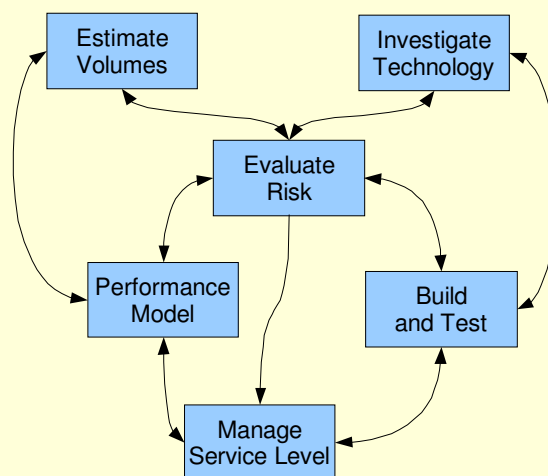# Managing Performance Problems

David Howard

Agenda

- What is the problem?
- Overall process
- Problem statement
- Performance modelling
- Evidence collection
- Resolution
- Avoiding performance issues

What is the problem?

- What are the symptoms?
  - Often user complaints
- What is the technical issue?
  - Hardware, Software etc…
- What is the process issue?
  - The real reason for production performance complaints
  - All newly developed systems are likely to have performance problems – just like bugs.

Overall Process

Estimate Volumes

Investigate Technology

Evaluate Risk

Performance Model

Build and Test

Manage Service Level

Process – Volumes…

- How many users?
- How many function per day per user?
- Usage variation per Hour, Day, Month, and Year
- Data growth profile?

Process - Technology

- Is it scalable?
  - Scalable means "Add more hardware and get better performance per user"
- Can it deal with the volume?
  - Order of algorithms
- Known bottlenecks?
  - Bottleneck means "Maxed out resource, or fundamental scalability limit"

Process – Risk…

- What cost is there to a slow system?
- What would a total failure cost?
- How likely is this in a defined time limit?
- If there is an issue, how long will it take to fix and what is the cost of this?
  - e.g. Time to add a new server.
- This defines how much effort needs to be put in to performance management.

Process – Build and <u>Test</u>

- What level of performance proof is justified?
  - Volume, Soak and Load testing
  - Optimisation
  - Scalability
  - Failover and fall-back under load
- How can this be proven?
- What is appropriate?

Process – Manage Service Level

- What load can the system cope with? (Average and Peak)
- What data volume?
- For how long?
- What is the likely uptime in these conditions?
- What happens when there is a failure?
- What response times are acceptable?

Problem statement

- What are the known symptoms?
- Does an SLA exist?
- Is there a mismatch between SLA and expectation?
- Do you know what users are experiencing?
- Do you know what users are doing?

Performance Modelling

- Use to work out likely response profile
- Can answer "Could this be caused by…"
- Beware: "Garbage in, Garbage out"
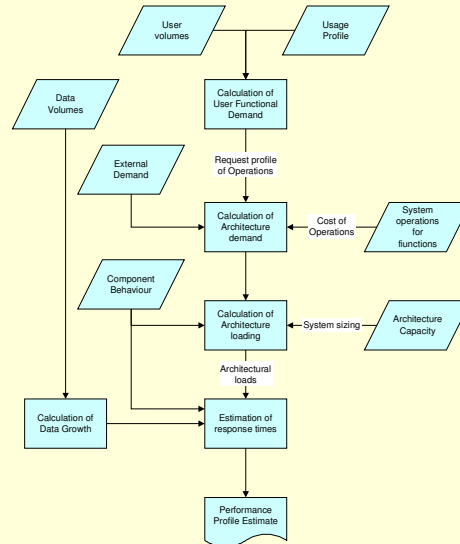
Evidence Collection

- Use scientific principles
  - … but backed up by common sense
- Where can real and decisive information be gathered from?
- What is actually going on?
  - Think whether a piece of evidence is a cause or a symptom.
  - What is the chance an effect is a "root cause"?

Evidence - Tools

- Performance monitoring
  - Standard: Perfmon, SQL execution plans…
  - Active: Record / Playback
- Code profiling
  - Which code is using up how much time?
  - CPU time or elapsed time?
- Tools can be very expensive, or quite cheap. Which is more cost-effective isn't obvious….

Resolution

- Objective: Improve end user experience.
- Likely to be multiple causes
- Hardware spend can help
  - …. but only temporarily
  - Tactical changes buy time to fix big issues
- Try not to over-sell a single solution
- If possible use:
  - A single and highly skilled team
  - Access to a production like environment

Avoiding performance issues

- Define and agree the expectation
- Early testing
  - Performance prototype
- Testing: Volume, Load and Soak
  - Prove scalability
  - Prove long term suitability
- Plan to take time over this
  - Fit through the project plan <u>not</u> after UAT

Sarquol… A quick plug

- Free Resources at sarquol.com:
  - Principles of Capacity Management
  - Coming soon: Generic Performance Model
- Irregular Bulletin (Approximately monthly):
  bulletin@sarquol.com
- E-mail: dh@sarquol.com
- Call: 07887 536083

Any Questions?