1:35 PM  | File folder |      |
| <b>insecure_private_key</b> | 8/1/2018 12:51 AM | File        | 2 KB |
| plugins.json                | 2/5/2019 6:54 PM  | JSON File   | 1 KB |
| setup_version               | 8/1/2018 12:51 AM | File        | 1 KB |

Then run: `vagrant up`

*Issue : The following SSH command responded with a non-zero exit status.*

Solution1 :

Vagrant login: <https://app.vagrantup.com/>

For the cloud publish boxes.

My credentials: [gowseshaik@gmail.com](mailto:gowseshaik@gmail.com)/[Vagrant@1122](mailto:Vagrant@1122)

#### Resize a Hard Disk for a Virtual Machine

Our Virtual Machines are provisioned using Vagrant from a Linux base box to run using VirutalBox. If the Hard Disk space runs out and you cannot remove files to free-up space, you can resize the Hard Disk using some VirtualBox and Linux commands.

Some assumptions

The following steps assume you've got a set-up like mine, where:

you use a Cygwin or Linux command-line terminal on your host machine

the VirtualBox install path is in your Windows (and therefore Cygwin bash) PATH environment variable

the vagrant boxes live at the path provisioning/boxes/mybox

your Cygwin HOME path is the same as your Windows %USERPROFILE% (see How do I change my Cygwin HOME folder after installation)

VirtualBox creates new Virtual Machines in the default location ~/VirtualBox\ VMs/

Steps to resize the hard disk

Stop the virtual machine using Vagrant.

```
# cd provisioning/boxes/mybox
# vagrant halt
```

Locate the VirtuaBox VM and the HDD attached to its SATA Controller. In this instance we're assuming the VM is located in the default location and is named mybox\_default\_1382400620.

```
# cd ~/VirtualBox\ VMs/mybox default 1382400620
# VBoxManage showvminfo mybox default 1382400620 | grep ".vmdk"
```

The showvminfo command should show you the location on the file-system of the HDD of type VMDK along with the name of the Controller it is attached to - it will look something like this:

```
SATA Controller (0, 0): C:\Users\user.name\VirtualBox VMs\mybox_default_1382400620\box-disk1.vmdk (UUID: 2f79610e-6c06-46d5-becb-448386ea40ec)
```

clone the VMDK type disk to a VDI type disk so it can be resized.

```
# cd ~/VirtualBox\ VMs/mybox default 1382400620
# VBoxManage clonehd "box-disk1.vmdk" "clone-disk1.vdi" --format vdi
```

*NOTE: We do this because VMDK type disks cannot be resized by VirtualBox. It has the added benefit of allowing us to keep our original disk backed-up during the resize operation.*

Find out how big the disk is currently, to determine how large to make it when resized. The information will show the current size and the Format variant. If Dynamic Allocation was used to create the disk, the Format variant will be "dynamic default".

```
# VBoxManage showhdinfo "clone-disk1.vdi"
```

Resize the cloned disk to give it more space. The size argument below is given in Megabytes (1024 Bytes = 1 Megabyte). Because this disk was created using dynamic allocation I'm going to resize it to 100 Gigabytes.

```
# VBoxManage modifyhd "clone-disk1.vdi" --resize 102400
```

*NOTE: If the disk was created using dynamic allocation (see previous step) then the physical size of the disk will not need to match its logical size - meaning you can create a very large logical disk that will increase in physical size only as space is used.*

*TIP: To convert a Gigabyte value into Megabytes use an online calculator.*

Find out the name of the Storage Controller to attach the newly resized disk to.

```
# VBoxManage showvminfo mybox default 1382400620 | grep "Storage"
```

Attach the newly resized disk to the Storage Controller of the Virtual Machine. In our case we're going to use the same name for the Storage Controller, SATA Controller, as revealed in the step above.

```

# VBoxManage storageattach mybox default 1382400620 --storagectl "SATA Controller" --port 0 --device 0 --type
hdd --medium clone-disk1.vdi
Reboot the Virtual Machine using Vagrant.
# cd provisioning/boxes/mybox
# vagrant up
Open a command-line shell as root on the Virtual Machine via ssh.
# vagrant ssh
# sudo su -
Find the name of the logical volume mapping the file-system is on (ie. /dev/mapper/VolGroupOS-lv_root).
# df
Find the name of the physical volume (or device) that all the partitions are created on (ie. /dev/sda).
# fdisk -l
Create a new primary partition for use as a Linux LVM
# fdisk /dev/sda
Press p to print the partition table to identify the number of partitions. By default there are two
- sda1 and sda2.
Press n to create a new primary partition.
Press p for primary.
Press 3 for the partition number, depending the output of the partition table print.
Press Enter two times to accept the default First and Last cylinder.
Press t to change the system's partition ID
Press 3 to select the newly creation partition
Type 8e to change the Hex Code of the partition for Linux LVM
Press w to write the changes to the partition table.
Reboot the machine, then ssh back in when it is up again and switch to the root user once more.
# reboot
# vagrant ssh
# sudo su -
Create a new physical volume using the new primary partition just created.
# pvcreate /dev/sda3
Find out the name of the Volume Group that the Logical Volume mapping belongs to (ie. VolGroupOS).
# vgdisplay
Extend the Volume Group to use the newly created physical volume.
# vgextend VolGroupOS /dev/sda3
Extend the logical volume to use more of the Volume Group size now available to it. You can either tell it to
add a set amount of space in Megabytes, Gigabytes or Terabytes, and control the growth of the Disk:
# lvextend -L+20G /dev/mapper/VolGroupOS-lv_root
Or if you want to use all the free space now available to the Volume Group:
# lvextend -l +100%FREE /dev/mapper/VolGroupOS-lv_root
Resize the file-system to use up the space made available in the Logical Volume
# resize2fs /dev/mapper/VolGroupOS-lv_root
Verifiy that there is now more space available
# df -h
A restart of the VM using vagrant may be a good idea here, to ensure that all services are running correctly now
that there is more space available. Exit the root user, exit the vagrant user and ssh session, then tell vagrant
to restart the machine.
# exit
# exit
# vagrant reload --provision

```

## What is the difference between Packer and Terraform

**Packer** : is used to create for images

**Terraform** : is used to create for virtual environments.

## Creating Boxes with Packer : is for images

<https://www.packer.io/intro/getting-started/install.html#precompiled-binaries>

Using Packer requires more up front effort, but the repeatable and automated builds will end any manual management of boxes. Additionally, all boxes will be stored and served from Vagrant Cloud, keeping a history along the way.

### »Creating Boxes via the Vagrant Cloud Web Interface

You'll first need to create a box file. This can be done via the vagrant package command or with Packer locally. After you've created the .box file, this guide can be followed.

1. Go to the Create Box page.
  2. Name the box and give it a simple description
  3. Create your first version for the box. This version must match the format [0-9].[0-9].[0-9]
  4. Create a provider for the box, matching the provider you need locally in Vagrant. virtualbox is the most common provider.
  5. Upload the .box file for each provider, or use a url to the .box file that is publicly accessible
- You can find all of your boxes in the Vagrant section of Vagrant Cloud.

Once you've created and released a box, you can release new versions of the box by clicking "Create New Version" under the versions sidebar on a box page. For more information on the release lifecycle of boxes, see the help page dedicated to box lifecycle.

## »Creating Boxes with the API

This example uses the API to upload boxes with curl. To get started, you'll need to get an access token. Then, prepare the upload:

```
$ curl 'https://vagrantcloud.com/api/v1/box/USERNAME/BOX_NAME/version/VERSION/provider/PROVIDER_NAME/upload?access_token=ACCESS_TOKEN'
```

This should return something like this:

```
{
  "upload path": "https://archivist.hashicorp.com/v1/object/630e42d9-2364-2412-4121-18266770468e"
}
```

Then, upload your box with the following command, with the filename in this case being foo.box:

```
$ curl -X PUT --upload-file foo.box https://archivist.hashicorp.com/v1/object/630e42d9-2364-2412-4121-18266770468e
```

When the upload finishes, you can verify it worked by making this request and matching the hosted\_token it returns to the previously retrieved upload token.

```
$ curl 'https://vagrantcloud.com/api/v1/box/USERNAME/BOX_NAME/version/VERSION_NUMBER/provider/PROVIDER_NAME?access_token=ACCESS_TOKEN'
```

Your box should then be available for download.

## Unix Commands

Command check the memory usage

<https://www.binarytides.com/linux-command-check-memory-usage/>

```
$ free -m
```

|                    | total | used | free | shared | buffers | cached |
|--------------------|-------|------|------|--------|---------|--------|
| Mem:               | 7976  | 6459 | 1517 | 0      | 865     | 2248   |
| -/+ buffers/cache: |       | 3344 | 4631 |        |         |        |
| Swap:              | 1951  | 0    | 1951 |        |         |        |

The m option displays all data in MBs. The total os 7976 MB is the total amount of RAM installed on the system, that is 8GB. The used column shows the amount of RAM that has been used by linux, in this case around 6.4 GB. The output is pretty self explanatory. The catch over here is the cached and buffers column. The second line tells that 4.6 GB is free. This is the free memory in first line added with the buffers and cached amount of memory.

Linux has the habit of caching lots of things for faster performance, so that memory can be freed and used if needed.

The last line is the swap memory, which in this case is lying entirely free.

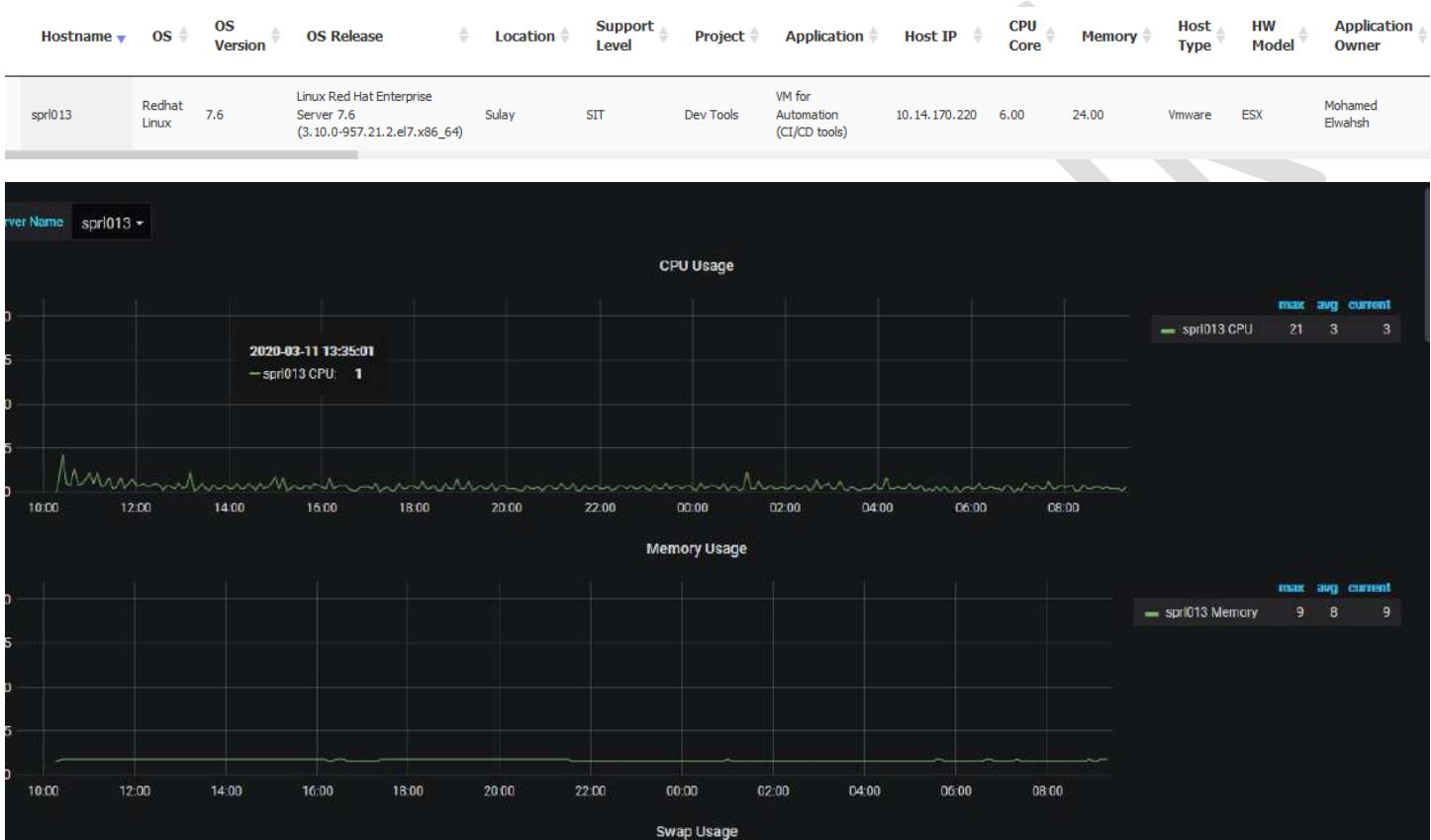
## How to display information about the CPU on Linux

Just run the lscpu command:

```
$ lscpu  
$ lscpu | egrep 'Model name|Socket|Thread|NUMA|CPU\s+'  
$ lscpu -p
```

<https://www.cyberciti.biz/faq/check-how-many-cpus-are-there-in-linux-system/>

example from memory snapshot:



## Docker [For MicroServices]

If you mean to install docker engine on unix server I think docker server is supported for windows, linux and mac computers.

I think, right now it's not possible docker doesn't support unix.



Detailed book for docker

and from <https://riptutorial.com>

To install Docker, we need to follow the steps given below.

For linux , step followed as per link.

[https://www.tutorialspoint.com/docker/installing\\_docker\\_on\\_linux.htm](https://www.tutorialspoint.com/docker/installing_docker_on_linux.htm)

### Step 1 : kernal check

- Before installing Docker, you first have to ensure that you have the right Linux kernel version running. Docker is only designed to run on Linux kernel version 3.8 and higher. We can do this by running the following command.

**uname**

This method returns the system information about the Linux system.

**Syntax**

```
uname -a
```

**Options**

**a** - This is used to ensure that the system information is returned.

**Return Value**

This method returns the following information on the Linux system -

kernel name  
node name  
kernel release  
kernel version  
machine  
processor  
hardware platform  
operating system

**Example**

```
uname -a
```

**Output**

When we run above command, we will get the following result -

```
demo@ubuntu:~$ uname -a
Linux ubuntu 4.2.0-27-generic #32~14.04.1-Ubuntu SMP Fri Jan 22 15:32:27 UTC 2016
i686 i686 i686 GNU/Linux
demo@ubuntu:~$ -
```

From the output, we can see that the Linux kernel version is 4.2.0-27 which is higher than version 3.8, so we are good to go.

**Step 2 :** yum update

go for apt-get update (or) yum update

**Step 3 :** install certificates for docker to install packages

The next step is to install the necessary certificates that will be required to work with the Docker site later on to download the necessary Docker packages. It can be done with the following command.

```
sudo apt-get install apt-transport-https ca-certificates
yum install apt-transport-https ca-certificates
```

**Step 4 :** add the new GPG key.

The next step is to add the new GPG key. This key is required to ensure that all data is encrypted when downloading the necessary packages for Docker.

The following command will download the key with the ID 58118E89F3A912897C070ADBF76221572C52609D from the **keyserver** `hkp://ha.pool.sks-keyservers.net:80` and adds it to the **adv** keychain. Please note that this particular key is required to download the necessary Docker packages.

## Docker and oracle : The Policy[Certified]

- Oracle support is limited to the version of Docker built by and downloaded from Oracle. Support is not provided for the upstream binary packages downloaded from the upstream Docker site.
- Oracle support for Docker is limited to customers with either Basic or Premier Oracle Linux Support Subscriptions.
- Oracle only provides Docker server binary packages for Oracle Linux 6 and Oracle Linux 7 (x86\_64).
- Oracle support is limited to severity 2 service requests and lower. Severity 1 service requests will not be accepted for Docker.
- Support is limited to assistance with the initial installation of Docker and subsequent configuration of Docker to enable downloading, creation and running of Dockerized applications.

- Oracle does not support Docker running on the Red Hat Compatible Kernel (RHCK). Because Docker requires newer features in more recent Linux kernel versions, Oracle only provides support for Docker with the Unbreakable Enterprise Kernel Release 4 (UEK4), identified by a kernel version of "4.1" or newer.
- Oracle Linux support does not include support for any Oracle or 3rd-party applications running within a Docker container on Oracle Linux.
- Oracle will periodically release updates from the upstream project in the addons channel. Interim patching and bug fixing will not be available.
- Support for Docker is provided for the latest available release on the Unbreakable Linux Network or Oracle Linux Yum Server (<http://yum.oracle.com/>). Customers are encouraged to upgrade to the most recent version before opening a service request.

To install docker on ubuntu:

## OS requirements for docker on ubuntu

To install Docker CE, you need the 64-bit version of one of these Ubuntu versions:

Cosmic 18.10

Bionic 18.04 (LTS)

Xenial 16.04 (LTS)

Docker CE is supported on x86\_64 (or amd64), armhf, arm64, s390x (IBM Z), and ppc64le (IBM Power) architectures.

```
$ sudo apt-get remove docker docker-engine docker.io containerd runc
$ apt-get update
```

```
$ apt-get install \
  apt-transport-https \
  ca-certificates \
  curl \
  gnupg-agent \
  software-properties-common
```

```
root@vagrant:/home/vagrant# curl -fsSL https://download.docker.com/linux/ubuntu/gpg | sudo apt-key add -
root@vagrant:/home/vagrant# add-apt-repository "deb [arch=amd64] https://download.docker.com/linux/ubuntu
$(lsb_release -cs) stable"
root@vagrant:/home/vagrant# apt-get update
root@vagrant:/home/vagrant# apt-cache policy docker-ce
root@vagrant:/home/vagrant# apt-get install -y docker-ce
root@vagrant:/home/vagrant# systemctl status docker
root@vagrant:/home/vagrant# docker version
to remove the docker from ubuntu:
root@vagrant:/home/vagrant# apt-get remove docker-ce
```

**Docker info :WARNING: No swap limit support**

Solution:

You can enable these capabilities on Ubuntu or Debian by following these instructions. Memory and swap accounting incur an overhead of about 1% of the total available memory and a 10% overall performance degradation, even if Docker is not running.

- 1) Log into the Ubuntu or Debian host as a user with sudo privileges.
- 2) Edit the /etc/default/grub file. Add or edit the GRUB\_CMDLINE\_LINUX line to add the following two key-value pairs:  
GRUB\_CMDLINE\_LINUX="cgroup\_enable=memory swapaccount=1"
- 3) Update GRUB.  
\$ sudo update-grub

OR

1. edit the file /etc/default/grub.d/50-cloudimg-settings.cfg and edit the line to  
GRUB\_CMDLINE\_LINUX\_DEFAULT="console=ttyS0 cgroup\_enable=memory swapaccount=1"  
sudo nano /etc/default/grub.d/50-cloudimg-settings.cfg
2. update grub  
sudo update-grub
3. reboot  
sudo reboot

### Steps to Install Docker on Oracle Linux

<https://oracle-base.com/articles/linux/docker-install-docker-on-oracle-linux-ol7>

<https://blogs.oracle.com/blogypuneeth/a-simple-guide-to-docker-installation-on-oracle-linux-75>

<https://blogs.oracle.com/virtualization/install-docker-on-oracle-linux-7-v2>

First, update your Oracle Linux 7 instance with correct yum setting. To install the latest Docker release (17.03.1.ce), you need ol7\_latest, ol7\_uekr4 and ol7\_addons enabled.

*Step 1 : get the repo in os.*

```
# cd /etc/yum.repos.d/  
# wget http://yum.oracle.com/public-yum-ol7.repo  
# vi public-yum-ol7.repo  
  
[ol7_latest]  
name=Oracle Linux $releasever Latest ($basearch)  
baseurl=http://yum.oracle.com/repo/OracleLinux/OL7/latest/$basearch/  
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle  
gpgcheck=1  
enabled=1  
  
[ol7_UEKR4]  
name=Latest Unbreakable Enterprise Kernel Release 4 for Oracle Linux $releasever ($basearch)  
baseurl=http://yum.oracle.com/repo/OracleLinux/OL7/UEKR4/$basearch/  
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle  
gpgcheck=1  
enabled=1  
  
[ol7_addons]  
name=Oracle Linux $releasever Add ons ($basearch)  
baseurl=http://yum.oracle.com/repo/OracleLinux/OL7/addons/$basearch/  
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle  
gpgcheck=1  
enabled=1
```

*Step 2 : install docker-engine*

Next you simply use yum install to start the installation.

```
# yum install docker-engine  
Once the installation is completed, you start the docker service.  
# systemctl start docker  
# systemctl enable docker  
# systemctl status docker
```

*Step 3 : Login to Oracle Container Registry with support web credentials*

Then you log into [Oracle Container Registry](#), trusted source of Oracle software as Docker containers, to pull the appropriate Docker containers.

```
# docker login container-registry.oracle.com  
Username: honglin.su@oracle.com  
Password:  
Login Succeeded
```

*Step 4 : Pull container from Oracle Container Registry with support web credentials*

```
# docker pull container-registry.oracle.com/os/oraclelinux:6.9
# docker pull container-registry.oracle.com/java/serverjre
# docker images

# docker pull container-registry.oracle.com/os/oraclelinux:7-slim

[root@orlinux7 yum.repos.d]# docker pull container-registry.oracle.com/os/oraclelinux:7-slim
Trying to pull repository container-registry.oracle.com/os/oraclelinux ...
Get https://container-registry-phx.oracle.com/v2/os/oraclelinux/manifests/7-slim: unauthorized: authentication required

[root@orlinux7 yum.repos.d]# docker login container-registry.oracle.com
Username: g.shaik.mit@mobilinfotech.com
Password: *****
Login Succeeded

[root@orlinux7 yum.repos.d]# docker pull container-registry.oracle.com/os/oraclelinux:7-slim
Trying to pull repository container-registry.oracle.com/os/oraclelinux ...
7-slim: Pulling from container-registry.oracle.com/os/oraclelinux
f25d82609194: Pull complete
Digest: sha256:50ad4bebb443da56ec7af6e5e53a7316ca518e58c0d1c9ab0cccd9aa2d14a0d
Status: Downloaded newer image for container-registry.oracle.com/os/oraclelinux:7-slim

Step 3 : check for docker images.
[root@orlinux7 yum.repos.d]# docker images
REPOSITORY          TAG      IMAGE ID      CREATED       SIZE
container-registry.oracle.com/os/oraclelinux  7-slim   b19454a5f17a  2 months ago  117MB
```

## Oracle /siebel Docker Images list

Dockerhub official images:

<https://github.com/docker-library/official-images>

docker siebel images:

<https://github.com/OracleSiebel/ConfiguringSiebel>  
<https://hub.docker.com/r/azeemdin/siebelapp>

docker oracle images:

<https://github.com/oracle/docker-images>

setting up with siebel ip15:  
<https://hub.docker.com/r/azeemdin/siebelapp>

To Run the Azeemudin siebelapp 15 container, Please follow the below steps

**DB Container:**

Starting up DB docker run -d --name siebeldb --shm-size=1g -p 8183:1521 azeemdin/siebeldb  
Connect to DB system/dis2004 siebel/siebel sadmin/sadmin

Install License Keys

**Siebel App Container:**

Starting Application docker run -d --hostname=siebelapp --name siebelapp -p 1222:22 -p 7777:7777 --link siebeldb:siebeldb azeemdin/siebelapp:v4 tail -f /dev/null  
hostname is important Port 7777 is very important for OpenUI to work properly.  
Steps to Start Siebel Services docker exec -it siebelapp /bin/bash Login with root/dis2004 service ssh start cd /usr/local/apache2/bin ./apachectl start  
Login with oracle/dis2004 cd /Siebel/15.0.0.0.0/ses/gtwysrvr ./cfgenv.sh ./siebenv.sh cd bin start\_ns cd /Siebel/15.0.0.0.0/ses/siebsrvr ./cfgenv.sh ./siebenv.sh cd bin start\_server all  
wait for some time to start Siebel Services (check by ps -ef)  
[http://\[Server\]:7777/ecomcommunications\\_enu/](http://[Server]:7777/ecomcommunications_enu/)

## linux core concepts for Docker:

cgroups  
iptables  
userspace  
user administration  
container capabilities : is a specialized areas which will give permissions to avoid sudo privileges..

Docker Commands :

<https://docs.docker.com/engine/reference/commandline/docker/>

docker start stop status:

- `systemctl start docker`
- `systemctl stop docker`
- `systemctl status docker`

`docker info`

**info** - It is used to ensure that the Docker command returns the detailed information on the Docker service installed.

It will show the below values.

- Number of containers
- Number of images
- The storage driver used by Docker
- The root directory used by Docker
- The execution driver used by Docker

`docker run image`

**Image** - This is the name of the image which is used to run the container.

This command will download the `ex:centos` image, if it is not already present, and run the OS as a container.

`Docker run -it Ubuntu bash`

Here you are telling Docker to run the command in the interactive mode via the `-it` option.

[Imp] \*\*\* `sudo docker run centos -it /bin/bash`

Note the following points about the above `sudo` command -

- We are using the `sudo` command to ensure that it runs with `root` access.
- Here, `centos` is the name of the image we want to download from Docker Hub and install on our Ubuntu machine.
- `-it` is used to mention that we want to run in `interactive mode`.
- `/bin/bash` is used to run the bash shell once CentOS is up and running.

```
ubuntu@ip-172-31-30-39:~/images/game-of-life$ docker pull tomcat:9.0-jre10
9.0-jre10: Pulling from library/tomcat
e7cb83f94a46: Pull complete
ea75eac4dae2: Pull complete
45c35a0f820b: Pull complete
b9ef45ead295: Pull complete
e8173ebf385f: Pull complete
aa4dbb9733d8: Pull complete
b8fdf937f089: Pull complete
21a815524fd4: Pull complete
1d862e2646b0: Pull complete
8ede20d4c0f2: Extracting 4.588MB/21.93MB
96040cfb2ea4: Download complete
```

Here from above, the every run statement will show as download while executing the run command for your image. If you write 1 run statement then it will download as 1 download statement. If inside that 1 run statement contains 2 more run commands it will download as 3 download statements.

It will be used as layers.  
 Every layer having a unique ID.  
 If you run the same command, it will show already exists, means these run statements already used in earlier images while pulling the image.

```
ubuntu@ip-172-31-30-39:~/images/game-of-life$ docker pull tomcat:9.0-jre8
9.0-jre8: Pulling from library/tomcat
cc1a78bfd46b: Already exists
d2c05365ee2a: Already exists
231cb0e216d3: Already exists
e8912f9d0ce2: Already exists
9bafe362f99b: Already exists
28b0652112a6: Already exists
da23ele20eae: Already exists
9d809d99b239: Already exists
10c61bb6d245: Already exists
50578eb745b9: Already exists
4b1be8d80f68: Pull complete
af8788795371: Pull complete
Digest: sha256:c41fcad410e6de3d04329910688589c82d4038de4e880f890c8ea35e5dfd4835
Status: Downloaded newer image for tomcat:9.0-jre8
ubuntu@ip-172-31-30-39:~/images/game-of-life$
```

So the image layer concept also used in storage part as well.

| REPOSITORY | TAG       | IMAGE ID     | CREATED        |
|------------|-----------|--------------|----------------|
| gol        | 1.1       | ef939ff11a56 | 5 minutes ago  |
| gol        | 1.0       | 429494e95e76 | 24 minutes ago |
| my-first   | latest    | c9d29cbd2918 | 2 days ago     |
| tomcat     | 9.0-jre10 | d498ddf22fb2 | 12 days ago    |
| tomcat     | 9.0-jre8  | 3b76e035b7c6 | 12 days ago    |
| tomcat     | 8.0-jre8  | 4227b4397b18 | 12 days ago    |
| ubuntu     | 16.04     | 5e8b97a2a082 | 2 weeks ago    |
| nginx      | latest    | cd5239a0906a | 2 weeks ago    |
| nginx      | 1.14-perl | f4d46d22d400 | 6 weeks ago    |
| nginx      | 1.14      | f759510436c8 | 6 weeks ago    |

Actual size was might be used as 100MB but it will show the entire 467MB with both versions of images.

Docker hard disk is a optimization on every where with the form as layers.

#### **docker rmi**

This command is used to remove Docker images.

Syntax: docker rmi ImageID  
 Example: sudo docker rmi 7a86f8ffcb25

#### **docker rm ContainerID**

\$sudo docker rm 9f215ed0b0d3  
 The above command will remove the Docker container 9f215ed0b0d3.

```
# docker rm 9f215ed0b0d3
# docker rm -f $(docker ps -a -q)
```

#### **docker images -q**

This command is used to return only the Image ID's of the images.

#### **docker inspect: How To Get Information About a Container In Docker (Doc ID 2149101.1)**

This command is used see the details of an image or container.

Syntax : docker inspect Repository

**Repository** - This is the name of the Image.

Return Value

The output will show detailed information on the Image.

**Example** :sudo docker inspect jenkins

## **docker inspect**

The "inspect" command will list the complete information of the container.

```
[root@OL7 ~]# docker inspect b77494b17772
[root@OL7 ~]# docker inspect --format='{{ .NetworkSettings.IPAddress }}' b77494b17772
172.17.0.2
```

```
sudo docker run -it centos /bin/bash
```

Running a Container ,Then hit Crtl+p and you will return to your OS shell.

## **Docker ps : Listing of Containers which are in running**

This command is used to return the currently running containers.

## **docker ps -a :**

This command is used to list all of the containers on the system

## **docker history ImageID**

**ImageID** - This is the Image ID for which you want to see all the commands that were run against it.  
Return Value : The output will show all the commands run against that image.

## **docker top ContainerID**

**ContainerID** - This is the Container ID for which you want to see the top processes.  
Return Value : The output will show the top-level processes within a container.

Example : sudo docker top 9f215ed0b0d3

## **docker stop ContainerID**

This command is used to stop a running container.

Examples: sudo docker stop 9f215ed0b0d3

## **docker stats ContainerID**

The output will show the CPU and Memory utilization of the Container.

Example: sudo docker stats 9f215ed0b0d3

## **docker attach**

This command is used to attach to a running container. When we run the above command, it will produce the following result -

```
demo@ubuntuserver:~$ sudo docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              NAMES
STATUS              PORTS               NAMES
07b0b6f434fe        centos:latest      "/bin/bash"        3 minutes ago     Up 3 minutes       cocky_pare
demo@ubuntuserver:~$ sudo docker attach 07b0b6f434fe
[root@07b0b6f434fe ~]#
```

Once you have attached to the Docker container, you can run the above command to see the process utilization in that Docker container.

```
top - 15:24:06 up 2:06, 0 users, load average: 0.00, 0.01, 0.02
Tasks: 2 total, 1 running, 1 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.3 sy, 0.0 ni, 99.7 id, 0.0 wa, 0.0 hi, 0.0 si, 0.
KiB Mem : 1484856 total, 1057152 free, 52368 used, 375336 buff/cache
KiB Swap: 1519612 total, 1519612 free, 0 used. 1403060 avail Mem
```

| PID | USER | PR | NI | VIRT  | RES  | SHR  | S | %CPU | %MEM | TIME+   | COMMAND |
|-----|------|----|----|-------|------|------|---|------|------|---------|---------|
| 1   | root | 20 | 0  | 11784 | 2992 | 2644 | S | 0.0  | 0.2  | 0:00.01 | bash    |
| 15  | root | 20 | 0  | 51864 | 3772 | 3272 | R | 0.0  | 0.3  | 0:00.00 | top     |

## **docker pause ContainerID**

**ContainerID** - This is the Container ID to which you need to pause the processes in the container.

Example : sudo docker pause 07b0b6f434fe

The above command will pause the processes in a running container **07b0b6f434fe**.

Output :When we run the above command, it will produce the following result -

```

demo@ubuntuserver:~$ sudo docker ps
[sudo] password for demo:
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              NAMES
07b0b6f434fe        centos:latest      "/bin/bash"        18 minutes ago   Up 18 minutes     cocky_pare
demo@ubuntuserver:~$ sudo docker pause 07b0b6f434fe
07b0b6f434fe
demo@ubuntuserver:~$ sudo docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              NAMES
07b0b6f434fe        centos:latest      "/bin/bash"        19 minutes ago   Up 19 minutes (Paused)   cocky_pare
demo@ubuntuserver:~$ 

```

#### *docker unpause ContainerID*

**ContainerID** - This is the Container ID to which you need to unpause the processes in the container.  
When we run the above command, it will produce the following result -

```

demo@ubuntuserver:~$ sudo docker unpause 07b0b6f434fe
07b0b6f434fe
demo@ubuntuserver:~$ 

```

#### *docker kill ContainerID*

**ContainerID** - This is the Container ID to which you need to kill the processes in the container.  
sudo docker kill 07b0b6f434fe  
Output :When we run the above command, it will produce the following result

```

demo@ubuntuserver:~$ sudo docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              NAMES
07b0b6f434fe        centos:latest      "/bin/bash"        23 minutes ago   Up 23 minutes     cocky_pare
demo@ubuntuserver:~$ sudo docker kill 07b0b6f434fe
07b0b6f434fe
demo@ubuntuserver:~$ 

```

#### *Docker container backup*

1. Commit the required container as an image  
`# docker commit -p <container-id> backup01`  
sha256:89682d4xxxxxx

**Now a new image backup01 will be created. Kindly note that this will not cover the data volume. You need take the backup of data-volume (if any) separately.**

To know this data-directory (data volume location) of a container, use the command 'docker inspect container-name'.

You will get a section called "Mounts". Location mentioned in "Source" is the data volume. You may directly backup this folder(here /site) to get backup of data volume.

```

"Mounts": [
{
"Source": "/site",
"Destination": "/usr/xx/xxx/xxx",
"Mode": "",
"RW": true,
"Propagation": "rprivate"

```

2. You can save the image backup01 to tar file using the following command:

```

#docker save -o backup01.tar backup01
# ls -al |grep back
-rw----- 1 root root 178697728 Aug 31 23:35 backup01.tar

```

You may choose to save the tar file on NFS mount point.

Another option is directly pushing the image backup01 to your local registry:

```

# docker tag backup01 localhost:5000/backup-image:v1

```

In this example, localhost is the hostname where the local registry is located and 5000 is the port number that the registry listens on. If you are working on a Docker engine located on a different host to the registry, you must change the hostname to point to the correct host. Note the repository and tag name, backup-image:v1 in the example, must all be in lower case to be a valid tag.

```

# docker push backup-image:v1

```

#### Steps to restore:

```
1. Image can be extracted from backup tar file using the following command  
# docker load -i /tmp/backup01.tar  
ff91b8b5abb1: Loading layer [=====>] 2.56 kB/2.56 kB  
Loaded image: backup01:latest  
You can create container from this image using "docker run"  
If you had data volume on the original container. You must restore the data volume too and run container with  
the data volume (docker run -v)  
2. In-case of pushed image. You can directly pull it.  
#docker pull localhost:5000/backup-image:v1
```

## Dockerfile

**Dockerfiles** use a simple DSL which allows you to automate the steps you would normally manually take to create an image.

#### Dockerfile to run:

```
ubuntu@ip-172-31-30-39: ~/images/game-of-life  
FROM tomcat:8.0-jre8  
MAINTAINER qualitythought  
ADD gameoflife.war /usr/local/tomcat/webapps/gameoflife.war  
EXPOSE 8080  
CMD ["catalina.sh", "run"]  
~
```

#### Difference between COPY and ADD in Dockerfile

**COPY** : is to copy files from source to destination. With COPY you cannot download files to destination.

**ADD** : is to copy/download files source(url/link/path) to destination

```
ubuntu@ip-172-31-30-39: ~/images/game-of-life  
FROM tomcat:8.0-jre8  
MAINTAINER qualitythought  
ADD gameoflife.war /usr/local/tomcat/webapps/gameoflife.war  
~
```

#### Docker build an image

```
#Cd /images/  
#Vi dockerfile  
FROM tomcat:8.0-jre8  
MAINTAINER techcircle  
ADD gameoflife.war /user/local/tomcat/webapps/gameoflife.war  
EXPOSE 8080  
CMD ["catalina.sh", "run"]  
  
ENTRYPOINT ["echo"]  
CMD["my-first-image"]
```

```
ubuntu@ip-172-31-30-39: ~/images/my-hello-world  
ubuntu@ip-172-31-30-39:~/images/my-hello-world$ docker run my-first echo dont  
dont  
ubuntu@ip-172-31-30-39:~/images/my-hello-world$
```

```
FROM nginx  
MAINTAINER khaja  
RUN touch test.txt  
ENTRYPOINT ["echo"]  
CMD ["my-first-image"]
```

Save it

```
#Docker build -t house:1.0 .
```

Now run the container as follows.

```
# docker run -p 8080:8080 --name ghouse house:1.0
```

#### Docker run -d

Run the container in detachment mode, which will run in background mode.

```
# docker run -d -p 8080:8080 --name ghouse house:1.0
```

#### Docker port forwarding

```
# docker run -d -p 8080:8080
```

#### Docker logs

To see the logs

```
# docker logs ghouse
```

#### Create and start a container

```
$ docker create -t -i fedora bash
```

#### Docker execute

```
Docker exec -it ghouse /bin/bash
```

#### Docker exit

To come out the terminal

CTL+pq

#### Docker Volumes

It is to manage data , Created and managed by **Docker**. You can create a **volume** explicitly using the **docker volume** command, or **Docker** can create a **volume** during container or service creation. When you create a **volume**, it is stored within a directory on the **Docker** host.

```
root@vagrant:/var/lib/docker/volumes#
```

#### Argument:

argument is a variable when and where the java path should be setted in your container before starting itself.

**Ex:** Java\_home=/usr/bin/java7

#### Environment Variable

environment variable for when you want to run the values inside the docker container.

#### Docker image creation steps for pingutils

```
azeemdin/siebeldb latest 5adc13783b5c 24 months ago 11.2GB
root@vagrant:~# docker pull ubuntu:16.04
16.04: Pulling from library/ubuntu
7b722c1070cd: Pull complete
5fbf74db61f1: Pull complete
ed41cb72e5c9: Pull complete
7ea47a67709e: Pull complete
Digest: sha256:e4a134999bea4abb4a27bc437e6118fffffb172e1b9d683129b74d254af51675
Status: Downloaded newer image for ubuntu:16.04
root@vagrant:~# docker run -it ubuntu:16.04 /bin/bash
root@iceebb4a4567:/# ping
bash: ping: command not found
try to execute the ping command whether the ping is working
or not, as it is not working
root@iceebb4a4567:/# apt-get install iputils
Reading package lists...
Building dependency tree...
Reading state information... Done
E: Unable to locate package iputils
root@iceebb4a4567:/# apt-get update
0% [Connecting to archive.ubuntu.com] [Connecting to security.ubuntu.com]
```

The screenshot shows a terminal session on a Vagrant box. It starts by pulling the 'ubuntu:16.04' image. Once pulled, a new container is created and started. Inside the container, the user tries to run 'ping', but it fails because the command is not found. To resolve this, the user runs 'apt-get install iputils', which installs the required package. Finally, the user runs 'apt-get update' to ensure all packages are up-to-date.

To automate the above commands with dockerfile

Mkdir images/ubuntupingutils

Vi dockerfile

```
FROM ubuntu:16.04
MAINTAINER gowseshaik@gmail.com
RUN apt-get update && apt-get install iputils-ping -y
CMD ["echo","Image created"]
```

```
root@vagrant:~/images/ubuntuwithping# vi dockerfile
root@vagrant:~/images/ubuntuwithping# docker build -t myubuntu .
```

## DevOpsIssues :

References

<https://github.com/qt-devops>

## Docker Compose:

### Docker compose port mapping

#### Requirement :

I have a `docker-compose.yml` file as in below

```
version: '2'
services:
  nodejs:
    build:
      context: .
      dockerfile: DockerFile
    ports:
      - "4000:4000"
    links:
      - redis
    expose:
      - "6379"
  redis:
    build:
      context: .
      dockerfile: Dockerfile-redis
```

My goal is to forward nodejs-127.0.0.1 port 6379 to the redis host. I can already ping redis from the nodejs machine, but the ports are not mapped. Tried expose options, but no chance either.

#### Solution:

It's important to point out that all of the above solutions map the port to every interface on your machine. This is less than desirable if you have a public IP address, or your machine has an IP on a large network. Your application may be exposed to a much wider audience than you'd hoped.

```
redis:
  build:
    context:
      dockerfile: Dockerfile-redis
    ports:
      - "127.0.0.1:3901:3901"
```

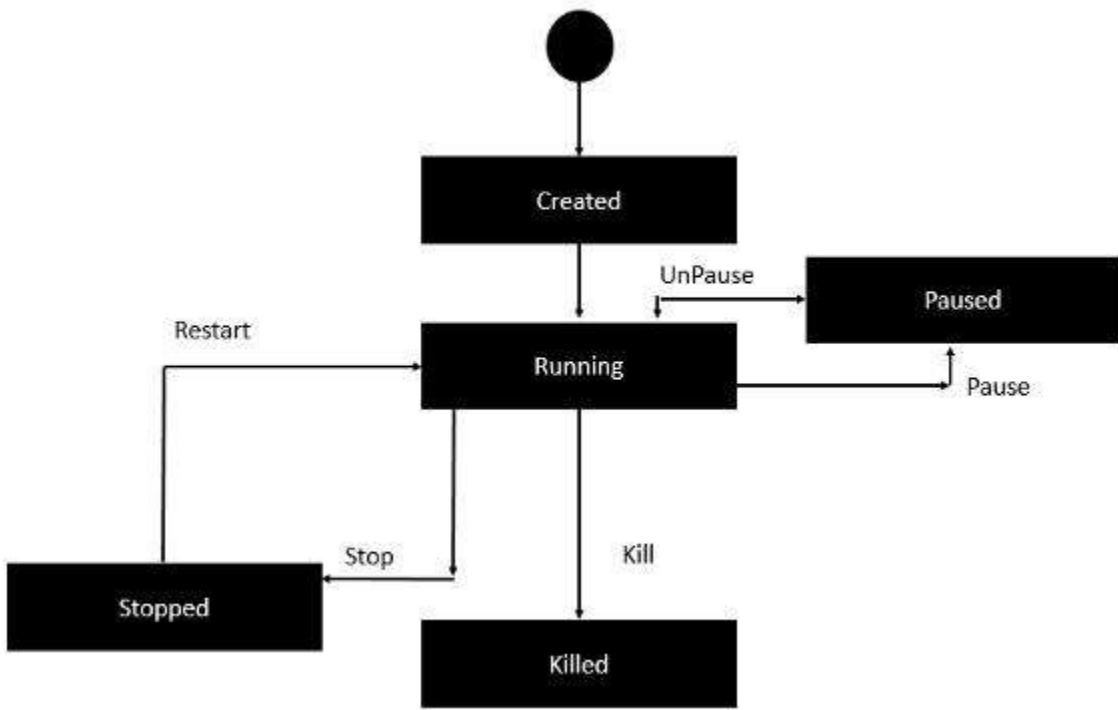
127.0.0.1 is the ip address that maps to the hostname localhost on your machine. So now your application is only exposed over that interface and since 127.0.0.1 is only accessible via your machine, you're not exposing your containers to the entire world.

The documentation explains this further and can be found here: <https://docs.docker.com/compose/compose-file/#ports>

## Docker – Container Lifecycle

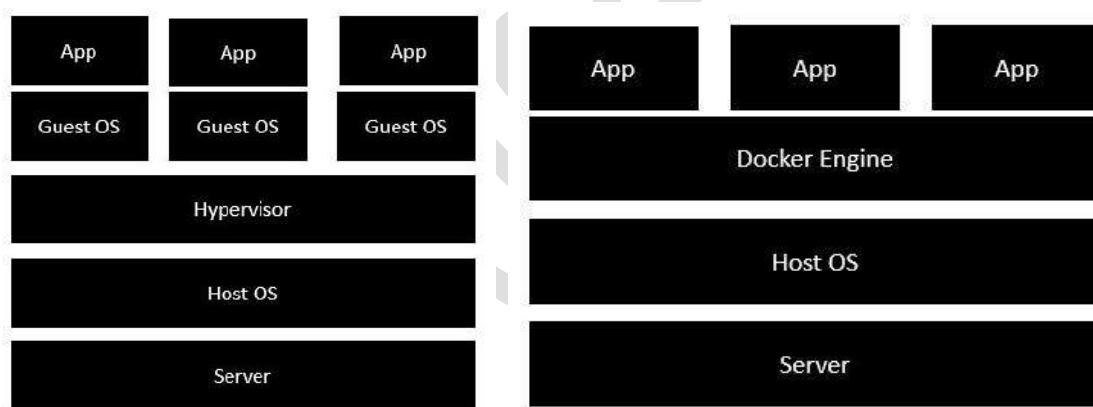
The following illustration explains the entire lifecycle of a Docker container.





- Initially, the Docker container will be in the **created** state.
- Then the Docker container goes into the **running** state when the Docker **run** command is used.
- The Docker **kill** command is used to kill an existing Docker container.
- The Docker **pause** command is used to pause an existing Docker container.
- The Docker **stop** command is used to pause an existing Docker container.
- The Docker **run** command is used to put a container back from a **stopped** state to a **running** state.

#### Difference between Virtualization and containerization architecture



`service docker stop`

This command is used to stop the Docker daemon process.

`service docker start`

This command is used to start the Docker daemon process.

#### Docker compose:

How are the

#### Docker Network

Docker's networking subsystem is pluggable, using drivers. Several drivers exist by default, and provide core networking functionality:

```

root@ip-172-31-30-39:/var/lib/docker/volumes# docker network
Usage: docker network COMMAND

Manage networks

Commands:
  connect      Connect a container to a network
  create       Create a network
  disconnect   Disconnect a container from a network
  inspect      Display detailed information on one or more networks
  ls           List networks
  prune        Remove all unused networks
  rm           Remove one or more networks

Run 'docker network COMMAND --help' for more information on a command.

```

### Docker network ls

```
root@vagrant:~# docker network ls
```

| NETWORK ID   | NAME   | DRIVER | SCOPE |
|--------------|--------|--------|-------|
| 263e51355b99 | bridge | bridge | local |
| adcad077fa88 | host   | host   | local |
| e2171afdd0ed | none   | null   | local |

docker network inspect bridge

```

{
    "Name": "bridge",
    "EndpointID": "601c8f5ec9db3900a7d280de22edc362b206a85b45e51b4882c2ab85e03696",
    "MacAddress": "02:42:ac:11:00:03",
    "IPv4Address": "172.17.0.3/16",
    "IPv6Address": ""
},
"cedd6d69340f3be13906e4908dc07280d52e8d353c882634e9ee521504be2310":
{
    "Name": "c1",
    "EndpointID": "41d719f49a0b5800823a8149244eb2de990ff759900878fa2454123ac28f6b",
    "MacAddress": "02:42:ac:11:00:02",
    "IPv4Address": "172.17.0.2/16",
    "IPv6Address": ""
}

```

Lets ping the containers

```

root@cedd6d69340f:/usr/local/tomcat# ping c2
ping: c2: Name or service not known
root@cedd6d69340f:/usr/local/tomcat# ping 172.17.0.3
PING 172.17.0.3 (172.17.0.3) 56(84) bytes of data.
64 bytes from 172.17.0.3: icmp_seq=1 ttl=64 time=0.090 ms
64 bytes from 172.17.0.3: icmp_seq=2 ttl=64 time=0.057 ms
64 bytes from 172.17.0.3: icmp_seq=3 ttl=64 time=0.057 ms
64 bytes from 172.17.0.3: icmp_seq=4 ttl=64 time=0.061 ms
64 bytes from 172.17.0.3: icmp_seq=5 ttl=64 time=0.060 ms
64 bytes from 172.17.0.3: icmp_seq=6 ttl=64 time=0.057 ms

```

#!/ Single-host netowking

>> In window, there is not bridge, only NAT enabled.

Docker network create

```
# Docker network create -d bridge --subnet 10.10.10.0/24 qt-bridge
```

- If you want to know about subnet values, please google with CIDR.
- I want create containers should be created in this network qt-bridge
- Service discovery is disabled in default network
- Server discovery is enabled in your custom network
- 

# Docker network ls

Now try to create two containers

```
# docker run -d --name c1 --network qt-bridge go1:0.01
# docker run -d --name c2 --network qt-bridge ubuntu:gouse
```

```
# docker inspect c1
# docker inspect c2
```

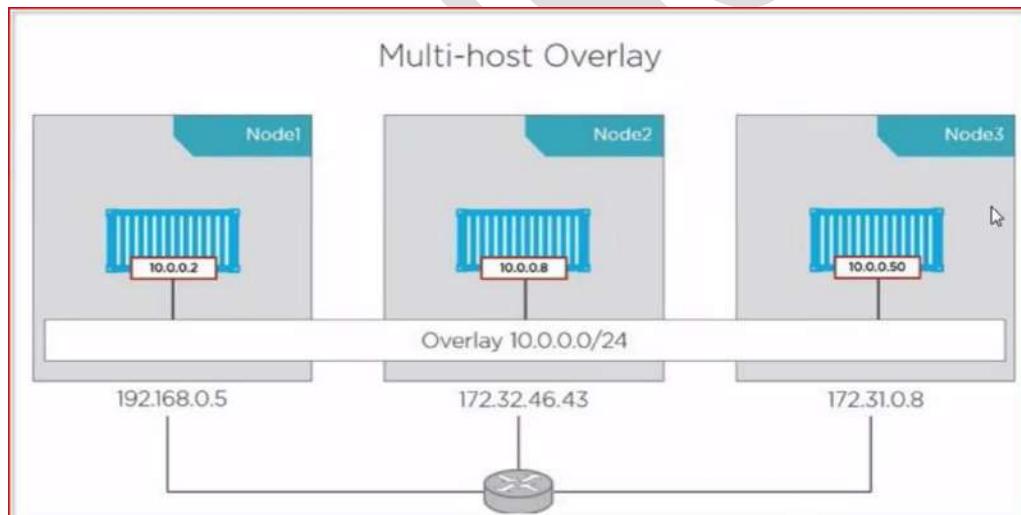
```
# docker exec -it c1 /bin/sh
Ping 10.10.10.2
Ping c2
```

```
# docker exec -it c2 /bin/sh
Ping 10.10.10.1
Ping c1
```

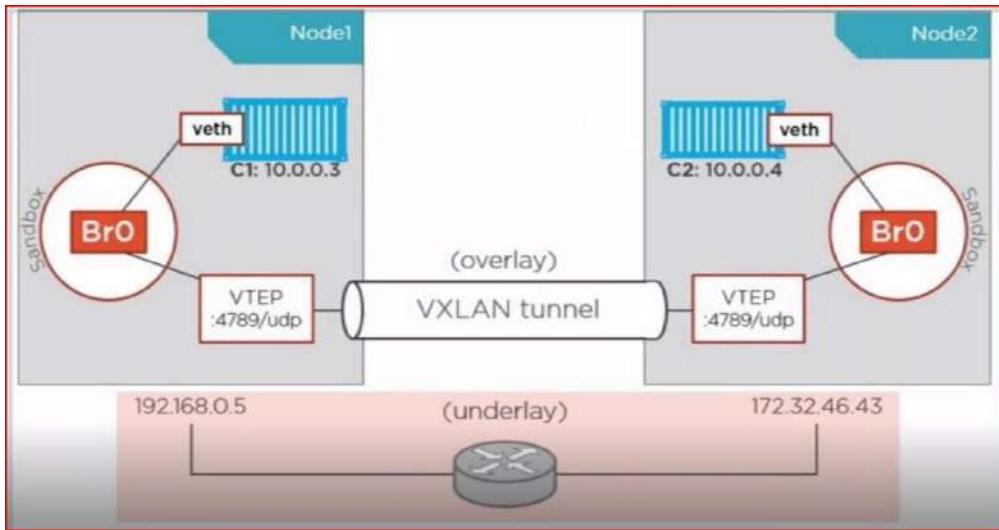
#### Docker Service network discovery

- Service discovery is disabled in default network
- Server discovery is enabled in your custom network

#### !# Multi-host overlay



In this multi-host overlay, all containers will be pinged on different hosts.



It is look as bridge, but it is not bridge.

Above figure is a logically speaking.

It will run on UDP ports or open all ports, you have to open those ports.

So first identify the main server.

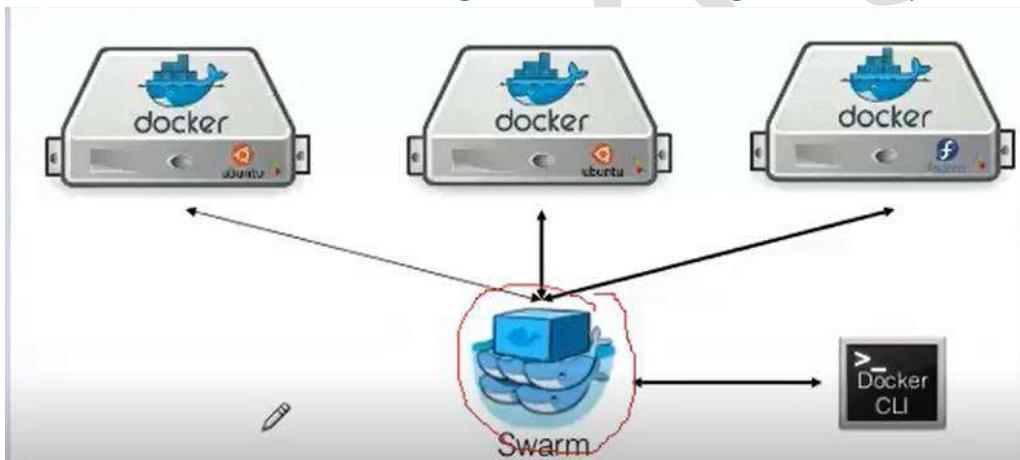
1. Docker swarm in one machine
2. On other machines join the docker swarm to it.

## Docker swarm

Current versions of Docker include swarm mode for natively managing a cluster of Docker Engines called a swarm.

Docker swarm will create a services, and **Services are nothing but a micro services.**

To create a Docker Swarm Manager : failover and High Availability



```
# apt-get update && apt-get install docker.io -y
```

```
# Docker swarm init
```

```
ubuntu@ip-172-31-30-39:~$ docker swarm init → create Swarm MASTER  
Swarm initialized: current node (t1s31vz7gfsxuwszbcb0t) is now a manager.
```

To add a worker to this swarm, run the following command:

```
docker swarm join --token SWMTKN-1-0rt6bgle37ze5pnnapuobb11iutsa149basyrey18  
t1zm8wnl9-729128k1lk96fzritx02zj765 172.31.30.39:2377
```

To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.

```
ubuntu@ip-172-31-30-39:~$ |
```

Copy this docker swarm join token on other machines to join

## Or

Get the ip of master machine and initialize the docker master from below command.

<https://rominirani.com/docker-swarm-tutorial-b67470cf8872>

```
# docker swarm init --advertise-addr 172.30.2.67
```

```
collisions:0 txqueuelen:1000  
RX bytes:44507833 (44.5 MB) TX bytes:151745 (151.7 KB)  
  
lo Link encap:Local Loopback  
inet addr:127.0.0.1 Mask:255.0.0.0  
inet6 addr: ::1/128 Scope:Host  
UP LOOPBACK RUNNING MTU:65536 Metric:1  
RX packets:160 errors:0 dropped:0 overruns:0 frame:0  
TX packets:160 errors:0 dropped:0 overruns:0 carrier:0  
collisions:0 txqueuelen:1  
RX bytes:11840 (11.8 KB) TX bytes:11840 (11.8 KB)  
  
root@Docker-Master:~# docker swarm init --advertise-addr 172.30.2.67  
Swarm initialized: current node (f4g21zt7orlut3ax86dq7cqtr) is now a manager.  
  
To add a worker to this swarm, run the following command:  
  
docker swarm join \  
--token SWMTKN-1-05v3mtk6uslbhowaj5scrqppq6cmvjtve5466umbw2cjuoie28c-3p9nn919pghju3ke1dlbkqbif \  
172.30.2.67:2377  
  
To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.  
root@Docker-Master:~#
```

```
# docker swarm leave
```

Use "docker swarm leave" to leave this swarm and join another one.

Practiced:

```
Name: manager  
ID: QW90:NM9X-UZIN:GRV2:37ND:LP2C:PJFN:4CHN:B2X4:XLTE:YQTE:QKQT  
Docker Root Dir: /var/lib/docker  
Debug Mode (client): false  
Debug Mode (server): false  
Registry: https://index.docker.io/v1/  
Labels:  
Experimental: false  
Insecure Registries:  
127.0.0.0/8  
Live Restore Enabled: false  
  
WARNING: No swap limit support  
root@manager:/home/vagrant# systemctl stop docker  
root@manager:/home/vagrant# docker info  
Cannot connect to the Docker daemon at unix:///var/run/docker.sock. Is the docker daemon running?  
root@manager:/home/vagrant# systemctl start docker  
root@manager:/home/vagrant# docker swarm init  
Error response from daemon: This node is already part of a swarm. Use "docker swarm leave" to leave this swarm and join another one.  
root@manager:/home/vagrant# docker node ls  
ID HOSTNAME STATUS AVAILABILITY MANAGER STATUS ENGINE VERSION  
8yxt2xkj13pvocnc0tm5mazh * manager Ready Active Leader 18.06.2-ce  
root@manager:/home/vagrant# docker swarm leave  
Error response from daemon: You are attempting to leave the swarm on a node that is participating as a manager. Removing the last manager erases all current state of the swarm. Use '--force' to ignore this message.  
root@manager:/home/vagrant# docker node ls  
ID HOSTNAME STATUS AVAILABILITY MANAGER STATUS ENGINE VERSION  
8yxt2xkj13pvocnc0tm5mazh * manager Ready Active Leader 18.06.2-ce  
root@manager:/home/vagrant# docker swarm leave --force  
Node left the swarm.  
root@manager:/home/vagrant# docker node ls  
Error response from daemon: This node is not a swarm manager. Use "docker swarm init" or "docker swarm join" to connect this node to swarm and try again.  
root@manager:/home/vagrant#
```

## # docker info

To check the information about docker master.

### # Docekr node ls

| ID                        | HOSTNAME | STATUS         | AVAILABILITY | MANAGER STATUS |
|---------------------------|----------|----------------|--------------|----------------|
| f4g21zt7orlut3ax86dg7cgtr | *        | ip-172-30-2-67 | Ready        | Active Leader  |

# on docker node1 join the swarm as worker.

```
root@Node1:~# docker swarm join \
> --token SWMTKN-1-05v3mtk6us1bhowaj5scrqpq6cmvjtve5466umbw2cjuoie28c-3p9nn9i9pghju3ke1dlbkqbif \
> 172.30.2.67:2377
This node joined a swarm as a worker.
```

### #docker inspect node1

# To retrieve the join command

<https://docs.docker.com/engine/swarm/join-nodes/>

[https://docs.docker.com/engine/reference/commandline/swarm\\_join-token/](https://docs.docker.com/engine/reference/commandline/swarm_join-token/)

**Join as a worker node:** To retrieve the join command including the join token for worker nodes, run the following command on a manager node:

```
$ docker swarm join-token worker
```

To add a worker to this swarm, run the following command:

```
Node2$ docker swarm join --token SWMTKN-1-3fbwk07niownlyy9icwmz4s4q3boaw7s68ptibhxanbuekovsc-4gh0bnnuufzbazpmf0tdwxkc 192.168.56.100:2377
```

### # docker service

Create a docker service to communicate with clusters.

```
# Docker service create --replicas 2 --name helloworld alpine ping docker.com
```

```
#docker service ls
```

| ID           | NAME       | REPLICAS | IMAGE  | COMMAND         |
|--------------|------------|----------|--------|-----------------|
| dph22mn7605g | helloworld | 2/2      | alpine | ping docker.com |

```
# docker service ps helloworld
```

| ID                                                      | NAME         | IMAGE  | NODE           | DESIRED STATE | CURRENT STATE              | ERROR |
|---------------------------------------------------------|--------------|--------|----------------|---------------|----------------------------|-------|
| 4e0jabj6g5mov5ymbkv5fkvs                                | helloworld.1 | alpine | ip-172-30-2-67 | Running       | Running 51 seconds ago     |       |
| 8uuj9gazj24m18fkj5jekm5be                               | helloworld.2 | alpine | Node2          | Running       | Running 51 seconds ago     |       |
| eeilu4fakylx89s68rdhhxpyp                               | helloworld.3 | alpine | Node1          | Running       | Running 2 seconds ago      |       |
| ajil0dqfouzcr3xzuwx6jp45s                               | helloworld.4 | alpine | Node1          | Running       | Running 2 seconds ago      |       |
| d6npcergyci85itoseje46aka                               | helloworld.5 | alpine | ip-172-30-2-67 | Running       | Running 2 seconds ago      |       |
| root@Docker-Master:~# docker service scale helloworld=5 |              |        |                |               |                            |       |
| helloworld scaled to 5                                  |              |        |                |               |                            |       |
| root@Docker-Master:~# docker service ps helloworld      |              |        |                |               |                            |       |
| 4e0jabj6g5mov5ymbkv5fkvs                                | helloworld.1 | alpine | ip-172-30-2-67 | Running       | Running about a minute ago |       |
| 8uuj9gazj24m18fkj5jekm5be                               | helloworld.2 | alpine | Node2          | Running       | Running about a minute ago |       |
| eeilu4fakylx89s68rdhhxpyp                               | helloworld.3 | alpine | Node1          | Running       | Running 2 seconds ago      |       |
| ajil0dqfouzcr3xzuwx6jp45s                               | helloworld.4 | alpine | Node1          | Running       | Running 2 seconds ago      |       |
| d6npcergyci85itoseje46aka                               | helloworld.5 | alpine | ip-172-30-2-67 | Running       | Running 2 seconds ago      |       |
| root@Docker-Master:~# docker service scale helloworld=4 |              |        |                |               |                            |       |
| helloworld scaled to 4                                  |              |        |                |               |                            |       |
| root@Docker-Master:~# docker service ps helloworld      |              |        |                |               |                            |       |
| 4e0jabj6g5mov5ymbkv5fkvs                                | helloworld.1 | alpine | ip-172-30-2-67 | Running       | Running about a minute ago |       |
| 8uuj9gazj24m18fkj5jekm5be                               | helloworld.2 | alpine | Node2          | Running       | Running about a minute ago |       |
| eeilu4fakylx89s68rdhhxpyp                               | helloworld.3 | alpine | Node1          | Shutdown      | Running 23 seconds ago     |       |
| ajil0dqfouzcr3xzuwx6jp45s                               | helloworld.4 | alpine | Node1          | Running       | Running 23 seconds ago     |       |
| d6npcergyci85itoseje46aka                               | helloworld.5 | alpine | ip-172-30-2-67 | Running       | Running 24 seconds ago     |       |

To get the mapping docker swarm manager to workers:

Copy the above token generated in swarm manager, and run it.

Worker1

```
ubuntu@ip-172-31-26-27:~$ docker swarm join --token SWMTKN-1-0rt6bg1e37ze5pnnapu0bb1iutsa149basyrey18tlzm8wn19-729128k11k96fzritx02zj765 172.31.30.39:2377
This node joined a swarm as a worker.
ubuntu@ip-172-31-26-27:~$ |
```

Worker2

```
ubuntu@ip-172-31-29-22:~$ docker swarm join --token SWMTKN-1-0rt6bg1e37ze5pnnapu0bb1iutsa149basyrey18tlzm8wn19-729128k11k96fzritx02zj765 172.31.30.39:2377
This node joined a swarm as a worker.
ubuntu@ip-172-31-29-22:~$ |
```

So I have 2 nodes joins as worker,

[Docker node ls](#)

| ID                        | TYPE    | MANAGER         | STATUS | HOSTNAME        | ENGINE VERSION | STATUS | AVAILABILITY |
|---------------------------|---------|-----------------|--------|-----------------|----------------|--------|--------------|
| smmfuxrdb25c9n949fbcf06g  | Manager | ip-172-31-26-27 | Ready  | ip-172-31-26-27 | 18.05.0-ce     | Ready  | Active       |
| msrcle1miwwth33n7m59mw051 | Worker  | ip-172-31-29-22 | Ready  | ip-172-31-29-22 | 18.05.0-ce     | Ready  | Active       |
| t1s31vz7gfsxuwszbcbzbev0t | Worker  | *               | Ready  | ip-172-31-30-39 | 18.05.0-ce     | Ready  | Active       |
|                           |         | Leader          |        |                 |                |        |              |

[To open the firewall for udp ports](#)

Execute "docker node ls" to find out nodes

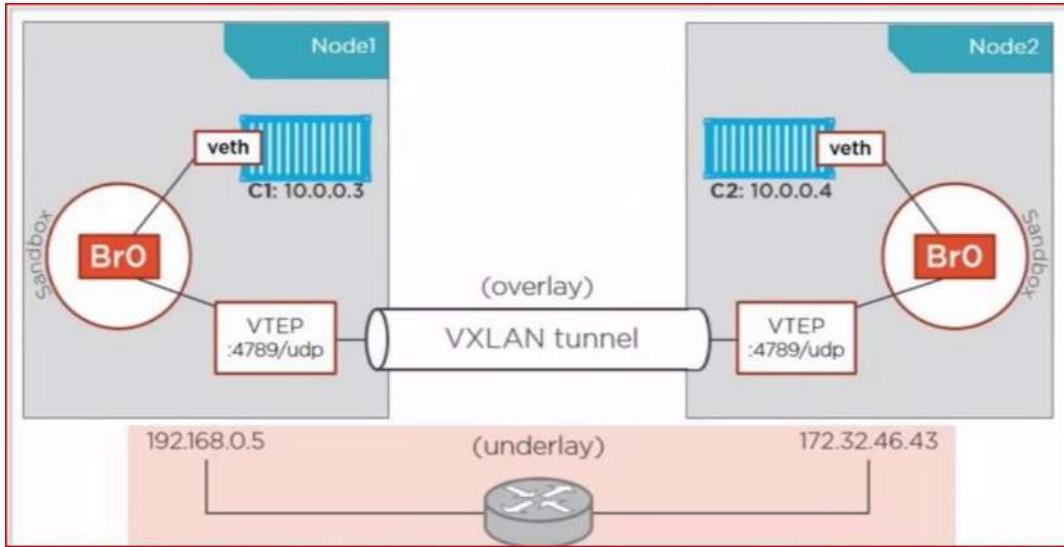
Note: Following ports should be open ==>

2789/udp  
7946/tcp/udp  
2377/tcp

[Create an overlay network](#)

```
# docker service create -d overlay qt-over
#docker network ls
```

```
ubuntu@ip-172-31-30-39:~$ docker network create -d overlay qt-over
mim77fnvuwc7afbjx09nie157
ubuntu@ip-172-31-30-39:~$ docker network ls
NETWORK ID      NAME      DRIVER      SCOPE
de0dc4d48eeef   bridge    bridge      local
0e0b25efacea   docker_gwbridge  bridge      local
06580a93d8e0   host      host       local
nkq6xcmr4mmd   ingress   overlay     swarm
0c95fccebf6b2   none      null       local
2c1f756bae27   qt-bridge  bridge     local
mim77fnvuwc7   qt-over   overlay     swarm
ubuntu@ip-172-31-30-39:~$ |
```



Sandbox containing bridge network.

Now I want to create 3 containers with the single command from manager on containers on different machines.

```
# Docker service create --name qt-svc --network qt-over --replicas 3 nginx sleep 1d
```

- Above command creating 3 nginx containers from a same machine or different machine as your wish.

To see the services status

```
# Docker service ps ps-svc
```

```
ubuntu@ip-172-31-30-39:~$ docker service create --name qt-svc --network qt-over --replicas 3 nginx sleep 1d
Inchyh1cc3i1wkwr5pmg5ni6w
overall progress: 3 out of 3 tasks
1/3: running
2/3: running
3/3: running
verify: Service converged
ubuntu@ip-172-31-30-39:~$ docker service ls
ID           NAME          MODE          REPLICAS
IMAGE        PORTS        desired/running
Inchyh1cc3i1    qt-svc      replicated     3/3
nginx:latest
ubuntu@ip-172-31-30-39:~$ |
```

Here I don't know where these above 3 containers are running.

So **docker service inspect qt-svc**

```
#docker service inspect qt-svc
```

You can see virtualIP details in inspect logs

```
, "VirtualIPs": [
  {
    "NetworkID": "mim77fnvuwc7afbjx09nie157",
    "Addr": "10.0.0.5/24"
```

Now go to other containers and run the ps-a

```
# Docker ps
```

```
ubuntu@ip-172-31-29-22:~$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              PORTS
78b612ee27c3        nginx:latest       "sleep 1d"         About a minute ago
Up About a minute   80/tcp
ubuntu@ip-172-31-29-22:~$ |
```

```
#docker ps
```

```

ubuntu@ip-172-31-26-27:~$ docker swarm join --token SWMTKN-1-0rt6bg1e37ze5pnnap
uobbl1iutsa149basyrey18t1zm8wn19-729128k11k96fzritx02zj765 172.31.30.39:2377
This node joined a swarm as a worker.
ubuntu@ip-172-31-26-27:~$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            NAMES
b4d35df0bb7f      nginx:latest       "sleep 1d"         About a minute ago   qt-svc.1.sjkpr3rpwhoeqoisdm1ih5lj9
ubuntu@ip-172-31-26-27:~$ 

```

Make sure when you are pulling the images make sure it should be available in central repository that is for us is docker hub.

Create a new 3 more containers from docker swarm master.

```
# docker service create --name qt-tom --network qt-over --replicas 3 tomcat
```

```

ubuntu@ip-172-31-30-39:~$ docker service create --name qt-tom --network qt-over
--replicas 3 tomcat
iej5zd6x1pgb06gwumr6gwygd
overall progress: 3 out of 3 tasks
1/3: running
2/3: running
3/3: running
verify: Service converged
ubuntu@ip-172-31-30-39:~$ 

```

Here we can understand that, we are controlling all containers from one machine.

```

ubuntu@ip-172-31-30-39:~$ docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            NAMES
STATUS              PORTS
e7a14781d825      tomcat:latest       "catalina.sh run"   41 seconds ago    qt-tom.1.cwgz1h2qtki4wbgkkcwmlb6i6
Up 40 seconds      8080/tcp
ce94d1640bf1       gol:1.0           "catalina.sh run"   About a minute ago  qt-gol.2.m19vxb1jtm9hml2gelpyzzjmc
Up About a minute  8080/tcp
6b9c57934bba       gol:1.0           "catalina.sh run"   About a minute ago  qt-gol.3.wewi2se9g6c88uw2tgt1bvhbd
Up About a minute  8080/tcp
7ada4ca2fcbb      gol:1.0           "catalina.sh run"   2 minutes ago     qt-gol.1.okz45zn6xokgkd8vcuds6j9rk
Up 2 minutes        8080/tcp
81cba78c1160       nginx:latest       "sleep 1d"          5 minutes ago     qt-svc.3.0uceroy9pxzc9pnk5yuxe4nug
Up 5 minutes        80/tcp
ubuntu@ip-172-31-30-39:~$ docker | 

```

## RealTime Scenario

In Real Time, we will focus on services not on container because might be your containers will be deleted on some part. We will create services on 3 ways in real time scenario.

Frontend service

Middleware service

Backend service

To operate from one mode of communication with the help of services.

Small group of containers under service.

## #docker service

```
ubuntu@ip-172-31-30-39:~$ docker service
Usage: docker service COMMAND
      Manage services
Commands:
  create      Create a new service
  inspect    Display detailed information on one or more services
  logs       Fetch the logs of a service or task
  ls        List services
  ps        List the tasks of one or more services
  rm        Remove one or more services
  rollback   Revert changes to a service's configuration
  scale     Scale one or multiple replicated services
  update    Update a service
Run 'docker service COMMAND --help' for more information on a command.
ubuntu@ip-172-31-30-39:~$
```

Kubernat and swarm are same purpose.

## # Practiced Our Swarm Cluster

Now that our machines are setup, we can proceed with setting up the Swarm.

The first thing to do is initialize the Swarm. We will SSH into the manager1 machine and initialize the swarm in there.

```
$ docker-machine ssh manager1
```

This will initialize the SSH session and you should be at prompt as shown below:

```
$ docker-machine ssh manager1
##          .
## ## ##
/### ## # ## ==#
/*****\ \ / ===
~~~ {~~ ~~~ ~~~ ~~~ ~~~ ~~~ ~ / === ~~~
\______ o
\ \ \ \ \ / \
\ \ \ \ \ /
Boot2Docker version 1.12.1, build HEAD : ef7d0b4 - Thu Aug 18 21:18:06 UTC 2016
Docker version 1.12.1, build 23cf638
docker@manager1:~$
```

Perform the following steps:

```
$ docker swarm init --advertise-addr MANAGER_IP
```

On my machine, it looks like this:

```
docker@manager1:~$ docker swarm init --advertise-addr 192.168.1.8
Swarm initialized: current node (500f62fetd4gry7o09jd9e0kf) is now a manager.
To add a worker to this swarm, run the following command:
docker swarm join \
- token SWMTKN-1-5mgzf6ehuc5pfmar00njd3oxv8nmjhateejaald3yzbef7os11-ad7b1k8k3b13aa3k3q13zivqd \
192.168.1.8:2377
To add a manager to this swarm, run 'docker swarm join-token manager' and follow the instructions.
docker@manager1:~$
```

Great!

You will also notice that the output mentions the docker swarm join command to use in case you want another node to join as a worker. Keep in mind that you can have a node join as a worker or as a manager. At any point in time, there is only one LEADER and the other manager nodes will be as backup in case the current LEADER opts out.

At this point you can see your Swarm status by firing the following command as shown below:

```
docker@manager1:~$ docker node ls
ID           HOSTNAME   STATUS  AVAILABILITY  MANAGER STATUS
500f62fetd... *   manager1  Ready  Active        Leader
```

This shows that there is a single node so far i.e. manager1 and it has the value of Leader for the MANAGER column.

Stay in the SSH session itself for **manager1**.

### Joining as Worker Node

To find out what docker swarm command to use to join as a node, you will need to use the join-token <role> command.

To find out the join command for a **worker**, fire the following command:

```
docker@manager1:~$ docker swarm join-token worker
To add a worker to this swarm, run the following command:
docker swarm join \
- token SWMTKN-1-5mgzf6ehuc5pfmar00njd3oxv8nmjhateejaald3yzbef7os11-ad7b1k8k3b13aa3k3q13zivqd \
192.168.1.8:2377
docker@manager1:~$
```

### Joining as Manager Node

To find out the the join command for a manager, fire the following command:

```

docker@manager1:~$ docker swarm join-token manager
To add a manager to this swarm, run the following command:
docker swarm join \
    --token SWMTKN-1-5mgzf6ehuc5pfmar00njd3oxv8nmjhteejaald3yzbef7os11-8xo0cmd6bryjrsh6w7op4enos \
    192.168.1.8:2377
docker@manager1:~$
```

Notice in both the above cases, that you are provided a token and it is joining the Manager node (you will be able to identify that the IP address is the same the **MANAGER\_IP** address).

Keep the SSH to manager1 open. And fire up other command terminals for working with other worker docker machines.

## Adding Worker Nodes to our Swarm

Now that we know how to check the command to join as a worker, we can use that to do a SSH into each of the worker Docker machines and then fire the respective join command in them.

In my case, I have 5 worker machines (worker1/2/3/4/5). For the first worker1 Docker machine, I do the following:

- SSH into the worker1 machine i.e. docker-machine ssh worker1
- Then fire the respective command that I got for joining as a worker. In my case the output is shown below:

```

docker@worker1:~$ docker swarm join \
    --token SWMTKN-1-5mgzf6ehuc5pfmar00njd3oxv8nmjhteejaald3yzbef7os11-ad7b1k8k3b13aa3k3q13zivqd \
    192.168.1.8:2377
This node joined a swarm as a worker.
docker@worker1:~$
```

I do the same thing by launching SSH sessions for worker2/3/4/5 and then pasting the same command since I want all of them to be worker nodes.

After making all my worker nodes join the Swarm, I go back to my manager1 SSH session and fire the following command to check on the status of my Swarm i.e. see the nodes participating in it:

```

docker@manager1:~$ docker node ls
ID          HOSTNAME  STATUS  AVAILABILITY  MANAGER STATUS
1ndqsslh7fpquc7fi35leig54  worker4  Ready   Active
1qh4aat24nts5izo3cgsboy77  worker5  Ready   Active
25nwmw5eg7a5ms4ch93aw0k03  worker3  Ready   Active
5oof62fetd4gry7o09jd9e0kf *  manager1  Ready   Active      Leader
5pm9f2pzs8ndijqkkblkgqbsf  worker2  Ready   Active
9yq4lcmfg0382p39euk81j9p4  worker1  Ready   Active
docker@manager1:~$
```

As expected, you can see that I have 6 nodes, one as the manager (manager1) and the other 5 as workers.

We can also do execute the standard docker info command here and zoom into the Swarm section to check out the details for our Swarm.

```

Swarm: active
  NodeID: 5oof62fetd4gry7o09jd9e0kf
  Is Manager: true
  ClusterID: 6z3sqrlaqank2uimyzijzapz3
  Managers: 1
  Nodes: 6
  Orchestration:
    Task History Retention Limit: 5
  Raft:
    Snapshot Interval: 10000
    Heartbeat Tick: 1
    Election Tick: 3
  Dispatcher:
    Heartbeat Period: 5 seconds
  CA Configuration:
    Expiry Duration: 3 months
  Node Address: 192.168.1.8
```

Notice a few of the properties:

- The Swarm is marked as active. It has 6 Nodes in total and 1 manager among them.
- Since I am running the **docker info** command on the manager1 itself, it shows the **Is Manager** as **true**.
- The Raft section is the Raft consensus algorithm that is used. Check out the details [here](#).

## Create a Service

Now that we have our swarm up and running, it is time to schedule our containers on it. This is the whole beauty of the orchestration layer. We are going to focus on the app and not worry about where the application is going to run.

All we are going to do is tell the manager to run the containers for us and it will take care of scheduling out the containers, sending the commands to the nodes and distributing it.

To start a service, you would need to have the following:

- What is the Docker image that you want to run. In our case, we will run the standard **nginx** image that is officially available from the Docker hub.
- We will expose our service on port 80.
- We can specify the number of containers (or instances) to launch. This is specified via the **replicas** parameter.

- We will decide on the name for our service. And keep that handy.

What I am going to do then is to launch 5 replicas of the nginx container. To do that, I am again in the SSH session for my manager1 node. And I give the following **docker service create** command:

```
docker service create --replicas 5 -p 80:80 --name web nginx
ctolqlt4h2o859t69j9pptye
```

What has happened is that the Orchestration layer has now got to work.

You can find out the status of the service, by giving the following command:

```
docker@manager1:~$ docker service ls
ID           NAME      REPLICAS   IMAGE      COMMAND
ctolqlt4h2o8  web       0/5        nginx
```

This shows that the replicas are not yet ready. You will need to give that command a few times.

In the meanwhile, you can also see the status of the service and how it is getting orchestrated to the different nodes by using the following command:

```
docker@manager1:~$ docker service ps web
ID           NAME      NODE      DESIRED STATE     CURRENT STATE      ERROR
71*         web.1    nginx    worker3    Running          Preparing 2 minutes ago
17*         web.2    nginx    manager1   Running          Running 22 seconds ago
ey*         web.3    nginx    worker2    Running          Running 2 minutes ago
bd*         web.4    nginx    worker5    Running          Running 45 seconds ago
dw*         web.5    nginx    worker4    Running          Running 2 minutes ago
```

This shows that the nodes are getting setup. It could take a while.

But notice a few things. In the list of nodes above, you can see that the 5 containers are being scheduled by the orchestration layer on **manager1, worker2, worker3, worker4 and worker5**. There is no container scheduled for **worker1** node and that is fine.

A few executions of **docker service ls** shows the following responses:

```
docker@manager1:~$ docker service ls
ID           NAME      REPLICAS   IMAGE      COMMAND
ctolqlt4h2o8  web       3/5        nginx
```

and then finally:

```
docker@manager1:~$ docker service ls
ID           NAME      REPLICAS   IMAGE      COMMAND
ctolqlt4h2o8  web       5/5        nginx
```

If we look at the service processes at this point, we can see the following:

```
docker@manager1:~$ docker service ps web
ID           NAME      NODE      DESIRED STATE     CURRENT STATE      ERROR
71*         web.1    nginx    worker3    Running          Running 4 minutes ago
17*         web.2    nginx    manager1   Running          Running 7 minutes ago
ey*         web.3    nginx    worker2    Running          Running 9 minutes ago
bd*         web.4    nginx    worker5    Running          Running 8 minutes ago
dw*         web.5    nginx    worker4    Running          Running 9 minutes ago
```

If you do a **docker ps** on the manager1 node right now, you will find that the nginx daemon has been launched.

```
docker@manager1:~$ docker ps
CONTAINER ID        IMAGE               COMMAND                  CREATED             STATUS              PORTS
NAMES
933309b04630        nginx:latest        "nginx -g 'daemon off'"   2 minutes ago      Up 2 minutes       80/tcp, 443/tcp
web.2.17d502y6qjhd1wqjle13nmjvc
docker@manager1:~$
```

## Accessing the Service

You can access the service by hitting any of the manager or worker nodes. It does not matter if the particular node does not have a container scheduled on it. That is the whole idea of the swarm.

Try out a curl to any of the Docker Machine IPs (manager1 or worker1/2/3/4/5) or hit the URL (<http://<machine-ip>>) in the browser. You should be able to get the standard NGINX Home page.

or if we hit the worker IP:

Nice, isn't it?

Ideally you would put the Docker Swarm service behind a Load Balancer.

## Scaling up and Scaling down

This is done via the **docker service scale** command. We currently have 5 containers running. Let us bump it up to 8 as shown below by executing the command on the **manager1** node.

```
$ docker service scale web=8
web scaled to 8
```

Now, we can check the status of the service and the process tasks via the same commands as shown below:

```
docker@manager1:~$ docker service ls
ID           NAME      REPLICAS   IMAGE      COMMAND
ctolqlt4h2o8  web       5/8        nginx
```

In the ps web command below, you will find that it has decided to schedule the new containers on worker1 (2 of them) and manager1(one of them)

```
docker@manager1:~$ docker service ps web
ID           NAME      NODE      DESIRED STATE     CURRENT STATE      ERROR

```

```

7i* web.1 nginx worker3 Running     Running 14 minutes ago
17* web.2 nginx manager1 Running   Running 17 minutes ago
ey* web.3 nginx worker2 Running   Running 19 minutes ago
bd* web.4 nginx worker5 Running   Running 17 minutes ago
dw* web.5 nginx worker4 Running   Running 19 minutes ago
8t* web.6 nginx worker1 Running   Starting about a minute ago
b8* web.7 nginx manager1 Running  Ready less than a second ago
Ok* web.8 nginx worker1 Running   Starting about a minute ago

```

We wait for a while and then everything looks good as shown below:

```

docker@manager1:~$ docker service ls
ID          NAME      REPLICAS IMAGE COMMAND
ctolq1t4h2o8 web 8/8 nginx
docker@manager1:~$ docker service ps web
ID          NAME      IMAGE NODE      DESIRED STATE CURRENT STATE ERROR
7i* web.1    nginx  worker3  Running     Running 16 minutes ago
17* web.2    nginx  manager1  Running   Running 19 minutes ago
ey* web.3    nginx  worker2  Running   Running 21 minutes ago
bd* web.4    nginx  worker5  Running   Running 20 minutes ago
dw* web.5    nginx  worker4  Running   Running 21 minutes ago
8t* web.6    nginx  worker1  Running   Running 4 minutes ago
b8* web.7    nginx  manager1  Running  Running 2 minutes ago
Ok* web.8    nginx  worker1  Running   Running 3 minutes ago
docker@manager1:~$
```

## Inspecting nodes

You can inspect the nodes anytime via the docker node inspect command.

For example if you are already on the node (for example manager1) that you want to check, you can use the name self for the node.

```
$ docker node inspect self
```

Or if you want to check up on the other nodes, give the node name. For e.g.

```
$ docker node inspect worker1
```

## Draining a node

If the node is ACTIVE, it is ready to accept tasks from the Master i.e. Manager. For e.g. we can see the list of nodes and their status by firing the following command on the **manager1** node.

```

docker@manager1:~$ docker node ls
ID          HOSTNAME  STATUS  AVAILABILITY  MANAGER STATUS
1ndqsslh7fpquc7fi35leig54  worker4  Ready  Active
lqh4aat24nts5izo3cgsboy77  worker5  Ready  Active
25nwmw5eg7a5ms4ch93aw0k03  worker3  Ready  Active
5oof62fetd4gry7o09jd9e0kf * manager1  Ready  Active  Leader
5pm9f2pqr8ndijqkkblkgqbsf  worker2  Ready  Active
9yq41cmfg0382p39euk81j9p4  worker1  Ready  Active
docker@manager1:~$
```

You can see that their AVAILABILITY is set to READY.

As per the documentation, When the node is active, it can receive new tasks:

- during a service update to scale up
- during a rolling update
- when you set another node to Drain availability
- when a task fails on another active node

But sometimes, we have to bring the Node down for some maintenance reason. This meant by setting the Availability to Drain mode. Let us try that with one of our nodes.

But first, let us check the status of our processes for the web services and on which nodes they are running:

```

docker@manager1:~$ docker service ps web
ID          NAME      IMAGE NODE      DESIRED STATE CURRENT STATE      ERROR
7i* web.1    nginx  worker3  Running     Running 54 minutes ago
17* web.2    nginx  manager1  Running   Running 57 minutes ago
ey* web.3    nginx  worker2  Running   Running 59 minutes ago
bd* web.4    nginx  worker5  Running   Running 57 minutes ago
dw* web.5    nginx  worker4  Running   Running 59 minutes ago
8t* web.6    nginx  worker1  Running   Running 41 minutes ago
b8* web.7    nginx  manager1  Running  Running 39 minutes ago
Ok* web.8    nginx  worker1  Running   Running 41 minutes ago

```

You find that we have 8 replicas of our service:

- 2 on manager1
- 2 on worker1
- 1 each on worker2, worker3, worker4 and workers5

Now, let us use another command to check what is going on in node **worker1**.

```

docker@manager1:~$ docker node ps worker1
ID          NAME      IMAGE NODE      DESIRED STATE CURRENT STATE      8t* web.6    nginx  worker1  Running      Running 44 minutes
ago
Ok* web.8    nginx  worker1  Running   Running 44 minutes ago
docker@manager1:~$
```

We can also use the docker node inspect command to check the availability of the node and as expected, you will find a section in the output as follows:

```
$ docker node inspect worker1
...
"Spec": {
  "Role": "worker",
  "Availability": "active"
},
...
or
docker@manager1:~$ docker node inspect --pretty worker1
ID: 9yq4lcmfg0382p39euk8lj9p4
Hostname: worker1
Joined at: 2016-09-16 08:32:24.5448505 +0000 utc
Status:
  State: Ready
  Availability: Active
Platform:
  Operating System: linux
  Architecture: x86_64
Resources:
  CPUs: 1
  Memory: 987.2 MiB
Plugins:
  Network: bridge, host, null, overlay
  Volume: local
Engine Version: 1.12.1
Engine Labels:
  - provider = hyperv
docker@manager1:~$
```

We can see that it is **“Active”** for its Availability attribute.

Now, let us set the **Availability** to **DRAIN**. When we give that command, the Manager will stop tasks running on that node and launches the replicas on other nodes with ACTIVE availability.

So what we are expecting is that the Manager will bring the 2 containers running on worker1 and schedule them on the other nodes (manager1 or worker2 or worker3 or worker4 or worker5).

This is done by updating the node by setting its availability to “drain”.

```
docker@manager1:~$ docker node update --availability drain worker1
```

Now, if we do a process status for the service, we see an interesting output (*I have trimmed the output for proper formatting*):

```
docker@manager1:~$ docker service ps web
ID      NAME      IMAGE      NODE      DESIRED STATE      CURRENT STATE
71*    web.1      nginx      worker3    Running     Running about an hour ago
17*    web.2      nginx      manager1   Running     Running about an hour ago
ey*    web.3      nginx      worker2    Running     Running about an hour ago
bd*    web.4      nginx      worker5    Running     Running about an hour ago
dw*    web.5      nginx      worker4    Running     Running about an hour ago
2u*    web.6      nginx      worker4    Running     Preparing about a min ago
8t*    \_ web.6   nginx      worker1    Shutdown   Shutdown about a min ago
b8*    web.7      nginx      manager1   Running     Running 49 minutes ago
7a*    web.8      nginx      worker3    Running     Preparing about a min ago
0k*    \_ web.8   nginx      worker1    Shutdown   Shutdown about a min ago
docker@manager1:~$
```

You can see that the containers on worker1 (which we have asked to be drained) are being rescheduled on other workers. In our scenario above, they got scheduled to worker2 and worker3 respectively. This is required because we have asked for 8 replicas to be running in an earlier scaling exercise.

You can see that the two containers are still in “Preparing” state and after a while if you run the command, they are all running as shown below:

```
docker@manager1:~$ docker service ps web
ID      NAME      IMAGE      NODE      DESIRED STATE      CURRENT STATE
71*    web.1      nginx      worker3    Running     Running about an hour ago
17*    web.2      nginx      manager1   Running     Running about an hour ago
ey*    web.3      nginx      worker2    Running     Running about an hour ago
bd*    web.4      nginx      worker5    Running     Running about an hour ago
dw*    web.5      nginx      worker4    Running     Running about an hour ago
2u*    web.6      nginx      worker4    Running     Running 8 minutes ago
8t*    \_ web.6   nginx      worker1    Shutdown   Shutdown 8 minutes ago
b8*    web.7      nginx      manager1   Running     Running 56 minutes ago
7a*    web.8      nginx      worker3    Running     Running 8 minutes ago
0k*    \_ web.8   nginx      worker1    Shutdown   Shutdown 8 minutes ago
```

This makes for cool demo, isn’t it?

## Remove the Service

You can simply use the service rm command as shown below:

```
docker@manager1:~$ docker service rm web
web
docker@manager1:~$ docker service ls
ID      NAME      REPLICAS      IMAGE      COMMAND
docker@manager1:~$ docker service inspect web
[]
Error: no such service: web
```

# Applying Rolling Updates

This is straight forward. In case you have an updated Docker image to roll out to the nodes, all you need to do is fire an service update command.

For e.g.

```
$ docker service update --image <imagename>:<version> web
```

Docker swarm join issues:

join Error response from daemon: rpc error: code = Unavailable desc = all SubConns are in TransientFailure, latest connection error: connection error: desc = "transport: Error while dialing dial tcp 192.168.56.100:2377: connect: protocol not available"

**Answer 1:** [Still not working.]

This is usually because either you have the firewall enabled (check to be sure - 'sudo systemctl status firewalld' and if enabled, disable with 'sudo systemctl stop firewalld').

**Answer 2:**

I'd say it's possibly firewall related. Ensure your ports are configured correctly on the third box. From the Docker docs:

Open protocols and ports between the hosts The following ports must be available. On some systems, these ports are open by default.

TCP port 2377 for cluster management communications TCP and UDP port 7946 for communication among nodes UDP port 4789 for overlay network traffic

**Answer 3 :** [Still not working.]

Try changing your docker swarm init command, change it to:

```
docker swarm init --advertise-addr Ip_address --listen-addr ip_address
```

then join other worker nodes. It should work.

```
root@manager:/home/vagrant# vi /etc/hosts
root@manager:/home/vagrant# cat /etc/hosts
127.0.0.1      localhost
127.0.1.1      manager manager
127.0.0.1      manager manager
```

## Docker swarm | Open protocols and ports between the hosts

The following ports must be available. On some systems, these ports are open by default.

**TCP port 2377** for cluster management communications

**TCP and UDP port 7946** for communication among nodes

**UDP port 4789** for overlay network traffic

If you plan on creating an overlay network with encryption (--opt encrypted), you also need to ensure **ip protocol 50 (ESP)** traffic is allowed.

#Github links :

<https://github.com/asquarezone>  
<https://github.com/asquarezone/DockerZone>

multi-machine scenario

<https://rominirani.com/docker-swarm-tutorial-b67470cf8872>

copying it in zip file

zip file

unzip

and then running one node application

pm2 is a open source tool

## Docker Process to build push to docker hub:

go to docker hub | the naming formate for docker image is loginname/imagename

docker build -t gouseshaik/gameoflife:1.0

docker login

and then push the image

docker push gouseshaik/gameoflife:1.0

# docker-machine and docker datacenters are all related to Enterprise products.

<https://rominirani.com/docker-swarm-tutorial-b67470cf8872>

## Docker stack

<https://docs.docker.com/get-started/part5/>

### Prerequisites

- Install Docker version 1.13 or higher.
- Get Docker Compose as described in Part 3 prerequisites.
- Get Docker Machine as described in Part 4 prerequisites.
- Read the orientation in Part 1.
- Learn how to create containers in Part 2.
- Make sure you have published the friendlyhello image you created by pushing it to a registry. We use that shared image here.
- Be sure your image works as a deployed container. Run this command, slotting in your info for username, repo, and tag: docker run -p 80:80 username/repo:tag, then visit <http://localhost/>.
- Have a copy of your docker-compose.yml from Part 3 handy.
- Make sure that the machines you set up in part 4 are running and ready. Run docker-machine ls to verify this. If the machines are stopped, run docker-machine start myvm1 to boot the manager, followed by docker-machine start myvm2 to boot the worker.
- Have the swarm you created in part 4 running and ready. Run docker-machine ssh myvm1 "docker node ls" to verify this. If the swarm is up, both nodes report a ready status. If not, reinitialize the swarm and join the worker as described in Set up your swarm.

### Kubernetes:

You will use bigger port numbers like 3320432 in kubernetes.

### How to Limit Memory and CPU for Docker Containers



A container without limits will have access to all system resources, potentially starving other services or containers. To combat this you may want to enforce some limits to ensure all containers are treated equally, or

that some become more equal than others. This tutorial will show you how to limit memory and CPU for Docker containers.

Setting Limits on Ubuntu 16.04

Out of the box a Docker installation on Ubuntu 16.04 we not be capable of setting limits. This is because cgroups swapping is disabled by default. When attempting to set limits you will be given the following error. **WARNING: Your kernel does not support swap limit capabilities or the cgroup is not mounted. Memory limited without swap.**

To address this error we can enable cgroup swapping by doing the following.

1. Open the grub configuration file in a text editor.

```
vi /etc/default/grub
```

2. Add the following line. If the **GRUB\_CMDLINE\_LINUX** optional already exists, modify it to include the values below.

```
GRUB_CMDLINE_LINUX="cgroup_enable=memory swapaccount=1"
```

3. Save your changes and exit the text editor.

4. Update the grub configuration.

```
sudo update-grub
```

5. Before the changes will be applied you will need to reboot your docker host.

Running Without Limits

Containers will automatically have access to the entire range of RAM and CPU processing power of its host. If you are running a single container, this may not be an issue. When you start hosting multiple containers, each one will than start stepping on each other.

We started an NGINX container as a demonstration of memory usage. As you can see from the screenshot below the container's limit is 8GB.

| CONTAINER ID | NAME          | CPU % | MEM USAGE / LIMIT   | MEM % | NET I/O   | BLOCK I/O | PIDS |
|--------------|---------------|-------|---------------------|-------|-----------|-----------|------|
| 08b3ba3b8d08 | happy_poitras | 0.00% | 1.344MiB / 7.796GiB | 0.02% | 828B / 0B | 0B / 0B   | 2    |

Docker stats output

We can apply memory limits to ensure the container never uses more than 256 MB of RAM, for example. We'll keep the first container running and launch a new one with the limits applied.

| CONTAINER ID | NAME           | CPU % | MEM USAGE / LIMIT   | MEM % | NET I/O   | BLOCK I/O | PIDS |
|--------------|----------------|-------|---------------------|-------|-----------|-----------|------|
| 1c7eb5235562 | eloquent_morse | 0.00% | 2.352MiB / 256MiB   | 0.92% | 648B / 0B | 0B / 0B   | 2    |
| 08b3ba3b8d08 | happy_poitras  | 0.00% | 1.344MiB / 7.796GiB | 0.02% | 828B / 0B | 0B / 0B   | 2    |

Docker stats with two running containers

Limiting Memory

To limit memory we use the **memory** flag when starting a container. For example, we used the following to limit our NGINX server to only 256 MB of RAM.

```
docker run -d -p 8081:80 --memory="256m" nginx
```

This sets a hard limit. That means that under no circumstances will the container be allowed to use more than 256 MB of RAM. Alternatively, we could set a soft limit. Soft limits ensure our container can still request additional memory after hitting its limit, preventing service outages.

The flag to set a soft limit is **memory-reservation**. To set a soft limit of 256 MB of RAM we would run the following command.

```
docker run -d -p 8081:80 --memory-reservation="256m" nginx
```

Limiting CPU

Allowing one container to monopolize the processors in your Docker host could cause service outages, by starving your other services and containers. Limit how much CPU a container can use.

Limit Number of Cores

We can limit the number of cores available to container by using the **cpus** flag.

Lock Container to Specific Cores

Just limiting the number of cores means your process will use any available core available. For most purposes this is fine. Sometimes, however, you may want to lock your containers to specific cores.

Limit CPU Time

Limiting CPU time ensure how often a process is able to interrupt the processor or a set of cores.

Shares and Weights

Rather than breaking out the calculator and being very specific about how many cores or CPU time a process can have, apply shares to your process instead. This allows more critical containers to have priority over the CPU when needed.

## How to Increase the CPU Share and Memory of a Container

To increase the Docker CPU share and the memory of a container, use the following example command:

```
docker update --cpu-shares 5120 -m 3000M ml-tm
```

**nsenter**

By default, when you launch a container, you will also use a **shell command** while launching the container as shown below. This is what we have seen in the earlier chapters when we were working with containers.

```
demo@ubuntuserver:~$ sudo docker run -it centos /bin/bash
[root@76d00fbce4c ~]# demo@ubuntuserver:~$ 
demo@ubuntuserver:~$ sudo docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              NAMES
STATUS              PORTS              NAMES
d76d00fbce4c        centos:latest      "/bin/bash"        7 seconds ago    Up 6 seconds   boring_goldstine
Up 6 seconds
demo@ubuntuserver:~$
```

In the above screenshot, you can observe that we have issued the following command –  
**sudo docker run -it centos /bin/bash**

We used this command to create a new container and then used the Ctrl+P+Q command to exit out of the container. It ensures that the container still exists even after we exit from the container.

We can verify that the container still exists with the Docker **ps** command. If we had to exit out of the container directly, then the container itself would be destroyed.

Now there is an easier way to attach to containers and exit them cleanly without the need of destroying them. One way of achieving this is by using the **nsenter** command.

Before we run the **nsenter** command, you need to first install the **nsenter** image. It can be done by using the following command –

```
docker run --rm -v /usr/local/bin:/target jpetazzo/nsenter
```

```
78ddc5c32d95: Pull complete
f68a85041029: Pull complete
a82810b57d94: Pull complete
b7593cf24c20: Pull complete
0cb2fb8d9656: Pull complete
f362bdd62dfa: Pull complete
dc5370e1cb31: Pull complete
5e3c4cd314f8: Pull complete
f3e4e1652690: Pull complete
29c0847bbd97: Pull complete
5cc9b463abfd: Pull complete
f36b6bd8d305: Pull complete
3731cc001b90: Pull complete
7b05e8e06727: Pull complete
1a0a7e2eaeb: Pull complete
a249cf324221: Pull complete
Digest: sha256:17dc70210e269e76d2f99c575bc9788ba9565aa9cb70b7132f895a6e846db86
Status: Downloaded newer image for jpetazzo/nsenter:latest
Installing nsenter to /target
Installing docker-enter to /target
Installing importenv to /target
demo@ubuntudemo:~$ 
demo@ubuntudemo:~$
```

Before we use the **nsenter** command, we need to get the Process ID of the container, because this is required by the **nsenter** command. We can get the Process ID via the Docker **inspect** command and filtering it via the **Pid**.

```
root@ubuntudemo:~# sudo docker ps
CONTAINER ID        IMAGE               COMMAND             CREATED            STATUS              NAMES
STATUS              PORTS              NAMES
ef42a4c5e663        centos:latest      "/bin/bash"        2 minutes ago    Up 2 minutes   stoic_banach
root@ubuntudemo:~# sudo docker inspect ef42a4c5e663 | grep Pid
    "PidMode": "",,
    "Pid": 2978,
root@ubuntudemo:~#
```

As seen in the above screenshot, we have first used the **docker ps** command to see the running containers. We can see that there is one running container with the ID of ef42a4c5e663.

We then use the Docker **inspect** command to inspect the configuration of this container and then use the **grep** command to just filter the Process ID. And from the output, we can see that the Process ID is 2978. Now that we have the process ID, we can proceed forward and use the **nsenter** command to attach to the Docker container.

**nsenter**

This method allows one to attach to a container without exiting the container.

#### Syntax

nsenter -m -u -n -p -i -t containerID command

#### Options

-**u** is used to mention the **Uts namespace**

-**m** is used to mention the **mount namespace**

-**n** is used to mention the **network namespace**

-**p** is used to mention the **process namespace**

-**i** s to make the container run in interactive mode.

-**t** is used to connect the I/O streams of the container to the host OS.

**containerID** - This is the ID of the container.

**Command** - This is the command to run within the container.

**Example** : sudo nsenter -m -u -n -p -i -t 2978 /bin/bash

**Output:**

```
root@ubuntudemo:~# sudo nsenter -m -u -n -p -i -t 2978 /bin/bash
```

```
bash-4.2# exit
```

```
exit
```

```
root@ubuntudemo:~# sudo docker ps
```

| CONTAINER ID       | IMAGE         | COMMAND      | CREATED       |
|--------------------|---------------|--------------|---------------|
| STATUS             | PORTS         | NAMES        |               |
| ef42a4c5e663       | centos:latest | "/bin/bash"  | 9 minutes ago |
| Up 9 minutes       |               | stoic_banach |               |
| root@ubuntudemo:~# |               |              |               |

From the output, we can observe the following points -

- The prompt changes to the **bash shell** directly when we issue the **nsenter** command.
- We then issue the **exit** command. Now normally if you did not use the **nsenter** command, the container would be destroyed. But you would notice that when we run the **nsenter** command, the container is still up and running.

#### GIT

Introduction to GIT on Linux - Install, Create Project, Commit Files

I have followed the below doc for installation

<https://oracle-base.com/articles/linux/git-2-2-installation-on-linux>

#### 1. Download and Install GIT

First, download the GIT from [here](#). Or, download it directly using wget as shown below.

```
cd
```

```
wget https://mirrors.edge.kernel.org/pub/software/scm/git/git-2.9.5.tar.gz
```

Next, extract the downloaded file.

```
tar xvfj git-1.7.6.tar.bz2  
cd git-1.7.6
```

Finally, install GIT as shown below using the default configure option. If you want to customize the installation, do "./configure -help" to view all available configuration options.

```
./configure
```

```
make
```

if you got below error while executing make command:

```
/usr/bin/perl Makefile.PL PREFIX='/usr/local/git' INSTALL BASE='' --localedir='/usr/local/git/share/locale'  
Can't locate ExtUtils/MakeMaker.pm in @INC (@INC contains: /usr/local/lib64/perl5 /usr/local/share/perl5  
/usr/lib64/perl5/vendor_perl5/vendor perl /usr/lib64/perl5 /usr/share/perl5 .) at Makefile.PL line 3.  
BEGIN failed--compilation aborted at Makefile.PL line 3.  
make[1]: *** [perl.mak] Error 2  
make: *** [perl/perl.mak] Error 2  
[oracle@orainux7 git-2.9.5]$ exit
```

```
yum install perl-devel
```

- make install
- or
- cd git-2.2.2
- make prefix=/usr/local/git all
- make prefix=/usr/local/git install

installation failed as below, so I have changed the permission with 755 for the directory /usr/local/git.

```
install -d -m 755 '/usr/local/git/bin'
install: cannot create directory '/usr/local/git': Permission denied
make: *** [install] Error 1
[oracle@orlinux7 git-2.9.5]$ ls -ltr /usr/local/git/bin
ls: cannot access /usr/local/git/bin: No such file or directory
[vagrant@orlinux7 ~]$ sudo su
[root@orlinux7 vagrant]# cd
[root@orlinux7 ~]# cd /usr/local/
[root@orlinux7 local]# ls
bin etc games include lib lib64 libexec sbin share src
[root@orlinux7 local]# mkdir git
[root@orlinux7 local]# chown -R oracle /usr/local/git
[root@orlinux7 local]# chmod 755 /usr/local/git
[root@orlinux7 local]# exit
exit
[vagrant@orlinux7 ~]$ sudo su oracle
[oracle@orlinux7 vagrant]$ cd
[oracle@orlinux7 ~]$ cd git-2.9.5/
[oracle@orlinux7 git-2.9.5]$ make prefix=/usr/local/git install
```

```
echo "export PATH=$PATH:/usr/local/git/bin" >> /etc/bashrc
[oracle@orlinux7 ~]$ vi .bash_profile
[oracle@orlinux7 ~]$ . ./bash_profile
[oracle@orlinux7 ~]$ whereis git
git: /usr/local/git /usr/local/git/bin/git
[oracle@orlinux7 ~]$ git -version
Unknown option: -version
usage: git [--version] [--help] [-C <path>] [-c name=value]
           [--exec-path[=<path>]] [--html-path] [--man-path] [--info-path]
           [-p | --paginate | --no-pager] [--no-replace-objects] [--bare]
           [--git-dir=<path>] [--work-tree=<path>] [--namespace=<name>]
           <command> [<args>]
[oracle@orlinux7 ~]$ git --version
git version 2.9.5
```

source /etc/bashrc

## 2. Initial Configuration

Git is installed by default under /usr/local/bin. Once you've installed GIT, verify it as shown below.

```
$ whereis git
git: /usr/local/bin/git

$ git --version
git version 1.7.6

$ git --help.
```

The first step is to specify your username and email address to your GIT repository using "git config" as shown below.

```
git config --global user.name "GIT Admin"
```

```
git config --global user.email gowseshaik@gmail.com
```

```
[oracle@orainux7 ~]$ git config --global user.name "GIT Admin"  
[oracle@orainux7 ~]$ git config --global user.email "gowseshaik@gmail.com"
```

Verify the git configuration information as shown below.

```
$ git config --list  
user.name=GIT Admin  
user.email=ramesh@thegeekstuff.com  
core.repositoryformatversion=0  
core.filemode=true  
core.bare=false  
core.logallrefupdates=true
```

This information is stored in the .gitconfig file under your home directory.

```
$ cat ~/.gitconfig  
[user]  
    name = GIT Admin  
    email = ramesh@thegeekstuff.com  
  
$ cat ~/.gitconfig  
[core]  
    autocrlf = true  
    excludesfile = C:\\\\Users\\\\80053806\\\\Documents\\\\gitignore_global.txt  
[gui]  
    recentrepo = C:/Gouse_1/Mobily_Daily_Activities
```

### 3. Create a Project

You can make any of your local directory as a GIT project (i.e repository). For example, if your project is located under /home/ramesh/projects/passworddragon, you can make that as your GIT project. First, cd to that directory, and execute git init as shown below.

```
$ cd /home/ramesh/projects/passworddragon  
$ git init  
Initialized empty Git repository in /home/ramesh/projects/passworddragon/.git/
```

This will create a .git directory under your project folder. Following is the content of the .git directory. GIT uses this directory to store information on how it is tracking the changes.

```
$ ls -altr .git  
total 40  
drwxrwxr-x  4 git git 4096 Aug 13 22:39 refs  
drwxrwxr-x  4 git git 4096 Aug 13 22:39 objects  
drwxrwxr-x  2 git git 4096 Aug 13 22:39 info  
drwxrwxr-x  2 git git 4096 Aug 13 22:39 hooks  
-rw-rw-r--  1 git git   23 Aug 13 22:39 HEAD  
-rw-rw-r--  1 git git   73 Aug 13 22:39 description  
-rw-rw-r--  1 git git   92 Aug 13 22:39 config  
drwxrwxr-x  2 git git 4096 Aug 13 22:39 branches  
drwxrwxr-x 36 git git 4096 Aug 13 22:39 ..  
drwxrwxr-x  7 git git 4096 Aug 13 22:39 .
```

Note: If you are sysadmin, who is trying to create a GIT central repository for your company, from where developers can download the projects, you may want to create a username called 'git' and organize all your projects under this account. For example: /home/git/project1, /home/git/project2, etc. Once you have the project organized, cd to the project directory, and do 'git init' from there as git user.

### 4. Add and Commit files to the Project

Once you've initialized the project using "git init", add the files located under this project directory, using "git add".

If there are different types of files under your project directory, and you want GIT to manage only certain types of files, add only those to the GIT as shown below. This example adds only the \*.java and \*.c files.

```
git add *.java  
git add *.c
```

Typically you would like to add all the files under the project directory to the GIT project. Just do "git add .", which will add all the files in the current directory and all the sub-directories to the GIT project.

```
git add .
```

Once you've added the files to the repository, you should commit those files, as shown below.

```
$ git commit -m 'Initial upload of the project'
create mode 100755 PasswordDragon.java

create mode 100755 pwm/ui/DataManager.java

create mode 100755 pwm/ui/PasswordFrame.java
create mode 100755 pwm/tools/StrongEncryption.java
create mode 100755 pwm/tools/PasswordStrength.java
..
```

If you didn't specify your username and email address using "git config" as explained above, you'll get the following error message.

```
$ git commit -m 'Initial upload of the project'
*** Please tell me who you are.
Run
  git config --global user.email "you@example.com"
  git config --global user.name "Your Name"
to set your account's default identity.
```

Omit --global to set the identity only in this repository.

```
fatal: empty ident  not allowed
```

## 5. Make Changes and Commit the File

You've installed GIT, created a project repository, committed all the files to the GIT project. Now it is time to start making some changes to a file and commit it to the repository.

```
vi PasswordDragon.java
```

Once you've modified a file locally, you can view the changes. i.e The difference between your local copy, and the copy already committed in the GIT project using "git diff" as shown below.

```
$ git diff
diff --git a/PasswordDragon.java b/PasswordDragon.java
index 6166ed1..fd82d32 100644
--- a/PasswordDragon.java
+++ b/PasswordDragon.java
@@ -2,7 +2,7 @@
-    public counter=10
+    public counter=55
```

Once you've made modifications to it, reviewed the changes, and happy with it, you can commit the file to GIT repository. This is a two step process. First, you should add the file to the staging area, and commit to the GIT project as shown below.

```
git add PasswordDragon.java
```

above added file will be staged.  
If you want to undo from the staged files.  
**git reset HEAD \***

When you perform commit, it will open your default editor, where you can enter the comment. Once you save your comment and exit the editor, it will commit the file to the GIT project and display the following message.

```
$ git commit
[master 80f10a9] Added password strength meter functionality
 1 files changed, 56 insertions(+), 7 deletions(-)
```

Note: You can also do "git commit -a", which will do the add and commit at the same time.

## 6. View Status and Commit Logs

From your local repository, when you perform "git status", it will display the current status. When the local copy is not changed (or when all the files are committed), you'll see the following message.

```
$ git status
# On branch master
nothing to commit (working directory clean)
```

If you've made changes to a file, and not committed yet, you'll see the following message.

```
$ git status
# On branch master
# Changes not staged for commit:
#   (use "git add ..." to update what will be committed)
#   (use "git checkout -- ..." to discard changes in working directory)
#
#       modified:   PasswordDragon.java
#
no changes added to commit (use "git add" and/or "git commit -a")
```

You can also view the history of a file as shown below.

```
$ git log PasswordDragon.java
commit c919ced7f42f4bc06d563c1a1ea107f2b2420d5
Author: GIT Admin
Date:   Sat Aug 13 22:54:57 2011 -0700

    Added password strength meter functionality

commit c141b7bdbff429de35e36bafb2e43edc655e9957
Author: GIT Admin
Date:   Sat Aug 13 20:08:02 2011 -0700
Initial upload of the project
```

JIRA [<http://vagrantlyhostip:8080>]  
<http://192.168.56.102:8080>

#### Check prerequisites as GIT, java, sudo user settings

- Install git, java jdk and jre
- Create dedicated user for this bitbucket not to facing any conflicts.
- # Allow members of group sudo to execute any command
- %sudo ALL=(ALL:ALL) ALL

#### Installation steps for JIRA

```
[atlbucket@orainux7 bin]$ cd
[atlbucket@orainux7 ~]$ ls
atlassian soft
[atlbucket@orainux7 ~]$ cd /vagrant
[atlbucket@orainux7 vagrant]$ ls
atlassian-bitbucket-5.15.0.tar.gz  git-2.9.5.tar.gz  linuxamd64_12102_database_1of2
linuxx64_12201_database.zip
atlassian-bitbucket-5.15.0-x64.bin  index.html      linuxx64_12201_database          VMs
[atlbucket@orainux7 vagrant]$ cd
[atlbucket@orainux7 ~]$ ls
atlassian soft
[atlbucket@orainux7 ~]$ cd soft
[atlbucket@orainux7 soft]$ ls
atlassian-bitbucket-5.15.0-x64.bin
[atlbucket@orainux7 soft]$ ls
atlassian-bitbucket-5.15.0-x64.bin
[atlbucket@orainux7 soft]$ cp /vagrant/atlassian-jira-software-7.12.3-x64.bin .
[atlbucket@orainux7 soft]$ ./atlassian-jira-software-7.12.3-x64.bin
Unpacking JRE ...
Starting Installer ...
Oct 31, 2018 1:02:21 PM java.util.prefs.FileSystemPreferences$2 run
INFO: Created system preferences directory in java.home.
You do not have administrator rights to this machine and as such, some installation options will not be
available. Are you sure you want to continue?
Yes [y, Enter], No [n]
```

**y**  
This will install JIRA Software 7.12.3 on your computer.

OK [o, Enter], Cancel [c]  
k

This will install JIRA Software 7.12.3 on your computer.

OK [o, Enter], Cancel [c]

Choose the appropriate installation or upgrade option.

Please choose one of the following:

Express Install (use default settings) [1], Custom Install (recommended for advanced users) [2, Enter], Upgrade  
an existing JIRA installation [3]

Where should JIRA Software be installed?  
[/home/atlbucket/atlassian/jira]

Default location for JIRA Software data

```
[/home/atlbitbucket/atlassian/application-data/jira]
```

Configure which ports JIRA Software will use.

JIRA requires two TCP ports that are not being used by any other applications on this machine. The HTTP port is where you will access JIRA through your browser. The Control port is used to startup and shutdown JIRA.

Use default ports (HTTP: 8080, Control: 8005) - Recommended [1, Enter], Set custom value for HTTP and Control ports [2]

Details on where JIRA Software will be installed and the settings that will be used.

Installation Directory: /home/atlbitbucket/atlassian/jira

Home Directory: /home/atlbitbucket/atlassian/application-data/jira

HTTP Port: 8080

RMI Port: 8005

Install as service: No

Install [i, Enter], Exit [e]

Extracting files ...

Please wait a few moments while JIRA Software is configured.

Installation of JIRA Software 7.12.3 is complete

Start JIRA Software 7.12.3 now?

Yes [y, Enter], No [n]

Please wait a few moments while JIRA Software starts up.

Launching JIRA Software ...

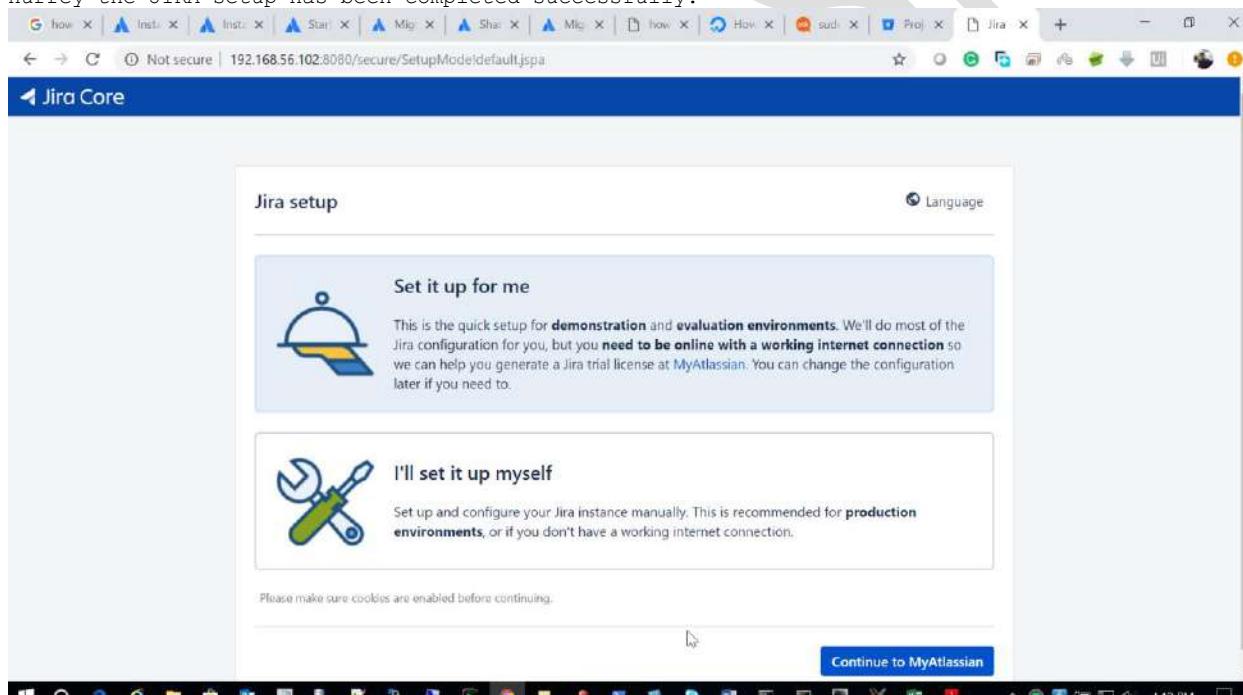
Installation of JIRA Software 7.12.3 is complete

Your installation of JIRA Software 7.12.3 is now ready and can be accessed via your browser.

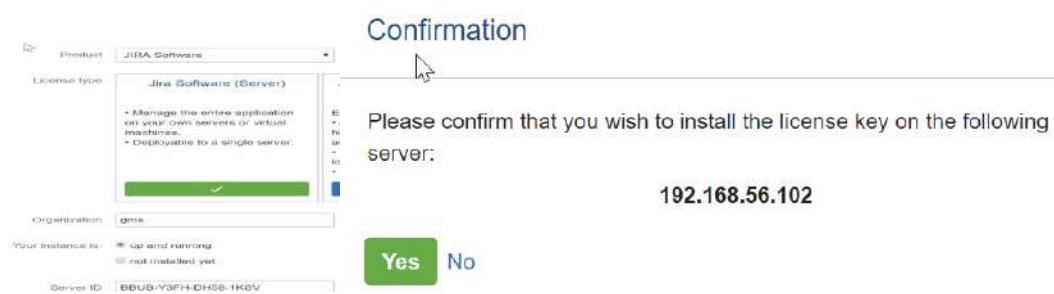
JIRA Software 7.12.3 can be accessed at http://localhost:8080

Finishing installation ...

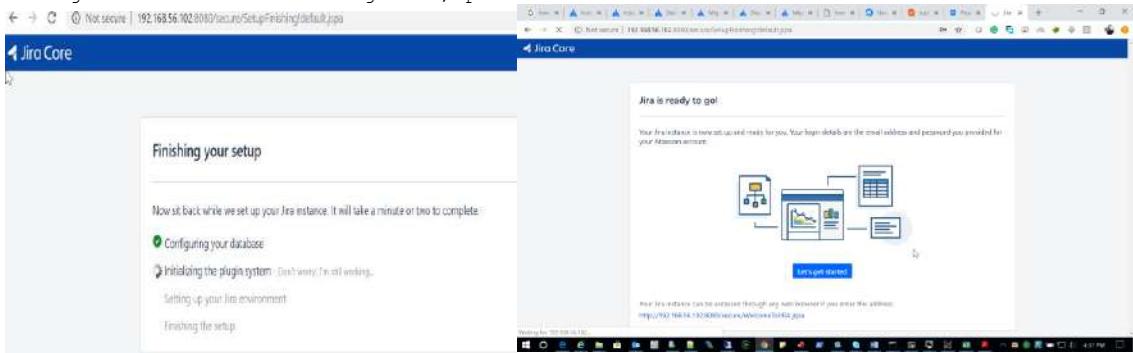
Hurrey the JIRA setup has been completed successfully.



Select with continue to Myatlassian option, by selecting "Set it up for me" or "I'll set it up myself". And select Jira software(server) and go down and select generate license. Click on Yes then.

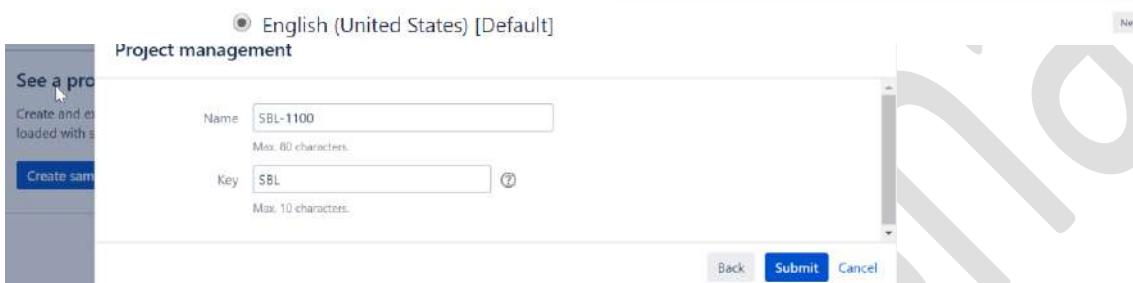


Configure credentials user: gouse , password=oracle and select Get Started.



## Welcome to Jira, gowseshaik@gmail.com!

Before we begin, choose the language you want to use in Jira.



Test it by creating one demo project and will launch the screen as follows.

For more things try to learn JIRA administration.

A screenshot of the Jira Software dashboard for project 'SBL-1100'. The left sidebar includes 'Summary', 'Issues' (selected), 'Reports', 'Project management ideas', 'Working in a project', and 'Add link'. The main area shows 'Open issues' with tasks like 'SBL-1', 'SBL-2', etc. A tooltip on the right side of the screen provides information about time zone synchronization.

## BitBucket

### Check prerequisites as GIT, java, sudo user settings

- Install git, java jdk and jre
  - Create dedicated user for this bitbucket not to facing any conflicts.
- 
- # Allow members of group sudo to execute any command
  - %sudo ALL=(ALL:ALL) ALL

For more steps please find the attachment of practical setup



MobaXterm\_bitbucke

t\_20181031\_150906:

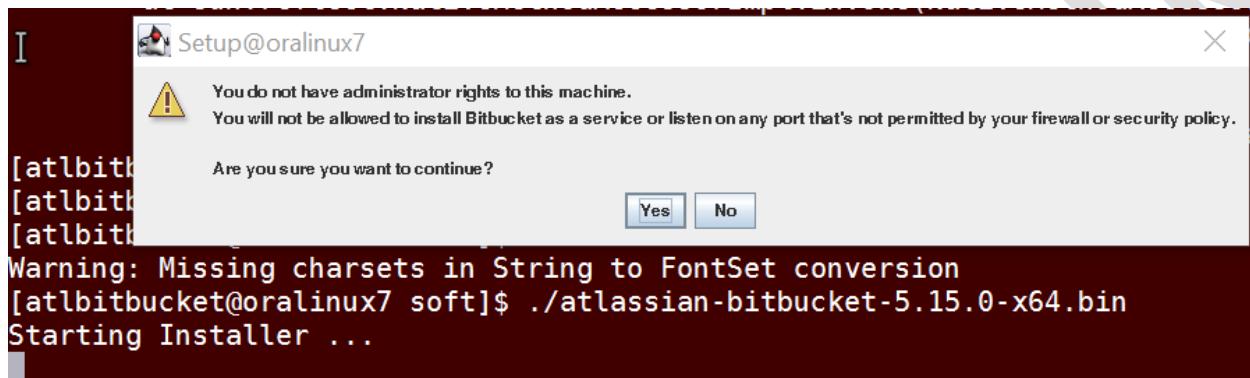
<https://confluence.atlassian.com/bitbucketserver/bitbucket-server-installation-guide-867338382.html>

<https://comtronic.com.au/how-to-install-atlassian-bitbucket-server/>

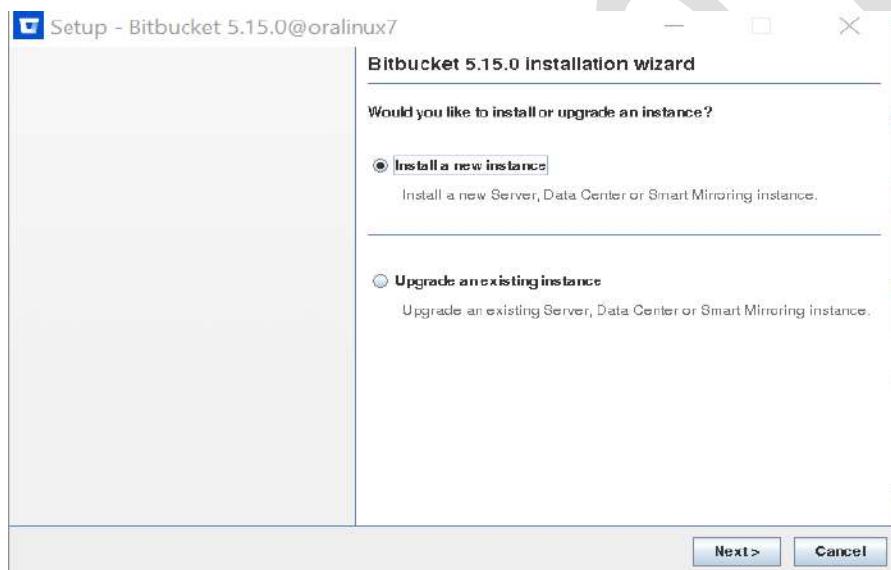
Set the display settings for GUI installation:

```
[atlbitbucket@orlinux7 soft]$ export DISPLAY=10.66.113.46:0.0
[atlbitbucket@orlinux7 soft]$ xclock
Warning: Missing charsets in String to FontSet conversion
```

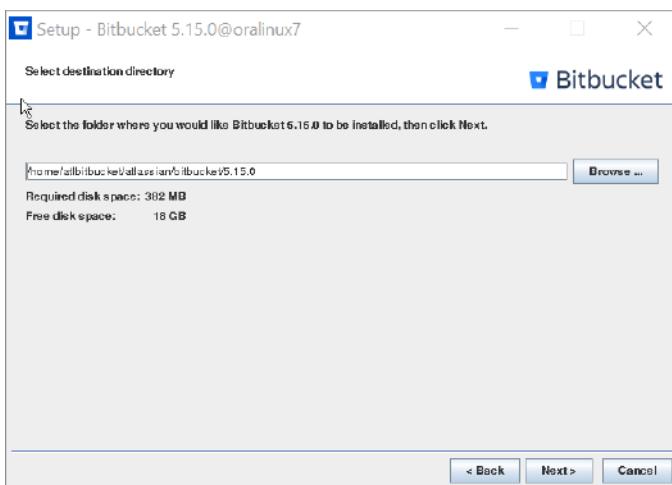
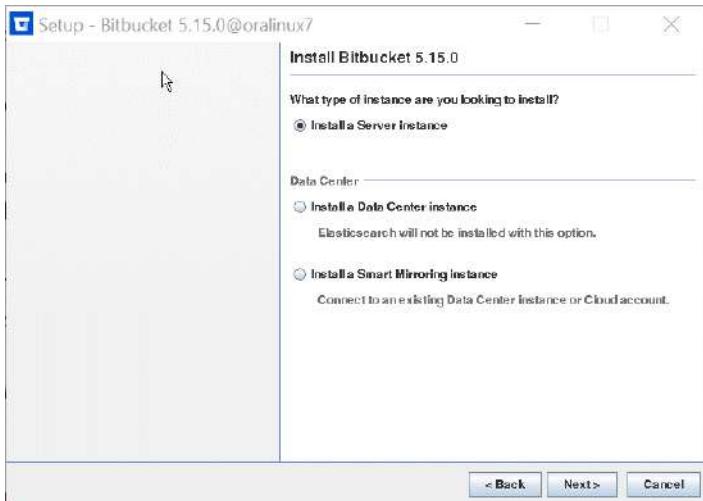
- Error for administration rights



Clicked on Yes, select "Install a new instance"

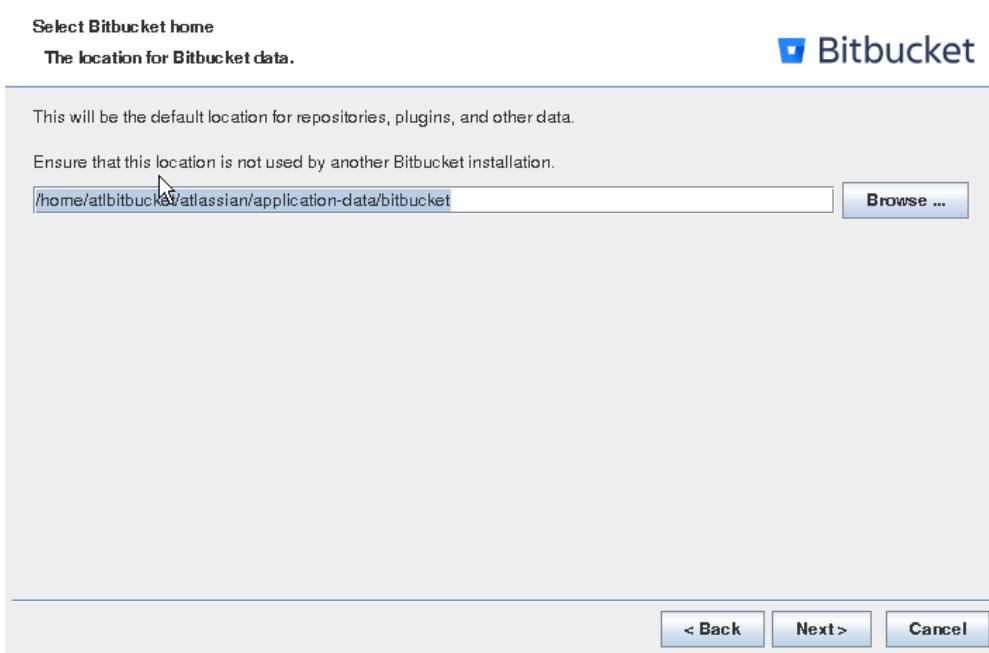


Select "Install a Server instance", click Next.



Home directory for the bitbucket : /home/atlassian/bitbucket5.15.0 , click on next.

/home/atlassian/application-data/bitbucket , click on next



Use default port is "http port :7990", click on Next

## Configure TCP Ports



Bitbucket requires a TCP port that isn't being used by other applications.

The HTTP port is where users access Bitbucket through their browsers.

Bitbucket also requires ports 7992 and 7993 are available to run an embedded Elasticsearch instance that provides search functionality to Bitbucket.

Use default HTTP port (7990) - Recommended

Set custom value for the HTTP port

< Back Next > Cancel

## Installation Summary

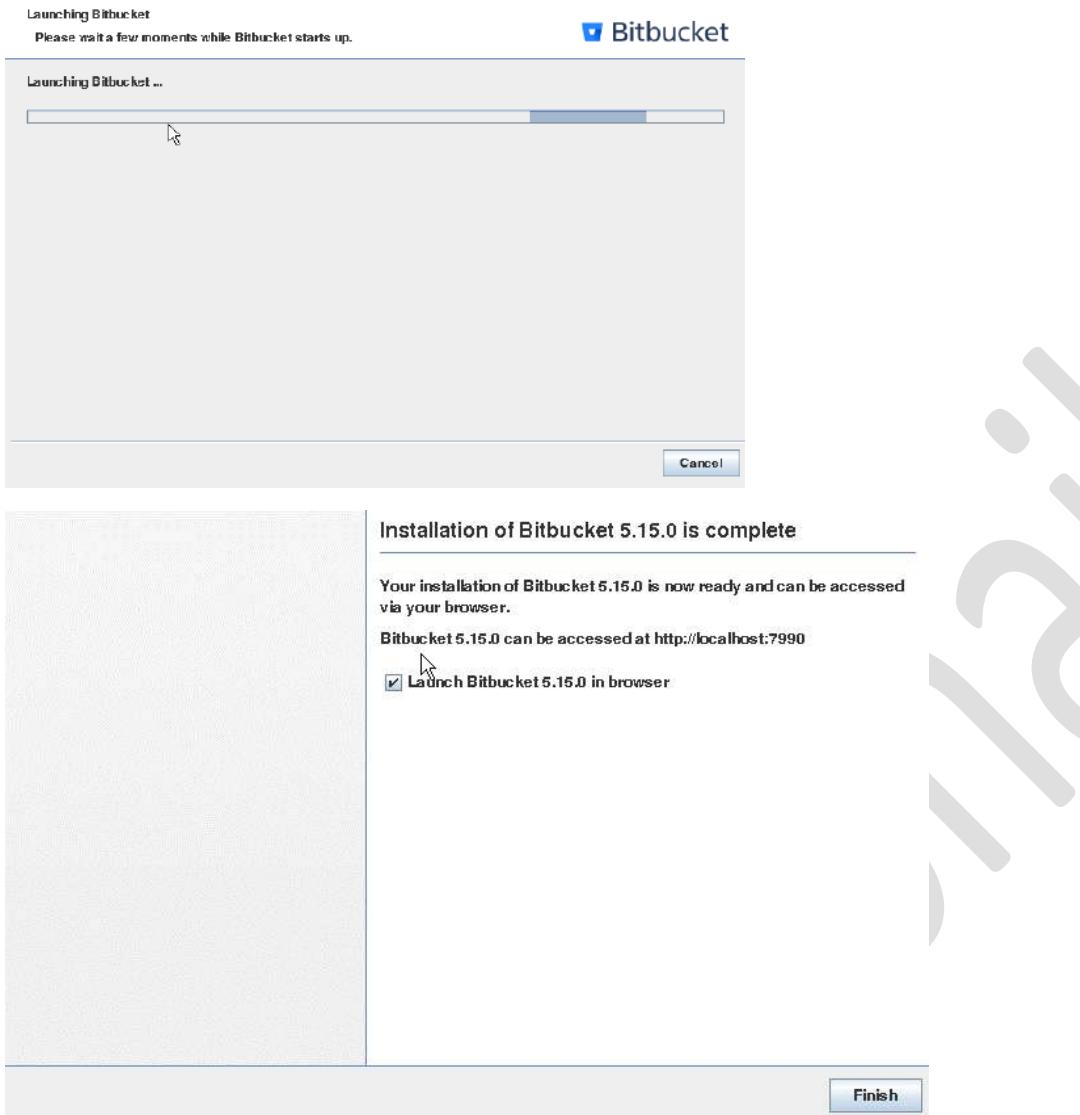


|                         |                                                          |
|-------------------------|----------------------------------------------------------|
| Installation Directory: | /home/atlibitbucket/atlassian/bitbucket/5.15.0           |
| Home Directory:         | /home/atlibitbucket/atlassian/application-data/bitbucket |
| HTTP Port:              | 7990                                                     |
| Install as a service:   | No                                                       |

< Back Next > Cancel

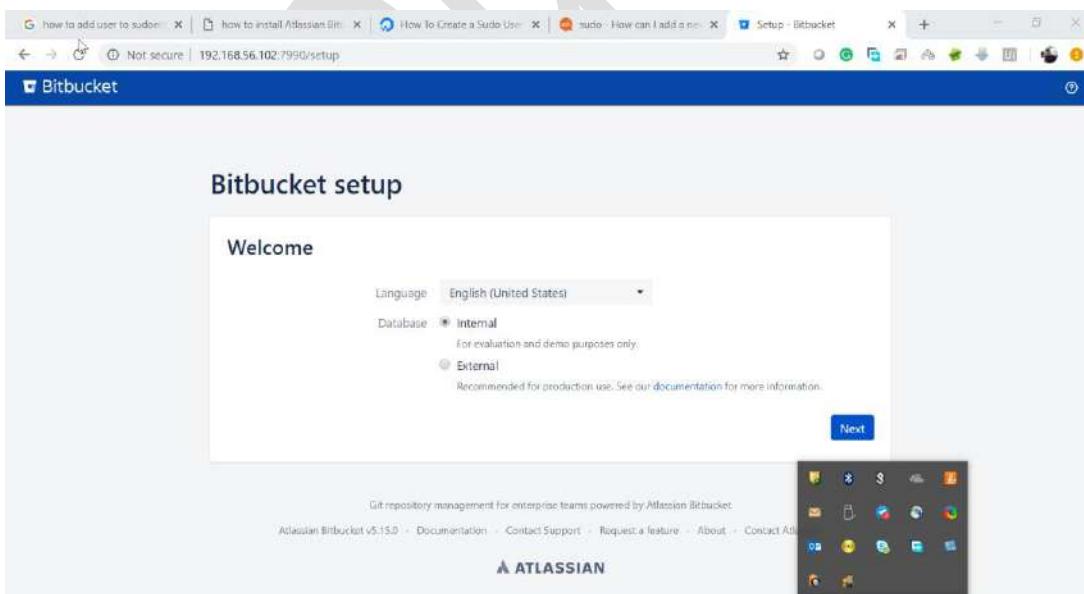
I can see the setup has been installed without service. Click on next.

The next prompt will ask as "Would you like to launch" for this click yes.



Launch the bitbucket from your browser "http://localhost:7990"

Get the vagrant ip and launch the url as follows. <http://192.168.56.102:7990/>



- Click next, by selecting option "Internal" as a demo purposes.

Licensing and settings

Application title\* Bitbucket

Base URL\* http://192.168.56.102:7990

All links created by Bitbucket (for emails, etc.) will be prefixed by this URL.

Server ID BK89-S3RS-3D6A-MDS3

License key\*  I need an evaluation license  
 I have a Bitbucket license key

Git repository management for enterprise teams powered by Atlassian Bitbucket

Atlassian Bitbucket v5.15.0 - Documentation - Contact Support - Request a feature - About - Contact Atlassian

**ATLASSIAN**

- Select License Key as "Evaluation License"
- And create an account if you don't have an account in atlassian.

### Generate license

Log in to your existing Atlassian account, or create a new one

[I have an account](#) [Create an account](#)

Get the authenticate with google account for getting the license key.

Administrator account setup

Username\* gouse

Full name\* gouse

Email address\* gouse@hikashield@gmail.com

Password\* \*\*\*\*\*

Confirm password\* \*\*\*\*\*

[Integrate with Jira](#) [Go to Bitbucket](#)

Git repository management powered by a low Niantic Blasted evaluation license

Atlassian Bitbucket v5.15.0 - Documentation - Contact Support - Request a feature - About - Contact Atlassian

Once activated, click on next and select the option for go to Bitbucket as of now I don't have installed jira on vagrant. So directly go to Bitbucket.

Username: gouse  
 Password: oracle

The screenshot shows a Bitbucket login page with a yellow background overlay. The overlay contains the text: "Hurry, we have done the setup of bitbucket successfully. Try to login." Below the overlay, the Bitbucket interface is visible, showing a pull request titled "Bugfix/TIB-57 buttons need to be red".

<http://192.168.56.102:7990/>

## BAMBOO

### Bamboo Best Practice - System Requirements

**Note** that Atlassian currently only supports Bamboo on x86 and 64 bit x86 derived hardware platforms.

<https://confluence.atlassian.com/bamboo/bamboo-best-practice-system-requirements-388401170.html>

| User scenario                          | Usage profile                               | Bamboo server    |
|----------------------------------------|---------------------------------------------|------------------|
| <b>Individual user/<br/>Small team</b> | 10 - 20 plans<br>Little concurrent building | 4 core, 4 GB RAM |

|                                                      |                                                                                 |                                       |
|------------------------------------------------------|---------------------------------------------------------------------------------|---------------------------------------|
|                                                      | Light server use                                                                |                                       |
| <b>Medium team</b>                                   | 10 - 20 plans<br>Medium concurrency<br>Light server use                         | 8 core, 8 GB RAM, remote agent use    |
| <b>Multiple small teams/<br/>Large team</b>          | 20 - 100s plans<br>Plan branches<br>High concurrency<br>Medium server use       | 8 core, 16 GB RAM, more remote agents |
| <b>Multiple large teams/<br/>Department/Division</b> | 1000s of plans<br>Frequent plan branches<br>High concurrency<br>High server use | 16 core, 16 GB RAM, all remote agents |

<https://confluence.atlassian.com/bamboo/installing-bamboo-on-linux-289276792.html>

As discussed, facing this issue from last 2 weeks and Still there is no updates/fix on below issue, waiting for an update and we need to identify the root cause why the installed bamboo software cleaned up from the server “SPRL013”. And after reinstalling the BAMBOO software, we are facing an issue at plan creation.

Configure plan - Create plan - A... +

Not secure | sprl013:8085/build/admin/create/newPlan.action?existingProjectKey=MAV

Plan access  Allow all users to view this plan

**Link repository to new build plan**

Repository host  Link new repository

Bitbucket Server / Stash Bitbucket Server / Stash

Display name Tafeely

Bitbucket Server / Stash details

Server Bitbucket

Repository TafeelyCode / prm ?

A new public key for the selected repository will be stored as a repository access key.

Branch git master

Repository access  Allow all users to reuse the configuration of this repository  
 Only you are allowed to reuse the configuration of this repository  
 None

Configure plan Cancel

Configure plan - Create plan - A... +

Not secure | sprl013:8085/build/admin/create/createPlan.action

Bamboo My Bamboo Projects Build Deploy Reports Create Search Create

**Create plan**

Configure plan Configure job

**Configure plan** How to create a build plan

Your build plan defines everything about your build process. Each plan has a Default job when it is created. More advanced configuration options, including those for apps, and the ability to add more jobs will be available to you after creating this plan.

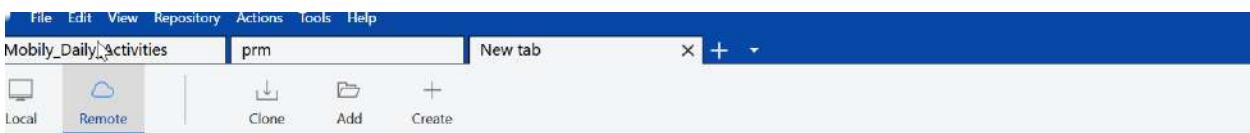
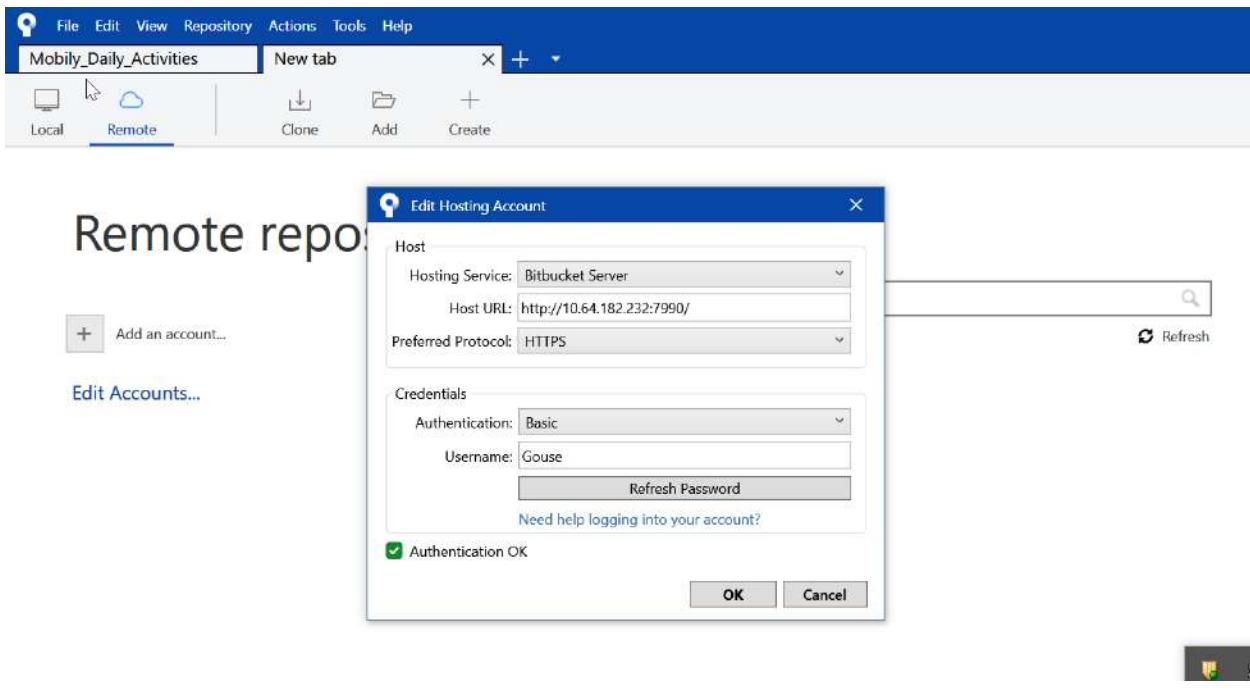
! • New access key is required but could not be added to Bitbucket Server.  
• ssh://git@10.64.182.232:7999/tafcd/prm.git: Connection timed out (Connection timed out)

Project and build plan name

Issue resolved by firewall team on **Port 7999 is not enable in Bamboo server and we raised a request to Firewall team to open the port, once it is done we will configure the bamboo Plans.**

### Bitbucket to Sourceeee configuration:

Go to Add tab + button and provide the hosting account details and get the repositories and clone it in local.



## Remote repositories

The screenshot shows the Bamboo interface with a 'Remote' tab selected. The dashboard lists several remote repositories under the user 'Gouse': 'Mobily-App-Android', 'CICD\_MobileApp', 'TafeelyDocuments', 'Tafeely-fixed', and 'prm'. Each repository entry includes a 'Clone' link.

Atlassian CLI

<https://bobswoosh.atlassian.net/wiki/spaces/BCLI/pages/31916685/Installation+and+Use>

## How to connect Bamboo and Bitbucket Server

### Installation and Configuration

We'll start with a clean install of Bamboo and Bitbucket Server. You can download them here:

- [Atlassian Bamboo](#)
- [Atlassian Bitbucket Server](#)

And simply follow the install instructions:

- [Bamboo installation instructions](#)
- [Bitbucket Server installation instructions](#)

For demo and testing purposes the embedded database install is perfect. Be aware that you might need to install certain prerequisites first!

Once you've installed both the tools we need to connect them, for this we need to set-up an application link. Take a look at [Integrating Bamboo with Bitbucket Server](#) to see how you can set this up.

### Task was canceled

#### Error:

Task was canceled - 'Checkout Default Repository(1)' of type com.atlassian.bamboo.plugins.vcs:task.vcs.checkout.  
error 26-Sep-2019 13:39:38 Failed to prepare the build 'Gouse\_BRM - Supp\_Service - Default Job #2 (GS-BRMSUP-JOB1-2)'

Solution:

Disable the default *Source Code Checkout* task

And run.

Above issue resolved and able to checkout. But facing an issue as in correct directory.

```
simple 26-Sep-2019 13:58:10 Executing build Gouse_BRM - Supp_Service - MavenJOB #8 (GS-BRMSUP-JOB1-8)
simple 26-Sep-2019 13:58:10 Skipping execution of task 'Checkout Default Repository' as it is disabled
simple 26-Sep-2019 13:58:10 Running pre-build action: VCS Version Collector
simple 26-Sep-2019 13:58:10 Skipping execution of task 'Checkout Default Repository' as it is disabled
simple 26-Sep-2019 13:58:10 Starting task 'Supp_Maven_clean' of type
'com.atlassian.bamboo.plugins.maven:task.builder.mvn3'
```

```
build 26-Sep-2019 13:58:15 [INFO] -----
build 26-Sep-2019 13:58:15 [INFO] Total time: 0.251 s
build 26-Sep-2019 13:58:15 [INFO] Finished at: 2019-09-26T07:58:15+03:00
build 26-Sep-2019 13:58:15 [INFO] Final Memory: 2m/659k
build 26-Sep-2019 13:58:15 [INFO] 
[ERROR] The goal you specified requires a project to execute but there is no POM in this directory (/export/home/bradmin/bamboo-agent-home/xml-data/build-dir/GS-BRMSUP-JOB1). Please verify you invoked Maven from the correct directory. -> [Help 1]
build 26-Sep-2019 13:58:15 [ERROR] 
build 26-Sep-2019 13:58:15 [ERROR] To see the full stack trace of the errors, re-run Maven with the -e switch.
build 26-Sep-2019 13:58:15 [ERROR] Re-run Maven using the -X switch to enable full debug logging.
build 26-Sep-2019 13:58:15 [ERROR] 
build 26-Sep-2019 13:58:15 [ERROR] For more information about the errors and possible solutions, please read the following articles:
build 26-Sep-2019 13:58:15 [ERROR] [Help 1] http://cwiki.apache.org/confluence/display/MAVEN/MissingProjectException
simple 26-Sep-2019 13:58:15 Falling task since return code of [/export/home/bradmin/maven325/bin/mvn --batch-mode -Djava.io.tmpdir=/var/tmp/GS-BRMSUP-JOB1 clean] was 1 while expected 0
simple 26-Sep-2019 13:58:15 Finished task 'Supp_Maven_clean' with result: Failed
simple 26-Sep-2019 13:58:15 Running next build plan in "Docker container cleanup"
```

```
build 26-Sep-2019 13:58:15 [INFO] -----
build 26-Sep-2019 13:58:15 [ERROR] The goal you specified requires a project to execute but there is no POM in this directory (/export/home/bradmin/bamboo-agent-home/xml-data/build-dir/GS-BRMSUP-JOB1). Please verify you invoked Maven from the correct directory. -> [Help 1]
build 26-Sep-2019 13:58:15 [ERROR] 
build 26-Sep-2019 13:58:15 [ERROR] To see the full stack trace of the errors, re-run Maven with the -e switch.
build 26-Sep-2019 13:58:15 [ERROR] Re-run Maven using the -X switch to enable full debug logging.
build 26-Sep-2019 13:58:15 [ERROR] 
build 26-Sep-2019 13:58:15 [ERROR] For more information about the errors and possible solutions, please read the following articles:
```

```
build 26-Sep-2019 13:58:15 [ERROR] [Help 1]
http://cwiki.apache.org/confluence/display/MAVEN/MissingProjectException
simple 26-Sep-2019 13:58:15 Failing task since return code of [/export/home/bradmin/maven325/bin/mvn --batch-
mode -Djava.io.tmpdir=/var/tmp/GS-BRMSUP-JOB1 clean] was 1 while expected 0
simple 26-Sep-2019 13:58:15 Finished task 'Supp_Maven_clean' with result: Failed
```

install git as well on solaries agent to download the repository build from bitbucket.

## Get some data into your tools.

Now you can import a test project into your Bitbucket Server. I choose the [run-bucket-run](#) project from my colleague [Tim Pettersen](#) which is hosted on Bitbucket Cloud.

Use these instructions to clone the repo to your local instance: [Tip of the Week – How to move a full Git repository.](#)

Once this has been done we can create a Bamboo build plan for our Bitbucket repo.

Go to Bamboo start creating a new plan, here is my configuration:

Due to the fact that Bitbucket Server and Bamboo are linked I can immediately select my repository.

The next step is adding [jobs and tasks](#) to the build plan.

For the purpose of this example I simply do a source code checkout:

Bamboo My Bamboo Build Deploy Reports Create Search

Create a new plan

Configure plan Configure tasks

### Configure tasks

Each plan has a default job when it is created. In this section, you can configure the Tasks for this plan's default job. You can add more jobs to this plan once the plan has been created.

A task is an operation that is run on a Bamboo working directory using an executable. An example of task would be the execution of a script, a shell command, an Ant Task or a Maven goal. Learn more about tasks.

**Source Code Checkout**

Checkout Default Repository

Final tasks Are always executed even if a previous task fails.  
Drag tasks here to make them final

Add task

**Source Code Checkout configuration**

Task description  
Checkout Default Repository

Disable this task

You can check out one or more repositories with this Task. You can choose to check out the Plan's Default Repository or specify a Specific Repository. You can add additional repositories to this Plan via the Plan configuration.

Repository\*  
Bitbucket Cloud - Towl Repo

Default always points to Plans default repository.

Checkout Directory

(Optional) Specify an alternative sub-directory to which the code will be checked out.

Force Clean Build  
Removes the source directory and checks it out again prior to each build. This may significantly increase build times.

Add repository

Save Cancel

**Enable this plan?**

Yes please!  
By selecting this option your plan will be available for building and change detection straight away. do not select this option if you have advanced configuration changes to make after creation.

Create Cancel

Continuous integration powered by Atlassian Bamboo version 5.10.1.1 build 51018 - 08 Feb 16  
Report a problem · Request a feature · Contact Atlassian · Contact Administrators

Atlassian

## Before clicking on create be sure to activate your build plan!

Immediately a first build will be triggered and displayed in Bamboo:

Bamboo My Bamboo Build Deploy Reports Create Search

Build projects / Towl - Cloud

**Tip of the Week - Cloud Repo Build**

Building our Cloud Repo, one build at a time...

Plan summary Recent failures History Tests

Showing Last 25 builds

**Plan summary**

Current activity  
No builds are currently running.

Recent history

| ①                                              | First build for this plan | 2 minutes ago | Testless build |
|------------------------------------------------|---------------------------|---------------|----------------|
| Latest build                                   | Last successful build     |               |                |
| Feed for all builds or just the failed builds. |                           |               |                |

**Plan statistics**

1 builds  
100% successful  
< 1s average duration

Continuous integration powered by Atlassian Bamboo version 5.10.1.1 build 51018 - 08 Feb 16  
Report a problem · Request a feature · Contact Atlassian · Contact Administrators

Atlassian

This build plan will be triggered every time you commit something to the repository.

You will also see the build result on each commit in Bitbucket Server. And be able click through to the detailed build results in Bamboo:

| Author             | Commit      | Message                                                      | Commit date | Builds                 |
|--------------------|-------------|--------------------------------------------------------------|-------------|------------------------|
| Thomas A. Anderson | 40018f21a19 | Added Stuff on Master                                        | 15 Jan 2016 | <a href="#">Builds</a> |
| Thomas A. Anderson | ff29e07d1fe | second totw commit                                           | 15 Jan 2016 | <a href="#">Builds</a> |
| Thomas A. Anderson | 6031510d262 |                                                              | 15 Jan 2016 | <a href="#">Builds</a> |
| Tim Petersen       | 00d56a52074 | TotW Project - TotW Build Plan                               | 4 mins ago  | <a href="#">Builds</a> |
| Tim Petersen       | 1add1e33a0  | #1 successful in < 1 second                                  |             | <a href="#">Builds</a> |
| Tim Petersen       | 8202fb0294b |                                                              |             | <a href="#">Builds</a> |
| Tim Petersen       | 7ed9c331b3f |                                                              |             | <a href="#">Builds</a> |
| Tim Petersen       | 1a66156ea3c |                                                              |             | <a href="#">Builds</a> |
| Tim Petersen       | 4558e3ba2bd | Favicon                                                      |             | <a href="#">Builds</a> |
| Tim Petersen       | 802ae930361 | Add GA tracking script.                                      |             | <a href="#">Builds</a> |
| Tim Petersen       | 7fa2010c0e  | Limit elapsed time in case the user tabs away from the game. |             | <a href="#">Builds</a> |
| Tim Petersen       | 95a79210523 | All documentation.                                           |             | <a href="#">Builds</a> |
| Tim Petersen       | 973ba268720 | Handle level end.                                            |             | <a href="#">Builds</a> |
| Tim Petersen       | ec557b47635 | Pressing F enters/demo mode.                                 |             | <a href="#">Builds</a> |
| Tim Petersen       | 612dab1e39  | Skip splash and end screen in demo mode.                     |             | <a href="#">Builds</a> |

And you will be able to see all commits in a build inside Bamboo. Clicking it will instantly bring you to the relevant commit or diff in BitBucket Server.:

| Author             | Commit      | Message     | Commit date  |
|--------------------|-------------|-------------|--------------|
| Thomas A. Anderson | 164ffcc8... | totw commit | A moment ago |

This is only the beginning! Let me show you what [Plan Branches](#) can do for you.

Configuring plan branches is as easy as going to your build plan configuration, go to the branches tab and selecting **Create plan branches for all new branches** in the dropdown next to the label **New Branches**:

Now Bamboo will create a plan for each new branch it will detect in your repository. This build plan will be a copy of your original build plan but will have its own logs, artifacts and triggers. Out of the box it will be triggered by a commit on the branch itself. So let's create a new branch **feature-totw** to test this out. Here is how Bamboo looks like after we've created the new Branch:

As you can see we can now select 2 branches (Master and feature-totw), this is what we see when feature-totw is selected:

**Plan summary**

**Current activity**  
No builds are currently running.

**Recent history**

| Build # | Description               | Completed     | Status         |
|---------|---------------------------|---------------|----------------|
| ① #1    | First build for this plan | 2 minutes ago | Testless build |

[Latest build](#) | [Last successful build](#) | [Feed for all builds or just the failed builds.](#)

**Plan statistics**

- 1 builds
- 100% successful
- < 1s average duration

Continuous integration powered by Atlassian Bamboo version 5.10.1.1 build 61016 - 08 Feb 16  
[Report a problem](#) | [Request a feature](#) | [Contact Atlassian](#) | [Contact Administrators](#)

**Atlassian**

It's a complete build plan overview for this new branch. If you have several branches at the same time you can even get a nice overview off how their builds are going in Bamboo:

**Plan summary**

**Branches**

| Plan         | Build | Completed     | Tests          | Reason                    |
|--------------|-------|---------------|----------------|---------------------------|
| bugfix-totw  | ① #1  | 1 minute ago  | No tests found | First build for this plan |
| feature-totw | ② #1  | 6 minutes ago | No tests found | First build for this plan |

**Atlassian**

This is very cool if you are working with a big team and using feature branches, you can immediately see which features are breaking their builds.

And this is not everything plan branches can do for you. If have been paying attention you would also have noticed the **Merging** title on the plan branches configuration tab. This option will give you the possibility to either update your branch with the latest commits from another branch (**Branch Updater**) or push your changes to another branch (**GateKeeper**). Both of these actions will happen after a successful build.

Bamboo My Bamboo Build Deploy Reports Create

Build projects / Totw - Cloud / Tip of the Week - Cloud Repo Build Configuration - Tip of the Week - Cloud Repo Build

Building our Cloud Repo, one build at a time...

**Plan Configuration**

- Stages & jobs 1
- Branches 2

Plan details Stages Repositories Triggers Branches Dependencies Permissions Notifications Variables Miscellaneous Audit log

**Branches**

Plan branches allow you to run builds across different branches in your source repository using the same plan configuration.

**Automatic branch management**

Plan branches can be created and deleted automatically based on branch creation and deletion in the primary source repository.

|                   |                                                              |
|-------------------|--------------------------------------------------------------|
| New branches      | Create plan branches for all new branches                    |
| Deleted branches  | Delete plan branches after a period of time<br>7 days        |
| Inactive branches | Delete plan branches after a period of inactivity<br>30 days |

**Merging**

Automatic merging can test the merge between branches and push changes back to the repository on a successful build. This setting will be applied to all new plan branches.

Branch merging enabled

|                                                                                          |                                                                                                   |
|------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------|
| <b>Branch updater</b>                                                                    | <b>Gatekeeper</b>                                                                                 |
| Checkout <input type="button" value="D Current branch"/>                                 | Checkout <input type="button" value="Tip of the Week - Cloud Repo Build"/>                        |
| Merge from <input type="button" value="Tip of the Week - Cloud Repo Build"/>             | Merge from <input type="button" value="D Current branch"/>                                        |
| Build Merge result                                                                       | Build Merge result                                                                                |
| Push on <input checked="" type="radio"/> <input type="button" value="D Current branch"/> | Push on <input type="radio"/> <input type="button" value="D Tip of the Week - Cloud Repo Build"/> |

**Notifications**

Notification preferences are applied to all new plan branches.

- Notify committers and people who have favourited the created branch  
All committers and people who have favourited the branch will be notified for all build failures and the first successful build.
- Use the plan's notification settings  
Use the same notification rules as configured for this plan
- Notifications should not be sent for the created branch  
No notifications will be sent for the created branch.

**Triggers**

How new plan branches should be triggered.

Branch triggers: Same as defined in parent plan

**Save Cancel**

Continuous integration powered by Atlassian Bamboo version 5.10.1.1 build 91018 - 08 Feb 16

Report a problem Request a feature Contact Atlassian Contact Administrators

Atlassian

These provide you with some extra tools to make working with feature branches so much easier. Simply use Branch Updater to keep your feature branch up-to-date with the latest changes on Master and you will never have to worry about suddenly breaking a build when you merge your feature into Master.

Here is the result of Branch Updater in Bamboo:



Build projects / TotW Project / TotW Build Plan

**Build #3** feature-totw

My Tip of the Week Build Plan

#3 was successful – Changes by Atlassian Bamboo <bamboo@example.com>

Branch summary Tests Commits Artifacts Logs Metadata

### Build result summary

#### Details

Completed 10 Feb 2016, 10:12:14 PM – 2 minutes ago

Duration < 1 second

Labels None

Show more

Write a comment...

#### Code commits

| Author                                | Commit     | Message                                                                               | Commit date |
|---------------------------------------|------------|---------------------------------------------------------------------------------------|-------------|
| Atlassian Bamboo <bamboo@example.com> | 6eba49c... | [bamboo] Automated branch merge (from master:a2482fdc1822f85d208b6014d2adb0cc23011b8) | 2 mins ago  |
| Thomas A. Anderson                    | a2482fd... | commit to master                                                                      | 37 mins ago |

Continuous integration powered by Atlassian Bamboo version 6.9.7 build 5920 - 14 Oct 16

Report a problem · Request a feature · Contact Atlassian · Contact Administrators

Atlassian

And here is what you see in Bitbucket Server:

Tip of the Week / run-bitbucket-run

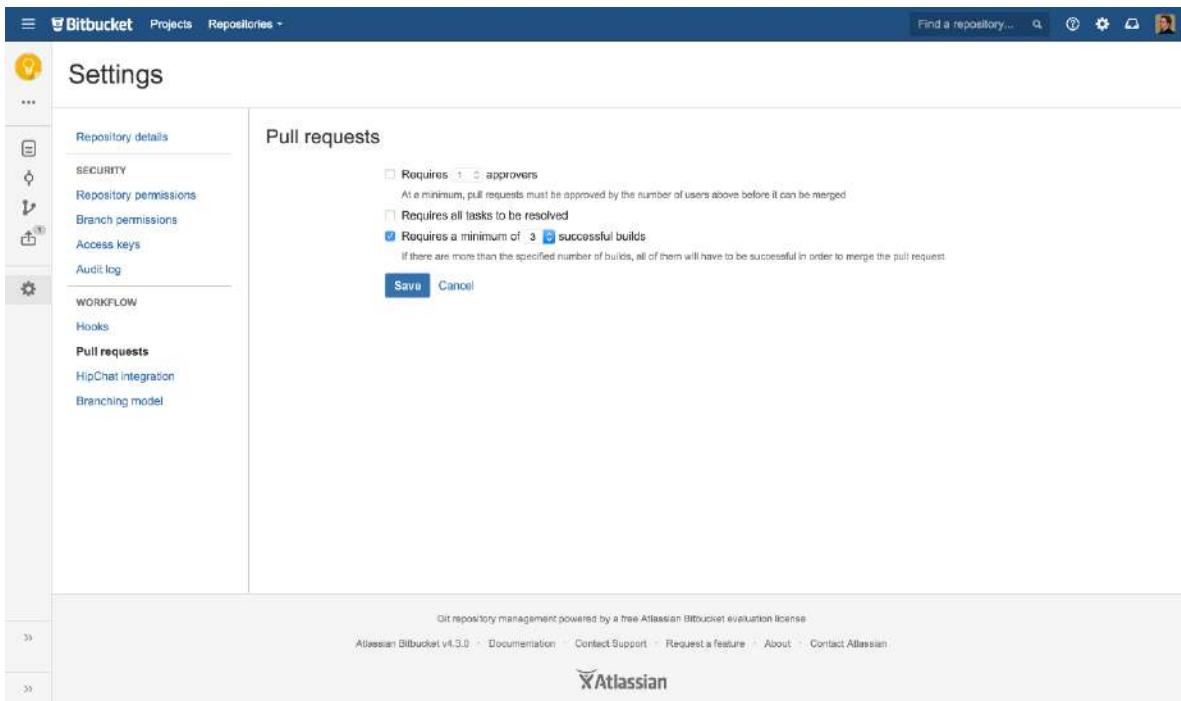
Commits

feature-totw ...

| Author             | Commit         | Message                                                                               | Commit date | Builds |
|--------------------|----------------|---------------------------------------------------------------------------------------|-------------|--------|
| Atlassian Bamboo   | 6eba49cc40b... | [bamboo] Automated branch merge (from master:a2482fdc1822f85d208b6014d2adb0cc23011b8) | 2 mins ago  |        |
| Thomas A. Anderson | 28f8aeb5d78... | commit to feature branch                                                              | 2 mins ago  |        |

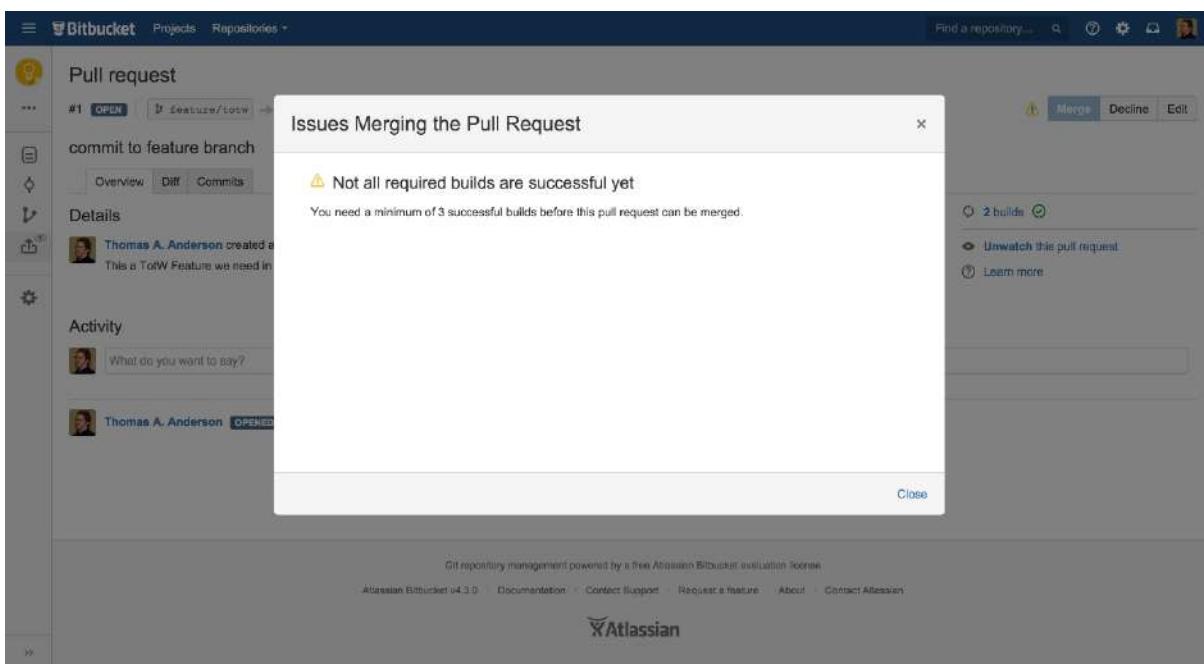
So this is all Bamboo, but what about Bitbucket Server? You already saw the build result that are available in Bitbucket Server and the automatic merges created by Bamboo. But what else is there?

You can use successful builds as a requirement in Pull Requests! Simply go to your repository settings and select the Pull requests menu item. Here you can set the number of successful builds a pull request needs before it can be merged.



The screenshot shows the Bitbucket Settings page for a repository. On the left, there's a sidebar with icons for Repository details, SECURITY, Repository permissions, Branch permissions, Access keys, Audit log, WORKFLOW, Hooks, and Pull requests. Under Pull requests, it lists HipChat integration and Branching model. The main content area is titled "Pull requests" and contains configuration options: "Requires 1 approvers" (unchecked), "Requires all tasks to be resolved" (unchecked), and "Requires a minimum of 3 successful builds" (checked). Below these are "Save" and "Cancel" buttons. At the bottom of the page, there's a footer with Atlassian branding and links to Documentation, Contact Support, Request a feature, About, and Contact Atlassian.

And if you try to merge a pull request before you have the right number of successful builds you'll see this:



The screenshot shows a Bitbucket pull request page for a pull request titled "#1 OPEN feature/total". A modal dialog box is open, titled "Issues Merging the Pull Request". It contains a warning message: "⚠ Not all required builds are successful yet" and "You need a minimum of 3 successful builds before this pull request can be merged." There are buttons for "Merge", "Decline", and "Edit". To the right of the dialog, there's a sidebar with "2 builds" (one green, one red), "Unwatch this pull request", and "Learn more". The footer of the page includes Atlassian branding and links to Documentation, Contact Support, Request a feature, About, and Contact Atlassian.

So this helps you again to be absolutely sure that anything you will merge into Master will not break your Master Build and will have no impact on your releases.

Share your own tips and tricks in the comments or on Twitter, be sure to mention me: [@pvdevoor](#) or [@atlassiandev](#) !

## Bamboo Specifying Global or Build-specific Variables

<https://confluence.atlassian.com/bamboo/bamboo-variables-289277087.html>

```
cp fm_inv_pol_custom.so /export/home/bradmin/BRM/Artifacts/Opcde/fm_inv_pol_custom-SIT-  
${bamboo.buildNumber}.so
```

Not sure how \${bamboo.shortPlanName} gives you the branch name, that's the name of the project or build plan. But I guess if it gets what you wanted.

We obtain the branch name from \${planRepository.branchName} or \${planRepository.branch}.

Both of those variables give the "real" branch name containing the slash (feature/whatever, not feature-whatever as displayed by Bamboo). I'm by no means suggesting that behavior is wrong, it's just not what I want in this case.

I got branch name with \${bamboo.planRepository.branchName}, not \${planRepository.branch}

```
cd /export/home/bradmin/.m2/repository/BillingPortalAddLineAdapter/BillingPortalAddLineAdapter/0.0.1-SNAPSHOT/  
cp BillingPortalAddLineAdapter-0.0.1-SNAPSHOT.jar BillingPortalAddLineAdapter-${bamboo.buildNumber}-${bamboo.planRepository.branchName}.jar  
cksum BillingPortalAddLineAdapter.jar > /export/home/bradmin/BRM/Artifacts/checksum_BillingPortalAddLineAdapter.txt
```

Just a note, the slash "/" can be set to a dash "-" in the branch. That's configured from Stash (Branching model under repository Settings).

Go to Bitbucket:

The screenshot shows the Bitbucket repository settings page for a project named 'BILL\_Pj AddLinePortalAdapter'. The left sidebar has a red box around the 'Repository settings' link. The main content area is titled 'Branching model' with a sub-section 'Project settings inheritance'. A red box highlights the radio button 'Use custom settings'. Below this, there are two sections: 'Development' and 'Production'. Under 'Development', the radio button 'Use default branch' is selected. Under 'Production', the radio button 'No production branch' is selected. The 'Branch prefixes' section at the bottom has four entries: 'Bugfix' (checkbox checked), 'Feature' (checkbox checked), 'Hotfix' (checkbox checked), and 'Release' (checkbox checked). Red boxes highlight the 'bugfix/' input field, the 'feature/' input field, the 'hotfix/' input field, and the 'release-' input field. At the bottom of the page, a terminal window shows the command: '-rwxr-xr-x 1 bradmin bradmin 2650908 Feb 19 18:49 fm\_inv\_pol\_custom-SIT-88.so'.

## Changing Your Shell with `chsh`

```
[ejr@hobbes ejr]$ cat /etc/shells  
/bin/bash  
/bin/sh  
/bin/tcsh  
/bin/csh  
/bin/zsh  
[ejr@hobbes ejr]$ chsh  
Changing shell for ejr.
```

## Confluence

<https://confluence.atlassian.com/doc/installing-confluence-on-linux-from-archive-file-255362363.html>

### JAVA

#### To install jre package:

```
$ sudo yum install jre
```

#### To check for updates to installed Java packages:

```
$ sudo yum list updates jre
```

#### To install jre updates:

```
$ sudo yum update jre
```

For more settings for open JDK, go to below link

<https://www.digitalocean.com/community/tutorials/how-to-install-java-with-apt-on-ubuntu-18-04>

### JAVA Basics

Java is a object oriented programming language.

Having concepts like:

- Encapsulation
- Inheritance
- Polymorphism

#### Prerequisites for Java Programming language:

##### Java used to program

1. Web Applications using Servlets and JSP
2. Native mobile apps for Android and BlackBerry
3. And Many other types of applications will be built with java

##### Who can learn Java

1. Used for people who want to develop software with java
2. Those with basic programming vocabulary.

##### Get ready for this course if you are able to answer below questions

1. What is a statement?
2. What is a variable?
3. What is function?
4. What is a condition?

If answer is NO, then go with java Fundamental programming.

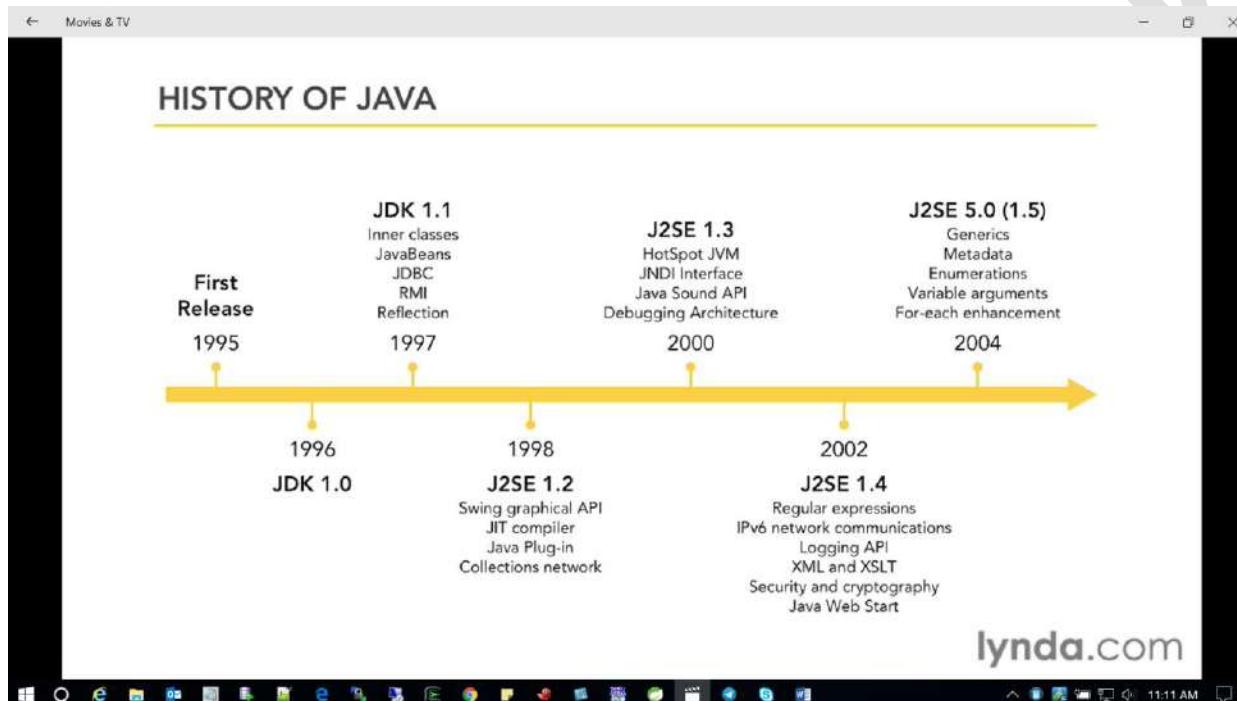
##### Similar Language:

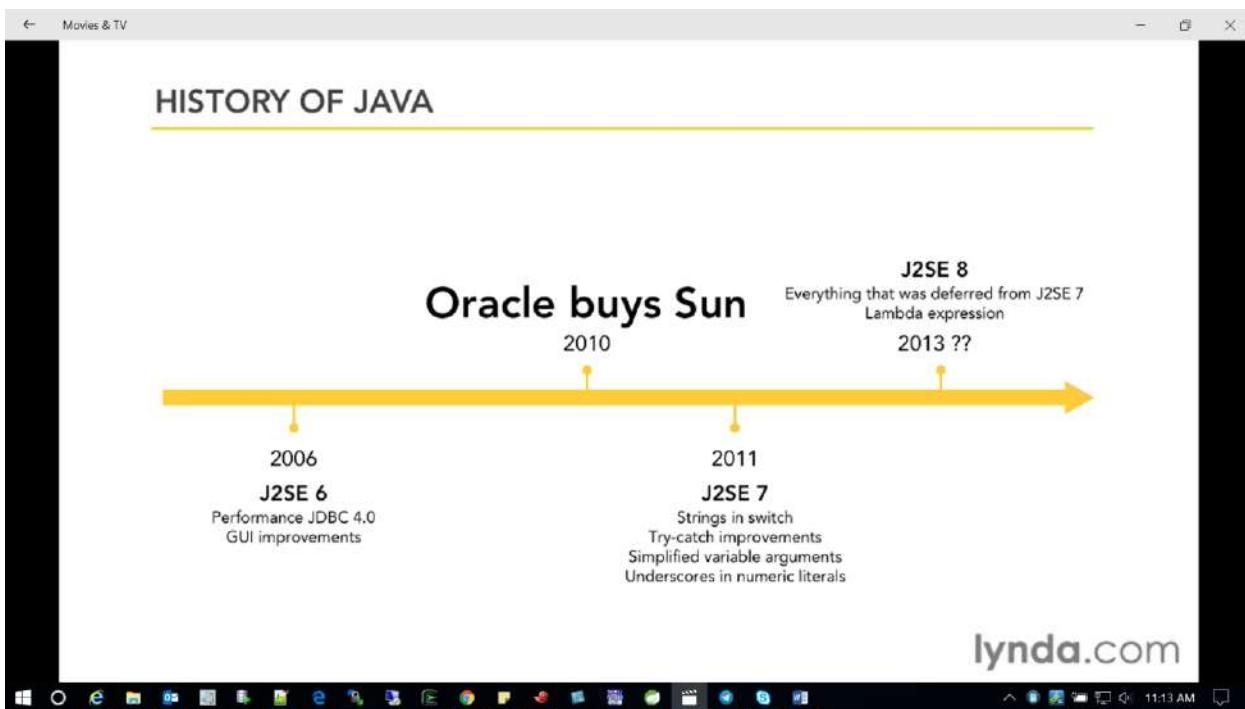
- Java is a “c-style” language

## SIMILAR LANGUAGES

- Java is a "C-Style" language
- Programmers of these languages will recognize the basic syntax:
  - C
  - C++
  - C#
  - JavaScript
  - PHP
- Java looks really different from:
  - Visual Basic
  - VB.NET
  - ColdFusion Markup Language
  - Assembly

Java History:





#### Principles of JAVA:

Having concepts like:

- Simple, Java is a object oriented programming language and familiar
- Robust and secure
- Architecture-neutral and portable
- High-Performance
- Interpreted is a compiled to a format in a interpreted time and run the program correctly
- Threaded, and Dynamic
- Encapsulation
- Inheritance
- Polymorphism

#### Runtime Architecture of JAVA

- Java is an interpreted language
- Compiles to bytecode instead of machine language
- The compiled application is portable between platforms without recompiling.

Java Virtual Machine (JVM)

Operating system

How to import Projects in Spring tool  
<https://www.youtube.com/watch?v=-PiXr5Emtd0>

#### Grails [Groovy and gsp]

- Grails is a framework, such as spring hibernate, quartz.. etc.
- Grails offer the best of available java.
- It's a concept of MVC. Not only MVC it also with GORM and build
- It's a fullstack framework.

<https://gsp.grails.org/latest/guide/index.html>

<https://www.youtube.com/watch?v=IPPFqjXGssw>

groovy support software spring verion :

<http://dist.springsource.com/release/STS/index.html>

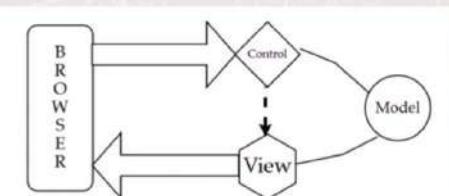
## GRAILS ARCHITECTURE



## GRAILS MVC



Grails built on concept of MVC, It does not merely recommend that you use MVC, It tells you to do so, then follows you home, and sits next to you to make sure you do



## WHY FULL STACK ?



All of these are made easy to use through the power of the Groovy language and the extensive use of Domain Specific Languages



[Move on to Groovy](#)

[What is Groovy](#)

## WHAT IS GROOVY?





**Groovy** is an agile and dynamic language for the Java Virtual Machine , builds upon the strengths of Java but has additional power features inspired by languages like Python, Ruby and Small talk makes modern programming features available to Java developers with almost-zero learning curve

Groovy = Java - boiler plate code

- + optional dynamic typing
- + closures
- + domain specific languages
- + builders
- + metaprogramming

[Why Groovy](#)

## WHY GROOVY?



**First the pseudo reason:** groovy runs on the JVM  
Groovy can be executed as a script using all legacy Java code and libraries

**Key Features:**

- A nice gateway to the Java platform
- Dynamic Language on JVM
- Focus on programmer productivity
- Superset of Java
- Native Java Library Integration
- Compiles into Java

## WHY GROOVY?



**Tons of Syntactic Features:**

- Tons of Options:
  - Optional parentheses
  - Optional semicolons
  - Optional Return
- Safe Dereferencing
- Autoboxing
- GStrings
- Regular Expressions
- Autogenerated Setters / Getters
- Even Elvis....

[Features](#)

# FEATURES



## A multi-faceted language for the Java platform

Apache Groovy is a powerful, optionally typed and dynamic language, with static-typing and static compilation capabilities, for the Java platform aimed at improving developer productivity thanks to a concise, familiar and easy to learn syntax. It integrates smoothly with any Java program, and immediately delivers to your application powerful features, including scripting capabilities, Domain-Specific Language authoring, runtime and compile-time meta-programming and functional programming.

### Flat learning curve

Concise, readable and expressive syntax, easy to learn for Java developers

### Smooth Java integration

Seamlessly and transparently integrates and interoperates with Java and any third-party libraries

### Vibrant and rich ecosystem

Web development, reactive applications, concurrency / asynchronous / parallelism library, test frameworks, build tools, code analysis, GUI building



### Powerful features

Closures, builders, runtime & compile-time meta-programming, functional programming, type inference, and static compilation



### Domain-Specific Languages

Flexible & malleable syntax, advanced integration & customization mechanisms, to integrate readable business rules in your applications



### Scripting and testing glue

Great for writing concise and maintainable tests, and for all your build and automation tasks

## Groovy VS JAVA

# GROOVY VS JAVA



### Closures

### Native syntax for lists and maps

```
def clos = [ println "hello!" ]  
clos() //prints "hello!"
```

### GroovyMarkup and GPath support

### Native support for regular expressions

### Polymorphic iteration and powerful switch statement

### Dynamic and static typing is supported - so you can omit the type declarations on methods, fields and variables

### Default Imports

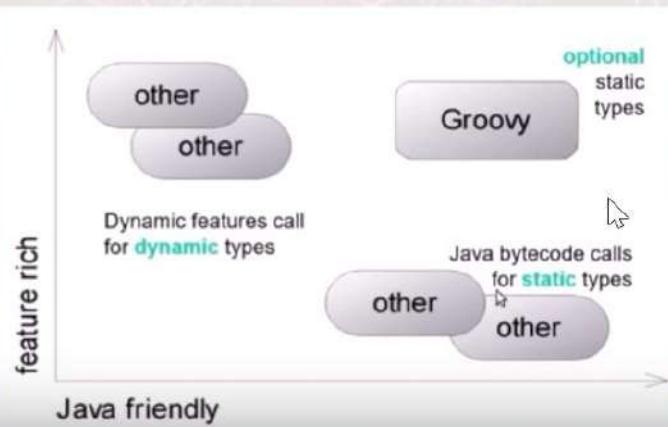
Example syntax of groovy for loop: it will also support java syntax.

```
For (I in 0..10)  
Assigning a variable as "def"  
Groovy.lan  
Groovy.io  
Groovy combines in both static and dynamic.
```

## GROOVY VS JAVA

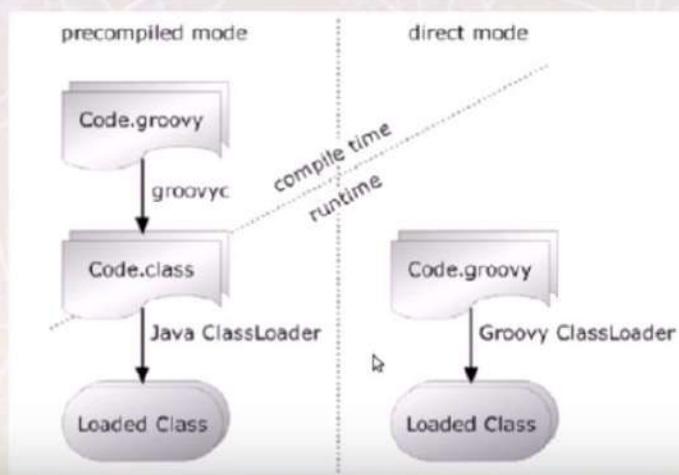


Groovy is feature rich and java friendly



Direct Mode VS precompiled Mode

## DIRECT MODE VS PRECOMPILED MODE



Getting Started with Library Management system Application using GGTS

### UNIX administration Topics

How to determine if AIX uses a 32-bit kernel or 64-bit kernel:

```
# getconf KERNEL_BITMODE  
64  
# getconf HARDWARE_BITMODE  
64  
# getconf DISK_SIZE /dev/hdisk0  
8678
```

You can also check by "ls -l /unix" it will give you like '/unix -> /usr/lib/boot/unix\_64' which means it is 64 bit.  
bootinfo -k or -y

```
$ORACLE_HOME/lib/    ==> 32 bit  
$ORACLE_HOME/lib64 ==> 64 bit
```

```
uname -a
```

```
[root@localhost ~]# uname -a
Linux localhost.localdomain 4.14.35-1818.3.3.el7uek.x86_64 #2 SMP Mon Sep 24 14:45:01 PDT 2018 x86_64 x86_64 x86_64 GNU/Linux
[root@localhost ~]#
```

Uname -m

```
[root@localhost ~]# uname -m
x86_64
```

Get complete CPU architecture

Command : lscpu

```
[root@localhost ~]# lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Byte Order:            Little Endian
CPU(s):                2
On-line CPU(s) list:  0,1
Thread(s) per core:   1
Core(s) per socket:   2
Socket(s):             1
NUMA node(s):          1
Vendor ID:             GenuineIntel
CPU family:            6
Model:                 158
Model name:            Intel(R) Core(TM) i7-7700HQ CPU @ 2.80GHz
Stepping:              9
CPU MHz:               2808.000
BogoMIPS:              5616.00
Hypervisor vendor:    KVM
Virtualization type:  full
L1 cache:              32K
L1i cache:             32K
L2 cache:              256K
L3 cache:              6144K
NUMA node0 CPU(s):    0,1
Flags:                 fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge nopl xtopology nonstop_tsc cpuid lm esx xsave avx rdrand hypervisor lahf_lm abm 3dnowprefetch invpcid si
```

#### Unix Login welcome message text

Have a look at /etc/motd.

If it's happening after /etc/motd is displayed it could be an "echo" within one of your shell initialisation files.

If it's not within /etc/motd, post which shell you're using.

#### For Getting xclock

Please execute below command.

Xclock on Ubuntu : apt-get install x11\*

Export DISPLAY=<your desktop/Laptop IP>:0.0

Ex : export DISPLAY=10.66.113.46:0.0

Xclock

Xclock on Oracle linux : yum install xorg-x11-apps.x86\_64

```
Installed:
  xorg-x11-apps.x86_64 0:7.7-7.el7
```

Dependency Installed:

```
  libXaw.x86_64 0:1.0.13-4.el7    libXcursor.x86_64 0:1.1.14-8.el7
```

Complete!

```
[root@localhost vagrant]# xclock
```

Warning: Missing charsets in String to FontSet conversion



#### Oracle sqlplus tips and tricks

##### LineSize & PageSize

To check the all attribute values for alignment

```
SQL> show all
```

To set the pagesize

```
SQL> set pagesize 15
```

To set the Column size/format

```
SQL> COLUMN name FORMAT A8
```

```
SQL> COLUMN address FORMAT A15
```

To set the linesize

```
SQL> set linesize 120
```

```
SQL> desc clients
```

```
SQL> set pagesize 100 linesize 120 newpage 0 feedback off
```

```
set pagesize 100 linesize 30 newpage 0 feedback off
```

```
select ENGLISH, ARABIC from CONFIG_ARABIC_CALTYPE_MAP_T where ENGLISH='Content Services*';
```

```
set pagesize 100 linesize 30 newpage 0 feedback off
```

```
COLUMN ENGLISH FORMAT A40
```

```
COLUMN ARABIC FORMAT A40,
```

will give you the columns in your table.

```
so you can select columns you need it or select all.  
SQL> COLUMN name FORMAT A8,  
SQL> COLUMN address FORMAT A15
```

If a name is longer than 8 characters, the remaining is displayed at the second line (or several lines)  
\*\* The set command can be used to change the default number of lines per page (14) and the number of characters per line (80).

For example, to set the number of lines per page to 60, use the following command:

```
SQL> SET PAGESIZE 60
```

All formatting remain active until they are cleared or reset or after you exit from SQL\*Plus.

```
SQL> CLEAR COLUMN
```

## Toad suite

<http://www.mediafire.com/file/1ftptwmcel1hgq/Toad+DBA+Suite+for+Oracle+11.5+Commercial.part1.rar>

### Oracle database installation on Oracle Linux:

<https://oracle-base.com/articles/12c/oracle-db-12cr1-installation-on-oracle-linux-7#12102-notestopics> :  
<http://mammqm.blogspot.com/2016/02/linuxunix.html>

### oracle-help

#### Host files on unix Env:

```
/etc/hosts  
/etc/hostname
```

#### Oracle Installation Prerequisites

✓ #yum install oracle-rdbms-server-12cR1-preinstall -y  
It is probably worth doing a full update as well, but this is not strictly speaking necessary.

✓ # yum update -y

Manual Setup for all packages prerequisites.

Add the following lines to the "/etc/sysctl.conf" file, or in a file called "/etc/sysctl.d/98-oracle.conf".

✓ Vi "/etc/sysctl.conf

```
✓ fs.file-max = 6815744  
✓ kernel.sem = 250 32000 100 128  
✓ kernel.shmmni = 4096  
✓ kernel.shmall = 1073741824  
✓ kernel.shmmax = 439804651104  
✓ kernel.panic_on_oops = 1  
✓ net.core.rmem_default = 262144  
✓ net.core.rmem_max = 4194304  
✓ net.core.wmem_default = 262144  
✓ net.core.wmem_max = 1048576  
✓ net.ipv4.conf.all.rp_filter = 2  
✓ net.ipv4.conf.default.rp_filter = 2  
✓ fs.aio-max-nr = 1048576  
✓ net.ipv4.ip_local_port_range = 9000 65500
```

Run the following command to change the current kernel parameters.

```
/sbin/sysctl -p
```

Add the following lines to a file called "/etc/security/limits.d/oracle-rdbms-server-12cR1-preinstall.conf" file.

Create the new groups and users.

```
groupadd -g 54321 oinstall  
groupadd -g 54322 dba  
groupadd -g 54323 oper  
#groupadd -g 54324 backupdba  
#groupadd -g 54325 dgdba  
#groupadd -g 54326 kmdba  
#groupadd -g 54327 asmdba  
#groupadd -g 54328 asmoper  
#groupadd -g 54329 asmadmin
```

```
useradd -u 54321 -g oinstall -G dba,oper oracle
```

Set the password for the "oracle" user.

```
passwd oracle
```

Set secure Linux to permissive by editing the "/etc/selinux/config" file, making sure the SELINUX flag is set as follows.

**SELINUX=permissive**

Once the change is complete, restart the server or run the following command.

# setenforce Permissive

If you have the Linux firewall enabled, you will need to disable or configure it, as shown here or here. To disable it, do the following.

# systemctl stop firewalld

# systemctl disable firewalld

Create the directories in which the Oracle software will be installed.

mkdir -p /u01/app/oracle/product/12.1.0.2/db\_1

chown -R oracle:oinstall /u01

chmod -R 775 /u01

Unless you are working from the console, or using SSH tunnelling, login as root and issue the following command.

xhost +<machine-name>

or

[root@orainux7 vagrant]# export DISPLAY=10.66.113.46:0.0

[root@orainux7 vagrant]# xclock

Add the following lines at the end of the "/home/oracle/.bash\_profile" file.

# Oracle Settings

export TMP=/tmp

export TMPDIR=\$TMP

export ORACLE\_HOSTNAME=orainux7

export ORACLE\_UNQNAME=cdb1

export ORACLE\_BASE=/u01/app/oracle

export ORACLE\_HOME=\$ORACLE\_BASE/product/12.1.0.2/db\_1

export ORACLE\_SID=cdb1

export PATH=/usr/sbin:\$PATH

export PATH=\$ORACLE\_HOME/bin:\$PATH

export LD\_LIBRARY\_PATH=\$ORACLE\_HOME/lib:/lib:/usr/lib

export CLASSPATH=\$ORACLE\_HOME/jlib:\$ORACLE\_HOME/rdbms/jlib

**Installation**

Sudo su oracle

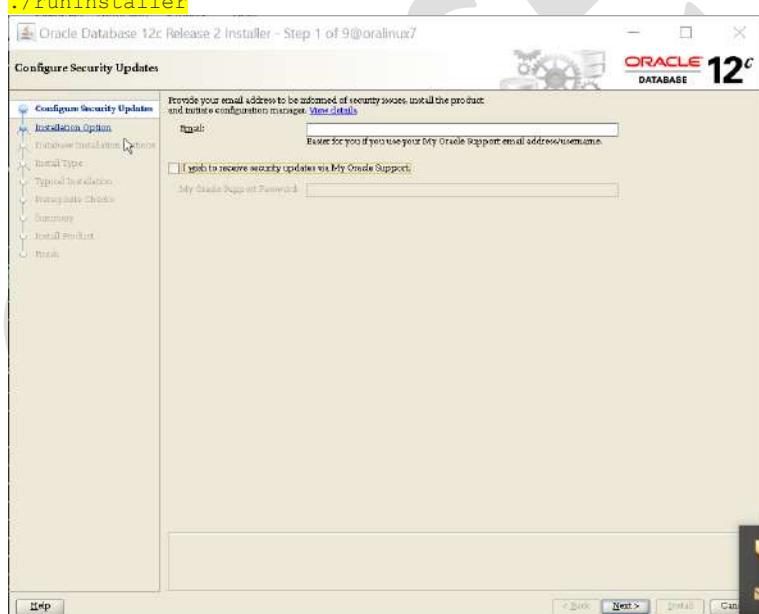
**Navigate to oracle database software directory which is having full permissions.**

Log into the oracle user. If you are using X emulation then set the DISPLAY environmental variable.

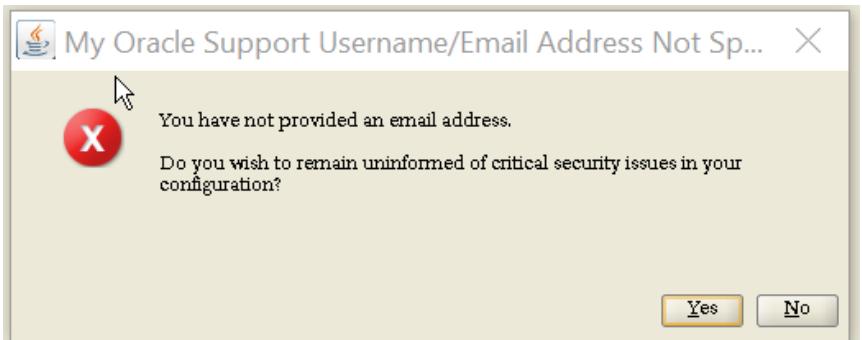
DISPLAY=<machine-name>:0.0; export DISPLAY

Start the Oracle Universal Installer (OUI) by issuing the following command in the database directory.

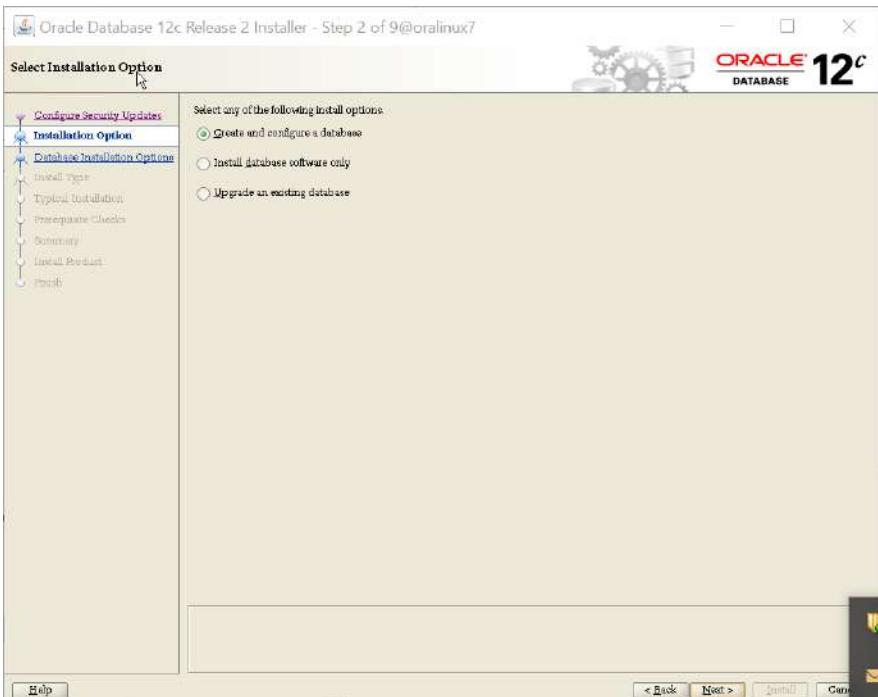
./runInstaller



Uncheck the value for My Oracle Support , click on next



Yes



Next

- Select Server class.



- Select Single instance database installation, click on Next

Select the type of database installation you want to perform.

- Single instance database installation
- Oracle Real Application Clusters database installation
- Oracle RAC One Node database installation

- Click on Typical install, click next

Typical install

Perform full Oracle Database installation with basic configuration.

Advanced install

Allows advanced selections such as different passwords for the SYS,SYSMAN, SYSTEM and DBSNMP accounts, database character set, product languages, automated backups, custom installation, and alternative storage options such as Oracle Automatic Storage Management.

- Db name: cdbl
- Password :cdbl
- And Pluggable DB Name : cpdbl
- Click on Next.

Perform full database installation with basic configuration.

|                                                                  |                                     |                   |
|------------------------------------------------------------------|-------------------------------------|-------------------|
| Oracle base:                                                     | /u01/app/oracle                     | Browse...         |
| Software location:                                               | /u01/app/oracle/product/12.2.0/db_1 | Browse...         |
| Storage type:                                                    | File system                         |                   |
| Database file location:                                          | /u01/app/oracle/oradata             | Browse...         |
| Database edition:                                                | Enterprise Edition (7.5GB)          |                   |
| OSDBA group:                                                     | dba                                 |                   |
| Global database name:                                            | cdbl                                |                   |
| Password:                                                        |                                     | Confirm password: |
| Service name:                                                    |                                     |                   |
| <input checked="" type="checkbox"/> Create as Container database |                                     |                   |
| Pluggable database name:                                         | cpdbl                               |                   |

- Click Next, don't change the oraInventory Group Name and path.

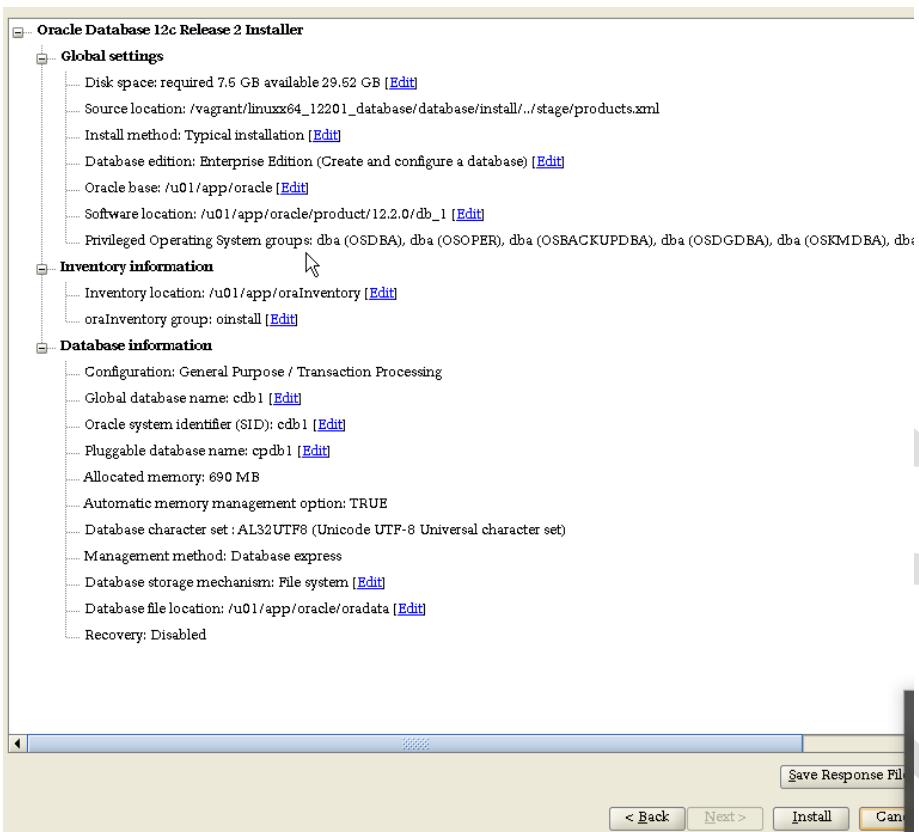
You are starting your first installation on this host. Specify a directory for installation metadata files (for example, install log files). This directory is called the "inventory directory". The installer automatically sets up subdirectories for each product to contain inventory data. The subdirectory for each product typically requires 150 kilobytes of disk space.

Inventory Directory: /u01/app/oraInventory     

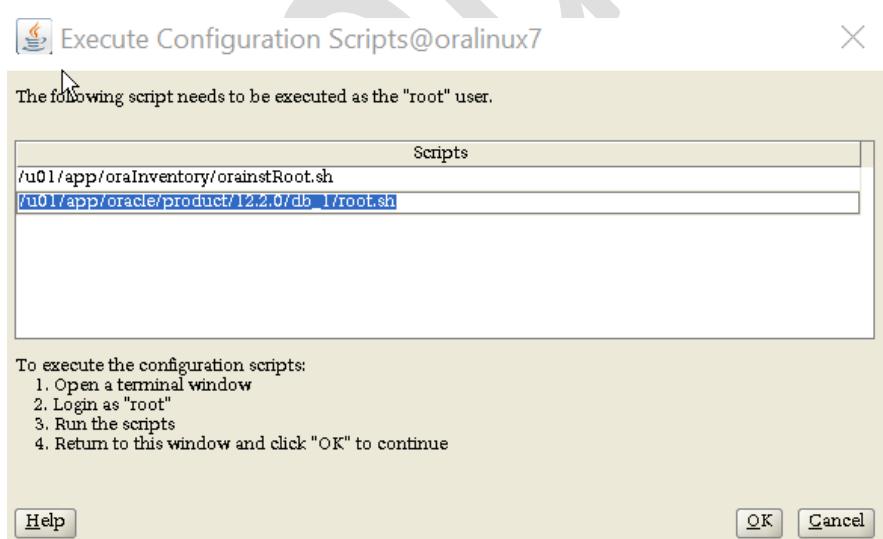
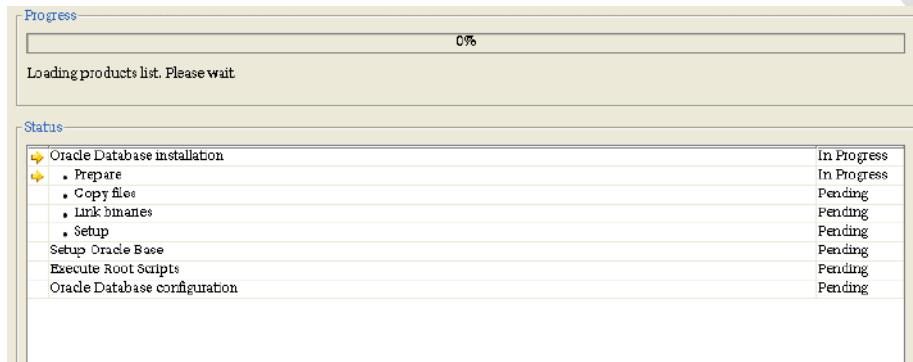
Specify an operating system group whose members have write permission to the inventory directory (oraInventory).

oraInventory Group Name: oinstall

- Click on Next and you will get the Response file and save it for further verification



- Now click on Install



- Please run the above scripts from root user, snap shot as follows.

```

For additional packages, updates, documentation and community help, see:
 * http://yum.oracle.com/
[vagrant@orinux7 ~]$ sudo su root
[root@orinux7 vagrant]# /u01/app/oraInventory/orainstRoot.sh
Changing permissions of '/u01/app/oraInventory'.
Adding read,write permissions for group.
Removing read,write,execute permissions for world.

Changing groupname of '/u01/app/oraInventory' to oinstall.
The execution of the script is complete.
[root@orinux7 vagrant]# /u01/app/oracle/product/12.2.0/db_1/root.sh
Performing root user operation.

The following environment variables are set as:
 ORACLE_OWNER= oracle
 ORACLE_HOME= /u01/app/oracle/product/12.2.0/db_1

Enter the full pathname of the local bin directory: [/usr/local/bin]:
Copying dbhome to /usr/local/bin ...
Copying oraenv to /usr/local/bin ...
Copying coraenv to /usr/local/bin ...

Creating /etc/oratab file...
Entries will be added to the /etc/oratab file as needed by
Database Configuration Assistant when a database is created
Finished running generic part of root script.
Now product-specific root actions will be performed.

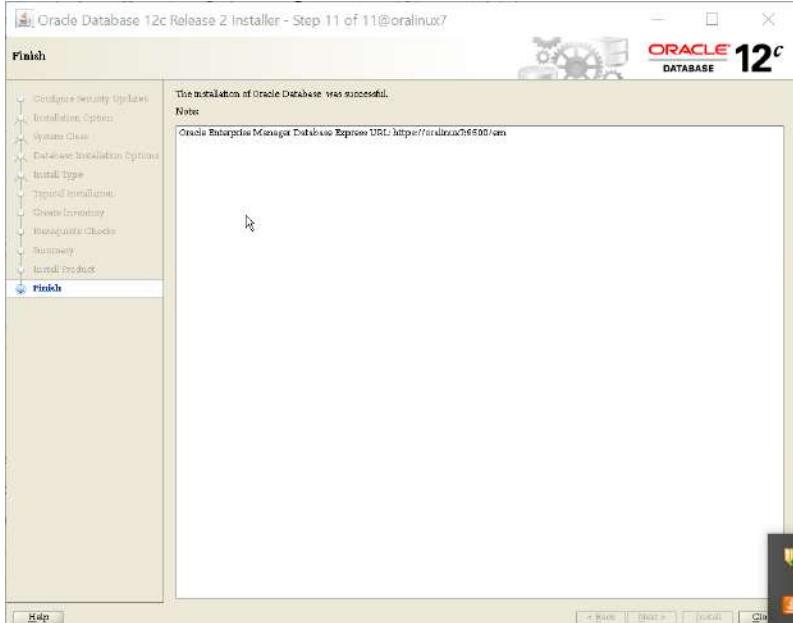
Do you want to setup Oracle Trace File Analyzer (TFA) now? yes|[no] :
Oracle Trace File Analyzer (TFA - User Mode) is available at :
/u01/app/oracle/product/12.2.0/db_1/suptools/tfa/release/tfa_home/bin/tfactl
OR

Oracle Trace File Analyzer (TFA - Daemon Mode) can be installed by running this script :
/u01/app/oracle/product/12.2.0/db_1/suptools/tfa/release/tfa_home/install/roottfa.sh
[root@orinux7 vagrant]#

```

<https://orinux7:5500/em>

installation successfully done as follows.



- After installing software you need to check the services working or not.

#### Oracle DB post checks:

Check are you able to connect database by sqlplus

- Sqlplus sys as sysdba.
- Startup
- Define

```
[oracle@orainux7 ~]$ sqlplus sys as sysdba
SQL*Plus: Release 12.2.0.1.0 Production on Mon Oct 22 08:31:05 2018
Copyright (c) 1982, 2016, Oracle. All rights reserved.

Enter password:
Connected to an idle instance.

SQL> startup
ORACLE instance started.

Total System Global Area 725614592 bytes
Fixed Size          8797008 bytes
Variable Size       557843632 bytes
Database Buffers   150994944 bytes
Redo Buffers        7979008 bytes
Database mounted.
Database opened.

SQL> define
DEFINE _DATE      = "22-OCT-18" (CHAR)
DEFINE _CONNECT_IDENTIFIER = "cdb1" (CHAR)
DEFINE _USER       = "SYS" (CHAR)
DEFINE _PRIVILEGE = "AS SYSDBA" (CHAR)
DEFINE _SQLPLUS_RELEASE = "1202000100" (CHAR)
DEFINE _EDITOR     = "vi" (CHAR)
DEFINE _O_VERSION  = "Oracle Database 12c Enterprise Edition Release 12.2.0.1.0 - 64bit P
DEFINE _O_RELEASE  = "1202000100" (CHAR)
SQL> 
```

#### Oracle DB Details:

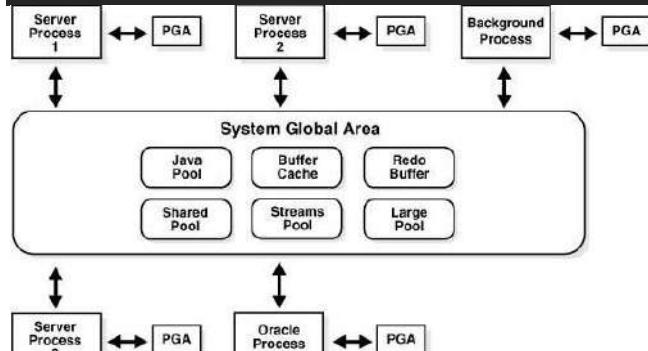
Oracle DB details:

Container DB name: cdb1  
 Pluggable DB name: CPDB1  
 Sys Password: oracle

#### Oracle DB Views & Important SQL's

```
SQL> select name from v$controlfile;
SQL> select member from v$logfile;
SQL> show sga
SQL> select con_id,name from v$tempfile;
SQL> select name from v$datafile;
SQL> select name from v$datafile where con_id=2;
SQL> select name, open_mode from v$pdbs;
$ sqlplus sys/password@localhost:1525/cdb12c as sysdba
```

#### Oracle DB Architecture:



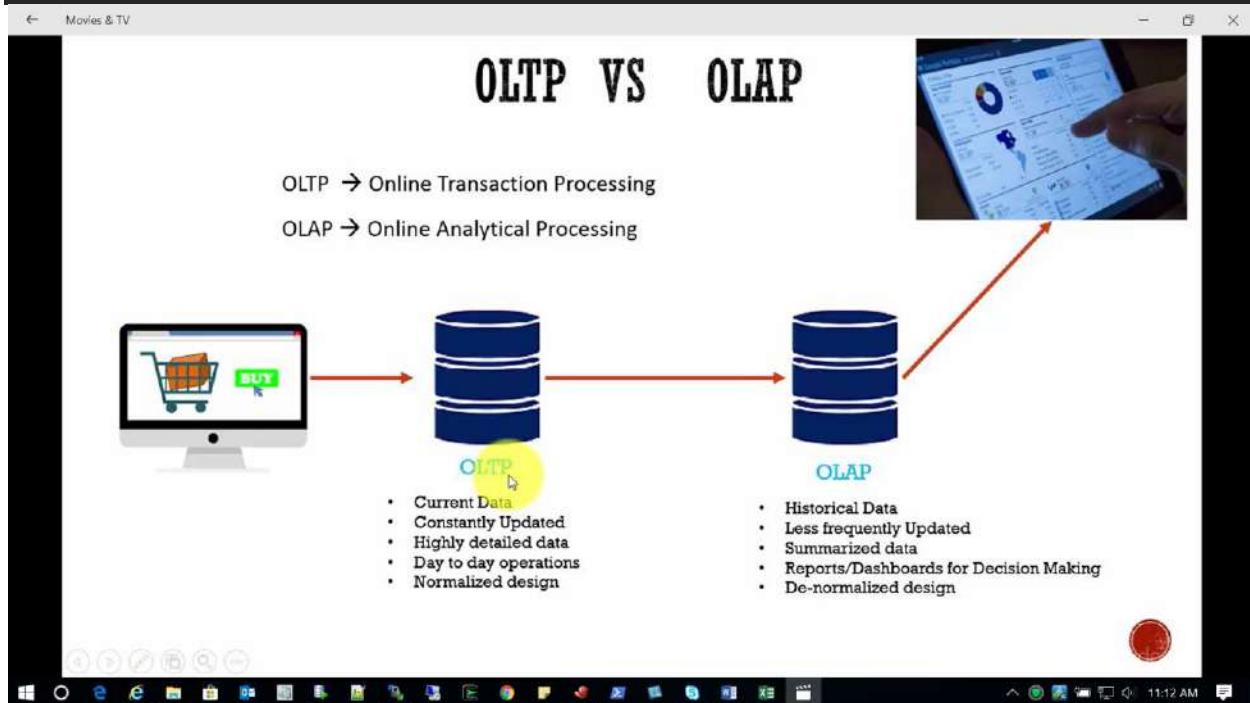
#### Difference between cdb and pdb in oracle 12c

- There is single network administration files like listener.ora, tnsnames.ora, and sqlnet.ora file for an entire CDB. All of the PDBs in the CDB use the same files.
- There is only one set of control files and online redo logs files for the entire CDB. Individual PDB's have their own data files (which contain the user data), but do not have distinct redo log or control files.
- There is a single SGA shared by all PDB's
- When the CDB is shut down then all PDB's are also automatically shut down

#### References :

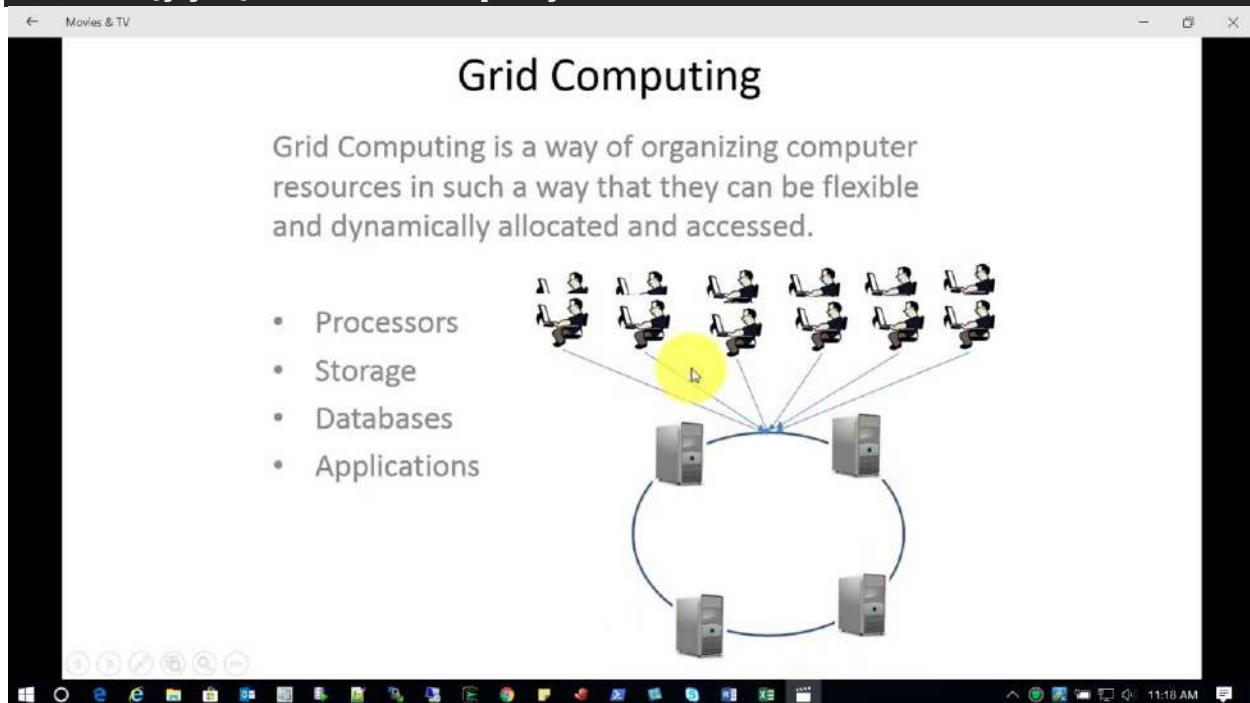
<https://www.linkedin.com/pulse/difference-between-cdb-pdb-oracle-12c-ankit-ashok-aggarwal>

About OLTP and OLAP

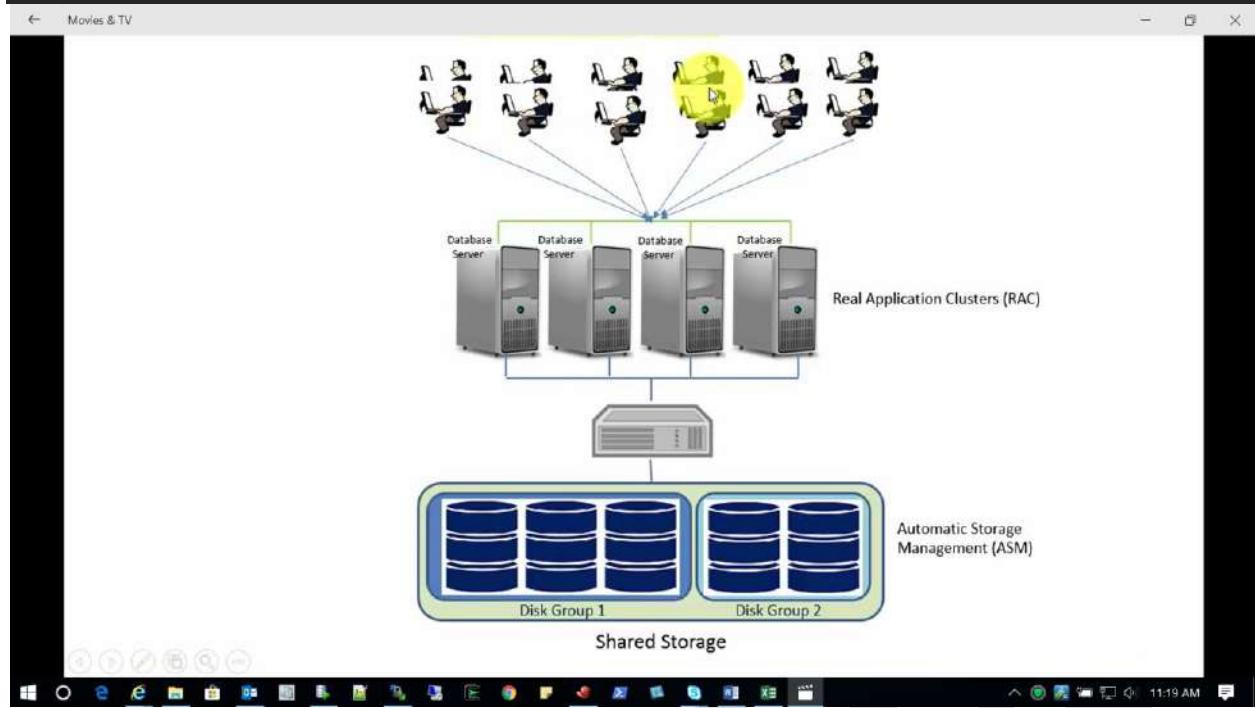


Highly normalized means, data is stored in multiple tables and relationships between them.

Oracle 11G [g-grid] : oracle Grid Computing.

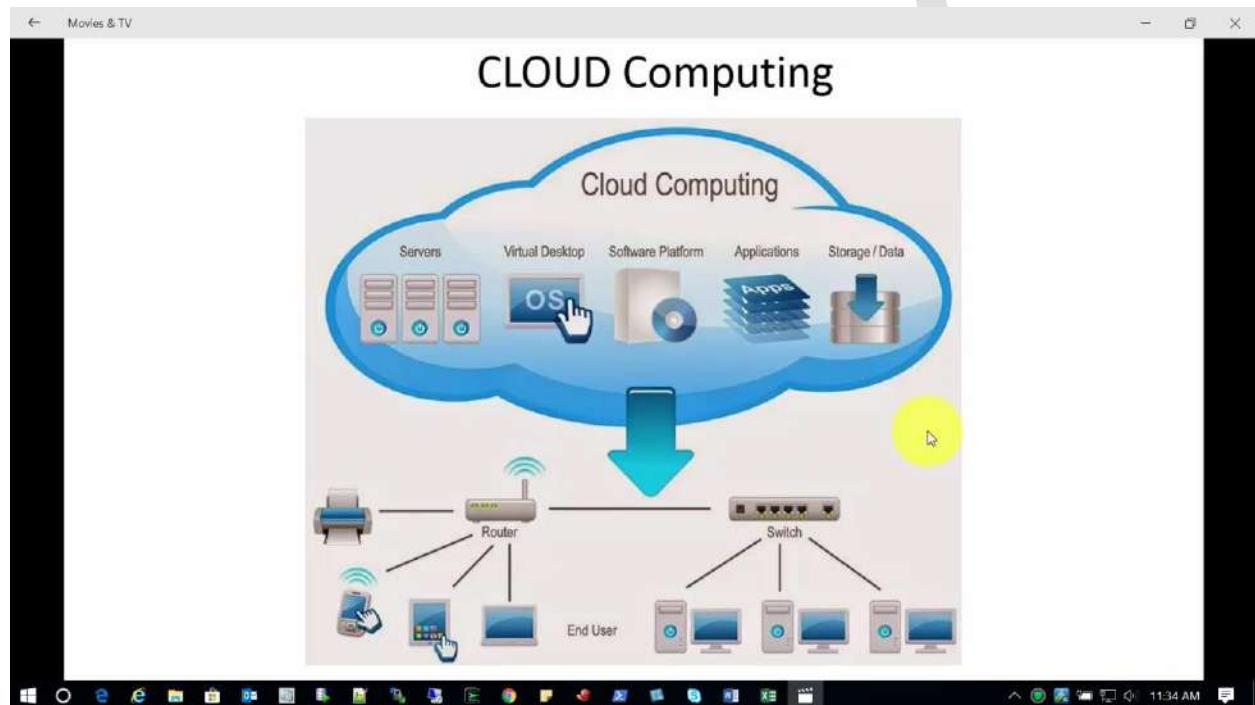


## RAC (Real application Cluster) and ASM(Automatic storage Management)



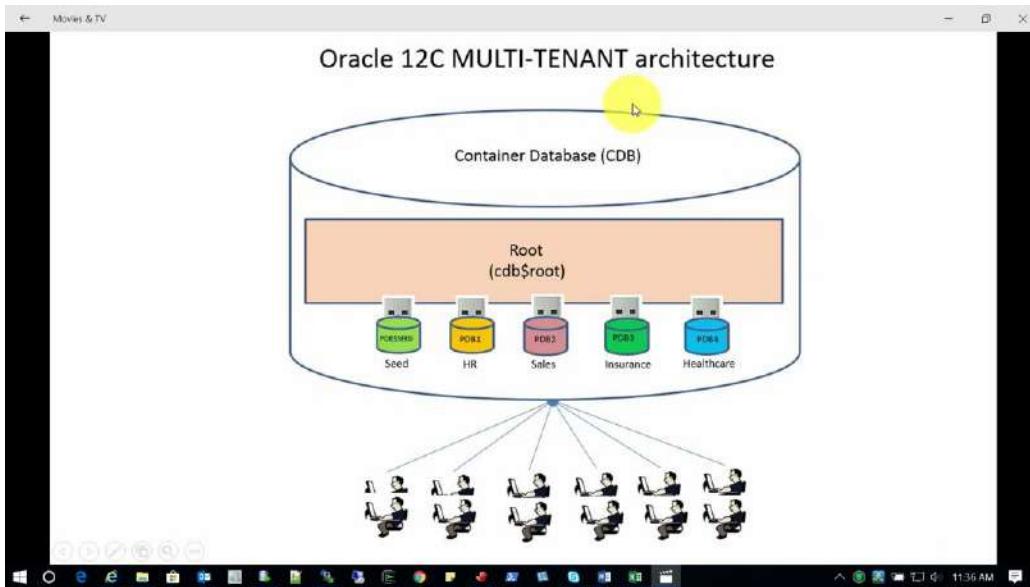
## Oracle 12C [Cloud computing]

Available over the network and for users as a services for the organizations which they don't have storage. No need to buy new servers.



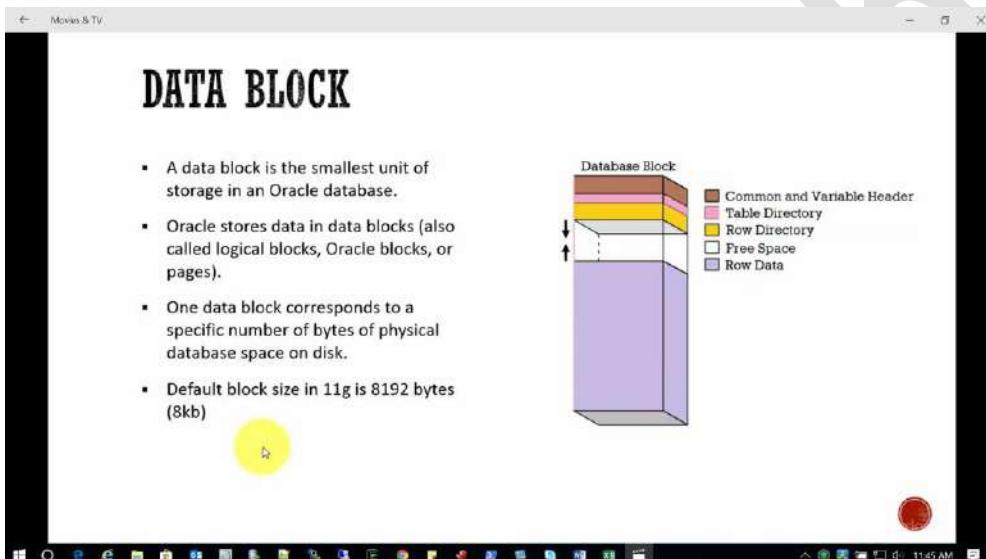
Supports as a container database (**Master Database**) and within we can configure/install multiple databases called pluggable database. It will connect from one database to other database. That's for oracle database as multi-tenant.

Unplug to one container1 database and plug to other container2 database → it's that simple is called as multi-tenant. Like you are plugging your mobile for charging and removing.



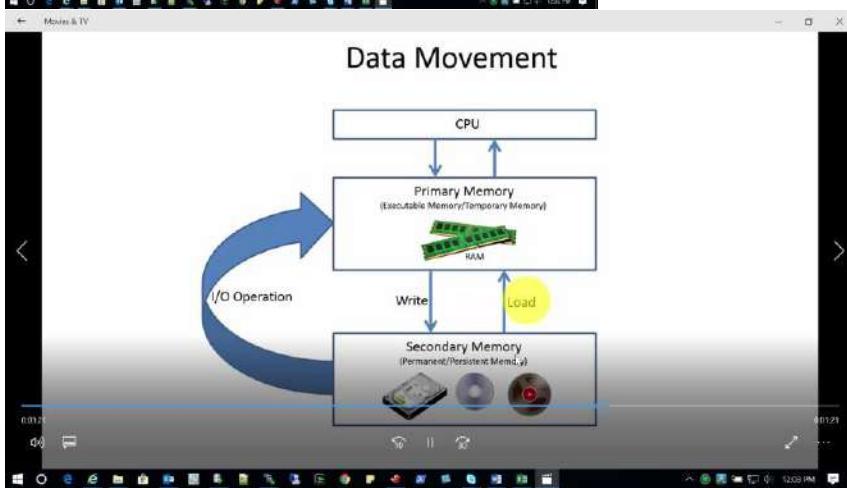
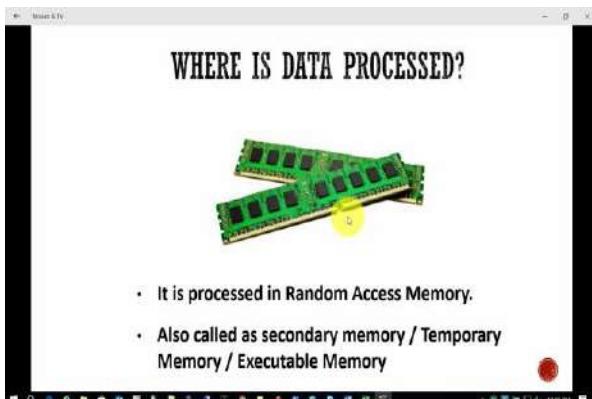
#### Oracle DBA Topics:

Computer Data will be in 0's and 1's, like in the same way oracle database data block's.



#### Where is DATA PROCESSED?

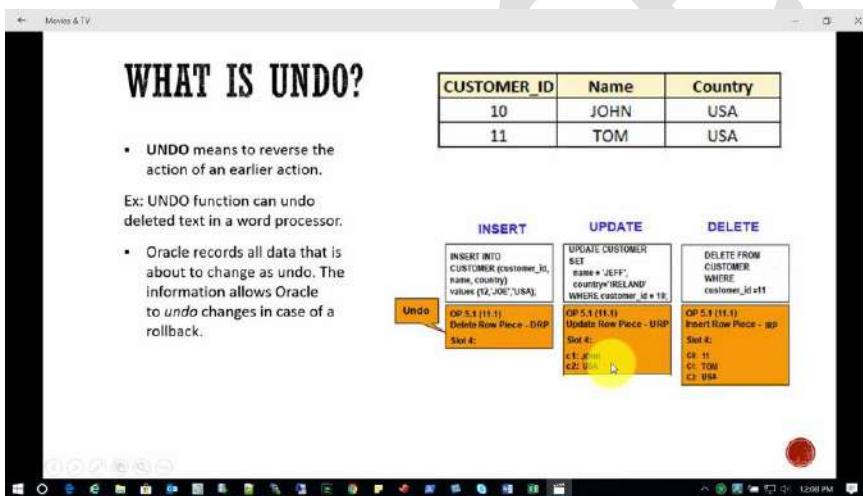
RAM(is a temporary memory), data will read in memory by DATA BLOCK's



## UNDO and REDO

What is an undo?

The deleted text will reappear. Means before going to do some change the oracle database maintained an undo data.



What is a REDO?

All data will be stored in REDO data, it will be used when if it only be applied some changes from redo data.

## WHAT IS REDO?

- REDO means to do something again.

Ex: You may type some text and then use UNDO to "untype" the text. You can use the REDO function to restore the text.

- Oracle records all changes made to the database. If the database crashes, this information allows Oracle to redo(reprocess) the changes.

| CUSTOMER_ID | Name | Country |
|-------------|------|---------|
| 10          | JOHN | USA     |
| 11          | TOM  | USA     |

**INSERT**

```
INSERT INTO CUSTOMER(customer_id, name, country)
values (12,'JOE','USA');
```

OP 11.2  
Insert Row Piece - DRP  
Slot 4:  
c0: 12  
c1: JOE  
c2: USA

**Redo**

**UPDATE**

```
UPDATE CUSTOMER
SET
name = 'JEFF',
country='IRELAND'
WHERE customer_id = 10;
```

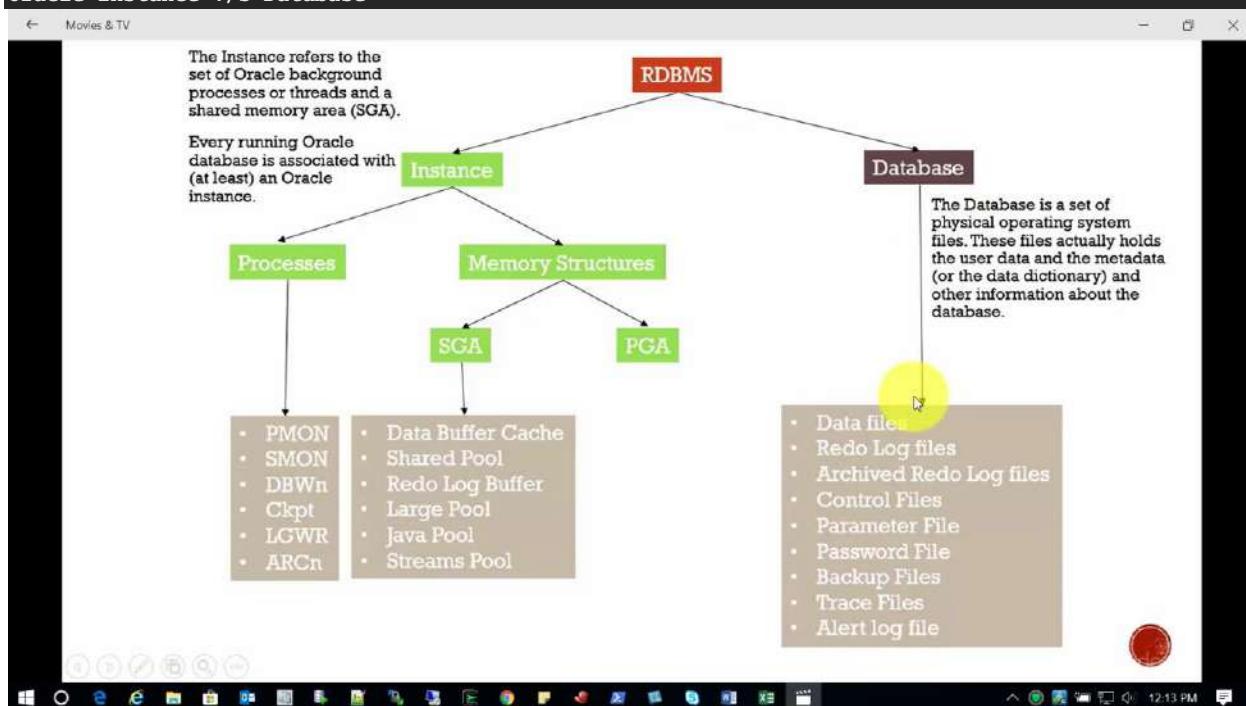
OP 11.5  
Update Row Piece - DRP  
Slot 4:  
c1: JEFF  
c2: IRELAND

**DELETE**

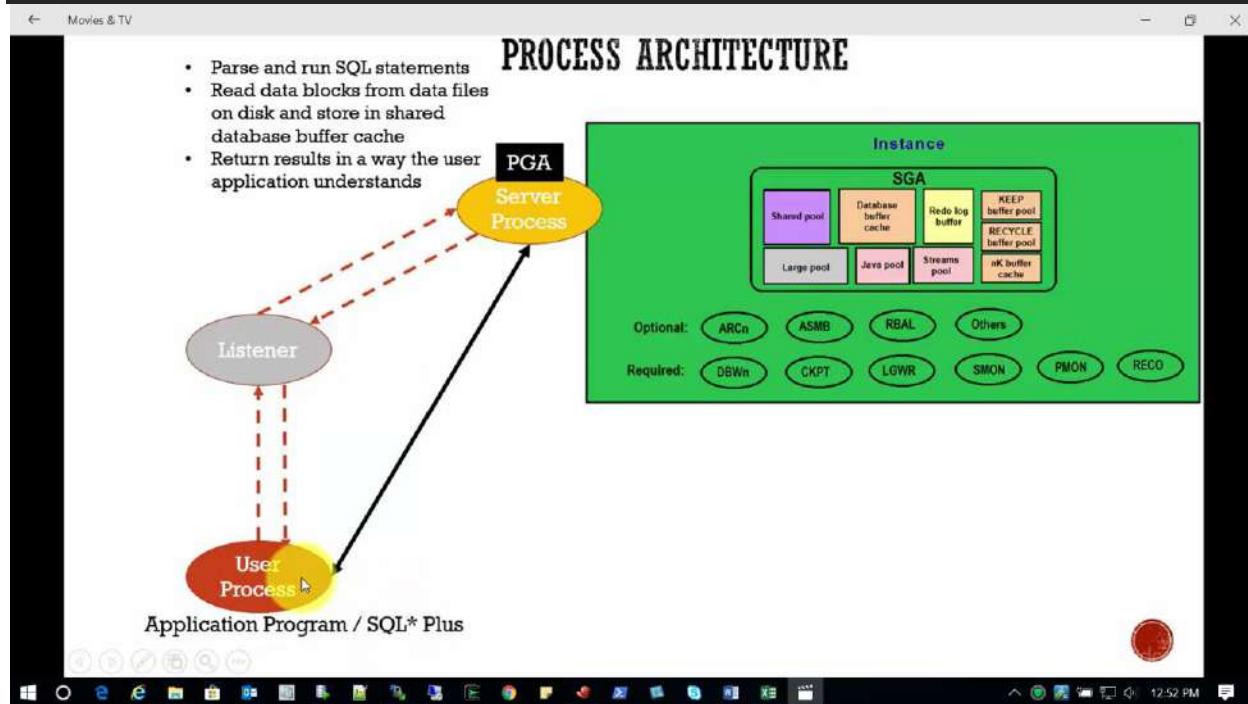
```
DELETE FROM CUSTOMER
WHERE
customer_id = 11;
```

OP 11.3  
Delete Row Piece - DRP  
Slot 4:

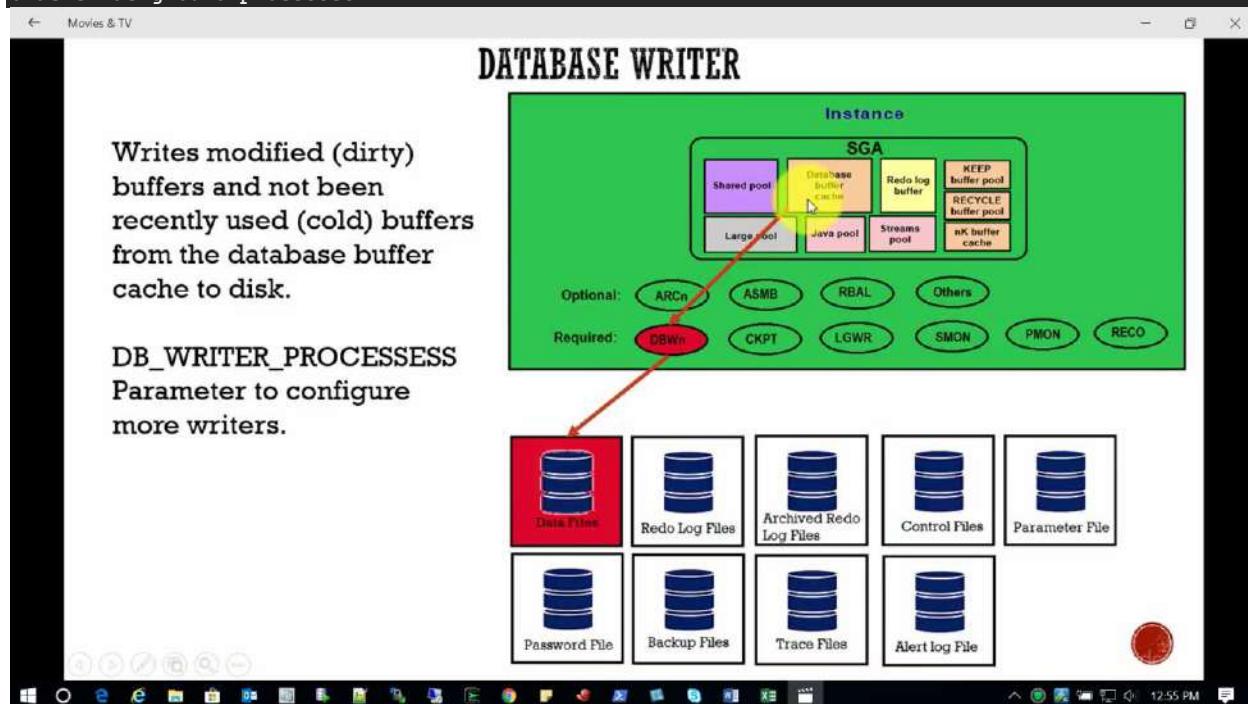
### Oracle instance v/s Database



## Oracle Process Architecture:

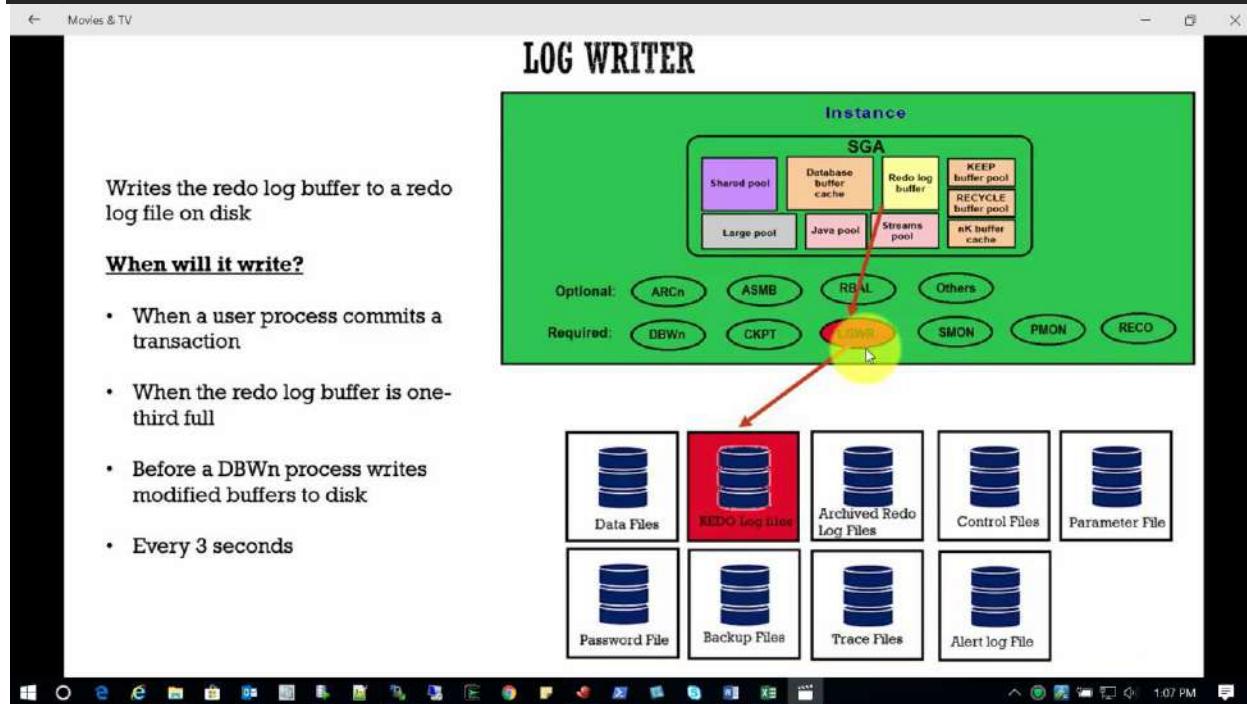


## Oracle Background processes:



1. Dirty Buffer : which are not using recently, from the data buffers cache
2. Cold Buffer :
3. Advance the check point CKPT, when oracle database writer will write a dirty buffer and cold buffer.

## LOG writer:



Writes the redo log buffer to a redo log file on disk

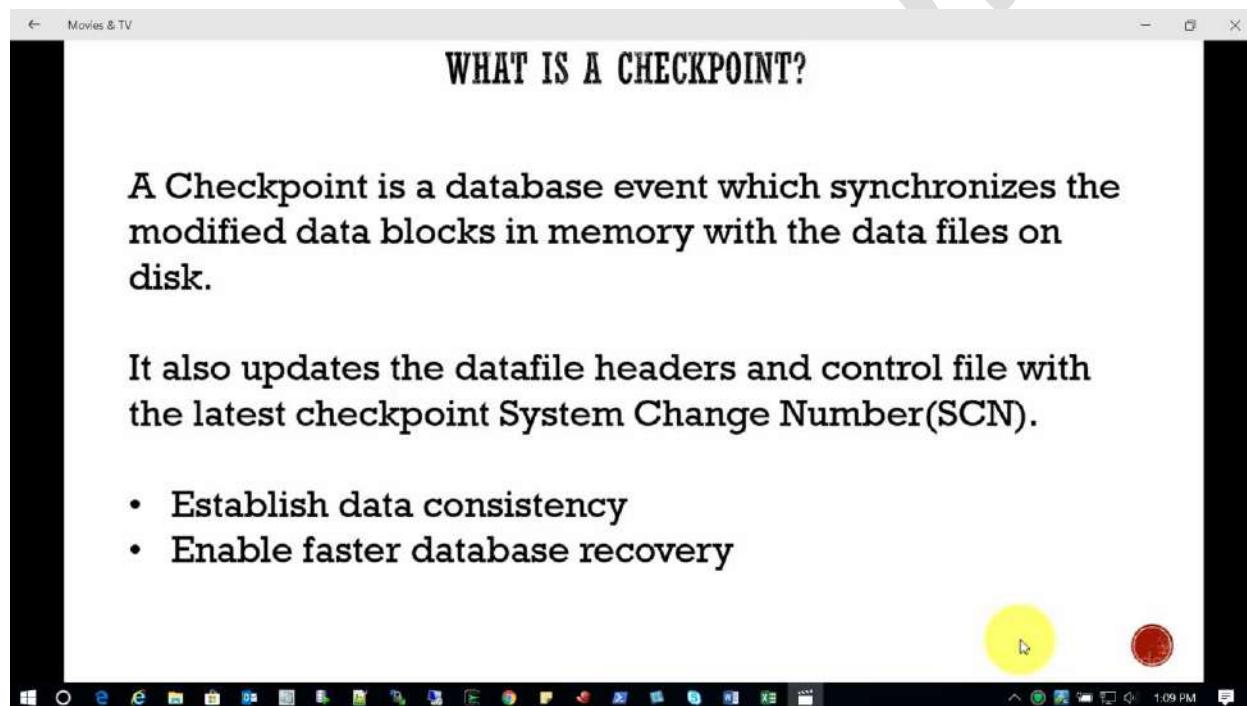
### When will it write?

- When a user process commits a transaction
- When the redo log buffer is one-third full
- Before a DBWn process writes modified buffers to disk
- Every 3 seconds

## Check Point Processes:

Establish data consistency and recovery.

SCN will be happening when the user commits every time.

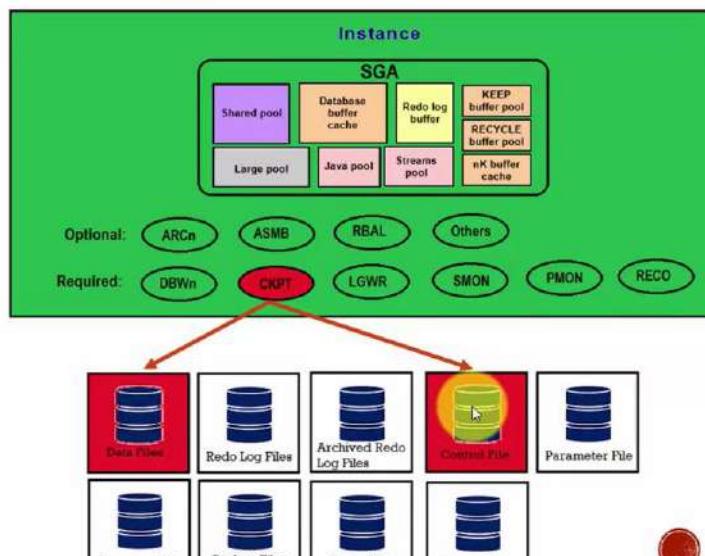


## WHEN WILL A CHECKPOINT OCCUR?

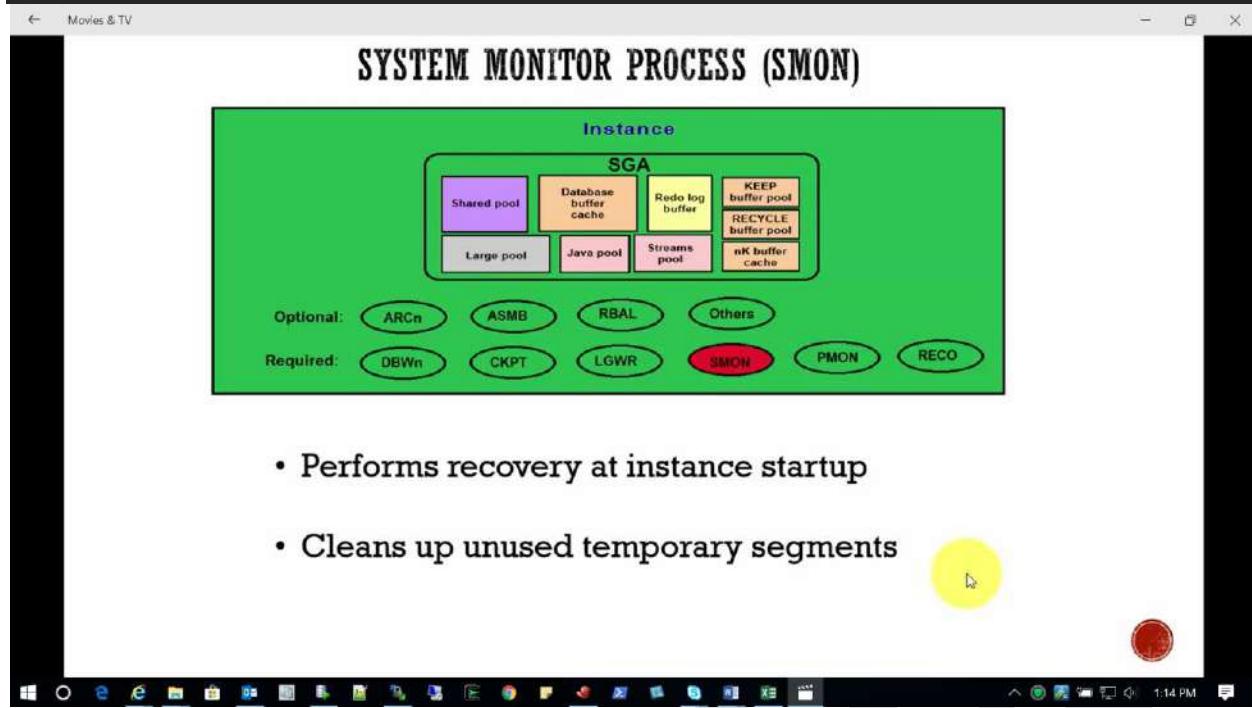
- At each switch of the redo log files.
- Once the number of seconds defined in the `LOG_CHECKPOINT_TIMEOUT` is reached.
- Once the current redo log file reaches the size  
`LOG_CHECKPOINT_INTERVAL * size of IO OS blocks`  
Example:  $10000 * 512 = 5120000$  (8M)
- Directly by the `ALTER SYSTEM SWITCH LOGFILE` command.
- Directly with the `ALTER SYSTEM CHECKPOINT` command.

Records checkpoint information (SCN) into the Control file and each data file header.

## CHECKPOINT



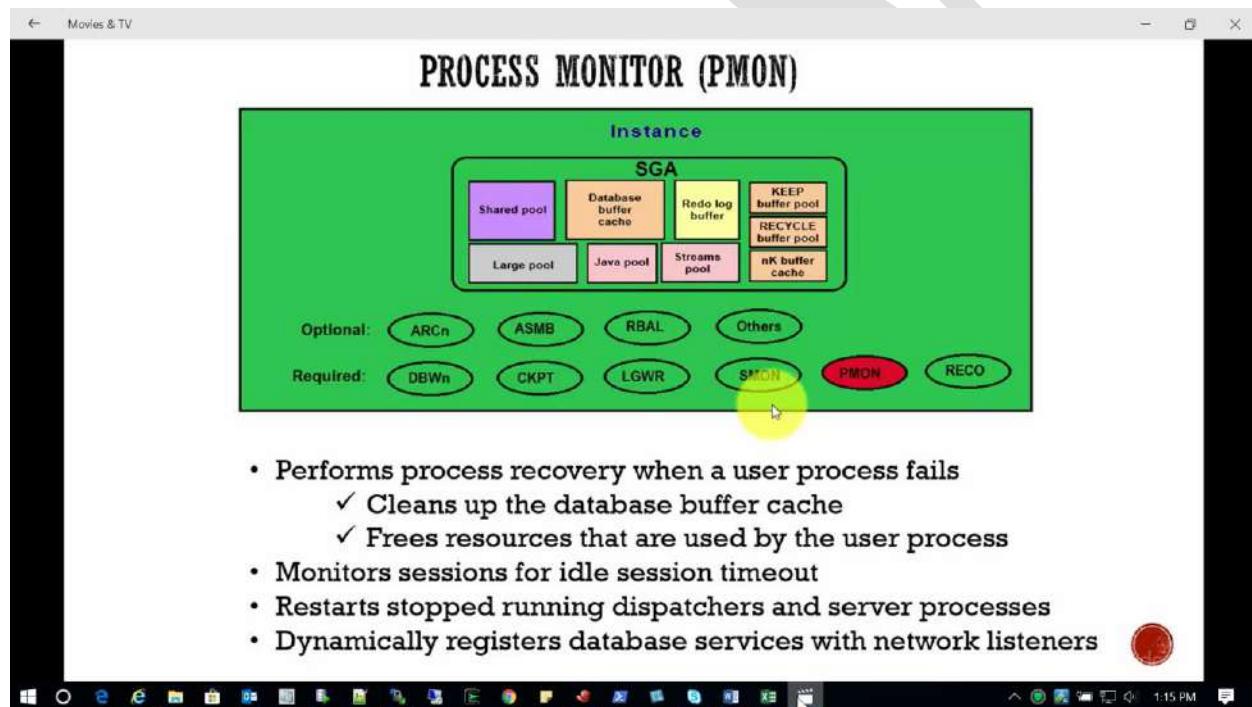
## System Monitor Process (SMON)



- Performs recovery at instance startup
- Cleans up unused temporary segments

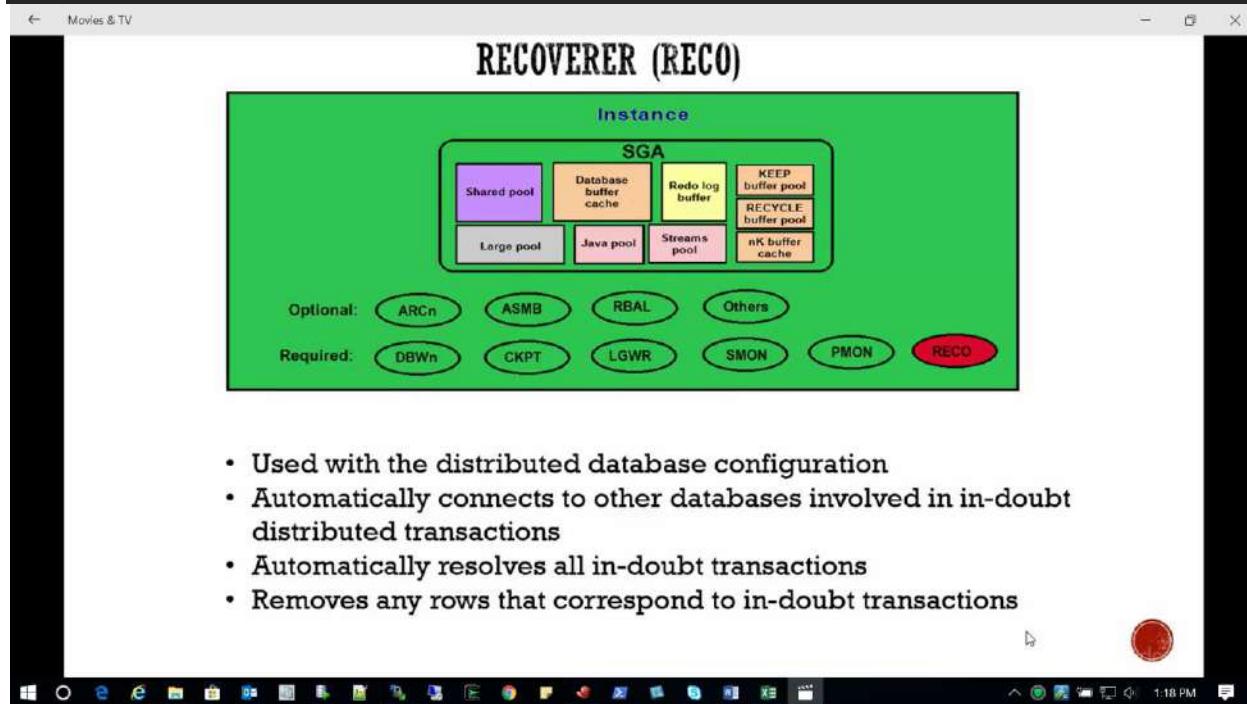
## Process Monitor

It's a manager for all processes



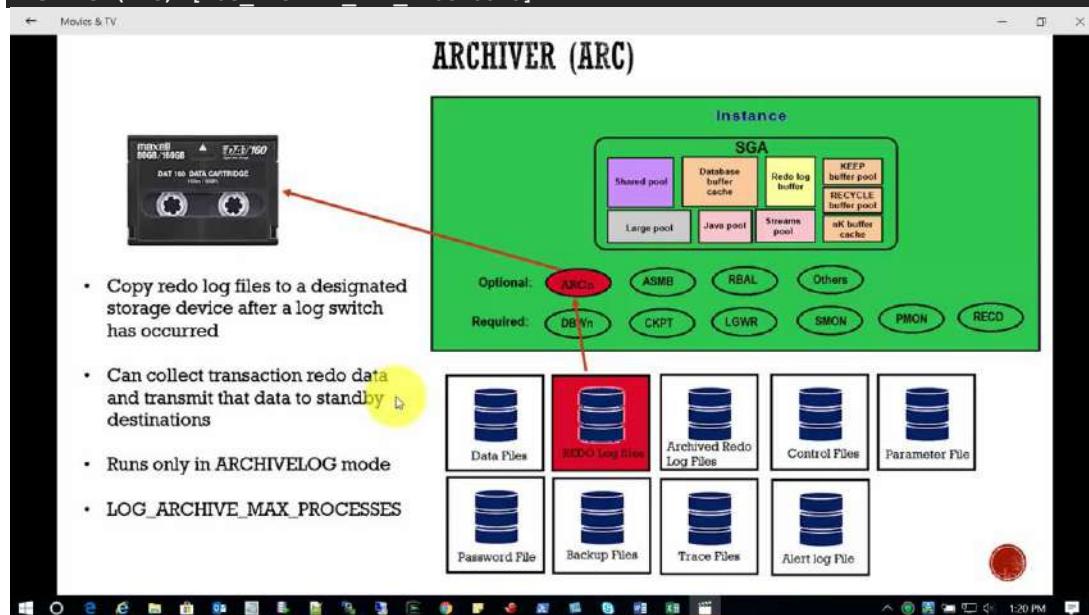
- Performs process recovery when a user process fails
  - ✓ Cleans up the database buffer cache
  - ✓ Frees resources that are used by the user process
- Monitors sessions for idle session timeout
- Restarts stopped running dispatchers and server processes
- Dynamically registers database services with network listeners

## Recoverer Process



- Used with the distributed database configuration
- Automatically connects to other databases involved in in-doubt distributed transactions
- Automatically resolves all in-doubt transactions
- Removes any rows that correspond to in-doubt transactions

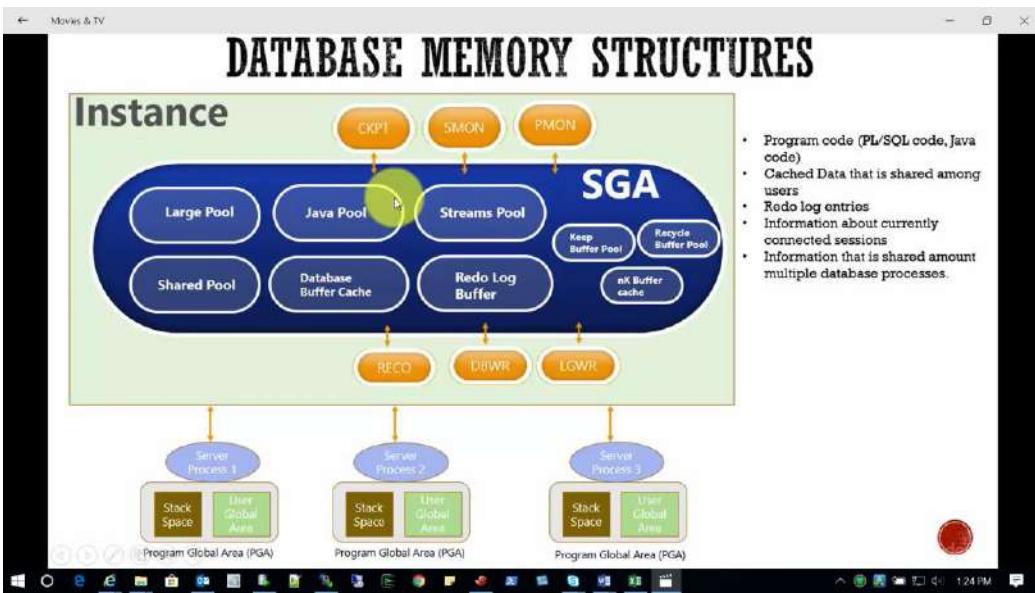
## Archiver (ARC) [LOG\_ARCHIVE\_MAX\_PROCESSES]



### 1. Database Memory Structure:

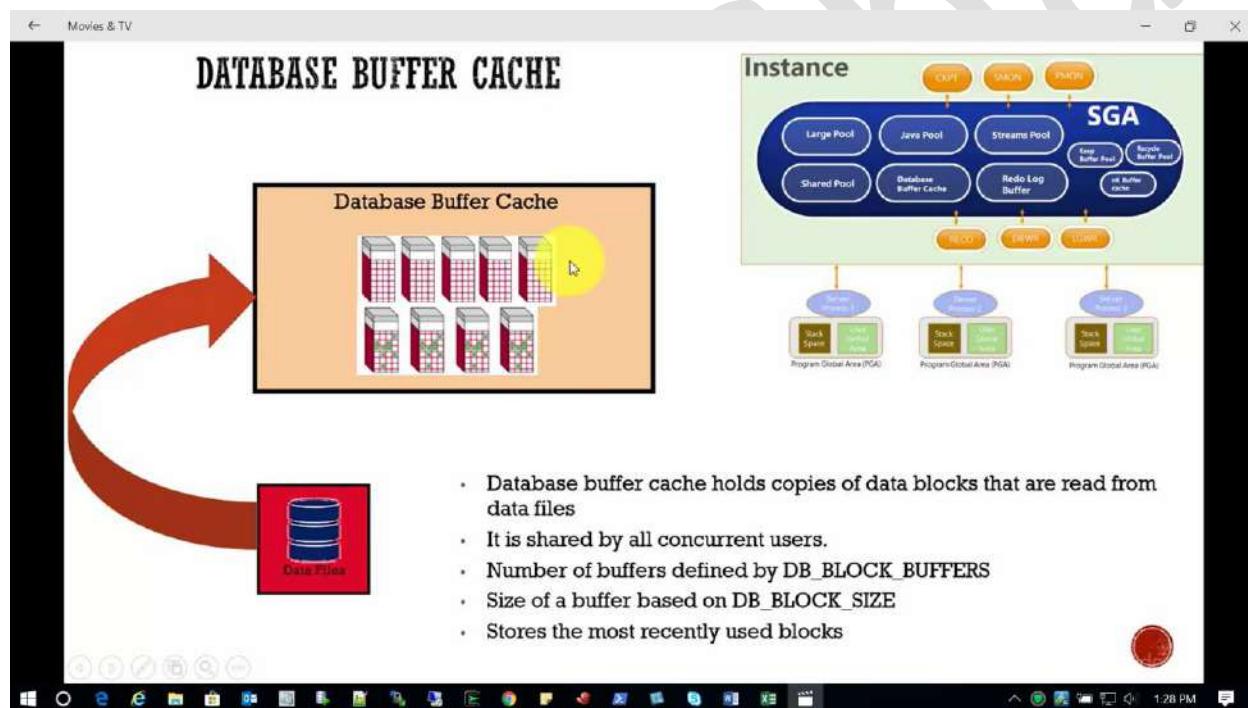
There are two types of memories

1. SGA (shared global area) → information shared among all processes
2. PGA (Programmable/Private global area) → information non-sharable is exclusive with server processes



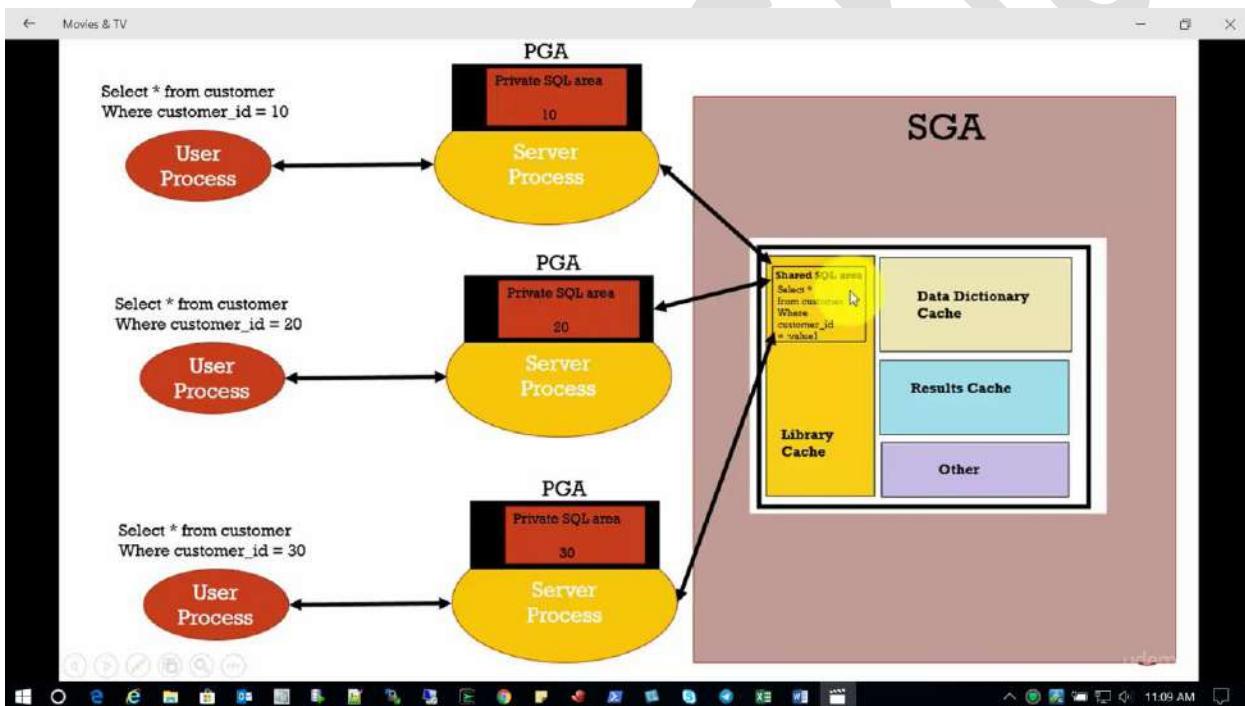
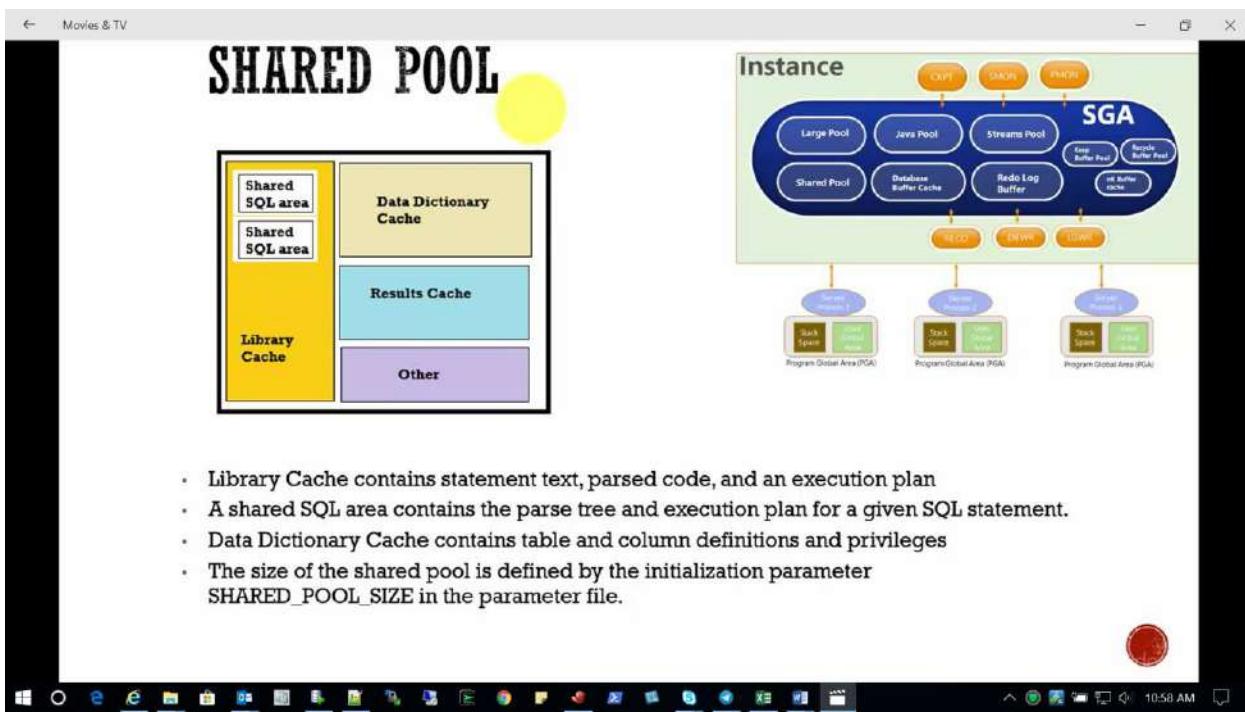
#### Database Buffer cache [DB\_BLOCK\_SIZE & DB\_BLOCK\_BUFFERS]

If user reads data, the data will be read from data files via data buffers in the form of data blocks buffer and size..



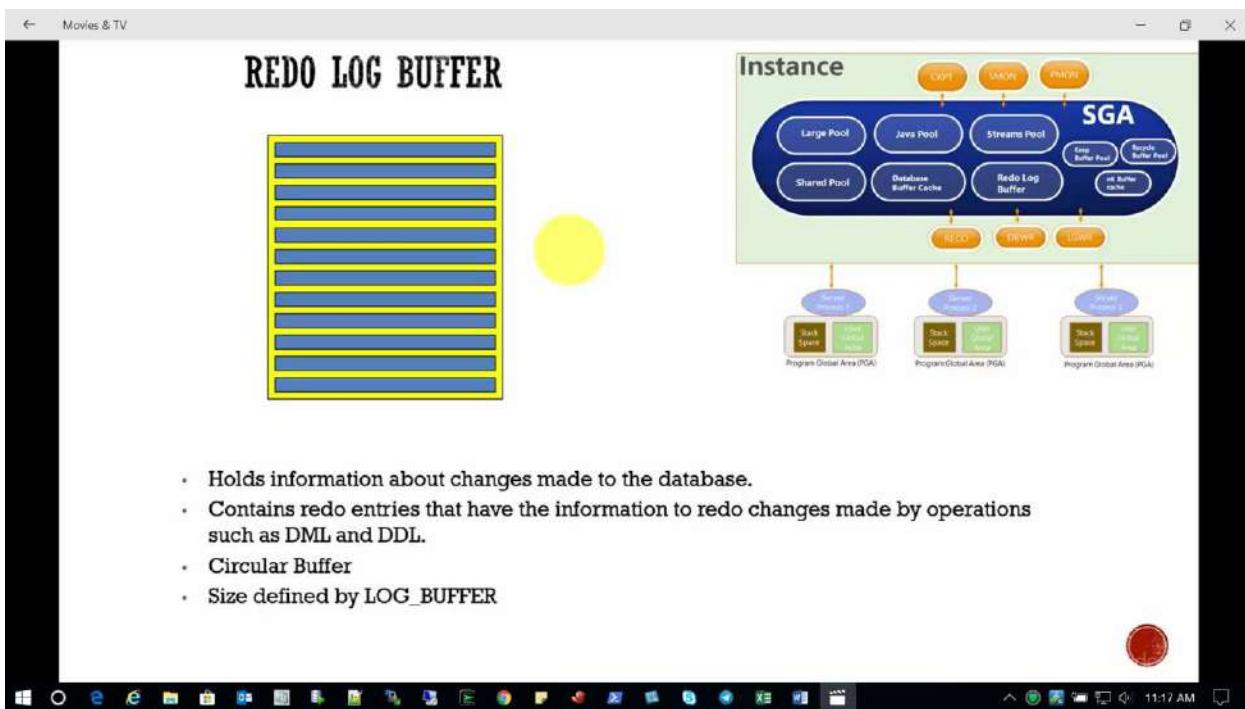
#### SHARED POOL [is a memory] [SHARED\_POOL\_SIZE]

One copy is stored in Library Cache (Present), and one other copy is stored in Data Dictionary Cache (Meta data). Results Cache is a both combination of Library Cache and Data Dictionary Cache.  
To get the Data Dictionary from sqldeveloper : dba\_[dba\_data\_file, dba\_free\_space]



#### Redo Log Buffer [LOG\_BUFFER]

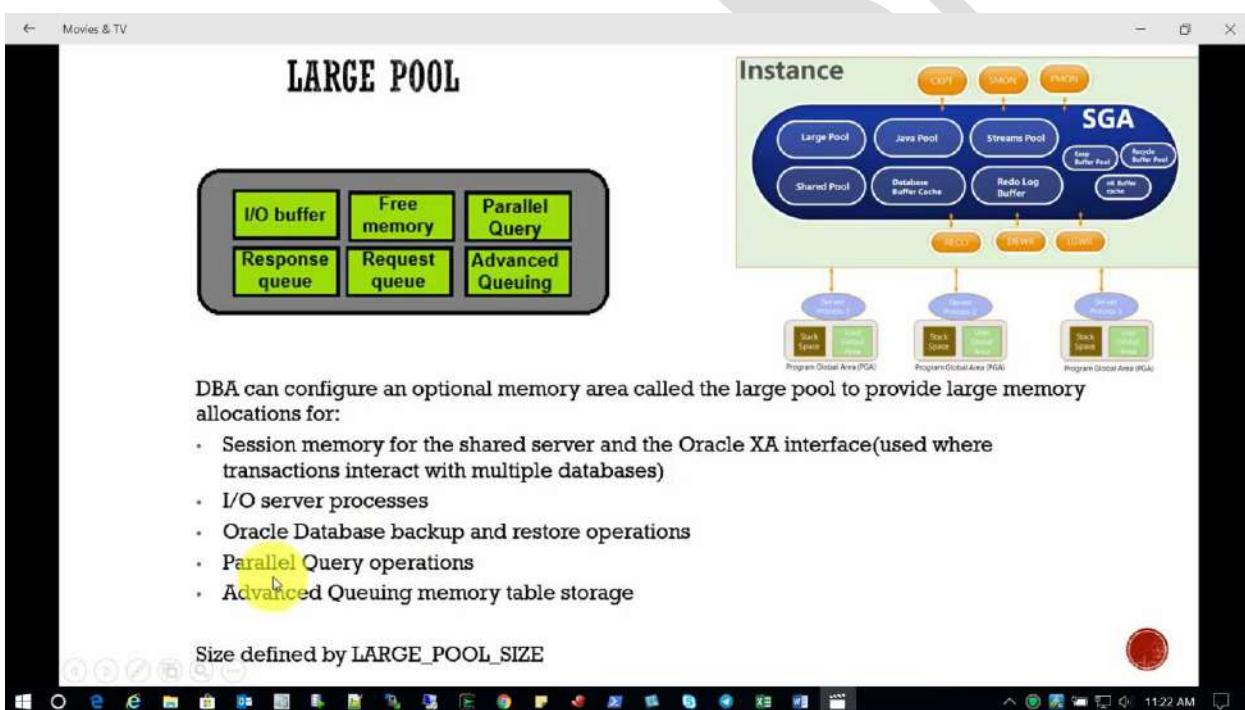
To help us for database recovery.



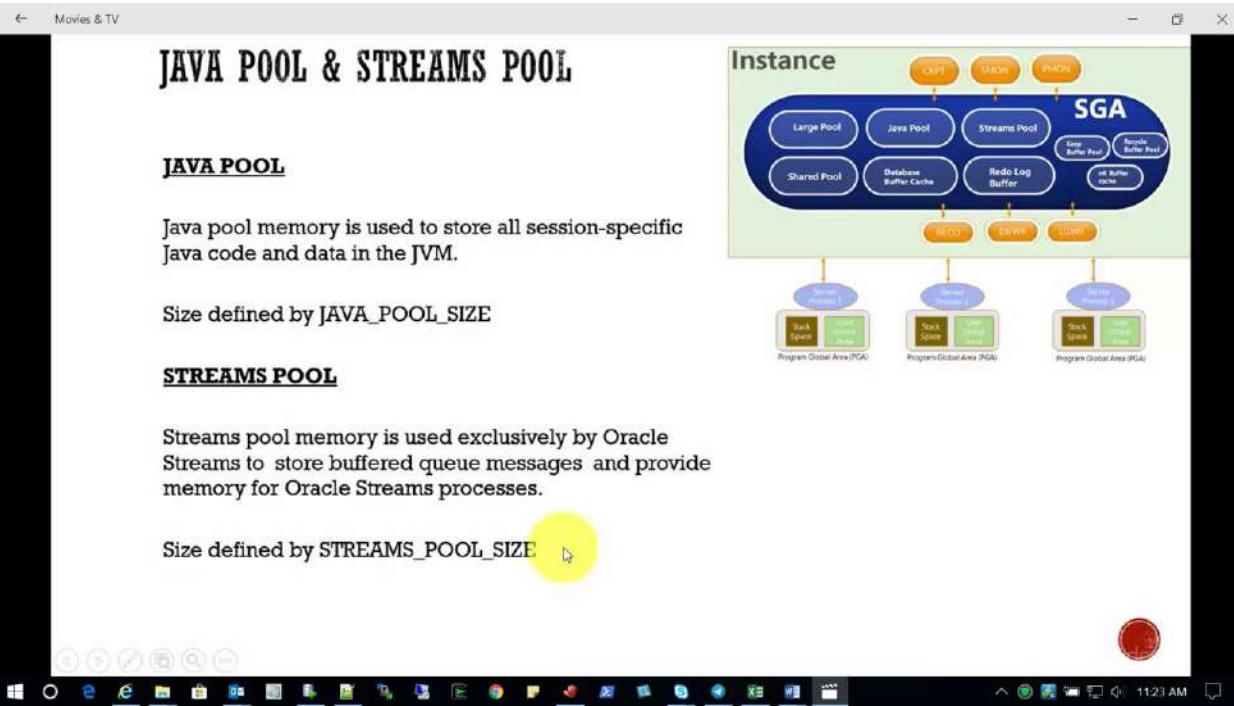
- Holds information about changes made to the database.
- Contains redo entries that have the information to redo changes made by operations such as DML and DDL.
- Circular Buffer
- Size defined by `LOG_BUFFER`

### Large pool [LARGE\_POOL\_SIZE]

As a DBA, sometimes we need an optional memory area needed, if you not configured Large pool it will take the help of shared pool.



### Java Pool & Streams Pool [STREAMS\_POOL\_SIZE]



### JAVA POOL

Java pool memory is used to store all session-specific Java code and data in the JVM.

Size defined by `JAVA_POOL_SIZE`

### STREAMS POOL

Streams pool memory is used exclusively by Oracle Streams to store buffered queue messages and provide memory for Oracle Streams processes.

Size defined by `STREAMS_POOL_SIZE`

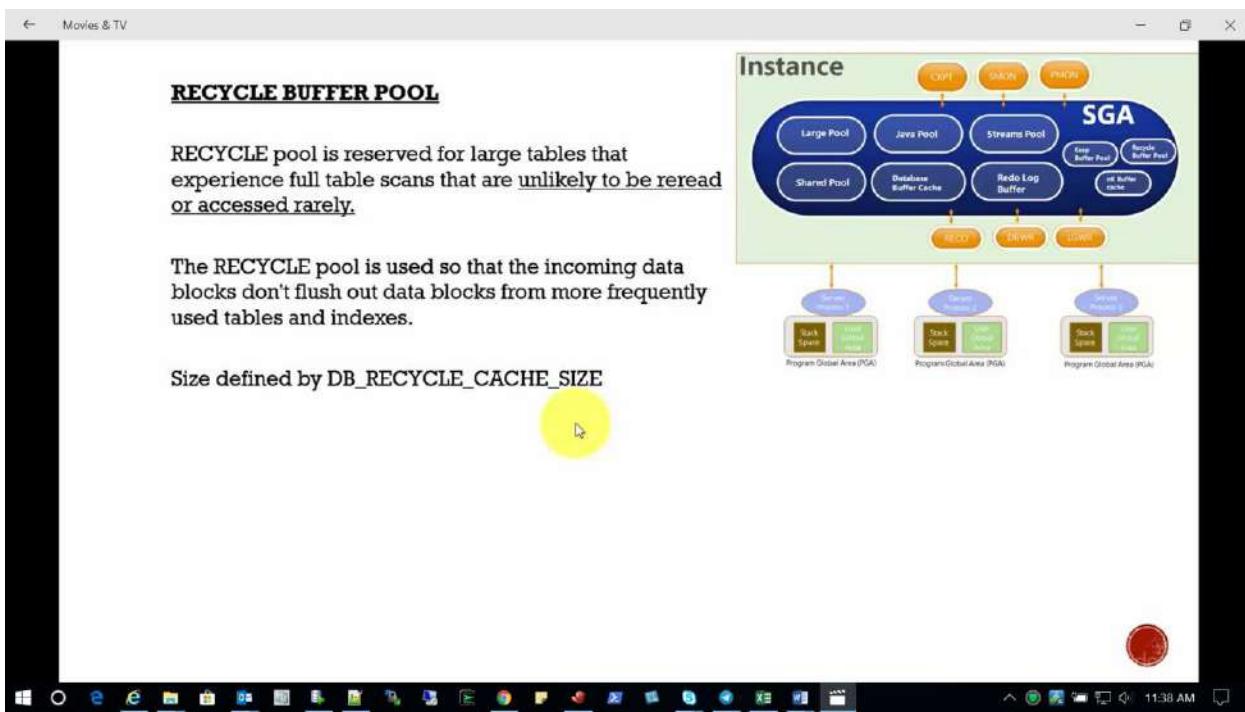
### Keep Buffer Pool [DB\_KEEP\_CACHE\_SIZE]

Keep Buffer Pool to overcome the issue of Database buffer Pool, it is used to keep frequently accessed blocks



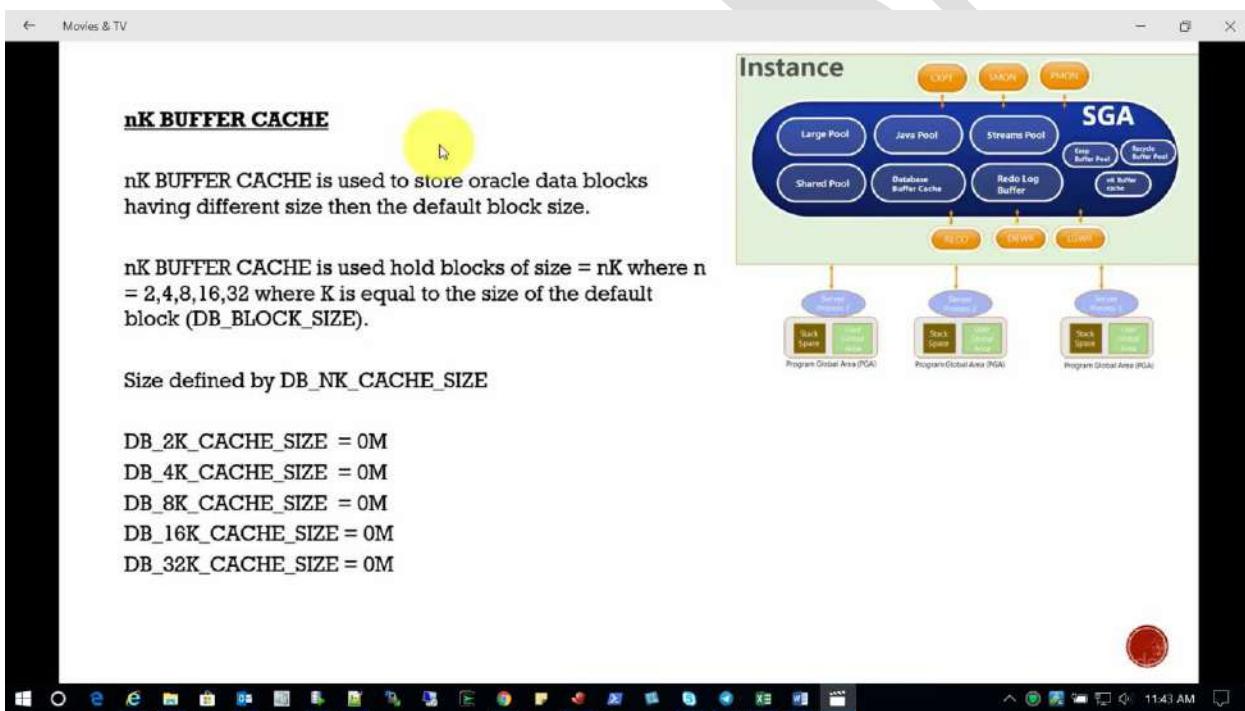
### Recycle Buffer Pool [DB\_RECYCLE\_CACHE\_SIZE]

Recycle buffer pool rarely used Data Blocks, if you need to scan big tables then it does not care more fluctuations for other table

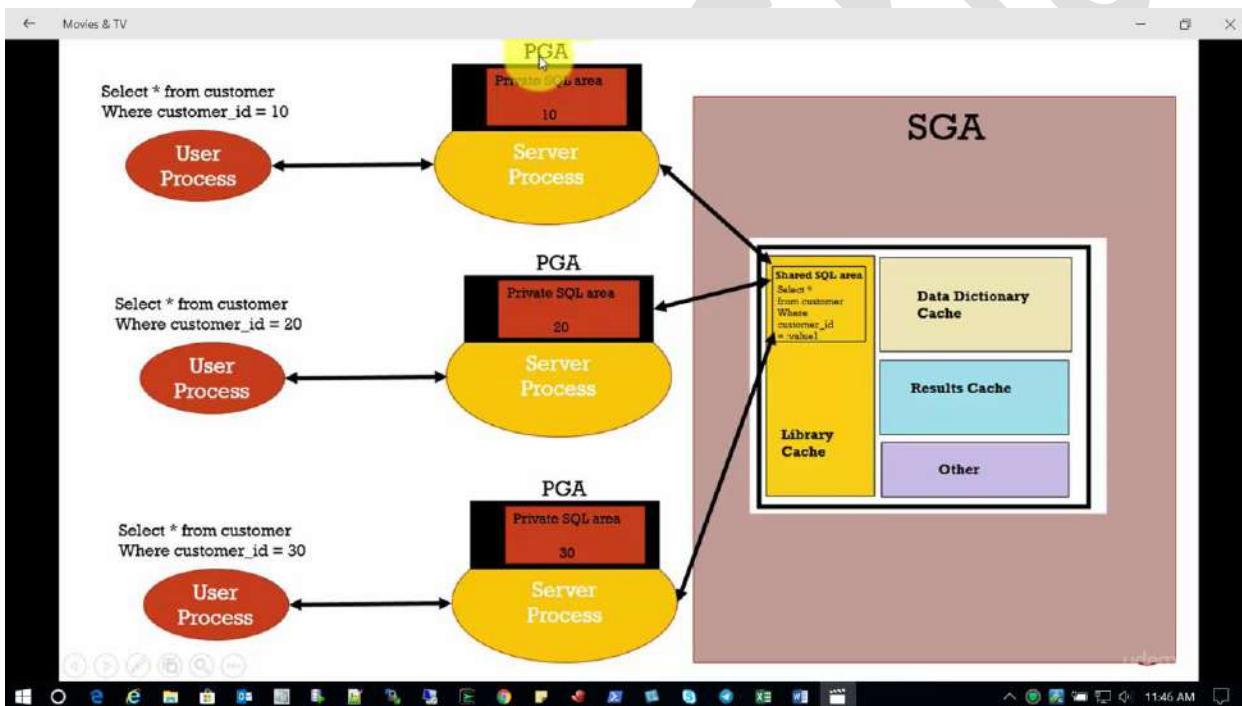
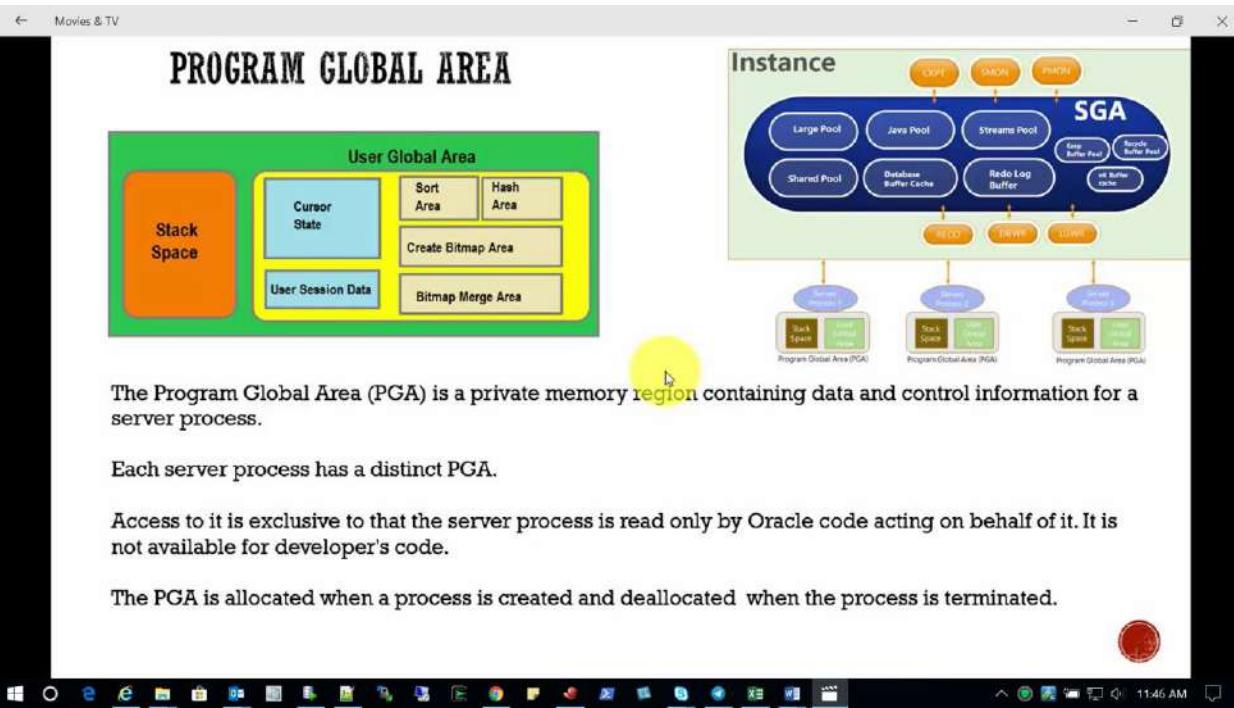


#### nK Buffer Cache [DB\_NK\_CACHE\_SIZE]

n is like number , k is a memory size. With this we can configure our data block size with the corresponding parameter value.

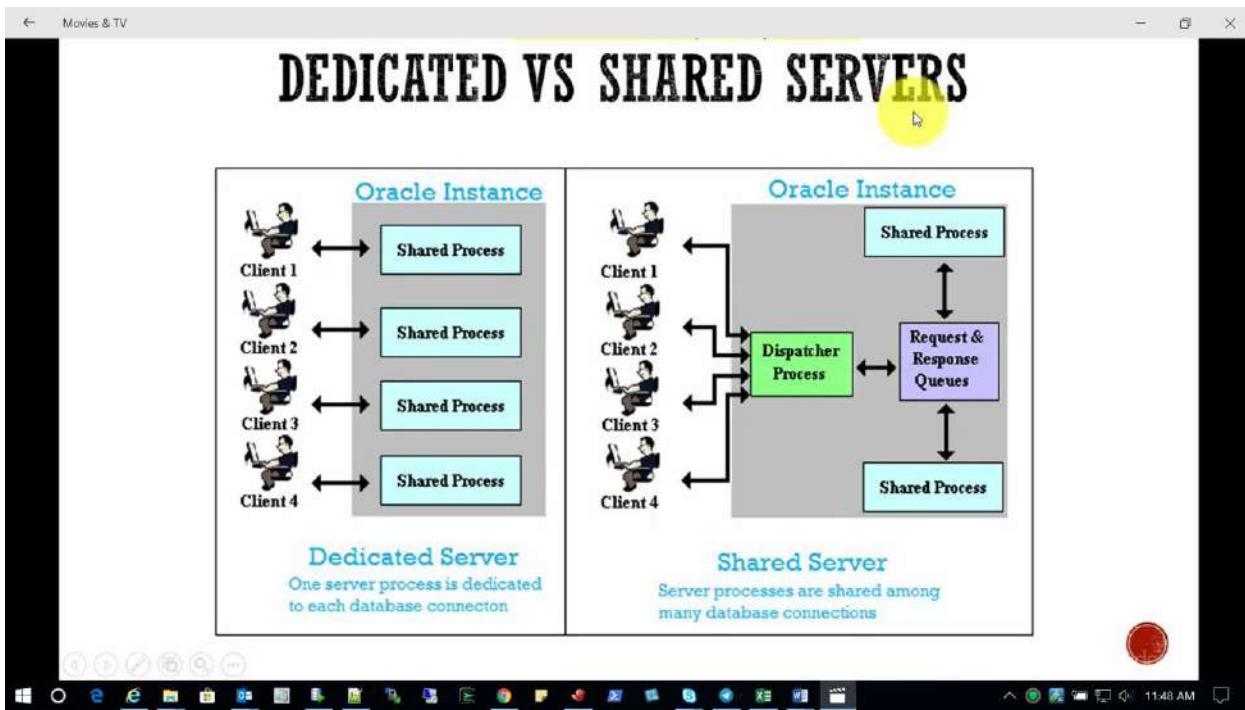


#### Program global Area



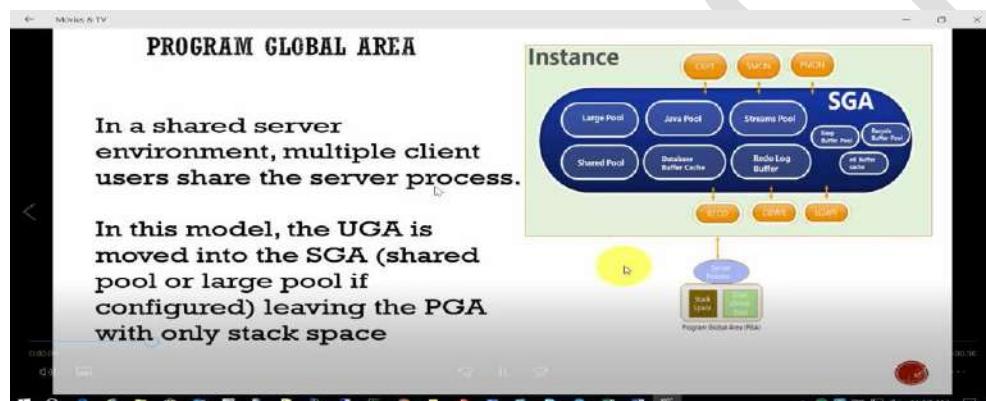
#### Dedicated Vs Shared Servers

Now a day all are going with dedicated servers only because of the memory is cheap, so we can get the huge memory capability servers.



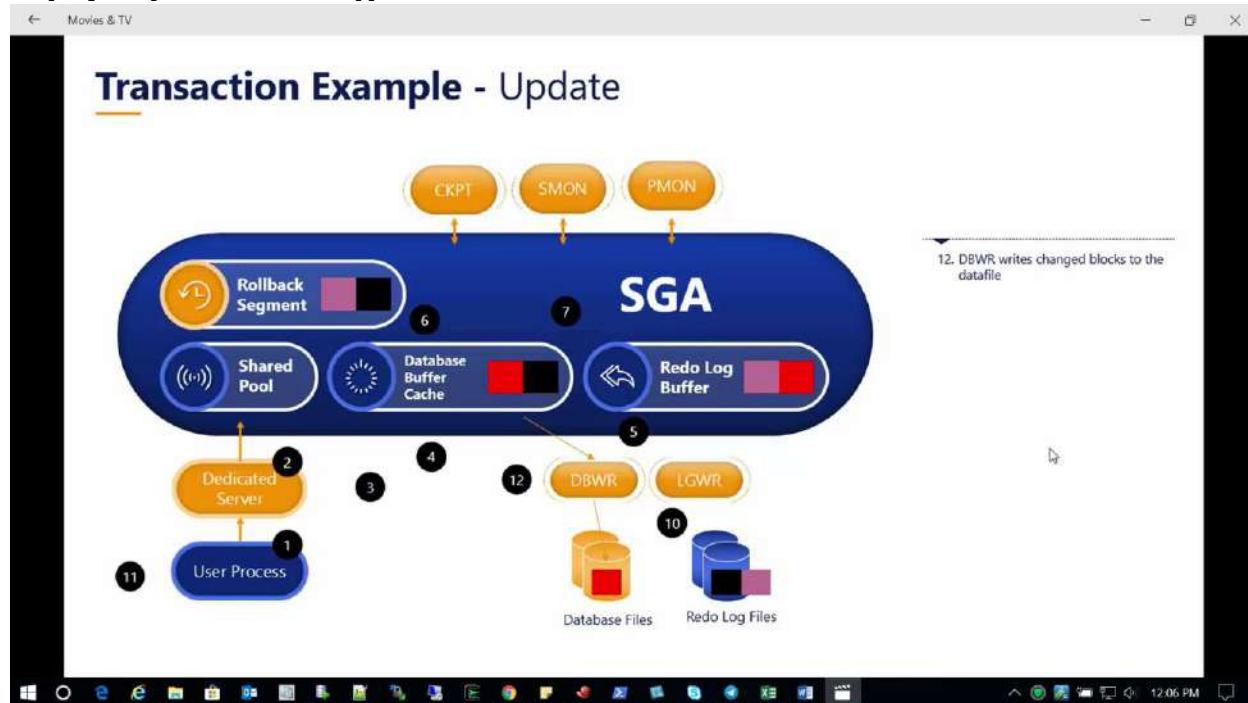
## Program Global Area in Shared server Environment

It contains a Stack Space.



## Database Transaction Example

Step by step transaction happened in DB



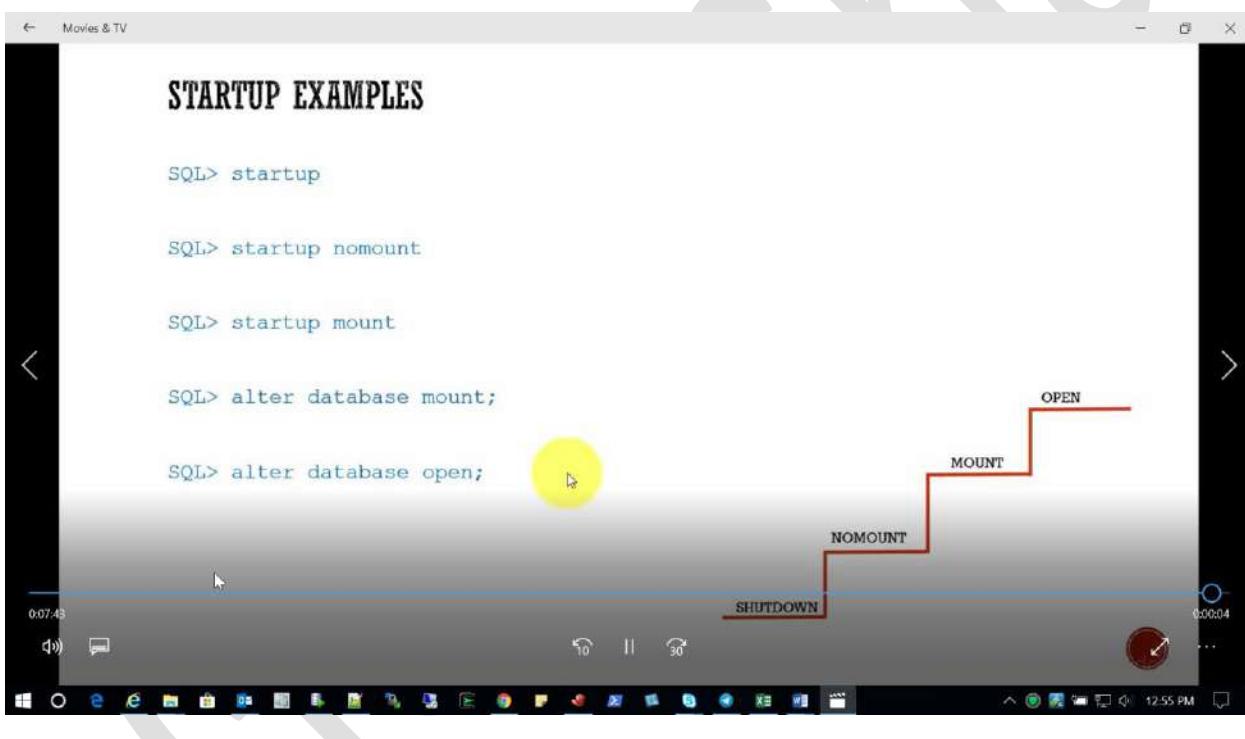
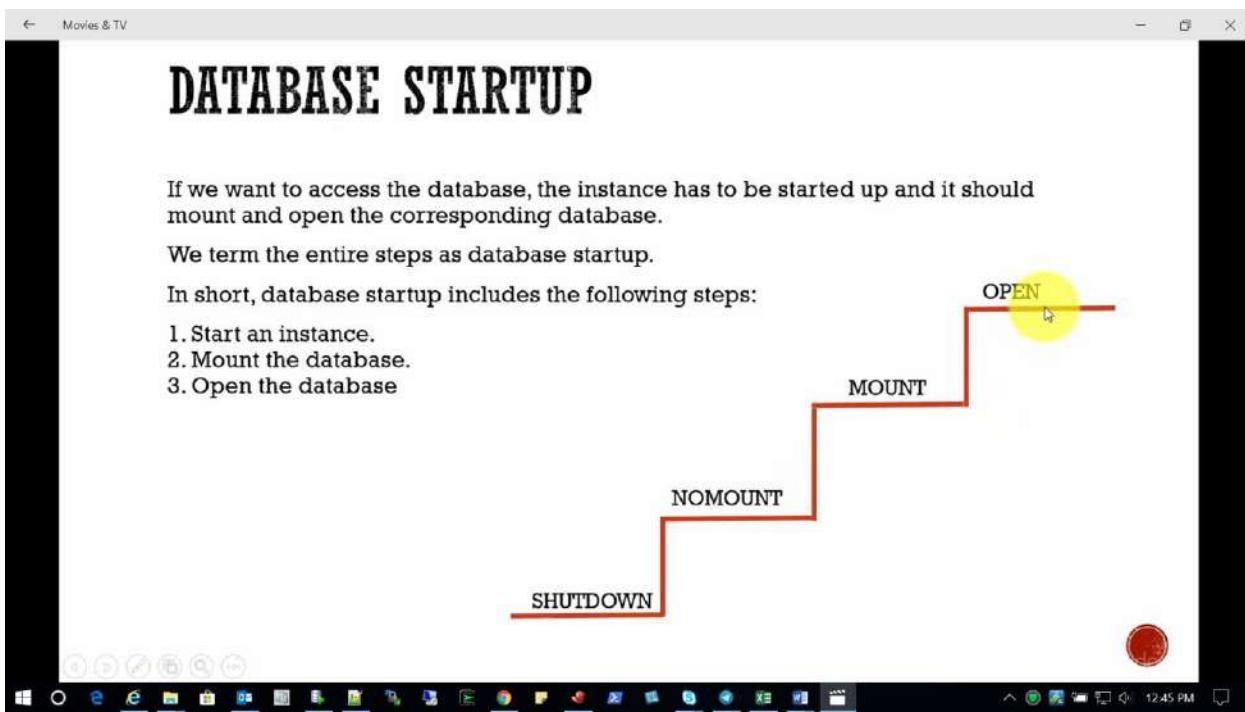
## Oracle Database Startup

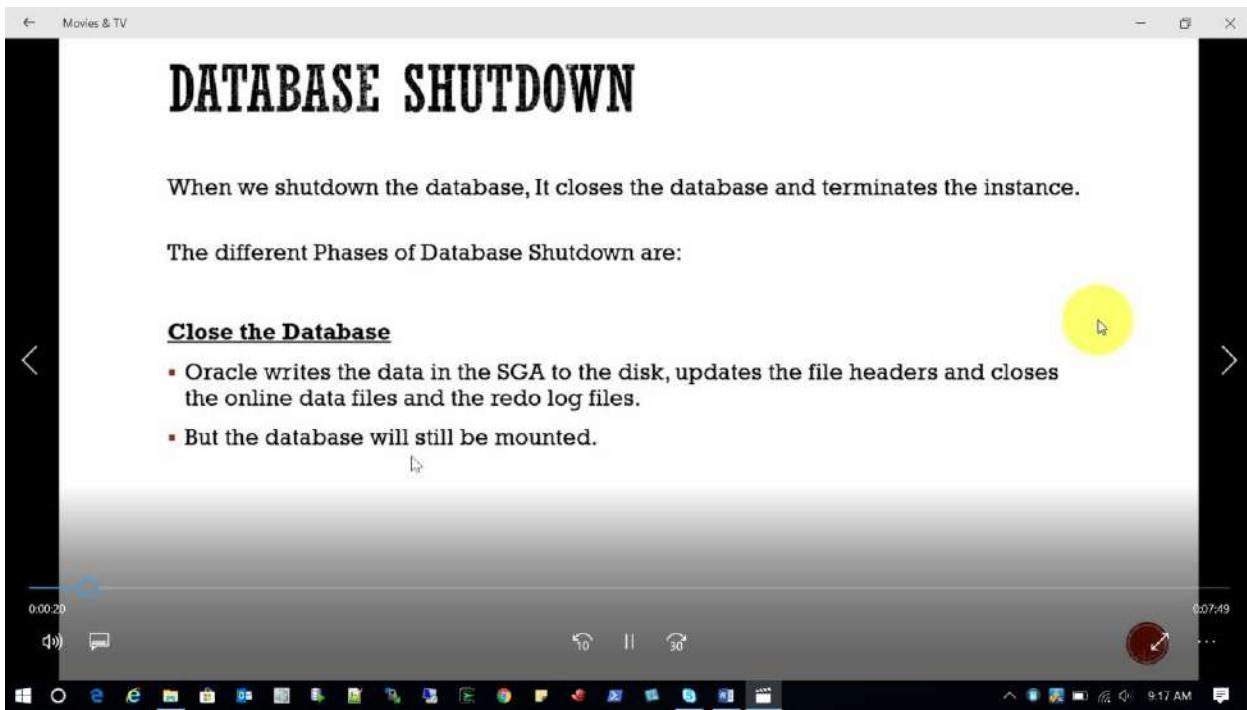
### ORACLE DATABASE

The Database is a set of physical operating system files. These files actually holds the user data and the metadata (or the data dictionary). Every running Oracle database is associated with (at least) an Oracle instance.

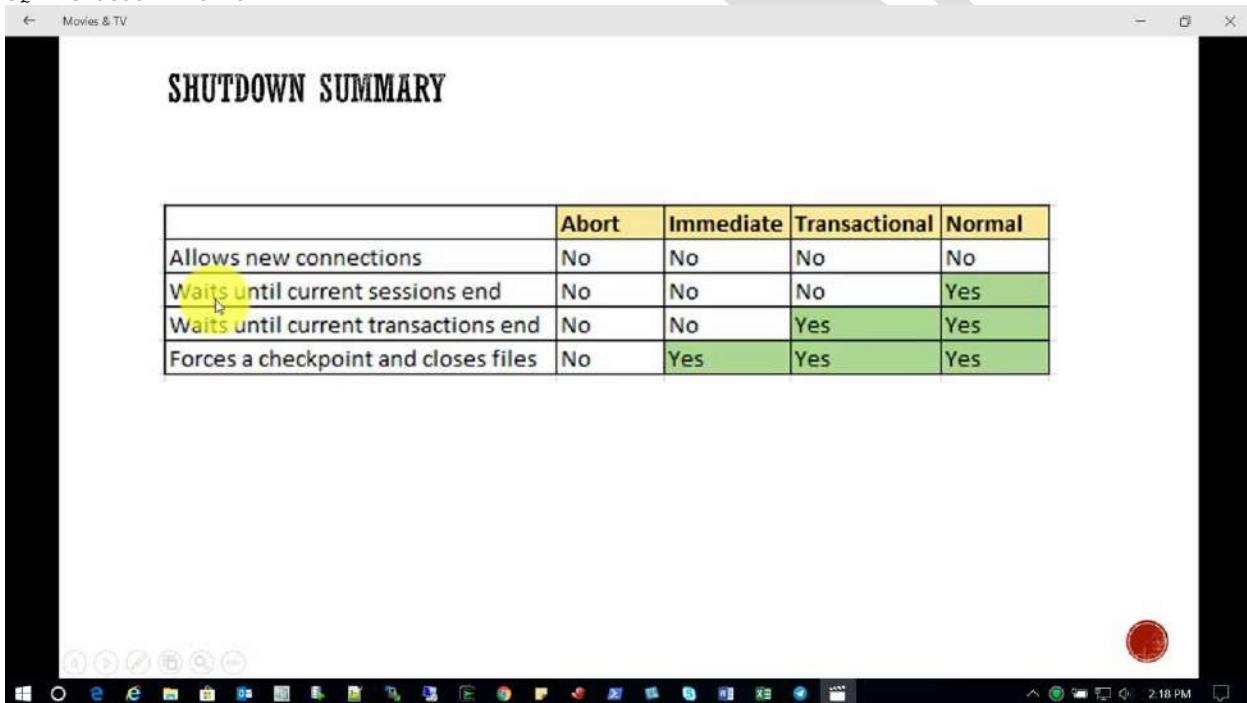
### ORACLE INSTANCE

The Instance refers to the set of Oracle background processes or threads and a shared memory area (SGA).





```
SQL> Shutdown Abort  
SQL> Shutdown immediate  
SQL> Shutdown Transactional  
SQL> Shutdown Normal
```



**Parameter File [Show parameter spfile]**  
More Initcdbl.ora file  
Files are two types  
1. Pfile (parameter file, is a text file)  
2. Spfile (server parameter file, is a binary file , note you cant edit.)

Movies & TV

# PARAMETER FILE

Parameter file is a text/binary file used to store the database initialization parameters.

The oracle instance reads the parameter file during startup which are then used to control the behavior of database instance.

It is also used to specify options like:

- memory allocation (SGA and PGA),
- startup of optional background processes,
- Setting of NLS parameters etc.

```
[oracle@linux dbs]$ more initprod1.ora
prod1._db_cache_size=226492416
prod1._java_pool_size=4194304
prod1._large_pool_size=8388608
prod1._oracle_base='/u01/app/oracle'#ORACLE_BASE set from environment
prod1._pga_aggregate_target=289406976
prod1._sga_target=545259520
prod1._shared_io_pool_size=0
prod1._shared_pool_size=238686720
prod1._streams_pool_size=4194304
*.audit_file_dest='/u01/app/oracle/admin/prod1/adump'
*.audit_trail='db'
*.compatible='11.2.0.4.0'
*.control_files='/disk2/prod1/control/control01.ctl','/disk2/prod1/contr
*.db_16k_cache_size=62914560
*.db_block_size=8192
*.db_domain=''
*.db_name='prod1'
*.db_recovery_file_dest='/disk2/prod1/fra'
*.db_recovery_file_dest_size=4385144832
*.diagnostic_dest='/u01/app/oracle'
*.dispatchers='(PROTOCOL=TCP) (SERVICE=prod1XDB)'
*.log_archive_dest_1='LOCATION=/disk2/prod1/arch/'
*.memory_target=834666496
*.open_cursors=300
*.processes=150
*.remote_login_passwordfile='EXCLUSIVE'
*.undo_retention=2400
*.undo_tablespace='UNDOTBS1'
```

9:09 AM

Movies & TV

# PARAMETER FILE

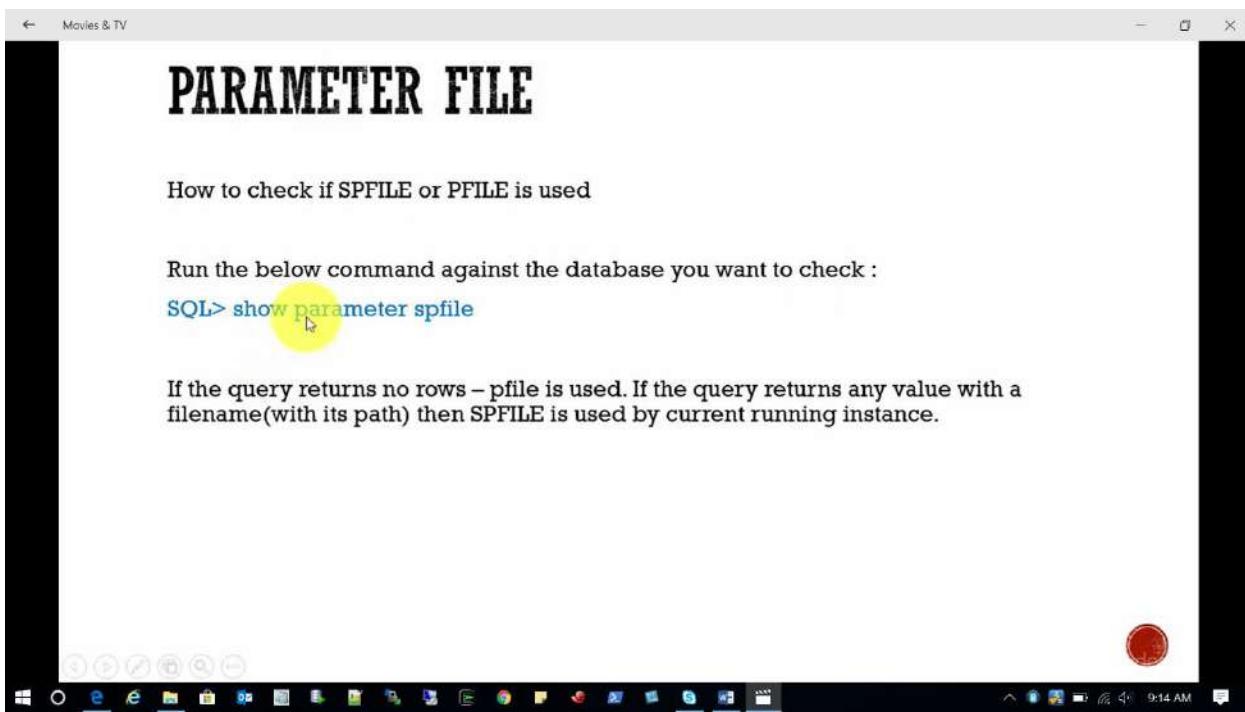
There are 2 types of parameter files.

1. pfile (parameter file) – older way [ not recommended by oracle ]
2. spfile (server parameter file) – newer way [ recommended by oracle ]

spfile was introduced starting from oracle 9i, until that time text based pfile was used to store database initialization parameters.

| PFILE                                                                                                  | SPFILE                                                                                                        |
|--------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------|
| Text file                                                                                              | Binary file                                                                                                   |
| Parameters in pfile can be edited using any text editor                                                | spfile can not be edited using a text editor. Instead it can only be altered using the "ALTER SYSTEM" command |
| Default location of pfile – \$ORACLE_HOME/dbs/init[SID].ora where [SID] – is the name of the instance. | Default location of spfile – \$ORACLE_HOME/dbs/spfile[SID].ora where [SID] – is the name of the instance.     |
| The RMAN utility can not take backup of a pfile                                                        | The RMAN utility can take backup of a spfile.                                                                 |

9:13 AM



How to rename a datafile in the primary database and manually  
[8:50 AM, 10/23/2018] +91 99002 12108: SOLUTION

The following steps describe how to rename a datafile in the primary database and manually propagate the changes to the standby database.

1. Set STANDBY\_FILE\_MANAGEMENT=MANUAL on both Primary and Standby Database.  
SQL>ALTER SYSTEM SET STANDBY\_FILE\_MANAGEMENT=MANUAL;
2. Take the Tablespace offline on the Primary Database:  
SQL> ALTER TABLESPACE tbs\_4 OFFLINE;
3. Rename Datafile on Primary Site:  
% mv /Disk1/oracle/oradata/mum/payroll\_1.dbf /Disk1/oracle/oradata/mum/payroll\_01.dbf
4. Rename the Datafile in the Primary Database and bring the Tablespace back online:  
SQL> ALTER TABLESPACE tbs\_4 RENAME DATAFILE '/Disk1/oracle/oradata/mum/payroll\_1.dbf' TO '/Disk1/oracle/oradata/mum/payroll\_01.dbf';  
SQL> ALTER TABLESPACE tbs\_4 ONLINE;
5. Stop Redo Apply on Standby Database:  
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE CANCEL;
6. Shutdown the Standby Database:  
SQL> SHUTDOWN;
7. Rename the Datafile at the Standby site :  
% mv /Disk1/oracle/oradata/mum/payroll\_1.dbf /Disk1/oracle/oradata/mum/payroll\_01.dbf
8. Start and mount the Standby Database:  
SQL> STARTUP MOUNT;
9. Rename the Datafile in the Standby Database control file.  
SQL> ALTER DATABASE RENAME FILE '/Disk1/oracle/oradata/mum/payroll\_1.dbf' TO '/Disk1/oracle/oradata/mum/payroll\_01.dbf';  
Note : STANDBY\_FILE\_MANAGEMENT initialization parameter must be set to MANUAL.
10. On the Standby Database, restart Redo Apply:  
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT FROM SESSION;
11. Set STANDBY\_FILE\_MANAGEMENT=AUTO on both Primary and Standby Database.  
SQL>ALTER SYSTEM SET STANDBY\_FILE\_MANAGEMENT=AUTO;  
Its for 11g  
Move a Datafile Online (Doc ID 1566797.1)

#### How To Check Oracle Database Memory and Storage Size

On certain occasions, I will be asked for an estimated resource requirements for a database that will be setup for a development environment or as a standby database. The following are my methods to calculate the memory and storage size based on the existing production database. The calculation on the storage size are meant for UNIX and LINUX.

#### 1. Get the estimated memory footprint of an existing database.

---- Get the SGA footprint of a database instance:  
SQL> SELECT sum(value)/1024/1024 "TOTAL SGA (MB)" FROM v\$sga;

```
---- Get the current PGA consumption of a database instance:
```

```
select sum(pga_max_mem)/1024/1024 "TOTAL MAX PGA (MB)" from v$process;
```

A more detailed breakdown of PGA memory usage:

```
SELECT spid, program,
       pga_max_mem      max,
       pga_alloc_mem    alloc,
       pga_used_mem     used,
       pga_freeable_mem free
  FROM V$PROCESS;
```

NOTE: The resulting program name having no associated process name (ex. PMON, SMON, RECO,...) is a session process.

Depending on the settings of the init parameter processes, it could be that the queries above only captures the active connections.

```
---- Get the estimated memory requirement for the unconnected sessions:
```

The query below gives the max allocated memory by a user session. We'll use this to calculate the remaining memory requirements for the unconnected processes.

```
select max(p.pga_max_mem)/1024/1024 "PGA MAX MEMORY OF USER SESSION (MB)"
  from v$process p, v$session s
 where P.ADDR = S.paddr and s.username is not null;
```

```
Get the maximum number of processes that an Oracle instance can handle.
```

```
select name, value from v$parameter
where name = 'processes';
```

```
Let's count the number of connected sessions:
```

```
select count(1) from v$session
where username is not null;
```

Get the total available connections by subtracting the connected sessions count from the processes parameter value.

The resulting value shall be multiplied by the resulting max allocated memory by a session done by the previous query.

This would then give you the estimated amount of reserve memory needed to accommodate additional connections.

```
ESTIMATED TOTAL MEMORY REQUIREMENT
```

SGA + PGA = EST MEMORY REQUIREMENT FOR CURRENT CONNECTIONS

SGA + PGA + UNCONNECTED SESSIONS = EST MEMORY REQUIREMENT AT MAXIMUM PROCESS UTILIZATION

where:

Unconnected Sessions (MB) = (processes - connected sessions) \* pga max memory of user session

```
2. Get the estimated storage sizing requirements of a database.
```

The procedure below in determining the storage size are meant for UNIX and LINUX environment. As for windows, just right click and select the properties on each of the drive letters given by the results of the queries.

@primary server,

```
-- Copy and paste each resulting commands onto a shell script and execute in primary server to get the mount point size
```

```
select unique 'df -k '||a.MTPPOINT MOUNT_POINT
```

```

from ( select substr(FILE_NAME,0,instr(FILE_NAME,'/',1,2)-1) MTPOINT
      from dba_data_files
      union
      select substr(FILE_NAME,0,instr(FILE_NAME,'/',1,2)-1) MTPOINT
      from dba_temp_files
      union
      select substr(MEMBER,0,instr(MEMBER,'/',1,2)-1) MTPOINT
      from v$logfile
      union
      select substr(NAME,0,instr(NAME,'/',1,2)-1) MTPOINT
      from v$controlfile
) a;

--- Query all parameter file destination and determine if they are of different path from the datafile mount
point or ORACLE_BASE dir.
      get the mount point size of file destinations

select name, value
from v$parameter
where (regexp_like(name, '^log_archive_(dest|dest\_[1-9]))$', 'i') or name like '%dump_dest' or name like
'%file_dest' or name like 'diag%dest' or name ='utl_file_dir') and value is not null;

--- Query the database directories for mount points

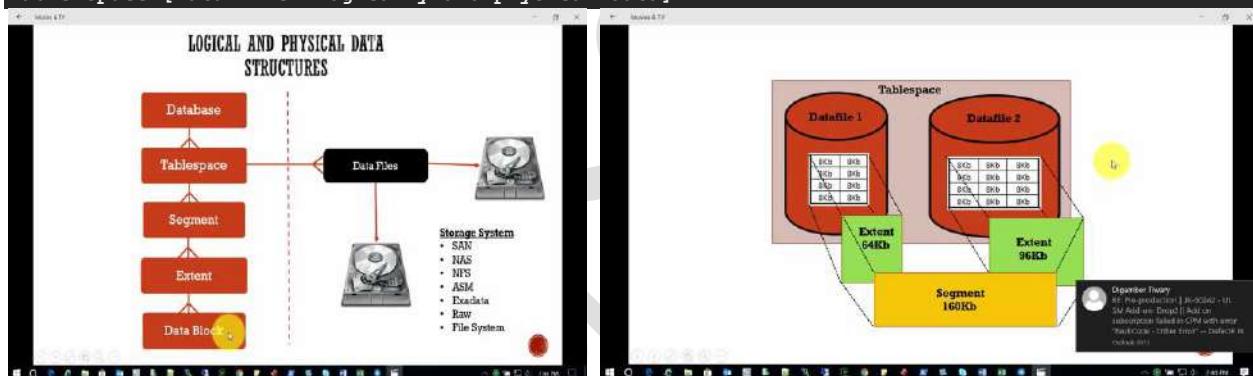
select 'df -k '||substr(DIRECTORY_PATH,0,instr(DIRECTORY_PATH,'/',1,2)-1) MTPOINT from dba_directories;

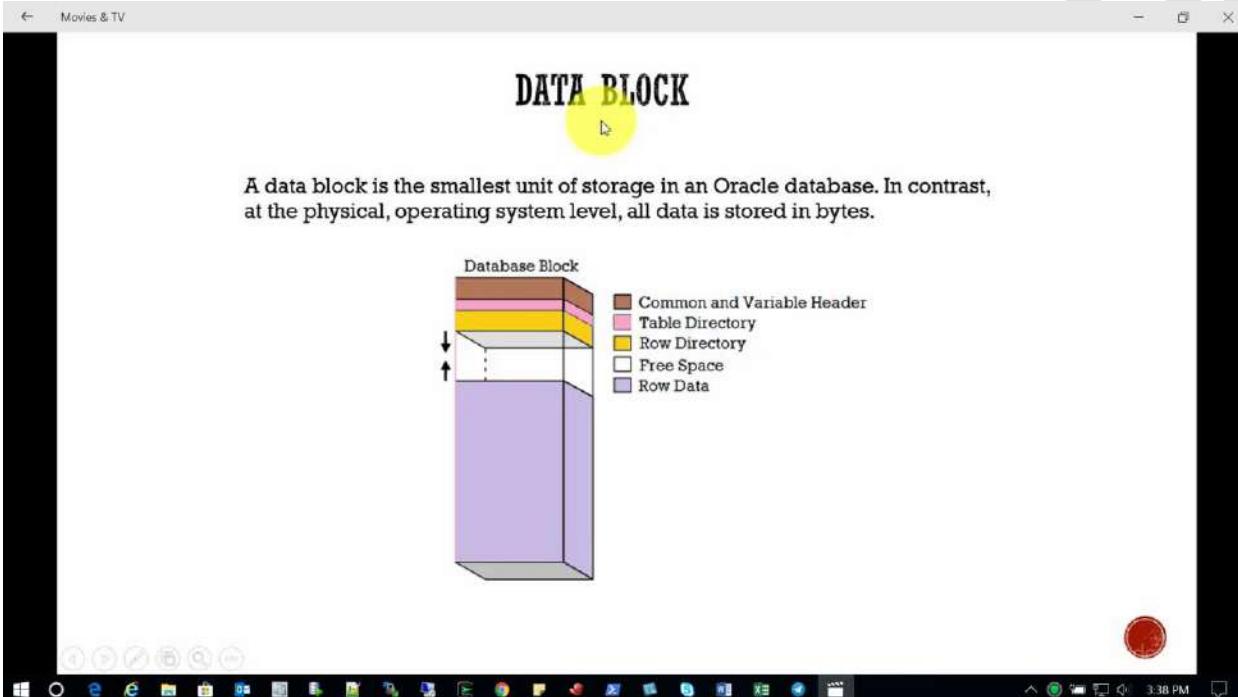
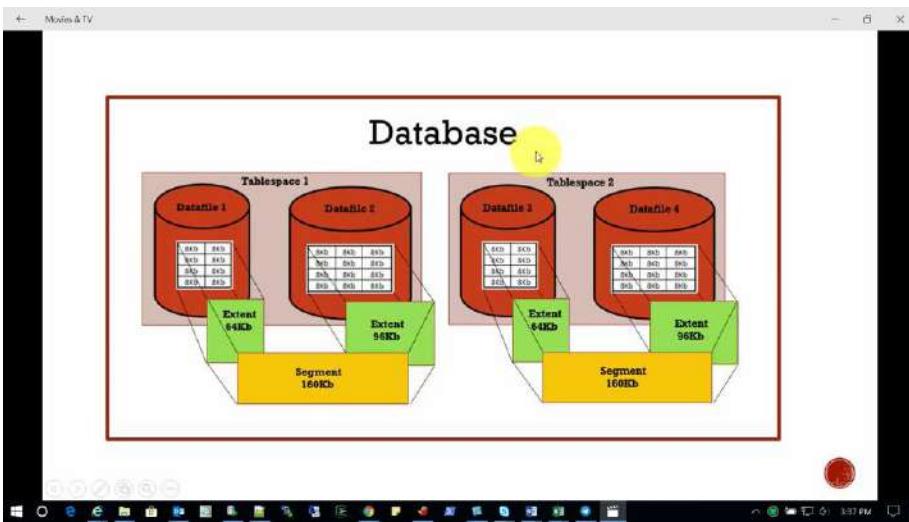
An alternative for a df command is to grep all the valid database related mount point in a single string to get
a much more concise report.
example:
$ df -k | grep -E '/u01|/u02|/utldir|/opt'

```

#### Oracle Tablespace Management

##### Table space [Data File= Logically and physical data]





#### TableSpaces Types

- Permanent : is used to store all permanent data like tables data
- UNDO : is used to provide the rollback the transactions
- Temporary : is used to sort and join operations

#### Default Tablespaces comes from installations:

- SYSTEM : is used by oracle servers, contains database information not recommended to store user data in this tablespace
- SYSAUX : system auxiliary tablespace is mandatory in 12c, same properties of SYSTEM tablespace
- TEMP : to store temporary data
- UNDOTBS1 : is used to store undo data
- USERS : it will store all users tables data in users tablespace.
- EXAMPLE : sample schema's , at the time of installation you can create your own.

#### Online and Offline TableSpace

- Online Tablespace: is up and running and database can read/write into to the tablespaces.
- Offline Tablespace: is offline, database cannot read/write into the tablespaces.
  - Offline tablespace backup.
  - Recover an individual tablespace or datafile.
  - Move a datafile without closing the database.

#### How to Create TableSpace

```
Create tablespace tbs1
datafile '/disk1/dev/data/data01.dbf'
```

```
size 50m  
autoextend on  
next 512k  
maxsize 250m;
```

#### How to add more space to Database

Method's to add more space for your database

- Create a new tablespace

```
Create tablespace tbs1 datafile '/disk1/dev/data/data01.dbf' size 50m autoextend on next 512k  
maxsize 250m;
```

- Add a datafile to an existing tablespace. It will be done with Alter  
`alter tablespace tbs1 add datafile '/disk1/dev/data/data01.dbf'`

- Increase the size of existing data file. It will be done with Alter  
`alter database datafile '/disk1/dev/data/data01.dbf' resize 100m;`

- Configure the dynamic growth of data file using "AUTOEXTEND on". It will be done with Alter

```
alter tablespace tbs1 add datafile '/disk1/dev/data/data01.dbf' size 50m autoextend on next 512k maxsize 250m;
```

#### ! Practiced on oraclelinux7 VM [Tablespace]

Trying to create a tablespace, but it's getting ORA-01119, ORA-17610, ORA-27037 and no such file or directory issue so then I created a table space with size and executed fine.

```
SQL> create tablespace tbs1 datafile '/u01/app/oracle/oradata/cdb1/data01.dbf';  
create tablespace tbs1 datafile '/u01/app/oracle/oradata/cdb1/data01.dbf'  
*
```

ERROR at line 1:

```
ORA-01119: error in creating database file '/u01/app/oracle/oradata/cdb1/data01.dbf'  
ORA-17610: file '/u01/app/oracle/oradata/cdb1/data01.dbf' does not exist and no size specified  
ORA-27037: unable to obtain file status  
Linux-x86_64 Error: 2: No such file or directory
```

Additional information: 7

#### Tablespace1: [with autoextent]

```
SQL> Create tablespace tbs1 datafile '/u01/app/oracle/oradata/cdb1/data01.dbf' size 50m autoextend on next 512k  
maxsize 250m;  
Tablespace created.
```

#### Tablespace2: [without autoextent]

```
SQL> create tablespace tbs2 datafile '/u01/app/oracle/oradata/cdb1/data02.dbf' size 5m;  
Tablespace created.
```

If the tablespace tbs1 contains only one datafile then go for below syntax To drop table space tbs1:

```
DROP TABLESPACE temp_tablespace including contents and datafiles;  
SQL>Drop tablespace tbs1 including contents and datafiles;  
SQL> select tablespace name, file name, bytes/1024/1024 from dba data files;
```

```
SQL> drop tablespace tbs1 including contents and datafiles;
```

Tablespace dropped.

```
SQL> select tablespace_name, file_name, bytes/1024/1024 from dba_data_files;
```

| TABLESPACE_NAME | FILE_NAME                                  | BYTES/1024/1024 |
|-----------------|--------------------------------------------|-----------------|
| SYSTEM          | /u01/app/oracle/oradata/cdb1/system01.dbf  | 810             |
| SYSAUX          | /u01/app/oracle/oradata/cdb1/sysaux01.dbf  | 480             |
| UNDOTBS1        | /u01/app/oracle/oradata/cdb1/undotbs01.dbf | 70              |
| USERS           | /u01/app/oracle/oradata/cdb1/users01.dbf   | 5               |
| TBS2            | /u01/app/oracle/oradata/cdb1/data02.dbf    | 5               |

```
SQL> drop tablespace tbs2 including contents and datafiles;
```

Tablespace dropped.

```
SQL> select tablespace_name, file_name, bytes/1024/1024 from dba_data_files;
```

| TABLESPACE_NAME | FILE_NAME                                  | BYTES/1024/1024 |
|-----------------|--------------------------------------------|-----------------|
| USERS           | /u01/app/oracle/oradata/cdb1/users01.dbf   | 5               |
| UNDOTBS1        | /u01/app/oracle/oradata/cdb1/undotbs01.dbf | 70              |
| SYSTEM          | /u01/app/oracle/oradata/cdb1/system01.dbf  | 810             |
| SYSAUX          | /u01/app/oracle/oradata/cdb1/sysaux01.dbf  | 480             |

```
SQL> |
```

To Rename the datafile for an existing tbspl for multiple datafiles

Step1:

```
SQL> alter tablespace tbs1 offline;
```

```
Tablespace altered.
```

Step2:

```
[oracle@orainux7 cdb1]$ mv /u01/app/oracle/oradata/cdb1/data02.dbf /u01/app/oracle/oradata/cdb1/data03.dbf
```

Step3:

```

SQL> alter tablespace TBS1 online;
alter tablespace TBS1 online
+
ERROR at line 1: [REDACTED]
ORA-01157: cannot identify/lock data file 16 - see DBWR trace file
ORA-01110: data file 16: '/u01/app/oracle/oradata/cdb1/data02.dbf'

```

#### Errors:

ORA-01157: cannot identify/lock data file 16 - see DBWR trace file  
 ORA-01110: data file 16: '/u01/app/oracle/oradata/cdb1/data02.dbf'

#### Solution:

##### Step3 repeat

```

SQL> alter database rename file '/u01/app/oracle/oradata/cdb1/data02.dbf' to '/u01/app/oracle/oradata/cdb1/data03.dbf'
2 ;

```

Database altered.

Step4: now the tablespace got in **online**.

```
SQL> alter tablespace tbs1 online;
```

Tablespace altered.

```
SQL> select tablespace_name, file_name, bytes/1024/1024 from dba_data_files;
```

| TABLESPACE_NAME | FILE_NAME                                  | BYTES/1024/1024 |
|-----------------|--------------------------------------------|-----------------|
| SYSTEM          | /u01/app/oracle/oradata/cdb1/system01.dbf  | 810             |
| SYSAUX          | /u01/app/oracle/oradata/cdb1/sysaux01.dbf  | 480             |
| UNDOTBS1        | /u01/app/oracle/oradata/cdb1/undotbs01.dbf | 70              |
| USERS           | /u01/app/oracle/oradata/cdb1/users01.dbf   | 5               |
| TBS1            | /u01/app/oracle/oradata/cdb1/data01.dbf    | 10              |
| TBS1            | /u01/app/oracle/oradata/cdb1/data03.dbf    | 10              |

6 rows selected.

To create a tablespace with blocksize

First check the present blocksize configured for your database.

```
SQL> show parameter db block size;
```

| NAME          | TYPE    | VALUE |
|---------------|---------|-------|
| db_block_size | integer | 8192  |

It showing 8kb, try to create tablespace with 16k. if not created please configure 16k for the your database.

```
SQL> create tablespace tbs1 datafile '/u01/app/oracle/oradata/cdb1/data02.dbf' size 10m blocksize 16k;
```

```
create tablespace tbs1 datafile '/u01/app/oracle/oradata/cdb1/data02.dbf' size 10m blocksize 16k;
```

```
create tablespace tbs1 datafile '/u01/app/oracle/oradata/cdb1/data02.dbf' size 10m blocksize 16k
```

```
*
```

ERROR at line 1:

ORA-29339: tablespace block size 16384 does not match configured block sizes

```
SQL> alter system set db_16k_cache_size=60m scope=both;
```

```
SQL> alter system set db_16k_cache_size=60m scope=both;
```

System altered.

Here scope both is the value for present 8k and newly configured 16k. and the configured with the size of 60mb for db\_16k\_cache\_size for the system data dictionary.

Now execute the step for creation of tablespace.

```
SQL> create tablespace tbs1 datafile '/u01/app/oracle/oradata/cdb1/data02.dbf' size 10m blocksize 16k;
```

```
create tablespace tbs1 datafile '/u01/app/oracle/oradata/cdb1/data02.dbf' size 10m blocksize 16k
```

```
*
```

ERROR at line 1:

ORA-01543: tablespace 'TBS1' already exists

You have to create new tablespace, for existing it can't create and also you can do it by alter

```
CREATE TABLESPACE lmtbsb DATAFILE '/u02/oracle/data/lmtbsb01.dbf' SIZE 50M
EXTENT MANAGEMENT LOCAL UNIFORM SIZE 128K
BLOCKSIZE 8K;
```

You can not change Blocksize on exist tablespace.

But You need to use 32k blocksize some tables.

you can change db\_32k\_cache\_size initialization parameter. and stop/start DB.

```
SQL> alter system set db_32k_cache_size=5G ;
```

```
*** db_32k_cache_size ,that share memory with sga_target ***
```

after that, create new tablespace (32k) and move tables or indexes to new tablespace

Example:

```
SQL> create tablespace test datafile '/oradata/testdb/test01.dbf' size 10M BLOCKSIZE 32k;
SQL> alter table test move tablespace test;
```

- The SGA (System Global Area) has a separate buffer cache for each block size:

DB\_2K\_CACHE\_SIZE

DB\_4K\_CACHE\_SIZE

DB\_8K\_CACHE\_SIZE

DB\_16K\_CACHE\_SIZE

DB\_32K\_CACHE\_SIZE

**To check the space on DB [DBA\_TABLESPACE\_USAGE\_METRICS]**

```
SQL>select * From DBA_TABLESPACE_USAGE_METRICS;
```

### [BestPractice]TableSpace Management Demo

Let's see what the tablespaces in db. And let's see what are the tablespaces and datafiles in db. And let's see the columns with the help of data dictionary table

```
SQL> desc dba_data_files
```

```
SQl> desc dba_data_files
Name          Null?    Type
-----  -----
FILE_NAME      VARCHAR2(513)
FILE_ID        NUMBER
TABLESPACE_NAME VARCHAR2(30)
BYTES          NUMBER
BLOCKS         NUMBER
STATUS         VARCHAR2(9)
RELATIVE_FNO   NUMBER
AUTOEXTENSIBLE VARCHAR2(3)
MAXBYTES       NUMBER
MAXBLOCKS     NUMBER
INCREMENT_BY   NUMBER
USER_BYTES     NUMBER
USR_BLOCKS    VARCHAR2(7)
```

### Tablespace Demo 1

```
SQL> select tablespace_name, file_name, bytes from dba_data_files;
```

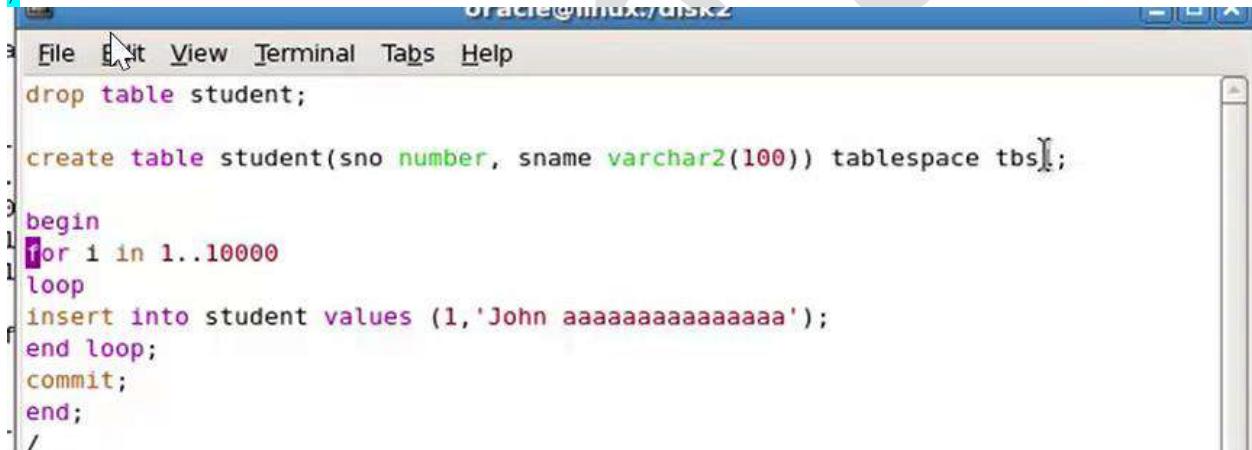
Create a table space script

```
Create tablespace tbs1 datafile '/disk1/dev/data/data01.dbf'
```

Then run the script below created.

#### Test1.sql:

```
drop table student;
create table student(sno number, sname varchar2(100)) tablespace tbs1;
begin
for i in 1..10000
loop
insert into student values(1,'Jhonaaaaaaaaaaaaaa');
end loop;
commit;
end;
/
```



The screenshot shows a terminal window titled "oracle@linux:/disk2". The menu bar includes File, Edit, View, Terminal, Tabs, Help. The main area contains the following SQL code:

```
File Edit View Terminal Tabs Help
drop table student;

create table student(sno number, sname varchar2(100)) tablespace tbs1;

begin
for i in 1..10000
loop
insert into student values (1,'Johnaaaaaaaaaaaaaa');
end loop;
commit;
end;
/
```

To check how much space occupied form the script on tablespace

```
SQL> select tablespace_name, bytes from dba_free_space where tablespace_name='tbs1';
```

```

← Movies & TV
File Applications Places System
MYNOTES (~) - gedit
File Edit View Search Tools Documents Help
File Edit View Terminal Tabs Help
New Open File Edit View Terminal Tabs Help
MYNOTES SYSTEM /disk2/prod1/data/sysaux01.dbf 490
SET LINE TBS1 /disk2/prod1/data/system01.dbf 748
COLUMN T /disk2/prod1/data/data01.dbf 1
COLUMN F
COLUMN P
SQL> @/disk2/test1.sql
drop table student
*
ERROR at line 1:
select t
ORA-00942: table or view does not exist

Table created.

PL/SQL procedure successfully completed.

SQL> select tablespace_name, bytes from dba_free_space where tablespace_name = 'TBS1';
TABLESPACE_NAME          BYTES
TBS1                      569824

SQL> select tablespace_name, file_name, bytes from dba_data_files;
TABLESPACE_NAME          FILE_NAME          BYTES
USERS                     /disk2/prod1/data/users01.dbf      5242880
UNDOTBS1                  /disk2/prod1/data/undotbs01.dbf  31457280
SYSAUX                    /disk2/prod1/data/sysaux01.dbf  513802240
SYSTEM                     /disk2/prod1/data/system01.dbf  775946240
TBS1                      /disk2/prod1/data/data01.dbf   1048576
SQL>

```

#### For deleting tablespace from database

SQL> drop tablespace tbs1 including contents and datafiles;

```

← Movies & TV
File Applications Places System
MYNOTES (~) - gedit
File Edit View Search Tools Documents Help
File Edit View Terminal Tabs Help
MYNOTES PL/SQL procedure successfully completed.
SET LINE T
COLUMN T
SQL> alter tablespace tbs1 add datafile '/disk2/prod1/data/data02.dbf' size 20m;
COLUMN F
Tablespace altered.
select t
SQL> select tablespace_name, file_name, bytes from dba_data_files;
select t
TABLESPACE_NAME          FILE_NAME          BYTES
USERS                     /disk2/prod1/data/users01.dbf      5
UNDOTBS1                  /disk2/prod1/data/undotbs01.dbf  30
SYSAUX                    /disk2/prod1/data/sysaux01.dbf  490
SYSTEM                     /disk2/prod1/data/system01.dbf  748
TBS1                      /disk2/prod1/data/data01.dbf   10
TBS1                      /disk2/prod1/data/data02.dbf   28
6 rows selected.
SQL> drop tablespace tbs1 including contents and datafiles;
Tablespace dropped.
SQL> select tablespace_name, file_name, bytes/1024/1024 from dba_data_files;
TABLESPACE_NAME          FILE_NAME          BYTES/1024/1024
USERS                     /disk2/prod1/data/users01.dbf      5
UNDOTBS1                  /disk2/prod1/data/undotbs01.dbf  30
SYSAUX                    /disk2/prod1/data/sysaux01.dbf  498
SYSTEM                     /disk2/prod1/data/system01.dbf  748
SQL>

```

#### To add more datafiles for an existing tablespace ex:tbs1

SQL> alter tablespace tbs1 add datafile '/u01/app/oracle/oradata/cdb1/data02.dbf' size 10m autoextend on next 512k maxsize 100m;

#### To Rename the existing Tablespace name with new name

SQL> alter tablespace tbs1 rename to tbs2;

SQL> alter tablespace tbs1 rename to tbs2;

Tablespace altered.

#### To drop the tablespace for a particular data file

SQL> alter tablespace tbs2 drop datafile '/disk2/prod1/data/data01.dbf'

```

MYNOTE:SQL> alter tablespace tbs2 add datafile '/disk2/prod1/data/data92.dbf' size 20M;
Tablespace altered.

select t
SQL> select tablespace_name, file_name, bytes/1024/1024 from dba_data_files;

TABLESPACE_NAME          FILE_NAME           BYTES/1024/1024
-----                  -----              -----
USERS                   /disk2/prod1/data/users01.dbf      5
UNDOTBS1                /disk2/prod1/data/undotbs01.dbf    50
SYSAUX                 /disk2/prod1/data/sysaux01.dbf   496
SYSTEM                  /disk2/prod1/data/system01.dbf   140
TBS2                    /disk2/prod1/data/data01.dbf     16
TBS2                   /disk2/prod1/data/data02.dbf     20

6 rows selected.

SQL> alter tablespace tbs2 drop datafile '/disk2/prod1/data/data02.dbf';
Tablespace altered.

SQL> select tablespace_name, file_name, bytes/1024/1024 from dba_data_files;

TABLESPACE_NAME          FILE_NAME           BYTES/1024/1024
-----                  -----              -----
USERS                   /disk2/prod1/data/users01.dbf      5
UNDOTBS1                /disk2/prod1/data/undotbs01.dbf    50
SYSAUX                 /disk2/prod1/data/sysaux01.dbf   496
SYSTEM                  /disk2/prod1/data/system01.dbf   140
TBS2                    /disk2/prod1/data/data01.dbf     16

SQL> 

```

To rename the datafile , you have to do first offline the tablespace

```

SQL> alter tablespace tbs2 offline;
Root# mv '/disk2/prod1/data/data01.dbf' to '/disk2/prod1/data/data99.dbf'
SQL> alter database rename file '/disk2/prod1/data/data01.dbf' to '/disk2/prod1/data/data99.dbf'
Now bring the table space to online
SQL> alter tablespace tbs2 online;

```

```

MYNOTE:SQL> 0 rows selected.

SET LINE
COLUMN T
COLUMN F
COLUMN P
Tablespace altered.

select t
SQL> select tablespace_name, file_name, bytes/1024/1024 from dba_data_files;

TABLESPACE_NAME          FILE_NAME           BYTES/1024/1024
-----                  -----              -----
USERS                   /disk2/prod1/data/users01.dbf      5
UNDOTBS1                /disk2/prod1/data/undotbs01.dbf    50
SYSAUX                 /disk2/prod1/data/sysaux01.dbf   496
SYSTEM                  /disk2/prod1/data/system01.dbf   140
TBS2                    /disk2/prod1/data/data01.dbf     16

SQL> alter tablespace tbs2 offline;
Tablespace altered.

SQL> alter database rename file '/disk2/prod1/data/data01.dbf' to '/disk2/prod1/data/data99.dbf';
alter database rename file '/disk2/prod1/data/data01.dbf' to '/disk2/prod1/data/data99.dbf'
*
ERROR at line 1:
ORA-01511: error in renaming log/data files
ORA-01110: error renaming data file 5: new file '/disk2/
ORA-01110: data file 5: '/disk2/prod1/data/data01.dbf'
ORA-27027: unable to obtain file status
Linux-x86_64 Error: 2: No such file or directory
Additional information: 3

SQL> 

[Terminal session continues with ls -l command showing renamed file]

```

Tablespace with different block sizes: [Used for full table scans]

```
SQL> show parameter db_block_size;
```

```

MYNOTE:SQL> show parameter db_block_size;

NAME          TYPE        VALUE
----          ----        -----
db_block_size integer     8192

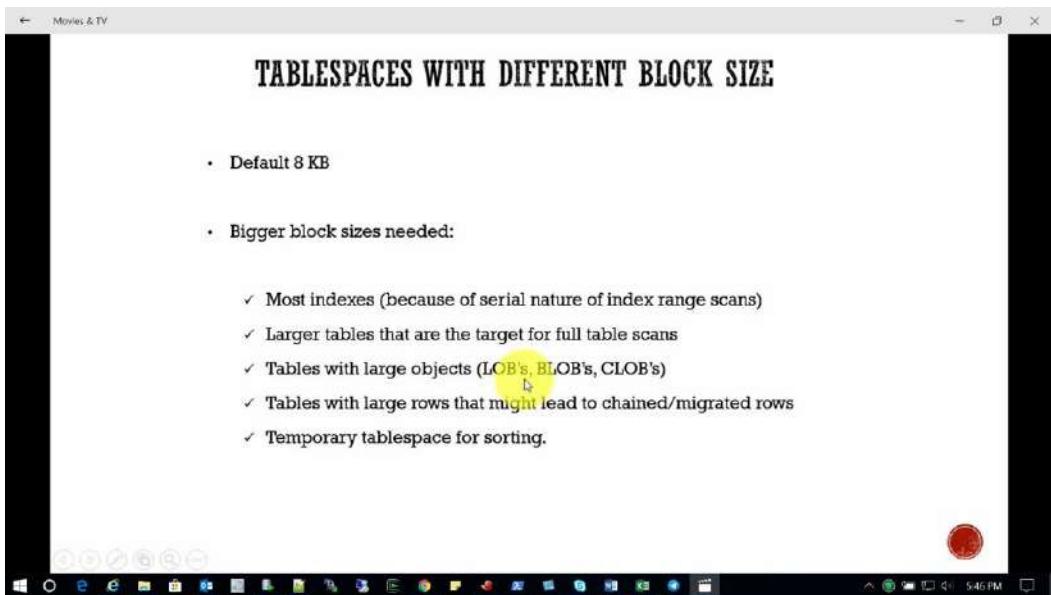
SQL> create tablespace tbs3 datafile '/disk2/prod1/data/data03.dbf' size 10m blocksize 16k;
Tablespace created.

SQL> alter system set db_16k_cache_size=0db scope=both;
System altered.

SQL> create tablespace tbs3 datafile '/disk2/prod1/data/data03.dbf' size 10m blocksize 16k;
Tablespace created.

SQL> 

```



#### Temporary Tablespace [DBA\_TEMP\_FILES]

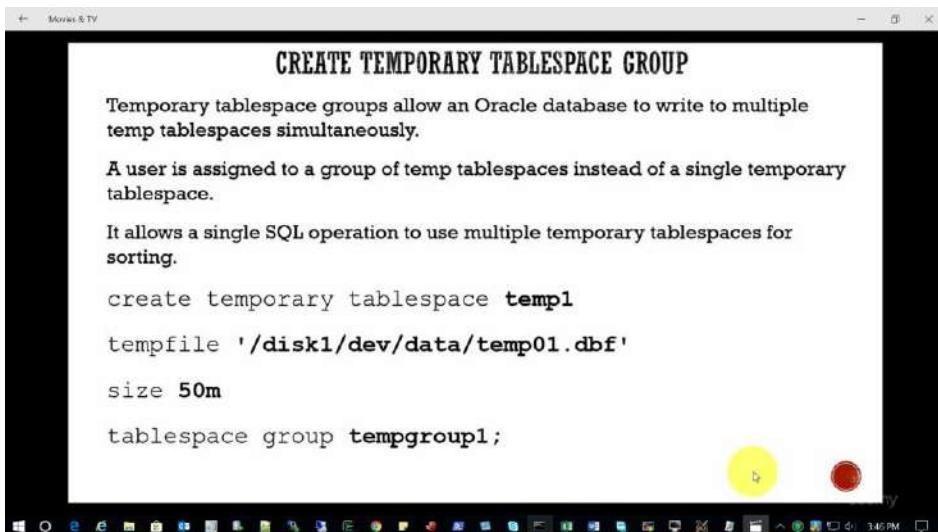
```
SQL>DESC DBA_TEMP_FILES;
SQL>select TABLESPACE_NAME, file_name, blocks from dba_temp_files;
SQL> select tablespace_name, file_name, bytes/1024/1024 from dba_temp_files;
TABLESPACE_NAME          FILE_NAME          BYTES/1024/1024
TEMP                     /u01/app/oracle/oradata/cdb1/temp01.dbf      33
SQL> create temporary tablespace tmp1 tempfile '/u01/app/oracle/oradata/cdb1/gtmp01.dbf' size 10m;
SQL> create temporary tablespace tmp1 tempfile '/u01/app/oracle/oradata/cdb1/gtmp01.dbf' size 10m;
Tablespace created.
SQL> select tablespace_name, file_name, bytes/1024/1024 from dba_temp_files;
TABLESPACE_NAME          FILE_NAME          BYTES/1024/1024
TMP1                    /u01/app/oracle/oradata/cdb1/gtmp01.dbf      10
TEMP                     /u01/app/oracle/oradata/cdb1/temp01.dbf      33
```

#### To check the database default properties

```
SQL> select * from database_properties where property_name like '%TABLESPACE%';
SQL> select * from database_properties where property_name like '%TABLESPACE%';
PROPERTY_NAME
PROPERTY_VALUE
DESCRIPTION
DEFAULT_PERMANENT_TABLESPACE
USERS
Name of default permanent tablespace
DEFAULT_TEMP_TABLESPACE
TEMP
Name of default temporary tablespace
PROPERTY_NAME
PROPERTY_VALUE
DESCRIPTION
```

#### To create a temporary tablespace group[DBA\_TABLESPACE\_GROUPS]

```
SQL> select * from dba_tablespace_groups;
if no records found, you should have at least one temp tablespace to move under tempgroup1.
```



Now create temporary tablespace under group.

```

SQL> create temporary tablespace temp1 tempfile '/u01/app/oracle/oradata/cdb1/gtmp02.dbf' size 10m tablespace group te_group;

```

```

SQL> create temporary tablespace temp2 tempfile '/u01/app/oracle/oradata/cdb1/gtmp02.dbf' size 10m tablespace group te_group;
Tablespace created.

```

To crosscheck whether it's created or not

```

SQL> select * from dba_tablespace_groups;

```

| GROUP_NAME | TABLESPACE_NAME |
|------------|-----------------|
| TE_GROUP   | TEMP2           |

How to add this group for the existing temp tablespace.

```

SQL> select tablespace_name, file_name from dba_temp_files;
SQL> select * from dba_tablespace_groups;
SQL> alter tablespace temp1 tablespace group te_group;

```



```

SQL> select * from dba_tablespace_groups;

```

| GROUP_NAME | TABLESPACE_NAME |
|------------|-----------------|
| TE_GROUP   | TEMP2           |

```

SQL> select tablespace_name, file_name from dba_temp_files;

```

| TABLESPACE_NAME | FILE_NAME                               |
|-----------------|-----------------------------------------|
| TMP1            | /u01/app/oracle/oradata/cdb1/gtmp01.dbf |
| TEMP2           | /u01/app/oracle/oradata/cdb1/gtmp02.dbf |
| TEMP            | /u01/app/oracle/oradata/cdb1/temp01.dbf |

```

SQL> select * from dba_tablespace_group;
select * from dba_tablespace_group

```

ERROR at line 1:  
ORA-00942: table or view does not exist

```

SQL> select * from dba_tablespace_groups;

```

| GROUP_NAME | TABLESPACE_NAME |
|------------|-----------------|
| TE_GROUP   | TEMP2           |

```

SQL> alter tablespace temp1 tablespace group te_group;
alter tablespace temp1 tablespace group te_group

```

ERROR at line 1:  
ORA-00959: tablespace 'TEMP1' does not exist

```

SQL> create temporary tablespace temp1 tempfile '/u01/app/oracle/oradata/cdb1/gtmp02.dbf' size 10m tablespace group te_group;
create temporary tablespace temp1 tempfile '/u01/app/oracle/oradata/cdb1/gtmp02.dbf' size 10m tablespace group te_group

```

ERROR at line 1:  
ORA-01537: cannot add file '/u01/app/oracle/oradata/cdb1/gtmp02.dbf' - file already part of database

```

SQL> alter tablespace TEMP1 tablespace group te_group;
Tablespace altered.

```

#### Locally VS Dictionary managed tablespaces

**Locally Managed Tablespaces:** undo/redo not generated. It's a good advantage and is the reason oracle moved from dictionary managed tablespaces to Locally managed tablespaces. Oracle will take care of all auto allocations for your tablespaces.

```

SQL> select tablespace_name, extent_management from dba tablespaces;

```

```

SQL> select tablespace_name, extent_management from dba_tablespaces;

```

| TABLESPACE_NAME | EXTENT_MAN |
|-----------------|------------|
| SYSTEM          | LOCAL      |
| SYSAUX          | LOCAL      |
| UNDOTBS1        | LOCAL      |
| TEMP            | LOCAL      |
| USERS           | LOCAL      |
| TBS2            | LOCAL      |
| BIGTBS1         | LOCAL      |
| TBS3            | LOCAL      |
| TEMP1           | LOCAL      |
| TEMP2           | LOCAL      |

10 rows selected.

#### Dictionary Managed Tablespaces:

```

SQL> create tablespace tbs3 datafile '/u01/app/oracle/oradata/cdb1/data03.dbf' size 10m extent management
dictionary default storage (initial 50k next 50k minextents 2 maxextents 50 pctincrease 0);

```

```

SQL> select tablespace_name, extent_management from dba_tablespaces;

```

| TABLESPACE_NAME | EXTENT_MAN |
|-----------------|------------|
| SYSTEM          | LOCAL      |
| SYSAUX          | LOCAL      |
| UNDOTBS1        | LOCAL      |
| TEMP            | LOCAL      |
| USERS           | LOCAL      |
| TBS2            | LOCAL      |
| BIGTBS1         | LOCAL      |
| TBS3            | LOCAL      |
| TEMP1           | LOCAL      |
| TEMP2           | LOCAL      |

10 rows selected.

```

SQL> create tablespace tbs3 datafile '/disk2/prod1/data/data05.dbf' size 10m extent management
2 dictionary default storage (initial 50k next 50k minextents 2 maxextents 50 pctincrease 0);
create tablespace tbs3 datafile '/disk2/prod1/data/data05.dbf' size 10m extent management dictionary
*
ERROR at line 1:
ORA-12913: Cannot create dictionary managed tablespace

```

## LOCALLY VS DICTIONARY MANAGED TABLESPACE

- Dictionary managed tablespaces are deprecated with Oracle 11g
- Default extent management is Locally managed
- Locally managed tablespaces track all extent information in the tablespace itself by using bitmaps ( Each bit corresponds to a block or group of blocks(Extents). The bitmap value of ON or OFF indicates whether an EXTENT is allocated or free for reuse)

**Why?**

- Fast, concurrent space operations.
- Enhanced performance due to reduced contention on data dictionary tables. User reliance on the data dictionary is reduced, because the necessary information is stored in file headers and bitmap blocks. This also reduces contention on the SYSTEM tablespace.
- Readable standby databases are allowed, because locally managed temporary tablespaces do not generate any undo or redo (UNDO/REDO is not generated as they do not update data dictionary tables)
- Space allocation is simplified, because when the AUTOALLOCATE clause is specified, the database automatically selects the appropriate extent size.
- Coalescing free extents is unnecessary for locally managed tablespaces

If the data is not showing properly in sql cmd.

```

Set linesize 150
Column tablespace_name format A30
Column file_name format A50
select tablespace_name, file_name, bytes from dba_data_files;

```

If you want to see in MB's

```

select tablespace_name, file_name, bytes/1024/1024 from dba_data_files;

```

Movies & TV

Applications Places System

MYNOTES (-) - gedit

File Edit View Journal Tags Help

New Open 775946249

MYNO

```

SET L1
SQL> SET LINESIZE 150
COLUMN TABLESPACE_NAME FORMAT A30
COLUMN FILE_NAME FORMAT A50
COLUMN
SQL> select tablespace_name, file_name, bytes from dba_data_files;

select TABLESPACE_NAME      FILE_NAME          BYTES
-----  -----  -----
select USERS                /disk2/prod1/data/users01.dbf      5242880
UNDOTBS1                 /disk2/prod1/data/undotbs01.dbf    31457280
SYSAUX                  /disk2/prod1/data/sysaux01.dbf    51386240
SYSTEM                   /disk2/prod1/data/system01.dbf   775946240

SQL> select tablespace_name, file_name, bytes from dba_data_files;

TABLESPACE_NAME      FILE_NAME          BYTES
-----  -----  -----
USERS                /disk2/prod1/data/users01.dbf      5242880
UNDOTBS1                 /disk2/prod1/data/undotbs01.dbf    31457280
SYSAUX                  /disk2/prod1/data/sysaux01.dbf    51386240
SYSTEM                   /disk2/prod1/data/system01.dbf   775946240

SQL> select tablespace_name, file_name, bytes/1024/1024 from dba_data_files;

TABLESPACE_NAME      FILE_NAME          BYTES/1024/1024
-----  -----  -----
USERS                /disk2/prod1/data/users01.dbf      5
UNDOTBS1                 /disk2/prod1/data/undotbs01.dbf    39
SYSAUX                  /disk2/prod1/data/sysaux01.dbf    499
SYSTEM                   /disk2/prod1/data/system01.dbf   749

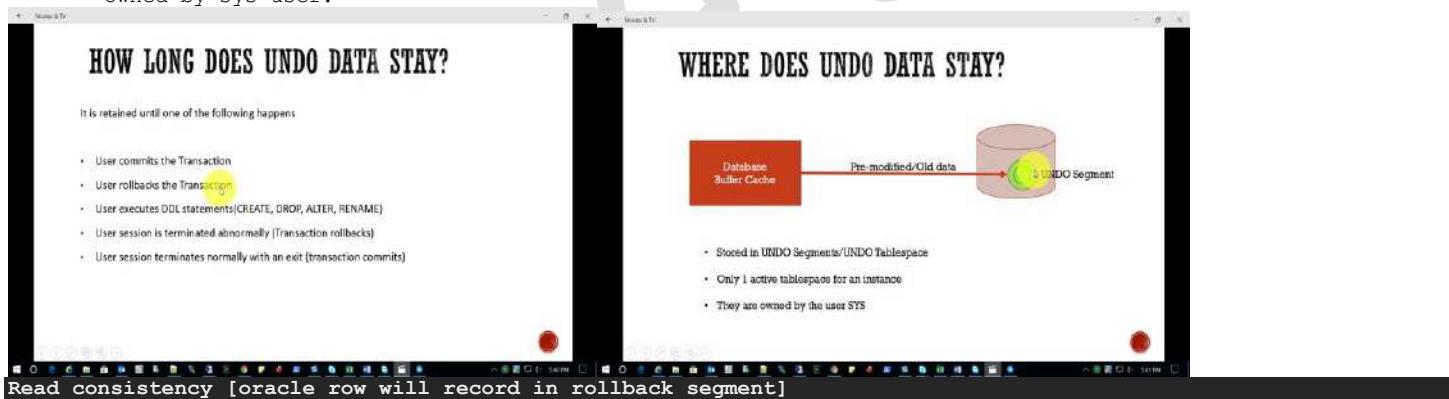
SQL> create tablespace tbs1 datafile '/disk2/';
```

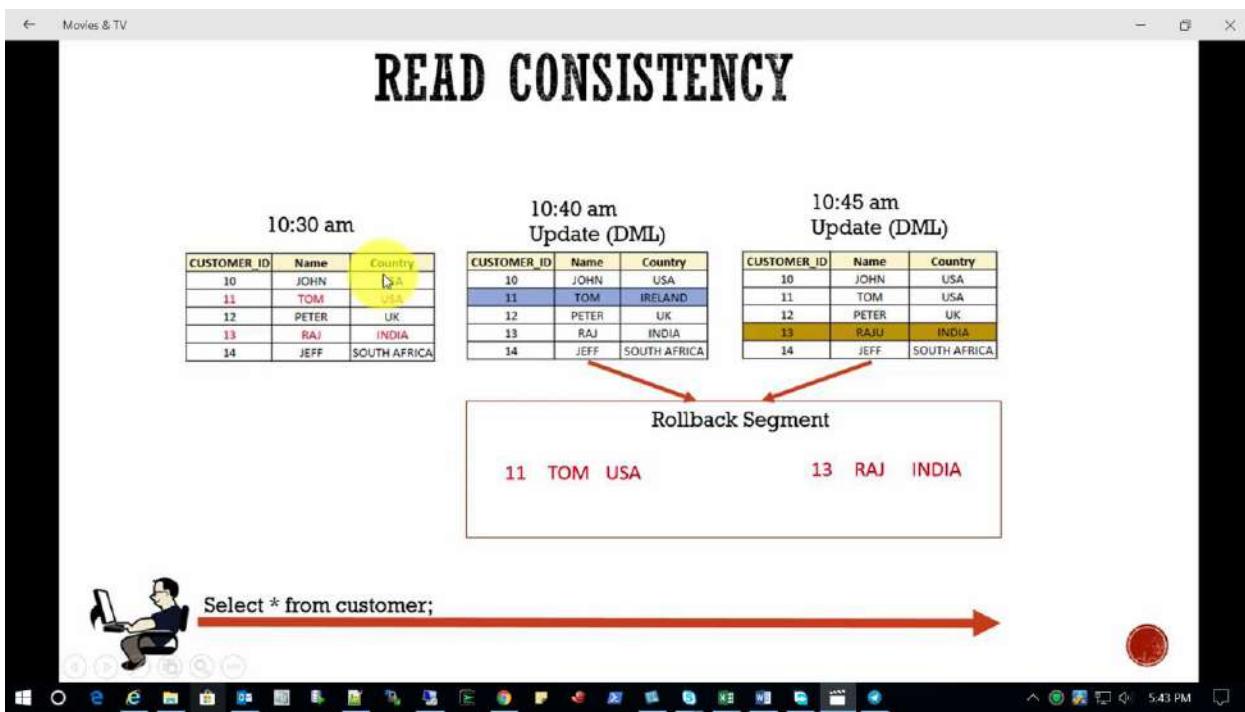
Ln 5, Col 1 INS

oracle@linux:~ MYNOTES (-) - gedit

## UNDO Management

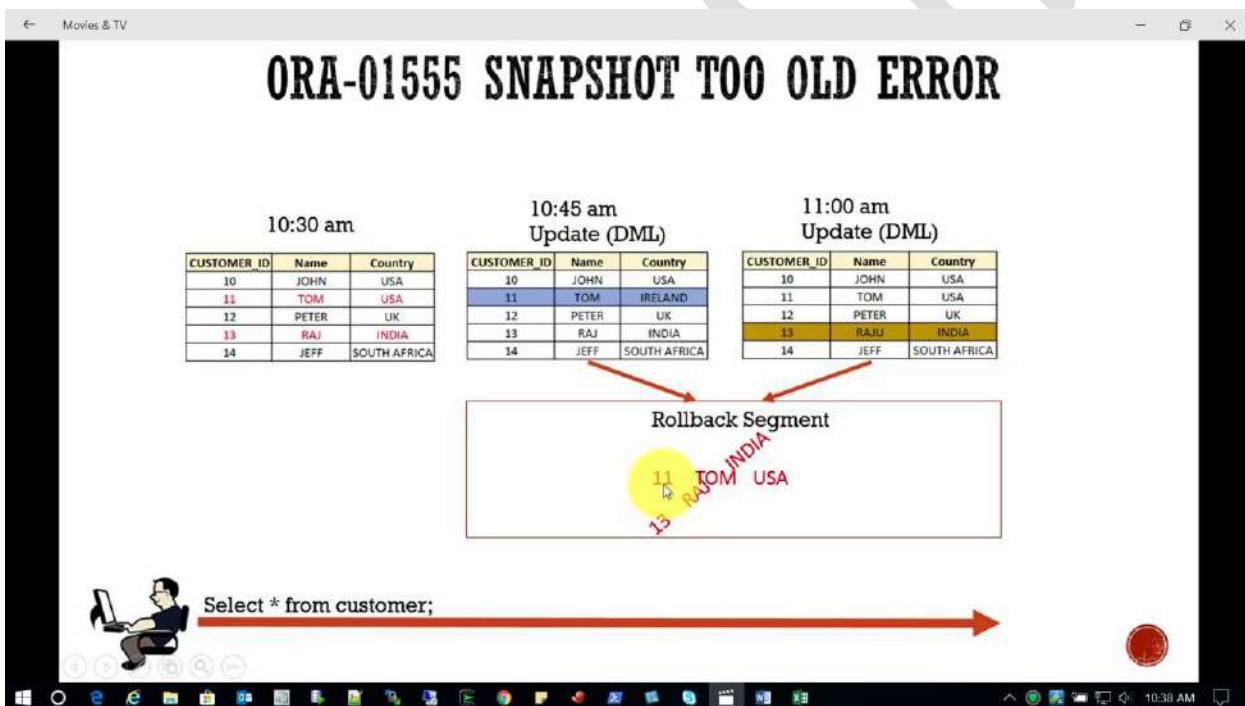
- If the row is deleted , to get back the deleted record undo will help.
- This data will store in entire row in undo segments.
- Undo data will be used for flashback operations.
- Undo data is used for recover the failed transactions.
- To know how the data transaction done yesterday
- How long does undo data stay? The answer is NO.
- Where does the undo data stay? The answer is, stored in UNDO segments and UNDO tablespace.
- A oracle instance will have a only one UNDO tablespace.
- Owned by sys user.





ORA-01555 SNAPSHOT TOO OLD ERROR

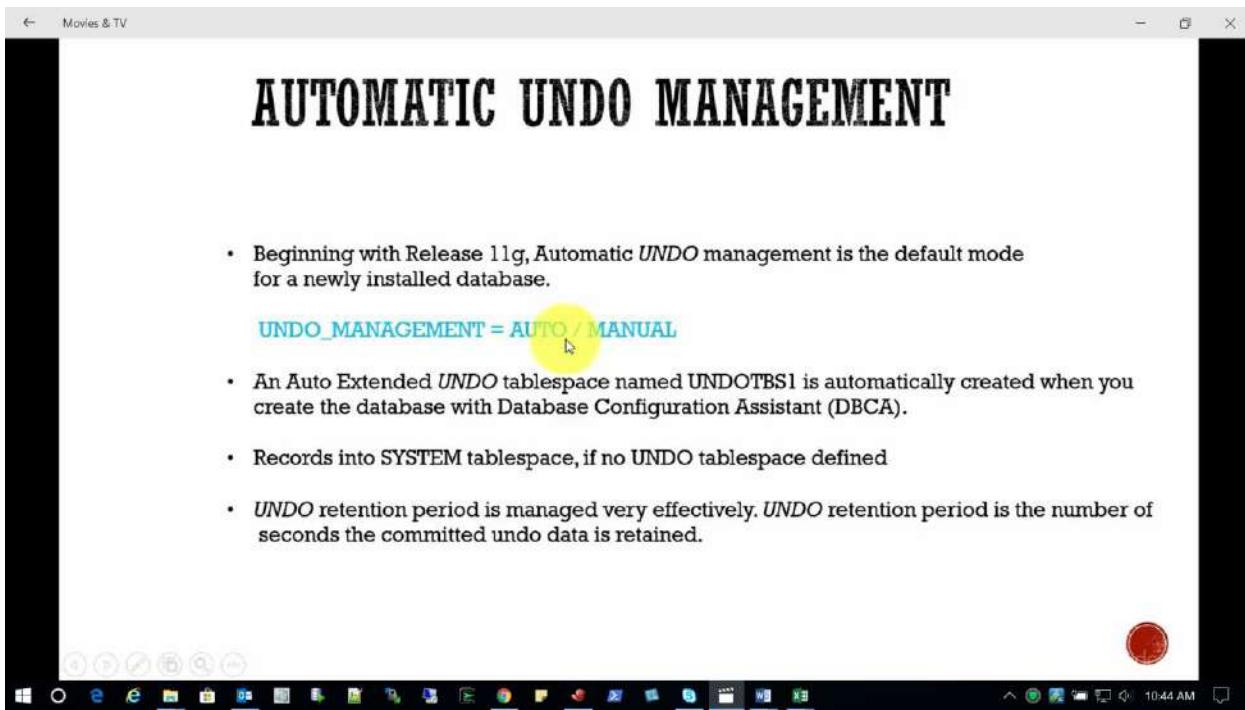
- If the record not found in UNDO data(Rollback segment), this error will show.



Automatic UNDO Management [Preferred than manual] undotbs1

Best approach to create the separate undo tablespace instead of using system undo tablespace.

```
SQL> show parameter undo;
NAME          TYPE    VALUE
undo_management      string  AUTO
undo_retention       integer  900
undo_tablespace      string  UNDOTBS1
SQL> show parameter undo;
```



```
# ! Practiced on UNDO Tablespace
SQL> select tablespace, contents, status from dba_tablespaces;
SQL> select tablespace_name, file_name, bytes/1024/1024 from dba_data_files;
SQL> show parameters UNDO;
SQL> alter tablespace undotbs1 add datafile '/u01/.../undodata01.dbf' size 10m autoextend on next 1m maxsize unlimited.
SQL> select tablespace_name, file_name, bytes/1024/1024 from dba_data_files;
SQL> alter tablespace undotbs2 add datafile '/u01/.../undodata01.dbf' size 5m reuse autoextend on;
SQL> select tablespace_name, file_name, bytes/1024/1024 from dba_data_files order by tablespace_name;
SQL> show parameters UNDO;
SQL> alter system set undo_tablespace=undotbs2;
SQL> show parameters UNDO;
SQL> select segment_name, owner, tablespace_name, status from dba_rollback_segs;
SQL> alter system set undo_tablespace=undotbs1;
SQL> select segment_name, owner, tablespace_name, status from dba_rollback_segs;
```

#### Configuring Undo Retention Period [UNDO\_RETENTION=1800]

Normally the undo\_retention parameter value set for 1800 seconds(30 sec) for committed undo data is retained in this period.

# CONFIGURING UNDO RETENTION PERIOD

UNDO retention period is the number of seconds the committed undo data is retained.

Set initialization parameter file.

UNDO\_RETENTION = 1800

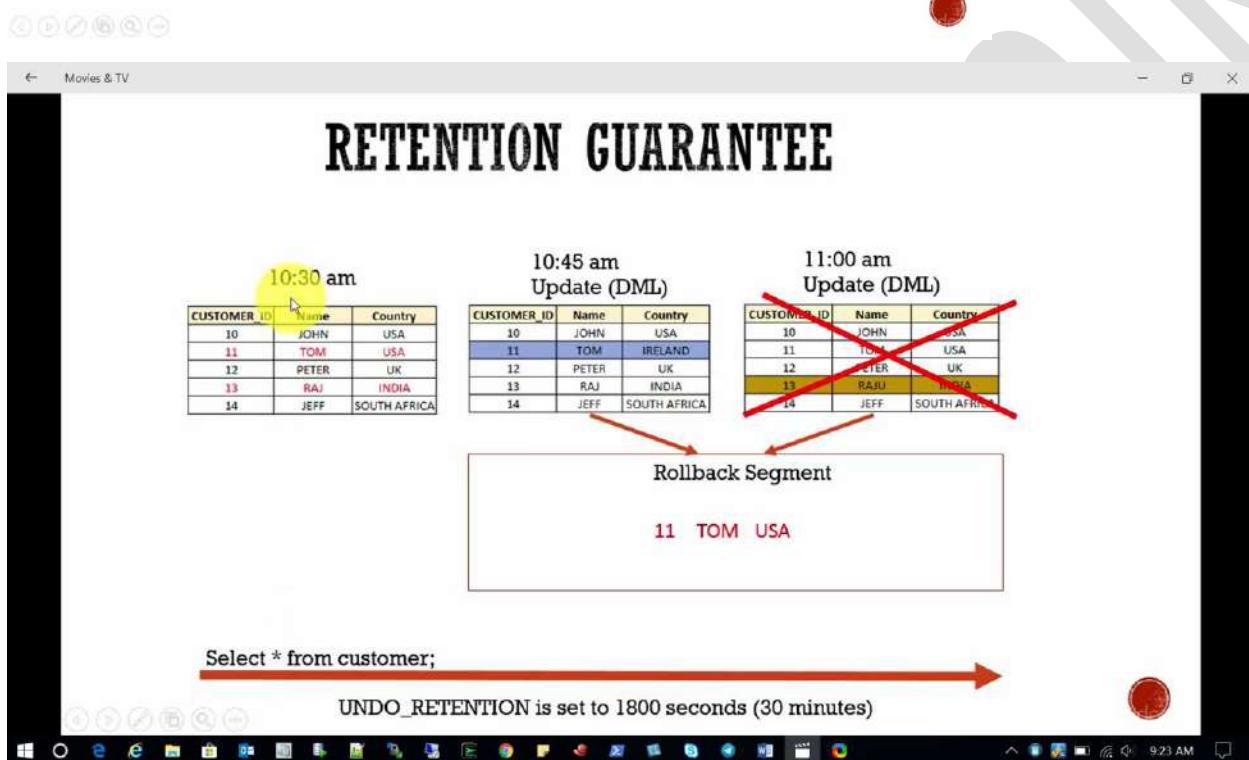
**ALTER**

ALTER SYSTEM SET UNDO\_RETENTION = 2400;

After a transaction is committed, undo data is no longer needed for rollback or transaction recovery purposes.

However, for consistent read purposes, long-running queries may require this old undo information for producing older images of data blocks.

Furthermore, the success of several Oracle Flashback features can also depend upda the availability of older undo information. For these reasons, it is desirable to retain the old undo information for as long as possible



```
SQL> show parameter undo;
SQL> alter system set undo_retention=2400;
SQL> show parameter undo;
SQL> select tablespace_name, retention from dba_tablespaces;
SQL> alter system undotbs1 RETENTION guarantee;
SQL> alter system undotbs1 RETENTION noguarantee;
```

## REDO Management

Their only purpose to enable recovery.

When the commits done their transaction

When the redo log files written multiples log files to database for every 3 seconds.

Only one redo log file, the database canot function when the redo log file not available.

So we have to maintain redo log groups.

Redo log file should be stored with in a same group in different disks

## Redo Logfile Types :

1. Current Log Files - Log writer
2. Active Log Files - For LGwr and archive log mode should be enabled here.
3. Inactive Log Files - which are no longer for instance recovery so overwrite this inactive log files.

**REDO LOG FILES**

Redo Log Files enable the Oracle Server or DBA to redo transactions if a database failure occurs.

Their only purpose is to enable recovery.

| CUSTOMER ID | Name | Country |
|-------------|------|---------|
| 10          | JOHN | USA     |
| 11          | TOM  | USA     |

Diagram illustrating redo log entries:

- INSERT:** OP 11.1 (10) Insert New Piece - DRP Stmt 4: C1: JOHN C2: USA
- UPDATE:** OP 11.2 (11) Update Row Piece - DRP Stmt 4: C1: JOHN C2: USA
- DELETE:** OP 11.3 (11) Delete Row Piece - DRP Stmt 4: C1: JOHN C2: USA

**WHEN IS REDO WRITTEN?**

Writes the redo log buffer to a redo log file on disk

**When will it write?**

- When a user process commits a transaction
- When the redo log buffer is one-third full
- Before a DBWn process writes modified buffers to disk
- Every 3 seconds

**REDO LOG FILE MANAGEMENT**

Groups for redo logs:

**Current Log file**  
The Redo Log file to which LGWR is actively writing is called the current log file.

**Active Log file**  
Log files required for instance recovery are categorized as active log files.

Active log files cannot be overwritten by LGWR until ARCn has archived the data when archiving is enabled.

**Inactive Log file**  
Log files no longer needed for instance recovery are categorized as inactive log files.

Redo log file should be stored with in a same group in different disks

#### # ! Practiced on REDO Tablespace

```
SQL> select * from v$logfile; → to get the list of redo logfiles
SQL> select * from v$log; → to get the sizes for the log groups.
SQL> select group#, bytes/1024/1024, status from v$log;
atleast we should have 2 log files under the redo groups.
```

```
SQL> alter database add logfile member '/disk2/prod1/log/redo01a.log' to group 1;
SQL> alter database add logfile member '/disk2/prod1/log/redo02a.log' to group 2;
SQL> alter database add logfile member '/disk2/prod1/log/redo03c.log' to group 3;
Now the all logs created will be shown as INVALIDED, make them VALID.
SQL> alter system switch logfile;
SQL> alter system switch logfile;
SQL> alter system switch logfile;
```

**Executed as practiced.**

#### Add log file to a groups

```
SQL> alter database add logfile member '/u01/app/oracle/oradata/cdb1/redo01a.log' to group 1;
Database altered.
```

```
SQL> alter database add logfile member '/u01/app/oracle/oradata/cdb1/redo02b.log' to group 2;
Database altered.
```

```
SQL> alter database add logfile member '/u01/app/oracle/oradata/cdb1/redo03c.log' to group 3;
Database altered.
```

```
SQL> select * from v$logfile;
GROUP# STATUS TYPE MEMBER IS_RECOVERY_DEST_FILE CON_ID
----- ----- ----- -----
 3 ONLINE  /u01/app/oracle/oradata/cdb1/redo03.log NO #####
 2 ONLINE  /u01/app/oracle/oradata/cdb1/redo02.log NO #####
 1 ONLINE  /u01/app/oracle/oradata/cdb1/redo01.log NO #####
 1 INVALID ONLINE  /u01/app/oracle/oradata/cdb1/redo01a.log NO #####
 2 INVALID ONLINE  /u01/app/oracle/oradata/cdb1/redo02b.log NO #####
 3 INVALID ONLINE  /u01/app/oracle/oradata/cdb1/redo03c.log NO #####
6 rows selected.
```

**Now make it valid which are add log files under the group.**

```
SQL> alter system switch logfile;
System altered.
```

```
SQL> select * from v$logfile;
GROUP# STATUS TYPE MEMBER IS_RECOVERY_DEST_FILE CON_ID
----- ----- ----- -----
 3 ONLINE  /u01/app/oracle/oradata/cdb1/redo03.log NO #####
 2 ONLINE  /u01/app/oracle/oradata/cdb1/redo02.log NO #####
 1 ONLINE  /u01/app/oracle/oradata/cdb1/redo01.log NO #####
 1 INVALID ONLINE  /u01/app/oracle/oradata/cdb1/redo01a.log NO #####
 2 INVALID ONLINE  /u01/app/oracle/oradata/cdb1/redo02b.log NO #####
 3 ONLINE  /u01/app/oracle/oradata/cdb1/redo03c.log NO #####
6 rows selected.
```

```
SQL> alter system switch logfile;
```

System altered.

```
SQL> alter system switch logfile;
System altered.
```

```
SQL> select * from v$logfile;
GROUP# STATUS TYPE MEMBER IS_RECOVERY_DEST_FILE CON_ID
----- ----- ----- -----
 3 ONLINE  /u01/app/oracle/oradata/cdb1/redo03.log NO #####
 2 ONLINE  /u01/app/oracle/oradata/cdb1/redo02.log NO #####
 1 ONLINE  /u01/app/oracle/oradata/cdb1/redo01.log NO #####
 1 ONLINE  /u01/app/oracle/oradata/cdb1/redo01a.log NO #####
 2 ONLINE  /u01/app/oracle/oradata/cdb1/redo02b.log NO #####
 3 ONLINE  /u01/app/oracle/oradata/cdb1/redo03c.log NO #####
6 rows selected.
```

```
SQL>
```

**To delete a group, please inactive first. Then delete it.**

```
SQL> alter database drop logfile group 3;
```

**To add logfiles to a group, please create as follows..**

```
SQL> alter database add logfile group
3('/u01/app/oracle/oradata/cdb1/redo03c.log','/u01/app/oracle/oradata/cdb1/redo03d.log','/u01/app/oracle/oradata/cdb1/redo03e.log') size 10m;
```

```
SQL> select * from v$log;
GROUP# THREAD# SEQUENCE# BYTES BLOCKSIZE MEMBERS ARC STATUS FIRST_CHANGE# FIRST_TIM NEXT_CHANGE# NEXT_TIME CON_ID
----- ----- ----- ----- ----- ----- ----- ----- -----
 1 1 7 209715200 512 2 NO INACTIVE 1732374 05-NOV-18 1732382 05-NOV-18 #####
 2 1 8 209715200 512 2 NO CURRENT 1732382 05-NOV-18 1.8447E+19 #####
 3 1 6 209715200 512 2 NO INACTIVE 1732394 05-NOV-18 1732374 05-NOV-18 #####
```

```
SQL> alter database drop logfile group 3;
```

Database altered.

```
SQL> alter database drop logfile group 3;
```

Database altered.

```
SQL> select * from v$logfile;
```

| GROUP# | STATUS | TYPE                                     | MEMBER | IS_RECOVERY_DEST_FILE | CON_ID |
|--------|--------|------------------------------------------|--------|-----------------------|--------|
| 2      | ONLINE | /u01/app/oracle/oradata/cdb1/redo02.log  | NO     | #####                 | #####  |
| 1      | ONLINE | /u01/app/oracle/oradata/cdb1/redo01.log  | NO     | #####                 | #####  |
| 1      | ONLINE | /u01/app/oracle/oradata/cdb1/redo01a.log | NO     | #####                 | #####  |
| 2      | ONLINE | /u01/app/oracle/oradata/cdb1/redo02b.log | NO     | #####                 | #####  |

now you can see group 3 droped from list

```
SQL> alter database add logfile group 3 ('/u01/app/oracle/oradata/cdb1/redo03c.log','/u01/app/oracle/oradata/cdb1/redo03d.log','/u01/app/oracle/oradata/cdb1/redo03e.log') size 10m;
alter database add logfile group 3 ('/u01/app/oracle/oradata/cdb1/redo03c.log','/u01/app/oracle/oradata/cdb1/redo03d.log','/u01/app/oracle/oradata/cdb1/redo03e.log') size 10m
*
ERROR at line 1:
ORA-06501: error in adding log file - '/u01/app/oracle/oradata/cdb1/redo03c.log' - file cannot be created
ORA-37030: created file already exists
Additional information: 3
```

```
SQL> select * from v$logfile;
GROUP# STATUS MEMBER IS_RECOVERY_DEST_FILE CON_ID
----- --- -----
2 ONLINE /u01/app/oracle/oradata/cdb1/redo02.log NO #####
1 ONLINE /u01/app/oracle/oradata/cdb1/redo01.log NO #####
1 ONLINE /u01/app/oracle/oradata/cdb1/redo01a.log NO #####
2 ONLINE /u01/app/oracle/oradata/cdb1/redo02b.log NO #####
SQL> rm /u01/app/oracle/oradata/cdb1/redo03c.log
SP2-0734: unknown command beginning "rm /u01/app..." - rest of line ignored.
SQL> alter database add logfile group 3 ('/u01/app/oracle/oradata/cdb1/redo03c.log','/u01/app/oracle/oradata/cdb1/redo03d.log','/u01/app/oracle/oradata/cdb1/redo03e.log') size 10m;
alter database add logfile group 3 ('/u01/app/oracle/oradata/cdb1/redo03c.log','/u01/app/oracle/oradata/cdb1/redo03d.log','/u01/app/oracle/oradata/cdb1/redo03e.log') size 10m
Database altered.
SQL> select * from v$logfile;
GROUP# STATUS MEMBER IS_RECOVERY_DEST_FILE CON_ID
----- --- -----
3 ONLINE /u01/app/oracle/oradata/cdb1/redo03c.log NO #####
2 ONLINE /u01/app/oracle/oradata/cdb1/redo02.log NO #####
1 ONLINE /u01/app/oracle/oradata/cdb1/redo01.log NO #####
1 ONLINE /u01/app/oracle/oradata/cdb1/redo01a.log NO #####
2 ONLINE /u01/app/oracle/oradata/cdb1/redo02b.log NO #####
3 ONLINE /u01/app/oracle/oradata/cdb1/redo03d.log NO #####
3 ONLINE /u01/app/oracle/oradata/cdb1/redo03e.log NO #####
7 rows selected.
```

```
SQL> select * from v$log;
GROUP# THREAD# SEQUENCE# BYTES BLOCKSIZE MEMBERS ARC STATUS FIRST_CHANGE# FIRST_TIM NEXT_CHANGE# NEXT_TIME CON_ID
----- --- -----
1 1 7 209715200 512 2 NO INACTIVE 1732374 05-NOV-18 1732382 05-NOV-18 #####
2 1 8 209715200 512 2 NO CURRENT 1732382 05-NOV-18 1.8447E+19 #####
3 1 0 10485760 512 3 YES UNUSED 0 0 #####
SQL> 
```

```
SP2-0734: unknown command beginning "ALTER" - rest of line ignored
SQL> alter database add logfile group 3('/u01/app/oracle/oradata/cdb1/redo03c.log','/u01/app/oracle/oradata/cdb1/redo03d.log','/u01/app/oracle/oradata/cdb1/redo03e.log') size 10m;
Database altered.
SQL> select * from v$logfile;
GROUP# STATUS MEMBER IS_RECOVERY_DEST_FILE CON_ID
----- --- -----
3 ONLINE /u01/app/oracle/oradata/cdb1/redo03c.log NO #####
2 ONLINE /u01/app/oracle/oradata/cdb1/redo02.log NO #####
1 ONLINE /u01/app/oracle/oradata/cdb1/redo01.log NO #####
1 ONLINE /u01/app/oracle/oradata/cdb1/redo01a.log NO #####
2 ONLINE /u01/app/oracle/oradata/cdb1/redo02b.log NO #####
3 ONLINE /u01/app/oracle/oradata/cdb1/redo03d.log NO #####
3 ONLINE /u01/app/oracle/oradata/cdb1/redo03e.log NO #####
7 rows selected.
```

```
SQL> select * from v$log;
GROUP# THREAD# SEQUENCE# BYTES BLOCKSIZE MEMBERS ARC STATUS FIRST_CHANGE# FIRST_TIM NEXT_CHANGE# NEXT_TIME CON_ID
----- --- -----
1 1 7 209715200 512 2 NO INACTIVE 1732374 05-NOV-18 1732382 05-NOV-18 #####
2 1 8 209715200 512 2 NO CURRENT 1732382 05-NOV-18 1.8447E+19 #####
3 1 0 10485760 512 3 YES UNUSED 0 0 #####
SQL> 
```

#### For clearing a Redo log:

```
SQL> Alter database clear logfile group 3;
SQL> Alter database clear unarchived logfile group 3;
```

#### Redo Log Data Dictionary Views

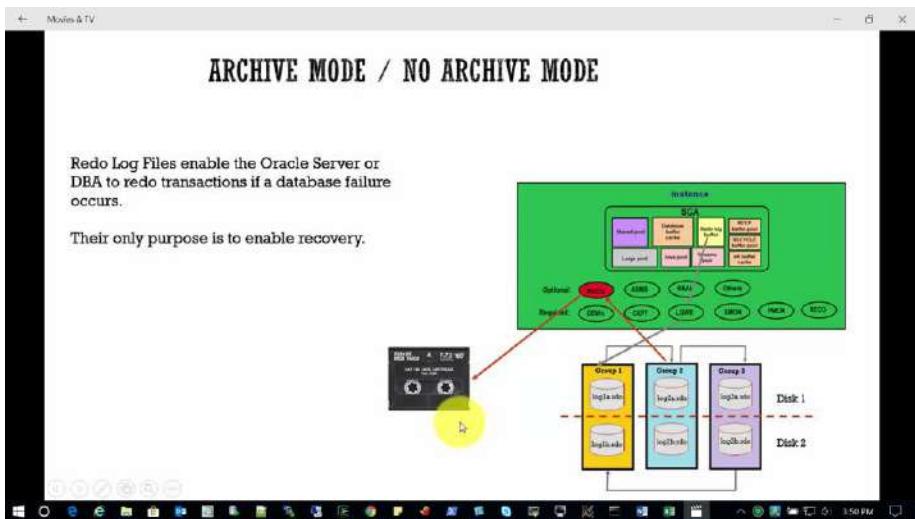
```
V$log
```

```
V$logfile
```

```
V$log history
```

#### Archive Mode / No Archive Mode

Why we need multiple groups. → maintain multiple groups, from the groups, one group will be used and remaining will be archived. Known as archive mode and saving in archive location.  
Archive mode will be used for recover mode.



**Let see if the database is archive mode or not.**

Check whether the database is in archive mode or not, if the database is in "No archive Mode".

For making the database in "Archive Mode"  
You have to shutdown the database.

Startup in mount mode.

SQL>shutdown immediate

SQL> startup mount

SQL> alter database archivelog;

SQL> alter system set log\_archive\_dest\_1='LOCATION=/u01/app/oracle/product/12.2.0/db\_1/arch' scope=both;

SQL> alter database open;

SQL> archive log list

```
SQL> archive log list
Database log mode          No Archive Mode
Automatic archival        Disabled
Archive destination        /u01/app/oracle/product/12.2.0/db_1/dbs/arch
Oldest online log sequence    9
Current log sequence       11
SQL> shutdown immediate
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> startup mount
ORACLE instance started.
```

Total System Global Area 725614592 bytes

Fixed Size 8797008 bytes

Variable Size 541066416 bytes

Database Buffers 167772160 bytes

Redo Buffers 7979008 bytes

Database mounted.

SQL> alter database archivelog;

Database altered.

SQL> alter system set log\_archive\_dest\_1 = 'LOCATION=/u01/app/oracle/product/12.2.0/db\_1/arch' scope=both;

2
SQL> alter system set log\_archive\_dest\_1='LOCATION=/u01/app/oracle/product/12.2.0/db\_1/arch' scope=both;

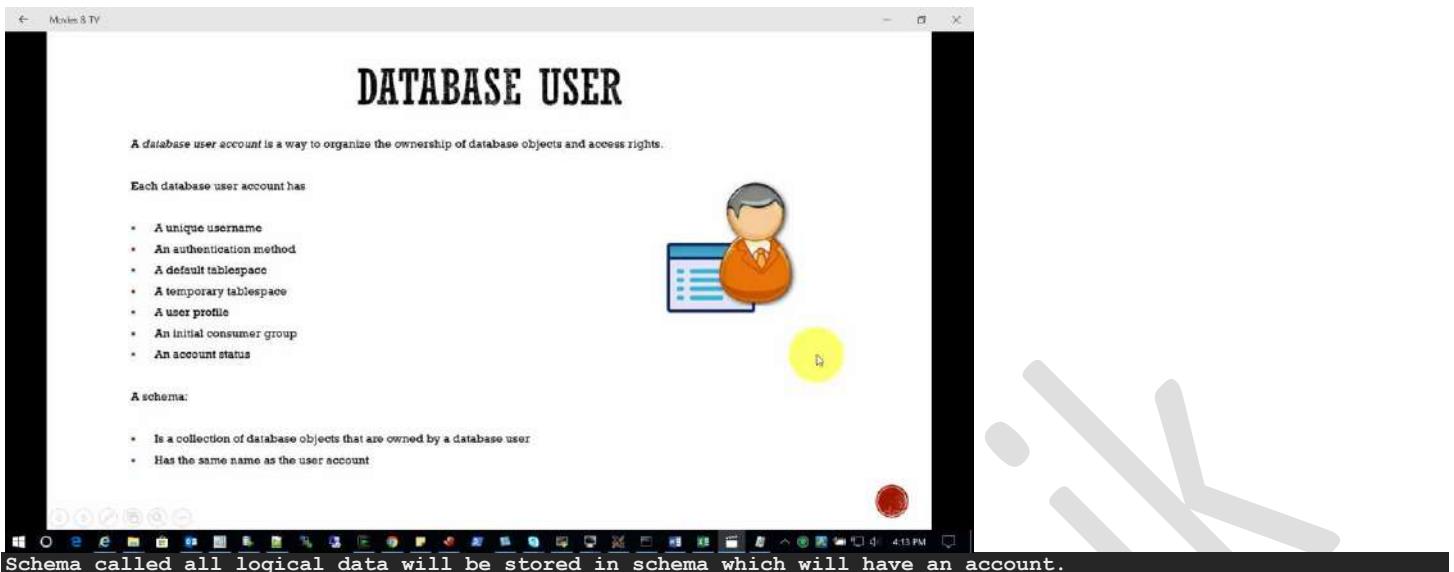
System altered.

```
SQL> archive log list
Database log mode          Archive Mode
Automatic archival        Enabled
Archive destination        /u01/app/oracle/product/12.2.0/db_1/arch
Oldest online log sequence    9
Next log sequence to archive 11
Current log sequence       11
SQL> alter database open;
```

Database altered.

#### USER Management

- User is a unique to identify
- Used for authentication
- A user has default table space, where the data is stored. Default temporary table space where the default temporary data will be stored for the user..
- Account status
- Initial consumer group
- Schema called all logical data will be stored in schema which will have an account.



#### Default ADMINISTRATIVE ACCOUNTS

TYPES OF dba ACCOUNTS:

**Sys** -- which will have all the privileges for the entire database. It's a master database.  
**System** - is much similar to sys, cannot perform database recovery/upgrades. Only tasks can be performed.  
**DBSMNP** - only used to manage the database, create users managed database to see what's going on and how many sessions are running. Only for monitoring the database  
**SYSMAN** - To perform administrative tasks, like creating users..

#### Create users

```
SQL> Create user jhon identified by jhon123;
SQL> Create user jhon identified by jhon123$ password expire;
SQL> Create user jhon Profile default Identified by jhon123$ Default tablespace USERS Temporary tablespace TEMP
Account UNLOCK;
SQL> Grant connect to jhon;
SQL> Grant RESOURCE to jhon;
SQL> grant create session to jhon;
SQL> Alter user jhon Identified by jhon2323@1;
SQL> select default_tablespace, temporary_tablespace from dba_users where username='JHON';
SQL> select tablespace_name, status from DBA_TABLESPACES;
SQL> alter user jhon quota 5m to tbs1; → to store the data we are increasing quota for the user's tablespace,
which can be shown from dba_data_files to know the space.
SQL> select username, tablespace_name, bytes, max_bytes from dba_ts_quotas;
SQL> drop user jhon cascade; [cascade is the option to delete all data for this user jhon]
SQL> alter user jhon account lock;
SQL> alter user jhon account unlock;
SQL> grant select on sys.customer to jhon;
SQL> grant delete on sys.customer to jhon;
SQL> grant select on sys.customer to jhon with grant option; → admin privileges to jhon, to give others this
select privileges to tom user jhon should have grant privileges.
SQL> grant delete on sys.customer to jhon with grant option;
SQL> revoke create table from jhon;
SQL> revoke select, delete on sys.customer from jhon;
```

#### QUOTA

```
SQL> alter user jhon quota 5m to tbs1; → to store the data we are increasing quota for the user's tablespace,
which can be shown from dba_data_files to know the space.
SQL> select username, tablespace_name, bytes, max_bytes from dba_ts_quotas;
SQL> select * from DBA_TS_QUOTAS where username = 'PIN';
```

#### Users privileges

Every user having 2 types of privileges

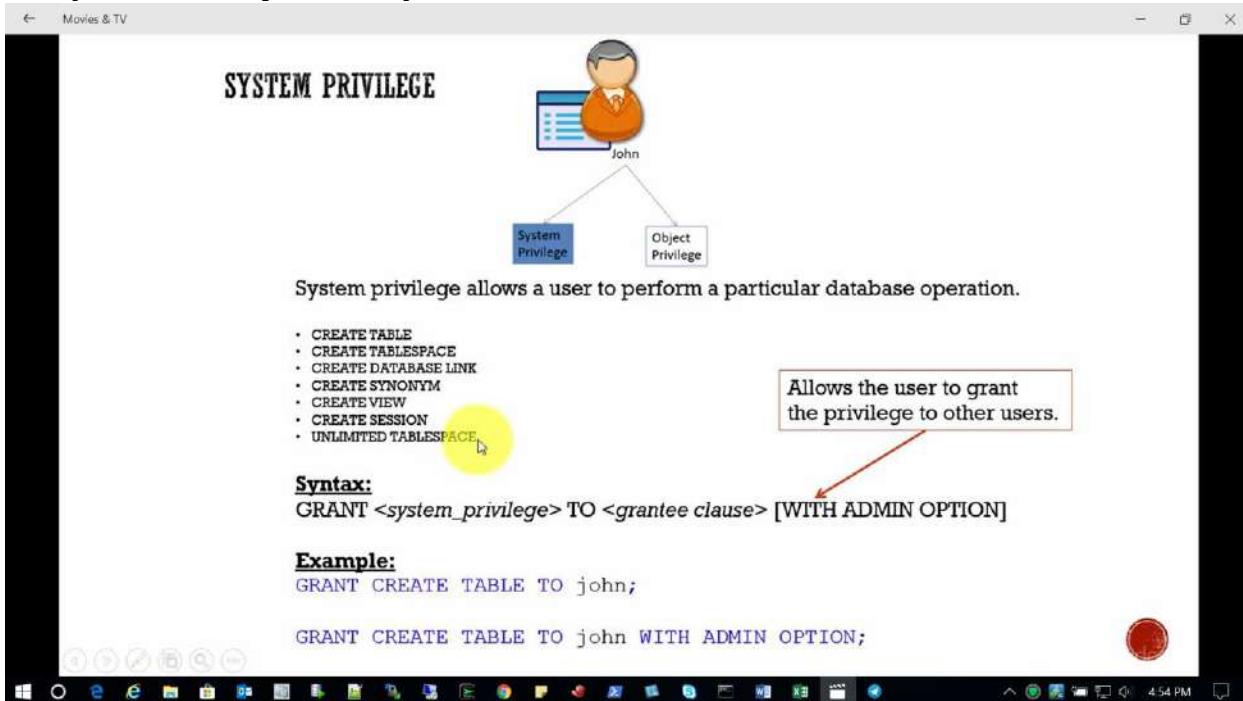
1. System privileges
2. Object privileges

#### System privileges

- RESTRICTED SYSTEM
- SYSDBA
- SYSOPER
- SYSASM
- DROP ANY OBJECT
- CREATE, MANAGE, DROP and ALTER TABLESPACE
- CREATE LIBRARY

- CREATE ANY DIRECTORY
- GRANT ANY OBJECT PRIVILEGE
- ALTER DATABASE
- ALTER SYSTEM

```
SQL> grant create table to jhon;
SQL> grant create table to jhon with admin option;
SQL> grant select any table to jhon;
```



#### Oracle PASSWORD Expiry for user

```
SQL> SELECT EXPIRY_DATE, PASSWORD, PROFILE FROM DBA_USERS WHERE USERNAME = 'INTEGRATE'; --now get the profile name and update the profile name in below query
SQL>ALTER PROFILE DEFAULT LIMIT PASSWORD_REUSE_TIME UNLIMITED;
SQL> commit;
SQL>ALTER PROFILE DEFAULT LIMIT PASSWORD_LIFE_TIME UNLIMITED;
SQL> commit;
SQL>select username, account status, EXPIRY_DATE from dba users where username='INTEGRATE';
```

```
SELECT EXPIRY_DATE, PASSWORD, PROFILE FROM DBA_USERS WHERE USERNAME = 'INTEGRATE';
--now get the profile name and update the profile name in below query
ALTER PROFILE DEFAULT LIMIT PASSWORD_REUSE_TIME UNLIMITED;
commit;
ALTER PROFILE DEFAULT LIMIT PASSWORD_LIFE_TIME UNLIMITED;
commit;

select username, account_status, EXPIRY_DATE from dba_users where username='INTEGRATE';
```

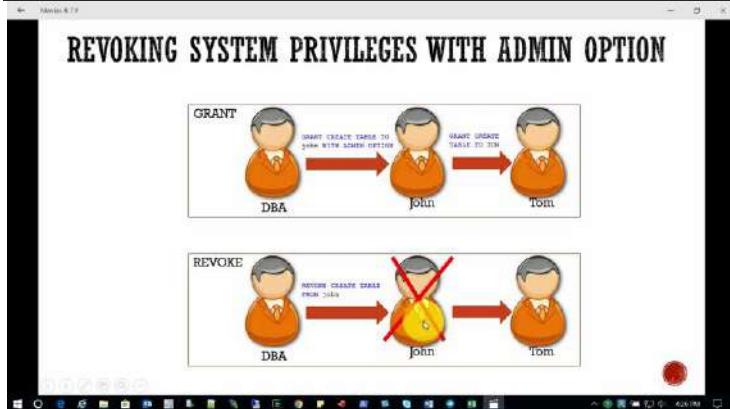
| Script Output |                |             | Query Result                         |
|---------------|----------------|-------------|--------------------------------------|
|               |                |             | All Rows Fetched: 1 in 0.009 seconds |
| USERNAME      | ACCOUNT_STATUS | EXPIRY_DATE |                                      |
| 1 INTEGRATE   | OPEN           | (null)      |                                      |

#### Object privilege

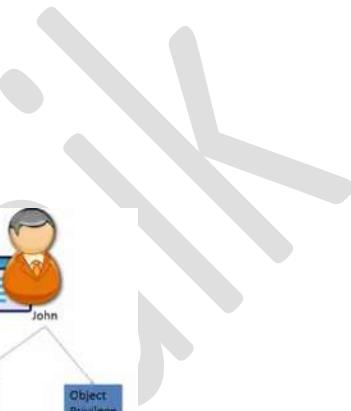
- SELECT
- UPDATE
- DELETE
- ALTER
- EXECUTE

```
SQL>Grant delete on jhon.stup to tom;
SQL>Grant delete on jhon.stud to tom with grant option;
```

## REVOKE SYSTEM PRIVILEGES WITH ADMIN OPTION



```
Sql> revoke create table from jhon;
```



## OBJECT PRIVILEGE LIST

| Privilege  | Object type                                          | Description                                                                                                                                                                                                                                                                                                                               |
|------------|------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SELECT     | Table, sequence, view, materialized view, or synonym | <p>Enables a user to select from a table, sequence, view, materialized view, or synonym.</p> <p>The SELECT privilege enables a user to perform all operations on a sequence.</p> <p>A user can be granted the SELECT privilege on a synonym or a view without being explicitly granted the SELECT privilege on the originating table.</p> |
| INSERT     | Table or synonym                                     | Enables a user to insert into a table or into the table through a synonym.                                                                                                                                                                                                                                                                |
| INDEX      | Table or materialized view                           | Enables a user to create an index on a table or materialized view.                                                                                                                                                                                                                                                                        |
| REFERENCES | Table or materialized view                           | Enables a user to create a foreign key dependency on a table or materialized view.                                                                                                                                                                                                                                                        |
| DELETE     | Table                                                | Enables a user to delete from a table.                                                                                                                                                                                                                                                                                                    |
| UPDATE     | Table                                                | Enables a user to update a table.                                                                                                                                                                                                                                                                                                         |
| EXECUTE    | PL/SQL package, procedure or function                | Enables a user to execute a PL/SQL package, procedure or function directly.                                                                                                                                                                                                                                                               |
| FLUSH      | Cache group                                          | Enables a user to flush a cache group.                                                                                                                                                                                                                                                                                                    |
| LOAD       | Cache group                                          | Enables a user to load a cache group.                                                                                                                                                                                                                                                                                                     |
| UNLOAD     | Cache group                                          | Enables a user to unload a cache group.                                                                                                                                                                                                                                                                                                   |
| REFRESH    | Cache group                                          | Enables a user to refresh a cache group.                                                                                                                                                                                                                                                                                                  |

### # ! Practiced on privileges

To check the privileges

```
SQL> select * from session_privs;
SQL> create user gouse identified by gouse;
ORA-65096: invalid common user or role name.
```

```
SQL> show
SQL> show user
USER is "SYS"
SQL> create user gouse identified by gouse;
create user gouse identified by gouse
*
ERROR at line 1:
ORA-65096: invalid common user or role name
```

Solution and Cause: An attempt was made to create a common user or role with a name that was not valid for common users or roles. In addition to the usual rules for user and role names, common user and role names must start with C## or c## and consist only of ASCII characters.

```
SQL> alter session set "_ORACLE_SCRIPT=true";
Session altered.
SQL> create user gouse identified by gouse;
User created.
SQL> select * from session_privs;
PRIVILEGE
-----
```

```

CREATE SESSION
SQL> create table stud(sno number);
create table stud(sno number)
*
ERROR at line 1:
ORA-01031: insufficient privileges
SQL>
SP2-0042: unknown command "" - rest of line ignored.
Now login to another session and try to grant the create session and create table to gouse from sysdba.

```

```

SQL> grant create session to gouse;
Grant succeeded.

```

```

SQL> grant create table to gouse;
Grant succeeded.

```

```

SQL> create table stud(sno number);
Table created.

```

```

SQL> insert into stud values(1);
insert into stud values(1)
*
```

```

ERROR at line 1:
ORA-00942: table or view does not exist
SQL> insert into stud values(1);
insert into stud values(1)
*
```

```

ERROR at line 1:
ORA-01950: no privileges on tablespace 'USERS'
Login as sysdba:

```

```

SQL> select TABLESPACE_NAME, FILE_NAME, BYTES/1024/1024 from dba_data_files;

```

| TABLESPACE_NAME | FILE_NAME                                  | BYTES/1024/1024 |
|-----------------|--------------------------------------------|-----------------|
| SYSTEM          | /u01/app/oracle/oradata/cdb1/system01.dbf  | 810             |
| SYSAUX          | /u01/app/oracle/oradata/cdb1/sysaux01.dbf  | 500             |
| UNDOTBS1        | /u01/app/oracle/oradata/cdb1/undotbs01.dbf | 70              |
| USERS           | /u01/app/oracle/oradata/cdb1/users01.dbf   | 5               |
| TBS1            | /u01/app/oracle/oradata/cdb1/data01.dbf    | 10              |
| TBS1            | /u01/app/oracle/oradata/cdb1/data03.dbf    | 10              |

6 rows selected.

SQL> alter user gouse quota 100M on USERS;

User altered.

as a DBA user (depending on how much space you need / want to grant).

Now try with gouse user

```

SQL> insert into stud values(1);
1 row created.

```

SQL> select \* from stud;

| SNO |
|-----|
| 1   |

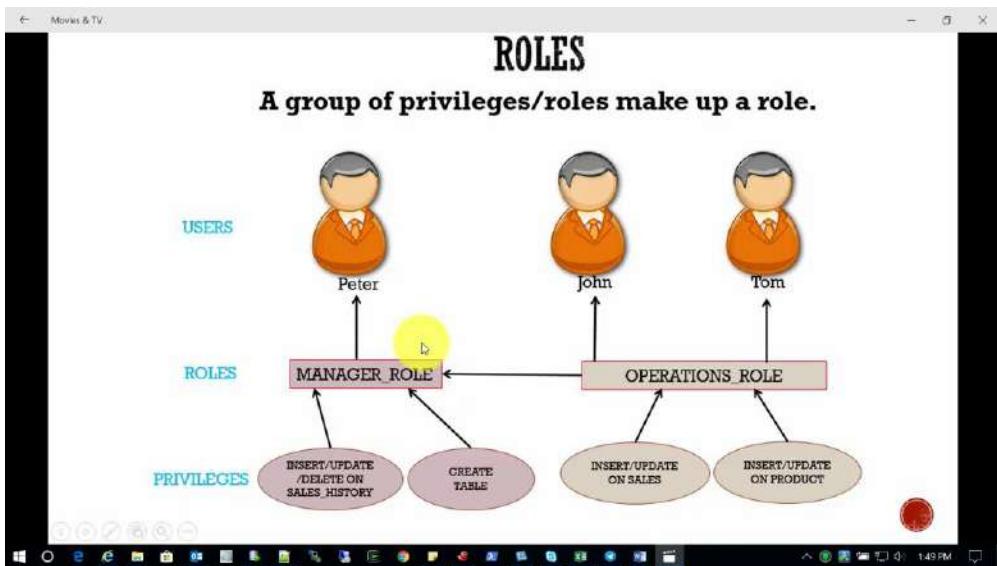
**ROLE [roles will be used instead of set of roles and assign to user]**

Assigning the roles to the users, based on providing privileges.

SQL> Create role <roleName>;

[OR]

SQL> Create role <roleName> identified by <password>; → for More secure Roles.



```
SQL> create ROLE MANAGER_ROLE;
SQL> grant insert, update, delete on DW.SALES_HISTORY TO MANAGER_ROLE;
SQL> grant create table to MANAGER_ROLE;
SQL> grant MANAGER_ROLE to PETER;
```

```
SQL> create ROLE OPERATIONS_ROLE;
SQL> grant insert, update on DW.SALES_HISTORY TO OPERATIONS_ROLE ;
SQL> grant insert, update on DW.PRODUCT_T TO OPERATIONS_ROLE ;
SQL> grant OPERATIONS_ROLE to MANAGER_ROLE; → yes we can assign one ROLE to another ROLE.
SQL> grant MANAGER_ROLE to JHON;
SQL> grant MANAGER_ROLE to GOUSE;
```

#### PROFILE

- Each user can have only one profile
- and default assigned to DEFAULT profile. It will be assigned when you are not assigned to user while initial stage of user creation.
- Profile is used to set of limits on database users as to manage users.
- Profiles are used to Resource consumption, manager accounts status and password expiration.
- How many cpus, Disk I/o, ideal time, connect Time, concurrent sessions, private SGA.

#### Profile | password Management

- We can manage passwords by using user profiles.
- The various security features we can use are:
  - Create profile <profile name> PASSWORD\_PRF LIMIT  
PASSWORD\_LIFE\_TIME 180 → days the password expiration days  
PASSWORD\_GRACE\_TIME 7 → after 180 days, the max of 7 days provided to user to change the password.  
PASSWORD\_REUSE\_TIME UNLIMITED → no.of days  
PASSWORD\_REUSE\_MAX UNLIMITED → no.of passwords attempts  
FAILED\_LOGIN\_ATTEMPTS 10 → wrong password attempts  
PASSWORD\_LOCK\_TIME 1 → the account lock for one day  
PASSWORD\_VERIFY\_FUNCTION verify function l1g; → built in function



```
SQL> select username, profile from dba_users where username='GOUSE';
SQL> select * from DBA_PROFILES;
SQL> select * from DBA_PROFILE where profile='DEFAULT';
```

File Edit View Terminal Tags Help

SQL> select username, profile from dba\_users where username='JOHN'; get the profile name based on user

| USERNAME | PROFILE |
|----------|---------|
| JOHN     | DEFAULT |

SQL> SELECT \* FROM DBA\_PROFILES WHERE PROFILE='DEFAULT'; Get the Resources which values has been configured for each and every default resource

| PROFILE | RESOURCE NAME             | RESOURCE LIMIT     |
|---------|---------------------------|--------------------|
| DEFAULT | COMPOSITE_LIMIT           | KERNEL UNLIMITED   |
| DEFAULT | SESSIONS_PER_USER         | KERNEL UNLIMITED   |
| DEFAULT | CPU_PER_SESSION           | KERNEL UNLIMITED   |
| DEFAULT | CPU_PER_CALL              | KERNEL UNLIMITED   |
| DEFAULT | LOGICAL_READS_PER_SESSION | KERNEL UNLIMITED   |
| DEFAULT | LOGICAL_READS_PER_CALL    | KERNEL UNLIMITED   |
| DEFAULT | IDLE_TIME                 | KERNEL UNLIMITED   |
| DEFAULT | CONNECT_TIME              | KERNEL UNLIMITED   |
| DEFAULT | PRIVATE_SGA               | KERNEL UNLIMITED   |
| DEFAULT | FAILED_LOGIN_ATTEMPTS     | PASSWORD 10        |
| DEFAULT | PASSWORD_LIFE_TIME        | PASSWORD 180       |
| PROFILE | RESOURCE_NAME             | RESOURCE LIMIT     |
| DEFAULT | PASSWORD_REUSE_TIME       | PASSWORD UNLIMITED |
| DEFAULT | PASSWORD_REUSE_MAX        | PASSWORD UNLIMITED |
| DEFAULT | PASSWORD_VERIFY_FUNCTION  | PASSWORD NULL      |
| DEFAULT | PASSWORD_GRACE_TIME       | PASSWORD 1         |
| DEFAULT | PASSWORD_GRACE_TIME       | PASSWORD 7         |

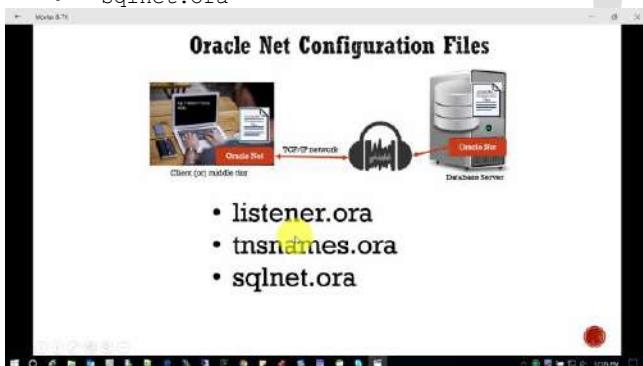
16 rows selected.

To know the version of oracle db

```
SELECT banner FROM v$version WHERE ROWNUM = 1;
```

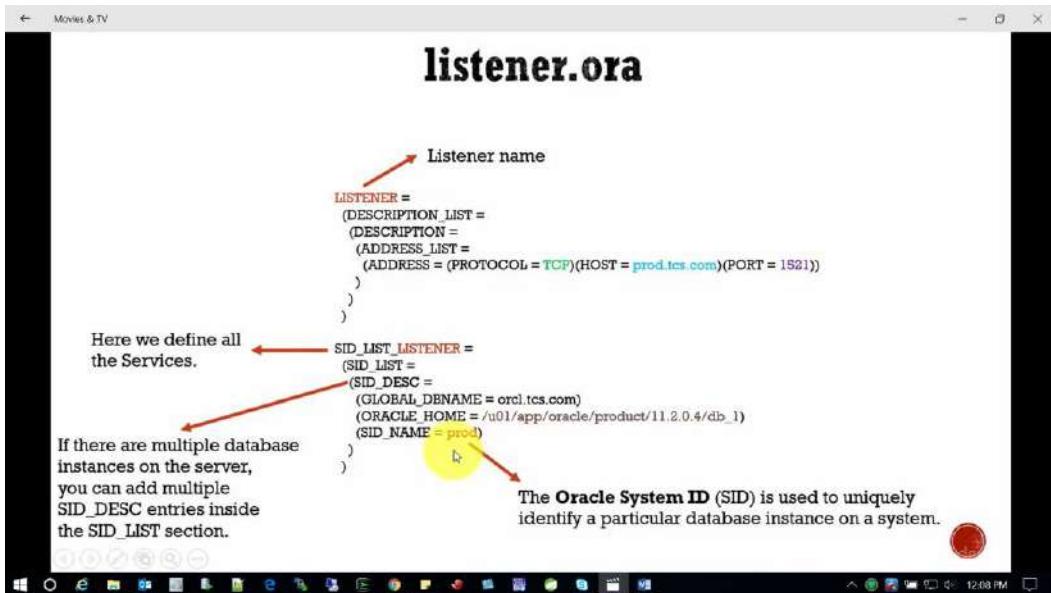
#### ORACLE NET SERVICES

- Listener.ora
- Tnsnames.ora
- Sqlnet.ora



#### Listener.ora [DB server]

All the network related configuration parameters location : \$ORACLE\_HOME/network/admin



Tools we can use for configuring and managing the Oracle Network

- Enterprise Manager
- Oracle Net Manager
- Oracle Net configuration Assistant
- Command Line

#### **Listener Control Utility:**

It enables you to control the listener

- To start the listener
- To stop the listener
- Check the status of the listener
- Reinitialize the listener from the configuration file parameters
- Dynamically configure many listeners
- Change the listener password.

```
$oracle> lsnrctl start
$oracle> lsnrctl status
$oracle> lsnrctl services
```

Copyright (c) 1991, 2013, Oracle. All rights reserved.

```
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=linux.oracle)(PORT=1521))
)
Services Summary...
Service "prod1" has 1 instance(s). → One instance is running on
Instance "prod1", status READY, has 1 handler(s) for this service...
  Handler(s):
    "DEDICATED" established:5 refused:0 state:ready
      LOCAL SERVER
Service "prod1XDB" has 1 instance(s). ← We can ignore this service
  Instance "prod1", status READY, has 1 handler(s) for this service...
    Handler(s):
```

#### **Client listner connection string:**

Connecting string take the following form

Username/password@hostname:port/service name

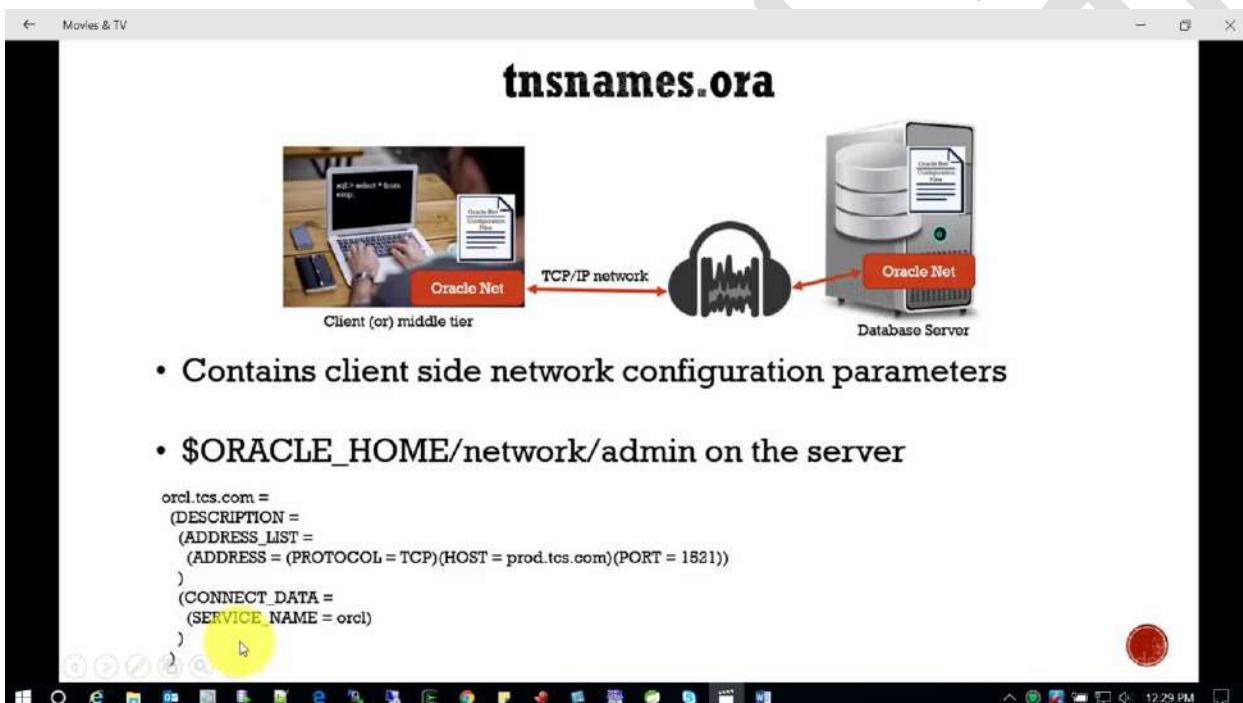
SQL> connect <hr/hr@prod.line.com:1521/orcl>



For the above connecting string we are making tnsnames.ora. It will place all files in Tnsnames.ora

### Tnsnames.ora [client machine]

It will be present in client machine to call the server listener.ora file with the help of connection strings in tnsnames.ora file.



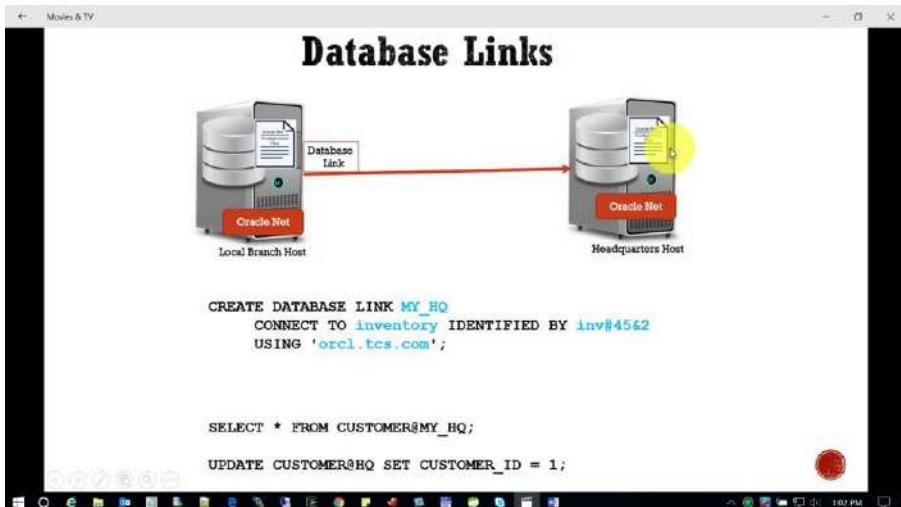
### Sqlnet.ora [Client Machine]

The **sqlnet.ora** file is the profile configuration file, and it resides on the client machines and the database server. The **sqlnet.ora** file is created by running the Network Configuration Assistant (NETCA). The **sqlnet.ora** is a text file that contains basic configuration details used by SQL\*Net.

### DataBase Link

Create a database link for connecting other databases like mysql, oracle and more..

```
SQL> Create database link MY_HQ connect to inventory identified by inv#12342 using 'orcl.tcs.com';
SQL> Select * from CUSTOMER@MY_HQ;
SQL> UPDATE CUSTOMER@MY_HQ SET CUSTOMER_ID=1;
```



1. Public database link
2. Private database link

#### Public database link

All the users in the database can use the database link

```
SQL> Create database public link Gouse_dblink connect to gouse identified by gouse#12342 using 'orcl.tcs.com';
← Public database link.
```

#### Private database link

All the user who created the database link can only use the Dblink

```
SQL> Create database link Gouse_dblink connect to gouse identified by gouse#12342 using 'orcl.tcs.com'; ←
Private database link.
```

#### # !Practiced DB Links

```
SQL> Create database link MY_HQ connect to inventory identified by inv#12342 using 'orcl.tcs.com';
SQL> Select * from CUSTOMER@MY_HQ;
SQL> UPDATE CUSTOMER@MY_HQ SET CUSTOMER_ID=1;
SQL> desc dba_db_links ← To know the what dblinks available in DB.
SQL> select * from dba_db_links
SQL> Create database link Gouse_dblink connect to gouse identified by gouse#12342 using 'orcl.tcs.com'; ←
Private database link.
SQL> select * from s$org_ext@gouse_dblink; ← to get the data from the created dblink.
```

#### DATA Dictionary [DBA Views for tables]

Is nothing but a set of **tables** and **views** information about the database. And are owned by the user SYS. Data about data is called as data dictionary.

User friendly views are built on top of these base tables.

**Exp:** DBA\_tables;

They are 1200 data dictionaries.

They are two types.

1. Static views : contains database objects , ddl operations are executed. ex: USER\_TABLES
2. Dynamic views : the data keeps on changing , ex : v\$PARAMETER
3. Global dynamic views : for multiple instances, ex: GV\$PARAMETER
4. Undocumented views : they did not documented, ex: x\$KSPPI

#### • Static views

Contains information data about objects, privileges, users.. etc.  
The views starting with USER\_, ALL\_, DBA\_

**STATIC VIEWS**

Static Views contain data about objects, privileges, users..etc

Contents are changed by DDL Operations.

Static Views are of three types...

**USER\_**

- ✓ Displays only the information about current user objects
- ✓ Ex: USER\_TABLES

**ALL\_**

- ✓ Displays information about current user objects and other objects the user has privileges on.
- ✓ Ex: ALL\_TABLES

**DBA\_**

- ✓ Displays information about all objects. (Should have a SELECT\_CATALOG\_ROLE role)
- ✓ Ex: DBA\_TABLES

**DBA VIEWS**

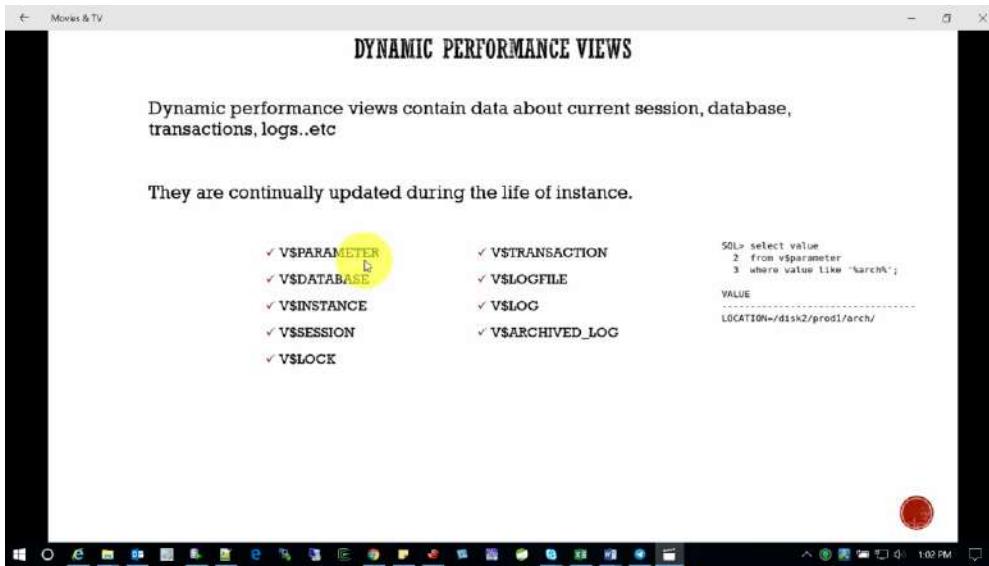
**COMMON DBA VIEWS**

|                   |                  |
|-------------------|------------------|
| ✓ DBA_TABLES      | ✓ DBA_EXTENTS    |
| ✓ DBA_TAB_COLUMNS | ✓ DBA_FREE_SPACE |
| ✓ DBA_CONSTRAINTS | ✓ DBA_VIEWS      |
| ✓ DBA_INDEXES     | ✓ DBA_SYNONYMS   |
| ✓ DBA_TABLESPACES | ✓ DBA_SYS_PRIVS  |
| ✓ DBA_DATA_FILES  | ✓ DBA_TAB_PRIVS  |
| ✓ DBA_OBJECTS     | ✓ DBA_TRIGGERS   |
| ✓ DBA_SEGMENTS    | ✓ DBA_TS_QUOTAS  |

#### DYNAMIC PERFORMANCE VIEWS

These views contains data about current sessions, database, transactions, logs.. etc. They are continually updated during the life of instance.

V\$PARAMETER    V\$DATABASE    V\$INSTANCE    V\$SESSION    V\$LOCK    V\$TRANSACTION    V\$LOGFILE    V\$LOG    V\$ARCHIVED\_LOG



### MISCELLANEOUS VIEWS

- ✓ Session Privileges : system privileges to the session which user logged in.
- ✓ Dictionary : it contains the information all data dictionary table/view.
- ✓ DICT\_COLUMNS : its like a dictionary and for each and every column in data dictionary
- ✓ TABLE\_PRIVILEGES : tables privileges for the current user.

### # Practiced

To get the all current user has the access for the tables.

#### USERS level:

```
SQL> describe user_tables;
SQL> select TABLE_NAME, TABLESPACE_NAME from user_tables;
SQL> desc all_tables;
SQL> select OWNER, TABLE_NAME, TABLESPACE_NAME, STATUS from all_tables;
SQL> select OWNER, TABLE_NAME, TABLESPACE_NAME, STATUS from all_tables where TABLESPACE_NAME= 'DATA01';
```

#### DBA Level:

To check whether he is having DBA privileges for the current user , execute below statement

```
SQL> desc DBA_TABLES;
SQL> select OWNER, TABLE_NAME, TABLESPACE_NAME, STATUS from DBA_tables where TABLESPACE_NAME= 'DATA01';
SQL> desc DBA_USERS;
SQL> select USERNAME, USER_ID, PASSWORD, ACCOUNT_STATUS from DBA_USERS;
SQL> desc DBA_TABLESPACES;
SQL> select TABLESPACE_NAME, BLOCK_SIZE from DBA_TABLESPACES;
```

#### Data files:

```
SQL> desc DBA_DATA_FILES;
SQL> select FILE_NAME, STATUS from DBA_DATA_FILES;
```

#### Views

##### Archive LogFiles:

```
SQL> desc v$ARCHIVED_LOG;
SQL> select name from v$ARCHIVED_LOG;
```

##### Sessions:

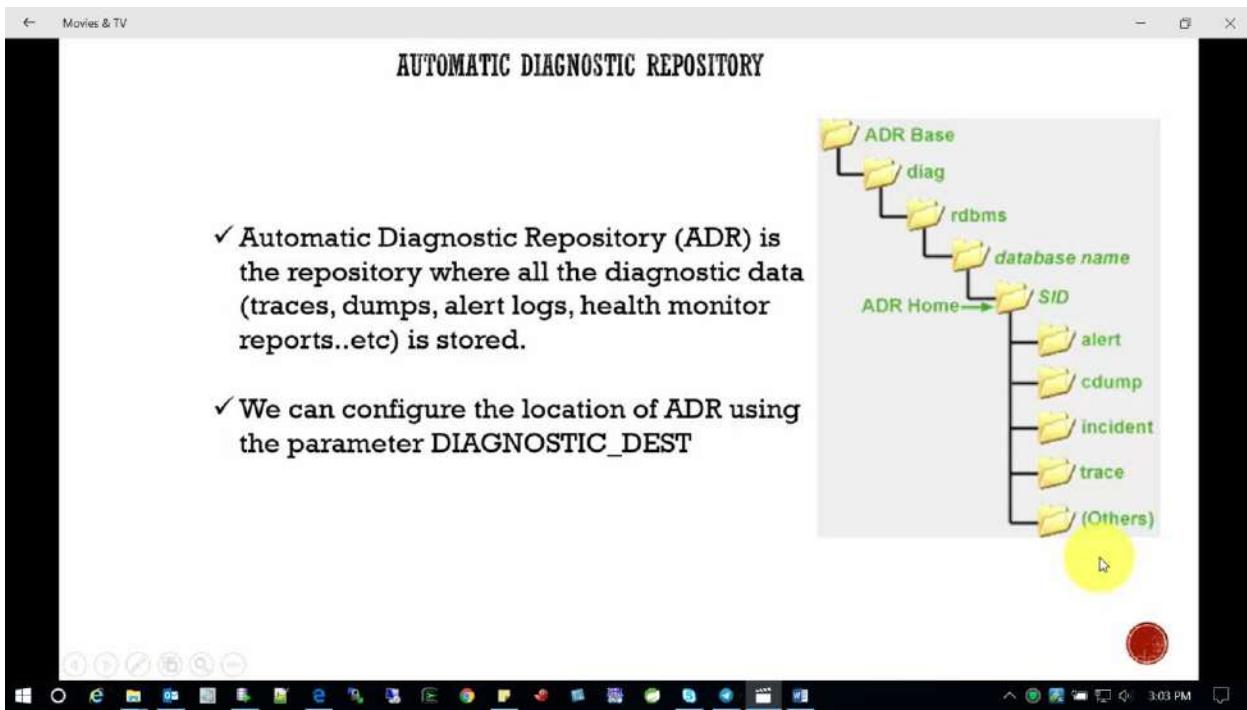
```
SQL> desc V$SESSION;
SQL> select SID from V$SESSION; → here sid(system id's ) will be run by system manager.
```

#### Diagnostic DATA

Diagnostic data is the set of information generated by the oracle database which is used to diagnose/research the errors occurred in the database.

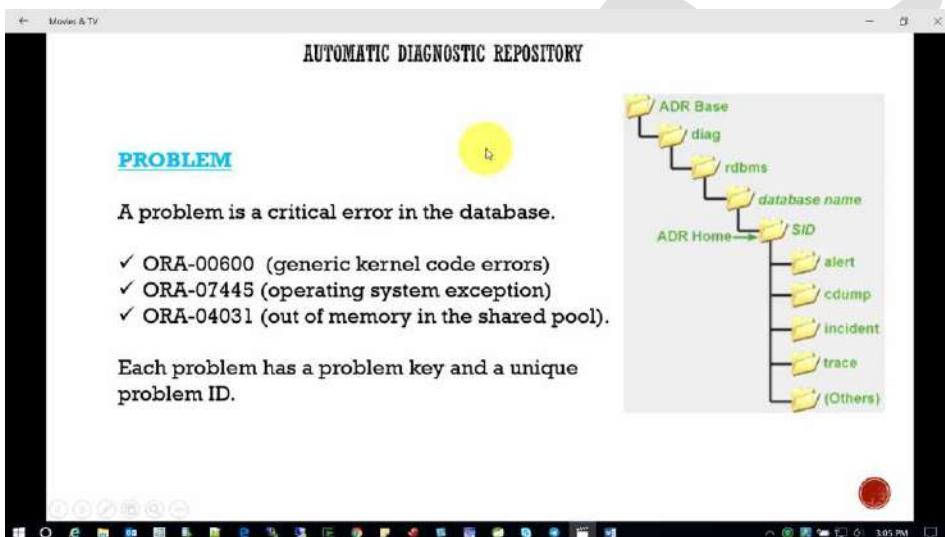
- **Alert Logs:** errors, messages, incidents for to check the logs.from 11g it will be in xml files.
- **Trace files :** for background process, individual processes. Each trace file has a uniq name.and it can be monitor by enabling the trace.
- **Core dump files :** all error related information will be stored in core dumps when the sevior errors faced by database.
- **Other kind of files..** which are stored to use for diagnostic information.

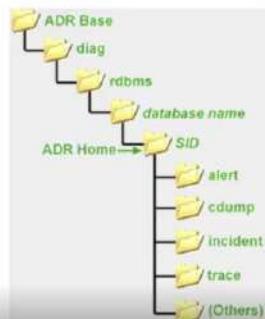
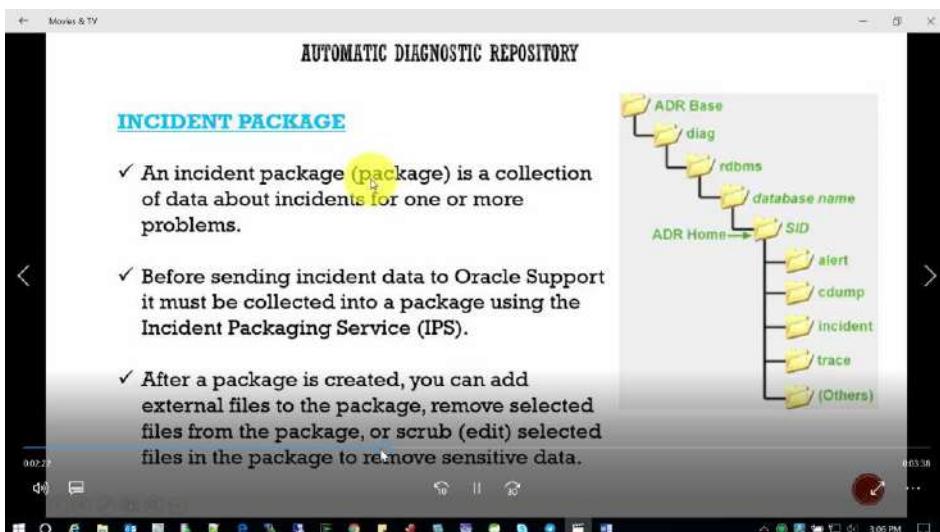
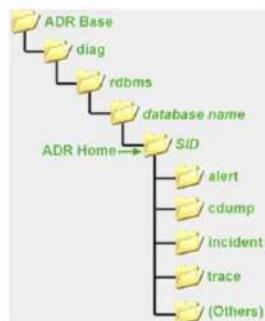
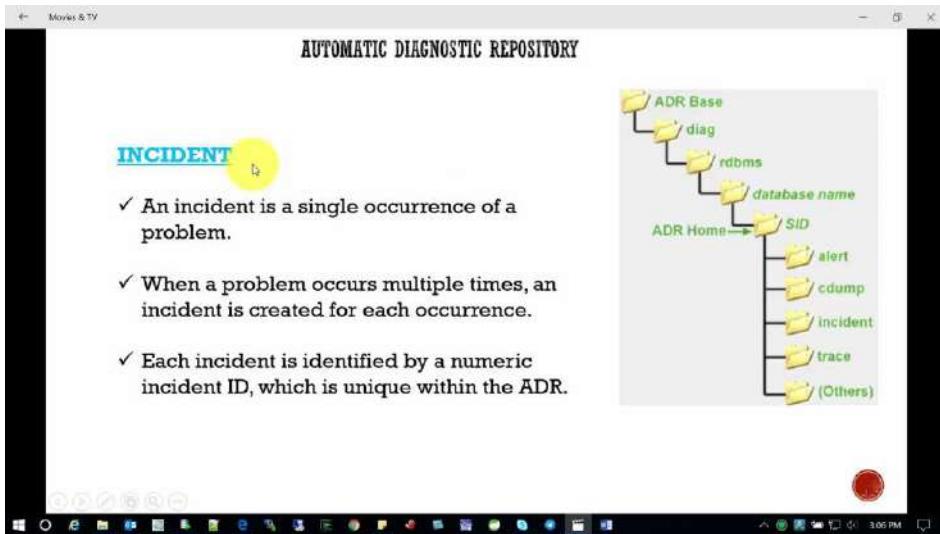
#### Automatic Diagnostic Repository (ADR)



From 11g all diagnostic data will be stored on ADR.

- **Problem** : is a critical error in DB, oracle support is needed. ORA-00600,ORA-07445,ORA-04031. Identified by problem id.
- **Incident** : is a single occurrence of a problem. It will be created for each occurrence. Identified by incident id
- **Incident package** : after package is created you can add for an incident.





```
# !Practiced:
SQL> show parameter diagnostic_dest;

NAME                                     TYPE        VALUE
-----                                     -----
diagnostic_dest                          string      /oracle/sbtsitn

Cd /oracle/sbtsitn

[oracle@linux ~]$ cd /u01/app/oracle
[oracle@linux oracle]$ ls -lrt
total 44
drwxr-xr-x  3 oracle oinstall 4096 Jan 23 21:32 product
drwxrwxr-x 11 oracle oinstall 4096 Jan 23 21:46 diag
drwxr-xr-x  2 oracle oinstall 4096 Jan 23 21:47 checkpoints
drwxr-x---  3 oracle oinstall 4096 Jan 29 19:28 admin
drwxr-x---  6 oracle oinstall 4096 Mar 12 20:29 cfgtoollogs
drwxr-x---  3 oracle oinstall 4096 Apr 26 13:38 oradata
```

product  
diag  
checkpoints  
admin  
cfgtoollogs  
oradata

Trace file reading snapshot, it will be available under diag directory.

Movies & TV

```

oracle@linux:u01/app/oracle/diag/rdbms/prod1/prod1/trace
File Edit View Terminal Tabs Help
STAT #140426831117736 id=6 cnt=42 pid=5 pos=1 obj=0 op='FIXED TABLE FULL X$KSLWT (cr=0 pr=0 pw=0 time=220 us cost=0 size=26
00 card=100'
STAT #140426831117736 id=7 cnt=1 pid=2 pos=2 obj=0 op='FIXED TABLE FIXED INDEX X$KSUSE (ind:1) (cr=0 pr=0 pw=0 time=14063 u
s cost=0 size=71 card=1)'
STAT #140426831117736 id=8 cnt=1 pid=1 pos=2 obj=0 op='FIXED TABLE FIXED INDEX X$KSLED (ind:2) (cr=0 pr=0 pw=0 time=5 us co
st=0 size=13 card=1)'

*** 2017-05-04 19:44:25.794
CLOSE #140426831117736:c=0,e=0,dep=0,type=0,tim=1493941465794371

PARSING IN CURSOR #140426831117736 len=42 dep=0 uid=0 oct=3 lid=0 tim=1493941465795328 hv=2270478018 ad='8b6b10e0' sqlid='d
nt6xy3p9nq'
select * from customer where customer_id=2
END OF STMT
PARSE #140426831117736:c=0,e=81,p=0,cr=0,cu=0,mis=1,r=0,dep=0,og=1,plh=2844954298,tim=1493941465795327
EXEC #140426831117736:c=0,e=14,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=1,plh=2844954298,tim=1493941465795395
FETCH #140426831117736:c=0,e=35,p=0,cr=6,cu=0,mis=0,r=1,dep=0,og=1,plh=2844954298,tim=1493941465795456
FETCH #140426831117736:c=0,e=4,p=0,cr=0,cu=0,mis=0,r=0,dep=0,og=1,plh=2844954298,tim=1493941465795717
STAT #140426831117736 id=1 cnt=1 pid=0 pos=1 obj=0 dep=7501 op='TABLE ACCESS FULL CUSTOMER (cr=6 pr=0 pw=0 time=31 us cost=3 siz
e=3 card=1)'

*** 2017-05-04 19:51:35.341
CLOSE #140426831117736:c=0,e=8,dep=0,type=0,tim=1493941895341011

PARSING IN CURSOR #140426831117736 len=289 dep=0 uid=0 oct=3 lid=0 tim=1493941895341618 hv=2462394820 ad='8b59de48' sqlid='
7crz5wy9caaf4'
SELECT NAME NAME_COL_PLUS_SHOW_PARAM,DECODE(TYPE,1,'boolean',2,'string',3,'integer',4,'file',5,'number',
6,'big integer','unknown') TYPE_DISPLAY_VALUE VALUE_COL_PLUS_SHOW_PARAM FROM V$PARAMETER WHERE UPPER(NAME) LIKE UPPER(:NMBIND_SHOW_0B
J) ORDER BY NAME_COL_PLUS_SHOW_PARAM,ROWNUM
END OF STMT
PARSE #140426831117736:c=0,e=472,p=0,cr=0,cu=0,mis=1,r=0,dep=0,og=1,plh=0,tim=1493941895341617

```

Movies & TV

```

oracle@linux:u01/app/oracle/diag/rdbms/prod1/alert
File Edit View Terminal Tabs Help
*** 2017-05-03 07:46:55.865
*** SESSION ID:(18,5) 2017-05-03 07:46:55.865
*** CLIENT ID:() 2017-05-03 07:46:55.865
*** SERVICE NAME:(SV$BACKGROUND) 2017-05-03 07:46:55.865
*** MODULE NAME:() 2017-05-03 07:46:55.865
*** ACTION NAME:() 2017-05-03 07:46:55.865

Initial buffer sizes: read 1024K, overflow 832K, change 805K
Log read is SYNCHRONOUS though disk_asynch_io is enabled!
Log read is SYNCHRONOUS though disk_asynch_io is enabled!
[oracle@linux trace]$ cd ..
[oracle@linux prod1]$ ls -lrt
total 124
drwxr-x--- 2 oracle oinstall 4096 Jan 29 19:28 metadata_pv
drwxr-x--- 2 oracle oinstall 4096 Jan 29 19:28 metadata_dgif
drwxr-x--- 2 oracle oinstall 4096 Jan 29 19:28 incpkd
drwxr-x--- 2 oracle oinstall 4096 Jan 29 19:28 hm
drwxr-x--- 2 oracle oinstall 4096 Jan 29 19:28 cdump
drwxr-x--- 2 oracle oinstall 4096 Jan 29 19:28 alert
drwxr-x--- 2 oracle oinstall 4096 Jan 29 19:30 metadata
drwxr-x--- 2 oracle oinstall 4096 Apr 28 21:54 iu
drwxr-x--- 7 oracle oinstall 4096 Apr 28 22:41 incident
drwxr-x--- 2 oracle oinstall 4096 Apr 28 22:42 sweep
drwxr-x--- 2 oracle oinstall 4096 Apr 28 22:42 stage
drwxr-x--- 2 oracle oinstall 4096 Apr 28 22:42 lck
drwxr-x--- 9 oracle oinstall 24576 May 4 19:28 trace
[oracle@linux prod1]$ cd alert
[oracle@linux alert]$ ls -lrt
total 948
-rw-r----- 1 oracle oinstall 953538 May 4 19:28 log.xml
[oracle@linux alert]$ 

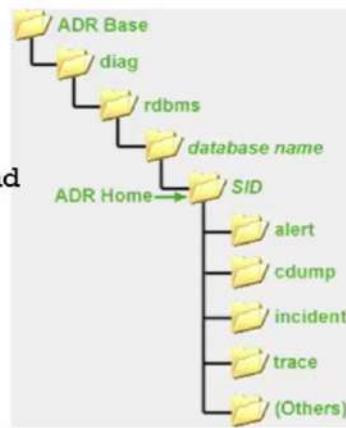
```

#### Automatic Diagnostic Repository command interpreter (ADRCI)

Is utility to read diagnostic files with this.

## ADRCI

- ✓ **ADRCI (Automatic Diagnostic Repository Command Interpreter)** is a utility to read the diagnostic files.



# !Practiced:

cd /u01/app/oracle/

Run the command to go in adrci utility.

\$adrci

adrci> help [To get all command in this utility]

```
[oracle@linux cdump]$ pwd
/u01/app/oracle/diag/rdbms/prod1/prod1/cdump
[oracle@linux cdump]$ cd /u01/app/oracle
[oracle@linux oracle]$ pwd
/u01/app/oracle
[oracle@linux oracle]$ ls -lrt
total 44
drwxr-xr-x  3 oracle oinstall 4096 Jan 23 21:32 product
drwxrwxr-x 11 oracle oinstall 4096 Jan 23 21:46 diag
drwxr-xr-x  2 oracle oinstall 4096 Jan 23 21:47 checkpoints
drwxr-x---  3 oracle oinstall 4096 Jan 29 19:28 admin
drwxr-x---  6 oracle oinstall 4096 Mar 12 20:29 cfgtoollogs
drwxr-x---  3 oracle oinstall 4096 Apr 26 13:38 oradata
[oracle@linux oracle]$ adrci
```

ADRDI: Release 11.2.0.4.0 - Production on Thu May 4 20:06:34 2017

Copyright (c) 1982, 2011, Oracle and/or its affiliates. All rights reserved.

ADR base = "/u01/app/oracle"  
adrci>

Creating a incident and make a package to send the problem to oracle support.

Adrci> show problem

Adrci> show incident

```

adrci> show problem
ADR Home = /u01/app/oracle/diag/rdbms/prod1/prod1
-----
PROBLEM_ID      PROBLEM_KEY                               LAST INCIDENT    LASTINC_TIME
-----
2              ORA 603                                     19281          2017-04-28 22:41:46,3
61000 -04:00
1              ORA 600 [krhpfn_03-1208]                   18017          2017-04-28 22:41:49,0
13000 -04:00
2 rows fetched

adrci> show incident
ADR Home = /u01/app/oracle/diag/rdbms/prod1/prod1
-----
INCIDENT_ID      PROBLEM_KEY                               CREATE TIME
-----
18153           ORA 600 [krhpfn_03-1208]                 2017-04-28 22:41:26,428000 -04:00
18154           ORA 600 [krhpfn_03-1208]                 2017-04-28 22:41:33,618000 -04:00
18155           ORA 600 [krhpfn_03-1208]                 2017-04-28 22:41:38,596000 -04:00
19201           ORA_603                                 2017-04-28 22:41:46,361000 -04:00
18017           ORA 600 [krhpfn_03-1208]                 2017-04-28 22:41:49,013000 -04:00
5 rows fetched

```

Adrci> show incident -mode detail -p "incident\_id='18153';"

Adrci> ips create package problem 1 correlate all

Adrci> ips generate package 1 in "/u01"

Now exit and go to /u01 and you can find the .zip file which one is packaged for the incident..

```

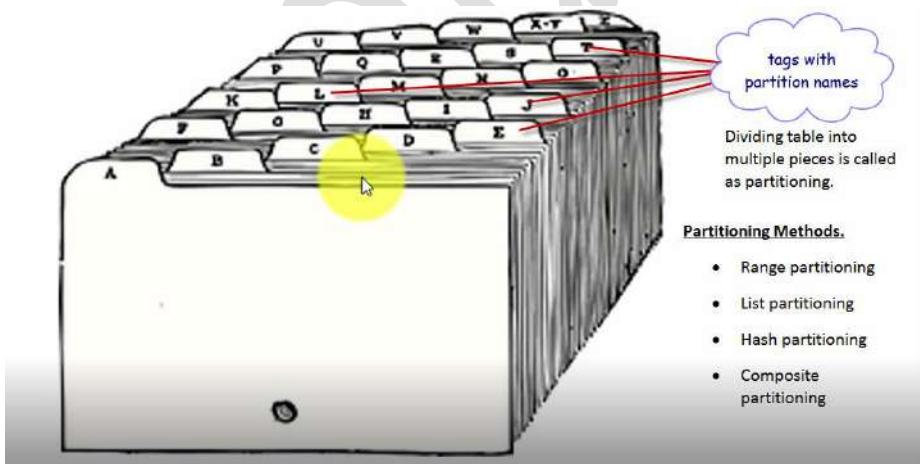
adrci> ips create package problem 1 correlate all
Created package 1 based on problem id 1, correlation level all
adrci> ips generate package 1 in "/u01"
Generated package 1 in file /u01/ORACLE600krh_20170504203117_COM_1.zip, mode complete
adrci> exit
[oracle@linux oracle]$ cd /u01
[oracle@linux u01]$ ls -lrt
total 12992
drwxr-x--x 2 oracle oinstall 16384 Jan 21 13:07 lost+found
drwxr-xr-x 4 oracle oinstall 4096 Jan 23 21:32 app
-rw-r--r-- 1 oracle oinstall 13257468 May 4 20:32 ORACLE600krh_20170504203117_COM_1.zip
[oracle@linux u01]$
```

#### Table Partitioning

Dividing table into multiple files is called as partitioning.

Oracle provide the following partitioning methods.

- Range partitioning
- List partitioning
- Hash partitioning
- Composite partitioning



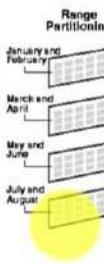
#### Range partitioning

Divided into 4 partition, and from month wise date to date.

## Range Partition

**Range partitioning** is a partitioning technique where data is stored separately in different sub-tables based on the data range.

```
CREATE TABLE SALES1
(
customer_id NUMBER,
order_date DATE,
order_amount number,
Region varchar2(10)
)
PARTITION BY RANGE (order_date)
(
PARTITION sales_p1507 VALUES LESS THAN (TO_DATE('2015-07-01', 'YYYY-MM-DD')),
PARTITION sales_p1508 VALUES LESS THAN (TO_DATE('2015-08-01', 'YYYY-MM-DD')),
PARTITION sales_p1509 VALUES LESS THAN (TO_DATE('2015-09-01', 'YYYY-MM-DD')),
PARTITION sales_pmax VALUES LESS THAN (MAXVALUE)
);
```



```
Create table sales1
( customer id NUMBER,
Order_date DATE,
Orader amount number,
Region varchar2(10)
)
PARTITION BY RANGE (order_date)
(
PARTITION sales_p1507 values less than (to_date('01-jan-2015','dd-mm-yyyy')),
PARTITION sales_p1507 values less than (to_date('01-feb-2015','dd-mm-yyyy')),
PARTITION sales_p1507 values less than (to_date('01-mar-2015','dd-mm-yyyy')),
PARTITION sales_p1507 values less than (to_date('01-jun-2015','dd-mm-yyyy')),
PARTITION sales_p1507 values less than (MAXVALUE)
);
```

## DB Schema migration Activity

As discussed, please provide the list of all objects from BRMPROD AND BRM-PP

- 1) Select owner,count(\*),object\_type from dba\_objects group by owner,object\_type;
- 2) Select owner,object\_type,object\_name from dba\_objects order by 1;

|      |                |         |          |                 |
|------|----------------|---------|----------|-----------------|
| PP   | Oracle BRM 7.5 | BRMPREP | brmprep  | eecsaruh2hoa190 |
| PROD | Oracle BRM 7.5 | BRMPROD | brmprod1 | eecsaruh6hoa121 |

Note: Excel format

Then compare the objects , which are not available in PP. migrate those objects for the same.

## Grarfana

Official Website : <https://grafana.com/grafana/download?platform=linux>  
For installation steps: <http://docs.grafana.org/installation/rpm/>

## Monitor and visualizing Data with grafana

- Grafana is a tool which will allows you to query, visualize, alert on understand your metrics.
- It can retrieve data from a wide range of data Source.
- It visualizes data with modern looking gauges
- Data can be visualized as a CURRENT state and/or as a TREND
- Metrics of different time periods can be compared.
- Grafana can raise alerts based on given criteria and to a variety of platforms:
  - Email
  - Slack

## Use Cases of Grafana:

- Monitor the health of your infrastructure e.g. via CloudWatch
- Monitor the status of your applications e.g. #of errors, average response time etc.
- Visualise your business data e.g # of orders per minutes, # of cancelled orders etc.
- Monitor your customer satisfaction by visualizing # of support tickets
- Quickly take action and resolve operational issues by setting up Alerts.

```
• yum install https://s3-us-west-2.amazonaws.com/grafana-releases/release/grafana-5.1.4-1.x86_64.rpm
```

## Start the server (init.d service)

- \$ sudo service grafana-server start
- This will start the grafana-server process as the grafana user, which is created during package installation. The default HTTP port is 3000, and default user and group is admin.
- Default login and password admin/ admin
- To configure the Grafana server to start at boot time:  
\$ sudo /sbin/chkconfig --add grafana-server

## Start the server (via systemd)

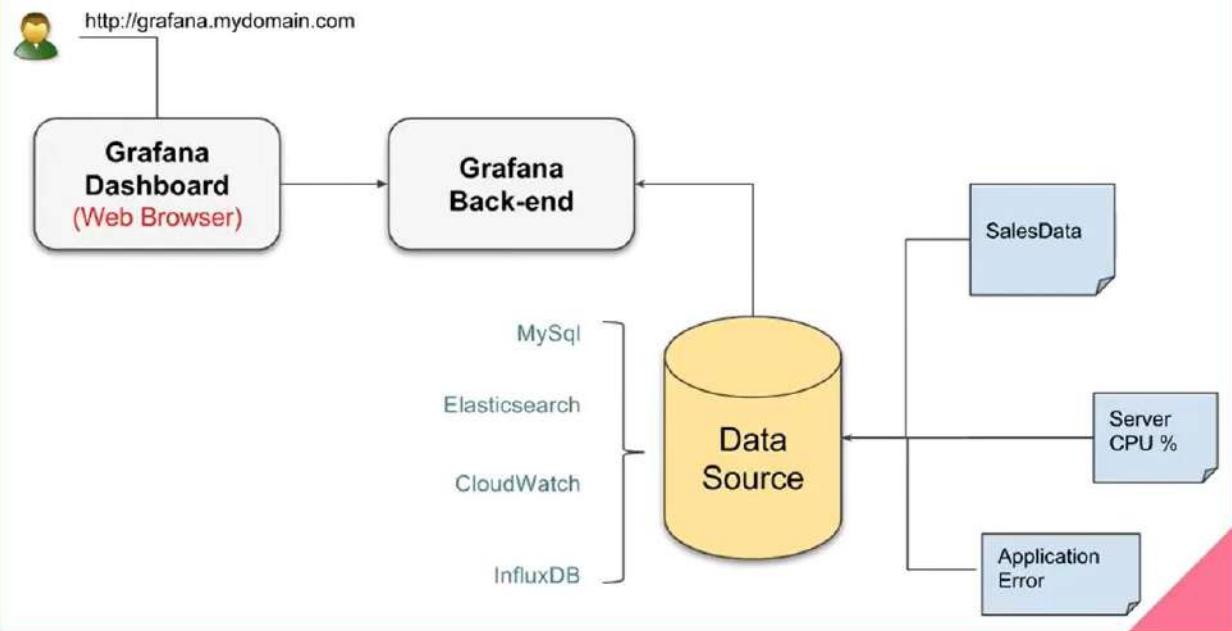
- \$ systemctl daemon-reload
- \$ systemctl start grafana-server
- \$ systemctl status grafana-server
- Enable the systemd service to start at boot
- sudo systemctl enable grafana-server.service
- Environment file
- The systemd service file and init.d script both use the file located at /etc/sysconfig/grafana-server for environment variables used when starting the back-end. Here you can override log directory, data directory and other variables.
- Logging
- By default Grafana will log to /var/log/grafana
- Database
- The default configuration specifies a sqlite3 database located at /var/lib/grafana/grafana.db. Please backup this database before upgrades. You can also use MySQL or Postgres as the Grafana database, as detailed on the configuration page.
- Configuration
- The configuration file is located at /etc/grafana/grafana.ini. Go the Configuration page for details on all those options.
- 

## Grafana Configuration Steps:

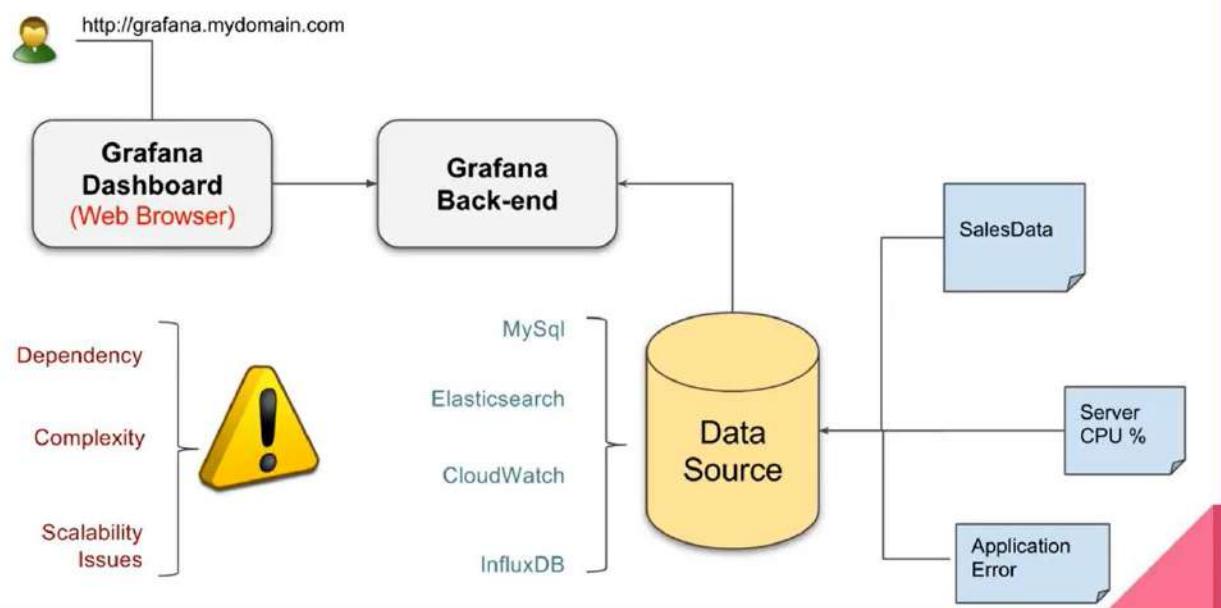
- The zip file contains a folder with the current Grafana version. Extract this folder to anywhere you want Grafana to run from. Go into the conf directory and copy `sample.ini` to `custom.ini`. You should edit `custom.ini`, never `defaults.ini`.
- The `default Grafana port is 3000`, this port requires extra permissions on Windows. Edit `custom.ini` and uncomment the `http_port` configuration option (; is the comment character in ini files) and change it to something like 8080 or similar. That port should not require extra Windows privileges.
- Default login and password `admin/ admin`
- Start Grafana by executing `grafana-server.exe`, located in the bin directory, preferably from the command line. If you want to run Grafana as Windows service, download NSSM. It is very easy to add Grafana as a Windows service using that tool.

## How Grafana Works:

# How Grafana works

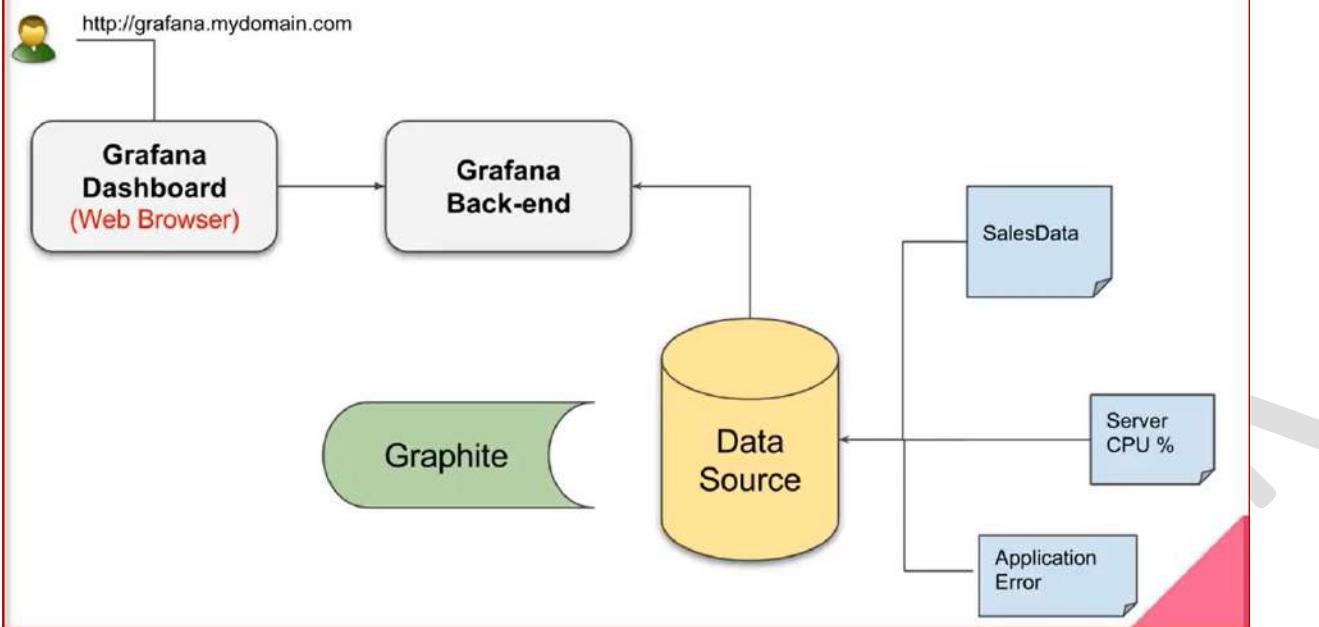


# How Graphite works



So better to use one datasource, i.e Graphite to avoid above issue [Dependency, Complexity, Scalability]

# How Graphite works



## \* What is Graphite:

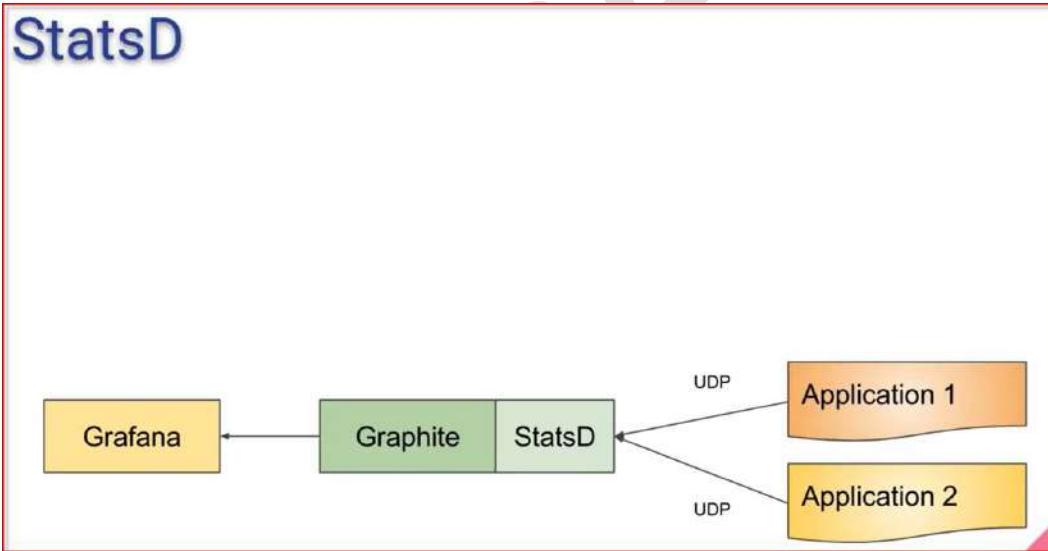
Graphite is a monitoring tool which is enterprise-ready and can be used alone to monitor websites, applications, business data and server.

## Why Graphite is a preferred data source?

It provides API which eliminates the need of dealing with various types of data sources

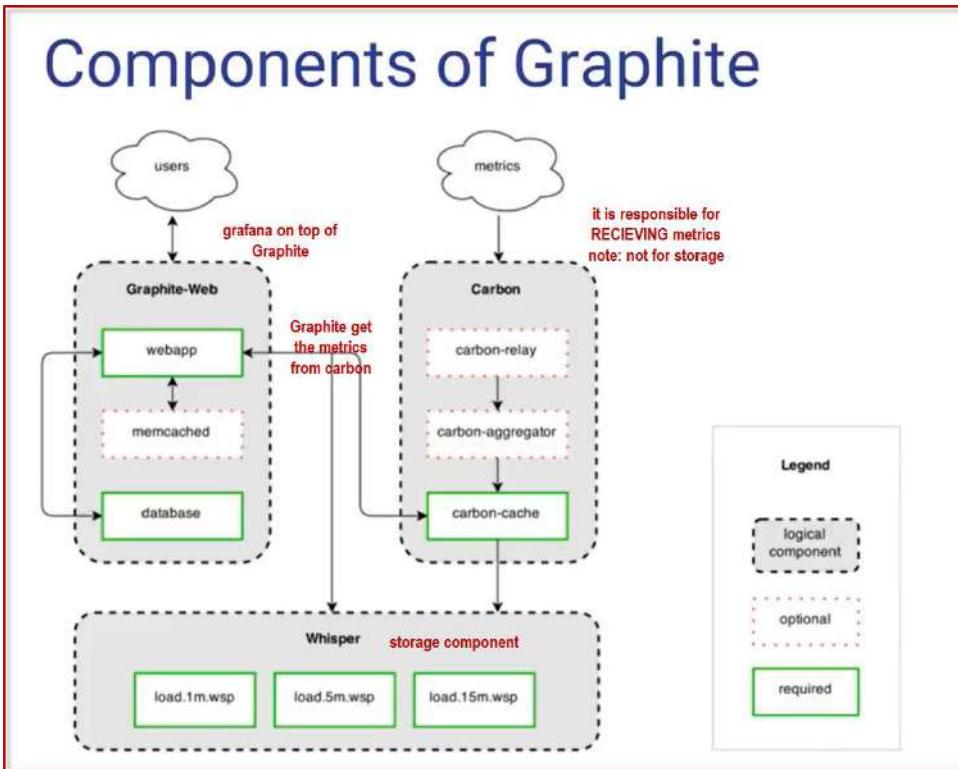
## What is STATSD?

# StatsD



- Statsd is normally paired with applications, which used to create a matrix for the applications.
- Statsd is a simple Daemon to aggregate and summarise metrics
- Metrics are in the form of plain text
- StatsD (by Default) works based on UDP which makes it lightweight and non-blocking

## Components of Graphite



## StatsD

StatsD gather metric values and sends them to Graphite (carbon)

**StatsD Gathers metric values:**

## StatsD

StatsD gathers metric values and sends them to Graphite (Carbon)

StatsD datagram:

`<bucket>:<value>|<type>@<sample rate>`

- A **bucket** is an identifier for the metrics. E.g. `MyService.Api.Invocation`
- **Value** is a number associated with a bucket. Values have different meanings depending on the metric type.
- **Type** determines that what type of event the metrics represents.
- **Sample Rate** indicates that what percentage of data must be sent to the server.

StatsD Diagram:

`<Bucket>:<value>|<type>@<sample rate>`

- A **bucket** is an identifier for the metrics. Ex: `MyService.Api.Invocation`
- **Value** is a number associated with a bucket. Values have different meanings depending on the metric type.
- **Type** determines that what type of event the metrics represents.
- **Sample Rate** indicates that what percentage of data must be sent to the server.

**Types of Metrics:**

**Counter:** is specified in the datagram with a "c" and shows the number of occurrences of an event. The Value must be a positive or negative whole number [`MyService.Api.Invocation:10|c`]

**Timer:** is specified in the datagram with "ms"[milli Seconds] and shows how long an event took to complete.

[MyService.Api.Invocation:120|ms]

**Gauge:** is specified in the datagram with "g" and represents a persistent value. Once set, the server will report the same value until it is changed. The + and - sign must be used to change the value

MyService.Product.AveragePrice:10|g (Set to average value price to 10)

MyService.Product.AveragePrice:+2|g (increase the average price to 12)

**Sets:** is specified with "s" and reports the number of unique occurrences of a metric. The value is the unique identifier of the occurrence.

Identity.users:Jack|s

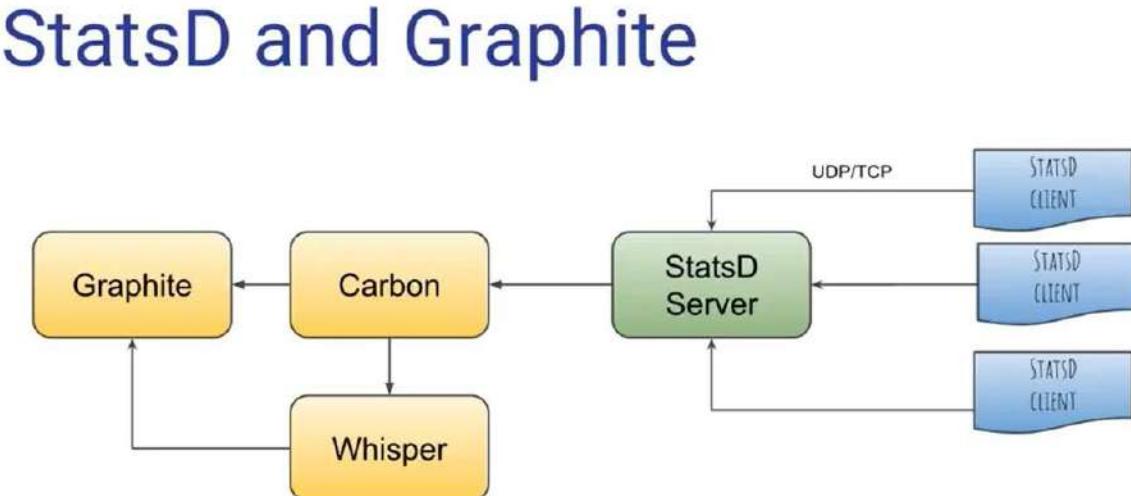
Identity.users:John|s

Sampling

- Is intended to reduce the number of metrics that is sent to the server
- Determines that what percentage of metrics should be sent to the server
- The server accounts for sampling by dividing the # of metrics to the sampling rate

Identity.users:John|s@0.1 [only 10% of samples are sent to the server, and server divides the number of received samples to 0.1]

## StatsD and Graphite



### StatsD Clients:

- The default protocol is UDP
- TCP is supported

### StatsD frequencies and histories/retentions

**Frequency** determines that how often the server must save metrics. Any metric received between two frequencies is ignored.

**History or retention** determines that for how long the metrics must be stored.

- s- second
- m- minute
- h- hour
- d- day
- w- week
- y- year

Multiple retention periods can be defined. Each retention period can have a different frequency.

```
1s:1d,10s:7d,1m:1y
```

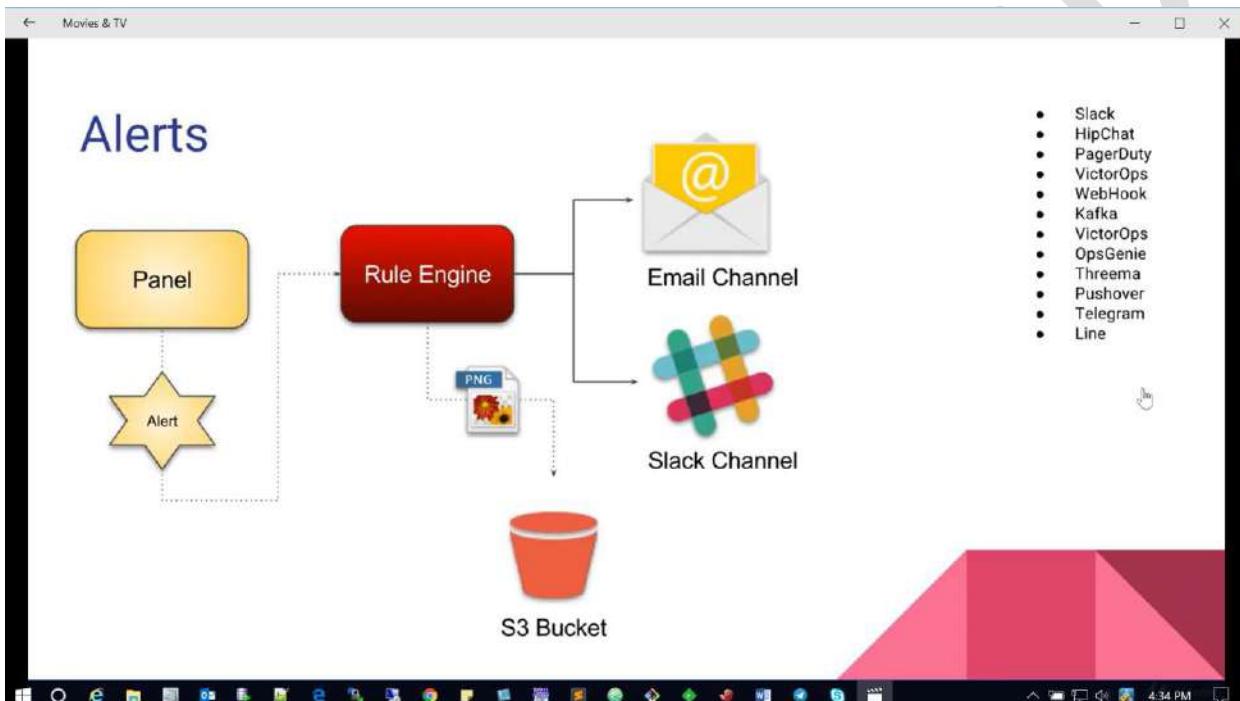
! Note: after 1 year the data will be wiped away.

- Frequencies and histories are defined in `/etc/carbon/storage-schemas.conf`
- Frequencies and histories can be applied on certain metrics/buckets using Regex

Example :

```
[default]
Pattern=*
Retentions=1s:1d,30s:7d,1m:1y
```

```
[logins]
Pattern=^login
Retentions=1m:1y
```



## Grafana installation:

```
Make sure firewall should be opened before implementing
http:80
ssh: 22
custom TCP port : 443 [https]
custom TCP Port : 8123 [Sends metrics to graphite]
```

## Install Mysql for grafana:

```
vagrant@manager:~$ sudo apt-get install
```

```
Reading package lists... Done
```

```
Building dependency tree
```

```
Reading state information... Done
```

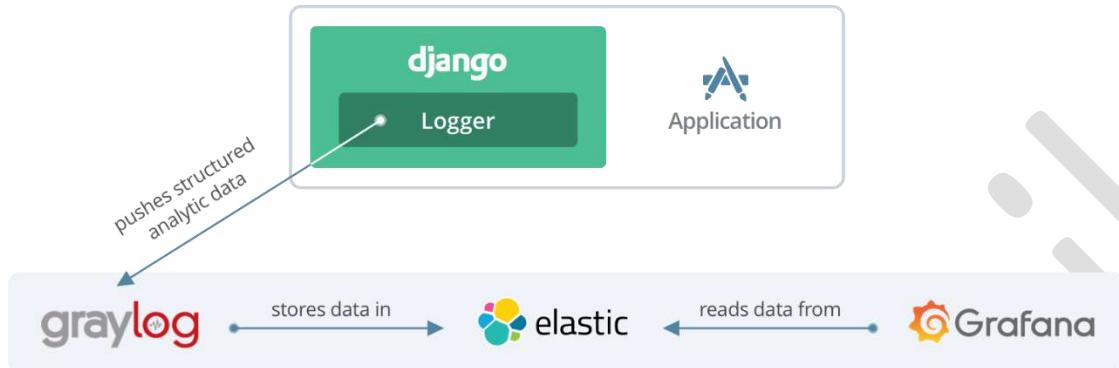
```
0 upgraded, 0 newly installed, 0 to remove and 95 not upgraded.
```

```
vagrant@manager:~$ sudo apt-get install mysql-server
```

```
Do you want to continue? [Y/n] y
```

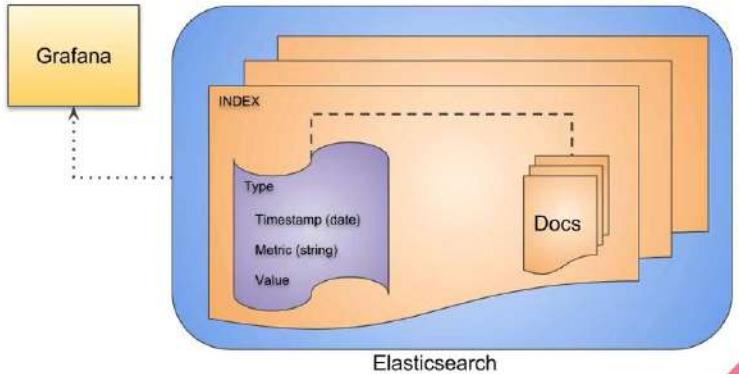
## Grafana implementation Example Architechture:

### Architechture 1 :



### Architechture 2 :

#### Grafana and Elasticsearch



To Xmanager encrypt password

<https://github.com/DoubleLabyrinth/how-does-Xmanager-encrypt-password>

Xshell Xftp password decrypt

<https://github.com/dzxs/Xdecrypt>

Oracle Linux CheatSheet

#### File Commands

|            |                                      |
|------------|--------------------------------------|
| ls         | -list directory                      |
| ls -al     | -formatted listing with hidden files |
| ls -ltr    | -formatted listing ordered by time   |
| cd dir     | -change to directory                 |
| cd -       | -go to directory in previously       |
| cd         | -go to home directory                |
| pwd        | -show current directory path         |
| mkdir dir  | -make directory dir                  |
| rm file    | -delete file                         |
| rm -r dir  | -delete directory dir                |
| rm -f file | -force remove file                   |
| rm -rf dir | -force remove directory dir          |

```
cp file1 file2           -copy file1 to file2
cp -r dir1 dir2          -copy dir1 to dir2; create dir2 if it doesn't exist
mv file1 file2            -rename/move file1 to file2 if file2 is an existing dir, moves file1
into dir file2
ln -s file link          -create symbolic link link to file
touch file                -create or update file
cat > file                -places standard input into file
more file                 -output the contents of file
head file                 -output the first 10 lines of file
tail file                  -output the last 10 lines of file
tail -f file               -output the contents of file as it grows, starting with the last 10 lines
```

### Process Management

```
ps                         -display your currently active processes
top                        -display all running processes
kill pid                   -kill process id pid
killall proc                -kill all processes named proc
bg                          -lists stopped or background jobs; resume a stopped job in the background
fg                          -brings the most recent job to foreground
fg n                        -brings job n to the foreground
```

### File Permissions

```
chmod octal file          -change the permissions of file to octal, which can be found separately for user, group, and world by adding:
```

- 4 - read (r)
- 2 - write(w)
- 1 - execute (x)

### Examples:

```
chmod 777                  -read, write, execute for all
chmod 755                  -rwx for owner, rx for group and world For more options, see man chmod.
chown user:group file      -change the owner and group for a file
```

### SSH

```
ssh user@host              -connect to host as user
ssh -p port user@host       -connect to host on port port as user
ssh-copy-id user@host        -add your key to host for user to enable a keyed or passwordless login
```

### Searching

```
grep pattern files          -search for pattern in files
grep -r pattern dir          -search recursively for pattern in dir
command | grep pattern       -search for pattern in the output of command
locate file                 -find all instances of file
find . file                  -find file within current directory
find / file                  -find file within whole system
```

### System Info

```
date                        -show the current date and time
cal                          -show this month's calendar
uptime                       -show current uptime
w                            -display who is online
whoami                      -who you are logged in as
finger user                  -display information about user
uname -a                     -show kernel information
cat /proc/cpuinfo             -cpu information
cat /proc/meminfo             -memory information
man command                  -show the manual for command
df                           -show disk usage
du                           -show directory space usage
du -sk | sort -n              -show sum of each directory space usage sorted by number in KB
free                         -show memory and swap usage
whereis app                  -show possible locations of app
which app                    -show which app will be run by default
```

### Compression

```
tar cf file.tar files       - create a tar named file.tar containing files
tar xf file.tar              - extract the files from file.tar
tar czf file.tar.gz files    - create a tar with Gzip compression
tar xzf file.tar.gz          - extract a tar using Gzip
tar cjf file.tar.bz2          - create a tar with Bzip2 compression
tar xjf file.tar.bz2          - extract a tar using Bzip2
gzip file                   - compresses file and renames it to file.gz
gzip -d file.gz              - decompresses file.gz back to file
```

### Network

```
ping host                   - ping host and output results
whois domain                 - get whois information for domain
dig domain                   - get DNS information for domain
dig -x host                  - reverse lookup host
wget file                   - download file
wget -c file                 - continue a stopped download
```

### Installation

```
make install                 - install a package (Debian)
dpkg -i pkg.deb              - install a package (RPM)
rpm -Uvh pkg.rpm
```

### **Shortcuts**

|                         |                                                               |
|-------------------------|---------------------------------------------------------------|
| Ctrl+C                  | - halts the current command                                   |
| Ctrl+Z                  | - stops the current command, resume with fg in the foreground |
| or bg in the background |                                                               |
| Ctrl+D                  | - log out of current session, similar to exit                 |
| Ctrl+W                  | - erases one word in the current line                         |
| Ctrl+U                  | - erases the whole line                                       |
| Ctrl+R                  | - type to bring up a recent command                           |
| !!                      | - repeats the last command                                    |
| exit                    | - log out of current session                                  |

### **Linux - Run command as a background process**

To run a command as a background process you must append & to the end of the command when issuing:

```
$ <command> &
```

If you wish the command to continue running even when you have logged off, you need to issue nohup at the beginning of the command:

```
$ nohup <command>
```

If you wish to run the command in the background even when you are logged off you should append with nohup and &:

```
$ nohup <command> &
```

Linux - Find processes running by user

To find which processes are running by a given user on linux you can issue the following:

```
$ ps -fu <username>
```

### **GZip decompress files**

Decompress Files

To decompress files use

```
gzip -d, gunzip,
```

or

```
gzcat
```

```
gzip filename.txt
```

This will produce a compressed file called filename.txt.gz, and filename.txt will be deleted.

To decompress filename.txt.gz run the following:

```
gunzip filename.txt.gz
```

### **Find Perl Version**

```
$ perl -v
```

### **How to find Operiting System is 32 bit or 64 bit?**

1) Linux

```
# uname -m  
i386 or i686 then it is 32 bit  
x86_64 is 64bit
```

Note: Using " uname -a " (or -m for short) only shows the type of kernel that is running.

A 32 bit kernel ( i386, i586, i686 ) can be run on a 64 bit capable CPU.

A 64 bit kernel ( x86\_64 ) requires a 64 bit capable CPU to run.

```
root@localhost ~]# uname -m  
i686
```

```
[root@localhost ~]# file /usr/bin/file  
/usr/bin/file: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), for GNU/Linux 2.6.9, dynamically  
linked (uses shared libs), for GNU/Linux 2.6.9, stripped
```

```
root@localhost ~]# getconf LONG_BIT
```

```
32
```

```
root@localhost ~]# uname -ar  
Linux localhost.localdomain 2.6.18-194.el5PAE #1 SMP Mon Mar 29 20:19:03 EDT 2010 i686 i686 i386 GNU/Linux  
root@localhost ~]# file /sbin/init  
/sbin/init: ELF 32-bit LSB executable, Intel 80386, version 1 (SYSV), for GNU/Linux 2.6.9, dynamically linked  
(uses shared libs), for GNU/Linux 2.6.9, stripped
```

```
root@localhost ~]# getconf WORD_BIT  
32
```

2) Sunsolaris  
isainfo -v

## **How to Find the Version of the Oracle Solaris OS That Your System Is Running**

To see the version of Oracle Solaris software that is running on your system, type either of the following commands.

```
$ uname -a
```

The `cat` command provides more detailed information.

```
$ cat /etc/release
```

To check the sun studio version:

```
eeccsaruh2hos053 - pin> cc -V  
cc: Sun C 5.13 SunOS_sparc 2014/10/20
```

For example, to see which package contains the `cc` binary use commands like this:

#### Solaris 10

```
pkgchk -l -p /path/to/.../bin/cc
```

#### Solaris 11

```
pkg search -lp /path/to/.../bin/cc
```

#### Linux

```
rpm -qf /path/to/.../bin/cc
```

To see which Oracle Solaris Studio packages are installed on the system:

#### Solaris 10

```
pkginfo | grep SPRO
```

#### Solaris 11

```
pkg list | grep -i studio
```

#### Linux

```
rpm -qa | grep solstudio | head
```

### How to find the number of cpu ?

1) Linux

```
[root@apps u02]# cat /proc/cpuinfo |grep -i processor|wc -l  
4  
[root@apps u02]#  
  
[root@apps ~]# grep -c /proc/cpuinfo -e "CPU"  
4  
[root@apps ~]#  
top ===> press numeric 1 and observe cpu column.  
top - 08:48:31 up 6:51, 2 users, load average: 0.11, 0.03, 0.01  
Tasks: 121 total, 1 running, 119 sleeping, 0 stopped, 1 zombie  
Cpu0 : 0.3% us, 0.7% sy, 0.0% ni, 99.0% id, 0.0% wa, 0.0% hi, 0.0% si  
Cpu1 : 1.7% us, 2.4% sy, 0.0% ni, 95.9% id, 0.0% wa, 0.0% hi, 0.0% si  
Cpu2 : 3.4% us, 1.7% sy, 0.0% ni, 94.9% id, 0.0% wa, 0.0% hi, 0.0% si  
Cpu3 : 0.0% us, 0.3% sy, 0.0% ni, 99.7% id, 0.0% wa, 0.0% hi, 0.0% si  
Mem: 7225436k total, 1931064k used, 5294372k free, 94796k buffers  
Swap: 15358100k total, 0k used, 15358100k free, 1645644k cached
```

```
[root@localhost ~]# cd /sys/devices/system/cpu  
[root@localhost cpu]# ls -ltrh  
total 0  
drwxr-xr-x 4 root root 0 Jan 31 19:33 cpu3  
drwxr-xr-x 4 root root 0 Jan 31 19:33 cpu2  
drwxr-xr-x 4 root root 0 Jan 31 19:33 cpu1  
drwxr-xr-x 4 root root 0 Jan 31 19:33 cpu0  
[root@localhost cpu]#  
$ cat /proc/cpuinfo | grep "physical id" | sort | uniq | wc -l  
2
```

### To find the number of cores per CPU:

```
$ cat /proc/cpuinfo | grep "cpu cores" | uniq  
cpu cores : 4
```

The total number of processors available is the number of physical CPUs multiplied by the number of cores per CPU.

To find the total number of processors:

```
$ cat /proc/cpuinfo | grep "processor" | wc -l  
16
```

2) Sunsolaris

```
# /usr/platform/sun4u/sbin/prtdiag -v  
Uname -x  
psrinfo | grep on-line | wc -l  
6
```

#### To check the RAM size in linux.

```
dmesg | grep RAM
```

There are three another way to check memory on Linux

```
1:- free -t -m  
2: vmstat  
3: top
```

Other commands for RAM :

```
cat /proc/meminfo  
$ vmstat  
$ vmstat -s
```

```
# dmidecode --type memory
```

Sunsolaris

```
# prtconf | grep Memory  
If you want the CPU speed(s) only:  
# psrinfo -v
```

#### To find only directories in Linux/Sunsolaris?

```
[root@apps u01]# ls -ltrh|grep -i ^d  
drwxr-xr-x 4 oracle dba 4.0K Feb 7 16:19 db  
drwxr-xr-x 3 oracle dba 4.0K Feb 7 17:01 oradiag_oracle  
drwxrwxrwx 2 oracle dba 16K Feb 7 19:12 lost+found  
[root@apps u01]#
```

#### To find operating system version?

```
[root@apps etc]# ls -tlrh /etc/redhat-release  
-rw-r--r-- 1 root root 66 Nov 21 2007 /etc/redhat-release  
[root@apps etc]# cat /etc/redhat-release  
Enterprise Linux Enterprise Linux AS release 4 (October Update 6)  
[root@apps etc]#  
  
[root@apps etc]# ls -tlrh /proc/version  
-r--r--r-- 1 root root 0 Feb 12 08:58 /proc/version  
[root@apps etc]# cat /proc/version  
Linux version 2.6.9-67.0.0.1.ELhugemem (mockbuild@ca-build15.us.oracle.com) (gcc version 3.4.6 20060404 (Red Hat 3.4.6-3)) #1 SMP Sun Nov 18 00:31:12 EST 2007  
[root@apps etc]#
```

#### To find the trc files which are older than 30 days?

```
[oracle@apps trace]$ find . -name "*.trc" -mtime +30 -exec ls -ltrh {} \;  
-rw-r----- 1 oracle dba 914 Nov 8 2008 ./R1211DB1_dbrm_21512.trc  
-rw-r----- 1 oracle dba 1.3K Nov 8 2008 ./R1211DB1_vktm_21506.trc  
-rw-r----- 1 oracle dba 12K Nov 8 2008 ./R1211DB1_ora_21500.trc  
  
oracle@apps trace]$ date
```

```
Tue Feb 12 09:25:47 IST 2013  
[oracle@apps trace]$
```

#### To delete the trace files older than 30 days.?

```
find . -name "*.trc" -mtime +30 -exec rm {} \;
```

To find the files more than 200MB.?

```
[oracle@apps ~]$ find . -name "*" -size +500000 -exec ls -ltrh {} \;
```

```
-rw-r--r-- 1 oracle dba 257M Jun  6 2007 ./db/tech_st/11.1.0/md/demo/SampleData/sample_data.zip-rw-r---- 1  
oracle dba 1.4G Feb 11 23:03 ./db/apps_st/data/a_txn_data01.dbf  
-rw-r--r-- 1 oracle dba 1.5G Feb 11 23:00 ./db/apps_st/data/a_ref01.dbf  
-rw-r--r-- 1 oracle dba 1014M Feb 12 08:52 ./db/apps_st/data/system01.dbf  
-rw-r----- 1 oracle dba 1.3G Feb 11 23:00 ./db/apps_st/data/system12.dbf
```

To find the req/out/log files which are more than 10Mb.

```
find . -name "*.req" -size +10000000c -exec ls -ltrh {} \;  
find . -name "*.out" -size +10000000c -exec ls -ltrh {} \;
```

Default block size in Linux

If you ever want to confirm the block size of any filesystem of any other Linux OS.

```
[root@app ~]# tune2fs -l /dev/sda1 | grep Block  
Block count:          3839527  
Block size:           4096  
Blocks per group:    32768
```

From this example, you can see that the default block size for the filesystem on /dev/sda1 partition is 4096 bytes, or 4k. That's the default block size for ext3 filesystem.

Confirm current filesystem parameters with tune2fs

The tune2fs -l command will show you all the information contained in a filesystem's superblock. Here's how it typically looks:

```
[root@app u02]# tune2fs -l /dev/sda1  
tune2fs 1.35 (28-Feb-2004)  
Filesystem volume name:   /  
Last mounted on:         <not available>
```

Creating 0 bytes (empty) files using touch command.

syntax: touch <filename>

```
[root@app u02]# touch file1 file2 file3  
[root@app u02]# ls -ld file*  
-rw-r--r-- 1 root root 0 Feb 12 10:04 file1  
-rw-r--r-- 1 root root 0 Feb 12 10:04 file2  
-rw-r--r-- 1 root root 0 Feb 12 10:04 file3  
[root@app u02]#
```

```
[root@app u02]# date  
Tue Feb 12 10:06:28 IST 2013  
[root@app u02]#
```

To change the timestamp (date and time of modification) of a file or directory.

syntax:  
touch <options> <argument> <file or directory>

touch -t YYYYMMDDhhmm <file or directory>

Example:

```
root@app u02]# date  
Tue Feb 12 10:15:41 IST 2013  
[root@app u02]# ls -ld file*  
-rw-r--r-- 1 root root 0 Feb 12 10:04 file1  
-rw-r--r-- 1 root root 0 Feb 12 10:04 file2  
-rw-r--r-- 1 root root 0 Feb 12 10:04 file3  
[root@app u02]# touch -t 201302121016 file*  
[root@app u02]# ls -ld file*  
-rw-r--r-- 1 root root 0 Feb 12 10:16 file1  
-rw-r--r-- 1 root root 0 Feb 12 10:16 file2  
-rw-r--r-- 1 root root 0 Feb 12 10:16 file3  
[root@app u02]# date  
Tue Feb 12 10:16:53 IST 2013
```

## User Administration

### creating an user

syntax:- useradd <options> <arguments> <username>

```
-u UID          -g primary group GUID  
-c comment      -d home directory  
-G secondary group -s shell  
-o override
```

**1) creating a user khaliq with default options.**

```
root@apps u02]# useradd khaliq
to check user has been created.
[root@apps u02]# tail -f /etc/passwd
ldap:x:55:55:LDAP User:/var/lib/ldap:/bin/false
oracle:x:500:500::/u01:/bin/bash
applmgr:x:501:500::/u02:/bin/bash
ali:x:914:914:associate:/home:/bin/csh
khaliq:x:915:915:/home/prakash:/bin/bash
```

**2) create a user ali with the following parameters.**

- 1)userid 914
- 2)comment associate
- 3)home directory /home
- 4)shell cshell

```
[root@apps u02]# useradd -u 914 -c associate -d /home -s /bin/csh ali
[root@apps u02]# tail -f /etc/passwd
postgres:x:26:26:PostgreSQL Server:/var/lib/pgsql:/bin/bash
fax:x:78:78:mtgetty fax spool user:/var/spool/fax:/sbin/nologin
oracle:x:500:500::/u01:/bin/bash
applmgr:x:501:500::/u02:/bin/bash
ali:x:914:914:associate:/home:/bin/csh
```

**To modify the properties of a user.**

```
syntax:- usermod <options> <arguments> <username>
-u UID           -g primary group GUID
-c comment       -d home directory
-G secondary group -s shell
-o override     -l login name
-L lock the account -U unlock the account
```

**1) change userid of ali 750**

```
[root@apps u02]# usermod -u 750 ali
[root@apps u02]#
[root@apps u02]# grep -i ali /etc/passwd
webalizer:x:67:67:Webalizer:/var/www/usage:/sbin/nologin
ali:x:750:914:associate:/home:/bin/csh
[root@apps u02]#
```

**2)Lock the account of tom.**

```
root@apps u02]# usermod -L tom
to check logoff as root and login as tom.
```

**3)To unlock the account of tom.**

```
[root@apps u02]# usermod -U tom
to check logoff as root and login as tom.
```

**4) to change the login name of user tom to jack.**

```
[root@apps u02]# usermod -l jack tom
```

**To delete a user**

**1)delete user abcxyz.**

```
[root@apps u02]# userdel abcxyz
[root@apps home]# ls /home
prakash raj tom
[root@apps home]#
```

**2)Delete the user abc along with his home directory and mailbox.**

```
[root@apps home]# ls /home
abc xyz 123
[root@apps home]# userdel -r abc
[root@apps home]# ls /home
xyz 123
```

**Creating a group**

```
groupadd <groupname>
or
groupadd <options> <argumetn> <groupname>
```

**1)creating a group dbal whose group id 751.**

```
[root@apps home]# groupadd -g 751 dbal
Modifying the properties of a group.
```

**1)change the groupid of dbal to 800**

```
[root@apps home]# groupmod -g 800 dbal
```

**2)Changing the groupname of dbal to db.**

```
[root@apps home]# groupmod -n db dbal
Deleting the group db.
```

**1)delete the group db.**

```
[root@apps home]# groupdel db
[root@apps home]#
```

**To view how long the system is running.**

```
[root@apps home]# uptime
11:30:47 up 9:34, 3 users, load average: 1.10, 1.10, 1.07
[root@apps home]#
```

**To find out hardware details.**

```
[root@apps ~]# lspci
00:00.0 Host bridge: Intel Corporation 440BX/ZX/DX - 82443BX/ZX/DX Host bridge (rev 01)
```

```
00:01:0 PCI bridge: Intel Corporation 440BX/ZX/DX - 82443BX/ZX/DX AGP bridge (rev 01)
00:0f:0 VGA compatible controller: VMware Inc [VMware SVGA II] PCI Display Adapter
Example's of grep command
oracle@apps ~]$ cp -r db /u02 &
[1] 26764
[oracle@apps ~]$ ps -ef|grep -i cp
root      15     10  0 02:45 ?          00:00:00 [kacpid]
root    24979      1  0 02:46 ?          00:00:00 /usr/sbin/acpid
root    26128 26106  1 02:46 ?          00:00:32 /usr/X11R6/bin/X :0 -audit 0 -auth /var/gdm/:0.Xauth -nolisten
tcp vt7
```

```
[oracle@apps u02]$ ls -ltrh
total 36K
drwxr-xr-x  3 applmgr dba  4.0K Feb  7 16:19 inst
drwxr-xr-x  4 applmgr dba  4.0K Feb  7 16:19 apps
drwxrwxrwx  2 applmgr dba  16K Feb  7 19:12 lost+found
drwxr-xr-x  2 root    root  4.0K Feb 12 10:02 stage
-rw-r--r--  1 root    root   0 Feb 12 10:16 file3
```

```
[oracle@apps u02]$ rm -rf db &
[1] 26807
[oracle@apps u02]$ ps -ef|grep -i rm
root    26592      1  0 02:47 ?          00:00:21 /usr/bin/gnome-terminal
oracle   26807 26737  5 03:49 pts/2        00:00:00 rm -rf db
oracle   26809 26737  0 03:49 pts/2        00:00:00 grep -i rm
[oracle@apps u02]$
```

**you can also check whether job is running or not.**

```
[oracle@apps u02]$ jobs
[1]+  Running                  rm -rf db &
```

**To stop remove job use kill cmd.**

```
kill -9 process id
kill -9 26807
```

**To check pmon is running or not.**

```
[oracle@apps appsutil]$ ps -ef|grep -i pmon
oracle   1393 26737  0 04:11 pts/2        00:00:00 grep -i pmon
oracle   27379      1  0 03:53 ?          00:00:00 ora_pmon_PROD
[oracle@apps appsutil]$
```

**To check fnd and http processes are running or not.**

```
[applmgr@apps scripts]$ ps -fu applmgr|grep -i fnd|wc -l
12
[applmgr@apps scripts]$ ps -fu applmgr|grep -i http|wc -l
11
[applmgr@apps scripts]$
```

**To check history.**

```
[applmgr@apps scripts]$ history|grep -i fnccpass
 686 ps -fu applmgr|grep -i FNDCPASS
 687 ps -fu applmgr|grep -i FNDCPASS
 688 ps -fu applmgr|grep -i FNDCPASS
 712 history|grep -i fnccpass
[applmgr@apps scripts]$ history|grep -i adstrtal
 20 ./adstrtal.sh apps/apps
 25 ./adstrtal.sh apps/apps
```

**You can also use fc -l cmd to check history in linux or Sunsolaris.**

```
[applmgr@apps scripts]$ fc -l
699 ps -fu applmgr|grep -i http|awk '{print "kill -9",$2}'
700 kill -9 28298
701 kill -9 28303
702 kill -9 28304
703 kill -9 28309
```

Using grep command in fc -l.

```
[applmgr@apps scripts]$ fc -l|grep -i ps|head
698 ps -fu applmgr|grep -i http|wc -l
699 ps -fu applmgr|grep -i http|awk '{print "kill -9",$2}'
711 ps -fu applmgr|grep -i http|wc -l
```

**How to check if a Port is listening for any Service?Checking webserver port no in the below example.**

```
[applmgr@apps scripts]$ grep -i s_login $CONTEXT_FILE
<login_page oa_var="s_login_page">http://apps.user.com:8000/OA_HTML/AppsLogin</login_page>
```

```
[applmgr@apps scripts]$ netstat -an|grep -i 8000
tcp        0      0 0.0.0.0:8000          0.0.0.0:*                      LISTEN
[applmgr@apps scripts]$
```

**Also you can check with lsof (list of open files).**

```
[applmgr@apps scripts]$ lsof |grep -i 8000
```

```

httpd      27226 applmgr   18u      IPv4          33089                  TCP *:8000 (LI STEN)
rotatelog 27235 applmgr   10w      REG          8,2        210    11846940 /u02/inst/
apps/PROD_apps/logs/ora/10.1.3/Apache/error_log.1360800000
rotatelog 27236 applmgr   11w      REG          8,2        1056   11846827 /u02/inst/
apps/PROD_apps/logs/ora/10.1.3/Apache/access_log.1360800000
httpd      27239 applmgr   18u      IPv4          33089                  TCP *:8000 (LI STEN)
httpd      27241 applmgr   18u      IPv4          33089                  TCP *:8000 (LI STEN)
httpd      27244 applmgr   18u      IPv4          33089                  TCP *:8000 (LI STEN)
httpd      27245 applmgr   18u      IPv4          33089                  TCP *:8000 (LI STEN)
httpd      27246 applmgr   18u      IPv4          33089                  TCP *:8000 (LI STEN)
httpd      27247 applmgr   18u      IPv4          33089                  TCP *:8000 (LI STEN)
[applmgr@apps scripts]$
```

egrep example

```
[applmgr@apps ~]$ egrep -i 's_login|s_applcsf|s_applout|s_appllog|s_appltmp|s_appltmp' $CONTEXT_FILE
<login_page oa_var="s_login_page">http://apps.user.com:8000/OA_HTML/AppsLogin</login_page>
<APPLCSF oa_var="s_applcsf">/u02/inst/apps/PROD_apps/logs/appl/conc</APPLCSF>
<APPLLOG oa_var="s_appllog">log</APPLLOG>
<APPLOUT oa_var="s_applout">out</APPLOUT>
<APPLTMP oa_var="s_appltmp">/u02/inst/apps/PROD_apps/appltmp</APPLTMP>
<APPLPTMP oa_var="s_appltmp" osd="UNIX">/usr/tmp</APPLPTMP>
```

#### How can I find out if my processor is 64 bit or 32, under linux?

Look for lm (Long Mode) under flags - if you can see it, you have a 64 bit CPU, if not, it is a 32 bit one.  
[**root@apps ~]**# cat /proc/cpuinfo

OR

```
[root@apps ~]# cat /proc/cpuinfo|grep -i lm
flags          : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx
fxsr sse sse2 ss ht nx rdtscp lm pni popcnt
flags          : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx
fxsr sse sse2 ss ht nx rdtscp lm pni popcnt
flags          : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx
fxsr sse sse2 ss ht nx rdtscp lm pni popcnt
flags          : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush dts acpi mmx
fxsr sse sse2 ss ht nx rdtscp lm pni popcnt
[root@apps ~]#
```

Look for lm (Long Mode) under flags - if you can see it, you have a 64 bit CPU, if not, it is a 32 bit one.

#### How to check all mount points of a server ?

```
[oracle@devuser u02]$ df -kh
Filesystem      Size  Used Avail Use% Mounted on
/dev/sda1       15G  6.9G  6.8G  51% /
none            2.2G     0  2.2G  0% /dev/shm
/dev/sda5       4.9G 344M  4.3G  8% /tmp
/dev/sda2       39G  2.4G  35G  7% /u01
```

#### How to make tar?

i have a share mount point that is backup and you must be in oracle directory to make it tar.  
in the below example it is taking input as 11.2.0 and sending the output to backup location.

```
nohup tar -cvf - 11.2.0 | gzip -c > /backup/share/1120.tar.gz &
nohup tar -cvf - appl | gzip -c > /backup/share/appl.tar.gz &
nohup tar -cvf - 10.1.2 | gzip -c > /backup/share/1012.tar.gz &
nohup tar -cvf - 10.1.3 | gzip -c > /backup/share/1013.tar.gz &
nohup tar -cvf - comn | gzip -c > /backup/share/comn.tar.gz &
```

#### How to untar?

**you must be in oracle directory.**

```
cd /u02/oracle
under oracle directoy 11.2.0 directory will be untar.as the gunzip command taking the input from backup
location
and making under it from where you fired the command in my case oracle directory.
cd /u02/oracle
gunzip -c /backup/share/1120.tar.gz | tar -xvf - &
```

#### Yum update issue

Getting an error, while migrating to vagrant box and vagrant up

```
failure: repodata/filelists.xml.gz from ol7_latest: [Errno 256] No more mirrors to try.
```

```
http://yum.oracle.com/repo/OracleLinux/OL7/latest/x86_64/repodata/filelists.xml.gz: [Errno 14] HTTPS Error 302 - Found
```

#### Solution:

hi after googling i found this and it worked -

```
http://marc.info/?l=fedora-test-list&m=115195214028430&w=2
```

**yum clean all**

```
yum makecache  
yum update
```

## No Space Left

Error: Error writing to file /var/cache/yum/x86\_64/7Server/ol7\_developer\_EPEL/g  
en/filelists.xml: [Errno 28] No space left on device

```
ol7_developer                                628/628  
ol7_developer                                628/628  
ol7_developer_EPEL                           25585/25585  
Error: Error writing to file /var/cache/yum/x86_64/7Server/ol7_developer_EPEL/g  
en/filelists.xml: [Errno 28] No space left on device  
[root@sbl01 siebel]# df -h  
Filesystem           Size   Used  Avail Use% Mounted on  
devtmpfs             1.4G     0  1.4G  0% /dev  
tmpfs                1.4G     0  1.4G  0% /dev/shm  
tmpfs                1.4G   8.5M  1.4G  1% /run  
tmpfs                1.4G     0  1.4G  0% /sys/fs/cgroup  
/dev/mapper/vg_main-lv_root    32G   32G  7.8M 100% /  
/dev/sda1              497M  155M  343M  32% /boot  
tmpfs                275M     0  275M  0% /run/user/54322  
tmpfs                275M     0  275M  0% /run/user/0  
[root@sbl01 siebel]#
```

Solution:

# Linux Commands Cheat Sheet

Did you know that there are literally hundreds of Linux commands? Even on a bare-bones Linux server install there are easily over 1,000 different commands.

The interesting thing is that most people only need to use a very small subset of those commands. Below you'll find a Linux "cheat sheet" that breaks down some of the most commonly used commands by category. To get your own PDF and printable copy, scroll to the bottom of the page.

Enjoy!

[Contents](#) [\[hide\]](#)

- 1 – SYSTEM INFORMATION
- 2 – HARDWARE INFORMATION
- 3 – PERFORMANCE MONITORING AND STATISTICS
- 4 – USER INFORMATION AND MANAGEMENT
- 5 – FILE AND DIRECTORY COMMANDS
- 6 – PROCESS MANAGEMENT
- 7 – FILE PERMISSIONS
- 8 – NETWORKING
- 9 – ARCHIVES (TAR FILES)
- 10 – INSTALLING PACKAGES
- 11 – SEARCH
- 12 – SSH LOGINS
- 13 – FILE TRANSFERS
- 14 – DISK USAGE
- 15 – DIRECTORY NAVIGATION

## 1 – SYSTEM INFORMATION

```
# Display Linux system information  
uname -a  
  
# Display kernel release information  
uname -r
```

```

# Show which version of redhat installed
cat /etc/redhat-release

# Show how long the system has been running + load
uptime

# Show system host name
hostname

# Display the IP addresses of the host
hostname -I

# Show system reboot history
last reboot

# Show the current date and time
date

# Show this month's calendar
cal

# Display who is online
w

# Who you are logged in as
whoami

```

## 2 – HARDWARE INFORMATION

```

# Display messages in kernel ring buffer
dmesg

# Display CPU information
cat /proc/cpuinfo

# Display memory information
cat /proc/meminfo

# Display free and used memory ( -h for human readable, -m for MB, -g for GB.)
free -h

# Display PCI devices
lspci -tv

# Display USB devices
lsusb -tv

# Display DMI/SMBIOS (hardware info) from the BIOS
dmidecode

# Show info about disk sda
hdparm -i /dev/sda

# Perform a read speed test on disk sda
hdparm -tT /dev/sda

# Test for unreadable blocks on disk sda
badblocks -s /dev/sda

```

## 3 – PERFORMANCE MONITORING AND STATISTICS

```

# Display and manage the top processes
top

# Interactive process viewer (top alternative)
htop

# Display processor related statistics
mpstat 1

# Display virtual memory statistics
vmstat 1

# Display I/O statistics
iostat 1

# Display the last 100 syslog messages (Use /var/log/syslog for Debian based systems.)
tail 100 /var/log/messages

# Capture and display all packets on interface eth0

```

```
tcpdump -i eth0

# Monitor all traffic on port 80 ( HTTP )
tcpdump -i eth0 'port 80'

# List all open files on the system
lsof

# List files opened by user
lsof -u user

# Display free and used memory ( -h for human readable, -m for MB, -g for GB.)
free -h

# Execute "df -h", showing periodic updates
watch df -h



## 4 – USER INFORMATION AND MANAGEMENT



# Display the user and group ids of your current user.
id

# Display the last users who have logged onto the system.
last

# Show who is logged into the system.
who

# Show who is logged in and what they are doing.
w

# Create a group named "test".
groupadd test

# Create an account named john, with a comment of "John Smith" and create the user's home directory.
useradd -c "John Smith" -m john

# Delete the john account.
userdel john

# Add the john account to the sales group
usermod -aG sales john



## 5 – FILE AND DIRECTORY COMMANDS



# List all files in a long listing (detailed) format
ls -al

# Display the present working directory
pwd

# Create a directory
mkdir directory

# Remove (delete) file
rm file

# Remove the directory and its contents recursively
rm -r directory

# Force removal of file without prompting for confirmation
rm -f file

# Forcefully remove directory recursively
rm -rf directory

# Copy file1 to file2
cp file1 file2

# Copy source_directory recursively to destination. If destination exists, copy source_directory into destination, otherwise create destination with the contents of source_directory.
cp -r source_directory destination

# Rename or move file1 to file2. If file2 is an existing directory, move file1 into directory file2
mv file1 file2

# Create symbolic link to linkname
ln -s /path/to/file linkname

# Create an empty file or update the access and modification times of file.
```

```

touch file

# View the contents of file
cat file

# Browse through a text file
less file

# Display the first 10 lines of file
head file

# Display the last 10 lines of file
tail file

# Display the last 10 lines of file and "follow" the file as it grows.
tail -f file

```

## 6 – PROCESS MANAGEMENT

```

# Display your currently running processes
ps

# Display all the currently running processes on the system.
ps -ef

# Display process information for processname
ps -ef | grep processname

# Display and manage the top processes
top

# Interactive process viewer (top alternative)
htop

# Kill process with process ID of pid
kill pid

# Kill all processes named processname
killall processname

# Start program in the background
program &

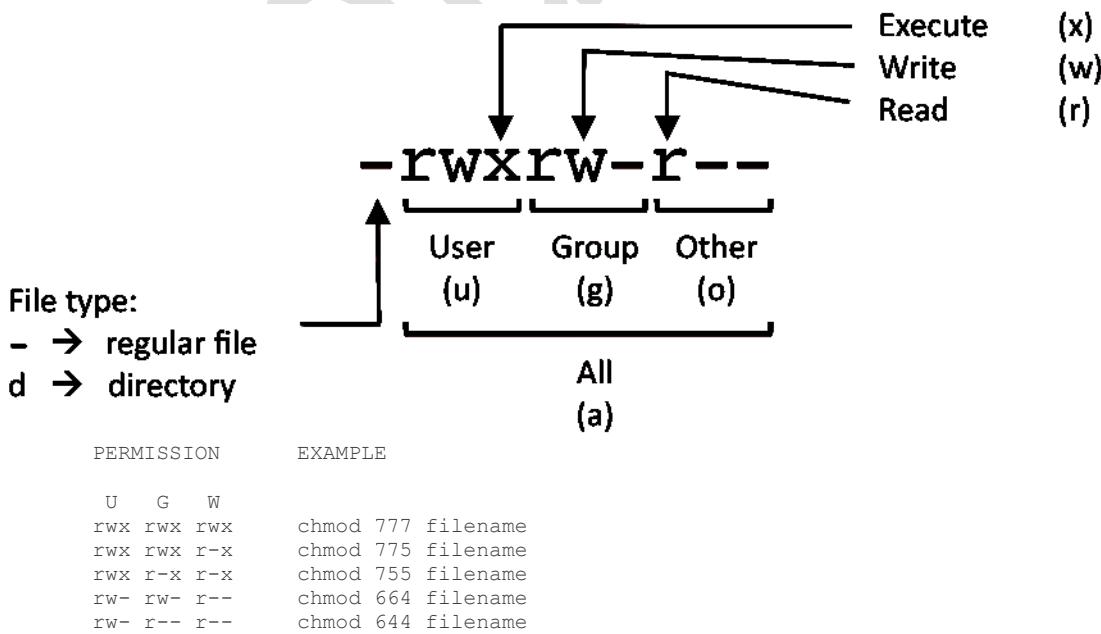
# Display stopped or background jobs
bg

# Brings the most recent background job to foreground
fg

# Brings job n to the foreground
fg n

```

## 7 – FILE PERMISSIONS



```
# NOTE: Use 777 sparingly!

LEGEND
U = User
G = Group
W = World

r = Read
w = write
x = execute
- = no access
```

**8 – NETWORKING**

```
# Display all network interfaces and ip address
ifconfig -a

# Display eth0 address and details
ifconfig eth0

# Query or control network driver and hardware settings
ethtool eth0

# Send ICMP echo request to host
ping host

# Display whois information for domain
whois domain

# Display DNS information for domain
dig domain

# Reverse lookup of IP_ADDRESS
dig -x IP_ADDRESS

# Display DNS ip address for domain
host domain

# Display the network address of the host name.
hostname -i

# Display all local ip addresses
hostname -I

# Download http://domain.com/file
wget http://domain.com/file

# Display listening tcp and udp ports and corresponding programs
netstat -nutlp
```

## 9 – ARCHIVES (TAR FILES)

```
# Create tar named archive.tar containing directory.
tar cf archive.tar directory

# Extract the contents from archive.tar.
tar xf archive.tar

# Create a gzip compressed tar file name archive.tar.gz.
tar czf archive.tar.gz directory

# Extract a gzip compressed tar file.
tar xzf archive.tar.gz

# Create a tar file with bzip2 compression
tar cjf archive.tar.bz2 directory

# Extract a bzip2 compressed tar file.
tar xjf archive.tar.bz2
```

## 10 – INSTALLING PACKAGES

```
# Search for a package by keyword.
yum search keyword

# Install package.
yum install package

# Display description and summary information about package.
```

```
yum info package  
# Install package from local file named package.rpm  
rpm -i package.rpm
```

```
# Remove/uninstall package.  
yum remove package
```

```
# Install software from source code.  
tar zxvf sourcecode.tar.gz  
cd sourcecode  
.configure  
make  
make install
```

## 11 – SEARCH

```
# Search for pattern in file  
grep pattern file
```

```
# Search recursively for pattern in directory  
grep -r pattern directory
```

```
# Find files and directories by name  
locate name
```

```
# Find files in /home/john that start with "prefix".  
find /home/john -name 'prefix*'
```

```
# Find files larger than 100MB in /home  
find /home -size +100M
```

## 12 – SSH LOGINS

```
# Connect to host as your local username.  
ssh host
```

```
# Connect to host as user  
ssh user@host
```

```
# Connect to host using port  
ssh -p port user@host
```

## 13 – FILE TRANSFERS

```
# Secure copy file.txt to the /tmp folder on server  
scp file.txt server:/tmp
```

```
# Copy *.html files from server to the local /tmp folder.  
scp server:/var/www/*.html /tmp
```

```
# Copy all files and directories recursively from server to the current system's /tmp folder.  
scp -r server:/var/www /tmp
```

```
# Synchronize /home to /backups/home  
rsync -a /home /backups/
```

```
# Synchronize files/directories between the local and remote system with compression enabled  
rsync -avz /home server:/backups/
```

## 14 – DISK USAGE

```
# Show free and used space on mounted filesystems  
df -h
```

```
# Show free and used inodes on mounted filesystems  
df -i
```

```
# Display disks partitions sizes and types  
fdisk -l
```

```
# Display disk usage for all files and directories in human readable format  
du -ah
```

```
# Display total disk usage off the current directory  
du -sh
```

## 15 – DIRECTORY NAVIGATION

```
# To go up one level of the directory tree. (Change into the parent directory.)  
cd ..
```

```
# Go to the $HOME directory
```

```

cd
# Change to the /etc directory
cd /etc

```

# Linux ip Command Networking Cheat Sheet

## IP QUERIES

| IP QUERIES   |                                                                                                                                                                                                                                                                                                                        |
|--------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SUBCOMMAND   | DESCRIPTIONS AND TASKS                                                                                                                                                                                                                                                                                                 |
| <b>addr</b>  | Display IP Addresses and property information (abbreviation of address)<br><b>ip addr</b><br>Show information for all addresses<br><b>ip addr show dev em1</b><br>Display information only for device em1                                                                                                              |
| <b>link</b>  | Manage and display the state of all network interfaces<br><b>ip link</b><br>Show information for all interfaces<br><b>ip link show dev em1</b><br>Display information only for device em1<br><b>ip -s link</b><br>Display interface statistics                                                                         |
| <b>route</b> | Display and alter the routing table<br><b>ip route</b><br>List all of the route entries in the kernel                                                                                                                                                                                                                  |
| <b>maddr</b> | Manage and display multicast IP addresses<br><b>ip maddr</b><br>Display multicast information for all devices<br><b>ip maddr show dev em1</b><br>Display multicast information for device em1                                                                                                                          |
| <b>neigh</b> | Show neighbour objects; also known as the ARP table for IPv4<br><b>ip neigh</b><br>Display neighbour objects<br><b>ip neigh show dev em1</b><br>Show the ARP cache for device em1                                                                                                                                      |
| <b>help</b>  | Display a list of commands and arguments for each subcommand<br><b>ip help</b><br>Display ip commands and arguments<br><b>ip addr help</b><br>Display address commands and arguments<br><b>ip link help</b><br>Display link commands and arguments<br><b>ip neigh help</b><br>Display neighbour commands and arguments |

## MULTICAST ADDRESSING

| MULTICAST ADDRESSING |                                                                                                                                              |
|----------------------|----------------------------------------------------------------------------------------------------------------------------------------------|
| SUBCOMMAND           | DESCRIPTIONS AND TASKS                                                                                                                       |
| <b>maddr add</b>     | Add a static link-layer multicast address<br><b>ip maddr add 33:33:00:00:00:01 dev em1</b><br>Add multicast address 33:33:00:00:00:01 to em1 |
| <b>maddr del</b>     | Delete a multicast address<br><b>ip maddr del 33:33:00:00:00:01 dev em1</b><br>Delete address 33:33:00:00:00:01 from em1                     |

## MODIFYING ADDRESS AND LINK PROPERTIES

| MODIFYING ADDRESS AND LINK PROPERTIES |                                                                                                                      |
|---------------------------------------|----------------------------------------------------------------------------------------------------------------------|
| SUBCOMMAND                            | DESCRIPTIONS AND TASKS                                                                                               |
| <b>addr add</b>                       | Add an address<br><b>ip addr add 192.168.1.1/24 dev em1</b><br>Add address 192.168.1.1 with netmask 24 to device em1 |
| <b>addr del</b>                       | Delete an address<br><b>ip addr del 192.168.1.1/24 dev em1</b><br>Remove address 192.168.1.1/24 from device em1      |
| <b>link set</b>                       | Alter the status of the interface<br><b>ip link set em1 up</b><br>Bring em1 online<br><b>ip link set em1 down</b>    |

|  |                                                                                                                                                            |
|--|------------------------------------------------------------------------------------------------------------------------------------------------------------|
|  | Bring em1 offline<br><b>ip link set em1 mtu 9000</b><br>Set the MTU on em1 to 9000<br><b>ip link set em1 promisc on</b><br>Enable promiscuous mode for em1 |
|--|------------------------------------------------------------------------------------------------------------------------------------------------------------|

#### ADJUSTING AND VIEWING ROUTES

| SUBCOMMAND           | DESCRIPTIONS AND TASKS                                                                                                                                                                                                                                                                                                                                                                                                                         |
|----------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>route add</b>     | Add an entry to the routing table<br><b>ip route add default via 192.168.1.1 dev em1</b><br>Add a default route (for all addresses) via the local gateway 192.168.1.1 that can be reached on device em1<br><b>ip route add 192.168.1.0/24 via 192.168.1.1</b><br>Add a route to 192.168.1.0/24 via the gateway at 192.168.1.1<br><b>ip route add 192.168.1.0/24 dev em1</b><br>Add a route to 192.168.1.0/24 that can be reached on device em1 |
| <b>route delete</b>  | Delete a routing table entry<br><b>ip route delete 192.168.1.0/24 via 192.168.1.1</b><br>Delete the route for 192.168.1.0/24 via the gateway at 192.168.1.1                                                                                                                                                                                                                                                                                    |
| <b>route replace</b> | Replace, or add if not defined, a route<br><b>ip route replace 192.168.1.0/24 dev em1</b><br>Replace the defined route for 192.168.1.0/24 to use device em1                                                                                                                                                                                                                                                                                    |
| <b>route get</b>     | Display the route an address will take<br><b>ip route get 192.168.1.5</b><br>Display the route taken for IP 192.168.1.5                                                                                                                                                                                                                                                                                                                        |

#### MANAGING THE ARP TABLE

| SUBCOMMAND           | DESCRIPTIONS AND TASKS                                                                                                                                                                                  |
|----------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>neigh add</b>     | Add an entry to the ARP Table<br><b>ip neigh add 192.168.1.1 lladdr 1:2:3:4:5:6 dev em1</b><br>Add address 192.168.1.1 with MAC 1:2:3:4:5:6 to em1                                                      |
| <b>neigh del</b>     | Invalidate an entry<br><b>ip neigh del 192.168.1.1 dev em1</b><br>Invalidate the entry for 192.168.1.1 on em1                                                                                           |
| <b>neigh replace</b> | Replace, or adds if not defined, an entry to the ARP table<br><b>ip neigh replace 192.168.1.1 lladdr 1:2:3:4:5:6 dev em1</b><br>Replace the entry for address 192.168.1.1 to use MAC 1:2:3:4:5:6 on em1 |

#### USEFUL NETWORKING COMMANDS (NOT NECESSARILY PROVIDED FROM IPRROUTE)

| SUBCOMMAND     | DESCRIPTIONS AND TASKS                                                                                                                                                                                                                                                                                                                                                   |
|----------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>arping</b>  | Send ARP request to a neighbour host<br><b>arping -l eth0 192.168.1.1</b><br>Send ARP request to 192.168.1.1 via interface eth0<br><b>arping -D -l eth0 192.168.1.1</b><br>Check for duplicate MAC addresses at 192.168.1.1 on eth0                                                                                                                                      |
| <b>ethtool</b> | Query or control network driver and hardware settings<br><b>ethtool -g eth0</b><br>Display ring buffer for eth0<br><b>ethtool -i eth0</b><br>Display driver information for eth0<br><b>ethtool -p eth0</b><br>Identify eth0 by sight, typically by causing LEDs to blink on the network port<br><b>ethtool -S eth0</b><br>Display network and driver statistics for eth0 |
| <b>ss</b>      | Display socket statistics. The below options can be combined<br><b>ss -a</b><br>Show all sockets (listening and non-listening)<br><b>ss -e</b><br>Show detailed socket information<br><b>ss -o</b><br>Show timer information<br><b>ss -n</b><br>Do not resolve addresses<br><b>ss -p</b><br>Show process using the socket                                                |

#### COMPARING NET-TOOLS VS. IPRROUTE PACKAGE COMMANDS

| COMPARING NET-TOOLS VS. IPROUTE PACKAGE COMMANDS          |                                                      |
|-----------------------------------------------------------|------------------------------------------------------|
| NET-TOOLS COMMANDS                                        | IPROUTE COMMANDS                                     |
| arp -a                                                    | ip neigh                                             |
| arp -v                                                    | ip -s neigh                                          |
| arp -s 192.168.1.1 1:2:3:4:5:6                            | ip neigh add 192.168.1.1 lladdr 1:2:3:4:5:6 dev eth1 |
| arp -i eth1 -d 192.168.1.1                                | ip neigh del 192.168.1.1 dev eth1                    |
| ifconfig -a                                               | ip addr                                              |
| ifconfig eth0 down                                        | ip link set eth0 down                                |
| ifconfig eth0 up                                          | ip link set eth0 up                                  |
| ifconfig eth0 192.168.1.1                                 | ip addr add 192.168.1.1/24 dev eth0                  |
| ifconfig eth0 netmask 255.255.255.0                       | ip addr add 192.168.1.1/24 dev eth0                  |
| ifconfig eth0 mtu 9000                                    | ip link set eth0 mtu 9000                            |
| ifconfig eth0:0 192.168.1.2                               | ip addr add 192.168.1.2/24 dev eth0                  |
| netstat                                                   | ss                                                   |
| netstat -neopa                                            | ss -neopa                                            |
| netstat -g                                                | ip maddr                                             |
| route                                                     | ip route                                             |
| route add -net 192.168.1.0 netmask 255.255.255.0 dev eth0 | ip route add 192.168.1.0/24 dev eth0                 |
| route add default gw 192.168.1.1                          | ip route add default via 192.168.1.1                 |

## Linux System Administrator Projects

If you are looking to gain some experience as a Linux system administrator, check out the list of ideas below. For each project on the list you could do it on multiple Linux distributions. You could also start about by learning the bare minimum necessary to complete one of these projects, then take it to the next level by making it fault tolerant or learning how to scale it. Create a virtual machine for your projects or try them on your own server hosted at Digital Ocean.

Contents [hide]

- Scratch Your Own Itch
- Configure the LAMP Stack
- Deploy an Open Source Application
- Configure Common Services – Client AND Server
- Configure Monitoring
- Create a Build System
- Create a Centralized Syslog Server
- System Automation
- Cluster All The Things
- Build a NAS
- Practice Migrating Data
- Create and Manage Users
- Configure a Backup Server
- Configure a Firewall
- Learn LVM
- Configure a Proxy Server
- Learn Revision Control

## Scratch Your Own Itch

Find a real need you have and fill it using a Linux based solution. This is one of the most rewarding ways to learn Linux. For example,

- Do you need somewhere to store work related documentation or notes? Then install a Wiki on Linux.
  - Want to start a blog? Host it on Linux!
  - Want to stream online content to your TV or other devices in your home? Then build a Linux media server.
- The next time you have a technical need and want to get some extra Linux practice, see if you can solve your problem with Linux.

## Configure the LAMP Stack

LAMP stands for Linux Apache MySQL and PHP. It's a very common architecture for web applications. If you think about the LAMP stack in broad terms you have an operating system (Linux), a web server (Apache), a database tier (MySQL), and a programming language (PHP). You can substitute different solutions for each component. Try configuring the LAMP stack on CentOS, then Ubuntu. Instead of using Apache you could use NGINX or lighttpd. For the database you could use PostgreSQL, MariaDB, or MongoDB. Instead of using PHP, why not try Python?

If you want to take it a step further and learn how to set up a highly available and scalable LAMP Stack, check out the High Availability for the LAMP Stack Course.

- Linux
- Apache
- MySQL
- PHP

## Deploy an Open Source Application

Find an application that you would be interested in using and deploy it. Even if you don't end up using the application you'll have gained the experience of setting it up.

- Alfresco
- Bugzilla
- DokuWiki
- Drupal
- Etherpad Lite
- Gitlab
- Joomla
- MediaWiki
- Moodle
- osTicket
- OwnCloud
- phpBB
- PunBB
- Redmine
- ServiceDesk Plus
- SugarCRM
- Trac
- Twiki
- WordPress
- Zen Cart

## Configure Common Services – Client AND Server

- cron
- CUPS
- DHCP
- DNS
- Email (SMTP, POP, IMAP)

- LDAP
- NFS
- NIS
- NTP
- SSH

## Configure Monitoring

- Cacti
- Icinga
- Monit
- Munin
- Nagios
- OpenNMS
- Zabbix
- Zenoss

## Create a Build System

- Cobbler
- FAI
- Foreman
- Kickstart
- Razor
- Spacewalk

## Create a Centralized Syslog Server

- ELK Stack: Elasticsearch, Logstash & Kibana
- Fluentd
- Logstash
- Kibana
- Splunk
- syslog-ng

## System Automation

- Ansible
- Chef
- MakeFiles and/or RakeFiles
- Puppet
- Salt Stack
- Write Shell Scripts

## Cluster All The Things

- Load balance web servers or other services using HAProxy
- Create a MySQL Cluster
- Create a GlusterFS Cluster
- Red Hat Cluster (Conga)

## Build a NAS

- NFS
- Samba

## Practice Migrating Data

- Migrate data from a disk on one server to a disk on another server.
- Migrate Databases.
  - Live migrations
  - Export / Import

## Create and Manage Users

- FreeIPA
- LDAP
- NIS
- User name ideas: <http://www.fakenamegenerator.com/>

## Configure a Backup Server

- Amanda
- Bacula
- Rsnapshot
- Rsync + SSH

## Configure a Firewall

- IPtables
- UFW

## Learn LVM

- Create and restore snapshots
- Extend volumes without downtime

## Configure a Proxy Server

- Forward Proxy
  - Apache
  - Squid
- Reverse Proxy
  - Apache
  - NGINX
  - Pound

## Learn Revision Control

- CVS
- Git
- RCS

If you are still looking for more ideas, check out Linux Server Hacks and Linux Server Hacks Volume Two. Simply looking at the table of contents will give you ideas for even more projects to tackle

#### Installing Portal BRM on RHE 5

I'm assuming that you have a fresh RHE 5 installed on a 64bit VM machine / actual machine, I preferred using a VMWare player image since I wanted to be able to snapshot and role back whenever I wanted to.

Start by knowing what you have on, by typing command

**\$spinrev**

this should yield all the portal products installed on your system, in this scenario I assume that nothing is installed on the system.

the next step should be installing Oracle DB 10g, reduce the SGA size to limit impact on your machine performance.

**NLS\_lang should be set to american\_america.UTF8**, if you didnt set correctly you'll get an ORA12705 error.

Copy the sources to /temp

Create user pin

**Start by running ThirdParty\*.bin as pin**

at this point you need to run **.source.me.sh** it can be found under **"/opt/portal/ThirdParty/"**

Make sure to export the java bin to path if not included in the .bashrc

The java home should be the one in the **"./opt/portal/Thirdparty/Jre/1.5"**

Install SNMP third party tool as mentioned in the official installation guide.

Source the source.me.sh file from thirdparty folder.

**\$source source.me.sh**

Create db user pin and grant it **resource/connect/DBMS\_lock/create\_session**

run **7\*Portal\_Base\_\*\_opt.bin** to install portal base.

Use default ports

**DM\_Port 12950**

**CM\_Port 11960**

I chose to enable partitioning event files to make sure to get real life like system.

After installing it you have to run

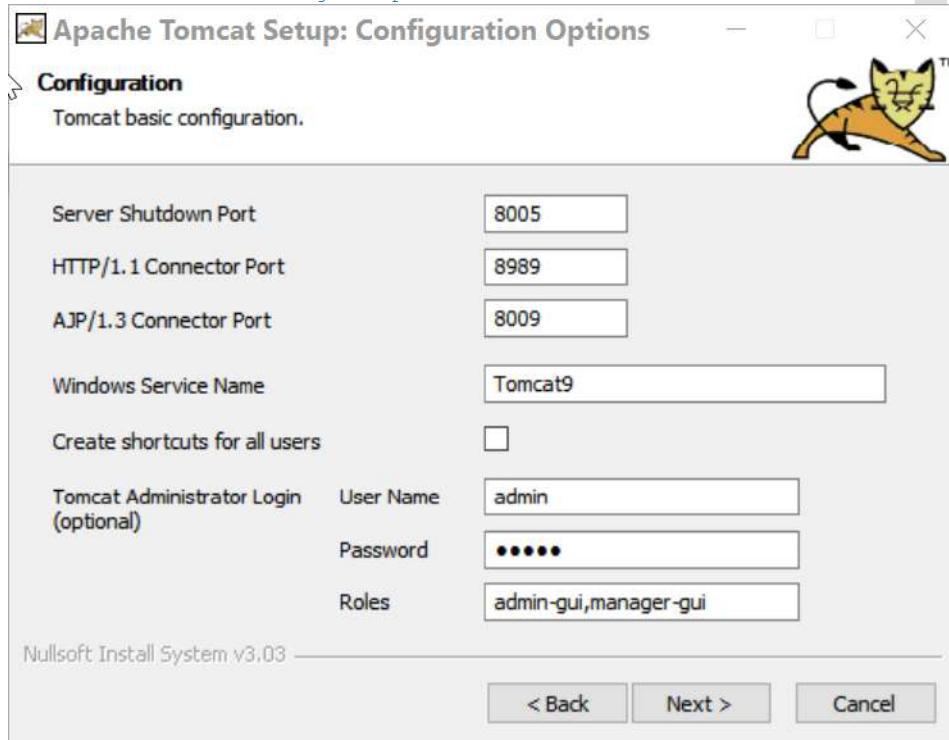
**source /opt/portal/7.3/source.me.sh**

**pin\_setup**

at which point I received an error stating that line 7 of code had an issue, just vi the file and add a space after the [ and it'll work, this is due to the differences between bash and ksh.

settings can be found under /opt/portal/7.3/setup/pin\_setup.values

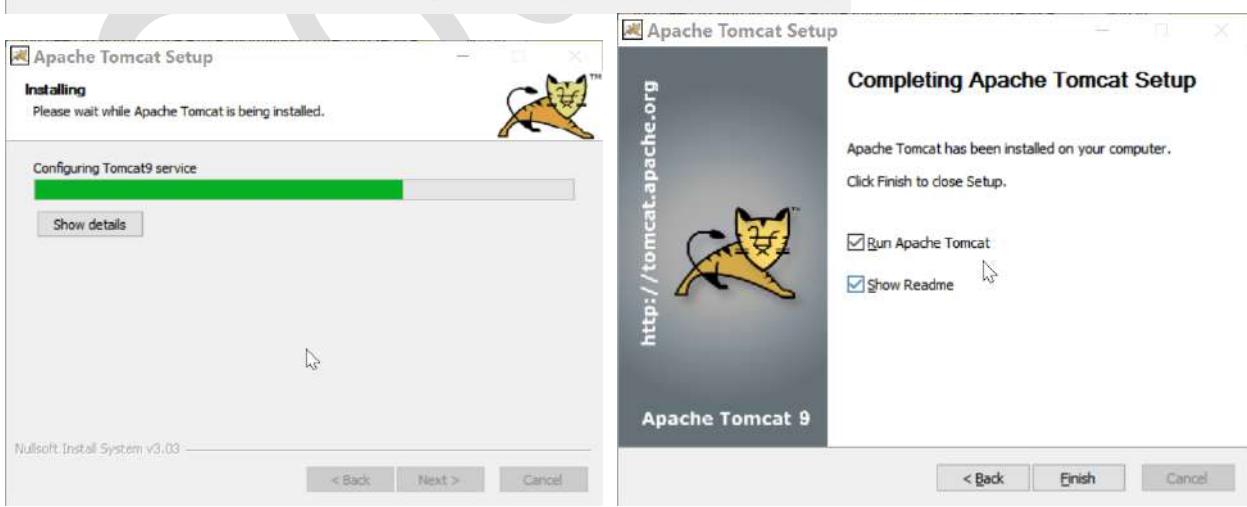
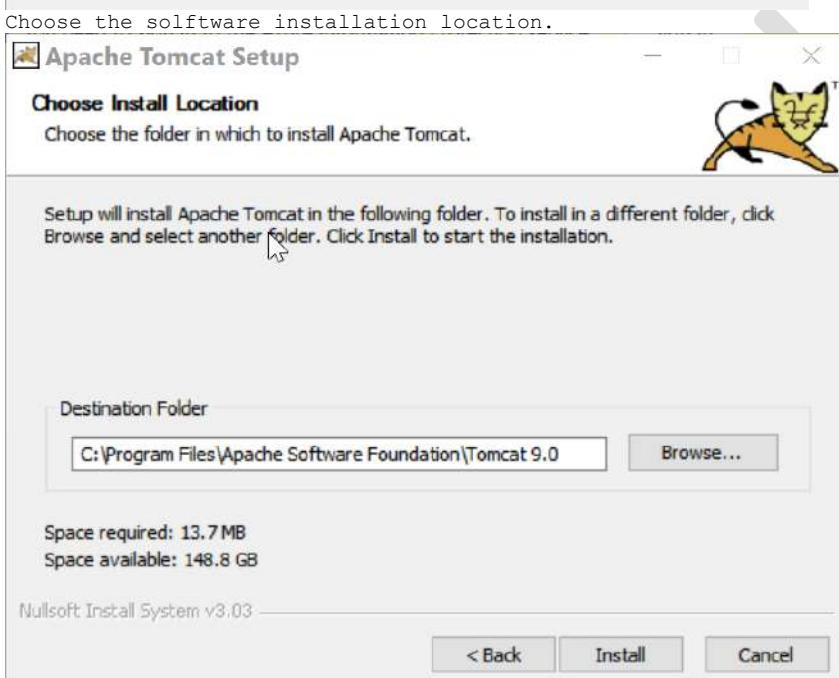
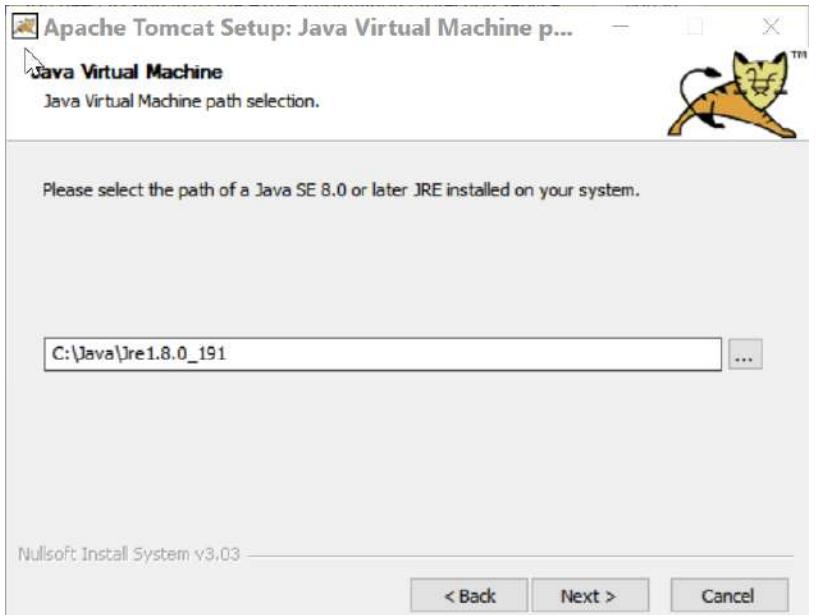
#### How to install and configure apache tomcat as service.



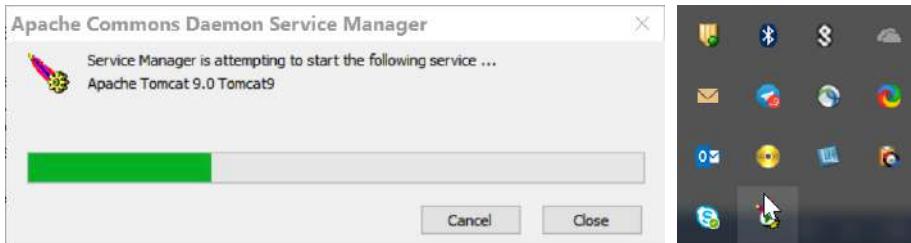
Change the HTTP port with 8989 instead of 8080

And give the administration login credentials, if you want or else leave as is.

Select the JRE

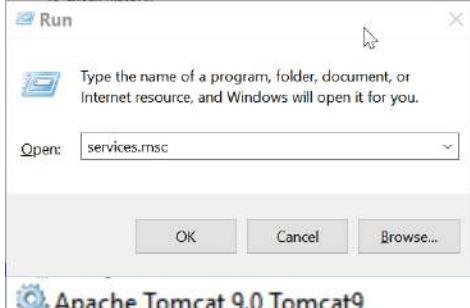


Installation completed.  
Now the service will start automatically, if you choose the "Run Apache tomcat" while installation done. As per above snapshot.



Now cross check the tomcat service in status bar as well as in services.

Check the from services.msc as well. As below its showing as in running state.



Now open Eclipse. And create a web develop project.

Clean steps provided in this link: <http://www.srccodes.com/p/article/2/JSP-Hello-World-Program-using-Eclipse-IDE-and-Tomcat-web-server>

#### Adding the Apache Tomcat runtimes

<https://o7planning.org/en/10285/create-a-simple-java-web-application-using-servlet-jsp-and-jdbc>

<https://o7planning.org/en/10209/installing-and-configuring-tomcat-server-in-eclipse>

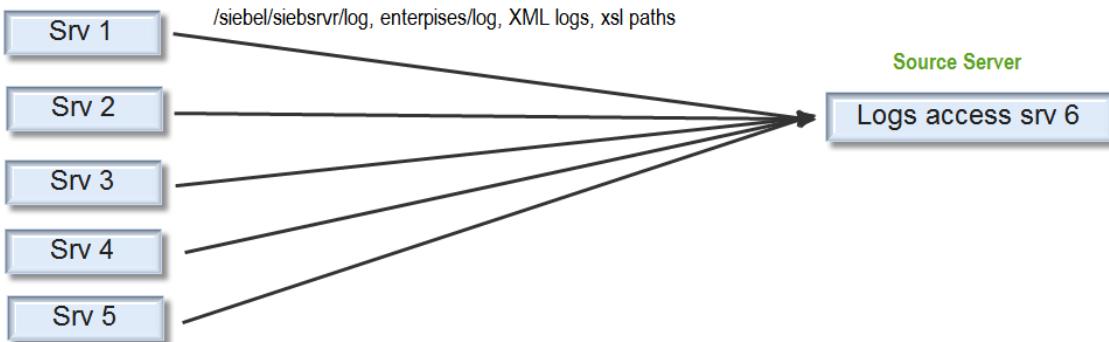
#### Centralized logging system Architecture

Make the sure the mounting points will be confirmed by UNIX team and this mounting point should not be disconnected.

1. Collect all directories from destination servers.
2. Mount it with NFS/other types of mounting methods on source with destination server.
3. It should have read write execute access on these mounting points
4. These mounting points should not be disconnected, when if the servers rebooted.
5. Install any webserver like tomcat, IBM http server, WebLogic etc.
6. Create a virtual directory
7. Create an html template page to access link files in that virtual directory.
8. **As discussed with unix team they are not providing support for mounting due to security concerns , we will create a 100G FS on sda022 as /siebellogs and enable ssh key based login from source to target host.**
9. Ssh key will be enabled in between systems.
10. Create a filesystem with the help of storage team as storage space with 100G
11. And create a cron job to place the files on /siebellogs filesystem.

## Centralize logging system

### Destination Server paths



### Source Server

| Source Host Names | Paths to Be Mounted                                                                                                                                         | Mount on destination server sda022                                                                                                                                                                                                        | Destination Host |
|-------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------|
| Sda011            | /siebel/siebsrvr/log<br>/siebel/siebsrvfenterprises/ETPPRD/srv_prd2/log<br>/home/java/BSL/xsl /home/java/EECC_EAI_API/DEB4<br>/home/java/EECC_EAI_API/DEBUG | /siebel/siebsrvr/log : sda011_srvr_log<br>/siebel/siebsrvfenterprises/ETPPRD/srv_prd2/log : sda011_ent_log<br>/home/java/BSL/xsl : sda011_xsl<br>/home/java/EECC_EAI_API/DEB4 : sda011_xmls<br>/home/java/EECC_EAI_API/DEBUG : sda011_pay | sda022           |
| Sda012            | /siebel/siebsrvr/log<br>/siebel/siebsrvfenterprises/ETPPRD/srv_prd2/log<br>/home/java/BSL/xsl /home/java/EECC_EAI_API/DEB4<br>/home/java/EECC_EAI_API/DEBUG | /siebel/siebsrvr/log : sda012_srvr_log<br>/siebel/siebsrvfenterprises/ETPPRD/srv_prd2/log : sda012_ent_log<br>/home/java/BSL/xsl : sda012_xsl<br>/home/java/EECC_EAI_API/DEB4 : sda012_xmls<br>/home/java/EECC_EAI_API/DEBUG : sda012_pay | sda022           |
| Sda014            | /siebel/siebsrvr/log<br>/siebel/siebsrvfenterprises/ETPPRD/srv_prd2/log<br>/home/java/BSL/xsl /home/java/EECC_EAI_API/DEB4<br>/home/java/EECC_EAI_API/DEBUG | /siebel/siebsrvr/log : sda014_srvr_log<br>/siebel/siebsrvfenterprises/ETPPRD/srv_prd2/log : sda014_ent_log<br>/home/java/BSL/xsl : sda014_xsl<br>/home/java/EECC_EAI_API/DEB4 : sda014_xmls<br>/home/java/EECC_EAI_API/DEBUG : sda014_pay | sda022           |
| Sda017            | /siebel/siebsrvr/log<br>/siebel/siebsrvfenterprises/ETPPRD/srv_prd2/log<br>/home/java/BSL/xsl /home/java/EECC_EAI_API/DEB4<br>/home/java/EECC_EAI_API/DEBUG | /siebel/siebsrvr/log : sda017_srvr_log<br>/siebel/siebsrvfenterprises/ETPPRD/srv_prd2/log : sda017_ent_log<br>/home/java/BSL/xsl : sda017_xsl<br>/home/java/EECC_EAI_API/DEB4 : sda017_xmls<br>/home/java/EECC_EAI_API/DEBUG : sda017_pay | sda022           |
| Sda039            | /siebel/siebsrvr/log<br>/siebel/siebsrvfenterprises/ETPPRD/srv_prd2/log<br>/home/java/BSL/xsl /home/java/EECC_EAI_API/DEB4<br>/home/java/EECC_EAI_API/DEBUG | /siebel/siebsrvr/log : sda039_srvr_log<br>/siebel/siebsrvfenterprises/ETPPRD/srv_prd2/log : sda039_ent_log<br>/home/java/BSL/xsl : sda039_xsl<br>/home/java/EECC_EAI_API/DEB4 : sda039_xmls<br>/home/java/EECC_EAI_API/DEBUG : sda039_pay | sda022           |

Create filesystem /sieberlogs

Create file system /sieberlogs on sda022 host. And then assign 100gb storage space to the same filesystem.

Hi Rami,

Please approve the below lun request for creating /sieberlogs file system.

Dear Storage team,

Please allocate the below lun request upon approval from Rami..

Hostname : sda022

Lun : 1 \* 100 GB

Below are the wwpn details ::

```
# hostname
suvio1
# pcmpath query wwpn
Adapter Name PortWWN
fscsi0 10000000C9AB3F52
fscsi1 10000000C9AB3F53
fscsi2 10000000C9AB3F32
fscsi3 10000000C9AB3F33
fscsi4 10000000C9AB64F0
fscsi5 10000000C9AB64F1
```

```
# hostname
suvio2
# pcmpath query wwpn
Adapter Name PortWWN
fscsi0 10000000C9AB3F2C
```

```
fscsil      10000000C9AB3F2D
fscsi2      10000000C9AB8138
fscsi3      10000000C9AB8139
fscsi4      10000000C9AB6696
fscsi5      10000000C9AB6697
```

100G storage added by storage team

| Name              | UID                               | Capacity   |
|-------------------|-----------------------------------|------------|
| SUVI01_02_Vol_057 | 600507680C80036898000000000000BCD | 100.00 GiB |

Scp commands:

```
scp xyz.txt siebel@sda022:/home/siebel
scp $(find . -mtime -1d) siebel@sda022:/siebellogs/sda011_xmls
```

Create directories in sda022

```
[siebel@sda022]/siebellogs>
```

```
mkdir sda011_srvr_log sda011_ent_log sda011_xsl sda011_xmls sda011_pay
mkdir sda014_srvr_log sda014_ent_log sda014_xsl sda014_xmls sda014_pay
mkdir sda017_srvr_log sda017_ent_log sda017_xsl sda017_xmls sda017_pay
mkdir sda039_srvr_log sda039_ent_log sda039_xsl sda039_xmls sda039_pay
```

Install tomcat on /siebellogs filesystem

```
set the support java jre path
```

```
export JAVA_HOME=/usr/java7_64
```

```
start tomcat
```

create a virtual directory in tomcat

before creating virtual directories, please edit the web.xml listing value to TRUE as follows.

```
Cd /siebel/tomcat/apache-tomcat-8.0.33/conf
```

**Web.xml**

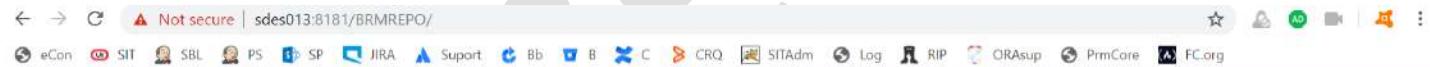
```
<param-name>listings</param-name>
<param-value>false</param-value> → <param-value>true</param-value>
```

```
cd /siebellogs/ApacheTomcat8/webapps
```

```
mkdir SBLPP
```

```
and start the tomcat with the command ./startup.sh
```

and now open the brower and confirm the the application is working or not.



## Directory Listing For [/]

| Filename     | Size   | Last Modified                 |
|--------------|--------|-------------------------------|
| textfile.txt | 0.0 kb | Thu, 28 Nov 2019 11:51:32 GMT |

## Apache Tomcat/7.0.96

Softlink or symbolic links creation:

```
ln -s <source> <destination>
```

```
ln -s /export/home/bradmin/.m2/repository /export/home/bradmin/tomcat7/webapps/BRMREPO
```

Cgi web page to execute unix commands:

<https://www.cyberciti.biz/tips/executing-linuxunix-commands-from-web-page-part-i.html>

CGI enabling to tomcat:

We have to configure to files, which contains in conf directory.

<https://dzone.com/articles/enable-cgi-mode-in-tomcat>

```
web.xml
context.xml
```



apache-tomcat-8.0.3  
3.tar

### My Script

Command Line - \$ csearch "argument"

```
#!/bin/sh
grep -ir -B 1 -A 4 "$*" /usr/local/var/rancid/CiscoDevices/configs
```

### My HTML page

```
<html>
<body>
<form method="POST" action="csearch.php">
    <input type="text" name="searchText">
    <input type="submit" value="Search">
</form>
</body>
</html>
```

### My PHP page

```
<?php
$searchText=$_POST['$searchText'];
?>

<html>
<?php

$output = shell_exec('/usr/local/bin/csearch $searchText');
echo "<pre>$output</pre>";

?>
</html>
```

Any help is greatly appreciated.

shell\_exec('/usr/local/bin/csearch \$searchText'); This isn't what you expect it to be:  
<?php

```
$searchText = 'foobar';
$cmd = '/usr/local/bin/csearch $searchText';
echo $cmd;

?>
outputs:
```

/usr/local/bin/csearch \$searchText

Change the string to use double quotes and \$searchText will actually be what you want to it be:

```
$output = shell_exec("/usr/local/bin/csearch $searchText");
More info on the use of quotes in PHP.
```

As @uri2x hints in a comment:

\$searchText=\$\_POST['\$searchText']; should be changed to \$searchText=\$\_POST['searchText']; for a similar reason.

## Tomcat TTILE change in Windows.

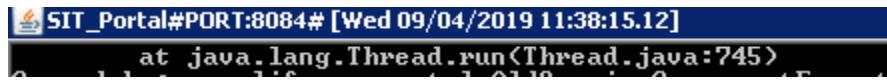
Requirement : need to show the title name as SIT or DEV project which were in same machine [Tafeely]

So I need to show the tomcat sit name instead of default name "Tomcat"

From



To



Go to tomcat bin directory and edit the batch file of catalina.bat and add the below line and save it.

```
set TITLE=Tomcat.Cluster#1.Server#1 [%DATE% %TIME%]
```

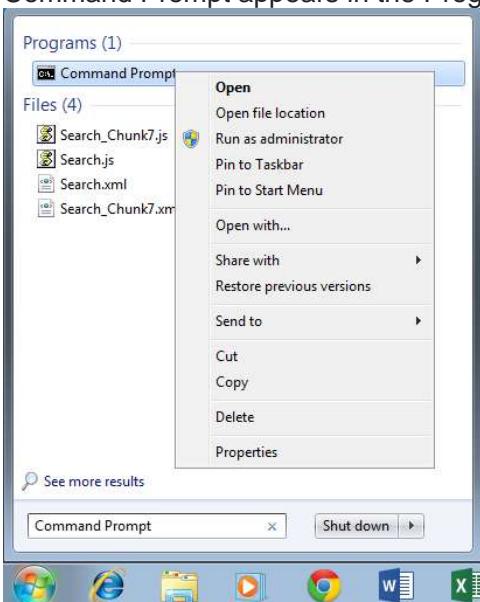
## Install Tomcat as a Windows Service

1. APM Connect uses Tomcat Server as the web server for the APM Connect Administration Center web application. This topic describes how to start Tomcat as a service.
2. Before You Begin
3. Before you install Tomcat, you must complete the following:
4. Install Java.

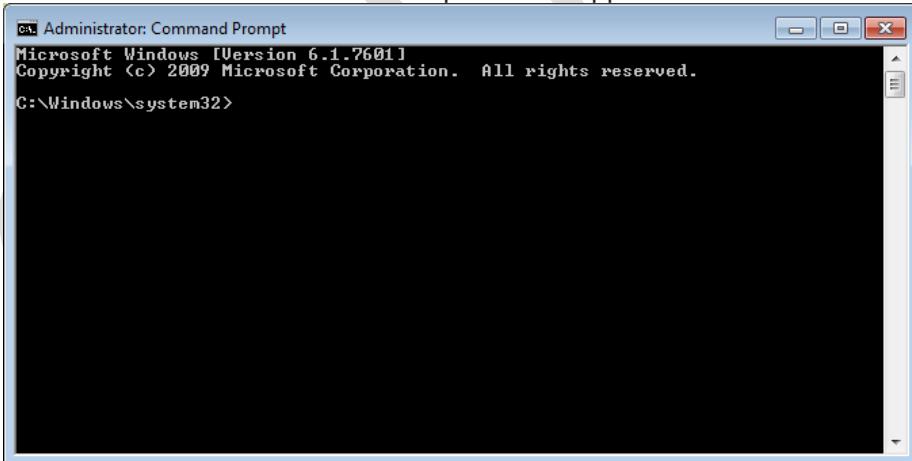
## Steps

To Install Tomcat as a Windows service:

5. From your desktop, select the Windows Start button to open the Windows Start Menu.
2. In the Search programs and files box, enter: Command Prompt.  
Command Prompt appears in the Programs list.



3. Right-click on Command Prompt, and then select Run as administrator.
4. The Administrator: Command Prompt window appears.



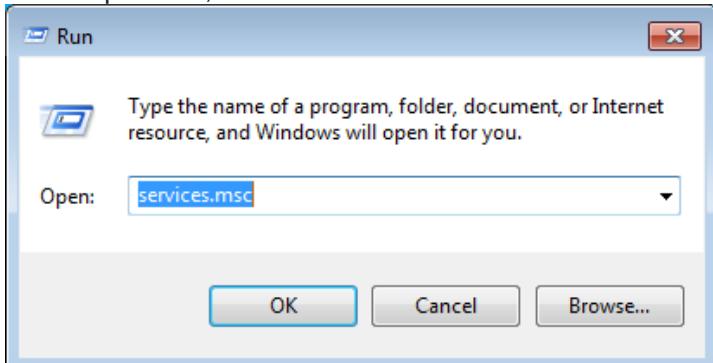
5. Change the directory path to C:\APMConnect\Utilities\Tomcat\bin, and enter the following command: service.bat install TAC\_Tomcat.
6. Press Enter.

The following message appears : C:\APMConnect\Utilities\Tomcat\bin>service.bat install TAC\_Tomcat  
Installing the service 'TAC\_Tomcat' ...Using CATALINA\_HOME: "C:\APMConnect\Utilities\Tomcat" Using  
CATALINA\_BASE: "C:\APMConnect\Utilities\Tomcat" Using JAVA\_HOME: "C:\Java\jdk1.7.0\_51" Using  
JRE\_HOME: "C:\Java\jdk1.7.0\_51\jre" Using JVM: "C:\Java\jdk1.7.0\_51\jre\bin\server\jvm.dll" The service  
'TAC\_Tomcat' has been installed.

Tomcat is installed as a Windows service.

To check Tomcat is started as a Widows service:

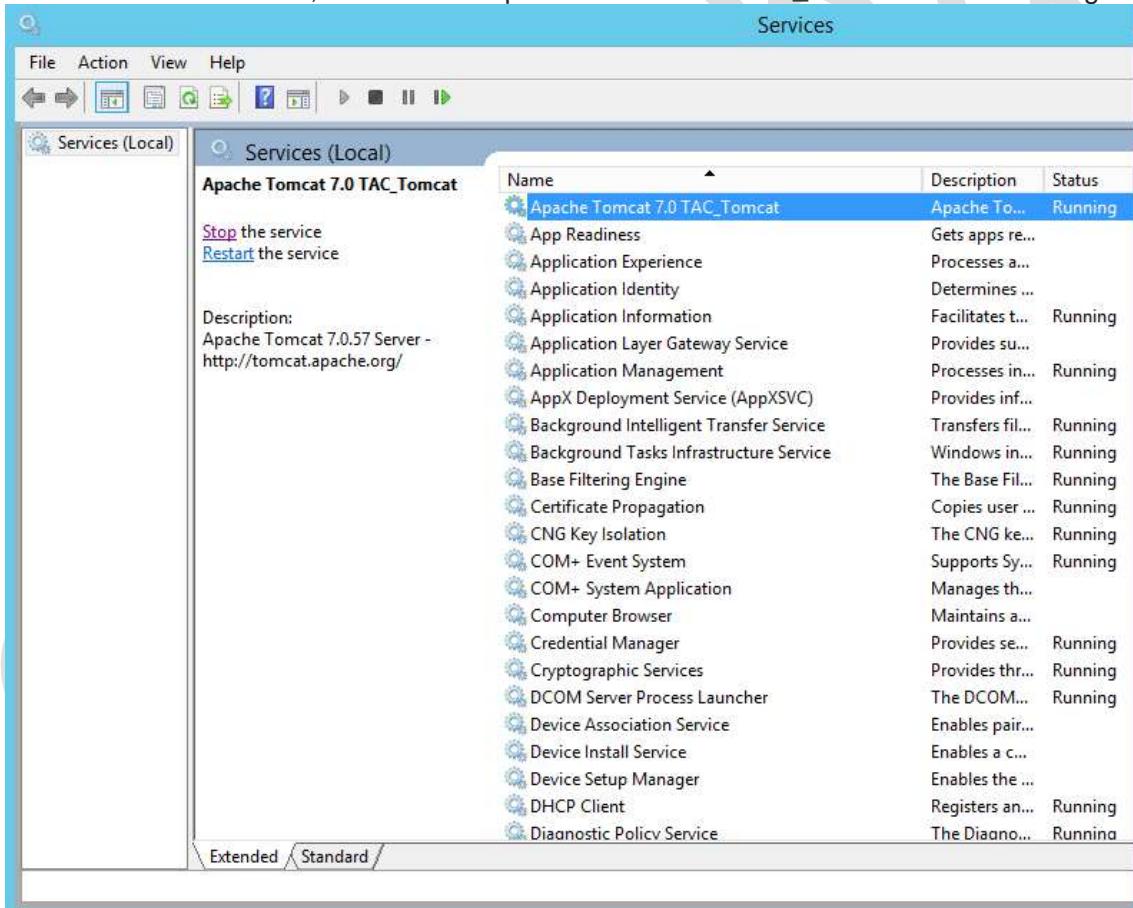
1. From your desktop, select the Windows Start button to open Windows Start Menu.
2. In the Search Programs and Files box, search for the Run program.  
The Run program appears in the Programs list.
3. Open the Run program.  
The Run window appears.
4. In the Open box, enter: services.msc.



5. Select OK.

The Services window appears.

If the service has started, the status of Apache Tomcat 7.0 TAC\_Tomcat will be Running.



6. If the service has not started, select Start.



Tomcat is installed and started as a Windows service.

## Other way of creating tomcat services for liferay project:

- Download and unzip Tomcat 7 for win 64 bit from <http://tomcat.apache.org/download-70.cgi> .
- Copy service.bat file and tomcat7.exe from the downloaded TOMCAT\_HOME/bin into the liferay-portal-6.1.0-ce-ga1/tomcat-7.0.23/bin directory of your liferay bundle.
- Follow the instructions contained here to run the "service.bat install" command (you must be Administrator of the machine to install the service).

```
D:\PRM\SitEnvironment\core\apache-tomcat-7.0.56\bin>service.bat install SiTafeely  
Installing the service 'SiTafeely' ...  
Using CATALINA_HOME: "D:\PRM\SitEnvironment\core\apache-tomcat-7.0.56"  
Using CATALINA_BASE: "D:\PRM\SitEnvironment\core\apache-tomcat-7.0.56"  
Using JAVA_HOME: "C:\Program Files\Java\jdk1.8.0_221"  
Using JRE_HOME: "C:\Program Files\Java\jdk1.8.0_221\jre"  
Using JVM: "C:\Program Files\Java\jdk1.8.0_221\jre\bin\server\jvm.dll"  
Access is denied.  
Failed installing 'SiTafeely' service
```

D:\PRM\SitEnvironment\core\apache-tomcat-7.0.56\bin>  
For above error, os is 64 bit and we are running 32bit tomcat. So this is the reason it is failing.

## Tasklist Administration:

```
TASKLIST /M
TASKLIST /V /FO CSV
TASKLIST /SVC /FO LIST
TASKLIST /APPS /FI "STATUS eq RUNNING"
TASKLIST /M wbem*
TASKLIST /S system /FO LIST
TASKLIST /S system /U domain\username /FO CSV /NH
TASKLIST /S system /U username /P password /FO TABLE /NH
TASKLIST /FI "USERNAME ne NT AUTHORITY\SYSTEM" /FI "STATUS eq running"
C:\Windows>
```

## TASKLIST cmd:

```
TASKLIST /M
TASKLIST /V /FO CSV
TASKLIST /SVC /FO LIST
TASKLIST /APPS /FI "STATUS eq RUNNING"
TASKLIST /M wbem*
TASKLIST /S system /FO LIST
TASKLIST /S system /U domain\username /FO CSV /NH
TASKLIST /S system /U username /P password /FO TABLE /NH
TASKLIST /FI "USERNAME ne NT AUTHORITY\SYSTEM" /FI "STATUS eq running"
```

## Services restart script:

172

For *exe* files, I suppose the differences are nearly unimportant.

But to start an *exe* you don't even need CALL.

When starting another batch it's a big difference,

**as CALL** will start it in the same window and the called batch has access to the same variable context.

So it can also change variables which affects the caller.

**START** will create a new cmd.exe for the called batch and without /b it will open a new window.

As it's a new context, variables can't be shared.

## Tafeely SIT services restart script and scheduled Task

```
@echo off
taskkill /F /IM java.exe
D:
cd D:\PRM\SitEnvironment\core\apache-tomcat-7.0.56\bin
start /wait /b cmd /c startup.bat
D:
cd D:\PRM\SitEnvironment\portal\liferay-portal-6.2-ce-ga3\tomcat-7.0.42\bin
start /wait /b cmd /c startup.bat
exit
```

## Tomcat Security

**Notice: This section is out of date with respect to recent version of Tomcat**

For a more up-to-date version of this content, please follow the TDS 5.0 guide starting at [https://docs.unidata.ucar.edu/tds/5.0/userguide/tomcat\\_permissions.html](https://docs.unidata.ucar.edu/tds/5.0/userguide/tomcat_permissions.html)

We believe Tomcat to be secure enough for typical scientific uses. There are no known security exploits or weaknesses. However, the Internet is a wild place, and we are not security experts so *caveat emptor*. These are best practices for running Tomcat.

Also see:

- Tomcat Security in TDS Tutorial
- <https://www.mulesoft.com/tcat/tomcat-security>

## Run as Unprivileged User

By default, Tomcat runs on port 8080 and therefore does not require root to run. It's important not to run as root. Create a special user, e.g. named "tomcat", which owns everything under  `${tomcat_home}`, and change to that user to run Tomcat. This special user will need read/write access to  `${tomcat_home}` and its subdirectories, and read access to your data directories. *Don't give the tomcat user any rights in any other directories.* If your operating system allows it (e.g. Unix, Linux), you might also not allow the tomcat user to log in, requiring instead that you log in as yourself, then switch to being the tomcat user.

## Restrict access to critical files

Make sure that everything under  `${tomcat_home}/conf/` can be read only by the tomcat user. Typically you would also give write access to the tomcat user.

## Use a Firewall

Unless you are on a private network, you need a firewall. A firewall restricts who is allowed to access network ports. Make the default setting to disallow all accesses, then enable just the ones that are needed.

Port **8080** should have unrestricted access.

If you are allowing remote management, you must also open up port **8443**.

Tomcat also uses port **8005** to enable shutdown. However, shutdown can only be run from the same machine as Tomcat is running from. As long as untrusted users aren't running on your server machine, you shouldn't have to worry about this port being open. However, you might want to restrict public access to it so hackers aren't tempted.

If you are also using Tomcat in conjunction with another web server like Apache to handle servlet/JSP requests, you need to allow that server access to port **8009**, but typically that can be restricted to accesses on the same machine or at least on your subnet. If you are running Tomcat only standalone (e.g. only running the THREDDS data server) then disable port 8009 in  `${tomcat_home}/conf/server.xml` file.

## Remove Default Tomcat Applications

Tomcat ships with several default web applications, found in the  `${tomcat_home}/webapps` directory. These defaults depend on the version of Tomcat and which installer you use.

- The **ROOT** application contains the server's main page. Any file that you add under  `${tomcat_home}/webapps/ROOT` will be served.
- The **admin** and **manager** applications are used for remote management. To use these applications, you must add users with roles **admin** and **manager**, respectively. The applications are then accessible from the main page, and can be used to add further users, start and stop webapps, etc. You should restrict the IP addresses that are allowed to run these applications by editing the files `admin.xml` and `manager.xml` in the  `${tomcat_home}/conf/Catalina/localhost/` directory.
- The **servlets-examples** and **jsp-examples** should probably be removed from a production server, to minimize security exposure. You can do this from the manager application or by deleting those directories from  `${tomcat_home}/webapps`.
- The **tomcat-docs**, **balancer** and **webdav** applications are probably harmless, but can also be removed if you want.

## Store Passwords as Digest

The file  `${tomcat_home}/conf/tomcat-users.xml` stores user names and passwords. By default the passwords are in clear text, e.g.:

```
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
  <role rolename="tdsConfig"/>
  <role rolename="manager"/>
  <role rolename="admin"/>
  <user username="sysadmin" password="yrPassword" roles="manager,admin"/>
```

```
<user username="cataloger" password="myPassword" roles="tdsConfig"/>
</tomcat-users>
```

Store them instead in digested form. First generate the digest (do this for each user):

```
> cd ${tomcat_home}/bin
> ./digest.sh -a SHA yrPassword
> yrPassword:aa01ea2afaae56c2b7da5e25ec18c505e58f12d7
```

If you are using DIGEST authentication, (only needed if you aren't using SSL), then use {username}:{realm}:{yrPassword} instead of {yrPassword} in calculating the digest, for example sysadmin:THREDDS Data Server:yrPassword. See this for more details.

Then cut and paste the digested passwords into the **tomcat-users.xml**:

```
<?xml version='1.0' encoding='utf-8'?>
<tomcat-users>
    <role rolename="tdsConfig"/>
    <role rolename="manager"/>
    <role rolename="admin"/>
    <user username="sysadmin" password="aa01ea2afaae56c2b7da5e25ec18c505e58f12d7"
roles="manager,admin"/>
    <user username="cataloger" password="5413ee24723bba2c5a6ba2d0196c78b3ee4628d1"
roles="tdsConfig"/>
</tomcat-users>
```

Then change the **server.xml** file to tell it to use digested passwords, by adding this **<Realm>** element to the **<Host>** element named **localhost**:

```
<Host name="localhost" debug="0" appBase="/opt/tomcat/webapps" unpackWARs="true" autoDeploy="true"
      xmlValidation="false" xmlNamespaceAware="false">
    <Realm className="org.apache.catalina.realm.MemoryRealm" digest="SHA" />
    ...
</Host>
```

## Separate User Roles For Restricted Datasets

The **tdsConfig**, **manager** and **admin** roles allow access to secure parts of Tomcat and TDS. These can only be accessed using HTTPS (TLS), and so are considered secure. If you are restricting access to datasets, you will also add other users who will have the **restrictedDatasetUser** role. *Do not give any of these users (including yourself!) any of the **tdsConfig**, **manager** or **admin** roles - keep them strictly separate.* This is because **restrictedDatasetUser** usage can also use non-HTTPS URLs, and so is vulnerable to session hijacking. By keeping the roles separate, you make sure that the worst that can happen is that someone can download some scientific data they shouldn't have access to.

## Restrict Access to Tomcat Manager Applications

The best way to secure the Tomcat manager and administration webapps is to restrict the set of IP addresses that can access them. This can be accomplished by including a **RemoteAddrValve** (Tomcat docs) in the **Context** element of these applications. For instance, to only allows connections from the localhost (127.0.0.1), i.e., the machine on which the Tomcat server is running, do the following:

1. Edit \${tomcat\_home}/conf/Catalina/localhost/admin.xml. It will probably look something like this:

```
2. <Context antiResourceLocking="false" privileged="true" />
```

Change it to include the highlighted line here:

```
<Context antiResourceLocking="false" privileged="true">
    <Valve className="org.apache.catalina.valves.RemoteAddrValve" allow="127\0\0\1"/>
</Context>
```

3. Edit \${tomcat\_home}/conf/Catalina/localhost/manager.xml and add the same line:

```
4. <Context antiResourceLocking="false" privileged="true">
5.     <Valve className="org.apache.catalina.valves.RemoteAddrValve" allow="127\0\0\1"/>
6. </Context>
```

NOTE: The value of the **allow** attribute must be a comma separated list of regular expressions used to match against the remote client's IP address. Here are several examples:

- `allow="128\.117\.140\.62"`
- `allow="128\.117\.140\.62,128\.117\.140\.63,128\.117\.140\.99"`
- `allow="128\.117\.140\..*"`

Similarly, you can use the **RemoteHostValve** to filter by host name. Again, the value of the **allow** and **deny** attributes must be a comma separated list of regular expressions which will be used to match against the remote client's host name. For instance:

```
<Valve className="org.apache.catalina.valves.RemoteHostValve" allow=".*\*.ucar\.edu"/>
```

## Java Security Manager

An additional level of security can be provided by running Tomcat with the Java Security Manager turned on. This can provide fine-grained security policies, at the cost of complexity in understanding what rights are needed to do any useful work, and how to grant them. This is needed if you allow untrusted servlets or JSPs to execute on your machine. If you are running just the THREDDS Data Server, you probably don't need to deal with this.

## Setting default timezone for the JVM of Tomcat

On Linux/Unix: add the "-Duser.timezone=Europe/Budapest" parameter (replacing the "Europe/Budapest" part with the ID of your timezone) to the JAVA\_OPTS environment variable in your {TOMCAT\_HOME}/bin/startup.sh. If you don't have a line in your startup.sh for setting the

JAVA\_OPTS, then you can add this to the start of the file (after the shell-specification in the first line of the shell-script, of course 😊):

```
export JAVA_OPTS="-Duser.timezone=Europe/Budapest"
```

On Windows: start the "Configure Tomcat" program from the Start Menu, select the Java tab and add the "-Duser.timezone=Europe/Budapest" to the Java options list.

On both platforms you'll have to restart Tomcat for the new default timezone setting to take effect.

## Cron Job:

### Syntax of crontab (field description)

The syntax is:

```
1 2 3 4 5 /path/to/command arg1 arg2
```

OR

```
1 2 3 4 5 /root/backup.sh
```

\* \* \* \* \* command to be executed

-----

| | | | |

| | | | ----- Day of week (0 - 7) (Sunday=0 or 7)

| | | | ----- Month (1 - 12)

| | ----- Day of month (1 - 31)

| ----- Hour (0 - 23)

----- Minute (0 - 59)

Your cron job looks as follows for system jobs:

```
1 2 3 4 5 /path/to/command arg1 arg2
```

OR

```
1 2 3 4 5 USERNAME /path/to/script.sh
```

## Job Failing in Cronjob's

This is not ok for a script which is set as a cron job:

```
mkdir jh  
cd jh
```

You should give the full path where jh directory must to be created. Also, in this path you should have permission to create new files/directories.

For example, your script should look like:

```
#!/bin/sh  
mkdir /home/lucky/jh  
cd /home/lucky/jh  
Also /usr/bin/sh is not the right path for sh. The right path is /bin/sh. You can check this with whereis  
sh command. And even so, your cron job should look like:  
20 * * * * /home/lucky/myfile.sh  
Don't forget to make the script executable:
```

```
chmod +x /home/lucky/myfile.sh
```

Very common problem.

Cron tasks are non-interactive and without having most of environment variables that a normal login session has. In your case, make sure that all commands have the proper path to them.

Typically, here is how it can be done in a safe manner:

Raw  
#!/bin/sh

```
PATH=/usr/bin:/sbin:; export PATH  
... the rest of the script
```

locate command is there in my environment and no permissions to install the package.

Finally,

Raw  
Good News Script working and getting the output through cron job.

```
Raw  
#!/bin/bash -x  
PATH=/usr/bin:/sbin:/usr/sbin:/oracle/base/11.1.0/db/client/bin;export PATH  
source /home/ibrahim/.bashrc  
ORACLE_HOME=/oracle/base/11.1.0/db/client; export ORACLE_HOME  
Cred=`cat < /home/ibrahim/Login`  
env  
RETVAL=` $Cred <<EOF  
SET LINESIZE 30000  
SET PAGESIZE 50000  
SET feedback off  
select * from emp;  
EXIT;  
EOF`  
echo "$RETVAL" >> /tmp/Crontest
```

Thank you so much Dusan Baljevic for solving a major problem for me.  
Thanks to all who helped for solving this issue.

You have added sqlplus to PATH instead of the directory of sqlplus. Please use:

```
Raw  
PATH=/usr/bin:/sbin:/oracle/base/11.1.0/db/client/bin  
export PATH
```

```
#!/bin/bash  
# environment path to set.  
=====  
PATH=/usr/bin:/sbin:/usr/sbin:/oracle/base/11.1.0/db/client/bin;export PATH source  
# this is the user you are working with on tty  
/home/ibrahim/.bashrc  
#ORACLE_HOME=/oracle/base/11.1.0/db/client  
Cred=`cat < /home/ibrahim/Scripts/Login`  
RETVAL=` $Cred <<EOF  
SET LINESIZE 30000  
SET PAGESIZE 50000  
SET feedback off  
select * from emp;  
EXIT;
```

EOF`

```
echo "$RETVAL" >> /tmp/Crontest
```

Python:

How to install Python on windows

<https://www.howtogeek.com/197947/how-to-install-python-on-windows/>

PIP installation on windows

On Windows pip3 should be in the Scripts path of your Python installation:

C:\path\to\python\Scripts\pip3

Use: `where python`

to find out your where the Python executable(s) is/are located. The result should look like this:  
C:\path\to\python\python.exe or C:\path\to\python\python3.exe

You can if check pip3 works with this absolute path:

C:\path\to\python\Scripts\pip3

if yes, add C:\path\to\python\Scripts to your environmental variable PATH .

> python -m pip install --upgrade pip

```
C:\Users\80053806\AppData\Local\Programs\Python\Python37-32>cd Scripts
C:\Users\80053806\AppData\Local\Programs\Python\Python37-32\Scripts>l1s
easy_install-3.7.exe easy_install.exe pip.exe pip3.7.exe pip3.exe

C:\Users\80053806\AppData\Local\Programs\Python\Python37-32\Scripts>pip3 install -r requirements.txt
You are using pip version 19.0.1, however version 18.1 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

C:\Users\80053806\AppData\Local\Programs\Python\Python37-32\Scripts>python -m pip install --upgrade pip
Collecting pip
  Downloading https://files.pythonhosted.org/packages/c2/d7/90f34cb0d83a6c5631cf71dfe64cc1054598c843a92b900e55575cc2ac37/pip-18.1-py2.py3-none-any.whl (1.3MB)
    100% |████████████████████████████████| 1.3MB 751kB/s
Installing collected packages: pip
  Found existing installation: pip 10.0.1
  Uninstalling pip-10.0.1...
    Successfully uninstalled pip-10.0.1
Successfully installed pip-18.1

C:\Users\80053806\AppData\Local\Programs\Python\Python37-32\Scripts>pips install -r requirements.txt
Requirement already up-to-date: pip in /c/Users/80053806/AppData/Local/Programs/Python/Python37-32/Scripts

C:\Users\80053806\AppData\Local\Programs\Python\Python37-32\Scripts>
```

>Pip3 install -r .\xdecrypt-master\requirements.txt

```
C:\Users\80053806\AppData\Local\Programs\Python\Python37-32\Scripts>pip3 install -r .\Xdecrypt-master\requirements.txt
Collecting pycryptodome (from -r .\Xdecrypt-master\requirements.txt (line 1))
  Downloading https://files.pythonhosted.org/packages/ab/80/3161bb89ad3d00e80b1a28fae899ec8ec2c1697356c74888f3d9a1f1dbd/pycryptodome-3.7.0-cp37-cp37m-win32.whl (7.9MB)
    84% |████████████████████████████████| 6.7MB 571kB/s eta 0:00:03
```

Python scripting on unix

Most useful unix modules are OS , subprocess, sys

Import os

os.system('ls')

And save the above code as test.py and execute as \$python test.py

Look at the subprocess module in the standard library:

```
from subprocess import call
call(["ls", "-l"])
```

The advantage of subprocess vs. system is that it is more flexible (you can get the stdout, stderr, the "real" status code, better error handling, etc...).

The official documentation recommends the subprocess module over the alternative os.system():

The subprocess module provides more powerful facilities for spawning new processes and retrieving their results; using that module is preferable to using this function [os.system()].

The "Replacing Older Functions with the subprocess Module" section in the subprocess documentation may have some helpful recipes.

Official documentation on the subprocess module:

- [Python 3 – subprocess](#)

The classical example from the subprocess module docs is:

```
import subprocess
import sys

# some code here

pid = subprocess.Popen([sys.executable, "longtask.py"]) # call subprocess

# some more code here
```

Python Webserver

Create a directory and navigate into that directory.

And run the command to start the webserver as follows. "python -m SimpleHTTPServer"

```
[siebel@eeecsaruh2hoa176]/home/siebel/gouse_bck/testweb>python -m SimpleHTTPServer
```

Webserver started.

```
[siebel@eeecsaruh2hoa176]/home/siebel/gouse_bck/testweb>python -m SimpleHTTPServer
Serving HTTP on 0.0.0.0 port 8000 ...
10.66.113.46 - - [15/Jan/2019 13:02:21] "GET / HTTP/1.1" 200 -
10.66.113.46 - - [15/Jan/2019 13:02:22] "GET /gouse/ HTTP/1.1" 200 -
10.66.113.46 - - [15/Jan/2019 13:02:25] "GET /test.txt HTTP/1.1" 200 -
10.66.113.46 - - [15/Jan/2019 13:03:10] "GET / HTTP/1.1" 200 -
10.66.113.46 - - [15/Jan/2019 13:03:12] "GET /test.py HTTP/1.1" 200 -
```

#### How to embed any script in HTML web page

I have written python script that blink LED. I want to make web page in HTML, contain button so when button will click on web page. Python script will be run on web server.

Here is html page

#### HTML:

1. <html>
2. <body>
3. <titletitle- 4. <h1h1>
- 5. <form>
- 6.   <input type="Button" value="Blink LED">
- 7. </form>
- 8. </body>
- 9. </html>

Web browser display following page



I want to embed python script in HTML web page. I think have to do some change on html file. I was searching on google and I found some example

#### Code (Text) :

1. <a href="/home/pi/Documents/LED/led.py">.
- 2.
3. <link rel="/home/pi/Documents/LED/led.py">

Where to add <href= "link location"> in my HTML Code. How to make complete HTML page that will run python script on browser ?

The python script is located in:

/home/pi/Documents/LED/led.py

Note : I am not talking only python script. I am looking way for any script like php,java or python. here I am just taking example of python script

#### Pip installation and virtual environment setup steps:

Microsoft Windows [Version 10.0.14393]

(c) 2016 Microsoft Corporation. All rights reserved.

```
C:\Users\80053806>python
```

```
Python 3.7.0 (v3.7.0:1bf9cc5093, Jun 27 2018, 04:06:47) [MSC v.1914 32 bit (Intel)] on win32
```

```
Type "help", "copyright", "credits" or "license" for more information.
```

```
>>> quit()
```

```

C:\Users\80053806>cd C:\Projects

C:\Projects>mkdir pyarflw

C:\Projects>cd pyarflw

C:\Projects\pyarflw>virtualenv --version
'virtualenv' is not recognized as an internal or external command,
operable program or batch file.

Virtual environment setup with the PIP:

=====
C:\Projects\pyarflw>pip install virtualenv
Collecting virtualenv
  Downloading https://files.pythonhosted.org/packages/8f/f1/c0b069ca6cb44f9681715232e6d3d65c75866dd231c5e4a88e80a46634bb/virtualenv-16.3.0-py2.py3-none-any.whl (2.0MB)
    100% |██████████| 2.0MB 447kB/s
Requirement already satisfied: setuptools>=18.0.0 in c:\users\80053806\appdata\local\programs\python\python37-32\lib\site-packages (from virtualenv) (39.0.1)
Installing collected packages: virtualenv
Successfully installed virtualenv-16.3.0

C:\Projects\pyarflw>virtualenv --version
16.3.0

pip virtual environment setup on windows:

=====
C:\Projects\pyarflw>pip install virtualenv
Requirement already satisfied: virtualenv in c:\users\80053806\appdata\local\programs\python\python37-32\lib\site-packages (16.3.0)
Requirement already satisfied: setuptools>=18.0.0 in c:\users\80053806\appdata\local\programs\python\python37-32\lib\site-packages (from virtualenv) (39.0.1)

pip can be found in side below path in window:

=====
C:\Projects\pyarflw>cd c:\users\80053806\appdata\local\programs\python\python37-32\scripts

c:\Users\80053806\AppData\Local\Programs\Python\Python37-32\Scripts>ls
Xdecrypt-master      __pycache__      easy_install.exe  pip3.7.exe  pywin32_postinstall.py  virtualenv.exe
Xdecrypt-master.zip  easy_install-3.7.exe  pip.exe          pip3.exe     pywin32 testall.py

c:\Users\80053806\AppData\Local\Programs\Python\Python37-32\Scripts>cd
c:\Users\80053806\AppData\Local\Programs\Python\Python37-32\Scripts

c:\Users\80053806\AppData\Local\Programs\Python\Python37-32\Scripts>cd /
c:>mkdir Projects
c:>cd "Projects"

```

```

Activate the python virtual Environment:
c:\Projects>mkdir pyarflw
c:\Projects>cd pyarflw

c:\Projects\pyarflw>ls

c:\Projects\pyarflw>virtualenv env
Using base prefix 'c:\\users\\80053806\\appdata\\local\\programs\\python\\python37-32'
New python executable in c:\Projects\pyarflw\env\Scripts\python.exe
Installing setuptools, pip, wheel...
done.

c:\Projects\pyarflw>ls
env
c:\Projects\pyarflw>cd env
c:\Projects\pyarflw\env>ls
Include Lib Scripts tcl
c:\Projects\pyarflw\env>cd Scripts
c:\Projects\pyarflw\env\Scripts>ls
activate activate.ps1 activate this.py easy install-3.7.exe pip.exe pip3.exe python3.dll
pythonw.exe
activate.bat activate.xsh deactivate.bat easy install.exe pip3.7.exe python.exe python37.dll
wheel.exe
cd c:\Projects\pyarflw>
c:\Projects\pyarflw>c:\Projects\pyarflw\env\Scripts\activate
(env) c:\Projects\pyarflw>

```

## Python pip install & Download

How to Install a Package in Python using PIP

[Python / May 17, 2020](#)

In this short guide, I'll show you how to install a package in Python using PIP. I'll also demonstrate how to uninstall a package that is no longer needed.

If you're using Windows, you'll be able to install a Python package by opening the Windows *Command Prompt*, and then typing this command:

```
pip install package name
```

Note: the above method would only work if you already added [Python to Windows path](#). Don't worry if you don't know what it means, as in the next section, I'll cover the full steps to install a package in Python using PIP.

Here are the topics to be reviewed:

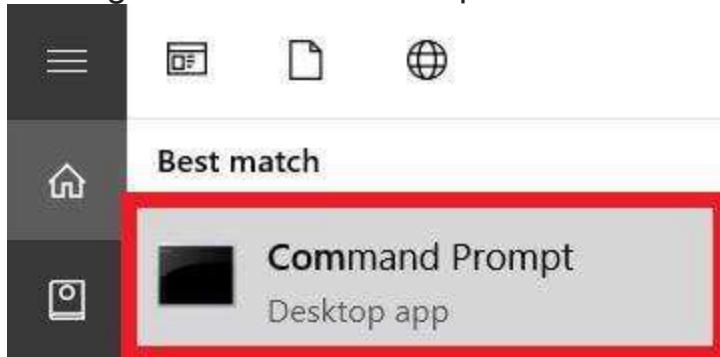
- Simple example with the steps to install the *pandas* package
- Uninstall a Python package
- Tool that will allow you to easily install/uninstall Python packages

## Steps to Install a Package in Python using PIP

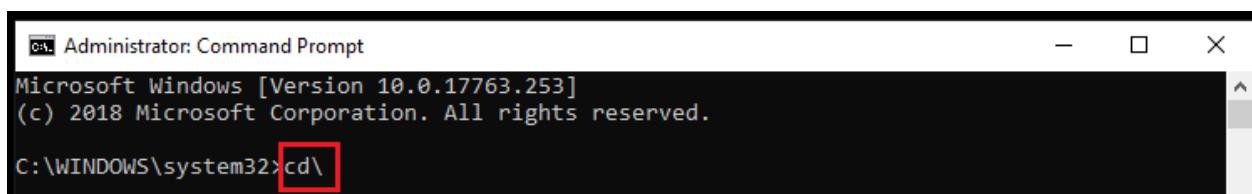
(1) First, type **Command Prompt** in the Windows search box:



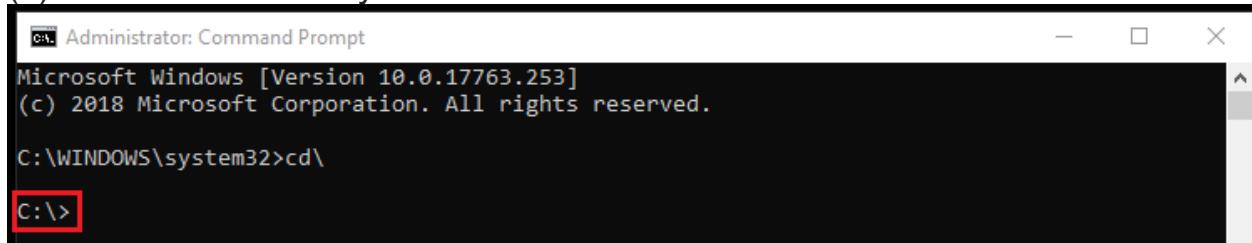
(2) Right click on the Windows Command Prompt. Then, select **Run as administrator** (by running the Command Prompt as an administrator, you'll avoid any permission issues):



(3) In the Command Prompt, type "cd\" as this command will ensure that your starting point has only the drive name:

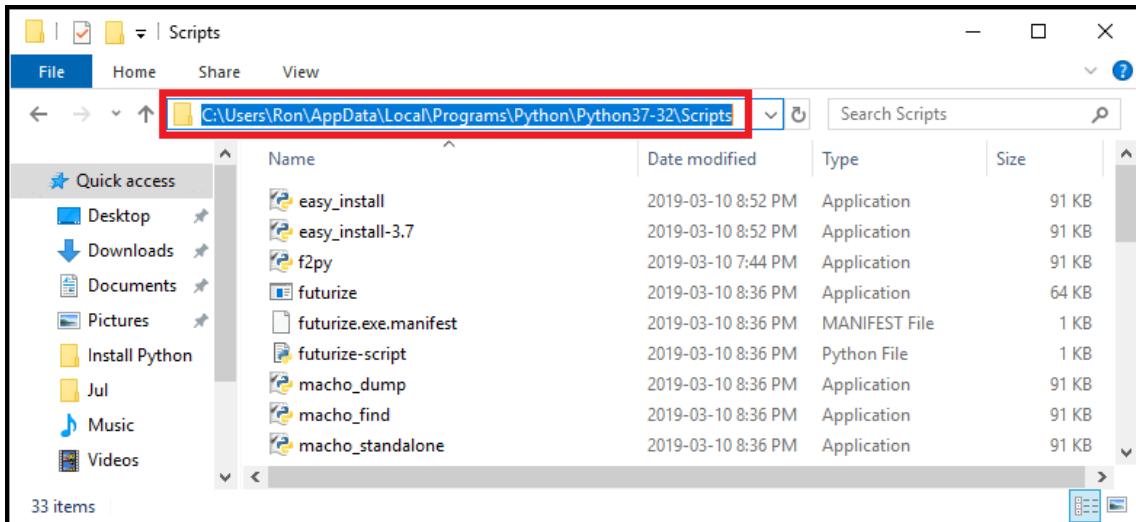


(4) Press **Enter**. Now you'll see the drive name of C:\>



(5) Locate your Python **Scripts** path. The Scripts folder can be found within the Python application folder, where you originally [installed Python](#).  
In my case, the Python Scripts path is:

**C:\Users\Ron\AppData\Local\Programs\Python\Python37-32\Scripts**



In the Command Prompt, type **cd** followed by your Python **Scripts** path:

```
C:\>cd C:\Users\Ron\AppData\Local\Programs\Python\Python37-32\Scripts
```

(6) Press **Enter**, and you'll see something similar to the following:

```
C:\Users\Ron\AppData\Local\Programs\Python\Python37-32\Scripts>
```

(7) Now, type the pip install command to install your Python package. The pip install command has the following structure:

```
pip install package name
```

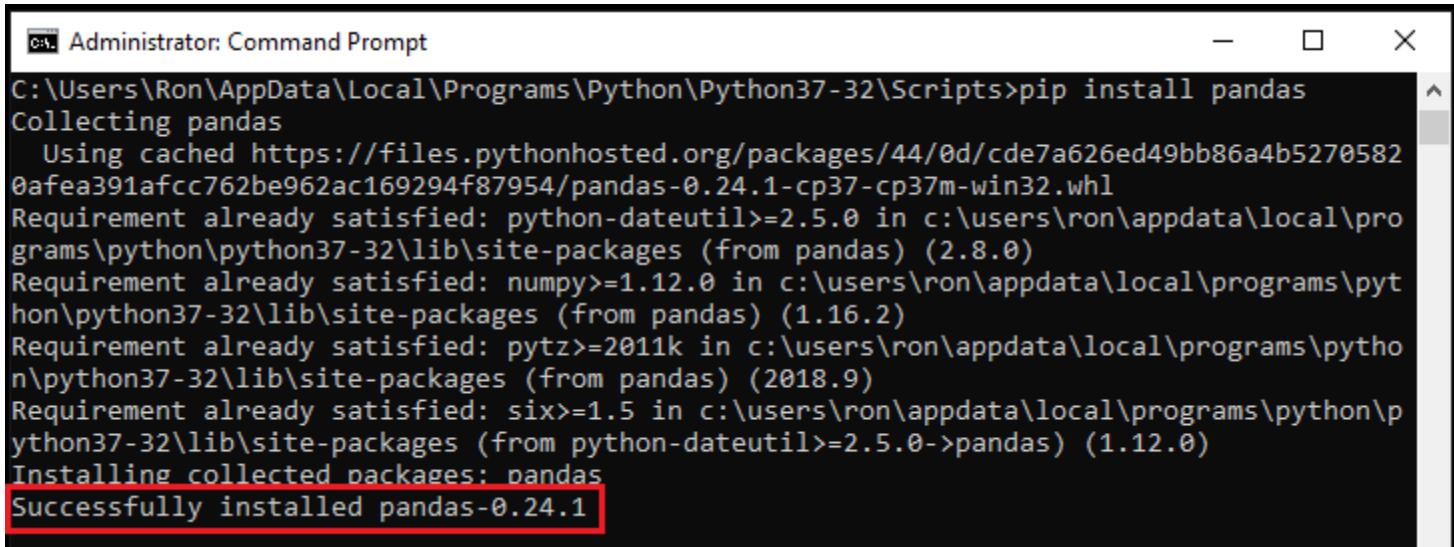
Since in our case, we would like to install the *pandas* package, then type the following command in the Command Prompt:

**pip install pandas**



```
C:\Users\Ron\AppData\Local\Programs\Python\Python37-32\Scripts>pip install pandas
```

(8) Finally, press **Enter**, and you'll notice that the package (here it's pandas) will be installed:



```
C:\Users\Ron\AppData\Local\Programs\Python\Python37-32\Scripts>pip install pandas
Collecting pandas
  Using cached https://files.pythonhosted.org/packages/44/0d/cde7a626ed49bb86a4b5270582
0afea391afcc762be962ac169294f87954/pandas-0.24.1-cp37-cp37m-win32.whl
Requirement already satisfied: python-dateutil>=2.5.0 in c:\users\ron\appdata\local\pro
grams\python\python37-32\lib\site-packages (from pandas) (2.8.0)
Requirement already satisfied: numpy>=1.12.0 in c:\users\ron\appdata\local\programs\pyt
hon\python37-32\lib\site-packages (from pandas) (1.16.2)
Requirement already satisfied: pytz>=2011k in c:\users\ron\appdata\local\programs\pytho
n\python37-32\lib\site-packages (from pandas) (2018.9)
Requirement already satisfied: six>=1.5 in c:\users\ron\appdata\local\programs\python\p
ython37-32\lib\site-packages (from python-dateutil>=2.5.0->pandas) (1.12.0)
Installing collected packages: pandas
Successfully installed pandas-0.24.1
```

You can quickly check if the package was successfully installed in Python, by opening the **Python IDLE** and then running the command “import pandas”  
If no errors appear (after you press F5 to run the import command), then the package was successfully installed.

Mission Accomplished!

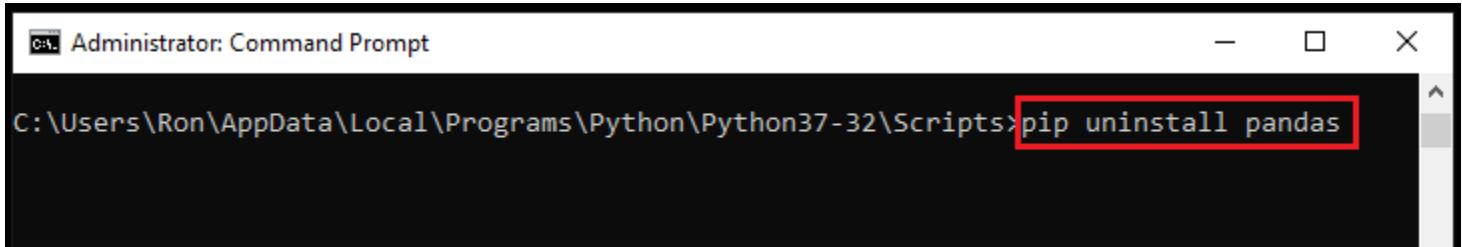
If you're dealing with Anaconda, you may also want to check how to [install a package using Anaconda](#).

Uninstall a Package using PIP

To [uninstall a package using PIP](#), simply type the following command in the Command Prompt (do not forget to set the Python Scripts path before you type this command):

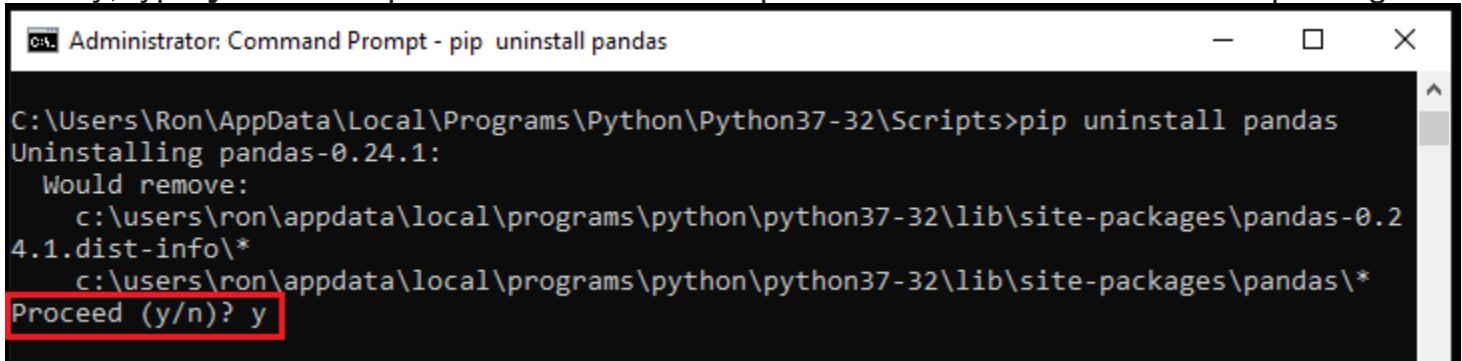
```
pip uninstall package name
```

In our case, type: **pip uninstall pandas** and then press **Enter**:



```
C:\Users\Ron\AppData\Local\Programs\Python\Python37-32\Scripts>pip uninstall pandas
```

Finally, type **y** and then press **Enter** in order to proceed with the removal of the package:



```
C:\Users\Ron\AppData\Local\Programs\Python\Python37-32\Scripts>pip uninstall pandas
Uninstalling pandas-0.24.1:
Would remove:
  c:\users\ron\appdata\local\programs\python\python37-32\lib\site-packages\pandas-0.2
4.1.dist-info\*
  c:\users\ron\appdata\local\programs\python\python37-32\lib\site-packages\pandas\*
Proceed (y/n)? y
```

## Tool to install/uninstall a Python Package

Now I'll share with you the code to install/uninstall Python packages using a simple tool I created.

Note that you'll need to [add Python to Windows path](#) before you can start using the tool. Here is the full Python code:

```
import os
import tkinter as tk
root= tk.Tk()

canvas1 = tk.Canvas(root, width = 300, height = 350, bg = 'gray90', relief = 'raised')
canvas1.pack()

label1 = tk.Label(root, text='Type Package:', bg = 'gray90')
label1.config(font=('helvetica', 14))
canvas1.create_window(150, 80, window=label1)

entry1 = tk.Entry (root, width=27)
canvas1.create_window(150, 120, window=entry1)

def installPackage ():
    global installPythonPackage
    installPythonPackage = 'pip install ' + entry1.get()

    os.system('start cmd /k ' + installPythonPackage)
def uninstallPackage ():
    global uninstallPythonPackage
    uninstallPythonPackage = 'pip uninstall ' + entry1.get()
```

```

os.system('start cmd /k ' + uninstallPythonPackage)

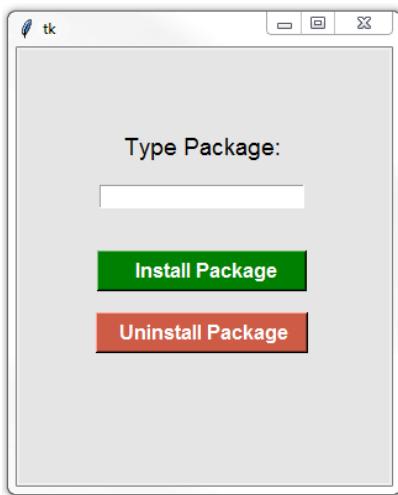
button1 = tk.Button(text='      Install Package      ', command=installPackage, bg='green',
fg='white', font=('helvetica', 12, 'bold'))
canvas1.create_window(150, 180, window=button1)

button2 = tk.Button(text='  Uninstall Package  ', command=uninstallPackage, bg='coral3',
fg='white', font=('helvetica', 12, 'bold'))
canvas1.create_window(150, 230, window=button2)

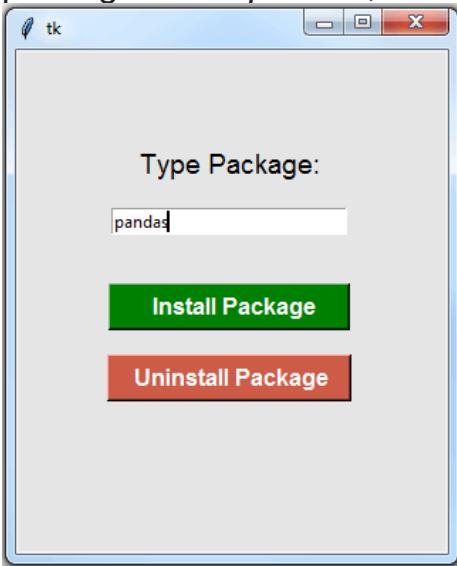
root.mainloop()

```

Simply run the code in Python and you would see the following screen:



Type the package name that you'd like to install or uninstall. For example, type the package name *pandas*, and then click on the green button to install the package:



That's it! You should be able to enjoy the [pandas](#) package.

## Python Automa's Functions & Classes

```
https://www.getautoma.com/docs
using module for this GUI automation is "from automa.api import *"

run the below script by saving into .at file extention
playback("C:\\\\Users\\\\80053806\\\\Desktop\\\\test.at")

start(r"E:\\siebeltoolsIP16\\Tools\\BIN\\siebdev.exe", "/c", r"E:\\siebeltoolsIP16\\Tools\\bin\\enu\\tools.cfg", "/u",
"sadmin", "/p", "Mobilyladm", "/d", "SBLSSITIP16")
switch_to(Window("Siebel Tools - Siebel Repository"))
press(CTRL + 'e')
press('a')
press(CTRL + 'q')
write('"ABO Bulk Request Explode Exceptions List Applet" or "ABO Bulk Request - Quote Form Applet Dashboard"')
press(ENTER)
press(CTRL + 'F7')
write("E:\\GouseShaik\\siebelBuild\\SRF\\siebel_sia_19thMarch2019.srf", into="Siebel repository file")
click("Compile")
scroll_down()
press(CTRL + 'F7')
write("E:\\GouseShaik\\siebelBuild\\SRF\\siebel_sia_19thMarch2019.srf", into="Siebel repository file")
click("Compile")
press(ALT + 'f')
press("Exit")
```



Automa.zip

## Apache Airflow:

*Airflow installation steps with pip on window [ will not work]:*

```
#=====
Once you have setup the virtual environment, now install Apache Airflow
C:\Projects\pyarflw>set SLUGIFY_USES_TEXT_UNIDECODE=yes
(env) c:\Projects\pyarflw> pip install apache-airflow
Failed : Command "c:\projects\pyarflw\env\scripts\python.exe c:\projects\pyarflw\env\lib\site-
packages\pip\_vendor\pep517\_in_process.py get_requires_for_build_wheel
C:\Users\80053806\AppData\Local\Temp\tmp5ne 5wc6" failed with error code 1 in
C:\Users\80053806\AppData\Local\Temp\pip-install-q_vyhbblu\pendulum

(env) c:\Projects\pyarflw> pip3 install apache-airflow
(env) c:\Projects\pyarflw> python -m pip install apache-airflow
```

We make heavy use of airflow, and we use VM's running Linux to get it running. We have Windows machines, but have to use VM's or mount drives on Linux/Mac boxes to get it to work. As far as I know it's not even on the road map to have Airflow run on Windows.

So, long answer short: No, even as of October 2017 airflow runs only on Unix based systems (it uses some python libraries that only work for unix underneath), and it's unlikely that anytime soon it will support Windows.

### *Airflow installation on ubuntu:*

1. Apt-get update

```

2. Install python : apt-get install python
3. Apt-get install python-pip
4. pip install --upgrade setuptools
5. cd
6. mkdir airflow
# install from pypi using pip
7. pip install apache-airflow
8. Open a new terminal (I was surprised, but this seemed to be required).
# initialize the database
1. airflow initdb
# start the web server, default port is 8080
1. airflow webserver -p 8080
access the airflow webserver once you started the above command.
http://192.168.56.100:8080/admin/airflow/tree?dag\_id=example\_bash\_operator
Tutorial: https://airflow.apache.org/tutorial.html

```

# Getting started with Apache Airflow



## Credit Airflow Official Site

In this post, I am going to discuss Apache Airflow, a workflow management system developed by Airbnb.

Earlier I had [discussed](#) writing basic ETL pipelines in Bonobo. Bonobo is cool for write ETL pipelines but the world is not all about writing ETL pipelines to automate things. There are other use cases in which you have to perform tasks in a certain order once or periodically. For instance:

- Monitoring Cron jobs
- transferring data from one place to other.
- Automating your DevOps operations.
- Periodically fetching data from websites and update the database for your awesome price comparison system.
- Data processing for recommendation based systems.
- Machine Learning Pipelines.
- Possibilities are endless.

Before we move on further to implement Airflow in our systems, let's discuss what actually is Airflow and it's terminologies.

## What is Airflow?

From the Website:

*Airflow is a platform to programmatically author, schedule and monitor workflows.*

*Use airflow to author workflows as directed acyclic graphs (DAGs) of tasks. The airflow scheduler executes your tasks on an array of workers while following the specified dependencies. Rich command line utilities make performing complex surgeries on DAGs a snap. The rich user interface makes it easy to visualize pipelines running in production, monitor progress, and troubleshoot issues when needed.*

Basically, it helps to automate scripts in order to perform tasks. Airflow is Python-based but you can execute a program irrespective of the language. For instance, the first stage of your workflow has to execute a C++ based program to perform image analysis and then a Python-based program to transfer that information to S3. Possibilities are endless.

## What is Dag?

From Wikipedia

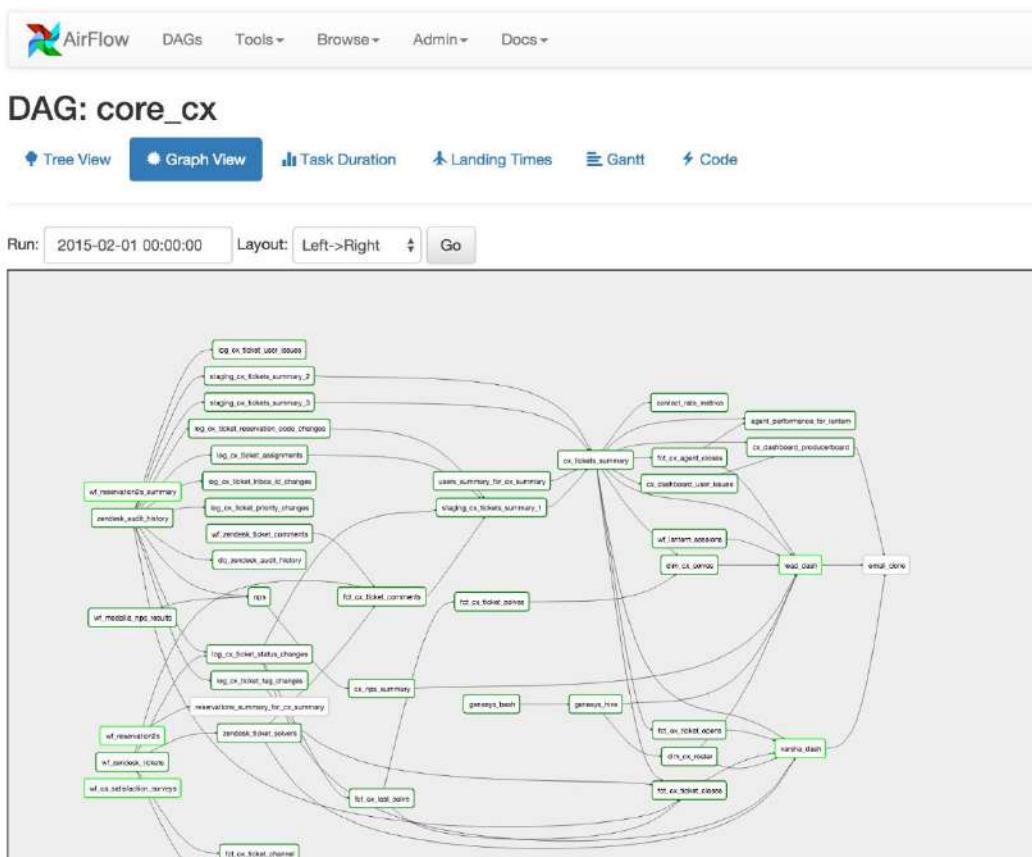
*In mathematics and computer science, a directed acyclic graph (DAG /'dæg/ (About this sound listen)), is a finite directed graph with no directed cycles. That is, it consists of finitely many vertices and edges, with each edge directed from one vertex to another, such that there is no way to start at any vertex v and follow a consistently-directed sequence of edges that eventually loops back to v again. Equivalently, a DAG is a directed graph that has a topological ordering, a sequence of the vertices such that every edge is directed from earlier to later in the sequence.*

Let me try to explain in simple words: You can only be a son of your father but not vice versa. OK, it's lame or weird but could not find a better example to explain a *directed cycle*.

## Why we use Airflow:

Here are a few reasons to use Airflow:

- **Open source:** After starting as an internal project at Airbnb, Airflow had a natural need in the community. This was a major reason why it eventually became an open source project. It is currently maintained and managed as an incubating project at Apache.
- **Web Interface:** Airflow ships with a Flask app that tracks all the defined workflows, and lets you easily change, start, or stop them. You can also work with the command line, but the web interface is more intuitive.
- **Python Based:** Every part of the configuration is written in Python, including configuration of schedules and the scripts to run them. This removes the need to use restrictive JSON or XML configuration files.



Airflow DAG (Credit: Apache Airflow)

In Airflow all workflows are DAGs. A Dag consists of *operators*. An operator defines an individual task that needs to be performed. There are different types of operators available( As given on Airflow Website):

- **BashOperator** - executes a bash command
  - **PythonOperator** - calls an arbitrary Python function
  - **EmailOperator** - sends an email
  - **SimpleHttpOperator** - sends an HTTP request
  - **MySqlOperator, SqliteOperator, PostgresOperator, MsSqlOperator, OracleOperator, JdbcOperator**, etc. - executes a SQL command
  - **Sensor** - waits for a certain time, file, database row, S3 key, etc...
- You can also come up with a custom operator as per your need.

### Installation and Setup

Airflow is Python based. The best way to install it is via pip tool.  
pip install apache-airflow

To verify whether it got installed, run the command: airflow version and it should print something like:

```
[2018-09-22 15:59:23,880] {__init__.py:51} INFO - Using executor SequentialExecutor
```

```
_____|_\(| )_____ / / / / | /| / /  
_____|_\|- / - / / / / / / / \ / | / /  
_/_/ |/_/ / / / / / / / / \ / | / /  
v1.10.0
```

You will need to install mysqlclient as well to incorporate MySQL in your workflows. It is optional though.

#### **pip install mysqlclient**

Before you start anything, create a folder and set it as AIRFLOW\_HOME. In my case it is airflow\_home. Once created you will call export command to set it in the path.

```
export AIRFLOW_HOME='pwd' airflow_home
```

Make sure you are a folder above of airflow\_home before running the export command. Within airflow\_home you will create another folder to keep DAGs. Call it dags

```
If you set load_examples=False it will not load default examples on the Web interface.
```

Now you have to call airflow initdb within airflow home folder. Once it's done it

```
creates airflow.cfg and unittests.cfg
```

airflow.db is an SQLite file to store all configuration related to run workflows. airflow.cfg is to keep all initial settings to keep things running.

In this file, you can see sql\_alchemy\_conn parameter with the value ..../airflow\_home/airflow.db

You can use MySQL if you want. For now, just stick with basic settings.

So far so good, now without wasting any time let's start the web server.

#### **airflow webserver**

When starts it shows the screen like:

```
2018-09-20 22:36:24,943] {__init__.py:51} INFO - Using executor SequentialExecutor
```

```
/anaconda3/anaconda/lib/python3.6/site-packages/airflow/bin/cli.py:1595: DeprecationWarning: The  
celeryd_concurrency option in [celery] has been renamed to worker_concurrency - the old setting has been used,  
but please update your config.
```

```
default=conf.get('celery', 'worker_concurrency'),
```

```
_____|_\(| )_____ / / / / | /| / /  
_____|_\|- / - / / / / / / / \ / | / /  
_/_/ |/_/ / / / / / / / / \ / | / /  
v1.10.0
```

```
[2018-09-19 14:21:42,340] {__init__.py:57} INFO - Using executor SequentialExecutor
```

```
_____|_\(| )_____ / / / / | /| / /  
_____|_\|- / - / / / / / / / \ / | / /  
_/_/ |/_/ / / / / / / / / \ / | / /  
/anaconda3/anaconda/lib/python3.6/site-packages/flask/ext/hook.py:71: ExtDeprecationWarning: Importing  
flask.ext.cache is deprecated, use flask_cache instead.
```

```
.format(x=modname), ExtDeprecationWarning
```

```
[2018-09-19 14:21:43,119] [48995] {models.py:167} INFO - Filling up the DagBag from  
/Development/airflow_home/dags
```

Running the Gunicorn Server with:

Workers: 4 sync

Host: 0.0.0.0:8080

Now when you visit 0.0.0.0:8080 it shows a screen like:



## DAGs

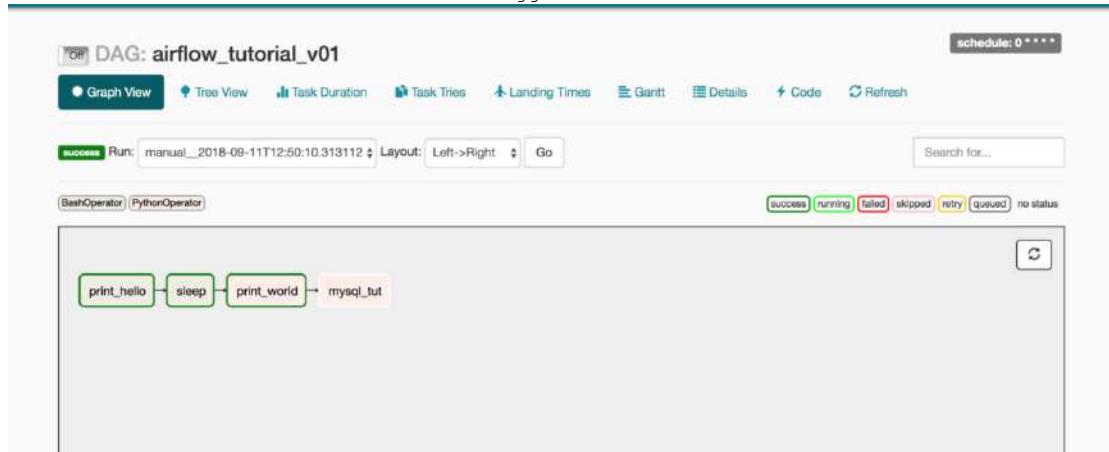
Search:

| # | DAG                                     | Schedule     | Owner   | Recent Tasks                                       | Last Run                                                       | DAG Runs | Links                                                                                                                                                                            |
|---|-----------------------------------------|--------------|---------|----------------------------------------------------|----------------------------------------------------------------|----------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1 | example_bash_operator                   | 0 * * * *    | airflow | <span style="color: lightgray;">0 0 0 0 0 0</span> | <span style="color: lightgray;">2018-09-22 10:48:00 UTC</span> | 0        | <span style="color: lightgray;">Run</span> <span style="color: lightgray;">Edit</span> <span style="color: lightgray;">Delete</span> <span style="color: lightgray;">Logs</span> |
| 2 | example_branch_dop_operator_v3          | * * * * *    | airflow | <span style="color: lightgray;">0 0 0 0 0 0</span> | <span style="color: lightgray;">2018-09-22 10:48:00 UTC</span> | 0        | <span style="color: lightgray;">Run</span> <span style="color: lightgray;">Edit</span> <span style="color: lightgray;">Delete</span> <span style="color: lightgray;">Logs</span> |
| 3 | example_branch_operator                 | daily        | airflow | <span style="color: lightgray;">0 0 0 0 0 0</span> | <span style="color: lightgray;">2018-09-22 10:48:00 UTC</span> | 0        | <span style="color: lightgray;">Run</span> <span style="color: lightgray;">Edit</span> <span style="color: lightgray;">Delete</span> <span style="color: lightgray;">Logs</span> |
| 4 | example_http_operator                   | 1 day, 00:00 | airflow | <span style="color: lightgray;">0 0 0 0 0 0</span> | <span style="color: lightgray;">2018-09-22 10:48:00 UTC</span> | 0        | <span style="color: lightgray;">Run</span> <span style="color: lightgray;">Edit</span> <span style="color: lightgray;">Delete</span> <span style="color: lightgray;">Logs</span> |
| 5 | example_kubernetes_executor             | None         | airflow | <span style="color: lightgray;">0 0 0 0 0 0</span> | <span style="color: lightgray;">2018-09-22 10:48:00 UTC</span> | 0        | <span style="color: lightgray;">Run</span> <span style="color: lightgray;">Edit</span> <span style="color: lightgray;">Delete</span> <span style="color: lightgray;">Logs</span> |
| 6 | example_passing_params_via_test_command | * * * * *    | airflow | <span style="color: lightgray;">0 0 0 0 0 0</span> | <span style="color: lightgray;">2018-09-22 10:48:00 UTC</span> | 0        | <span style="color: lightgray;">Run</span> <span style="color: lightgray;">Edit</span> <span style="color: lightgray;">Delete</span> <span style="color: lightgray;">Logs</span> |
| 7 | example_python_operator                 | None         | airflow | <span style="color: lightgray;">0 0 0 0 0 0</span> | <span style="color: lightgray;">2018-09-22 10:48:00 UTC</span> | 0        | <span style="color: lightgray;">Run</span> <span style="color: lightgray;">Edit</span> <span style="color: lightgray;">Delete</span> <span style="color: lightgray;">Logs</span> |
| 8 | example_short_circuit_operator          | 1 day, 00:00 | airflow | <span style="color: lightgray;">0 0 0 0 0 0</span> | <span style="color: lightgray;">2018-09-22 10:48:00 UTC</span> | 0        | <span style="color: lightgray;">Run</span> <span style="color: lightgray;">Edit</span> <span style="color: lightgray;">Delete</span> <span style="color: lightgray;">Logs</span> |
| 9 | example_skip_dag                        | 1 day, 00:00 | airflow | <span style="color: lightgray;">0 0 0 0 0 0</span> | <span style="color: lightgray;">2018-09-22 10:48:00 UTC</span> | 0        | <span style="color: lightgray;">Run</span> <span style="color: lightgray;">Edit</span> <span style="color: lightgray;">Delete</span> <span style="color: lightgray;">Logs</span> |

## Airflow Web UI in action

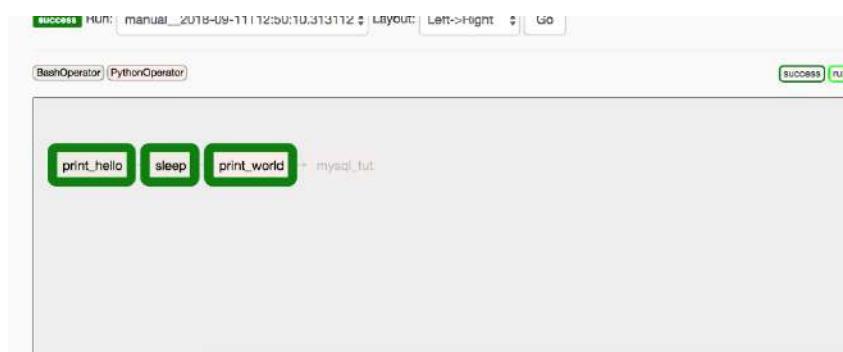
You can see a bunch of entries here. These are the example shipped with the Airflow installation. You can turn them off by visiting airflow.cfg file and set load\_examples to FALSE

**DAG Runs** tell how many times a certain DAG has been executed. **Recent Tasks** tells which task out of many tasks within a DAG currently running and what's the status of it. The **Schedule** is similar to the one you would have used when scheduling a Cron, therefore, I won't emphasize on it at the moment. The **Schedule** is responsible at what time this certain DAG should be triggered.



### DAG (Graph View)

Here is the screenshot from a DAG I created earlier and executed. You can see rectangular boxes representing a task. You can also see different color boxes on the top right of the greyed box, named: **success**, **running**, **failed** etc. These are legends. In the picture above you can all boxes have a green border, still, if you are unsure then hover your mouse on success legend and you will see a screen like below:



You might have noticed the background/filling color of these boxes which is green and red. On top-left of the greyed box, you can see why are they in such colors, these background color represents the different types of operators being used in this DAG. In this case, we are using **BashOperator** and **PythonOperator**.

### Basic Example

We will work on a basic example to see how it works. I will be explaining the example. In the dags folder which was earlier created in airflow\_home/ we will create our first sample DAG. So, I am going to create a file with name, my\_simple\_dag.py

The very first thing you are going to do after imports is to write routines that will serve as tasks for **Operators**. We will be using a mixture of BashOperator and PythonOperator.

```
import datetime as dt
from airflow import DAG
from airflow.operators.bash_operator import BashOperator
from airflow.operators.python_operator import PythonOperator

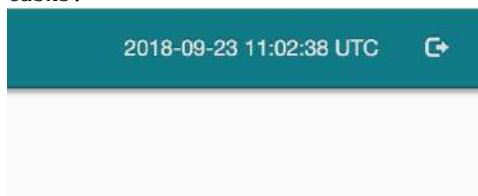
def greet():
    print('Writing in file')
    with open('path/to/file/greet.txt', 'a+', encoding='utf8') as f:
        now = dt.datetime.now()
        t = now.strftime("%Y-%m-%d %H:%M")
        f.write(str(t) + '\n')
    return 'Greeted'
def respond():
    return 'Greet Responded Again'
```

These are two simple routines which are doing nothing but returning a text. I will tell you later why am I writing something in a text file. Next things I am going to do is to define default\_args and create a DAG instance.

```
default_args = {  
    'owner': 'airflow',  
    'start_date': dt.datetime(2018, 9, 24, 10, 00, 00),  
    'concurrency': 1,  
    'retries': 0  
}
```

Here you set a bunch of parameters in the default\_args dict variable.

start\_date tells since when this DAG should start executing the workflow. This start\_date could belong to the past. In my case, it is 22 September and 11 AM UTC. This date is past for me now because it's already 11:15 AM UTC for me. You can always change this parameter via airflow.cfg file and set your own local timezone. For now, UTC is fine for me. In case you are still curious what time is being used by Airflow, check on the top right of the Airflow Web UI, you should see something like given below. You can use this as a reference to schedule your tasks.



Current time on Airflow Web UI

The retries parameter retries to run the DAG **X** number of times in case of not executing successfully. The concurrency parameter helps to dictate the number of processes needs to be used running multiple DAGs. For instance, your DAG has to run 4 past instances, also termed as [Backfill](#), with an interval of 10 minutes(I will cover this complex topic shortly) and you have set concurrency to 2 then **2 DAGs** will run at a time and execute tasks in it. If you already have implemented multiprocessing in your Python then you should feel like home here.

```
with DAG('my_simple_dag',  
        default_args=default_args,  
        schedule_interval='*/10 * * * *',  
        ) as dag:  
    opr_hello = BashOperator(task_id='say_Hi',  
                            bash_command='echo "Hi!!"')  
  
    opr_greet = PythonOperator(task_id='greet',  
                               python_callable=greet)  
    opr_sleep = BashOperator(task_id='sleep_me',  
                            bash_command='sleep 5')  
  
    opr_respond = PythonOperator(task_id='respond',  
                                python_callable=respond)  
opr_hello >> opr_greet >> opr_sleep >> opr_respond
```

Now using [Context Manager](#) we are defining a DAG with its properties, the first parameter is the ID of the dag, in our case it is my simple dag , the second parameter we already have discussed, the 3rd parameter is something that needs to be discussed along with start\_date that mentioned in default\_args.

Within that Context Manager, you are assigning operators along with task Ids. In our case these operators labeled as: opr\_hello opr\_greet opr\_sleep and opr\_respond. These names then appear in rectangular boxes discussed above.

Before I move further, I better discuss *DAG Runs* and *scheduler* and what role do they play in the entire workflow.

#### [What is Airflow Scheduler?](#)

Airflow Scheduler is a monitoring process that runs all the time and triggers task execution based on schedule interval and execution date.

#### [What is DagRun?](#)

A *DagRun* is the instance of a DAG that will run at a time. When it runs, all task inside it will be executed.



Above is the diagram which might help to figure out about a *DAGRun* :-)  
 Assume the `start_date` is **September, 24, 2018 12:00:00 PM UTC** and you have started the DAG at **12:30:00 PM UTC** with the `schedule_interval` of **\*/10 \* \* \* \* (After every 10 minutes)**. By using the same `default_args` discussed above, the following will be the entries of DAG that will run instantly, one by one in our case due to concurrency is 1:

| DAG_TIME | DAG No |
|----------|--------|
| 12:00:00 | 1      |
| 12:10:00 | 2      |
| 12:20:00 | 3      |
| 12:30:00 | 4      |

Running DAGS since the start date

Why is it happening? Well, you are responsible for it. Airflow gives you the facility to run past DAGs. The process of running past DAGs is called **Backfill**. The process of Backfill actually let Airflow forget some status of all DAGs since its inception. The feature has been given for scenarios where you are running a DAG which queries some DB or API like Google Analytics to fetch previous data and make it part of the workflow. Even if there is no past data, Airflow will run it anyway to keep the state of the entire workflow intact.

Once past DAGs are run, the next(the one you intend to run will run) at **12:40:00 PM UTC**. Do remember that whatever the schedule you set, the DAG runs AFTER that time, in our case if it has to run **after** every 10 mins, it will run once 10 minutes are passed.

Let's play with it. I turn my simple dag on and then start the scheduler.

|                                         |                          |              |         |  |
|-----------------------------------------|--------------------------|--------------|---------|--|
| <input checked="" type="checkbox"/> Off | latest_only_with_trigger | 4:00:00      | Airflow |  |
| <input checked="" type="checkbox"/> On  | my_simple_dag            | */10 * * * * | airflow |  |
| <input checked="" type="checkbox"/> Off | test_utils               | None         | airflow |  |

Starting the DAG

airflow scheduler

As soon as you run you will see the dag screen like this:

|                                         |                          |              |         |  |
|-----------------------------------------|--------------------------|--------------|---------|--|
| <input checked="" type="checkbox"/> Off | latest_only_with_trigger | 4:00:00      | Airflow |  |
| <input checked="" type="checkbox"/> On  | my_simple_dag            | */10 * * * * | airflow |  |
| <input checked="" type="checkbox"/> Off | test_utils               | None         | airflow |  |

DAG with status "Running"

Some of the tasks are queued. If you click on the DAG Id, `my_simple_dag` you will see a screen like below:

x State equals running

|   | State   | Dag Id        | Execution Date       | Run Id                              | External Trigger |
|---|---------|---------------|----------------------|-------------------------------------|------------------|
| ✓ | Running | my_simple_dag | 09-24T10:20:00+00:00 | scheduled_2018-09-24T10:20:00+00:00 | ●                |
| ✓ | running | my_simple_dag | 09-24T10:10:00+00:00 | scheduled_2018-09-24T10:10:00+00:00 | ●                |
| ✓ | running | my_simple_dag | 09-24T10:00:00+00:00 | scheduled_2018-09-24T10:00:00+00:00 | ●                |

DAGs backfilled

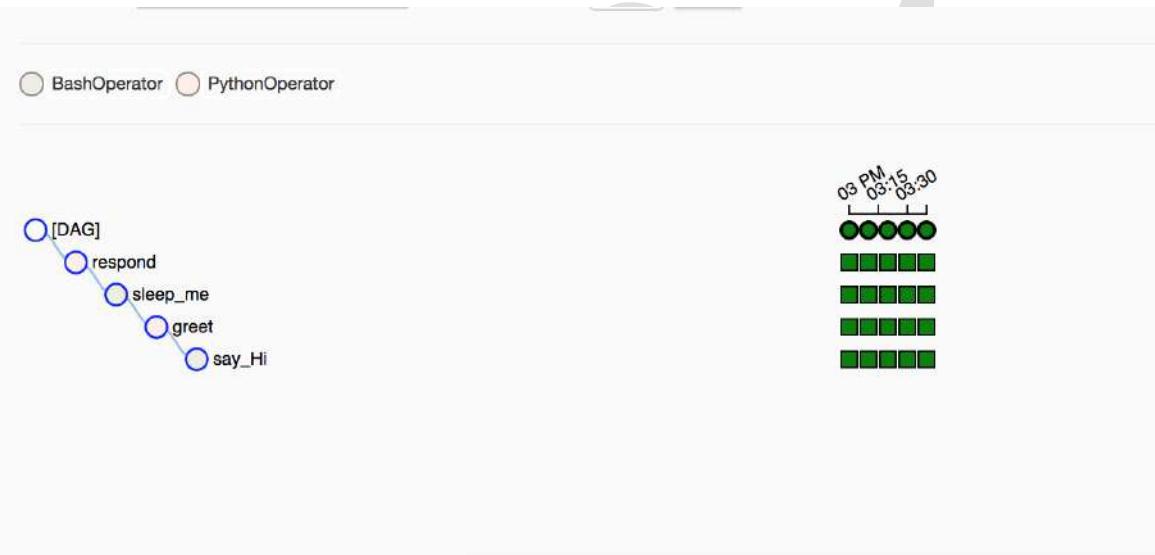
Notice the timestamp in **Run Id** column. Do you see the pattern? The first one executed at 10:00, then 10:10, 10:20. It then stops, let me clarify again that the DAG runs once the time duration which is 10minutes is passed. The scheduler started at 10:30 AM. so it filled passed **3** with the difference of **10 mins** of the interval.

x State equals success

|   | State   | Dag Id        | Execution Date       | Run Id                              | External Trigger |
|---|---------|---------------|----------------------|-------------------------------------|------------------|
| ✓ | success | my_simple_dag | 09-24T10:00:00+00:00 | scheduled_2018-09-24T10:00:00+00:00 | ●                |
| ✓ | success | my_simple_dag | 09-24T10:20:00+00:00 | scheduled_2018-09-24T10:20:00+00:00 | ●                |
| ✓ | success | my_simple_dag | 09-24T10:10:00+00:00 | scheduled_2018-09-24T10:10:00+00:00 | ●                |
| ✓ | success | my_simple_dag | 09-24T10:00:00+00:00 | scheduled_2018-09-24T10:00:00+00:00 | ●                |

DAG with Backfills and the current one

The DAG that was executed for **10:30:00 AM UTC** was actually done at **10:40:00 AM UTC**, The latest DAGRun record will always be a one minus than the current time. In our case, the machine time was **10:40:00 AM UTC**



DAG Tree View

If you hover on one of the circles you can see the timestamp in front of **Run:** that tells the time it was executed. You can see that these green circles have a time difference of 10 minutes. The Tree View gives is a bit complicated but gives a complete picture of your entire workflow. In our case, it was run 4 times and all tasks ran successfully, the dark green color.

You can avoid Backfilling in two ways: You set `start_date` of the future or set `catchup = False` in DAG instance. For instance, you can do something like below:

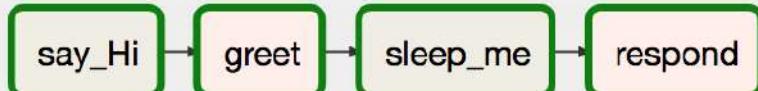
```

with DAG('my_simple_dag',
        catchup=False,
        default_args=default_args,
        schedule_interval='*/10 * * * *',
        # schedule interval=None,
        ) as dag:
  
```

By setting `catchup=False` it then does not matter whether your start date belongs to the past or not. It will be executing from the current time and continues. By setting `end_date` you can make a DAG stop running itself.

`opr_hello >> opr_greet >> opr_sleep >> opr_respond`

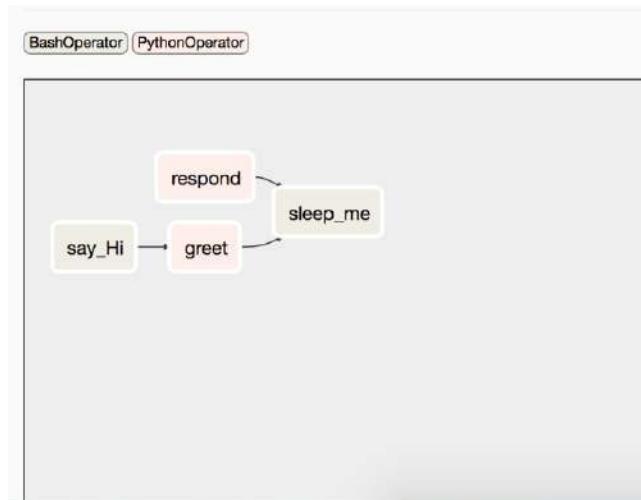
The line you are seeing above tells the relationship between operators hence constructs the entire workflow. The bitwise operator here is telling the relationship between operators. Here `opr hello` runs first and then the rest. The flow executes from left to right. In pictorial form it looks like below:



DAG In GraphView

`opr_hello >> opr_greet >> opr_sleep << opr_respond`

If you change the direction of the last operator the flow will look like below:



The `respond` task will execute in parallel and `sleep` will execute in both cases.

## Conclusion

In this post, I discussed how you can introduce a comprehensive workflow system to schedule and automate your workflows. In part 2, I will come up with a real-world example to show how Airflow can be used. I wanted to cover it up in this post but it already got enough lengthy and explaining the `DAGRun` concept was necessary as it took me quite a time to figure it out.

# Apache Airflow Installation on Ubuntu

This is the documentation of Apache Airflow installation using Ubuntu on Windows.

## Preparation pip

Apache Airflow requires `pip` for installation. Run below command.

`sudo apt-get install python-pip`

Perhaps if the default installed `pip` is not the up-to-date version, you may want to consider updating it:  
`sudo pip install --upgrade pip`

## Installing Database Backend (PostgreSQL)

Follow the steps : <https://tecadmin.net/install-postgresql-server-on-ubuntu/>

```
sudo apt-get install postgresql postgresql-contrib
Now we already have Postgres installed. Next, we need to create
bamboo@manager:~$ sudo su - postgres
postgres@manager:~$ psql
psql (10.9 (Ubuntu 10.9-0ubuntu0.18.10.1))
Type "help" for help.
postgres=# \conninfo
```

You are connected to database "postgres" as user "postgres" via socket in "/var/run/postgresql" at port "5432".

- a database for Airflow
- a user having access to the database

## How To Install phpPgAdmin on Ubuntu 16.04 and 14.04 LTS

Follow the Steps : <https://tecadmin.net/install-phppgadmin-in-ubuntu/>

### Creating Postgres User and Database

- Creating new Linux user airflow.

```
sudo adduser airflow
[sudo] password for tole:
Adding user `airflow' ...
Adding new group `airflow' (1001) ...
Adding new user `airflow' (1001) with group `airflow' ...
Creating home directory `/home/airflow' ...
Copying files from `/etc/skel' ...
Enter new UNIX password:
Retype new UNIX password:
passwd: password updated successfully
Changing the user information for airflow
Enter the new value, or press ENTER for the default
  Full Name []: airflow
  Room Number []:
  Work Phone []:
  Home Phone []:
  Other []:
Is the information correct? [Y/n] Y
```

- Use usermod to add new user airflow into the sudo group.

```
sudo usermod -aG sudo airflow
```

- Now we change the shell into user airflow.

```
su - airflow
```

- Now we use user postgres to create psql role for airflow.

```
sudo -u postgres psql
```

Note: when I run above line in Ubuntu on Windows, error was thrown like below:

```
sudo: setresuid() [1000, 113, 1000] -> [-1, 0, -1]: Operation not permitted
sudo: unable to set runas group vector: Operation not permitted
sudo: PERM_ROOT: setresuid(0, -1, 0): Operation not permitted
```

Temporary solution was using:

```
user@server:~$sudo su postgres
postgres@server:/home/user$ psql
```

Now we're in **postgres=#**

Let's create Postgres user for airflow. Still in psql console.

```
postgres=# CREATE USER airflow PASSWORD 'airfl0w';
CREATE ROLE
postgres=# CREATE DATABASE airflow;
CREATE DATABASE
postgres=# GRANT ALL PRIVILEGES ON ALL TABLES IN SCHEMA public TO airflow;
GRANT
```

```
postgres=# \du
List of roles
Role name | Attributes | Member of
-----+-----+-----+-----
```

|          |                                                            |    |
|----------|------------------------------------------------------------|----|
| airflow  |                                                            | {} |
| postgres | Superuser, Create role, Create DB, Replication, Bypass RLS | {} |

Now let's try to check whether the database already set up and can be accessed by user airflow.

```
psql -d airflow
```

```
psql (9.5.9)
```

Type "help" for help.

```
airflow=> \conninfo
```

You are connected to database "airflow" as user "airflow" via socket in "/var/run/postgresql" at port "5432".  
airflow=>

If something like above shown, then you're good.

### Change pg\_hba.conf Setting

We also need to reconfigure **pg\_hba.conf** to allow connection from airflow.

```
sudo nano /etc/postgresql/9.5/main/pg_hba.conf
```

```

# DO NOT DISABLE!
# If you change this first entry you will need to make sure that the
# database superuser can access the database using some other method.
# Noninteractive access to all databases is required during automatic
# maintenance (custom daily cronjobs, replication, and similar tasks).
#
# Database administrative login by unix domain socket
local    all      postgres          peer
#
# TYPE   DATABASE        USER        ADDRESS         METHOD
#
# "local" is for Unix domain socket connections only
local    all      all            peer
# IPv4 local connections:
host    all      all      127.0.0.1/32      md5
# IPv6 local connections:
host    all      all      ::1/128        md5
# Allow replication connections from localhost, by a user with the
# replication privilege.
#local  replication  postgres      peer
#host  replication  postgres      127.0.0.1/32  md5
#host  replication  postgres      ::1/128        md5

```

Change IPV4 address to 0.0.0.0/0 and the IPV4 method to trust.

```

# IPv4 local connections:
host all all 0.0.0.0/0 trust

```

**// Restart the service**

```
sudo service postgresql restart
```

Next, we configure **postgresql.conf**.

```

sudo nano /etc/postgresql/9.5/main/postgresql.conf
# -----
# CONNECTIONS AND AUTHENTICATION
# -----
# - Connection Settings -
listen_addresses = 'localhost' # what IP address(es) to listen on;
listen_addresses = '*' # for Airflow connection
// Restart the service
sudo service postgresql restart

```

#### Airflow Installation:

Set Up Airflow Default Home  
`export AIRFLOW_HOME=~/airflow`

#### Installing Airflow

```
pip install "[airflow[postgres, mssql, celery, rabbitmq]]"
```

```
tole@LAPTOP-NU9GHDFE:~$ pip install airflow
collecting airflow
  Downloading airflow-1.8.0.tar.gz (8.4MB)
    100% |██████████| 8.4MB 125kB/s
collecting alembic<0.9,>=0.8.3 (from airflow)
  Downloading alembic-0.8.10.tar.gz (976kB)
    100% |██████████| 983kB 460kB/s
collecting croniter<0.4,>=0.3.8 (from airflow)
  Downloading croniter-0.3.19.tar.gz
collecting dill<0.3,>=0.2.2 (from airflow)
  Downloading dill-0.2.7.1.tar.gz (64kB)
    100% |██████████| 71kB 461kB/s
collecting flask<0.12,>=0.11 (from airflow)
  Downloading Flask-0.11.1-py2.py3-none-any.whl (80kB)
    100% |██████████| 81kB 471kB/s
collecting flask-admin==1.4.1 (from airflow)
  Downloading Flask-Admin-1.4.1.tar.gz (922kB)
    100% |██████████| 931kB 369kB/s
collecting flask-cache<0.14,>=0.13.1 (from airflow)
  Downloading Flask-Cache-0.13.1.tar.gz (45kB)
    100% |██████████| 51kB 472kB/s
collecting flask-login==0.2.11 (from airflow)
  Downloading Flask-Login-0.2.11.tar.gz
collecting flask-swagger==0.2.13 (from airflow)
  Downloading flask-swagger-0.2.13.tar.gz
collecting flask-wtf==0.12 (from airflow)
  Downloading Flask_WTF-0.12-py2-none-any.whl
collecting funcsigs==1.0.0 (from airflow)
  Downloading funcsigs-1.0.0-py2.py3-none-any.whl
collecting future<0.16,>=0.15.0 (from airflow)
  Downloading future-0.15.2.tar.gz (1.6MB)
    100% |██████████| 1.6MB 371kB/s
collecting gitpython>=2.0.2 (from airflow)
  Downloading GitPython-2.1.7-py2.py3-none-any.whl (446kB)
    100% |██████████| 450kB 416kB/s
collecting gunicorn<19.4.0,>=19.3.0 (from airflow)
  Downloading gunicorn-19.3.0-py2.py3-none-any.whl (110kB)
    100% |██████████| 112kB 328kB/s
collecting jinja2<2.9.0,>=2.7.3 (from airflow)
  Downloading Jinja2-2.8.1-py2.py3-none-any.whl (264kB)
    100% |██████████| 266kB 227kB/s
collecting lxml<4.0,>=3.6.0 (from airflow)
  Downloading lxml-3.8.0-cp27-cp27mu-manylinux1_x86_64.whl (6.8MB)
    100% |██████████| 6.8MB 81kB/s
collecting markdown<3.0,>=2.5.2 (from airflow)
  Downloading Markdown-2.6.9.tar.gz (271kB)
    100% |██████████| 276kB 189kB/s
collecting pandas<1.0.0,>=0.17.1 (from airflow)
  Downloading pandas-0.20.3-cp27-cp27mu-manylinux1_x86_64.whl (22.4MB)
    100% |██████████| 22.4MB 51kB/s
collecting psutil<5.0.0,>=4.2.0 (from airflow)
  Downloading psutil-4.4.2.tar.gz (1.8MB)
    100% |██████████| 1.8MB 296kB/s
collecting pygments<3.0,>=2.0.1 (from airflow)
  Downloading Pygments-2.2.0-py2.py3-none-any.whl (841kB)
    100% |██████████| 849kB 487kB/s
collecting python-daemon<2.2,>=2.1.1 (from airflow)
  Downloading python_daemon-2.1.2-py2.py3-none-any.whl
collecting python-dateutil<3,>=2.3 (from airflow)
  Downloading python_dateutil-2.6.1-py2.py3-none-any.whl (194kB)
    100% |██████████| 194kB 645kB/s
collecting python-nvd3==0.14.2 (from airflow)
  Downloading python-nvd3-0.14.2.tar.gz
collecting requests<3,>=2.5.1 (from airflow)
  Downloading requests-2.18.4-py2.py3-none-any.whl (88kB)
    100% |██████████| 92kB 819kB/s
collecting setproctitle<2,>=1.1.8 (from airflow)
  Downloading setproctitle-1.1.10.tar.gz
collecting sqlalchemy>=0.9.8 (from airflow)
  Downloading SQLAlchemy-1.1.14.tar.gz (5.2MB)
    100% |██████████| 5.2MB 154kB/s
collecting tabulate<0.8.0,>=0.7.5 (from airflow)
  Downloading tabulate-0.7.7-py2.py3-none-any.whl
collecting thrift<0.10,>=0.9.2 (from airflow)
  Downloading thrift-0.9.3.tar.gz
collecting zope.deprecation<5.0,>=4.0 (from airflow)
  Downloading zope.deprecation-4.3.0-py2.py3-none-any.whl
```

Check [here](#) or more details on the list of the subpackages and what they enable.

## Starting Up Airflow Database

After successful packages installation, we can start Airflow's database by issuing:

```
airflow initdb
```

The command will generate `airflow.cfg` file in Airflow's home directory we set up earlier.

## Set up `airflow.cfg`

- we should use CeleryExecutor instead of SequentialExecutor if we want to run the pipeline in the webUI

```
executor = CeleryExecutor
```

- we should pass along the connection info of the postgresql database `airflow` we just created

`sql_alchemy_conn = postgresql+psycopg2://ubuntu@localhost:5432/airflow`  
Save it and run `airflow initdb`

```
airflow initdb
[2017-10-01 22:41:30,583] {__init__.py:57} INFO - Using executor celeryExecutor
[2017-10-01 22:41:30,782] {driver.py:120} INFO - Generating grammar tables from /usr/lib/python2.7/lib2to3/Grammar.txt
[2017-10-01 22:41:30,809] {driver.py:120} INFO - Generating grammar tables from /usr/lib/python2.7/lib2to3/PatternGrammar.txt
DB: postgresql+psycopg2://airflow@localhost:5432/airflow
[2017-10-01 22:41:31,181] {db.py:28} INFO - creating tables
INFO [alembic.runtime.migration] Context impl PostgresqlImpl.
INFO [alembic.runtime.migration] Will assume transactional DDL.
INFO [alembic.runtime.migration] Running upgrade e3a74de0dc1 -> 1507a2892f7, create is encrypted
INFO [alembic.runtime.migration] Running upgrade 1507a2892f7 -> 13eb5f81e27, maintain history for compatibility with earlier migrations
INFO [alembic.runtime.migration] Running upgrade 13eb5f81e27 -> 338e90f54d61, More logging into task_instance
INFO [alembic.runtime.migration] Running upgrade 338e90f54d61 -> 52d714493f0, job_id indices
INFO [alembic.runtime.migration] Running upgrade 52d714493f0 -> 50289887784, Addinfo extra to log
INFO [alembic.runtime.migration] Running upgrade 50289887784 -> 10c63411adcf0, dagrun
INFO [alembic.runtime.migration] Running upgrade 10c63411adcf0 -> 2e541a1dcfc, task_instance
INFO [alembic.runtime.migration] Running upgrade 2e541a1dcfc -> 40e67319e3a9, dagrun_config
INFO [alembic.runtime.migration] Running upgrade 40e67319e3a9 -> 561833c1c74b, add password column to user
INFO [alembic.runtime.migration] Running upgrade 561833c1c74b -> 4446608588, dagrun_start end
INFO [alembic.runtime.migration] Running upgrade 4446608588 -> bbc3705a13e, Add notification_sent column to sla_miss
INFO [alembic.runtime.migration] Running upgrade bbc3705a13e -> bba5a7fc896, Add a column to track the encryption state of the 'Extra' Field in connection
INFO [alembic.runtime.migration] Running upgrade bba5a7fc896 -> 1988acfcc09e3, add is_encrypted column to variable table
INFO [alembic.runtime.migration] Running upgrade 1988acfcc09e3 -> 2682a2033, rename user_table
INFO [alembic.runtime.migration] Running upgrade 2682a2033 -> 382aab8ef20, add task_index
INFO [alembic.runtime.migration] Running upgrade 382aab8ef20 -> 644e9ccdf6c9, add task fails journal table
INFO [alembic.runtime.migration] Running upgrade 644e9ccdf6c9 -> f2ca10b85618, add dag_stats table
INFO [alembic.runtime.migration] Running upgrade f2ca10b85618 -> 4addfa1236f1, add fractional seconds to mysql tables
INFO [alembic.runtime.migration] Running upgrade 4addfa1236f1 -> 8504051e801b, xcom dag task indices
INFO [alembic.runtime.migration] Running upgrade 8504051e801b -> 5e7d17757c7a, add pid field to TaskInstance
INFO [alembic.runtime.migration] Running upgrade 5e7d17757c7a -> 127d2bf2dfa7, Add dag_id/state index on dag_run table
Done.
$
```

## Disabling Examples

```
nano airflow/airflow.cfg
```

```
....
```

```
# Whether to load the examples that ship with Airflow. It's good to
# get started, but you probably want to set this to False in a production
# environment
load_examples = False
```

## Starting Airflow Web Server

This command will start web server in your localhost at port 8080.

| DAG                                     | Schedule | Owner   | Recent Tasks | Last Run | DAG Runs | Links |
|-----------------------------------------|----------|---------|--------------|----------|----------|-------|
| example_bash_operator                   |          | airflow |              |          |          |       |
| example_branch_dop_operator_v3          |          | airflow |              |          |          |       |
| example_branch_operator                 |          | airflow |              |          |          |       |
| example_http_operator                   |          | airflow |              |          |          |       |
| example_passing_params_via_test_command |          | airflow |              |          |          |       |
| example_python_operator                 |          | airflow |              |          |          |       |
| example_short_circuit_operator          |          | airflow |              |          |          |       |
| example_skip_dag                        |          | airflow |              |          |          |       |
| example_subdag_operator                 |          | airflow |              |          |          |       |

Next. We will cover how to integrate Airflow with Microsoft SQL Server database.

### # ! How to run bash script file in Airflow

From the tutorial this is OK:

```
#!/bin/bash
#create_file.sh

file=filename.txt

if [ ! -e "$file" ] ; then
    touch "$file"
fi

if [ ! -w "$file" ] ; then
    echo cannot write to $file
    exit 1
fi
```

```
t2 = BashOperator(
    task_id='sleep',
    bash_command='sleep 5',
    retries=3,
```

```
dag=dag)
```

But you're passing a multi-line command to it

```
create_command = """  
./scripts/create_file.sh  
"""
```

should be

```
create_command = "./scripts/create_file.sh"
```

Moreover, you also have to make sure that you are in the correct directory to avoid cryptic errors. Do it like this for example:

```
create_command = "./scripts/create_file.sh"  
if os.path.exists(create_command):  
    t1 = BashOperator(  
        task_id='create_file',  
        bash_command=create_command,  
        dag=dag  
    )  
else:  
    raise Exception("Cannot locate {}".format(create_command))
```

- add space after .sh : "./scripts/create\_file.sh" – [Karol Sudol](#) Feb 21 '17 at 10:17

- @KarolSudol can you elaborate? – [Jean-François Fabre](#) Feb 21 '17 at 10:19
- 5 sometimes you might receive error: This fails with Jinja template not found, in order to overcome add space at the end of the script, not sure what drives this behaviour:  
ref: [cwiki.apache.org/confluence/display/AIRFLOW/Common+Pitfalls](http://cwiki.apache.org/confluence/display/AIRFLOW/Common+Pitfalls) – [Karol Sudol](#) Feb 21 '17 at 10:49
- 2 @KarolSudol Its because airflow checks the end of the line you pass in and if it ends in .sh it tries to treat it as a template. A space breaks that check and it doesn't treat it as a template. I still can't figure out the why, though. – [Gregory Arenius](#) Aug 3 '17 at 19:40
- thanks for that space tip - would noooot have figured that out myself

## Learn Airflow basics



### Important Composer Concepts



#### Apache Airflow (incubating)

Create workflows as DAGs  
Airflow schedules and executes  
Ensures dependencies satisfied



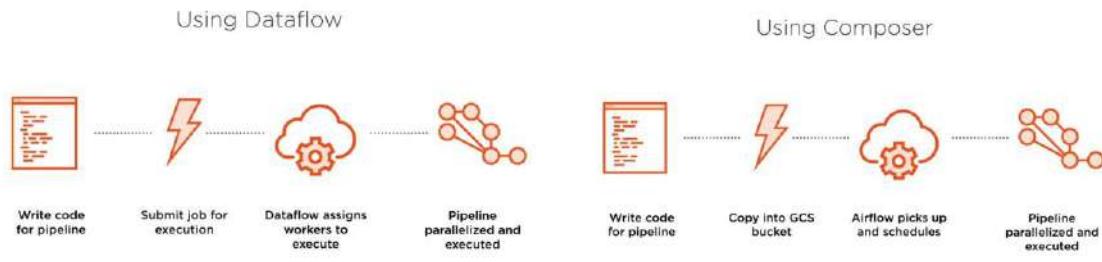
Simple Python API  
Scalable architecture  
Powerful operators  
Jinja templates



Airflow workers  
Web server to managed pipelines  
Scheduler tracks and executes DAGs  
Celery task queue to scale workers  
Redis as message broker for Celery

- Webserver is based on python flask

- Celery is a tool/technology to execution of tasks
- Redis is as message broker.
- Kubernetes is the cluster for the cluster mechanism.



### DAG:

is nothing but a workflows/pipelines of your tasks

Directed-Acyclic-Graph

Define in Airflow python script

- Must execute instantaneously
- Merely Dag Definint

[Python | web2py](#)

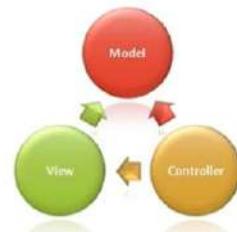
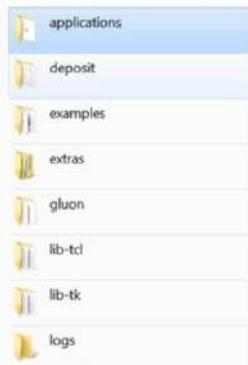
**Operators:**

| Operator                                                              | Description                                                                                                  |
|-----------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------|
| <code>**</code>                                                       | Exponentiation (raise to the power)                                                                          |
| <code>~, +, -</code>                                                  | Complement, unary plus and minus<br>(method names for the last two are <code>+@</code> and <code>-@</code> ) |
| <code>*, /, %, //</code>                                              | Multiply, divide, modulo and floor division                                                                  |
| <code>+, -</code>                                                     | Addition and subtraction                                                                                     |
| <code>&gt;&gt;, &lt;&lt;</code>                                       | Right and left bitwise shift                                                                                 |
| <code>&amp;</code>                                                    | Bitwise 'AND'                                                                                                |
| <code>^</code>                                                        | Bitwise exclusive 'OR'                                                                                       |
| <code> </code>                                                        | Bitwise 'OR'                                                                                                 |
| <code>in, not in, is, is not, &lt;, &lt;=, &gt;, &gt;=, !=, ==</code> | Comparison operators, equality operators, membership and identity operators                                  |
| <code>not</code>                                                      | Boolean 'NOT'                                                                                                |
| <code>and</code>                                                      | Boolean 'AND'                                                                                                |
| <code>or</code>                                                       | Boolean 'OR'                                                                                                 |
| <code>=, %=, /=, //=, -=, +=, *=, **=</code>                          | Assignment operators                                                                                         |



Web2py is inbuild with rocket webserver, browser based integrated environment, inbuild SQLite.

## MVC – File Structure - URL



URL Mapping:

<http://server/app/cntr/view/args>

All structure will be stored in application directory

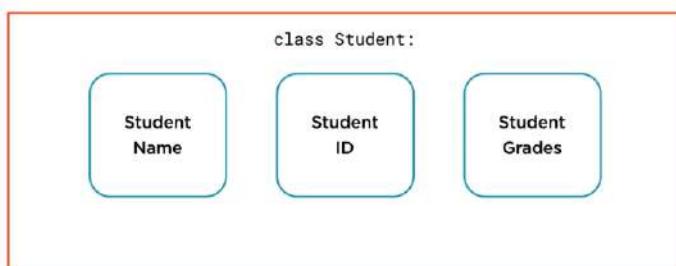


### Classes why we need them ?

A Class : a logical group of method and contains data

Data stored in dictionary

A Class



Function is what is really does for the class.

Define a class in python is so easy

Class student:

Instance will be created for the class

Building blocks are students()

Student = student()

Creating an instance for a class.

student = Student()

Creating a new instance of a Student class

They did not depend on each other.

## Class Instances

```
mark = Student("Mark")
james = Student("James")
jessica = Student("Jessica")

mark != james != jessica
```

The screenshot shows the PyCharm IDE interface with a project named 'PluralsightCourse' containing files like classes.py, exceptions.py, functions.py, loops.py, students.txt, and yield.py. The current file is classes.py, which contains the following code:

```
class Student:
    pass

student = Student()
print(student)
```

Annotations with arrows explain the code execution:

- An arrow points from the line "class Student:" to the text "Class 'Student'".
- An arrow points from the line "student = Student()" to the text "instance created for the class".
- An arrow points from the line "print(student)" to the output window below, which shows the memory location of the object.

The output window shows the result of running the script:

```
/Users/Deusdies/venvs/pluralsightgettingstarted/bin/python /Users/Deusdies/Work/Projects/PluralsightCourse/classes.py
<__main__.Student object at 0x1084e0270>
Process finished with exit code 0
```

A callout box highlights the memory location "0x1084e0270" in the output with the text "Memory location".

The screenshot shows the PyCharm IDE interface. On the left, the project structure for 'PluralsightCourse' is visible, containing files like classes.py, exceptions.py, functions.py, loops.py, students.txt, and yield.py. In the center, the code editor for 'classes.py' is open, showing the following Python code:

```
1 class Student:
2     pass
3
4     student = Student()
5
6     print(student)
7
8     new_student = Student()
9
10    print(new_student)
```

Annotations with arrows point from the two 'Student()' calls to the explanatory text 'individual Instances created for a class'. Below the code editor, the 'Run' tool window shows the output of running the script:

```
/Users/Deusdies/venvs/pluralsightgettingstarted/bin/python /Users/Deusdies/Work/Projects/PluralsightCourse/classes.py
<__main__.Student object at 0x108d8c278>
<__main__.Student object at 0x108d8c2d0>
```

Annotations with arrows point from the two output lines to the explanatory text 'Stored in memory, reason still the values are not assigned to instances of class Student'.

### Adding-Methods-to-Our-Class

The screenshot shows the PyCharm IDE interface. The project structure for 'PluralsightCourse' is visible, with 'functions.py' selected. In the code editor for 'classes.py', the code has been modified to include a method:

```
1 students = []
2
3
4 class Student:
5     def add_student(self, name, student_id=332):
6         student = {"name": name, "student_id": student_id}
7         students.append(student)
8
9
10 student = Student()
11 student.add_student("Mark")
12
13 print(students)
```

The 'Run' tool window shows the output of running the script:

```
/Users/Deusdies/venvs/pluralsightgettingstarted/bin/python /Users/Deusdies/Work/Projects/PluralsightCourse/classes.py
[{"name": "Mark", "student_id": 332}]
```

Annotations with arrows point from the 'print(students)' call and its output to the explanatory text 'Stored in memory, reason still the values are not assigned to instances of class Student'.

```

1 students = []
2
3
4 class Student:
5     def add_student(self, name, student_id=332):
6         student = {"name": name, "student_id": student_id}
7         students.append(student)
8
9
10 student = Student()
11 student.add_student("Mark")
12
13 print(students)
14
15
16

```

Run → classes

```

$ /Users/Deusdies/venvs/pluralsightgettingstarted/bin/python /Users/Deusdies/Work/Projects/PluralsightCourse/classes.py
[{"name": "Mark", "student_id": 332}
Process finished with exit code 0

```

Constructors and other special methods.

`_init_`

```

1 students = []
2
3
4 class Student:
5     def add_student(self, name, student_id=332):
6         student = {"name": name, "student_id": student_id}
7         students.append(student)
8
9
10 student = Student()  # Constructor call
11 student.add_student("Mark")  # Method call
12
13
14
15
16

```

Run → classes

```

$ /Users/Deusdies/venvs/pluralsightgettingstarted/bin/python /Users/Deusdies/Work/Projects/PluralsightCourse/classes.py
[{"name": "Mark", "student_id": 332}
Process finished with exit code 0

```

```

1 students = []
2
3
4 class Student:
5     def __init__(self, name, student_id=332):
6         student = {"name": name, "student_id": student_id}
7         students.append(student)
8
9
10
11
12
13
14
15
16

```

Run → classes

```

$ /Users/Deusdies/venvs/pluralsightgettingstarted/bin/python /Users/Deusdies/Work/Projects/PluralsightCourse/classes.py
[{"name": "Mark", "student_id": 332}
Process finished with exit code 0

```

In background constructor method will call.

#### Solaris - checking whether an account is locked or not

Recently we needed to be able to check what the account status on one of the servers was as there was an access problem.

Was it locked out? How do we check from a terminal session?

```
# passwd -s <account_name>
```

This comes back with the accounts status. In this instance it came back with:

```
# <account_name> LK
```

Status information:

PS = a normal working account.

LK = locked out account.

NP = account has no password.

Okay, so the account is locked. How do I unlock it?

```
# passwd -u <account_name>
```

Account is now unlocked - now to find the script that locked the account in the first place.....

NOTE: If you want to lock the account on purpose

```
# passwd -l <account_name>
```

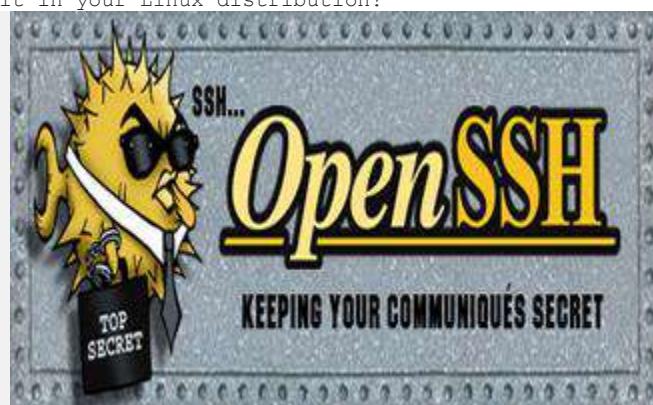
#### OpenSSH :

##### How to Install and Configure OpenSSH Server In Linux

by Oltjano Terpollari | Published: November 13, 2013 | Last Updated: January 7, 2015

Download Your Free eBooks NOW - [10 Free Linux eBooks for Administrators](#) | [4 Free Shell Scripting eBooks](#)

Being a network administrator requires a deep knowledge about remote login protocols such as rlogin, telnet and ssh. The one I will discuss in this article is ssh, a secure remote protocol which is used to work remotely on other machines or transfer data between computers using [SCP \(Secure Copy\)](#) command. But, what is OpenSSH and how to install it in your Linux distribution?



*Install OpenSSH in Linux*

#### What is OpenSSH?

OpenSSH is a free open source set of computer tools used to provide secure and encrypted communication over a computer network by using the ssh protocol. Many people, new to computers and protocols, create a misconception about OpenSSH, they think it is a protocol, but it is not, it is a set of computer programs that use the ssh protocol.

OpenSSH is developed by the Open BSD group and it is released under Simplified BSD License. A main factor which has made possible for OpenSSH to be used so much among system administrators is its multi-platform capability and very useful nice features it has. The latest version is OpenSSH 6.4 which has been released on November 8, 2013.

This version of OpenSSH comes with many new features and patches, so if you already use OpenSSH for administering your machines, I suggest you to do an upgrade.

#### **Why Use OpenSSH And Over Telnet Or Ftp?**

The most important reason why should use OpenSSH tools over ftp and telnet is that all communications and user credentials using OpenSSH are encrypted, they are also protected from man in the middle attacks. If a third party tries to intercept your connection, OpenSSH detects it and informs you about that.

#### **What Are Some Of The OpenSSH Features?**

1. Secure Communication
2. Strong Encryption (3DES, Blowfish, AES, Arcfour)
3. X11 Forwarding (encrypt X Window System traffic)
4. Port Forwarding (encrypted channels for legacy protocols)
5. Strong Authentication (Public Key, One-Time Password and Kerberos Authentication)
6. Agent Forwarding (Single-Sign-On)
7. Interoperability (Compliance with SSH 1.3, 1.5, and 2.0 protocol Standards)
8. SFTP client and server support in both SSH1 and SSH2 protocols.
9. Kerberos and AFS Ticket Passing
10. Data Compression

#### **Installation of OpenSSH in Linux**

To install OpenSSH, open a terminal and run the following commands with superuser permissions.

##### **On Ubuntu/Debian/Linux Mint**

```
$ sudo apt-get install openssh-server openssh-client
```

##### **On RHEL/Centos/Fedora**

Type the following yum command to install openssh client and server.

```
# yum -y install openssh-server openssh-clients
```

#### **Configuration of OpenSSH**

It's time to configure our OpenSSH behaviour through the ssh config file, but before editing the /etc/ssh/sshd config file we need to backup a copy of it, so in case we make any mistake we have the original copy.

Open a terminal and run the following command to make a copy of the original sshd configuration file.

```
$ sudo cp /etc/ssh/sshd_config /etc/ssh/sshd_config.original_copy
```

As you can see from the command I typed, I added the original copy suffix, so every time I see this file I know it is an original copy of the sshd config file.

#### How Do I Connect to OpenSSH

Before we go further, we need to verify if our openssh server is working or not. How to do that? You can try to connect to the openssh server from your localhost through your openssh client or do a portscanwith nmap, but I like to use a small tool called netcat, also known as the TCP/IP Swiss army knife. I love working with this amazing tool on my machine, so let me show it to you.

```
# nc -v -z 127.0.0.1 22
```

Referring to the netcat results, the ssh service is running on port 22 on my machine. Very good! What if we want to use another port, instead of 22? We can do that by editing the sshd configuration file.

Set your OpenSSH to listen on TCP port 13 instead of the default TCP port 22. Open the `sshd_config` file with your favourite text editor and change the port directive to 13.

```
# What ports, IPs and protocols we listen for
```

```
Port 13
```

Restart OpenSSH server so the changes in config file can take place by typing the following command and run netcat to verify if the port you set for listening is open or not.

```
$ sudo /etc/init.d/ssh restart
```

Should we verify is our openssh server is listening on port 13, or not?. This verification is necessary, so I am calling my lovely tool netcat to help me do the job.

```
# nc -v -z 127.0.0.1 13
```

Do you like to make your openssh server display a nice login banner? You can do it by modifying the content of `/etc/issue.net` file and adding the following line inside the sshd configuration file.

## Conclusion

There are many things you can do with the openssh tools when it comes to the way you configure your openssh server, I can say that your imagination is the limit!.

Read Also: [5 Best Practices to Secure and Protect OpenSSH Server](#)

### Playing with firewall:

```
service iptables stop  
service iptables status  
service iptables start
```

How to Disable the Firewall for Oracle Linux or Red Hat Enterprise Linux

1. Stop the ipchains service:

```
# service ipchains stop
```

2. Stop the iptables service:

```
# service iptables stop
```

3. Stop the ipchains service from starting when you restart the server:

```
# chkconfig ipchains off
```

4. Stop the iptables service from starting when you restart the server:

```
# chkconfig iptables off
```

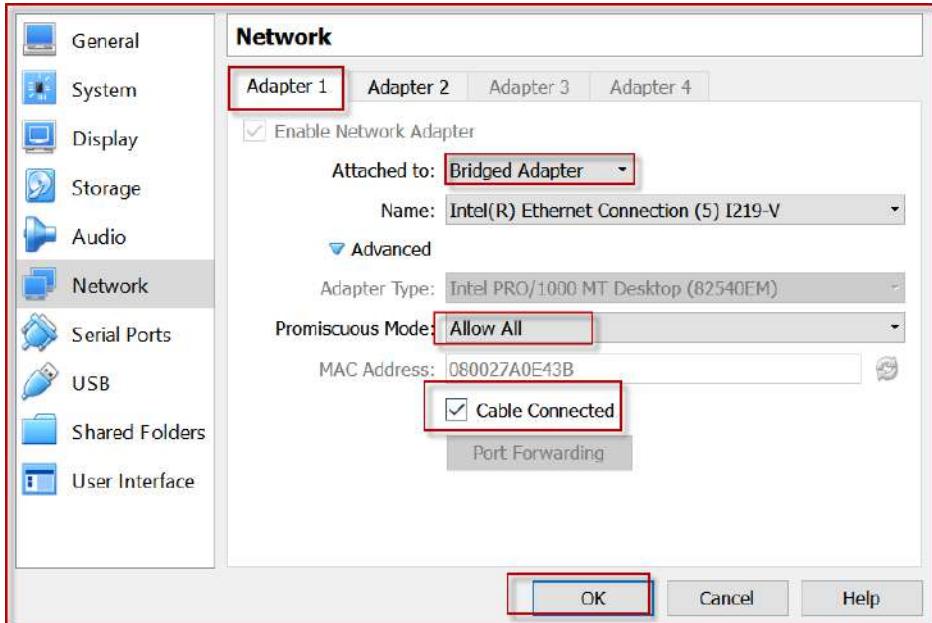
[VirtualBox: How to set up networking so both host and guest can access internet and talk to each other](#)

The best way to do this is to use a Bridge Adapter in virtualbox. In virtual box go to the settings for your machine->Network->Adapter 1 and select Bridged Adapter. This will make your virtual machine part of your main network.

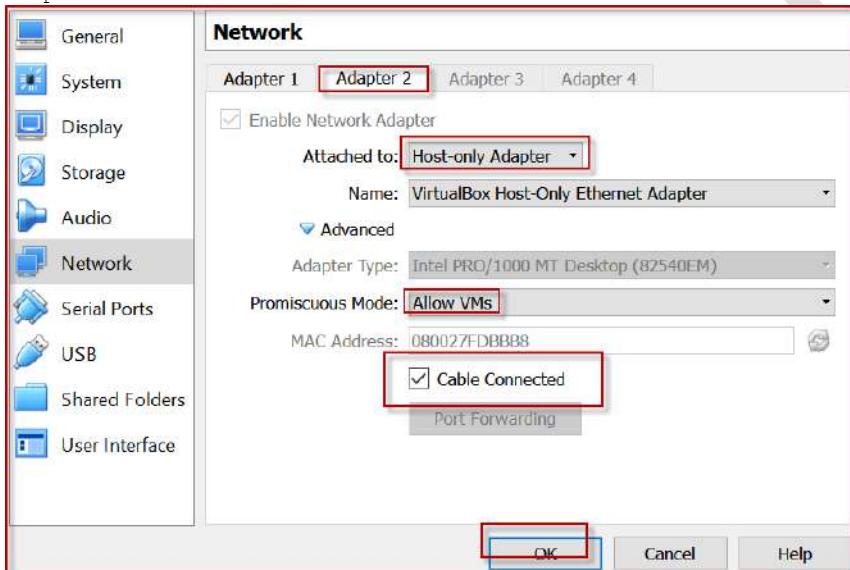
If you have a dhcp server it should supply an address etc. to the virtual machine which will allow it to communicate with the rest of your systems and vice versa.

If you want to connect the virtualbox from host, Please make sure to configure with below settings.

Adapter 1:



Adapter 2:



In virtual box operating system ip: ifconfig

```
[root@BRM75 Desktop]# ifconfig
eth2      Link encap:Ethernet HWaddr 08:00:27:FD:BB:B8
          inet addr:192.168.56.101 Bcast:192.168.56.255 Mask:255.255.255.0
          inet6 addr: fe80::a00:27ff:fed:b8/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:9159 errors:0 dropped:0 overruns:0 frame:0
          TX packets:59 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:909214 (887.9 KiB) TX bytes:18594 (18.1 KiB)

eth3      Link encap:Ethernet HWaddr 08:00:27:A0:E4:3B
          inet6 addr: fe80::a00:27ff:fea0:e43b/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST MTU:1500 Metric:1
          RX packets:188 errors:0 dropped:0 overruns:0 frame:0
          TX packets:124 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:24523 (23.9 KiB) TX bytes:14102 (13.7 KiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1 Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING MTU:65536 Metric:1
          RX packets:96 errors:0 dropped:0 overruns:0 frame:0
          TX packets:96 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:7576 (7.3 KiB) TX bytes:7576 (7.3 KiB)

virbr0    Link encap:Ethernet HWaddr 52:54:00:77:3A:B5
          inet addr:192.168.122.1 Bcast:192.168.122.255 Mask:255.255.255.0
          UP BROADCAST MULTICAST MTU:1500 Metric:1
          RX packets:0 errors:0 dropped:0 overruns:0 frame:0
          TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)
```

Now try to ping the virtual box ip from host cmd.

```
Microsoft Windows [Version 10.0.14393]
(c) 2016 Microsoft Corporation. All rights reserved.

C:\Users\80053806>ping 192.168.56.101

Pinging 192.168.56.101 with 32 bytes of data:
Reply from 192.168.56.101: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.56.101:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Users\80053806>
```

## ZABBIX

Hello, the zabbix-server works only on Linux/Unix machines. The zabbix-agent works on windows!

Zabbix-server: Unix/Linux  
Zabbix-agent: Windows/Linux

## ssh-passwordless-login-using-ssh-keygen-in-5-easy-steps

SSH (Secure SHELL) is an open source and most trusted network protocol that is used to login into remote servers for execution of commands and programs. It is also used to transfer files from one computer to another computer over the network using secure copy (SCP) Protocol.

In this article we will show you how to setup password-less login on RHEL/CentOS

7.x/6.x/5.x and Fedora using ssh keys to connect to remote Linux servers without entering password. Using

Password-less login with SSH keys will increase the trust between two Linux servers for easy file synchronization or transfer.

#### Setup SSH Passwordless Login

My Setup Environment

SSH Client : 192.168.0.12 ( Fedora 21 )  
SSH Remote Host : 192.168.0.11 ( CentOS 7 )

If you are dealing with number of Linux remote servers, then SSH Password-less login is one of the best way to automate tasks such as automatic backups with scripts, synchronization files using scp and remote command execution.

In this example we will setup SSH password-less automatic login from server 192.168.0.12 as user tecmint to 192.168.0.11 with user sheena.

#### Step 1: Create Authentication SSH-Kegen Keys on - (192.168.0.12)

First login into server 192.168.0.12 with user tecmint and generate a pair of public keys using following command.

```
[tecmint@tecmint.com ~]$ ssh-keygen -t rsa
```

Generating public/private rsa key pair.

Enter file in which to save the key (/home/tecmint/.ssh/id\_rsa): [Press enter key]  
Created directory '/home/tecmint/.ssh'.

Enter passphrase (empty for no passphrase): [Press enter key]

Enter same passphrase again: [Press enter key]

Your identification has been saved in /home/tecmint/.ssh/id\_rsa.

Your public key has been saved in /home/tecmint/.ssh/id\_rsa.pub.

The key fingerprint is:

5f:ad:40:00:8a:d1:9b:99:b3:b0:f8:08:99:c3:ed:d3 tecmint@tecmint.com

The key's randomart image is:

+--[ RSA 2048]----+

```
| ..ooE.++|  
| o. o.o |  
| .. . |  
| o . . o|  
| S . . + |  
| . . . o|  
| . o o ..|  
| + + |  
| +. |  
+---+
```

#### Step 2: Create .ssh Directory on - 192.168.0.11

Use SSH from server 192.168.0.12 to connect server 192.168.0.11 using sheena as user and create .ssh directory under it, using following command.

```
[tecmint@tecmint ~]$ ssh sheena@192.168.0.11 mkdir -p .ssh
```

The authenticity of host '192.168.0.11 (192.168.0.11)' can't be established.

RSA key fingerprint is 45:0e:28:11:d6:81:62:16:04:3f:db:38:02:1a:22:4e.

Are you sure you want to continue connecting (yes/no)? yes

Warning: Permanently added '192.168.0.11' (ECDSA) to the list of known hosts.

sheena@192.168.0.11's password: [Enter Your Password Here]

Step 3: Upload Generated Public Keys to - 192.168.0.11

Use SSH from server 192.168.0.12 and upload new generated public key (id\_rsa.pub) on server 192.168.0.11 under sheena's .ssh directory as a file name authorized\_keys.

```
[tecmint@tecmint ~]$ cat .ssh/id_rsa.pub | ssh sheena@192.168.0.11 'cat >> .ssh/authorized_keys'
```

sheena@192.168.1.2's password: [Enter Your Password Here]

Step 4: Set Permissions on - 192.168.0.11

Due to different SSH versions on servers, we need to set permissions on .ssh directory and authorized\_keys file.

```
[tecmint@tecmint ~]$ ssh sheena@192.168.0.11 "chmod 700 .ssh; chmod 640 .ssh/authorized_keys"
```

sheena@192.168.0.11's password: [Enter Your Password Here]

Step 5: Login from 192.168.0.12 to 192.168.0.11 Server without Password

From now onwards you can log into 192.168.0.11 as sheena user from server 192.168.0.12 as tecmintuser without password.

```
[tecmint@tecmint ~]$ ssh sheena@192.168.0.11
```

## Solaris Admin provided steps for ssh-keygen:

ssh-keygen -t rsa

enter -

cd ~/.ssh

cat rsa.pub → Public key

copy content paste it in below location

destination server --

`~/.ssh/authorized_keys` -- create folder and file if not exist and make sure folder has 700 permission on source and destination.

## How to configure Password-less authentication from Windows to Linux using Putty

I have already written few articles to [configure passwordless authentication between two Linux servers](#), here in this article I will share the steps to configure password less login from Windows to Linux box using Putty

My setup:

=====

**Windows Box: 192.168.1.2**

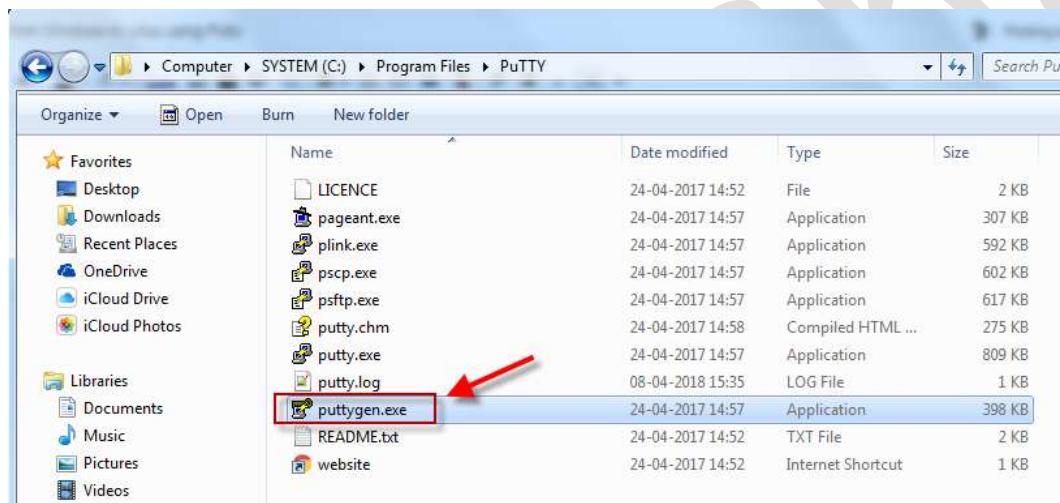
**Linux Box: 192.168.1.5**

=====

You can download and install Putty from their [official page](#) based on your Windows environment.

My Putty files are available at C:\Program Files\PuTTY as below

Here double click on "puttygen.exe" to open this tool which can be used to generate new private and public key



This tool is similar to Linux based ssh-keygen, next in the puttygen tool, select the key type and bit size which you would like use for the generated key.

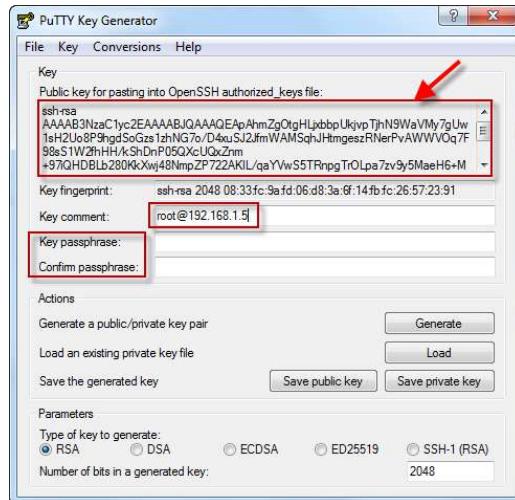
Here for my example I am using **2048** bit size **RSA-2** key



Once you click on "Generate" both private and public key will be generated.

Next change the comment to something which can be more meaningful and you can remember in future, like for me I have used `root@192.168.1.5` since I plan to use this key for root user for my Linux node **192.168.1.5**

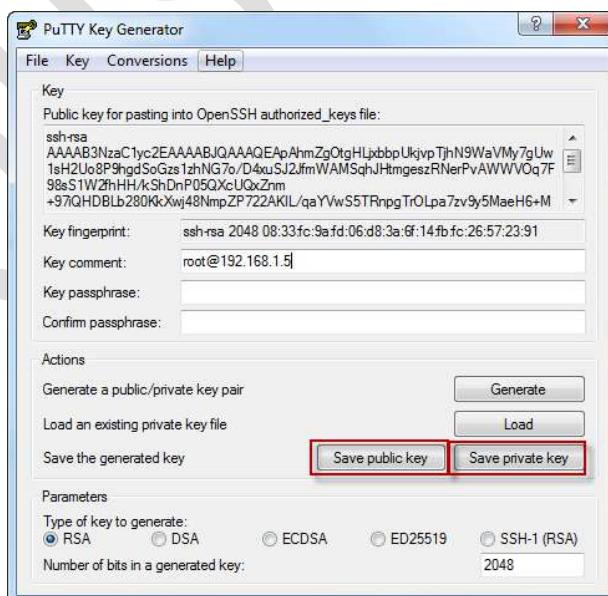
Since we want to perform a password less authentication hence I will leave the password section blank, in case if you wish to assign a passphrase then provide the same here



Next save the private key to your machine.

I am saving it by below name

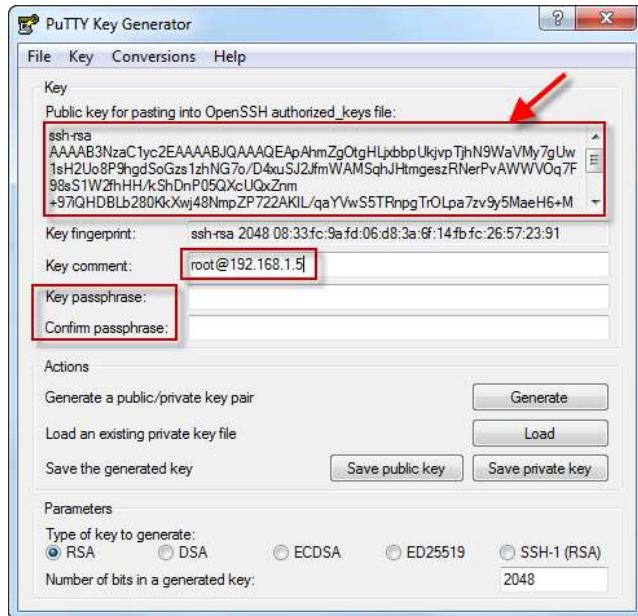
`golinuxhub-server-id_rsa`



Now it is time to copy the public key to the Linux box

Since I want to use this for root user I will append the public key to root user's `authorized_keys`

The public key can be seen in the highlighted section, copy this and append the key to `/root/.ssh/authorized_keys` on 192.168.1.5 (my Linux box)



## On my Linux box

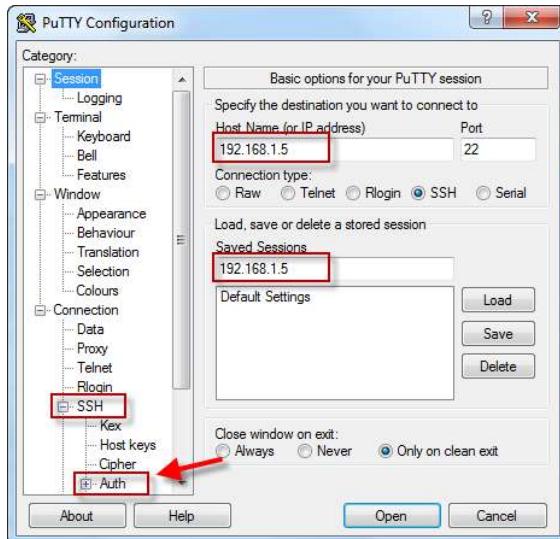
**NOTE:** If `/root/.ssh` does not exists you can manually create this

```
# mkdir /root/.ssh  
# chmod 700 /root/.ssh
```

as you see below I have appended my public key to the root user's `authorized_keys` file

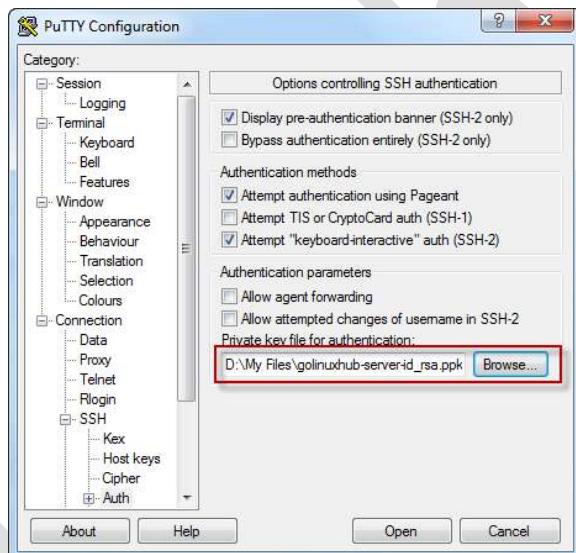
```
[root@golinuxhub-server root :~]# ls -ld .ssh/  
drwx----- 2 root root 80 Apr  8 16:18 .ssh/  
  
[root@golinuxhub-server root :~]# cat .ssh/authorized_keys  
ssh-rsa  
AAAAB3NzaC1yc2EAAAABJQAAQEApAhmZg0tgHLjxbbpUkjvpTjhN9WaVMy7gUw1sH2Uo8P9hgdSoGzs1zhNG7o/D4xuSJ2Jfm  
WAMsQhJHtmgeszRNerPvAkwVOq7F98sS1w2fhHH/kShDnP05QXcUQxZnm+97iQHDBLb280KkXwj48NmpZP722AKIL/qaYVwS5T  
RnpgTrOLpa7zv9y5MaeH6+MmMmV6n0U5A8+LFF0qYPQYJ4oSFolA0anHJDDM9Gqw3JXV2mgyEOR7+H6R36LE01Fy5kgSBSX8Kj  
GyF6VsssiUF1aDWsB3KE03eHRuRIfBFMTXV0K+rjFLSqnVj/m0/+pLx9JngYaGSFZNEHe4/4Irw== root@192.168.1.5
```

Now since everything is done it is time to create and save a session for my Linux box in the Putty with the private key. Open Putty and add the Linux box IP as shown below

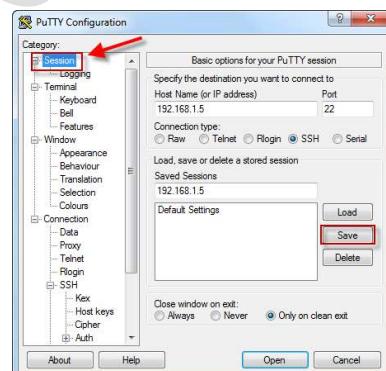


Navigate to **SSH -> Auth**

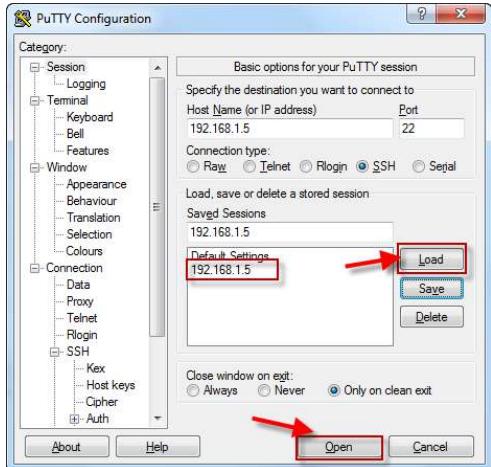
and **Browse** to the private key which was created above



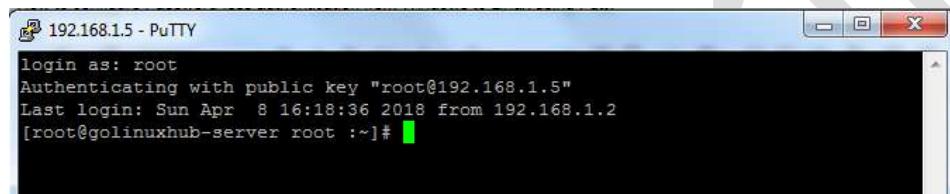
Next navigate back to "Session" as shown below and "save" the changes



This will create a new shortcut with the host name as shown below. Select the hostname and click on **Load**, next click on **"Open"** to open a new connection

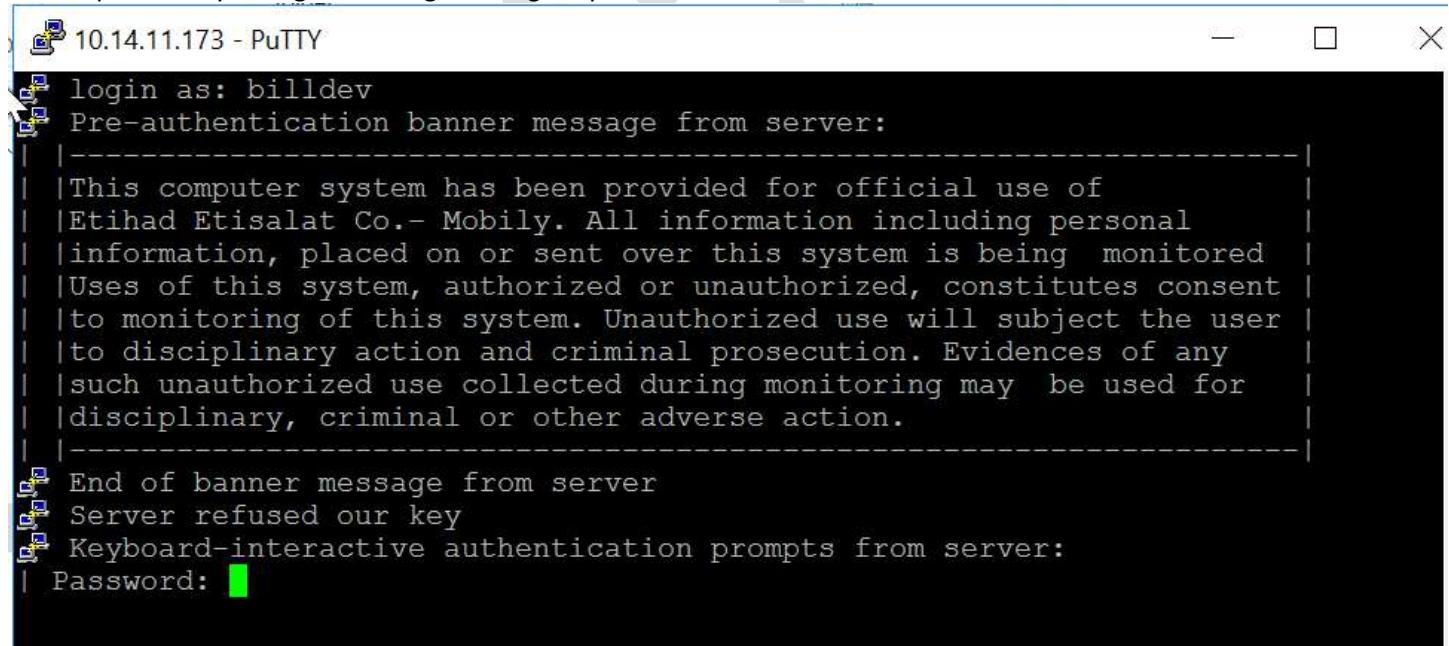


Next attempt to login using root, as you see it does not prompts for password this time and logs in without and passphrase



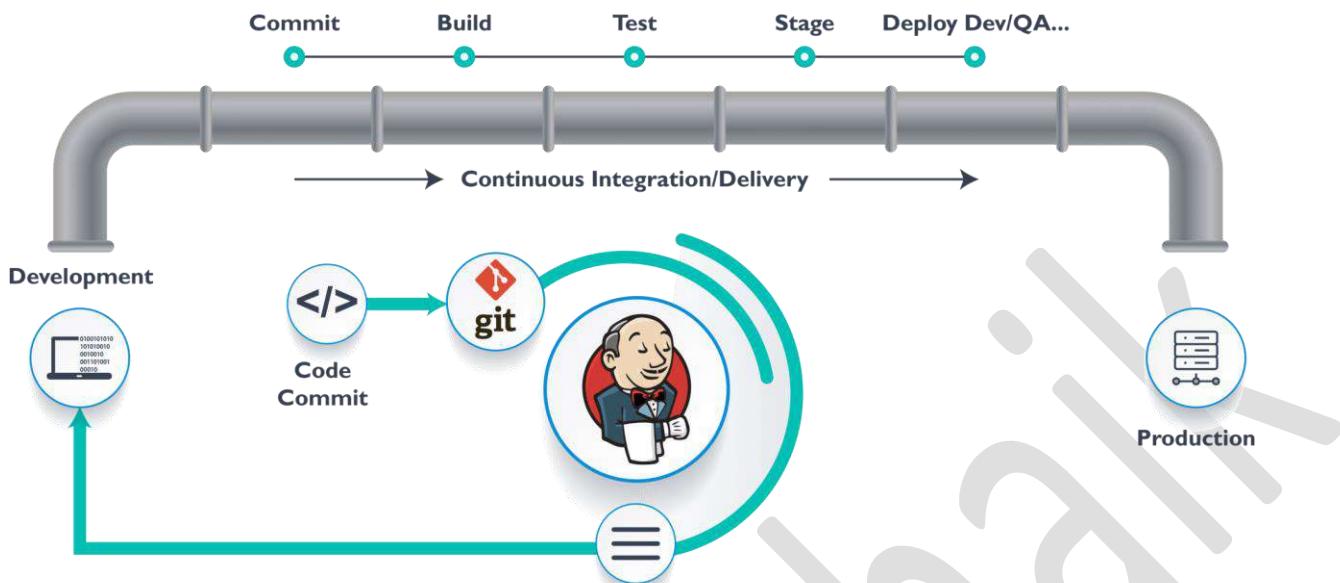
I hope the article was useful.

Issue : private key configuration login asking for password.



# Jenkins

Install java <https://www.java.com/en/download/manual.jsp>



## Jenkins installation on windows

If you are unable to find the secret file, please uninstall and re-install Jenkins again. Else, delete the .jenkins folder in C:\Users\...\jenkins\secrets.

=====

Are you using jenkins.war or jenkins.exe to install? If you are using jenkins.exe, try to download jenkins.war from Jenkins war.

Open Command Prompt, run `java -jar jenkins.war`. Click on the images to see the output

Open browser with URL `localhost:8080/`

**Error: java.io.IOException: Failed to start Jetty**

The cause of your error:

Caused by: java.net.BindException: Address already in use: bind

Explanation: As a default, Jenkins uses the port 8080 (like many other web applications do, too). Obviously, you already have another application listening on port 8080.

So you should bind Jenkins to another port. See Starting and Accessing Jenkins for more information. For example, you can start with

`java -jar jenkins.war --httpPort=8081`

open with Administration-Rights →

`cd C:\GouseShaik\Jenkins>jenkins.exe install`

`2019-08-19 15:47:38,644 INFO - Installing the service with id 'jenkins'`

## Jenkins now working unfortunately

Make sure to take the backup of existing jenkins installed directory every time you setup newly.

Now install the software freshly.

And change the port in settings.xml file.

Test the url with newly configured port.

Now Please override the backup copy of Jenkins content in freshly installed directory.

And restart the service from services.msc and test it.

## Error 1067: The process terminated unexpectedly



Services

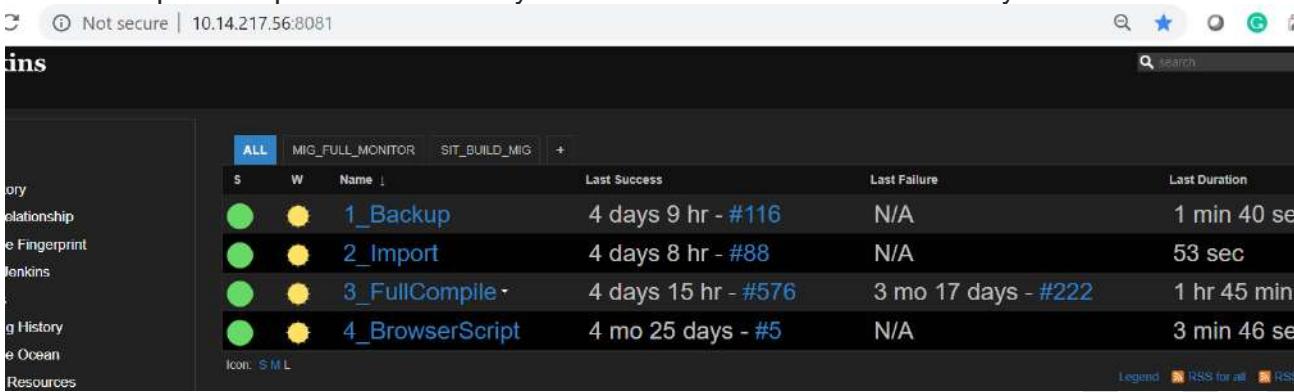
Windows could not start the Jenkins service on Local Computer.  
Error 1067: The process terminated unexpectedly.

OK

**Solution:**

With the help of jenkins.msi software, I have repair the software. Then worked fine.

After the repair completes successfully it will start the service automatically.



The screenshot shows the Jenkins dashboard with a sidebar on the left containing links like 'Category', 'Relationship', 'Fingerprint', 'Jenkins', 'History', 'Ocean', and 'Resources'. The main area displays a table of jobs under the 'ALL' tab. The columns are 'S' (Status), 'W' (Workdir), 'Name', 'Last Success', 'Last Failure', and 'Last Duration'. There are four jobs listed: '1\_Backup' (last success 4 days 9 hr - #116, last failure N/A, duration 1 min 40 sec), '2\_Import' (last success 4 days 8 hr - #88, last failure N/A, duration 53 sec), '3\_FullCompile' (last success 4 days 15 hr - #576, last failure 3 mo 17 days - #222, duration 1 hr 45 min), and '4\_BrowserScript' (last success 4 mo 25 days - #5, last failure N/A, duration 3 min 46 sec). A legend at the bottom indicates 'Icon: S M L' and RSS feeds for all and Jenkins.

## How to install a plugin in Jenkins manually

Yes, you can. Download the plugin (\*.hpi file) and put it in the following directory:

`<JenkinsHome>/plugins/`

Afterwards you will need to restart Jenkins.

## BUILDING PIPELINES BY LINKING JENKINS JOBS

9. November 2011 | Clement Escoffier

Continuous integration servers have become a corner stone of any professional development environment. By letting a machine integrate and build software, developers can focus on their tasks: fixing bugs and developing new features. With the emergence of trends such as continuous deployment and delivery, the continuous integration server is not limited to integrating your products, but has become a central piece of infrastructure.

However, organizing jobs on the CI server is not always easy.

This blog post describes a couple of strategies for creating dependent tasks with Jenkins (or Hudson).

### On keeping things small

To make your continuous server really efficient for your team, you need to give feedback to your development team as fast as possible. After a commit (or a push), we all wait for a notification to make sure we didn't introduce any obvious bugs (at some point, we've all forgotten to add a file to the SCM, or introduced a wrong import statement). However, builds tend to be long for enterprise applications; tests (both unit and smoke) can take hours to execute. So it's imperative to reduce feedback time and therefore, to divide massive jobs into small and fast units.

For example, a regular build process would generally take the following actions (for each module/component):

- compile the code
- run unit tests
- package the code
- run integration tests
- extract quality metric
- generate reports and documentation
- deploy the package on a server

It's clear that carrying out all those tasks on a multi-module project can take a considerable length of time, leaving the development team waiting before they can switch to the next task. It's not rare to see a Maven build taking (just for the compilation / test / packaging) up to 30 minutes.

Moreover, smaller jobs offer much more flexibility. For example, one can restart from a failing step without restarting the full build from scratch.

## A true story : Restarting a test server

Recently, in one of our projects, we had to clean up and re-populate a test server. In the first version, a script was executing the following actions in one massive job:

- Stop the server
- Clean up the file system
- Drop tables
- Create tables and populate with test data
- Start the server

The process was taking more or less 10 minutes, and in case of failure didn't allow restarts from the failing step.

In a second version, each action was executed in its own job. However, jobs were not dependent on each other, which required job/wait cycles of 10 minutes.

| S | W | Name ↓                                                                  |
|---|---|-------------------------------------------------------------------------|
|   |   | <a href="#">Pipeline Blog Post - Cleanup</a>                            |
|   |   | <a href="#">Pipeline Blog Post - Creating tables and Inserting data</a> |
|   |   | <a href="#">Pipeline Blog Post - Dropping tables</a>                    |
|   |   | <a href="#">Pipeline Blog Post - Start server</a>                       |
|   |   | <a href="#">Pipeline Blog Post - Stop Server</a>                        |

scenario

Finally, we linked jobs together to trigger the whole process by simply starting the first job. To create those dependencies, we tried three approaches of which two were successful (I'll let you guess about the third one).

### One-to-One Relationship

The first approach was pretty simple: one job triggers another one after successful completion. So in our case: Stop server

```
|-> Clean up filesystem
    |-> Drop database
        |-> Create table and insert data
            |-> Start server
1   INFO: Pipeline Blog Post - Stop Server #3 main build action completed: SUCCESS
2   07.11.2011 15:12:20 hudson.model.Run run
3   INFO: Pipeline Blog Post - Cleanup #3 main build action completed: SUCCESS
4   07.11.2011 15:15:27 hudson.model.Run run
5   INFO: Pipeline Blog Post - Dropping tables #3 main build action
6   completed: SUCCESS
7   07.11.2011 15:17:29 hudson.model.Run run
8   INFO: Pipeline Blog Post - Creating tables and Inserting data #3 main build action completed: SUCCESS
9   07.11.2011 15:22:31 hudson.model.Run run
10  INFO: Pipeline Blog Post - Start server #3 main build action completed: SUCCESS
```

To achieve these sorts of relationships with Jenkins, you can either configure a post-build action starting the next job (downstream) or configure the trigger of the current build (upstream). Both ways are totally equivalent.

### TRIGGERING A JOB AFTER SUCCESSFUL COMPLETION OF THE CURRENT JOB

This first way is probably the most intuitive. It consists of configuring the job triggered after the current job. In our scenario, the 'stop server' job triggers, on successful completion, the 'cleanup' job. To configure this dependency, we configure a post-build action in the Configure page of the 'stop server' job:

## Build other projects

Projects to build

- Trigger only if build succeeds
- Trigger even if the build is unstable
- Trigger even if the build fails

build once the current job finishes

Trigger a

### STARTING THE CURRENT JOB AFTER COMPLETION OF ANOTHER BUILD

With this second way, you configure which job triggers the current job. In our scenario, the 'cleanup' job is triggered after the 'stop server' job. So in the Configure page of 'cleanup', in the Build trigger section, we select the Build after other projects are built and specify the 'stop server' job:

## Build Triggers

### Build after other projects are built

Projects names

Multiple projects can be specified like 'abc, def'

Using this one-to-one dependency approach is probably the simplest way to orchestrate jobs on Jenkins. With such easy configuration, decomposing complex activities / build processes is very simple. You can restart the stream from any point, and get feedback after every success or failure.

### Über Job: the wrong good idea

One attempt to optimize the previous method was to create a kind of über job, triggering all other jobs. Unfortunately, even if it was a brilliant idea on paper, it doesn't work. Indeed, there are two issues:

- Even if a job fails, others are triggered
- The job execution order is not deterministic

The first point is simple to explain: the jobs are not interconnected, so even if one fails, others are still executed. This can be really annoying if the initial requirement of a job is not set up correctly.

```
1 07.11.2011 15:32:51 hudson.model.Run run
2 INFO: Pipeline Blog Post - Über Job #6 main build action completed: SUCCESS
3 07.11.2011 15:32:59 hudson.model.Run run
4 INFO: Pipeline Blog Post - Stop Server #5 main build action completed: SUCCESS
5 07.11.2011 15:33:01 hudson.model.Run run
6 INFO: Pipeline Blog Post - Cleanup #3 main build action completed: SUCCESS
7 07.11.2011 15:33:03 hudson.model.Run run
8 INFO: Pipeline Blog Post - Dropping tables #3 main build action completed: FAILURE
9 07.11.2011 15:33:05 hudson.model.Run run
10 INFO: Pipeline Blog Post - Creating tables and Inserting data #3 main build action completed: SUCCESS
11 07.11.2011 15:33:07 hudson.model.Run run
12 INFO: Pipeline Blog Post - Start server #3 main build action completed: SUCCESS
```

In this log, the 'drop tables' job failed. We would expect the whole scenario to come to a halt at this point, but unfortunately that's not the case. When this happens, we can't be sure of the resulting state on our restarted server.

The second point is more tricky. Jenkins is built on an asynchronous model: jobs are scheduled and will be executed later. The order of execution can depend on a lot of different parameters, so the order is hard to plan. In our case, we have seen:

```
1 07.11.2011 15:32:51 hudson.model.Run run
2 INFO: Pipeline Blog Post - Pipeline #18 main build action completed: SUCCESS
3 07.11.2011 15:32:59 hudson.model.Run run
4 INFO: Pipeline Blog Post - Stop Server #15 main build action completed: SUCCESS
5 07.11.2011 15:33:01 hudson.model.Run run
6 INFO: Pipeline Blog Post - Cleanup #13 main build action completed: SUCCESS
7 07.11.2011 15:33:03 hudson.model.Run run
8 INFO: Pipeline Blog Post - Creating tables and Inserting data #13 main build action completed: SUCCESS
9 07.11.2011 15:33:05 hudson.model.Run run
10 INFO: Pipeline Blog Post - Dropping tables #13 main build action completed: SUCCESS
11 07.11.2011 15:33:07 hudson.model.Run run
12 INFO: Pipeline Blog Post - Start server #13 main build action completed: SUCCESS
```

You can see that the tables were dropped after the data was inserted. Well... I'll let you guess the state of the server after this execution.

So, even if this method seemed to be a good idea, it's actually a pretty bad idea if you want your process executed reliably.

A bit of optimization: Fork and Join

This last method uses a Jenkins plugin named 'Join plugin' (see the [Join plugin page](#)). In brief, this plugin allows you to configure fork/join patterns: once downstream projects are completed, other projects are triggered.

If you have jobs that can be run in any order, this plugin will reduce the amount of configuration you need. In our scenario, in the 'stop server' job, it can be used as follows:

Build other projects

Projects to build

Pipeline Blog Post - Cleanup, Pipeline Blog Post - Dropping tables

- Trigger only if build succeeds
- Trigger even if the build is unstable
- Trigger even if the build fails

Groovy Postbuild

Join Trigger

Trigger even if some downstream projects are unstable

Projects to build once, after all downstream projects have finished

Pipeline Blog Post - Creating tables and Inserting data, Pipeline Blog Post - Start server

So, the 'stop server' job triggers the 'clean up' and 'drop table' jobs. Once those (independent) jobs are completed, we trigger the data insertion and restart the server. In our experience, these two 'join' jobs were always executed in right order (and on the same executor). But we recommend triggering only one join job with a one-to-one dependency:

/-> Cleanup -\

Stop server \* -> Insert data -> Start server

\-> Drop table -/

This approach reduces the number of builds to configure, but should be used only if your jobs are independent.

A last tip

Especially in the one-to-one approach, the pipeline can become pretty long. Jenkins has a nice plugin to visualize the downstream builds: [Downstream buildview plugin](#).

We recommend using this plugin to track the progress and visualize the result of complex/long pipelines.

Conclusion

Even if the advantages of splitting a long build process into small jobs are obvious, doing so may be more difficult than expected. This post has presented several ways to create dependencies between your jobs for Jenkins/Hudson.

Usefull link:

<https://linuxize.com/post/how-to-install-jenkins-on-centos-7/>

Prerequisites and steps to setup Jenkins:

1. Get the sudo access for Jenkins.
2. Install jdk8/ openjdk8
3. And make sure Java 8 is the default Java version.
4. enable the Jenkins repository
5. And add the repository to your system
6. Once the repository is enabled, install the latest stable version of Jenkins
7. start the Jenkins service
8. check whether it started successfully

9. Finally enable the Jenkins service to start on system boot.
10. Adjust the Firewall to open the necessary port
11. And browse the URL <http://localhost:8080>
12. And follows the default steps like password and install services for jenkins

## To change the port:

Go to the directory where you installed the Jenkins (In default, it is under Program Files/Jenkins)

Open the Jenkins.xml or in UNIX default location:

```
sudo vim /etc/default/jenkins
```

Search --httpPort=8080 and replace the 8080 with the new port number that you wish

Jenkins: How to change JENKINS\_HOME on Windows

Issue : JENKINS\_HOME is almost full.

Your Jenkins data directory C:\Users\80053806\.jenkins (AKA JENKINS\_HOME) is almost full. You should act on it before it gets completely full.

| Variable     | Value                         |
|--------------|-------------------------------|
| JENKINS_HOME | E:\GouseShaik\jenkins\Jenkins |

Solution:

If you install Jenkins using MSI installer, it copies files to C:\Program Files (x86)\jenkins and uses the install directory as JENKINS\_HOME (on 64bit machine).

This means that Jenkins stores all data (including plugins, workspace and job data) to C:\Program Files (x86)\jenkins. On windows, it is recommended that application data is stored to ProgramData and AppData. So, I wanted to change JENKINS\_HOME.

First, I added JENKINS\_HOME to environment value, but it doesn't make any sense. Jenkins prefers the value set on jenkins.xml when we use MSI installer.

Now, let's change JENKINS\_HOME to C:\ProgramData\Jenkins (Jenkins version: 1.550).

1. Stop Jenkins service.
2. Open jenkins.xml in the install folder, and edit

3. <env name="JENKINS\_HOME" value="%BASE%"/>

as follows:

```
<env name="JENKINS_HOME" value="%ProgramData%\Jenkins"/>
```

4. Move all files in install folder except for the following files (Don't copy because it breaks symbolic links in jobs folder):

- jre folder
- jenkins.err.log
- jenkins.exe
- jenkins.exe.config
- jenkins.out.log
- jenkins.war
- jenkins.war.bak
- jenkins.war.tmp
- jenkins.wrapper.log
- jenkins.xml

5. Start Jenkins service.

6. Open <http://localhost:8080/systemInfo> and check JENKINS\_HOME has been changed.

Enjoy!

Jenkins PNG Images size: 278x68

Title.png

## Build Jenkins Job with Condition and Parameter



### Scenario:

Suppose we want to start or stop apache service via reading user parameter.  
Install Conditional BuildStep plugin.

### Configure Job

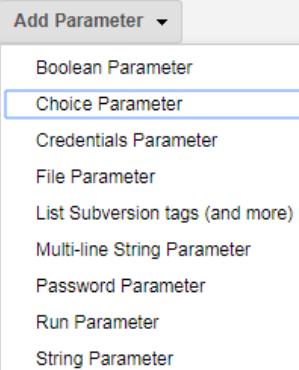
1. Go to Jenkins Job and tick This project is parameterized.

This project is parameterized

Add Parameter ▾

2. Click on Add Parameter and select Choice Parameter

This project is parameterized



## Source Code

None

3. Here give a name of the parameter and also define Choices of a parameter.

Remember name is act as a Variable name

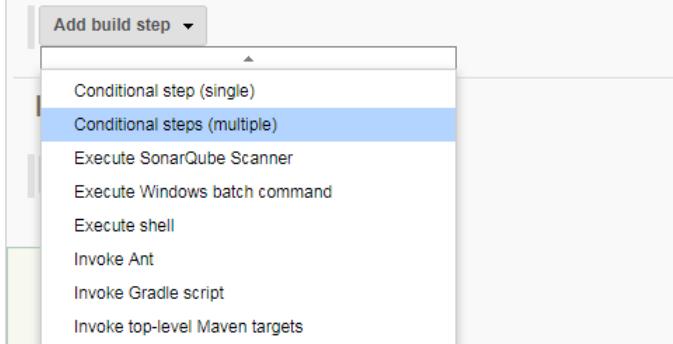
This project is parameterized

The screenshot shows the 'Choice Parameter' configuration dialog. It has fields for 'Name' (set to 'select'), 'Choices' (containing 'START' and 'STOP'), and 'Description' (with the placeholder 'choose what you want'). There are also 'Plain text' and 'Preview' buttons at the bottom.

4. Scroll down to the Build option and Add build step.

Select Conditional steps(multiple)

## Build



5. Select Run? is Strings Match

Create two Conditional steps as per below

### Condition 1: (for apache START)

Conditional steps (multiple)

Run? Strings match

String 1 \$select  
String 2 STOP  
Case insensitive

Advanced...

**Steps to run if condition is met**

Execute shell

Command service httpd start

See the list of available environment variables

Advanced...

Add step to condition ▾

### Condition 2: (for apache STOP)

Conditional steps (multiple)

Run? Strings match

String 1 \$select  
String 2 STOP  
Case insensitive

Advanced...

**Steps to run if condition is met**

Execute shell

Command service httpd start

See the list of available environment variables

Advanced...

Add step to condition ▾

Also, you can create a simple Execute shell script

Execute shell

Command if [ \$select = START ]  
then  
service httpd start  
else  
service httpd start  
fi

See the list of available environment variables

Advanced...

6. Click Save
7. Now build job with a parameter.

The screenshot shows the Jenkins interface for a project named 'Apache'. The left sidebar contains links: 'Back to Dashboard', 'Status', 'Changes', 'Workspace', 'Build with Parameters' (which is highlighted with a red box), 'Delete Project', 'Configure', and 'Rename'. Below the sidebar, a message says 'Select your choice and Build.' A dropdown menu is open under 'Build with Parameters', showing options 'START', 'STOP', and 'Build'. The 'START' option is selected.

Select your choice and Build.

## Project Apache

This build requires parameters:

A dropdown menu is open, showing options 'START', 'STOP', and 'Build'. The 'START' option is selected. Below the menu, there is a message 'you want'.

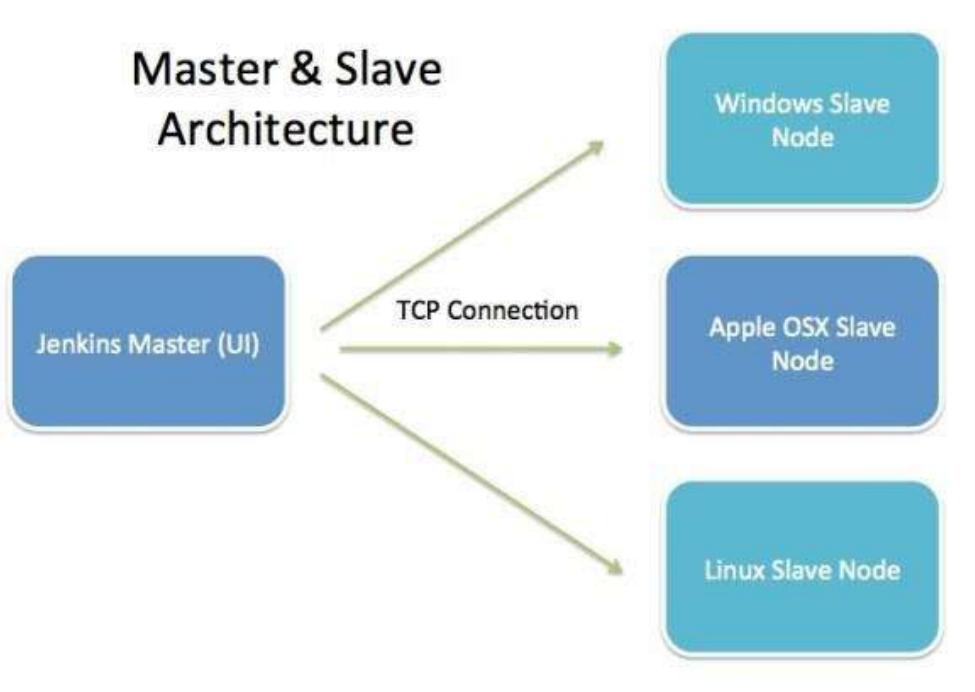
## Jenkins Master and slave node configuration

<https://mohitgoyal.co/2017/02/14/add-linux-slave-node-in-the-jenkins/>

Jenkins is a cross-platform, continuous integration and continuous delivery application that can be used to build and test your software projects continuously making it easier for developers to integrate changes to the project, and making it easier for users to obtain a fresh build, which in turn increases your productivity. When you are working on a number of projects which get built on a regular basis or want to run multiple jobs or might need several different environments to test your builds, then a single Jenkins server cannot simply handle the entire load.

Jenkins supports the master-slave architecture, i.e. many slaves work for a master. It is also known as Jenkins Distributed Builds. It also allows you to run jobs on different environments like Linux, Windows, MacOS, etc. We can also run the same test case on different environments in parallel using Jenkins Distributed Builds, which in turn helps you to achieve the desired results quickly using this distributed approach. All of the job results are collected and combined on the master node for monitoring.

## Master & Slave Architecture



### Jenkins Master

Your main Jenkins server is the master machine. The tasks performed by the master are :

- Scheduling build jobs.
- Dispatching builds to the slaves for the execution.
- Monitor the slaves.
- Recording and presenting the build results.
- Can also execute build jobs directly.

### Jenkins Slave

A slave is a Java executable that runs on a remote machine. The characteristics of the slave are :

- It hears requests from the Jenkins Master instance.
- Slaves can run on a variety of operating systems.
- The job of a Slave is to do as they are told to, which involves executing build jobs dispatched by the Master.
- We can configure a project to always run on a particular Slave machine or a particular type of Slave machine, or simply let Jenkins pick the next available Slave.

### Setting Up A Jenkins Slave

For this example, I have used one AWS EC2 instance as a slave machine and my system as the master.

1. Go to Jenkins dashboard -> Manage Jenkins ->Manage Nodes.

2. Give a name to the node and then click OK. For the first time user, only one option of "Permanent Agent" will be present. After that, we get another option, "Copy Existing Node" which is self-explanatory.  
3. Then, a page will open in which we have to specify the configuration of the slave machine. All of them are not required but let's get an overview of the all parameters asked in the configuration :

The screenshot shows the Jenkins 'TestSlave/configure' page. The configuration fields are as follows:

- Name:** TestSlave
- Description:** (empty)
- # of executors:** 1
- Remote root directory:** /home/ec2-user/jenkins
- Labels:** slave
- Usage:** Use this node as much as possible
- Launch method:** Launch slave agents via SSH
- Host:** 13.127.1.226
- Credentials:** ec2-user
- Host Key Verification Strategy:** Known hosts file Verification Strategy
- Availability:** Keep this agent online as much as possible

A 'Save' button is located at the bottom left.

- **Name:** Name of the Slave which should be unique.
- **Description:** Description of this slave. It is optional but giving a description of it would be really helpful for other team members.
- **# of executors:** The maximum number of concurrent builds that Jenkins may perform on this agent. I have used 1 executor for testing purpose, but a good practice would be the number of CPU cores on the machine.
- **Remote root directory:** An agent needs to have a directory dedicated to Jenkins. Specify the path to this directory on the agent. I've specified /home/ec2-user/jenkins as the remote root directory.
- **Labels:** Labels are used to group multiple agents into one logical group. Multiple labels must be separated by a space.
- **Usage:** Controls how Jenkins schedules builds on this node.
- **Launch method:** It Controls how Jenkins starts this agent.
  - **Launch agent via execution of command on the master:** Starts an agent by having Jenkins execute a command from the master. Use this when the master is capable of remotely executing a process on another machine, e.g. via SSH or RSH.
  - **Launch slave agents via SSH:** Starts a slave by sending commands over a secure SSH connection. The slave needs to be reachable from the master, and you will have to supply an account that can log in to the target machine. No root privileges are required. If we select this, it will further need following credentials :
    - **Host:** The IP address of the slave machine.
    - **Credentials:** Select the credentials to be used for logging in to the remote host.
    - **Host Key Verification Strategy:** Controls how Jenkins verifies the SSH key presented by the remote host whilst connecting. These are of 4 types :
      - **Known Host file verification strategy:** Checks the known hosts file (~/.ssh/known hosts) for the user Jenkins is executing under, to see if an entry exists that matches the current connection.
      - **Manually provided key verification strategy:** Checks the key provided by the remote host matches the key set by the user who configured this connection.
      - **Manually trusted key verification strategy:** Require a user with Computer.CONFIGURE permission to authorise the key presented during the first connection to this host before the connection will be allowed to be established.
      - **Non-Verifying verification strategy:** Does not perform any verification of the SSH key presented by the remote host, allowing all connections regardless of the key they present. It's not advisable to select as it may open the path for attackers.
- **Availability:** Controls when Jenkins starts and stops this agent. It has two options available.
  - **Keep this slave on-line as much as possible:** This one is the default and normal setting. In this mode, Jenkins tries to keep the slave on-line as much as possible. If Jenkins can start the slave without user assistance,

it will periodically attempt to restart the slave if it is unavailable. Jenkins will not take the slave off-line.

- Take this slave on-line when in demand and off-line when idle: In this mode, if Jenkins can launch the slave without user assistance, it will periodically attempt to launch the slave while there are unexecuted jobs else the slave will be taken off-line by Jenkins
- After giving all the required configurations, click on the Save and your slave machine will be up and running.

The screenshot shows the Jenkins interface at [localhost:8080/computer/](http://localhost:8080/computer/). On the left, there's a sidebar with links like 'Back to Dashboard', 'Manage Jenkins', 'New Node', and 'Configure'. The main area is titled 'Nodes' and displays a table with two rows:

| S | Name      | Architecture  | Clock Difference | Free Disk Space | Free Swap Space | Free Temp Space | Response Time |
|---|-----------|---------------|------------------|-----------------|-----------------|-----------------|---------------|
| 1 | master    | Linux (amd64) | In sync          | 826.51 GB       | 7.63 GB         | 826.51 GB       | 0ms           |
| 2 | TestSlave | Linux (amd64) | In sync          | 5.30 GB         | 0 B             | 5.30 GB         | 2951ms        |

Below the table, it says 'Data obtained 1 min 3 sec' and 'Build Queue'. A 'Refresh status' button is at the bottom right.

You now have a master and a slave machine. You could add more slaves or try various configurations available. even, you can try using the setup for performing various jobs, which itself can be configured according to the master-slave setup.

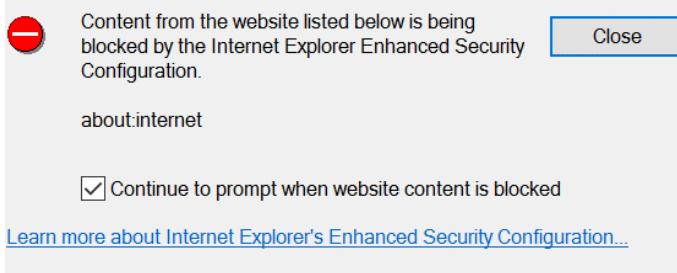
## Jenkins blocked by Internet explorer:

Error:

getting content from the website listed below is being blocked by the internet explorer enhanced security configuration

Internet Explorer

X



**Solution:** lower the security disable

## Jenkins-linking-to-my-local-git-repository

In this case, the URL should start with the file protocol followed by the path to the repository.

E.g., <file:///home/rbkcbecq/dev/git/gitsandbox>.

## Access local git repository in Jenkins docker container

In case somebody wants to connect to a local git repository from a Jenkins which runs in a docker container I would recommend to mount the local git repository folder to the docker image via the Volumes flag (see [Use Volumes](#) for more details).

To mount your local git repository folder to a Jenkins container run the container with an additional `-v` or `--volume` flag.

The basic docker run statement from the official docker image documentation would then look like.

```
docker run -p 8080:8080 -p 50000:50000
           -v <PATH_TO_LOCAL_GIT_REPO>:<MOUNT_POINT_IN_CONTAINER>:ro
           jenkins/jenkins:latest
```

The `:ro` option is optional. It will mount the volume as read-only. So you won't be able to write to the repos from the container.

Then you can simply access your git repository via the file protocol.

```
file://<MOUNT_POINT_IN_CONTAINER>
```

No need to use ssh.

If you run Jenkins within Docker, one possible solution would be via SSH:

```
ssh://user@IP_of_your_host/path_to_your_project/project_name
```

If you did clone a remote repository into local where Jenkins is running. You can just put the path of the local repository then it will work.

For example, `/home/username/local_repository_path`.

## Apache Maven

```
export M2_HOME=/usr/local/apache-maven/apache-maven-3.3.1
```

```
export M2=$M2_HOME/bin
```

```
export MAVEN_OPTS=-Xms256m -Xmx512m
```

[To generate WAR file from WEB-INF file project directory from CommandLine](#)

```
Cd \prm-core-1.0.0-SNAPSHOT\WEB-INF
```

```
Jar -cvf test.war *
```

## Jenkins FIX Performance Issues:

### Solution:

I have used “Discard old build” plugin, issue resolved for the slowness.

<https://mincong.io/2018/12/25/discard-old-build-artifacts-on-jenkins/>

Do you know that feeling when you've finished working on a feature, pushed the code, but then your CI system refuses to respond?

<https://engineering.taboola.com/5-simple-tips-boosting-jenkins-performance/>

Lagging or slow responsiveness is very common among Jenkins users, and you can find many [reported issues](#) on it. Slow CI systems are frustrating, they make you develop slower and waste your time. I have worked on several Jenkins systems with various versions (1.x and 2.x), tens of slaves and hundreds of builds per day. I have managed to improve the performance of those systems using a few simple guidelines. In the following post I will share 5 tips that can make your Jenkins better, and put a smile on your developers' faces.

## Tip 1: Minimize the amount of builds on the master node

The master node is where the application is actually running, this is the brain of your Jenkins and, unlike a slave, it is not replaceable. So, you want to make your Jenkins master as “free” from work as you can, leaving the CPU and memory to be used for scheduling and triggering builds on slaves only. In order to do so, you can restrict your jobs to a node label, for example:

Restrict where this project can be run

Label Expression

ParallelModuleBuild

Label is serviced by 35 nodes

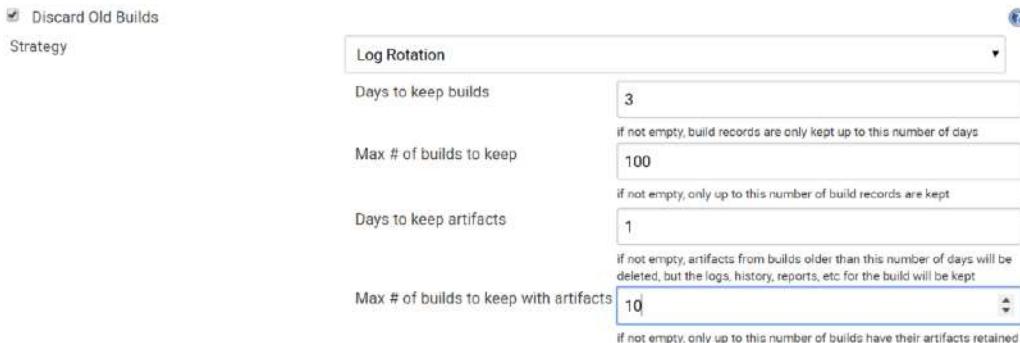
In the case of pipeline jobs, put the label when allocating the node, for example:

```
stage("stage 1"){
    node("ParallelModuleBuild"){
        sh "echo \"Hello ${params.NAME}\""
    }
}
```

In those cases, the job and node block will only run on slaves labeled with *ParallelModuleBuild*.

## Tip 2: Do not keep too much build history

When you configure a job, you can define how many of its builds, or for how long they, will be left on the filesystem before getting deleted. This feature, called Discard Old Builds, becomes very important when you trigger many builds of that job in a short time. I have encountered cases where the history limit was too high, meaning too many builds were kept on the filesystem. In such cases, Jenkins needed to load many old builds – for example, to display them in the history widget, – and performed very slowly, especially when trying to open those job pages. Therefore, I recommend limiting the amount of builds you keep to a reasonable number.



## Tip 3: Clear old Jenkins data

In continuation to the build data from the previous tip, another important thing to know is the old data management feature. As you probably know, Jenkins keeps the jobs and builds data on the filesystem. When you perform an action, like upgrading your core, installing or updating a plugin, the data format might change. In that case, Jenkins keeps the old data format on the file system, and loads the new format to the memory. It is very useful if you need to rollback your upgrade, but there can be cases where there is too much data that gets loaded to the memory. High memory consumption can be expressed in slow UI responsiveness and even OutOfMemory errors. To avoid such cases, it is best to open the old data management page (<http://JenkinsUrl/administrativeMonitor/OldData/manage>), verify that the data is not needed, and clear it.

## Tip 4: Define the right heap size

This tip is relevant to any Java application. A lot of the modern Java applications get started with a maximum heap size configuration. When defining the heap size, there is a very important JVM feature you should know. This feature is called UseCompressedOops, and it works on 64bit platforms, which most of us use. What it does, is to shrink the object's pointer from 64bit to 32bit, thus saving a lot of memory. By default, this flag is enabled on heaps with sizes up to 32GB (actually a little less), and stops working on larger heaps. In order to compensate the lost space, [the heap should be increased to 48GB\(!\)](#). So, when defining heap size, it is best to stay below 32GB. In order to check if the flag is on, you can use the following command (jinfo comes with the JDK):

## Tip 5: Tune the garbage collector

The garbage collector is an automatic memory management process.

Its main goal is to identify unused objects in the heap and release the memory that they hold. Some of the GC actions cause the Java application to pause (remember the UI freeze?). This will mostly happen when your application has a large heap (> 4GB). In those cases, GC tuning is required to shorten the pause time. After dealing with these issues in several Jenkins environments, my tip contains a few steps:

- Enable G1GC – this is the most modern GC implementation (default on JDK9)
- Enable GC logging – this will help you monitor and tune later
- Monitor GC behavior – I use [http://gceeasy.io/](http://gceasy.io/)
- Tune GC with [additional flags](#) as needed
- Keep monitoring
- Read [this post](#) and [this one](#) – these are great posts about GC tuning

## Now it's your turn

Jenkins responsiveness issues are very common, but can be dealt with. In one of the environments I worked on, I had a configuration page which took over a minute to open. After implementing the above tips, especially G1GC and old data, this page opened in a split second.

If you got this far, you are probably familiar with this issue. So, try my tips and share what worked for you and what hasn't. There is a good chance your Jenkins will speed up too.

### [Reload Configuration from Disk](#)

You must either restart Jenkins or reload the data, in order to reconcile the UI with its back end. There is no other way.

If you have access to the "Manage Jenkins" menu there is a link to "Reload Configuration from Disk", which is a little less invasive than a restart.

Keep in mind that the system will be not usable for a few seconds while the data is reloaded, just like during a full restart. If you have a lot of builds, this might take a little. So, make sure that this is ok with the other users of your system.

Maven coordinates:

**Full naming convention: Groupid:artifactid:packaging:version**

- ✓ Groupid :
- ✓ Artifactid :
- ✓ Version :
- ✓ Packaging :

### Mvn install

Mvn install is the very first command to download your dependency in .m2/repository directory.

Repository version will be checked from below maven centralized repository.

<http://repo.maven.apache.org/maven2/>

External repository : <https://mvnrepository.com/>

if you want junit dependency, get it developer defined version from below path.

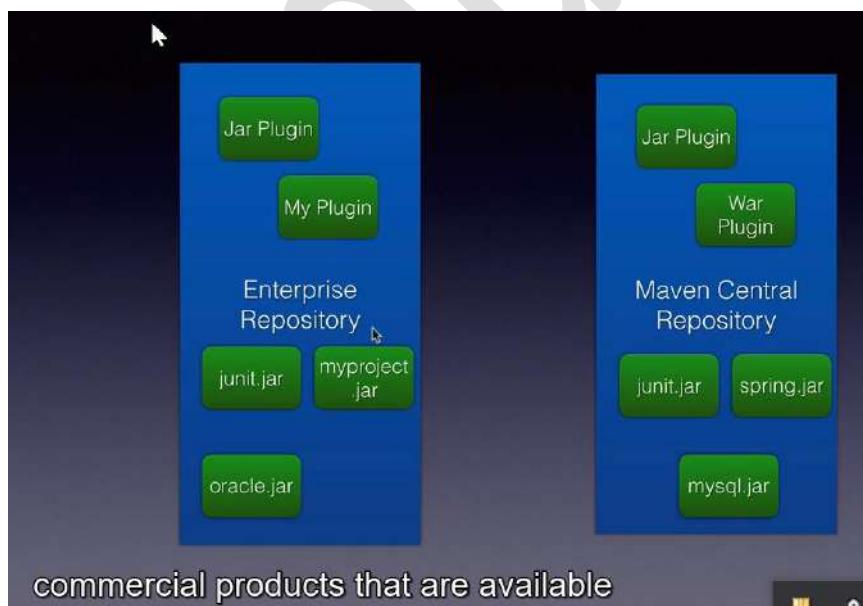
<http://repo.maven.apache.org/maven2/junit/junit/>

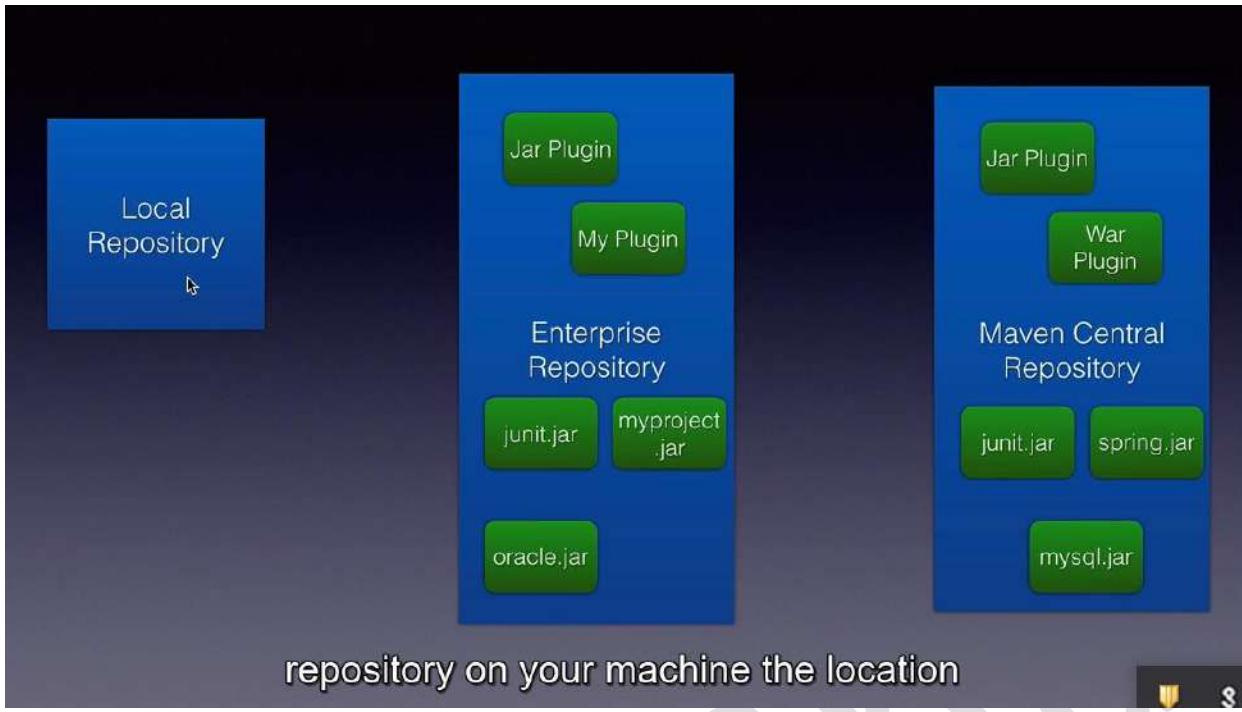
  
Not secure | repo.maven.apache.org/maven2/junit/junit/

### junit/junit

|                        |                  |
|------------------------|------------------|
| <a href="#">...</a>    |                  |
| <a href="#">3.7/</a>   | 2005-09-20 05:49 |
| <a href="#">3.8/</a>   | 2005-09-20 05:49 |
| <a href="#">3.8.1/</a> | 2005-09-20 05:49 |
| <a href="#">3.8.2/</a> | 2007-05-14 21:34 |
| <a href="#">4.0/</a>   | 2006-02-16 19:24 |
| <a href="#">4.1/</a>   | 2006-10-07 12:02 |
| <a href="#">4.10/</a>  | 2011-09-29 19:12 |

If you don't find the dependency files from Maven centralized repository, then go to Enterprise repository which are some times as commercial.





### Assignment:

```

mvn archetype
Command Line Arguments
Sum of two Numbers
mvn install
jar -cp

```

## Apache Archiva

### How to install Apache Archiva in Windows

By [mkyong](#) | November 26, 2009 | Updated : August 30, 2012 | Viewed : 13,576 | +15 pv/w

Apache Archiva is a powerful repository manager, and usually hook up with Maven development.

Apache Archiva installation

1. Download Apache Archiva

Visit Apache Archiva official website, Click on the download link, e.g Archiva 1.2.2.

*P.S Choose the standalone version (Archiva will use jetty as build-in web container)*

2. Extract it

Extract the downloaded zip file to folder "D:\apache-archiva-1.2.2\"

### 3. Install as Windows service

Find the **archiva.bat** "{Archiva\_Home}\bin" folder, and issue the "**archiva install**" command to register it as a Windows service.

```
D:\apache-archiva-1.2.2\bin>archiva install  
wrapper | Apache Archiva installed.  
Copy
```

### 4. Start Archive

Two ways to start the Apache Archive :

1. Start it with Archiva command - "archiva start".

```
D:\apache-archiva-1.2.2\bin>archiva start  
wrapper | Starting the Apache Archiva service...  
wrapper | Apache Archiva started.  
Copy
```

2. Start it as Windows service under control panel.

### 4. Archive web admin

Access your Apache Archiva web admin via 8080 port - <http://localhost:8080/archiva/>.

P.S Apache Archiva standalone version is using jetty as build-in web container and 8080 as default port.

### 5. Done

#### Post Installation

Please change the Apache Archiva default 8080 port to avoid port conflict issue, as you know, the 8080 is a very hot port number.

#### 1. Archiva configuration file

Locate the Apache Archiva web container configuration file - "**D:\apache-archiva-1.2.2\conf\jetty.xml**"

#### 2. Update port number

Find the following pattern

```
<Call name="addConnector">  
  <Arg>  
    <New class="org.mortbay.jetty.nio.SelectChannelConnector">  
      <Set name="host"><SystemProperty name="jetty.host"/></Set>  
      <Set name="port"><SystemProperty name="jetty.port" default="8080"/></Set>  
      <Set name="maxIdleTime">30000</Set>  
      <Set name="Acceptors">2</Set>  
      <Set name="statsOn">false</Set>  
      <Set name="confidentialPort">8443</Set>  
      <Set name="lowResourcesConnections">5000</Set>  
      <Set name="lowResourcesMaxIdleTime">5000</Set>  
    </New>  
  </Arg>  
</Call>  
Copy
```

Change the jetty port to other port - "8888".

```
<Set name="port"><SystemProperty name="jetty.port" default="8888"/></Set>  
Copy
```

#### 3. Restart Apache Archiva

Restart it to take effect. Now you can access the Apache Archive web admin via <http://localhost:8888/archiva/>.

## Siebel Installation with OEL7 and Docker

<https://www.youtube.com/watch?v=MvETSSryqok&feature=youtu.be>

### Quick links

[00:00](#) – Intro

[00:30](#) - OEL Install

Ram : 12GB

Storage space: 200 GB

Auto Mount the shared folder

Map the iso , which is download from edelivery

Add your user under sudoers

```
Sudo yum install kernel-uek-devel-$uname -r  
Sudo usermod -aG docker useradmin
```

## 22:25 - Setting up Virtual Box extensions

Install virtual box guest additional iso files

## 27:00 - Setting up Docker

<https://blogs.oracle.com/virtualization/install-docker-on-oracle-linux-7-v2>

```
# cd /etc/yum.repos.d/  
# wget http://yum.oracle.com/public-yum-ol7.repo  
# vi public-yum-ol7.repo  
  
[ol7_latest]  
name=Oracle Linux $releasever Latest ($basearch)  
baseurl=http://yum.oracle.com/repo/OracleLinux/OL7/latest/$basearch/  
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle  
gpgcheck=1  
enabled=1  
  
[ol7_UEKR4]  
name=Latest Unbreakable Enterprise Kernel Release 4 for Oracle Linux $releasever ($basearch)  
baseurl=http://yum.oracle.com/repo/OracleLinux/OL7/UEKR4/$basearch/  
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle  
gpgcheck=1  
enabled=1  
  
[ol7_addons]  
name=Oracle Linux $releasever Add ons ($basearch)  
baseurl=http://yum.oracle.com/repo/OracleLinux/OL7/addons/$basearch/  
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-oracle  
gpgcheck=1  
enabled=1  
  
$ sudo yum install docker-engine  
$ docker version  
$ sudo vi /etc/docker/daemon.json  
{  
    "experimental": true  
}  
$ sudo systemctl restart docker  
$ docker version
```

## 34:04 - Setting up Git

```
$ sudo yum install git
```

35:15 - Pulling Github Repository

<https://github.com/OracleSiebel/ConfiguringSiebel>

git clone <https://github.com/OracleSiebel/ConfiguringSiebel.git>

cd ConfiguringSiebel/Containerization/Docker/ol7/build/**32-bit\_instantclient\_12c/**

and create a empty tnsname.ora file under above path

follow the steps under this link [https://github.com/OracleSiebel/ConfiguringSiebel/tree/master/Containerization/Docker/ol7/build/32-bit\\_instantclient\\_12c](https://github.com/OracleSiebel/ConfiguringSiebel/tree/master/Containerization/Docker/ol7/build/32-bit_instantclient_12c)

## 35:45 - Building 32-bit client

```
docker build -t registry.local.com:5000/oracle/database-instantclient/32bit:12.2.0.1 --squash . 2>&1 | tee build.$(date +%F_%R).log
```

## 42:16 - Building IP17 Base

Copy the siebel enterprise server image to this path

Cd /cygdrive/c/vms/siebelol7/ConfiguringSiebel/Containerization/Docker/ol7/build/base\_siebel

Cp -R siebel\_enterprise\_server .

50:20 - Disabling SELinux

58:48 - Applying 19.1 Update

1:08:55 - Adding Persistence Layer

1:10:33 - Launching Example Enterprise

1:12:20 - Ensure user exists and persistent volumes are created and accessible

1:20:13 - Auto Configuration of Enterprise

1:25:58 - Call Center Login

Show less

REPLY

## Setting up Siebel Provisioning Server

**This would take some time. Please wait...**

This is a normal message, and it will go away when the server set up is complete.

The script presents the following actions:

```
=====
Provisioning Oracle Siebel - Actions
=====

1: Provision Single Siebel App Tier + Demo DB
2: Provision Single Siebel App Tier + Base DB
3: Provision Single Siebel App Tier + Demo DB DBAAS
4: Provision Single Siebel App Tier + Base DB DBAAS
5: Provision Multiple Siebel App Tiers + Base DB
6: Provision Multiple Siebel App Tiers + Base DB DBAAS
7: Delete Existing Environment
8: Exit Without Making Changes

Choose Action: [ ]
```

Note: Action 7, Delete Existing Environment, should be used if you provisioned either the Demo DB or Base DB, and the provisioning failed. You can delete the environment, resolve the problem, and re-provision the database.

1. Select one of the server provisions.
2. Using the information you collected in your note document, respond to the following prompts:

```
=====
Provisioning Oracle Siebel - Cloud Credentials
=====

Enter Oracle Compute Cloud Service Endpoint      :
Enter Oracle Cloud Identity Domain            :
Enter Oracle Cloud Username                   :
Enter Oracle Cloud Password                  :
Enter Oracle DBaaS Endpoint                 :
Enter Oracle DBaaS Identity Domain          :
Enter Oracle DBaaS Username                 :
Enter Oracle DBaaS Password                :
```

After you enter the above information, the provisioning script contacts the Provisioning Server and validates your credentials.

3. Select an available image from the list presented.

```
Available image(s) for Application Tier
=====
1: /Compute-emeaaaausebldeva/ [REDACTED]@oracle.com/Siebel_IP2016_SIEBELMT
2: /Compute-emeaaaausebldeva/ [REDACTED]@oracle.com/Siebel_IP2016_SIEBELMT

Choose image for Application Tier from above list:
```

4. Select one of the industries from the list presented.

```
Available Industry(s)
```

```
1: Automotive  
2: Financial Services  
3: Life Sciences  
4: Sales  
5: Service  
6: Partner Relationship Management  
7: Public Sector  
8: Telecommunications  
9: Loyalty  
10: Consumer Goods
```

```
Choose an Industry key from the above list: [ ]
```

5. Create a username and password for the landing page of your application. You can add these values to your note document.

```
Available Industry(s)
```

```
1: Automotive  
2: Financial Services  
3: Life Sciences  
4: Sales  
5: Service  
6: Partner Relationship Management  
7: Public Sector  
8: Telecommunications  
9: Loyalty  
10: Consumer Goods
```

```
Choose an Industry key from the above list:
```

```
Enter username for your app :  
Enter password for your app :  
re-Enter password for your app :
```

6. Select a shape for the application tier, and then again for the database tier. Select the shape that uses the least memory.

```
Available Shapes
```

| Name    | CPU  | Memory |
|---------|------|--------|
| 1: oc1m | 2.0  | 15360  |
| 2: oc2m | 4.0  | 30720  |
| 3: oc3  | 2.0  | 7680   |
| 4: oc3m | 8.0  | 61440  |
| 5: oc4  | 4.0  | 15360  |
| 6: oc4m | 16.0 | 122880 |
| 7: oc5  | 8.0  | 30720  |
| 8: oc5m | 32.0 | 245760 |
| 9: oc6  | 16.0 | 61440  |
| 10: oc7 | 32.0 | 122880 |

```
Choose Shape for Application Tier:
```

```
Choose Shape for Demo Database Tier:
```

7. if you are provisioning DBaaS, enter a password that meets the following criteria:

Length of 8 to 30 characters

Contains at least one lowercase letter

Contains at least one uppercase letter

Contains at least one number

Contains one of the following special characters: - (dash), - (underscore), or # (octothorpe).

Does not contain any space characters.

Re-enter the same password when prompted.

```
Enter DBaaS password. (Note: Use a password that is 8 to 30 characters long, contains at least one lowercase letter, one uppercase letter, one number, one of the following characters -_# and no white space character.) :  
re-Enter DBaaS password :
```

8. Select a database image from the list presented.

```

Available image(s) for Demo Database Tier
-----
1: /Compute-emeaaaauseblpma1/Siebel_IP2016_DEMODB

Choose image for Demo Database Tier from above list: [ ]
```

9. Create a unique name for your environment. You can add this name to your note document.
10. Select an SSH key from the list presented.

```

Available SSH Key(s)
-----
1: /Compute-emeaaaauseblpma1/SSH_PK_PS

Choose an SSH key from the above list: [ ]
```

11. Review the summary and enter Y to accept the values.

```

=====
Provisioning Oracle Siebel - Review
=====

Oracle Cloud User : emeaaaausebldeva
Oracle Cloud Identity Domain : Provision Single Siebel App Tier + Demo DB DBAAS
Action : /Compute-emeaaaausebldeva/[REDACTED]@oracle.com/Siebel_IP2016_SIEBELMT
Application Tier Image : Life Sciences
Application Industry : tc1
Application User Name : oc3
Shape of Application Tier : oc3
Shape of DBaaS : omega
Environment Name : 1
Number of Application Tier Nodes : d16309key,SSH_PK_PS
SSH Key Name : no
Use Logical Hostname : dbaas
Database Service Type : 100
Dbaas Usable Storage : 12.1.0.2
Dbaas Version

=====

Do you wish to proceed with the above action (Y/N)?: [ ]
```

The provisioning script now starts to create the environment as you specified. This is not instantaneous. There will be many status messages as the script runs.

The script is done when you get this message: "Your Setup is ready. Please find the environment details below".

```

-----Your Setup is ready. Please find the environment details below-----
description: SINGLE TIER SIEBEL ENVIRONMENT
name: omega
oracle:
siebel:
apptier:
components: Enterprise Server, Siebel Server and Siebel Web Server Extension
hostname: e699c7
imagename: /Compute-emeaaaauseblpma1/Siebel_IP2016_SIEBELMT
instancename: omega-appl01
instanceshape: oc3
privateipaddress: 10.196.164.174
publicipaddress: 140.86.33.45
siebelenvurl: http://140.86.33.45:7780/
version: 161
dbtier:
connectstring: DESCRIPTION=(ADDRESS_LIST=(ADDRESS=(PROTOCOL=TCP)(HOST=140.86.13.197)(PORT=1521)))(CONNECT_DATA=(SERVICE_NAME=siebel))
hostname: f60765
imagename: /Compute-emeaaaauseblpma1/Siebel_IP2016_DEMODB
instancename: omega-db01
instanceshape: oc3
privateipaddress: 10.196.164.178
publicipaddress: 140.86.13.197
version: 12.1.0.2
```

Copy the value of the **siebelenvurl** field and paste that into your note document. You will use this URL to log in to your environment later.

12. Return to the Oracle Cloud Services interface and click the Network tab.

# OpenShift

## 01. OpenStack Introduction:

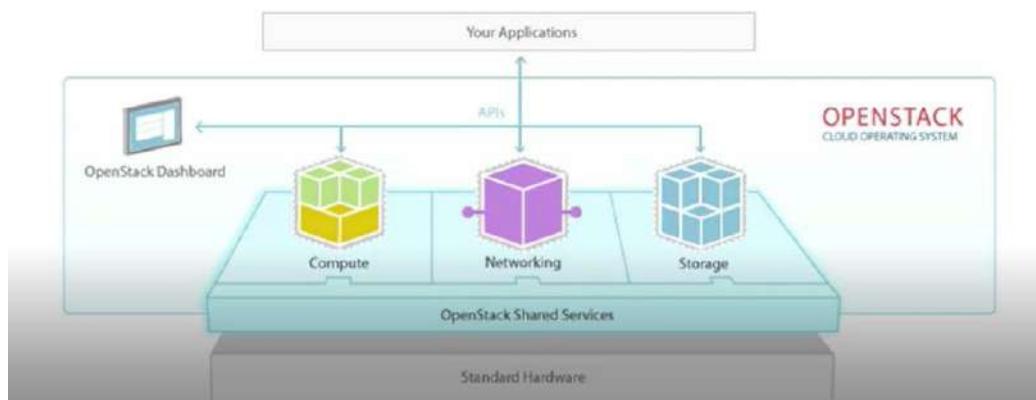
OpenStack is Cloud echoSystem to control large pools of storage, compute, and networking resources throughout a datacenter. This is all to manager dashboard(horizon) that gives and administrator control while yet empowering their users to provision resources through a web interface.

OpenStack is globally collaboration of developers and cloud computing technologist producing the ubiquitous open source computing platform for both public and private clouds.

Deploy and manager OpenStack with below topics

- Identity Service is know as “Keystone”
- Image Service is know as “Glance”
- Computer Service is know as “Nova”
- Networking is know as “neutron”
- Dashboard is know as “Horizon”
- Block Storage is know as “Cinder”
- Object Storage is know as “Swift”
- Orchestration is know as “Heat”
- Telemetry is know as “Ceilometer”

## OpenStack: The Open Source Cloud Operating System



## 02. OpenStack Architecture

## 03. OpenStack Messaging Server

## 04. Identity Service – Keystone

## 05. OpenStack Clients

## 06. Image Service – Glance

- 07. Compute Service – Nova
- 08. Networking Service – Neutron
- 09. OpenStack Dashboard – Horizon
- 10. Block Storage - Cinder
- 11. Object Storage - Swift
- 12. Orchestration Service - Heat
- 13. OpenStack Telemetry - Ceilometer
- 14. Launching Instances

## Ansible

### Oracle Linux - Install Ansible

Ansible is an open-source automation engine that automates software provisioning, configuration management, and application deployment. Ansible is based upon a push mechanism where you will push configurations to the servers rather than pulling them as is done by, for example, puppet. When you want to start using Ansible the first step required will be configuring that central location from where you will push the Ansible configurations. Installing Ansible on a Oracle Linux machine is rather straight forward and can be achieved by following the below steps.

#### Step 1

To be able to install Ansible via the YUM command you will have to ensure that you have the EPEL release RPM installed which will take care of ensuring that you have the fedora YUM repository in place. This is due to the fact that the RPM's for ansible are placed on the fedora repository.

You can do so by first executing a wget to download the file and than install it with the RPM command:

```
 wget http://download.fedoraproject.org/pub/epel/6/x86_64/epel-release-6-8.noarch.rpm
```

```
 rpm -ivh epel-release-6-8.noarch.rpm
```

If done correct you will now have something like the below in your YUM repository directory:

```
[root@localhost ~]# ls -la /etc/yum.repos.d/
total 24
drwxr-xr-x. 2 root root 4096 Aug 25 09:22 .
drwxr-xr-x. 63 root root 4096 Aug 25 08:36 ..
-rw-r--r-- 1 root root 957 Nov 5 2012 epel.repo
-rw-r--r-- 1 root root 1056 Nov 5 2012 epel-testing.repo
-rw-r--r--. 1 root root 7533 Mar 28 10:13 public-yum-ol6.repo
[root@localhost ~]#
```

if you check the epel.repo file you should have at least the "Extra packages for Enterprise Linux 6" channel active. You can see this in the example below:

```
[root@localhost ~]# cat /etc/yum.repos.d/epel.repo
[epel]
name=Extra Packages for Enterprise Linux 6 - $basearch
#baseurl=http://download.fedoraproject.org/pub/epel/6/$basearch
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-6&arch=$basearch
failovermethod=priority
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-6

[epel-debuginfo]
name=Extra Packages for Enterprise Linux 6 - $basearch - Debug
#baseurl=http://download.fedoraproject.org/pub/epel/6/$basearch/debug
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-debug-6&arch=$basearch
failovermethod=priority
enabled=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-6
gpgcheck=1

[epel-source]
name=Extra Packages for Enterprise Linux 6 - $basearch - Source
#baseurl=http://download.fedoraproject.org/pub/epel/6/SRPMS
mirrorlist=https://mirrors.fedoraproject.org/metalink?repo=epel-source-6&arch=$basearch
failovermethod=priority
enabled=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-6
gpgcheck=1
[root@localhost ~]#
```

## Steps 2

As soon as you have completed the needed steps in step 1 you should be able to do an installation of Ansible on Oracle Linux by executing a simple yum install command.

```
yum install ansible
```

## Step 3

In basic your installation should be done and Ansible should be available and ready to be configured. To ensure you have the installation right you can conduct the below test to verify.

```
[root@localhost init.d]# ansible localhost -m ping
[WARNING]: provided hosts list is empty, only localhost is available
```

```
localhost | SUCCESS => {
    "changed": false,
    "ping": "pong"
}
```

```
[root@localhost init.d]#
# cat /etc/hosts
```

```
10.14.11.197 cherry
```

```
# cat /etc/ansible/hosts
```

```
[sblapp]
```

## Cherry

```
[siebel@db root]$ ansible -m ping all
```

```
SITIP16 | SUCCESS => {  
    "changed": false,  
    "ping": "pong"  
}  
[siebel@db root]$ ansible -m raw -a '/usr/bin/uptime' all
```

```
SITIP16 | CHANGED | rc=0 >>  
03:35PM up 10 days, 16:18, 9 users, load average: 2.82, 3.08, 2.99  
Shared connection to sitip16 closed.
```

```
[siebel@db root]$ ansible -m shell -a 'python -V' SITIP16
```

```
SITIP16 | CHANGED | rc=0 >>  
Python 2.7.2
```

## MongoDB

MongoDB is an open-source document database and leading NoSQL database. MongoDB is written in C++. Is a noSQL, nothing but a schema less.

MongoDB is a cross-platform, document oriented database that provides, high performance, high availability, and easy scalability. MongoDB works on concept of collection and document.

### Prerequisites

Before proceeding with this tutorial, you should have a basic understanding of database, text editor and execution of programs, etc. Because we are going to develop high performance database, so it will be good if you have an understanding on the basic concepts of Database (RDBMS).

### Database

Database is a physical container for collections. Each database gets its own set of files on the file system. A single MongoDB server typically has multiple databases.

### Collection [Tables]

Collection is a group of MongoDB documents. It is the equivalent of an RDBMS table. A collection exists within a single database. Collections do not enforce a schema. Documents within a collection can have different fields. Typically, all documents in a collection are of similar or related purpose.

### Document [Schema/row]

A document is a set of key-value pairs. Documents have dynamic schema. Dynamic schema means that documents in the same collection do not need to have the same set of fields or structure, and common fields in a collection's documents may hold different types of data.

The following table shows the relationship of RDBMS terminology with MongoDB.

| RDBMS                      | MongoDB                                                  |
|----------------------------|----------------------------------------------------------|
| Database                   | Database                                                 |
| Table                      | Collection                                               |
| Tuple/Row                  | Document                                                 |
| column                     | Field                                                    |
| Table Join                 | Embedded Documents                                       |
| Primary Key                | Primary Key (Default key _id provided by mongodb itself) |
| Database Server and Client |                                                          |
| Mysqld/Oracle              | mongod                                                   |
| mysql/sqlplus              | mongo                                                    |

### Sample Document

Following example shows the document structure of a blog site, which is simply a comma separated key value pair.

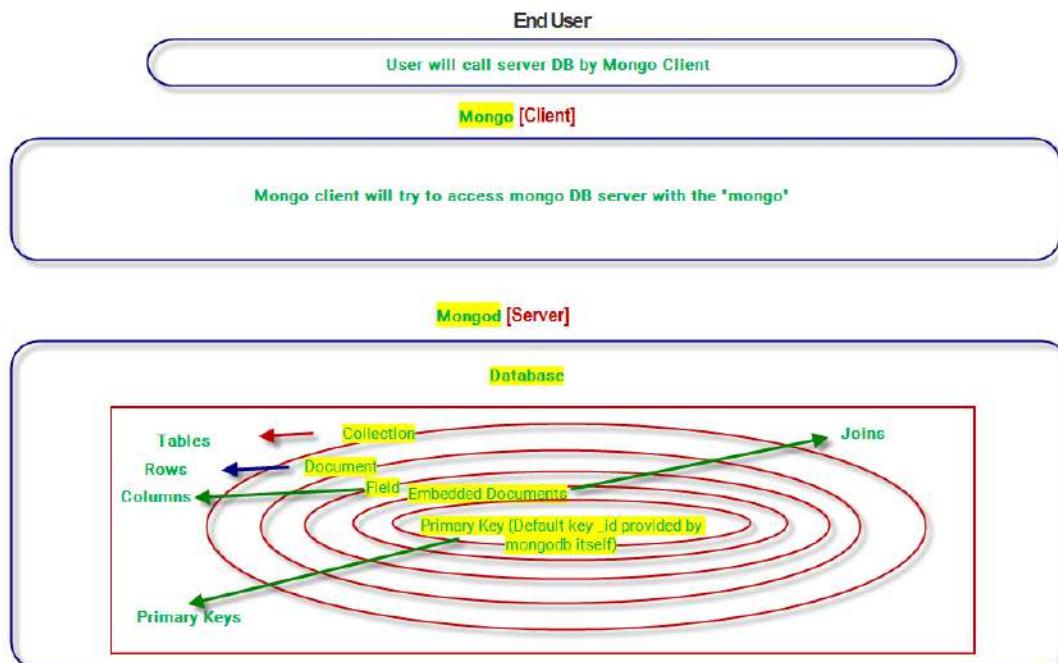
```
{
  _id: ObjectId('7df78ad8902c'),
  title: 'MongoDB Overview',
  description: 'MongoDB is no sql database',
  by: 'tutorials point',
  url: 'http://www.tutorialspoint.com',
  tags: ['mongodb', 'database', 'NoSQL'],
  likes: 100,
  comments: [
    {
      user: 'user1',
      message: 'My first comment',
      dateCreated: new Date(2011, 1, 20, 2, 15),
      like: 0
    },
    {
      user: 'user2',
      message: 'My second comments',
      dateCreated: new Date(2011, 1, 25, 7, 45),
      like: 5
    }
  ]
}
```

**\_id** is a 12 bytes' hexadecimal number which assures the uniqueness of every document. You can provide `_id` while inserting the document. If you don't provide then MongoDB provides a unique id for every document. These 12 bytes first 4 bytes for the current timestamp, next 3 bytes for machine id, next 2 bytes for process id of MongoDB server and remaining 3 bytes are simple incremental VALUE.

1. why MongoDB 05:29
2. How data insert in databases 06:20
3. Collections in mongoDB 03:55
4. Documents in mongoDB 04:48
5. Downloading mongoDB 05:32
6. Installing mongoDB in windows 07:44
7. Checking databases in mongodb 03:28
8. Collections in mongoDB 05:25
9. Documents in mongoDB 05:33
10. Inserting documents in mongodb 08:43
11. Introduction to objects in mongoDB 07:54
12. Arrays in mongoDB 07:54
13. Nested documents in mongoDB 09:39
14. Retrieve data from mongodb 07:34
15. Greater than and less than in mongodb 07:06
16. LIKE in mongoDB 04:27

MongoDb will connect based on drivers to other tools. In mongodb documentation, it available about drivers which will support based on your requirements.

## About MongoDB Structure



## MongoDB vs. RDBMS

- Collection vs. table
- Document vs. row
- Field vs. column
- Collection isn't strict about what goes in it  
(it's schema-less)

MongoDB : DB → Collections → Documents (json format {java script object notation})

```
$array['key']=  
db=>collections  
collection=>documents  
var user = {  
    fname:"Shehzad Ahmed",  
    lname:"Shakzee",  
    age:20  
}
```

A diagram shows a JSON object representing a document. The object has properties: 'fname' with value 'Shehzad Ahmed', 'lname' with value 'Shakzee', and 'age' with value 20. These properties are highlighted with red boxes and circled with green lines. A green arrow points from the circled area to the word 'Values'. Another green arrow points from the same circled area to the word 'Keys'.

## MongoDB installation:

Before getting installing mongodb on windows

```
C:>wmic os get osarchitecture  
OSArchitecture  
64-bit  
C:>  
I a full straight forward, nothing complicated like RDBMS.
```

1. After installation navigate to mongodb installed directory.
2. And navigate to bin direcotry.
3. Create 2 directories i.e, data and log direcotires
4. Navigate to data directory and create db directory.



### To setup mongoDB server:

Cd C:\MongoDB\bin

```
mongod --directoryperdb dbpath C:\MongoDB\data\db --logpath C:\MongoDB\log\mongo.log --logappend --rest --install
```

```
mongod --directoryperdb dbpath C:\MongoDB\data\db --logpath C:\MongoDB\log\mongo.log --logappend --install
```

OR

```
D:\set up\mongodb\bin>mongod.exe --dbpath "C:\MongoDB\data"
```

This will show **waiting for connections** message on the console output, which indicates that the mongod.exe process is running successfully.  
Now to run the MongoDB, you need to open another command prompt and issue the following command.

```
D:\set up\mongodb\bin>mongo.exe
```

## MongoDB Commands Cheatsheet

The following is the list of the commands:

### Start and stop the MongoDB Database:

```
sudo service mongod start  
sudo service mongod stop
```

### Access the MongoDB

### Access the MongoDB database using Shell:

```
mongo --host localhost:27017
```

#### Show all databases:

```
show dbs
```

#### Create a database

### Create a database, say, testdb; Switch to the database:

```
use testdb  
use devopsdb
```

Until a collection is created in a database, the database name is not listed as a result of execution of the command, "show dbs."

#### Add a collection:

```
db.createCollection("user")
```

## Show all collections

Show all collections in a database; Execute the "use dbname" command to access the database before executing the command given below.

```
show collections  
show tables
```

The following command also work:

```
db.getCollectionNames()
```

## Insert a record in the collection

Insert a record in the collection; A record is inserted in the collection, "user."

```
db.user.insert({ "name": "Ajitesh Shukla", "location": "hyderabad", "username": "eajitesh" })
```

## Display list of records of a collection

Display list of records of a collection; "user" collection is used.

```
db.user.find()  
db.user.find().pretty()
```

## Display a list of records matching with value (s) of specific fields:

```
db.user.find({ "username": "eajitesh" })  
db.user.find({ "username": "eajitesh", "location": "hyderabad" })
```

## Drop the collection:

```
db.user.drop()
```

## Create users in the database

Create users in the database; The below command creates a user with username as "ajitesh" and having the role such as "readWrite" and "dbAdmin"

```
db.createUser({ "user": "ajitesh", "pwd": "gurukul", "roles": [ "readWrite", "dbAdmin" ] })  
db.createUser({ "user": "gouse", "pwd": "gouse", "roles": [ "readWrite", "dbAdmin" ] })
```

## Show users

Show users; If executed without selecting a database, it displays all users along with database information.

```
show users
```

## Login into the database with username and password:

```
mongo -u USERNAME -p PASSWORD --authenticationDatabase DATABASENAME  
mongo -u GOUSE -p GOUSE --authenticationDatabase devopsdb → not worked  
mongo -u gouse -p gouse --authenticationDatabase devopsdb → work because case sensitive,  
I have created an user as gouse[small letters]
```

For user created in above command, the login command would look like the following:

```
mongo -u ajitesh -p gurukul --authenticationDatabase testdb
```

## MongoDB Server Service is not starting

### Error :

Windows could not start the MongoDB Server service on Local Computer.

Error 1053: The service did not respond to the start or control request in a timely fashion.

**Solution:**

reinstall the software or create a data and db directories under c drive.

C:\data\db

**Error:**

**Service 'MongoDB Server' (MongoDB) could not be installed.**

**Verify that you have sufficient privileges to install system services.**



**Solution : runs on windows2008server**

[MongoDBCompass :](#)

For some reason in Windows 10 it is installed into the hidden AppData directory. At least I found first a shortcut 'MongoDB Compass' here:

C:\Users\|AppData\Roaming\Microsoft\Windows\Start Menu\Programs\MongoDB Inc  
and then in the properties of it the actual location seems to be  
C:\Users\|AppData\Local\MongoDBCompass\MongoDBCompass.exe

[GlassFish](#)

[Install the Oracle GlassFish Server on Windows](#)

To install the Oracle GlassFish Server on Windows:

1. Go to the following URL:

<http://www.oracle.com/technetwork/middleware/GlassFish/overview/index.html>

or

<https://javaee.github.io/glassfish/download>

<https://www.youtube.com/watch?v=ETExks000Ck>

follows above link, and it will work fine to get access.

```
C:\Windows\system32\cmd.exe
Use "exit" to exit and "help" for online help.
asadmin> start-domain domain1
Waiting for domain1 to start .....
Successfully started the domain : domain1
domain  Location: C:\Gouse_1\Devops\GlassFish\glassfish4\glassfish\domains\domain1
Log File: C:\Gouse_1\Devops\GlassFish\glassfish4\glassfish\domains\domain1\logs\server.log
Admin Port: 4848
Command start-domain executed successfully.
asadmin>
```

<http://localhost:4848/>

<http://localhost:4848/common/index.jsf>

## 2. Select [Click here to download Oracle GlassFish Server \(3.1.2.x\)](#).

Where 3.1.2.x is the latest version available (for example, 3.1.2.2), as shown below:

**Note:** Do not select the Web Profile version of the Oracle GlassFish Server.

The screenshot shows the Oracle Technology Network website. The navigation bar includes links for Sign In/Register for Account, Help, Select Country/Region, Communities, I am a..., I want to..., Search, and Oracle Technology Network. The main menu has categories like PRODUCTS AND SERVICES, SOLUTIONS, DOWNLOADS, STORE, SUPPORT, TRAINING, PARTNERS, and ABOUT. Below the menu, a breadcrumb trail shows Oracle Technology Network > Middleware > GlassFish Server > Overview. On the left, a sidebar lists various Oracle products: Fusion Middleware Home, AIA Foundation Pack, Business Intelligence, Coherence, Developer Tools, Event-Driven Architecture, GlassFish Server, Identity Management, JRockit (highlighted with a red arrow), SOA Suite, TopLink, Tuxedo, WebCenter, WebCenter Content, WebCenter Sites, WebCenter Portal, Social Network, WebLogic Server, Event Processing, Business Intelligence Foundation, Data Integration, Application Server, and Beehive. The main content area features a section for 'GlassFish Server 3.1.2.2 Now Available!' with links to download the server or web profile. Below this, there are sections for 'Technical Information' (with links to supported configurations, datasheets, tech casts, webinars, and podcasts) and 'News & Community' (with links to the GlassFish blog, Facebook, Twitter, and press releases). To the right, a 'Popular Downloads' sidebar lists items like Berkeley DB, Enterprise Manager, Database EE and XE, Developer VMs, Enterprise Pack for Eclipse, Java, JDeveloper and ADF, Oracle Linux and Oracle VM, MySQL, NetBeans IDE, NoSQL Database, Solaris, SQL Developer, VirtualBox, and WebLogic Server. A 'More Middleware Downloads' section includes a link to VM Templates for Middleware. At the bottom right is an 'Oracle Cloud' button with a 'Get Started' link.

## 3. Select [Accept License Agreement](#).

## 4. Select the [Windows Installer](#) download (for example, *ogs-3.1.2.2-windows.exe*).

**Important:** Do not run the installer once it has downloaded.

## 5. Open a command line prompt in the directory where the installer was downloaded (for example, *C:\Users\User1\Downloads*).

## 6. Enter the following command:

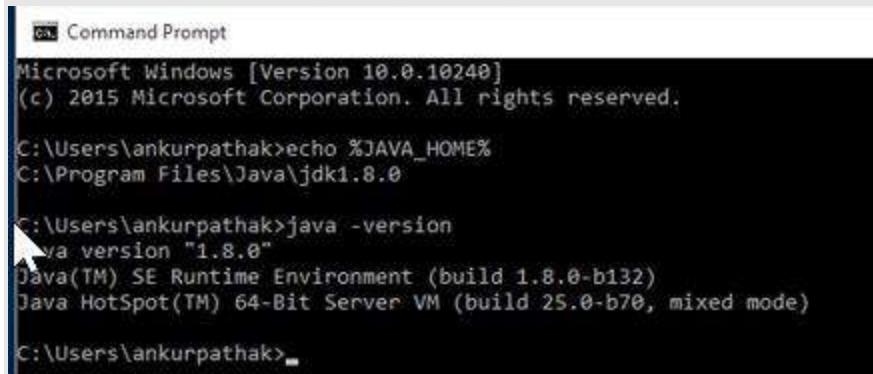
```
C:\Users\User1\Downloads>ogs-<3.1.2.x>-windows.exe -j "<java>"
```

Where:

- <3.1.2.x> is the GlassFish version
- <java> is the location of the JDK on the machine

**Important:** This must be the JDK, not the JRE.

Pre-Checks:



```
Windows [Version 10.0.10240]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Users\ankurpathak>echo %JAVA_HOME%
C:\Program Files\Java\jdk1.8.0

C:\Users\ankurpathak>java -version
java version "1.8.0"
Java(TM) SE Runtime Environment (build 1.8.0-b132)
Java HotSpot(TM) 64-Bit Server VM (build 25.0-b70, mixed mode)

C:\Users\ankurpathak>
```

<https://www.youtube.com/watch?v=WWCsvea8fGk>

For example:

```
C:\Users\User1\Downloads>ogs-3.1.2.2-windows.exe -j "C:\Program
Files\Java\jdk1.7.0_51"
```

```
Cd C:\Users\User1\Downloads>
ogs-3.1.2.2-windows.exe -j "C:\Java\jdk1.8.0_191"
```



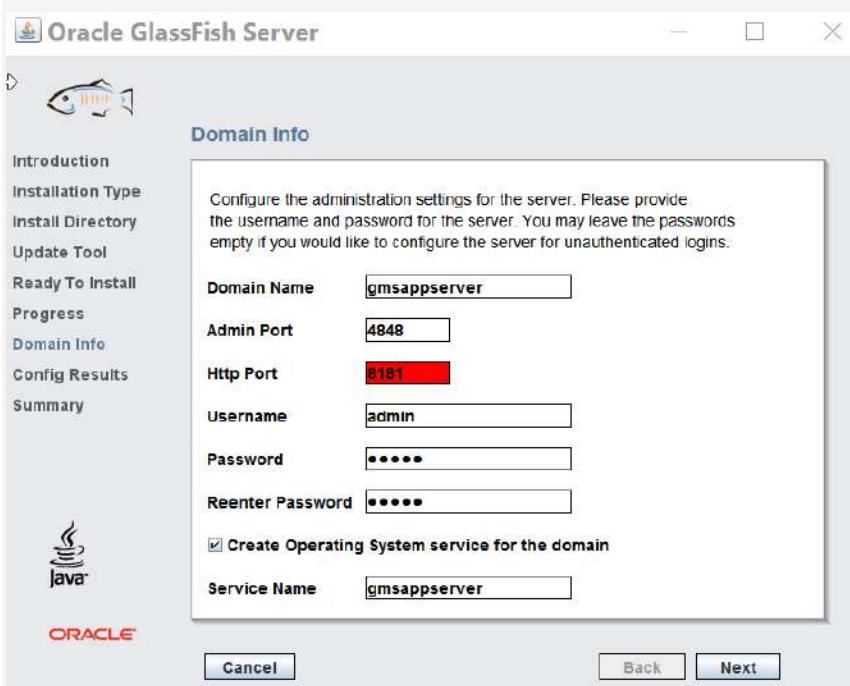
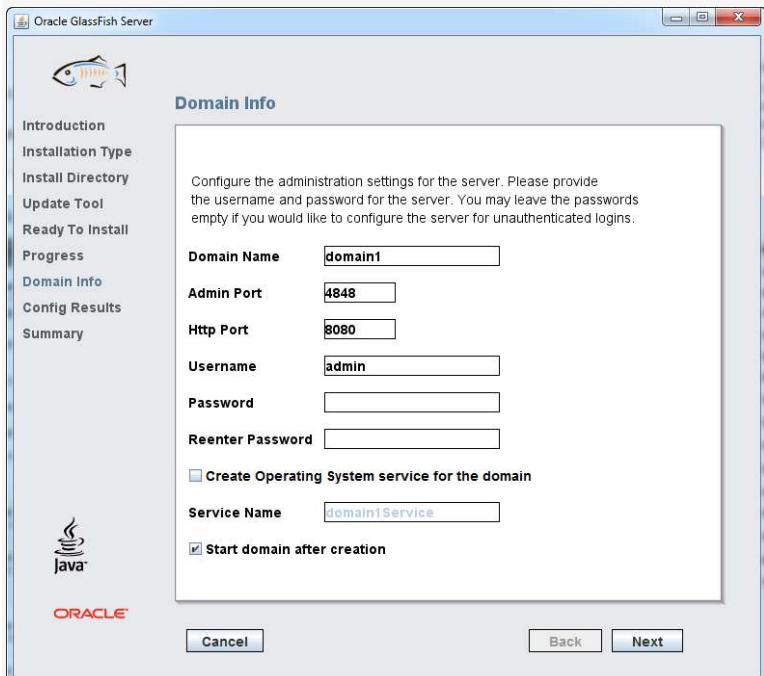
The installation process starts and the Oracle GlassFish Server **Introduction** window is displayed.

7. Click **Next**.
8. Select **Typical Installation** and click **Next**.
9. Keep the default installation directory and click **Next**.
10. **Optional:** If desired, activate the **Install Update Tool** and enter proxy server information as applicable.
11. Click **Next**.



## 12. Click Install.

The Oracle GlassFish Server is installed and the **Domain Info** window is displayed.



13. Enter the GlassFish Server domain information:

- **Domain Name:** Enter cmsappserver
- **Admin Port:** Enter 4848 (keep default)
- **HTTP Port:** Enter a port that is not currently in use (for example, 8080)
- **Username:** Enter the username for accessing the GlassFish Administration Console; recommended value is admin
- **Password:** Enter the password for the GlassFish Administration Console username; for example, admin
- **Reenter Password:** For example, admin
- **Click Create Operating System service for the domain**
- **Service Name:** Enter cmsappserver
- **Keep Start domain after creation checked**

14. Click **Next**.

The domain is configured.



15. Click **Next** and then **Exit** to finish the installation.

16. Enable the access to the GlassFish Administration Console remotely, as follows:

- a. Open a command-line prompt.
- b. Go to the following directory:

%GlassFishDir%/bin/

For example:

```
cd C:\glassfish3\glassfish\bin
```

- c. Enter the following command:

```
asadmin --host localhost --port 4848 enable-secure-admin
```

A message similar to the following is displayed:

You must restart all running servers for the change in secure admin to take effect.

Command enable-secure-admin executed successfully.

17. To restart the GlassFish server, click the Start button and select **All Programs >Oracle GlassFish Server > Start Application Server**.

## ElasticSearch

Install java jdk on server and set the JAVA\_HOME

```
sudo update-alternatives --config java
```

# How to Install Java 11 on CentOS 7/6 & Fedora 30-25

Written by [Rahul](#), Updated on May 2, 2019

[JAVA](#) [java](#), [Java 11](#), [Java 11 LTS](#), [java-install](#)

The Oracle team has discontinued the security updates for Java 8. Also released the latest Java 11 LTS version. Java 11 has been released for general availability with the many featured enhancements. You can read the details [here](#). This article will help you to install Oracle Java 11 on CentOS 7/6, RHEL 7/6 and Fedora 29/28/27/26/25 system.

### Read this:

- [Install Java 12 on CentOS/RHEL & Fedora](#)

## Step 1 – Download Java 11

Download the latest Java SE Development Kit 11 LTS release from its [official download page](#) or use following commands to download from command line.

```
wget --no-cookies --no-check-certificate --header "Cookie: oraclelicense=accept-securebackup-cookie" "http://download.oracle.com/otn-pub/java/jdk/11.0.2+9/f51449fc52f4d52b93a989c5c56ed3c/jdk-11.0.2_linux-x64_bin.rpm"
```

## Step 2 – Install Java 11

After extracting archive file use **alternatives** command to install it. alternatives command is available in **chkconfig** package.

```
rpm -Uvh jdk-11.0.2_linux-x64_bin.rpm
```

```
alternatives --config java
```

There are 2 programs which provide 'java'.

| Selection | Command                       |
|-----------|-------------------------------|
| + 1       | /opt/jdk1.8.0_201/bin/java    |
| * 2       | /usr/java/jdk-11.0.2/bin/java |

Enter to keep the current selection[+], or type selection number: **2**

According to the above screenshot, there are 3 versions installed. Java 11 is listed on number 2, So input numeric 2 and press Enter. Now Java 11 is configured as default Java on my CentOS and Fedora system.

## Step 3 – Check Java Version

Check the installed Java version on your system using the following command.

```
java -version
```

```
java version "11.0.2" 2019-01-15 LTS
```

```
Java(TM) SE Runtime Environment 18.9 (build 11.0.2+9-LTS)
```

```
Java HotSpot(TM) 64-Bit Server VM 18.9 (build 11.0.2+9-LTS, mixed mode)
```

## Step 4 – Setup Java Environment Variables

Most of the Java-based applications uses environment variables to work. Set the Java environment variables using the following commands

- Setup **JAVA\_HOME** Variable

```
export JAVA_HOME=/usr/java/jdk-11.0.2
```

- Setup **PATH** Variable

```
export PATH=$PATH:/usr/java/jdk-11.0.2/bin
```

Also put all above environment variables in **/etc/environment** file for auto loading on system boot.

# How to Install Elasticsearch on Fedora & CentOS

Written by [Rahul](#), Updated on June 4, 2019

[BIG-DATA](#) [BIG-DATA](#), [cluster](#), [data](#), [database](#), [elasticsearch](#), [Elasticsearch Cluster](#)

**Elasticsearch** is flexible and powerful open source, distributed real-time search and analytics engine. Using a simple set of APIs, it provides the ability for full-text search. Elastic search is freely available under the Apache 2 license, which provides the most flexibility.

This tutorial will help you to setup Elasticsearch single node cluster on CentOS, Red Hat, and Fedora systems.

## Step 1 – Prerequisites

Java is the primary requirement for installing Elasticsearch on any system. You can check the installed version of Java by executing the following command. If it returns an error, install Java on your system using [this tutorial](#).

```
java -version
```

## Step 2 – Download Elasticsearch

Now download the latest Elasticsearch archive from its [official download page](#). At the time of last update of this article Elasticsearch 7.1.1 version is latest version available to download. After download extract the archive.

```
wget https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-7.1.1-linux-x86_64.tar.gz  
tar xzf elasticsearch-7.1.1-linux-x86_64.tar.gz
```

## Step 3 – Configure Elasticsearch

Now we need to set up the Elasticsearch cluster and node name. Elasticsearch uses “elasticsearch” as default cluster name, We recommend to change it as per your setup.

```
mv elasticsearch-7.1.1 /usr/share/elasticsearch  
cd /usr/share/elasticsearch
```

To change cluster named edit **config/elasticsearch.yml** file and update following values. Node names are dynamically generated, but to keep a fixed user-friendly

name change it also. You may also need to change network host to access elasticsearch from remote hosts.

```
vim config/elasticsearch.yml
```

```
network.host: 192.168.10.100
```

```
cluster.name: TecAdmin_Cluster1
```

```
node.name: "California Datacenter"
```

```
#####
# Cluster name identifies your cluster for auto-discovery. If you're running
# multiple clusters on the same network, make sure you're using unique names.
#
cluster.name: TecAdmin_Cluster1

#####
# Node names are generated dynamically on startup, so you're relieved
# from configuring them manually. You can tie this node to a specific name:
#
node.name: "California DataCenter"
```

## Step 4 – Start Elasticsearch and Test

As the Elasticsearch setup is completed. Let the start Elasticsearch cluster using following command.

```
./bin/elasticsearch &
```

You have all done, just need to verify setup. Elasticsearch works on port default port 9200, open your browser to point your server on port 9200, You will find something like below output

```
http://localhost:9200/
```



A screenshot of a web browser window titled "localhost:9200". The address bar shows "localhost:9200" and "svr1.tecadmin.net:9200". The page content displays the following JSON response:

```
{  
  "status" : 200,  
  "name" : "California DataCenter",  
  "cluster_name" : "TecAdmin_Cluster1",  
  "version" : {  
    "number" : "1.7.0",  
    "build_hash" : "929b9739cae115e73c346cb5f9a6f24ba735a743",  
    "build_timestamp" : "2015-07-16T14:31:07Z",  
    "build_snapshot" : false,  
    "lucene_version" : "4.10.4"  
  },  
  "tagline" : "You Know, for Search"  
}
```

[ TecAdmin.net ]

## Step 5 – Elasticsearch Uses Examples

Following examples will help you to add, fetch and search data in Elasticsearch cluster.

### Create New Bucket

```
curl -XPUT http://localhost:9200/mybucket
```

*Output:*

```
{"acknowledged":true}
```

### Adding Data to Elasticsearch

Use following commands to add some data in Elasticsearch.

**Command 1:**

```
curl -XPUT 'http://localhost:9200/mybucket/user/johny' -d '{ "name" : "Rahul Kumar" }'
```

*Output:*

```
{"_index": "mybucket", "_type": "user", "_id": "johny", "_version": 1, "created": true}
```

**Command 2:**

```
curl -XPUT 'http://localhost:9200/mybucket/post/1' -d '  
{  
    "user": "Rahul",  
    "postDate": "01-15-2015",  
    "body": "This is Demo Post 1 in Elasticsearch" ,  
    "title": "Demo Post 1"  
}'
```

*Output:*

```
{"_index": "mybucket", "_type": "post", "_id": "1", "_version": 1, "created": true}
```

**Command 3:**

```
curl -XPUT 'http://localhost:9200/mybucket/post/2' -d '  
{  
    "user": "TecAdmin",  
}'
```

```
"postDate": "01-15-2015",  
  
"body": "This is Demo Post 2 in Elasticsearch" ,  
  
"title": "Demo Post 2"  
  
}'
```

### *Output:*

```
{"_index":"mybucket","_type":"post","_id":"2","_version":1,"created":true}
```

## Fetching Data from Elasticsearch

Use the following command to GET data from ElasticSearch and read the output.

```
curl -XGET 'http://localhost:9200/mybucket/user/johny?pretty=true'  
  
curl -XGET 'http://localhost:9200/mybucket/post/1?pretty=true'  
  
curl -XGET 'http://localhost:9200/mybucket/post/2?pretty=true'
```

## Searching in Elasticsearch

Use the following command to search data from elastic search. Below command will search all data associated with user johny.

```
curl 'http://localhost:9200/mybucket/post/_search?q=user:TecAdmin&pretty=true'
```

### *Output:*

```
{  
  
"took" : 145,
```

```
"timed_out" : false,  
  
"_shards" : {  
  
    "total" : 5,  
  
    "successful" : 5,  
  
    "failed" : 0  
  
},  
  
"hits" : {  
  
    "total" : 1,  
  
    "max_score" : 0.30685282,  
  
    "hits" : [ {  
  
        "_index" : "mybucket",  
  
        "_type" : "post",  
  
        "_id" : "2",  
  
        "_score" : 0.30685282,  
  
        "_source":  
  
    {  
  
        "user": "TecAdmin",  
  
        "postDate": "01-15-2015",  
  
        "body": "This is Demo Post 2 in Elasticsearch" ,  
  
    }  
}]}
```

```
        "title": "Demo Post 2"  
  
    }  
  
}  
  
]  
  
}  
  
}
```

Congratulation's! You have successfully configured elasticsearch single node cluster on your Linux system.

## How to Install and Configure Elasticsearch on Linux and Windows



What is Elasticsearch? In simple terms, we could possibly say elastic search is a NoSQL database. Since there are so many NoSQL databases, let us understand how Elasticsearch is different from them.

### Introduction to Elasticsearch

Elasticsearch is a real time, document based, distributed, NoSQL database, full text based search engine, and a powerful analytics engine, it is REST based. The following are the key features of elasticsearch.

- 1. Real Time:** Inserting and retrieving data from elasticsearch is super-fast, it's called near real time data retrieval. It is useful in low latency web applications, which has large amount of data to process.
- 2. Document Based:** Elasticsearch is schema less database. It stores JSON documents without knowing schema in advance. At run time it can infer from the data inserted what should be its mapping.

**3. Distributed:** Elasticsearch is distributed database. It is clustered. Data is distributed across multiple nodes to avoid single point of failure. If one node goes down then it can recover data from other nodes.

**4. NoSQL database:** Elasticsearch is NoSQL database like Mongo, Redis. It supports only JSON documents insertion and retrieval.

**5. Full text based search:** Full text is advanced way of searching occurrence of a term in documents, without scanning whole document. It works by storing text indexes for all the terms in document.

**6. Analytics engine:** Elasticsearch provides tools, APIs, to analyze the stored documents. We can search for popular patterns, metrics, reporting, and powerful data charting dashboard support.

**7. REST based APIs:** Elasticsearch only uses REST based APIs to insert and retrieve data. GET, PUT, DELETE, POST.

## Download Elasticsearch

Download the latest version of the install package for respective platform from [Elasticsearch download site](#)

The install process is same for both Linux and Windows. In Windows, it'll be Zip file. In Linux, it'll be .tar.gz file. Also, in the .yml file make sure the path follow the correct syntax for your corresponding OS.

On Linux:

```
wget https://artifacts.elastic.co/downloads/elasticsearch/elasticsearch-6.7.0.tar.gz
```

```
tar xvfz elasticsearch-6.7.0.tar.gz
```

On Windows:

```
C:> unzip elasticsearch-6.7.0.zip
```

```
Archive: elasticsearch-6.7.0.zip
```

```
creating: elasticsearch-6.7.0/lib/
```

```
inflating: elasticsearch-6.7.0/lib/elasticsearch-6.7.0.jar
```

```
inflating: elasticsearch-6.7.0/lib/elasticsearch-x-content-6.6.2.jar
```

It will extract the all the JARs required

```
creating: elasticsearch-6.7.0/logs/  
creating: elasticsearch-6.7.0/plugins/
```

It will create all the required folders for the elasticsearch.

The following is the minimum requirements to set up elastic search.

## Modify elasticsearch.yml file

Open elasticsearch/config/elasticsearch.yml configuration file and edit following configuration options

```
vi elasticsearch-6.7.0/config/elasticsearch.yml
```

```
# Use a descriptive name for your cluster:
```

```
#
```

```
cluster.name: TGS_CLUSTER
```

```
# Use a descriptive name for the node:
```

```
#
```

```
node.name: TGS-1
```

```
# Path to directory where to store the data
```

```
# (separate multiple locations by comma):
```

```
#
```

```
path.data: /root/tgs/data
```

```
# Path to log files:
```

```
#  
  
path.logs: /root/tgs/logs  
  
# Set the bind address to a specific IP (IPv4 or IPv6):  
  
#  
  
network.host: 127.0.0.1  
  
# Set a custom port for HTTP:  
  
#  
  
http.port: 9200
```

As shown in the above example, in the elasticsearch.yml file, set the following values appropriately:

- Cluster name
- Node name
- Data path
- Path to logs
- Bind Address
- Custom Port

## Modify Elasticsearch jvm.options

To set up Java options, open elasticsearch/config/jvm.options file and set the memory to be allocated, for heap memory

-Xms1g

-Xmx1g

- Xms represents the initial size of total heap space
- Xmx represents the maximum size of total heap space

Make sure your system has minimum enough memory, otherwise elasticsearch will be super slow, or, would not work at all or might throw exceptions after a while.

Further java options are advanced options, editing them without deep understanding would cause unexpected behavior.

## Start Elasticsearch

Execute following command on shell or windows command prompt, from top level elasticsearch directory

On Linux:

```
cd /root/elasticsearch-6.7.0  
bin/elasticsearch
```

On Windows:

```
C:\elasticsearch-6.7.0> bin\elasticsearch
```

There will be series of log but important part is the following:

```
[2019-03-23T12:32:23,196][INFO ][o.e.n.Node] [TGS-1] initialized  
[2019-03-23T12:32:23,196][INFO ][o.e.n.Node] [TGS-1] starting ...  
[2019-03-23T12:32:24,001][INFO ][o.e.t.TransportService] [TGS-1] publish_address {127.0.0.1:9300}, bound_addresses {127.0.0.1:9300}  
[2019-03-23T12:33:52,437][INFO ][o.e.c.s.MasterService] [TGS-1] zen-disco-elected-as-master ([0] nodes joined), reason: new_master {TGS-1}{Q97KJ6A2QW2i0ehLL1nwWg}  
{nR6YOn0dTY-6VCn8y0rtEQ}{127.0.0.1}{127.0.0.1:9300}  
{ml.machine_memory=8521035776, xpack.installed=true, ml.max_open_jobs=20, ml.enabled=true}  
[2019-03-23T12:33:52,444][INFO ][o.e.c.s.ClusterApplierService] [TGS-1] new_master {TGS-1}  
{Q97KJ6A2QW2i0ehLL1nwWg}{nR6YOn0dTY-6VCn8y0rtEQ}{127.0.0.1}{127.0.0.1:9300}  
{ml.machine_memory=8521035776, xpack.installed=true, ml.max_open_jobs=20, ml.enabled=true},
```

```
reason: apply cluster state

(from master [master {TGS-1}{Q97KJ6A2QW2i0ehLL1nwWg}

{nR6YOn0dTY-6VCn8yOrtEQ}{127.0.0.1}{127.0.0.1:9300}

{ml.machine_memory=8521035776, xpack.installed=true, ml.max_open_jobs=20, ml.enabled=true}
committed version [1] source [zen-disco-elected-as-master ([0] nodes joined)])
```

```
[2019-03-23T12:33:52,627][INFO ][o.e.h.n.Netty4HttpServerTransport]
```

```
[TGS-1] publish_address {127.0.0.1:9200}, bound_addresses {127.0.0.1:9200}
```

```
[2019-03-23T12:33:52,628][INFO ][o.e.n.Node] [TGS-1] started
```

In the above output, TGS-1 is started, means our elasticsearch is up.

This setup only contains single node. This is not a cluster.

## Verify Elasticsearch Installation

Elasticsearch has REST based cluster management interfaces, and it provides REST APIs to manage the cluster, using curl or web browser we can check the state of cluster.

Using Curl:

```
$ curl -XGET http://127.0.0.1:9200/
```

Talking about curl, you might find this helpful: [wget vs curl: How to Download Files Using wget and curl](#)

The following is the output of the above curl command:

```
{
  "name" : "TGS-1",
  "cluster_name" : "TGS-CLUSTER",
```

```
"cluster_uuid" : "QHwc5GpmQbuc5Tjg30pTjA",

"version" : {

"number" : "6.6.2",

"build_flavor" : "default",

"build_type" : "zip",

"build_hash" : "3bd3e59",

"build_date" : "2019-03-06T15:16:26.864148Z",

"build_snapshot" :* false,

"lucene_version" : "7.6.0",

"minimum_wire_compatibility_version" : "5.6.0",

"minimum_index_compatibility_version" : "5.0.0"

},



>tagline" : "You Know, for Search"

}
```

Just enter following link in browser <http://127.0.0.1:9200/> , hit enter, you'll see the same output as above.

## Installing On Debian / Ubuntu

Installing from debian repository

```
santosh@ubuntu:~$ sudo apt-get install elasticsearch

[sudo] password for santosh:

Reading package lists... Done

Building dependency tree
```

```
Reading state information... Done
```

It will pull deb packages from repositories, and will install elasticsearch.

Follow similar steps discussed above to configure, and to verify the setup of elasticsearch.

Elasticsearch would not start automatically after installing, you will have to start the elasticsearch manually or register it to system services and run service commands to execute it.

Directory structure will be different on Debian / Ubuntu.

Configuration files:

- /etc/elasticsearch/elasticsearch.yml
- /etc/elasticsearch/jvm.options

Default configuration settings are good to go, but if you want to modify the default setting, edit the configuration files at respective paths and restart the elasticsearch.

## Groovy Installation

These instructions describe how to install a binary distribution of Groovy.

- First, [Download](#) a binary distribution of Groovy and unpack it into some file on your local file system.
- Set your `GROOVY_HOME` environment variable to the directory you unpacked the distribution.
- Add `GROOVY_HOME/bin` to your `PATH` environment variable.
- Set your `JAVA_HOME` environment variable to point to your JDK. On OS X this is `/Library/Java/Home`, on other unixes its often `/usr/java` etc. If you've already installed tools like Ant or Maven you've probably already done this step.

You should now have Groovy installed properly. You can test this by typing the following in a command shell:

```
groovysh
```

Which should create an interactive groovy shell where you can type Groovy statements. Or to run the [Swing interactive console](#) type:

```
groovyConsole
```

To run a specific Groovy script type:

```
groovy SomeScript
```

## Gradle

Gradle for building Mobile App APK file generation steps from command line.

Cd c:\<your developer provided project directory>

cd C:\MobileApp

\* make sure to cross check JAVA supported version for gradle

**Gradle 5.1.1 requires Java 8 or later to run. You are currently using Java 7.**

C:\MobileApp> gradle tasks

```
c:\MobileApp>gradle tasks
<-----> 0% CONFIGURING [11s]
> root project > Resolve dependencies of :classpath
> IDLE
```

```
ktlintDebugCheck - Runs ktlint on all kotlin sources for android debug variant in this project.
ktlintReleaseCheck - Runs ktlint on all kotlin sources for android release variant in this project.
lint - Runs lint on all variants.
lintDebug - Runs lint on the Debug build.
lintRelease - Runs lint on the Release build.
lintVitalRelease - Runs lint on just the fatal issues in the release build.
test - Run unit tests for all variants.
testDebugUnitTest - Run unit tests for the debug build.
testReleaseUnitTest - Run unit tests for the release build.
```

To see all tasks and more detail, run gradle tasks --all

To see more detail about a task, run gradle help --task <task>

Deprecated Gradle features were used in this build, making it incompatible with Gradle 6.0.  
Use '--warning-mode all' to show the individual deprecation warnings.  
See [https://docs.gradle.org/5.5/userguide/command\\_line\\_interface.html#sec:command\\_line\\_warnings](https://docs.gradle.org/5.5/userguide/command_line_interface.html#sec:command_line_warnings)

```
BUILD SUCCESSFUL in 5m 16s
1 actionable task: 1 executed
c:\MobileApp>-
```

**As per above fig, it will download all dependencies.**

You can execute all the build tasks available to your Android project using the Gradle wrapper command line tool. It's available as a batch file for Windows (gradlew.bat) and a shell script for Linux and Mac (gradlew.sh), and it's accessible from the root of each project you create with Android Studio.

To run a task with the wrapper, use one of the following commands from a Terminal window (from Android Studio, select View > Tool Windows > Terminal):

On Windows:

**gradlew task-name**

On Mac or Linux:

```
./gradlew task-name
```

To see a list of all available build tasks for your project, execute tasks:

```
gradlew tasks
```

```
1 actionable task: 1 executed
c:\MobileApp>gradlew tasks
Starting a Gradle Daemon (subsequent builds will be faster)

> Configure project :app
NDK is missing a "platforms" directory.
If you are using NDK, verify the ndk.dir is set to a valid NDK directory. It is C:\android_ndk.
If you are not using NDK, unset the NDK variable from ANDROID_NDK_HOME or local.properties, or
set ANDROID_NDK_DIR to point to your NDK directory.

Observed package id 'add-ons;addon-google_apis-google-19' in inconsistent location
    at [C:\android_sdk\sdk\add-ons\addon-google_apis-google-19] (Expected 'C:\android_sdk\sdk\add-ons\addon-google_apis-google-19-1')
Observed package id 'build-tools;20.0.0' in inconsistent location 'C:\android_sdk\sdk\build-tools\20.0.0'
registerResGeneratingTask is deprecated, use registerGeneratedResFolders(FileCollection)
registerResGeneratingTask is deprecated, use registerGeneratedResFolders(FileCollection)

> Configure project :RecyclerBanner
NDK is missing a "platforms" directory.
If you are using NDK, verify the ndk.dir is set to a valid NDK directory. It is C:\android_ndk.
If you are not using NDK, unset the NDK variable from ANDROID_NDK_HOME or local.properties, or
set ANDROID_NDK_DIR to point to your NDK directory.

<=====> 66% CONFIGURING [45s]
> :app > Resolve dependencies of :app:debugRuntimeClasspath
```

Make sure to execute in new command CMD session.

```
lintVitalRelease - Runs lint on just the fatal issues in the release build.
test - Run unit tests for all variants.
testDebugUnitTest - Run unit tests for the debug build.
testReleaseUnitTest - Run unit tests for the release build.

To see all tasks and more detail, run gradlew tasks --all

To see more detail about a task, run gradlew help --task <task>

Deprecated Gradle features were used in this build, making it incompatible with Gradle 6.0.
Use '--warning-mode all' to show the individual deprecation warnings.
See https://docs.gradle.org/5.1.1/userguide/command_line_interface.html#sec:command_line_warnings

BUILD SUCCESSFUL in 22s
1 actionable task: 1 executed
c:\MobileApp>
```

The rest of this page describes the basics to build and run your app with the Gradle wrapper. For more information about how to set up your Android build, see [Configure your build](#).

If you'd prefer to use the Android Studio tools instead of the command line tools, see [Build and run your app](#).

[About build types](#)

By default, there are two build types available for every Android app: one for debugging your app—the debug build—and one for releasing your app to users—the release build. The resulting output from each build must be signed with a certificate before you can deploy your app to a device. The debug build is automatically signed with a debug key provided by the SDK tools (it's insecure and you cannot publish with it to the Google Play Store), and the release build must be signed with your own private key.

If you want to build your app for release, it's important that you also sign your app with the appropriate signing key. If you're just getting started, however, you can quickly run your apps on an emulator or a connected device by building a debug APK.

You can also define a custom build type in your `build.gradle` file and configure it to be signed as a debug build by including `debuggable true`. For more information, see [Configure Build Variants](#).

### Build and deploy an APK

Although building an app bundle is the best way to package your app and upload it to the Play Console, building an APK is better suited for when you want quickly test a debug build or share your app as a deployable artifact with others.

### Build a debug APK

For immediate app testing and debugging, you can build a debug APK. The debug APK is signed with a debug key provided by the SDK tools and allows debugging through adb.

To build a debug APK, open a command line and navigate to the root of your project directory. To initiate a debug build, invoke the `assembleDebug` task:

```
gradlew assembleDebug
```

```
c:\MobileApp>gradlew assembleDebug
```

```
BUILD SUCCESSFUL in 4m 22s  
56 actionable tasks: 56 executed  
c:\MobileApp>
```

This creates an APK named `module_name-ebug.apk` in `project_name/module_name/build/outputs/apk/`. The file is already signed with the debug key and aligned with zipalign, so you can immediately install it on a device.

Or to build the APK and immediately install it on a running emulator or connected device, instead invoke `installDebug`:

```
gradlew installDebug
```

The "Debug" part in the above task names is just a camel-case version of the build variant name, so it can be replaced with whichever build type or variant you want to assemble or install. For example, if you have a "demo" product flavor, then you can build the debug version with the `assembleDemoDebug` task.

To see all the build and install tasks available for each variant (including uninstall tasks), run the `tasks` task.

Also see the section about how to run your app on the emulator and run your app on a device.

### Build a release APK

When you're ready to release and distribute your app, you must build a release APK that is signed with your private key. For more information, go to [section about how to sign your app from the command line](#).

Deploy your app to the emulator

To use the Android Emulator, you must create an Android Virtual Device (AVD) using Android Studio.

Once you have an AVD, start the Android Emulator and install your app as follows:

In a command line, navigate to android\_sdk/tools/ and start the emulator by specifying your AVD:

**emulator -avd avd\_name**

If you're unsure of the AVD name, execute `emulator -list-avds`.

Now you can install your app using either one of the Gradle install tasks mentioned in the section about how to build a debug APK or the adb tool.

If the APK is built using a developer preview SDK (if the targetSdkVersion is a letter instead of a number), you must include the -t option with the install command to install a test APK.

**adb install path/to/your\_app.apk**

All APKs you build are saved in `project_name/module_name/build/outputs/apk/`.

For more information, see Run Apps on the Android Emulator.

## Apache Ant & Apache IVY

<https://www.codetab.org/tutorial/apache-ivy/installation/>

To work through the tutorial, we require Apache Ivy, Ant and IvyDE, and this chapter covers installation of software.

| Software                | Version | Size (approximate) |
|-------------------------|---------|--------------------|
| Apache Ivy              | 2.2     | 1 MB               |
| Apache Ant              | 1.8.2   | 8 MB               |
| IvyDE<br>Eclipse Plugin |         | 8 MB               |

### Apache Ant and Apache Ivy Installation

We require Apache Ant and Apache Ivy to work through the tutorial. Download latest distribution of Apache Ivy and Ant from Apache site. In case Ant is already installed in your system, ignore the its installation commands. Following commands installs Apache Ant at /opt/ant, but any other location is perfectly fine. Adjust settings accordingly.

```
tar -C /opt/ant/ -xzvf apache-ant-1.8.2-bin.tar.gz
tar -xzvf apache-ivy-2.2.0-bin-with-deps.tar.gz
cp apache-ivy-2.2.0/ivy-2.2.0.jar /opt/ant/apache-ant-1.8.2/lib
export ANT_HOME=/opt/ant/apache-ant-1.8.2
export PATH=$PATH:$ANT_HOME/bin
```

Environment variables ANT\_HOME and PATH are essential to run Ant. These exports may be moved to .bash\_profile in Linux so that they are always set when you login.

### Work Dir

Throughout the tutorial, we use a work directory to run Apache Ivy examples, This directory is some location like \$HOME/work, but you are free to choose any other location. We will be using phrases like "create a file in work dir", "add a ivy.xml to workdir" and "run from work dir" etc., which simply means do that in the work directory.

### Test the Apache Ivy installation

To test the Ant and Apache Ivy installation, add following build.xml file to work dir.

```
build.xml
<project name="test ivy" default="test" xmlns:ivy="antlib:org.apache.ivy.ant">
  <target name="test" description="Test ivy installation">
    <ivy:settings />
  </target>
</project>
```

and run Ant.

ant

Successful build indicates that ant and ivy installation is fine.

### Build Error

Typical error installation error is

build.xml:5: Problem: failed to create task or type antlib:org.apache.ivy.ant:settings

check the following to resolve this

- environment variable ANT\_HOME is not set properly
- ivy.jar is missing from ANT\_HOME/lib

### Install IvyDE Eclipse Plugin

IvyDE Plugin for Eclipse integrates Apache Ivy with Eclipse IDE. From IvyDE Plugin for Eclipse onwards we use the IvyDE plugin to enable dependency management of Eclipse projects. Install IvyDE using one of the following installation methods.

**Update Site** : To install IvyDE, start Eclipse and go to **Help → Install New Software ...** and it displays the Available Softwares window. In **Work with** text box, enter IvyDE update

site <http://www.apache.org/dist/ant/ivyde/updatesite/> and click Add and then in Add Repository window click OK. It displays the following plugins.

- Apache Ivy library
- Apache IvyDE Eclipse plugin

Select both and proceed. This will install Apache Ivy, Ivy Ant Tasks and IvyDE.

**Manual Installation** : This method involves the installation of two plugins - IvyDE and Apache Ivy plugins.

**IvyDE Plugin** : This plugin links core Ivy to Eclipse. Download the IvyDE plugin

from <https://ant.apache.org/ivy/ivyde/download.html>. For Eclipse Indigo download apache-ivyde- 2.1.0.201008101807-RELEASE.tar.gz and uncompress. to get two folders containing the artifacts to deploy. Copy the plugin and features to Eclipse installation directory as follows:

- plugins/org.apache.ivyde.eclipse\_2.1.0.201008101807-RELEASE.jar to \$ECLIPSE\_HOME/plugins
- features/org.apache.ivyde.feature\_2.1.0.201008101807-RELEASE.jar to \$ECLIPSE\_HOME/features

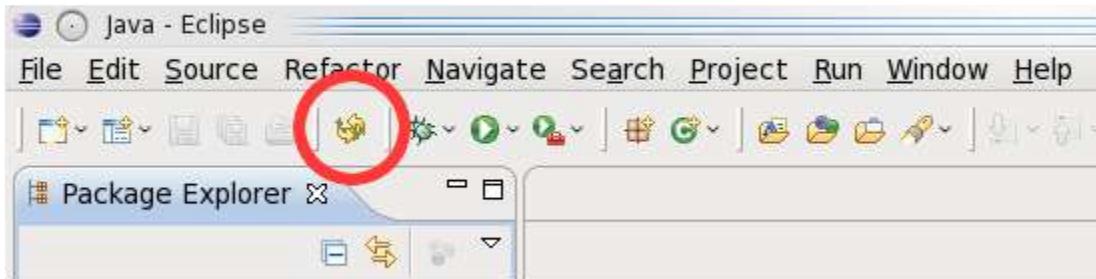
**Apache Ivy Plugin** : This plugin contains core Apache Ivy and Ant Tasks. Get the latest Ivy feature and plugin from following sites

- <http://www.apache.org/dist/ant/ivyde/updatesite>

For Eclipse Indigo download and copy them to your eclipse installation directory as follows

- org.apache.ivy\_2.2.0.final\_20100923230623.jar to \$ECLIPSE\_HOME/plugin
- org.apache.ivy.eclipse.ant\_2.2.0.final\_20100923230623.jar to \$ECLIPSE\_HOME/plugin
- org.apache.ivy.feature\_2.2.0.final\_20100923230623.jar to \$ECLIPSE\_HOME/features

After installation, restart Eclipse and menu bar should show the Resolve button of IvyDE.



With these installations we are ready to Automate the dependency management with Apache Ivy.

[ELK Installation Guide](#)

## Summary

Beat Installation guide is only a reference guide for metric beat & file beat Elastic components.

## Beat Overview

Beats is the platform for single-purpose data shippers. They send data from hundreds or thousands of machines and systems to Logstash or Elasticsearch.

Beats consists of many members as a family like: Filebeat, Metricbeat, Heartbeat etc.

These are lightweight data shippers, which sits on server, gather data and then centralize data into Elasticsearch.

To maintain a strong muscle for processing, it can also ship the data to Logstash for transformation and parsing. It gathers the logs and metrics from unique environments and document them with essential metadata from hosts, container platforms like Docker and Kubernetes, and cloud providers before shipping them to the Elastic Stack.

## Metricbeat

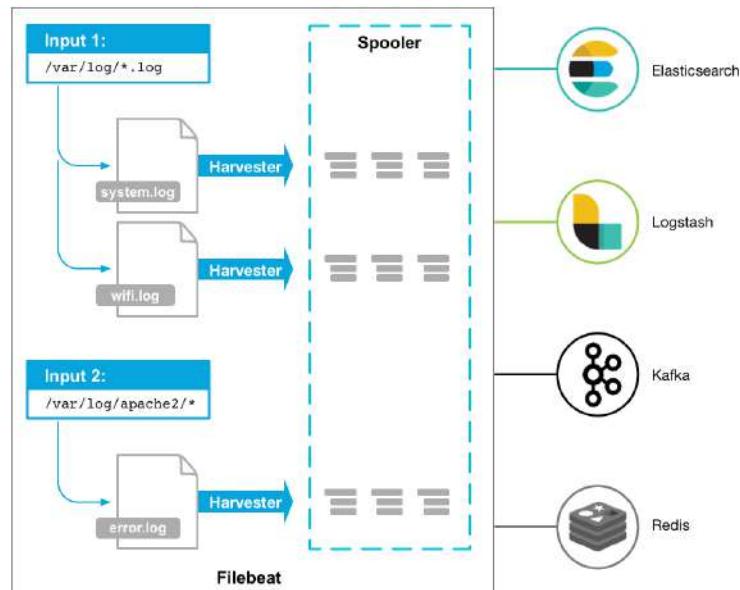
Metricbeat is a lightweight shipper that is installed on application / any server and periodically collects metrics from the operating system and from services running on the server. Metricbeat takes the metrics and statistics that it collects and ships them to the output that you specify, such as Elasticsearch or Logstash.

Metricbeat helps to monitor servers by collecting metrics from the system and services running on the server

## Filebeat

Filebeat is a lightweight shipper for forwarding and centralizing log data. Installed as an agent on application servers, Filebeat monitors the log files or locations that you specify, collects log events, and forwards them to either to Elasticsearch or Logstash for indexing.

Here's how Filebeat works: When you start Filebeat, it starts one or more inputs that look in the locations you've specified for log data. For each log that Filebeat locates, Filebeat starts a harvester. Each harvester reads a single log for new content and sends the new log data to libbeat, which aggregates the events and sends the aggregated data to the output that you've configured for Filebeat.



## Installation & Configuration Steps

### Metric beat

#### Installation Steps

1. Download the metricbeat-7.2.0-x86\_64.rpm file from below link

<https://teams.microsoft.com/#/files/General?threadId=19%3Ac24766ec82e347628ff39dbdb733d6e9%40thread.skype&ctx=channel&context=ELK%2520Installer%2520Files>

[It can also be downloaded from the elastic website URL

<https://www.elastic.co/downloads/beats/metricbeat> ]

2. Create user "metricbeat" user on the server where metricbeat needs to be installed.
3. Connect to the application server using "metricbeat" user where metricbeat needs to be installed.
4. Provide the below listed sudo permissions to "metricbeat" user
  - `/usr/bin/rpm -ivh metricbeat*`,
  - `/usr/bin/rpm -e metricbeat*`,

- /usr/bin/systemctl status metricbeat\*,
  - /usr/bin/systemctl status -l metricbeat\*,
  - /usr/bin/systemctl stop metricbeat\*,
  - /usr/bin/systemctl start metricbeat\*,
  - /usr/bin/journalctl -u metricbeat
5. Copy .rpm file into the application server (\home\<username> folder). Make sure you have single metrickbeat RPM file exists into \home\<username> folder
  6. Go to “\Home\<username>” folder and then run the below command
    - i. sudo /usr/bin/rpm -vih metricbeat\*
  7. Check the below paths are created

| Type          | Description                                      | Location                  |
|---------------|--------------------------------------------------|---------------------------|
| <b>Home</b>   | Home of the Metricbeat installation.             | /usr/share/metricbeat     |
| <b>Bin</b>    | The location for the binary files.               | /usr/share/metricbeat/bin |
| <b>Config</b> | The location for configuration files.            | /etc/metricbeat           |
| <b>Data</b>   | The location for persistent data files.          | /var/lib/metricbeat       |
| <b>Logs</b>   | The location for the logs created by Metricbeat. | /var/log/metricbeat       |

8. Run the following commands to check if the metric beat services are running fine
  - To check the status of service  
systemctl status metricbeat
  - To stop the service
9. sudo systemctl stop metricbeat
10. To start the service
11. sudo systemctl start metricbeat
12. Please make sure the below files is owned by root user only
13. /etc/metricbeat/metricbeat.yml
14. Provide the below SUDO permission to “metricbeat” user
15. sudo vi metricbeat.yml

## Configuration of Metric beat

[Currently keep the values as default and do not change anything there]

**Important Note : Any configuration changes into the below mentioned file will be done using root. Only ROOT user should be**

**allowed to write to beat configuration files. NO other user should have written access to these configuration files.**

To configure Metricbeat, edit the configuration file placed @ /etc/metricbeat/metricbeat.yml file. The default configuration file is called metricbeat.yml.

To ship the information / metric data from application servers to elastic servers below configuration needs to be changed into metricbeat.yml file

```
#----- Elasticsearch output -----
output.elasticsearch:
  # Array of hosts to connect to.
  hosts: ["10.64.50.42:9200"]

  # Optional protocol and basic auth credentials.
  #protocol: "https"
  #username: "elastic"
  #password: "changeme"
[This needs to be validated, if the application server would be able to directly connect to the elastic server , then user / password is not required; else user / password would be required to access the elastic]
```

The Application servers should be connected with elastic server (10.64.50.42) using 9200 port

| Host Name | Server IP   | Port | Node Type | User          |
|-----------|-------------|------|-----------|---------------|
| mprl1129  | 10.64.50.42 | 9200 | Elastic   | elasticsearch |

## File beat

### Installation Steps

1. Download the filebeat-7.2.0-x86\_64.rpm file from below link

[https://teams.microsoft.com/\\_#/files/General?threadId=19%3Ac24766ec82e347628ff39dbdb733d6e9%40thread.skype&ctx=channel&context=ELK%2520Installer%2520Files](https://teams.microsoft.com/_#/files/General?threadId=19%3Ac24766ec82e347628ff39dbdb733d6e9%40thread.skype&ctx=channel&context=ELK%2520Installer%2520Files)

[It can also be downloaded from the elastic website URL  
<https://www.elastic.co/downloads/beats/filebeat> ]

2. Create user “filebeat” user on the server where filebeat needs to be installed.
3. Connect to the server using “filebeat” user where filebeat needs to be installed.
4. Provide the below listed sudo permissions to “filebeat” user
  - /usr/bin/rpm -ivh filebeat\*,
  - /usr/bin/rpm -e filebeat\*,
  - /usr/bin/systemctl status filebeat\*,
  - /usr/bin/systemctl status -l filebeat\*,
  - /usr/bin/systemctl stop filebeat\*,
  - /usr/bin/systemctl start filebeat\*,
  - /usr/bin/journalctl -u filebeat

5. Copy the .rpm file into the application server (\home folder)
6. Go to “home” folder and then run the below command
7. sudo rpm -vih filebeat\*
8. Check the below paths are created

| Type          | Description                                      | Location                |
|---------------|--------------------------------------------------|-------------------------|
| <b>Home</b>   | Home of the filebeat installation.               | /usr/share/filebeat     |
| <b>Bin</b>    | The location for the binary files.               | /usr/share/filebeat/bin |
| <b>Config</b> | The location for configuration files.            | /etc/filebeat           |
| <b>Data</b>   | The location for persistent data files.          | /var/lib/filebeat       |
| <b>Logs</b>   | The location for the logs created by Metricbeat. | /var/log/filebeat       |

9. Run the following commands to check if the metric beat services are running fine
  - To check the status of service  
`systemctl status filebeat`
  - To stop the service  
`sudo systemctl stop filebeat`
  - To start the service  
`sudo systemctl start filebeat`
10. Please make sure the below files is not owned by “filebeat” user  
`/etc/filebeat/filebeat.yml`
11. Provide the below SUDO permission to “filebeat” user  
`sudo vi filebeat.yml`

## Configuration of File beat

[Currently keep the values as default and do not change anything there]

**Important Note : Any configuration changes into the below mentioned file will be done using root. Only ROOT user should be allowed to write to beat configuration files. NO other user should have written access to these configuration files.**

To configure Metricbeat, edit the configuration file placed @ /etc/filebeat/filebeat.yml file. The default configuration file is called filebeat.yml.

To ship the log / metric data from application servers to logstash below configuration needs to be changed into filebeat.yml file

```
#----- Logstash output -----
-----
output.logstash:
  # The Logstash hosts
  hosts: ["10.64.50.60:5044"]

  # Optional SSL. By default is off.
  # List of root certificates for HTTPS server verifications
  #ssl.certificateAuthorities: ["/etc/filebeat/logstash-forwarder.crt"]

  # Certificate for SSL client authentication
  #ssl.certificate: "/etc/pki/client/cert.pem"

  # Client Certificate Key
  #ssl.key: "/etc/pki/client/cert.key"
```

The Host entry will be changed based on the load distribution on logstash servers.

**The Application servers should be connected with logstash Servers**

| Host Name | Server IP   | Port | Node Type | User     |
|-----------|-------------|------|-----------|----------|
| mprl127   | 10.64.50.60 | 5044 | Logstash  | logstash |
| mprl128   | 10.64.50.61 | 5044 | Logstash  | logstash |

## Make Build configuration:

<https://thoughtbot.com/blog/the-magic-behind-configure-make-install>

make clean;make clobber;make all;make clean

Nexus:

### Windows Commands:

<https://docs.microsoft.com/en-us/windows-server/administration/windows-commands/forfiles>

<https://www.windows-commandline.com/get-file-modified-date-time/>

dir /T:W /A:-D

forfiles /M \*.pdf /C "cmd /c echo @path @fdate @ftime"