PERFORMANCE TESTING

CASE STUDY

Part 2: Understanding the Situation -- Suggested Answers to Part 1

Ross Collard
rcollard@attglobal.net
Collard & Company

Version 4.9, January 2004

Case Study 2.1

Suggested Answers to Part 1 of the Case Study

Instructions

Review and critique these suggested answers to the questions in Part 1, and see if you agree. Do not assume that the answers provided here are the only possible answers or necessarily the best answers.

Explore and surface the main assumptions underlying these answers (which often are unstated). Ask if the assumptions are reasonable, and where changes in the assumptions would lead to different answers.

Expect to spend 30 to 45 minutes on this critique.

Answers to Exercise 1: Determining the Objectives

(1.1) What do we want to accomplish with this performance testing project?

(a) Why are we doing this test?

A common answer to this question is: "Because the boss told me to!" Unless we have a clear justification, a sense of purpose and a sense of the expectations, though, it is not good idea to mindlessly proceed.

The performance and robustness of the order processing is critical to the success of the book club. The primary objective of the new system is to improve the book club's competitive edge and profitability, and system performance is a key to achieving these objectives:

- The primary impetus for this performance testing project is to assess if the new system will improve the book club's competitive position. Externally, visitors to the Web site will discontinue their sessions and not return if the TB site is perceived to be too slow, unreliable or unavailable. The competition is only three clicks away.

- A major reason for the new system is to improve productivity by 25%. The desired productivity improvement depends on the system performance for the internal staff, so we are interested in the question of whether the new system will facilitate these internal users' productivity. This productivity improvement also depends on transferring tasks from the internal staff to the visitors to the Web site, so the performance of the features for the external visitors will also impact the internal staff productivity.

- Both the business groups and the IS group which will be supporting the system want a confirmation that the performance goals, and the service level agreements (SLAs)

derived from those goals, are realistic and achievable.

In summary, there are significant risks of going live and finding that the performance of the system in live operation is not acceptable. The risks could materialize in several ways: interruption of business services, lost sales, dissatisfied customers and a poor image among potential future members.

The need to mitigate these risks is sufficient to justify the performance measurement project. While the probability of a major disruption may be low, the organization cannot afford the worst-case consequences if the new system goes live and the result is a fiasco.

(b) Who do we need to satisfy?

The constituencies, who have a vested interest, needs and expectations of the performance test results, include:

- The senior managers of the book club, who have invested considerable funds to build the new order processing system

- The book club customers and Web site visitors also have a vested interest in the system performance, but this is an indirect interest -- they will not be involved in performance testing, unless some kind of beta test or field test is done. Because these external parties cannot directly participate in the project, we advocate that an internal group such as the marketing group be appointed as the champion and representative of the external parties.

- The business group managers from each of the main business groups, namely the customer service group, the catalog publishing group, and the warehouse distribution group. Each of these managers is responsible for the productivity of his or her group in the on-going operations, and so is interested in the system's performance for his or her specific group.

- The information systems (IS) group, who will be responsible for maintaining and supporting the system in live operation after the testing, and for meeting the service level agreements (SLAs), and who may be penalized if these SLAs are not met.

  The system developers and any others who are responsible for delivering an acceptable working system. These people may need to tune, debug, and if necessary re-architect the system in order to meet its performance and robustness goals. They include systems architects, network engineers and database administrators, and they need specific, detailed feedback from the testing efforts in order to perform the tuning and debugging.

Suggested Answers to Part 1 of the Case Study

While it was not listed earlier as one of the questions to consider, an important question to ask -- once we have identified the constituencies -- is what the success factors are for each distinct constituency.

(c) What do they want to know?

The section entitled: "Performance Testing Objectives", in the Description of the Situation (in Part 1 of this case study), provides the list of questions which the managers want to see addressed by the performance test project.

These stated objectives should not necessarily be accepted as holy, sacrosanct and not to be questioned by the performance test team. The performance test team needs to be involved early in the discussion and formulation of these objectives, and ask these questions:

- Do the managers' stated performance objectives seem reasonable?
- Are their performance objectives measurable and testable?
- What questions do the decision makers plan to answer, with the data gathered in this testing project?
- How can the performance test team "reverse engineer", that is, start with the set of questions which need to be answered and work backwards, in order to determine what to measure?

(d) What SHOULD they want to know?

We will leave this question open for debate.

(e) How will they use the results of the performance test?

To be able to answer his question, we need to assume that the performance testers have obtained a sense of what the decision makers want to know. The performance testing project will provide information to decision-makers, so that they can make informed decisions on these issues:

- The senior managers:
  - Will this system be able to sustain our competitive advantage in the marketplace?
  - Is the system ready to place in live operation, and if not, what needs to be done to get there?

  - Is additional performance tuning needed before the system goes live?
  - Is re-work (e.g., re-writing code to optimize it) needed before the system goes live?
  - What increases in the hardware and network capacity are needed, if any, before the system goes live?
  - Is the system performance is so poor that tuning unlikely to solve the performance problems, and more drastic actions are needed? These more drastic actions may include a re-design or re-writing of the system, moving to some back-up contingency plan, such as continuing to use the existing system on a temporary basis, threatening the vendors with litigation, etc.
- Can the system scale to accommodate future growth?

o The business group managers:
- Will the system enable my particular business unit to be productive?
- Will internal user satisfaction and morale be improved by the new system? Can the system handle spikes (peak demands)?

o The information systems (IS) group:
- Can we maintain and support this system?
- Can the service level agreements (SLAs) be met?

o The book club customers and Web site visitors:
- Do I get stress-free service (prompt, reliable and efficient service), from TB?
- Is the Web site accessible when I want to use it?

o The system developers and implementors:
- Is the client (the book club) satisfied? Are we going to get paid?
- How does the system need to be tuned or re-worked?

This list of questions and information needs is not complete. In addition, I have grouped the questions together under the constituencies (i.e., the types of people) involved in the system. However, there is not necessarily an exclusive ownership of the issues. For example, the question "Can the system scale to accommodate future growth?" is listed under the senior manager category above, but every constituency has an interest in this issue.

(1.2) Which of the business objectives can -- or cannot -- reasonably be addressed in performance testing project?

The business objectives for the new system are reproduced below, together with an opinion of

Suggested Answers to Part 1 of the Case Study

whether each can or should be addressed in performance testing:

- o "Support the operations of the book club, by providing the features listed below."
  - No, this business objective does not need to be addressed by the performance testing project, because the feature correctness should have been separately addressed by a feature testing project.

- o "Improve productivity in the customer service and warehouse distribution groups by 25%."
  - Yes and no: this business objective can be partly addressed. The performance testers can assess whether the new system will facilitate or hinder user productivity, but they cannot evaluate whether the business unit managers will take the steps to actually achieve these possible productivity improvements with the system in live operation.

- o "Distribute 95% of all books within 15 working days from the date of an order and at the least available shipping cost."
  - No, this business objective should not be addressed by the performance testers, because it is more appropriate for the warehouse distribution group and mail room to address this area.

- o "Distribute books with 98% accuracy, i.e., only 2% of the books or less should be returned because the wrong book was shipped to a member or because the member's address was incorrect."
  - No, this business objective should not be addressed by the performance testers, because it is more appropriate for the warehouse distribution group and mail room to address this area.

- o "Answer 95% of customer telephone calls (e.g., queries about order status), within 3 minutes."
  - Yes and no: this business objective can be partly addressed. The performance testers can arrange a telephone-based test, using either tools to mimic people making phone calls or actual people calling. However, it will be difficult for the testers to put together a big enough group of "volunteers" to realistically load the system in test mode, and the tools are expensive.

- o "Support the projected growth in membership for the next 5 years."
  - Yes: this business objective can be addressed in the performance and load testing.

Let's pause here and reflect what these opinions mean. Out of the six objectives listed above

Suggested Answers to Part 1 of the Case Study

(which were reproduced from the earlier background description of the situation), only one is unambiguously addressed by the performance testing project. This is not a high degree of matching, and there are three implications of the mismatch:

(a) The performance testers had better be careful about what they commit to, and manage expectations.

(b) If there's no linkage of the performance test results to what the business managers care about, we might want to question the whole rationale for performance testing.

(c) There may be an unstated but very important business objective here. (Just because it is unstated does not mean it is not real.) The unstated objective is to ensure that the system is competitive (e.g., sufficiently fast, productive and able to handle the work volume), and efficient (e.g., uses expensive resources efficiently). This unstated objective if true is enough to justify the entire performance testing project. Since the objective was not explicitly stated, though, it is a very good idea to go back and double-check with the business managers that this objective is as important as we think it is.

(1.3) Overall, are the performance goals for the system (i) relevant and significant, (ii) realistic and probably feasible to attain, and (iii) testable or measurable? Has any major goal been omitted?

Based on the available information, the goals all appear to be relevant and significant. No major goal has been omitted. However, the stated goals are not specific enough to judge if they are realistic or testable.

(1.4) Which performance testing objectives can be directly linked back to one or more specific business objectives? Which cannot?

It is always a good idea to ensure the testing objectives are aligned with the business objectives, including the unstated ones. This section restates the testing objectives which were listed earlier, together with an opinion on each one.

(a) Are response times satisfactory when the system is operating under a realistic load?
- Yes, this ties back to the business objectives for user satisfaction, service levels and productivity.

(b) Does the system operate correctly when accessed simultaneously by multiple users?
- Yes and no. This ties back to a business objective, which is to ensure the system works correctly. But feature interactions are better addressed as part of feature

testing; this is outside the scope of the performance test.

(c) How well does the system handle heavy loads?
- Yes, this ties back to the business objectives for user satisfaction, productivity and ability to grow.

(d) Will performance and reliability levels be maintained over an extended period of use?
- Yes, this ties back to the business objectives.

(e) Will the system availability (uptime) be adequate in live operation?
- Yes, this ties back to the business objectives.

(f) Is the entire system tuned optimally?
- Yes and no. This ties back to a business objective, which is to be efficient. However, the word "optimally" is dangerous here and implies expectations that the performance testers probably cannot meet. It is difficult, perhaps impossible, to determine if a system is optimally tuned. At the risk of being nit-picking, this objective is better re-stated in these words: "Is the entire system tuned reasonably well, in terms of efficient use of resources and with reasonable flexibility and spare capacity to handle changes in demand?"

(g) Does the system degrade gracefully, or fail catastrophically, when it is pushed up to, and beyond, its planned maximum capacity?
- Yes, this ties back to the business objectives.

(h) Is the system scalable: can it be upgraded to accommodate the projected growth without re-writes or a major re-structuring?
- Yes, this ties back to the business objectives.

(1.5) Are all the stated testing objectives valid and within the scope of the performance testing project?

This question has already been answered under questions (1.2) and (1.3) above.

(1.6) Are the stated testing objectives specific enough to be measurable, and is it feasible to measure them? If not, what do you think we should do about it?

At this time, the objectives are not tangible enough to be measured and evaluated. That's OK -- for the moment -- the background reading provides an initial statement of the situation, not a detailed one. Nevertheless, it is important to have an explicit and early

task in the project work plan to evolve the objectives and make them tangible.

Each testing objective needs to become more specific before the project can progress very far. Questions and issues to be resolved fairly quickly are listed below for each testing objective:

(a) Are response times satisfactory when the system is operating under a realistic load?

- We need to quantify the response time goals – what do we mean by "satisfactory"?
- We need to quantify the test load, the expected live operational load, and the relationship between them.
- We need to specify the test infrastructure or environment, and its comparability to the live operational environment which is described in Appendix B.
- Response time goals have already been specified in service level agreements (SLAs) which have been negotiated between the IS group and the business units. These service level agreements are described in the Description of the Situation, Part B. You do not need to review Part B in order to answer the questions in Exercises 1 and 2.
- The guideline for response time consistency has been set as plus or minus 25% of the average. That is, if the average response time for a transaction is expected to be 2 seconds, then 90% or more of the measured response times should fall within the range of 1.5 and 2.5 seconds.

(b) Does the system operate correctly when accessed simultaneously by multiple users?

Problems such as features interfering with each other, database optimization, resource contention and transaction priorities need to be considered here.
We need to define the term "correctly" – what suite of feature test cases will be used to check correctness, and what percentage pass rate for that suite is considered acceptable?
We need to quantify "multiple", specify the interactions and possible interferences which we will test, and define what each virtual user will be doing (as a series of test cases coordinated across the multiple users).

(c) How well does the system handle heavy loads?

- We cannot assume that the performance degradation from adding additional users or performing extra work is linear: a significant increase in response time may occur when only a few users are added or the workload increases by an increment.
- We need to quantify the heavy loads.
- We need to specify the system's acceptable behavior under heavy loads. If the system is expected to degrade gracefully, what level of service is acceptable at each increasing level of load? For example, what increase in response time is

acceptable?

(d) Will performance and reliability levels be maintained over an extended period of use?

Insidious problems such as memory leaks will not reveal themselves in short run tests or by testing with a small number of users. Such problems usually lead to performance degradation and eventually to system failures.

We need to specify the test duration and the (varying) load during the test.

We need to quantify reliability in terms of mtbf or some other relevant, measurable entity.

Reliability goals or targets need to be established.

(e) Will the system availability (uptime) be adequate in live operation?

Apart from the planned downtime for maintenance, is there reasonable confidence that the unplanned downtime will be held to acceptable levels?

We need to specify the acceptable level of availability in live operation, the level in the test lab and their relationship.

(f) Is the entire system tuned optimally?

- The real objective here is to ensure the system and its support resources are "rightsized", that is, the deployed resources are utilized efficiently with reasonable reserve capacity but also with minimal waste.
- Since the test lab may not exactly match the live environment, the optimal tuning in one may not be best for the other.
- We will never know if the system is tuned optimally, even if the test lab is a mirror image of the live environment. We can only confirm that the tuning is adequate, in the sense that performance and reliability appear to be acceptable for the more common or more likely ways the system will be configured.
- Successful tuning requires the empirical comparison of various hardware (server and client), support software, database and network configurations, under carefully controlled test workloads.
- As an example of the questions that tuning may address: are upgrades from single-processor servers to dual-processor or quad-processor servers required to achieve the performance goals?
- On the other hand, a well-tuned -- even if not an optimally tuned -- system may allow the book club to increase the number of users without costly hardware upgrades. Experience has shown us that very few systems are initially tuned for high or even adequate performance.

(g) Does the system degrade gracefully, or fail catastrophically, when it is pushed up to, and beyond, it's planned maximum capacity?

Suggested Answers to Part 1 of the Case Study

- This system can fail in an indefinitely large number of ways (yes, that's infinite). We need to define the universe of ways in which the system can fail, the failure modes we will specifically test, and what the system's expected behavior is – i.e., specifically how it will fail in each test scenario, or how it will degrade gracefully.
- The planned maximum capacity must be specified.

(h) Is the system scalable: can it be upgraded in the future to accommodate the projected growth over the next 3 to 5 years, without major software re-writes or a major re-structuring or conversion of the database?
- Should this objective include the identification of bottlenecks and recommendations on how to alleviate them?

We also recommend that three new test objectives be added in these areas:

(i) Design reviews.
- Testers should be involved early and immersed in the system design reviews.

(j) Impact of database conversion.
- Testers need to evaluate the consequences for the testing project.

(k) Adequacy of the basic design – architecture of both the application software and its supporting infrastructure.

(1.7) What do we want to measure, in order to evaluate the performance of this system?

In order to evaluate the performance of this system, the performance test team needs to measure four items:

(a) Response time of the major transactions.

(b) Throughput of the major transactions.

(c) Availability of the system while it is under load.

(d) Resource utilization of the major components of the system, while it is under load.

The above list is only an outline -- the specific details (e.g., the nodes in the network at which the response times will be measured) will be added later.

Suggested Answers to Part 1 of the Case Study

The team, to the degree possible, will also attempt to identify and isolate bottlenecks in the system.

Answers to Exercise 3: Calculating the Work Loads

(3.1) How many web site visits or sessions are expected per month?

100,000 book club members (regular users) each have 2 visits or sessions a month = 200,000 sessions. Casual visitors add another 15,000 sessions monthly, for a total of 215,000 sessions per month.

(3.2) How many web site visits or sessions are expected in a typical hour?

215,000 sessions divided by 30 days = 7,167 daily sessions on average. 7,167 daily sessions divided by 17 hours a day = 422 hourly sessions on average. Let's say 450 to be conservative.

(3.3) How many web site visits or sessions are expected in a peak-demand hour?

7,167 daily sessions multiplied by 20% = 1,434 sessions on average during the peak hour of a typical day. Let's say 1,500 to be conservative.

(3.4) What is the expected peak number of concurrent visitors in a typical week?

1,500 peak-hour sessions divided by 4 (as each session lasts 15 minutes) = 375 overlapping sessions or concurrent users.

(3.5) What is the expected peak number of concurrent visitors in a typical month?

Include the safety factors of 4 and 8, and we have expected peaks of 1,500 (4 x 375) concurrent sessions in a typical week and 3,000 (8 x 375) in a typical month.

(3.6) How many hits will happen to the home page in a typical hour?

The number of hits to the home page in a typical hour is 450 (one per session).

(3.7) How many hits will happen to the home page in the peak hour in a typical month?

The number of hits to the home page in the peak hour in a typical month is 3,600 (8 times more than the average hour).

Suggested Answers to Part 1 of the Case Study

(3.8) What other significant assumptions besides the ones listed below, if any, have you made in this exercise?

We will leave this question open for debate.

Answers to Exercise 4: Outlining the Test Environment

(4.1) Where will the measurements be taken?

(a) Externally

These performance characteristics (response time, throughput, availability and resource utilization), will be measured end-to-end, at several different external points of access to the system.

The locations of these end points for measurement will be selected to provide a representative set of readings of the response times, throughputs, etc., for each major constituency mentioned earlier: the senior managers, the customer service group, the catalog publishing group, the warehouse distribution group, and the IS group.

Users at the remote satellite office, who are connected by the WAN, are likely to experience slower response time than users directly connected to the main headquarters LAN. So the response times and throughput should be measured at the satellite office as distinct from the main office.

The performance characteristics will be measured at these end points under several different loads, as described later in the answer to the next question.

(b) Internally.

It is unlikely that the project objectives can be satisfied only with external measurements. In addition, internal measurements will be needed to assess resource utilization and try to identify bottlenecks.

At a minimum, the system performance will be measured internally in these areas (which are usually fruitful areas to investigate bottlenecks):

- Processing time spent within the database server, for example, to service a query, as opposed to the transportation time within the network, back and forward from the local client to the database server.

Suggested Answers to Part 1 of the Case Study

- o Network utilization, measured as a percentage of the theoretical full bandwidth of selected major network links, under the test loads.

- o Load balancing, for example, the division of the work load between the application servers. If the load balancing software is not installed and working correctly, this measurement will reveal it.

- o Response times inside and outside the firewall. Firewall are often bottlenecks, and this measurement will show whether the problem is trivial or not.

- o Response times and throughput during degraded modes of operation.

(c) Exploratory data gathering.

Initially, the behavior of the system is not likely to be known well enough to decide specifically what measurements to collect, and where. The internal measurements proposed above (database vs. network, load balancing, and inside/outside the firewalls) are based on hunches. Hunches are a useful way to get started, but sometimes can be off base.

For this reason, the performance test team initially will conduct exploratory testing, and collect measurements from a wide number of points in the system. After the initial data has been collected, the performance test team will have a better sense of which measurements are material and where to focus their attention.

Measuring resource utilization and identifying bottlenecks in the system requires that probes be placed into the system during testing. These probes will be inserted into a representative sets of clients and servers, and will gather information such as processor utilization, memory utilization, and lengths of queues.

(4.2) What load(s) should we place on this system, while measuring its performance, such as an average load, a peak load or an overload?

Since system performance can vary dramatically with the load placed on the system, several different loads need to be utilized during performance measurement.

The test loads (benchmarks) used during performance measurement should include the following mixes of demands:

Suggested Answers to Part 1 of the Case Study

(a) typical daily business workloads;

(b) daily periods of peak demand;

(c) weekly and monthly periods of peak demand;

(d) specialized demands, such as
(i) the occasional need to print or electronically distribute high volumes of catalogs or other documents, or
(ii) the periodic need to back up the database;

(e) senior managers' use of the system, such as computations of "what if?" queries during to aid decision making during key business meetings; and

(f) the likely future growth in demand over the next 3 years and the next 5 years.

The operational profile which was documented in Part 1 will be used to determine what the specific mix of demands will be in each of these test loads. We will use the 90%-10% rule: the most popular 10% of the transactions account for about 90% the work load, so the other 90% of the transactions can safely be ignored when the test loads are put together. Each of these loads also should include appropriate levels of simulated background noise.

(4.3) What types of load testing should we utilize in our strategy, such as spike and bounce, hot spot or duration testing?

Several techniques hold the promise of delivering useful information, and since the feasibility and relative ease of using these techniques have not yet been assessed, it is too early to choose or eliminate techniques. Any of these techniques may be used (pending further evaluation):

A. Violations of Pre-Conditions (i.e., Invalid Input Data)
A.1 Negative testing
A.2 Boundary value testing
A.3 De-stabilization

B. Heavy Loads
B.1 Load testing
B.2 Limit testing
B.3 Stress testing
B.4 Hot spot testing

Suggested Answers to Part 1 of the Case Study

C. Probing for System Limits
C.1 Bottleneck identification
C.2 Duration or endurance testing
C.3 Accelerated life testing
C.4 Spike and bounce testing
C.5 Breakpoint testing
C.6 Extreme configuration testing
C.7 Dirty configuration testing

D. Interactions
D.1 Rendezvous testing
D.2 Feature interaction / interference testing
D.3 Interoperability and interface testing
D.4 Deadlock testing
D.5 Synchronization testing

E. Human Errors
E.1 Bad day testing
E.2 Soap opera testing

F. Catastrophes
F.1 Disaster recovery testing

G. Physical Failures
G.1 Environmental testing

H. Before-and-After Comparisons
H.1 Volume testing
H.2 Parallel testing

I. Handling Changes
I.1 Live change testing
I.2 Extreme configuration testing
I.3 Modification testing
I.4 Regression testing

J. Handling Errors
J.1 Error detection & recovery testing
J.2 Degraded mode of operation testing
J.3 Software fault injection

Suggested Answers to Part 1 of the Case Study

Other types of testing which are associated with robustness testing, and which we will consider using, are:

Scalability testing
Compatibility and configuration testing
Interoperability testing
Risk-based testing
Hazard or threat identification
Failure modes effects and assessment (FMEA)

(4.4) How do we generate and drive these test loads?

The loads will be generated and entered into the system:

(a) manually,

(b) by using the existing copies of WinRunner, and

(c) by using load testing tools such as LoadRunner for both client/server traffic and for Web traffic.

These load testing tools will simulate many simultaneously active users on the network.

The copies of WinRunner will still be used, even though the load testing tools will contribute the bulk of the test work load. WinRunner will be used to check feature correctness by running the feature-level test cases during the performance testing.

(4.5) What environment (test equipment and facilities) do we need for the performance testing?

Ideally, the full operational environment should be made available for performance testing. Additional equipment may be needed in the test environment also, such as additional servers connected to the network to host the load testing tools.

A test database needs to be developed, which is roughly equal in size to the live production database, and containing a similar mix of data to the live production database.

Providing the full operational environment and full database is preferred for testing, but is expensive and time consuming and may not be feasible. The alternatives available, if it is not feasible to fully mirror the live environment in the test lab, have not been addressed as yet. (This is an open issue.)

Suggested Answers to Part 1 of the Case Study

(4.6) What kinds of tools are needed for this performance testing, if any?

The tools needed for this performance test project are as follows: (a) the feature testers' copies of WinRunner, (b) load testing tools, to simulate both internal client/server users and external Web site visitors, (c) profiler(s), which place probes into the system and monitor resource consumption within the system, resident on both the client-side and the server-side, (d) a background noise generator, (e) configuration manager(s), loaded on both the servers and the clients, and (f) monitoring and diagnostic tools, such as a network sniffer.

(4.7) What skills are required for this performance test? What kinds of people should be on the performance test team?

The performance test team should include:

- Testing and performance measurement professionals, who are experienced in assessing and predicting system performance.
- Business professionals, who understand the user community and know how the system is going to be used.
- Technical specialists who are involved with the system, such as the database administrator and the network administrator.
- Vendor technical support people should be on call, to resolve any issues specific to one vendor's equipment or software.
- Tool experts are needed, for the load testing and other tools used in this project.

(4.8) How should the data (the test results) be analyzed and interpreted?

The section mentioned before, entitled: "Performance Testing Objectives", in Part 1 of this case study, provides the list of questions which the managers want to see addressed and hopefully resolved by the conclusion of this performance testing project.

After the initial exploratory testing, the performance test team will "reverse engineer" and work backwards from each of the questions on this list, and determine what data needs to be collected and analyzed, in order to develop an informed conclusion on each of these questions.

Suggested Answers to Part 1 of the Case Study

(4.9) What are the likely difficulties involved with this performance test project? What are the risks, if any, that the performance test will not deliver trustworthy or usable results?

There are many risks that the performance testing project will not be successful. Anticipating these risks is smart, rather than "sticking our heads in the sand", because contingency plans can be developed for the risks which have been identified.

Some of the major risks of the performance testing project are:

- The performance testing time is cut short because of deadline pressures, if the software developers, the equipment suppliers or the feature testers are late.
- The performance testing is done so late in the development cycle that the design is locked in concrete, and little can be done to improve performance if needed.
- There may be intense last-minute pressures to compromise and install the system with its existing but inadequate performance. (Some people would argue that this is not a test project risk, since the testing project delivers the right information about the system's performance, but an overall system project risk.)
- The operational profile used in testing is not representative of the real world. For example, if the peak and stress loads used in testing are too light, there may be a misleading impression of the system's robustness.
- The performance test environment does not adequately mirror the real world.
- The tools provide problems -- either the budget does not allow the needed tools to be used, the tools have long and difficult learning curves, or the tools in themselves are buggy.
- There is insufficient involvement and support from the user community (the internal business groups). These people may be already over-committed, doing double duty by continuing the on-going day-by-day business operations, and also getting ready for the new system, and may have no energy left for participating in the performance test.

(4.10) What additional information, if any, do you need to know before you can develop an adequate performance test plan?

Suggested Answers to Part 1 of the Case Study

Additional information which will be helpful to the performance test project includes:

- The performance specifications, if any have been defined, or at least the performance expectations, for the system. (These could come from the senior managers, the business groups, competitive analysis by marketing, or the contractual obligations spelled out in the contract with the system developers.)

  Any hunches from the system developers and equipment suppliers about likely resource utilization where bottlenecks maybe lurking. (Caution -- these hunches could be dead wrong.)

- A copy of the functional specs. for the book order processing system.

- A copy of the specific topology and design of the system.

- An inventory of all the vendors, products and version numbers of products used in the infrastructure, together with contact names within the vendors.

- A copy of the contract(s) with the system developers and equipment suppliers, to help determine what assistance they are contractually obligated to provide, if any, in the performance tuning and measurement efforts.

- Information about the measurement capabilities which are built into the existing software, such as Windows XP and Oracle 9i, and which might be utilized in performance measurement.

- A better sense of the managers' tolerance for risk and willingness to spend time and money to mitigate the risk, including the project budget and schedule numbers which are likely to be acceptable.

(4.11) What assumptions have been made?

The main assumptions are (this is not a complete list):

- Measuring the system's performance is a justifiable and feasible project, in terms of the availability of the expertise to do the job, and management's willingness to fund the effort and allow sufficient time to do it.

- Other people who will be needed to support the performance testing, such as the system developers and users, will be available as needed and cooperative.

Suggested Answers to Part 1 of the Case Study

- The tools will not be an impediment to this project. We will be able to find one or more commercially available load testing tools which will generate suitable loads and capture the performance data we need, will be affordable, reasonably straightforward to learn, and not buggy. (Ho ho, that's a good one.)

- The scope of this project is limited to performance measurement, not performance tuning. Tuning requires a different set of skills than measurement, and usually requires specific and deep technical knowledge of the various components of the system.

- The performance measurement will be a one-time activity. There will not be an indefinite number of iterations of tuning the system in response to the reported performance, then re-measuring to see what the cause-and-effect impact of the tuning is on the system performance.

- The performance data which will be captured will actually be useful. It will be sufficient to develop clear and unambiguous answers to the book club managers' questions about the acceptability of the system.

(4.12) How would you respond to the suggestion that we skip the performance testing, and instead simply "beef up" the servers, clients and communications channels as and when these actions become necessary?

The risk of this approach is that sometimes throwing money at performance problems does not work. The bottlenecks may not be identified correctly, and new capacity is added to the system in areas where it does no good.

The main reason for performance measurement before the system goes live is to avoid possible business interruption. The testing will help assess and mitigate this risk.

The system is critical to the day-to-day operations of the book club. If, when the system goes into live operation, it cannot handle the work load, then the costs of business interruption could be severe.

(4.13) Is an initial impact assessment (IIA) needed for this project?

The first response of the testers to this question is no. The purpose of the IIA is to determine whether a performance test project is needed or not. In this situation, the need for performance testing is obvious so the IIA can be skipped – it will not tell the testers

anything extra beyond what they already know, which is that the risk of going into live operation without performance testing is unacceptable.

On the other hand, an argument can be made that performing an IIA is not such a bad idea in this case. Some of the managers are apparently skeptical, because the question has been raised about whether this project can be skipped. The IIA may persuade them of the necessity. In addition, the IIA could give the testers an additional level of understanding about the system fairly quickly, which will be useful in formulating the performance test strategy.