*For instructions on using this template, please see Notes to Author/Template Instructions on page 22. Notes on accessibility: This template has been tested and is best accessible with JAWS 11.0 or higher. For questions about using this template, please contact* [*CMS IT Governance*](mailto:IT_Governance@cms.hhs.gov)*. To request changes to the template, please submit an* [*XLC Process Change Request*](https://www.cms.gov/Research-Statistics-Data-and-Systems/CMS-Information-Technology/XLC/Downloads/XLCProcessChangeRequestCR.docx) *(CR).*

|  | Centers for Medicare & Medicaid Services  CMS eXpedited Life Cycle (XLC) |
| --- | --- |

<Project Name / Acronym>

Performance Test Plan and Results Template

Version X.X

MM/DD/YYYY

**Document Number:** <document’s configuration item control number>

**Contract Number:** <current contract number of company maintaining document>

Table of Contents

[1. Executive Summary 1](#_Toc420668192)

[1.1 Overview: Project Background and Scope 1](#_Toc420668193)

[2. Performance Requirements and Planning 2](#_Toc420668194)

[2.1 Performance Requirements 2](#_Toc420668195)

[2.1.1 Load Model 2](#_Toc420668196)

[2.2 Performance Test Approach 3](#_Toc420668197)

[2.2.1 Assumptions, Constraints, and Risks 4](#_Toc420668198)

[2.2.2 Milestones 5](#_Toc420668199)

[2.2.3 Test Organization 6](#_Toc420668200)

[2.2.4 Performance Test Script Steps 7](#_Toc420668201)

[2.2.5 Performance Test Data Planning 7](#_Toc420668202)

[2.2.6 Performance Test Scenario Runs 8](#_Toc420668203)

[2.2.7 Performance Test Environment 11](#_Toc420668204)

[2.3 Performance Test Monitoring Tools and Metrics 13](#_Toc420668205)

[3. Execution and Analysis 16](#_Toc420668206)

[3.1 Performance Test Results and Analysis 16](#_Toc420668207)

[3.1.1 Test Run 1 – Baseline Two Hour – Peak Hour Load Test 16](#_Toc420668208)

[3.1.2 Test Run 2 17](#_Toc420668209)

[3.1.3 Test Run 3 17](#_Toc420668210)

[3.1.4 Test Run 4 17](#_Toc420668211)

[3.1.5 Test Run 5 18](#_Toc420668212)

[3.1.6 Test Run 6 18](#_Toc420668213)

[Appendix A: Test Sign-off Sheet 2](#_Toc420668214)

[Appendix B: Record of Changes 3](#_Toc420668215)

[Appendix C: Acronyms 4](#_Toc420668216)

[Appendix D: Glossary 5](#_Toc420668217)

[Appendix E: Referenced Documents 6](#_Toc420668218)

[Appendix F: Notes to the Author / Template Instructions 7](#_Toc420668219)

[Appendix G: XLC Template Revision History 8](#_Toc420668220)

[Appendix H: Additional Appendices 9](#_Toc420668221)

List of Figures

**No table of figures entries found.**

List of Tables

[Table 1: Load Model 3](#_Toc420668168)

[Table 2: Change Requests (CRs) 3](#_Toc420668169)

[Table 3: Assumptions 4](#_Toc420668170)

[Table 4: Constraints 4](#_Toc420668171)

[Table 5: Risks 5](#_Toc420668172)

[Table 6: Schedule of Milestones 5](#_Toc420668173)

[Table 7: Test Organization 6](#_Toc420668174)

[Table 8: Performance Test (Script 1 Steps) 7](#_Toc420668175)

[Table 9: Performance Test (Script 2 Steps) 7](#_Toc420668176)

[Table 10: Performance Test Scenarios 8](#_Toc420668177)

[Table 11: Performance Test Scenario Runtime Settings 9](#_Toc420668178)

[Table 12: Performance Test Environment Hardware 11](#_Toc420668179)

[Figure 1: Logical Test Environment 12](#_Toc420668180)

[Figure 2: Physical Test Environment 13](#_Toc420668181)

[Table 13: Tools 13](#_Toc420668182)

[Table 14: Application Server Tier 14](#_Toc420668183)

[Table 15: Database Server Tier 14](#_Toc420668184)

[Table 16: Business Process/Transactions Goals 16](#_Toc420668185)

[Table 17: Business Transactions Statistics Summary 16](#_Toc420668186)

[Table 18: Record of Changes 3](#_Toc420668187)

[Table 19: Acronyms 4](#_Toc420668188)

[Table 20: Glossary 5](#_Toc420668189)

[Table 21: Referenced Documents 6](#_Toc420668190)

[Table 22: XLC Template Revision History 8](#_Toc420668191)

# Executive Summary

The Enterprise Testing Center (ETC) supports the commercial off-the-shelf (COTS) product HP Performance Center (PC) to conduct performance tests and analyze results from executed tests. This Test Plan needs to be completed by vendors seeking to test their applications in the ETC. Performance tests help to determine a system’s and application’s limitations, as well as the maximum number of active users utilizing the application throughout servers.

The Performance Test Plan and Results is a combined document designed to more closely integrate performance test planning and reporting. This document prescribes the performance requirements, load model, performance test approach, assumptions, issues, risks, constraints, milestone/schedule, test organization, performance test script steps, performance test data planning, performance test scenario runs, performance test environment, performance test monitoring tools and metrics, performance test results, and analysis steps for Performance Testing (PT).

The objective of this document is to summarize the tasks that have been performed by each system and/or department supporting the Performance Test (PT) phase.

## Overview: Project Background and Scope

Instructions: Provide an objective of the Application/Project and what it is intended to do in support of the Government Agency.

Briefly describe the changes or new modernization that this Application/Project will include. Examples of changes or new modernization include new software code, enhanced capabilities, or hardware component changes.

# Performance Requirements and Planning

## Performance Requirements

List the performance requirements in the Load Model format located in Section 2.1.1. Write any other details, factors, or performance attributes related to the Performance Requirements. The baseline performance attributes are Business Process, Service Level Agreement (SLA)/Response Time, and Transactions per hour.

The Performance Requirements should define the expected performance standard for the Application and Infrastructure that supports this project. The Performance Requirements should specifically define the transaction rate per Business Process and the expected SLAs/response times per Business Process.

The Business Processes of interest within the Application should be defined based on their level of importance to the application execution.

### Load Model

Complete the Load Model detailed in the table below. Add as many load models as needed to support the Performance Test. Add columns to the Load Model if there are more performance attribute information to share.

The Load Model is used as a repository for the Performance Requirements and the workloads that drive Demand Management. Demand Management is the measurement of the demand for workloads that process on infrastructure. The Load Model includes the following performance attributes: Business Processes, SLA/Response Times, and Transactions per Hour. The Load Model allows for mapping all of the performance attributes together to make specific and measurable Performance Requirements. The Load Model and Performance Requirements complement each other as the Load Model is a repository for the Performance Requirements.

The Load Model can be adjusted, dependent on what the key business processes are for the application. Several Load Models can be placed in this section to explain different workloads. For example, there could be a Production Load Model located here to indicate this project’s application along with other regression workloads that also will be sharing the infrastructure in Production. There could be a Load Model for this application’s workload only. There could be a Load Model for what will be executed in a Performance Test environment which is 50% of Production.

The base Load Model table is presented below:

Table 1: Load Model

| Business Transactions | SLA/response times | Transactions per hour |
| --- | --- | --- |
| Business Process 1 | 4 seconds | 1500 |
| Business Process 2 | 4 seconds | 1900 |
| Business Process 3 | 1 second | 1400 |
| Business Process 4 | 3 seconds | 1350 |
| Business Process 5 | xxxx | xxxx |

## Performance Test Approach

Write the high level approach for performance-testing the Application and Infrastructure.

Document the Performance Test Approach by covering five key areas:

1. Review the System Design Document (SDD) if available, or collect project background information on the application and infrastructure.
2. Leverage a Production Load Model to determine what regression load should also be considered for execution during the performance tests, if available.
3. Determine what load test types should be executed; examples include Peak hour Load Tests, Stress Tests, and 4–24-hour Endurance Tests.
4. What key performance metric areas will be important to monitor or define the pass criteria? The Performance Requirements are strongly recommended to drive the pass/fail criteria, but previous results comparisons can also be considered in the pass/fail decision process.
5. Based on the Application capabilities or Infrastructure changes, what scope will be performance tested? Are there any Change Requests (CR) s related to this activity?

Table 2: Change Requests (CRs)

| Task ID | Description | Project Affected |
| --- | --- | --- |
| CR10000 | TBD | XXXXXX |
| CR10001 | TBD | XXXXXX |
| CR10002 | TBD | XXXXXX |

### Assumptions, Constraints, and Risks

#### Assumptions

Assumptions should be documented concerning the available release software, test environment, dependencies, tools, and test schedule associated with the performance test. Examples are shown below.

Table 3: Assumptions

| No. | Assumption |
| --- | --- |
| 1 | Release x.x Code will be fully functional, having passed Functional and Automated Pass I and II before Performance Testing begins. |
| 2 | All Performance Center Tool software has been installed and configured. |
| 3 | The fully deployed, installed, and configured Web tier, middleware tier, and database servers must be operational in order for performance testing shake-out to begin. |
| 4 | xxxxxxx |

#### Constraints

Constraints should be documented concerning the available release software, test environment, dependencies, tools, test schedule, and other items pertaining to the performance test. Examples are shown below.

Table 4: Constraints

|  |  |  |
| --- | --- | --- |
| No. | Constraint | Impact |
| 1 | The Performance Test environment has 50% of the servers that Production has. | The scaling factor of Performance Test to Production is 50%. All Production Load Models that are executed in Performance Test should be run at 50% of the full Production load Model to represent a 100% Load Test in the AJ Test environment. |
| 2 | The Performance Test environment does not have some of the older data that Production has, which limits some of the data scenarios that can be simulated. | The data in Production has not been purged since 2009; searches in Production intermingle with older data than Performance Test can. This could limit the capability of reproducing some Production issues. |
| 3 | The Performance Test team does not have a commercial tool or an approved Wire Shark-like tool that allows for measuring network response times using packet captures. | The impact of network response times will not be measurable as we determine what areas within the Architecture are responsible for transaction response time cost. This constraint will leave network response time cost-related questions unanswered. |
| 4 | xxxx | xxxx |

#### Risks

Risks should be documented concerning the test schedule, release software, dependencies, tools, test approach test environment and other items pertaining to the performance test. Examples are shown below.

Table 5: Risks

| No. | Risk | Impact | Action/Mitigation | Assigned To |
| --- | --- | --- | --- | --- |
| 1 | If functional errors from validation testing occur and prevent the creation of performance test scripts or performance test execution, execution of performance test project tasks will be delayed until functional errors can be addressed. | HIGH | The team will start Performance Test execution once environment certification, test script validation, and data staging efforts are completed. | Project Manager |
| 2 | If a performance-tuning effort is conducted in the middle of the performance test execution schedule and as a result configuration or code changes are made to the environment, any tests executed prior to the performance-tuning changes should be re-executed. | HIGH | It is recommended that any tests that were executed before the performance tuning changes should be re-executed after the performance-tuning changes. | Project Manager, Performance Engineering |
| 3 | Xx | Xx | Xx | xx |

### Milestones

Key milestones are listed in the table below. Each of the milestones represents a group of tasks on which completion of Performance Testing is dependent. If any of the milestones are listed as “At Risk”, the milestones that follow it will most likely be delayed as well.

Table 6: Schedule of Milestones

| ID | % Done | At Risk | Task | Due Date | Interface |
| --- | --- | --- | --- | --- | --- |
| 1 | 0-100 | Yes or No | Preliminary Project Plan submitted | xx/xx/xxxx | Project Management |
| 2 | 0-100 | Yes or No | Final Project Plan submitted | xx/xx/xxxx | Project Management |
| 3 | 0-100 | Yes or No | Performance Requirements and Production Load Model reviewed and verified | xx/xx/xxxx | Requirements Management and Performance Engineer |
| 4 | 0-100 | Yes or No | Environment Planning | xx/xx/xxxx | Environment Team and Project Management |
| 5 | 0-100 | Yes or No | Test Plan | xx/xx/xxxx | Performance Engineer |
| 6 | 0-100 | Yes or No | Script Development and Data Planning | xx/xx/xxxx | Performance Engineer and Vendor Project Team |
| 7 | 0-100 | Yes or No | Environment Certification and Test Script Validation | xx/xx/xxxx | Project Management and Environment Team |
| 8 | 0-100 |  | Data Staging and Setup | xx/xx/xxxx | Performance Engineer and Vendor Project Team |
| 9 | 0-100 |  | Performance Monitoring Configuration | xx/xx/xxxx | Environment Team and Performance Engineer |
| 10 | 0-100 |  | Test Execution and Analysis | xx/xx/xxxx | Performance Engineer, Monitoring Tool administrators, and Development |

### Test Organization

Document the test organization and any other departments that will be supporting the Performance Test Phase.

Table 7: Test Organization

| Name | Functional Role | Responsibilities |
| --- | --- | --- |
| Name | Project Manager | Facilitating and coordinating all schedules related to SDLC phases and infrastructure |
| Name | Performance Engineering Lead | Manages schedules and activities related to Performance Testing projects |
| Name | Performance Engineer | Prepares for performance test execution, executes performance tests, analyzes performance tests, and tracks problem reports |
| Name | Performance Engineer | Prepares for performance test execution, executes performance tests, analyzes performance tests, and tracks problem reports. |
| Name | Monitoring Support | Monitors performance tests using Performance monitors |
| Name | Application Support | Supports performance test execution as configuration or application issues are found |
| Name | Performance Test Environment Support | Supports and maintains the Performance Test environment |

### Performance Test Script Steps

In this section, the performance test scripts that need to be developed are detailed by user action step as shown in the tables below. For each key Business Process within the Application under Test, a Performance Test script needs to be developed.

Performance Test scripts will be built using Performance Center’s Virtual User Generator (VuGen) component. The following performance test script steps map back to the Load Model defined earlier in Section 2.1.1. The performance test scripts are designed to simulate the Business Processes/Transactions in the Load Model.

Develop performance test scripts that simulate all of the actions in the Business Processes/Transactions documented in the Load Model.

Table 8: Performance Test (Script 1 Steps)

| Name | Application Name: Pecos (for example)  Business Process Name: Submit Health information (for example) |
| --- | --- |
| 1 | Login |
| 2 | Click link |
| 3 | Enter personal ID |
| 4 | Fill out .PDF form and submit |
| 5 | xxxxx |
| 6 | xxxxx |

Table 9: Performance Test (Script 2 Steps)

| Name | Application Name: Pecos (for example)  Business Process Name: Search for Health information (for example) |
| --- | --- |
| 1 | Login |
| 2 | Click Link |
| 3 | Search |
| 4 | Search |
| 5 | xxxxx |
| 6 | xxxxx |

### Performance Test Data Planning

Provide a summary of the test data that will be needed for the Performance Test phase. Provide the details of what data will be needed to support the execution of the performance test scripts for several iterations per the Load Model. There could be needs for dynamic login data and variable data in order to not allow for iterations to be static. The performance test execution iterations should be varied, and the data drives this process.

There could be a need to research the problematic data flows in Production if the application under test is already deployed in Production. Determine if there is a need for performance test data to be pre-loaded in the Performance Test Environment database.

#### Data Preparation

Define the procedure that will be used to prepare the test data for Performance Test. Define the procedures needed to create the test data. Some of the procedures include, but are not limited to:

* Testing the data and database to ensure that they are ready for the current test stage and align with test requirements.
* Identifying/defining any tools needed to create, modify, and manipulate the test data.
* Developing a strategy for obtaining and refreshing test data for every cycle or pass conducted in performance testing. During the performance test, the volume of the test data is large relative to other test stages.

### Performance Test Scenario Runs

The table below provides an example of a short summary of each of the Performance Test scenario runs.

*Instructions: In the short paragraphs below the table, describe the short objective of each of the Performance Test scenarios to be executed. Examples are included.*

Table 10: Performance Test Scenarios

| Test Run | Date | Execution |
| --- | --- | --- |
| Shakeout test | To Be Determined (TBD) | Shake-out performance test scripts and monitors |
| Run 1 | TBD | Baseline – 2 Hour Load Test (Peak Hour Load) |
| Run 2 | TBD | Repeat Baseline – 2 Hour Load Test (Peak Hour Load) |
| Run 3 | TBD | 1 Hour Stress Test (150–300% of Peak Hour Load) |
| Run 4 | TBD | Repeat 1-Hour Stress Test (150–300% of Peak Hour Load) |
| Run 5 | TBD | 4–24-Hour Endurance Test (Average Hour or Varied Load) |
| Run 6 | TBD | Repeat 4–24-Hour Endurance Test (Average Hour or Varied Load) |

Table 11: Performance Test Scenario Runtime Settings

| Test Run | Pacing between Iterations | Think Times |
| --- | --- | --- |
| Run 1 | 6 seconds | 10 seconds |
| Run 2 | 6 seconds | 10 seconds |
| Run 3 | 3 seconds | 10 seconds |
| Run 4 | 3 seconds | 10 seconds |
| Run 5 | 12 seconds | 10 seconds |
| Run 6 | 12 seconds | 10 seconds |

#### Shake-out Test

The Shake-out test is designed to ensure that the performance test scripts are working in the Performance Test Environment. The Shake-out test is also used for making sure the Performance Monitors that are configured for metrics collection are operating as expected. The Shake-out test can also be used to run a 1–5-user test to determine how long it takes for transaction steps to complete. This method is valuable for the runtime settings pacing of the test.

#### Test Run One, Baseline 2 Hour - Peak Hour Load Test (REPEAT in Run 2)

Test Run One is a 2-hour peak hour baseline test that will exercise the top workload transactions in the Application under Test, which are documented above in Table 2-1, Load Model and Table 2-8, Performance Test Script Steps. Peak hour transaction processing will be under examination to determine if the system can maintain response times under the highest anticipated load. Test Run One is designed to collect performance metrics on transaction throughput, response times, and system resource utilization, in comparison to Performance requirements.

Test Run Two is a repeat of Test Run One.

#### Test Run Three, 1 Hour Stress Test (150-300% of Peak Hour Load) (REPEAT in Run 4)

Test Run Three is a 1-hour stress test that will exercise the top workload transactions in the Application under Test, which are documented above in Table 2-1, Load Model and Table 2-8, Performance Test Script Steps. Stressing the system to view how the Application and Infrastructure scales if this Application’s workload grows in the near future will be examined to determine if response times can be maintained. Test Run Three is designed to collect performance metrics on transaction throughput, response times, and system resource utilization, in comparison to Performance requirements.

Test Run Four is a repeat of Test Run Three.

#### Test Run Five, 4-24 Hour Endurance Test (Average Hour or Varied Load) (REPEAT in Run 6)

Test Run Five is a 2-hour peak hour baseline test that will exercise the top workload transactions in the Application under Test, which are documented above in table 2-1, Load Model and Table 2-8, Performance Test Script Steps. This Endurance test will determine if the system resources are recycled for re-use while processing transactions over long periods. Proper re-cycling of memory, CPU, and other system utilization resources is healthy for performance. Test Run Five is designed to collect performance metrics on transaction throughput, response times, and system resource utilization, in comparison to Performance requirements.

Test Run Six is a repeat of Test Run Five.

### Performance Test Environment

*Instructions: Document the Performance Test Environment components based on the categories listed in the table below.*

*Performance testing will be performed in the XXXX Test environment. The table below lists an example of the components within the Performance Test environment that XXXX Application will use.*

Table 12: Performance Test Environment Hardware

| Server Name | Environment Tier | Server vendor model | Hardware Version | OS | Memory (GB) | CPU count | Total Drive Space |
| --- | --- | --- | --- | --- | --- | --- | --- |
| CM5687 | Web Service | Dell | M620 | Linux | 32 GB | 8 cores | 512 GB |
| CM5688 | Web Service | Dell | M620 | Linux | 32 GB | 8 cores | 512 GB |
| CM5689 | Middleware | Dell | M620 | Linux | 32 GB | 8 cores | 512 GB |
| CM5690 | Middleware | Dell | M620 | Linux | 32 GB | 8 cores | 512 GB |
| CM5683 | Middleware | Dell | M820 | Linux | 32 GB | 16 cores | 1 TB |
| CM5685 | Database | Dell | M820 | Linux | 32 GB | 16 cores | 1 TB |
| xxx | xxx | xxx | xxx | xxx | xxx | xxx | xxx |
| xxx | xxx | xxx | xxx | xxx | xxx | xxx | xxx |

As listed below, describe what the Scaling factor between the Production environment that will support the Application under Test, and the Performance Test environment that will support the Application under Test.

The Scaling factors are as follows:

1. Is the hardware in the same vendor family?
2. Does the hardware have the same number of CPUs (processors)?
3. Does the hardware have the same version?
4. Is the OS the same platform (UNIX, Linux, or Windows)?
5. Are the configuration files in Test a match to Production or pulled from Production?
6. Is data populated in the Database to the same level as in Production? If not, what is the ratio?

#### Server Software

*Instructions: Provide information on the software used on the Performance Test environment components.*

*Example:*

*The software used in the performance test environment is:*

* *WebSphere Application Server (WAS) software.*
* *DB/2 database*

#### Logical Test Environment

*Instructions: Provide a Logical Test Environment diagram to illustrate where transactions will process.*

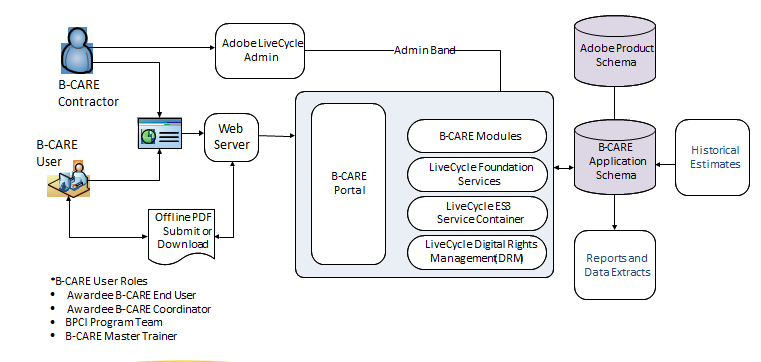


Figure 1: Logical Test Environment

#### Physical Test Environment

*Instructions: Provide a Physical Test Environment diagram to illustrate what components will be used in the Performance Test Environment.*

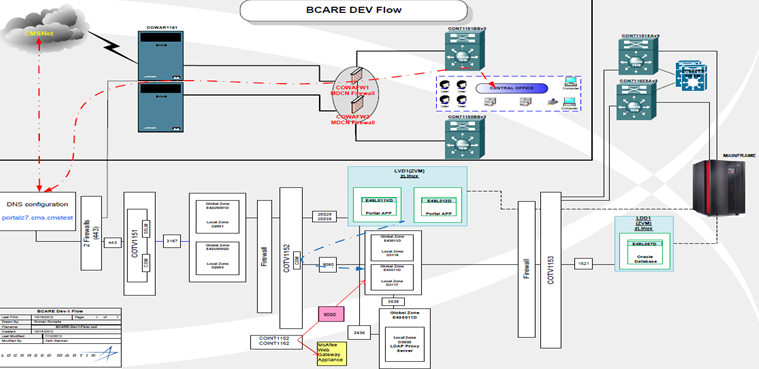


Figure 2: Physical Test Environment

## Performance Test Monitoring Tools and Metrics

*Instructions: List and provide a description of the tools to be used in supporting monitoring the performance tests.*

*The tools listed below will be used to support Performance Testing efforts. They will be used to capture performance metrics as needed, located in the table below.*

*An example is given below.*

Table 13: Tools

| Tool | Purpose |
| --- | --- |
| HP Performance Center | To build performance test scripts  To execute performance test scenarios  To collect transaction response times |
| CA Wily | To measure physical server level utilization statistics and performance metrics |
| HP Deep Diagnostics | To measure physical server level utilization statistics and performance metrics  To measure utilization statistics at the code level (classes and methods) |
| Site Scope | To measure physical server level utilization statistics and performance metrics |

The two tables below describe examples of the various performance metrics that can be captured during the Performance Test stage to view resource usage trends.

Table 14: Application Server Tier

| Metrics | Value Measured |
| --- | --- |
| CPU utilization | CPU utilization |
| Physical Memory Percentage used | Physical Memory Percentage used |
| Memory | Memory utilization |
| Java Virtual Machine (JVM) Runtime/Total Memory | Total memories in the JVM runtime |
| JVM Runtime/Free Memory | Free memories in the JVM runtime |
| Used memories in the JVM runtime |
| JDBC Connections/Concurrent Waiters | Number of threads that are currently waiting for connections |
| JDBC DB Connections/Percent used | Average percent of pool that is in use |
| JDBC DB Connections/Percent maxed | Average percent of the time that all connections are in use |
| Thread Creates | Total number of thread creates |
| Thread Destroys | Total number of threads destroyed |
| Thread Pool/Active Threads | Number of concurrently active threads |
| Thread Pool/Pool Size | Average number of threads in pool |
| Thread Pool/Percent Maxed | Average percent of the time that all threads are in use |
| Heap size | Amount of heap allocated. |
| Memory | Memory utilization  Processes in run queue (Procs r), User Time (cpu us), System time(cpu sy), Idle time (cpu id), Context Switching (cs), Interrupts |
| Disk I/O | Disk I/O utilization  Read/Write per sec (r/s, w/s), Percentage busy (%b), Service Time (svc\_t) |
| Network | Collisions (Collis), Output Packets (Opkts), Input errors (Ierrs), Input Packets (Ipkts) |
| Queue Depth | Measurement of queue depths during the test execution |

Table 15: Database Server Tier

| Metrics | Value Measured |
| --- | --- |
| CPU utilization | CPU utilization |
| Physical Memory Percentage used | Physical Memory Percentage used |
| Memory | Memory utilization  Processes in run queue (Procs r), User Time (cpu us), System time(cpu sy), Idle time (cpu id), Context Switching (cs), Interrupts |
| Disk I/O | Disk I/O utilization  Read/Write per sec (r/s, w/s), Percentage busy (%b), Service Time (svc\_t) |
| Network | Collisions (Collis), Output Packets (Opkts), Input errors (Ierrs), Input Packets (Ipkts) |

# Execution and Analysis

*Instructions: In this section, compile the Performance Test run log details and the Performance Test results. Also provide an overall analysis of the Application and Infrastructure being examined.*

*An example is provided below.*

## Performance Test Results and Analysis

### Test Run 1 – Baseline Two Hour – Peak Hour Load Test

**Performance Requirement**

Each Business Process should achieve a transaction per hour goal while processing within the allocated SLA response times.

Table 16: Business Process/Transactions Goals

|  |  |  |
| --- | --- | --- |
| Business Transactions | SLA/ Response Times | Transactions per Hour |
| Business Process 1 | 4 seconds | 1500 |
| Business Process 2 | 4 seconds | 1900 |
| Business Process 3 | 1 second | 1400 |
| Business Process 4 | 3 seconds | 1350 |
| Business Process 5 | xxxx | xxxx |

**Observations**

This load test was executed with 35 virtual users processing the load model. The server processing time for all transactions (except Business Process 1) were within the 4-second response time. Business Process 4 had a 90th percentile measurement of 9.28 seconds during this test run. Numerous exceptions were noticed during this test run related to obtaining the same record in the queue by two users.

**Pass/Fail Analysis Summary**

Business Process 4 failed during Performance Test execution. It has been determined that a code change is needed to improve Business Process 4. A decision will be made if the application can be deployed with this issue or if more time will be allocated for the fix to be developed and tested.

Table 17: Business Transactions Statistics Summary

| Business Transactions | SLA/response times Requirement | SLA/response times Actual | Transactions per hour Requirement | Transactions per hour Actual |
| --- | --- | --- | --- | --- |
| Business Process 1 | 4 seconds | 3.6 secs | 1500 | 1500 |
| Business Process 2 | 4 seconds | 2.4 secs | 1900 | 1900 |
| Business Process 3 | 1 second | 0.6 secs | 1400 | 1400 |
| Business Process 4 | 3 seconds | 9.28 secs | 1350 | 1214 |
| Business Process 5 | 5 seconds | 3.9 secs | xxxx | xxxx |

### Test Run 2

Performance Requirement

Observations

Pass/Fail Analysis Summary

### Test Run 3

Performance Requirement

Observations

Pass/Fail Analysis Summary

### Test Run 4

Performance Requirement

Observations

Pass/Fail Analysis Summary

### Test Run 5

Performance Requirement

Observations

Pass/Fail Analysis Summary

### Test Run 6

Performance Requirement

Observations

Pass/Fail Analysis Summary

1. Test Sign-off Sheet

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Sign-Off Sheet | | | | | |
| This sign-off sheet indicates that the signing reviewers agree with the final test deliverables in relation to release <XX.XX> of <Project>. The signatures attest that all deliverables have been completed and all required test activities have been successfully accomplished and verified, thereby acknowledging the completion of the test stage.  Additional comments: | | | | | |
| Concur: | | | | | |
|  |  |  |  |  |  |
|  | Government Task Lead (GTL) |  | Signature |  | Date |
| Concur: | | | | | |
|  |  |  |  |  |  |
|  | Project/Release Manager |  | Signature |  | Date |
| Concur: | | | | | |
|  |  |  |  |  |  |
|  | Project/Release Manager |  | Signature |  | Date |
| Concur: | | | | | |
|  |  |  |  |  |  |
|  | Performance Test Lead |  | Signature |  | Date |
| Approve: | | | | | |
|  |  |  |  |  |  |
|  | Performance Engineer |  | Signature |  | Date |

1. Record of Changes

*Instructions: Provide information on how the development and distribution of the Master Performance Testing Template will be controlled and tracked. Use the table below to provide the version number, the date of the version, the author/owner of the version, and a brief description of the reason for creating the revised version.*

Table 18: Record of Changes

| Version  Number | Date | Author/Owner | Description of Change |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. Acronyms

Instructions: Provide a list of acronyms and associated literal translations used within the document. List the acronyms in alphabetical order using a tabular format as depicted below.

Table 19: Acronyms

| Acronym | Literal Translation |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

1. Glossary

Instructions: Provide clear and concise definitions for terms used in this document that may be unfamiliar to readers of the document. Terms are to be listed in alphabetical order.

Table 20: Glossary

| Term | Definition |
| --- | --- |
|  |  |
|  |  |
|  |  |
|  |  |

1. Referenced Documents

Instructions: Summarize the relationship of this document to other relevant documents. Provide identifying information for all documents used to arrive at and/or referenced within this document (e.g., related and/or companion documents, prerequisite documents, relevant technical documentation, etc.).

Table 21: Referenced Documents

| Document Name | Document Location and/or URL | Issuance Date |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

1. Notes to the Author / Template Instructions

This document is a template for creating a Master Performance Testing Template for a given investment or project. The final document should be delivered in an electronically searchable format. The Master Performance Testing Template should stand on its own with all elements explained and acronyms spelled out for reader/reviewers, including reviewers outside CMS who may not be familiar with CMS projects and investments.

This template includes instructions, boilerplate text, and fields. The developer should note that:

* *Each section provides instructions or describes the intent, assumptions, and context for content included in that section. Instructional text appears in blue italicized font throughout this template.*
* *Instructional text in each section should be replaced with information specific to the particular investment.*
* *Some text and tables are provided as boilerplate examples of wording and formats that may be used or modified as appropriate.*

When using this template, follow these steps:

1. *Table captions and descriptions are to be placed centered, above the table.*
2. *Modify any boilerplate text, as appropriate, to your specific investment.*
3. *Do not delete any headings. If the heading is not applicable to the investment, enter “Not Applicable” under the heading.*
4. *All documents must be compliant with Section 508 requirements.*
5. *Figure captions and descriptions are to be placed centered, below the figure. All figures must have an associated tag providing appropriate alternative text for Section 508 compliance.*
6. *Delete this “Notes to the Author / Template Instructions” page and all instructions to the author before finalizing the initial draft of the document.*
7. XLC Template Revision History

The following table records information regarding changes made to the XLC template over time. This table is for use by the XLC Steering Committee only. To provide information about the controlling and tracking of this artifact, please refer to the Record of Changes section of this document.

Table 22: XLC Template Revision History

| Version  Number | Date | Author/Owner | Description of Change |
| --- | --- | --- | --- |
| 1.0 | 11/7/2014 | Enterprise Test Center | Baseline document. See CR 14-006. |
| 1.1 | 02/02/2015 | Surya Potu,  CMS/OEI/DPPIG | Updated CMS logo. |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. Additional Appendices

Instructions: Utilize additional appendices to facilitate ease of use and maintenance of the document.