



DBT-3™ Query Benchmark

Standalone Execution of 21 Standard DBT-3™ Queries

Ingres 9.3 versus Ingres VectorWise 1.0, and Hard Disk versus Flash RAID 5

Author(s): Dr. Wojtek Rappak and Roy Hann
Revision: 0.9
Date: July 2, 2010

Rational Commerce Ltd.
88 Tyrwhitt Road,
London, SE4 1QB
Tel: +44 (0)20 8691 2089

eMail: wr@rationalcommerce.com
www.rationalcommerce.com

DBT-3™ Individual Query Benchmark: Standalone execution of 21 standard DBT-3™ queries									
Report Date	June 30, 2010								
Hardware	Rational Commerce V16 Vector Appliance , with: <ul style="list-style-type: none"> • Dual Intel 5520 Nehalem/Gainestown quad-core processors (8 cores total) • 32GB RAM • 450GB flash memory RAID 5 device • 1.5TB 7200rpm 64mb cache Western Digital hard disk 								
OS	Centos release 5.4 (Final). Kernel 2.6.18								
RDBMS	Ingres VectorWise (IVW) 1.0 build 114 and ‘Conventional’ Ingres 9.3								
Availability Date	IVW 1.0 build 112 was on GA release on June 8, 2010. Build 114 will be released in July 2010								
Database model	Standard DBT-3™ database (derived from TPC-H) geared towards decision support, data warehouse and analytic queries. Model represents multi-regional sales data represented by familiar entities such as lineitem, orders, part, supplier, customer, etc.								
Database generation	The database was generated using the standard DBT-3™ data generation toolkit <i>dbgen</i> .								
DBT-3™ Data Model	<p>The diagram illustrates the DBT-3™ data model with the following tables and their attributes:</p> <ul style="list-style-type: none"> PART (P_) (SF*200,000): PARTKEY, NAME, MFGR, BRAND, TYPE, SIZE, CONTAINER, RETAILPRICE, COMMENT PARTSUPP (PS_) (SF*800,000): PARTKEY, SUPPKEY, AVAILQTY, SUPPLYCOST, COMMENT CUSTOMER (C_) (SF*150,000): CUSTKEY, NAME, ADDRESS, NATIONKEY, PHONE, ACCTBAL, MKTSEGMENT, COMMENT NATION (N_) (25): NATIONKEY, NAME, REGIONKEY, COMMENT SUPPLIER (S_) (SF*10,000): SUPPKEY, NAME, ADDRESS, NATIONKEY, PHONE, ACCTBAL, COMMENT LINEITEM (L_) (SF*6,000,000): ORDERKEY, PARTKEY, SUPPKEY, LINENUMBER, QUANTITY, EXTENDEDPRICE, DISCOUNT, TAX, RETURNFLAG, LINESTATUS, SHIPDATE, COMMITDATE, RECEIPTDATE, SHIPINSTRUCT, SHIPMODE, COMMENT REGION (R_) (5): REGIONKEY, NAME, COMMENT ORDERS (O_) (SF*1,500,000): ORDERKEY, CUSTKEY, ORDERSTATUS, TOTALPRICE, ORDERDATE, ORDER-PRIORITY, CLERK, SHIP-PRIORITY, COMMENT <p>Relationships (Foreign Keys):</p> <ul style="list-style-type: none"> PARTSUPP (PARTKEY) → PART (PARTKEY) PARTSUPP (SUPPKEY) → SUPPLIER (SUPPKEY) CUSTOMER (CUSTKEY) → CUSTOMER (CUSTKEY) CUSTOMER (NATIONKEY) → NATION (NATIONKEY) SUPPLIER (NATIONKEY) → NATION (NATIONKEY) LINEITEM (ORDERKEY) → ORDERS (ORDERKEY) LINEITEM (PARTKEY) → PART (PARTKEY) LINEITEM (SUPPKEY) → SUPPLIER (SUPPKEY) REGION (REGIONKEY) → NATION (REGIONKEY) 								
Scaling	A scaling factor of 30 was used. This equates to a database of approximately 40GB								
Actual Data Volumes: Rows in tables	<table border="0"> <tr> <td>part: 6,000,000</td> <td>partsupp: 24,000,000</td> </tr> <tr> <td>lineitem: 180,000,000</td> <td>orders: 45,000,000</td> </tr> <tr> <td>supplier: 300,000</td> <td>customer: 4,500,000</td> </tr> <tr> <td>nation: 25</td> <td>regions: 5</td> </tr> </table>	part: 6,000,000	partsupp: 24,000,000	lineitem: 180,000,000	orders: 45,000,000	supplier: 300,000	customer: 4,500,000	nation: 25	regions: 5
part: 6,000,000	partsupp: 24,000,000								
lineitem: 180,000,000	orders: 45,000,000								
supplier: 300,000	customer: 4,500,000								
nation: 25	regions: 5								

DBT-3™ Individual Query Benchmark: Standalone execution of 21 standard DBT-3™ queries	
General Approach	<ul style="list-style-type: none"> • This is not a TPC-H benchmark but it uses open source DBT-3 tools which are based on it. ‘DBT-3 is a fair usage implementation of the TPC’s TPC-H Benchmark specification’. See http://osdl/dbt.sourceforge.net/ for more details • the database was generated through the dbgen toolkit that is part of the DBT-3 package. It is written to the TPC-H specification with a data model which follows to the Table Layout section of the TPC-H spec (section 1.4 http://www.tpc.org/tpch/spec/tpch2.11.0.pdf) • The 21 SQL queries used in the benchmark implement the 'Business Questions' set out in the TPC-H spec (sections 2.4 to 2.24), they are coded in concordance to the specified SQL and the mandated types of parameters. • The aim of the benchmark was to compare the timings of identical queries running against identical data repositories held in four different database technologies • Four separate databases were used in the benchmark. The data held in these was identical <ul style="list-style-type: none"> ○ Conventional Ingres with data on standard hard disk (<i>ing hdd</i>) ○ Conventional Ingres with data on ssd flash memory device (<i>ing ssd</i>) ○ Ingres Vectorwise with data on standard hard disk (<i>vw hdd</i>) ○ Ingres Vectorwise with data on ssd (<i>vw ssd</i>) • The benchmark used the 21 standard DBT-3™ queries and ran them one at a time on a dedicated system with no other user processes running. • Each query was run successively four times: ing hdd, ing ssd, vw hdd, vw ssd. The data returned by each query run had to be identical. Different return sets would invalidate the run. None occurred. • Timings measured the length of time between the submission of the query and the completed arrival of the return data set • Where possible, the databases were optimized according DBT-3™ rules. The indexing on the <i>hdd</i> databases was adapted from the postgres DBT-3™ benchmarks. This was used to create secondary indexes. All databases had the same primary key constraint. All Ingres ‘conventional’ tables were ISAM.
Configuration	<ul style="list-style-type: none"> • databases were optimized through the optimizedb utility. This can significantly improve query performance • All data sets were identical except that the two <i>ssd</i> databases were held on the ssd flash memory device

Query run setup and caching	An attempt was made to ensure that each query was unaffected by the previous one. The OS cache was cleared before each query run and the Ingres Vectorwise server was restarted. Ingres cache was cleared using a trace point.
Query run sequence	<p>Clear OS Cache Restart Ingres Vectorwise Clear Ingres Cache Time run of query <i>n</i> against <i>ing hdd</i> database</p> <p>Clear OS Cache Restart Ingres Vectorwise Clear Ingres Cache Time run of query <i>n</i> against <i>ing ssd</i> database</p> <p>Clear OS Cache Restart Ingres Vectorwise Clear Ingres Cache Time run of query <i>n</i> against <i>vw hdd</i> database</p> <p>Clear OS Cache Restart Ingres Vectorwise Clear Ingres Cache Time run of query <i>n</i> against <i>vw ssd</i> database</p> <p>(Repeat for query <i>n+1</i> up to query 21)</p>

DBT-3™ QUERY 1	
Business Question	DBT-3™/TPC-H Pricing Summary Report Query (Q1) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre>select l_returnflag, l_linestatus, 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_shipdate <= date('1998-12-01') -'90 days' group by l_returnflag, l_linestatus order by l_returnflag, l_linestatus;</pre>	
Rows returned	4

Benchmark timings:

DBT-3™ QUERY 1	
Run Type	Time (hh:mm:ss)
Ingres hdd	0:11:46
Ingres ssd	0:16:07
VW hdd	0:00:39
VW ssd	0:00:08

DBT-3™ QUERY 2	
Business Question	DBT-3™/TPC-H Minimum Cost Supplier Query (Q2) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre> select s_acctbal, s_name, n_name, p_partkey, p_mfgr, s_address, s_phone, s_comment from part, supplier, partsupp, nation, region where p_partkey = ps_partkey and s_suppkey = ps_suppkey and p_size = 15 and p_type like '%BRASS' and s_nationkey = n_nationkey and n_regionkey = r_regionkey and r_name = 'EUROPE' and ps_supplycost = (select min(ps_supplycost) from partsupp, supplier, nation, region where p_partkey = ps_partkey and s_suppkey = ps_suppkey and s_nationkey = n_nationkey and n_regionkey = r_regionkey and r_name = 'EUROPE') order by s_acctbal desc, n_name, s_name, p_partkey </pre>	
Rows returned	14,155

Benchmark timings:

DBT-3™ QUERY 2	
Run Type	Time (hh:mm:ss)
Ingres hdd	0:01:52
Ingres ssd	0:00:50
VW hdd	0:00:07
VW ssd	0:00:02

DBT-3™ QUERY 3	
Business Question	DBT-3™/TPC-H Shipping Priority Query (Q3) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre> select l_orderkey, sum(l_extendedprice * (1 - l_discount)) as revenue, o_orderdate, o_shippriority from customer, orders, lineitem where c_mktsegment = 'BUILDING' and c_custkey = o_custkey and l_orderkey = o_orderkey and o_orderdate < date '1995-03-15' and l_shipdate > date '1995-03-15' group by l_orderkey, o_orderdate, o_shippriority order by revenue desc, o_orderdate </pre>	
Rows returned	339,695

Benchmark timings:

DBT-3™ QUERY 3	
Run Type	Time (hh:mm:ss)
Ingres hdd	0:15:50
Ingres ssd	0:04:29
VW hdd	0:00:44
VW ssd	0:00:08

DBT-3™ QUERY 4	
Business Question	DBT-3™/TPC-H Order Priority Checking Query (Q4) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre> select o_orderpriority, count(*) as order_count from orders where o_orderdate >= date '1993-07-01' and o_orderdate < date('1993-07-01') + '3 month' and exists (select * from lineitem where l_orderkey = o_orderkey and l_commitdate < l_receiptdate) group by o_orderpriority order by o_orderpriority; </pre>	
Rows returned	5

Benchmark timings:

DBT-3™ QUERY 4	
Run Type	Time (hh:mm:ss)
Ingres hdd	0:07:53
Ingres ssd	0:03:18
VW hdd	0:00:22
VW ssd	0:00:06

DBT-3™ QUERY 5	
Business Question	DBT-3™/TPC-H Local Supplier Volume Query (Q5) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre> select n_name, sum(l_extendedprice * (1 - l_discount)) as revenue from customer, orders, lineitem, supplier, nation, region where c_custkey = o_custkey and l_orderkey = o_orderkey and l_suppkey = s_suppkey and c_nationkey = s_nationkey and s_nationkey = n_nationkey and n_regionkey = r_regionkey and r_name = 'ASIA' and o_orderdate >= date '1994-01-01' and o_orderdate < date ('1994-01-01') + '1 year' group by n_name order by revenue desc; </pre>	
Rows returned	5

Benchmark timings:

DBT-3™ QUERY 5	
Run Type	Time (hh:mm:ss)
Ingres hdd	0:24:25
Ingres ssd	0:09:17
VW hdd	0:00:45
VW ssd	0:00:08

DBT-3™ QUERY 6	
Business Question	DBT-3™/TPC-H Forecasting Revenue Change Query (Q6) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre>select sum(l_extendedprice * l_discount) as revenue from lineitem where l_shipdate >= date '1993-07-01' and l_shipdate < date('1993-07-01') + '1 year' and l_discount between .06 - 0.01 and .06 + 0.01 and l_quantity < 24;</pre>	
Rows returned	1

Benchmark timings:

DBT-3™ QUERY 6	
Run Type	Time (hh:mm:ss)
Ingres hdd	0:06:00
Ingres ssd	0:02:37
VW hdd	0:00:12
VW ssd	0:00:01

DBT-3™ QUERY 7	
Business Question	DBT-3™/TPC-H Volume Shipping Query (Q7) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2.
SQL:	
<pre> select supp_nation, cust_nation, l_year, sum(volume) as revenue from (select n1.n_name as supp_nation, n2.n_name as cust_nation, extract(year from l_shipdate) as l_year, l_extendedprice * (1 - l_discount) as volume from supplier, lineitem, orders, customer, nation n1, nation n2 where s_suppkey = l_suppkey and o_orderkey = l_orderkey and c_custkey = o_custkey and s_nationkey = n1.n_nationkey and c_nationkey = n2.n_nationkey and ((n1.n_name = 'FRANCE' and n2.n_name = 'GERMANY') or (n1.n_name = 'GERMANY' and n2.n_name = 'FRANCE')) and l_shipdate between date '1995-01-01' and date '1996-12-31') as shipping group by supp_nation, cust_nation, l_year order by supp_nation, cust_nation, l_year; </pre>	
Rows returned	4

Benchmark timings:

DBT-3™ QUERY 7	
Run Type	Time (hh:mm:ss)
Ingres hdd	0:14:24
Ingres ssd	0:08:55
VW hdd	0:00:07
VW ssd	0:00:06

DBT-3™ QUERY 8	
Business Question	DBT-3™/TPC-H National Market Share Query (Q8) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre> select o_year, sum(case when nation = 'BRAZIL' then volume else 0 end) / sum(volume) as mkt_share from (select extract(year from o_orderdate) as o_year, l_extendedprice * (1 - l_discount) as volume, n2.n_name as nation from part, supplier, lineitem, orders, customer, nation n1, nation n2, region where p_partkey = l_partkey and s_suppkey = l_suppkey and l_orderkey = o_orderkey and o_custkey = c_custkey and c_nationkey = n1.n_nationkey and n1.n_regionkey = r_regionkey and r_name = 'AMERICA' and s_nationkey = n2.n_nationkey and o_orderdate between date('1995-01-01') and date ('1996-12-31') and p_type = 'ECONOMY ANODIZED STEEL') as all_nations group by o_year order by o_year </pre>	
Rows returned	2

Benchmark timings:

DBT-3™ QUERY 8	
Run Type	Time (hh:mm:ss)
Ingres hdd	00:12:11
Ingres ssd	00:08:41
VW hdd	00:00:22
VW ssd	00:00:05

DBT-3™ QUERY 9	
Business Question	DBT-3™/TPC-H Product Type Profit Measure Query (Q9) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre> select nation, o_year, sum(amount) as sum_profit from (select n_name as nation, extract(year from o_orderdate) as o_year, l_extendedprice * (1 - l_discount) - ps_supplycost * l_quantity as amount from part, supplier, lineitem, partsupp, orders, nation where s_suppkey = l_suppkey and ps_suppkey = l_suppkey and ps_partkey = l_partkey and p_partkey = l_partkey and o_orderkey = l_orderkey and s_nationkey = n_nationkey and p_name like '%green%') as profit group by nation, o_year order by nation,o_year desc </pre>	
Rows returned	175

Benchmark timings:

DBT-3™ QUERY 9	
Run Type	Time (hh:mm:ss)
Ingres hdd	00:26:39
Ingres ssd	00:09:19
VW hdd	00:01:16
VW ssd	00:00:22

DBT-3™ QUERY 10	
Business Question	DBT-3™/TPC-H Returned Item Reporting Query (Q10) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre> select c_custkey, c_name, sum(l_extendedprice * (1 - l_discount)) as revenue, c_acctbal, n_name, c_address, c_phone, c_comment from customer, orders, lineitem, nation where c_custkey = o_custkey and l_orderkey = o_orderkey and o_orderdate >= date('1993-10-01') and o_orderdate < date('1993-10-01') + '3 month' and l_returnflag = 'R' and c_nationkey = n_nationkey group by c_custkey, c_name, c_acctbal, c_phone, n_name, c_address, c_comment order by revenue desc </pre>	
Rows returned	1,148,325

Benchmark timing

DBT-3™ QUERY 10	
Run Type	Time (hh:mm:ss)
Ingres hdd	00:20:25
Ingres ssd	00:06:05
VW hdd	00:01:14
VW ssd	00:00:14

DBT-3™ QUERY 11	
Business Question	DBT-3™/TPC-H Important Stock Identification Query (Q11) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre> select ps_partkey, sum(ps_supplycost * ps_availqty) as value from partsupp, supplier, nation where ps_suppkey = s_suppkey and s_nationkey = n_nationkey and n_name = 'GERMANY' group by ps_partkey having sum(ps_supplycost * ps_availqty) > (select sum(ps_supplycost * ps_availqty) * 0.0001 from partsupp, supplier, nation where ps_suppkey = s_suppkey and s_nationkey = n_nationkey and n_name = 'GERMANY') order by value desc </pre>	
Rows returned	0

Benchmark timings:

DBT-3™ QUERY 11	
Run Type	Time (hh:mm:ss)
Ingres hdd	00:02:34
Ingres ssd	00:00:49
VW hdd	00:00:07
VW ssd	00:00:02

DBT-3™ QUERY 12	
Business Question	DBT-3™/TPC-H Shipping Modes and Order Priority Query (Q12) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre>select l_shipmode, sum(case when o_orderpriority = '1-URGENT' or o_orderpriority = '2-HIGH' then 1 else 0 end) as high_line_count, sum(case when o_orderpriority <> '1-URGENT' and o_orderpriority <> '2-HIGH' then 1 else 0 end) as low_line_count from orders, lineitem where o_orderkey = l_orderkey and l_shipmode in ('MAIL', 'SHIP') and l_commitdate < l_receiptdate and l_shipdate < l_commitdate and l_receiptdate >= date('1994-01-01') and l_receiptdate < date('1994-01-01') + '1 year' group by l_shipmode order by l_shipmode</pre>	
Rows returned	2

Benchmark timings:

DBT-3™ QUERY 12	
Run Type	Time (hh:mm:ss)
Ingres hdd	00:10:11
Ingres ssd	00:03:52
VW hdd	00:00:27
VW ssd	00:00:08

DBT-3™ QUERY 13	
Business Question	DBT-3™/TPC-H Customer Distribution Query (Q13) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre> select c_count, count(*) as custdist from (select c_custkey, count(o_orderkey) from customer left outer join orders on c_custkey = o_custkey and o_comment not like '%special%requests%' group by c_custkey) as c_orders (c_custkey, c_count) group by c_count order by custdist desc, c_count desc; </pre>	
Rows returned	46

Benchmark timings:

DBT-3™ QUERY 13	
Run Type	Time (hh:mm:ss)
Ingres hdd	00:12:28
Ingres ssd	00:14:19
VW hdd	00:00:55
VW ssd	00:00:12

DBT-3™ QUERY 14	
Business Question	DBT-3™/TPC-H Promotion Effect Query (Q14) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre>select 100.00 * sum(case when p_type like 'PROMO%' then l_extendedprice * (1 - l_discount) else 0 end) / sum(l_extendedprice * (1 - l_discount)) as promo_revenue from lineitem, part where l_partkey = p_partkey and l_shipdate >= date('1995-09-01') and l_shipdate < date('1995-09-01') + '1 month';</pre>	
Rows returned	1

Benchmark timings:

DBT-3™ QUERY 14	
Run Type	Time (hh:mm:ss)
Ingres hdd	00:07:26
Ingres ssd	00:06:51
VW hdd	00:00:45
VW ssd	00:00:04

DBT-3™ QUERY 15	
Business Question	DBT-3™/TPC-H Top Supplier Query (Q15) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre> create view v_revenue (supplier_no, total_revenue) as select l_suppkey, sum(l_extendedprice * (1 - l_discount)) from lineitem where l_shipdate >= date('1996-01-01') and l_shipdate < date('1996-01-01') + '90 days' group by l_suppkey; select s_suppkey, s_name, s_address, s_phone, total_revenue from supplier, v_revenue where s_suppkey = supplier_no and total_revenue = (select max(total_revenue) from v_revenue) order by s_suppkey; drop view v_revenue; </pre>	
Rows returned	1

Benchmark timings:

DBT-3™ QUERY 15	
Run Type	Time (hh:mm:ss)
Ingres hdd	00:12:03
Ingres ssd	00:05:49
VW hdd	00:00:42
VW ssd	00:00:04

DBT-3™ QUERY 16	
Business Question	DBT-3™/TPC-H Parts/Supplier Relationship Query (Q16) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre> select p_brand, p_type, p_size, count(distinct ps_suppkey) as supplier_cnt from partsupp, part where p_partkey = ps_partkey and p_brand <> 'Brand#45' and p_type not like 'MEDIUM POLISHED%' and p_size in (49,14,23,45,19,3,36,9) 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 </pre>	
Rows returned	27,840

Benchmark timings:

DBT-3™ QUERY 16	
Run Type	Time (hh:mm:ss)
Ingres hdd	00:00:42
Ingres ssd	00:00:32
VW hdd	00:00:12
VW ssd	00:00:09

DBT-3™ QUERY 17	
Business Question	DBT-3™/TPC-H Small-Quantity-Order Revenue Query (Q17) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre> select sum(l_extendedprice) / 7.0 as avg_yearly from lineitem, part where p_partkey = l_partkey and p_brand = 'Brand#23' and p_container = 'MED BOX' and l_quantity < (select 0.2 * avg(l_quantity) from lineitem where l_partkey = p_partkey); </pre>	
Rows returned	1

Benchmark timings:

DBT-3™ QUERY 17	
Run Type	Time (hh:mm:ss)
Ingres hdd	00:37:08
Ingres ssd	00:10:35
VW hdd	00:00:40
VW ssd	00:00:04

DBT-3™ QUERY 18	
Business Question	DBT-3™/TPC-H Large Volume Customer Query (Q18) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre> select c_name, c_custkey, o_orderkey, o_orderdate, o_totalprice, sum(l_quantity) from customer, orders, lineitem where o_orderkey in (select l_orderkey from lineitem group by l_orderkey having sum(l_quantity) > 300) and c_custkey = o_custkey and o_orderkey = l_orderkey group by c_name, c_custkey, o_orderkey, o_orderdate, o_totalprice order by o_totalprice desc, o_orderdate </pre>	
Rows returned	1,917

Benchmark timings:

DBT-3™ QUERY 18	
Run Type	Time (hh:mm:ss)
Ingres hdd	00:17:40
Ingres ssd	00:11:58
VW hdd	00:00:40
VW ssd	00:00:16

DBT-3™ QUERY 19	
Business Question	DBT-3™/TPC-H Discounted Revenue Query (Q19) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre> select sum(l_extendedprice* (1 - l_discount)) as revenue from lineitem, part where (p_partkey = l_partkey and p_brand = 'Brand#12' and p_container in ('SM CASE', 'SM BOX', 'SM PACK', 'SM PKG') and l_quantity >= 1 and l_quantity <= 1+10 and p_size between 1 and 5 and l_shipmode in ('AIR', 'AIR REG') and l_shipinstruct = 'DELIVER IN PERSON') or (p_partkey = l_partkey and p_brand = 'Brand#23' and p_container in ('MED BAG', 'MED BOX', 'MED PKG', 'MED PACK') and l_quantity >= 10 and l_quantity <= 10+10 and p_size between 1 and 10 and l_shipmode in ('AIR', 'AIR REG') and l_shipinstruct = 'DELIVER IN PERSON') or (p_partkey = l_partkey and p_brand = 'Brand#34' and p_container in ('LG CASE', 'LG BOX', 'LG PACK', 'LG PKG') and l_quantity >= 20 and l_quantity <= 20+10 and p_size between 1 and 15 and l_shipmode in ('AIR', 'AIR REG') and l_shipinstruct = 'DELIVER IN PERSON'); </pre>	
Rows returned	1

Benchmark timings:

DBT-3™ QUERY 19	
Run Type	Time (hh:mm:ss)
Ingres hdd	00:06:15
Ingres ssd	00:01:48
VW hdd	00:00:47
VW ssd	00:00:05

DBT-3™ QUERY 20	
Business Question	DBT-3™/TPC-H Potential Part Promotion Query (Q20) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre> select s_name, s_address from supplier, nation where s_suppkey in (select distinct (ps_suppkey) from partsupp, part where ps_partkey=p_partkey and p_name like 'forest%' and ps_availqty > (select 0.5 * sum(l_quantity) from lineitem where l_partkey = ps_partkey and l_suppkey = ps_suppkey and l_shipdate >= date('1994-01-01') and l_shipdate < date('1994-01-01') + '1 year')) and s_nationkey = n_nationkey and n_name = 'CANADA' order by s_name </pre>	
Rows returned	5,689

Benchmark timings:

DBT-3™ QUERY 20	
Run Type	Time (hh:mm:ss)
Ingres hdd	00:08:05
Ingres ssd	00:04:23
VW hdd	00:00:51
VW ssd	00:00:06

DBT-3™ QUERY 21	
Business Question	DBT-3™/TPC-H Suppliers Who Kept Orders Waiting Query (Q21) TPC-H Specification, http://www.tpc.org/tpch/spec/tpch2.11.0.pdf , Section 2
SQL:	
<pre> select s_name, count(*) as numwait from supplier, lineitem l1, orders, nation where s_suppkey = l1.l_suppkey and o_orderkey = l1.l_orderkey and o_orderstatus = 'F' and l1.l_receiptdate > l1.l_commitdate and exists (select * from lineitem l2 where l2.l_orderkey = l1.l_orderkey and l2.l_suppkey <> l1.l_suppkey) and not exists (select * from lineitem l3 where l3.l_orderkey = l1.l_orderkey and l3.l_suppkey <> l1.l_suppkey and l3.l_receiptdate > l3.l_commitdate) and s_nationkey = n_nationkey and n_name = 'SAUDI ARABIA' group by s_name order by numwait desc, s_name </pre>	
Rows returned	11,957

Benchmark timings:

DBT-3™ QUERY 21	
Run Type	Time (hh:mm:ss)
Ingres hdd	00:36:52
Ingres ssd	00:13:56
VW hdd	00:00:44
VW ssd	00:00:19

Filename: V16-Benchmark-20100630.doc
Directory: C:\Documents and Settings\ingres\Desktop\VectorWise
Template: C:\Documents and Settings\ingres\Application
Data\Microsoft\Templates\Normal.dot
Title: Ratcom V16 Vector Appliance Benchmark
Subject: VectorWise versus Ingres on V16 Appliance
Author: WR, RH
Keywords: VectorWise, Ingres, HDD, SSD, RAID, performance
Comments: DBT-3 Individual Query Benchmark: Standalone
execution of 21 standard TPC-H queries
Creation Date: 7/4/2010 11:44:00 AM
Change Number: 5
Last Saved On: 7/4/2010 2:46:00 PM
Last Saved By: Roy Hann
Total Editing Time: 15 Minutes
Last Printed On: 7/4/2010 2:47:00 PM
As of Last Complete Printing
Number of Pages: 34
Number of Words: 3,728 (approx.)
Number of Characters: 22,369 (approx.)