



# A Hitchhiker's Guide to Oracle10g RAC

How to setup RAC on a single node or laptop

# Why Hitchhike?

- Save on transportation
- Feeling of adventure
- For challenge of travel w/limited resources
- For new & interesting experiences
- For sheer & simple love of it

# What's a Cluster?

- Reasons to cluster (from Wikipedia):
  - high availability
  - load balancing
  - high performance
  - grid computing

# What's a Cluster

- Types of clusters
  - IBM shared nothing (hashed data access)
    - convoy effect
    - data integrity moved to application
    - loss of any server means entire db down
  - Microsoft federated
    - each db needs own integrity constraints
    - need db triggers to check integrity violations
    - complex to maintain, many failure points

# Oracle's Model

- shared disk/database
- each node has own instance, SGA, undo
- applications scale easier
- no data partitioning req'd (as with OPS)
- no single point of failure
  - raided disks
  - redundant switches

# Hitchhiker's Model

- travel light, few worries, cares
- easy to setup, maintain
- low cost, low barrier to entry
- single server - open road, informal
- single disk, eth0 – no HA, no performance
- many nodes share same CPU, memory
- learn concepts, commands, architecture

# Steps A to Z

- Setup IP addresses
- Setup ssh + rsh w/autologin
- Setup Raw devices
- Install Clusterware + 10g
- Setup Listener + ASM
- Create Instance, Start & Register
- Create 2nd Instance, Undo2 + Register

# Virtual IP Addresses

- Oracle wants a few interfaces available
- edit /etc/hosts file
- hostname: route66 (192.168.0.19)
- private name: route66-priv (192.168.0.75)
- virtual IP: route66-vip (192.168.0.76)
- comment out route66-vip for now

# Virtual Interfaces

- eth0:0, eth0:1, eth0:2 etc
- same physical ethernet device
- OS created device files
  - $ /sbin/ifconfig eth0:1 192.168.0.75 netmask 255.255.255.0 broadcast 192.168.0.255
  - $ /sbin/ifconfig eth0:2 192.168.0.76 netmask 255.255.255.0 broadcast 192.168.0.255

# RSH Install

- download latest rsh client + server
  - Use http://rpmfind.net
  - Use up2date, yumm, or related tools
  - $ rpm –Uvh rsh-server-0.17-34.1.i386.rpm
  - $ rpm –Uvh rsh-0.17-34.1.i386.rpm

# RSH Autologin

- Oracle’s installer wants to autologin
- between “servers”, self, virtual interfaces
- edit /home/oracle/.rhosts
  - route66
  - route66-vip
  - route66-priv
- $ chmod 644 /home/oracle/.rhosts
- Test, test, test!!

# SSH Setup

- again, Oracle wants to autologin
- between “servers”, self, virtual interfaces
- ssh/sshd probably already installed
- be sure sshd is running:
  - /etc/rc.d/init.d/sshd start
- generate private and public keys
  - $ ssh-keygen -t dsa  (as oracle OS user)

# SSH Setup

- Copy public key data into authorized_keys
  - $ cp id_dsa.pub authorized_keys
- Try ssh route66-vip, verify no pwd req'd
- Test, test, test!!!

# Raw Devices

- Can create virtual disk system
- Similar to virtual ethernet devices
- Cluster Ready Services (CRS) will use
- Auto Storage Manager (ASM) will use
- Can use new disk, or existing main HD

# Raw Devices

- Create 3x 2G “disks”
  - $ mkdir /asmdisks
  - $ dd if=/dev/zero of=/asmdisks/disk1 bs=1024k count=2000
  - $ dd if=/dev/zero of=/asmdisks/disk2 bs=1024k count=2000
  - $ dd if=/dev/zero of=/asmdisks/disk3 bs=1024k count=2000

# Raw Devices

- Create a 100M OCR disk:
  - $ dd if=/dev/zero of=/asmdisks/disk4 bs=1024k count=100
- Create small 20M voting disk:
  - $ dd if=/dev/zero of=/asmdisks/disk5 bs=1024k count=20

# Raw Devices

- Convince Linux our files are raw disks:
  - $ /sbin/losetup /dev/loop1 /asmdisks/disk1
  - $ raw /dev/raw/raw1 /dev/loop1
  - $ chown oracle.dba /dev/raw/raw1
- Do the same for disk2 through disk5

# Install Clusterware

- We are tricking installer into thinking we have clustered hardware
- Tell Oracle about voting + OCR devices
  - $ ln -sf /dev/raw/raw4 /home/oracle/product/disk_ocr
  - $ ln -sf /dev/raw/raw5 /home/oracle/product/disk_vot
- Setup environment & fire up runInstaller

# Install Clusterware

- ssh/rsh autologin setup is key
- raw devices must be setup
- A few steps very tricky…
- ./runInstaller  (-ignoreSysPrereqs)
- Possibly use CentOS.org RHEL
- Ignore phys memory prerequisite warning
- Specify Cluster Config
  - The following node names invalid route66-vip
  - comment in /etc/hosts file

# Install Clusterware

- Specify Cluster Config
  - The node route66-priv not reachable
    - sshd running?
    - /etc/rc.d/init.d/iptables stop
    - eth0:1, eth0:2 created?
  - Error thrown when IP address of host cannot be determined
    - Comment out route66-vip in /etc/hosts file
    - Click retry

# Install Clusterware

- Specify Network Interface Usage
  - Edit -> specify PUBLIC
- Specify Oracle Cluster Registry Location
  - external redundancy option
  - /home/oracle/product/disk_ocr
- Specify Voting Disk Location
  - external redundancy option
  - /home/oracle/product/disk_vot

# Install Clusterware

- At end, need to run root.sh
- got error that vipca failed, run manually
  - $CRS_HOME/bin/vipca
  - specify route66, route66-vip, 192.168.0.76
- all configuration assistants must succeed!

# Reinstall Clusterware

- rm –rf /etc/oracle
- rm –rf /home/oracle/oraInventory
- rm –rf $CRS_HOME
- /bin/dd if=/dev/zero of=/asmdisks/disk4 bs=1024k count=100
- /bin/dd if=/dev/zero of=/asmdisks/disk5 bs=1024k count=20
- rm /etc/rc.d/init.d/init.crs
- rm /etc/rc.d/init.d/init.crsd
- rm /etc/rc.d/init.d/init.cssd
- rm /etc/rc.d/init.d/init.evmd
- rm /etc/rc.d/rc3.d/S96init.crs
- rm /etc/rc.d/rc5.d/S96init.crs
- reboot server – kills any CRS processes

# Test Clusterware

- Verify CRS is running
  - $ ps auxw | grep ora

```
root   /bin/su -l oracle -c sh -c 'ulimit -c unlimited; cd /home/oracle/product/10.2.0s/log/b\
  ebel/evmd; exec /home/oracle/product/10.2.0s/bin/evmd '
root   /home/oracle/product/10.2.0s/bin/crsd.bin reboot
oracle /home/oracle/product/10.2.0s/bin/evmd.bin
root   /bin/su -l oracle -c /bin/sh -c 'ulimit -c unlimited; cd /home/oracle/product/10.2.0s/\
  log/bebel/cssd;  /home/oracle/product/10.2.0s/bin/ocssd  || exit $?'
oracle /bin/sh -c ulimit -c unlimited; cd /home/oracle/product/10.2.0s/log/bebel/cssd;  /home\
  /oracle/product/10.2.0s/bin/ocssd  || exit $?
oracle /home/oracle/product/10.2.0s/bin/ocssd.bin
oracle /home/oracle/product/10.2.0s/bin/evmlogger.bin -o /home/oracle/product/10.2.0s/evm/log\
  /evmlogger.info -l /home/oracle/product/10.2.0s/evm/log/evmlogger.log
oracle /home/oracle/product/10.2.0s/opmn/bin/ons -d
oracle /home/oracle/product/10.2.0s/opmn/bin/ons -d
```

  - $ olsnodes –n

```
route66 1
```

# Install Oracle 10g

- fairly straightforward as Oracle installs go
  - ./runInstaller  (-ignoreSysPrereqs)
- specify Enterprise Edition
- Oracle will recognize CRS is installed

# Create Listener

- edit $TNS_ADMIN/listener.ora
  - Usual setup
    - ORACLE_HOME=/home/oracle/product/10.2.0
  - Include both hostnames in address list:
    - HOST=route66
    - HOST=route66-vip
- use new 10g srvctl to start:
  - srvctl start nodeapps –n route66

# Create ASM Instance

- strange name +ASM1
- create init+ASM1.ora, specifying user, bkg and core_dump_dest,
  - large_pool_size=15m
  - instance_type=asm
  - asm_diskstring='/dev/raw/raw1', '/dev/raw/raw2','/dev/raw/raw3', '/dev/raw/raw4','/dev/raw/raw5'

# Start ASM

- ORACLE_SID=+ASM1
- SQL> startup nomount
  - create diskgroup DBDATA external redundancy disk '/dev/raw/raw1','/dev/raw/raw2‘;
  - create diskgroup DBRECO external redundancy disk ‘/dev/raw/raw3’;
- edit init+ASM1.ora and add:
  - asm_diskgroups=‘DBDATA',‘DBRECO'
- use sqlplus to startup ASM instance
  - SQL> startup force

# Register +ASM1

- register with srvctl
  - $ srvctl add asm -n route66 -i +ASM1 -o /home/oracle/product/10.2.0.1
- shutdown with sqlplus:
  - SQL> shutdown immediate
- now startup with srvctl:
  - $ srvctl start asm -n route66
- use ps to check for +ASM1 instance

# Create BEATNIK Instance

- Edit your usual init.ora params such as db_block_size, db_multiblock_read_count, archiving params etc
  - db_name=kerouac
- instances are BEATNIK & HIPPY
- Set instance specific params like this:
  - BEATNIK.background_dump_dest=/path/a

# Create BEATNIK Instance

- specify user, bkg, core_dump_dest,
  - BEATNIK.instance_number=1
  - BEATNIK.instance_name=BEATNIK
  - BEATNIK.thread=1
  - BEATNIK.undo_tablespace=UNDOTBS1
- Create admin/BEATNIK/* & admin/HIPPY/* dirs
- Edit crKEROUAC.sql as follows:
  - CREATE DATABASE “KEROUAC"<br>DATAFILE SIZE 250M<br>EXTENT MANAGEMENT LOCAL<br>SYSAUX DATAFILE SIZE 125M<br>DEFAULT TEMPORARY TABLESPACE TEMP TEMPFILE SIZE 20M<br>UNDO TABLESPACE "UNDOTBS1" DATAFILE SIZE 200M<br>LOGFILE GROUP 1 SIZE 10240K,<br>GROUP 2 SIZE 10240K,<br>GROUP 3 SIZE 10240K ;
- To simplify - no character set, max params, user specified here

# Create BEATNIK Instance

- Fire up sqlplus and issue:
  - startup nomount pfile=/path/a/initBEATNIK.ora
  - @crKEROUAC.sql
- Select ctl files from v$parameter & add to initBEATNIK.ora

# Make BEATNIK Clustered

- Add these two params to your initBEATNIK.ora:
  - *.cluster_database=true
    *.cluster_database_instances=5
- use sqlplus to shutdown instance:
  - SQL> shutdown immediate
- Use sqlplus to startup again:
  - SQL> startup force

# Register BEATNIK

- Let OCR know what we've done:
  - $ srvctl add database –d KEROUAC -o /home/oracle/product/10.2.0.1/
  - $ srvctl add instance -d KEROUAC -i BEATNIK -n route66
- Use sqlplus to shutdown immediate
- Startup again with srvctl:
  - $ srvctl start instance -d KEROUAC -i BEATNIK

# Create HIPPY Instance

- First we need second undo tablespace
  - SQL> create undo tablespace undo2 datafile '+DBDATA' size 100M;
- Create  init.ora file:
  - $ cp initBEATNIK.ora initHIPPY.ora

# Startup & Register HIPPY

- $ export ORACLE_SID=HIPPY
- SQL> startup
- $ srvctl add instance -i HIPPY -d KEROUAC -n route66

# Create Data Dictionary

- SQL>@?/rdbms/admin/catalog.sql
  SQL>@?/rdbms/admin/catblock.sql
  SQL>@?/rdbms/admin/catproc.sql
  SQL>@?/rdbms/admin/catoctk.sql
  SQL>@?/rdbms/admin/owminst.plb
  SQL>@?/rdbms/admin/catclust.sql

# Some Things To Try

- SQL> select instance_name from gv$instance;
- Add a third instance
- Kill -9 the processes of one of the instances
- Cross-instance transaction example

# Some Things To Understand

- Automatic Storage Management
- Cluster Ready Services
- Global Cache Services
- Global Enqueue Services
- Clusterware procs – crsd, evmd, ocssd,oprocd
- RAC processes lms, lmd, lmon, lck0
- GV$ data dictionary

# And Don’t Forget

- add eth0:1, eth0:2 to rc.local
- add loop device setup to rc.local
- add raw device setup to rc.local
- add +ASM1,BEATNIK,HIPPY to /etc/oratab
- add instance startup to /etc/rc.d/init.d/dbora
  - start)
    - srvctl start instance –i BEATNIK –d KEROUAC
  - stop)
    - srvctl start instance –i HIPPY –d KEROUAC

# Futher Reading & Thanks

- Clusterware & RAC Install & Config Guide for Linux
- Clusterware & RAC Admin Deploy Guide
- http://otn.oracle.com
- Please visit http://www.iheavy.com or email me at <shull@iheavy.com>
- Thanks to Amit Poddar & dizwell.com
- Apress, Oracle Press books