



Performax provides software load testing and performance engineering services to help our clients build, market, and deploy highly scalable applications.

Bernie Velivis
President, Performax Inc

Load Testing

*Load testing is an essential component of quality assurance. It can identify problems with an application's efficiency, stability, or scalability **before** it's released.*

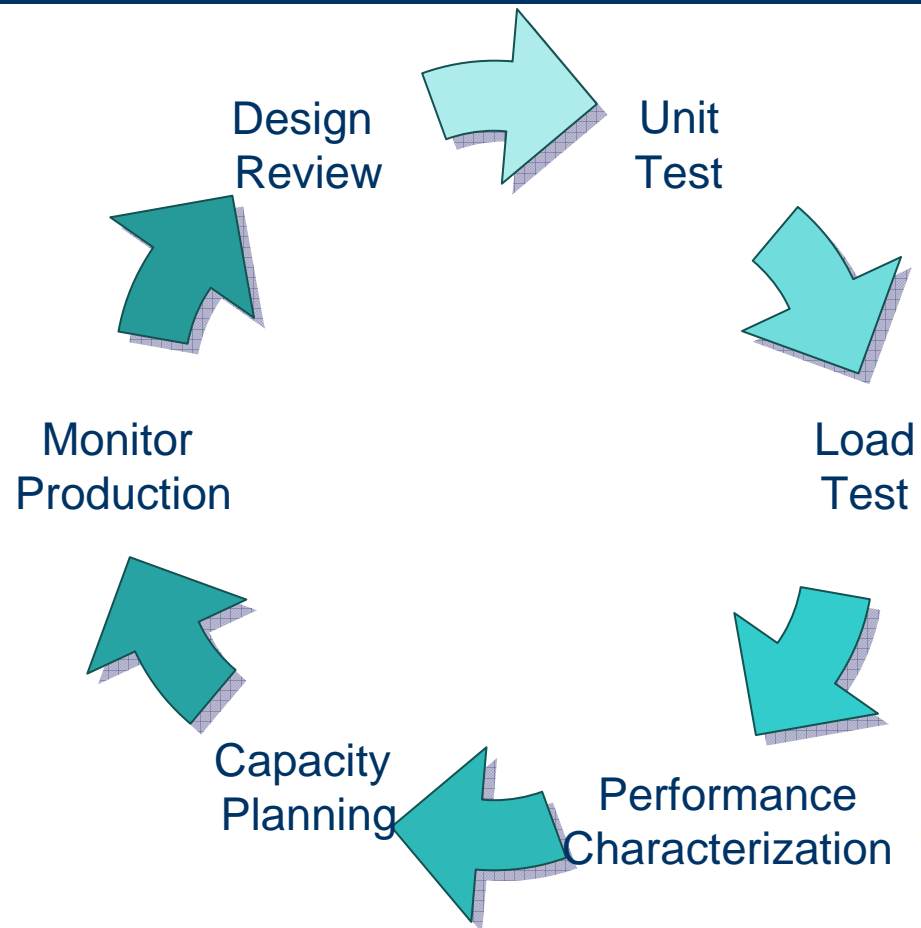
The cost of fixing performance problems after an application enters production is high. The cost of restoring reputations and confidence is higher yet.

Performance Characterization

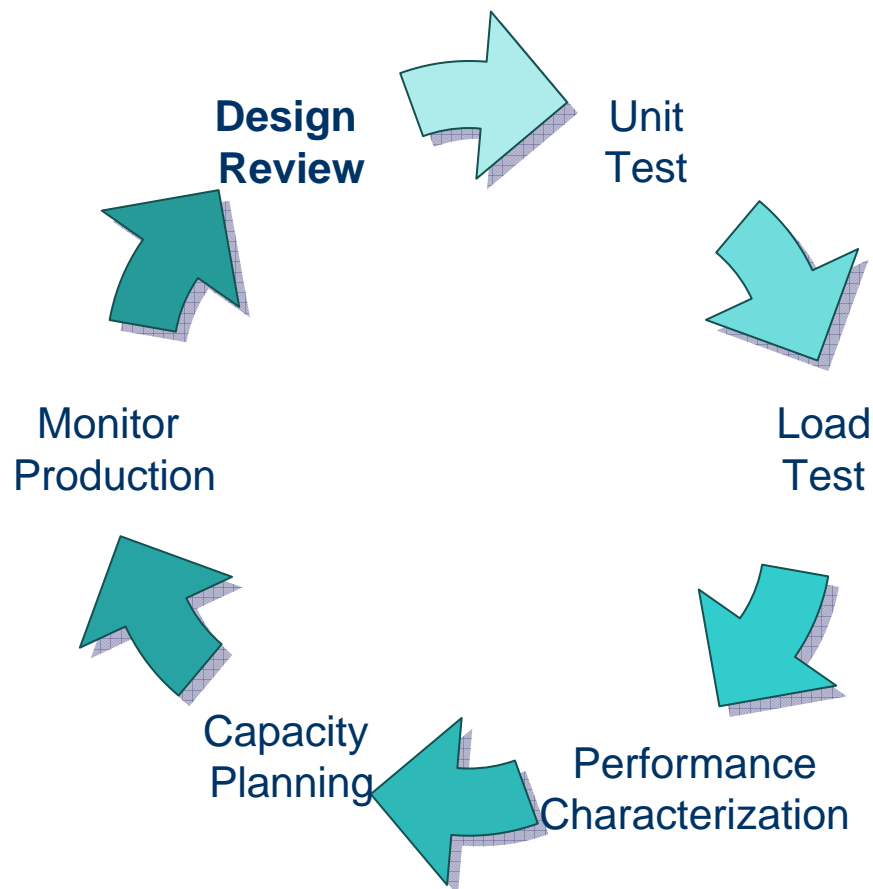
There is a tremendous value in understanding when a company is going to have to spend money on their IT infrastructure.

Performance characterization projects determine what spending is required on the infrastructure and when it will be needed to meet growing capacity demands.

Performance Engineering Methodology



Design review and rapid prototyping



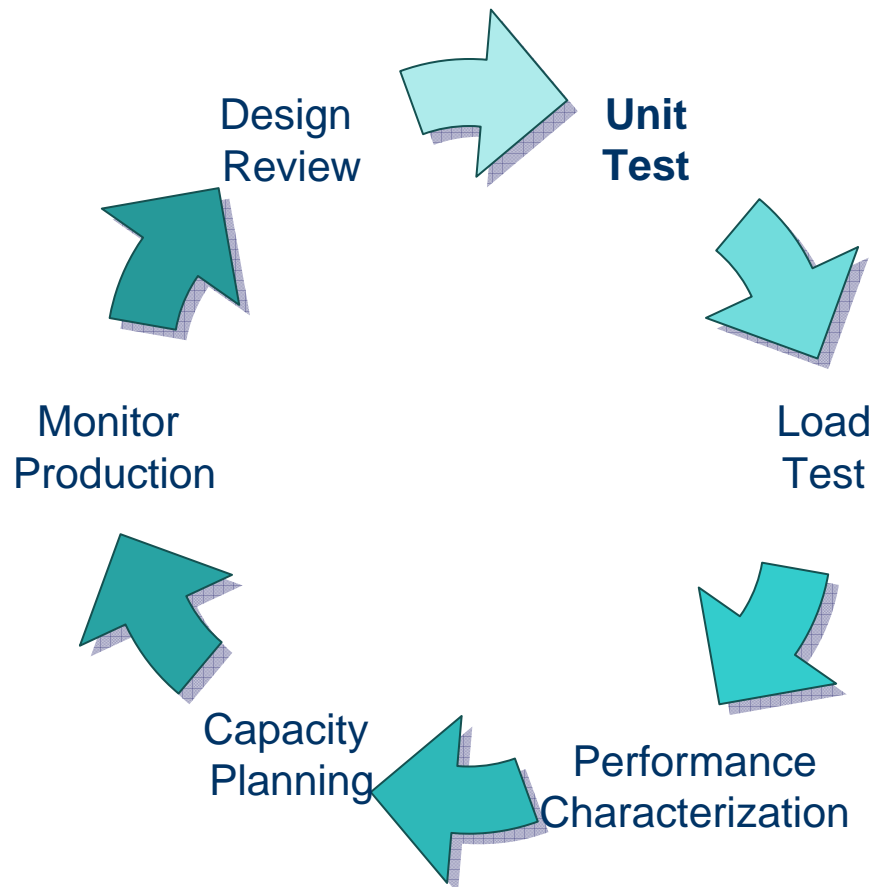
Design Review

(pay \$ now or \$\$\$ later)

- Scalability and availability requirements
- Service level agreements
- Architecture review
- Response time budget by logical tier
- Agile prototyping & evaluation



Unit test code during development

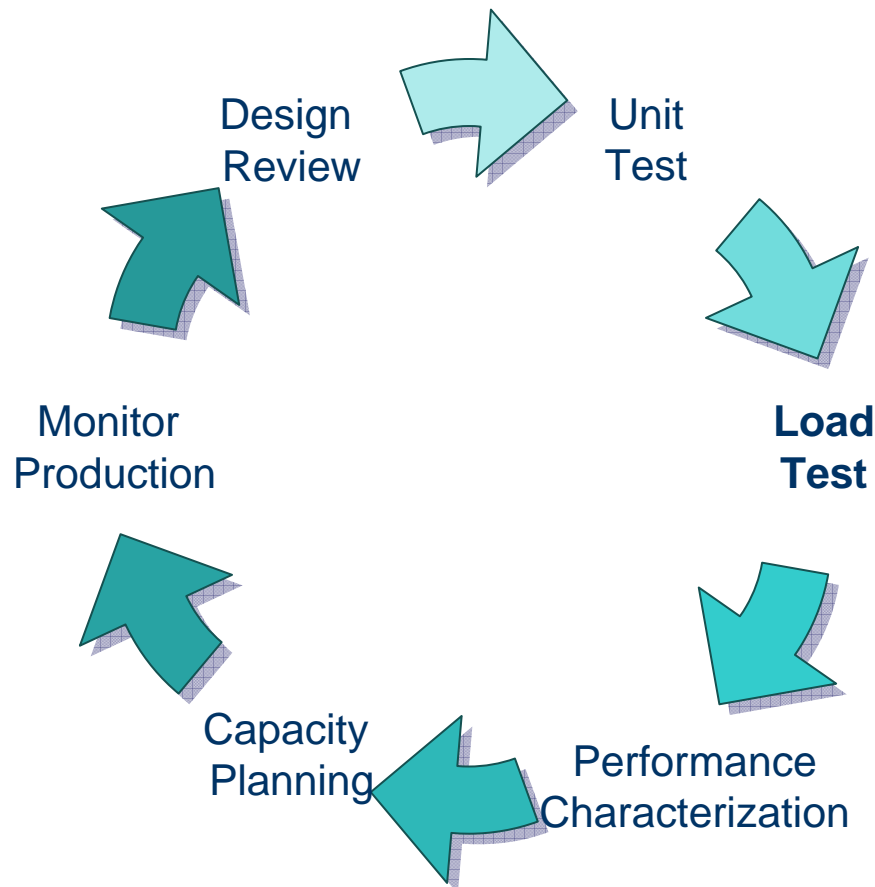


Unit Test

- Profile long running, frequently executed, or critical transactions across distributed services
- Optimize to reduce hardware demand and increase concurrency
- Compare results against resource budget
- Regression analysis of single user performance



Load test multiple users and a mix of transactions (system test)

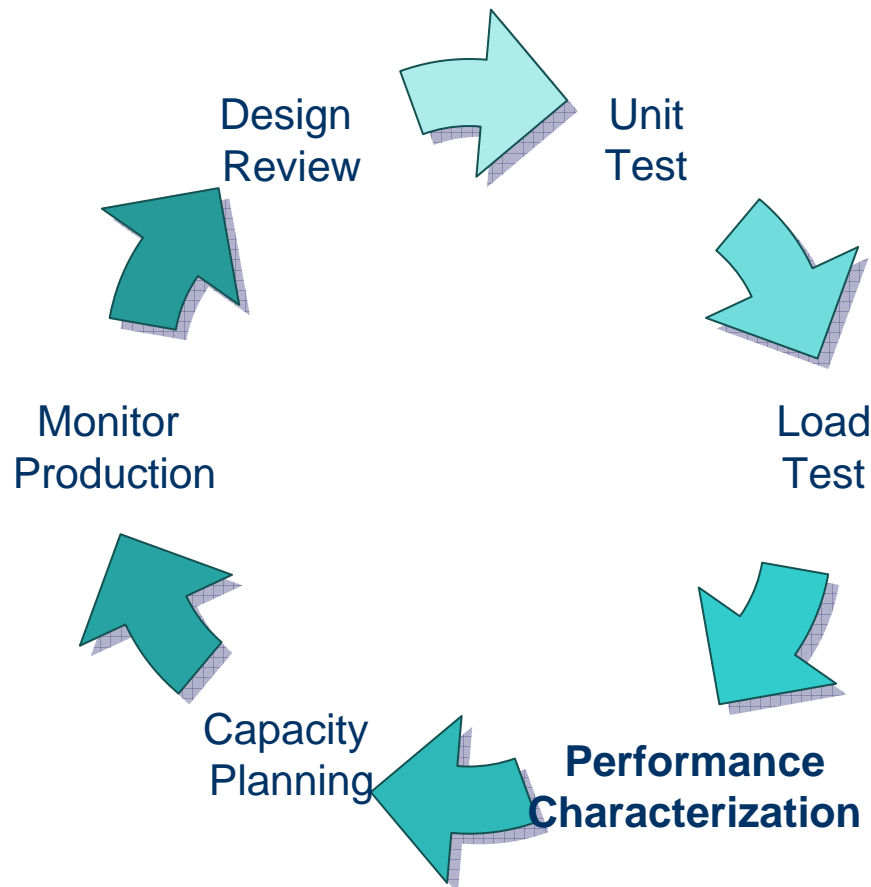


Load Test

- Define usage scenarios (online/batch/maintenance)
- Measure response time and throughput versus load on reference hardware
- Monitor and tune System performance
- Multi-user performance regression analysis
- Stability under sustained load
- Failover characteristics under load



Characterize performance versus load, hardware, data, and workload



Performance Characterization

- Test different usage scenarios to determine impact on configuration
- Characterize capacity vs. the speed and number of CPUs by tier
- Heuristics for configuring network, memory, and disk bandwidth
- Spreadsheet model showing server capacity by physical tier
- Determine scalability limit of the architecture
- Document modeling methodology so that models can be extended



Recap Goals of Performance Testing

Unit testing

- Profile long running, frequently executed, or critical transactions across distributed services
- Optimize to reduce hardware demand and increase concurrency
- Compare results against resource budget
- Regression analysis of transaction performance

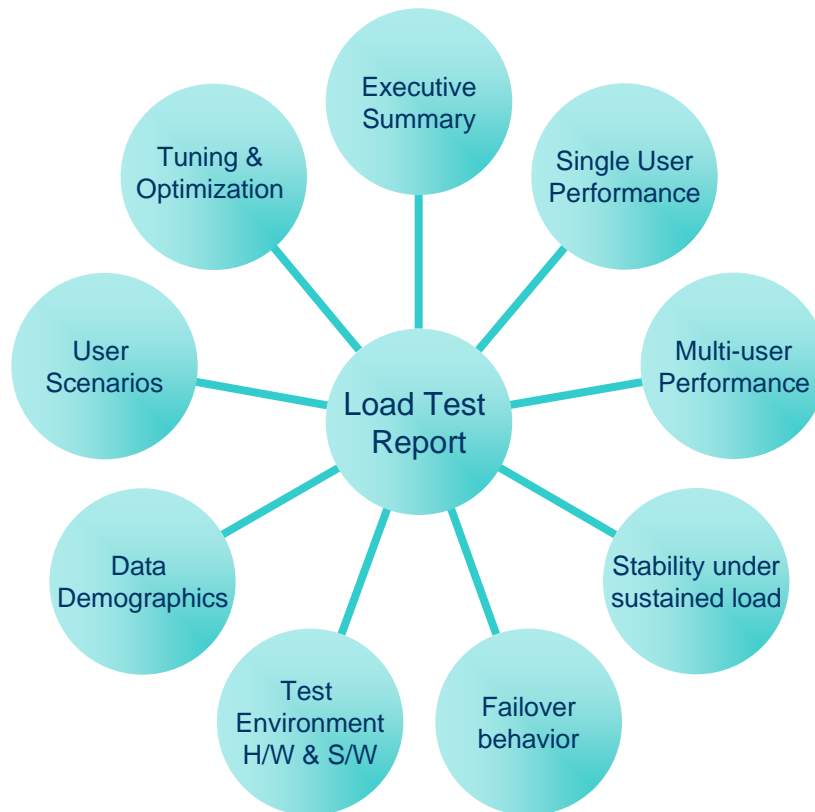
Load testing

- Find and fix obvious system bottlenecks
- Regression analysis of system performance
- Measure application stability under sustained load.
- Determine capacity limits of a specific hardware configuration
- Determine failover characteristics

Performance Characterization

- Determine hardware required to meet service objectives at various loads
- Decide when hardware needs to be added to stay ahead of demand
- Find the limits of the architecture's scalability
- Recommendations for enhancing scalability if necessary

Deliverables



Load Test

Load test report
Multi-user scripts
Test data repository

Performance Characterization

Modeling methodology
Capacity vs. server CPU configuration
Network bandwidth requirements
Disk and memory considerations
Sensitivity to data demographics
Sensitivity to workload



Client Responsibilities

Identify and describe top user scenarios

Supply synthetic data to support load testing

Provide dedicated hardware for testing

Provide load test tool licenses
(none required with OpenSTA)

Optional Training

Tools & methodology training

- OpenSTA basics (1 day)
- OpenSTA Advanced topics (1 day)
- Performance testing workshop (1 day)

Contact Information



Bernard Velivis
President

☎ 603 860 7900
✉ Bernie.Velivis@iPerformax.com
💻 www.iPerformax.com