ORACLE DATABASE 12c

Comparison of Database Cloud Services

Benchmark Testing Overview

ORACLE WHITE PAPER SEPTEMBER 2016

ORACLE

Table of Contents

Disclaimer

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Preface

Cloud computing is revolutionizing how information is consumed, delivered, and created. The demands for instant access to information is driving changes that impact IT processes and infrastructures, as well as how information is analyzed and the way applications and databases are developed and delivered.

The traditional datacenter with databases running on dedicated servers and storage creates an inflexible database and application divide that slows down deployment, blocks resource sharing, and keeps businesses from responding effectively to rapidly changing conditions.

Many companies are embracing database as a service (DBaaS) in the cloud and the trend is on the increase. Why the strong momentum toward DBaaS adoption?

DBaaS eliminates the need to purchase, build, and manage disparate silos of server and storage systems. It makes database resources and capabilities available online so users can consume them whenever, and wherever, they're needed. This approach lowers operational expenses as you only pay for what you use.

The instant availability of database resources simplifies management, freeing up both developers and database administrators to focus on the higher-value tasks that provide more-direct value to the business.

DBaaS also allows companies to become more agile and therefore competitive.

Introduction

As more customers begin to transition their mission critical systems to the Cloud, it has become increasingly important to choose the right Database Cloud solution that offers agility and elasticity but also industry-leading security, high availability and performance.

To make the comparison easier for customers, Oracle is documenting both a transaction processing and analytic workload that can be used to compare different Cloud solutions.

This whitepaper provides an overview of both workloads and outlines the conditions under which they should be tested.

Cloud OLTP Workload

The Cloud OLTP Workload is not a TPC-C benchmark test. It is not any way comparable to published TPC-C results.

The workload uses the TPC-C schema, transaction definitions, transaction mix, transaction inputs and scaling rules but has no Remote Terminal Emulator, think times or keying times.

A driver program on the System Under Test (SUT) generates the transaction inputs using the same rules that the TPC-C Remote Terminal Emulator uses to generate transaction inputs. No think times or keying times are used.

The metric is OLTP Transactions per Minute, which includes all transaction not just New Orders per Minute.

To match the security and availability requirements of today's cloud customers, the database must be encrypted and the durability requirement is strengthened. The system must preserve the effects of committed transactions and ensure database consistency after the permanent irrecoverable failure of any two durable medium (instead of recovering from the failure of a single durable medium as the TPC-C specification requires).

All other ACID requirements of TPC-C remain in effect, as do all of the other TPC-C rules.

No audit is required. Customers are free to run this benchmark, but Oracle must certify all **published** results for this benchmark to ensure the rules are properly followed.

Cloud Analytic Workload

The Cloud Analytics Workload is not a TPC-H benchmark test. It is not in any way comparable to published TPC-H results.

The Cloud Analytics Workload is run on a scale factor 10,000 database created and loaded according to the rules in the TPC-H specification. The workload uses 13 new analytics queries instead of the 22 TPC-H queries.

The run consists of a power test of the queries. The TPC-H refresh functions are not run, but the database must be able to perform the TPC-H refresh functions. No throughput test is required. The configuration and execution rules of the TPC-H remain in effect.

The metric is the geometric mean of the query timing intervals from the power test.

To match the security and availability requirements of today's cloud customers, the database must be encrypted and the durability requirement is strengthened. The system must preserve the database and the effects of committed updates after the permanent irrecoverable failure of any two durable medium (instead of recovering from the failure of a single durable medium as the TPC-H specification requires).

All other ACID requirements of TPC-H remain in effect, as do all of the other TPC-H rules.

No audit is required. Customers are free to run this benchmark, but Oracle must certify all **published** results for this benchmark to ensure the rules are properly followed.

Conclusion

As more customers begin to transition their mission critical systems to the Cloud, it has become increasingly important to choose the right Database Cloud solution that offers agility and elasticity but also industry-leading security, high availability and performance.

Using the workloads outlined in this paper, customers can compare different Database Cloud services to determine which is right for them.

Appendix A Analytic Queries Used

Below are a description and full syntax for each analytical query used in the comparison.

Query 1.sql

Business question: The Top Brand Query finds the top 10 revenue-generating brands in 1997 and 1998.

```
SELECT *
FROM   (SELECT p_brand,
               SUM(l_extendedprice * ( 1 - l_discount )) AS revenue
        FROM   lineitem,
               part
        WHERE  l_partkey = p_partkey
               AND l_shipdate >= To_date ('1997-01-01', 'YYYY-MM-DD')
               AND l_shipdate < Add_months(To_date ('1997-01-01', 'YYYY-MM-DD'),
                                24)
        GROUP  BY p_brand
        ORDER  BY revenue DESC)
WHERE  ROWNUM <= 10;
```

Query 2.sql

Business question: The Median Customer Query finds the median order price of all German customers.

```
SELECT Median (o_totalprice)
FROM   orders,
       customer,
       nation
WHERE  c_custkey = o_custkey
       AND c_nationkey = n_nationkey
       AND n_name = 'GERMANY';
```

Query 3.sql

Business question: Computes the 75 percentile of orders placed by German customers based on the total price of the order.

```
SELECT Percentile_cont(0.75)
         within GROUP (ORDER BY o_totalprice)
FROM   orders,
       customer,
       nation
WHERE  c_custkey = o_custkey
       AND c_nationkey = n_nationkey
       AND n_name = 'GERMANY';
```

Query 4.sql

Business question: The Median Discount Query finds the median discount given on individual line items of all orders by German customers between January 1, 1995 and December 31, 1995.

```
SELECT Median (l_discount)
FROM   orders,
       customer,
       lineitem,
       nation
WHERE  c_custkey = o_custkey
       AND o_orderkey = l_orderkey
       AND c_nationkey = n_nationkey
       AND n_name = 'GERMANY'
       AND o_orderdate BETWEEN To_date('1995-01-01', 'YYYY-MM-DD') AND
                               To_date('1995-12-31', 'YYYY-MM-DD');
```

Query 5.sql

Business question: The Clerk Summary Report provides a summary pricing report for all line items ordered through a specific clerk within a given week.

```
SELECT SUM(l_quantity)                                          AS sum_qty,
       SUM(l_extendedprice)                                     AS
       sum_base_price,
       SUM(l_extendedprice * ( 1 - l_discount ))                AS
       sum_disc_price,
       SUM(l_extendedprice * ( 1 - l_discount ) * ( 1 + l_tax )) AS sum_charge,
       Avg(l_quantity)                                          AS avg_qty,
       Avg(l_extendedprice)                                     AS avg_price,
       Avg(l_discount)                                          AS avg_disc,
       Count(*)                                                 AS count_order
FROM   lineitem
WHERE  l_orderkey IN (SELECT o_orderkey
                      FROM   orders
                      WHERE  o_orderdate >= To_date('1995-01-01', 'YYYY-MM-DD')
                             AND o_orderdate < To_date( '1995-01-01',
                                                'YYYY-MM-DD') + 6
                             AND o_clerk = 'Clerk#007373565');
```

Query 6.sql

Business question: The Frequent Customer Query lists the top 100 customers with the most orders that have all line items completely fulfilled. Only orders, which contain at least one part with size less than 5 or at least one part of type "Standard Plated Tin", are considered.

```
SELECT * FROM   (SELECT c_name,
                Count(*) ocount
        FROM    orders,
                customer
        WHERE   o_custkey = c_custkey
                AND o_orderstatus = 'F'
                AND ( EXISTS (SELECT 1
                              FROM   lineitem,
                                     part
                              WHERE  l_orderkey = o_orderkey
                                     AND l_partkey = p_partkey
                                     AND p_size < 5)
                      OR EXISTS (SELECT 1
                                 FROM   lineitem,
                                        part
                                 WHERE  l_orderkey = o_orderkey
                                        AND l_partkey = p_partkey
                                        AND p_type = 'STANDARD PLATED TIN') )
        GROUP   BY c_name
        ORDER   BY 1, 2)
WHERE   ROWNUM <= 100;
```

Query 7.sql

Business question: The Low Discount Part Query lists how often parts of a certain size are discounted at a lower rate (2%).

```
SELECT * FROM   (SELECT p_partkey,
                Count(*) ocount
        FROM    lineitem,
                supplier,
                orders,
                part
        WHERE   l_orderkey = o_orderkey
                AND l_partkey = p_partkey
                AND l_suppkey = s_suppkey
                AND l_discount < 0.02
                AND p_size < 41
        GROUP   BY p_partkey
        ORDER   BY 1, 2)
WHERE   ROWNUM <= 100;
```

Query 8.sql

Business question: The Top Revenue Generating Part Query lists the overall top 10 revenue-generating parts in orders that were placed on a specific date, marked by customers as urgent, and had an overall order price greater than $480,000.

```
SELECT *
FROM   (SELECT p_name,
               p_mfgr,
               p_brand,
               p_type,
               p_size,
               p_container,
               p_retailprice,
               p_comment,
               qty,
               qty * p_retailprice
        FROM   (SELECT l_partkey       partkey,
                       SUM(l_quantity) qty
                FROM   lineitem
                WHERE  l_orderkey IN (SELECT o_orderkey
                                      FROM   orders
                                      WHERE  o_orderdate = To_date('1996-04-30',
                                                           'YYYY-MM-DD')
                                             AND o_orderpriority = '1-URGENT'
                                             AND o_totalprice > 480000)
                GROUP  BY l_partkey) PartiallyFullfiledOrders, part
        WHERE  p_partkey = PartiallyFullfiledOrders.partkey
        ORDER  BY qty * p_retailprice)
WHERE  ROWNUM <= 10;
```

Query 9.sql

Business question: The Fulfillable Small Order Query lists detailed line item information, including its supplier information, of orders that can be fulfilled by a single supplier. Only orders placed within a specific month that have no priority specified and whose total price does not exceed $850 are considered.

```
SELECT l.l_shipdate,
       l.l_discount,
       l.l_extendedprice,
       l.l_quantity,
       l.l_returnflag,
       l.l_linestatus,
       l.l_tax,
       l.l_commitdate,
       l.l_receiptdate,
       l.l_shipmode,
       l.l_linenumber,
       l.l_shipinstruct,
       l.l_comment,
       s.s_comment,
       s.s_name,
       s.s_address,
       s.s_phone,
       s.s_acctbal
FROM   (SELECT l_orderkey,
               l_suppkey,
               SUM(l_quantity)   sqty,
               SUM(ps_availqty) aqty
        FROM   lineitem,
               partsupp
        WHERE  l_orderkey IN (SELECT o_orderkey
                              FROM   orders
                              WHERE  o_orderdate BETWEEN To_date('1996-04-01',
                                                         'YYYY-MM-DD')
                                                         AND
                                     Add_months(To_date('1996-04-01',
                                                'YYYY-MM-DD'
                                                ), 1)
                                     AND o_orderpriority = '4-NOT SPECIFIED'
                                     AND o_totalprice < 850)
               AND l_partkey = ps_partkey
        GROUP  BY l_orderkey, l_suppkey) t,
       lineitem l,
       supplier s
WHERE  t.l_orderkey = l.l_orderkey
       AND t.l_suppkey = s.s_suppkey
       AND sqty < aqty;
```

Query 10.sql

Business question: The Not Fulfillable Lineitem Query shows how often parts occur in line items which have no corresponding order and no supplier that can fulfill the order.

```
SELECT * FROM
       (SELECT p_partkey,
               Count(*) ocount
        FROM   lineitem,
               part
        WHERE  l_partkey = p_partkey
               AND NOT EXISTS (SELECT o_orderkey
                               FROM   orders
                               WHERE  o_orderkey = l_orderkey)
               AND NOT EXISTS (SELECT 1
                               FROM   supplier
                               WHERE  l_suppkey = s_suppkey)
               AND l_discount < 1.1
               AND p_size < 45
        GROUP  BY p_partkey
        ORDER  BY 1, 2);
```

Query 11.sql

Business question: The Missing Order and Supplier Query counts the number of parts that are listed in line items for which there are no orders and no suppliers in the database. Only orders with less than 50% discount and specific sizes are investigated.

```
SELECT *
FROM   (SELECT p_partkey,
               Count(*) ocount
        FROM   lineitem,
               part
        WHERE  l_orderkey NOT IN (SELECT o_orderkey
                                  FROM   orders)
               AND l_partkey = p_partkey
               AND l_suppkey NOT IN (SELECT s_suppkey
                                     FROM   supplier)
               AND l_discount < 0.5
               AND p_size < 41
        GROUP  BY p_partkey
        ORDER  BY 1, 2)
WHERE  ROWNUM < 100;
```

Query 12.sql

Business question: The Late Shipping Query lists detailed order information of those orders under $50005 that were placed within a specific year and that were committed to be shipped one month after the order date, but were actually shipped to the customer within 1 year of their order date.

```
SELECT *
FROM   (SELECT o_orderkey,
               o_custkey,
               o_orderdate,
               o_orderstatus,
               o_totalprice,
               o_orderpriority
        FROM   orders
        WHERE  o_totalprice < 50005
               AND o_orderdate >= To_date('1995-01-01', 'YYYY-MM-DD')
               AND o_orderdate < Add_months(To_date('1995-01-01', 'YYYY-MM-DD'),
                                 12)
               AND ( o_orderkey, Add_months(o_orderdate, -1) ) NOT IN (SELECT
                       CASE
                       WHEN l_orderkey > 5 THEN l_orderkey
                       ELSE NULL
                       END,
                       l_commitdate
                                                                       FROM
                   lineitem
                                                                       WHERE
                       l_extendedprice < 1001
                       AND l_shipdate >= To_date('1995-01-01', 'YYYY-MM-DD')
                       AND l_shipdate < Add_months(To_date('1995-01-01',
                                                   'YYYY-MM-DD'),
                                        12))
        ORDER  BY 1, 2, 3, 4, 5)
WHERE  ROWNUM <= 100;
```

Query 13.sql

Business question: The Approximate Supplier Query estimates the number of suppliers that supply specific parts for which no customer complaint was filed. The query provides estimates for all suppliers and lists them in ascending order of part information.

```
SELECT p_brand,
       p_type,
       p_size,
       Approx_count_distinct(ps_suppkey) AS supplier_cnt
FROM   partsupp,
       part
WHERE  p_partkey = ps_partkey
       AND p_brand <> 'Brand#15'
       AND p_type NOT LIKE 'LARGE PLATED%'
       AND p_size IN ( 21 )
       AND ps_suppkey NOT IN (SELECT s_suppkey
                              FROM   supplier
                              WHERE  s_comment LIKE '%Customer%Complaints%')
GROUP  BY p_brand,
          p_type,
          p_size
ORDER  BY supplier_cnt DESC,
          p_brand,
          p_type,
          p_size;          p_brand,
         p_type,
         p_size;
```

Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

ORACLE

CONNECT WITH US


blogs.oracle.com


facebook.com/oracle


twitter.com/oracle


oracle.com

Integrated Cloud Applications & Platform Services

White Paper Comparison Of Database Cloud Services
September 2016


Oracle is committed to developing practices and products that help protect the environment