Cranfield UNIVERSITY

Performance testing as a full life cycle activity

Julian Harty

Cranfield UNIVERSITY

Scope of Performance

Performance

- What is performance testing?
- Various views
- 3 outcomes
- 3 evaluation techniques
- A few techniques
- 5 layers of abstraction
- Scalability & Capacity planning

Objectives of the open season

- Definition of Performance
- Definition of Performance Testing
- Agreement with techniques, any other test case design techniques
  - TURA
  - Traversal & Interaction
  - Scenarios and workflows
  - Capture Edit and Replay

Cranfield UNIVERSITY

What is Performance Testing?

All about:

- numbers and quantities
- resource constraints
- effects of interactions

Definition (ISO 9126) of Efficiency

The capability of the software to provide appropriate performance, relative to the amount of resources used, under stated conditions.

Cranfield UNIVERSITY

TURA

Performance techniques

- Starts by adding additional requirements and conditions to existing functional test cases
  - Time
  - Users
  - Resources
  - Accuracy
- Simple and effective
- X + Y = Z in time T, with U users, and R resources, to A accuracy

Traversal and Interactivity

Models typical behaviour of users:

- How they use the system
- Interaction between users

  e.g. buyers and sellers
- Frequency of use
- Variations in behaviour

Generally converted into test scripts for automated tools, although large teams can act out the roles directly.

Cranfield UNIVERSITY

Scenarios and Workloads

Load on a system is seldom uniform. The variations in load means the demand on the system will also vary.

Scenarios and Workloads (similar to operational profiles) are used to exercise the software in order to find out whether the software meets its performance requirements as the load varies.

Cranfield UNIVERSITY

Capture, Edit, Replay (CER)

Performance techniques

Capture, Edit, Replay (CER) testing is commonly used to determine how the software responds when subjected to traffic recorded from an existing system or environment.

Editing is often required e.g. to replace dynamic values, and may happen before the test is started, or dynamically while the captured information is being replayed

Very prone to generating erroneous results

Cranfield UNIVERSITY

Technical views of performance testing

- Isolated, single user performance
- Aggregated, multi-user performance
- Deviation, best, average, and worst case
- Scaling, capacity planning

- Queue management
- Concurrency issues
- Cost per service request

Performance Testing as a FLCA

- Analytical and Simulation Modelling
- Selection of:
  - Algorithms
  - Protocols
  - Architecture
- Performance testing and quality gates - throughout the V model
- Performance tuning and monitoring

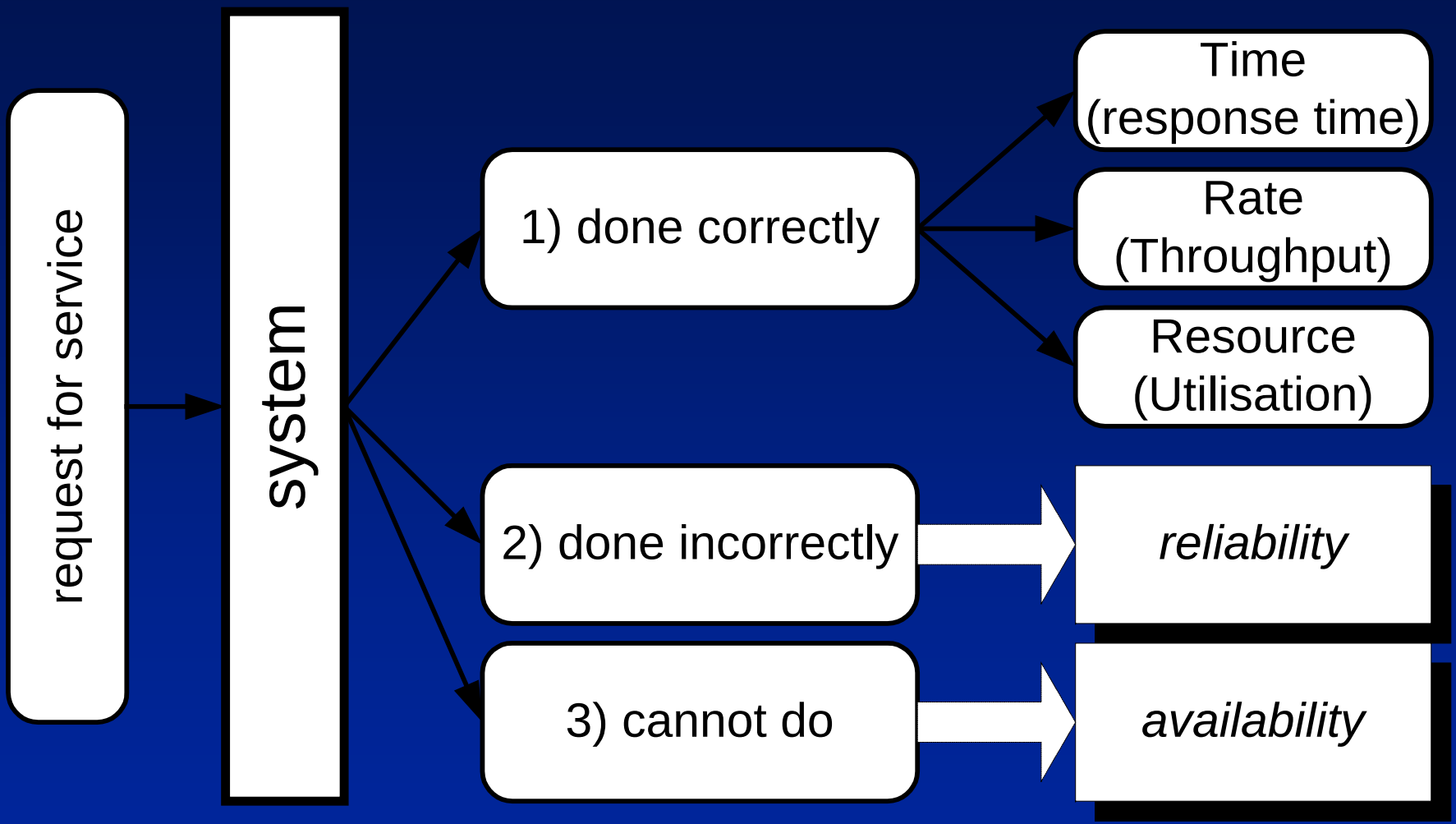
3 evaluation techniques

- Analytical modelling

  e.g. Queuing models

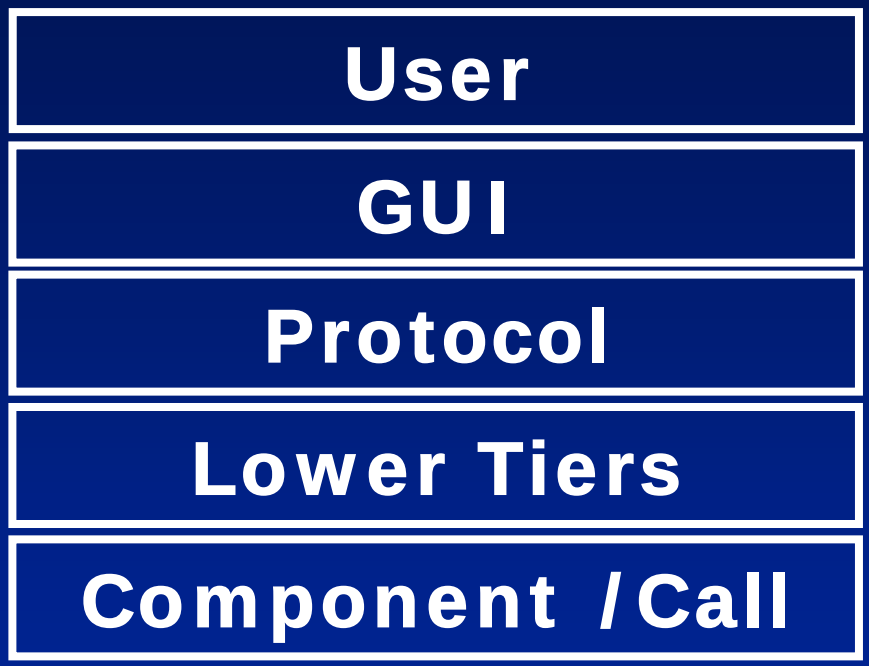
- Simulation
  - Allows the planned system to be simulated, generally on a computer
  - Can be driven by traces taken from an existing system
- Measurement
  - The software is exercised and the outcome is measured

3 possible outcomes



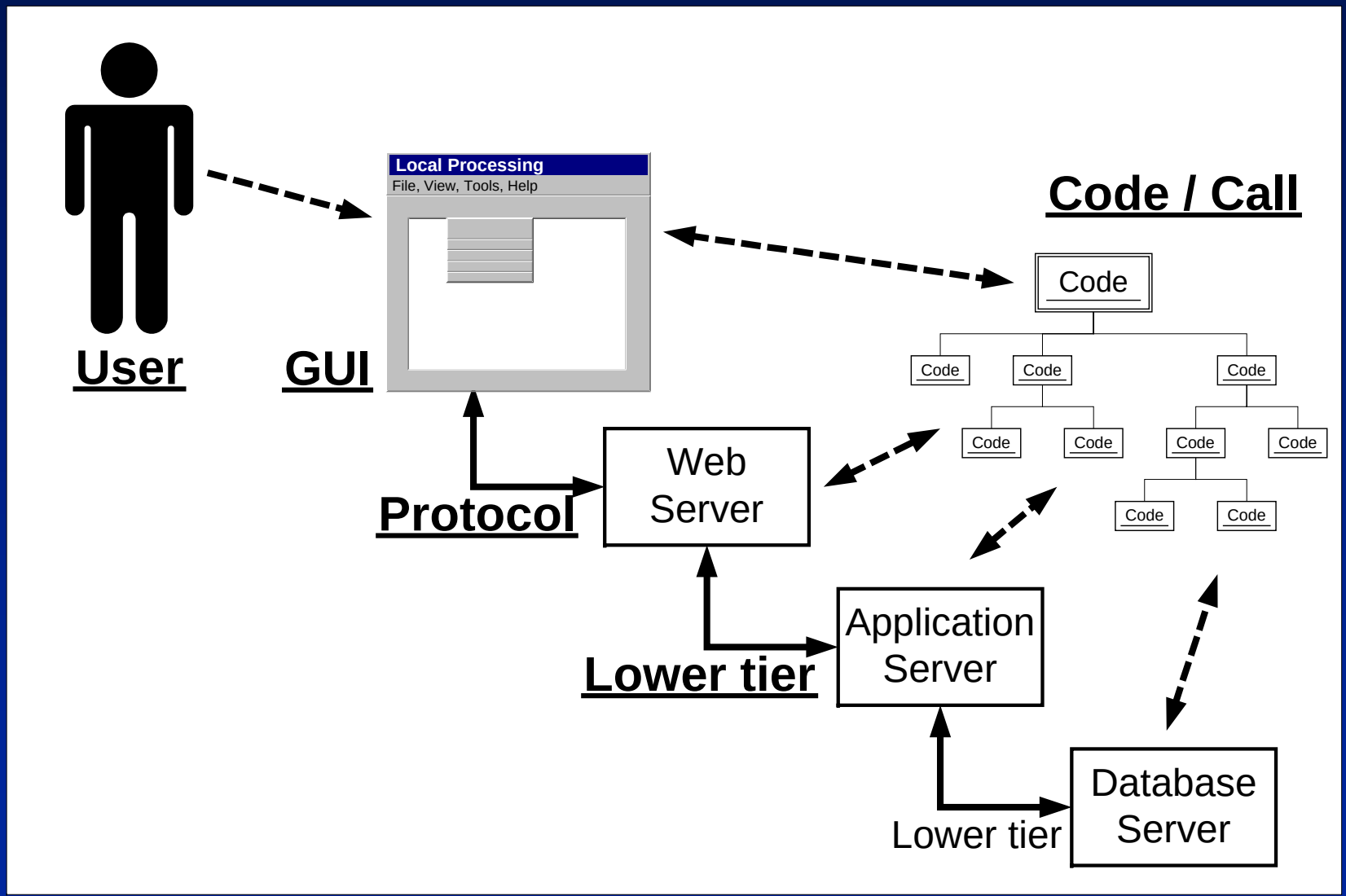
Based on Figure 3.1 page 33 of "The Art of Computer Systems Performance Analysis" Raj Jain

5 Layers of abstraction for performance testing (1)



5 layers of abstraction for performance testing

5 Layers of abstraction for performance testing (2)



Cranfield UNIVERSITY

Scalability

Scalability – the degree or extent to which the software can cope with more work.

- Measures how much a system can grow, particularly across multiple servers.
- Particularly necessary for Internet Applications and other applications where the volume of users, etc. is hard to predict.
- Can include changing database engine, etc.

1

10

50

100

1000

50000

Scale UP ↑ **bigger** computers – easier

Scale OUT ↔ more computers - cheaper

Cranfield UNIVERSITY

Capacity Planning & Scalability

Capacity planning – how to make best use of the current resources, and how to increase the capacity before the current resources are exhausted.

Capacity planning and scalability have similar goals:

To ensure the users get the performance they expect, given limited hardware resources and the current software.

Sometimes the software is the limiting factor, and would need to be modified to take advantage of any increased hardware resources.

Cranfield UNIVERSITY

How to improve performance

If the performance requirement isn't met

- Tuning and optimisation
- Change the hardware
- Select more suitable algorithms
- Recode hotspots in more efficient programming language
- Change the architecture

- NB: Use testing to identify, pinpoint and isolate the occurrences and underlying causes of bottlenecks

- *If all else fails:*
  - *Change the requirement*
  - *Leave the project…*

Performance References

- The art of computer systems performance analysis

  Raj Jain,

  ISBN 0-471-50336-3, Wiley, © 1991
- Capacity Planning for web performance

  Daniel A. Menascé and Virgilio A. F. Almeida

  ISBN 0-13-693822-1, Prentice Hall, © 1998
- Gain econfidence & load testing for eConfidence

  Privately published by Segue.com
- The Web Testing Handbook

  Steve Splaine and Stefal P. Jaskiel

  ISBN 0-9704363-0-0, STQE Publishing, © 2001