HP Business Analytics

For the Linux ® **Operating System**

Software Version: 10.0

Performance Benchmark Document

Document Release Date: May 2015

Software Release Date: May 2014

Table of Contents

1. Introduction

1.1 Purpose and Scope

This document outlines the performance benchmark for HP IT Business Analytics (BA) v10.0, and highlights the performance improvements when comparing with the previous release v9.50.

The scenarios designed in this document follow those critical paths that every BA customer will go through: DCS & ETL, KPI Calculation, and UI operations.

Note that all the tests are conducted at HP Software R&D labs.

1.2 Environment Setup

The following tables list the properties of the machines in use:

Configuration 1: Single Node Vertica Setup[1]						
Server	**VM**	**CPU**	**Memory**	**Disk**	**Network**	**OS**
BA	1	8 cores, 2.13 Ghz	16 GB	100 GB	1 GB	Red Hat Linux 6.5
Vertica	1	8 cores, 2.13 Ghz	16 GB	100 GB	1 GB	Red Hat Linux 6.5

Configuration 2: 3-Nodes Vertica Setup (ksafe1)[2]						
Server	**VM**	**CPU**	**Memory**	**Disk**	**Network**	**OS**
BA	1	8 cores, 2.13 Ghz	16 GB	100 GB	1 GB	Red Hat Linux 6.5
Vertica	3	8 cores, 2.13 Ghz	16 GB	100 GB	1 GB	Red Hat Linux 6.5

For the cluster setup, Native Connection Load Balancing is enabled. It is a feature built into Vertica server and client libraries. A host in Vertica cluster can redirect a client's attempt to another currently-active host in the mix, and this redirection is based on ROUNDROBIN policy. According to Vertica Document, in most situation it is the right choice, as it is easy to set up, less at risk of host failures, less memory and CPU intensive, and supported by HP.

[1] Configuration 1 is referred as [single setup] in this document.
[2] Configuration 2 is referred as [cluster setup] in this document

2. Scenarios

2.1 DCS & ETL

In this scenario, we perform end-to-end DCS & ETL operations for PPM, ALM, CSA, and SM. All the sources are restored from customer dumps. It's acknowledged that data distribution often has a certain impact on the performance, so by using customer dumps we ensure that the test results reflect the real world situation.

In v10.0, we have replaced the heavy weight BODS with a light weight SQL-based workflow. Generally speaking, for DCS, the load is on the BA server, i.e. multiple threads are leveraged to extract the source data from the database or the web server, while, for ETL, the load is on the Vertica nodes(s) as multiple entities are processed in parallel.

PPM: *In comparison with v9.50, the size of the flat-file is smaller, the requirement of the database storage is lower, and total time cost is reduced significantly for both DCS and ETL.* With these two PPM sources: CBA and UHG; one is smaller, the other is larger; it is proved that the resources consumption of the operating system as well as the performance are linearly grown with the size of the content pack increases.

PPM: CBA	v10.0		v9.50	
Initial load, ~1.21 m rows in target tables	**Flat-File Size**			
	0.57 GB		1.68 GB	
	Database Storage Requirement Per Node			
	2+ GB		~15 GB	
	DCS Time Cost	**ETL Time Cost**	**DCS Time Cost**	**ETL Time Cost**
	Single Setup			
	1+ mins	6+ mins	4 mins	102 mins
	Cluster Setup			
	1+ mins	11+ mins	n/a	n/a

PPM: UHG	v10.0		v9.50	
Initial load, ~3.91 m rows in target tables	**Flat-File Size**			
	2.89 GB		7.79 GB	
	Database Storage Requirement Per Node			
	6+ GB		~75 GB	
	DCS Time Cost	**ETL Time Cost**	**DCS Time Cost**	**ETL Time Cost**
	Single Setup			
	5+ mins	21+ mins	13 mins	242 mins
	Cluster Setup			
	5+ mins	41+ mins	n/a	n/a

ETL in a cluster setup is also much slower than in a single setup. Clusters take care of the high availability of the underlying databases, which essentially means that in a 3-nodes cluster even if one node goes down, the load can be forwarded to the other nodes that are still up and running. This is critical in production environments to avoid system downtime. However, this advantage comes at a cost, i.e. each load of the data to the database has to be duplicated **ksafe** time(s) (in our cluster setup, it is set to 1) to populate its main and buddy projections. If our cluster setup is configured with no backup, ETL time cost is close to that of a single setup, but be careful as this type of usages is NOT recommended according to Vertica Support.

ALM: *It shows similar performance improvement for both DCS and ETL*. However, source extraction is still the bottleneck of the scenario, and it's most likely caused by the network transmission speed. We sampled 20 packets, 2 m in size each, against our remote ALM web server, the average round trip takes ~241.103 ms; while for a local web server or database, it is ~4 ms. Therefore if customer uses a local ALM web server, DCS should be much faster.

ALM, MSA	v10.0		v9.50	
Initial load, ~0.36 m rows in target tables	**Flat-File Size**			
	0.41 GB		n/a	
	Database Storage Requirement Per Node			
	2+ GB		n/a	
	DCS Time Cost	**ETL Time Cost**	**DCS Time Cost**	**ETL Time Cost**
	Single Setup			
	40+ mins	4+ mins	81 mins	29 mins
	Cluster Setup			
	41+ mins	8+ mins	n/a	n/a

CSA: *It shows similar performance improvement for ETL*. However, based on DCS time cost, source extraction is shown as the bottleneck. Unlike ALM, the round trip of 20 2 m packets between BA server and local CSA web server takes ~4 ms. Further analysis of the CSA extractor indicates that the execution of the CSA REST request is extremely time consuming. Therefore unless CSA enhances the API, this bottleneck can hardly be avoided.

CSA, UHC	v10.0		v9.50	
Initial load, ~3.73 m rows in target tables	**Flat-File Size**			
	0.15 GB		n/a	
	Database Storage Requirement Per Node			
	1+ GB		n/a	
	DCS Time Cost	**ETL Time Cost**	**DCS Time Cost**	**ETL Time Cost**
	Single Setup			
	283+ mins	8+ mins	330 mins	30 mins
	Cluster Setup			
	286+ mins	15+ mins	n/a	n/a

SM:*, It shows similar disk usages reduction and performance improvement for both initial load and 1% delta load.* Surprisingly, unlike other content packs, its cluster setup is almost as fast as its single setup.

SM, ES	v10.0		v9.50	
Initial load, ~13.95 m rows in target tables	**Flat-File Size**			
	1.00 GB		3.49 GB	
	Database Storage Requirement Per Node			
	9+ GB		~75 GB	
	DCS Time Cost	**ETL Time Cost**	**DCS Time Cost**	**ETL Time Cost**
	Single Setup			
	9+ mins	26+ mins	20 mins	453 mins
	Cluster Setup			
	10+ mins	29+ mins	n/a	n/a

SM, ES	v10.0		v9.50	
1% Delta load, ~14.05 m rows in target tables	**Flat-File Size**			
	14.1 MB		n/a	
	Database Storage Requirement Per Node			
	< 1GB		n/a	
	DCS Time Cost	**ETL Time Cost**	**DCS Time Cost**	**ETL Time Cost**
	Single Setup			
	10+ mins	3+ mins	19 mins	56 mins
	Cluster Setup			
	10+ mins	3+ mins	n/a	n/a

2.2 KPI Calculation

In this scenario, we perform a series of KPI Engine calculations using a simple formula with various breakdowns, calculation types, and calculation periods. As a result, we observe that:

1> *Comparing with v9.50, we have achieved certain level of performance improvement*.
2> The total time cost is linearly grown with the increase of calculation units. Generally speaking more breakdowns and calculation periods lead to more calculation units.
3> With a large number of distinct values in breakdown(s) there is no consecutive load on the Vertica database, the stress is all on the Postgres database. With a smaller number of distinct values, the load goes back to the Vertica database, as there are less historical results to be processed.

KPI Definition	Breakdown Details
Source: PPM, CBA customer dump Context: ProjectPortfolioManagement Formula: COUNT(ProjectTask , *)	<u>Breakdown Case #1</u> ➔ ProjectTask: OverdueTaskIndicator ➔ Project: Class ➔ Person: Name (x125)
	<u>Breakdown Case #2</u> ➔ ProjectTask: OverdueTaskIndicator ➔ Project: Class ➔ Person: Name (x25)
	<u>Breakdown Case #3</u> ➔ ProjectTask: OverdueTaskIndicator ➔ Project: Class ➔ Person: Name (x5)
	<u>Breakdown Case #4</u> ➔ ProjectTask: OverdueTaskIndicator ➔ Project: Class
	<u>Breakdown Case #5</u> ➔ ProjectTask: OverdueTaskIndicator

Breakdown Case #	Calculation Type	Calculation Period	Calculation Units		Total Time Cost (s)	
			v10.0 Single Setup	v9.50	v10.0 Single Setup	v9.50
1	Recalculate, 1year	Yearly	1518	1578	35	49
1	Recalculate, 1 year	Quarterly	3795	3945	92	133
1	Recalculate, 1 year	Monthly	9867	10257	249	365
1	Recalculate, 1 year	Weekly	40227	43558	1064	1506

1	Recalculate, 1 year	Daily	277794	287985	8034	9840
1	Recalculate, 2 years	Daily	554829	575240	16173	19665
5	Recalculate, 2 years	Daily	2924	2920	647	1011
4	Recalculate, 2 years	Daily	10965	10950	928	1547
3	Recalculate, 2 years	Daily	39474	39420	1612	3050
2	Recalculate, 2 years	Daily	127925	133590	3038	5487
1	Calculate Now	Daily	759	790	17	22

For a cluster setup, it shows very similar numbers.

Breakdown #	Calculation Type	Calculation Period	Calculation Units		Total Time Cost (s)	
			v10.0 Cluster Setup	**v9.50**	**v10.0 Cluster Setup**	**v9.50**
1	Recalculate, 1year	Yearly	1518	1578	41	49
1	Recalculate, 1 year	Quarterly	3795	3945	97	133
1	Recalculate, 1 year	Monthly	9867	10257	261	365
1	Recalculate, 1 year	Weekly	40227	43558	1068	1506
1	Recalculate, 1 year	Daily	277794	287985	7770	9840
1	Recalculate, 2 years	Daily	554829	575240	15348	19665
5	Recalculate, 2 years	Daily	2924	2920	716	1011
4	Recalculate, 2 years	Daily	10965	10950	1057	1547
3	Recalculate, 2 years	Daily	39474	39420	1847	3050
2	Recalculate, 2 years	Daily	127925	133590	3133	5487
1	Calculate Now	Daily	759	790	16	22

2.3 UI Operations

In this scenario, we simulate a large number of concurrent users viewing pages and exploring the web portal by HP LoadRunner [v11.52]. Note that we do not use the data from any of those customer dumps, as in v9.50, this test is designed using a demo CAP, so in order to draw fair comparison with the previous release, we stick to the CAP data.

Viewing a BOE report is excluded in this test, as its behavior depends on the performance of a 3rd party library, its weight is assigned to other UI operations accordingly.

The following table presents the workflow transactions, their weights in percentage, and corresponding simulated steps:

Transaction	Weight, %	Simulated Steps
TX_BA_open_VPOps_Main_double	10%	Open a page with four components: 1) Scorecard 2) KPI View (8 KPIs) 3) Historical View (3 KPIs) 4) KPI Rolodex (15KPIs)
TX_BA_open_VPOps_8_components	10%	Open a page with eight components: 1) Scorecard 2) KPI View (4 KPIs) 3) Historical View (3 KPIs) 4) KPI Rolodex (8KPIs) 5) KPI List (7 KPIs) 6) Historical Metric View (1 Metric) 7) Pie Chart View (2 KPIs) 8) KPI View (7 KPIs)
TX_BA_open_VPOps_Main	10%	Open a page with four components: 1) Scorecard 2) KPI View (5 KPIs) 3) Historical View (1 KPIs) 4) KPI Rolodex (6KPIs)
TX_BA_refresh_VPOps_Main	7%	Click the Refresh button on the page TX_BA_open_VPOps_Main
TX_BA_view_Reduce_Risk	10%	Simulate the user action that displays the information while setting the cursor on the shortcut of a KPI or Objective
TX_BA_explore_Reduce_Cost	10%	Click the shortcut link of a KPI in a page to open Explorer
TX_BA_view_Met_SLAs	10%	Simulate the user action that displays the information while setting the cursor on the shortcut of a KPI or Objective
TX_BA_explore_Affected	10%	Click the shortcut link of a KPI in a page to open Explorer

TX_BA_change_to_current	7%	Change the Period to current in Explorer
TX_BA_dataset	1%	Click the Data Set button in Explorer
TX_BA_goalmap	5%	Click the Goal Map button in Explorer
TX_BA_forecast	5%	Click the Forecast button in Explorer
TX_BA_annotation	5%	Add and Delete annotation while viewing the KPI details in the Dashboard Page

When comparing with v9.50, we can support more concurrent users with larger TTPS. Note that the test results exclude ramp-up and ramp-down periods to reflect only the peak load.

	Total Transactions Per Second, Pass	**Total Transactions Per Second, Fail**	**Concurrent Users[3]**	**Think Time, Second**
v10.0, Single Setup	5.15	0.001	300	60 (90% - 120%)
v10.0, Cluster Setup	5.10	0.003	300	60 (90% - 120%)
v9.50	1.00	< 0.01	100	60 (90% - 120%)

The following tables list the details for both single setup and cluster setup. Although both can support up to 300 concurrent users and 5+ TTPS, cluster setup is slightly slower than single setup in terms of Avg. and 90% response time, i.e. some cross-nodes queries should be optimized in future releases.

v10.0, Single Setup	**Min.**	**Avg.**	**Max.**	**Std. Dev.**	**90%**	**Pass**	**Fail**	**Stop**
TX_annotation	0.0	0.1	2.9	0.1	0.2	926	4	0
TX_change_to_current	0.1	0.2	3.3	0.1	0.3	1301	0	0
TX_dataset	0.5	0.6	1	0.1	0.8	157	0	0
TX_explore_Affected	0.3	0.4	2.9	0.1	0.4	1886	0	0
TX_explore_Reduce_Cost	0.6	0.9	4.9	0.2	1.0	1828	0	0
TX_forecast	0.2	0.5	1.4	0.2	0.7	935	0	0
TX_goalmap	0.0	0.1	3.1	0.1	0.1	924	0	0
TX_open_VPOps_Main	0.7	1.5	7.7	0.6	2.4	1924	0	0
TX_refresh_VPOps_Main	0.1	0.3	1.9	0.1	0.5	1307	0	0
TX_view_Met_SLAs	0.1	0.2	3.9	0.1	0.2	1804	0	0
TX_view_Reduce_Risk	0.1	0.1	5.4	0.2	0.1	1785	0	0

[3] Concurrent Users represents the number of users currently logged in the system, it may be inactive due to the think time configuration; and all these users are not logged in and out the system simultaneously, they are instead controlled by user ramp-up settings: one every 00:00:05 (HH:MM:SS); and user ramp-down settings: one every 00:00:05 (HH:MM:SS).

TX_open_VPOps_8_components	1.4	3.0	12.4	1.0	4.9	1825	0	0
TX_open_VPOps_Main_double	0.6	1.7	5.1	0.6	2.6	1874	0	0

v10.0, Cluster Setup	**Min.**	**Avg.**	**Max.**	**Std. Dev.**	**90%**	**Pass**	**Fail**	**Stop**
TX_annotation	0.0	0.2	0.3	0.0	0.2	895	10	0
TX_change_to_current	0.1	0.2	0.7	0.1	0.3	1236	0	0
TX_dataset	0.9	1.2	2.0	0.2	1.5	181	0	0
TX_explore_Affected	0.3	0.4	1.3	0.1	0.5	1926	0	0
TX_explore_Reduce_Cost	0.7	1.0	1.6	0.1	1.1	1750	0	0
TX_forecast	0.2	0.7	2.2	0.2	0.9	859	0	0
TX_goalmap	0.0	0.1	0.2	0.0	0.1	927	0	0
TX_open_VPOps_Main	0.8	2.0	7.7	1.0	3.4	1808	0	0
TX_refresh_VPOps_Main	0.1	0.4	1.8	0.2	0.5	1323	0	0
TX_view_Met_SLAs	0.1	0.2	0.4	0.0	0.2	1872	0	0
TX_view_Reduce_Risk	0.1	0.1	0.2	0.0	0.2	1828	0	0
TX_open_VPOps_8_components	1.5	4.0	14.4	1.7	6.7	1919	0	0
TX_open_VPOps_Main_double	0.7	2.1	8.8	1.0	3.4	1877	0	0

3. Conclusion

Performance is one of the high priority projects in this release. We expended considerable efforts optimizing the performance, as well as making a significant move by shifting to the Linux operating system and leveraging the world-famous real time analytics platform, Vertica. This document proves that we have made the right decision: faster with less hardware requirement.

4. Appendix I – v9.50 Environment Setup

Configuration: Distributed Setup						
Server	**Model**	**CPU**	**Memory**	**Disk**	**Network**	**OS**
BA	VM	8 cores, 2.67 Ghz	16 GB	100 GB	1 GB	Windows 2008 R2
BOE	VM	8 cores, 2.67 Ghz	16 GB	100 GB	1 GB	Windows 2008 R2
DWH	VM	8 cores, 2.67 Ghz	16 GB	100 GB	1 GB	Windows 2008 R2
SQLServer	ProLiant DL580	24 Cores 2.93Ghz	64 GB	1 TB	1 GB	Windows 2008 R2

5. Appendix II – Configuration References

In v10.0 Support Matrix, there are two set of configurations, minimum and recommended. In this performance benchmark we adopt the recommended one with 16 GB physical memory. However, there are differences in terms of application performance among these configurations based on the results of additional tests. This section can be used as guideline for capacity planning.

Configuration	CPU (Cores)		Memory (GB)	
	BA	**Vertica**	**BA**	**Vertica**
Minimum	4	4	12	12
Recommended, 1	8	8	16	16
Recommended, 2	8	8	16	24

DCS and ETL: On a BA server, no extractor can utilize more than 50% of the processing power of the 8-cores system, so as expected, a 4-cores system will not affect its performance. A smaller number of cores for Vertica server means lower concurrency, and more entities queued up for processing, which in turn slows down the ETL process. For example, the initial load of PPM_UHG takes 21+ minutes with an 8-cores configuration; while it increases to 27+ minutes with a 4-cores configuration. Similarly for the initial load of SM_ES: with an 8-cores configuration it spends 26+ minutes; while with a 4-cores configuration it goes up to 31+ minutes. Also note that for small content packs or content pack with very skewed data distribution, i.e. a few large entities, the rest of the entities much smaller, the impact is trivial.

We support running multiple content packs concurrently. In general, the 12 GB physical memory for the Vertica server is sufficient for the initial load of 1 to 2 content packs; the 16 GB physical memory for Vertica server is sufficient for the initial load of 2 to 3 content packs. However, if additional content packs are required, and there are entities with millions of rows, we suggest to upgrade the physical memory to at least 24 GB.

KPI Calculation: It is expected that there are no significant differences among these configurations.

UI Operations: A smaller number of cores for aBA server results in a smaller number of concurrent users, 150; lower TTPS, 2.6. What will happen if there are more concurrent users logging on and browsing the portal, both the avg. and 90% response time will increase dramatically, which is often the indicator of bad end user experience.