Features, strengths and weaknesses comparison between MS SQL 2005 (Yukon) and Oracle 10g databases

A Technical White Paper
By
David Gornshtein
Boris Tamarkin
WisdomForce Technologies, Inc
http://www.wisdomforce.com

Revision 1.2

This topic contains alternative comparison between latest versions of Oracle 10g and Microsoft SQL Server 2005 databases.

Table of Contents

Preface

The purpose of the following white paper is to provide a feature comparison between MSSQL 2005[1] and Oracle 10g. We will compare VLDB/OLTP related features and discuss issues with performance, utilities and replication. We will discuss several new features, which were developed by Microsoft in order to provide competitive functionality to its commercial rival Oracle database. At the same time we will discuss the changes from a Database Administrator stand point that were done to MSSQL 2005 release compared to previous releases. Since this topic is intended mostly to Database Administrators, we will not add information such as minimum requirements, supported platforms and max number of columns per table as redundant. We have never seen anyone purchase a database product only because it can hold up to 32k columns per table and not "just" 1024.

Microsoft published "Top 10 Features for Database Administration" (actually all of them are very nice features). Nearly all of these features are present in Oracle since early 8i or prior. The "Snapshot Isolation" is only a kind of superset feature that provides midway functionality to Oracle row versioning although with such cons as higher overhead. However MSSQL database was and still is several steps ahead of Oracle in such great features like the ease of installation, configuration, performing basic tuning and the use of development tools.
When comparing Oracle 8i/9i with MSSQL 2000 for VLDB/OLTP and enterprise applications on high end machines, the important limitation of Oracle always was the need for a highly skillful Oracle DBA, while almost every experienced MSSQL developer can perform DBA role successfully.

Early conclusion: MS SQL Server made important progress with its new Yukon release. However if you have a complicated application or systems running on high end machines you may still want to consider using Oracle. For department level servers and small/mid range applications the MSSQL server would be a preferred choice.

---

[1] MSSQL 2005 features based on publicly available beta 2

Major limitations of MSSQL prior to Yukon (such as MSSQL 2000)

Locking strategy

The main problem of the MSSQL 2000 compared with Oracle 8i/9i for VLDB/OLTP and enterprise applications on the high-end machines was the well-known MSSQL locking strategy. This locking would cause a high lock contention with a very high possibility of deadlocks in the correct transactional flows, which could work well on any databases that provide row versioning. In addition, MSSQL 2000 lock escalation mechanism caused locks to be escalated to the block or even table level in case of high concurrency, which is very common on high end, SMP machines. [2]

Here the impact of this locking strategy from the author's DBA practices:
It was late 2001, when one of our customers had MSSQL 2000 EE installed on an IBM server with 2 CPUs. That customer had an additional backup server of the same configuration. The largest table in the system (large ISP radius server with widely used AAA e.g. authentication, authorization, and accounting) was the accounting history table. This table contained information of connects/disconnects made to the ISP by its customers. The table has grown and became about mid size table (from the Oracle point of view) about 100GB. However the system was based on MSSQL 2000 and not Oracle. As you may know a 100GB table with concurrent inserts is pretty large for an MSSQL database. The customer intended to upgrade their main server to 4 CPUs machine. They switched to a backup server with an empty accounting table and have run it with this configuration for about 2 days. At that time the table already had about 750K rows. Customer then tried to copy those 750K rows to the main table with about 60x10^6 rows (100GB) by using DTS. We performed the test of copying 750K rows via DTS to an empty table, which took about 12 minutes, and then we copied this data to the real, large table. The copying process itself took the same 12 minutes, but immediately after copy MSSQL started to rebalance the 20GB index on this table and this fact halted the upgraded server with 4 XEON CPUs for 5 hours.

---

[2] Even in MSSQL server 7 and/or 2000 you can disable lock escalations by using the undocumented (for these versions) trace flags 1211 or 1224, where 1211 will completely disable lock escalations and 1224 will disable lock escalations until lock structures will consume more then 40% of memory used by MSSQL in the low 2GB (e.g. excluding AWE memory in case of Windows 2000 advanced). To turn on one of these flags (for example 1211) in a single session, run DBCC TRACEON (1211, -1). To do this permanently, on the server level go to the windows service manager, MSSQL service, and add -t1211.
For example, the service definition may look like:
`%SQL_SERVER_HOME%\mssql\binn\sqlservr.exe -t1211`
If both trace flags 1211 and 1224 are turned on, 1224 takes precedence over 1211.
For more details, see http://support.microsoft.com/default.aspx?scid=kb;en-us;323630&sd=tech

The table had no clustered indexes, e.g. it had heap instead of B-Tree organized table[3].

Best guess of what did happen, is that due to a large number of inserts, SQL server escalated the lock to exclusive table level lock.

Note: Here we are coming to an additional MSSQL 2000 problem that seems to be solved in the MSSQL 2005: Dedicated Administrator Connection which will be described with more details in this document in Administration.

Online space reorganization and table/index partitioning

Starting from version 8.1.x and until the current 10g, Oracle has been constantly increasing the number of DDL operations that can be performed online. In Oracle 10g you can rebuild and move an index or even a table without the need to hold an exclusive lock for the duration of the rebuild. However a momentary lock is required to complete the operation. During the rebuild the table is fully operational and can be updated, new indexes can be built and so on. Oracle 9.2 until 9.2.0.5 patch set experienced a number of bugs related to online index and table reorganizations, especially in the RAC environments but it seems that from 9.2.0.5 and 10.1.0.3 most of these issues have been resolved.

In MSSQL 2000 the only online partial index compact / reorganization has been available by DBCC INDEXDEFRAG, however this operation skipped blocks which could not be exclusively locked. MSSQL 2005 has several additional online reorganization options, which will be explained in more details and compared with options available in Oracle 10g in online data reorganization.

---

[3] Now we will try to explain this:
1.   Select from the table without a clustered index, does not returns the rows in the same order, as they have been inserted. The order of inserts has the following effect on a non parallel select:
a.   Non-parallel select will return sorted sequences of values and in most case these sequences will be dense. For instance, table filled with rows having key values of 1…to 100000000, the select (without order by clause) could return the following sequence of keys:
1, 200, 212, 320, … , 440, 441, 446,   2, 53, 78, 719, 729,   4, 9, 10, 15, 28, 35. After say 30000 all (or almost all) of the sequences will be dense, e.g. 40000, 40001, 40002, 40003 … 42000, … ,30000, 30001, 30002, 30003 … 32000, (… Continued on next page footnote)
*Note: Such behavior is not documented but our tests show that MSSQL 2000 works this way.
b.   Sorting these sequences by, for instance, quick sort and merge to the single, dense, sorted sequence or to build B+Tree index from it, is much cheaper (in CPU cycle) than building such index from randomly distributed data.
c.   Most of these sequences will be dense. Per our understanding of this issue, select scans blocks according to IAM index for specific table (without clustered index) begins from the First IAM block. Then the non dense sequences appear after each 8 blocks extents filled up and new block in empty or mixed extent allocated. In addition, according to our tests, if the table is smaller than a single extent (e.g. 8 blocks in size), it will be scanned by the select in the order of insertion.
2.   Let's consider two wide tables that were built with 30 columns each. Both tables contain several variable length columns such that row length is more then 1000 bytes. The first table will have a clustered index. The second table will not have clustered index, but instead will contain a three columns regular, unique index. Filling the first table with a clustered index with 10 million rows will take significantly longer than filling the second one.

Rows to the table have been inserted in a chronological order and up to this moment we have never deleted historical records.

Statistics

MSSQL 2000 had far less time-based numeric statistics than comparable v$ views in Oracle.
In MSSQL 2000 (and in 2005 as well) there is just a small amount of tuning related, time based
statistics. However MSSQL exports rich set of online statistics by using Windows Performance
monitor, which is very visually appealing. The only problem is that taken constantly, the statistics
consume huge amount of disk space. These statistics are very useful and nearly equivalent to the
Oracle 10g OEM provided statistics.

A very small number cumulative statistics is available in MSSQL database since "instance startup"
or another "point in time" by using DBCC PSS, DBCC PROCCACHE, DBCC OPENTRAN and
several online statistics via system stored procedures such as sp_whp, sp_lock and so on.

In addition, starting from MSSQL 7, Microsoft provides MSSQL Profiler and Server-Side Trace,
which provide functionality similar to Oracle event 10046 or sql_trace one. MSSQL 2005 extends
these capabilities with Oracle-style dynamic performance views. The default user "dbo" now owns
all the tables in the database master, which previously began with sys. In MSSQL 2005
master.dbo.sysdatabases became a dynamic view belonging to the user "sys":
master.sys.sysdatabases. (What does this remind you of?☺) . For the details on features comparison
see further in this document the tuning.

Clustering and high availability

Starting from late versions in MSSQL 6.5 and 7 Microsoft provides a fast failover cluster
configuration. This is equivalent to the DataGuard cold failover in Oracle. However the fast
failover in MSSQL is pretty rapid even compared to Oracle 9i RAC Transparent Application
Failover, which is advertised as a superior high-availability solution as compared to DataGuard.
This is due to the fact that all RAC nodes experience "Reconfiguration" on node failure on top of,
for example, Sun Cluster 3.1 halt of at least 10 - 30 seconds. The exact time of "Reconfiguration"
depends on the amount of locks present in the failed node and overall transaction load. At the same
time a well-configured MSSQL 2000 on top of a Microsoft Cluster will failover to the other server
within about 15 seconds.

However, neither MSSQL 2000 nor MSSQL 2005 provides a shared disk cluster based scalability
solution similar to the concept of the Oracle 10g data grid.

Oracle 10g "data grid" in reality is nothing more than just another buzz used mostly in the business,
since Oracle 10g supports grids of 4 to 8 instances on most platforms. In addition, most of today's
storage vendors do not understand exactly, what the term "storage grid" means. However the idea
itself looks promising. We should keep in mind that it will take a long time for MSSQL 2005 to
become a stable release. Perhaps Oracle will have working grids at that time as well.

MSSQL database uses another term for "shared nothing cluster". The idea is to have several servers
and MANUALLY divide the data between these servers. In other words, perform MANUAL
horizontal partitioning of all tables. At same time there is a need to create a special type of
partitioned views where each view is a union of tables identified by linked server definitions in the
style of server_name.database_name.owner_name.table_name. For instance, it can be
history_server_1999.sells.dbo.order_lines, where table order_lines, which resides on server linked
as "history_server_1999" has constraint on a partitioning column. The constraint ranges of all the
tables involved in the union view must not overlap.

All this might sounds good, but it is not as good as Oracle shared disk cluster configuration. In order to scale out Oracle RAC, you need just to install Oracle software on another machine, create several configuration files, and add redo logs to the thread. Then you can startup additional instances without performing a single change to the application's database schema. Only several instance level changes are necessary such as the addition of an undo tablespace and redo log groups to every new instance. However, these changes do not require shutdown of the working database. Now let's discuss the case when you need to scale out the system on MSSQL. MSSQL with its share nothing architecture, requires, after adding a new sever, the creation of additional tables for each cluster partitioned view you have, move part of the data, rebuild all existing tables and views. Such an operation requires a significant maintenance effort and cannot be called "add node on demand easily" while in Oracle 10g RAC such operation can be done within the minutes. In addition, Oracle RAC in versions prior to 10g (7/8/8i OPS and 9i RAC) has been running on top of OS specific cluster solution such as Veritas cluster, HP MC Service Guard, Sun Cluster etc. (although from OPS 8i Oracle has it's own cluster services for Windows e.g. OSD cluster software and from RAC 9i also on Linux) this required additional installation, licenses and additional expenses. The new Oracle 10g comes with a built in cluster solution, called "Cluster Ready Services (CRS)".

In addition to free proprietary clusterware, Oracle 10g provides an automatic storage management solution, called "Automatic Storage Management (ASM)". ASM is supposed to simplify 10g RAC storage management as compared to raw devices. Be aware though that all these features are yet quite unstable. Even the current release is called "release version with several patch sets", e.g. current is 10.1.0.3. We should add here a real life account.

Several Oracle DBAs in one large billing solution provider tried to install 10g Cluster Ready Services on HPUX 11i. There is no documentation about existing issues. Even Oracle support was not able to find the support engineer or BDE (Oracle Support Engineer having directly connection to development unit) who has a good understanding of Oracle clusterware, since this subject is closer to system management rather then to the database. Even Oracle RAC PAC team, which has always been the best source for RAC related problems solution, had expertise mostly in Linux based CRS.

So, those DBAs realized that in order to install 10g RAC on HPUX they need MC Service Guard with HP Serviceguard Extension for RAC exactly as they need it when install Oracle 9i RAC. This is despite of Oracle marketing information that Oracle brings proprietary clusterware for 10g RAC. The guys installed MC Service Guard, HP Serviceguard Extension, Oracle 10g CRS on HPUX, installed Oracle, ran several tests and everything was fine. When the tests completed, they wanted to simply uninstall CRS from HPUX servers. And here problems started. There is a guide on how to install CRS (and all this installation is under root account) but there is not even a single document on how to uninstall the CRS and there are several processes running under root account. The system administrators decided to "kill -9" all these process, remove CRS from all the init scripts hopped to end it there. However after they did "kill -9" both servers restarted and restarted and restarted forever ... until HPUX OS has been reinstalled.

So, although 10g grid idea is much better than a specific MSSQL share nothing cluster implementation, it is still far from being simple in installation and use.

The idea of a share nothing cluster itself has several pros and cons over other shared disk clusters, but describing this would require an additional topic.

By the way, a share nothing cluster could be implemented more efficiently than it was done in SQL Server. For instance, consider IBM EEE DBi2 v8 share nothing cluster. IBM allows easily creating a DB2 instance on additional computer, creating a partition and defining its configuration, adding partition on additional computer and using a single command "redistribute nodegroup GLOBAL uniform" you will redistribute all the tables. Just be sure, that the partitioning key (unique or not) is defined on all the tables in that node group.

Administration

MSSQL 2005 administration is as simple as every Microsoft product administration with a nice GUI and only a few parameters required a manual change.
In addition, in the case of the not so uncommon SQL server halt (it is much more common that all Windows would halt together with SQL server) the special Dedicated Administrator Connection has been added.

MSSQL	Oracle
a) Dedicated Administrator Connection is allowed only from SQLCMD command line utility for these MSSQL server DBAs that never used command line. The expected syntax is: `SQLCMD  -S davidg01\COMP01 -U system -P manager -A` where `system` is any user having sysadmin fixed server role (my test user) and manager is password davidg01\COMP01 SQL server instance `-A` means this is Dedicated Administrator Connection. *Unfortunately, this option does not really works in MSSQL 2005 beta 2*	a) `sqlplus "/ as sysdba"` connected OS user will become a member of dba group.
b) Additional administration related feature is to allow enforcing policies for SQL Server login passwords and allowing for the separation of owners and schemas. *Unfortunately, this option does not really implemented in MSSQL 2005 beta 2*	b) Oracle provides password complexity enforcement rule started from version 7. This is implemented in function sys.verify_function, where dem function defined in file $ORACLE_HOME/rdbms/admin/utlpwdmg.sql Starting from version 8i, Oracle provides an option to connect with some user and then change the schema to different user. For instance, if you connected with user "scott", but want to work in schema "apps". It is possible to

	do by executing
	`" ALTER SESSION SET CURRENT_SCHEMA =`
	`apps;"`

Locking

The most notable feature added in MSSQL 2005 is probably a new isolation level called Snapshot Isolation (SI). The idea has been to add row versioning option to MSSQL so that

- Updates will not block the select operation.
- If working in so called "Read committed snapshot" isolation level which is equivalent to SCN based consistent read mode, then all fetched rows will be returned in the same state as it just was <u>on select statement execution</u>.
- If working in so called "Snapshot Isolation (SI)" isolation level, then all fetched rows will be returned in the same state as they were <u>on transaction begin</u>

The only reason to use (SI) mode is probably to provide an option to create a report that represents a consistent snapshot of complete system in point-in-time. (SI) mode has no direct equivalent mode in Oracle. It can be simulated although by, for instance, Oracle Flashback Query mode that is a switch to flashback mode in some point-in-time and create report using a consistent snapshot of the system.

There are three general differences between Oracle and MSSQL in the row versioning implementation:
  I. Oracle implemented the row versioning on the DB block level, while Microsoft implemented it on the level of each single row. Microsoft claims that its own implementation is better and faster. However we did not see any evidence of this claim since Oracle provides similar performance with block based row versioning on the same hardware and OS compared to MSSQL with no row versioning (please see the TPCC results below).
 II. Oracle uses undo segments to store undo records while Microsoft uses TempDB for that same purpose.
III. Oracle metadata is managed in the same manner as table data. So during data querying most of online DDL statements on the table (add or drop indexes, add partitions and even reorganize space) can be performed simultaneously. The same is also correct in the case of sessions when using some kind of Flashback Query. On the other hand, in SQL Server all DDL operations that are currently running on tables belong to database "Snapshot Isolation". "Snapshot Isolation" queries are prohibited.

After we went through reading Microsoft TechNet note about SQL Server 2005 Beta 2 Snapshot Isolation, there is couple of things we want to mention.

We are left very skeptical after looking at the terms of optimistic/pessimistic concurrency control. There are many reasons for working in non-blocking mode rather than blocking mode. While

working in non-blocking mode, Oracle database's performance is similar to MS SQL working in blocking mode when considered all aspects: of scalability, manageability, simplicity of design and code.

Several SQL Server whitepapers (for instance http://www.microsoft.com/technet/prodtechnol/sql/2005/SQL05B.mspx) mention that it is recommended working using pessimistic locking mode in order to achieve better performance. However Oracle, which is always running with an optimistic locking mode, achieves the same performance on the same hardware as SQL Server with pessimistic locking mode. It looks like SQL Server just needs to optimize the code a little bit. For example, look at the following TPCC results on Unisys ES7000 Aries 420 Enterprise Server.

Note: since Oracle database license is more expensive than SQL Server, the hardware used for the MSSQL 2000 benchmark was somewhat better:

Database	tpcc/m	$/tpcc
Oracle Database 10g EE	291413	4.98 US $
Microsoft SQL 2000 EE	309036	4.49 US $

From the results table above it is easy to see that MSSQL performed 6% better. However Oracle runs more efficiently on UNIX platform and not on Windows. Apparently, Oracle database provides nearly the same performance as MSSQL, which uses pessimistic locking, while Oracle using optimistic locking.

In addition, the MSSQL Server whitepaper in some places compares MSSQL 2005 with some early version of Oracle 8i or bellow. It is similar to comparing the latest StarOffice7 with Microsoft Word 2.0. For instance, the white paper discusses manual rollback segments management.

The table below analyzes the document "SQL Server 2005 Beta 2 Snapshot Isolation" (http://www.microsoft.com/technet/prodtechnol/sql/2005/SQL05B.mspx) published on July 13, 2004. We found four facts that we have a different opinion on how authors of that document are criticizing Oracle database in favor of MSSQL Server. Our opinion is based on expertise and experience.

"SQL Server 2005 Beta 2 Snapshot Isolation"	**Our opinion**
Requires complex configuration of ROLLBACK SEGMENTS (definition & on/offline) & transaction level USE ROLLBACK SEGMENT statements to avoid ORA-01555: "Snapshot too old." caused by "long running" transactions overwriting their versioned pages in the rollback segment. *Note: Oracle does not have a definition for "long running" transactions*	1. Starting from Oracle 9i the configuration of ROLLBACK SEGMENTS is automatic. 2. In Oracle10g a new parameter UNDO_RETENTION_PERIOD insures minimum undo retention period. This will insure ORA-01555 will not occur for transaction as long as needed. *Oracle doesn't need to have a definition for "long running" transactions*

ROLLBACK SEGMENTS don't support PCTINCREASE and hence don't "autogrow" so you must get the size right when they are created	Undo segments will be increased automatically until undo tablespace is filled. After that undo tablespace datafiles will be automatically extended (if defined to be auto extend)

MSSQL 2005	Oracle	There is no need for those pessimistic locking modes as the same performance is achieved with Oracle optimistic locking mode
READ COMMITTED (locking)	No Equivalent	
REPEATABLE READ	No Equivalent	
SERIALIZABLE	No Equivalent	

Must handle conflicts (ORA-08177: data page updated outside of the transaction) & retry failed transactions. Requires use of INITRANS >= 3 & MAXTRANS on CREATE/ALTER TABLE DDL to enable space for on-page transaction information before SERIALIZABLE can be used. Page-based data versioning – Oracle must reconstruct the entire page. When using SERIALIZABLE access to other rows on an updated page by other transactions causes ORA-08177: "Can't serialize access for this transaction."	There is no need for those pessimistic locking modes as the same performance is achieved with Oracle optimistic locking mode. If you are an Oracle DBA then you have probably never used or seen anyone use SERIALIZABLE on Oracle.
Application always sees potentially stale data, as there is no choice between concurrency models.	Use "select for update" if there is a need for fresh, unchanged data.

Online data reorganization

Historically, MSSQL provided a very limited online data and index reorganization as compared to competing databases such as Oracle or DB2. Prior to MSSQL 2005 there was only DBCC INDEXDEFRAG operation which was comparable to Oracle "alter index coalesce" that could be performed without an exclusive lock on the underlying table. Statements such as "DBCC DBREINDEX" statement or "`CREATE INDEX ... WITH DROP_EXISTING`" used to cause exclusive locks. Starting from MSSQL 2005 the online index options[4] will allow concurrent modifications (such as updates, deletes, and inserts) to the underlying table or clustered index data and any associated indexes during index data definition language (DDL) execution. For example, while a clustered index is being rebuilt, you can continue to make updates to the underlying data and perform queries against the data.

Well, Oracle has the option to rebuild most kind of indexes or even a whole table without the need to hold an exclusive lock during the rebuild operation. However a momentary lock is required to complete the operation. We expect Microsoft to provide an option for online table reorganization or movement to another filegroup online.

---

[4] All these options are explained in the MSSQL online manual. We will only provide some example for each option. There are just 2 main cases when indexes should be rebuilt either in Oracle or MSSQL:

Case A:

The index becomes highly skewed or fragmented because of significant number if deletes. Since MSSQL B-Tree indexes are balanced, the case of a skewed index is not relevant, but especially in the case of a clustered index and a high DML concurrency, the index may became heavily fragmented.

To identify such cases use "DBCC SHOWCONTIG" and look at the Level in MSSQL and look at the BLEVEL column of user_indexes / dba_ind_partitions in Oracle (just insure that the indexes has been analyzed). In addition, in the case of Oracle you can dump several index blocks on the different levels to see that index is not very skewed and or fragmented.

To perform a treedump in Oracle:

    ALTER SESSION SET EVENTS 'IMMEDIATE TRACE NAME TREEDUMP LEVEL XXXXX';

Where XXXXX is object id for specific index. In Oracle in such case you should run "`alter index index_name rebuild`" statement or "`alter index index_name coalesce`" just to remove empty blocks from the index structure and readjust the high water mark for segment. In Oracle 10g the equivalent statement is "`ALTER INDEX ... SHRINK SPACE COMPACT`" which instructs Oracle to merge the contents of index blocks where possible to free blocks for reuse. Coalesce operation is equivalent.

Case B.

You are expecting a significant number of inserts and need a near real-time response time. In this case you should rebuild index to adjust pctfree in Oracle or fillfactor and pad_index in MSSQL to avoid frequent index page (MSSQL)/blocks(Oracle) splits which will cause a very serious performance degradation to both Oracle and MSSQL, especially in case of huge and deep B-Tree indexes.

For more in depth information on MSSQL indexes, block splits and so on, you can take a look at http://www.microsoft.com/resources/documentation/msa/idc/all/solution/en-us/rag/ragc06.mspx or read the book "Inside Microsoft SQL Server 2000" by Kalen Delaney. This book is certainly the best book on MS SQL Server. Kalen, we are looking forward to reading "Inside Microsoft SQL Server 2005" :-)

Partitioning

Physical table and index partitioning is something new in MSSQL 2005. In MSSQL 2000 only the updateable union view was described in the discussion on clustering.
Oracle provides several fixed types of partitioning. The DBA should choose whether the table would be partitioned based on ranges, hash or list. However these partitioning schemes do cover the most useful situations. In the Oracle database the DBA defines the partitioning by using "`create table`" DDL statement. In addition, in order to assure better maintainability, Oracle provides the option for dual level partitioning. For example a table partitioned by range and each partition is partitioned by hash. Such an option makes sense in the case of a multi milliard rows table and is not supported in MSSQL.
However such an option can be simulated, by using a complicated partitioning function.

MSSQL 2005 provides a more flexible partitioning options than Oracle in the field of tables and index partitioning. In MSSQL 2005 partitioning is implemented by a user-defined function in Transact-SQL or any available .NET language.

There are three steps required to define a partitioned table, as described in the MSSQL 2005 Books Online topic "Creating Partitioned Tables and Indexes":

1) Create a partition function to specify how a table that uses the function will be partitioned.
2) Create a partition scheme to specify the placement of a partition function's partitions on filegroups.
3) Create a table or index using the partition scheme.

Multiple tables can use the same partition scheme. Table and index partitioning both work well in MSSQL 2005 beta 2.

Indexes

Oracle provides richer indexing options than MSSQL 2005 does. While MSSQL still has only BTree based indexes comparable to Oracle regular indexes and clustered indexes such as Oracle IOT, Oracle provides bitmap indexes, which work best for columns with low selectivity. In addition, Oracle provides reverse BTree indexes that are good to reduce contention on sequential values inserts. The last index related feature is Oracle key based cluster, which provides better performance on select from several related tables with the same key columns, included into the cluster.

Clustering

MSSQL provides solid high availability (HA) failover clustering support. However, MSSQL performance, in share nothing clustering is very limited, its administration and design is very complex. All clustering views should be redesigned and rebuild in order to scale the cluster out.

We discussed in previous paragraph the major MSSQL limitations in [Clustering and High availability](#) and also how MSSQL server share nothing and view based cluster works.
Oracle 10g adds RAC/Grid based clustering that can be easily scaled out even in standard edition. This is one of the notable advantages of Oracle 10g compared with MSSQL 2005.
Oracle provides a new, built-in software storage management solution called ASM (Automatic Storage Management) that assists with RAC/Grid based cluster deployment. In addition, Oracle provides proprietary and free (!) cluster file system on Linux and Windows platforms that helps in avoiding the use of raw devices.

Tuning

Function based indexes via indexed computed column

Oracle provides an option to create an index on a function applied to a specific column. MSSQL provides with ability to create a so-called "computed column" and the ability to create an index on this column. These features are basically equivalent. Yukon introduces the `PERSISTED` keyword in the `CREATE TABLE` and `ALTER TABLE` statements to physically store computed columns in the table. Their values are updated when any columns that are part of their calculation change. Marking a computed column as `PERSISTED` allows creating an index on a computed column that is deterministic, but not guaranteed to be precise.

MSSQL 2005 Indexed View vs. Oracle Materialized Views

Starting from MSSQL 2000 and from Oracle 8i, both databases started to provide Materialized Views. Those views help in avoiding the expensive heavy join of multiple tables, in particular in Data Warehouse environments. The result of an SQL statement representing a view is materialized on the disk in a form similar to a table. In MSSQL 2000 the materialized views have been limited in the means of data types and functions available.

MSSQL 2005 still has a significant list of restricted functions and aggregations. The Indexed View in MSSQL still has a lot of limitations as compared with the Oracle one. For example, the Indexed View definition could not contain the following very useful aggregations, functions and options, namely derived table (sub query in `FROM` list), reference to another view, `DISTINCT`, `EXISTS`, `NOT EXISTS`, self-join, expressions on aggregate results (e.g. `SUM(x)+SUM(x)`), `STDEV, STDEVP, VAR, VARP, AVG, Subquery, SUM` on nullable expressions and `UNION`.

In MSSQL it is enough to create a clustered index on this view definition in order to make "Indexed View" from regular view definition:

```
CREATE UNIQUE CLUSTERED INDEX index_name on pre_defined_view(column_list)
```

To create "Materialized View" in Oracle, view should be created as materialized:

```
CREATE MATERIALIZED VIEW view_name view_definition (instead of just CREATE VIEW)
```

See our future whitepapers for MSSQL 2005 vs. Oracle 10g dynamic performance views comparison.

Data export/import

Oracle 10g provides with a new utility to perform data export called "data pump". Unfortunately data pump output is yet another proprietary Oracle binary format. Oracle has two export and two import options, exp/data pump and imp/data pump respectively. In addition Oracle provides with the sqlldr utility to import textual data and the so-called external tables, which allow performing SQL statements on structured text files as if they were RDBMS tables.

MSSQL 2005 has two export and import options, namely bcp and DTS. Bcp can export and import data using a portable textual format which can be easily imported to the RDBMS of another vendor. DTS is a high performance, programmable ETL tool with tasks and several wizards, which has no compatible options within Oracle RDBMS.

In addition MSSQL 2005 DTS provides the option to build workflow similar to those built by Oracle Workflow Builder. However DTS is used mostly to populate data warehouses rather then to build business flow. An in-depth comparison between DTS and Workflow features useful to build business logic flow is not in the scope of this whitepaper.

WisdomForce FastReader

WisdomForce FastReader (http://www.wisdomforce.com) is complementing a hole in Oracle database data import/export functionality by providing with ability quickly unload/extract data from large Oracle tables into portable text files with various format. This text format can be easily configured. FastReader extracts the data very efficiently and with no production system overhead by bypassing Oracle engine and using a proprietary algorithm. At the same time FastReader generates input (control files) for high-speed loaders of databases from different vendors such as Oracle (sqlldr), MSSQL (bcp), DB2, MySQL, etc. For more information please visit http://www.wisdomforce.com or download WisdomForce FastReader datasheet at http://www.wisdomforce.com/dweb/resources/docs/fastreader_data_sheet.pdf

Replication

Peer-to-Peer Transactional Replication

MSSQL 2005 includes a new peer-to-peer transactional replication feature that improves support for data scale out using replication. However this feature has the following limitations:
- All participating databases should contain the identical schema.
- Object names, object schema, and publication names should be identical among the participating databases.
- Row and column filtering is not supported in peer-to-peer replication.

Transactional replication might help to scale up. However, the limitations, which are described above make this feature not applicable for real-time BI tasks, such as real-time replication with transformation.

Oracle streams are very similar to MSSQL. However Oracle 9i replication streams rules were extended in Oracle 10g with a new feature called "rules engine". "Rules engine" allows applying online transformation of the replication data of the streams. In addition, Oracle 10g support row and column filtering, column and table mapping, manual conflict resolution procedures, and so on.

Database Mirroring (Extend log shipping, automatic failover to a standby server)

SQL Server 2005 significantly enhances the capabilities of log shipping by providing a database mirroring option. Database mirroring allows continuous streaming of the transaction log from a source server to a single destination server. In the event of failure to the primary system, applications can reconnect to the database on the secondary server almost immediately, without waiting for recovery to finish. The secondary database instance detects the failure of the primary server within several seconds and accepts database connections almost immediately after detecting the failure. Unlike failover clustering, the mirrored server is fully cached and ready to accept workloads because of its synchronized state. Well, what this really means is that you need three MSSQL server instances. However only a single instance is actually active and then up-to-date standby database should be available (which can not be accessed by users until failover)

Oracle has a very similar mirroring as compared to MSSQL by having Oracle physical standby with dataguard. The only difference is that with Oracle just two instances are required: active and passive.

Backup and Recovery

Attach/restore database

By using Online Restore with SQL Server 2005, database administrators will be able to perform a restore operation while an instance of SQL Server is running. Online Restore improves the availability of SQL Server because only data being restored is unavailable; the rest of the database remains online and available.

Oracle has a similar structure called a tablespace. For instance, consider transportable tablespace drop and import that can be done while Oracle is up and running. However it is impossible to take old (like with SCN before current) non-transportable tablespace and restore it to the running instance. Such operation may compromise data consistency. In addition Oracle tablespace is not a complete entity, since dictionary metadata is used to describe structure of the objects hold in each tablespace.

In our opinion, MSSQL databases structure is simpler and designed better than Oracle tablespace. Each MSSQL database contains its own dictionary. On other hand, Oracle has more recovery options for corrupted database, redo log or datafile than MSSQL.

From our experience, once back in the early 2000 we have seen an MSSQL database have a media failure (three disks where gone at the same time from five disks raid 5 on IBM eSeries server) and on this raid 5 the transaction log resided for a production database. While all datafiles resided on the other disk and were available, we did not succeeded to open database even in the inconsistent state just to export the data via MS bcp or dts.

Here we should note, that since MSSQL server does not have a separate structure to maintain undo information and it is maintained together with redo in the transaction log, such a situation is comparable to a loss of the current redo log together with undo segment corruption which is a very hard situation in Oracle as well. The recovery is complicated if the database can be recovered or even open in the inconsistent state so it is not always possible. We have not found a way (and it might be that such a way doesn't exist at whole) to open MSSQL database in an inconsistent state in such a case (neither now in MSSQL 2005). Oracle does allow opening and we did it in several very hard situations by changing two init.ora parameters:_allow_resetlogs_corruption and _corrupted_rollback_segments. By the way, both those parameters are unsupported by Oracle Support ☺

Microsoft provides some undocumented flags as well (inherited from Sybase SQL server), that could be used in the case of corrupted or completely lost transaction logs. For example, trace flag 3608. If this flag is turned on, MSSQL will skip automatic recovery (at startup) for all databases with exception of the master database. Using the flag (3608) may cause MSSQL to generate a new empty transaction log instead of the lost one. However, in this historical case in year 2000, this flag did not help us. To turn this flag on, just go to the windows service manager, MSSQL service and add -t3608. For example: %MSSQL2000_HOME%\mssql\binn\sqlservr.exe -t3608.

If a transaction log for a database other than master was lost, the new transaction log named %YOUR_DB_NAME%_log.LDF will be created. If there are problems with the master database, it can be easily recreated by rebuildm.exe utility. Having such a utility makes life much easier as compared to the Oracle database when the system tablespace is lost.

20

Oracle has a rich set of logical dumps that can help to identify the problem that needs recovery, online and backup files timestamps, and so on. MSSQL provides several DBCC PAGE and DBCC LOG based dumps. DBCC PAGE is explained well in the Inside Microsoft SQL Server 2000 and DBCC log is documented in the Sybase whitepapers (since MSSQL inherited most of DBCC from Sybase). It is almost impossible to perform some kind of unusual recovery in MSSQL due to a very limited number of parameters available, even if the database administrator understands well the problem.

Hot Backup

Starting from MSSQL 2000, there is an optional VDI C based API available which enables performing hot backup via mirror split. It allows third party storage vendors such as NetApp or EMC and volume manager vendors such as Veritas to use their Mirror/Split based solutions to perform online backups.

Oracle hot backup mode provides an option to split mirrors without compromising data consistency. It is preferred over VDI in the sense of 24/7 operation, since in hot backup mode Oracle instance is fully operational. VDI, on other hand, causes complete suspension of all databases of the MSSQL instance (including select statements) for all the time of the split. Time consumed by such a split is dependant on the replication vendor.

Although Oracle provides an option not to suspend operations for a time of a mirror split for the purposes of online backup, from our experience it does not always work properly. For example, we have seen a heavily loaded Oracle instance on a high end HP SuperDome server connected to EMC DMX storage, when we tried to perform online backup by splitting a BCV volume. We did not succeed to recover on the splitting side of the mirror for several times.

These problems on heavily loaded high end machines are not so rare, so Oracle added two new "`alter system`" options to handle such situations:

- `alter system` QUIESCE `RESTRICTED`
- `alter system` suspend

These statements cause I/O suspension to allow, for instance, safe split of mirror.

Fast Recovery

A new faster recovery option will improve availability of SQL Server databases. Administrators will be able to reconnect to recovering database after the transaction log has been rolled forward. Oracle works in this way since version 8 (or may be even 7.4.3). This just means that previously MSSQL server used a very suboptimal algorithm.

Real life example: several months ago, one of the companies we worked with, performed MSSQL high availability test on a high end Unisys Windows based server. After failure on startup, we observed some strange error: it looked like database recovery was completed. Then the server

started to work but data was corrupted. At the time we have seen this log, it was thought that may be it happened during the rollback even thought the database has not opened yet. After long hours of investigation we didn't found why recovery has completed (roll-forward) but the database still didn't opened. Oracle DBAs would expect the database to open on the end of roll-forward. Anyway, starting from MSSQL 2005 it should work this way as well.

RMAN with Legato and Oracle Job Scheduler vs. MSSQL maintenance plans

Oracle provides with a rich scripting based interface to automatic backup with multiple features, including incremental backups, compressed backups, and automatic backup retention and so on. In addition, Oracle EE distribution includes Legato backup tape manager that supports most kinds of directly attached tapes and some options to catalog tapes.

In Oracle 10g, PL/SQL package DBMS_JOB has been extended and became DBMS_SCHEDULER. This package allows running background jobs such as collect statistics or reorganizes database objects once or recurrently interact with the resource manager and monitor all these tasks execution.

MSSQL provides a feature named "maintenance plans" which functionality is comparable to Oracle Job Scheduler. In addition, maintenance plans allow exact definitions for performing automated backup.

Application features

Exception Handling

TRY/CATCH Transact-SQL construct provides functionality comparable with Oracle PL/SQL exception.

Online Analytical Functions topic

Transact-SQL Enhancements SQL Server 2005 provides new language capabilities for developing scalable database applications. These enhancements include error handling, recursive query capabilities, relational operator PIVOT, APPLY, ROW_NUMBER and other row ranking functions, and more. Starting from Oracle 8i and until 10g, Oracle constantly extending the number of the available Online Analytical functions. In Oracle 10g you can use almost one hundred analytical functions in different combinations. However PIVOT operator used mostly for Online Analytical Functions to Excel or some other spreadsheet integration has no analog in Oracle RDBMS. In order to achieve PIVOT functionality use one of the beans from set of special BI beans in Oracle OLAP options.

Queuing

MSSQL 2005 Service Broker adds reliable, scaleable, distributed, asynchronous functionality to database applications. This feature is basically comparable with Oracle Advanced Queuing, although Oracle Advanced Queue provides much more functionality. It can be persistent/non persistent, synchronous / asynchronous, has much more activation options and so on

Note: we are planning to discuss in depth comparison between Oracle Streams Advanced Queuing and MSSQL 2005 Service Broker in our future white papers.

The main idea is that some application (or database managed code) put messages to the queue/queues while other applications (or database managed code) receive notification via trigger or regular SEND/RECEIVE in MSSQL or via trigger like "Asynchronous Notifications" in Oracle. "Asynchronous Notifications" implemented by Oracle based PL/SQL package or java servlet (using following API functions: sys.aq$_reg_info to create registration objects, sys.aq$_reg_info_list create list of these objects, and sys.dbms_aq.register to register the list) in Oracle or just by check/browse queue in Oracle and provide a mechanism for message processing.
In MSSQL each queue may have a notification trigger in Transact SQL or .NET fired after message arrives to the queue. In both MSSQL and Oracle bulk enqueue / dequeue functions are implemented due to performance reasons.

In our opinion, using a database as a message queue is quite expensive. If persistent queue is really needed, then the preferred way would be using MSMQ on Windows or a Sleepy Cat BerklyDB based queue. From our experience with Oracle AQ, database based queues rarely provide good performance while consume a lot of CPU, disk and other resources.

Possibly after MSSQL 2005 release we will provide with data of a competitive benchmark between both Oracle and SQL Server queuing.

XML Support

Starting from MSSQL 2005, SQL Server provides a way to store, index and query XML documents inside of a relational database, including XML <-> Relational data integration. Oracle has similar feature called Oracle XML DB.

Asynchronous array of processed statements

Multiple Active Result Sets via multiple asynchronous array processing statements processed in the single session context is implemented in MSSQL 2005.
Oracle is able to provide similar functionality by having several sql statements on single connection, performed in the asynchronous mode e.g. via OCI.

.NET in MSSQL vs. Java in Oracle

It is looks like MSSQL 2005 has a serious advantage over Oracle. Java inside Oracle database can be called only indirectly meaning that it should be wrapped by PL/SQL or by using either CORBA or EJB interfaces while .NET in MSSQL can be called absolutely transparently as Transact-SQL. In addition, uploading and maintaining of Java objects in Oracle is the same as maintaining a complete java application server. On the other hand, MSSQL provides a simple interface to develop, debug and use .NET triggers, stored procedures and packages in the MSSQL 2005.

Useful Oracle features that will have no compatible features in MSSQL 2005

Oracle Logminer

Oracle Logminer provides a way to extract the history of all DML/DDL operations performed on the database. This means, that you can take the old redo log and see a complete transactional protocol. It is used for audit, replication or creation of the change rollback script.
MSSQL 2005 has no equivalent functionality, although third party tools such as Lumigent Log Explorer can provide it. However, one of MSSQL 2005 enhancements, namely DDL Event Notifications allows raising events on DDL statements and other system events and processing these event data as messages by using MSSQL service broker queues.

Flashback Query

Another useful feature in the domain of security and human error based disaster recovery is Flashback Query together with "Dropped Objects Recycle Bin" that is new in 10g. The idea is to see a snapshot of database object, for instance see the table in some point-in-time in the past. The only constraint is that undo space is required to store undo records to "rollback" data blocks in memory to the required point-in-time.
MSSQL 2005 does not have similar features, although Flashback Query can be simulated for several points-in-time, for instance, by opening transactions with snapshot isolation. Then all selects queries that are performed within the context of these transactions will return data right to the point-in-time of the first statement of this transaction.
With all respect, Oracle 10g "Dropped Objects Recycle Bin" still cannot be simulated and will not be supported, since in MSSQL 2005 metadata has no row versioning and DDL on all the tables locked by even a shared lock, meaning select is not allowed.

Auditing

With a new init.ora parameter audit_trail = db_extended all sql queries can be audited together with the bind variables values.
In MSSQL the only way to audit is to use MSSQL trace, which has no option to show bind variables, and causes a significant performance overhead.

Statistics History

Since Oracle statistics are actually data blocks in the dictionary tables, the flashback option provides an infrastructure, which allows the DBA to restore statistics up to some time in the past. If the update queries execution plans have changed to the worse, due to some analyze or due to recurrent statistics, it possible to "rollback" statistics and flush shared pool to reparse queries with the old statistics.

Rollback statistics

If you have a heavy transaction which failed and is rolling back, how can you estimate how much time will it rollback and slowdown your instance? Well, in MSSQL server you cannot.

In Oracle before 10g you could estimate by a script, such as this one:
http://www.wisdomforce.com/dweb/resources/docs/complete_rollback_script.pdf

In Oracle 9i and10g you can easily check out Oracle estimation by querying new V$FAST_START_TRANSACTIONS fixed view.

Automatic Workload Repository

Oracle 10g provides with a new set of fixed views named V$SOME_STATISTIC_HISTORY where SOME_STATISTIC is the name of one of the general-purpose views. Those views contain automatically collected historical statistics information divided by workload type or wait event classes with automatic retention mechanism.

Examples of these views are
- v$active_session_history   for v$active_session
- v$servicemetric_history   for v$servicemetric
- v$session_wait_history   for v$session_wait
- v$sysmetric_history   for v$sysmetric
- v$waitclassmetric_history for v$waitclassmetric.

The idea is that if you aware that at some time, less then 24 hours before, yours RDBMS was very busy, or response time was very long, or some application failed due to a database performance reason, you can immediately investigate the problem. However you did not turn on any online statistics collection mechanism.
Since the Automatic Workload Repository is an internal Oracle mechanism, the performance impact is insignificant. Before Oracle 10g the same result could be achieved with costly third party tools such as Precise (Veritas) Indepth or Quest SQLab Vision.

Object Oriented Features such as objects and vararrays

Oracle provides several basic object oriented features. Those features allow Oracle to call themselves ORDBMS or "object relational". However, these features are actually relatively basic and similar to the option of storing in the data raw structures like data, variable length arrays and nested tables. From a pure theoretical point of view, nested tables have no advantages over two correlated tables that can be joined by PK -> FK (primary key -> foreign key). However, these features exist and are nice-to-have. Please note, that the performance will be demoted a little bit when using these features. Performance lost would be relative to standard RDBMS features that can be used to implement the same design patterns.

Optional compilation of PL/SQL to native binary (shared library) via C compiler

Today, most of the development efforts done are toward using more and more virtual machines such as Java, CLR and so on. However, native code still provides better performance. Starting from Oracle 9i, Oracle provides an option to compile PL/SQL code to a native shared library. In most cases, if PL/SQL code contains complex calculation, such compilation will improve performance by dozens percent. However, support of this feature for RAC was not so native in Oracle 9i. Shared libraries were not distributed automatically between nodes. In addition, this feature causes some administration overhead. Starting from Oracle 10g, the support for native PL/SQL compilation has been simplified. All compiled shared libraries are now stored in the database, so RAC support is native and administrative overhead has been minimized.

Useful MSSQL features that have no similar features in Oracle 10g

Identity

One of the most useful features that Oracle historically does not have is the identity. Almost every database provides with an option to generate primary key automatically, while Oracle remains only with sequences. Oracle sequence is a useful mechanism however it causes serious administration overhead. Implementation of something similar to MSSQL Identity by using Oracle sequence would require reflecting the sequence name in the application or creating a trigger for each table/sequence pair.

The second feature is Rules, which can be implemented as Oracle triggers. However, in MSSQL the implementation of Rule is very lightweight and convenient.

The third feature is user defined data types that are easy to use.

Notification services

Notification Services enable businesses to build rich notification applications that deliver personalized and timely information to any device. This information can include stock market alerts, news subscriptions, package delivery alerts, and airline ticket prices. In MS SQL Server 2005 the Notification Services are more tightly integrated with technologies such as Analysis Services and SQL Server Management Studio.

Reporting Services

One of the really nice and useful features provided by MSSQL 2000/2005 is the integrated reporting services which includes PDF creation support and a nice IDE. Although MS SQL 2005 Reporting Services provides less functionality than, for instance, Oracle Reports with Oracle IAS, the MS SQL 2005 Reporting Services are integrated into RDBMS. This is very cost effective since it does not require additional license fees, very simple in installation, use and development.

Replication

MS SQL 2005 provides with built in replication from MSSQL to Oracle and even replication from Oracle to MSSQL. However this feature is broken in MSSQL 2005 Beta 2. The good thing is that according to Microsoft documentation the replication from MSSQL 2005 to Oracle is working via the same Microsoft Log Reader Agent as Microsoft transactional replication, so this replication has good chances to work well.

Accordingly to MSSQL documentation, the replication from Oracle to MSSQL server 2005 is implemented in the form of trigger based replication. Using trigger-based replication is slowing down the replication source, does not support metadata changes and very error prone. For example, on each insert operation on Oracle, three additional operations will be actually performed (then the total number of operations will be four):

- The Original insert
- Trigger   inserting the data to replication queue (another table)
- Microsoft pooling process selects and fetches this data from the replication queue
- Microsoft pooling process deletes this data from the replication queue.

Note*: The latest copy of the document is available at
http://www.wisdomforce.com/dweb/resources/docs/MSSQL2005_ORACLE10g_compare.pdf