|  |
| --- |
| International Production Center |
| Workday Enhancement Integration  Final Report | |

Author: Arindam Das

Version: 1.0

Date: 21/02/2019

Ref: 201901\_WorkdayEnhancement\_Performance\_Test\_Plan\_v0.2

[1. EXECUTIVE SUMMARY 4](#_Toc536712770)

[1.1. Functional Summary 4](#_Toc536712771)

[1.2. Technical Recommendations 5](#_Toc536712772)

[1.3. Project History 7](#_Toc536712773)

[2. CONTACT INFORMATION 9](#_Toc536712774)

[2.1. BU 9](#_Toc536712775)

[2.2. IPC Engineering Benchmark 9](#_Toc536712776)

[3. Load Test Architecture 10](#_Toc536712777)

[3.1. Test Platform 10](#_Toc536712778)

[3.2. Platform – Material and Software 11](#_Toc536712779)

[3.3. Testing tools 11](#_Toc536712780)

[4. Load Test Characteristics 12](#_Toc536712781)

[4.1. Functional scenarios 12](#_Toc536712782)

[4.2. The load test models 12](#_Toc536712783)

[5. BENCH RESULTS 13](#_Toc536712784)

[5.1. Load tests Summary 13](#_Toc536712785)

[6. Conclusion 15](#_Toc536712786)

Preamble

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Author** | **Recipients** |
| 1.0 | 21/02/2019 | Arindam Das | Abhishek Ranjan  Sridhar Polisetty  Jyothi Pujari  Jordi Redondo  Ravi Kumar Rongali  Ramu Kommanapalli  SatyasuryaSivakumar Suthapalli  Sai Chanda  Sridhar Polisetty  Satyavathi Salikanti  Srinivasarao Vivaramneni  Deepa Maddula  Amit Kulkarni  Pooja Mehta  James Conway  Vinay Gudivada |

Track changes

|  |  |
| --- | --- |
| **Version number** | **Change description** |
| 1.0 | Document creation |

Associated documents

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Author** | **Version** | **Date** |
| 201901\_WorkdayEnhancement\_Performance\_Test\_Plan\_v0.2 | Arindam Das | 0.2 | 25/01/2019 |

# EXECUTIVE SUMMARY

## Functional Summary

The application under test was Workday Enhanced Integration API v2.0. This application captures local payroll data of the Clients via Workday system using REST API Services.

Primary objective of the Benchmark was to performance test China and Singapore APIs to determine Application API’s stability and capacity in terms of transaction rate supported. Also to provide an overall user experience under load and identify bottlenecks if any.

7 unique APIs from CN and SG were identified for this purpose and combined tests were executed.

Both tests were performed in plateau model with max 200 users (Plateau model – 100, 150, 180 and 200 users) to identify the exact breaking point and capacity threshold.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rounds | Objective | API Locale | Scenarios | Concurrent Users |
| Round 1 | Capacity Objective | Singapore | SG APIs [SC1 – SC7] | 200 users plateau model  (100 – 150 – 180 – 200 users) |
| Round 2 | China | CN APIs [SC1 – SC7] |

From these test results we could summarize the below notes-

* From Singapore API test on TQ3 environment, we can infer that 🡪
  + System is able to support a scenario rate of **~7888 Transaction/hr. with ~150 user** concurrency, and is considered to be the breaking point, after which response time degrades and intermittent errors (HTTP 500) start appearing with increased SAP server CPU utilization
* From China API test on TQ3 environment, we can infer that 🡪
  + System is able to support **~5159 transactions/hr. with ~100 user concurrency**, beyond which response times begin to degrade and we hit the bottleneck with SAP server resources (Mainly CPU).
* During initial China API test we observed JCO connection pool saturation with pool size – 30. Further test was executed with pool size increased to 100. With increased connection pool capacity, we saw response times improved by avg. **55%▼** and transaction rate increased by avg. **16%▲** at 100 user concurrency.

*Since tests were performed in TQ3 environment, which is scaled down as compared to production in terms of infrastructure capacity (Number of servers in each layer, CPU, Memory etc.), the results should be extrapolated to predict actual Production behavior.*

A detailed results and comparison from all these tests have been documented in [*Bench Results*](#_BENCH_RESULTS_1) section.

## Technical Recommendations

* Performance test results indicate that it is the SAP layer which exhibits high CPU intensive processing and workload. Wherever, Mule layer acts as a router/converter and does not consume high resource utilization.

It is recommended to monitor the SAP server CPU utilization to look out for any high CPU utilization and causing performance bottleneck.

* During China API testing, we hit a bottleneck where JCO connection pool was saturated at Mule layer. Due to which transaction response time increased and intermittent errors were thrown.

From Mule server log (mule-app-mnc-wei.log…) we see below error snippet –

*Root Exception stack trace:*

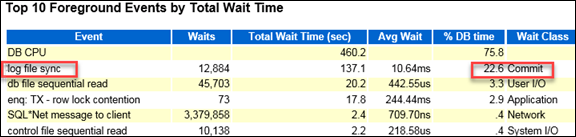
*com.sap.conn.jco.JCoException: (106) JCO\_ERROR\_RESOURCE: Connection pool xssadmin@null:null:002?ft3qAbYRwKWv+StI36ML9w==|CONFIGURED\_USER&002|XSSADMIN|EN|201F00B5CA5D65A1C118E5E32431514C is exhausted. The current pool size peak limit is 30 connections.*

To remediate the issue connection pool capacity was increased to 100. This had positive impact on both response times and infrastructure stability.

It is recommended to monitor mule server logs to look out for any such error messages related to connection pool saturation.

* From ORACLE AWR report, we did not see any major DB bottleneck. Though below areas can be looked at for further improvement scope –

1. Wait class “Commit” consumed around 22.6% of total DB Time for event – “Log File Sync”.

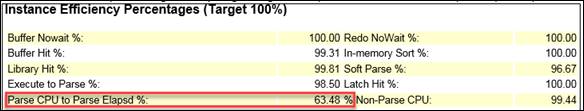


* High “log file sync” is observed in case of slow disk writes (LGWR takes long time to write), or because the application commit rate is very high.As seen below, there were around 29k waits for “log file parallel write” event with an average 1.34 ms delay with a total of 40 sec delay.

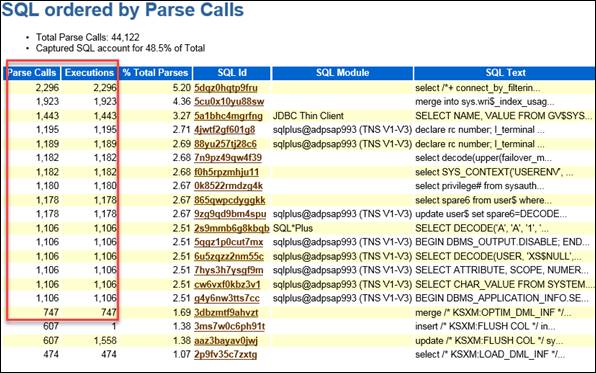
cid:image002.jpg@01D4BEF0.E25CFB80

Please check application logic to understand if this is normal or there’s any scope to reduce number of commits.

1. For “Instance Efficiency Percentages” everything looked good except for Parse CPU to Parse Elapsed % (63.48%). Mostly Soft parsing were observed.



For most SQL queries, Parse to Execution ratio was 1:1.



“Session\_Cached\_Cursors” can be increased to 100 (currently set to default 50) to see if parsing is improved.

1. SQL ID - 9ry7cj3jc3xdm was executed almost 413k times with a total elapsed time of 12,243 sec.

cid:image014.png@01D4C201.3F232250

Please check application logic if this is normal or if there’s any scope of reducing the SQL call.

## Project History

Primary objective of the Benchmark was to performance test WEI APIs (version 2.0) for China and Singapore APIs. 7 APIs were identified for this purpose and 2 rounds of test were executed for China and Singapore APIs respectively.

Both the tests were executed in Plateau model with max 200 users (Plateau model – 100, 150, 180 and 200 users) to identify the exact breaking point and capacity threshold.

Please see below a brief overview of both the tests.

**Round 1 – Singapore APIs**

* At 100 users
* Total scenario Rate achieved - 5358 Transactions/Hr.
* 90th percentile response time for all APIs (except SC1 – Landing Page Service) were < 3 sec. Response time was stable.
* SAP server (adpsap915) CPU utilization was -  57% avg.
* At 150 users
* Total scenario Rate achieved - 7888 Transactions/Hr.
* Response time increased with increasing CPU utilization for adpsap915. SAP server (adpsap915) CPU utilization was increased to -  82% avg.
* 90th percentile response time for all APIs were <6 sec.
* At 180 users
* CPU saturation (>98%) observed for adpsap915.
* 90th percentile response time for all APIs were more than 10 sec.
* Intermittent HTTP 500 errors started appearing and failures increased.
* Scenario Rate dropped due to high CPU and errors  - 7456 Transactions/Hr.
* At 200 users
* CPU completely saturated for adpsap915.
* 90th percentile response time for all APIs were more than 12 sec.
* HTTP 500 error rate increased exponentially

For SG APIs on TQ3 environment, 150 concurrent users at a scenario rate of 7888 Transaction/hr. is considered to be the breaking point, after which response time degrades and intermittent errors (HTTP 500) start appearing with increased SAP server CPU utilization.

**Round 2 – China APIs**

Initial combined test for China APIs revealed JCO Connection pool saturation at Mule layer. Initial (default) value for connection pool was set to 30.

Further test was executed with increased JCO connection pool size to 100. Please see below results.

* At 100 users
* Response times have improved at 100 user load after increasing connection pool size on the mule server (From 30 to 100).
* Scenario rate achieved was 5159 Transactions/hr.
* However, CPU utilization of ADAPSAP915 is on the higher side. It is between 80 to 85%.

* At 150 users
* At 150 users, we start observing intermittent HTTP Status-Code=502 (Proxy Error) for <https://ut2adm.ehc.adp.com/mncws/v1/token?iss=workday-rddev&aud=globalview-rdqua&tenant=002&sub=21001&mncpd=adp>
* Processor utilization on SAP server ADPSAP915 is more than 90% and clearly response times are impacted

* At 180 users
* 90th percentile response time for all APIs were >40sec.
* Intermittent HTTP 500 errors increased.
* SAP server (ADPSAP915) CPU was saturated.

From the China API test results we can infer that with the current configuration system is able to support 5159 transactions/hr beyond which response times begin to degrade and we hit the bottleneck with SAP server resources.

A detailed results and comparison from all tests have been documented in [Bench Results](#_BENCH_RESULTS_1) section.

# CONTACT INFORMATION

## BU

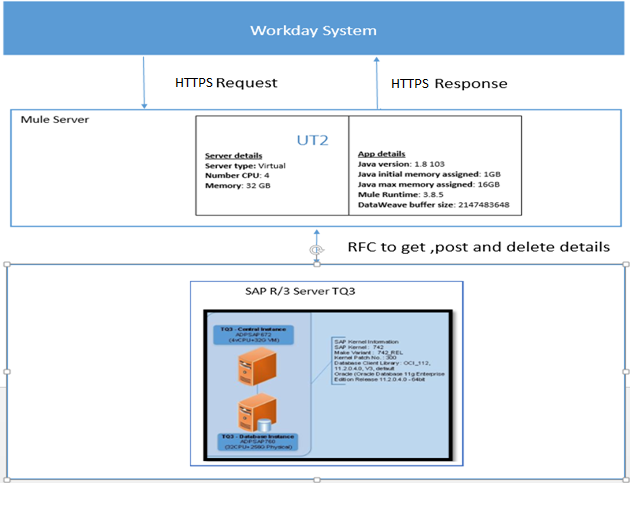
|  |  |  |
| --- | --- | --- |
| Name | Email 🖅 Telephone 🖁 | Role |
| Sridhar Polisetty | Sridhar.Polisetty@ADP.com | Application Development Manager |
| Amit Kulkarni | Amit.kulkarni@adp.com | Quality Assurance Manager |
| Jyothi Pujari | Jyothi.Pujari@adp.com | Product Specialist |
| Sai Chada | Sai.Chada@adp.com | Solution Architect |

## IPC Engineering Benchmark

|  |  |  |
| --- | --- | --- |
| Name | Email 🖅 Telephone 🖁 | Role |
| Arindam Das | [Arindam.Das1@adp.com](mailto:Arindam.Das1@adp.com) | GETS – Project Leader |
| Pooja Dilip-Mehta | [Pooja.Mehta@adp.com](mailto:Pooja.Mehta@adp.com) | GETS – Project Manager |
| James Conway | James.Conway@adp.com | GETS – Performance Testing Manager |

# Load Test Architecture

## Test Platform



Tests were performed in (UT2 – TQ3) environment.

Workday Enhancement API endpoint URL: <https://portal106.globalview.adp.com/mnccc/workday>...

## Platform – Material and Software

### Servers description

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Name | Model | OS | CPU | RAM |
| Reverse proxy - euprx02.gv.esi.adp.com:8080 (adpsap662) | VM | Linux SUSE11 64 bits | 4 vCPU | 32 GB |
| Mule server - ut2nod01.gv.esi.adp.com (adpsap852) | VM | RedHat Server 7.5 | 4 vCPU | 32 GB |
| SAP Server - tq3aci00.gv.esi.adp.com (adpsap915) | VM | Linux SUSE12 64 bits | 4 vCPU | 32 GB |
| Oracle DB - tq3dbi00.gv.esi.adp.com – (adpsap993) | Physical | Linux Suse 12 64 bits | 80 vCPU | 256 GB |

All VM servers were connected via 1 Gbps and Physical servers via 20 Gbps.

### Servers components

|  |  |  |  |
| --- | --- | --- | --- |
| Service | Logical Hostname | Physical Hostname | Middleware / Software |
| Reverse Proxy | euprx02.gv.esi.adp.com | adpsap662 | Apache/httpd |
| Application Layer | ut2nod01.gv.esi.adp.com | adpsap852 | Mule-enterprise-standalone-3.8.5 |
| Business Layer | tq3aci00.gv.esi.adp.com | adpsap915 | SAP Kernel |
| Oracle Database | tq3dbi00.gv.esi.adp.com | adpsap993 | Oracle |

## Testing tools

HP Loadrunner 12.60 was used to create automated scripts for APIs

HP Performance Center 12.60 has been used to simulate user activity and collate system and software metrics.

HP Sitescope 11.51 has been used to collect system and software metrics during user simulation.

# Load Test Characteristics

## Functional scenarios

14 API’s (7-CN, 7-SG) were identified for the benchmark; these were tested for 2 clients – China and Singapore.

|  |  |  |
| --- | --- | --- |
| # of Scenarios | Workday Enhancement Integration | User type |
| SC1 | Landing Page Service | Employee |
| SC2 | Subtype Service | Employee |
| SC3 | Record Overview Service | Employee |
| SC4 | Metadata Service | Employee |
| SC5 | Master data Service | Admin |
| SC5.1 | Create Record | Admin |
| SC5.2 | Edit Record | Admin |
| SC5.3 | Delete Record | Admin |
| SC6 | Dropdown Value Service | Employee |
| SC7 | Value Callback Service | Employee |

## The load test models

This section describes the load test model of the different tests that have been conducted on the application.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| # | Objective | Concurrent VUsers | Scenario used | Ramp up rate | Steady state | Ramp down rate |
| LT8 | Capacity objective | 200 users plateau model  (100 – 150 – 180 - 200 users) | SC1 – SC7 combined  (Equal Distribution) | 1 user started every 5 seconds | 100 users 30 min 🡪  150 users 30 min 🡪  180 users 15 min 🡪  200 users 15 min | 1 user stopped every 5 seconds |

LT8 was executed in 2 rounds – for China API and Singapore API.

# BENCH RESULTS

## Load tests Summary

Tests executed:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Rounds | Objective | API Locale | Scenarios | Concurrent Users |
| Round 1 | Capacity Objective | Singapore | SG APIs [SC1 – SC7] | 200 users plateau model  (100 – 150 – 180 – 200 users) |
| Round 2 | China | CN APIs [SC1 – SC7] |

Please find below results from all the above test rounds.

* **Round 1** (Singapore API results):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameters | At 100 Users | At 150 Users | At 180 Users | At 200 Users |
| 90th percentile Response Time (sec) | SC1 - 4.2  SC2 - 1.6  SC3 - 1.8  SC4 - 1.8    SC5.1 - 1.6    SC5.2 - 1.7    SC5.3 - 1.5  SC6 - 2.1  SC7 - 2.3 | SC1 - 6.2  SC2 - 4.1  SC3 - 4.5  SC4 - 4.2    SC5.1 - 4.1    SC5.2 - 3.7    SC5.3 - 3.5  SC6 - 4.8  SC7 - 4.9 | SC1 - 14.8  SC2 - 10.2  SC3 - 10.1  SC4 - 10.3    SC5.1 - 9.6    SC5.2 - 10.5    SC5.3 - 10.1  SC6 - 10.7  SC7 - 10.9 | SC1 - 17.1  SC2 - 11.8  SC3 - 12.1  SC4 - 11.5    SC5.1 - 12.5    SC5.2 - 11.5    SC5.3 - 10.8  SC6 - 12.5  SC7 - 12.3 |
| Scenario rate achieved (Transactions/Hr.) | SC1 - 792  SC2 - 808  SC3 - 806  SC4 - 808  SC5 - 488  SC6 - 802  SC7 - 854  **Total - 5358** | SC1 - 1142  SC2 - 1204  SC3 - 1190  SC4 - 1192  SC5 - 678  SC6 - 1188  SC7 - 1294  **Total - 7888** | SC1 - 1060  SC2 - 1200  SC3 - 1124  SC4 - 1112  SC5 - 564  SC6 - 1156  SC7 - 1240  **Total - 7456** | SC1 - 1104  SC2 - 1204  SC3 - 1184  SC4 - 1184  SC5 - 544  SC6 - 1144  SC7 - 1260  **Total - 7624** |
| RP server adpsap662 - CPU utilization (%) | Avg – 2.7  Max – 3.9 | Avg – 2.7  Max – 4.7 | Avg – 3.5  Max – 5.9 | Avg – 2.5  Max – 3.1 |
| Mule server adpsap852 - CPU utilization (%) | Avg – 5.3  Max – 6.9 | Avg – 6.0  Max – 8.1 | Avg – 6.5  Max – 7.7 | Avg – 6.1  Max – 7.1 |
| SAP server adpsap915 - CPU utilization (%) | Avg – 57.8  Max – 62.3 | Avg – **82.0**  Max – **87.0** | Avg – **98.1**  Max – **98.5** | Avg – **98.2**  Max – **98.3** |
| DB server adpsap993 - CPU utilization (%) | Avg – 1.1  Max – 1.7 | Avg – 1.2  Max – 2.1 | Avg – 1.2  Max – 1.3 | Avg – 1.0  Max – 1.2 |
| Max Mule Heap Utilization (GB) | 2.2 (Total 16 GB) | 2.3 (Total 16 GB) | 2.5 (Total 16 GB) | 2.5 (Total 16 GB) |
| Errors | NA | NA | Intermittent HTTP 500 errors | HTTP 500 error rate increased exponentially |

* **Round 2** (China API results):

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Parameters | At 100 Users | At 150 Users | At 180 Users | At 200 Users |
| 90th percentile Response Time (sec) | SC1 - 12.7  SC2 - 4.9  SC3 - 5.1  SC4 - 7.2    SC5.1 - 11.8    SC5.2 - 4.6    SC5.3 - 4.4  SC6 - 4.8  SC7 - 5.2 | SC1 - 31.9  SC2 - 20.6  SC3 - 21.0  SC4 - 24.3    SC5.1 - 33.8    SC5.2 - 20.8    SC5.3 - 19.3  SC6 - 20.2  SC7 - 20.5 | SC1 - 41  SC2 - 30.5  SC3 - 31.0  SC4 - 35.3    SC5.1 - 45.5    SC5.2 - 30.1    SC5.3 - 29.5  SC6 - 30.2  SC7 - 29.4 | SC1 - 52.1  SC2 - 41.6  SC3 - 41.7  SC4 - 47.4    SC5.1 - 60.8    SC5.2 - 41.8    SC5.3 - 41.4  SC6 - 41.7  SC7 - 41.0 |
| Scenario rate achieved (Transactions/Hr.) | SC1 - 738  SC2 - 788  SC3 - 792  SC4 - 774  SC5 - 418  SC6 - 799  SC7 - 850  **Total - 5159** | SC1 - 738  SC2 - 839  SC3 - 803  SC4 - 788  SC5 - 245  SC6 - 796  SC7 - 846  **Total - 5055** | SC1 - 763  SC2 - 896  SC3 - 871  SC4 - 803  SC5 - 212  SC6 - 850  SC7 - 940  **Total - 5335** | SC1 - 770  SC2 - 871  SC3 - 857  SC4 - 799  SC5 - 184  SC6 - 835  SC7 - 853  **Total - 5169** |
| RP server adpsap662 - CPU utilization (%) | Avg – 2.8  Max – 5.3 | Avg – 2.7  Max – 4.8 | Avg – 2.3  Max – 2.7 | Avg – 2.6  Max – 4.7 |
| Mule server adpsap852 - CPU utilization (%) | Avg – 5.8  Max – 6.6 | Avg – 7.8  Max – 25.1 | Avg – 5.6  Max – 7.0 | Avg – 5.7  Max – 6.3 |
| SAP server adpsap915 - CPU utilization (%) | Avg – **83.3**  Max – **85.7** | Avg – **90.6**  Max – **96.4** | Avg – **94.9**  Max – **96.2** | Avg – **94.5**  Max – **95.0** |
| DB server adpsap993 - CPU utilization (%) | Avg – 2.8  Max – 3.0 | Avg – 3.0  Max – 3.4 | Avg – 3.0  Max – 3.1 | Avg – 3.0  Max – 3.1 |
| Max Mule Heap Utilization (GB) | 2.2 (Total 16 GB) | 2.2 (Total 16 GB) | 2.3 (Total 16 GB) | 2.3 (Total 16 GB) |
| Errors | NA | HTTP Status-Code=502 (Proxy Error) for [Token](https://ut2adm.ehc.adp.com/mncws/v1/token?iss=workday-rddev&aud=globalview-rdqua&tenant=002&sub=21001&mncpd=adp) URL | Intermittent HTTP 500 errors for API URLs  +  Intermittent HTTP 502 errors for token URL | |

# Conclusion

Form all the tests performed we could conclude below:

* From Singapore API test on TQ3 environment, we can infer that 🡪
  + System is able to support a scenario rate of ~7888 Transaction/hr. and is considered to be the breaking point, after which response time degrades and intermittent errors (HTTP 500) start appearing with increased SAP server CPU utilization
* From China API test on TQ3 environment, we can infer that 🡪
  + System is able to support ~5159 transactions/hr. beyond which response times begin to degrade and we hit the bottleneck with SAP server resources (Mainly CPU).
* Since tests were performed in TQ3 environment, which is scaled down as compared to production in terms of infrastructure capacity (Number of servers in each layer, CPU, Memory etc.), the results should be extrapolated to predict actual Production behavior.