Oracle 11g Release 2

-

Install Guide

Auhtor: Ronny Egner
Email: ronnyegner@gmx.de

Date: 25. September 2009

Status: early Draft

The information outlined in this document are from the authors experience and research.

At this point of time this document is in an early development state.

You can use the information freely without WARRANTY!

The author

- Name: Ronny Egner
- Status: Freelancer
- Contact data:
  - email: ronnyegner@gmx.de
  - mobile phone: +49 1708139903

- > 7 years experience with Oracle Databases and UNIX operating systems
  - experience with 8i to 11g Release 2
  - High Availability (RAC, Data Guard)
  - Backup and Cloning (RMAN, custom Scripts, Storage-based Cloning)
  - Database Upgrades

- EMC Networker expert
  - Planning and implementing company-wide backup concepts
  - Backing up all kinds of data (including databases, exchange server, sql server, …) to tape / virtual tape libraries / disk based storage, ...

Contents

- Overview
- Part 1 – Single Database installation and creation
- Part 2 – ASM (including ACFS and ADVM) installation, configuration and tests
- Part 3 – ADVM and ACFS
- Part 4 – RAC with ASM installation and configuration

Overview

Overview

- Oracle Database 11g Release 2 released on 1st September 2009
- Download from http://otn.oracle.com (requires registration but no support identifier)

- Currently available platforms:
  - Linux x86 (32-bit)
  - Linux x86_64 (64-bit)

Overview – New Features

- Announcement on OTN:

  www.oracle.com/global/de/community/dbadmin/tipps/11gr2/index.html

- New Features – a non-complete overview:
  - Automatic Block repair in data guard scenarios (i.e. Replace defective block on standby with valid block from primary database or vice versa)
  - Automatic determance of parallelism degree based on object size, query complexity and hardware resouces
  - RAC installation complexity dramatically reduced → we will evaluate this :-)
  - Storage of Oracle Cluster Registry and Voting Disk in ASM
  - „Oracle Restart“: automatically (re)starts database instance, ASM, listener and so on
  - Utility for complete uninstallation of Oracle RAC
  - Reduced („zero downtime“) for patching RAC clusters

Overview – New Features

- New Features – a non-complete overview (con´t):
  - Instance Caging: Limit number of CPUs used
  - Scheduler improvements (emailing, file watch, run procedures on remote db)
  - ACFS (ASM Cluster File System): Cluster file system based on ASM for non-database datafiles, i.e. (Oracle) Binary installations, trace files, alert logs and so on
  - ASM Dynamic Volume Manager: create volumes out of disk groups and use it to create a file system (ext3, reiserfs, etc pp) on top of it
  - ASM FS Snapshots
  - Intelligent data placement: frequently accessed blocks are placed on the edge of the disk where I/O performance is higher by ASM
  - ASM File Access Control
  - New compression algorithm (LZO) offers fast compression and de-compression
  - → most „cool“ ASM features require 11g release 2 database and asm compatible level (= everything 11.2)

Overview - Components

- Oracle Database
  - traditional database (rdbms)
  - ASM module
  - Listener et al
- Oracle Infrastructure
  - foundation for Oracle RAC
  - Includes
    - Clusterware
    - ASM Module
    - Oracle Restart

Overview - Conclusion

- ?? Is 11g Release 2 ready for production yet ??
  - From the Authors point of view: NO!
  - Release of 11g Release 2 was driven by marketing and promise „release in September 2009“
  - Latest beta of 11g R2 (released two weeks before „productive release“ still contained critical bugs
  - 11.2.0.1.0 good for testing and educational purpose
  - For productive environments:

    Wait at least until 11.2.0.2.0!
    (i.e. The first patchset)

Part 1 –
Single Database installation

Installation Overview

- Local System configuration
  - Installation took place on laptop with CentOS 5 x86_64 (Kernel 2.6.18-128.2.1.el5)
  - 4 GB Memory
  - 500 GB local disk

Installation Overview

- Steps required to install Oracle 11g Release 2

1. Configure Storage
2. Check and fulfill pre-requirements
3. Binary installation of database
4. Listener configuration
5. Creation of database

Installation Prerequisites

- Storage Requirements
  - As always - recommendation: SAME (stripe and mirror everything)
  - Valid storage options for single database instance:
    - file system (ext3, reiser, xfs, etc al)
    - ASM
    - ACFS → only for non-database files (i.e. Binary files, trace files and so on)
    - NFS
    - ISCSI
    - RAW Disks

Installation Prerequisites

- SWAP
  - Between 1 and 2 GB RAM → 1.5 times the size of RAM
  - Between 2 and 16 GB RAM → equal to size of RAM
  - > 16 GB RAM → 16 GB SWAP
- Automatic Memory Management
  - Required /dev/shm with appropriate size (i.e. SGA of 16 GB required /dev/shm to be 16 GB+)
  - Huge Pages and autom. Memory Management are INCOMPATIBLE

Installation Prerequisites

- Supported Operating Systems
  - on 32-bit Linux
    - Asianux 2 Update 7 (Kernel 2.6.9 or later)
    - Asianux 3 (Kernel 2.6.18 or later)
    - Oracle Enterprise Linux 4 Update 7 (Kernel 2.6.9 or later)
    - Oracle Enterprise Linux 5 Update 2 (Kernel 2.6.18 or later)
    - Red Hat Enterprise Linux 4 Update 7 (Kernel 2.6.9 or later)
    - Red Hat Enterprise Linux 5 Update 2 (Kernel 2.6.18 or later)
    - SUSE Linux Enterprise Server 10 SP2 (Kernel 2.6.16.21 or later)
    - SUSE Linux Enterprise Server 11 (2.6.27.19 or later)

!! ACFS and ADVM are ONLY supported on RHEL 5 and OEL 5 !!

Installation Prerequisites

- Supported Operating Systems
  - on 64-bit Linux
    - Asianux 2 (Kernel 2.6.9 or later)
    - Asianux 3 (Kernel 2.6.18 or later)
    - Oracle Enterprise Linux 4 Update 7 (Kernel 2.6.9 or later)
    - Oracle Enterprise Linux 5 Update 2 (Kernel 2.6.18 or later)
    - Red Hat Enterprise Linux 4 Update 3 (Kernel 2.6.9 or later)
    - Red Hat Enterprise Linux 5 Update 2 (Kernel 2.6.18 or later)
    - SUSE Linux Enterprise Server 10 SP2 (Kernel 2.6.16.21 or later)
    - SUSE Linux Enterprise Server 11 (2.6.27.19 or later)

!! ACFS and ADVM are ONLY supported on RHEL 5 and OEL 5 !!

Installation Prerequisites

- Required Packages - refer to:

  http://download.oracle.com/docs/cd/E11882_01/install.112/e10840/pre_install.htm

- Multihoming (i.e. System with multiple network cards)
  - Recommended to set ORACLE_HOSTNAME
  - If not set – first entry from /etc/hosts is taken

Installation Prerequisites

- Users and Groups
  - For separation of rights (i.e. Manage ASM storage, manage database instance)
  - Available groups:
    - OSDBA (typical: „dba“; have SYSDBA privileges on database)
    - SYSOPER (typical: „sysoper“; optional, limited set of administrative priv)
    - OSDBA for ASM (typical: „asmdba“; full administrative access to ASM instance)
    - OSASM for ASM (typical: „asmadmin“; administrative access to ASM instance via SQL*Net)
    - OSOPER for ASM (typical: „asmoper“; optional, like SYSOPER group for limited access)
  - Minimal group needed: OSDBA group
    (in this document the osdba group is named „dba“)

Installation Prerequisites

- Shell Limits
  - In /etc/security/limits.conf

```
oracle          soft    nproc    16384
oracle          hard    nproc    16384
oracle          soft    nofile   65536
oracle          hard    nofile   65536
```

    (replace „oracle“ with user holding the installation)
  - In /etc/pam.d/login add if not exists

```
session    required     pam_limits.so
```

Installation Prerequisites

- Kernel Limits (MINIMUM values)
  - /etc/sysctl.conf

    kernel.sem=250 32000 100 128
    kernel.shmall=2097152
    kernel.shmmax=536870912
    kernel.shmmni=4096
    fs.file-max=6815744
    fs.aio-max-nr=1048576
    net.ipv4.ip_local_port_range=9000 65500
    net.core.rmem_default=262144
    net.core.rmem_max=4194304
    net.core.wmem_default=262144
    net.core.wmem_max=1048576

    - SuSE only -
    vm.hugetlb_shm_group=<gid of osdba group>

  - The values in /etc/sysctl.conf should be tuned (i.e. according to the number of instance, available memory, number of connections,...)

Installation Prerequisites

- Kernel Limits
  - The values in /etc/sysctl.conf should be tuned (i.e. according to the number of instance, available memory, number of connections,...)
  - see Part 2 for guides how to calculate the kernel parameters

Installation Prerequisites

- User Profile file (minimum file)
  - ~/.bash_profile (RHEL, OEL) or ~/.profile (SuSE)

```
export ORACLE_BASE=/u01/app/oracle
export
ORACLE_HOME=$ORACLE_BASE/product/11.2/ora11r2p
export ORACLE_SID=ORA11R2P
export PATH=$ORACLE_HOME/bin:
$ORACLE_HOME/OPatch:$PATH
 umask 022
```

  - ATTENTION: This profile file is for use with a dedicated user for each database binary installation. If you wish to install several binary installation under one single user make sure ORACLE_HOME and TNS_ADMIN is unset

Installation – notes before installation

- We will install an Oracle 11g Release 2 database (stand-alone without grid infrastructure, without asm), operating system user is named „ora11“, osdba group named „dba“ with home directory „/u01/app/oracle/product/11.2.0/ora11“

- Note: According to OFA the directory shall be named „...11.2.0/db_1“. For easy management we install a dedicated binary installation under a dedicated user for every database which runs on the system and name them accordingly.

  For instance binary database installation for database with SID „ORA11“ is held by operating system user named „ora11“ with home directory „...11.2.0/ora11“.
  The binary installation for the database with SID „ORA11T“ is held by the user named „ora11t“ with home directory „...11.2.0/ora11t“.

  You can – of course – use one single user on operating system level for having one or more binary installations. It´s up to you.

Installation – the actual installation

- Create User

  mkdir -p /u01/app/oracle/product/11.2.0/ora11
  useradd -g dba -d /u01/app/oracle/product/11.2.0/ora11 ora11
  passwd ora11

- Create profile file

  export ORACLE_BASE=/u01/app/oracle
  export ORACLE_HOME=$ORACLE_BASE/product/11.2.0/ora11
  export ORACLE_SID=ORA11
  export NLS_LANG=AMERICAN_AMERICA.WE8MSWIN1252
  export TMP=$ORACLE_HOME/tmp
  export TNS_ADMIN=$ORACLE_HOME/network/admin
  export TEMP=$TMP
  export PATH=$ORACLE_HOME/bin:$ORACLE_HOME/OPatch:$PATH

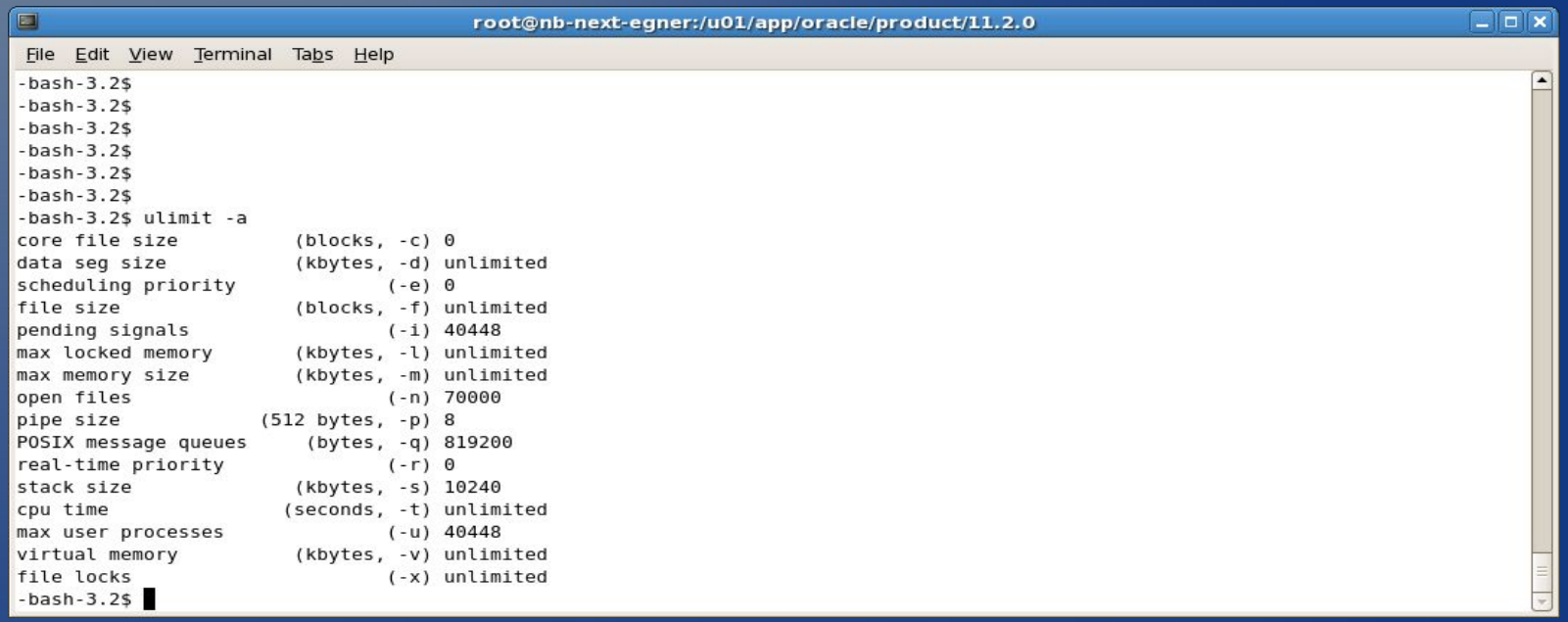
Installation – the actual installation

- Check, if X11 works

  export DISPLAY=localhost:0.0
  xterm

- Check

  ulimit -a (as user „ora11“)



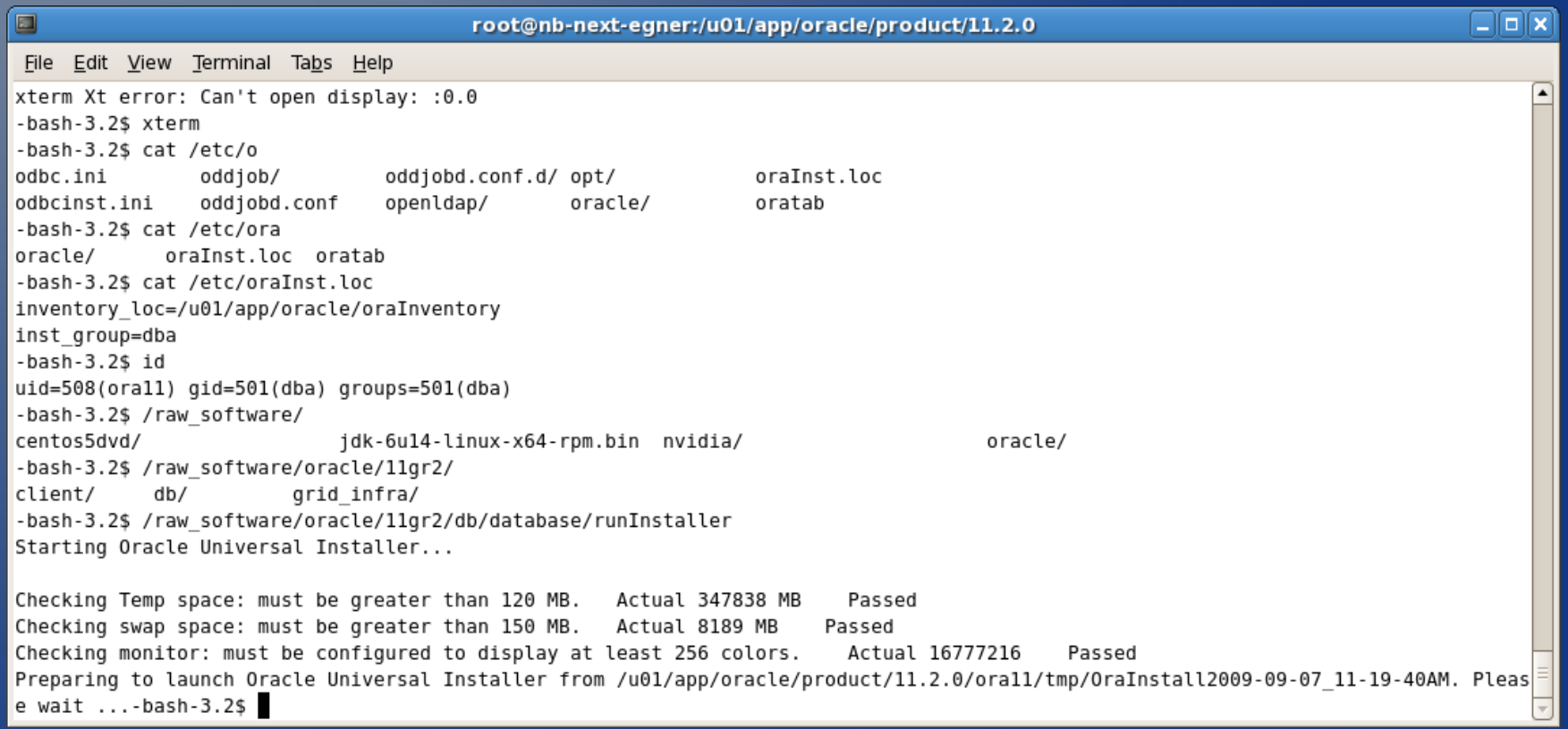
„open files“ and „max user processes“ should show values greater or equial the requisites

Installation – the actual installation

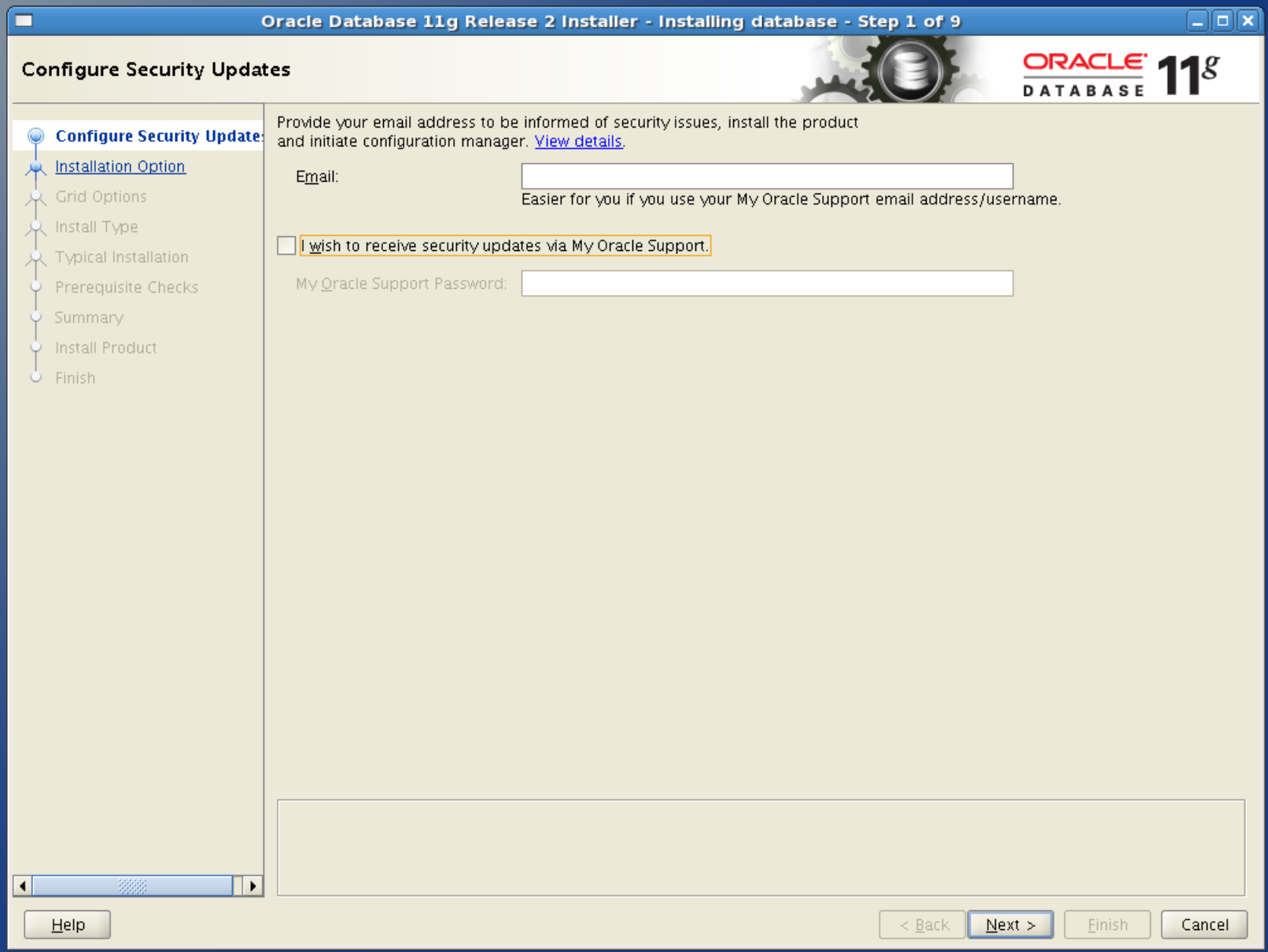
- Check, if X11 works

  export DISPLAY=localhost:0.0
  xterm

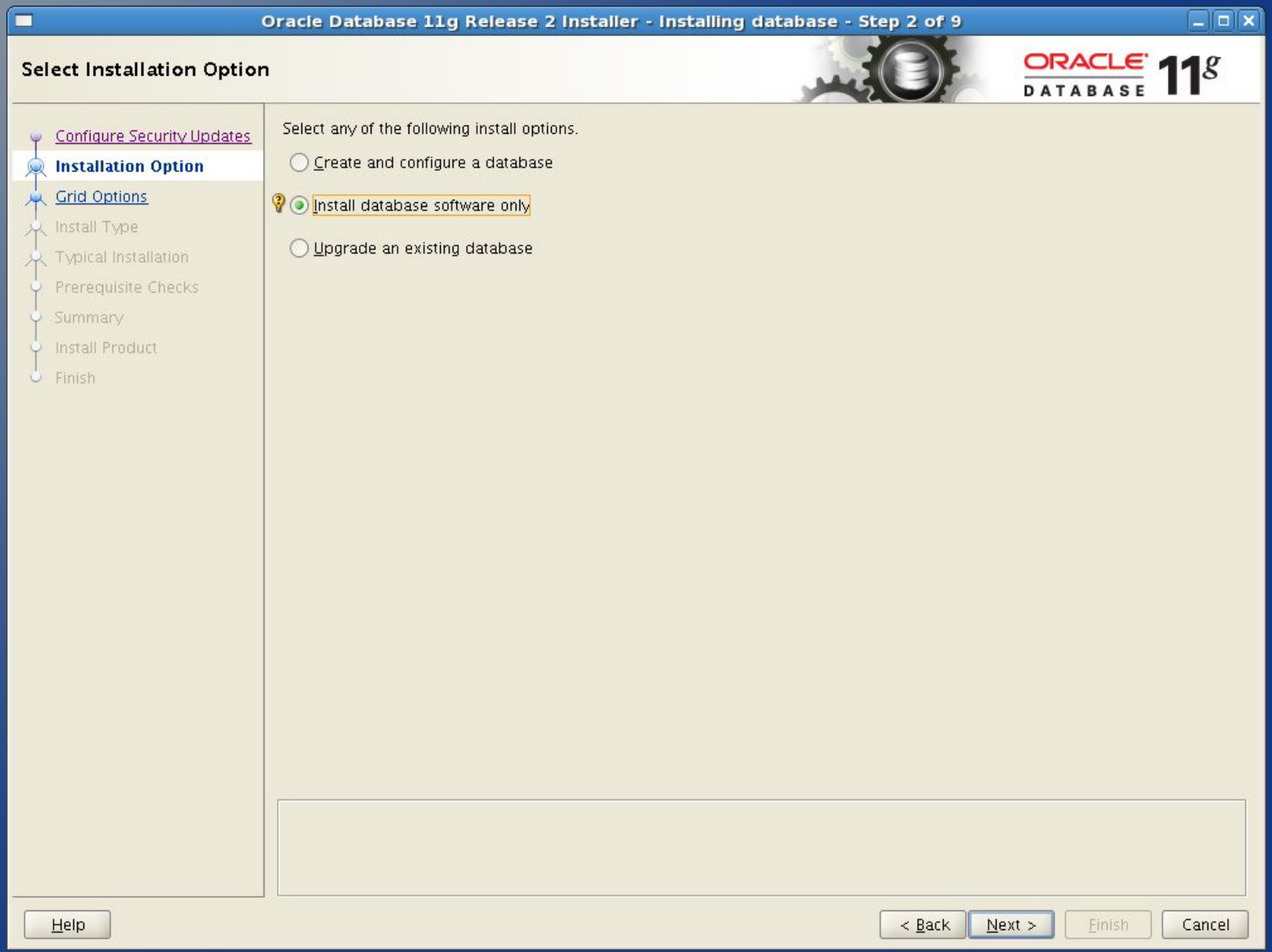
- Start Installation



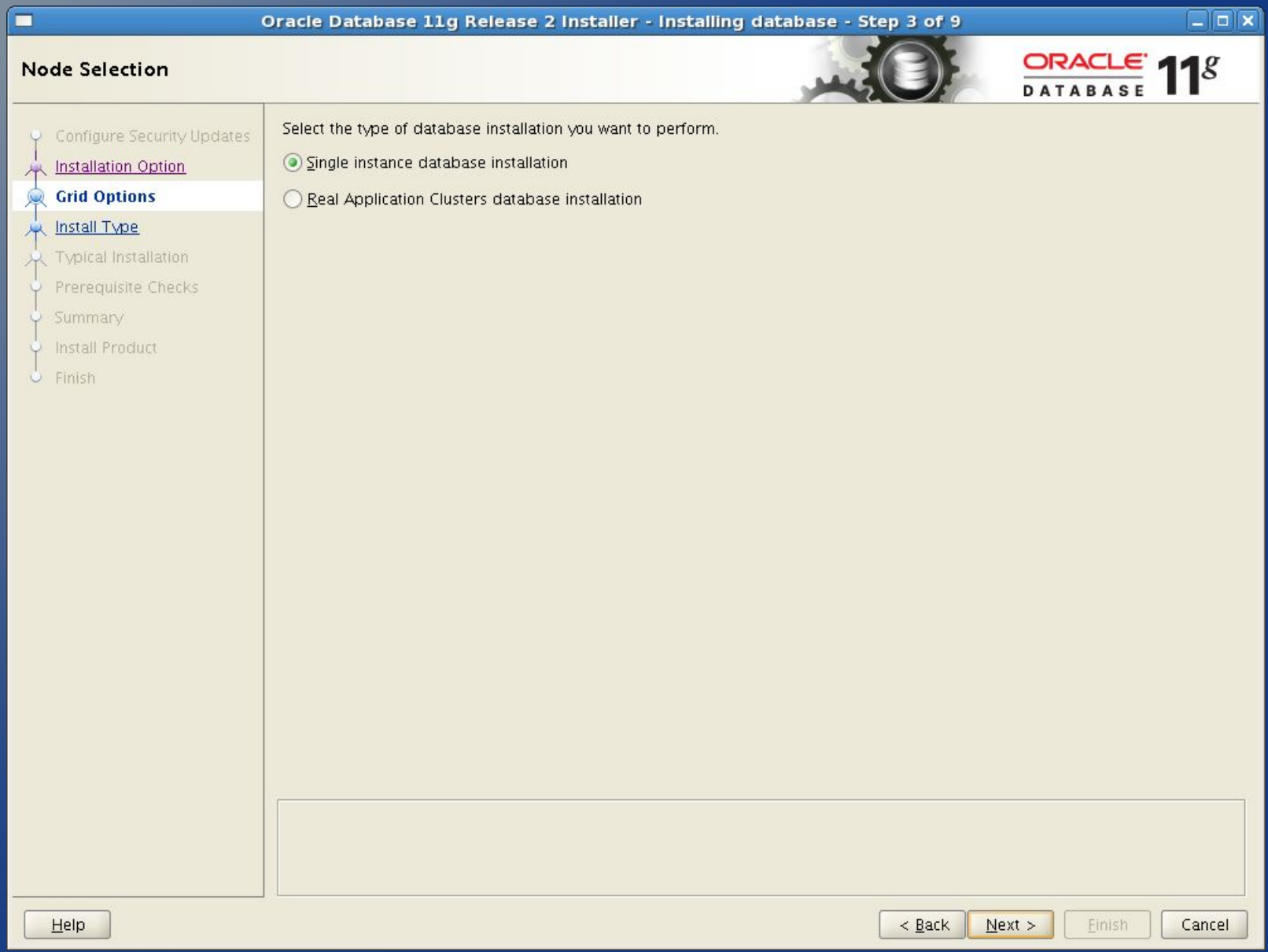
Installation – the actual installation



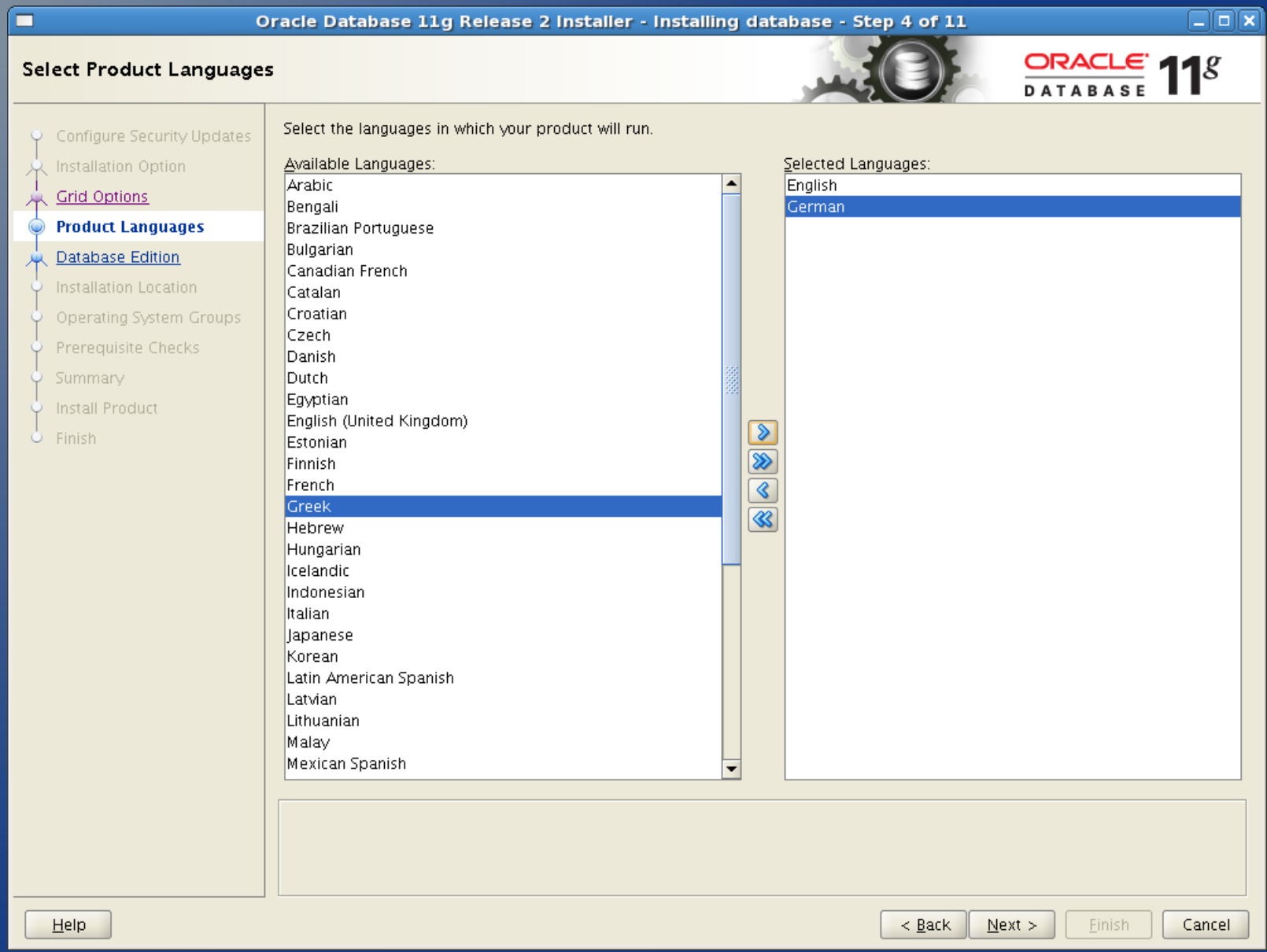
Installation – the actual installation



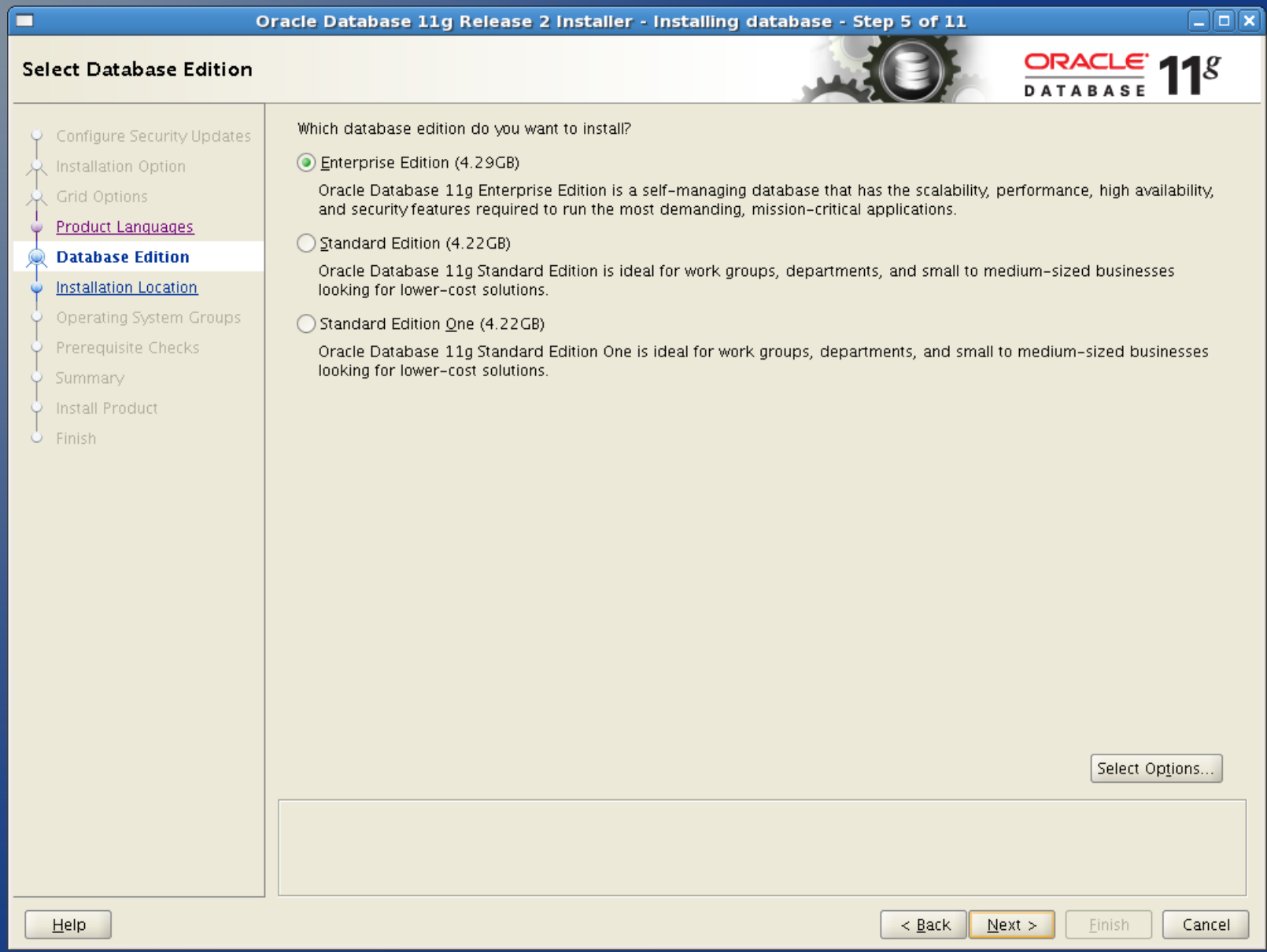
Installation – the actual installation



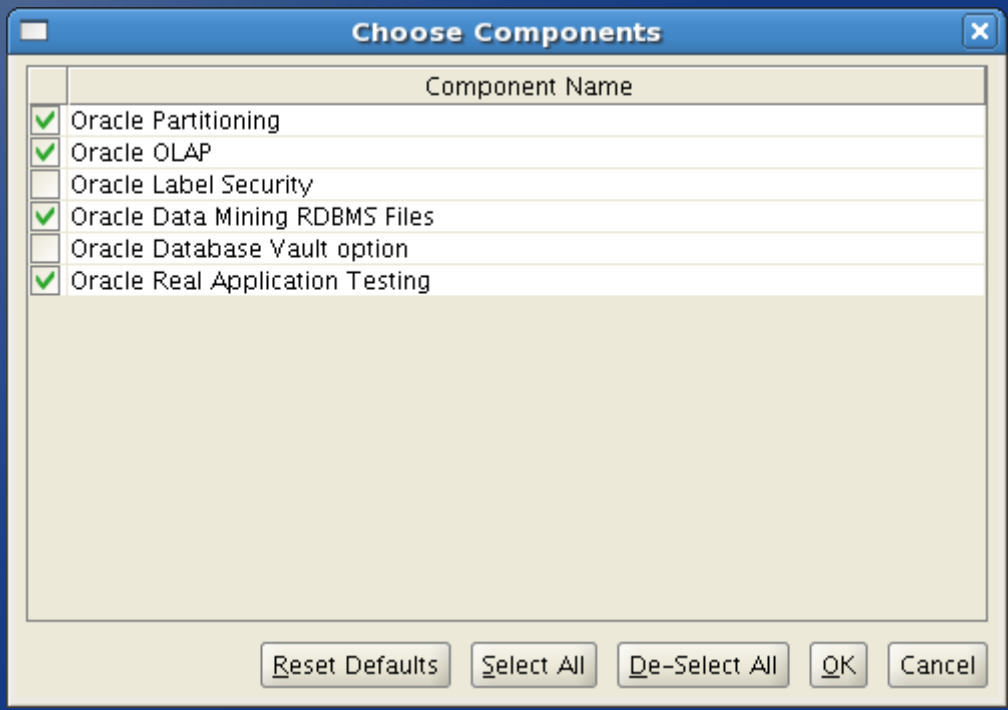
Installation – the actual installation



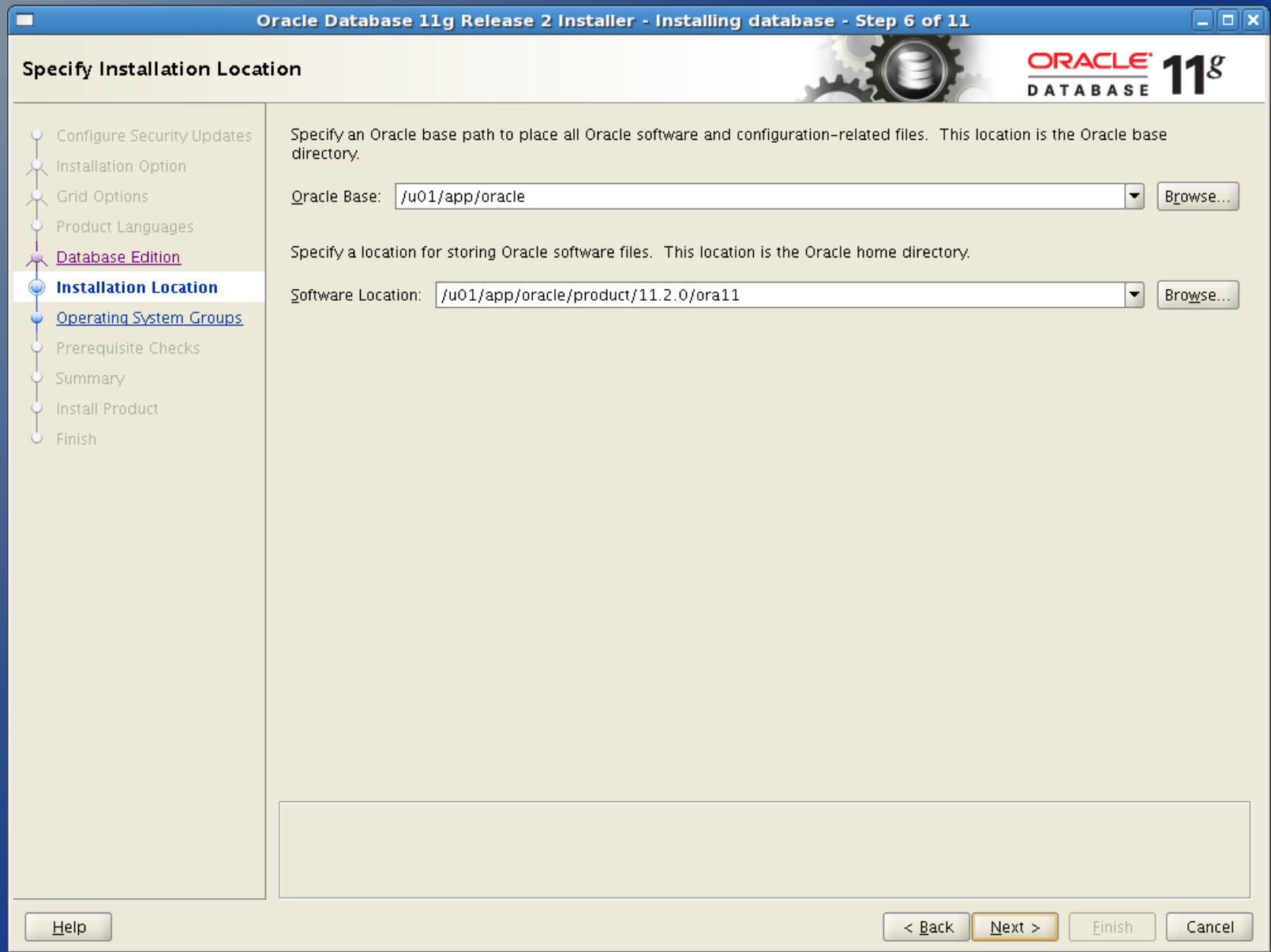
Installation – the actual installation



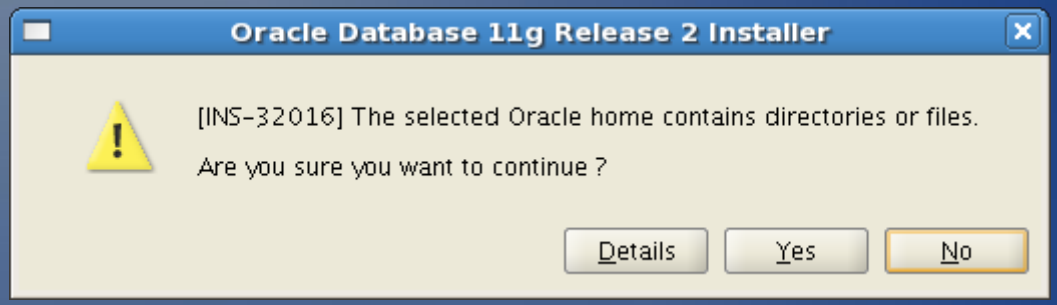
Installation – the actual installation



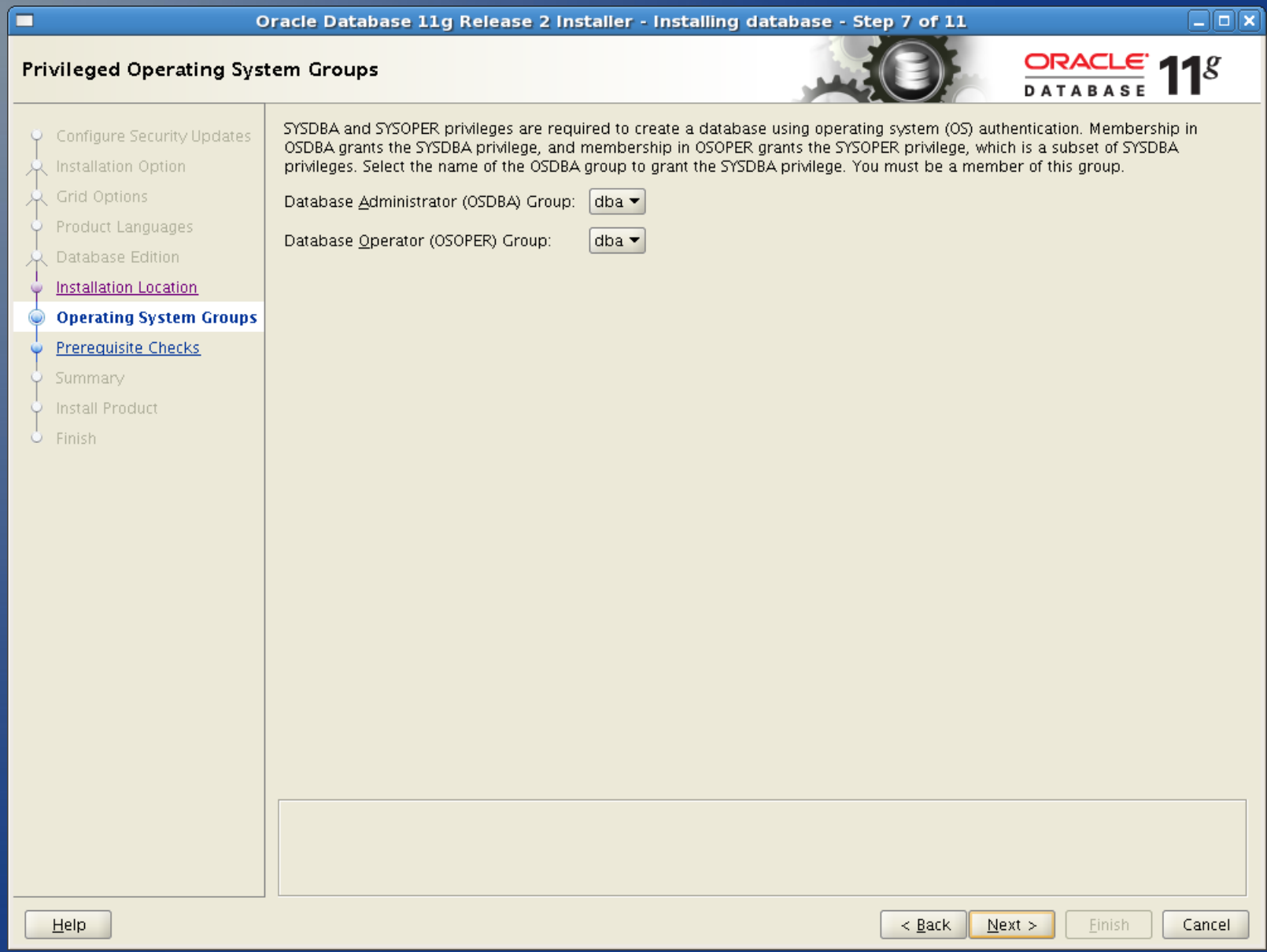
Installation – the actual installation



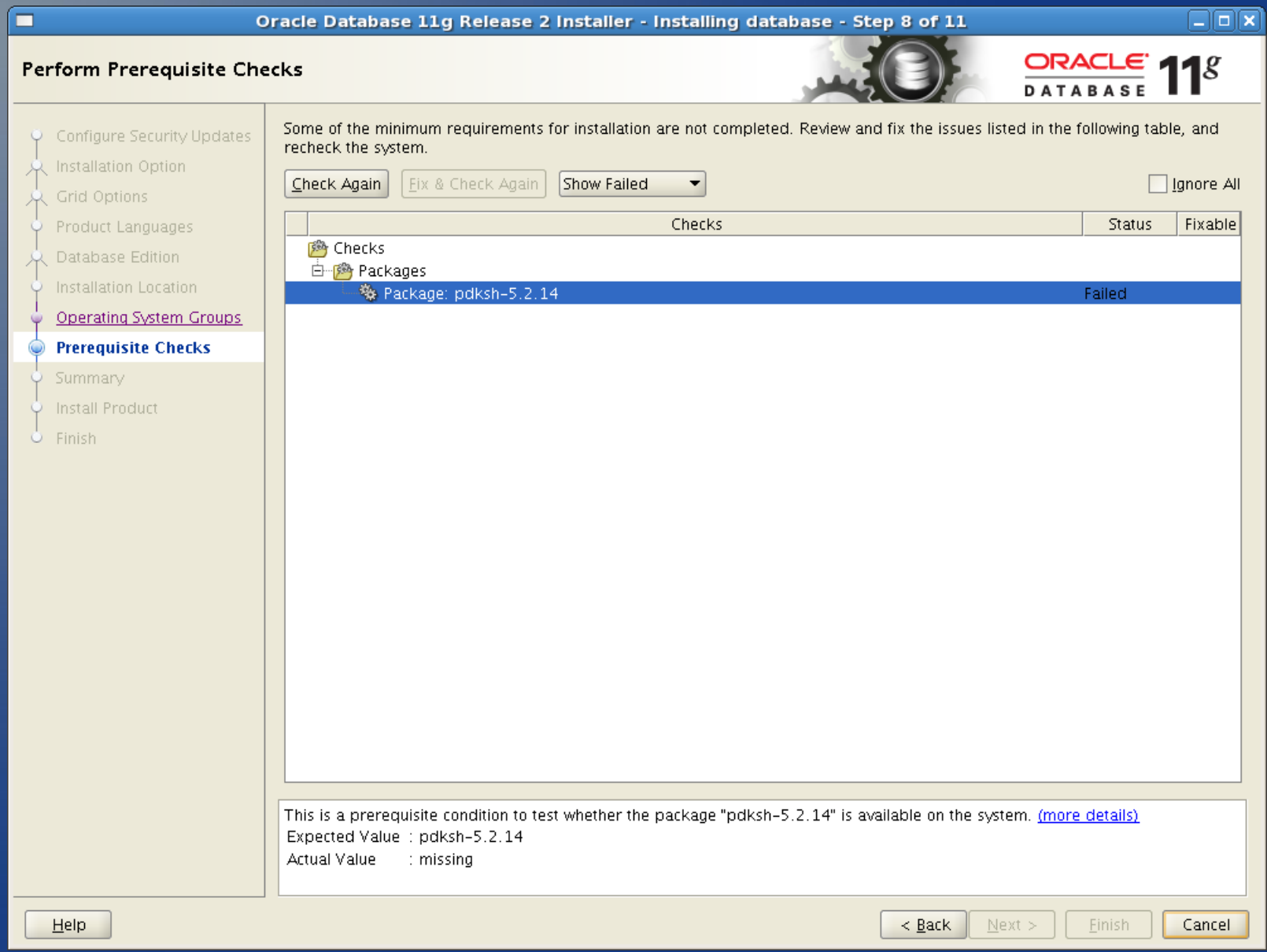
Installation – the actual installation



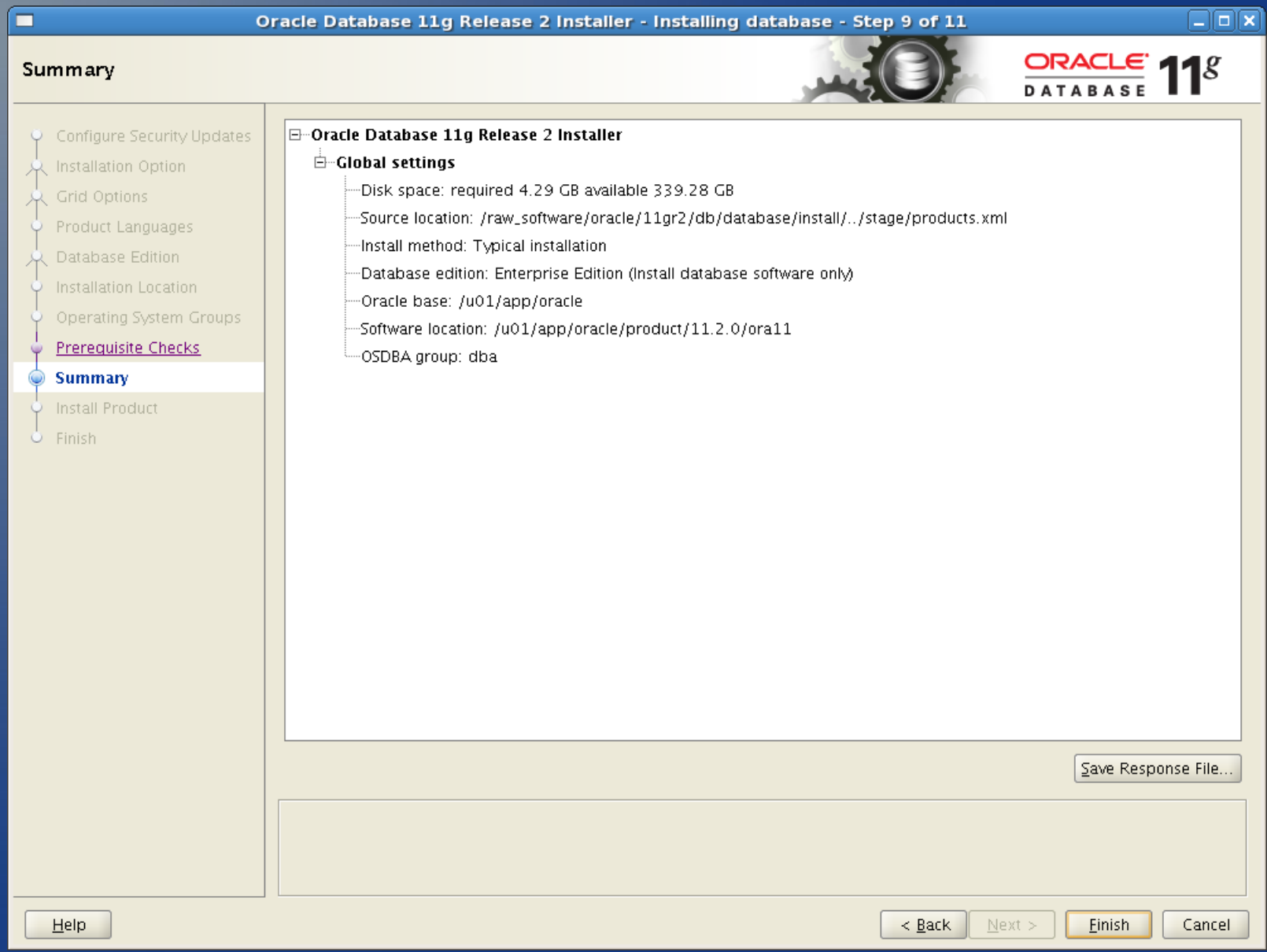
Installation – the actual installation



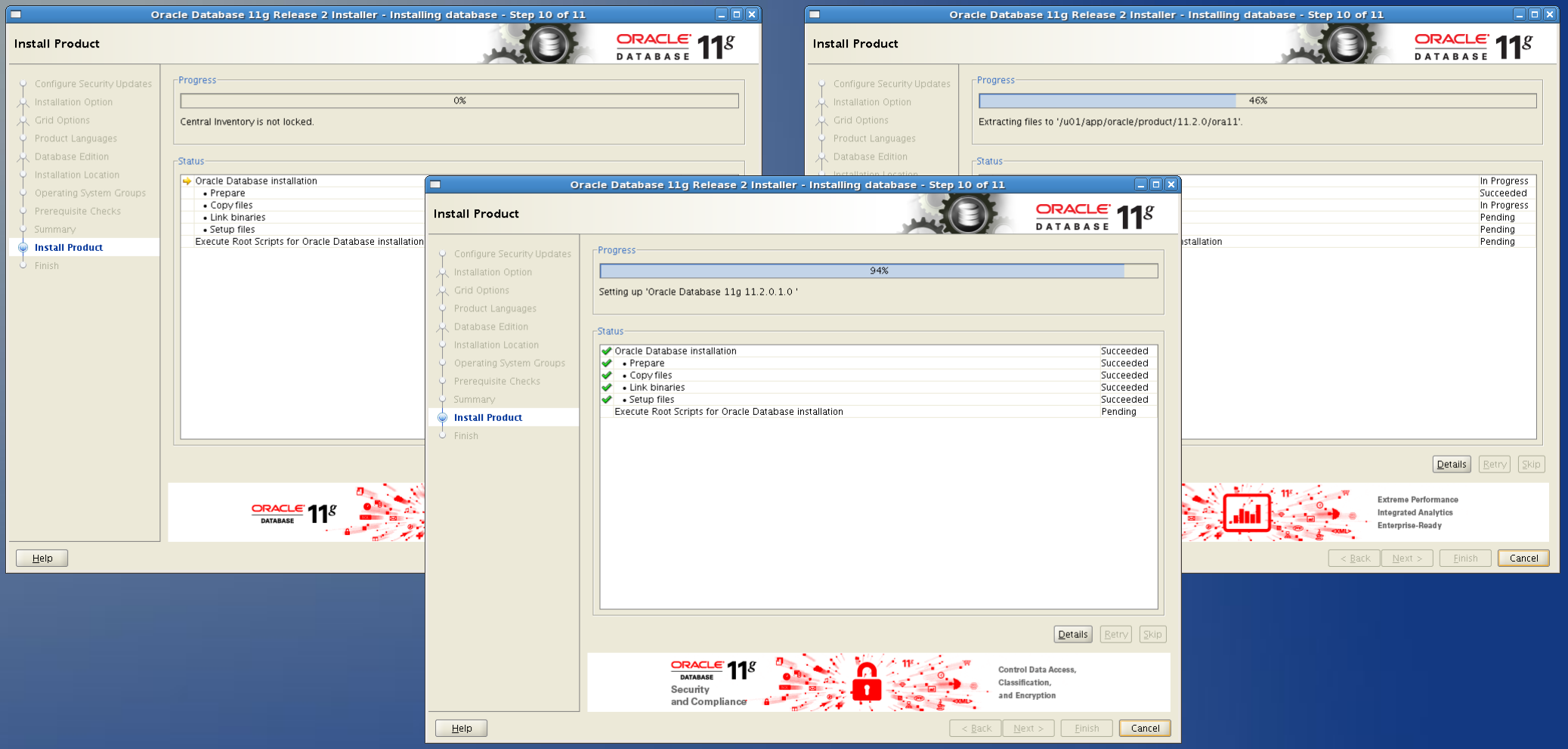
Installation – the actual installation



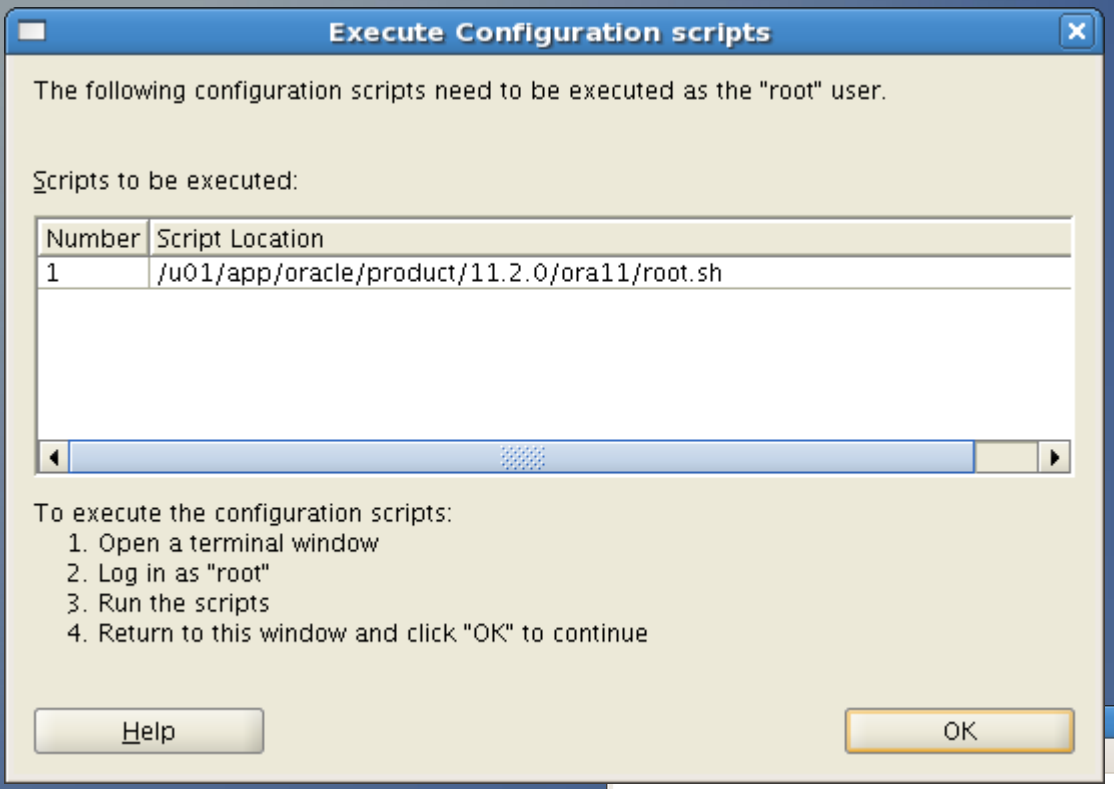
Installation – the actual installation

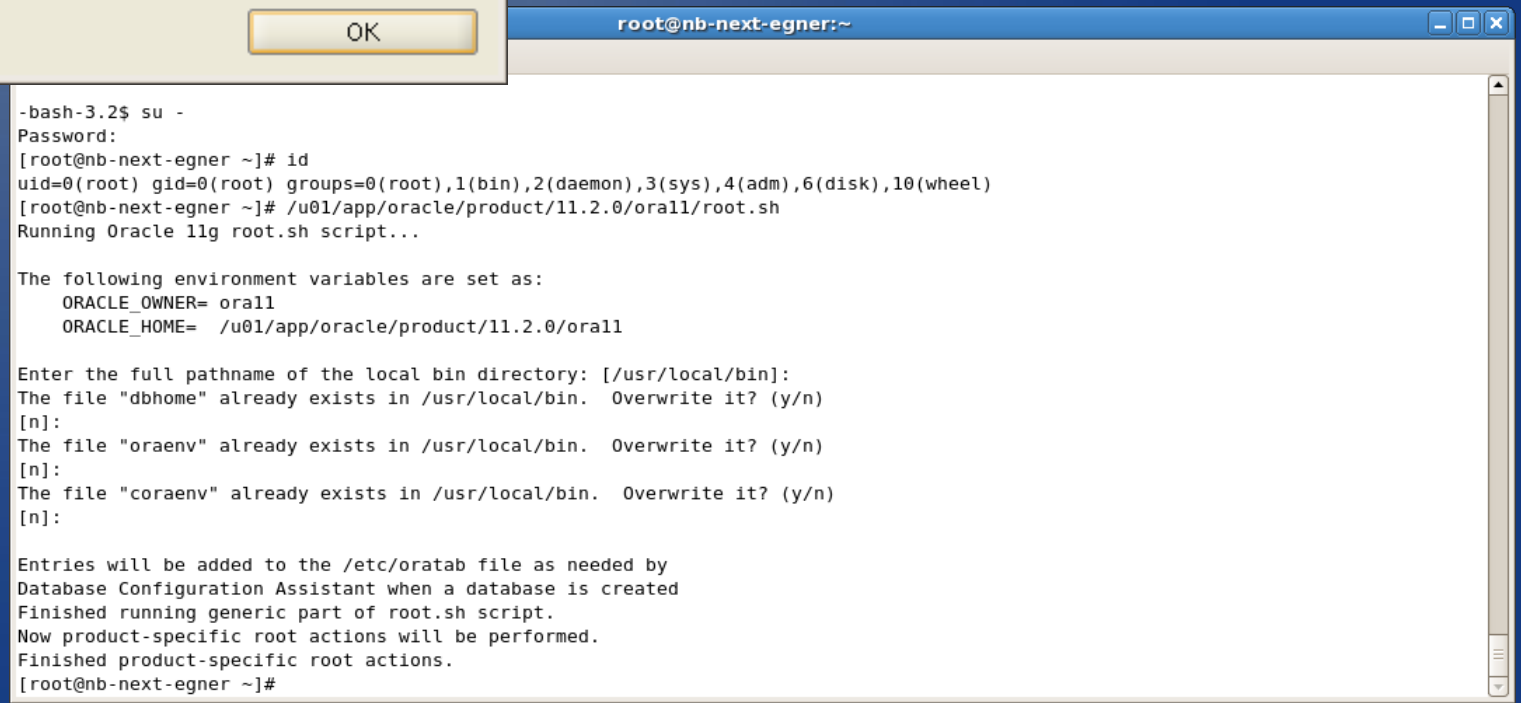


Installation – the actual installation

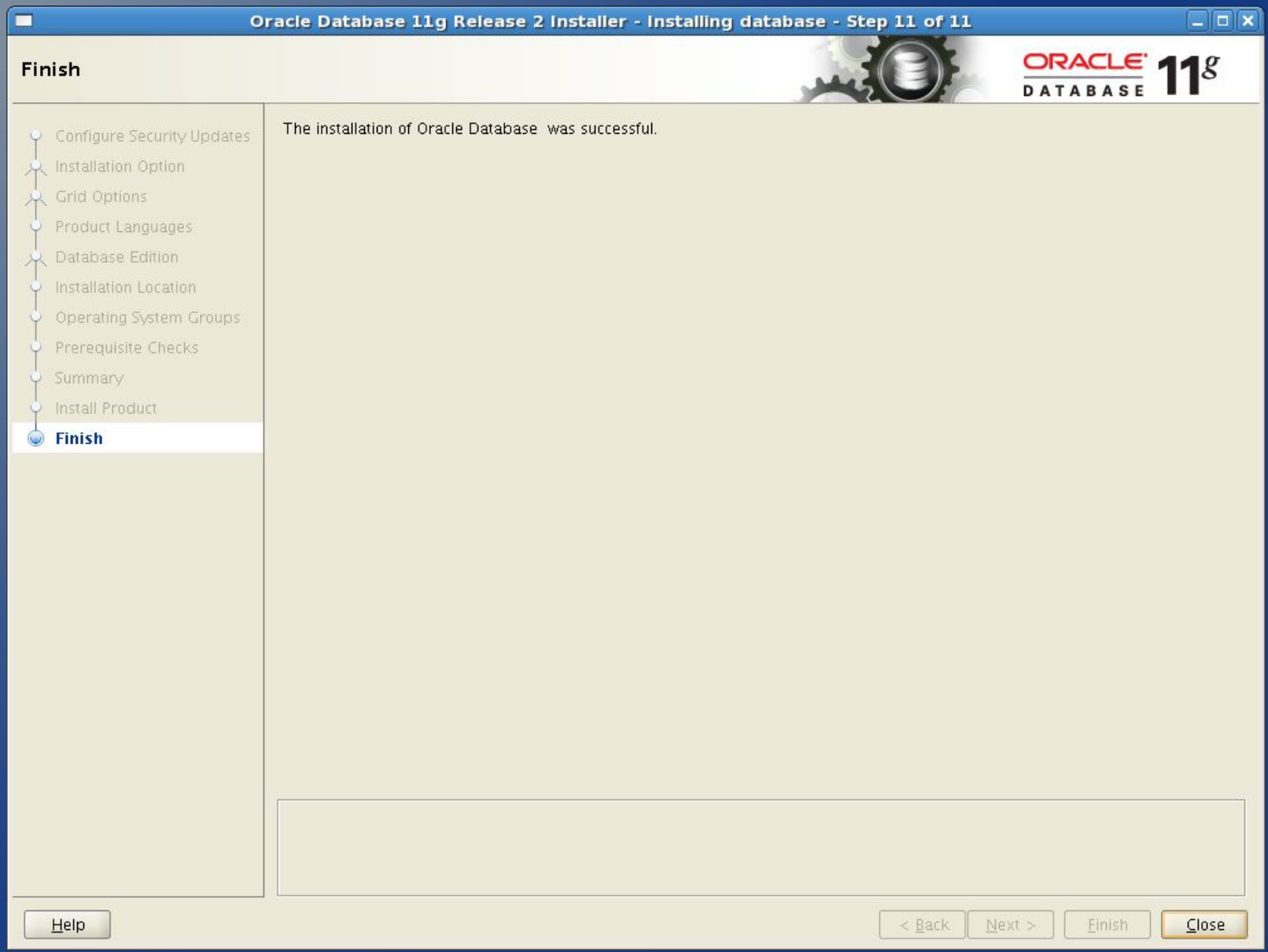


Installation – the actual installation

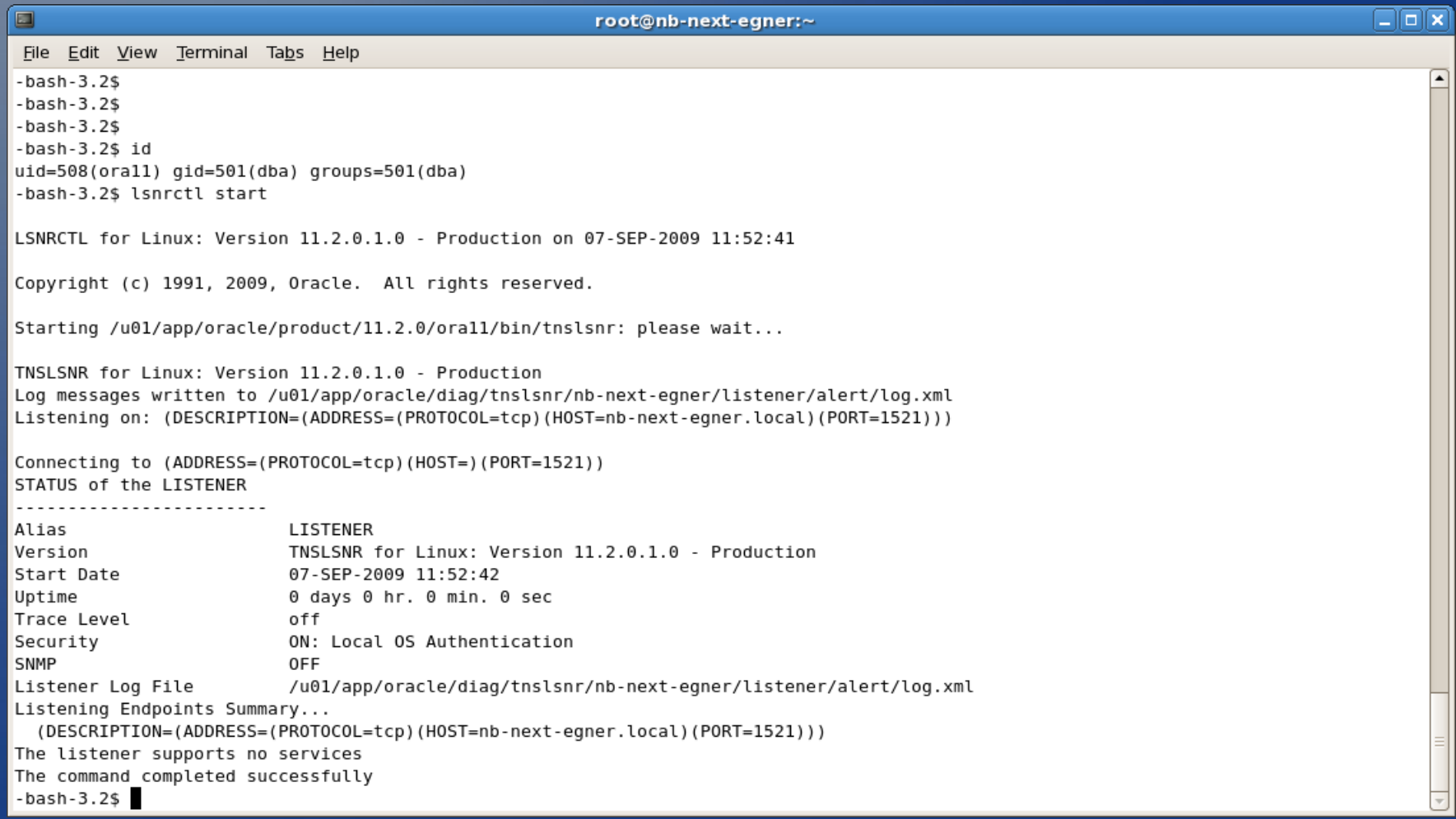




Installation – the actual installation

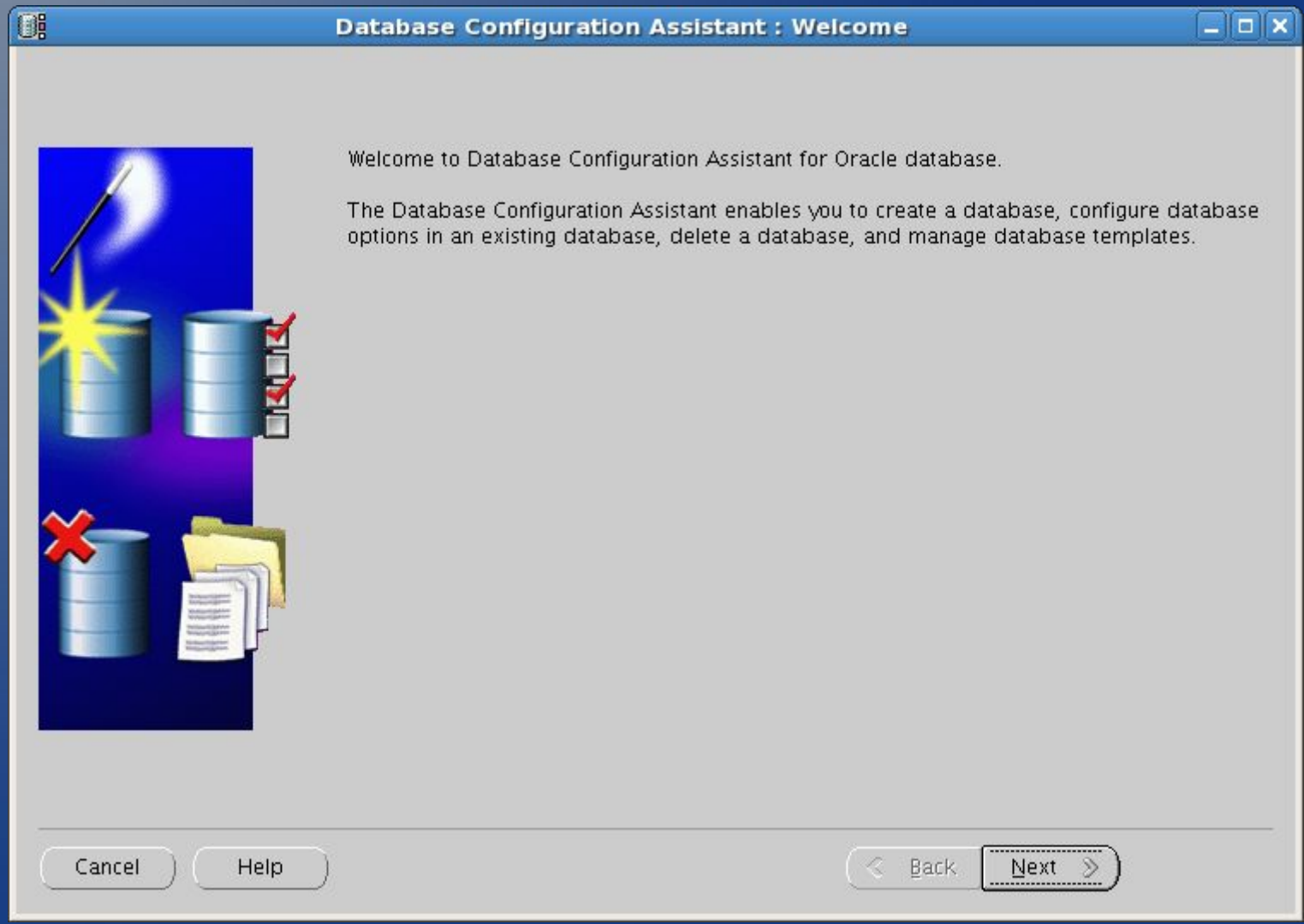


Installation – configure the listener

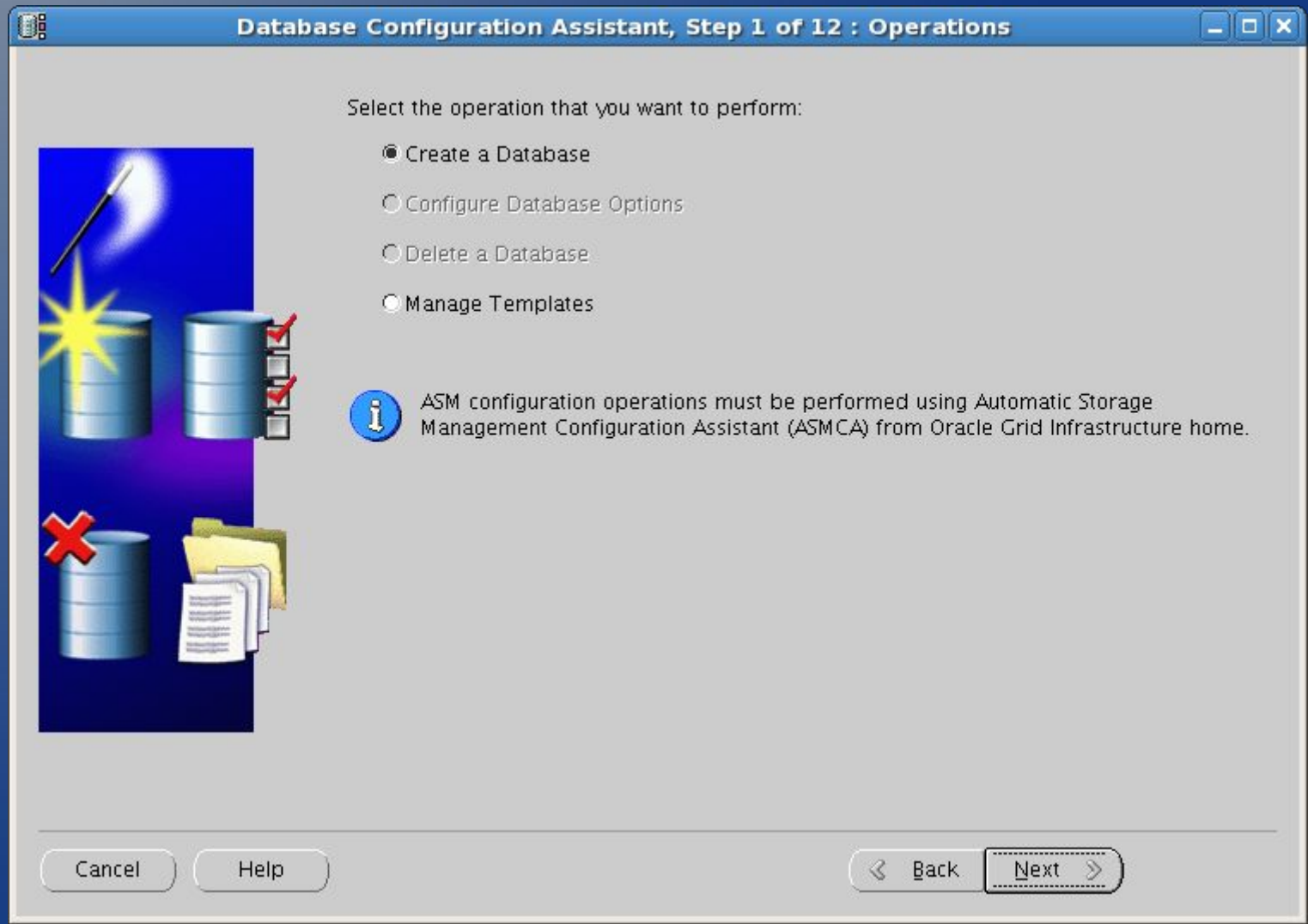


Installation – create the database

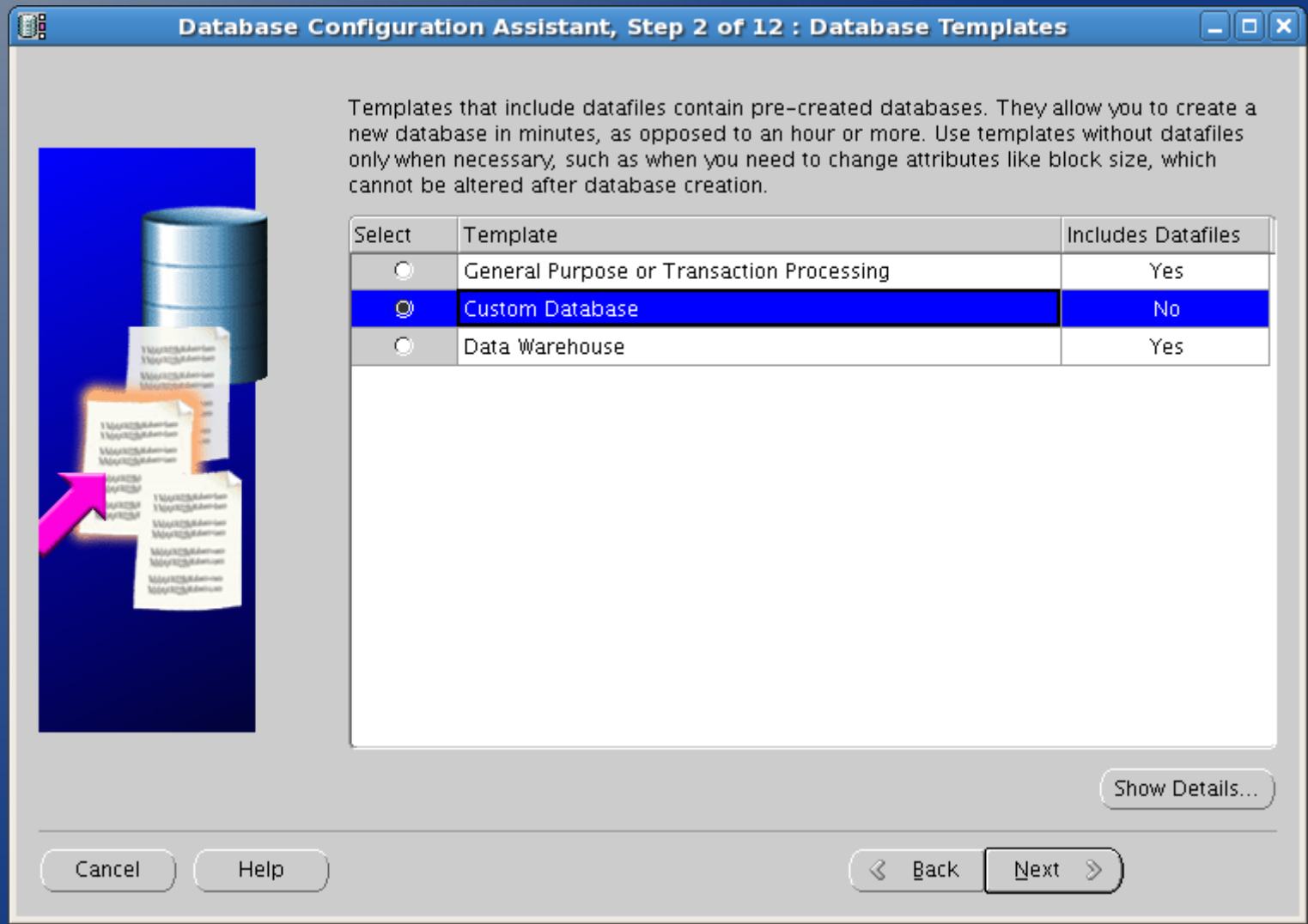
Installation – create the database



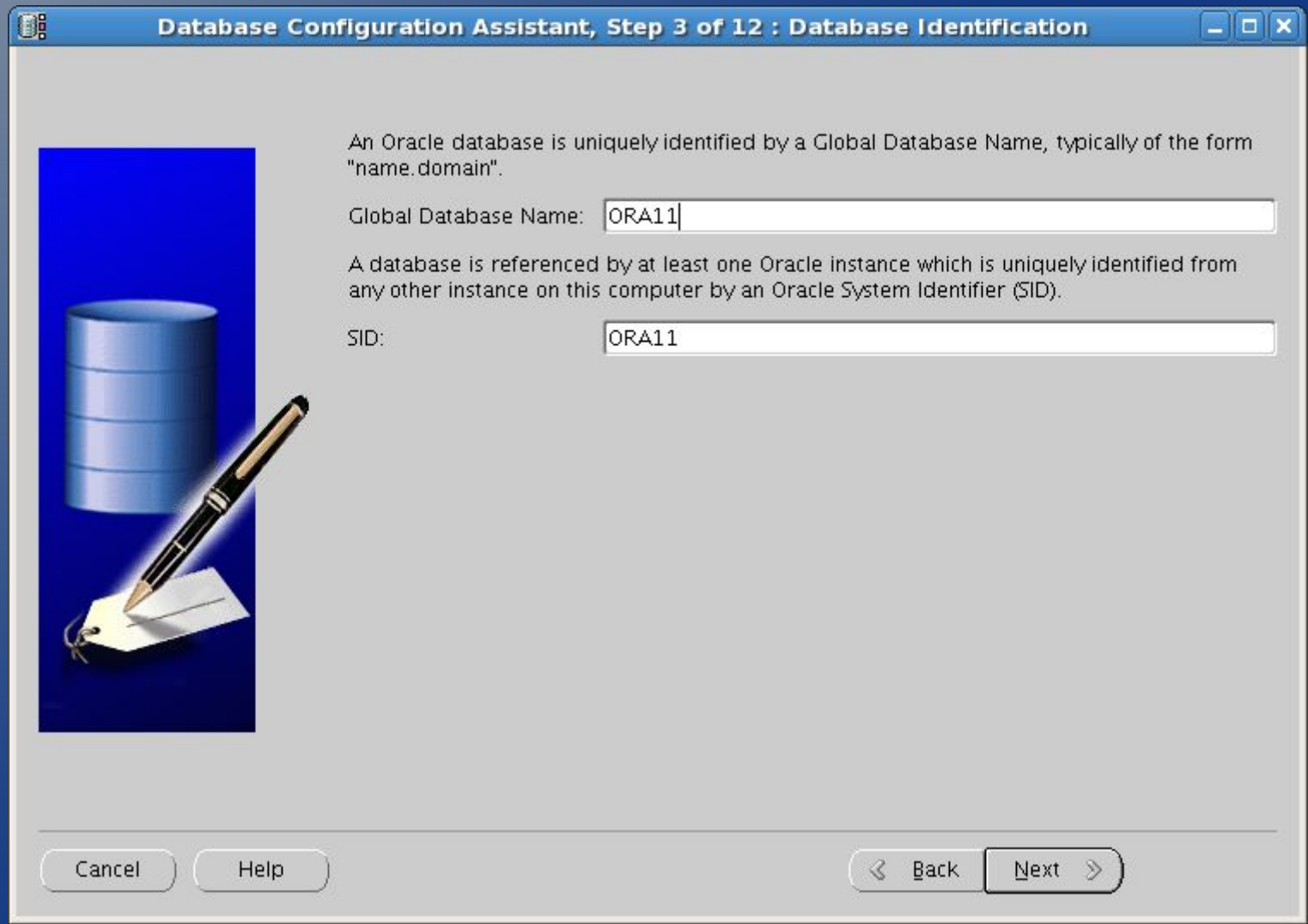
Installation – create the database



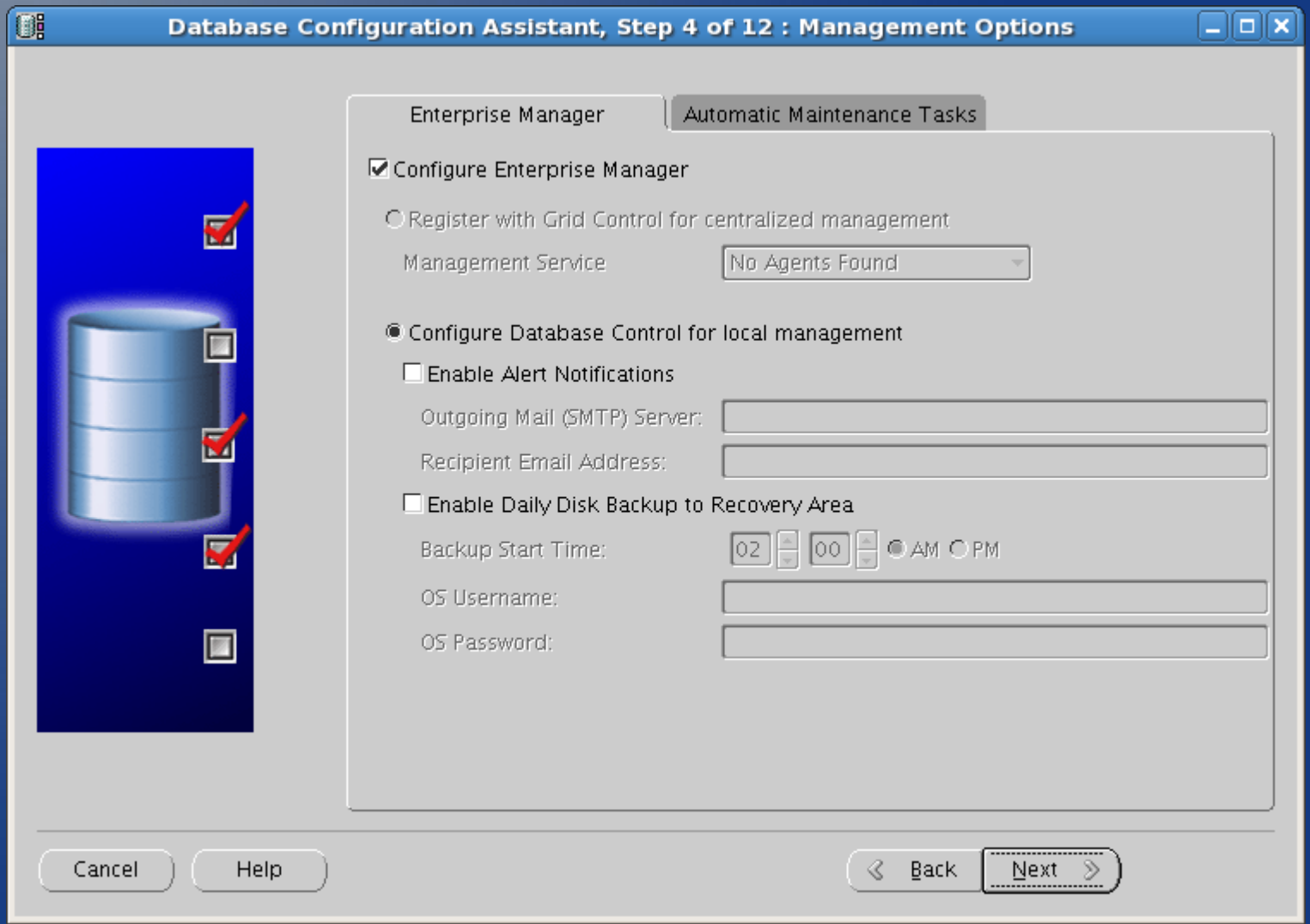
Installation – create the database



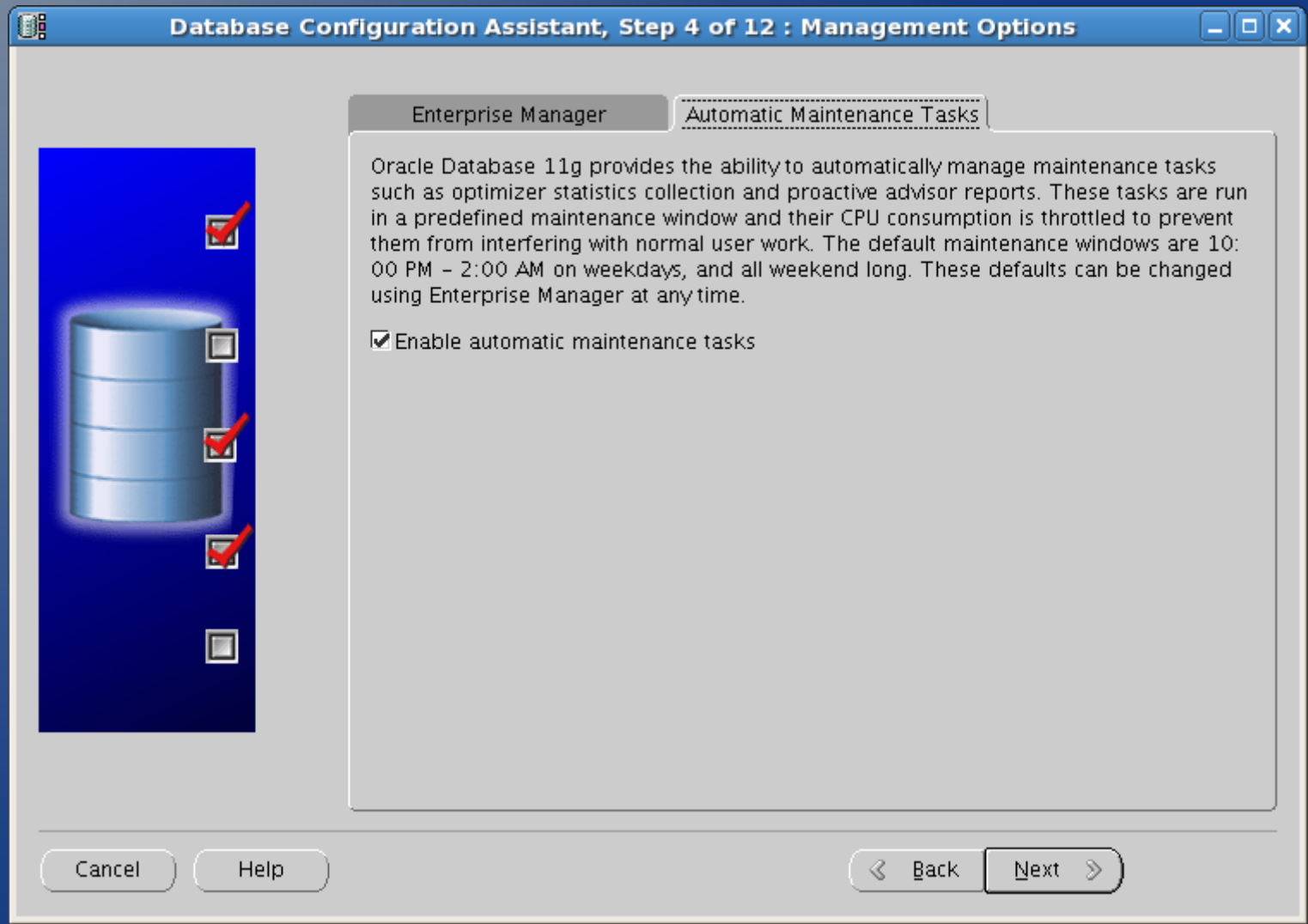
Installation – create the database



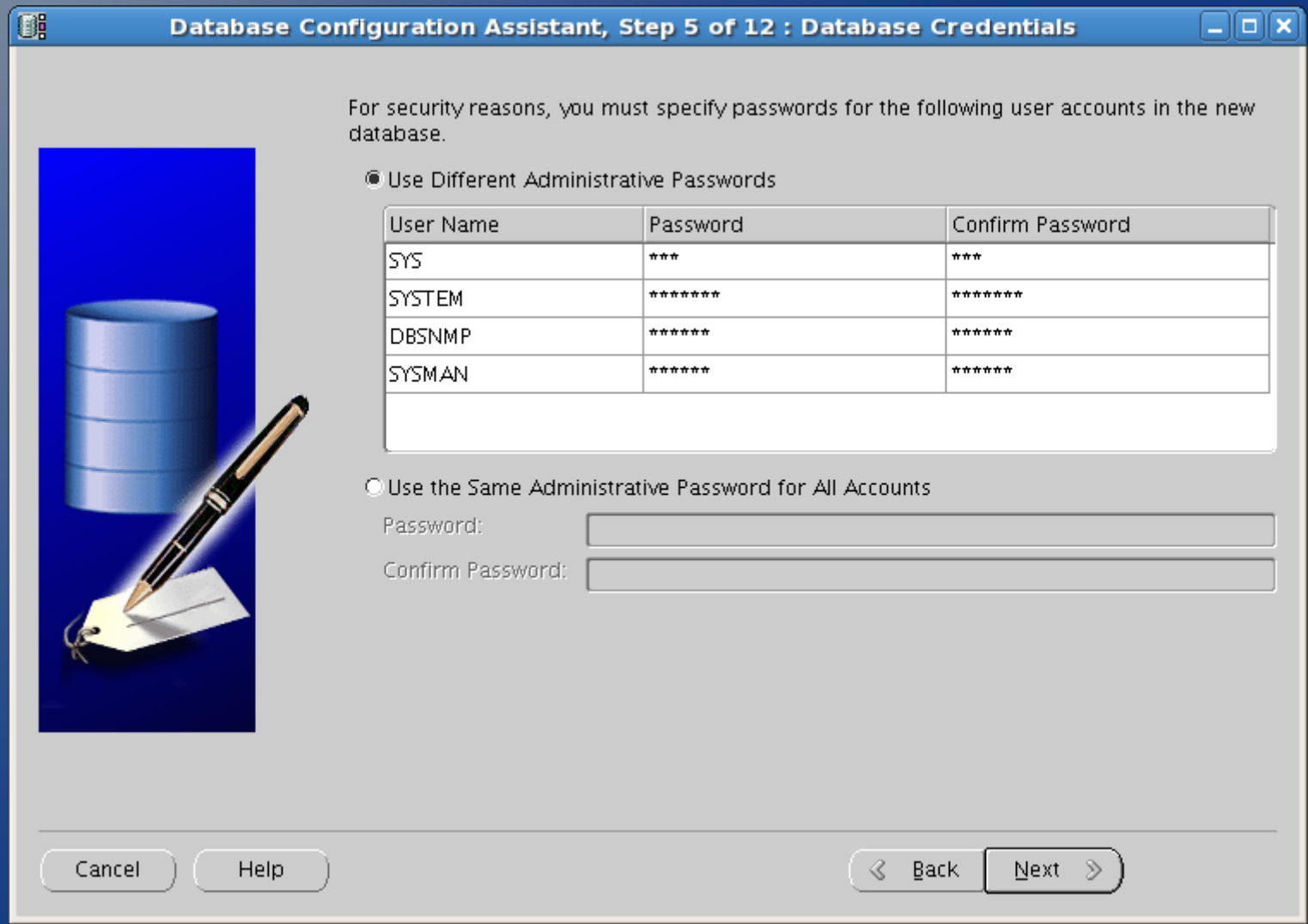
Installation – create the database



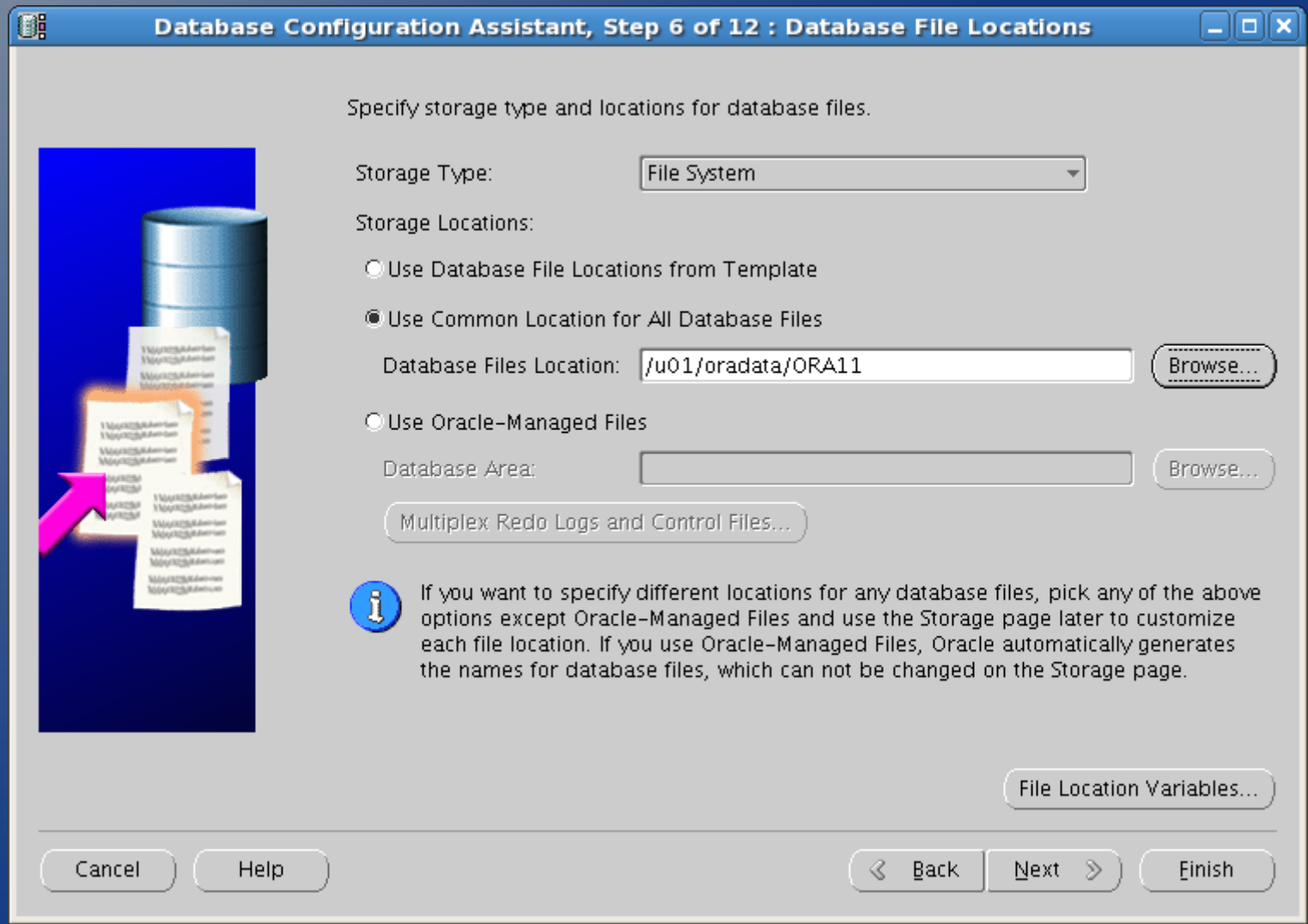
Installation – create the database



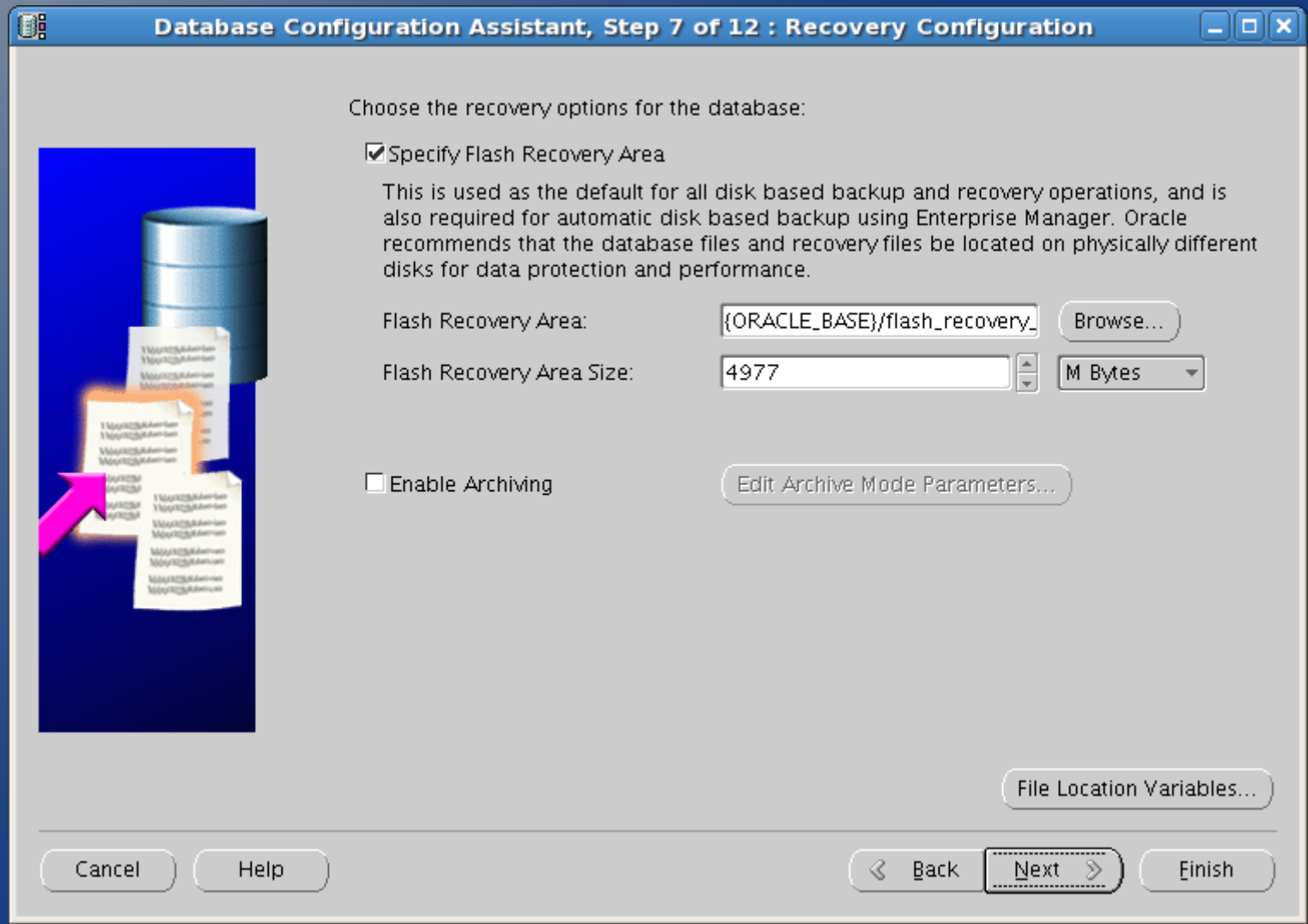
Installation – create the database



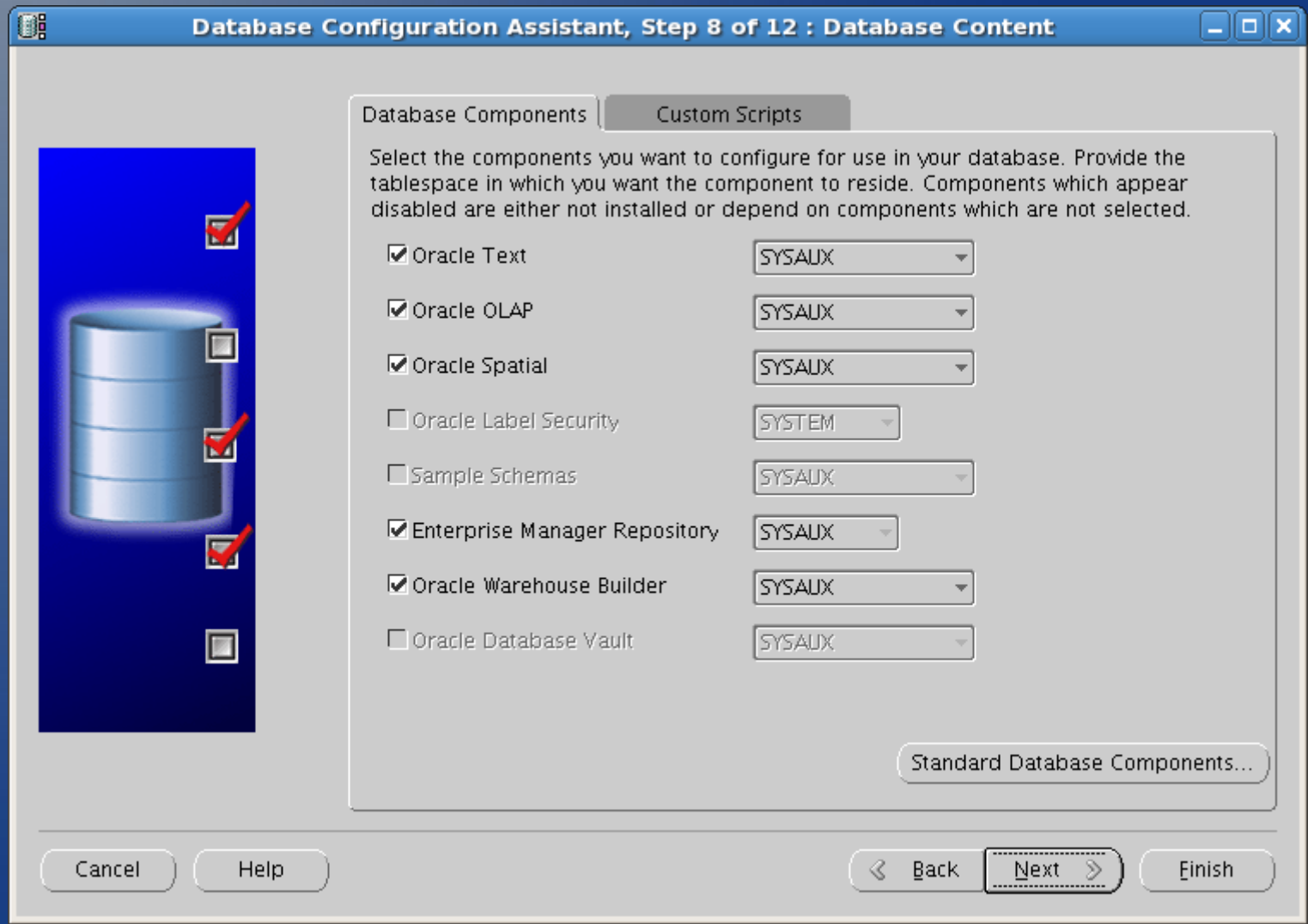
Installation – create the database



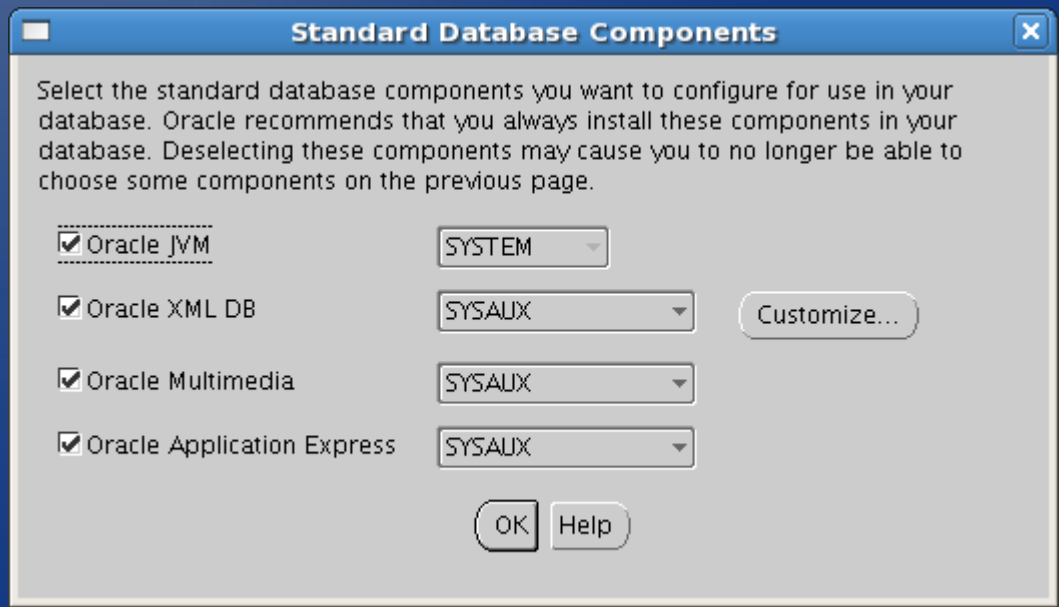
Installation – create the database



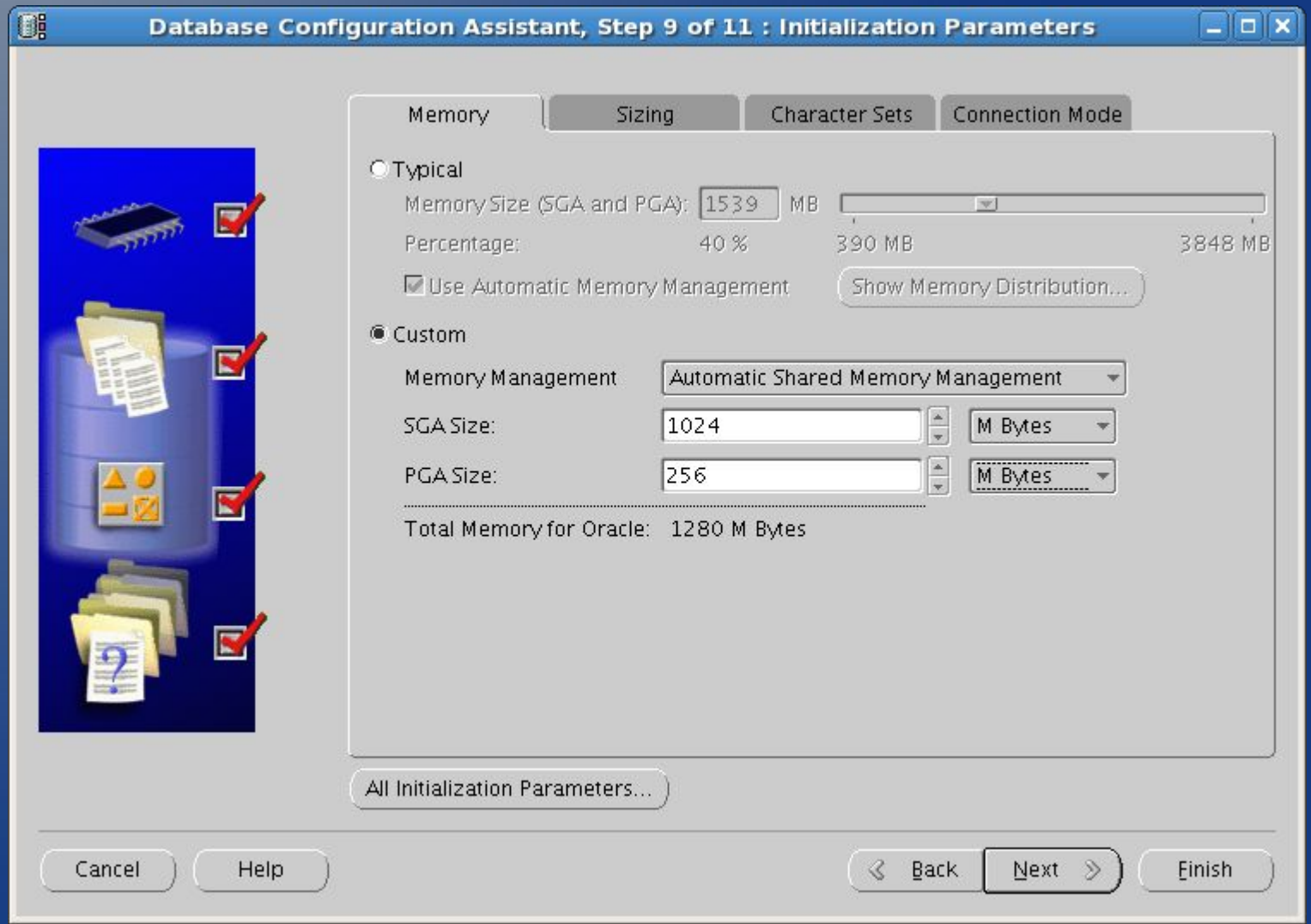
Installation – create the database



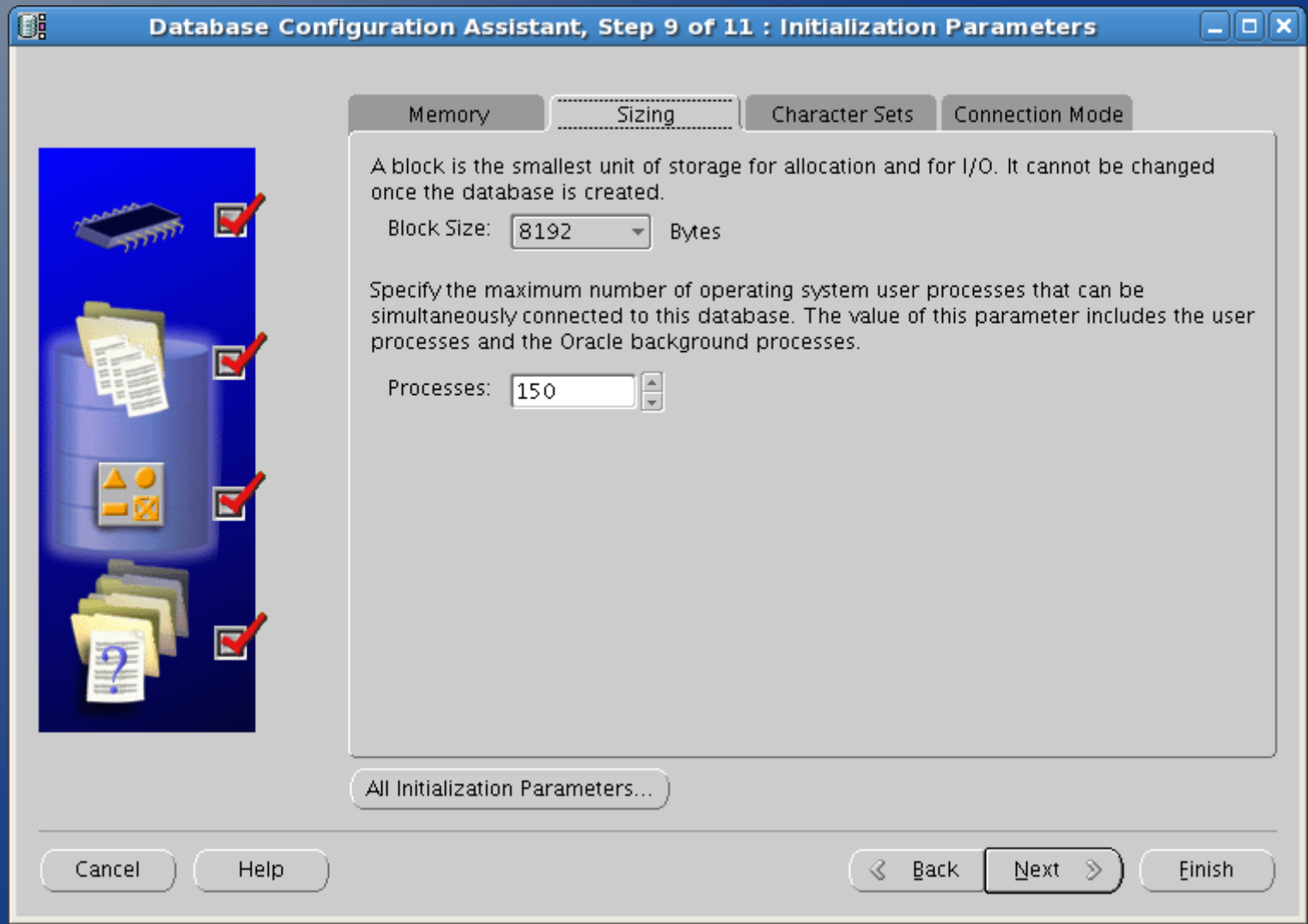
Installation – create the database



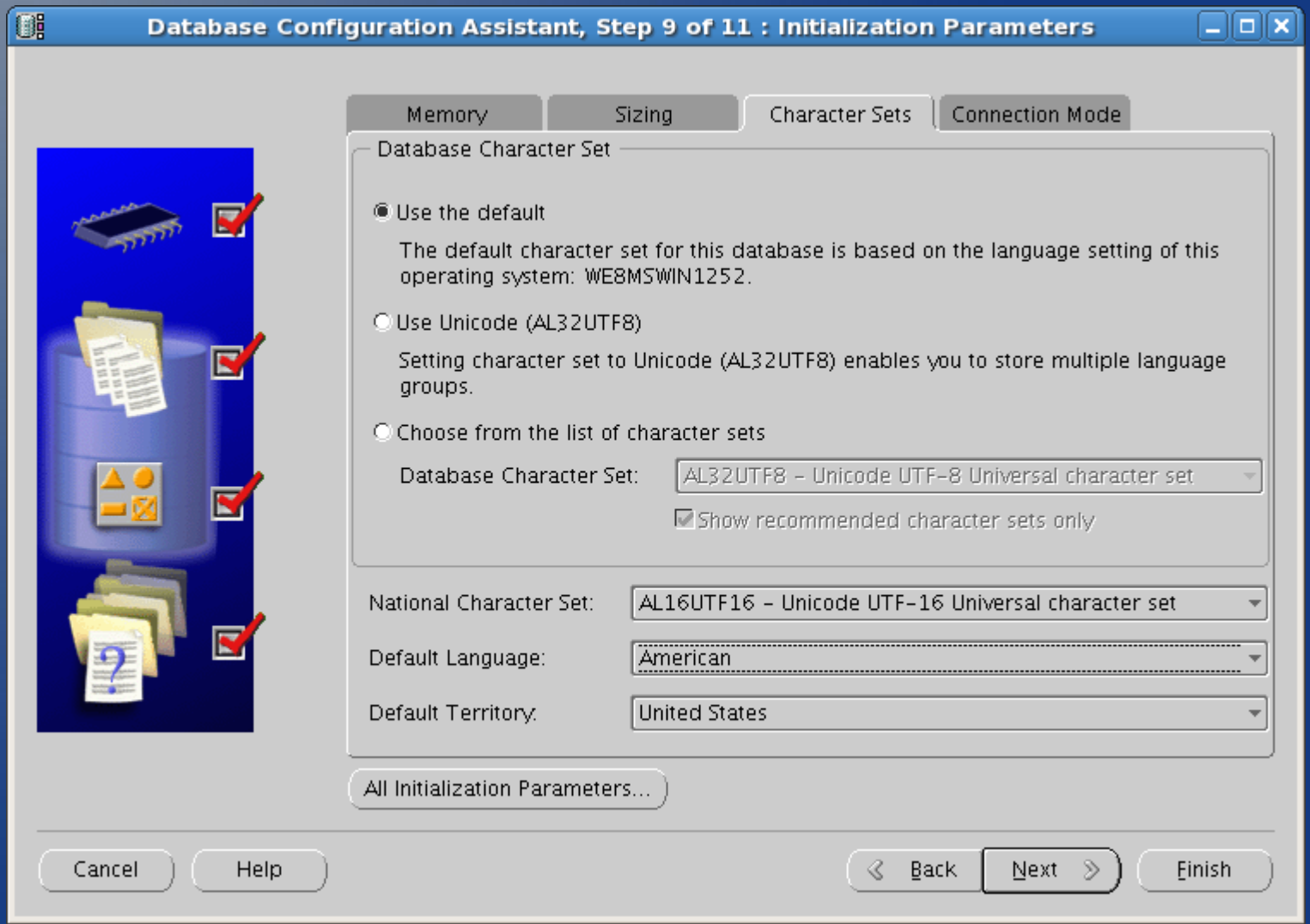
Installation – create the database



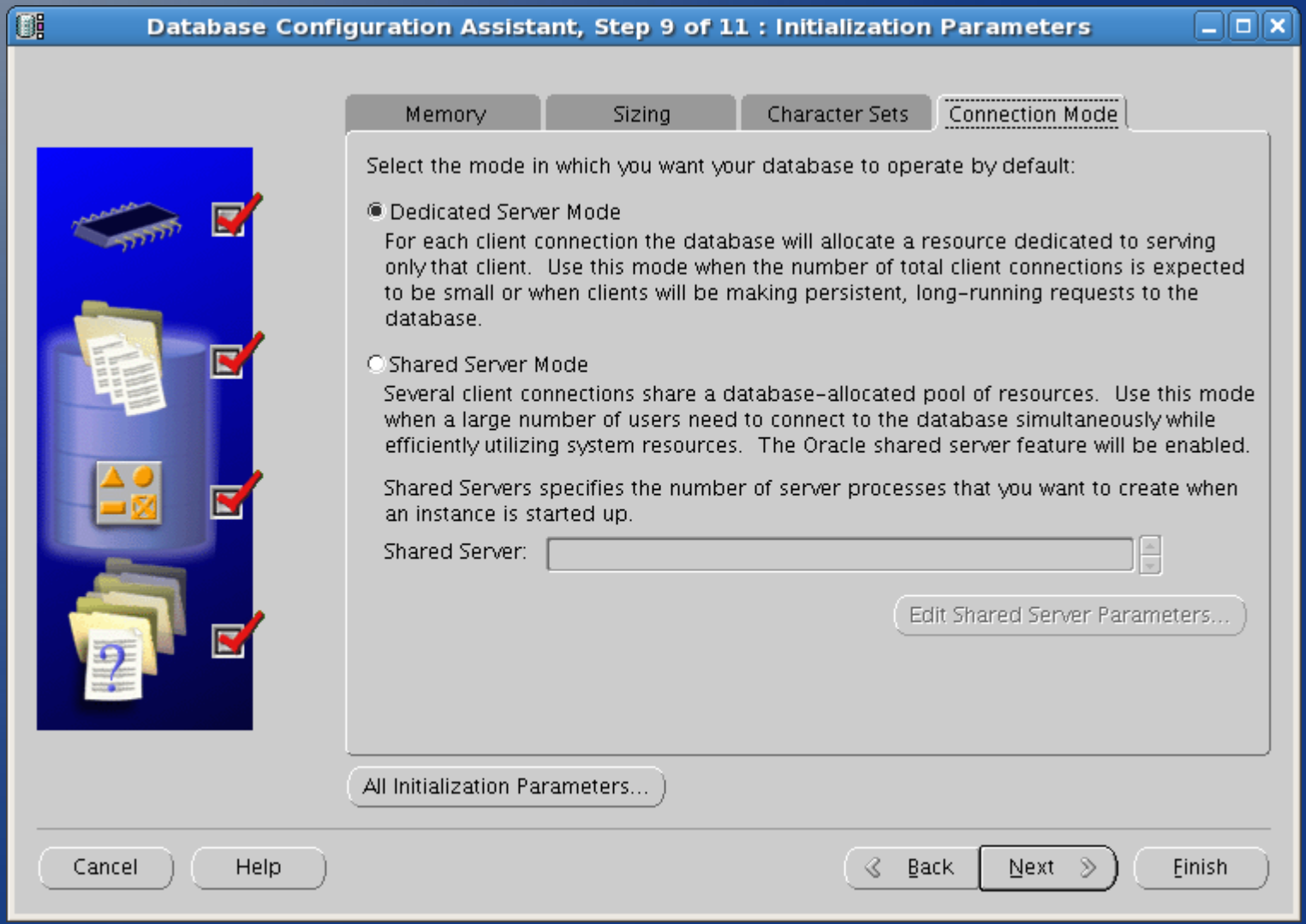
Installation – create the database



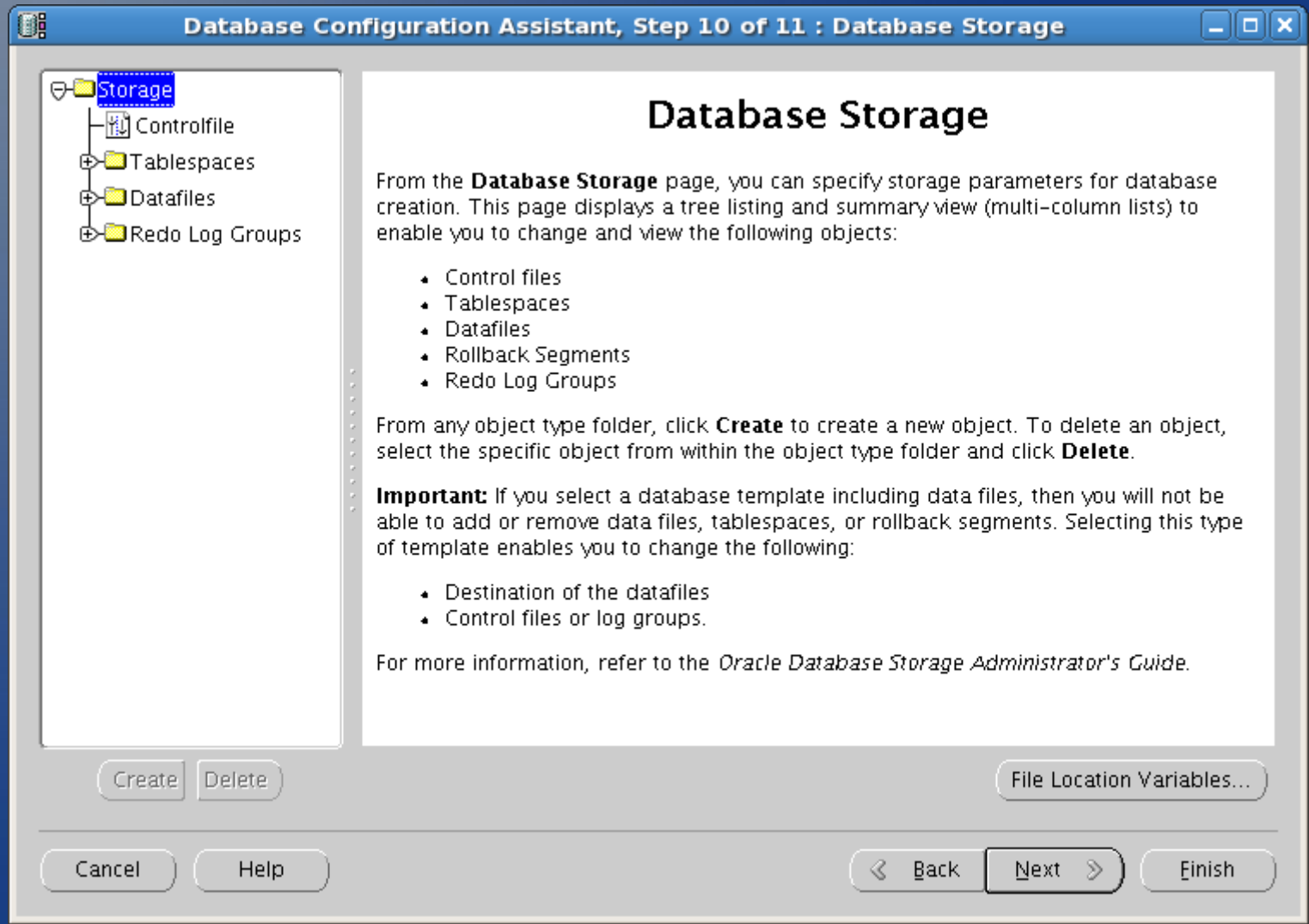
Installation – create the database



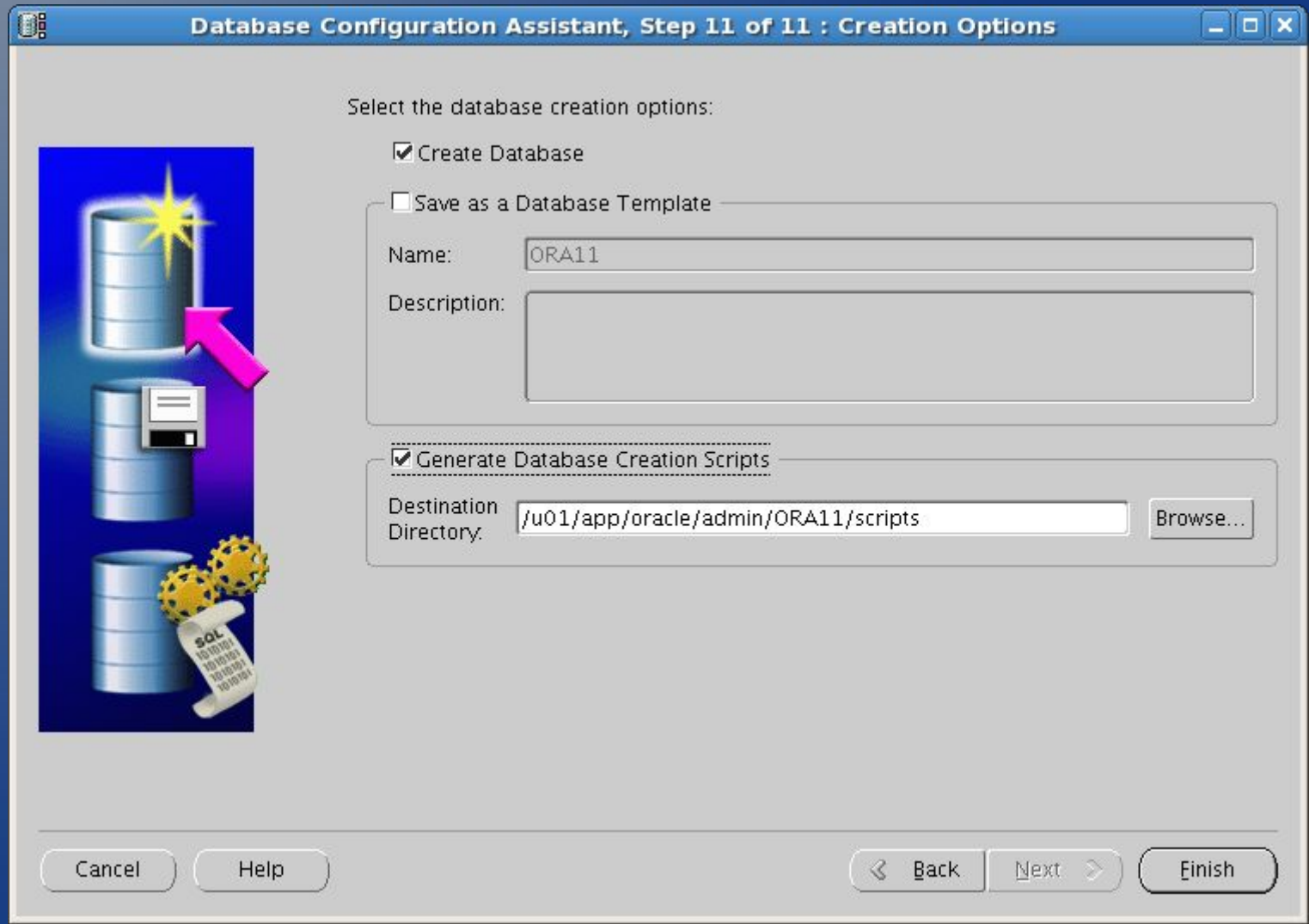
Installation – create the database



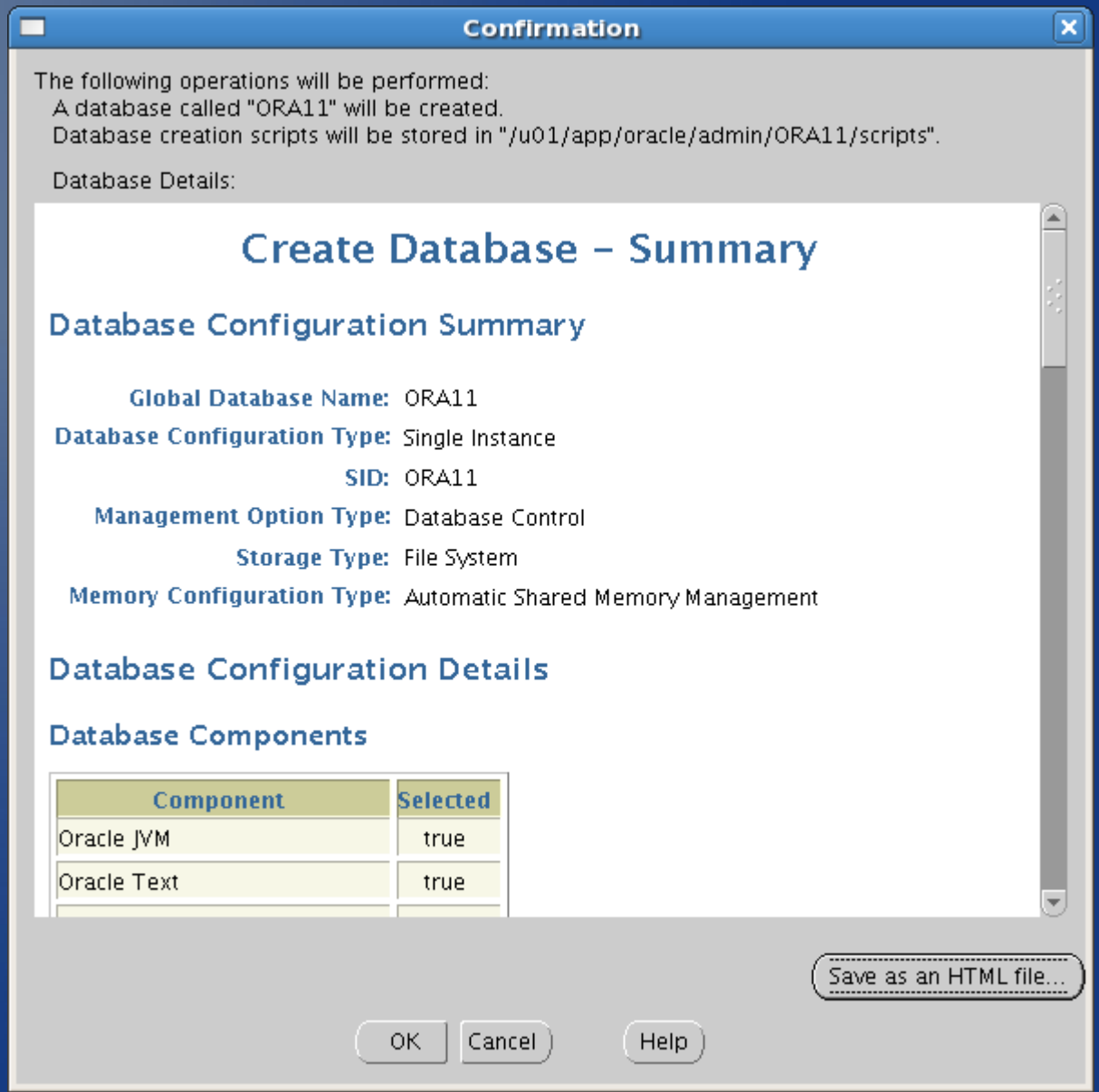
Installation – create the database



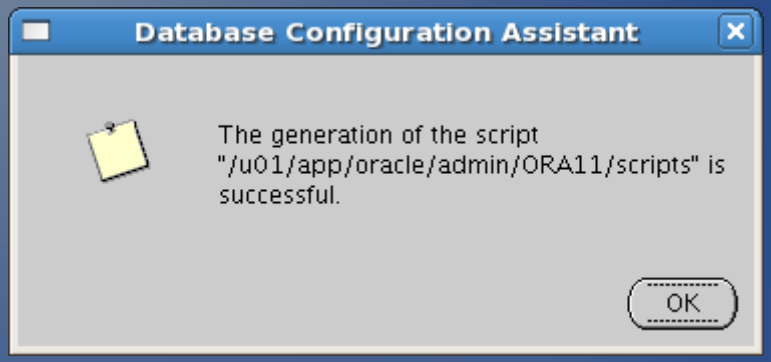
Installation – create the database



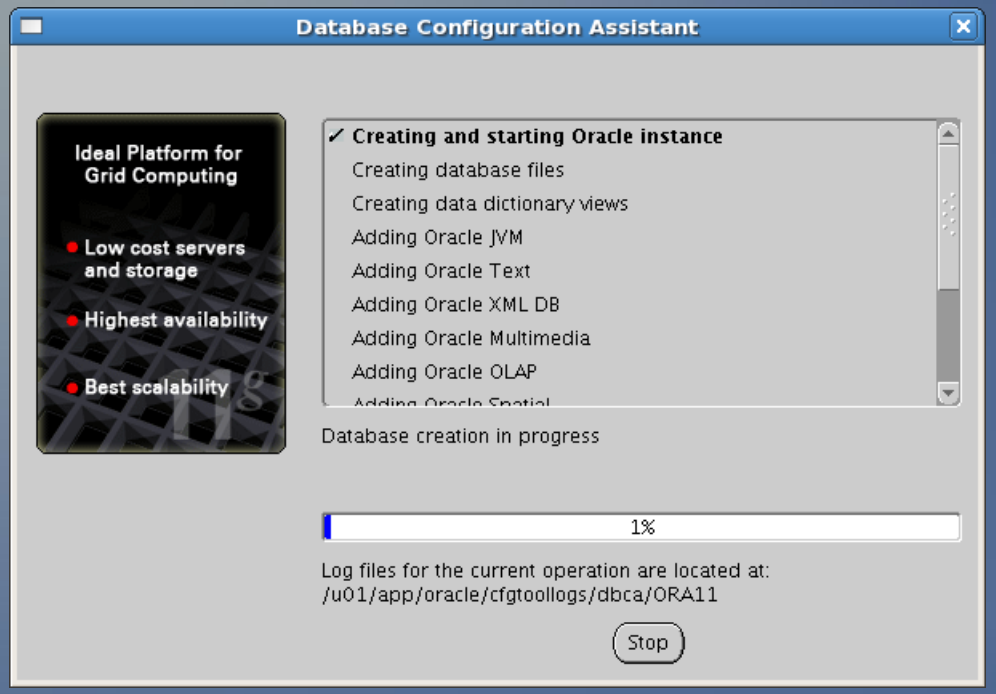
Installation – create the database

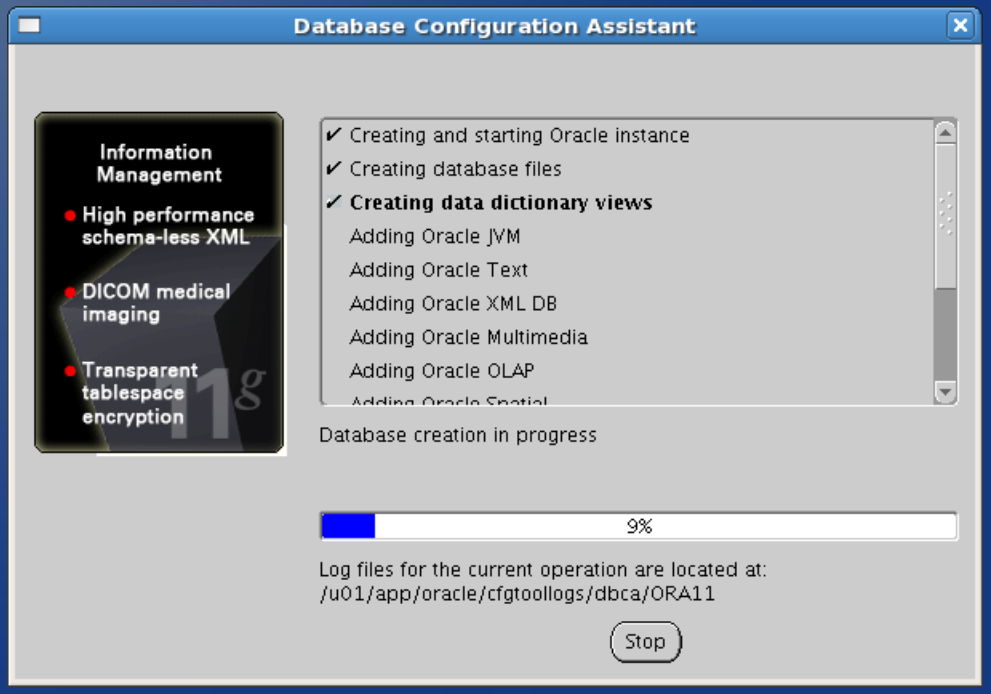


Installation – create the database

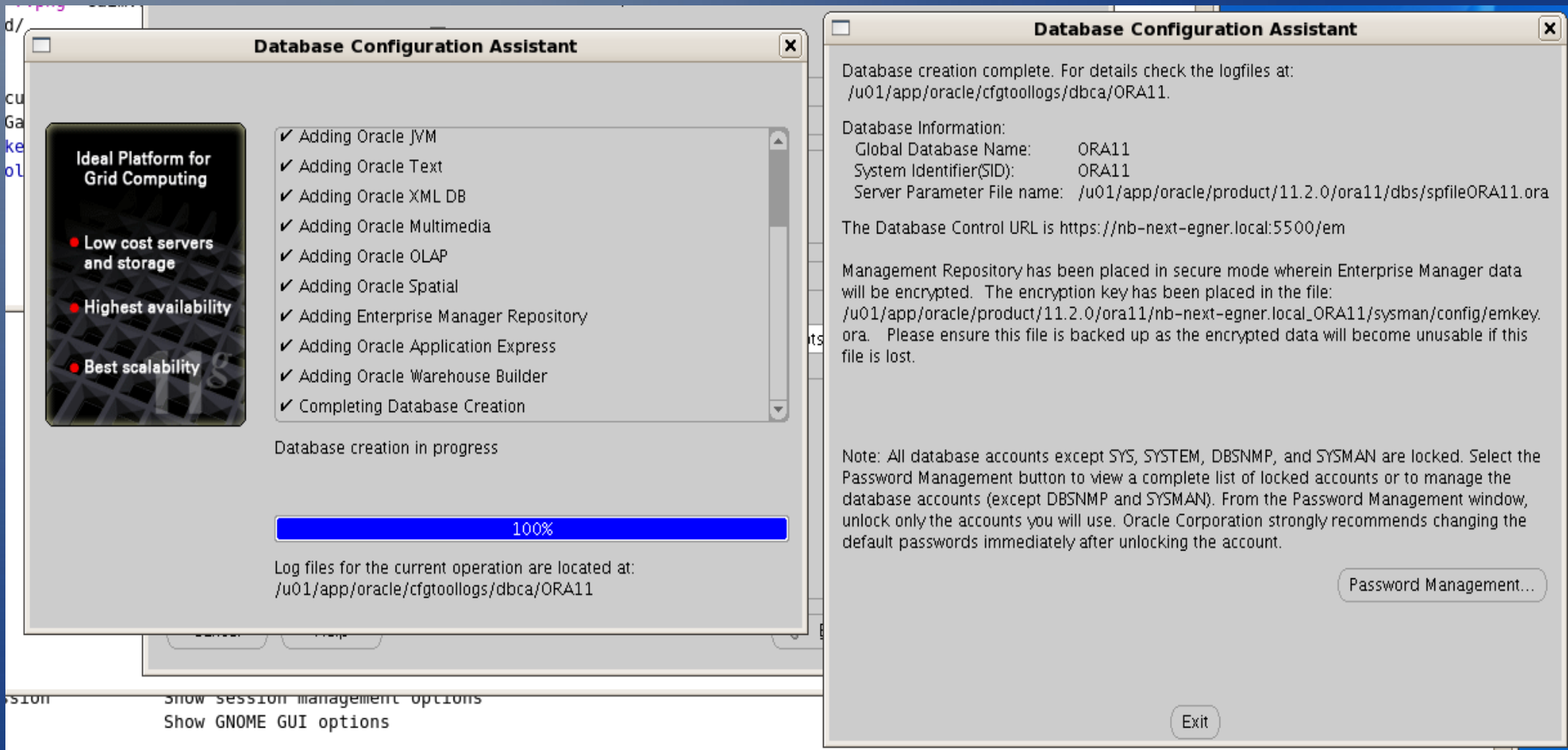


Installation – create the database





Installation – create the database



Part 2 –
Oracle Grid Infrastructure

System configuration

- System configuration:
  - two virtual machines (VMWARE)
    - 1 vCPU
    - 2 GB RAM → bare minimum possible
    - 40 GB Disk
  - Storage exported via ISCSI
    - 4 LUNs with 2 GB each
    - 2 LUNs with 30 GB each
  - Operating system configuration
    - Oracle Enterprise Linux 5.3 x86_64 (Kernel 2.6.18-128.el5)
    - Installed packages: default system + development packages
  -

System configuration

- System configuration:
  - Cluster Name: „RAC“
  - Binary installation on local disk
  - OCR, Voting and datafiles stored in ASM

Installation Overview

- Installation of Oracle 11g Release 2 Grid Infrastructure
  - Configure Linux and pre-requirements
  - Configure Storage
  - Binary installation of grid infrastructure
- Installation of Oracle 11g Release 2 Database (either single or rac installation)

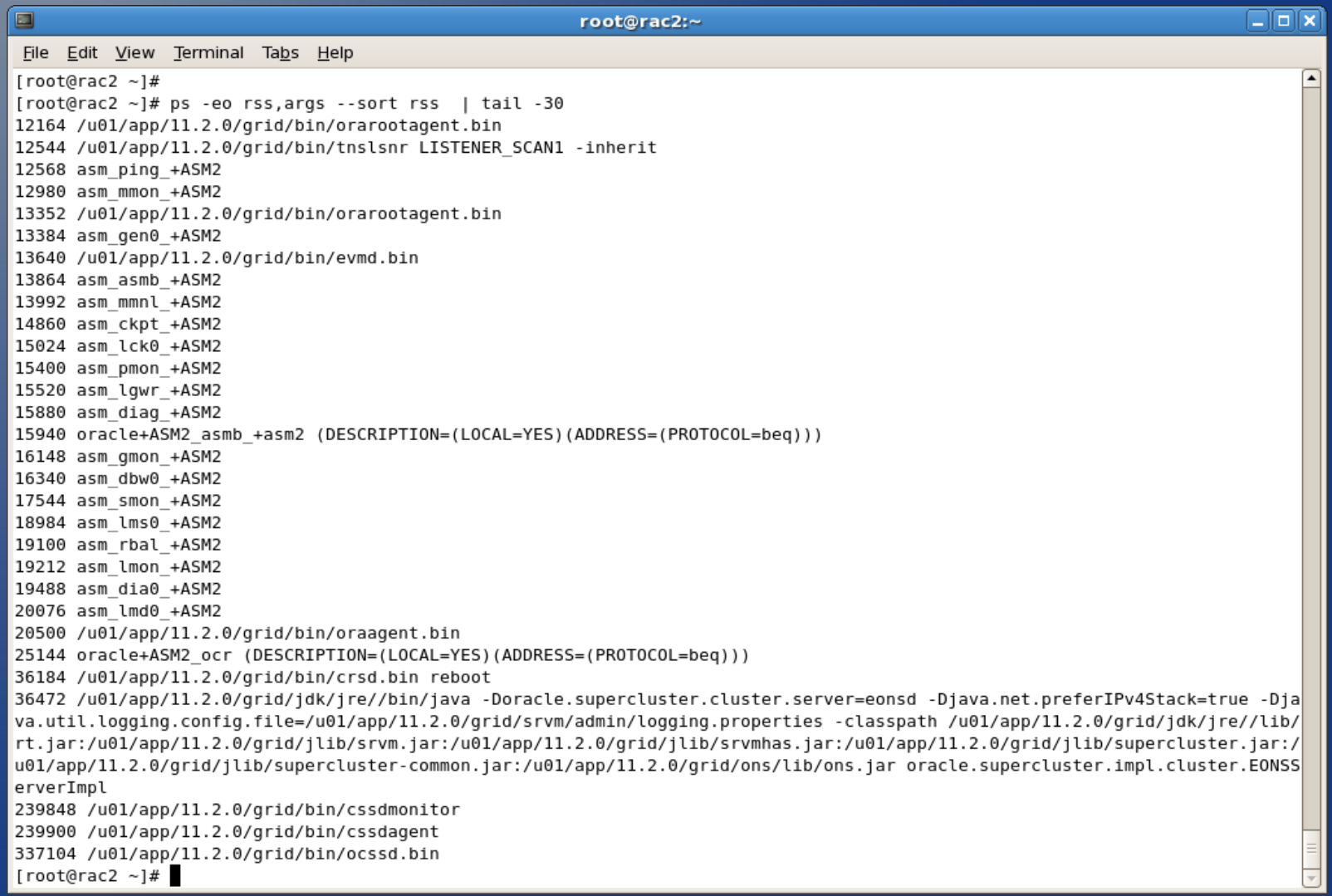
Installation – configure linux and requirements

- SWAP
  - Between 1 and 2 GB RAM → 1.5 times the size of RAM
  - Between 2 and 16 GB RAM → equal to size of RAM
  - > 16 GB RAM → 16 GB SWAP
- Memory
  - according to grid infrastructure documentation „>= 1 GB Memory“
  - bare minumum from authors experience:
    - 1 GB for grid infrastructure components
    - 500 MB for operating system
    - 1 GB for cluster database SGA/PGA/UGA

    = 2,5 GB bare minimum!

Installation – configure linux and requirements

- Memory (con´t)
  - See below for memory consumption with grid infrastructure installed → > 800 MB for infrastructure processes



Installation – configure linux and requirements

- Automatic Memory Management
  - Required /dev/shm with appropriate size (i.e. SGA of 16 GB required /dev/shm to be 16 GB+)
  - Huge Pages and autom. Memory Management are INCOMPATIBLE

Installation – configure linux and requirements

- Checking required packages (see required packages for single database installation; this applies here as well cause we will end up install a database in the end)
  - According to the documentation the following packages are needed:

    Binutils-2.17.50.0.6, compat-libstdc++-33-3.2.3, compat-libstdc++-33-3.2.3 (32 bit), elfutils-libelf-0.125, elfutils-libelf-devel-0.125, gcc-4.1.2, gcc-c++-4.1.2, glibc-2.5-24, glibc-2.5-24 (32 bit), glibc-common-2.5, glibc-devel-2.5, glibc-devel-2.5 (32 bit), glibc-headers-2.5, ksh-20060214, libaio-0.3.106, libaio-0.3.106 (32 bit), libaio-devel-0.3.106, libaio-devel-0.3.106 (32 bit), libgcc-4.1.2, libgcc-4.1.2 (32 bit), libstdc++-4.1.2, libstdc++-4.1.2 (32 bit), libstdc++-devel 4.1.2, make-3.81, sysstat-7.0.2, unixODBC-2.2.11, unixODBC-2.2.11 (32 bit), unixODBC-devel-2.2.11, unixODBC-devel-2.2.11 (32 bit)
- On sample system with default + development packages installed only the following rpms were missing:

  rpm -ihv libaio-devel-0.3.106-3.2.* libstdc++43-devel-4.3.2-7.el5.* sysstat-7.0.2-3.el5.x86_64.rpm unixODBC-2.2.11-7.1.* unixODBC-devel-2.2.11-7.1.*

Installation – configure linux and requirements

- Shell Limits
  - In /etc/security/limits.conf

```
grid                soft    nproc     16384
grid                hard    nproc     16384
grid                soft    nofile    65536
grid                hard    nofile    65536
grid                soft    stack     10240
grid                hard    stack     10240
```

  - In /etc/pam.d/login add if not exists

```
session    required     pam_limits.so
```

Installation – configure linux and requirements

- Kernel Limits (MINIMUM values)
  - /etc/sysctl.conf

    kernel.sem=250 32000 100 128
    kernel.shmall=2097152
    kernel.shmmax=536870912
    kernel.shmmni=4096
    fs.file-max=6815744
    fs.aio-max-nr=1048576
    net.ipv4.ip_local_port_range=9000 65500
    net.core.rmem_default=262144
    net.core.rmem_max=4194304
    net.core.wmem_default=262144
    net.core.wmem_max=1048576

    - SuSE only -
    vm.hugetlb_shm_group=<gid of osdba group>

  - The values in /etc/sysctl.conf should be tuned (i.e. according to the number of instance, available memory, number of connections,...)

Installation – configure linux and requirements

- Kernel Limits on Linux (Calculate them) - „kernel.sem“

```
semmns = Total number of semaphores systemwide =

        2 * sum (process parameters of all database instances on the system)
        + overhead for background processes
        + system and other application requirements

semmsl = total semaphoren for each set

semmni = total semaphore sets =

        semmns divided by semmsl, rounded UP to nearest multiple to 1024

kernel.sem = <semmsl semmns semopm semmni>

    semmsl              → set to 256
    semmns              → set total number of semaphoren (see above!)
    semopm              → 100; in documentation not explicitly described
    semmni              → see calculcation above
```

Installation – configure linux and requirements

- Kernel Limits on Linux (Calculate them) - „kernel.shmall

kernel.shmall = This parameter sets the total amount of shared memory **pages** that can be used system wide. Hence, SHMALL should always be at least ceil(shmmax/PAGE_SIZE).
PAGE_SIZE is usually 4096 bytes unless you use Big Pages or Huge Pages which supports the configuration of larger memory pages.

(quoted from: www.puschitz.com/TuningLinuxForOracle.shtml)

Installation – configure linux and requirements

- Kernel Limits on Linux (Calculate them) - „kernel.shmmax“

```
kernel.shmmax = the maximum size of a single shared memory segment in bytes that
                a linux process can allocate

If not set properly database startup can fail with:

    ORA-27123: unable to attach to shared memory segment
```

Installation – configure linux and requirements

- Kernel Limits on Linux (Calculate them) - misc parameter

```
kernel.shmmni = system wide number of shared memory segments; Oracle recommendation
                for 11g Release 1 „at least to 4096“; i did not found anything
                for Release 2....

fs.file-max = maximum number of open files system-wide; must be at least „6815744“

fs.aio-max-nr = concurrent outstanding i/o requests; must be set to „1048576“

net.ipv4.ip_local_port_range = mimimum and maximum ports for use; must be set to
                               minimal „9000“ and „65500“ as maximum

net.core.rmem_default = the default size in bytes of the receive buffer; must
                        be set at least to „262144“

net.core.rmem_max = the maximum size in bytes of the receive buffer; must be
                    set at least to „4194304“

net.core.wmem_default = the default size in bytes of the send buffer; must
                        be set at least to „262144“

net.core.wmem_max = the maximum size in bytes of the send buffer; must be
                    set at least to „1048576“
```

Installation – configure linux and requirements

- Networking
  - Works completely different than 10g or 11g R1!
  - At least two separated networks (public and private) and therefore two network interfaces required
  - ATTENTION: Interface names must be equal on ALL nodes! (i.e. If private network interface on node A is eth2 the private network interface name on all other nodes must be eth2 as well....)
  - Recommendation: Use bonding for:
    - Static naming (even if you use only one interface per bond)
    - Failover / Load Sharing
    - → we will use network bonding with only one interface in the following
  - IP adresses can be given by two schemes:
    - GNS (grid naming service) – automatic ip numbering
    - Manual Mode
    - → we will use manual ip adressing mode in the following

Installation – configure linux and requirements

- Networking (con´t)
  - GNS mode requires:
    - one fixed public IP for each node
    - one dhcp virtual IP for each node
    - one hdcp for fixed private IP for each node
    - three dhcp IP for the SCAN

    - Thougths by the author:
      - new
      - more complex
      - if working quite easy adding of an node; at least from the ip numbering point of view – but how often do you add a node?

Installation – configure linux and requirements

- Networking (con´t)
  - Manual Mode ip adressing requires:
    - one public IP for each node
    - one virtual IP for each node
    - one private IP for each node
    - one to three (recommended) IPs for providing the SCAN name

Installation – configure linux and requirements

- Networking (con´t)
  - Naming schema used in the following (remember: 2-node-cluster)

Identity	Home Node	Host Node	Given Name	Type	Address
Node 1 Public	Node 1	Node 1	rac1	Public	192.168.180.10
Node 1 VIP	Node 1	sel. by clusterware	rac1-vip	Virtual	192.168.180.100
Node 1 Private	Node 1	Node 1	rac1-priv	Private	192.168.181.10
Node 2 Public	Node 2	Node 2	rac2	Public	192.168.180.20
Node 2 VIP	Node 2	sel. by clusterware	rac2-vip	Virtual	192.168.180.200
Node 2 Private	Node 2	Node 2	rac2-priv	Private	192.168.181.20
SCAN VIP 1	none	sel. by clusterware	rac-scan	Virtual	192.168.180.5
SCAN VIP 2	none	sel. by clusterware	rac-scan	Virtual	192.168.180.6
SCAN VIP 3	none	sel. by clusterware	rac-scan	Virtual	192.168.180.7

cluster name: „RAC“

Installation – configure linux and requirements

- Configure Network Bonding
  - In /etc/modprobe.conf add line:

```
alias bond0 bonding
alias bond1 bonding
options bonding miimon=100 mode=1 max-bonds=2
```

(„mode=1“ means active/passive failover... see „bonding.txt“ in kernel sources for more options)

  - /etc/sysconfig/network-scripts/ifcfg-bond0 looks like:

```
DEVICE=bond0
BOOTPROTO=none
ONBOOT=yes
NETWORK=192.168.180.0
NETMASK=255.255.255.0
IPADDR=192.168.180.10
USERCTL=no
```

Installation – configure linux and requirements

- Configure Network Bonding (con´t)
  - /etc/sysconfig/network-scripts/ifcfg-eth0 looks like:

```
DEVICE=eth0
BOOTPROTO=none
ONBOOT=yes
MASTER=bond0
SLAVE=yes
USERCTL=yes
```

(Note: Add a second interface to achive real fault tolerance.... for our testing environment we use bonding to provide a consistent name schema)

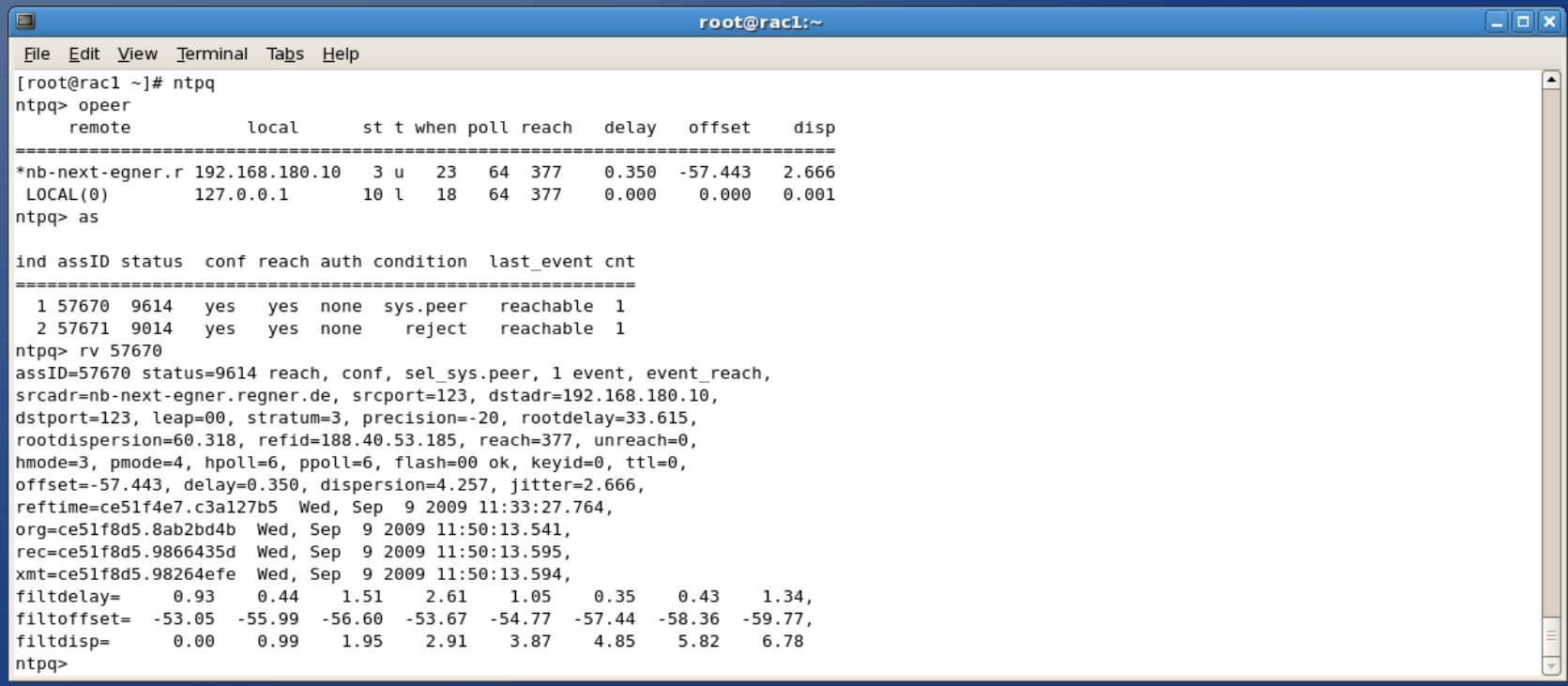
  - configuration for bond1 is not shown... just alter interface names and IPs

Installation – configure linux and requirements

- Configure NTP
  - Grid Infrastructure provides ntp-like time synchronization with „ctssd“ (cluster time synchronization service)
  - ctssd is provided in case connections to ntp servers are not possible
  - If **no** running („chkconfig ntpd off“) and configured („rm /etc/ntp.conf“) ntpd is found ctssd will be used
  - If ntpd is found ctssd will start in observer mode
  - ATTENTION: Set the „-x“ flag if you use ntp to prevent ntp from stepping the clock in /etc/sysconfig/ntpd

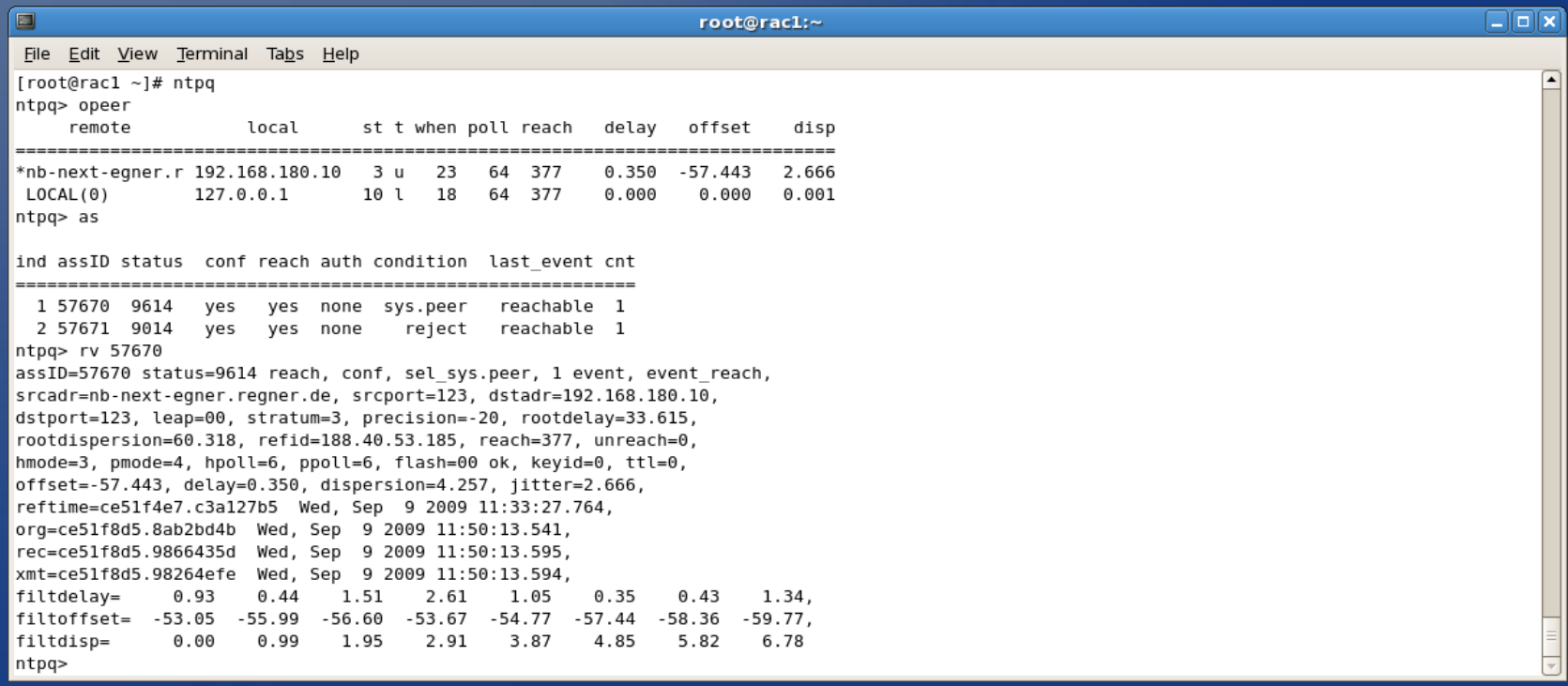
Installation – configure linux and requirements

- Check if NTP is working
  - start „ntpq“
  - enter „opeer“ to see list of all peers
    - In our example two peers: host „nb-next-egner“ and the local clock



Installation – configure linux and requirements

- Check if NTP is working
  - enter „as“ to see associations
    - „sys.peer“ means the clock is synchronized against this; the order in which the entries apper is like „opeer“ - so first entry means host „nb-next-egner“ → fine!
    - reject means not synchronized against due to various reasons
  - enter „rv <assID>“ for detailed information



Installation – configure linux and requirements

- SCAN
  - SCAN = Single Client Access Name; new concept in 11g R2
  - DNS-based
  - nameing notation: <name of cluster>-scan.<domain>

    for our cluster named „rac“ with domain „regner.de“ this will be

    „rac-scan.regner.de“

  - You need at least ONE – better three IPs for the new database access schema called „SCAN“
  - IPs are configured in DNS (forward and reverse lookup);
    !! using local hosts file failed verification after grid installation !!

Installation – configure linux and requirements

- SCAN
  - Name in DNS „<rac-name>-scan.<domain>“
    - for our to be installed infrastructure we choose the name „rac“
    - forward- and reverse lookup needs to be configured
    - so scan name is „rac-scan“
    - excerpt from zone file:

```
rac-scan        IN A        192.168.180.6
rac-scan        IN A        192.168.180.7
rac-scan        IN A        192.168.180.8
```

Installation – configure linux and requirements

- SCAN (con´t)
  - After installation we will find three listeners running from grid infrastructure home:

```
bash# srvctl status scan_listener
SCAN Listener LISTENER_SCAN1 is enabled
SCAN listener LISTENER_SCAN1 is running on node rac1
SCAN Listener LISTENER_SCAN2 is enabled
SCAN listener LISTENER_SCAN2 is running on node rac2
SCAN Listener LISTENER_SCAN3 is enabled
SCAN listener LISTENER_SCAN3 is running on node rac2
```

Installation – configure linux and requirements

- SCAN (con´t)
  - Connection to database „RAC11P“ using SCAN would use this tnsnames entry

```
RAC11P =
(DESCRIPTION=
  (ADDRESS=(PROTOCOL=tcp)(HOST=rac-scan.regner.de)
(PORT=1521))
  (CONNECT_DATA=(SERVICE_NAME=RAC11P))
)
```

  - The „old fashioned“ way still works:

```
RAC11P_old =
(DESCRIPTION=
  (ADDRESS_LIST=
    (ADDRESS=(PROTOCOL=tcp)(HOST=rac1-vip.regner.de)
(PORT=1521))
    (ADDRESS=(PROTOCOL=tcp)(HOST=rac2-vip.regner.de)
(PORT=1521))
  )
  (CONNECT_DATA=(SERVICE_NAME=RAC11P))
)
```

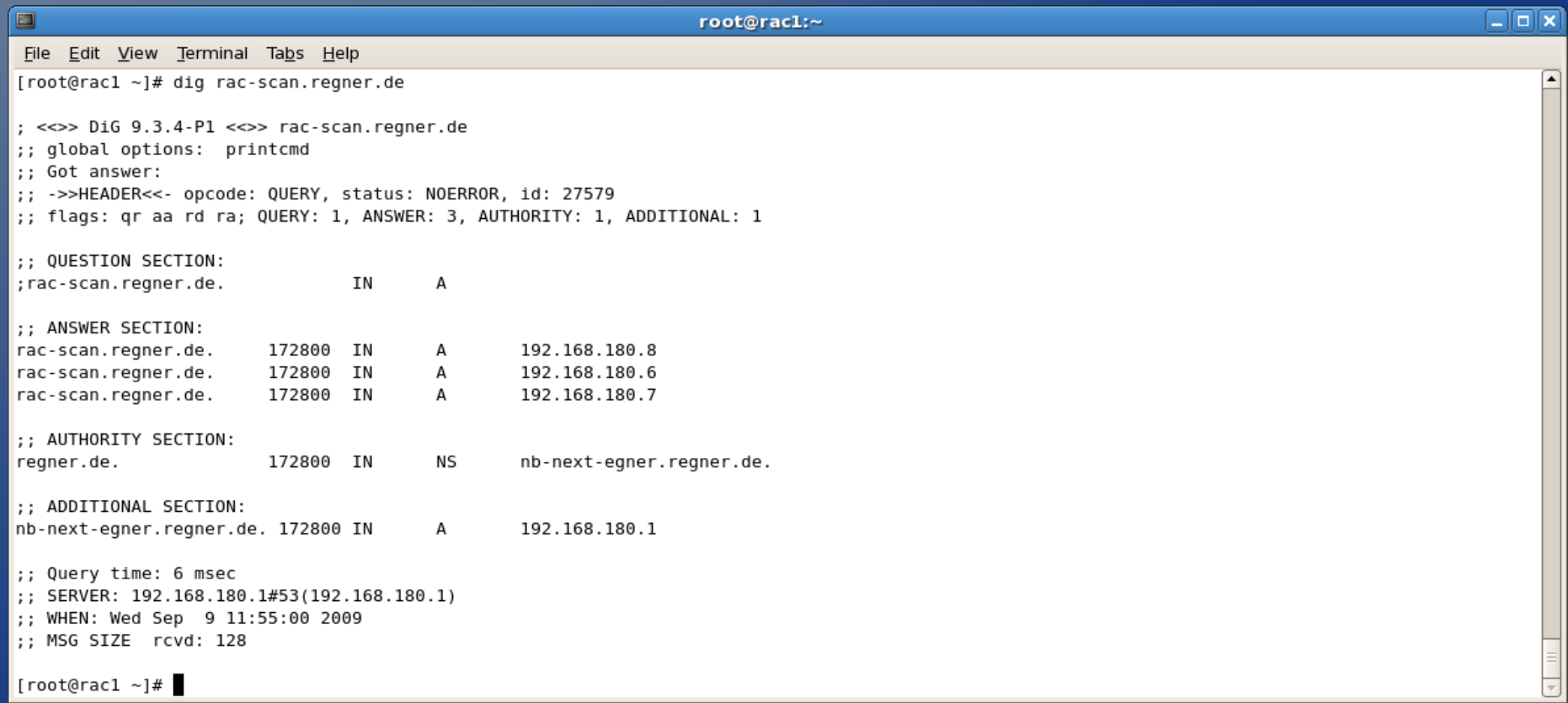
Installation – configure linux and requirements

- SCAN (con´t)
  - Connecting to a named instance:

```
RAC11P =
(DESCRIPTION=
  (ADDRESS=(PROTOCOL=tcp)(HOST=rac-scan.regner.de)
(PORT=1521))
  (CONNECT_DATA=(SERVICE_NAME=RAC11P)
  (INSTANCE_NAME=RAC11P1))
)
```

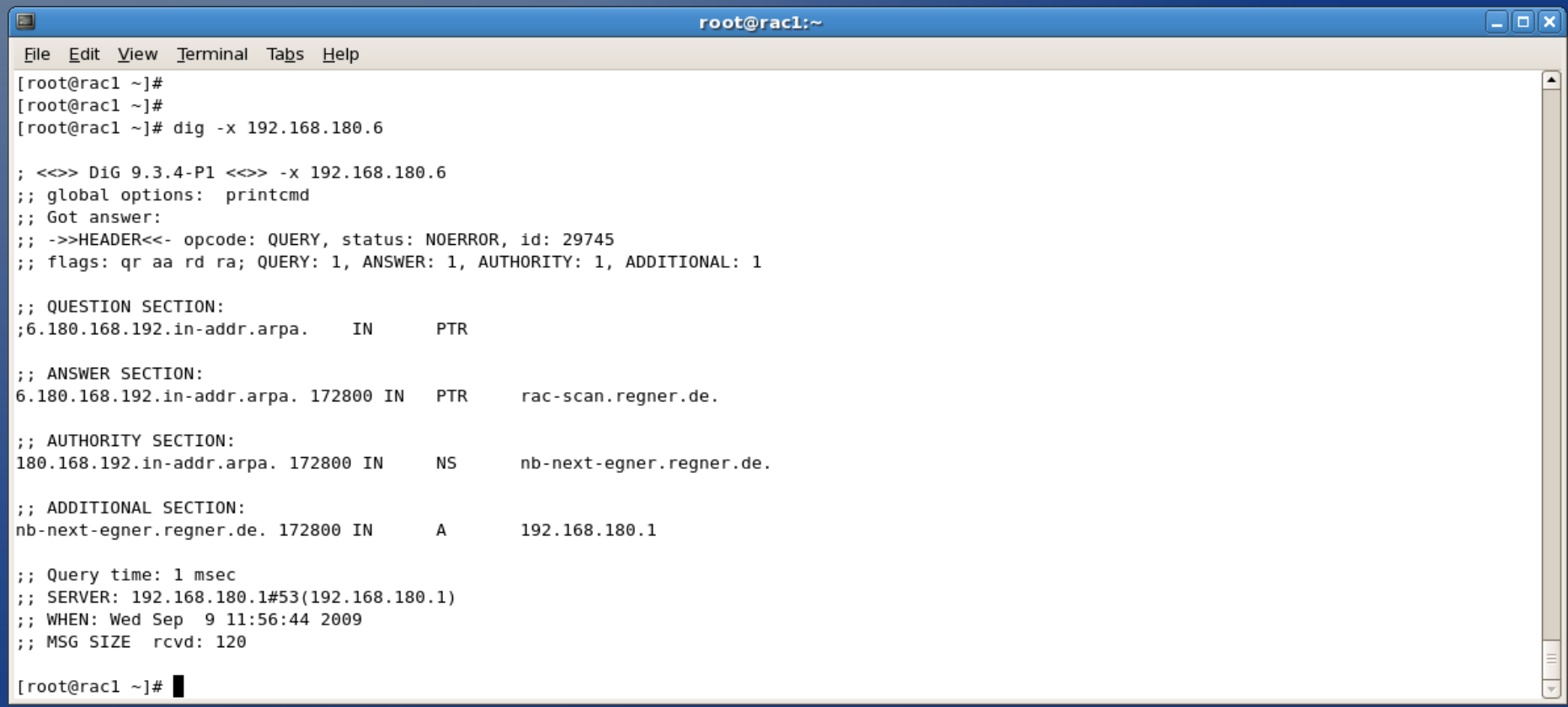
Installation – configure linux and requirements

- Check DNS for SCAN
  - Forward lookup
    - Use „dig“ to check
    - „dig rac-scan.regner.de“



Installation – configure linux and requirements

- Check DNS for SCAN
  - Reverse lookup
    - Use „dig“ to check
    - „dig -x 192.168.180.6“
    - „dig -x 192.168.180.7“
    - „dig -x 192.168.180.8“



Installation – configure linux and requirements

- Create Group

  groupadd -g 500 dba

  Note: For educational purposes we use only one group. In productive enviroments there should be more groups to separate administrative duties.

- Create User and directories

  ```
  mkdir -p /u01/app/11.2.0/grid
  chown -R root:dba /u01
  chmod -R 775 /u01
  chown -R grid:dba /u01/app/11.2.0/grid
  useradd -g dba -u 500 -d /u01/app/11.2.0/grid grid
  passwd grid
  ```

  Note: Oracle recommends different users for grid and database installation!

Make sure groupid and userid are the same on ALL nodes!

Installation – configure linux and requirements

- Create profile file (~/.bash_profile or ~/.profile on SuSE) for user „grid“

```
umask 022
if [ -t 0 ]; then
        stty intr ^C
fi
```

Installation – configure storage

- Prepare Storage – Requirements
  - must be visible on all nodes
  - as always - recommendation: SAME (stripe and mirror everything)

- Storage – what to store where:
  - OCR and Voting disk
    - ASM
    - NFS
    - RAW disks
  - Oracle Clusterware binaries
    - NFS
    - Local disk

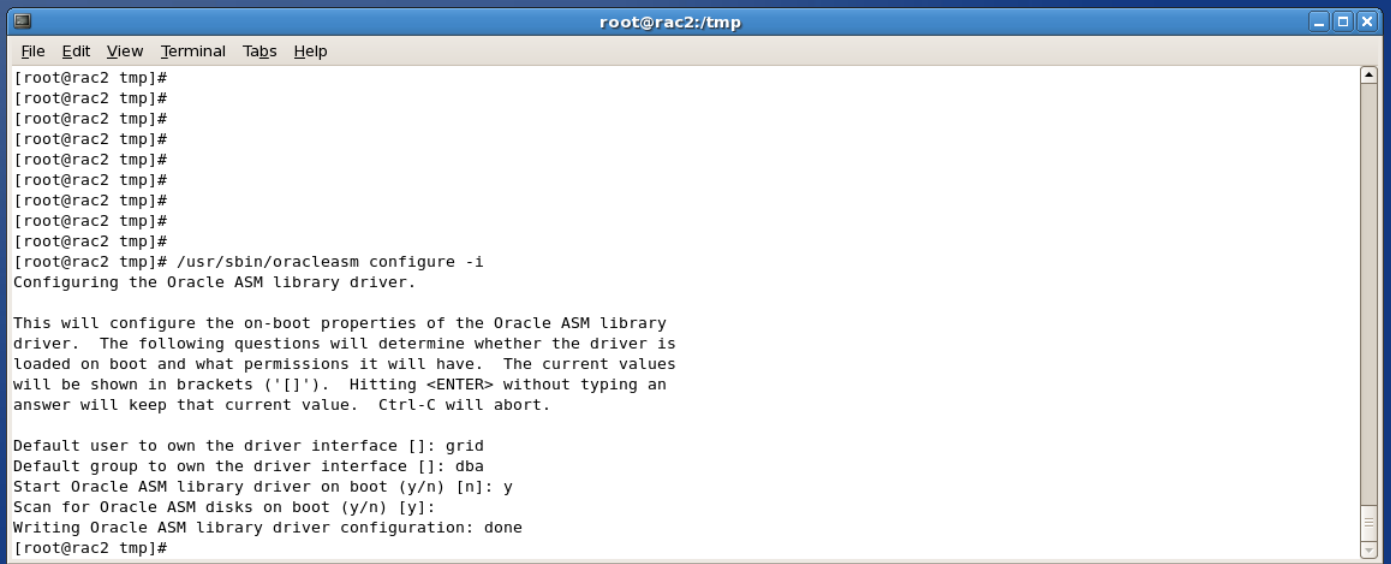
Installation – configure storage

- Storage – what to store where (con´t):
  - Oracle RAC binaries
    - ACFS
    - NFS
    - local disk
  - Oracle database files
    - ASM
    - NFS
    - RAW disks
  - Oracle recovery files
    - ASM
    - NFS

Installation – configure storage

- Prepare and Configure Storage (for use with asm)
  - Install RPMs
    - oracleasmsupport
    - oracleasmlib
    - oracleasm-<kernel-version>
      (see „Sources“ for download location“)
  - Configure ASM

    /usr/sbin/oracleasm configure -i



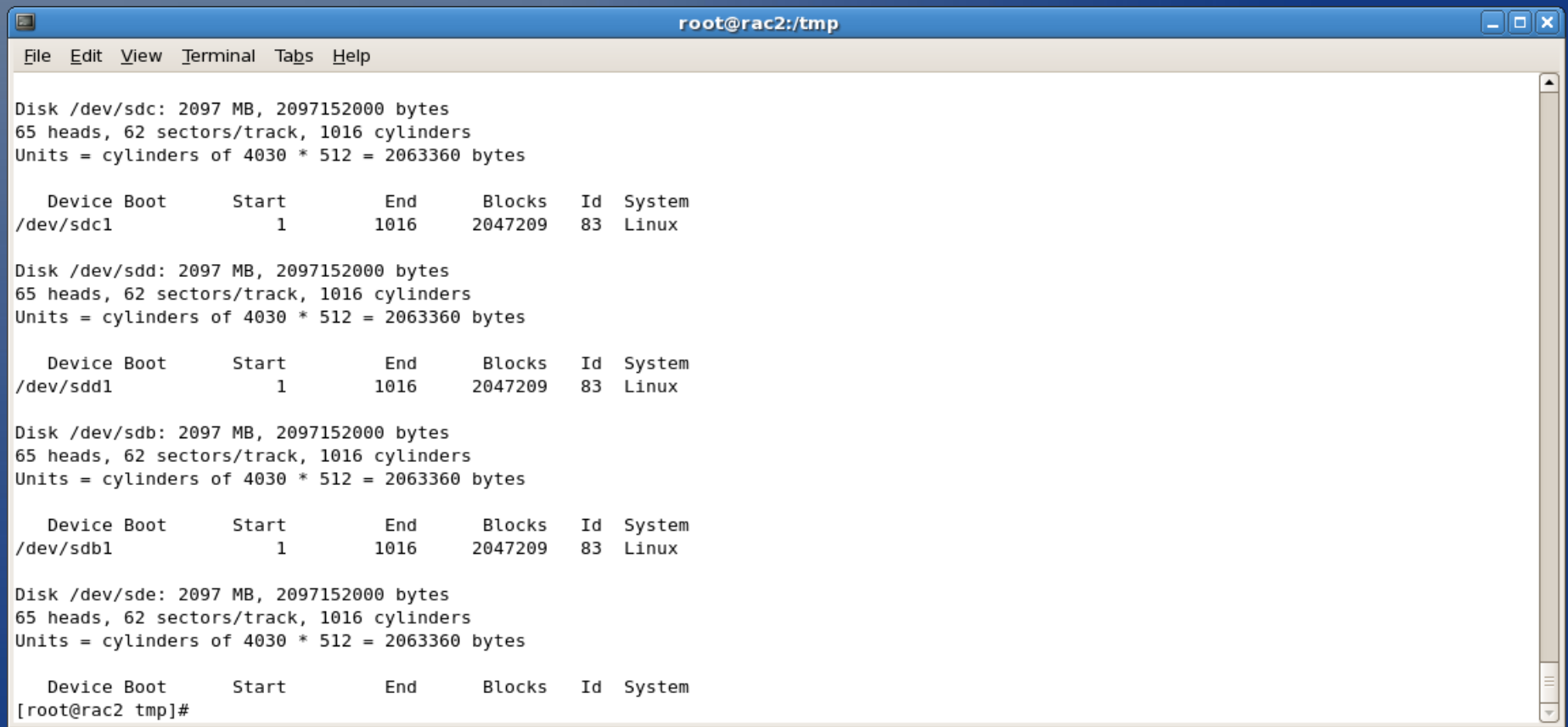
Installation – configure storage

- Prepare and Configure Storage (for use with asm)
  - init ASM

```
/usr/sbin/oracleasm init
```

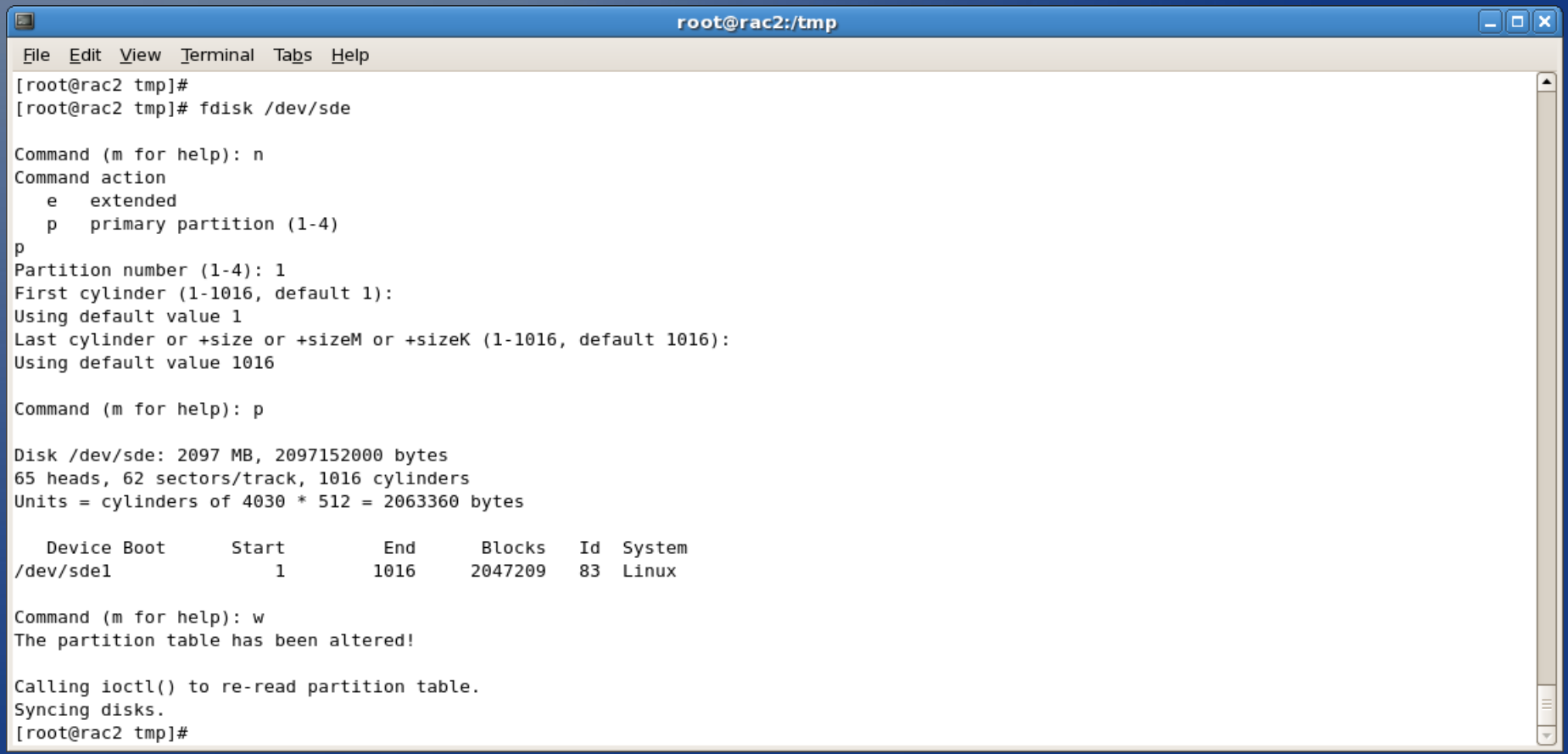
Installation – configure storage

- Prepare and Configure Storage (for use with asm)
  - Create Partitions on disk with fdisk
    - Query all available disks with „fdisk -l“
    - In the following example disk /dev/sde (this is our iSCSI storage) does not contain a partition at all – we will create one



Installation – configure linux operating system

- Prepare and Configure Storage (for use with asm)
  - Create Partitions on disk with fdisk
    - Create one whole disk partition on /dev/sde

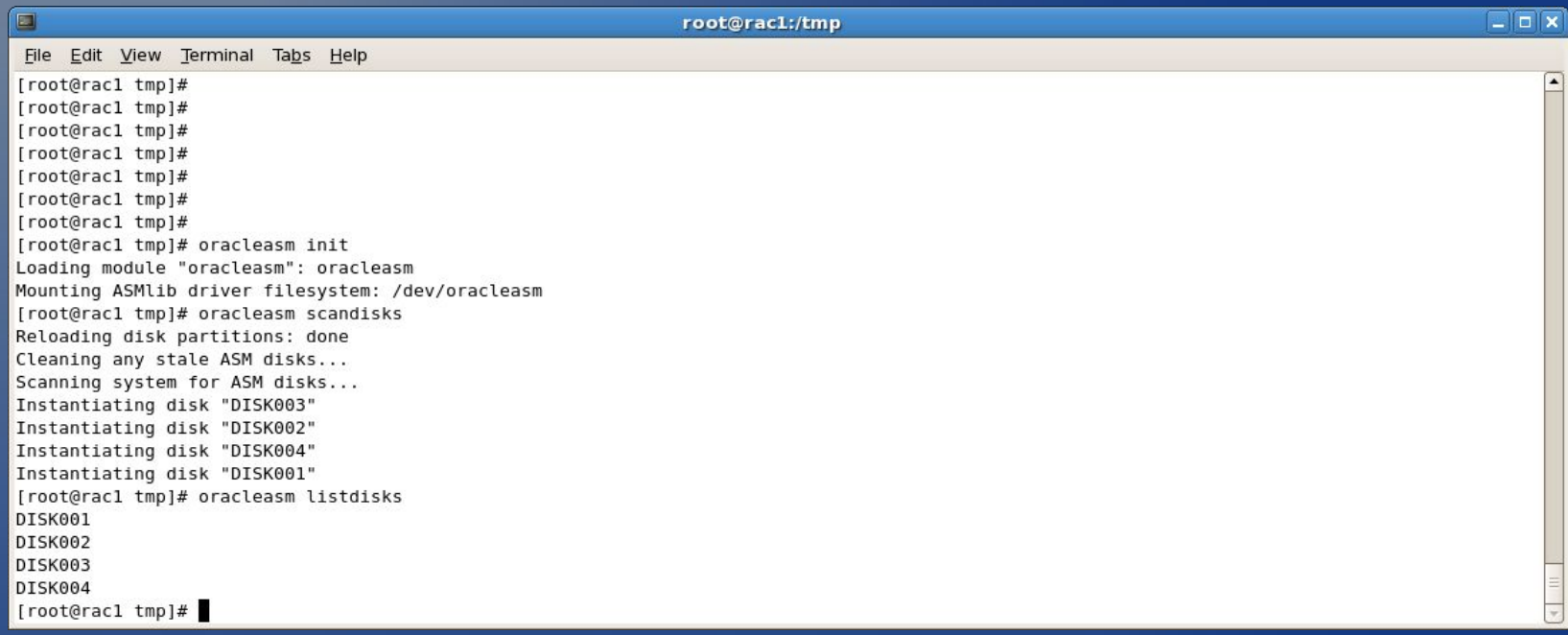


Installation – configure storage

- Prepare and configure Storage (for use with asm)
  - Label all disks with asm label

Installation – configure storage

- Prepare and configure Storage (for use with asm)
  - Query disks on all nodes
    - Node „rac1“



→ all disks visible with correct label

Installation – configure storage

- Prepare and configure Storage (for use with asm)
  - Query disks on all nodes
    - Node „rac2“ (the second node)

→ also all four LUNs visible

Installation – configure storage

- Prepare and configure Storage (for use with asm)
  - OCR and Voting disks
    - Will be placed in ASM (new in 11g R2)
    - three different redundancy levels:
      - External - 1 disk minimum needed
      - Normal - 3 disks minumum needed
      - High - 5 disks minimum needed
    - Storage Requirments
      - External - 280 MB OCR + 280 MB Voting Disk
      - Normal - 560 MB OCR + 840 MB Voting Disk
      - High - 840 MB OCR + 1,4 GB Voting Disk
      - plus Overhead for ASM Metadata

Installation – configure storage

- Prepare and configure Storage (for use with asm)
  - Overhead for ASM metadata

```
total =
   [2 * ausize * disks]
+ [redundancy * (ausize * (nodes * (clients + 1) + 30) + (64 * nodes) + 533)]

redundancy = Number of mirrors: external = 1, normal = 2, high = 3.
ausize = Metadata AU size in megabytes.
nodes = Number of nodes in cluster.
clients - Number of database instances for each node.
disks - Number of disks in disk group.
```

  - For example, for a four-node Oracle RAC installation, using three disks in a normal redundancy disk group, you require 1684 MB of space for ASM metadata

```
   [2 * 1 * 3]
+ [2 * (1 * (4 * (4 + 1)+ 30)+ (64 * 4)+ 533)]
= 1684 MB
```

Installation – configure storage

- Prepare and configure Storage (for use with asm)
  - OCR and Voting disks – recommendations
    - use high redundancy for OCR and Voting disks - the correct function of your cluster depends on it!
    - use 5 disks with 10 GB each – enough space for all files plus asm metadata plus space for futher growth

Installation – binary installation of grid infra

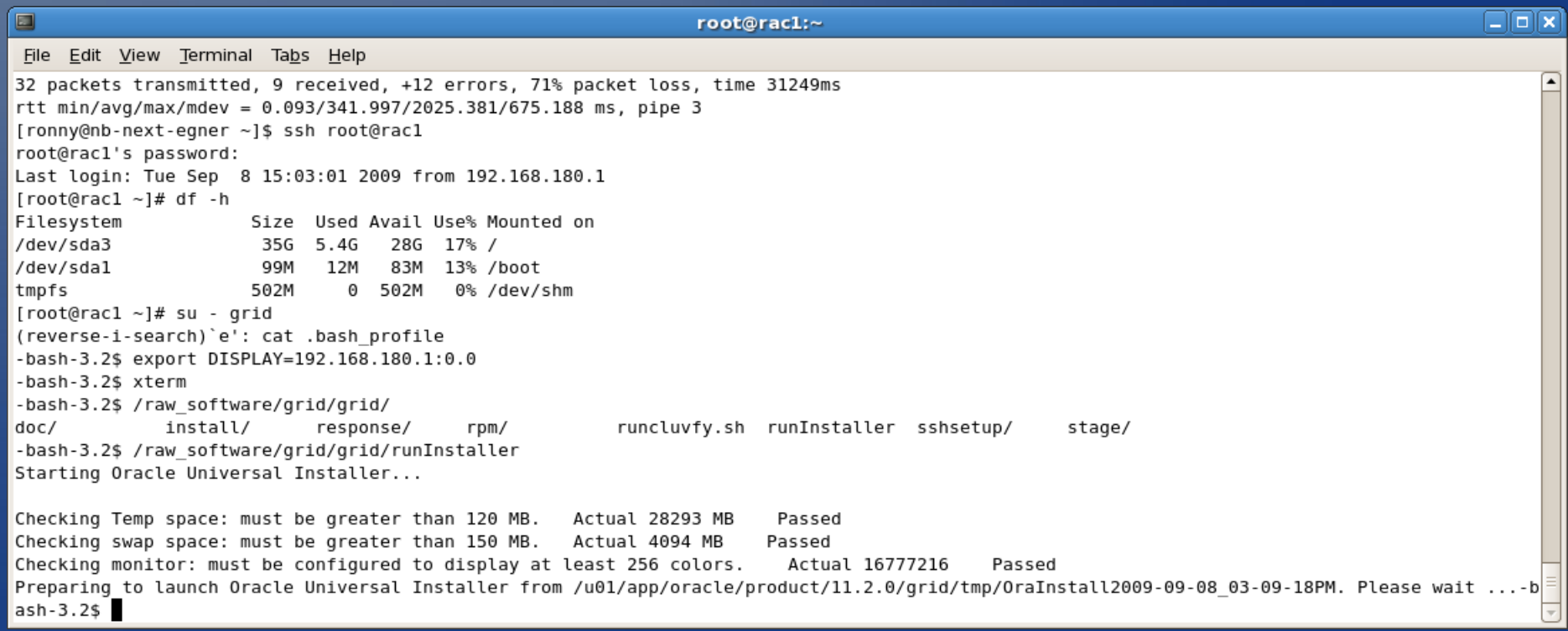
- Checklist
  - Storage visible ✔
  - user and groups created ✔
  - Kernel parameters configured ✔
  - RPM Packages checked / installed ✔
  - NTP working ✔
  - DNS working ✔
  - Connection (ping, ssh) between nodes working? ✔

  - Backup available for rollback? ✔

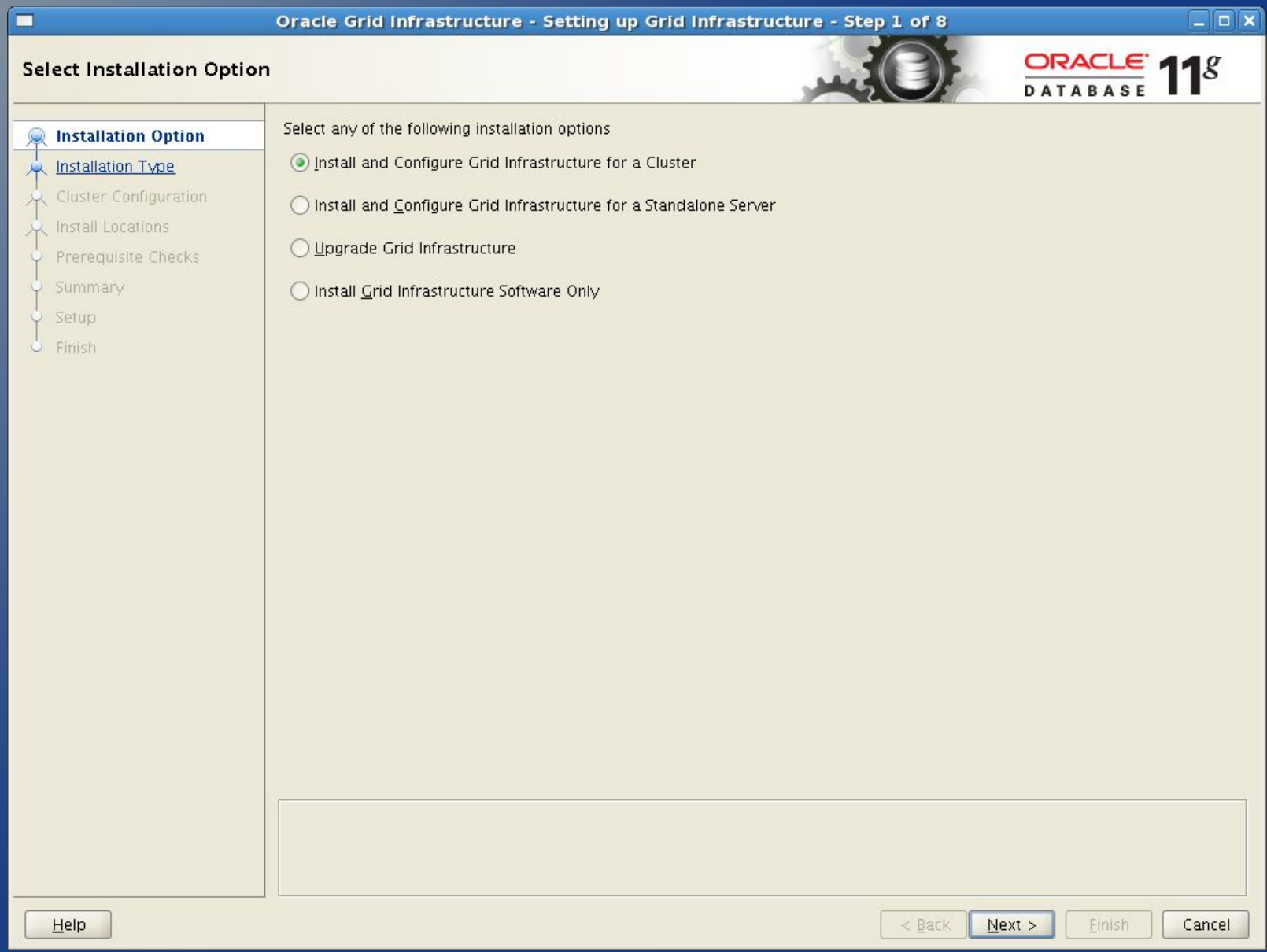
  - Alright! Lets start binary installation

Installation – binary installation of grid infra

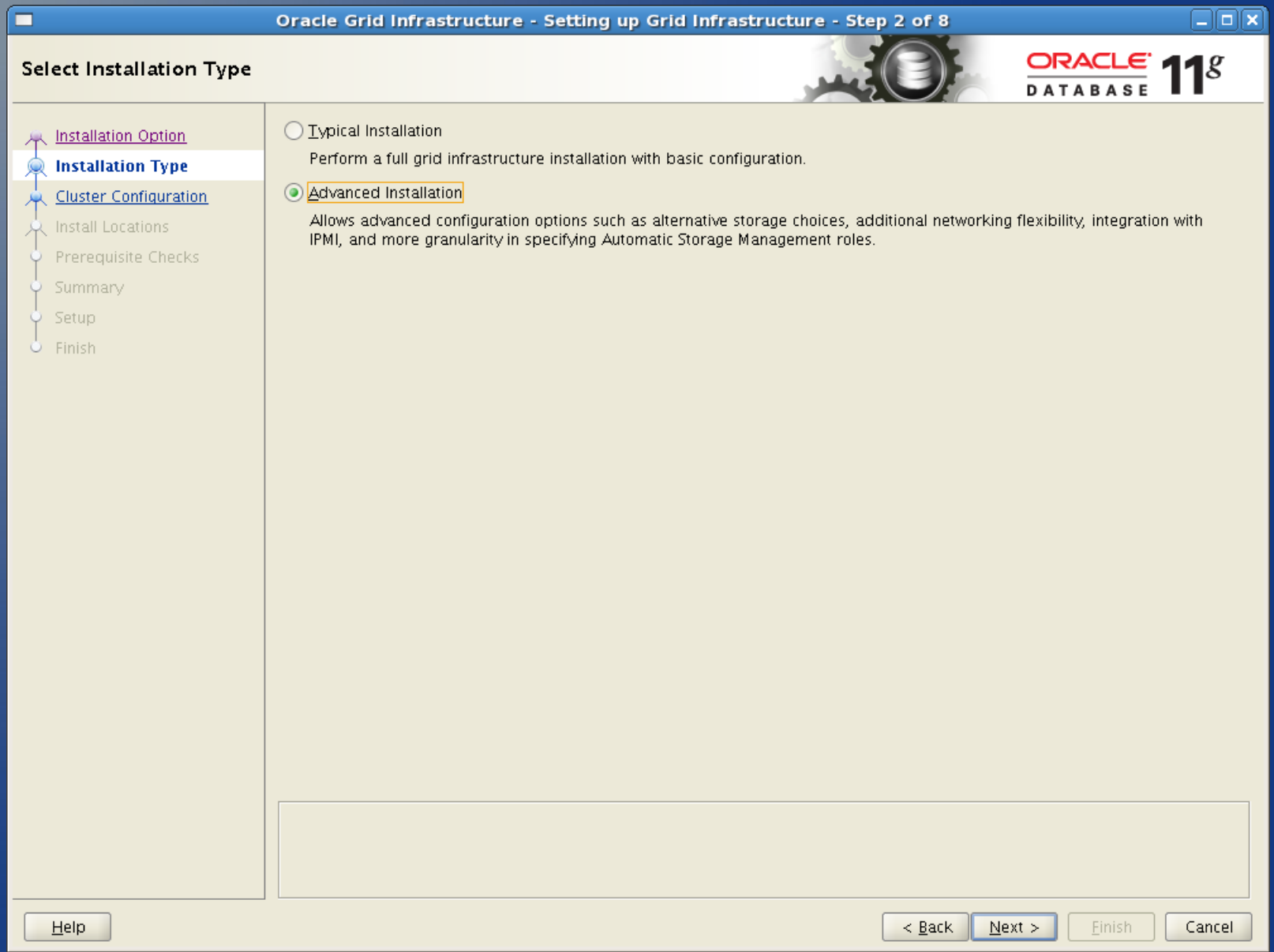
- Start installation as user „grid“ on one node (here on node „rac1“)



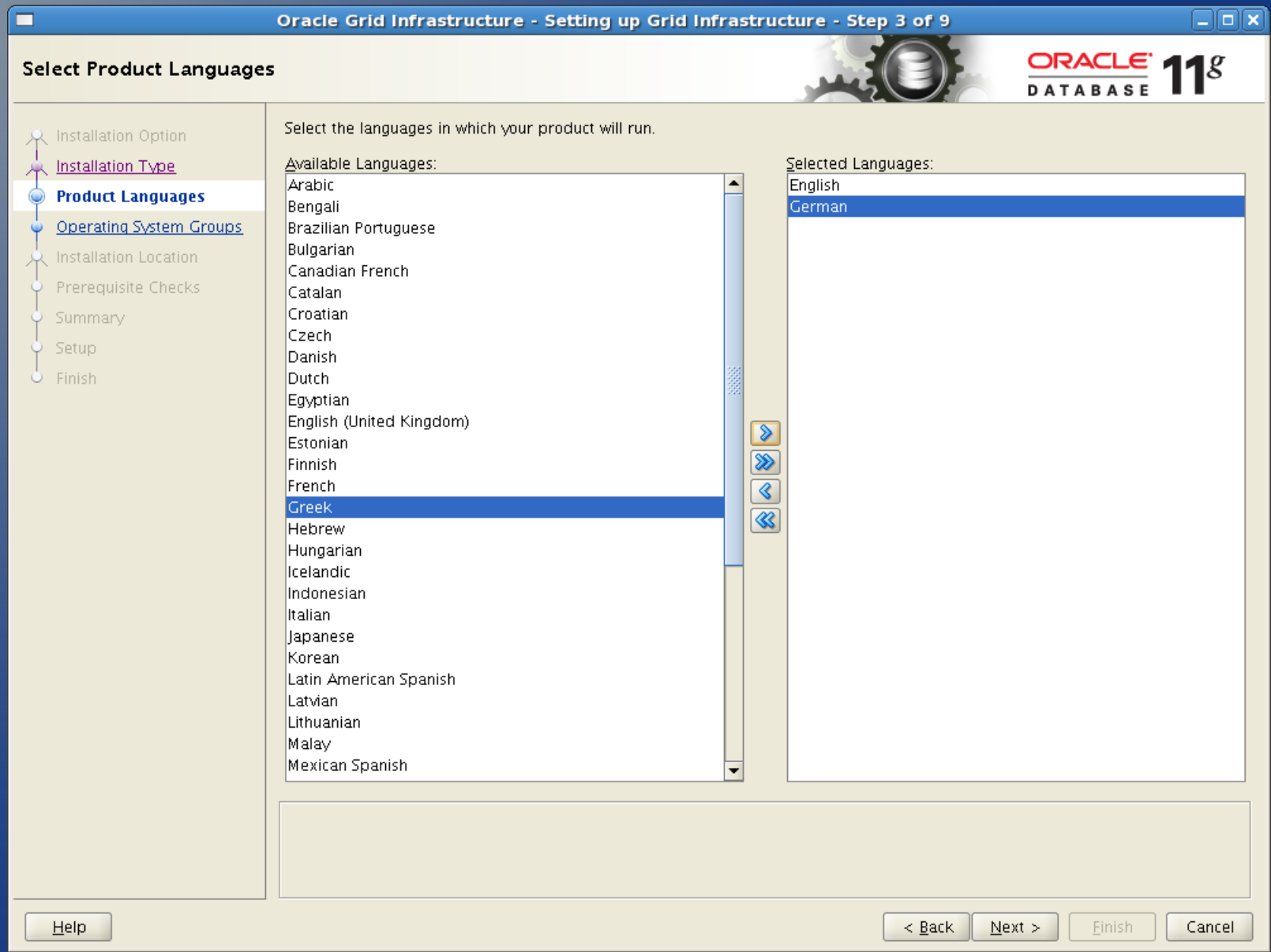
Installation – binary installation of grid infra



Installation – binary installation of grid infra

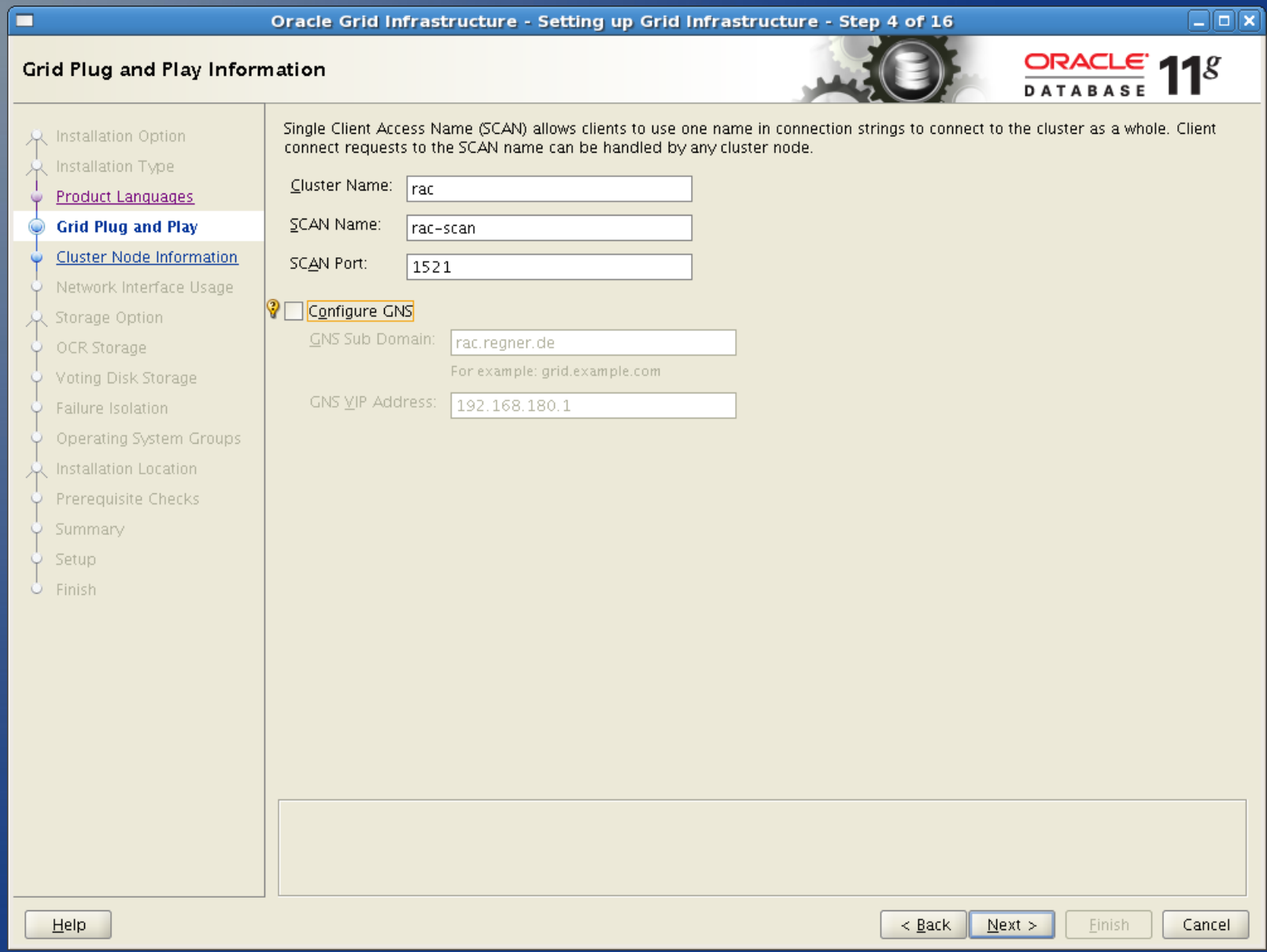


Installation – binary installation of grid infra



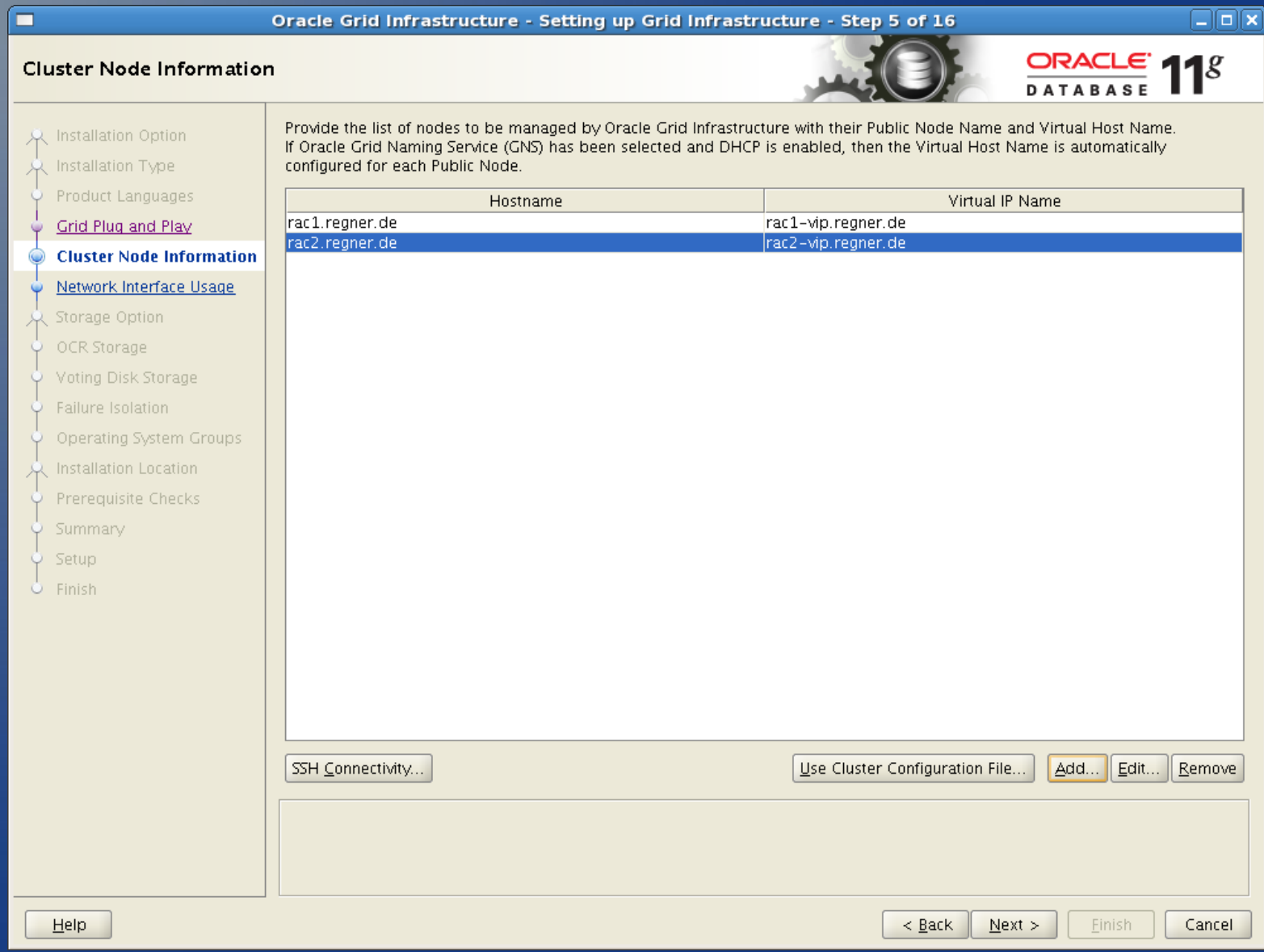
Installation – binary installation of grid infra

- Remember: We choose not to use GNS; so it is deselected



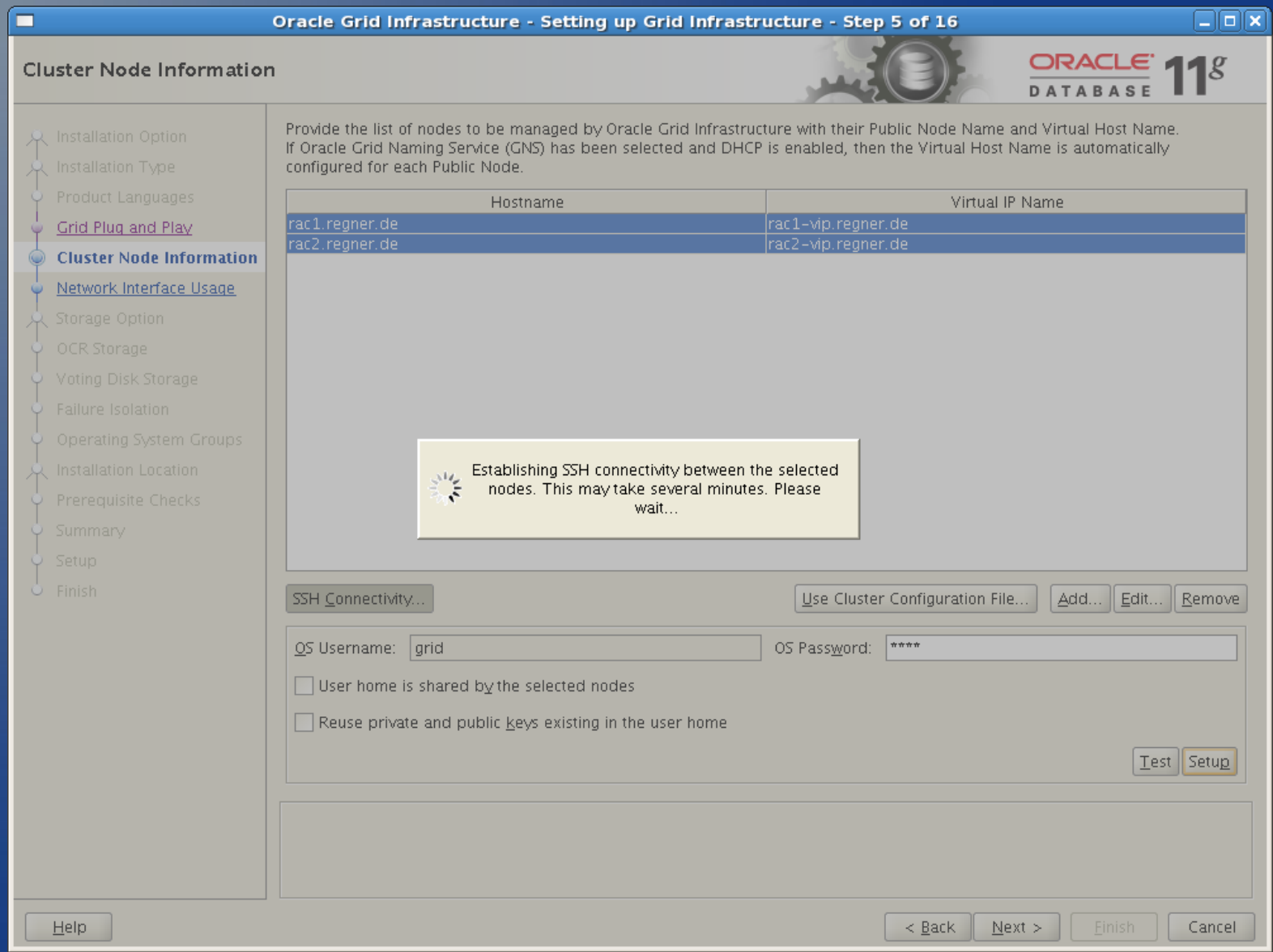
Installation – binary installation of grid infra

- the node the installer was started is already added by default; add here all other nodes (in our case we added „rac2“)



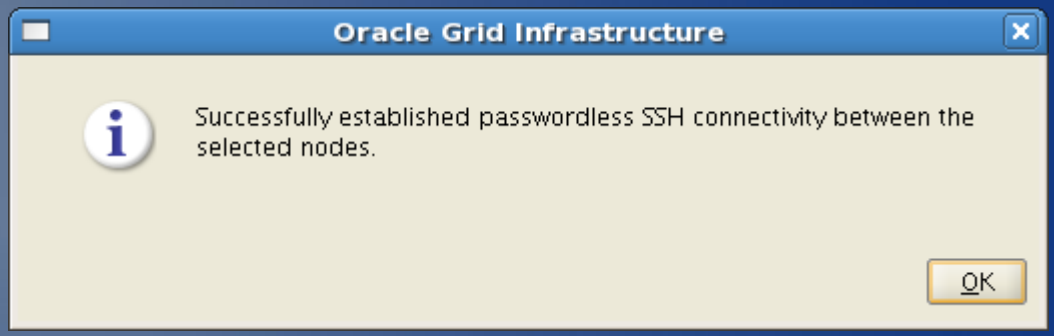
Installation – binary installation of grid infra

- Click on „SSH Connectivity“, enter username and password and click on „Setup“



Installation – binary installation of grid infra

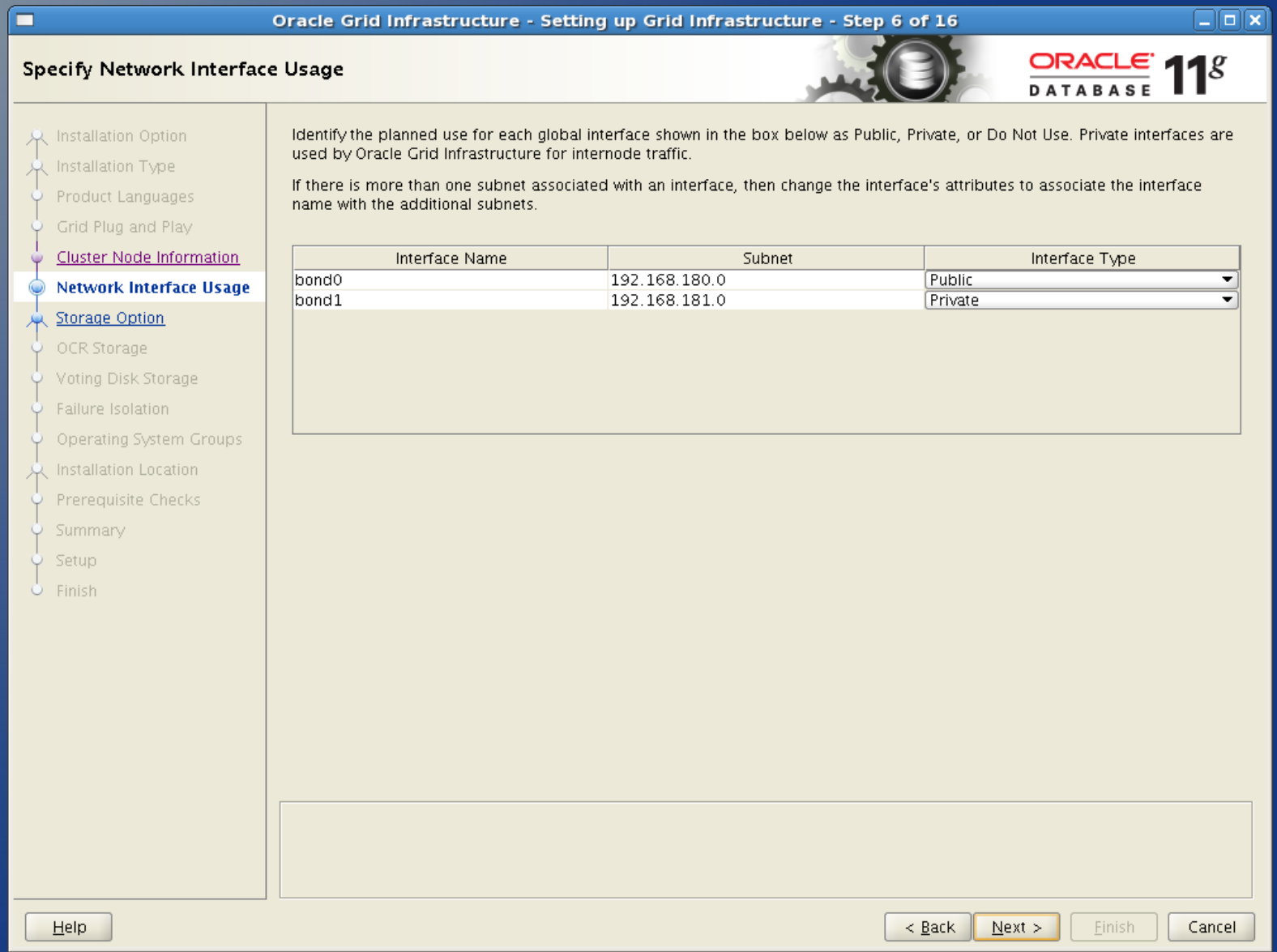
- If everything worked the following message appears



- If there are problems check:
  - Group ID and User ID on both nodes
  - Connectivity between both nodes
  - Passwords

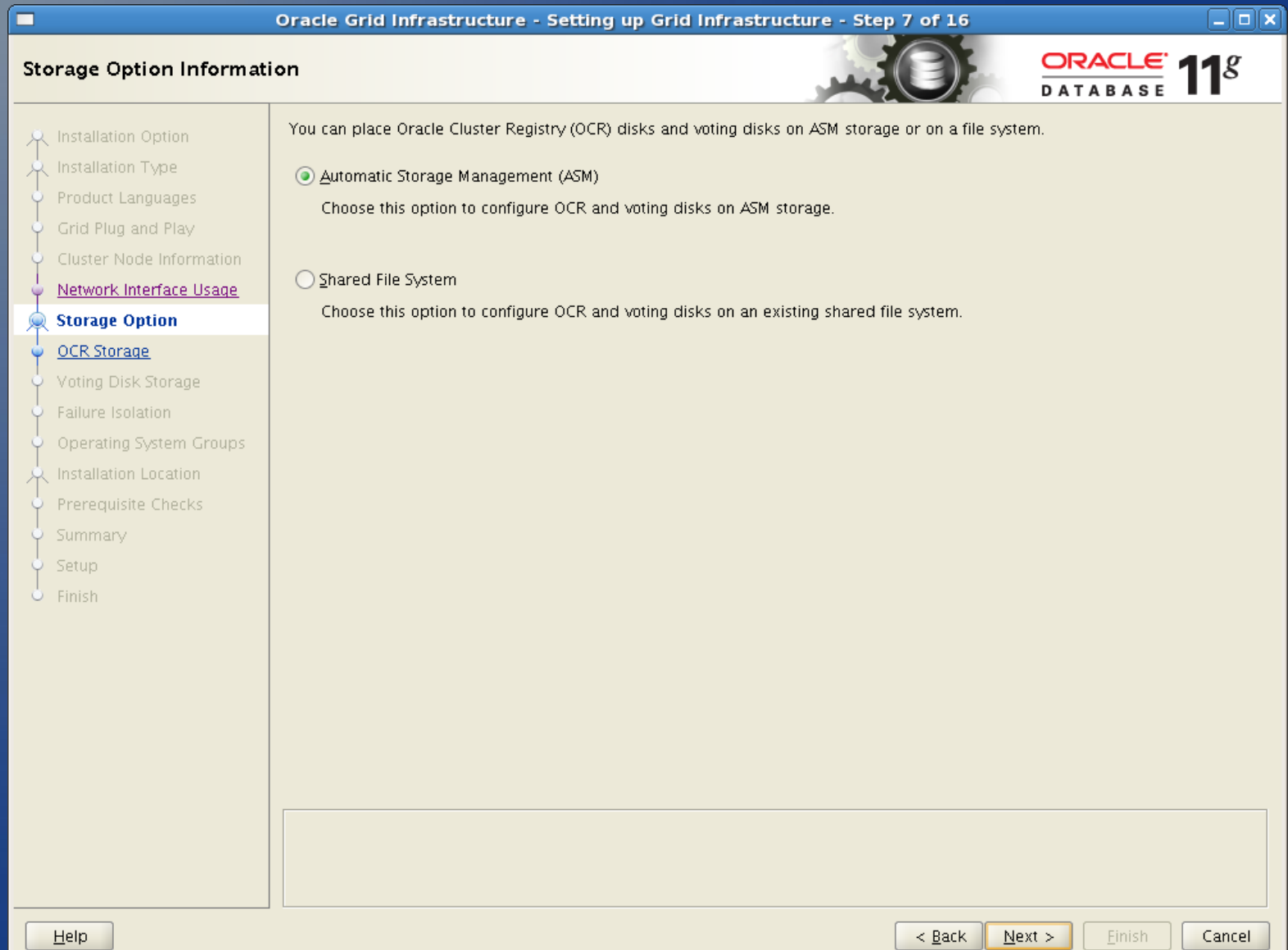
Installation – binary installation of grid infra

- Select which interface is the public and which the private one



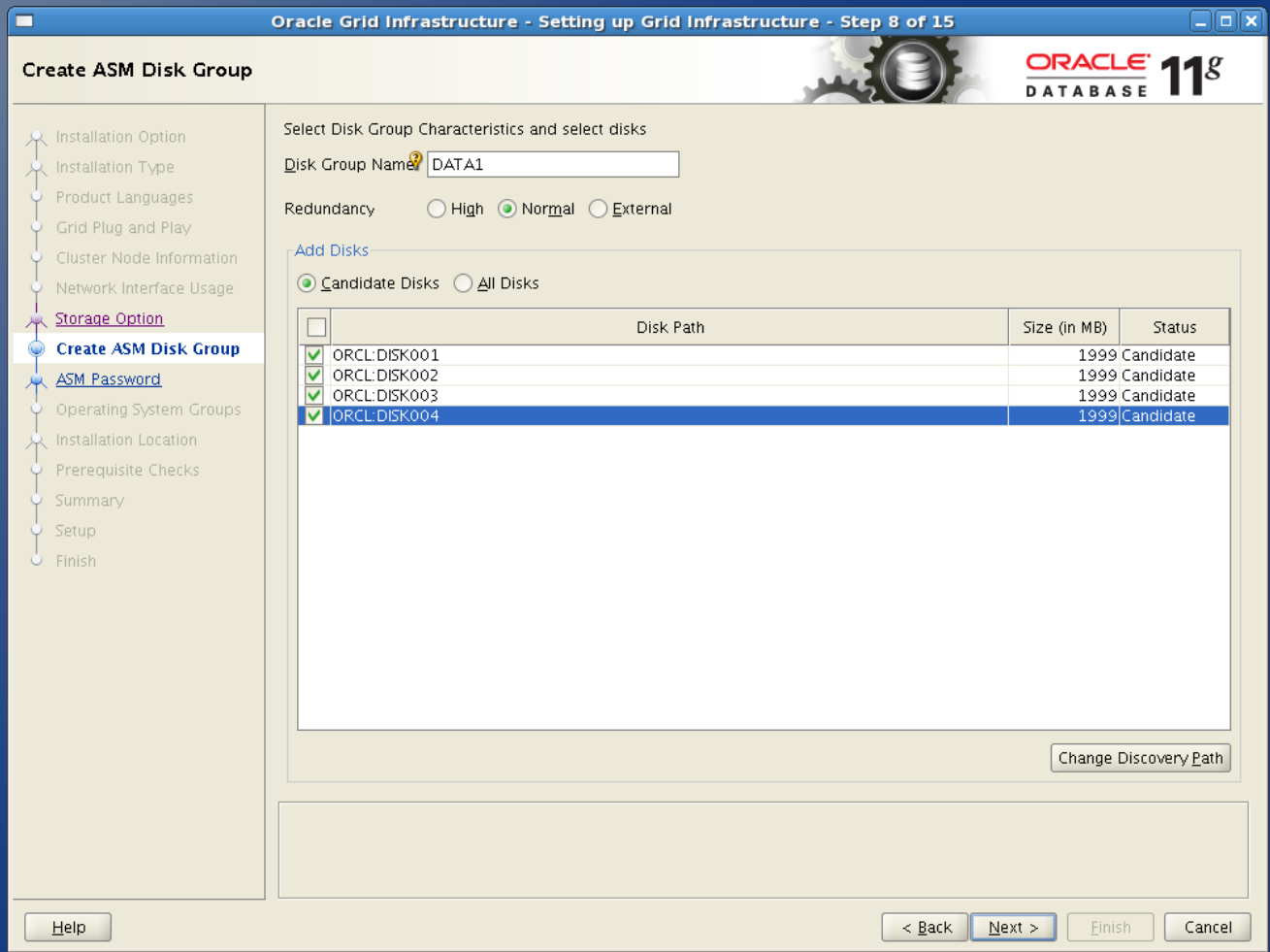
Installation – binary installation of grid infra

- Where to place OCR and Voting disk... in our case we use ASM for everything



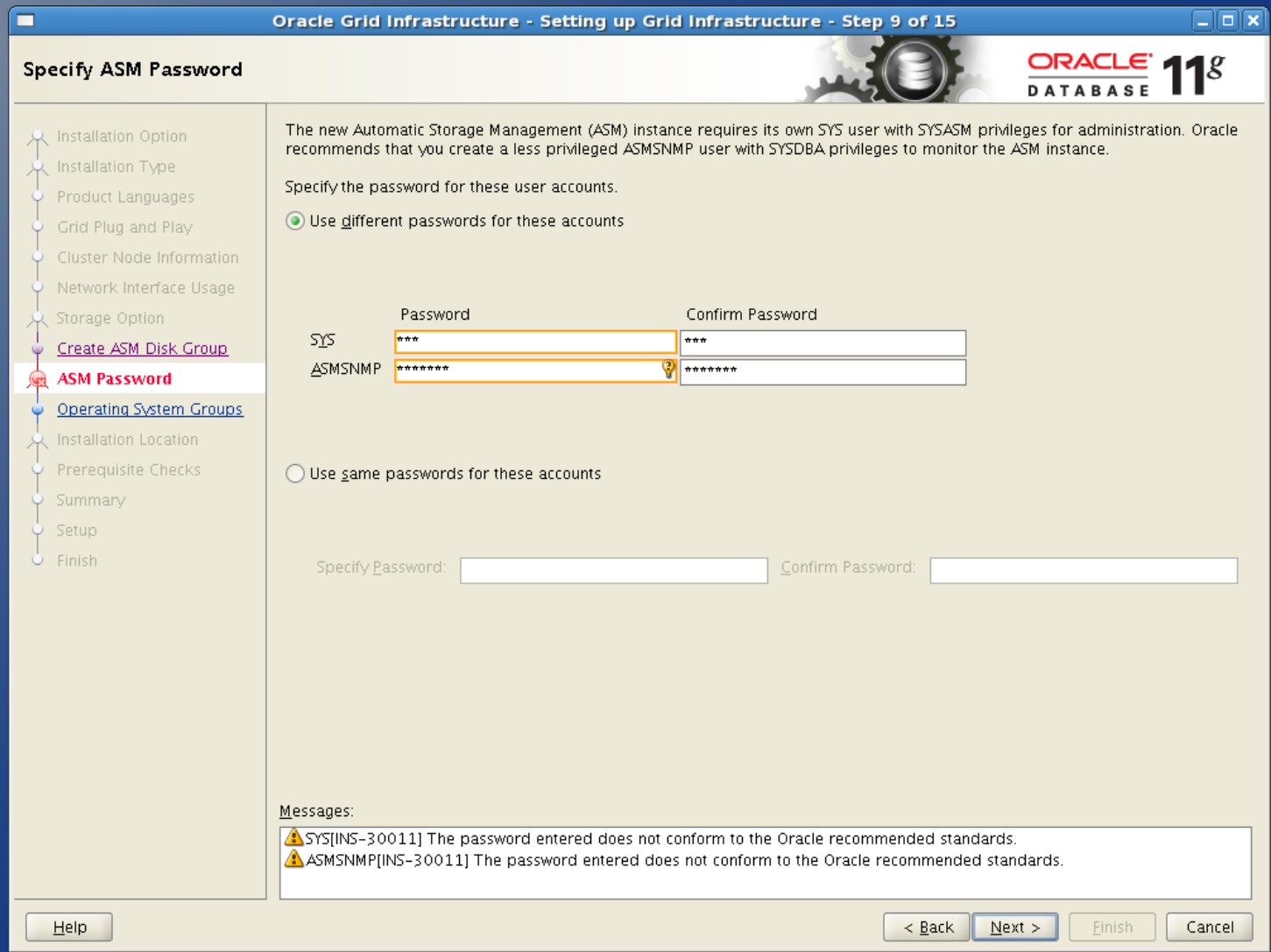
Installation – binary installation of grid infra

- For storing OCR and Voting disk we need to create a data group; our first data group is called „DATA1“ and consists of the four LUNs we prepared and labeled before... here we see the disk names we labeled the disks with again
- We choose „normal“ redundancy which will create a mirror



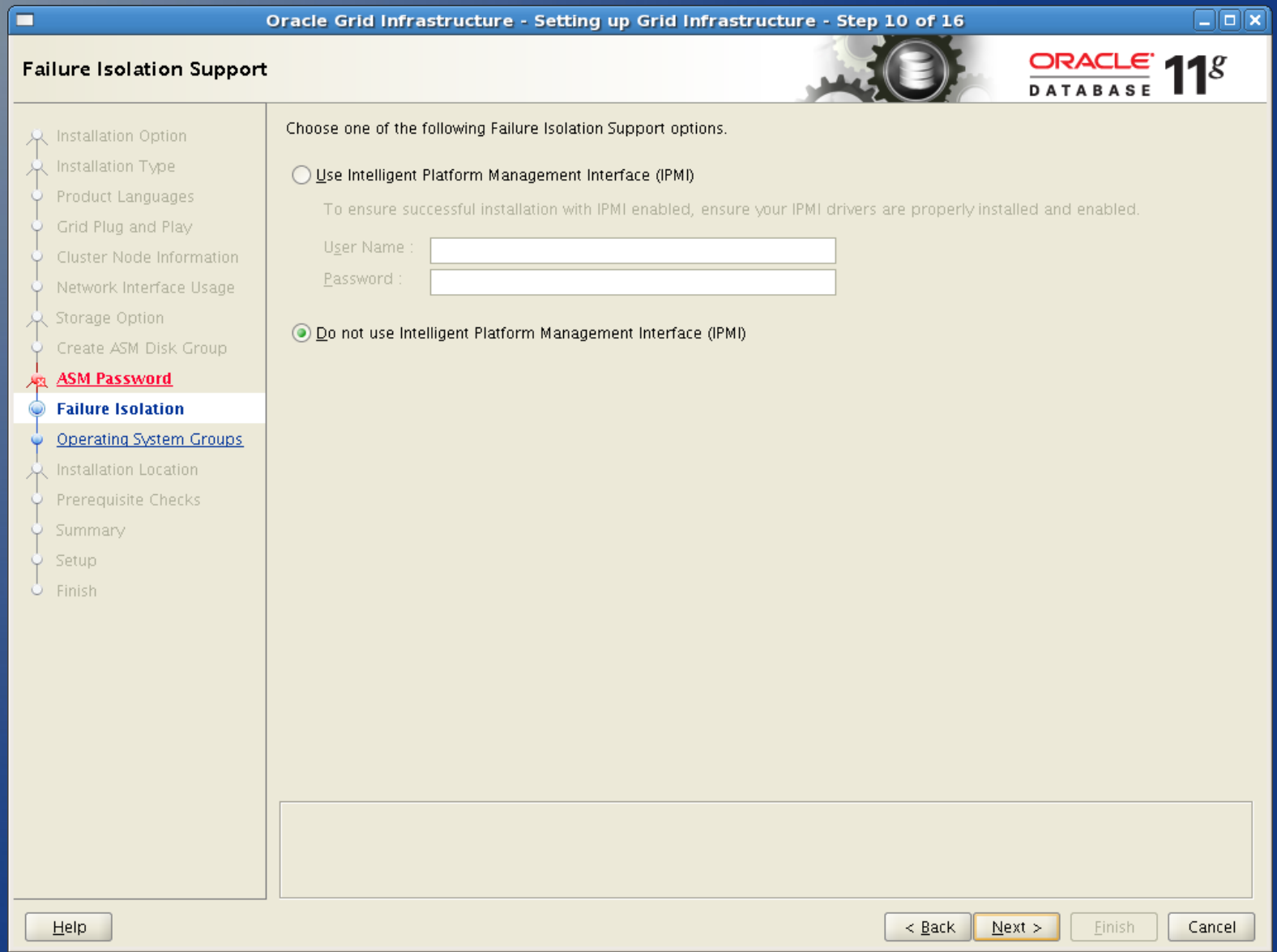
Installation – binary installation of grid infra

- specify passwords for ASM and ASMSNMP.. choose strong passwords if possible (i was lazy and chose not that strong ones – acceptable for educational purposes but not in real productive scenarios)



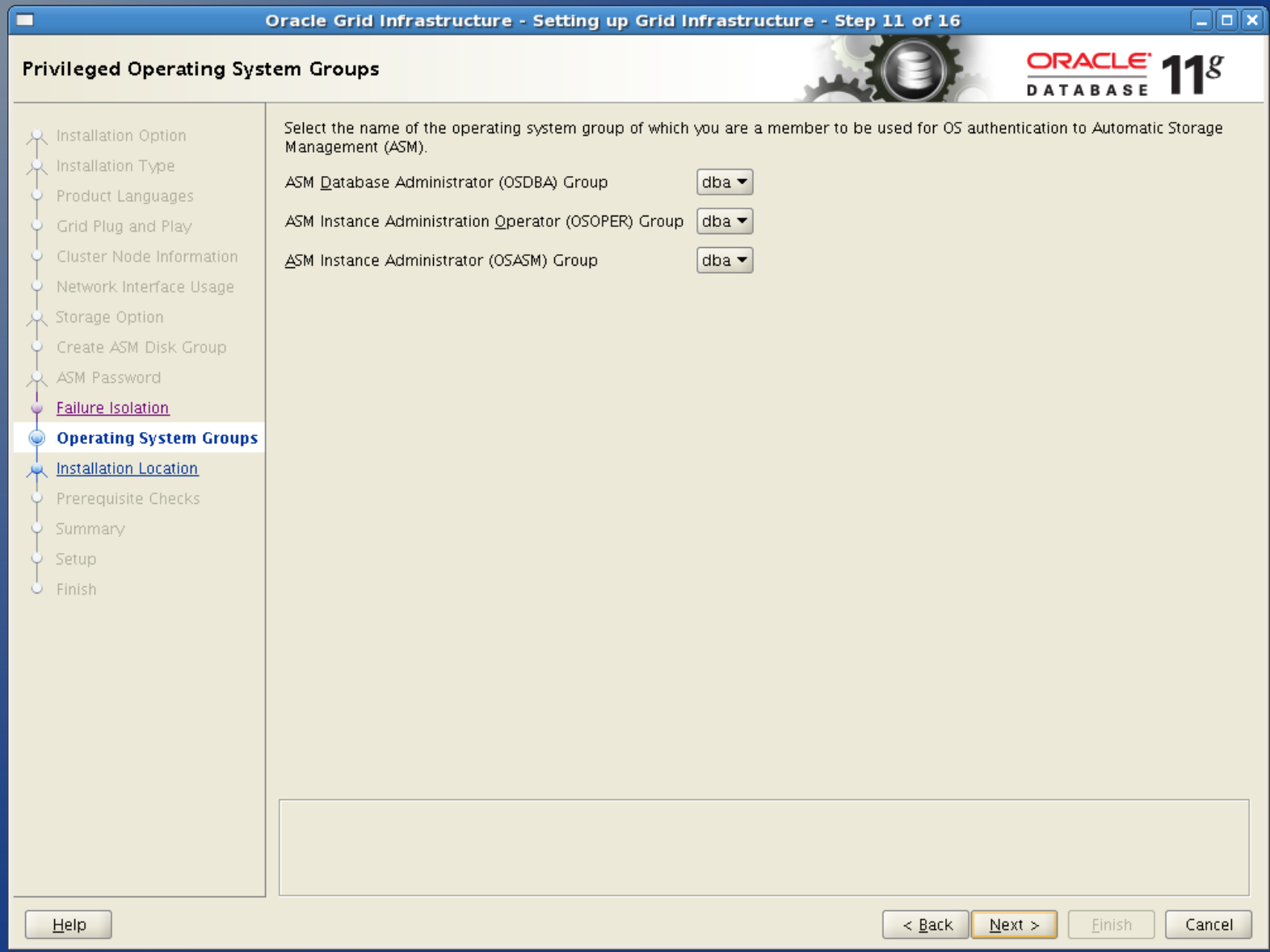
Installation – binary installation of grid infra

- Grid Infrastructure can use IPMI for fencing... VMWARE does not have IPMI



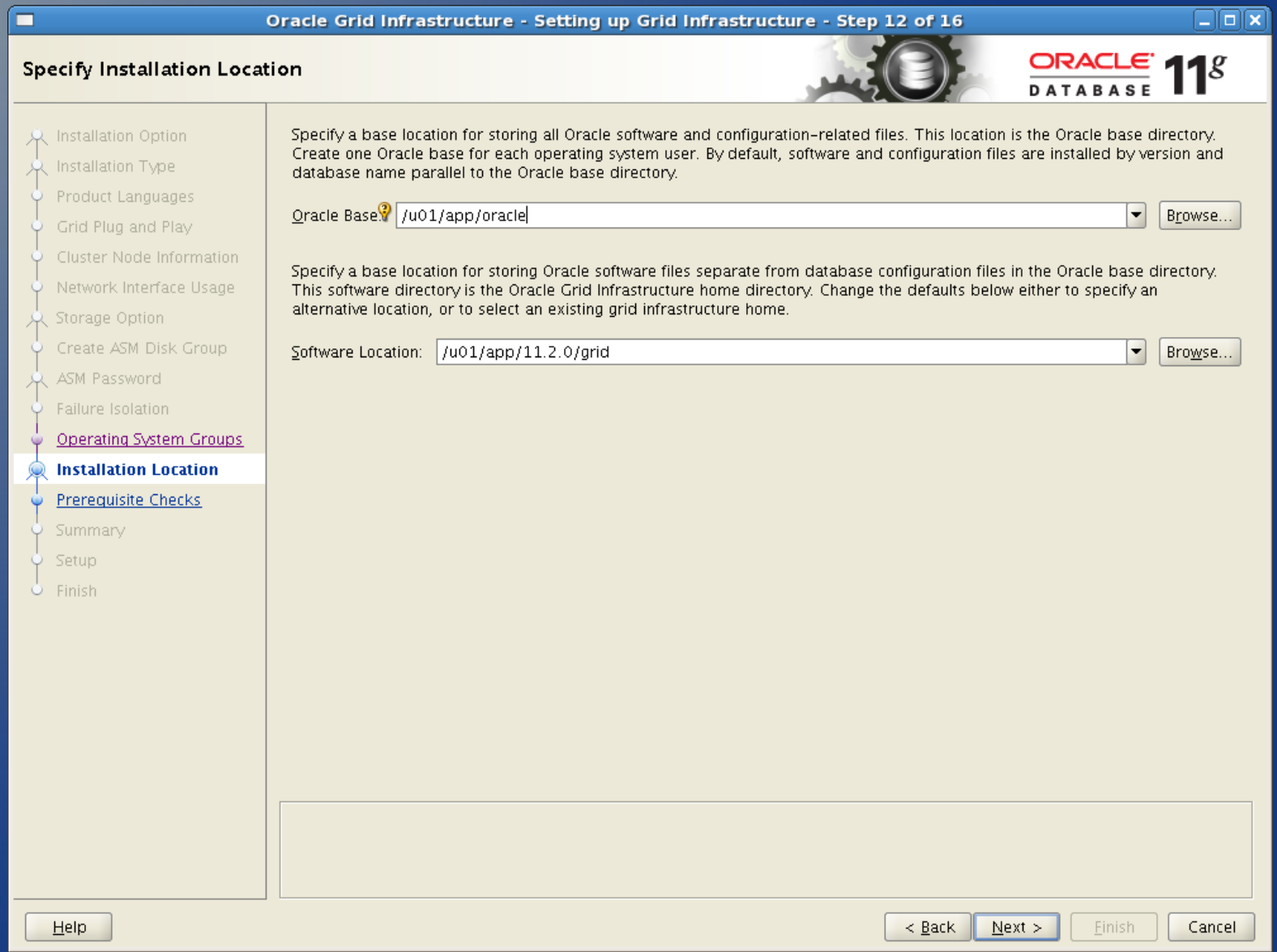
Installation – binary installation of grid infra

- group mapping...for role separation... we have only „dba“ … change accordingly to your needs



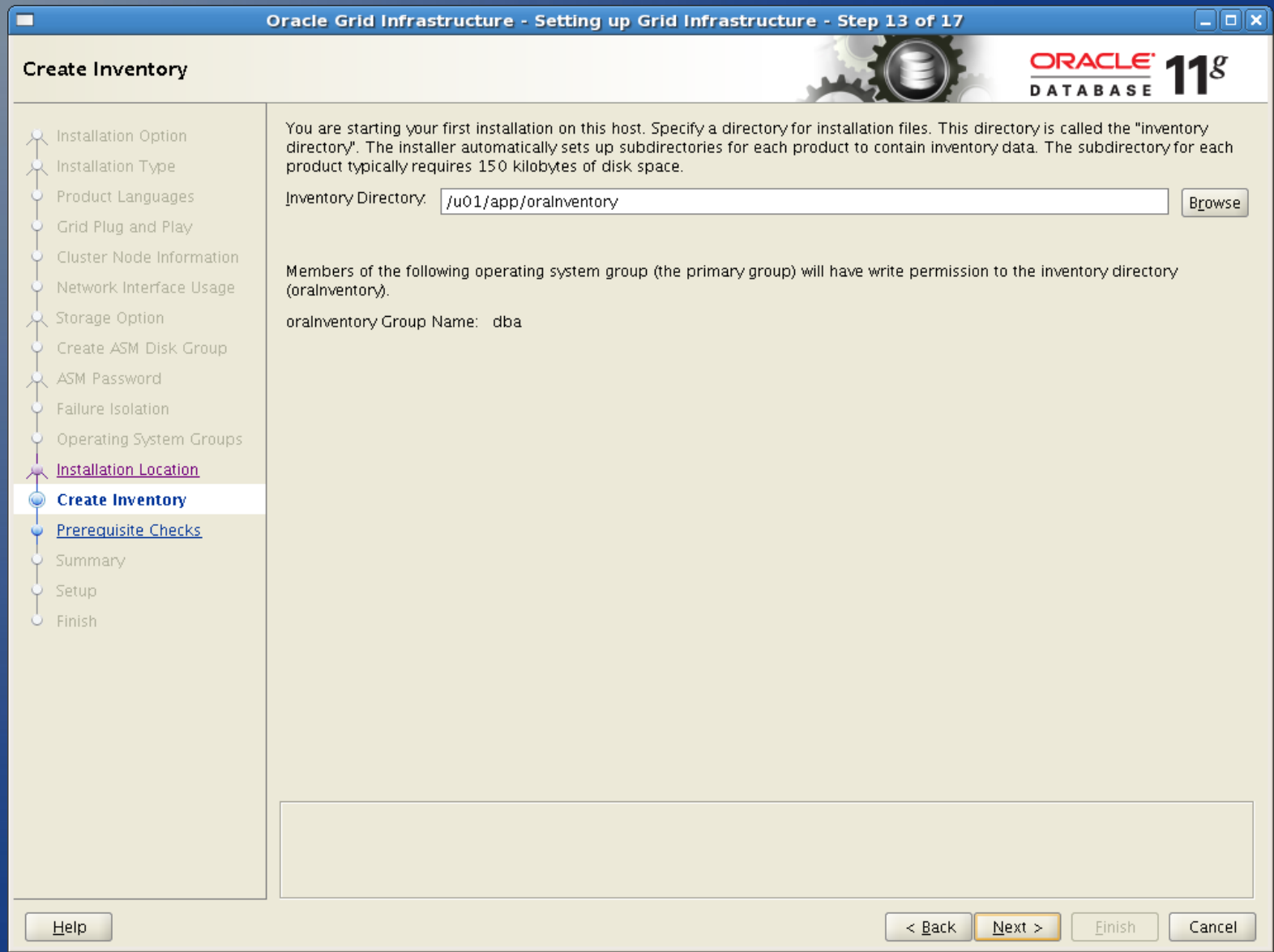
Installation – binary installation of grid infra

- Set Oracle Base and software (install) location... software location must not be under oracle base location... else the installer throws an error saying so



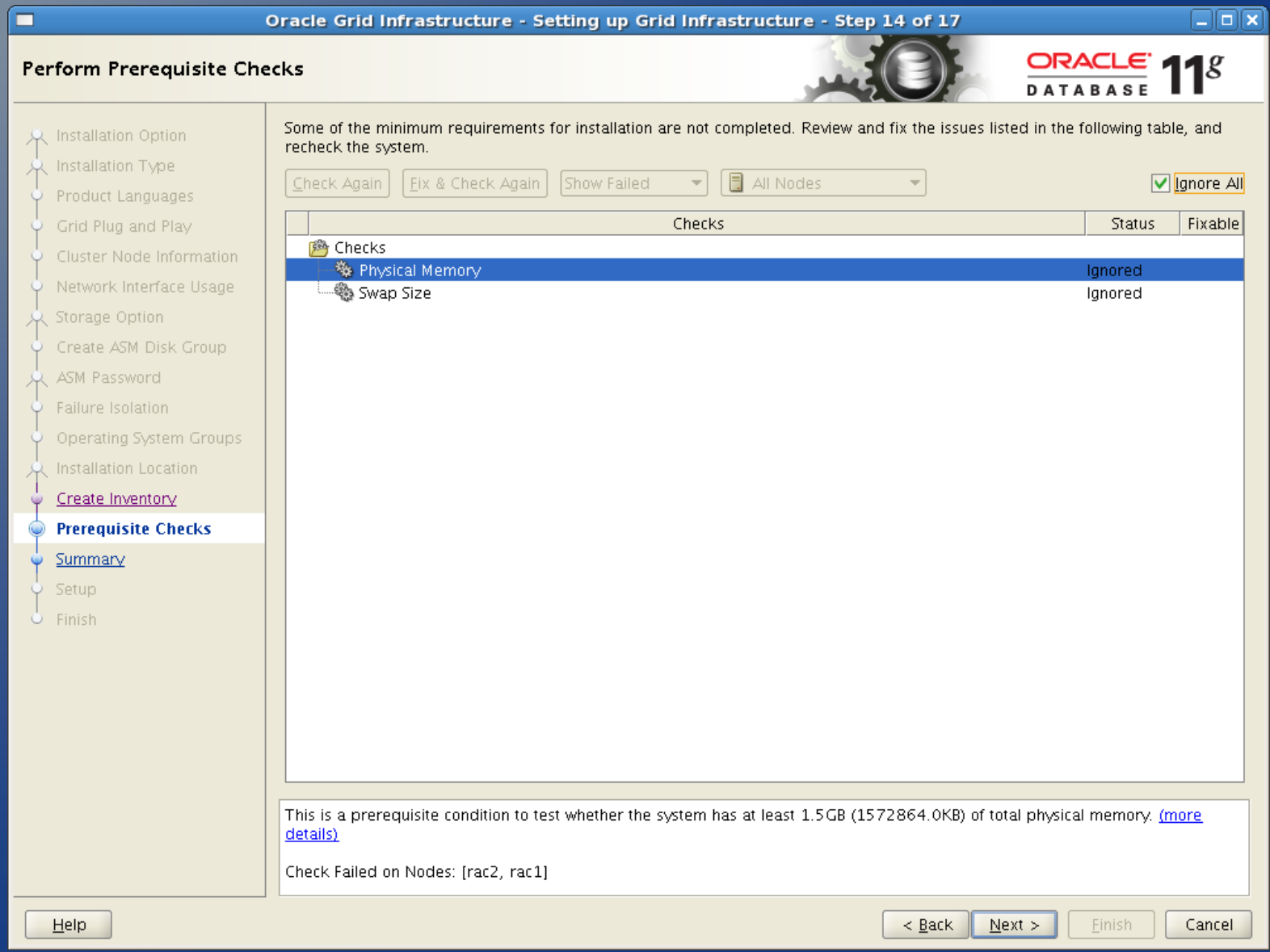
Installation – binary installation of grid infra

- Inventoriy location...



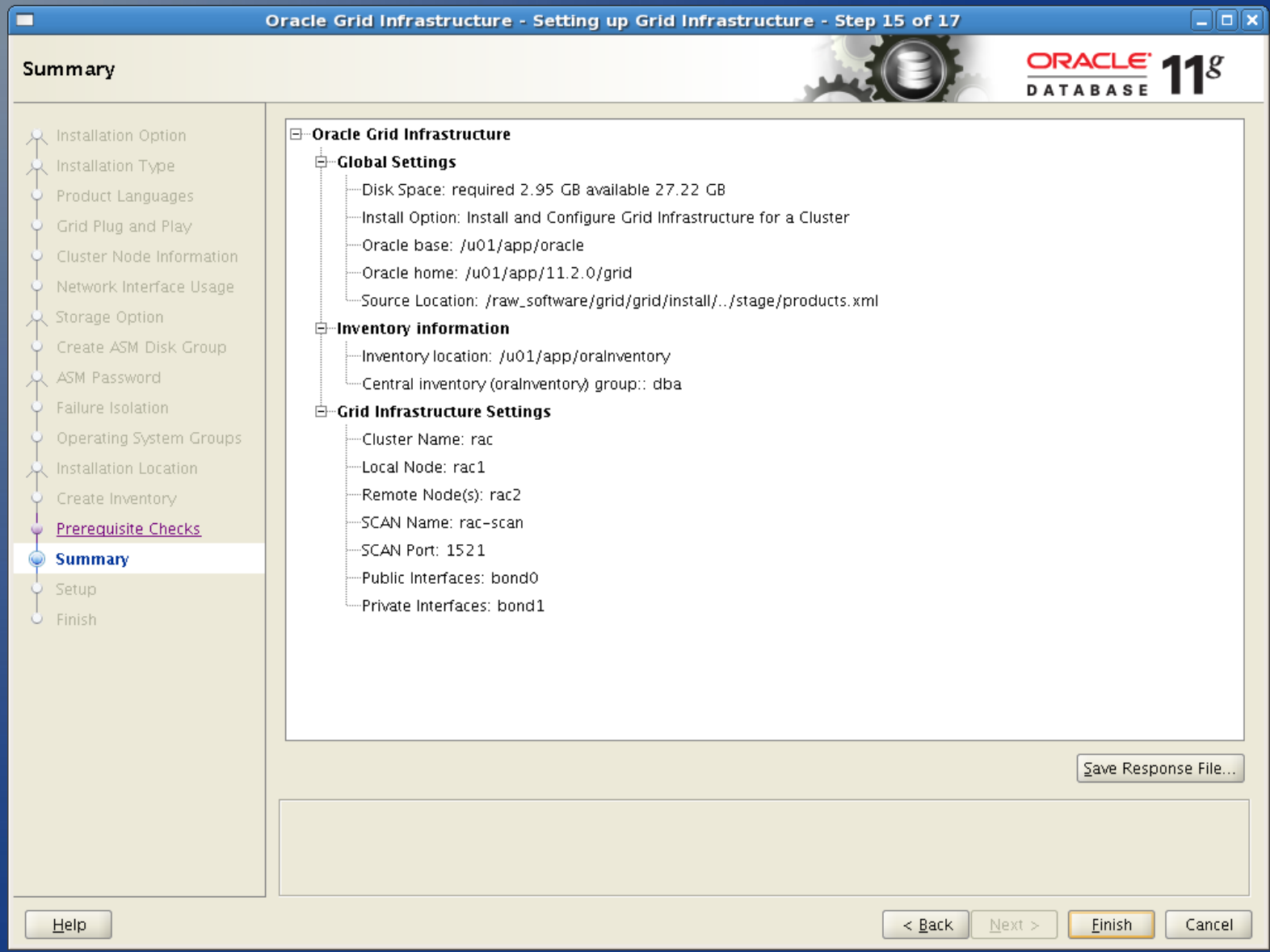
Installation – binary installation of grid infra

- Make sure you fix every issue reported here (memory and swap size are limited on virtual machine so this is not fixable...but should anyway)



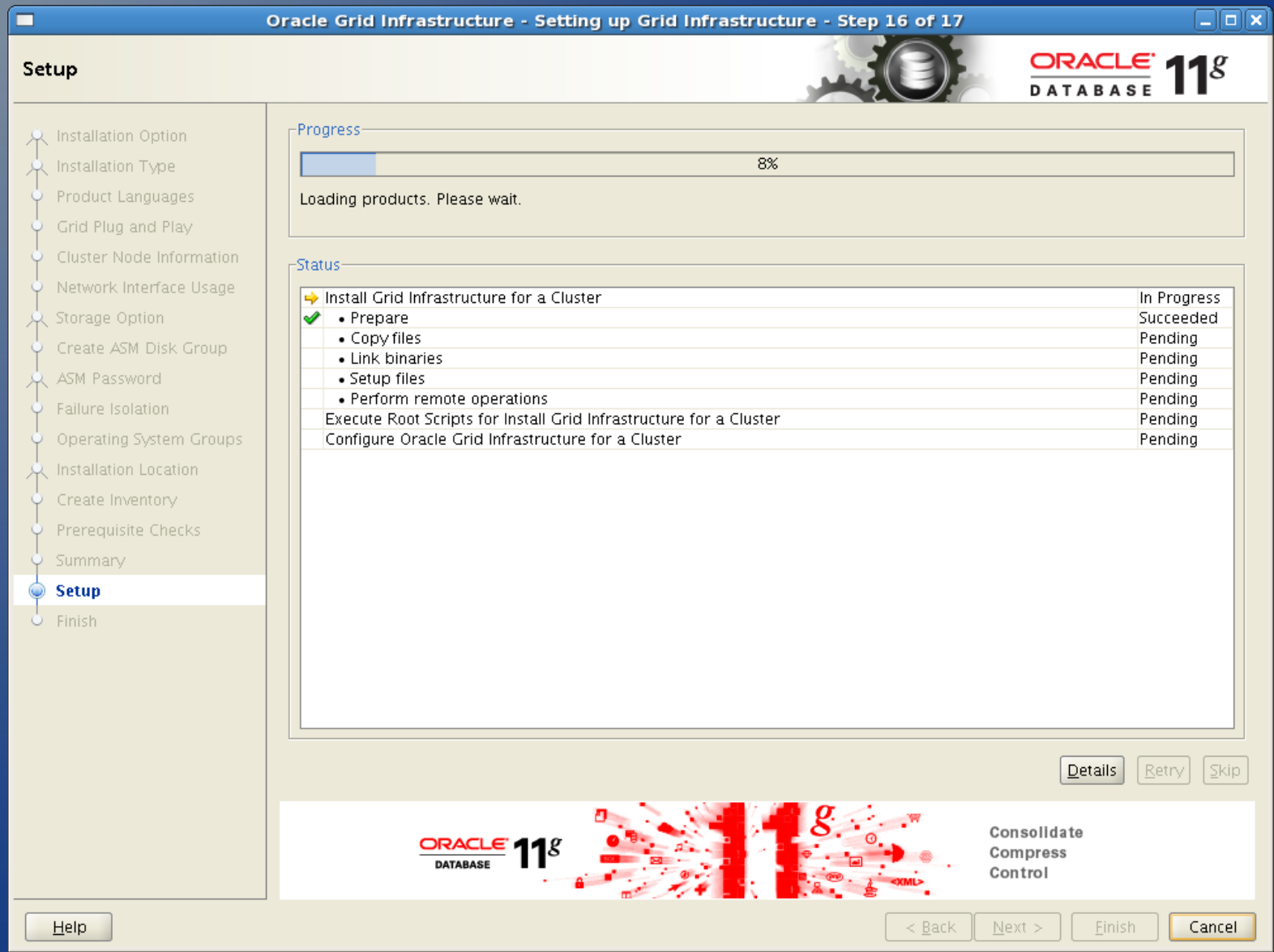
Installation – binary installation of grid infra

- Ready...



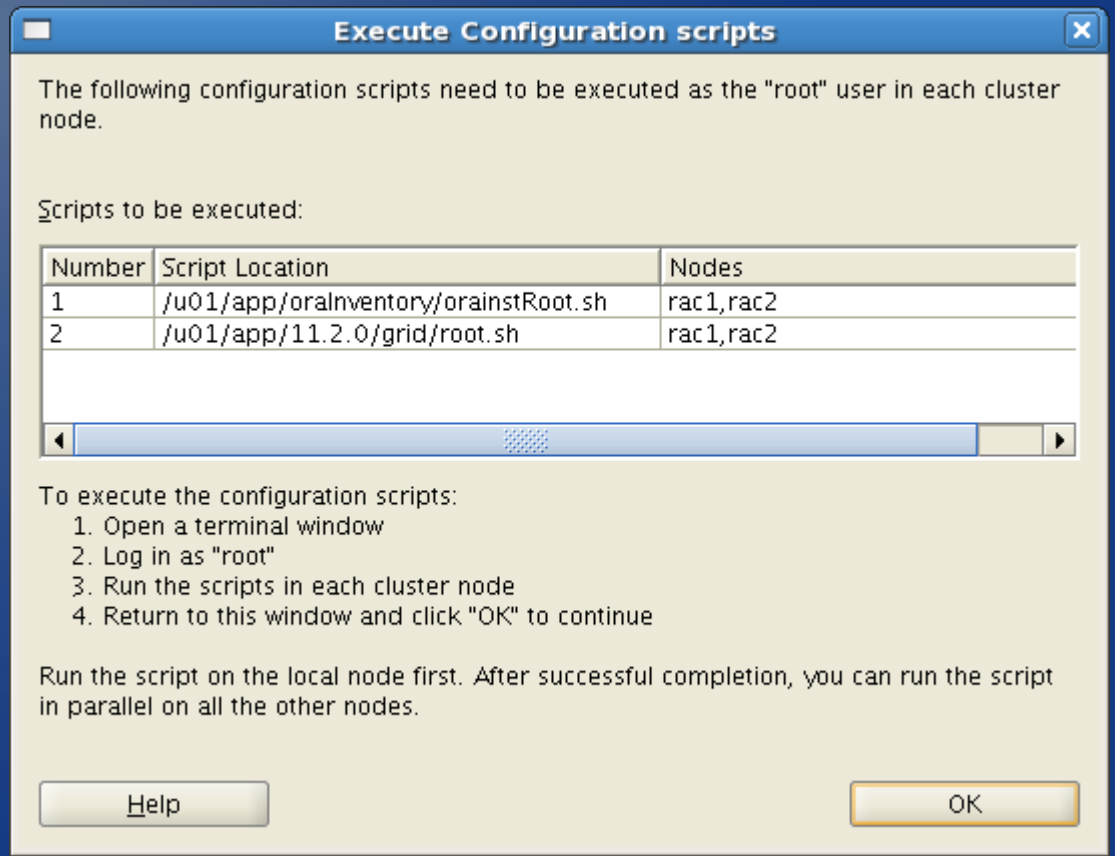
Installation – binary installation of grid infra

- Installing...



Installation – binary installation of grid infra

- Post-Installation scripts to be started in the following order:
  - orainstRoot.sh on node rac1
  - orainstRoot.sh on node rac2
  - root.sh on node rac1
  - root.sh on node rac2



Installation – binary installation of grid infra

- Sample of root.sh output

root@rac1:~

File Edit View Terminal Tabs Help

```
The execution of the script is complete.
[root@rac1 ~]# /u01/app/11.2.0/grid/root.sh
Running Oracle 11g root.sh script...

The following environment variables are set as:
    ORACLE_OWNER= grid
    ORACLE_HOME=  /u01/app/11.2.0/grid

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.sh script.
Now product-specific root actions will be performed.
2009-09-09 07:30:40: Parsing the host name
2009-09-09 07:30:40: Checking for super user privileges
2009-09-09 07:30:40: User has super user privileges
Using configuration parameter file: /u01/app/11.2.0/grid/crs/install/crsconfig_params
Creating trace directory
LOCAL ADD MODE
Creating OCR keys for user 'root', privgrp 'root'..
Operation successful.
  root wallet
  root wallet cert
  root cert export
  peer wallet
  profile reader wallet
  pa wallet
  peer wallet keys
  pa wallet keys
  peer cert request
  pa cert request
  peer cert
  pa cert
  peer root cert TP
  profile reader root cert TP
  pa root cert TP
  peer pa cert TP
  pa peer cert TP
  profile reader pa cert TP
  profile reader peer cert TP
  peer user cert
  pa user cert
Adding daemon to inittab
```

Installation – binary installation of grid infra

- Full output of root.sh on node 1

```
[root@rac1 ~]# /u01/app/11.2.0/grid/root.sh
Running Oracle 11g root.sh script...
The following environment variables are set as:
    ORACLE_OWNER= grid
    ORACLE_HOME=  /u01/app/11.2.0/grid
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.sh script.
Now product-specific root actions will be performed.
2009-09-09 07:30:40: Parsing the host name
2009-09-09 07:30:40: Checking for super user privileges
2009-09-09 07:30:40: User has super user privileges
Using configuration parameter file: /u01/app/11.2.0/grid/crs/install/crsconfig_params
Creating trace directory
LOCAL ADD MODE
Creating OCR keys for user 'root', privgrp 'root'..
Operation successful.
 root wallet
  root wallet cert
  root cert export
  peer wallet
  profile reader wallet
  pa wallet
  peer wallet keys
  pa wallet keys
  peer cert request
  pa cert request
  peer cert
  pa cert
  peer root cert TP
  profile reader root cert TP
  pa root cert TP
  peer pa cert TP
  pa peer cert TP
  profile reader pa cert TP
  profile reader peer cert TP
  peer user cert
  pa user cert
Adding daemon to inittab
CRS-4123: Oracle High Availability Services has been started.
ohasd is starting
CRS-2672: Attempting to start 'ora.gipcd' on 'rac1'
CRS-2672: Attempting to start 'ora.mdnsd' on 'rac1'
CRS-2676: Start of 'ora.gipcd' on 'rac1' succeeded
CRS-2676: Start of 'ora.mdnsd' on 'rac1' succeeded
CRS-2672: Attempting to start 'ora.gpnpd' on 'rac1'
CRS-2676: Start of 'ora.gpnpd' on 'rac1' succeeded
CRS-2672: Attempting to start 'ora.cssdmonitor' on 'rac1'
CRS-2676: Start of 'ora.cssdmonitor' on 'rac1' succeeded
CRS-2672: Attempting to start 'ora.cssd' on 'rac1'
CRS-2672: Attempting to start 'ora.diskmon' on 'rac1'
CRS-2676: Start of 'ora.diskmon' on 'rac1' succeeded
CRS-2676: Start of 'ora.cssd' on 'rac1' succeeded
CRS-2672: Attempting to start 'ora.ctssd' on 'rac1'
CRS-2676: Start of 'ora.ctssd' on 'rac1' succeeded
ASM created and started successfully.
DiskGroup DATA created successfully.


clscfg: -install mode specified
Successfully accumulated necessary OCR keys.
Creating OCR keys for user 'root', privgrp 'root'..
Operation successful.
CRS-2672: Attempting to start 'ora.crsd' on 'rac1'
CRS-2676: Start of 'ora.crsd' on 'rac1' succeeded
CRS-4256: Updating the profile
Successful addition of voting disk 6d8af7c66c874f28bfa9af46f39c5533.
Successful addition of voting disk 72b8bfad59be4f78bfd53ab49a9f064a.
Successful addition of voting disk 43ec07c3e81c4fcbbf2e03d640d75294.
Successfully replaced voting disk group with +DATA.
CRS-4256: Updating the profile
CRS-4266: Voting file(s) successfully replaced
##  STATE    File Universal Id               File Name Disk group
--  -----    -----------------               --------- ---------
 1. ONLINE   6d8af7c66c874f28bfa9af46f39c5533 (ORCL:DISK001) [DATA]
 2. ONLINE   72b8bfad59be4f78bfd53ab49a9f064a (ORCL:DISK002) [DATA]
 3. ONLINE   43ec07c3e81c4fcbbf2e03d640d75294 (ORCL:DISK004) [DATA]

Located 3 voting disk(s).
```

Installation – binary installation of grid infra

- Full output of root.sh on node 1 (con´t)

```
CRS-2673: Attempting to stop 'ora.crsd' on 'rac1'
CRS-2677: Stop of 'ora.crsd' on 'rac1' succeeded
CRS-2673: Attempting to stop 'ora.asm' on 'rac1'
CRS-2677: Stop of 'ora.asm' on 'rac1' succeeded
CRS-2673: Attempting to stop 'ora.ctssd' on 'rac1'
CRS-2677: Stop of 'ora.ctssd' on 'rac1' succeeded
CRS-2673: Attempting to stop 'ora.cssdmonitor' on 'rac1'
CRS-2677: Stop of 'ora.cssdmonitor' on 'rac1' succeeded
CRS-2673: Attempting to stop 'ora.cssd' on 'rac1'
CRS-2677: Stop of 'ora.cssd' on 'rac1' succeeded
CRS-2673: Attempting to stop 'ora.gpnpd' on 'rac1'
CRS-2677: Stop of 'ora.gpnpd' on 'rac1' succeeded
CRS-2673: Attempting to stop 'ora.gipcd' on 'rac1'
CRS-2677: Stop of 'ora.gipcd' on 'rac1' succeeded
CRS-2673: Attempting to stop 'ora.mdnsd' on 'rac1'
CRS-2677: Stop of 'ora.mdnsd' on 'rac1' succeeded
CRS-2672: Attempting to start 'ora.mdnsd' on 'rac1'
CRS-2676: Start of 'ora.mdnsd' on 'rac1' succeeded
CRS-2672: Attempting to start 'ora.gipcd' on 'rac1'
CRS-2676: Start of 'ora.gipcd' on 'rac1' succeeded
CRS-2672: Attempting to start 'ora.gpnpd' on 'rac1'
CRS-2676: Start of 'ora.gpnpd' on 'rac1' succeeded
CRS-2672: Attempting to start 'ora.cssdmonitor' on 'rac1'
CRS-2676: Start of 'ora.cssdmonitor' on 'rac1' succeeded
CRS-2672: Attempting to start 'ora.cssd' on 'rac1'
CRS-2672: Attempting to start 'ora.diskmon' on 'rac1'
CRS-2676: Start of 'ora.diskmon' on 'rac1' succeeded
CRS-2676: Start of 'ora.cssd' on 'rac1' succeeded
CRS-2672: Attempting to start 'ora.ctssd' on 'rac1'
CRS-2676: Start of 'ora.ctssd' on 'rac1' succeeded
CRS-2672: Attempting to start 'ora.asm' on 'rac1'
CRS-2676: Start of 'ora.asm' on 'rac1' succeeded
CRS-2672: Attempting to start 'ora.crsd' on 'rac1'
CRS-2676: Start of 'ora.crsd' on 'rac1' succeeded
CRS-2672: Attempting to start 'ora.evmd' on 'rac1'
CRS-2676: Start of 'ora.evmd' on 'rac1' succeeded
CRS-2672: Attempting to start 'ora.asm' on 'rac1'
CRS-2676: Start of 'ora.asm' on 'rac1' succeeded
CRS-2672: Attempting to start 'ora.DATA.dg' on 'rac1'
CRS-2676: Start of 'ora.DATA.dg' on 'rac1' succeeded
CRS-2672: Attempting to start 'ora.registry.acfs' on 'rac1'
CRS-2676: Start of 'ora.registry.acfs' on 'rac1' succeeded
```

```
rac1     2009/09/09 07:46:29     /u01/app/11.2.0/grid/cdata/rac1/backup_20090909_074629.olr

Preparing packages for installation...
Cvuqdisk-1.0.7-1
Configure Oracle Grid Infrastructure for a Cluster ... succeeded
Updating inventory properties for clusterware

Starting Oracle Universal Installer...
Checking swap space: must be greater than 500 MB.   Actual 2801 MB    Passed
The inventory pointer is located at /etc/oraInst.loc
The inventory is located at /u01/app/oraInventory
'UpdateNodeList' was successful.
```

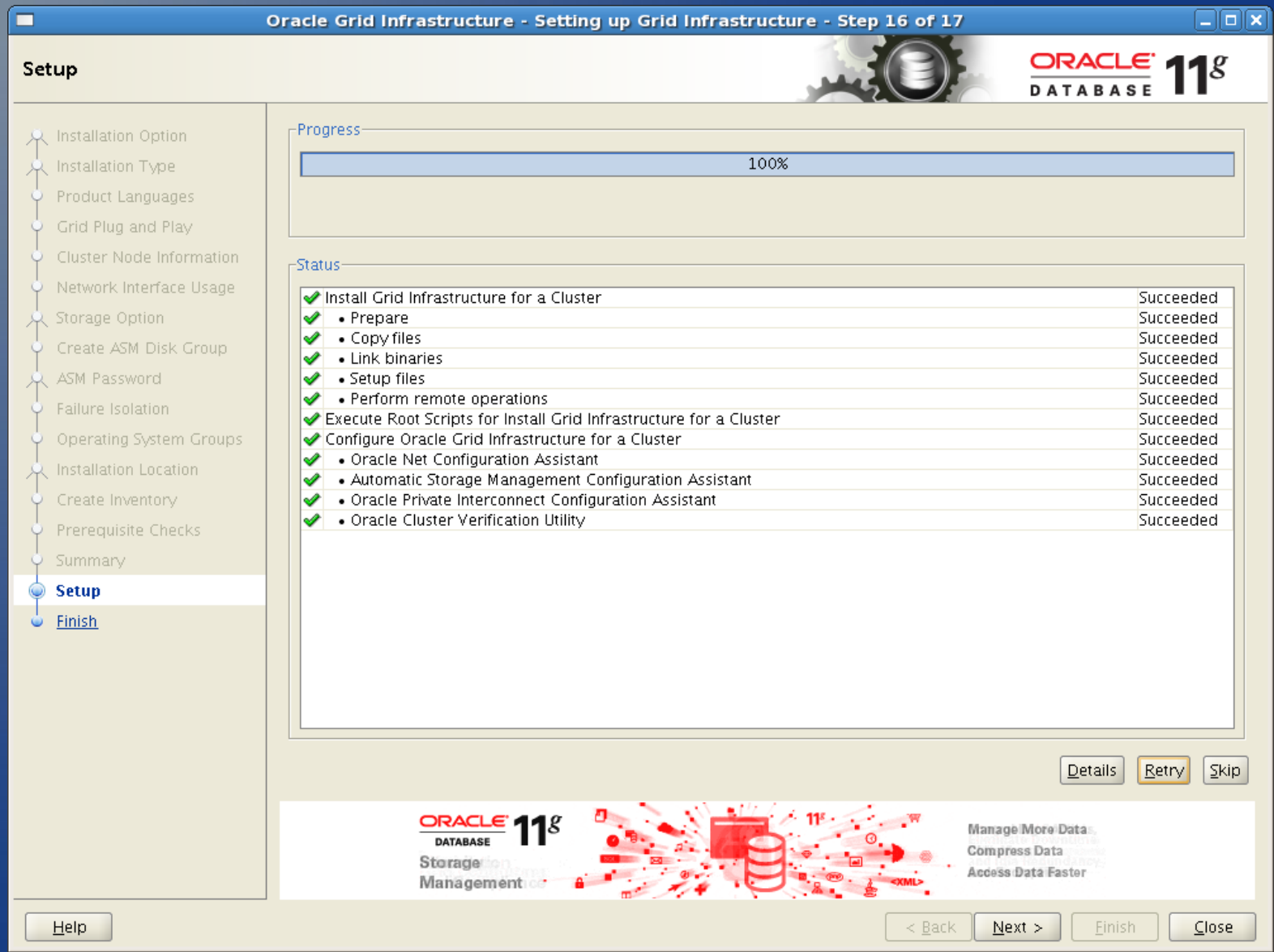
Installation – binary installation of grid infra

- Full output of root.sh on node 2

```
[root@rac2 11.2.0]# /u01/app/11.2.0/grid/root.sh
Running Oracle 11g root.sh script...

The following environment variables are set as:
    ORACLE_OWNER= grid
    ORACLE_HOME=  /u01/app/11.2.0/grid

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.sh script.
Now product-specific root actions will be performed.
2009-09-09 07:48:45: Parsing the host name
2009-09-09 07:48:45: Checking for super user privileges
2009-09-09 07:48:45: User has super user privileges
Using configuration parameter file: /u01/app/11.2.0/grid/crs/install/crsconfig_params
Creating trace directory
LOCAL ADD MODE
Creating OCR keys for user 'root', privgrp 'root'..
Operation successful.
Adding daemon to inittab
CRS-4123: Oracle High Availability Services has been started.
ohasd is starting

CRS-4402: The CSS daemon was started in exclusive mode but found an active CSS daemon on
node rac1, number 1, and is terminating
An active cluster was found during exclusive startup, restarting to join the cluster
CRS-2672: Attempting to start 'ora.mdnsd' on 'rac2'
CRS-2676: Start of 'ora.mdnsd' on 'rac2' succeeded
CRS-2672: Attempting to start 'ora.gipcd' on 'rac2'
CRS-2676: Start of 'ora.gipcd' on 'rac2' succeeded
CRS-2672: Attempting to start 'ora.gpnpd' on 'rac2'
CRS-2676: Start of 'ora.gpnpd' on 'rac2' succeeded
CRS-2672: Attempting to start 'ora.cssdmonitor' on 'rac2'
CRS-2676: Start of 'ora.cssdmonitor' on 'rac2' succeeded
CRS-2672: Attempting to start 'ora.cssd' on 'rac2'
CRS-2672: Attempting to start 'ora.diskmon' on 'rac2'
CRS-2676: Start of 'ora.diskmon' on 'rac2' succeeded
CRS-2676: Start of 'ora.cssd' on 'rac2' succeeded
CRS-2672: Attempting to start 'ora.ctssd' on 'rac2'
CRS-2676: Start of 'ora.ctssd' on 'rac2' succeeded
CRS-2672: Attempting to start 'ora.drivers.acfs' on 'rac2'
CRS-2676: Start of 'ora.drivers.acfs' on 'rac2' succeeded
CRS-2672: Attempting to start 'ora.asm' on 'rac2'
CRS-2676: Start of 'ora.asm' on 'rac2' succeeded
CRS-2672: Attempting to start 'ora.crsd' on 'rac2'
CRS-2676: Start of 'ora.crsd' on 'rac2' succeeded
CRS-2672: Attempting to start 'ora.evmd' on 'rac2'
CRS-2676: Start of 'ora.evmd' on 'rac2' succeeded

rac2     2009/09/09 08:01:00     /u01/app/11.2.0/grid/cdata/rac2/backup_20090909_080100.olr

Preparing packages for installation...
cvuqdisk-1.0.7-1
Configure Oracle Grid Infrastructure for a Cluster ... succeeded
Updating inventory properties for clusterware
Starting Oracle Universal Installer...
Checking swap space: must be greater than 500 MB.   Actual 2969 MB    Passed
The inventory pointer is located at /etc/oraInst.loc
The inventory is located at /u01/app/oraInventory

'UpdateNodeList' was successful.
```

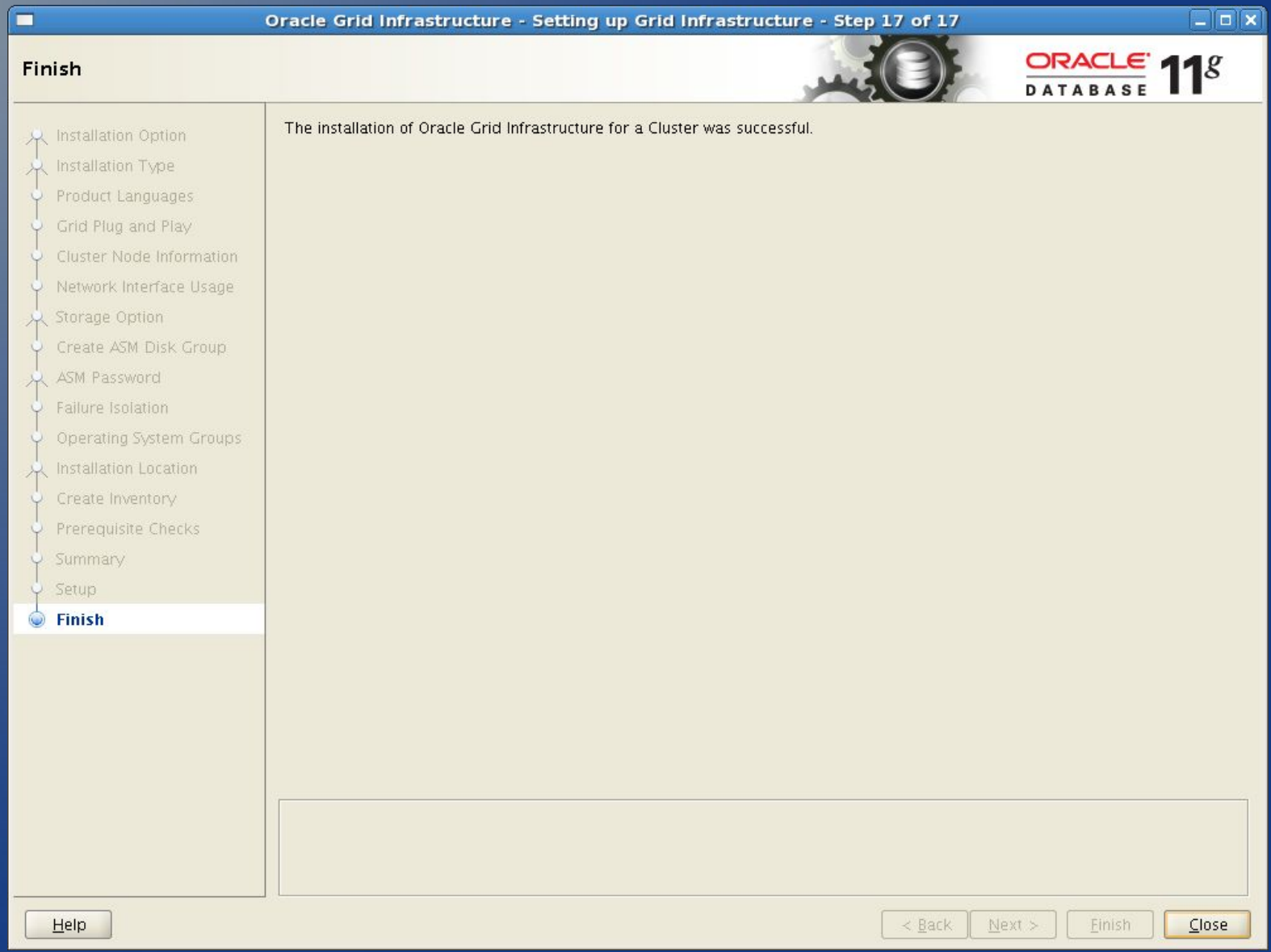
Installation – binary installation of grid infra

- If everything works as expected the result should look like this:



Installation – binary installation of grid infra

- FINISHED!



Installation – binary installation of grid infra

- If there are problems:
  - Look at the log files located on /u01/app/oraInventory/logs
  - Fix the issues noted here
  - If this does not work out: Search Metalink / Open SR

Installation – binary installation of grid infra

- Where to go now?
  - We just installed the Infrastructure needed for RAC, i.e. ASM and Clusterware
  - Install diagnostic utilities (strongly recommended)
    - OSWatcher
    - and RACDDT
  - Tune Kernel parameters (if not done before)
  - Create at least TWO more disk groups:
    - one for holding database files (i.e. datafiles and binary installation files)
    - one to be used as flashback recovery area
  - now we need to install a RAC database → this will be Part 4
  - Part 3 will cover ADVM (ASM Dynamic Volume Manager) and ACFS (ASM Cluster File system) shortly
  - Backup current configuration

Part 3 –
ADVM and ACFS

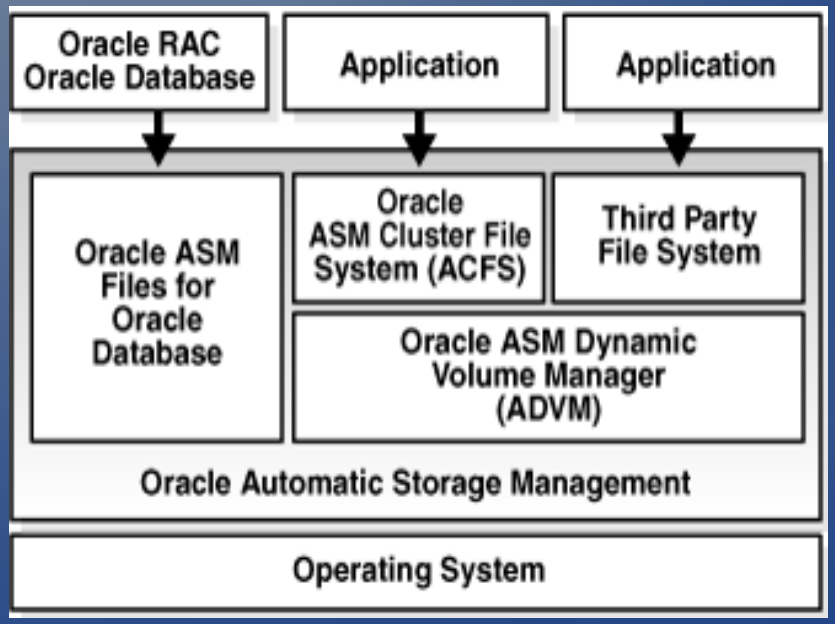
System configuration - overview

- We will use the system configured and installed in part 2

ADVM and ACFS - what it is

- ADVM = ASM dynamic volume manager
- ACFS = ASM cluster file system

- Basic layout



(Source: Oracle Database Storage Administrator's Guide 11g Release 2)

ADVM and ACFS - what it is

- ADVM - Advantages
  - Integrated in ASM; this can be an disadvantage as well :-)
  - Inhertits storage from ASM hence enables host-based mirroring (either 2- or 3-way-mirroring)
  - multiple volumes within a disk group can be created with an file system such as ext3, ext4, reiserfs, … on top of it and will support storage of any file type as the file system normally woud – EXCEPT files which can be place in ASM directly
  - ADVM volume dynamically resizeable

ADVM and ACFS - what it is

- ADVM - Disadvantages
  - ADVM volumes may be resized online; but the used file system must support it as well (ext3 on OEL 5 does support online resizing but does not support online shrinking)
  - Storing files which can be stored in ASM directly in ADVM + file system is not supported
  - NFS on top of ADVM is also not supported
  - ASM configuration assistant (asmca) only supports creation of volumes / file system... delete a volume / file system requires command line

ADVM and ACFS - what it is

- ACFS - Advantages
  - cluster file system on top of ASM and ADVM
  - as available as ASM is (inherits storage from ASM disk group and ADVM volume)
  - Supports storage of files which cannot be directly stored in ASM, i.e.
    - executables
    - trace files
    - log files
    - …
    - Supports even oracle database binary installations
  - On ACFS read-only Snapshots can be created
  - dynamically resizeable

ADVM and ACFS - what it is

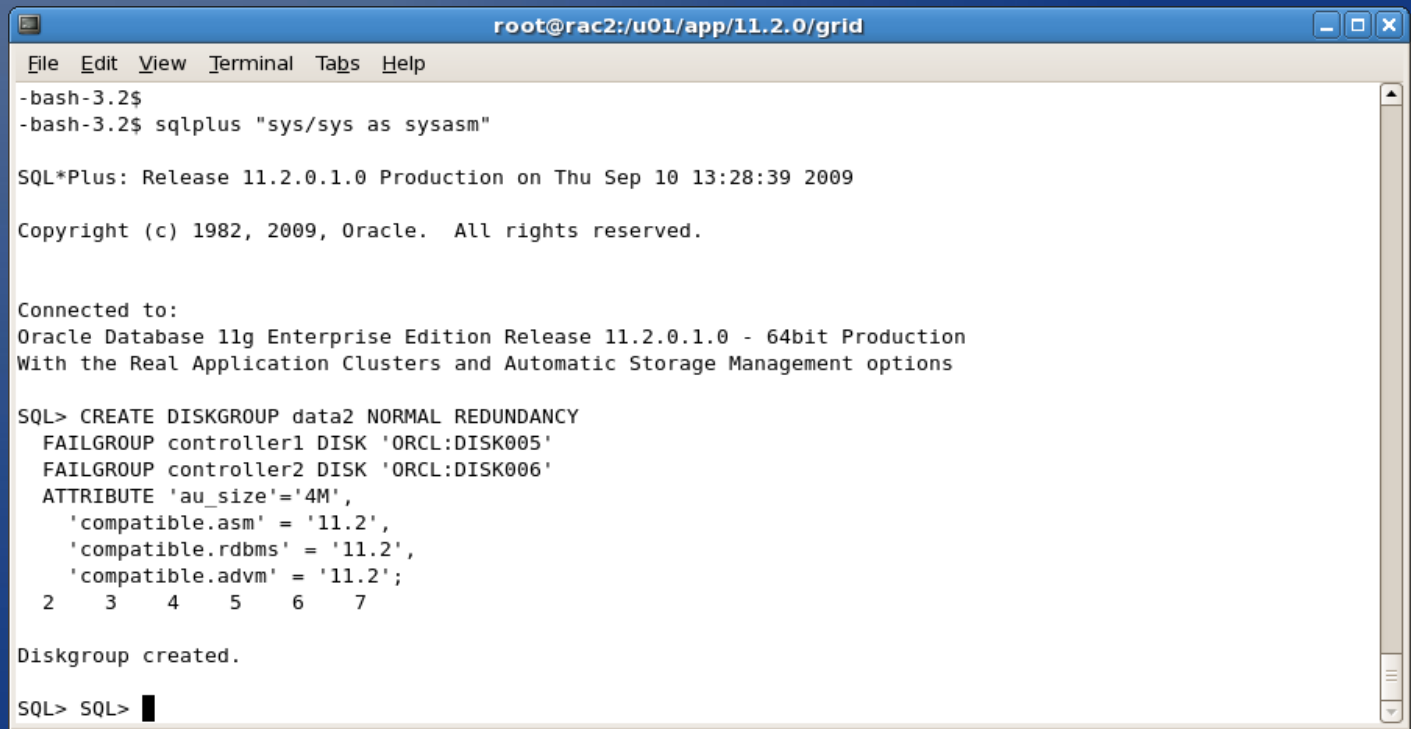
- ACFS - Advantages
  - Useable accross plattforms
    - Thoughts
      - Do i need licenses for grid infrastructrue?
      - If not: What if grid infrastructure + ASM used to provide host-based mirroring and cluster file system for non-oracle applications, for instance web servers
  - ACFS Mount registry: used for mouting ACFS and ADVM file system across reboots

ADVM and ACFS - what it is

- ACFS – Disadvantages
  - for example storing database files in ACFS is not supported, according to the documentation
    „Oracle Support Services will not take calls and development will not fix bugs associated with storing unsupported file types in Oracle ACFS“
  - Only available with RedHat Server 5 or Oracle Enterprise Linux 5 !
  - Disk group compatible parameter COPATBILE.ASM and COMPATIBLE.ADVM must be set so 11.2
  - ASM configuration assistant (asmca) only supports creation of volumes / file system... delete a volume / file system requires command line

ADVM and ACFS - some tests

- Lets first create an additional disk group called „DATA2“ which consists for two iSCSI LUNs with 30 GB each
  - Preparation:
    - LUNs visible with „fdisk -l“
    - Partition created (one on each LUN)
    - disk labeled with „oracleasm createdisk <name> <devpath>“
  - Create disk group in ASM
    (remember to connect as „sys as sysASM“!)

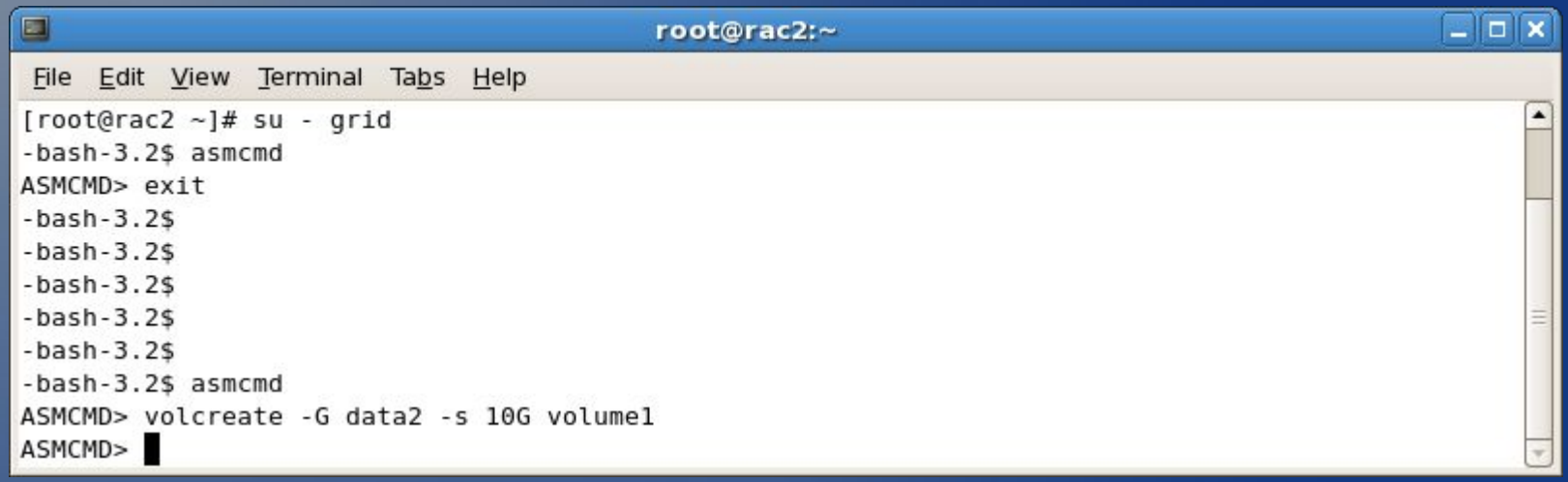


ADVM and ACFS - some tests

- Notes on disk groups
  - AUSIZE of 4 MB recommended by Oracle documentation due to:
    - Increased I/O through the I/O subsystem if the I/O size is increased to the AU size.
    - Reduced SGA size to manage the extent maps in the database instance.
    - Faster datafile initialization if the I/O size is increased to the AU size.
    - Increased file size limits.
    - Reduced database open time.
  - Large AUSIZE requires as well
    - Increasing size of maximum IO request to at least 4 MB in operating system, drive, HBA, storage system
    - Larger stripe size in storage system (pre 11g R2: 1 MB stripe size, with 11g R2: 4 MB? → to be tested)
  - Read the documentation on COMPATIBLE parameters;most „cool“ features are only available with 11.2 COMPATIBLE parameter hence require 11g R2 database

ADVM and ACFS - some tests

- Creating ACFS
  - Requires ADVM in which ACFS can be created
    - volcreate creates ADVM volume

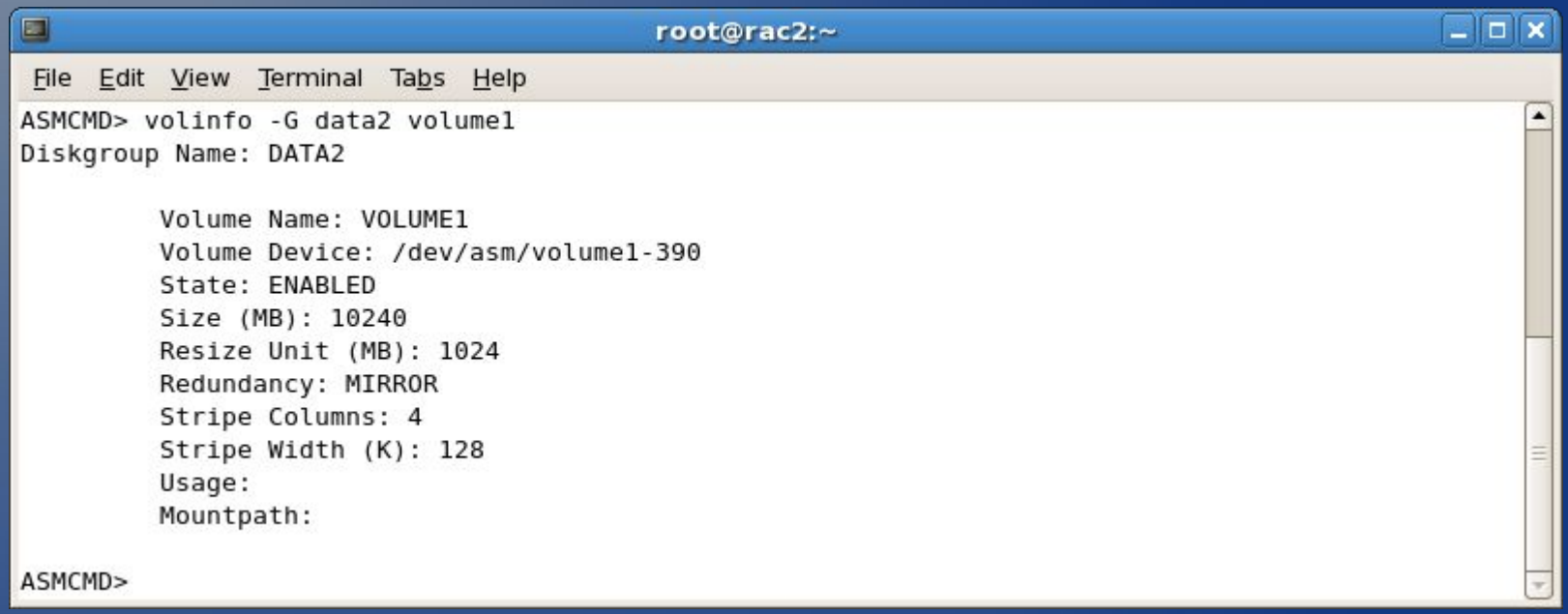


(command above shows minimal command creating an ADVM volume; redundancy is derived from disk group, our data group was created with „normal“ redundancy so the volume inherits this as well)

  - Creation with SQL also possible: „ALTER DISKGROUP data2 ADD VOLUME volume1 SIZE 10G;“

ADVM and ACFS - some tests

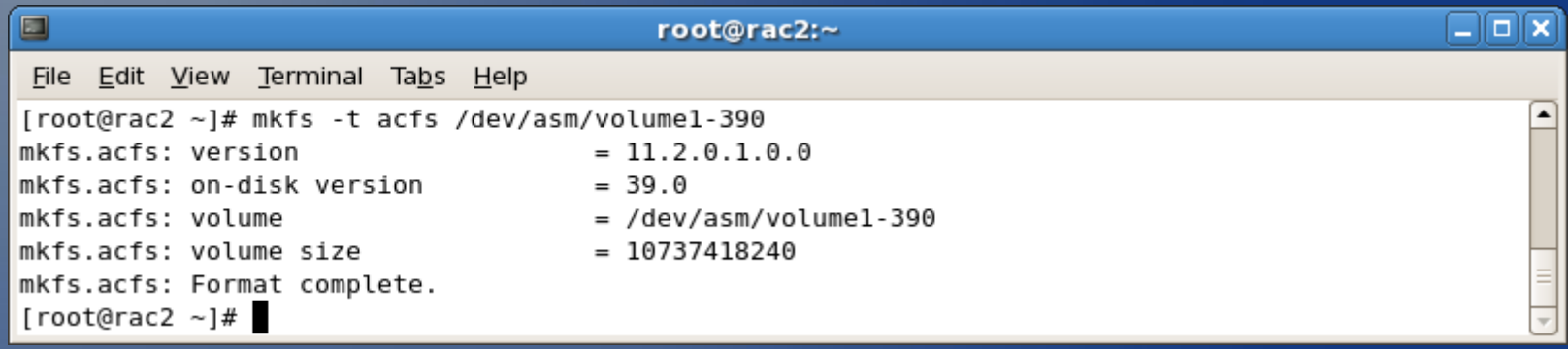
- Creating ACFS
  - Requires ADVM in which ACFS can be created
    - volinfo shows detailed information



    - Especially device path is important for creating the file system

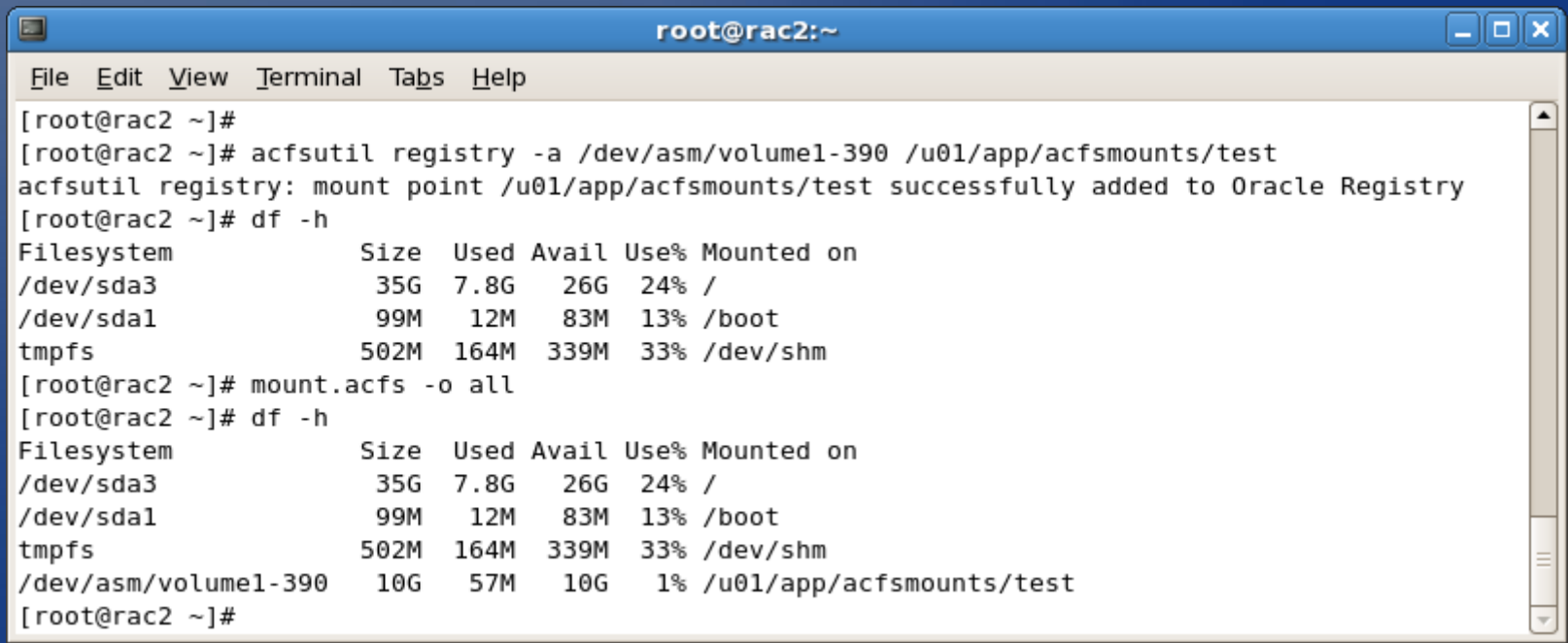
ADVM and ACFS - some tests

- Creating ACFS
  - Requires ADVM in which ACFS can be created
    - create ACFS from operating system (only on one node)



ADVM and ACFS - some tests

- Creating ACFS
  - Requires ADVM in which ACFS can be created
    - register acfs in registry to be mounted across reboots with „acfsutil“
    - ATTENTION: DO NOT register shared oracle home directories; this will be done by the clusterware
    - test of everything works by issueing „mount.acfs -o all“ on all nodes; the file system should be mounted and accessible



ADVM and ACFS - some tests

- Simple Performance tests:

  dd if=/dev/zero bs=1024k of=<path> count=1000 oflag=direct
  → direct I/O used; no caching, performed 10 times

  - write to ACFS
    - ACFS 2-way mirror: ~ 6 MB/s average
    - ACFS unprotected: ~ 12 MB/s averga
    - → expected... one disk, double I/O halfed throughput
  - direct write to iSCSI LUN: ~ 14.3 MB/s average

  - Tests were performed within a VMWare... so results are most likely not accurate... but we get an impression.. we will check this on real hardware later!

Part 4 – Installation Real Application Cluster Database (RAC)

System configuration

- System configuration:
  - two virtual machines (VMWARE)
    - 1 vCPU
    - 4 GB RAM
    - 40 GB local Disk
  - Storage exported via ISCSI
    - 4 LUNs with 10 GB each
    - 2 LUNs with 30 GB each
  - Operating system configuration
    - Oracle Enterprise Linux 5.3 x86_64 (Kernel 2.6.18-128.el5)
    - Installed packages: default system + development packages

System configuration

- System configuration (con´t):
  - Cluster Name: „RAC“
  - Grid Binary installation on local disk
  - OCR, Voting and datafiles stored in ASM
  - Oracle HOME stored on ACFS file system (this menas we will have a SHARED home directory among all nodes!)
  - Oracle HOME will be installed unser user „ora11p“

Installation Overview

- Installation of Oracle 11g Release 2 Grid Infrastructure
  - → done in chapter 2

- Installation of Oracle 11g Release 2 Database (rac installation)
  - Review system requirements
  - Install database software (aka „binaries“)
  - Create database

Installation – Review requirements

- All requirments from chapter 2 (grid infrastructure installation) applies here as well, for instance:
  - Kernel parameter
  - Limits (esp. configure limits for new user „ora11p“ which will hold the binary database installation)
  - Synchronous time
  - dns-resolvable SCAN name
  - working public and private interconnect
  - shared and accessible storage
  - at least ONE better two more disk groups created:
    - one for database files and binary installation
    - one for flashback recovery area
  - Note: SSH equivalence will be set up by installer

Installation – Install database binaries

- Create ACFS file system:
  - Create further ACFS mount directory

    mkdir -p /u01/app/oracle/product/11.2.0/ora11p
  - Create ADVM volume

```
-bash-3.2$
-bash-3.2$
-bash-3.2$
-bash-3.2$ asmcmd
ASMCMD> volcreate -G data2 -s 10g ora11p_home
ASMCMD>
```

  - Create ACFS file system

```
[root@rac2 ~]#
[root@rac2 ~]# mkfs.acfs /dev/asm/ora11p_home-132
mkfs.acfs: version                   = 11.2.0.1.0.0
mkfs.acfs: on-disk version           = 39.0
mkfs.acfs: volume                    = /dev/asm/ora11p_home-132
mkfs.acfs: volume size               = 10737418240
mkfs.acfs: Format complete.
[root@rac2 ~]#
```

Installation – Install database binaries

- Create ACFS file system:
  - DO NOT register ACFS file system mountpoint for the shared oracle home; we will do this later (when creating the database) as a clusterware resource
  - Mount ACFS file system on both nodes

    mount /dev/asm/ora11p_home-132
    /u01/app/oracle/product/11.2.0/orap11p

```
[root@rac1 ~]# df -h
Filesystem            Size  Used Avail Use% Mounted on
/dev/sda3              26G  8.8G   16G  37% /
/dev/sda1              99M   12M   83M  13% /boot
tmpfs                 1.9G  164M  1.7G   9% /dev/shm
/dev/asm/ora11p_home-132
                       10G   85M   10G   1% /u01/app/oracle/product/11.2.0/ora11p
[root@rac1 ~]#
```

Installation – Install database binaries

- Create User on both nodes:

```
useradd -u 501 -g dba -d /u01/app/oracle/product/11.2.0/ora11p ora11p
passwd ora11p

chown -R root:dba /u01/app/oracle/product
chmod -R 775 /u01/app/oracle/product
chown -R ora11p:dba /u01/app/oracle/product/11.2.0/ora11p
```

- Create .bash_profile for user ora11p (note: changes on node A will be visible on node B cause were using ACFS; so changing profile file is needed only once)

```
export ORACLE_BASE=/u01/app/oracle
export ORACLE_HOME=$ORACLE_BASE/product/11.2.0/ora11p
export ORACLE_SID=ORA11P
if [ -t 0 ]; then
stty intr ^C
fi
```

Installation – Install database binaries

- One word on Cluster Verification utility (cluvfy):
  - should be used prior installation
  - At this point the user „ora11p“ does not have SSH equivalence configured
  - So there are two options:
    - start cluvfy after installer set up ssh equivanlence
    - set up SSH equivalence manually
  - I personally prefer to run cluvfy after installer set up ssh equivalence
  - However the database installer seems to run a more or less complete check automatically
  - If you want to start cluvfy here is the syntax:

```
cluvfy stage -pre dbinst -fixup -n nodeA,nodeB -r <release> -osdba
<name of sysdba group on unix> -verbose

    <release> is: 10gR1, 10gR2, 11gR1, 11gR2
```

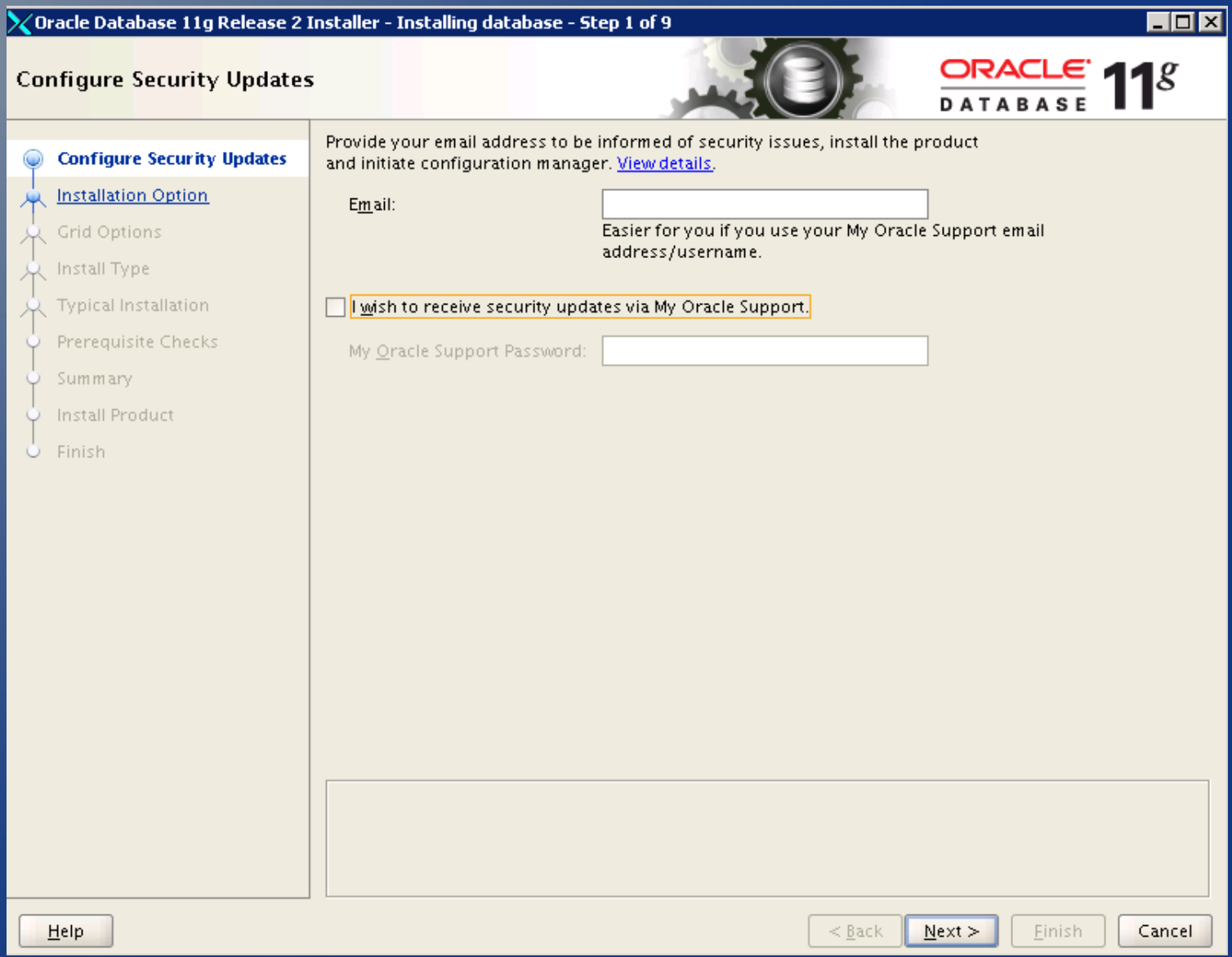
Installation – Install database binaries

- Start installer als user „ora11p“ on any node

```
-bash-3.2$ /raw_software/database/runInstaller
Starting Oracle Universal Installer...

Checking Temp space: must be greater than 120 MB.   Actual 15415 MB    Passed
Checking swap space: must be greater than 150 MB.   Actual 4094 MB    Passed
Checking monitor: must be configured to display at least 256 colors.    Actual 16777216    Passed
Preparing to launch Oracle Universal Installer from /tmp/OraInstall2009-09-14_10-21-53AM. Please wait .
..
```

Installation – Install database binaries

- For testing purposes i deselected update notifications... in productive environments this is highly recommended

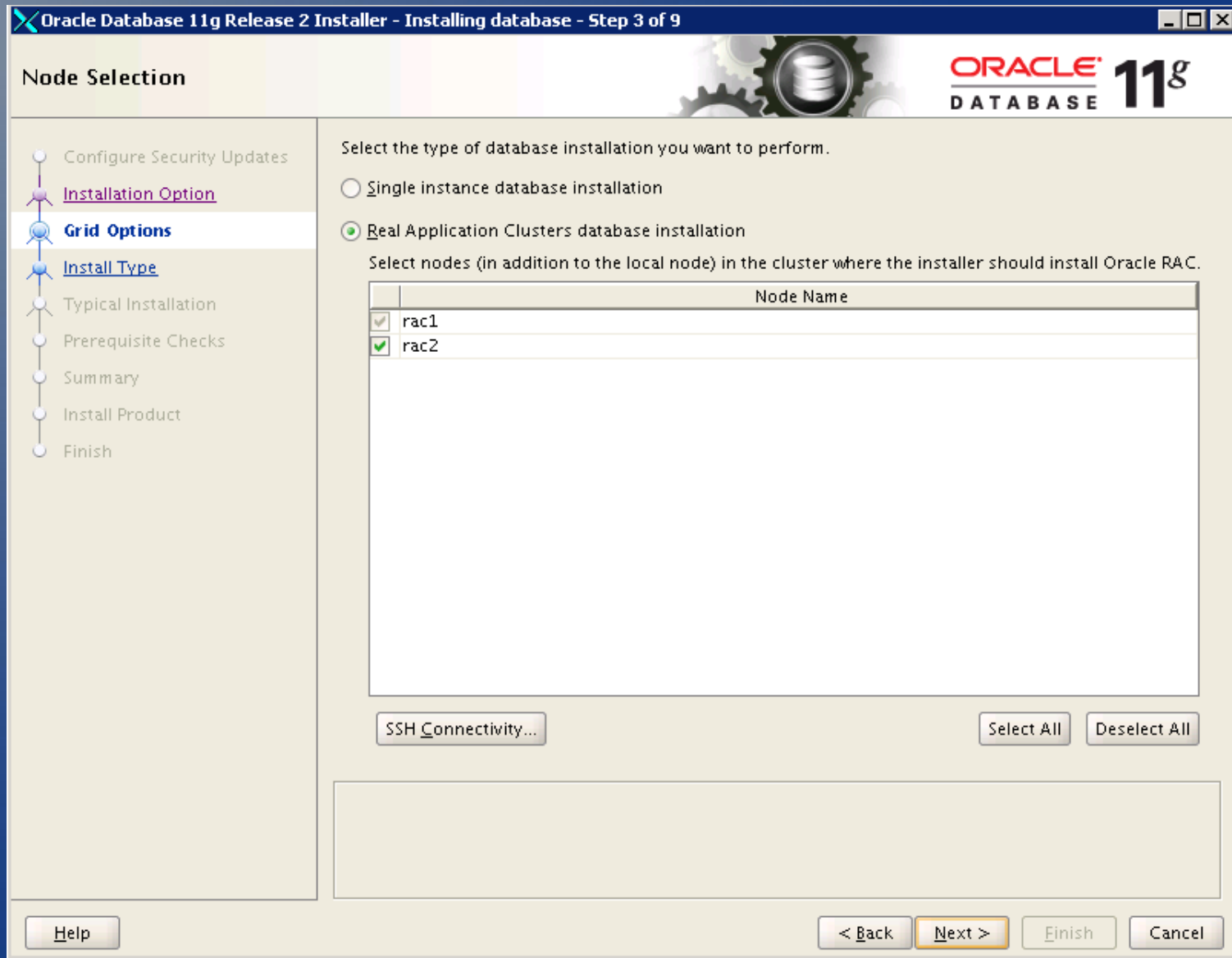


Installation – Install database binaries

- we will install database software only and create database later

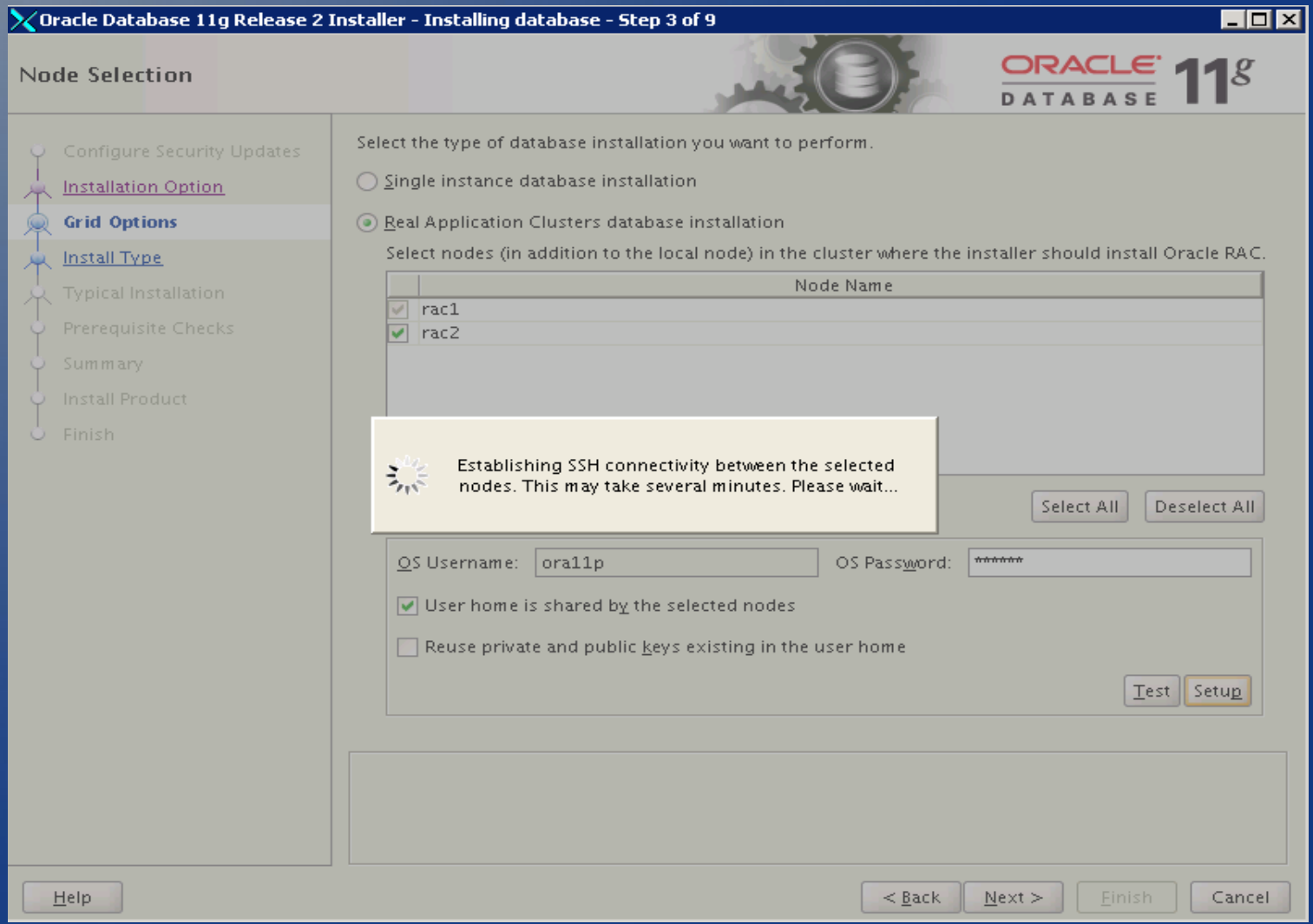
Installation – Install database binaries

- both nodes were discovered correctly….



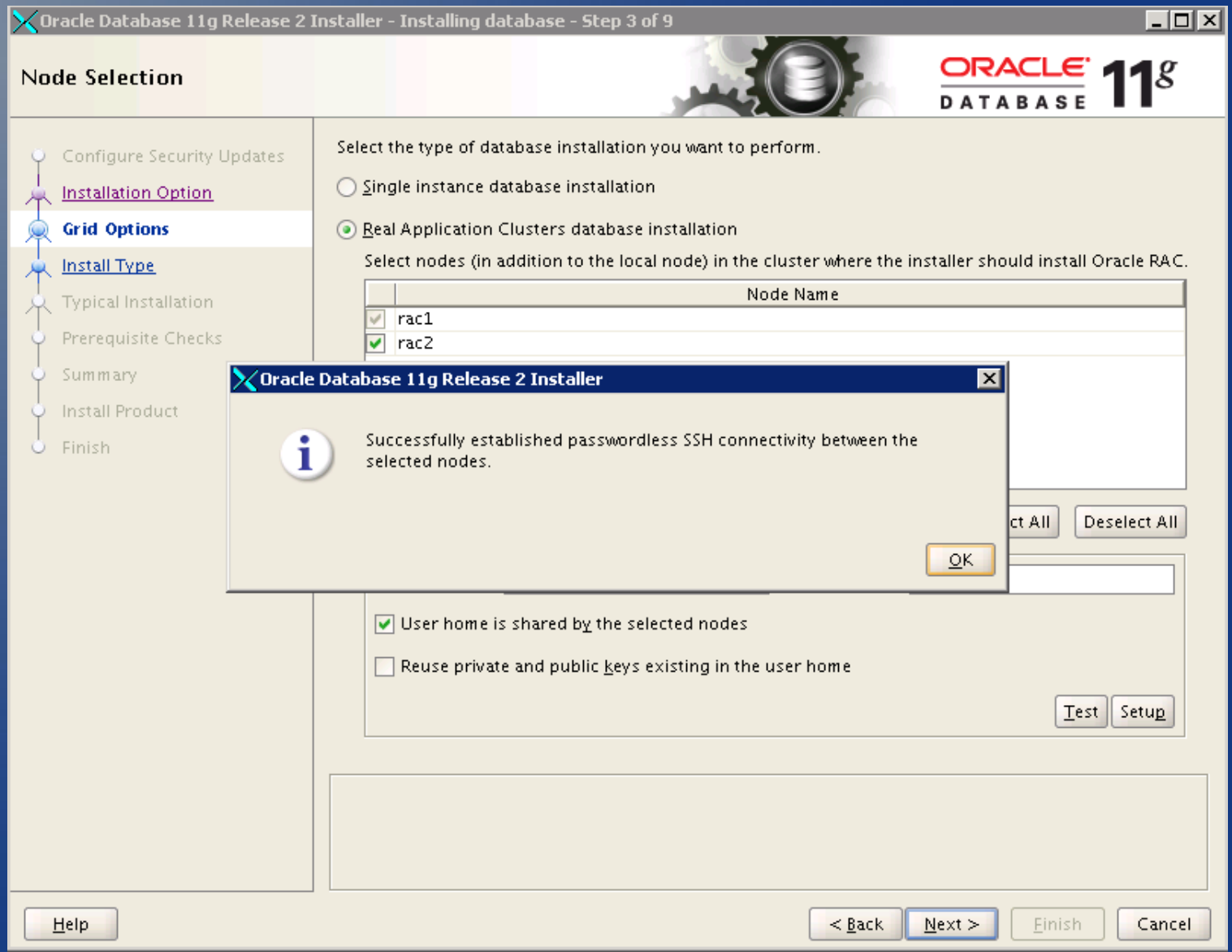
Installation – Install database binaries

- because were installing as user „ora11p“ which is different from the user holding our infrastructure installation we need to create passwordless ssh connectitivty
- you also need to select „user home is shared“ cause were using ACFS



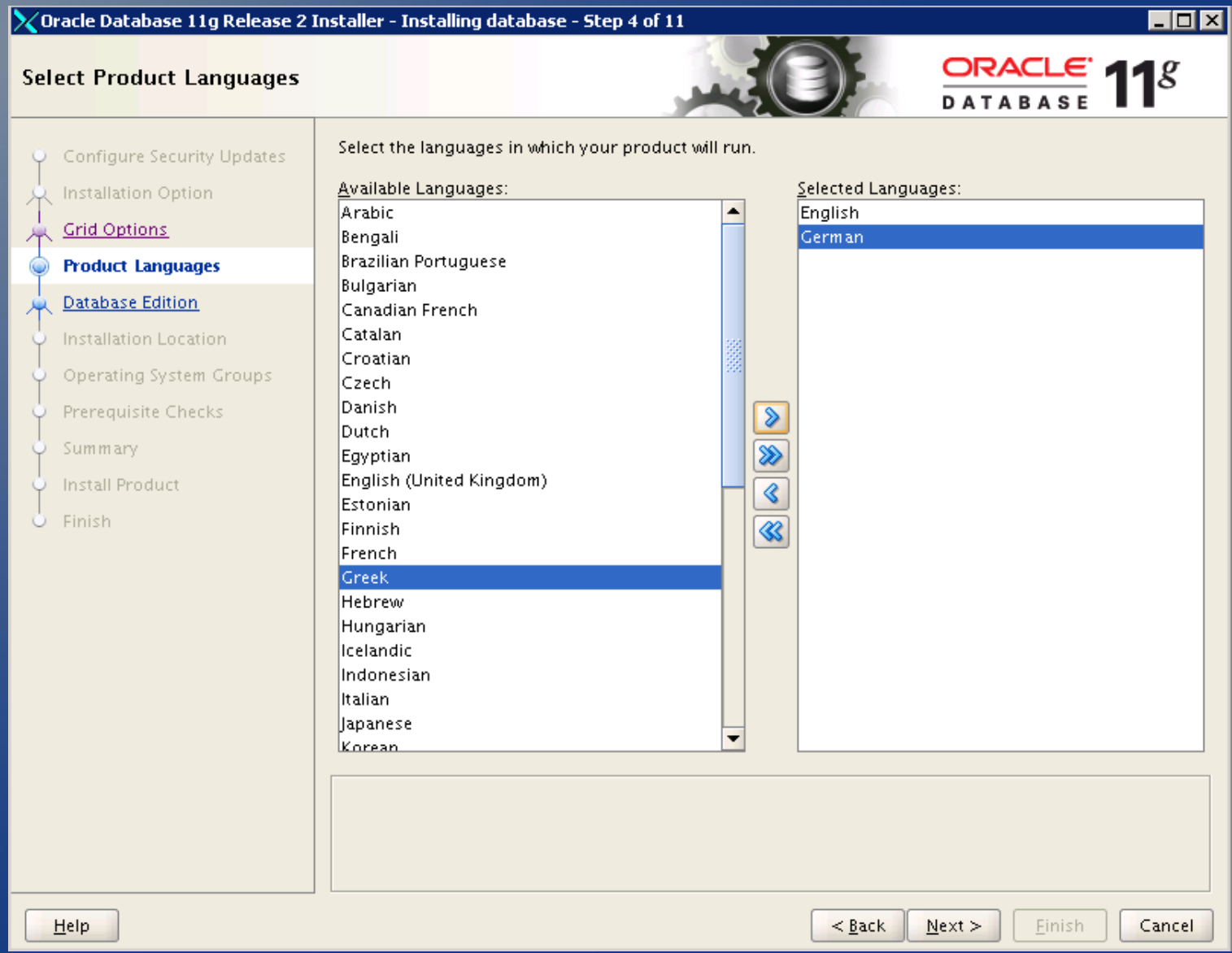
Installation – Install database binaries

- SSH connectivity successfully established...



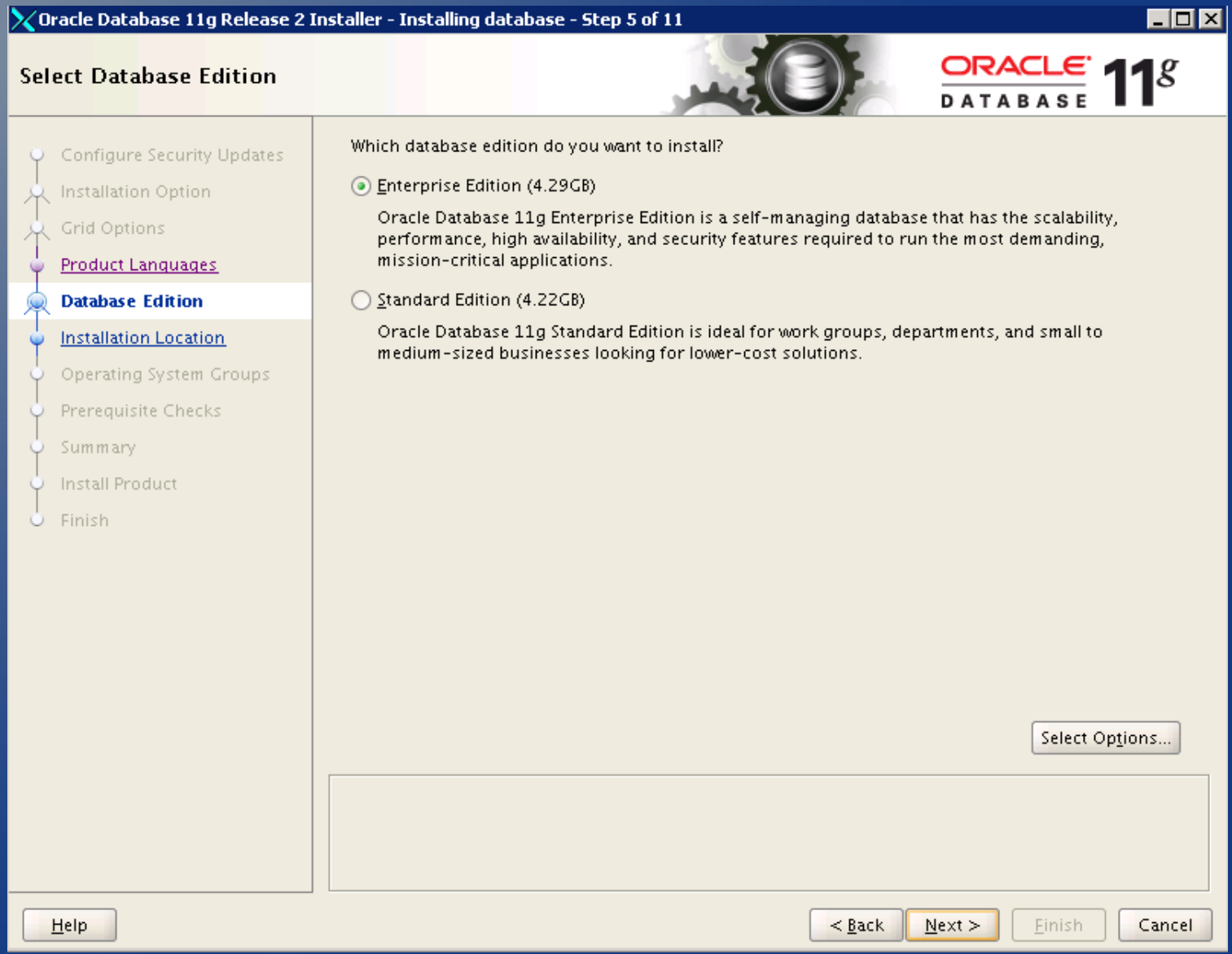
Installation – Install database binaries

- select language support



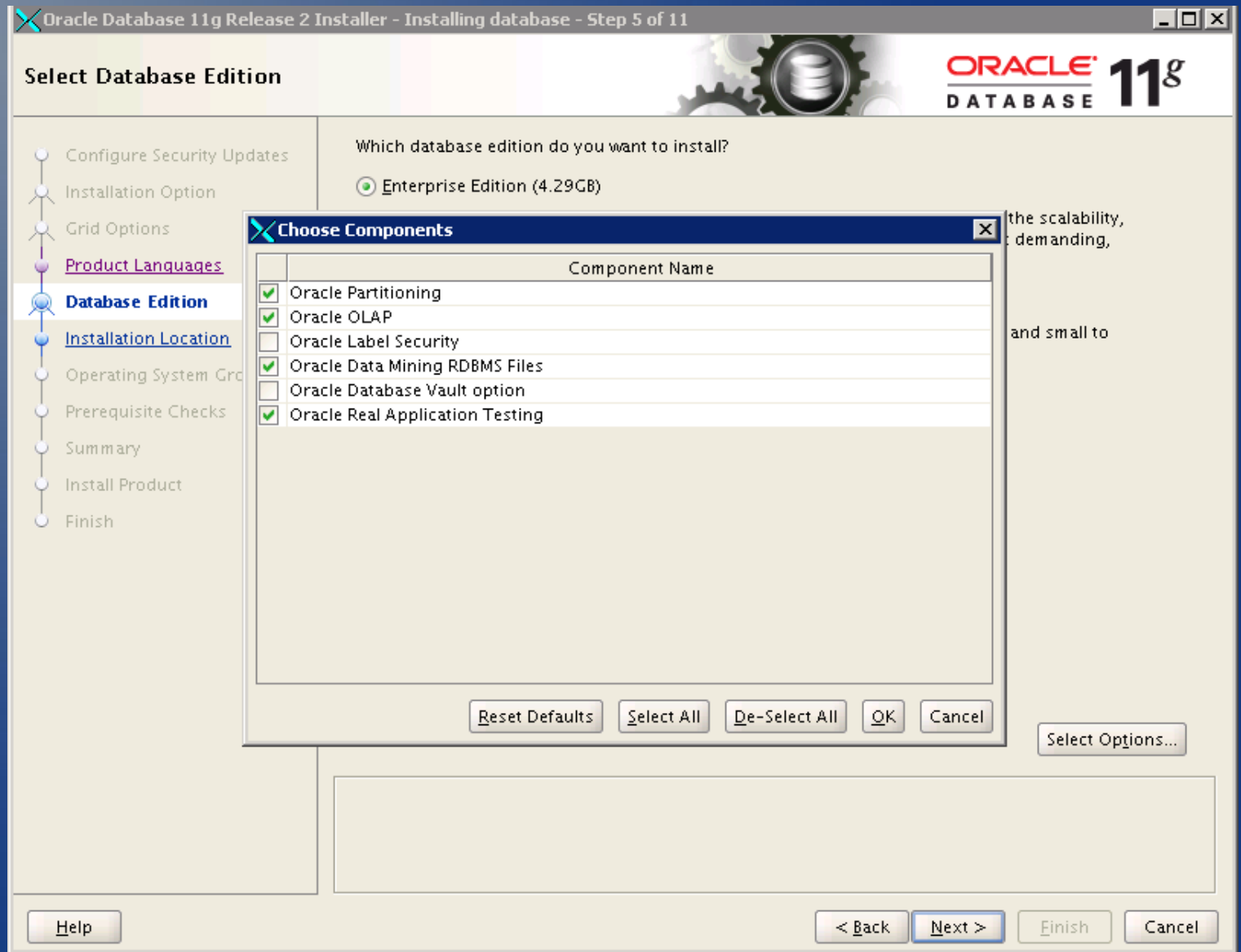
Installation – Install database binaries

- select edition



Installation – Install database binaries

- select edition and options



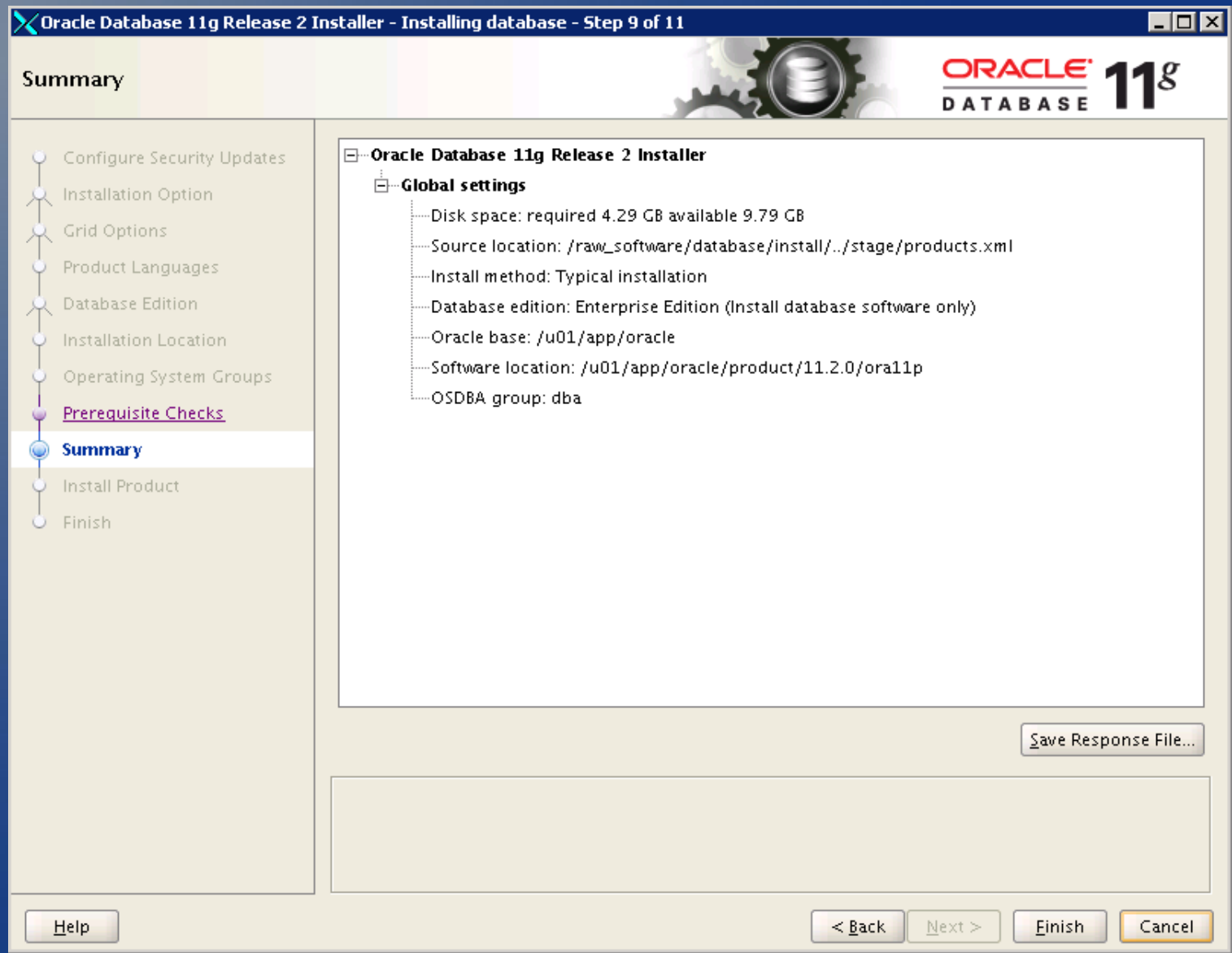
Installation – Install database binaries

- select ORACLE_BASE and ORACLE_HOME; we set it in profile file so it is entered by the installer automatically

Installation – Install database binaries

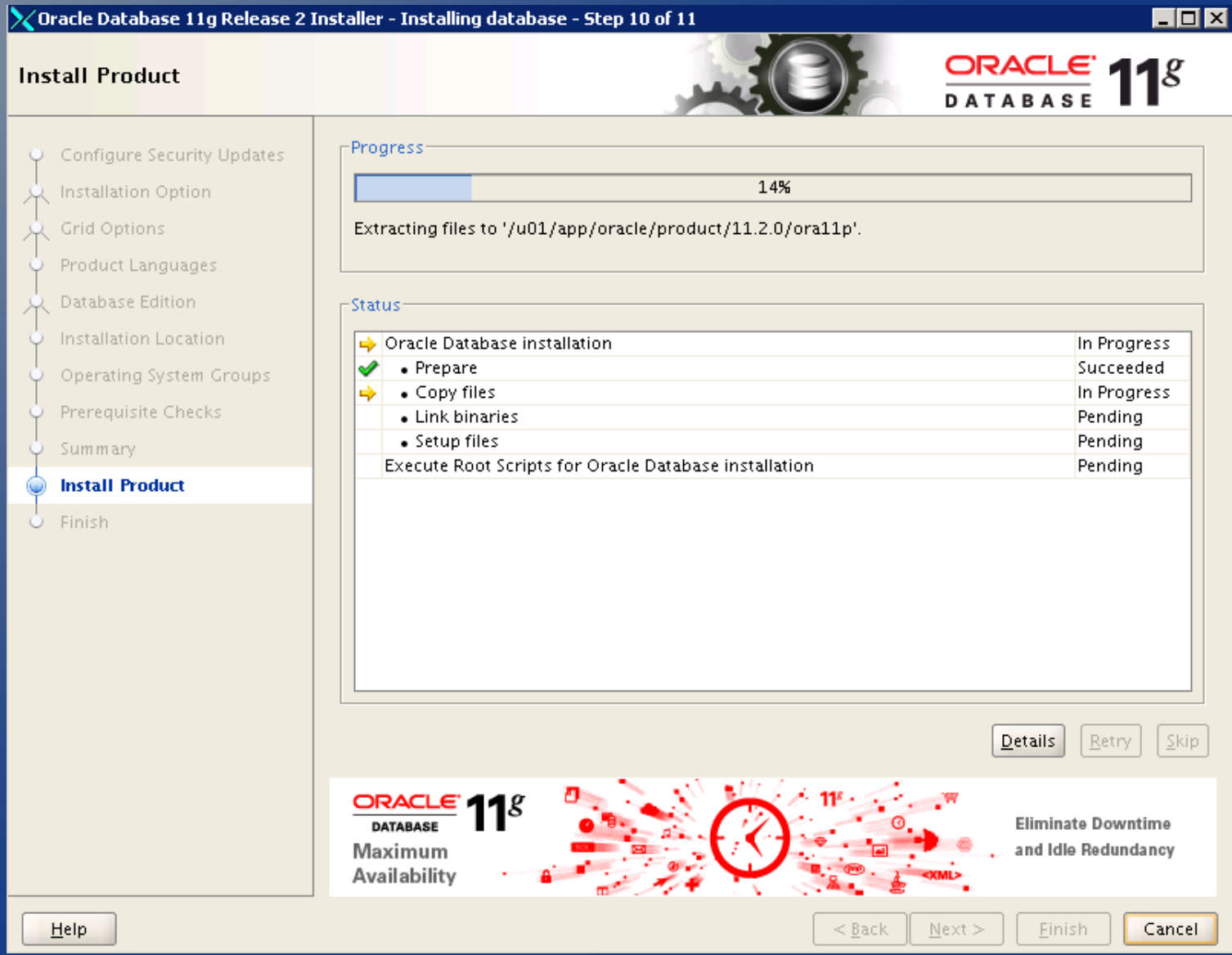
Installation – Install database binaries

- Installation Summary



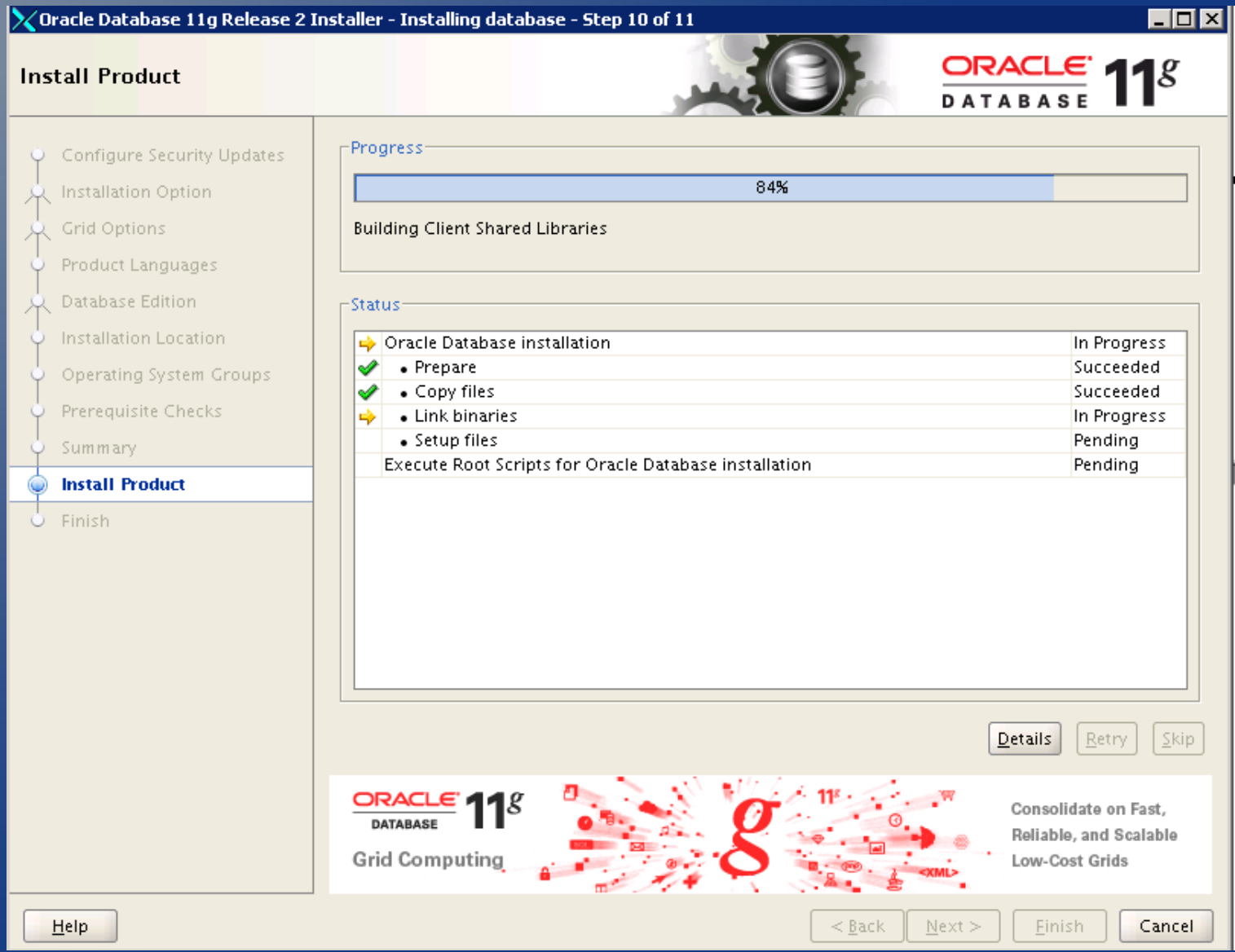
Installation – Install database binaries

- Install process



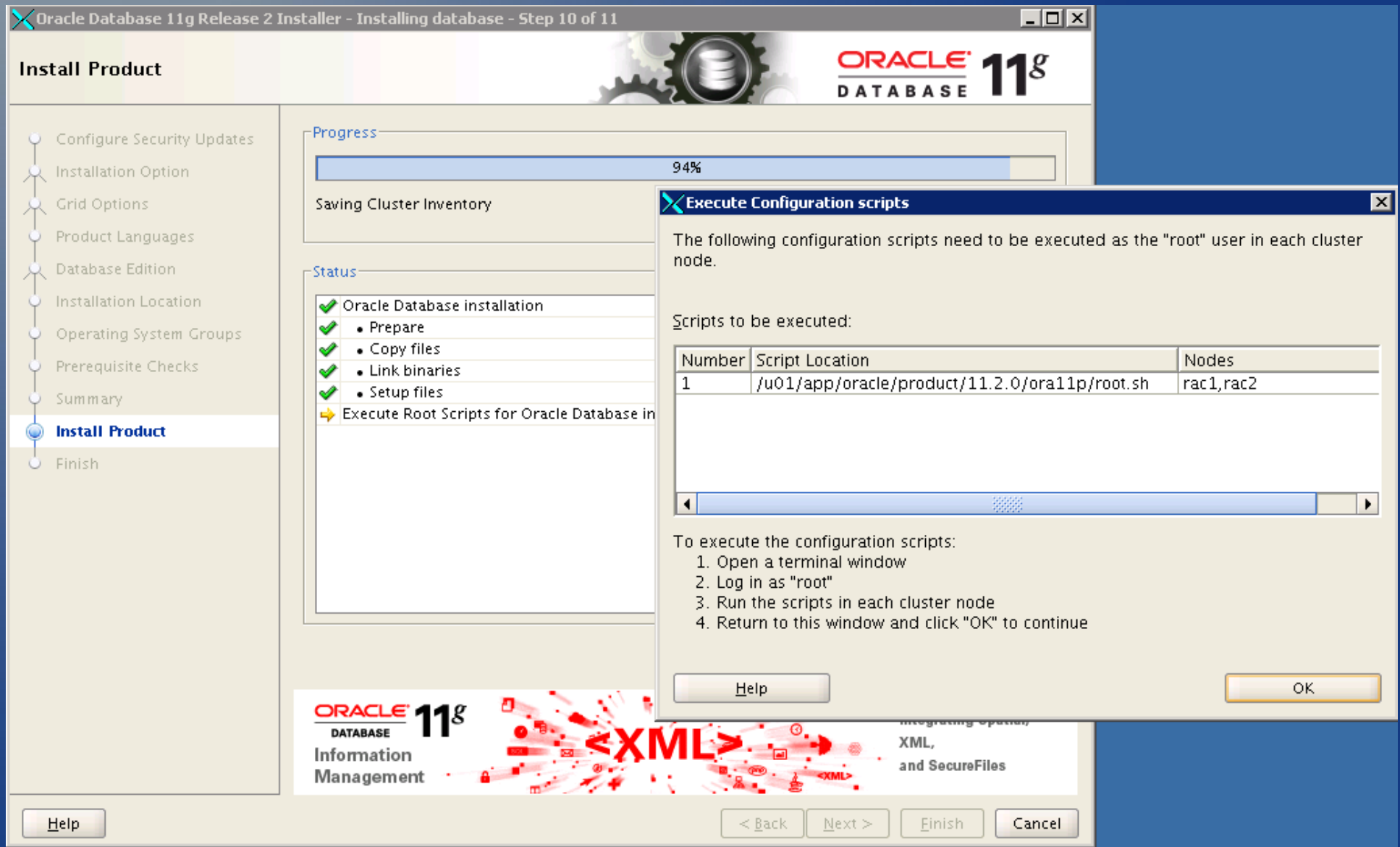
Installation – Install database binaries

- Install process (con´t)



Installation – Install database binaries

- Installation nearly finished.. just start „root.sh“ as root on all nodes



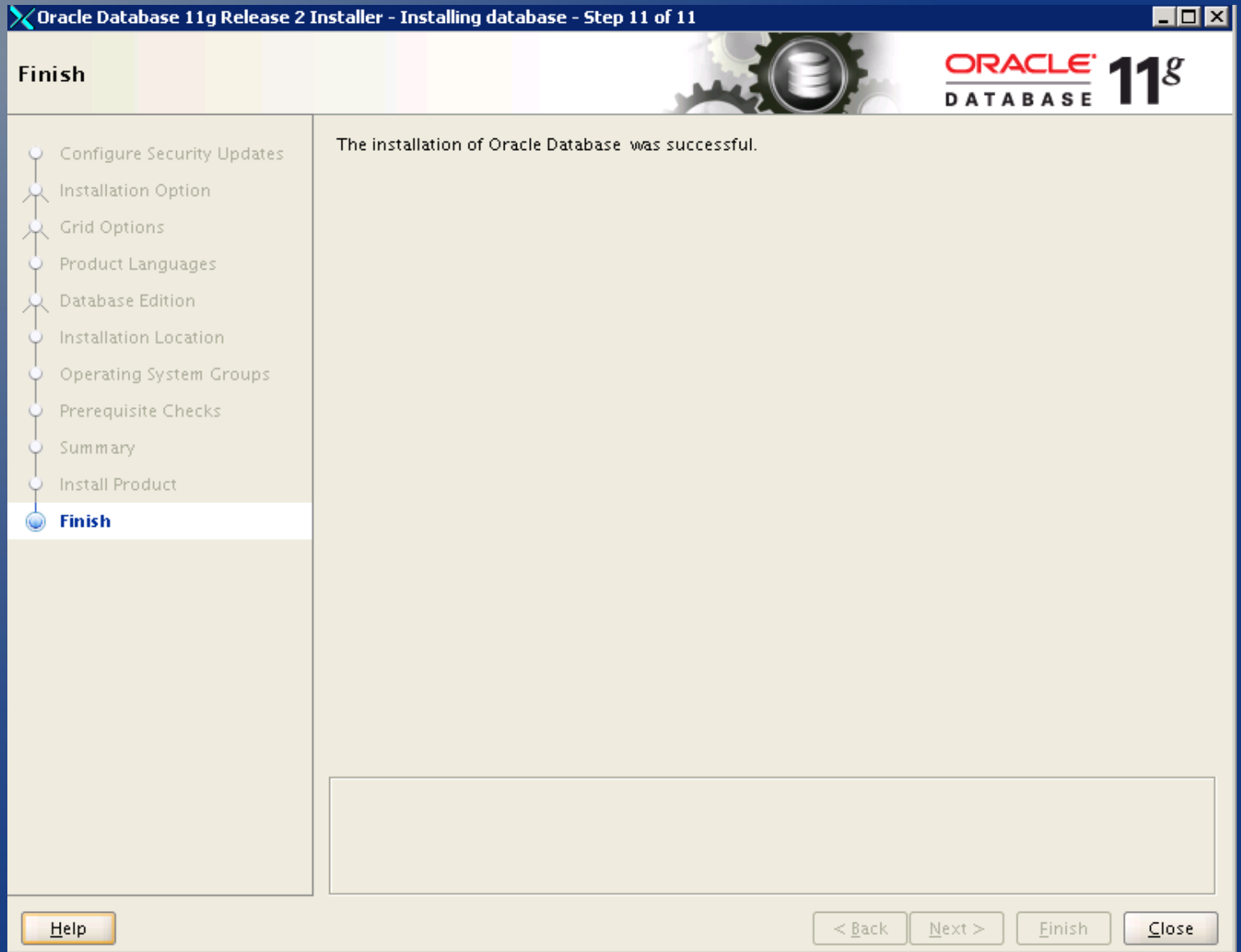
Installation – Install database binaries

- root.sh sample output

```
[root@rac2 ~]# /u01/app/oracle/product/11.2.0/ora11p/root.sh
Running Oracle 11g root.sh script...

The following environment variables are set as:
    ORACLE_OWNER= ora11p
    ORACLE_HOME=  /u01/app/oracle/product/11.2.0/ora11p

Enter the full pathname of the local bin directory: [/usr/local/bin]:
The file "dbhome" already exists in /usr/local/bin.  Overwrite it? (y/n)
[n]:
The file "oraenv" already exists in /usr/local/bin.  Overwrite it? (y/n)
[n]:
The file "coraenv" already exists in /usr/local/bin.  Overwrite it? (y/n)
[n]:

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.sh script.
Now product-specific root actions will be performed.
Finished product-specific root actions.
[root@rac2 ~]#
```

Installation – Install database binaries

- Finished



Installation – binary installation of grid infra

- Where to go now?
  - We just installed all neccessary components for creating our first RAC database → this will be the next chapter
  - Backup current configuration

Part 5 –
Creation of a RAC database

Create a RAC database - Preparations

- Based upon previous chapters (especially grid infrastructure installation, advm/acfs, rac binary installation)

- Oracle database and grid infrastructure binaries are installed
- at least ONE or better two disk groups configured for:
  - database and binary files
  - flash recovery area

- desired database name: ORA11P

Create a RAC database – with dbua

- Preparations
  - create directory for cfgtools

```
create /u01/app/oracle/cfgtoollogs
mkdir -p /u01/app/oracle/cfgtoollogs
chown root:dba /u01/app/oracle/cfgtoollogs
chmod 775 /u01/app/oracle/cfgtoollogs
```

Create a RAC database – with dbua

- Preparations
  - Check SSH equivalence once again – execute as user „ora11p“ (this is the user holding the oracle database binary installation):

    /usr/bin/ssh -o FallBackToRsh=no -o PasswordAuthentication=no -o StrictHostKeyChecking=yes -o NumberOfPasswordPrompts=0 rac2 /bin/date

    /usr/bin/ssh -o FallBackToRsh=no -o PasswordAuthentication=no -o StrictHostKeyChecking=yes -o NumberOfPasswordPrompts=0 rac1 /bin/date

    Both commands should return the current date.

Create a RAC database – with dbua

- Preparations
  - If there are errors like „connection refused“, check:
    - network connectivity between both nodes
    - does a „normal“ SSH connection works?:

      ssh root@rac1-priv
      ssh root@rac1
      ssh root@rac2-priv
      ssh root@rac2

      both commands should prompt for a password

    - If commands above prompt for password (they will most likely) something happend to the SSH equivalence configured by the installer

Create a RAC database – with dbua

- Preparations
  - here is how to re-enable SSH equivalence:

    <PATH_TO_DATABASE_INSTALLATION>/sshsetup/sshUserSetup.sh -user ora11p -hosts rac1.regner.de rac2.regner.de -shared

    Arguments:
    -user: the name of the user for ssh equivalence to be configured
    -hosts: space separated list of node names
    -shared: indicates a shared oracle home (on acfs or nfs)

Create a RAC database – with dbua

- Preparations
  - add ACFS oracle home to cluster registry

    $GRID_HOME/bin/srvctl add filesystem -d /dev/asm/ora11p_home-132 -v ora11p_home -g DATA2 -m /u01/app/oracle/product/11.2.0/ora11p -u ora11p

    Arguments:
    -d: path of advm volume
    -v: name of volume
    -m: mount point path

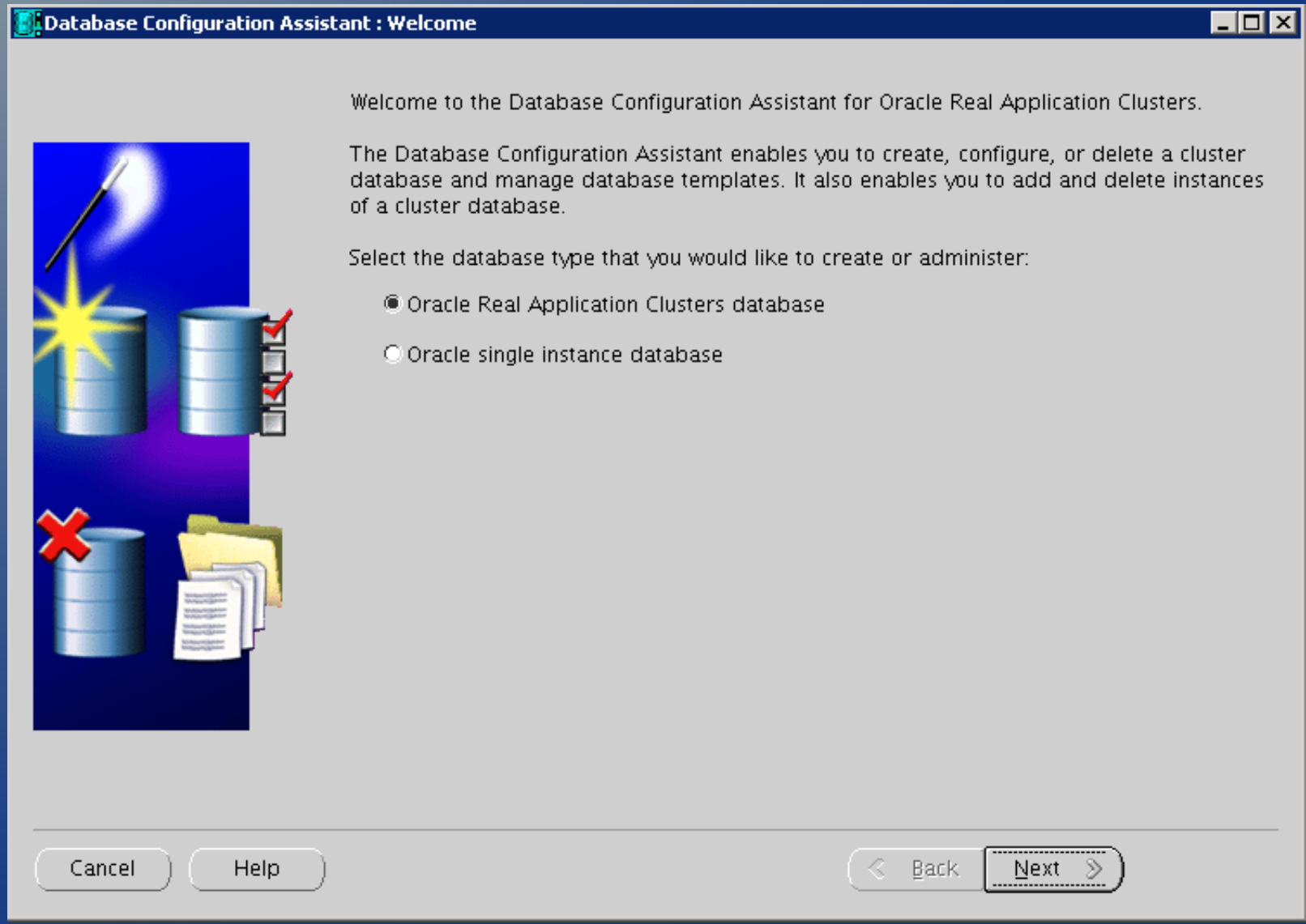
    „-u ora11p“ ist important and must match the owner of the database binary installation; else dbca will raise error compaining about missing permissions

Create a RAC database – with dbua

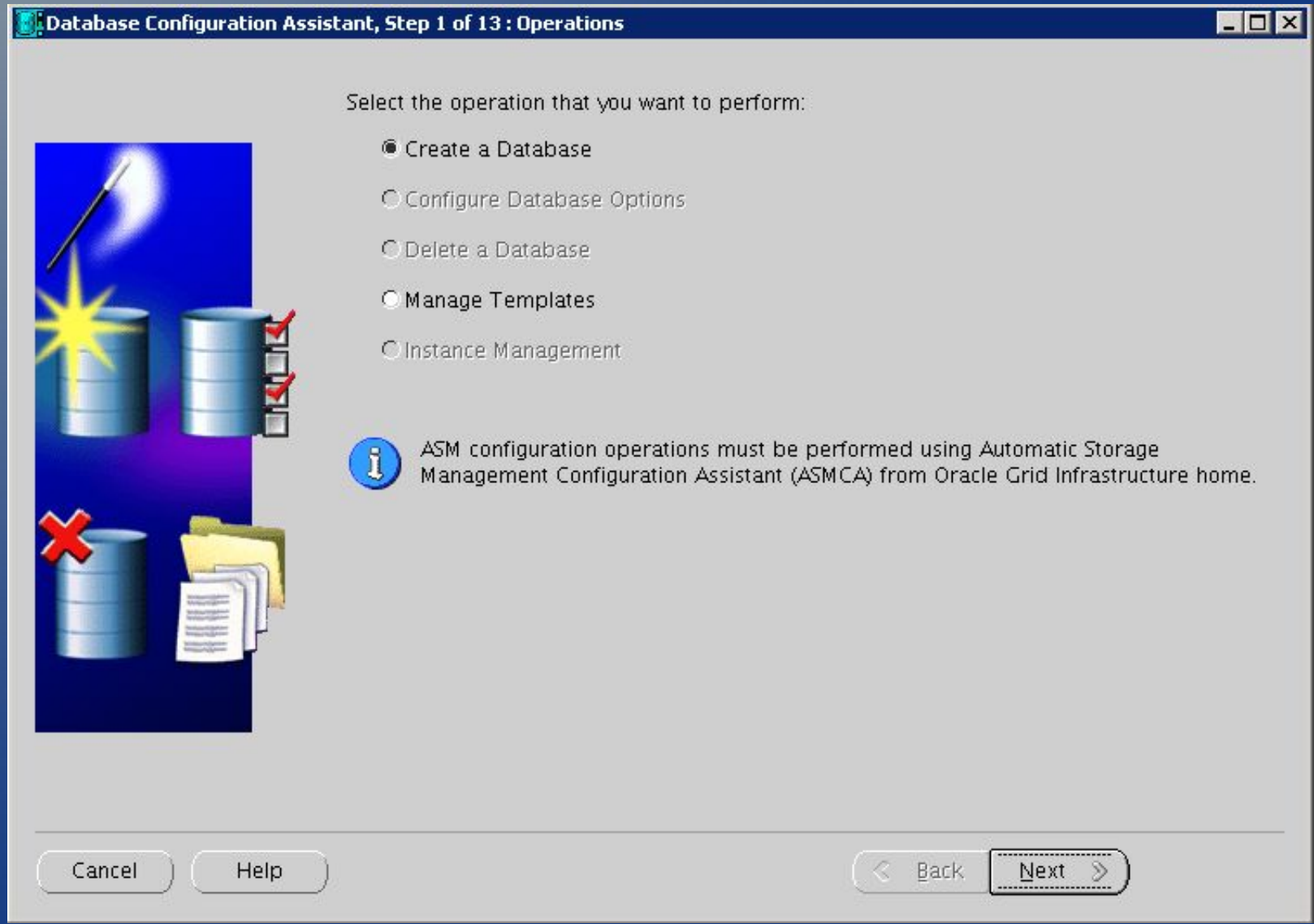
- Start dbca

```
-bash-3.2$
-bash-3.2$
-bash-3.2$ export DISPLAY=172.23.0.170:17.0
-bash-3.2$ $ORACLE_HOME/bin/dbca &
```

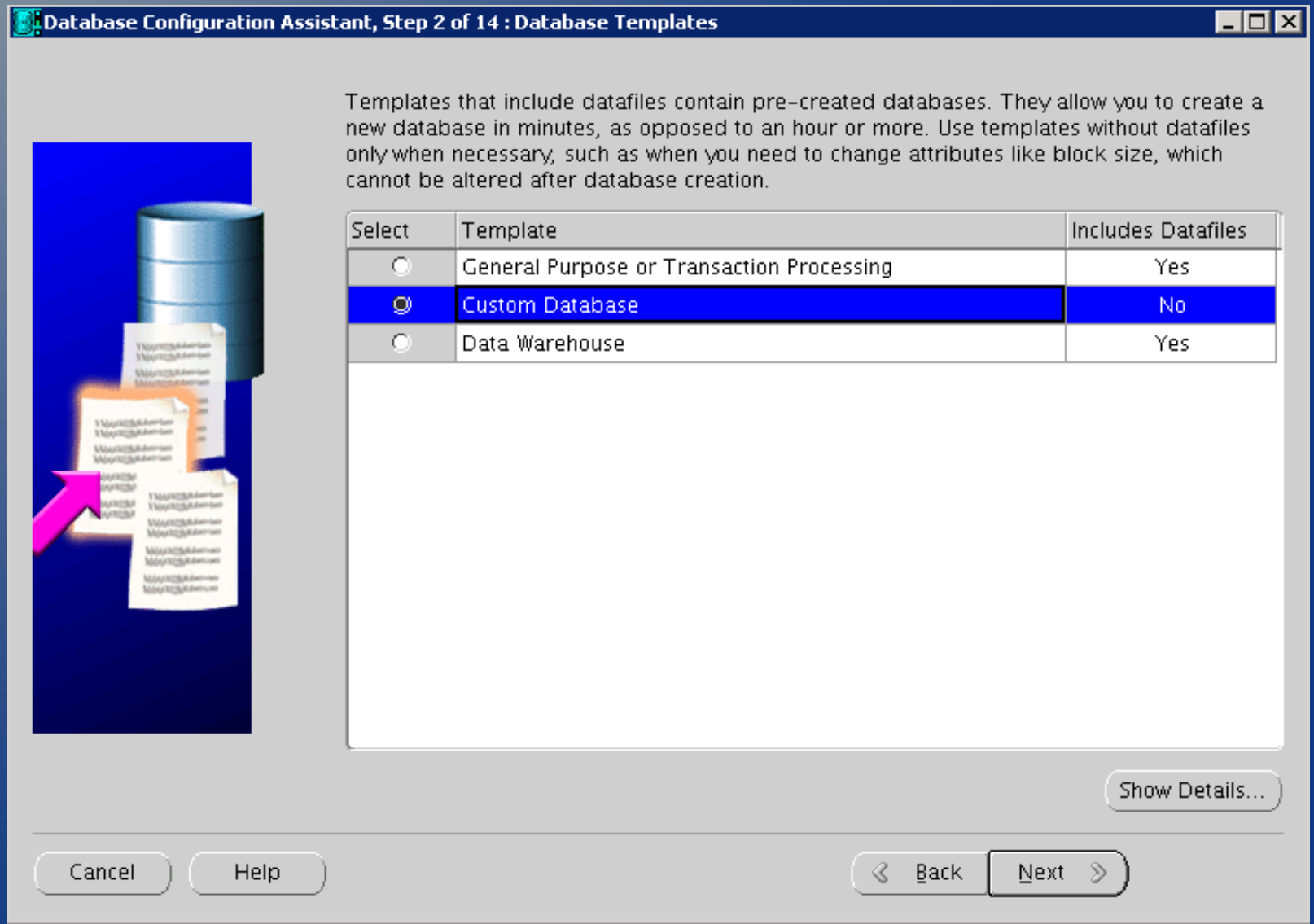
Create a RAC database – with dbua



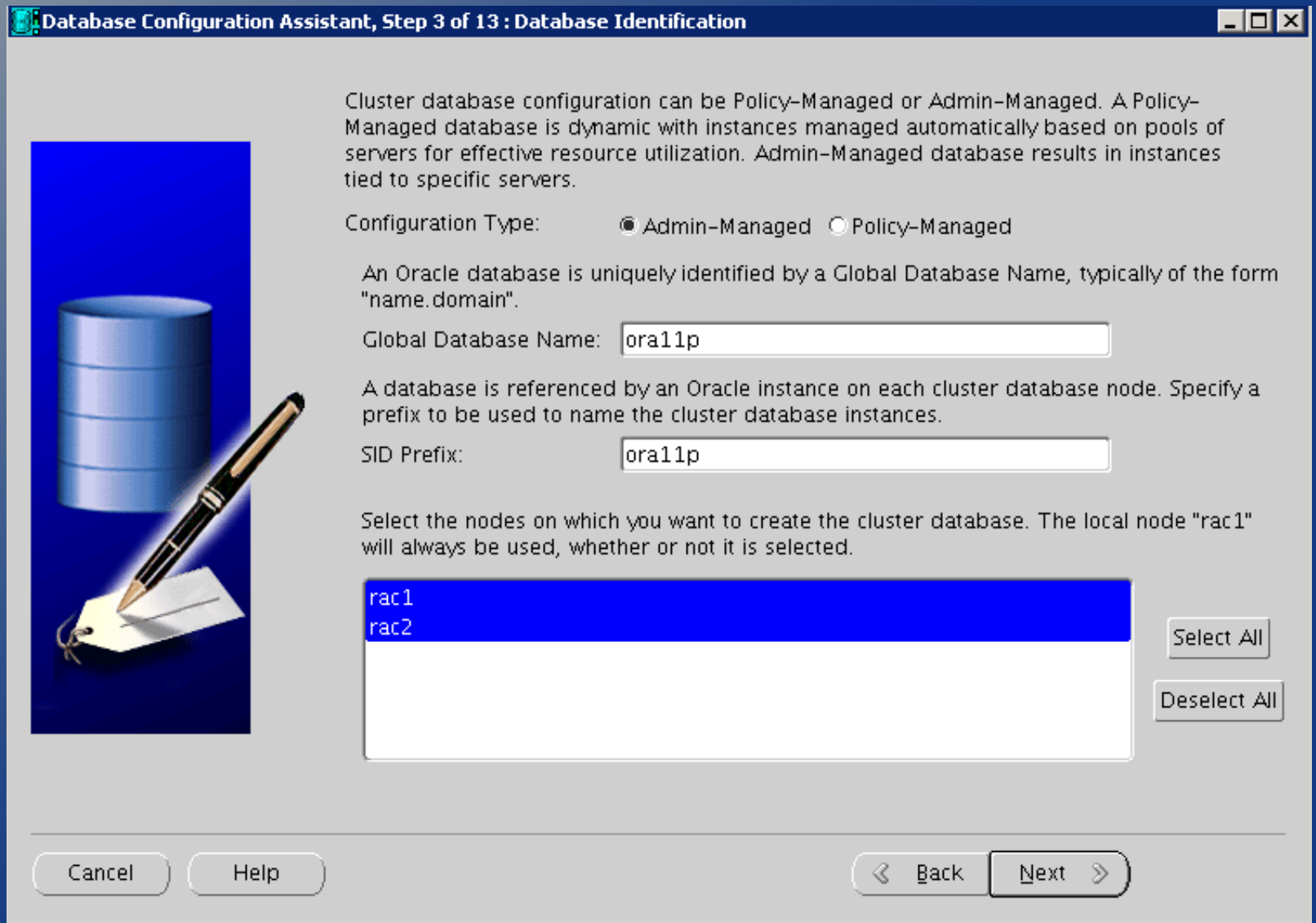
Create a RAC database – with dbua



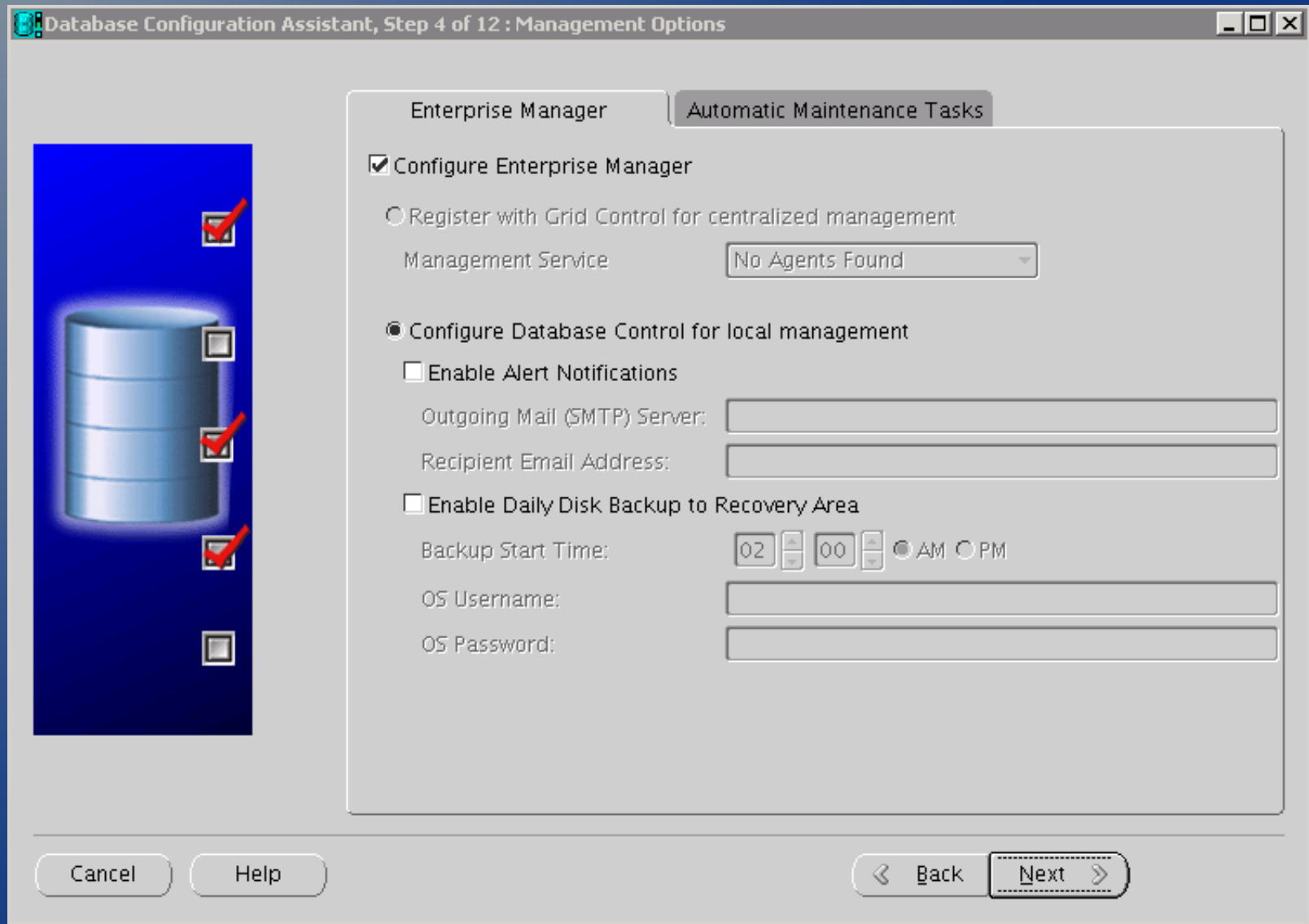
Create a RAC database – with dbua



Create a RAC database – with dbua

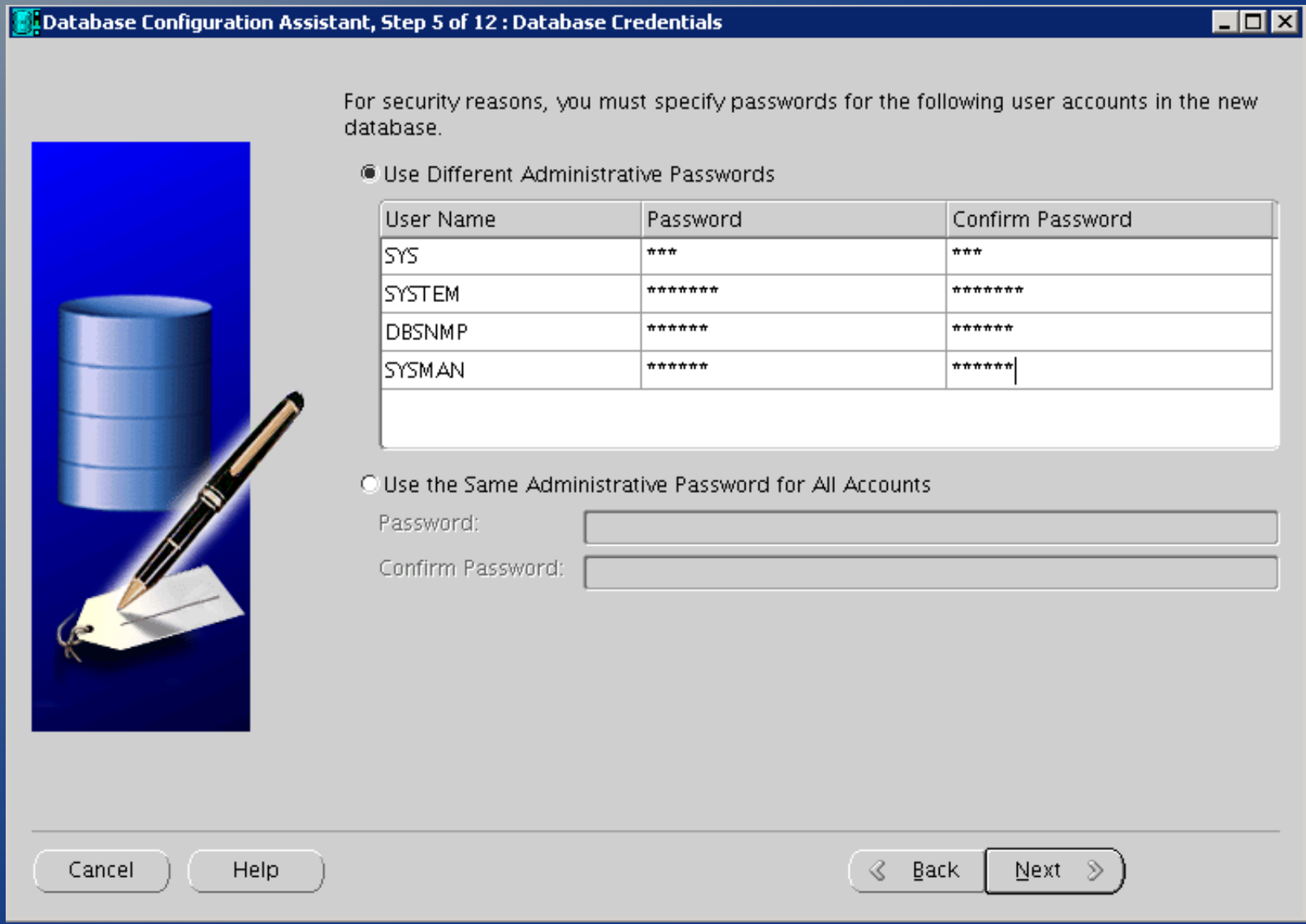


Create a RAC database – with dbua

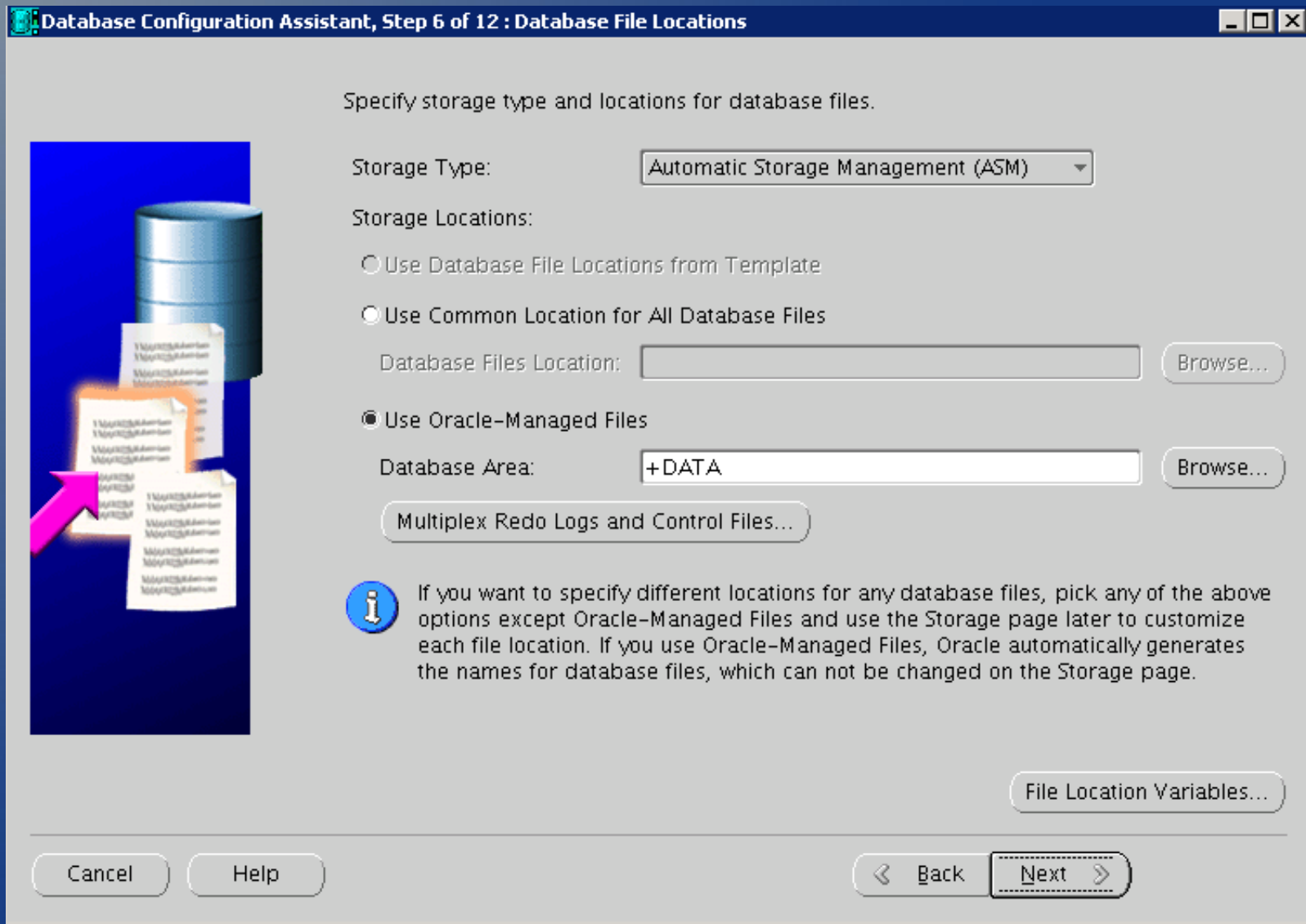


Create a RAC database – with dbua

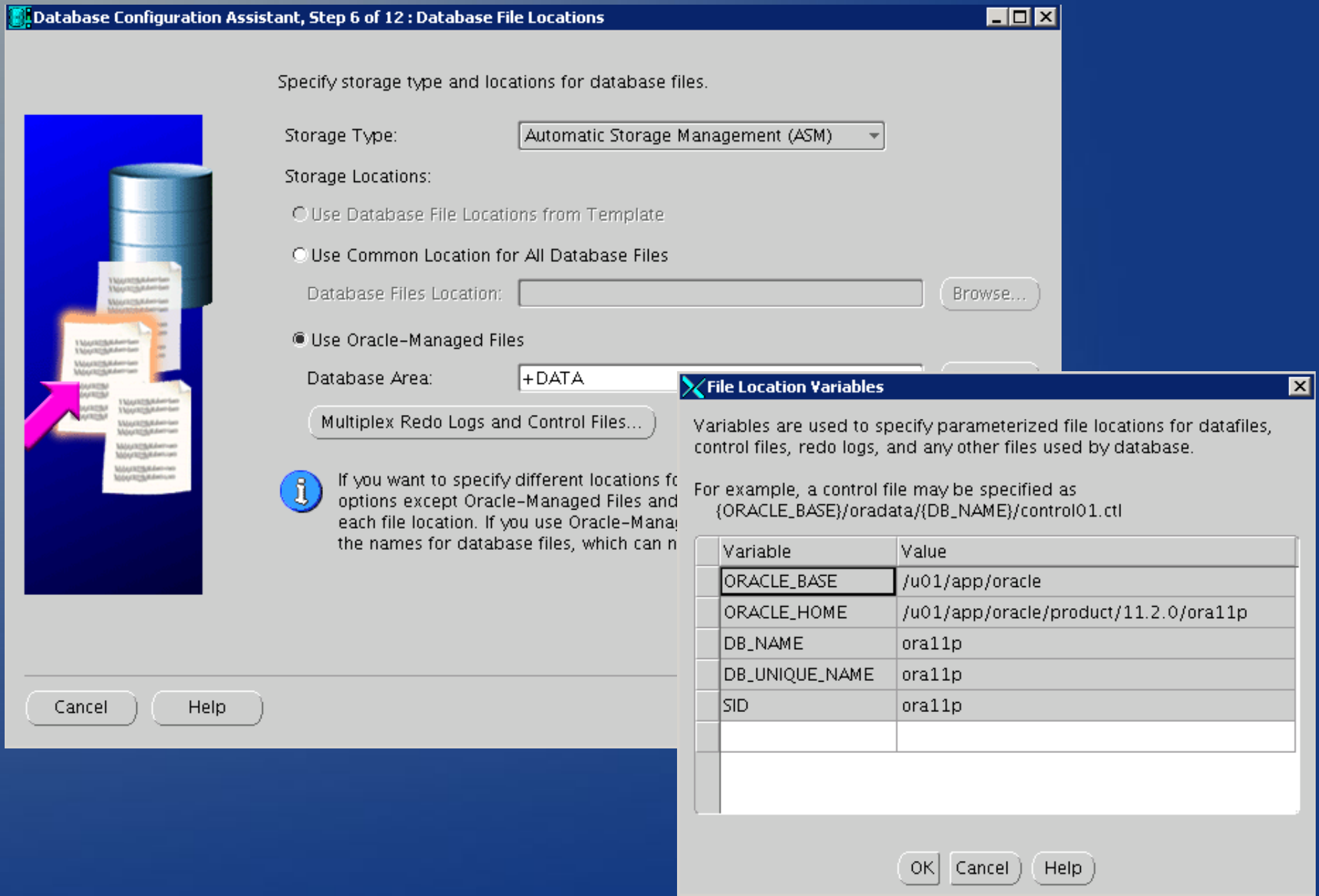
Create a RAC database – with dbua



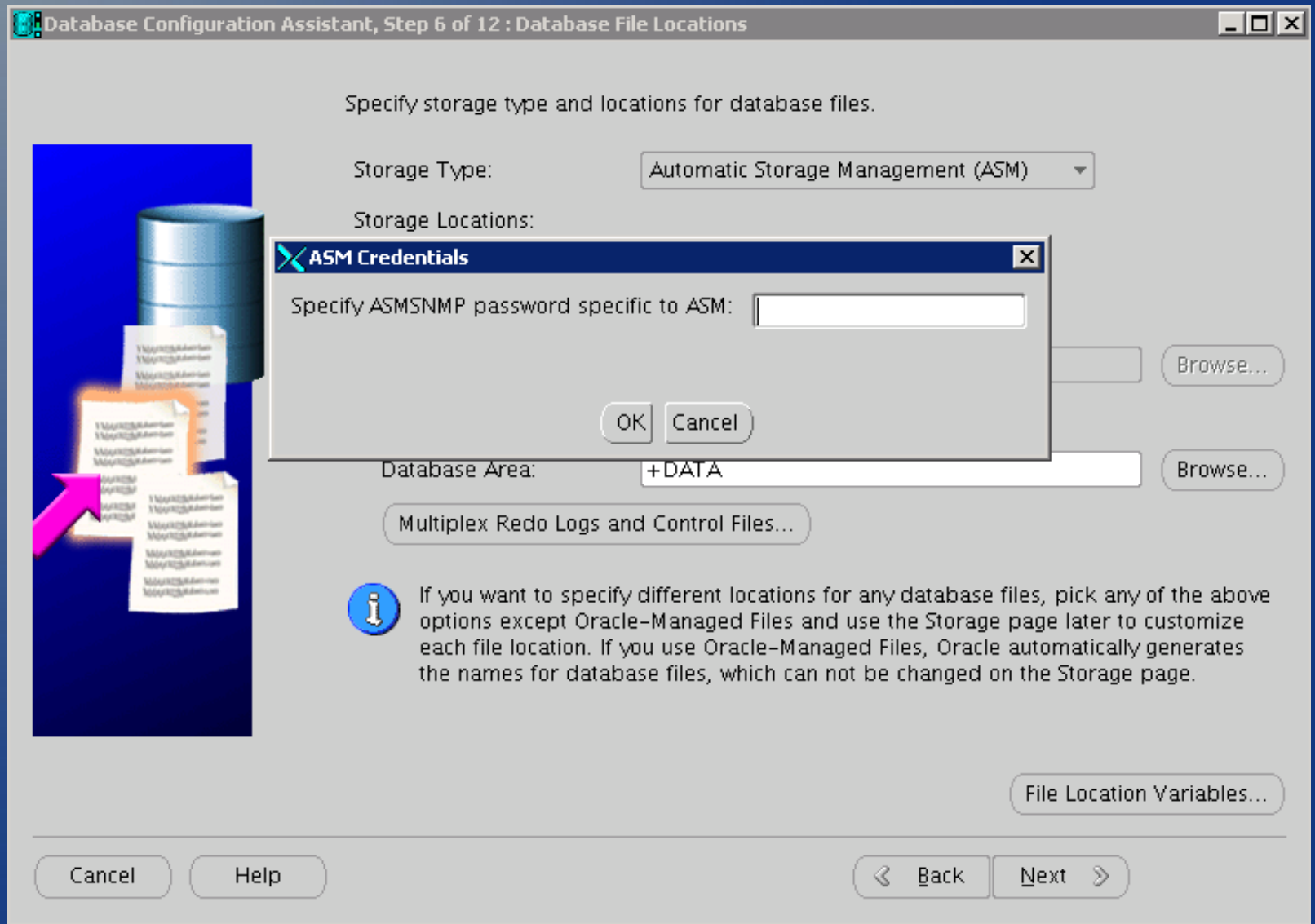
Create a RAC database – with dbua



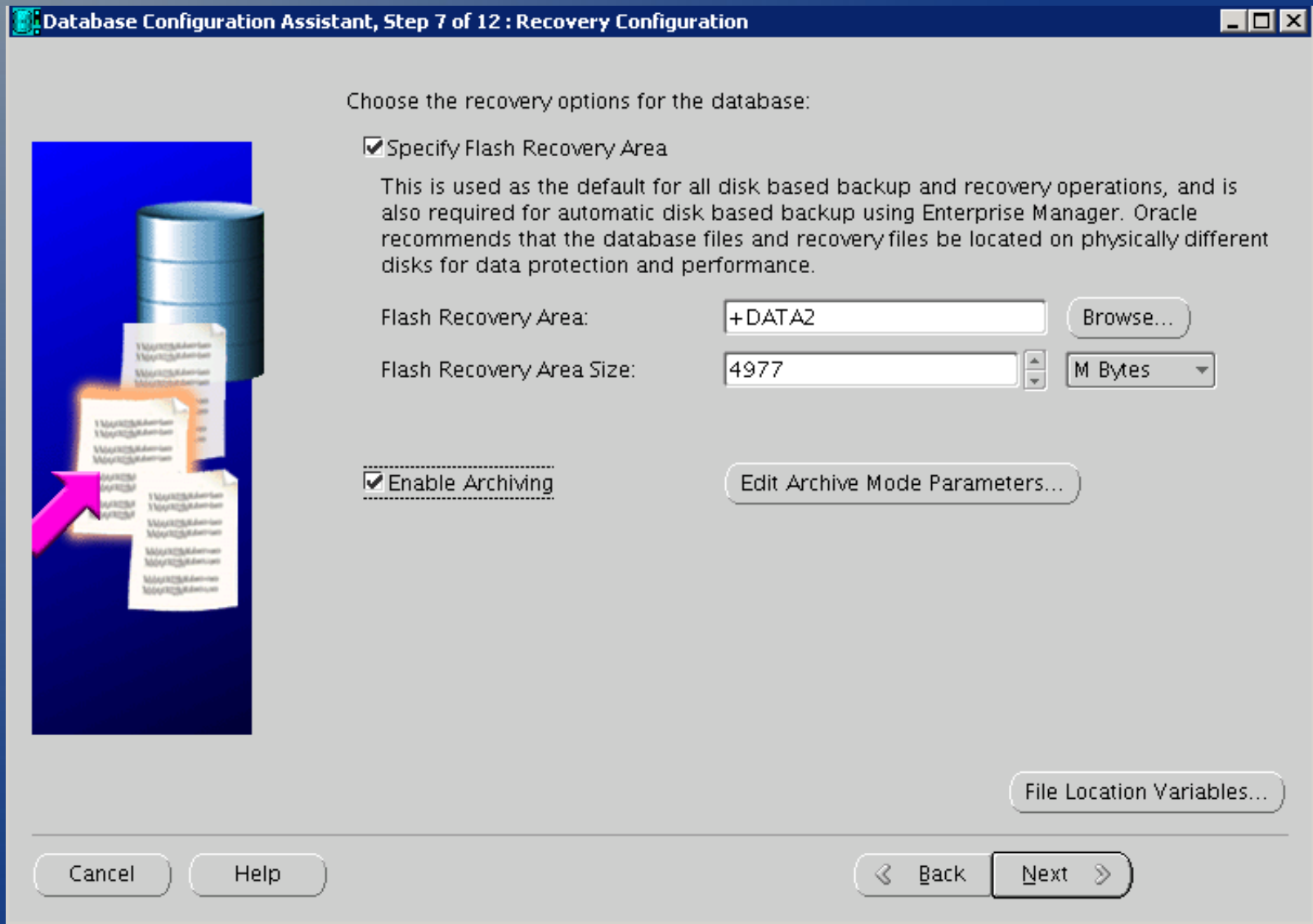
Create a RAC database – with dbua



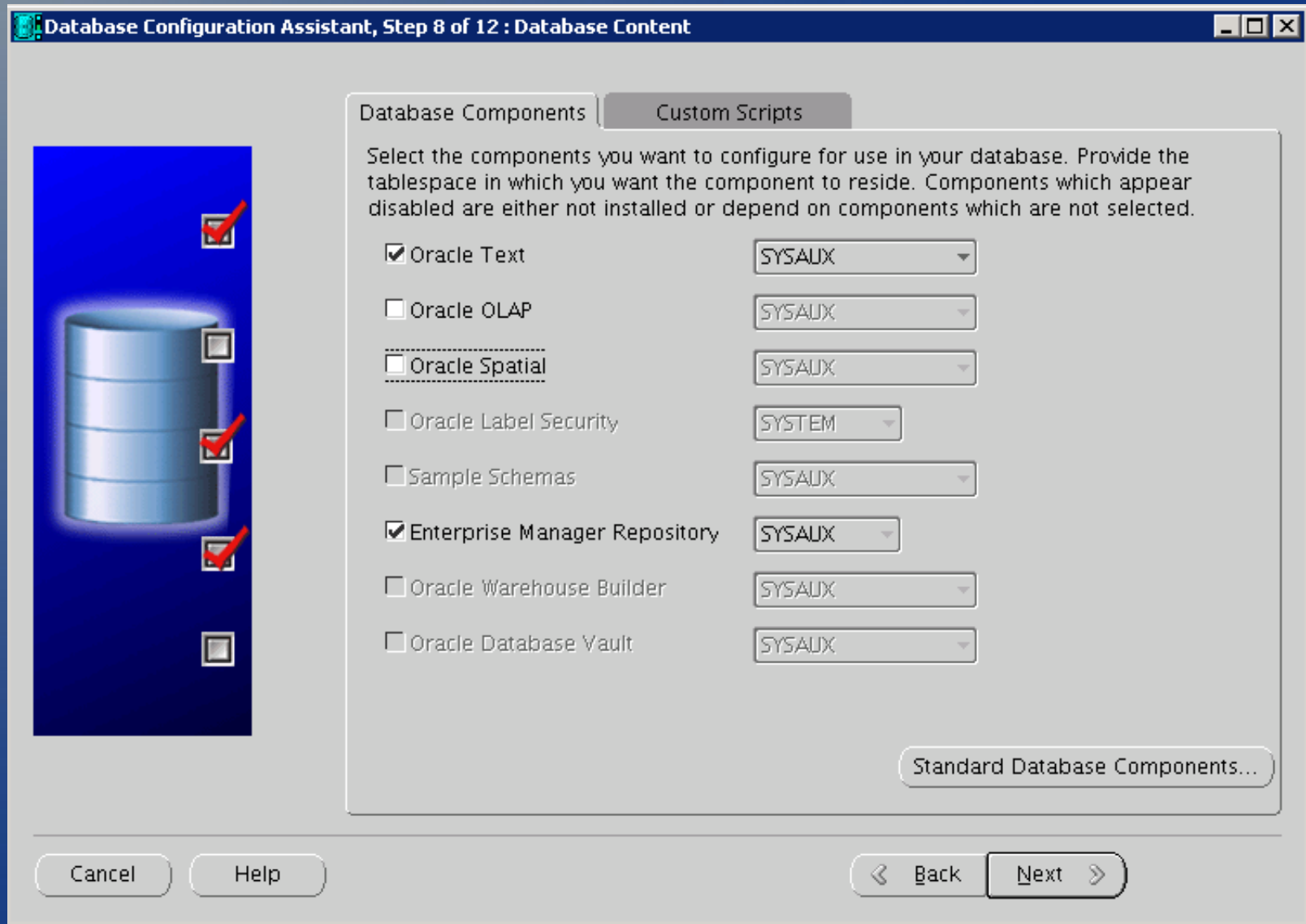
Create a RAC database – with dbua



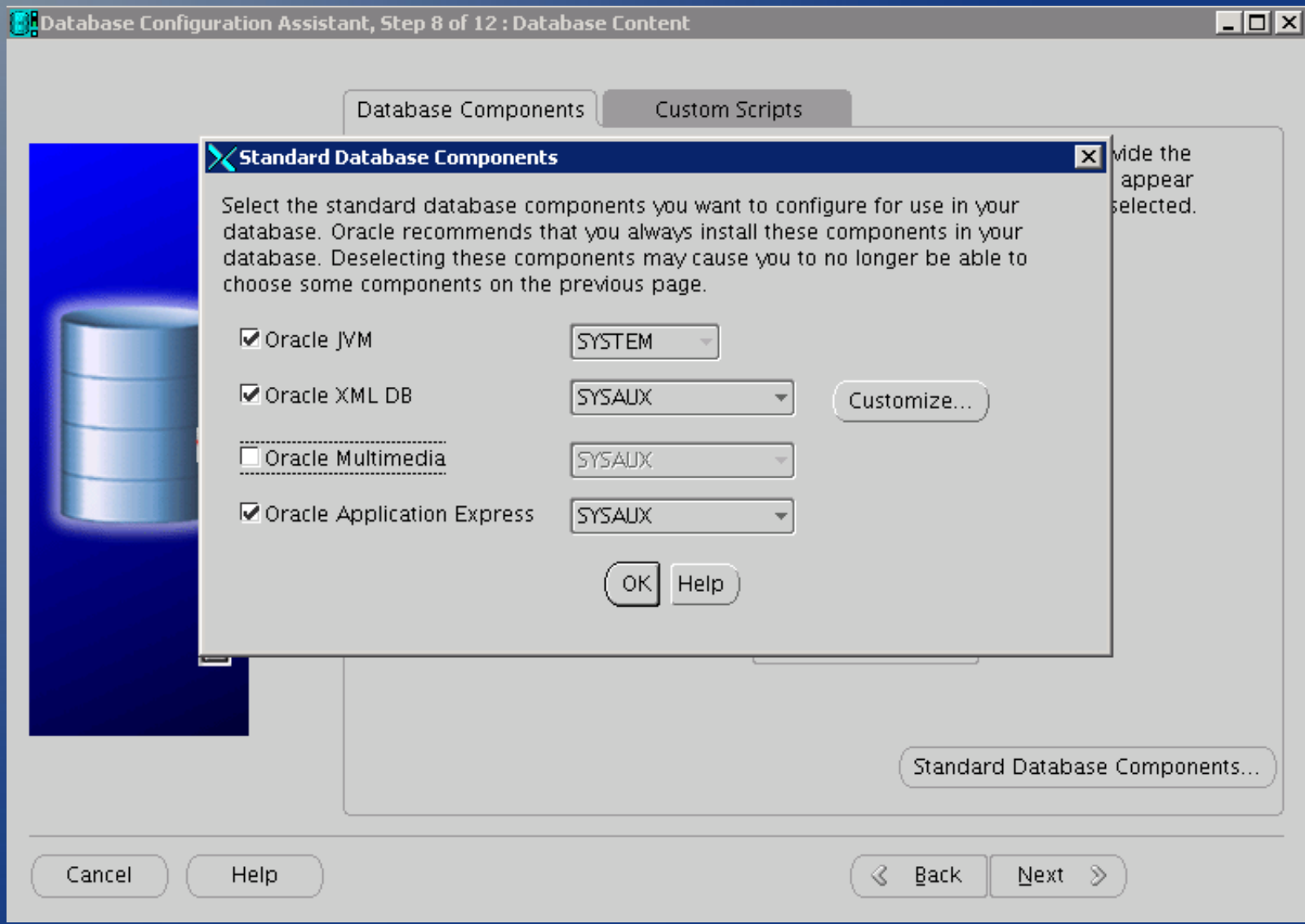
Create a RAC database – with dbua



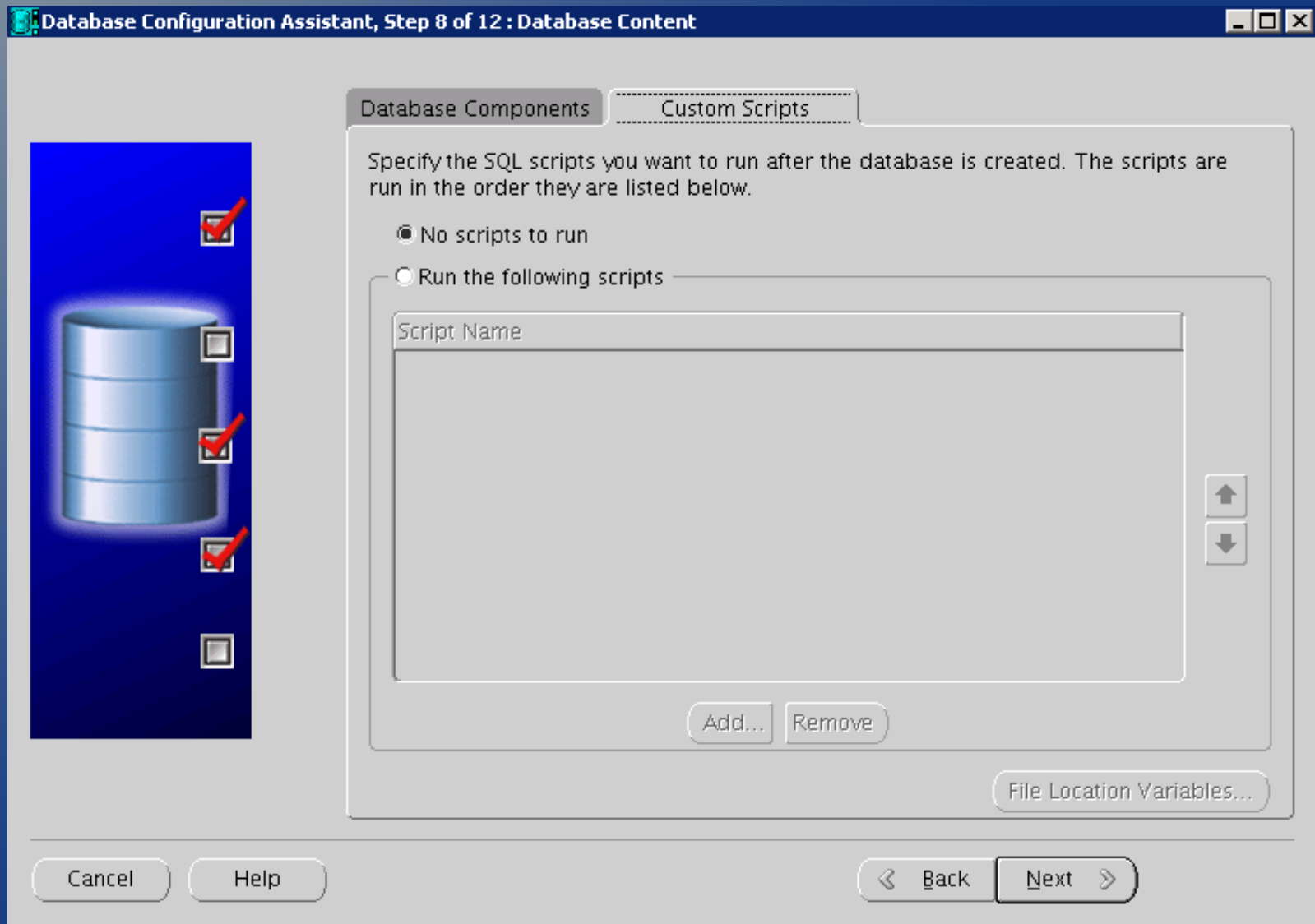
Create a RAC database – with dbua



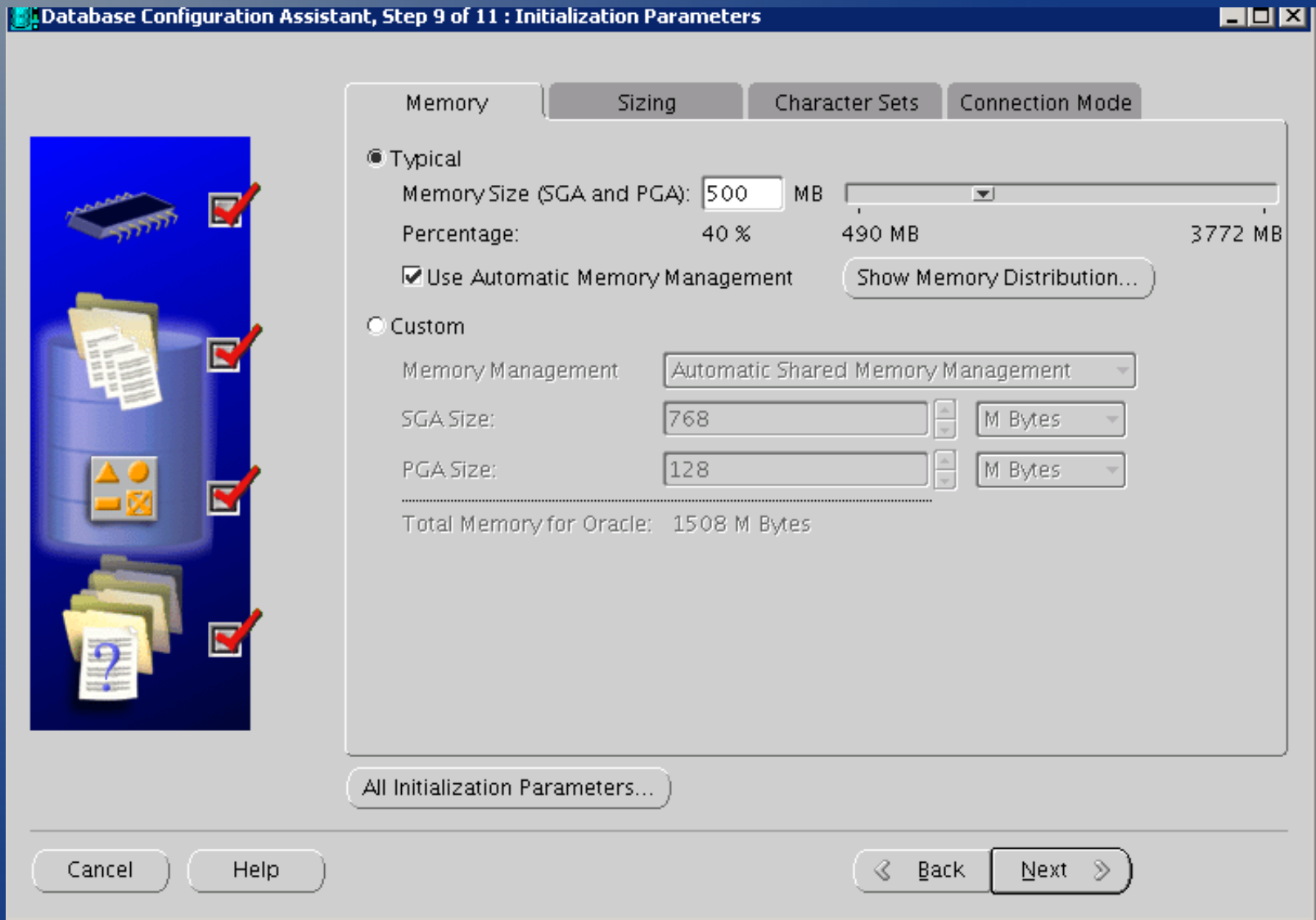
Create a RAC database – with dbua



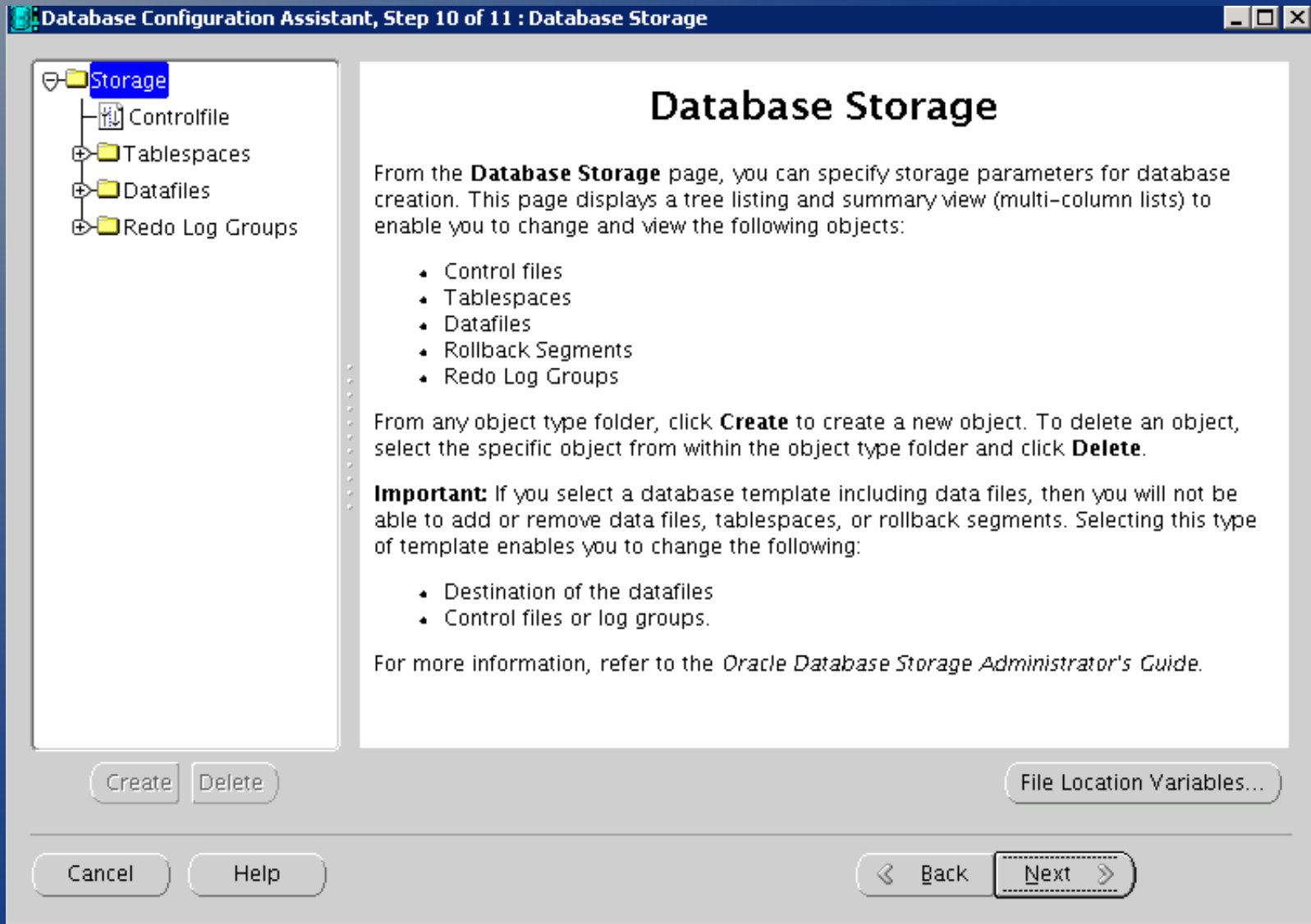
Create a RAC database – with dbua



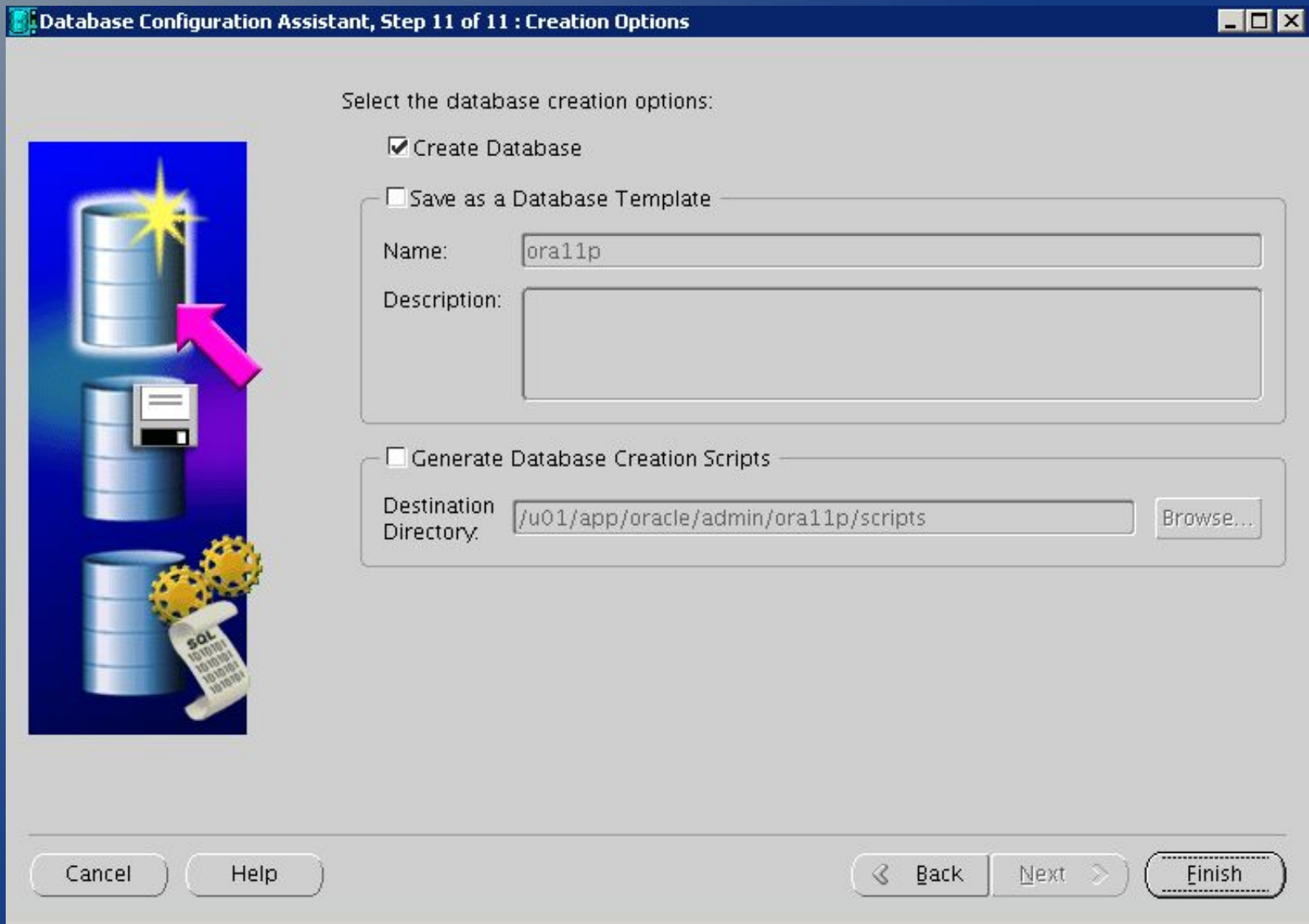
Create a RAC database – with dbua



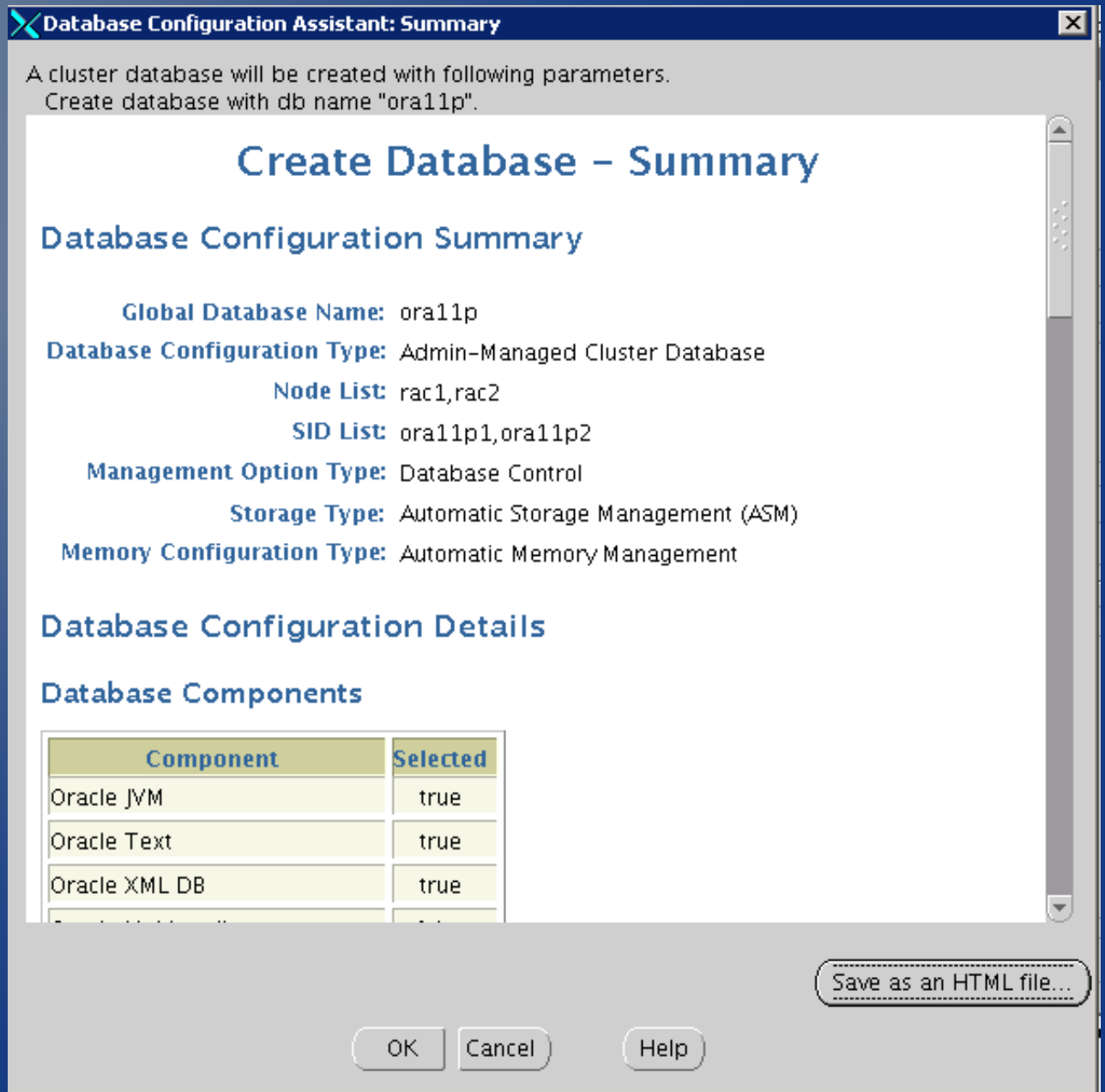
Create a RAC database – with dbua



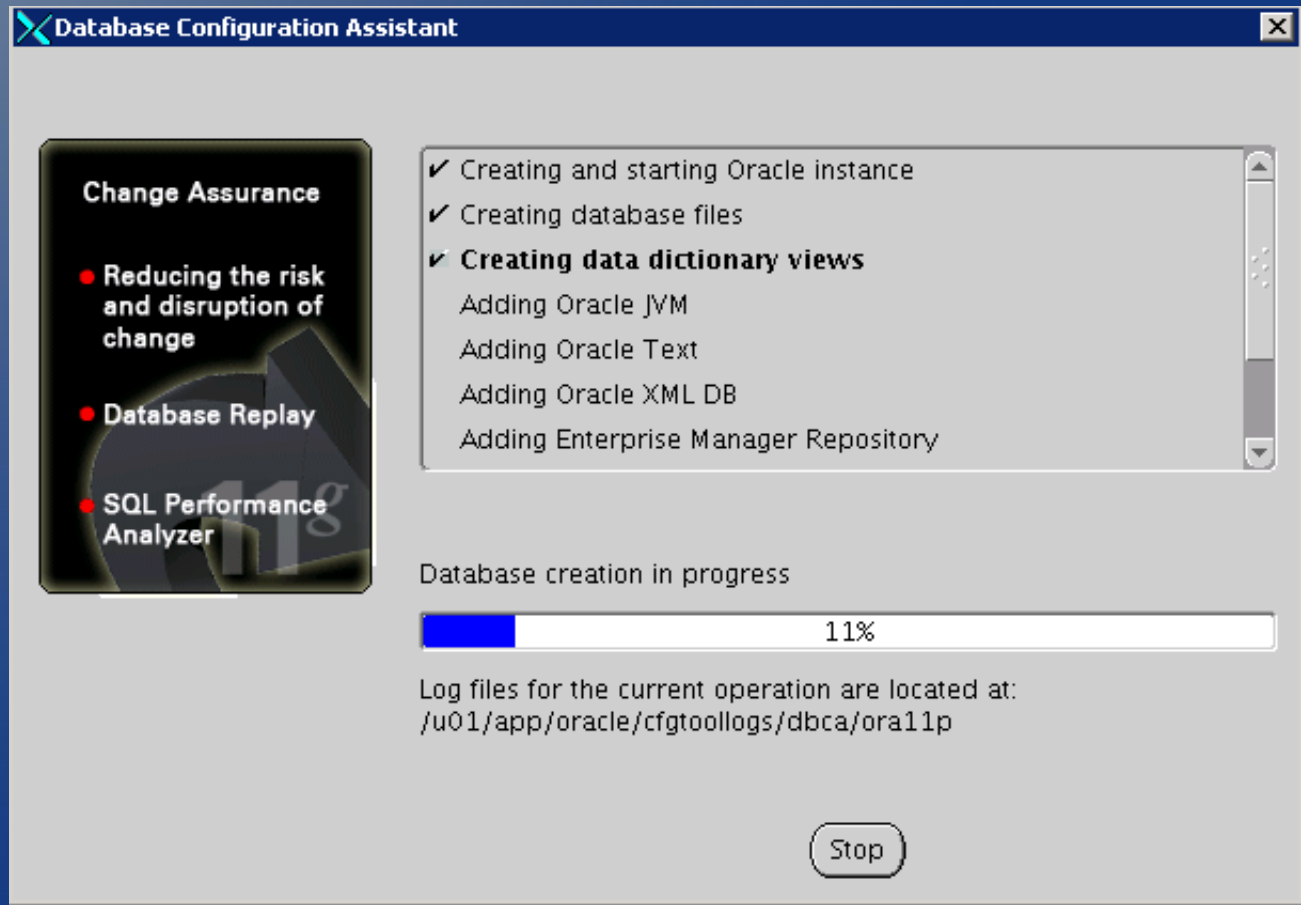
Create a RAC database – with dbua



Create a RAC database – with dbua

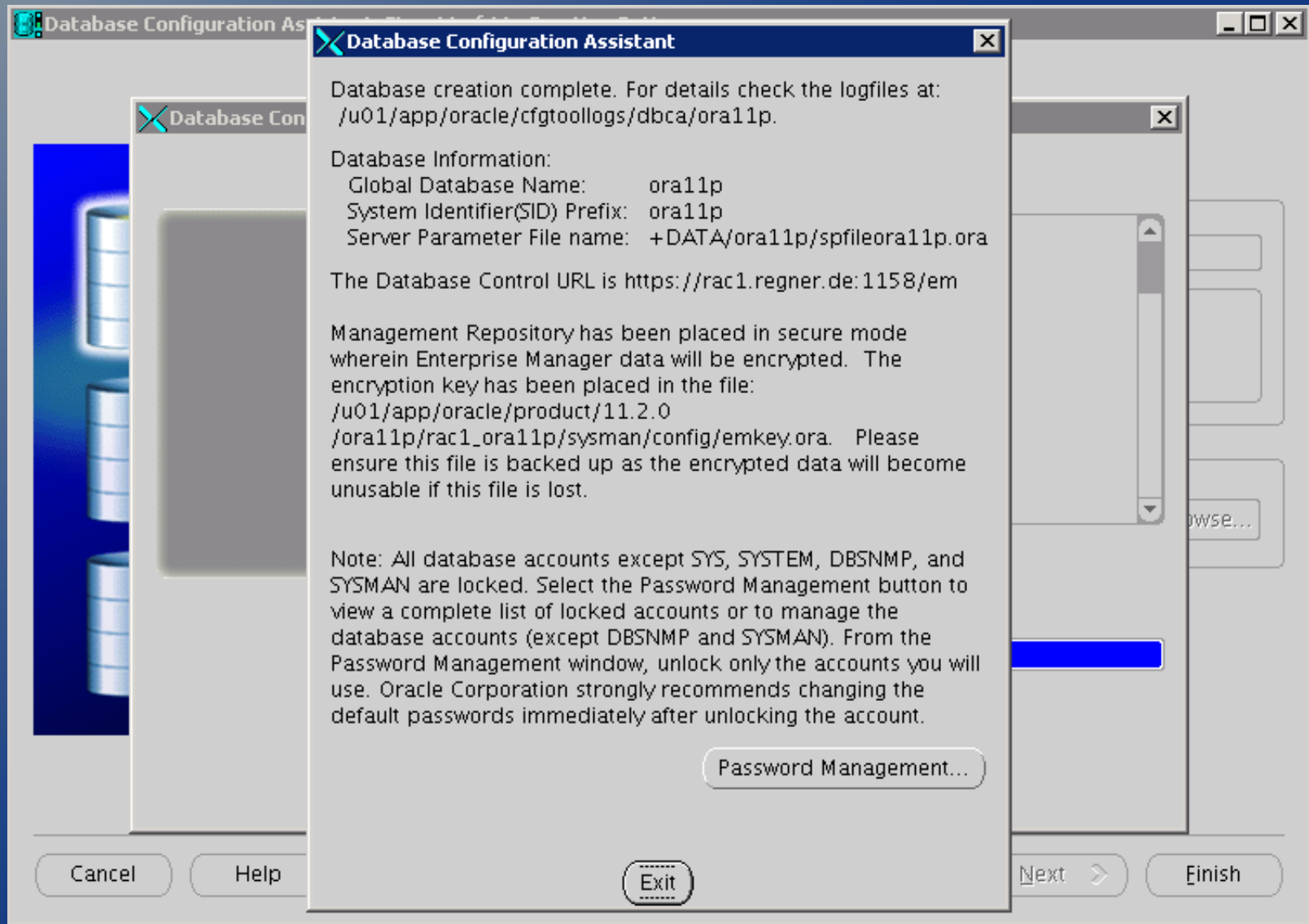


Create a RAC database – with dbua



Create a RAC database – with dbua

- Done!



Create a RAC database – with dbua

- Where to go now?
  - Tune Instance, for instance:
    - memory_target
    - db_writer_processes
    - sessions parameter
    - …

Appendix

Sources

- General documentation
  - Oracle Database 11g R2 book index
    http://www.oracle.com/pls/db112/homepage
  - Database Installation Guide for Linux
    http://download.oracle.com/docs/cd/E11882_01/install.112/e10840/toc.htm

- Downloads
  - ASM (asmlib and kernel modules)
    http://www.oracle.com/technology/tech/linux/asmlib/index.html

    Note: If you do not have access to the unbreakable linux update repository you can take the RPMs for Red Hat Enterprise Server 5... just make sure it matches your kernel version!
  - Oracle Enterprise Linux (OEL)
    http://edelivery.oracle.com/linux

Further questions?

Suggestions?

Comments?

Need help?

Send me an email:
ronnyegner@gmx.de