Simulating storage system performance: a useful approach for SuperB?

Moreno Marzolla
Dipartimento di Scienze dell'Informazione
Università di Bologna
marzolla@cs.unibo.it

http://www.moreno.marzolla.name/

SuperB Computing R&D Workshop, feb 9—12 2010, Ferrara, Italy
Introduction

- The term *capacity planning* refers to the activity of estimating the right amount of resources required to meet future service demands.
Introduction

• The performance and scalability of a new system can be estimated
  • by measuring an existing system
  • by building and analyzing an appropriate model
    – Analytical model
    – Simulation model
    – Combination of the two above
  • You can use a running system as a source of “reliable” parameter estimates to be used as inputs for an analytical model
Measuring existing systems

• PROs
  • You definitely get the most realistic parameter estimates (response times, throughput...)

• CONs
  • There could be no system to measure!
  • Need to modify the system to insert “probes”
  • It is difficult to forecast how the workload varies in the future (the “new” system might be used differently from a similar, “old” one)
Analytical models

• Develop a mathematical model of the system using an appropriate notation:
  • E.g, Queueing Network, Markov Chain, Petri Nets...

• PROs
  • Models can be evaluated quickly and efficiently

• CONs
  • Analytical models require significant simplifying assumptions in order to be tractable
  • The performance modeler must be acquainted with the specific modeling notation (QN, MC, PN...)
Simulation models

• Write a simulation program which behaves like the "real system"

• PROs
  • Simulation is very powerful

• CONs
  • Writing a simulation program requires a lot of effort
  • Even moderately complex simulations could require a long running time to produce accurate results
  • Care must be taken in analyzing the simulation results (initialization bias, confidence intervals)
Performance modelling study (simplified)

- Statement of objectives; examine design and evaluation criteria; estimate manpower, cost and time for the performance study
- Input data analysis, identify probability distributions; collect data for validation; start with a simple model
- Implement the model (write the simulation program, or implement equations with some numerical evaluation tool)
- Check adequacy of probability distributions and simplifying assumptions; involve people familiar with the system;
- Evaluate the model; run simulation
- Test sensitivity of output to changes in input parameters; compare output with a current system if available
- Document and present the results; compute confidence intervals, compare alternatives

Formulate the problem and define the goals

Collect input data and define the model

Implement the model

Verified?

Evaluate the model

Test sensitivity of output to changes in input parameters; compare output with a current system if available

Analyze and present output data

Valid?

no

Document and present the results; compute confidence intervals, compare alternatives

no
Some common mistakes
(so we can try to avoid them)
Common mistakes

No goal

- **Before** starting a performance study, a clear goal must be stated
  - Ok: “*Compare the throughput of system A and system B with respect to workload W*”
  - Ok: “*Which caching policy among A, B and C provides the higher hit ratio for access pattern P?*”
  - Not ok: “*Analyze system A*”
  - Not ok: “*Prove that A is better than B*”
Common mistakes: *Inappropriate level of detail*

• Avoid complexity if possible
  • Complex models require a deep understanding of the inner working of the system being modeled
    – this understanding is often not available
  • Detailed input parameters (e.g., service times) are needed to produce meaningful results
    – such parameters are often unknown
• Prefer simpler models which depends on few parameters
Common mistakes: 
*Underestimating the manpower*

- Doing a proper capacity planning study requires time and experience analysts
  - Performance analysis, like programming, cannot be mastered in a week
  - The level of required competence increases with the complexity of the model
- More on manpower later
Common mistakes: *Taking no action*

- Are you willing to take into account the results of the performance study?

  I completed the performance study. Results show that system A provides higher throughput than B for workload W.

  System A is too expensive. We are going to buy system B anyway.
Performance evaluation of Storage Systems

• Some relevant literature

  • “Towards a Performance Model for Virtualised Multi-Tier Storage Systems”, USENIX
    http://www.usenix.org/events/fast08/wips_posters/lebrecht-wip.pdf
  
  • “Analytical Performance Modeling of Hierarchical Mass Storage Systems”, IEEE ToC
    http://portal.acm.org/citation.cfm?id=268702
  
  • “A Performance Model of Disk Array Storage Systems”,
    http://www.cs.unh.edu/~varki/publication/cmg.ps
  
  • Others...
Performance evaluation of Storage Systems

- Existing studies consider relatively simple storage systems (single disks, RAID arrays)
- Existing models are highly specific and cannot be easily generalized
  - In general, models tend to depend on the goal of the study
- Lack of software tools for simulation/modeling of storage systems
  - General-purpose simulation engines exist, but models must be defined/coded by hand
Performance evaluation of Storage Systems

- The DiskSim Simulation Environment: http://www.pdl.cmu.edu/DiskSim/
  - Targeted at accurate, single-disk modeling
- The MONARC project at CERN: http://www.cern.ch/MONARC/
  - Early effort for modeling of a distributed computing infrastructure
  - Unfortunately this project finished long ago
“Toy” example

- Analysys farm with a number of computing nodes connected to a storage system
- Fixed number of N jobs
  - Some are using the CPUs
  - Some are waiting for data access
- The storage system is two-tiered
  - A small disk pool provides a fast data cache
  - A large, slow tape library holds all data
System model

- Based on a closed Queueing Network
  - Service times are exponentially distributed
  - Routing is purely probabilistic

![System model diagram]
Model parameters

- $N$ (unknown)
  - Number of concurrent jobs
- $S_{CPU}$ (measured)
  - Average length of computation (w/o data access)
- $S_{disk}$ (measured)
  - Average time to access data from disk
- $S_{tape}$ (measured)
  - Average time to access data on tape
- $p_{hit}$ (estimated, e.g. by simulation using actual access traces)
  - Cache hit probability
Question 1

- Given $S_{\text{CPU}} = 1000$, $S_{\text{tape}} = 200$, $S_{\text{disk}} = 1$

- What is the max system throughput when $p_{\text{hit}} = 0.5$?
- What is the max system throughput when $p_{\text{hit}} = 0.8$?

The system saturates at about $N = 20$

The system saturates at about $N = 40$
Question 2

- Given $S_{CPU} = 1000$, $S_{tape} = 200$, we can buy:
  - Disks model A (cheap, slow)
    - $S_{diskA} = 1.0$, $p_{hitA} = 0.4$
  - Disks model B (expensive, 20% faster than model A)
    - For the same money we can buy less disk space
    - We estimate the cache hit probability decreases to 0.3
    - $S_{diskB} = 0.8$, $p_{hitB} = 0.3$
Answer to question 2

- In this case a large cache of slow disks is better than a small cache of faster disks
Model extensions

- Non-exponential service times
- Multiple classes of requests
- Priorities
- Batch arrivals
- ...
- Note: any of the above makes the model non-trivial to solve
  - Nevertheless, it is possible to handle all extensions above, and many others...
Back to the original question

• “Simulating storage system performance: a useful approach for SuperB?”

• My anwer: of course it is!!
  • Proper capacity planning helps to identify and discard wrong design decisions early
  • Potential limitations of the system can be identified and a plan to address them prepared in advance
  • Many questions can be given informed answers
Manpower estimates

• Disclaimer
The most (IMHO) difficult steps

1. Formulate the problem and define the goals
2. Collect input data and define the model
3. Implement the model
   - Verified?
     - no
       - Evaluate the model
         - Valid?
           - no
             - no
               - Analyze and present output data
Tasks and durations

- Define Goals
- Collect data
- Implement & validate
- Analyze
- Present

- Increase by 1PM if training on performance modelling is needed
Additional resources

- JavaSim: a discrete-event simulator written in Java: http://javasim.codehaus.org/
- The Java Modelling Tools from Politecnico di Milano, Italy: http://jmt.sourceforge.net/
- The qnetworks toolbox: a Queueing Networks analysis package for GNU Octave http://www.moreno.marzolla.name/software/qnetworks/
- The DiskSim Simulation Environment: http://www.pdl.cmu.edu/DiskSim/
- The MONARC project at CERN: http://www.cern.ch/MONARC/
Post Scriptum
Garbage in, Garbage Out

On two occasions I have been asked, —"Pray, Mr. Babbage, if you put into the machine wrong figures, will the right answers come out?" ... I am not able rightly to apprehend the kind of confusion of ideas that could provoke such a question.”

— Charles Babbage, Passages from the Life of a Philosopher