**What to include in a performance test plan**

<https://searchsoftwarequality.techtarget.com/tip/What-to-include-in-a-performance-test-plan>

Before performance testing can be performed effectively, a detailed plan should be formulated that specifies how performance testing will proceed from a business perspective and technical perspective. At a minimum, a performance testing plan needs to address the following:

* Overall approach
* Dependencies and baseline assumptions
* Pre-performance testing actions
* Performance testing approach
* Performance testing activities
* In-scope business processes
* Out-of-scope business processes
* Performance testing scenarios
* Performance test execution
* Performance test metrics

As in any testing plan, try to keep the amount of text to a minimum. Use tables and lists to articulate the information. This will reduce the incidents of miscommunication.

**Overall approach**  
This section of the performance plan lays out the overall approach for this performance testing engagement in non-technical terms. The target audience is the management and the business. Example:

"The performance testing approach will focus on the business processes supported by the new system implementation. Within the context of the performance testing engagement, we will:

* Focus on mitigating the performance risks for this new implementation.
* Make basic working assumptions on which parts of the implementation need to be performance-tested.
* Reach consensus on these working assumptions and determine the appropriate level of performance and stress testing that shall be completed within this compressed time schedule.

This is a living document, as more information is brought to light, and as we reach consensus on the appropriate performance testing approach, this document will be updated."

**Dependencies and baseline assumptions**  
This section of the performance test plan articulates the dependencies (tasks that must be completed) and baseline assumptions (conditions testing believes to be true) that must be met before effective performance testing can proceed. Example:

"To proceed with any performance testing engagement the following basic requirements should be met:

* Components to be performance tested shall be completely functional.
* Components to be performance tested shall be housed in hardware/firmware components that are representative or scaleable to the intended production systems.
* Data repositories shall be representative or scaleable to the intended production systems.
* Performance objectives shall be agreed upon, including working assumptions and testing scenarios.
* Performance testing tools and supporting technologies shall be installed and fully licensed."

|  |  |
| --- | --- |
| https://cdn.ttgtmedia.com/images/spacer.gif | |
| |  |  |  | | --- | --- | --- | | |  | | --- | | **More information about performance testing** | | [Testing for performance, part 1: Assess the problem space](https://searchsoftwarequality.techtarget.com/tip/Testing-for-performance-part-1-Assess-the-problem-space)  [Testing for performance, part 2: Build out the test assets](https://searchsoftwarequality.techtarget.com/tip/Testing-for-performance-part-2-Build-out-the-test-assets)  [Testing for performance, part 3: Provide information](https://searchsoftwarequality.techtarget.com/tip/Testing-for-performance-part-3-Provide-information) | | | https://cdn.ttgtmedia.com/images/spacer.gif |
| https://cdn.ttgtmedia.com/images/spacer.gif | |

**Pre-performance testing actions**  
This section of the performance test plan articulates pre-testing activities that could be performed before formal performance testing begins to ensure the system is ready. It's the equivalent to smoke testing in the functional testing space. Example:

"Several pre-performance testing actions could be taken to mitigate any risks during performance testing:

* Create a "stubs" or "utilities" to push transactions through the QA environment -– using projected peak loads.
* Create a "stubs" or "utilities" to replace business-to-business transactions that are not going to be tested or will undergo limited performance. This would remove any dependencies on B2B transactions.
* Create a "stubs" or "utilities" to replace internal components that will not be available during performance testing. This would remove any dependencies on these components.
* Implement appropriate performance monitors on all high-volume servers."

**Performance testing approach**  
This section of the performance plan expands on the overall approach, but this time the focus is on the both the business and technical approach. As an example:

"The performance testing approach will focus on a logical view of the new system implementation. Within the context of the performance testing engagement, we will:

* Focus on mitigating the performance risks for this new implementation.
* Make basic working assumptions on which parts of the implementation need to be performance-tested.
* Reach consensus on these working assumptions and determine the appropriate level of performance that shall be completed.
* Use a tier 1 performance testing tool that can replicate the expected production volumes.
* Use an environment that replicates the components (as they will exist in production) that will be performance-tested -– noting all exceptions.
* Use both production and non-production (testing) monitors to measure the performance of the system during performance testing."

**Performance testing activities**  
This section of the performance test plan specifies the activities that will occur during performance testing. Example:

"During performance testing the following activities shall occur:

* Performance test shall create appropriate loads against the system following agreed-upon scenarios that include:
  + User actions (workflow)
  + Agreed-upon loads (transactions per minute)
  + Agreed-upon metrics (response times)

* Manual testing and automated functional tests shall be conducted during performance testing to ensure that user activities are not impacted by the current load.

* System monitors shall be used to observe the performance of all servers involved in the test to ensure they meet predefined performance requirements.

* Post-implementation support teams shall be represented during performance testing to observe and support the performance testing efforts."

**In-scope business processes**  
This section of the performance test plan speaks to which aspects of the system are deemed to be in-scope (measured). Example:

"The following business processes are considered in-scope for the purposes of performance testing:

* User registration
* Logon/access
* Users browsing content
* Article sales & fulfillment
* Billing

Business process list formed in consultation with: Business Analysts, Marketing Analyst, Infrastructure, and Business Owner."

**Out-of-scope business processes**  
This section of the [performance testing plan](https://searchsoftwarequality.techtarget.com/feature/Compare-software-performance-and-load-testing-tools) speaks to which aspects of the system are deemed to be out-of-scope (measured). Example:

"Business processes that are considered out-of-scope for the purposes of testing are as follows:

* Credit check
  + Assumption: Credit check link shall be hosted by a third party -- therefore no significant performance impact.
* All other business functionality not previously listed as in-scope or out-of-scope
  + Assumption: Any business activity not mentioned in the in-scope or out-of-scope sections of this document does not present a significant performance risk to the business."

**Formulating performance testing scenarios**  
The existence of this section within the body of the performance testing plan depends on the maturity of the organization within the performance testing space. If the organization has little or no experience in this space, then include this section within the plan otherwise include it as an appendix. Example:

"Formulation of performance testing scenarios requires significant inputs from IT and the business:

* Business scenario
  + The business scenario starts as a simple textual description of the business workflow being performance-tested.
  + The business scenario expands to a sequence of specific steps with well-defined data requirements.
  + The business scenario is complete once IT determines what (if any) additional data requirements are required because of the behavior of the application/servers (i.e. caching).

* Expected throughput (peak)
  + The expected throughput begins with the business stating how many users are expected to be performing this activity during peak and non-peak hours.
  + The expected throughput expands to a sequence of distinguishable transactions that may (or may not) be discernable to the end user.
  + The expected throughput is completed once IT determines what (if any) additional factors could impact the load (i.e. load-balancing)

* Acceptance performance criteria (acceptable response times under various loads)
  + Acceptance performance criteria are stated by the business in terms of acceptable response times under light, normal and heavy system load. System load being day-in-the-life activity. These could be simulated by other performance scenarios.
  + The performance testing team then restates the acceptance criteria in terms of measurable system events. These criteria are then presented to the business for acceptance.
  + The acceptance criteria are completed once IT determines how to monitor system performance during the performance test. This will include metrics from the performance testing team.

* Data requirements (scenario and implementation specific)
  + The business specifies the critical data elements that would influence the end-user experience.
  + IT expands these data requirements to include factors that might not be visible to the end user, such as caching.
  + The performance testing team working with IT and the business creates the necessary data stores to support performance testing."

**Performance test execution**  
Once again the existence of this section of the performance test plan is dependent upon the maturity of the organization within the performance testing space. If the organization has significant performance testing experience, then this section can become a supporting appendix. Example:

"Performance testing usually follows a linear path of events:

* Define performance-testing scenarios.
* Define day-in-the-life loads based on the defined scenarios.
* Execute performance tests as standalone tests to detect issues within a particular business workflow.
* Execute performance scenarios as a "package" to simulate day-in-the-life activities that are measured against performance success criteria.
* Report performance testing results.
* Tune the system.
* Repeat testing as required."

**Performance test metrics**  
The performance test metrics need to track against acceptance performance criteria formulated as part of the performance testing scenarios. If the organization has the foresight to articulate these as performance requirements, then a performance requirements section should be published within the context of the performance test plan. The most basic performance test metrics consist of measuring response time and transaction failure rate against a given performance load -- as articulated in the performance test scenario. These metrics are then compared to the performance requirements to determine if the system is meeting the business need.

**Closing comments**  
This article can speak only to generalities; the specifics depend on the system and situations being tested. As a closing note, non-performance testing activities are often referred to as performance testing -- what I like to call warm-and-fuzzy performance testing. If you are not replicating expected production loads, then you are not doing performance testing.