PROJECT Slide Templates

Presented by Donny Soh

Types of Performance Testing

Presented by Donny Soh

Performance Testing

Performance Test	
Load testing (volume testing)	examines behaviour under a range of loads tests whether or not the system meets its performance requirements for the load expected on the system simulates what happens in normal use of the real application
Stress testing	examines the behaviour of the system when operating beyond capacity establishes the limits in which the system can be expected operate checks for graceful degradation
Soak testing (stability testing)	examines behaviour over a long period of time and checks for problems that appear slowly over time e.g. memory leaks

Performance Testing

Performance Test	
Performance Regression testing (smoke testing)	Smoke testing gets its name from a simple analogy: just as you would check for smoke to see if a new piece of hardware is functioning properly without catching fire, smoke testing in software development is a preliminary test to check if the basic functions of a program work without crashing. checks for changes in performance after code changes this applies to performance as well as functionality. unacceptable performance is a “defect” adding new features frequently impacts performance
Performance Benchmarking	establishes performance against standardised loads reviewers often use benchmarks to compare products, or different versions of the same product benchmarks can also be used internally to evaluate whether performance has improved or degraded the performance of a system prior to tuning is called the baseline

Performance Testing

Performance Test	
Pipe clean testing	validates the performance tests themselves checks that instrumentation is working and the test scripts are correct
Isolation testing	checks individual components of the system
User acceptance testing	e.g. users may complain that the system is too slow

Eliciting Performance Requirements

Presented by Donny Soh

Eliciting Performance Requirements

Questions to be answered might include:

- what aspects of performance matter to whom and why?
- what resources are available to improve performance?
- what load is expected, and how is it shaped?
  - are surges expected?
  - will the load vary seasonally?
  - how is the load expected to grow over time?
- what transaction times are expected and why?
- are there any safety issues or regulation surrounding the system's use?
- are there any competing products and how do they perform?

Eliciting Performance Requirements

If a system is already in place, questions might also include:

- are there any legacy systems that need to be worked with?
- what performance measurements already exist?
- what performance tests have been run?
- is the current system overloaded, or perceived to be?
- how does the existing architecture affect performance?
  - where are the bottlenecks?
  - does the existing architecture constrain the options for addressing performance?
- does the organisation have performance and capacity planning processes in place?

Documenting Performance Requirements

Bondi (2015) suggests the following structure for a performance requirement:

1. Requirement number
2. Title
3. Statement of requirement
4. Supporting commentary
5. List of precedents, sources, standards
6. Derivation of quantities
7. List of dependent requirements
8. List of assumptions and precedent performance requirements
9. Sources of measurement data
10. Name of a subject matter expert on this requirement

(adapted from Bondi, Table 7.2)

Document Example

Suggested Project SLIDE FOUR

Requirement number

Requirement No. PERF-01

Title

Landing Page Response Time

Statement of Requirement

Statement of Requirement

- The page index.html, including all graphics, shall load within two seconds for at least 99% of requests made from computers within the local area network, when the system is unloaded.

Supporting Commentary

Supporting Commentary

- Loading times for computers located outside the local area network may be greater due to network configuration outside our control.

List of precedents, sources, standards

- **Sources**
- Response time sufficient for uninterrupted work; see Nielsen (1993).

Sources of measurement data

- **Measurement**
- The response time shall be measured using Firefox Developer Tools.

Derivation of quantities

- **Derivation of quantities**
- The response time of loading image information should be less than 2 seconds as it is found that up to 2 seconds is linked with casual work.

Assumptions

- **Assumptions**
- Bandwidth of the server is constant

Subject Matter Expert

- **Subject Matter Expert**
- Audrey Cheong, User Experience Manager

Suggested Documentations for Performance Testing

- a *performance requirements* document that describes the different **performance requirements** the computer system has to fulfill

- a *performance measurement plan* that describes the **instruments** to be used to measure performance

- a *capacity management plan* that describes how the system will be **monitored after deployment**

- a *performance test plan* that describes **how the performance requirements are to be tested**

Suggested Project SLIDE FIVE

Performance Requirements- Patterns

Eckhardt et al. (2016) suggest the following patterns for requirements:

Response time									
In x% of all cases	the system	must/shall		a		of		milliseconds	
In case of x	the operation	could/may	have	a mean	response time	of less than	x	seconds	[between event A and event B]
	the component	must not		a median		of no more than		minutes	
				a maximal		of no less than		hours	
				a minimal		of more than			

Specific Requirements

Throughput									
In x% of all cases	the system	must/shall		a	transaction rate	of		items	millisecond
In case of x	the operation	could/may	have	a mean	throughput	of less than	x	requests per	second
	the component	must not		a median	processing speed	of no more than		orders	minute
				a maximal		of no less than		bytes	hour
				a minimal		of more than		etc.	

Specific Requirements

Document Example

Requirement number

Requirement No. PERF-01

Title

Landing Page Response Time

Statement of Requirement

Statement of Requirement

- The page index.html, including all graphics, shall load within two seconds for at least 99% of requests made from computers within the local area network, when the system is unloaded.

Supporting Commentary

Supporting Commentary

- Loading times for computers located outside the local area network may be greater due to network configuration outside our control.

List of precedents, sources, standards

- **Sources**
- Response time sufficient for uninterrupted work; see Nielsen (1993).

Sources of measurement data

- **Measurement**
- The response time shall be measured using Firefox Developer Tools.

Derivation of quantities

- **Derivation of quantities**
- The response time of loading image information should be less than 2 seconds as it is found that up to 2 seconds is linked with casual work.

Assumptions

- **Assumptions**
- Bandwidth of the server is constant

Subject Matter Expert

- **Subject Matter Expert**
- Audrey Cheong, User Experience Manager

Executing Performance Tests

Presented by Donny Soh

Test Environment

Performance testing has much in common with other kinds of scientific experiments:

- test results should be reproducible
  - the test procedure should be unambiguous and well-documented
  - control the parameters, e.g. do not run other programs while testing
- instrumentation should be validated
  - use quality tools, be aware of bugs
- test results should be recorded

Test Environment

For a test to represent the behaviour of the real system, it must be set up similarly to the real system

- the test configuration should match the real configuration
- databases may need to be prepared with data similar to that of the real system
  - record the start and end states of the database if possible
- hardware devices might need to be simulated

Suggested Project SLIDE SIX

Test Environment

Setting up a test environment requires several steps (Molyneaux, 2014):

- Provisioning of hardware and software.
- Provisioning of load testing tools.
- Development of use cases and scripts.
- Creation of test data.
- Instrumentation.
- Dealing with problems identified by tests.

Be aware that the measurement instruments may themselves affect the test

- profilers, system monitors, etc., are themselves processes that consume computing resources
- inserting measurement tools into your code changes the program
- measurement tools may themselves contain bugs!
- it may be necessary to test the tools themselves to ensure efficiency and correctness

Test Plan and Playbook

To improve reproducibility, performance tests can be conducted according to a playbook

- the playbook sets out the steps that should be taken to conduct the test
- how is the system initialised?
- what steps are to be taken?
- how are the results to be recorded?
- load generation tools can follow scripts to generate a standardised load

Test Environment

The test environment need not be of the same scale as the real environment but some care is necessary:

- hosting all of the components of a distributed system on a single host may not model network delays
- using a single processor or single disk may not test how the system scales to multiprocessor systems or disk arrays

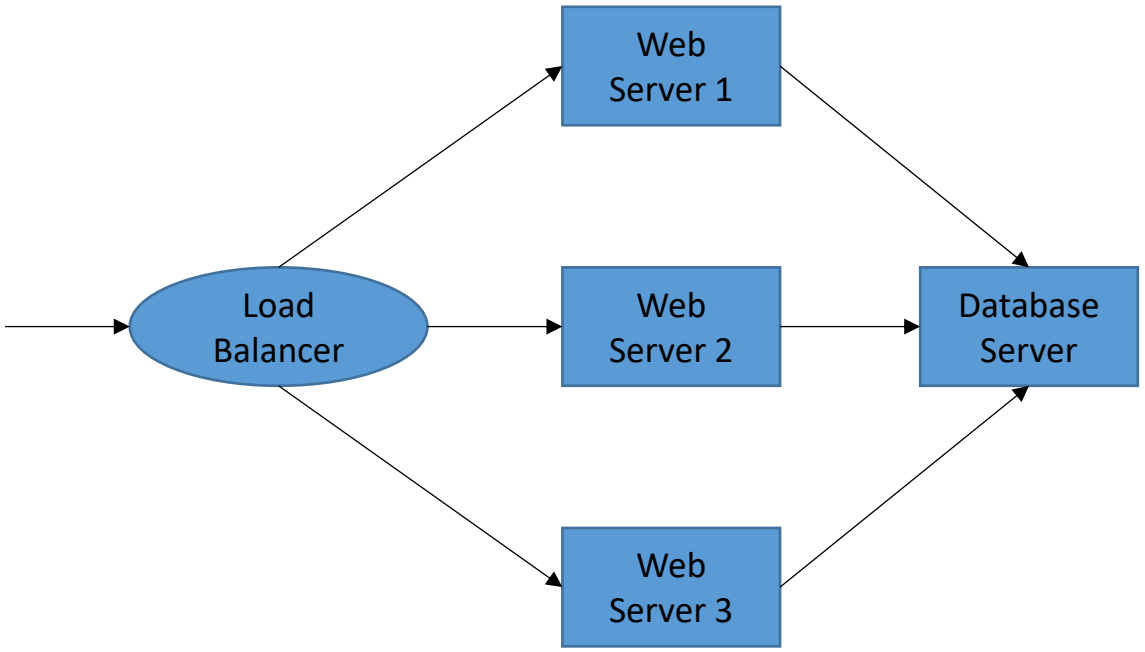
The test environment should have the same structure as the production environment.

If the test is conducted on a scaled-down version of the real system, all components should be scaled similarly:

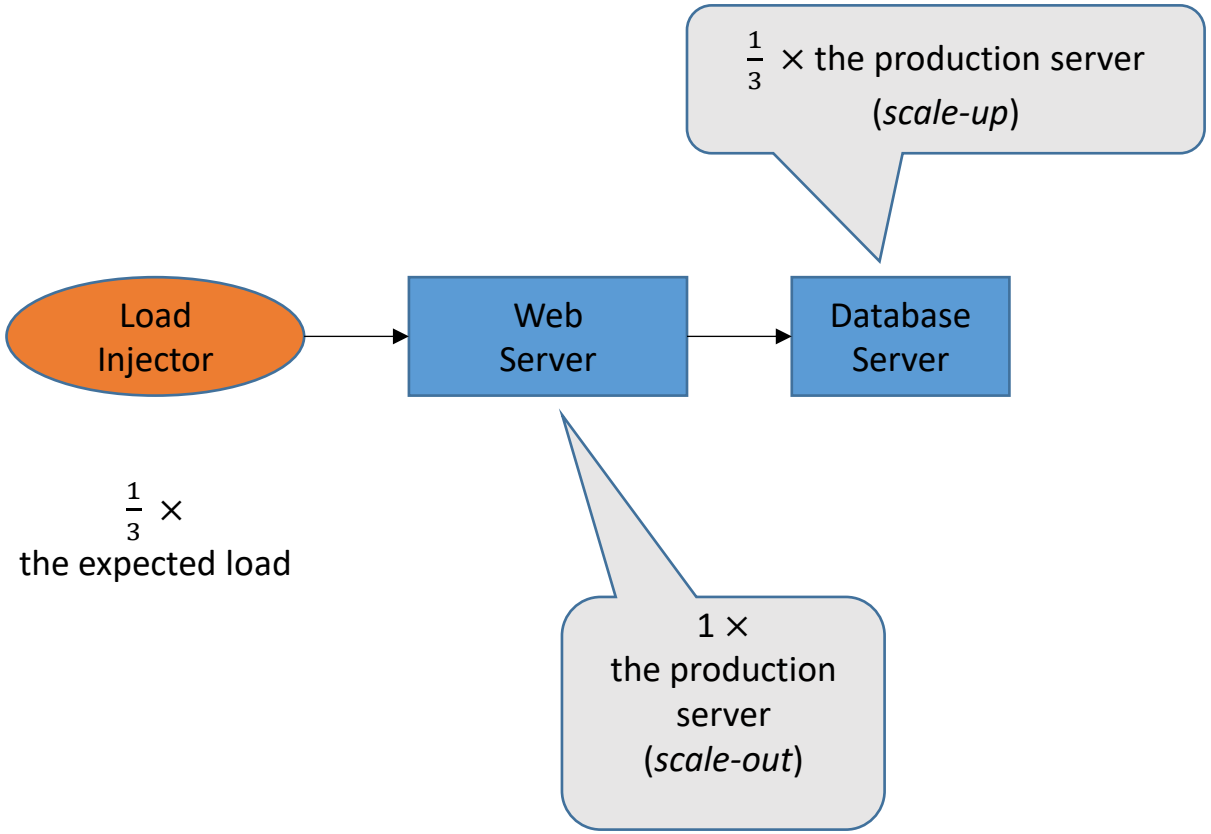
- test data size
- number of CPUs
- amount of storage
- etc.

Test Environment

Production Environment



Test Environment



Suggested Project SLIDE SEVEN

Test Plan: Example

- **Requirement No.** PERF-02
- **Upload File**

- **Test Environment**

Gigabit Ethernet LAN

Windows 10, 64-bit
Java 9 Standard Edition, 64-bit
Jmeter Version 4.0.
…

Ubuntu Server 18.04 LTS
Apache httpd 2.4
…

- **Test Playbook**

1. Initialise the database with an empty table.
2. Start the database server.
3. Start the web server.
4. jmeter –n –t PERF-02-Upload-File.jmx
5. Monitor JMeter output for errors.
6. Verify that no response time exceeded 1000ms when the load was fewer than forty virtual users.