Ferdinand - Middleware measurement

Lukáš Marek
Department of Distributed and Dependable Systems
http://d3s.mff.cuni.cz

CHARLES UNIVERSITY PRAGUE
Faculty of Mathematics and Physics
Ferdinand project

• Targeted mainly on performance
• Who is involved
  ▪ Charles University in Prague
    • Faculty of Mathematics and Physics – DDDS
  ▪ Academy of Sciences of the Czech Republic
  ▪ Karlsruhe Institute of Technology
Middleware measurement

• JMS measurement
  ▪ Lukáš + Petr + Luboš
  ▪ Goal
    • Measure various aspects of several JMS brokers
    • Identify points affecting throughput and message delivery time
    • Create generalized performance model

• EJB measurement
  ▪ Andrej + Luboš + Petr
  ▪ Same goal as above
Performance Annotations

• Performance Annotations
  ▪ Jarda + Tomáš
  ▪ Performance-based unit testing
  ▪ Conditions in big-oh notation
KIT Projects

• Storage Virtualization
  ▪ Qais Noorshams
  ▪ Performance prediction of virtualized storage

• Improving the Quality of Model Transformations
  ▪ Andreas Rentschler

• Infrastructure Parameter Detection for Performance Prediction based on Goal-oriented Measurements
  ▪ Michael Hauck
• Goal
  - Measure various aspects of several JMS brokers
  - Identify points affecting throughput and message delivery time
  - Create generalized performance model

• Methodic
  - How to cover all performance relevant factors
  - Analyze data
  - ...
Generalized performance model

- Performance model for the whole class of middlewares
  - JMS middlewares or even messaging mid.
- Why not for single JMS implementation?
  - Changes over time
  - Factors that are relevant only for single implementation
  - Model is outdated even before is published
Generalized performance model II

• What will be included in the model?
  ▪ Factors that are relevant for all (majority) implementations
    • Length of the send/received messages
    • Guaranteed delivery
    • Message rate (number of messages processed concurrently on broker)
    • ...

Lukáš Marek
Measurements

- JMS Brokers
  - ActiveMQ
  - HornetQ
- Benchmarks
  - SPECjms2007
SPECjms2007 – overview

source: www.spec.org
SPECjms2007 – overview

source:www.spec.org
SPECjms2007 – topology

source:www.spec.org
Microbenchmarks vs. real workloads

- Microbenchmarks
  + designed to trigger only single perf. effect
  + easier to evaluate
  + available (you have to program them)

- Real workloads
  + more representative
  + triggers all sorts of unpredicted effects
  + easier to publish
Measurements – instrumented points
Results
SPECjms2007 results
Why ???
Welcome to the real workload
Real workload

- Too many effects in one picture
- Real workload is much more unpredictable than microbenchmarks
- Not enough (“too much”) data
- (We are doing something wrong)
How to solve this

- Data evaluation using statistics
- Gather more data
  - Observe more parameters
- Modify parameters of the workload
  - Make it more predictable
  - Generate higher load
  - Kai Sachs et al.: Performance evaluation of message-oriented middleware using the SPECjms2007 benchmark
Generalized performance model

- Performance model for the whole class of middlewares
  - Valid for all (most of) JMS brokers
- General model captures the influences between performance parameters
- Parametrized model can predict performance
- If the implementation does not comply with our model, something is (probably) oddly designed
Motivation for generalized model

• Several JMS models
  ▪ Michael Menth et al.: Throughput Comparison of Professional JMS Servers
    • Filters and replication
  ▪ Jens Happe et al.: Statistical Inference of Software Performance Models for Parametric Performance Completions
    • Blackbox models

• Overall behavior of JMS not covered

  • TODO: Kai Sachs et al.: Performance evaluation of message-oriented middleware using the SPECjms2007 benchmark
Conclusion

• Experiments started
• Lessons learned
  ▪ Enterprise software needs enterprise service
  ▪ Performance evaluation of “real workload” is hard
• First results point to statistical analysis and benchmarking under more observed (controlled) environment
Partial end

Questions?
My stay in Lugano - Switzerland
Program of the stay

- Ferdinand project
- Walter Binder: Towards an Aspect-Oriented Domain-Specific Language for Efficient Dynamic Program Analysis
JPMF

• Java Performance Measurement Framework
  ▪ dev. lead Luboš
  ▪ Basic instrumentation done

• Support for better (easier) instrumentation needed

• Tasks for the stay
  ▪ Integrate Walter's project with JPMF
  ▪ Use JPMF in Ferdinand project
Object instance profiling

- New way of profiling applications
- Profiling instances (not classes – code)
- The outcome of the stay could be also used here
Another questions?