Installation Instructions for

Oracle 8.1.7

on

SuSE Linux Enterprise Server 8[(*)]

and United Linux 1.0

(**) May be used for Oracle 8.1.7 on other SuSE Linux versions as well – in conjunction with the support pages at http://www.suse.com/oracle/.

SuSE Inc. ©2002

The SuSE Oracle Team <oracle@suse.com>

Table of Contents

Preface

This document describes the step−by−step installation process of Oracle 8.1.7 on the **SuSE Linux Enterprise Server 8**. For detailed Oracle instructions see the Oracle Installation Guide and the Oracle Administration Guide, both very good documents, that come with Oracle 8.1.7. For more Oracle books see http://technet.oracle.com/, where they are made available online.

The Oracle Installation Guide provides a much more comprehensive overview over **all** the possible installation options. This document on the other hand uses only one typical installation method as an example. If you want to know all the details about the Oracle installer and the Oracle tools like the Database Creation Assistant, see the Oracle documentation.

This document is based on the Oracle Installation Guide provided by Oracle. It leaves out all the things that unnecessary because it is the default in SuSE Linux Enterprise Server anyway. An example is the creation of an Oracle user and groups, or the *umask* setting for this user, these tasks are done automatically by an add−on package SuSE provides.

There are two sections: A very short, text−only one for (somewhat) experienced users, and a much longer and more detailed one. The Quick Installation section is at the end of this document.

You will find additional information online at the SuSE support pages for Oracle located at *http://www.suse.com/oracle/*

Requirements

Hardware

For a detailed description of hardware– and space requirements see the generic Oracle Installation Guide that comes with Oracle 8.1.7 for Linux.

- Any Intel– or AMD– based PC with a 32bit CPU

- At least 128 MB of RAM, more if you use the Java VM in the Oracle database

- We recommend ca. 1 GB of free disk space for Oracle and a demo database

Software

- SuSE Linux Enterprise Server 8

  We recommend that during OS installation you select the *Default* installation option in the SuSE Linux Enterprise Server 8 software selection screen and go from there.

  These are not unbreakable rules. If you know what and why you want to break them it can be okay to do so.

  Must Install:

  - ✔ Make sure that **X–Window** is installed and running, or that at least the X–Window libraries are present for a remote installation.

  - ✔ You will also need to have the basic developments tools installed, like **make**, **gcc**, and the **binutils** package.

  - ✔ To view the Oracle documentation you need a **web browser** (e.g. Netscape or Konqueror) and a **PDF reader** (e.g. Adobe Acrobat Reader™, gv/ghostscript or xpdf).

  - ✔ You may need to install package **pdksh**, which provides a Korn shell. Some Oracle scripts require this shell.

  - ✔ You should install the **Java Development Kit** (JDK) 1.1.8. SuSE Linux Enterprise Server comes with the Sun JDK 1.3.1 in package *java2*. If you will not work with Java this is not needed, however. The Java GUI tools Oracle comes with use their own JRE (Java Runtime Environment) bundled with Oracle.

  Must NOT Install:

  - ✗ Do not install any webserver, like Apache. Oracle already comes with Apache.

  - ✗ Do not install OpenLDAP, an LDAP directory server, if you do not plan to use it or if you plan to use Oracle Internet Directory (OID), which is Oracle's LDAP service.

  - ✗ We recommend you do not install any other server software. Production Oracle servers should run only Oracle and nothing else. One exception could be a Mail Transfer Agent (MTA) like sendmail or postfix.

  - ✗ Do not replace any core components of the Linux system with packages from outside sources. Oracle is supported and certified only with the original SuSE Linux Enterprise Server 8, plus SuSE patches and patchsets. This is especially true for key components like the kernel. You loose certification and support if you use anything but the kernels SuSE provides. You can compile your own, but you must use the SuSE kernel sources.

- · Oracle 8.1.7 CD or archive

  You either need an Oracle 8.1.7 CD or the archive you can download for free from OTN
  (http://technet.oracle.com/). The OTN archive is exactly the same version you get from Oracle
  on CD, the limitation is the Oracle Technet Evaluation License you have to acknowledge before
  getting to the download area. The binaries themselves are not limited in any way.

- · Patches/Patchsets for SuSE Linux Enterprise Server 8

  See the SuSE Maintenance Web. You must be a SuSE Maintenance customer to have access.
  Install the SuSE patches after the operating system installation, and before or after the Oracle
  installation, it does not matter for the installation process. When we created this document we
  did the installation on a pristine SuSE Linux Enterprise Server 8 without any additional patches
  installed.

- · Patches/Patchsets for Oracle 8.1.7

  See Oracle Metalink. You must be an Oracle support customer to have access. The Oracle
  patches must be installed after the Oracle installation, obviously. You install patchsets using the
  Oracle Universal Installer. Each patch and each patchset comes with detailed installation
  instructions.

- · SuSE Support Website for Oracle: http://www.suse.com/oracle/

  Please check the page for *Oracle 8.1.7 on SuSE Linux Enterprise Server 8*, reachable directly
  from the main page location above by clicking on the appropriate icon in the support matrix
  (direct URL: http://www.suse.com/en/support/oracle/db/817_sles8.html) for any updates and
  additional information.

Pre–Installation

Tips for the OS Installation

Partitioning: Ideally you have partitioned your system in a way that allows you to install Oracle and the database file on different partitions than the OS. The advantage is that when you have to update the operating system at some point you can do a complete new, fresh installation and reformat the OS partition without loosing your Oracle installation and/or data. We recommend to place the Oracle installation in */opt/oracle*, for the only reason that this is what our add–on package *orarun.rpm*, described below, has as a default.

Here is an example for a setup we would recommend for a production system:
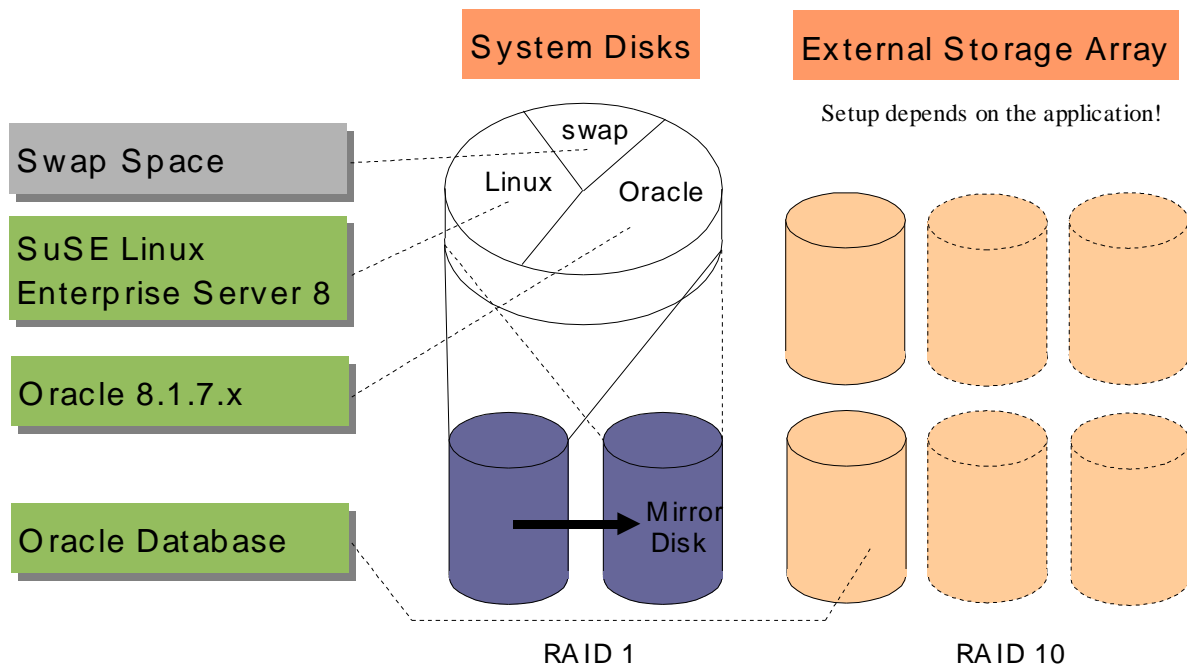
The hardware consists of two main systems: A computer system (the server) with one or better two internal disks and a hardware RAID controller, and an external storage unit with lots of space, typically connected via an external SCSI cable or a fiber channel cable (more robust).

This document only considers the server unit, the configuration of the external array where the database contents will be stored on depends on the application. There are Oracle performance guides available – in bookstores and for free from Oracle on OTN as online books and whitepapers – that deal with that database administration topic. We do not need to concern ourselves with it at this point.

We suggest that you have at least two internal drives in the server unit, configured in a RAID–1 (mirroring) configuration in the hardware RAID controller of the server to protect us against the loss of one of the drives. There should be three main areas (partitions) on the disk (the virtual disk the SuSE Linux installer sees as provided by the hardware RAID controller). One is the swap space, the other one is for the operating system – possibly further divided e.g. to put logfiles and the temporary directory in a different partition – and the third one is for the Oracle installation.

You should give ca. 1.5–4 GB to Linux, 1–2 GB for swap and the rest is for Oracle.

Memory requirements: We recommend a total of at least 512 MB total RAM (physical RAM and swap space). How much you actually need and how much of it should be physical RAM completely depends on how you are going to use the database and how many users are going to connect to it, and how your application works (does it do a lot of computing in the database, or does it pull the data to another tier , e.g. an application server, and does the processing there?). The extremes go from 64MB physical RAM for a slow demo database doing some swapping to a big server system with several GB of RAM. Oracle provides sizing studies. This is an Oracle issue and not really a Linux issue.

After the OS Installation

Remote access: After the OS installation the only way to access the system remotely is via Secure Shell (*ssh*). Anything else, e.g. *telnet*, *ftp*, or *rsh*, will not work. For security reasons SuSE Linux by default does not activate the *inetd* daemon providing those services, they have to be enabled explicitly either by editing */etc/inetd.conf* to enable services and then by calling (as *root*) *"insserv /etc/init.d/inetd; rcinetd start"* or by using the YaST2 Control Center (call "yast2" – works with and without X–Window – go to "Network/Basic" and select option "Start/Stop services (inetd)") which does the same. See the SuSE Linux Enterprise Server documentation for details.

Special Partitioning: It is now time to setup the space for the Oracle installation. At least the database files should be on different disks than the operating system, but the Oracle installation should also go in a partition different from the operating system, see above, we recommend a clean separation. Read the Oracle Administration Guide that comes with Oracle 8.1.7 about the Oracle Flexible Architecture (OFA), the disk– and mount point setup Oracle recommends.

Since there are many different methods and all have their use, and depend a lot on how the database is going to be used, we will not make any suggestions. A small demo/test system can very well have everything, OS, Oracle and data files in one partition, but a big production database on a server with many disks may have a very different setup.

Logical Volume Management and possibly also using raw I/O are more sophisticated methods to manage space. These make sense for larger systems and databases. See the Logical Volume Manager Whitepaper at *http://www.suse.com/en/support/whitepapers/lvm/*. Note that the PDF version there may be slightly more up–to–date since the conversion to HTML is done separately.

Kernel parameters are something you do not need to worry about, we will set them dynamically later during runtime. You can and should use the kernel the SuSE installer installed in your system, unless you are really familiar with the process of compiling a new kernel. If you do compile your own please use the SuSE kernel sources. Check if there is an official SuSE update kernel available for SLES–8 on the Maintenance web.

Refer to either the *Oracle Installation Guide for Linux* for which variables to set to what values, or simply read on, install the *orarun* package (see Appendix), and get that information from the file /etc/sysconfig/oracle. For the installation you do not need to do this in any case, only for actually running Oracle.

Users and groups for Oracle (user *oracle*, groups *oinstall*, *dba*) are setup automatically when you install package *orarun* as described below.

Environment variables are another subject we will take care of during the installation, by installing a package *orarun* SuSE created to make these things easier.

Detailed Installation Instructions

Starting point: SuSE Linux Enterprise Server 8 has been installed and all partitioning has been done. You have Oracle 8.1.7, CD or as tar archive (which you already unpacked to some location).

1. Depending on if you have a GUI login window or are on a text console:

   Start X−Window and login (as yourself, the user you setup during the system installation). Or the other way around, depending on if you booted into the *xdm* runlevel 5 (X−Window is always on) or into the *console* runlevel 3 (X−Window is off by default).

   The picture shows the default SuSE login screen which you get if you configured X−Windows during the installation process **and** included the KDE packages in the installation (the *Default* software selection does that, the *Minimum+X11* does not and looks slightly different).
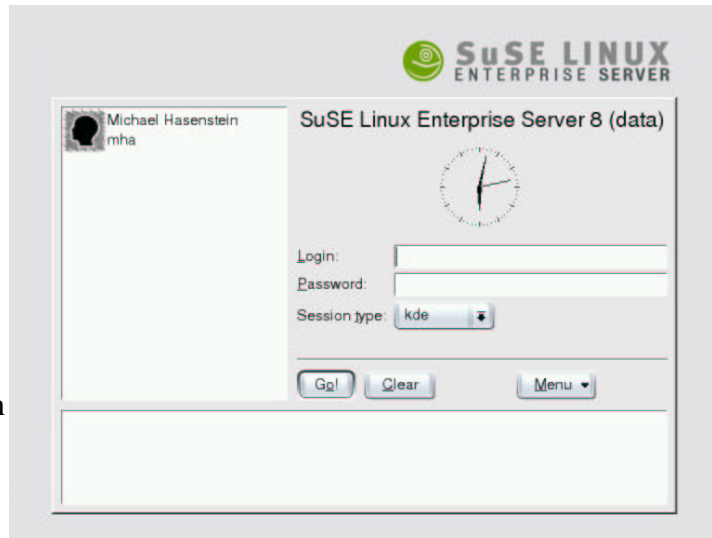
   Click on the icon representing the user, or enter the user name manually. Then enter the password − note that you will not see anything you type for security reasons (someone might be looking over your shoulder, for example). We suggest to keep *KDE*, then press *GO*.

2. Open two *xterms* or KDE *konsoles*, or simply anything that provides a *shell* (a command line prompt).
   See the picture, this is how your screen should look like afterwards. First (**1.**) you open a KDE *Terminal*, then you press the *New* button (**2.**) to open a second shell, and within the terminal window you can switch between the two shells by pressing the two buttons, marked *root* and *oracle* in the picture.
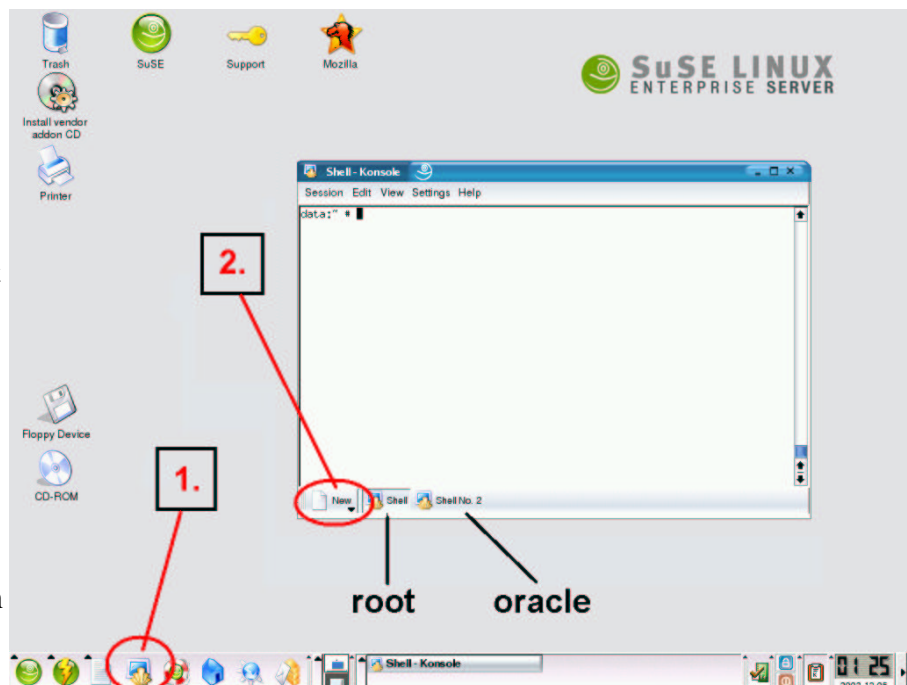
Shell #1 (*root*):

3. To become user *root*, execute this command and enter the *root* password when prompted for it. Note that you will not see what you type at the password prompt, as always in a Unix system.
   su –
   Now you are logged in as the administrator – but only in this shell, everywhere else you are still the regular user you logged in as.

4. Get package *orarun.rpm* and install it. You can use any web browser to download it, or you can also use the command below if you have *wget* installed (ignore the line break in the URL, the paper this is printed on is just not wide enough!).
   **wget *ftp://ftp.suse.com/pub/suse/i386/supplementary/
        commercial/Oracle/sles-8/orarun.rpm***
   rpm –Uvh orarun.rpm

5. Optional, since the defaults are okay for the installation process and for running a demo database): Check and edit file **/etc/sysconfig/oracle**

   A description of this file is in the appendix of this document.

6. Check the two files **/etc/profile.d/oracle.[c]sh**

   There are two files for setting environment variables. The one ending in *.csh* is for C–shell users, the other one ending in *.sh* for Bourne shell users. Default in SuSE Linux is a Bourne shell.

   The three most important variables are ORACLE_BASE, ORACLE_HOME and ORACLE_SID.
   - ORACLE_BASE is the base for everything Oracle installs, like the product itself, the Java Runtime Environment (JRE) used to run the GUI tools, the documentation, the inventory directory where the Oracle Universal Installer keeps track of what components are installed, the Oracle Universal Installer itself, and by default it is also the base for the database files, logfiles and parameter files – but you can select a completely different location when you create your own database.
   - ORACLE_HOME is a directory below ORACLE_BASE (at least it should be!) where the Oracle product, in this case the database, is installed. If you install multiple Oracle products they all get their own ORACLE_HOME, but we recommend you use the same ORACLE_BASE directory.
   - ORACLE_SID is an identifier for a database instance. Your Oracle installation can run several different databases on one machine, simultaneously, and they all get a different SID. We suggest to use only four characters for the SID, although more are possible. The default is *mydb*. For example, if you run Oracle Parallel Server each instance on each node gets a different SID but they all work on the same database.

   By default SuSE points ORACLE_BASE to the directory /opt/oracle, so if you do not change this setting this is where Oracle will get installed.

7. Prepare the Oracle user. To set password for the *oracle* user execute the command:
   passwd oracle

   Recommended but optional: Create extra /home/oracle for oracle user. By default it is in
   /opt/oracle and no dot−files like .profile exist since we, SuSE, by universally accepted
   standards are not allowed to install anything into the /home directory. Execute the following
   commands to create a full home directory:
   cp −a /etc/skel/ /home/oracle
   chown -R oracle:oinstall /home/oracle
   usermod −d /home/oracle oracle

8. If you have an Oracle CD, mount CD−ROM, insert it into the CD−ROM drive and mount it by
   positioning your mouse pointer over the CD−ROM icon on your KDE desktop, then press the
   right mouse button and select *Mount* in the menu that appears.

   Alternatively, just execute this command in a shell that does the same:
   mount /cdrom

   Shell #2 (*oracle*):

9. Switch to the other shell and become user oracle by executing the command (do not use
   "su", "sux" takes care of the X−Window environment):
   sux − oracle

10. Get Oracle's glibc patch from technet.oracle.com and store it somewhere. Command
    (username/password is your free Oracle OTN account on *http://technet.oracle.com/* and please
    ignore the line breaks in the command, the paper this is printed on is just not wide enough):
    wget −−http−user=??? −−http−passwd=??? http://download.
    oracle.com/otn/linux/oracle8i/glibc−2.1.3−stubs.tar.gz

    You could also use any regular web browser, of course.

11. Execute this shell command to work around a problem of the IBM JRE Oracle uses. You loose
    Oracle's internationalization − but only for the installer texts and help, this has nothing to do
    with the database:
    unset LANG

12. Make sure the NumLock key on your keyboard is OFF. If it is not, pressing any of the buttons
    in the Oracle Universal Installer with the mouse will have no effect. This is a bug in Java or the
    Oracle Universal Installer.

13. Do **not** change into the Oracle source directory (e.g. /cdrom), remain in user *oracle*'s home
    directory. Start the installer (assuming you have a CD, otherwise adapt the path to wherever
    your Oracle installation source is located if it is not a CD):
    /cdrom/runInstaller

Oracle Universal Installer:

14. This is the welcome window of the Oracle Universal Installer and what you should see after starting it.
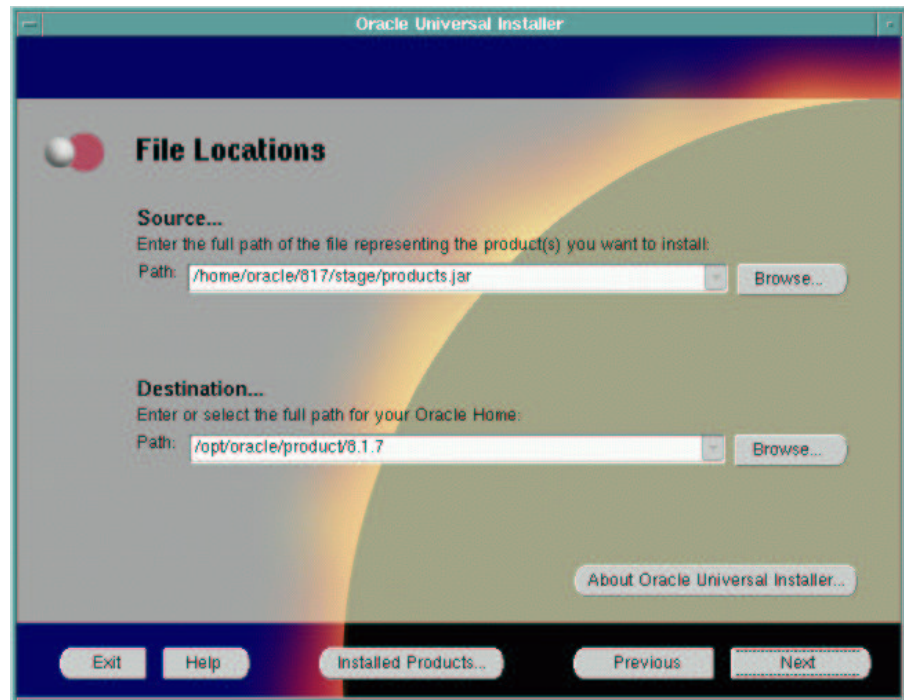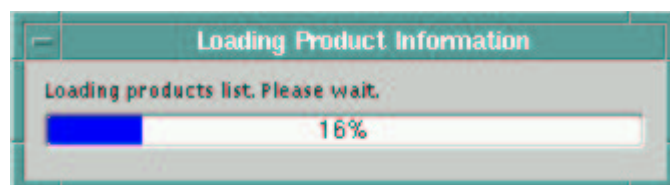
Press *Next*.

15. The destination for the installation is already okay, the installer simply takes the value of ORACLE_HOME.

Should you want a different value you should edit ORACLE_HOME as well.

Press *Next*.

16. A small window with a progress indicator will show up, wait until it is finished:

17. You will see the following two windows **only if there is <u>no</u> file `/etc/oraInst.loc`** yet on your machine, which this step will create.

The second sentence in this window is useless, since you should **never** start the Oracle Universal Installer as *root*. What it should say is that if you leave the field blank only the user you are running the installation as will be able to update the Oracle software.

We put the group *oinstall* in the field, but since only user *oracle* is a member of this group by default this is equivalent to leaving it blank, but leaves you with the flexibility to add user names to this group later should you wish so.
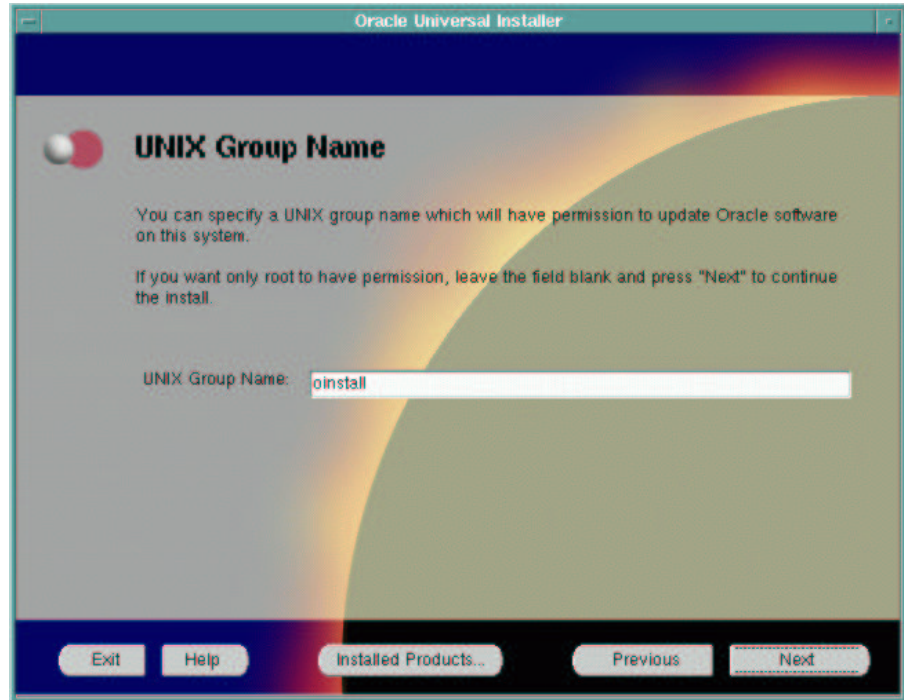
When this gray window pops up after you pressed *Next* in the previous screen *UNIX Group Name*, switch to the *root* shell window temporarily:
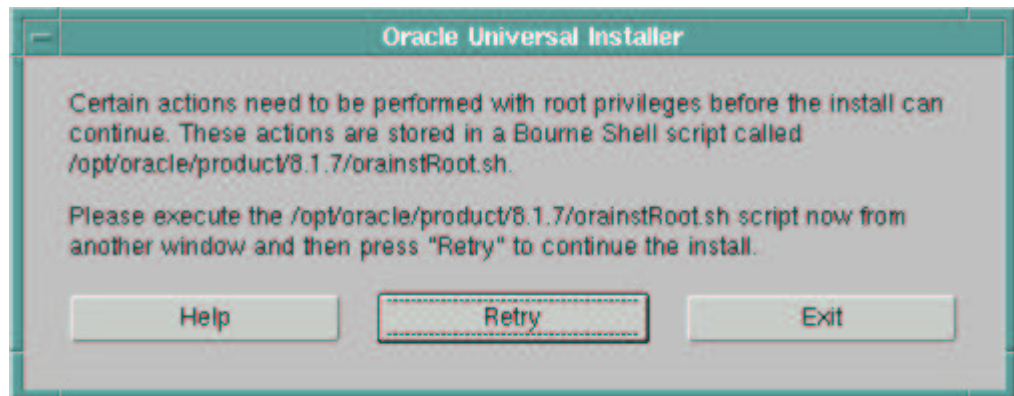
Shell #1 (*root*):

Just run the script as told. Type the location – which depends on your setting of $ORACLE_HOME – as it appears in the window you see on your screen.

After running the script Press *Retry* in that gray Oracle Installer window to continue with the installation.

This is the end of this optional step, which creates a global Oracle configuration file `/etc/oraInst.loc` with only two entries which will be used by any Oracle Universal Installer for any other Oracle product as well. If this file exists this step does not exist.
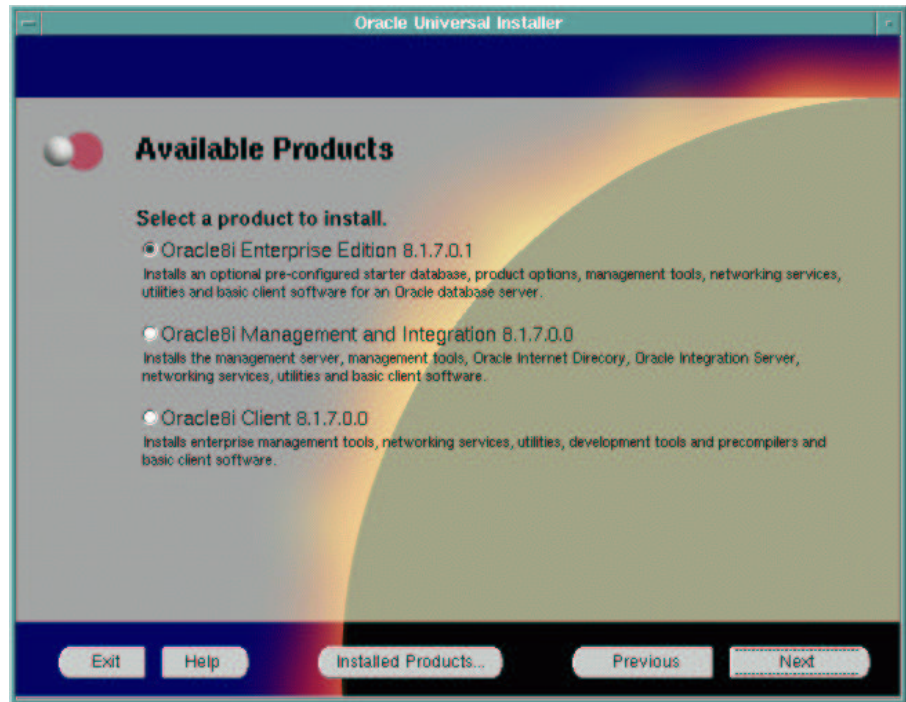
Oracle Universal Installer:

18. Select the installation type.

Info: The client tools of the third item are included when you select the first or second one.

For these instructions we selected the first item, the **server installation**.

Press *Next*.


You get three more choices now.

If you want to install Oracle Parallel Server you have to select *Custom*.

Tip: If you select *Typical* or *Custom* will also have significance towards the end of the installation when the Network Assistant and the Database Creation Assistant are run. In a *typical* installation those tools will perform a default operation, in a *custom* installation they will be much more interactive and will let you set a lot of parameters.
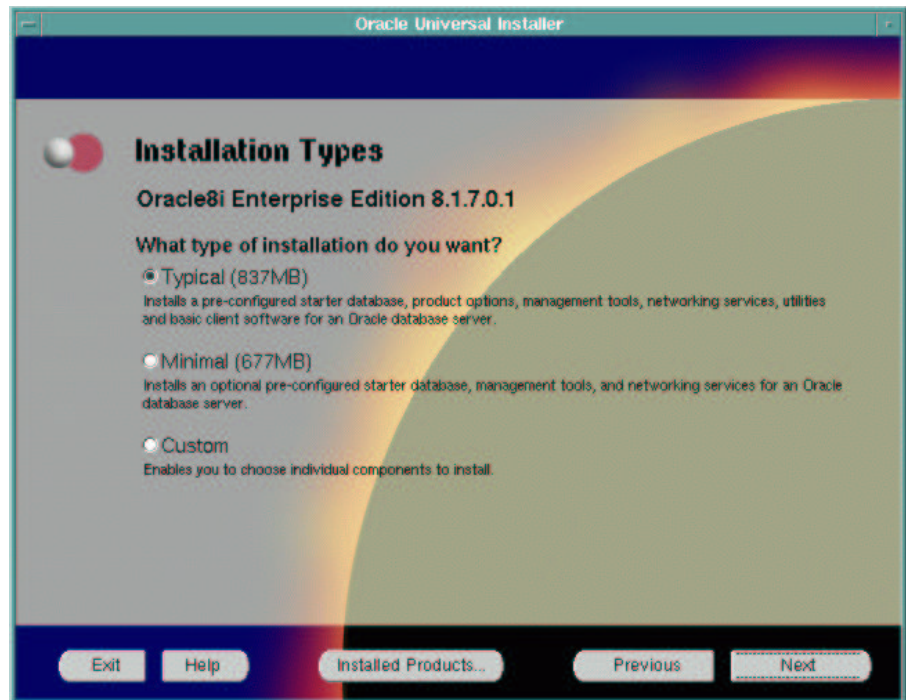
19. Specify the location of the JDK 1.1.8v3 installed on your system. The path is */usr/lib/jdk1.1.8*.
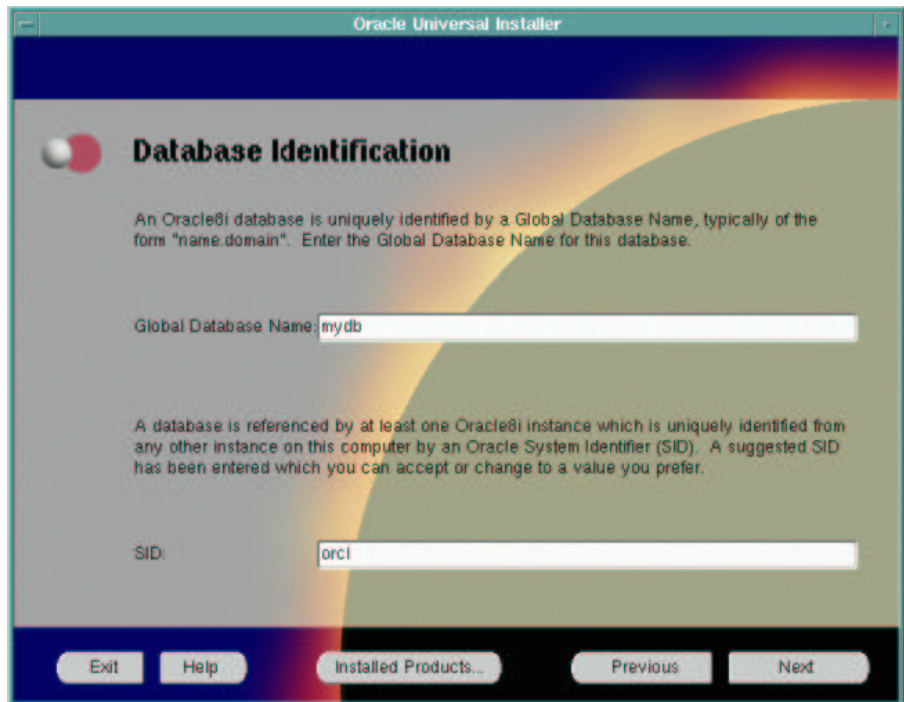
Press *Next*.
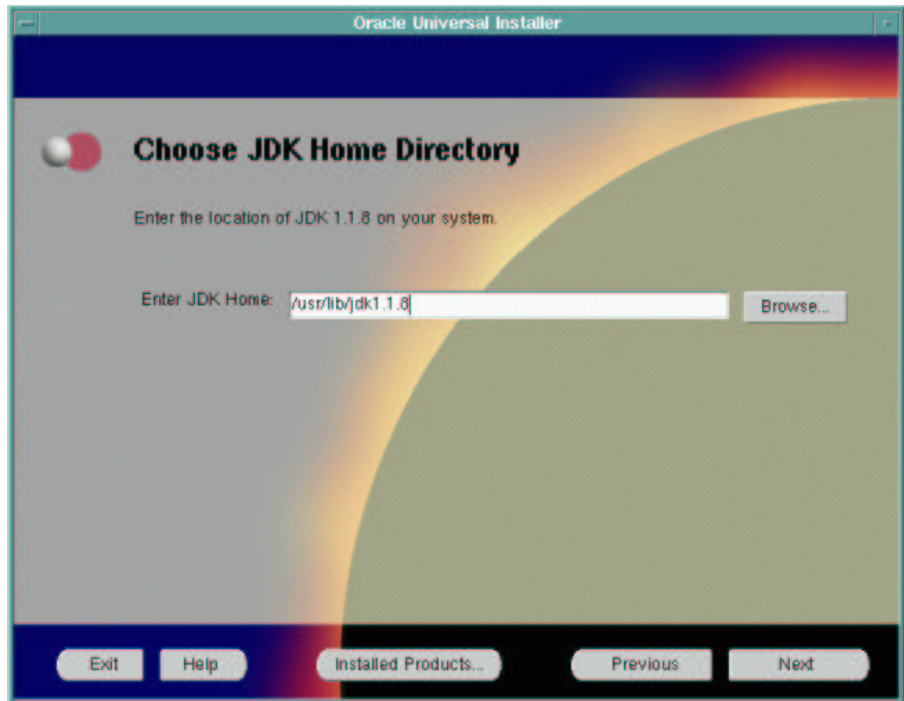
20. Database Identification

Tip: If you plan to use iAS 9*i* Database Cache for the database you create here the name and the SID must be equal.

If you do not know what all of this means do not worry, to create and use a demo database you do not need to know. We used *mydb* as name.

21. Summary screen – review your settings and press *Install* and go and have a cup of coffee, this will take a few minutes.
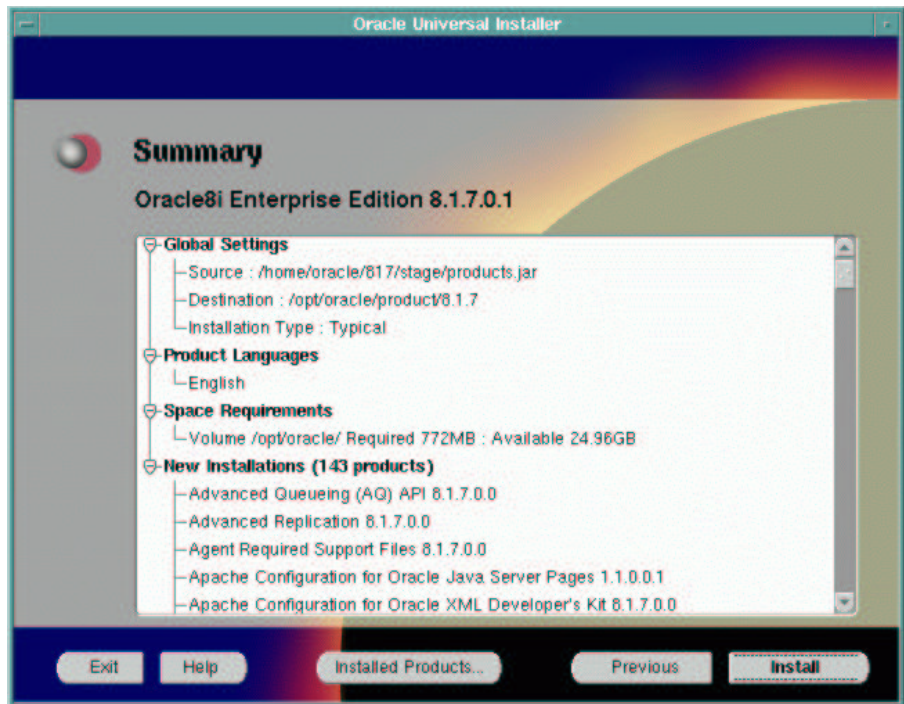
This can take a few minutes or a half hour, depending on how fast your system is.

22.After some time:

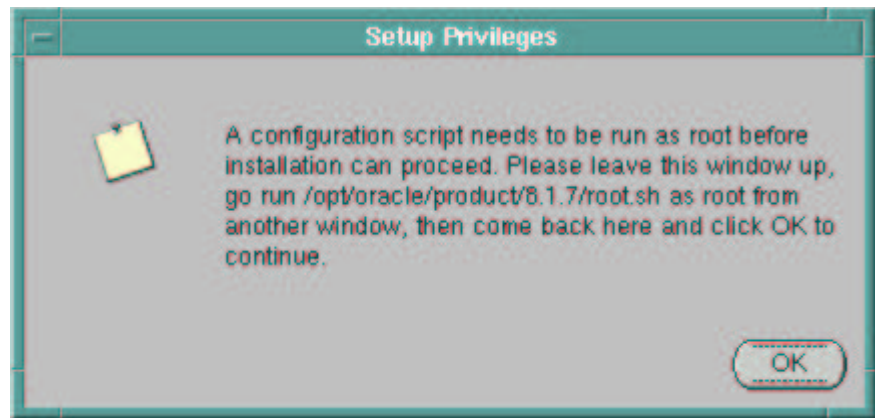When the gray window with the message that tells you to run *root.sh* pops up we have to do a few things in the two terminals before we can continue with the Oracle Installer.

Setup Privileges

A configuration script needs to be run as root before installation can proceed. Please leave this window up, go run /opt/oracle/product/8.1.7/root.sh as root from another window, then come back here and click OK to continue.

OK

Shell #2 (*oracle*):

23.Go to the #2 terminal (user *oracle*). Now it is time to install the Oracle patch we downloaded earlier:
```
cd $ORACLE_HOME
tar xvzf where-is-it/glibc-2.1.3-stubs.tar.gz
```
(Info: creates directory $ORACLE_HOME/lib/stub/ and files
$ORACLE_HOME/setup_stubs.sh and $ORACLE_HOME/README.stub)
Optional: Read README.stub (application developers should definitely read it before doing any linking against the Oracle libraries)

24.Run **./setup_stubs.sh**
While you wait for this command to finish you can already take on the next three steps if you like, just make sure that this step is finished before you continue with the Oracle Installer.

25.Switch to the other (user *root*) terminal we opened and used earlier.

Shell #1 (*root*):

26.There are two errors in Oracle's script. Edit **$ORACLE_HOME/root.sh**:
1.) Edit the line that says
RUID=`/usr/bin/id|$AWK -F\( '{print $2}'|$AWK -F\) '{print $1}`
and add a "'" just before the very last character "`".
2.) The line
 RMF=rm -f
exists TWICE, first correctly with quotes around "rm –f", but the second time they are missing.
Add them, so that it looks like the first one:
 RMF="rm -f"

27.Now run the script:
```
$ORACLE_HOME/root.sh
```
Wait until it is finished, which may take several minutes.

Also make sure that the other shell command we issued earlier, ./setup_stubs.sh, is finished before you continue.

Oracle Universal Installer:

28. Close the gray message window by pressing *OK*.

29. The Oracle Installer will now configure Net8, create a demo database, and start the Apache webserver that comes with Oracle.

At this point Oracle is **fully installed** and if any of these tools fail it does not really matter. You can always call them on the command line manually (*netasst* to create or edit a Net8 configuration, *netca* to create a Net8 configuration using a wizard, *dbassist* to create a database; **Note:** you

must unset the environment variable for internationalization with the command `unset LANG` before starting **any** of the Oracle Java GUI tools or they will crash immediately).
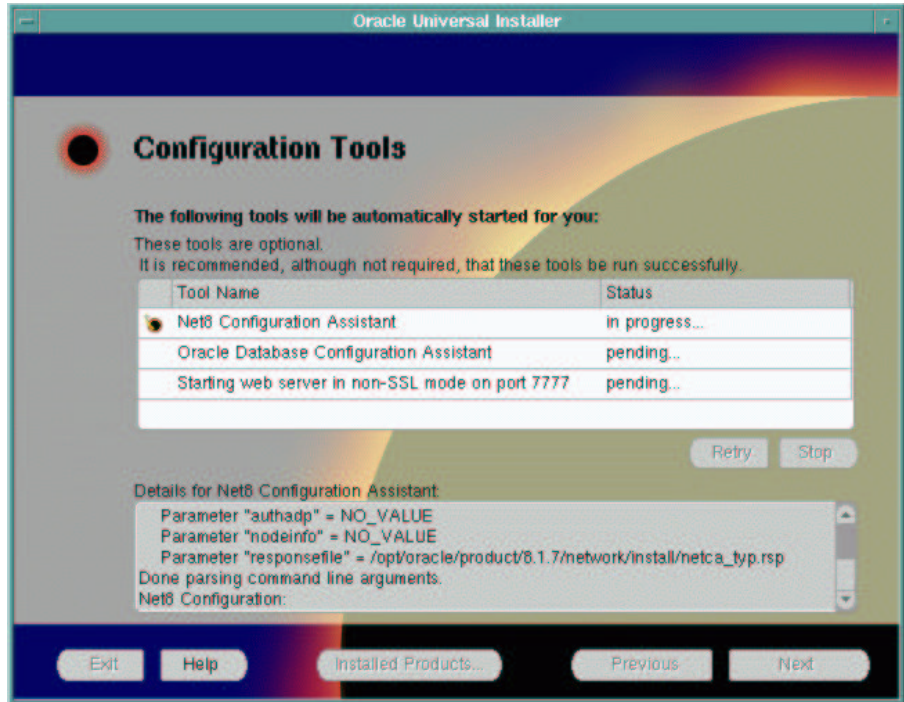
Net8 Configuration Assistant

Configure Net8. If you do not know much about Net8 just accept the *Typical Configuration* option the tool suggests. Note: the connect string will be the name of the database selected at the beginning – not the SID, the name! – plus the machines DNS domainname). Example for db–name=mydb and domain=us.oracle.com: connect to the database using


```
sqlplus scott/tiger@mydb.us.oracle.com
```
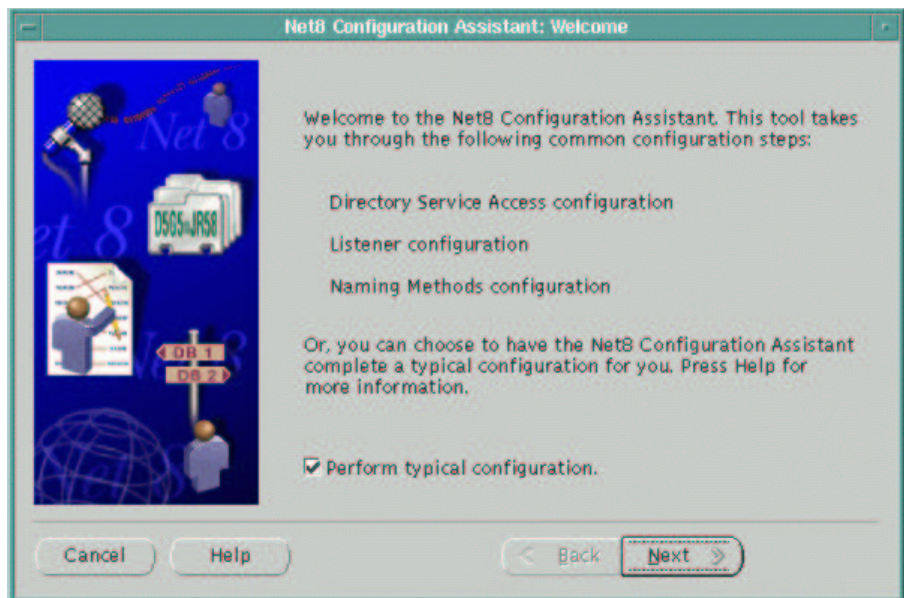
Database Configuration Assistant

Create demo database. If you selected a *Typical* configuration at the beginning you are not asked any questions, the database is not created new, instead default binary demo database files are copied from the Oracle installation CD to the system.

Since we selected a *Typical* installation we get a default database copied into the system, already created and ready to be used immediately. No parameters can be set. In a *Custom* installation this tool will offer a lot more dialog boxes!

At the end we get a summary of the database name and the system identifier, and what the default passwords for the DBA are.
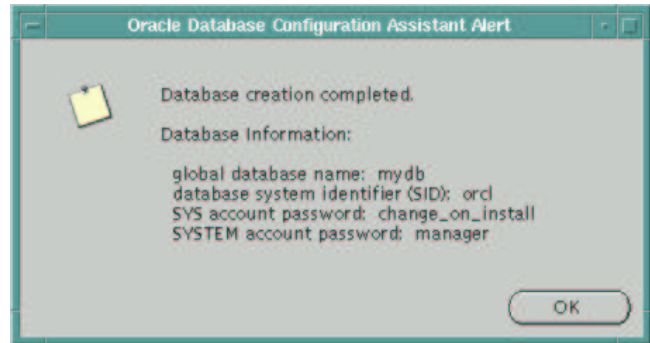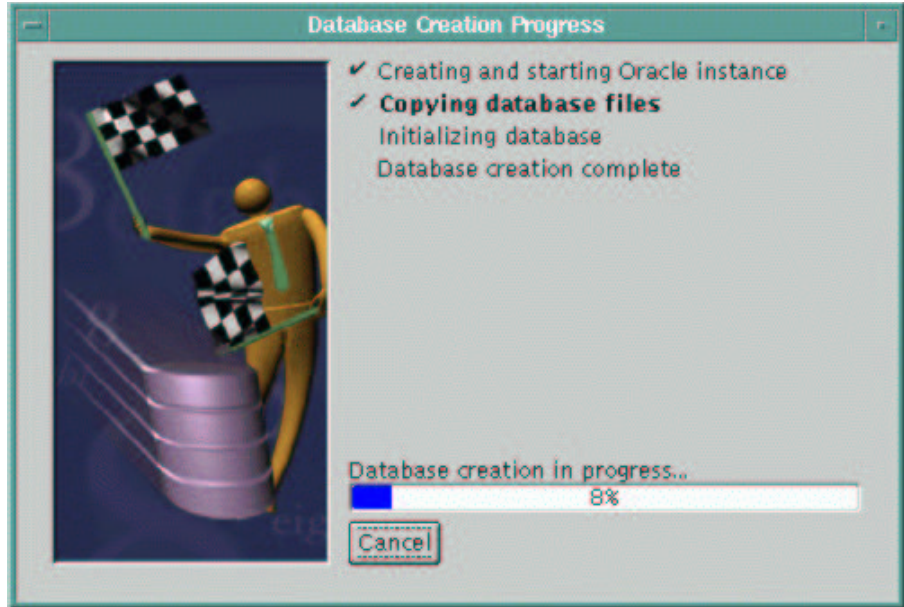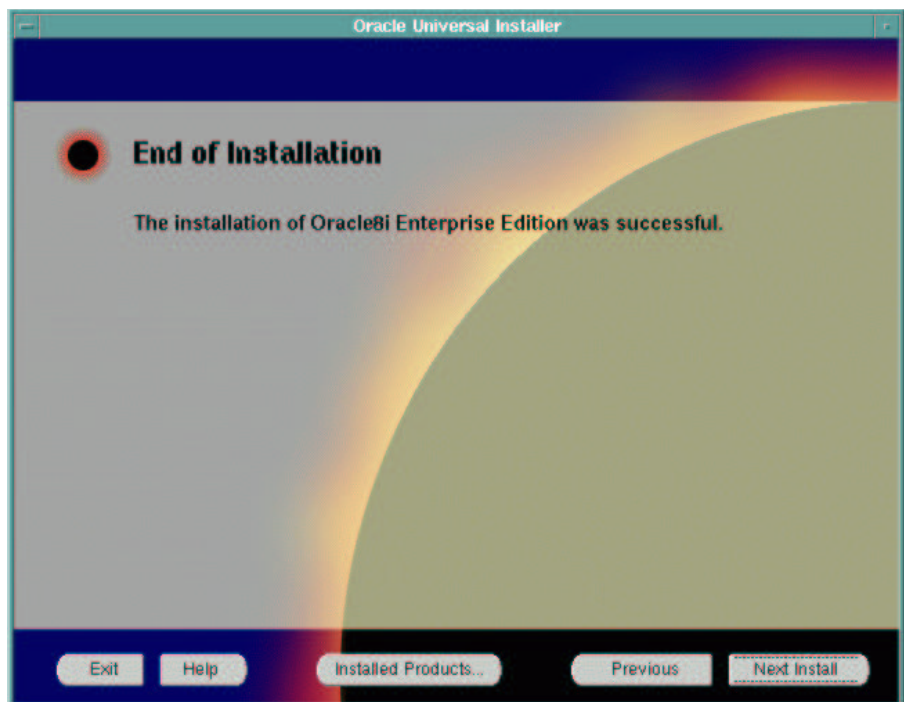
30. The end. Select *Next Installation* e.g. to install Pro−C/C++, or exit the installer.

Note: <u>Installing Oracle 8*i* Patchsets</u>

Please note that you **must** re−run step #24 (also see #23) to re−apply the stub−libs patch after installing an Oracle patchset!

END

Quick Installation Instructions

The SHORT version of the same – For the knowledgeable and impatient users.

Starting point: SuSE Linux Enterprise Server 8 has been installed and all partitioning has been done. You have Oracle 8.1.7, CD or as tar archive (which you already unpacked to some location).

1. Start X–Window and login (as yourself). Or the other way around, depending on if you booted into the *xdm* runlevel (X–Window is on) or into the *console* runlevel (X–Window is off).

2. Open two *xterms* or KDE *konsoles*, or simply anything that provides a shell.

Shell #1 (*root*):

3. Become user *root*:
   ```
   su –
   ```

4. Get package *orarun.rpm* and install it (do not type the line break in the URL):
   ```
   wget ftp://ftp.suse.com/pub/suse/i386/supplementary/
        commercial/Oracle/sles-8/orarun.rpm
   rpm -Uvh orarun.rpm
   ```
   You could also use any web browser, of course.
   For your information: There is a file /usr/share/doc/packages/orarun/README.

5. Check /etc/profile.d/oracle.[c]sh: ORACLE_HOME, ORACLE_SID, ...

6. Check /etc/rc.config.d/oracle.rc.config: START_ORACLE, etc.

7. Set password for oracle user: as *root*, do
   ```
    passwd oracle
   ```
   Optional: Create extra /home/oracle for oracle user (by default it is in /opt/oracle since we must not install anything into /home during OS installation): as *root*, do
   ```
    cp -a /etc/skel/ /home/oracle
    chown -R oracle:oinstall /home/oracle
    usermod -d /home/oracle oracle
   ```

8. If you have an Oracle CD, mount CD–ROM: as user *root*, do
   ```
    mount /cdrom
   ```

Shell #2 (*oracle*):

9. Become user oracle (do not use "su"):
   ```
   sux – oracle
   ```
   Make sure that step 5 has been done or the environment will not be set for the oracle user!

10. Get the glibc patch from technet.oracle.com and store it somewhere. Command (user/password is your Oracle OTN account info, *http://technet.oracle.com/* and please ignore the line breaks in the command, the paper is just too small):
   ```
    wget --http-user=??? --http-passwd=??? http://download.
    oracle.com/otn/linux/oracle8i/glibc-2.1.3-stubs.tar.gz
   ```
   You could also use any web browser, of course.

11. Shell command to work around a problem of the IBM JRE Oracle uses (you loose Oracle's internationalization – but only for the installer, this has nothing to do with the database):
    ```
    unset LANG
    ```

12. Make sure the NumLock key on your keyboard is OFF. If it is not, pressing any of the buttons in the Oracle Universal Installer will have no effect.

13. Do not change into the Oracle source directory (e.g. `/cdrom`), remain in user *oracle*'s home directory. Start the installer:
    ```
    /cdrom/runInstaller
    ```

Oracle Universal Installer:

14. Follow the instructions of the Oracle Universal Installer to install Oracle. The destination for the installation is already okay, the installer simply takes the value of $ORACLE_HOME.

    If a gray window pops up when you press *Next* in the screen *UNIX Group Name* with the buttons *Help*, *Retry*, *Exit*, switch to the *root* shell window temporarily:

Shell #1 (*root*):

15. If there is no file `/etc/oraInst.loc` yet, the Oracle Installer will come up with a gray window that tells you to run some script as root. In this case: just run that script.

16. Press "Retry" in that gray Oracle Installer window to continue with the installation.

Oracle Universal Installer:

17. Follow the easy instructions of the Oracle installer. Note that if you select a *Typical* installation later in the process the installer will install a fixed, pre–created binary demo database, but if you select *Custom* you will get the choice to create a new database with lots of parameters to influence.

18. When you get to the end of the Q&A screens, when you get to review all settings before starting the actual installation process: Press *Install*. This will take a while.

    Wait....

    When the gray window with the message that tells you to run *root.sh* pops up we have to do a few things in the two terminals before we can continue with the Oracle Installer.

Shell #2 (*oracle*):

19. Go to the #2 terminal (user *oracle*). Now it is time to install the Oracle patch we downloaded earlier:
    ```
    cd $ORACLE_HOME
    tar xvzf where-is-it/glibc-2.1.3-stubs.tar.gz
    ```
    (Info: creates directory `$ORACLE_HOME/lib/stub/` and files
    `$ORACLE_HOME/setup_stubs.sh` and `$ORACLE_HOME/README.stub`)
    Optional: Read README.stub (application developers should definitely read it before doing any linking against the Oracle libraries)

20. Run `./setup_stubs.sh` You can already take the next three steps, just make sure that this step is finished before you continue with the Oracle Installer.

21. Switch to the other (user *root*) terminal we opened and used earlier.

Shell #1 (*root*):

22. There are two errors in Oracle's script. Edit `$ORACLE_HOME/root.sh`:
    1.) Edit the line that says
    `RUID='/usr/bin/id|$AWK -F\( '{print $2}'|$AWK -F\) '{print $1}'`
    and add a "'" just before the very last character "'".
    2.) The line
    `   RMF=rm -f`
    exists TWICE, first correctly with quotes around "rm –f", but the second time they are missing. Add them, so that it looks like the first one:
    `   RMF="rm -f"`

23. Run the script:
    `  $ORACLE_HOME/root.sh`
    Wait until it is finished, which may take several minutes. Also make sure that the other shell command we issued earlier, `./setup_stubs.sh`, is finished before you continue.

Oracle Universal Installer:

24. Close the gray window by pressing the *Retry* button.

25. The Oracle Installer will now start a few tools to configure Net8, create a demo database, and to start Apache. At this point Oracle is FULLY INSTALLED, and if any of these tools fail it does not matter, you can always call them on the command line manually (*netasst* to create or edit a Net8 configuration, *netca* to create a Net8 configuration using a wizard, *dbassist* to create a database; you **must** `unset LANG` before starting **any** of the Oracle Java GUI tools or they will crash even before they start).

26. Configure Net8. If you do not know much about Net8 just accept the *Typical Configuration* option the tool suggests.
    Note: the connect string will be the name of the database selected at the beginning – not the SID, the name! – plus the machines DNS domainname).
    Example for db–name=mydb and domain=us.oracle.com: connect to the database using
    `   sqlplus scott/tiger@mydb.us.oracle.com`

27. Create demo database.
    If you selected a *Typical* configuration at the beginning you are not asked any questions, the database is not created new, instead default binary demo database files are copied from the Oracle installation CD to the system.

28. The end. Select *Next Installation* e.g. to install Pro–C/C++, or exit the installer.

Note about installing Oracle 8*i* Patchsets: You **must** re–run step #20 (also see #19) to re–apply the stub–libs patch after installing an Oracle patchset!

END (Quick Install section)

Appendix – SuSE Package orarun.rpm

Package information

Name   : orarun
License   : GPL

Summary: Environment for running Oracle 8*i* and 9*i*

Description: This package
- sets the Oracle environment variables for each user, like ORACLE_HOME and PATH
- sets the recommended kernel parameters, e.g. SHMMAX
- provides for automated start/stop of Oracle processes at system startup/shutdown

You may want to or even have to edit /etc/profile.d/oracle.[c]sh (environment variables), /etc/sysconfig/oracle (which components to start/stop, values for kernel parameters).

File list:
Shell script: `/etc/init.d/oracle`
Shell variables: `/etc/profile.d/oracle.csh`
C–Shell variables: `/etc/profile.d/oracle.sh`
Link (in *root*'s PATH): `/usr/sbin/rcoracle -> ../../etc/init.d/oracle`
Shell variables: `/etc/sysconfig/oracle`
Documentation: `/usr/share/doc/packages/orarun/README`

1.) It **provides the environment variables for running Oracle. It does so for all** users, by placing it in the `/etc/profile.d/` directory, and it provides a file each for Bourne shell users and for C shells (SuSE Linux default shell is *bash*, a Bourne shell). Some administrators may not like to have an Oracle environment set for all users, they can simply place those files somewhere else and ignore any errors rpm produces later when it finds that the two files that art part of the orarun package are not there.

2.) The package also provides a script for **automated startup and shutdown** (when the system starts up or shuts down) of the Oracle database and of other Oracle components like the Apache webserver or the listener. Which components should be started by the script can be controlled by setting certain variables to *yes* or *no* in the text file `/etc/sysconfig/oracle`.

3.) The startup script also takes care of **setting the kernel parameters** for running Oracle. The file storing the values is `/etc/sysconfig/oracle`. Have a look and edit it according to the instructions contained in that text file! For small to medium databases no values need to be adjusted at all since SuSE already provides reasonable defaults, only for bigger production systems might it become necessary.

Note that the startup script does its two tasks – setting kernel parameters and starting/stopping Oracle processes – independent of one another.