Introduction

Oracle database is available on Oracle's website for free downloading. All someone need to do is just sign up there and get Oracle software for his/her business without paying a penny at all. Now this database software can be used in the company's/institution's/personal systems. By default from 11g onwards most of the features (aka database options) are bundled with the software so one can use them directly without worrying/thinking on downloading and installing them again and again (like rpm packages in Linux). After installing the free downloaded copy you get a full featured Oracle database up and running in your organization which can be used for critical/non-critical all kind of data and the best part is that you have got everything free of cost but there are some points to be considered : --

- o Is this legal?   Using a Product with all the features without purchasing a license of it. Though all the features can be used easily but this is not at all legal.

- o What about support? As long as you can manage things well, it is well (though it is illegal) but what if you stuck somewhere and seeking for assistance.

- o If Oracle audits your company? Definitely there will be a hefty impact. (financial as well as copyright)

The RDBMS market is seeing a tough competition now with non-RDBMS softwares, one of the major reason is the licensing cost. The amount required to buy a license for any RDBMS software can be too much for a small scale company. There are many RDBMS vendors currently present in the market and Oracle is one of the major player here.

So, in this document we will try to find out how much costly an Oracle database is, also we will try to ascertain the reasons why small scale companies don't find it feasible to go with Oracle, why some of the small companies are migrating from Oracle to open source database.

Oracle Licensing

Oracle has a simple formula to calculate the license cost.

Total cost = Total licenses * cost per license + total licenses * cost per add-on pack + Support cost for each pack

You have to purchase license of each add-ons pack which you will be using in the database.

The first term i.e., Total license can be of 2 types.

1. Per Processor Licensing

This licensing method takes number of Processors (Sockets) of the server into consideration, the server where Oracle database is hosted has physical CPUs and based on number of sockets and cores per socket the licensing is determined.

The formula used here is –

Total cost = Total licenses * cost per license + total licenses * cost per add-on pack + Support cost for each pack

Total licenses = No. of Sockets * No. of cores per Socket * multi core factor

The multi core factor is a fixed value and it is written in the Oracle documentation for each processor. (Please find the link for the doc in references section).

2. Per Named User Licensing

The total cost depends on Number of users using the database. All the connections to the database (application, process, session, human intervention) everything will be considered as a user and whatever has been the maximum limit of connections, that much entities of Oracle licenses to be purchased. This kind of license should be purchased if and only if you know the maximum no. of connections that can be in your database at any point of time.

e.g., in a database where maximum connection made is 5, so 5 Oracle licenses to be purchased. It means 5 licenses of the installed database edition + 5 licenses of each add-on pack being used.

The formula used here is –

Total cost = Total licenses * cost per license + total licenses * cost per add-on pack + Support cost for each pack

Total licenses = No. of Users (Connections :– Human/Non-human)

Note: - User here refers to connections from outside not the users inside database.

Test Scenarios

This test has been performed in our SVT environment's databases, we will be calculating the total cost here for 2 different scenarios – in 2 different operating systems, 2 different Processor architectures and 2 different database architectures (one standalone database and another is a 3-node RAC database). We fetch AWR, ADDM, ASH reports on regular intervals, run tuning advisor/segment advisor and have partitioned tables in our databases. These all are licensed features and one must purchase licenses for all these utilities. We will be discussing only type 1 here i.e., per processor license because in larger scale purchasing per named user license does not make sense.

Oracle has its own way to find out which feature is being used in the database currently or which was used in the past. It maintains its internal tables for this. After creating a database most of the features are ready to use but that does not mean that those should be used without procuring licenses for those. Let's begin with the test:-

- **Standalone Database hosted on Linux**

Getting basic OS details –

$ `uname -a`

`Linux xhhmy92853 2.6.32-504.16.2.el6.x86_64 #1 SMP Tue Mar 10 17:01:00 EDT 2015 x86_64 x86_64 x86_64 GNU/Linux`

$ `cat /etc/redhat-release`

`Red Hat Enterprise Linux Server release 6.6 (Santiago)`

$ `lscpu | grep Architecture`

`Architecture:          x86_64`

Getting number of sockets –

$ `cat /proc/cpuinfo | grep "physical id" | sort -u | wc -l`

4

$ `lscpu| grep "Socket(s)"`

`Socket(s):             `**`4`**

Getting cores per socket –

$ `lscpu| grep "Core(s) per socket"`

`Core(s) per socket:    `**`6`**

Getting Multi core factor –

$ `cat /proc/cpuinfo| grep "model name" | head -1`

`model name     : `**`Intel(R) Xeon(R) CPU E5-4607 v2 @ 2.60GHz`**

Oracle has a document in place for each processor and based on that doc the multi-core factor (also known as core processor licensing factor) for above CPU family is 0.5.

Vendor and Processor	Core Processor Licensing Factor
Intel Xeon Series 56XX, Series 65XX, Series 75XX, Series E7-28XX, E7-28XX v2, Series E7-48XX, E7-48XX v2, E7-48XX v3, Series E7-88XX, E7-88XX v2, E7-88XX v3, Series E5-24XX, Series E5-26XX, E5-26XX v2, Series E5-46XX, E5-46XX v2, Series E5-16XX, Series E3-12XX, E5-26XX v3, E5-24XX v2, E5-16XX v3 and E5-16XX v2 or earlier Multicore chips	0.5

For more information on this doc, please check the web links given in references section. From the above table we can see that the core licensing factor for Intel Xeon E5-4607 is 0.5

Now we have got all the terms to calculate total number of licenses required.

Total licenses = No. of Sockets * No. of cores per Socket * multi core factor

Total licenses   = 4 * 6 * 0.5

**   = 12**

Hence a total of 12 licenses should be purchased for database edition and all the add-on packs those are to be used or currently being used in the database. Now let's check what all the packs are being used and the edition of the database.

Getting Database edition details –

```
SQL> select * from v$version;

BANNER
--------------------------------------------------------------------------
Oracle Database 11g Enterprise Edition Release 11.2.0.4.0 – 64bit
Production
PL/SQL Release 11.2.0.4.0 - Production
CORE     11.2.0.4.0      Production
TNS for Linux: Version 11.2.0.4.0 – Production
NLSRTL Version 11.2.0.4.0 – Production
```

Getting Database feature usage details –

Oracle maintains various internal tables for tracking the usage of its features. Few of the important tables are --

   ✓ wri$_dbu_usage_sample
   ✓ wri$_dbu_feature_usage
   ✓ wri$_dbu_feature_metadata

These 3 tables, after joining on certain internal conditions, form a 'documented' view called DBA_FEATURE_USAGE_STATISTICS. This view gets updated every 7 days (default value) called 'taking samples of database' and keeps the records like feature x was used n number of times on date z with DBID y. the column 'DETECTED_USAGE' acts like a counter and keeps increasing the value whenever it finds the feature is used.

```
SQL> show parameter control_management_pack_access;

NAME                                 TYPE        VALUE
------------------------------------ ----------- ----------------------------
control_management_pack_access       string      DIAGNOSTIC+TUNING
```

The above parameter tells that both the packs diagnostic and tuning can be used irrespective of license procurement. Generally **v$option** tells what all the features are enabled in the database and can be used.

Now we will find out what all the features have been used so far in our database by querying the view **DBA_FEATURE_USAGE_STATISTICS.**

```
SQL> alter session set nls_date_format='dd-mm-yy hh24:mi:ss';

Session altered.

SQL> col name for a55

SQL> set lines 300

SQL> set pages 300

SQL> select distinct name,dbid, version, detected_usages,TOTAL_SAMPLES,
SAMPLE_INTERVAL, currently_used, first_usage_date, last_usage_date  from
DBA_FEATURE_USAGE_STATISTICS where detected_usages > 0 order by 1;
```

The output of the query is very large so cannot be written here hence attached the same.

query_output.txt

PFB, some section from the output so that we can get the details of licensed features.

NAME	DBID	VERSION	DETECTED_USAGES	TOTAL_SAMPLES	LAST_USAGE_DATE
ADDM	806692467	11.2.0.4.0	2	8	21-10-14 01:46:34
ADDM	3303547349	11.2.0.4.0	4	42	05-08-15 00:48:33

ADDM report was taken from the database 6 times – 2 times when the DBID was 806692467 and 4 times when the DBID was 3303547349.AWR, ADDM, ASH all comes under Oracle Diagnostic pack.

```
HeapCompression      806692467 11.2.0.4.0   2    8   30-09-14 00:16:29

HeapCompression  3303547349 11.2.0.4.0   1   42   12-11-14 01:56:31
```

For Heap compression Oracle has licenses under add-on pack named Advanced Compression.

```
Partitioning (user)  803305409 11.2.0.4.0    1   6   11-09-14 00:29:42

Partitioning (user)  806692467 11.2.0.4.0    8   8   04-11-14 01:52:47

Partitioning (user) 3303547349 11.2.0.4.0   42  42   19-08-15 01:02:15
```

Partitioning is again a licensed pack, as we have partitioned tables in our database hence Oracle detects its usage every time whenever a sample was taken (42 out of 42).

```
SQL Tuning Advisor    806692467 11.2.0.4.0   2   8   21-10-14 01:46:34

SQL Tuning Advisor   3303547349 11.2.0.4.0   3  42   24-06-15 00:39:06
```

The above feature which was used 5 times in the database, requires a license called Tuning pack. For more information on which pack comes under which license please refer the links given in references section.

So based on above findings we can make a table as below:-

OS	CPU	DB Architecture	Database Edition	Add-ons Being Used
Linux	Intel Xeon Series E5-46XX	Standalone	Enterprise Edition	Diagnostic
				Tuning
				Partitioning
				Advanced Compression

All the entries in the table have a separate cost associated with them. They do have different support cost as well. For pricing of each pack please refer the link in references section. So our formula was -

Total cost = Total licenses * cost per license + total licenses * cost per add-on pack + Support cost for each pack

Now we have already got the first term i.e., Total licenses which was 12. So the final cost would be-

Database edition and add-ons packs	Cost per license in USD	Total Licenses	Support Cost	Sum (cost per license * total licenses + support cost)
Enterprise edition	47500	12	10450	580450
Diagnostic pack	7500	12	1650	91650
Tuning pack	5000	12	1100	61100
Advanced compression	11500	12	2530	140530
Partitioning	11500	12	2530	140530

SO the total cost for a standalone enterprise edition database with 4 add-on packs costs just under INR 7 Cr. In case extended support required for any old version then there will be some more addition in these figures.

Grant Total	USD	1014260
	INR	66941160

- ▪ **3-node RAC Database hosted on Solaris**

Getting basic OS details –

```
$ uname -a
```

```
SunOS ssdb0188 5.10 Generic_150400-03 sun4u sparc SUNW,Sun-Fire-15000
```

```
$ isainfo -kv
```

```
64-bit sparcv9 kernel modules
```

Getting number of sockets –

```
$ echo "`psrinfo -p` socket(s)"
```

8 socket(s)

Getting cores per socket –

```
$ echo "`kstat -m cpu_info|grep -w core_id|uniq|wc -l` core(s) "
```

16 core(s)

That means cores per sector are – 16/8 = **2**

Getting Multi core factor –

```
$ psrinfo -pv | head -2
```

The physical processor has 2 virtual processors (384 388)

UltraSPARC-IV+ (portid 384 impl 0x19 ver 0x22 clock 1500 MHz)

Vendor and Processor	Core Processor Licensing Factor
Sun UltraSPARC IV, IV+, or earlier Multicore chips	0.75

For the processor family Sun UltraSPARC IV+ the multi-core factor is 0.75. Now let's calculate the total number of licenses required for this database. The formula for a RAC database differs here a little bit than the formula of standalone database. For a standalone the formula for getting total licenses was –

***<mark>Total licenses</mark>** **= No. of Sockets * No. of cores per Socket * multi core factor*

But for RAC database, Oracle considers number of instances (or number of servers where different RAC nodes exist), so the formula here is-

***<mark>Total licenses</mark>** **= No. of Sockets * No. of cores per Socket * No. of Instances * multi core factor*

Total licenses **= 8 * 2 * 3 * 0.75**

= 36

Hence a total of 36 licenses should be purchased for database edition and all the add-on packs those are to be used or currently being used in the database. We checked the view **DBA_FEATURE_USAGE_STATISTICS** for this database as we checked in the database above and formed below table.

OS	CPU	DB Architecture	Database Edition	Add-ons Being Used
Solaris	SUN UltraSPARC IV+	3-node RAC	Enterprise Edition	Diagnostic
				Tuning
				Partitioning
				Real Application Cluster

Now applying the formula on the above we can calculate the total cost of this database.

Database edition and add-ons packs	Cost per license in USD	Total Licenses	Support Cost)	Sum(cost per license * total licenses + support cost
Enterprise edition	47500	36	10450	1720450
Real Application Cluster	23000	36	5060	833060
Diagnostic pack	7500	36	1650	271650
Tuning pack	5000	36	1100	181100
Partitioning	11500	36	2530	416530

So for a 3-node RAC database with 4 add-on packs like above, you have to pay around INR 22.5 Cr.

Grant Total	USD	3422790
	INR	225904140

Now after the above analysis we can see that the cost of just 2 databases touched nearly INR 30 Crores. Might be now we have got the answer of the question which I asked earlier – why small scale companies can't find it feasible to go with Oracle database.

But the good thing which I can say as concluding remark is –

As a DBA we administer a database which is 100 to 10000 times more expensive than our salary. ☺

De-installing 'unwanted' features

Now the question arises can we reduce the licensing cost of Oracle, as we can see by default most of the option are enabled and can be used accidently or unintentionally. So is there any mechanism in place to restrict the usage of some unwanted features?

The answer is yes!! This method is useful for those small databases or companies who don't want to pay this hefty amount by ignoring some 'unwanted/advanced' features but still want to exploit all the other advantages provided by Oracle.

The methods are –

- **Parameter file changes**

Oracle database forces customers to use diagnostic and tuning pack as this option is enabled by default after creating the database.

```
SQL> show parameter control_management_pack_access;

NAME                                 TYPE        VALUE
------------------------------------ ----------- ---------------------------
control_management_pack_access       string      DIAGNOSTIC+TUNING
```

To disable this option just set the value to none so that neither of the packs can be used in the database. This is a dynamic parameter so can be changed without bounce.

```
SQL> alter system set control_management_pack_access=none;

System altered.

SQL> sho parameter control_management_pack_access;

NAME                                 TYPE        VALUE
------------------------------------ ----------- ----------------------

control_management_pack_access       string      NONE
```

Doing the above is useful for those customers who have their home-grown script or they are using stats-pack report (no licensing required) instead of AWR/ADDM etc. After changing the parameter we have saved the license cost of Diagnostic and Tuning pack.

Disabling the above parameter does not mean Oracle will not collect internal stats or MMON will not flush instance statistics to sysaux tablespace, rather Oracle internally will still collect the stats for its internal purpose but we will not be able to generate a human friendly report like AWR or ADDM. This method is only useful when you do not need instance statistics frequently or do not require recommendations from ADDM or do not require recommendations from SQL tuning advisor. In this case stats pack report or any home-grown script can be used.

After changing this parameter you have saved USD 12000 (license cost) per license + USD 2750 (support cost) which can give you a big relief but this method works only for diagnostic and tuning pack, For other options we can use Method 2 or 3.

- **Chopt (Change Options) utility**

From 11gR2 onwards Oracle has provided chopt utility to de-install some specific products which were mistakenly installed or products which are to be removed for their extra licensing cost. It allows you to remove those specific features that you are not licensed to use, so that you can reduce the likelihoods of accidental usage of extra features.

Chopt is a utility which resides in the directory $ORACLE_HOME/bin.

```
$ chopt

usage:

chopt <enable|disable> <option>

options:

                dm = Oracle Data Mining RDBMS Files

                dv = Oracle Database Vault option

                lbac = Oracle Label Security

                olap = Oracle OLAP

                partitioning = Oracle Partitioning

                rat = Oracle Real Application Testing

e.g. chopt enable rat
```

 Suppose we are not intended to use partitioned tables in our database so we can remove this option. In this case there is no need to keep this option enabled as it can be used accidently. The disabling is very easy through chopt.

```
SQL> select * from v$option where parameter='Partitioning';

PARAMETER                          VALUE

-------------------        -------------------------------

Partitioning                       TRUE
```

Shutdown the database now and then run chopt utility to disable the partitioning option.

```
SQL> shutdown immediate;

Database closed.

Database dismounted.

ORACLE instance shut down.

SQL> exit
```

The chopt utility automatically runs make –f command in background to enable/disable the option but if it does not work, it will generate the commands and that can be run by user, the same as below-

```
$ chopt disable partitioning

Writing to
/u01/app/oracle/product/11.2.0/dbhome_1/install/disable_partitioning.log...

%s_unixOSDMakePath% -f
/u01/app/oracle/product/11.2.0/dbhome_1/rdbms/lib/ins_rdbms.mk part_off

%s_unixOSDMakePath% -f
/u01/app/oracle/product/11.2.0/dbhome_1/rdbms/lib/ins_rdbms.mk ioracle
```

Here it could not run the make automatically so it generated the commands to be run in a specific order.

```
$ make -f /u01/app/oracle/product/11.2.0/dbhome_1/rdbms/lib/ins_rdbms.mk
part_off

/usr/bin/ar cr /u01/app/oracle/product/11.2.0/dbhome_1/rdbms/lib/libknlopt.a
/u01/app/oracle/product/11.2.0/dbhome_1/rdbms/lib/ksnkkpo.o

$ make -f /u01/app/oracle/product/11.2.0/dbhome_1/rdbms/lib/ins_rdbms.mk
ioracle
```

As both the generated commands have been executed now, so let's start the database to check if the value of the parameter is changed.

```
SQL> startup

SQL>  select * from v$option where parameter='Partitioning';

PARAMETER                          VALUE

-------------------                --------------------------------

Partitioning                       FALSE
```

The option is disabled now and we have saved a significant licensing cost because by default this option was enabled and could be used accidently.

In the same manner we can disable other below features as well in the database.

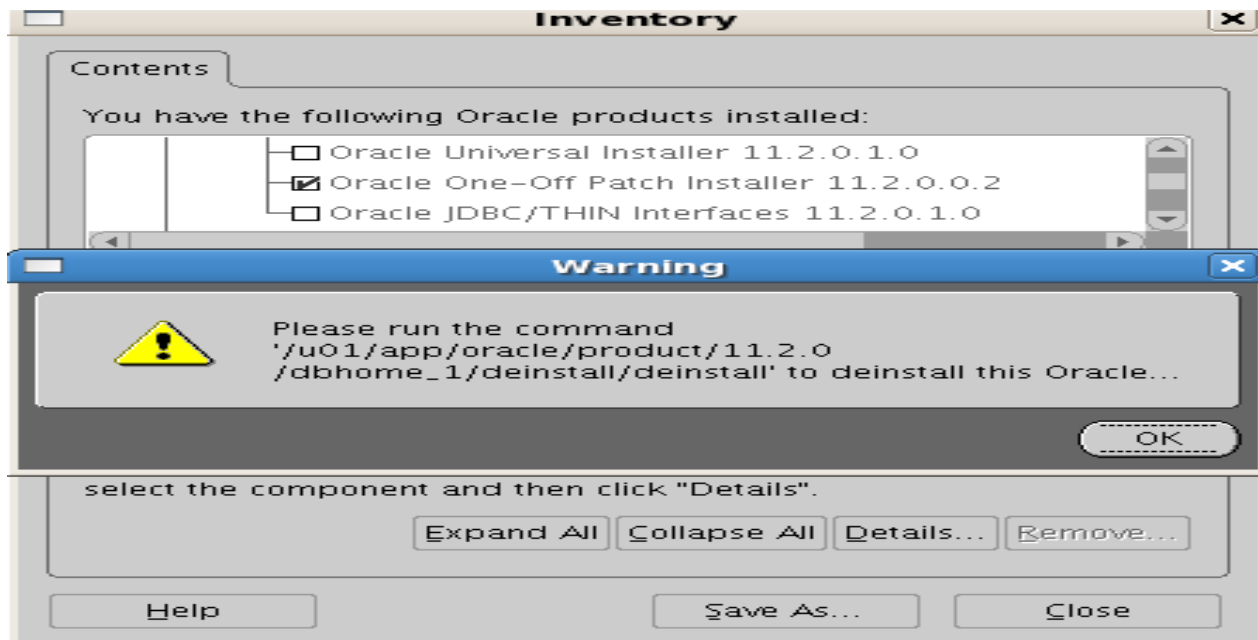| Product/Component	Option
Oracle Data Mining	dm
Database Vault	dv
Oracle OLAP	olap
Oracle Label Security	lbac
Oracle Partitioning	partitioning
Real Application Testing	rat

The above table shows the options that can be disabled/re-enabled through chopt utility.

- **Using OUI (Oracle Universal Installer)**

Navigate to the bin directory under Oracle home and invoke runInstaller.

```
$ cd $ORACLE_HOME/bin

$ ./runInstaller
Starting Oracle Universal Installer...
```
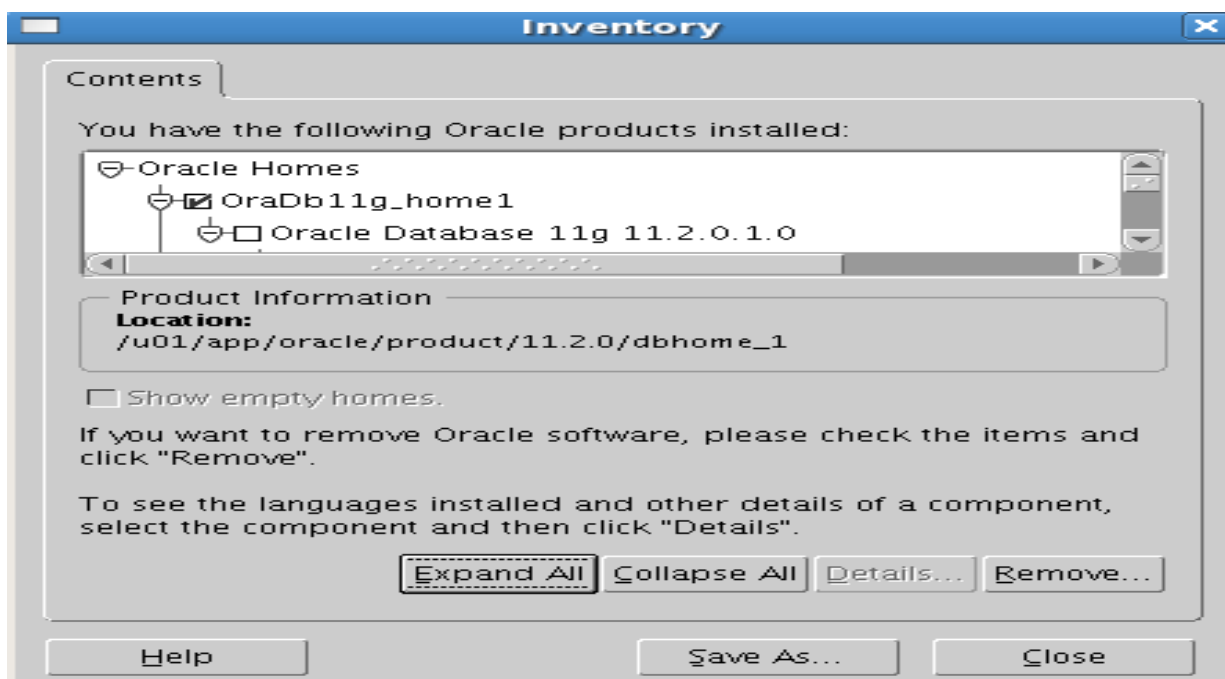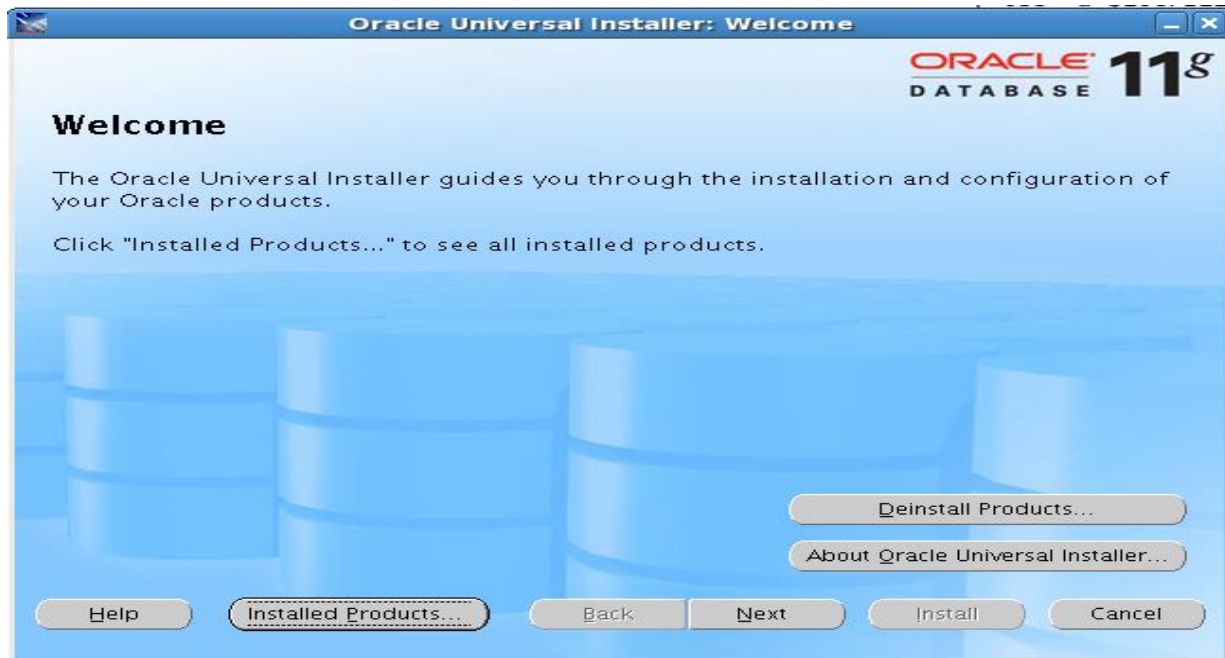
Click on installed products → expand all → check the component → remove

Using this method also the components can be de-installed or installed again.

Reference(s)

1. http://www.oracle.com/us/corporate/pricing/technology-price-list-070617.pdf

2. http://www.oracle.com/us/corporate/contracts/processor-core-factor-table-070634.pdf

3. http://www.oraclealchemist.com/news/a-few-words-on-oracle-licenses/

4. http://docs.oracle.com/cd/E11882_01/license.112/e47877/options.htm#DBLIC173