Leistungsanalyse von Rechnersystemen

18. Oktober 2006

Nöthnitzer Straßen 46
Raum 1026
Tel. +49 351 - 463 - 35048

Holger Brunst (holger.brunst@tu-dresden.de)

---

Recommended Literature

- Raj Jain: The Art of Computer Systems Performance Analysis
- Rainer Klar, Peter Dauphin, Fran Hartleb, Richard Hofmann, Bernd Mohr, Andreas Quick, Markus Siegle
  Messung und Modellierung paralleler und verteilter Rechnersysteme
Announcements

- Class slides are now available at:
  http://tu-dresden.de/die_tu_dresden/zentrale_einrichtungen/zih/lehre/ws0607/lars
- Access information in the future:
  - User: lars06
  - Password: larsdnld
- Guided tour through our new machine room:
  - Thursday 19th at 9:45, Zellescher Weg 12 below the bridge

Summary of Previous Lecture: Motivation

- Invent of computer changed our lives regarding data processing
- Can do computations much faster. Factor: $10^{10}$ - $10^{14}$
- Interplay of memory, cache, registers, and I/O
- HPC: A key technology? Its role in the USA, Japan, and Europe
- Required know how for HPC: Algorithms, performance, programming, and machine operation
- Challenges: Languages, parallelization, network, scheduling, system architecture, programming models
- Software is a key technology
Summary of Previous Lecture: Our Center

- ZIH is a Central Scientific Unit at Technische Universität Dresden
- Build from former URZ and ZHR
- Support real users
- Development of algorithms and methods in close cooperation with all departments
- Provides HPC infrastructure and IT services
- Research in: Performance analysis, programming methods, software tools, modeling of biological algorithms, mathematical models and algorithms
- Role of mediator between vendors, developers, and users
- Teaching and education

Summary of Previous Lecture: New HPC Infrastructure

- **HPC -SAN**
  - Capacity: > 50 TB
  - Main memory: ≥ 4 TB
  - 8 GB/s

- **HPC -Server**
  - Main memory: ≥ 4 TB

- **PC -SAN**
  - Capacity: > 50 TB
  - 4 GB/s

- **PC Farm**
  - 4 GB/s

- **PetaByte Tape Silo**
  - Capacity: ≥ 1 PB
  - 1.8 GB/s
Summary of Previous Lecture: Activities at ZIH

- Vampir: Performance analysis of parallel programs
- BenchIT: Benchmark based evaluation of computer architectures
- Parbench: Modeling and analysis of parallel workloads

Preliminary Remarks

- The development of computer systems in respect of hard- and software is accompanied by performance measurements and modeling since the 1960s
- However, only a small fraction of the research work is applied in the field
- Ferrari (1986):
  The study of performance evaluation as an independent subject has sometimes caused researchers in the area to lose contact with reality.
- Why is it that performance measurements are by no means an integrated and natural part of computer system development?
  - The primary duty of system developers is to create functionally correct systems!
  - Performance evaluation tends to be optional. Some people compare it to the freestyle event in ice-skating
Preliminary Remarks

- The term 'performance' is ambiguous in computer science. It can stand for:
  - “Well, it’s functioning (more or less)”;
  - A short development cycle;
  - High throughput;
  - Short response times;
  - Good reliability.
- Doherty (1970)
  Performance is the degree to which a computing system meets expectations of the persons involved in it.
- Graham (1973)
  Performance ... is the effectiveness with which the resources of the host computer system are utilized toward meeting the objectives of the software system. Or short: How well does the system enable me to do what I want to do?
- Ferrari (1978)
  We use the term ‘performance’ to indicate how well a system, assumed to perform correctly, works.
- DIN-Norm 66273
  The German DIN-Norm considers a computer system as a black box and “ ... baut die Messung und Bewertung der Schnelligkeit ausschließlich auf das Verhalten der Datenverarbeitungsanlage an der vom Anwender sichtbaren Schnittstelle auf.”
- Jain (1991)
  Contrary to common belief, performance evaluation is an art. ... Like artist, each analyst has a unique style. Given the sample problem, two analysts may choose different performance metrics and evaluation methodologies.
Objectives

- Performance analysis: Get highest performance for a given cost
- "Performance Analyst": Anyone who is associated with computer systems, i.e. systems engineers and scientists but also users

Tasks:
- Specification of performance requirements
- Evaluation of design alternatives
- Comparison of two or multiple systems
- Finding the best value of given system parameter (system tuning)
- Identification of bottlenecks
- Workload characterization for a given system
- Finding the right size and number of hardware and software components (capacity planning)
- Performance prediction at future workloads (forecasting)

Conventions

System:
- An arbitrary collection of hardware, software, and firmware e.g.:
  - CPU
  - Database
  - Network of computers

Metric:
- A criteria used to evaluate the performance of a system e.g.:
  - Response time
  - Throughput
  - Floating point operations per second

Workload(s):
- The overall sum of user requests to a system e.g.:
  - CPU workload: Instructions to execute
  - Database workload: Which queries to perform
Example 1: Selection of Technique, Metric, and Workload

- What performance metrics should be used to compare the performance of the following systems?
- How and where would you start?
  - Two disk drives or SANs?
  - Two transaction processing systems?
  - Two packet retransmission algorithms?
- Examples: Capacity, read/write throughput, seek latency, energy consumption, mean-time to failure, emission of heat and noise, form factor etc.

Example 2: Correctness of performance measurements

- How to measure the performance of a computer system?
- At least two tools are required:
  - Load generator
  - Performance monitor
- Which type of monitor would be more suitable (software or hardware) for measuring the following quantities?
  - Number of instructions executed by a processor
  - Degree of multiprogramming on a timesharing system
  - Response time of packets on a network
Example 3: Measurement and Simulation Design

The performance of a system depends on the following three factors:

- Garbage collection used: G1, G2, or none
- Type of workload
  - Editing
  - Computing
  - Artificial intelligence
- Type of CPU: C1, C2, or C3

How many experiments are needed?
How does one estimate the performance impact of each factor?

Example 4: Application of Simple Queuing Models

The average response time of a database system is three seconds. During a 1-minute observation interval, the idle time on the system was 10 seconds.

A queuing model for the system can be used to determine the following:

- System utilization
- Average service time per query
- Number of queries completed during the observation interval
- Average number of jobs in the system
- Probability of number of jobs in the system > 10
- 90 percentile response time
  - German: Perzentile/Prozentränge oder allg.: Quantile
  - Reminder: 50th percentile is the median
- 90 percentile waiting time
Example 4: Quantil

\[
\begin{align*}
Q_p & \quad 1 - p \\
p &
\end{align*}
\]

The Art of Performance Evaluation

- Successful evaluation cannot be produced mechanically
- Evaluation requires detailed knowledge of the system to be modelled
- Careful selection of methodology, workload, and tools
- Conversion from an abstract feeling or notion to a real problem which needs to formalized in a way that can be handled by established tools
- Analysts tend to have different “styles”
- A typical case:
  - Assume two systems A and B with the following throughputs measured in transactions per second:

<table>
<thead>
<tr>
<th>System</th>
<th>Workload 1</th>
<th>Workload 2</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>20</td>
<td>10</td>
</tr>
<tr>
<td>B</td>
<td>10</td>
<td>20</td>
</tr>
</tbody>
</table>
The Ratio Game

Three ways to compare the two systems:

1. Comparing the average throughput:

<table>
<thead>
<tr>
<th>System</th>
<th>Workload 1</th>
<th>Workload 2</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>20</td>
<td>10</td>
<td>15</td>
</tr>
<tr>
<td>B</td>
<td>10</td>
<td>20</td>
<td>15</td>
</tr>
</tbody>
</table>

- Throughput with respect to system B:

<table>
<thead>
<tr>
<th>System</th>
<th>Workload 1</th>
<th>Workload 2</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>2.00</td>
<td>0.50</td>
<td>1.25</td>
</tr>
<tr>
<td>B</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
</tbody>
</table>

- Throughput with respect to system A:

<table>
<thead>
<tr>
<th>System</th>
<th>Workload 1</th>
<th>Workload 2</th>
<th>Average</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>1.00</td>
<td>1.00</td>
<td>1.00</td>
</tr>
<tr>
<td>B</td>
<td>0.50</td>
<td>2.00</td>
<td>1.25</td>
</tr>
</tbody>
</table>

Well, equal performance

Great! System A is 25% better!

But no! System B is 25% better!

Performance Games

- Ratio game is just one representative
- Similar games are popular
- Selection of the right workload, measurement, or presentation method
- Sometime intentional to show the superiority of a certain system
- Lack of performance evaluation techniques

Therefore: Knowledge of common mistakes and games is important for
- choosing the right methodology as an analyst;
- questioning offers, recommendations, and advertisements as a consumer or buying agent.
Common Mistake Classes in Performance Analysis

<table>
<thead>
<tr>
<th>Performance Mistakes</th>
<th>Goals</th>
<th>Methodology</th>
<th>Completeness</th>
<th>Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>None</td>
<td>None</td>
<td>Unsystematic</td>
<td>Overlooking Parameters</td>
<td>None</td>
</tr>
<tr>
<td>Biased</td>
<td>Biased</td>
<td>Incorrect Metrics</td>
<td>Ignore Factors</td>
<td>Erroneous</td>
</tr>
<tr>
<td>Problem?</td>
<td>Problem?</td>
<td>Bad Workload</td>
<td>Wrong Level of Detail</td>
<td>Too Complex</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Wrong Technique</td>
<td></td>
<td>Ignore Input Errors</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Bad Experiments</td>
<td></td>
<td>Irgnoring Variability</td>
</tr>
</tbody>
</table>

Presentation
- Improper Presentation
- Ignoring Social Aspects
- Omitting Assumptions
- Omitting Limitations

Common Mistakes: What are the goals?

- No goals with a good understanding of the problem
  - Many performance efforts are started without clear goals
  - Performance model must be developed with a particular goal in mind
  - First, understand the system and the problem (40%)
  - Then, start writing the simulation code
  - Not trivial. Goals often change with a better understanding of the problem

- Biased goals
  - “show that one system is better than another”
  - Metric and workload are not selected for proper comparison but for highlighting a given system
  - Performance analyst are to be unbiased!
  - The role of a performance analyst is like that of a jury
  - Depend your conclusions on results rather than on believes
Common Mistakes: Methodology Selection

- Unsystematic approach
  - Arbitrary selection of system parameters, factors, metrics, and workloads lead to inaccurate conclusions. Be complete!
- Incorrect performance metrics
  - Example 1: Comparison of MIPS of a RISC and a CISC architecture
  - Example 2: ALDI computer “datasheets” for MHz, MB, and Megapixel fans
- Unrepresentative workload
  - Workload should represent the actual usage of the system in practice
  - Example: Packet sizes in a network
- Wrong evaluation technique
  - Analysts are often "married" to one technique i.e. measurement, or simulation, or analytical modeling
  - Resulting in a model optimized for the analyst rather than the problem
  - An analyst should have a basic knowledge of all three techniques

Common Mistakes: Completeness and Balance

- Overlooking important parameters
  - List system and workload characteristics that affect performance
  - System: quantum (CPU) and working set (memory) size
  - Workload: number of users, request patterns, priorities
- Inappropriate level of detail
  - Very different alternatives: Use high-level model
  - Slight variations: Use more detailed model
  - Do not take a detailed approach when a high-level model will do and vice versa
- Ignoring significant factors
  - Varied parameters are called factors
  - Usually, not all parameters are factors.
  - Identify the ones that significantly alter performance if varied e.g. response time: packet size vs. arrival rate
  - Favor factors that are directly controlled by the user
  - The choice of factors should be based on relevance, not on their knowledge
Common Mistakes: Analysis

- No analysis
  - Analysts are good at collecting enormous amounts of data but often cannot analyze the data and write understandable summaries.
  - Result: No useful analysis at all or a thick report with many graphs but no interpretation.
  - Teamwork can help.

- Erroneous analysis
  - Let’s average ratios! Short simulation runs or so much more convenient!
  - No sensitivity analysis (German: Empfindlichkeitsanalyse).
  - Do not present your results as facts but as evidence.
  - Performance results may be sensitive to workload and system parameters.

- Ignoring errors in input
  - Parameters of interest cannot be measured. Example: Network device.

- Improper treatment of outliers: Measurement error vs. system phenomenon.

- Analysis too complex: Published models are often too complex for the real world.

- Ignoring variability: Common to analyze only the mean performance. Example: Daily averages of computer demands which ignore the large hourly peaks.

Common Mistakes: Presentation

- Improper presentation of results
  - Help decision making.
  - “The right metric to measure the performance of an analyst is not the number of analyses performed but the number of analyses that helped the decision makers.”

- Ignoring social aspects
  - Presentation requires social and substantive skills.
  - Analysts typically have good substantive skills…
  - Trust between analyst and decision makers.
  - Conflict of interest: Innovativeness of the modeling approach (analyst) vs. quickly getting to the final results (decision maker).

- Omitting assumptions and limitations
  - Users will try to reproduce your results under their assumptions which is likely to reveal different results.
Checklist for Avoiding Common Mistakes I

- Is the system correctly defined and the goals clearly stated?
- Are the goals stated in an unbiased manner?
- Have all the steps of the analysis followed systematically?
- Is the problem clearly understood before analyzing it?
- Are the performance metrics relevant for this problem?
- Is the workload correct for this problem?
- Is the evaluation technique appropriate?
- Is the list of parameters that affect performance complete?
- Have all parameters that affect performance been chosen as factors to be varied?
- Is the experimental design efficient in terms of time and results?
- Is the level of detail proper?
- Is the measured data presented with analysis and interpretation?

Checklist for Avoiding Mistakes II

- Is the analysis statistically correct?
- Has the sensitivity analysis been done?
- Would errors in the input cause an insignificant change in the results?
- Have the outliers in the input or output been treated properly?
- Have the future changes in the system and workload been modeled?
- Has the variance of input been taken into account?
- Has the variance of the results been analyzed?
- Is the analysis easy to explain?
- Is the presentation style suitable for its audience?
- Have the results been presented graphically as much as possible?
- Are the assumptions and limitations of the analysis clearly documented?
A Systematic Approach to Performance Evaluation

1. State goals and define the system
2. List services and outcomes
3. Select metrics
4. List parameters
5. Select factors to study
6. Select evaluation technique
7. Select workload
8. Design experiments
9. Analyze and interpret data
10. Present results

Example: The Acquisition Process of our new HPC system

1. State goals and define the system
2. List services and outcomes
3. Select metrics
4. List parameters
5. Select factors to study
6. Select evaluation technique
7. Select workload
8. Design experiments
9. Analyze and interpret data
10. Present results
Thank You!

Nöthnitzer Straße 46
Raum 1026
Tel. +49 351 - 463 - 35048

Holger Brunst (holger.brunst@tu-dresden.de)