Performance analysis and performance modeling of web-applications

Dr. Heinz Kredel
Dr. Hans-Günther Kruse
Dr. Ingo Ott

University of Mannheim – IT-Center

3PGCIC 2011 - Barcelona

Agenda

Motivation for modeling

Performance measurement

Performance modeling

Summary

Motivation for modeling

Highly integrated systems, many customers

Enhancing customer satisfaction (by)

Improving response time

Analysis needs rebuilding infrastructure

You won‘t want to analyze in productive environment

Very expensive

Decision making is often estimated

gut feeling plus a buffer

Matches reality?

Performance measurement

Analyzing a realistic scenario:

- After “Freshman student” event
- 1.600 Students planned their study plan
- concurrent and on a 1:1:1-infrastructure
  (1 Web-, 1 App-, 1 DB-Server)

Problems arised:

- Slower response times
- Even timeouts
- No login possible
- Not all customers were satisfied!

Performance measurement

Solution: More hardware

Three Questions:

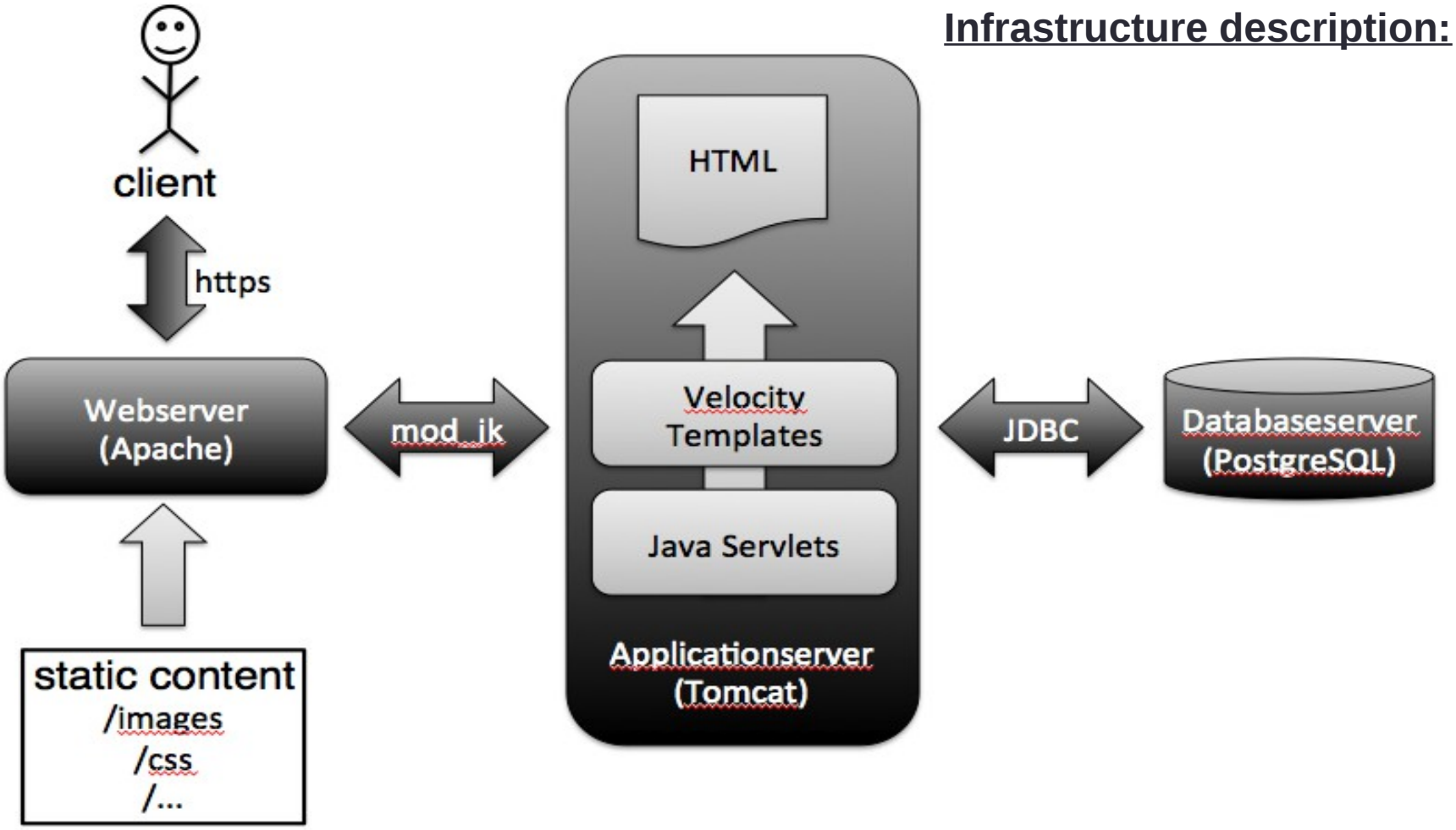
- What is the applicable amount?
- Where is the bottleneck?
- What can be done in realistic time?

What we have done:

- Rebuilding Infrastructure
- Simulation of a specific scenario with funkload
- Evaluating results

Performance measurement

Infrastructure description:



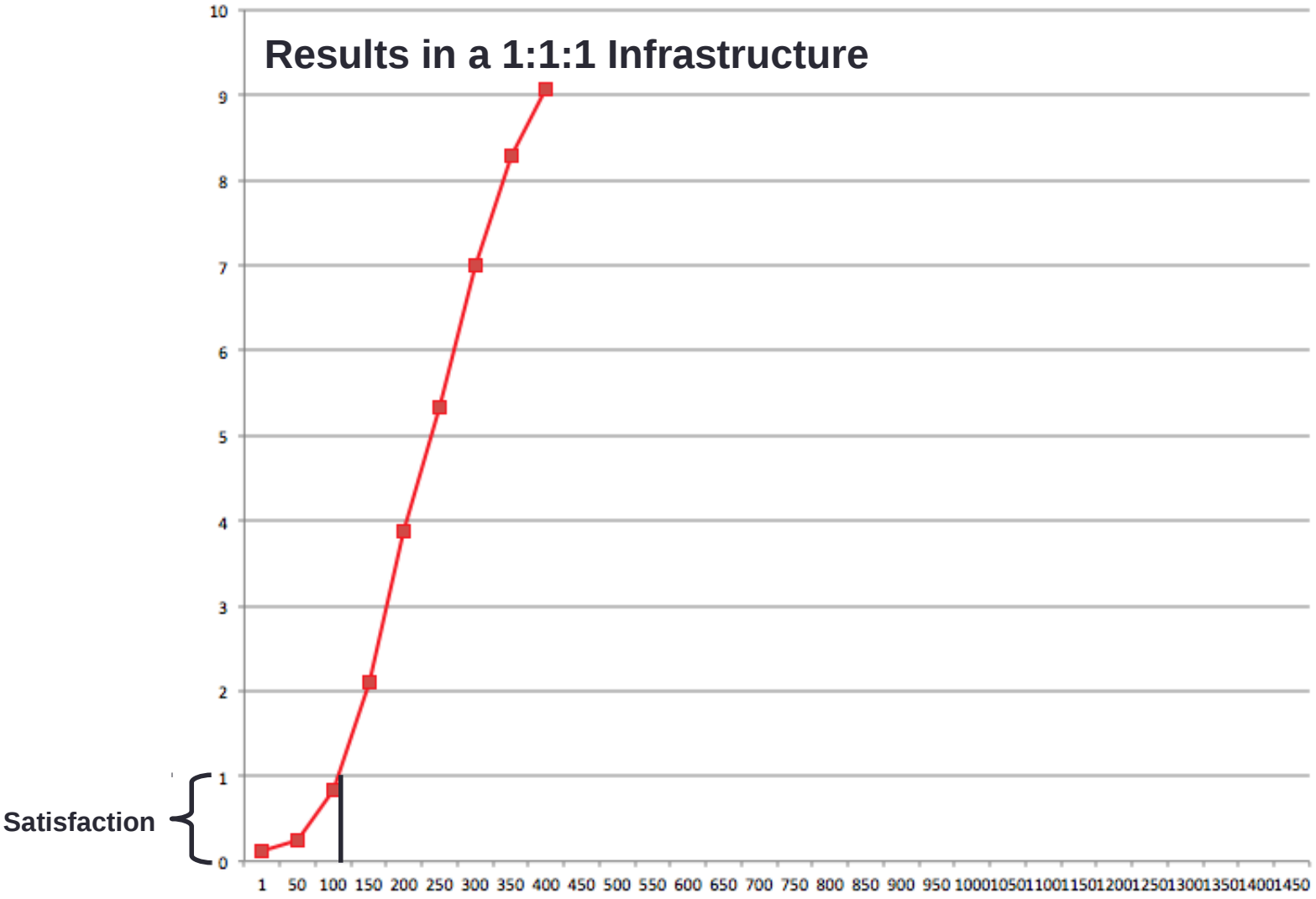
Performance measurement

Simulation scenario:

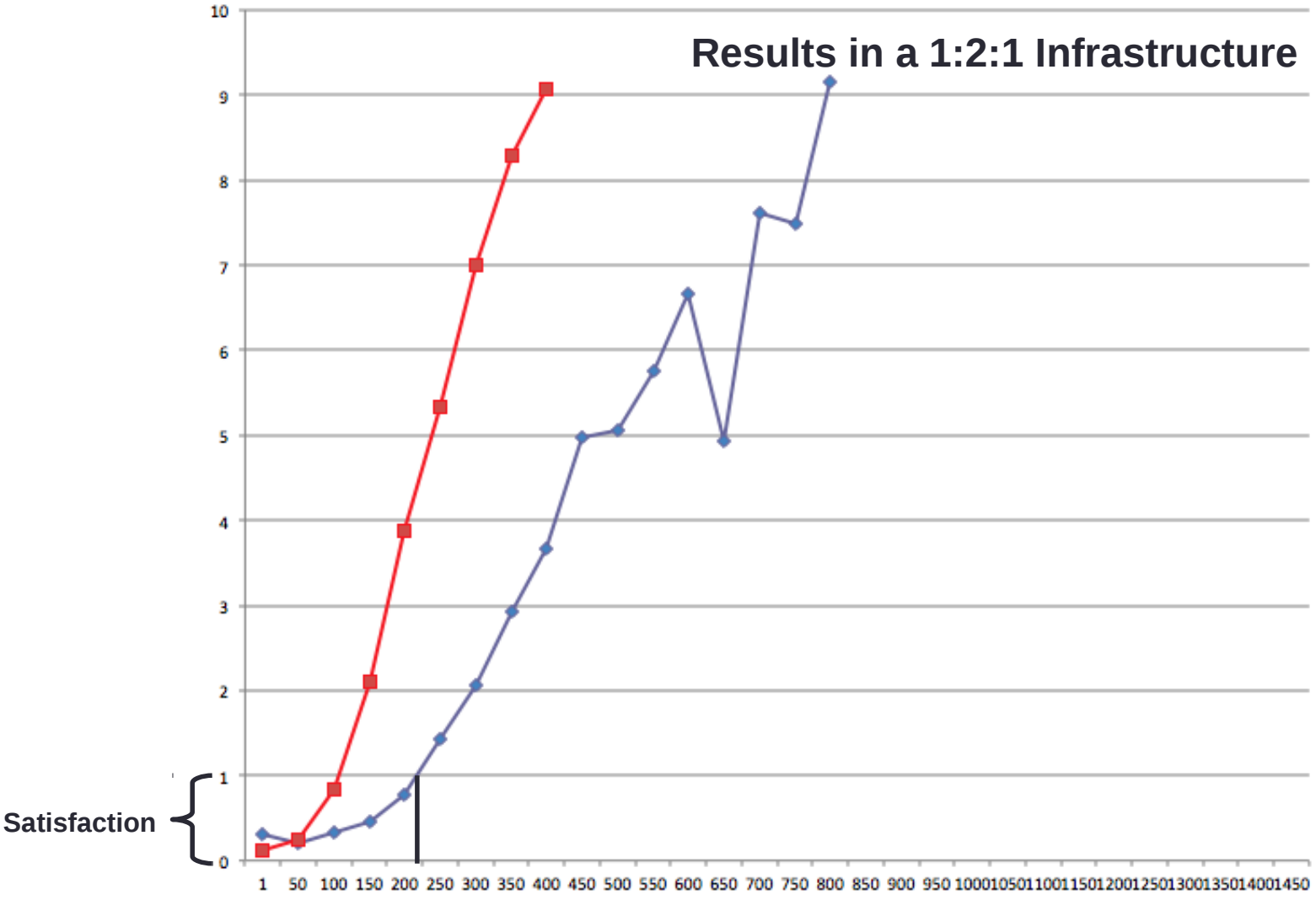
- Call the start page
- Login with a random user
- Navigate through the lecture index
- Load a defined webpage of a lecture
- Logoff

Calculate the base load
Increase app-server until “system fits”

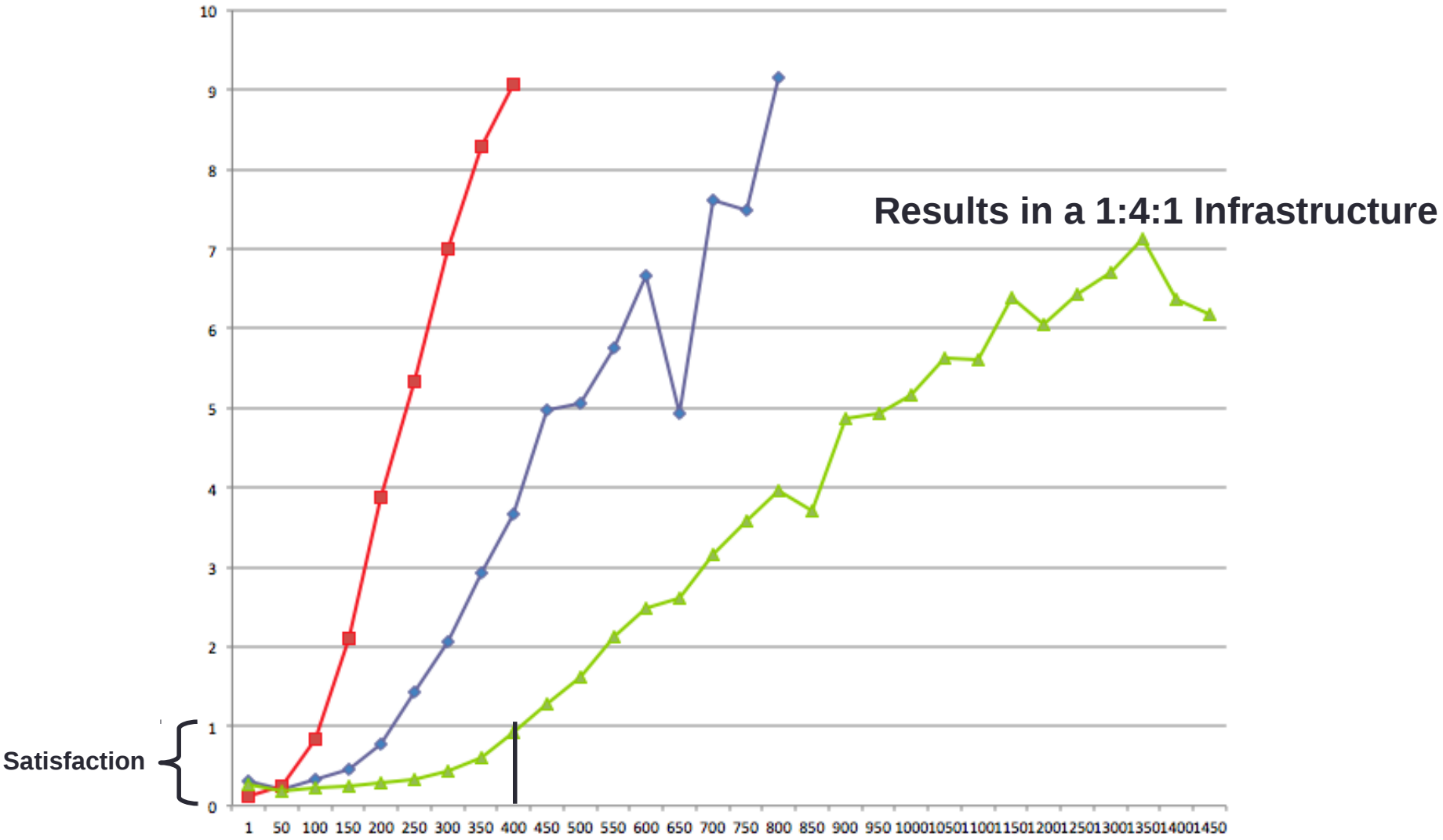
Performance measurement



Performance measurement



Performance measurement



Performance measurement

Simulation-Result:

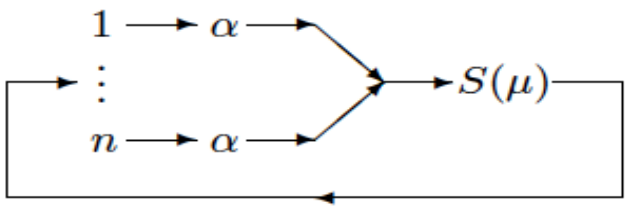
- Duplication of app-server leads to duplication of maximum number of concurrent users

- Average response time is cut in half and increases more slowly by duplication

Performance modeling

Based on results and current infrastructure

Modeled system:



n: # clients
α: client request rate
μ: service rate of server S

Constraints:

service and interarrival times exponentially distributed

Our modeling process is based on classical file-server-model

Point of Interest:

Probability of waiting pw

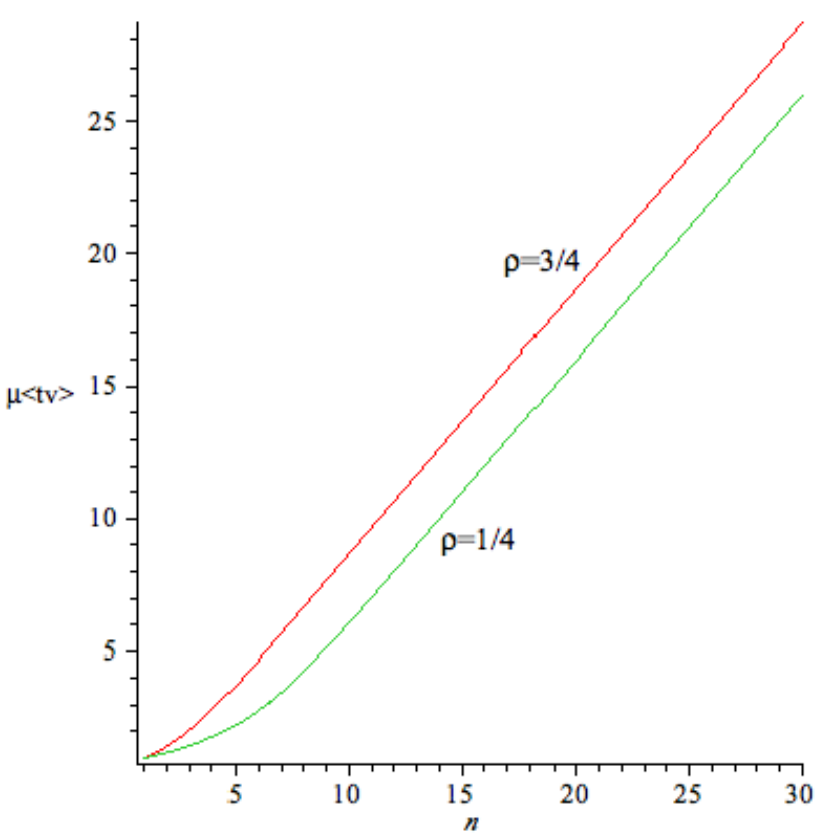
Average response time μ<tv>

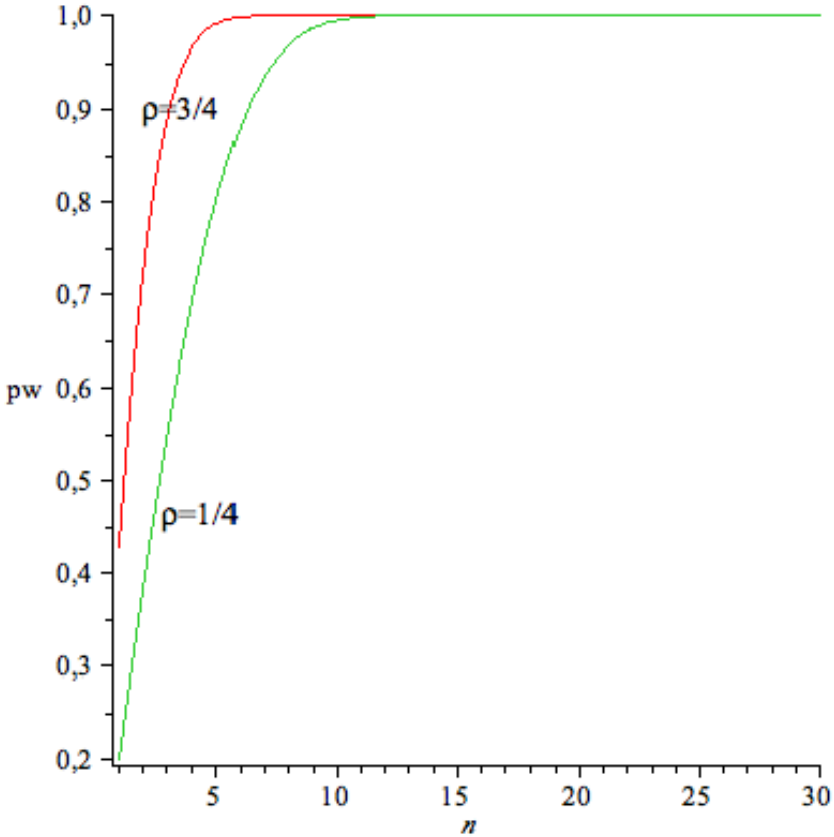
Performance modeling

After solving equations…

$$\mu < t_v >= \frac{n}{p_w} - \frac{1}{\rho} \qquad p_w = 1 - \pi_0 = 1 - \frac{1}{S_n(\rho)}$$

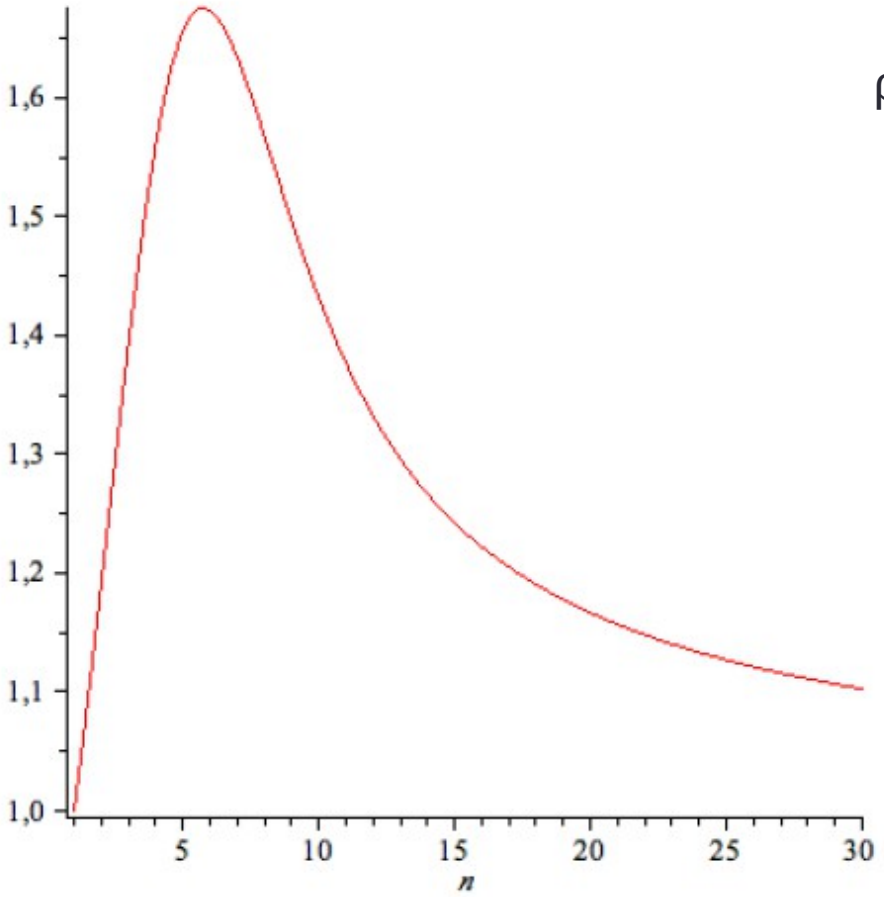
Load: ρ=α/μ





Performance modeling

Speedup matches observed data



ρ=3/4 and ρ=1/4

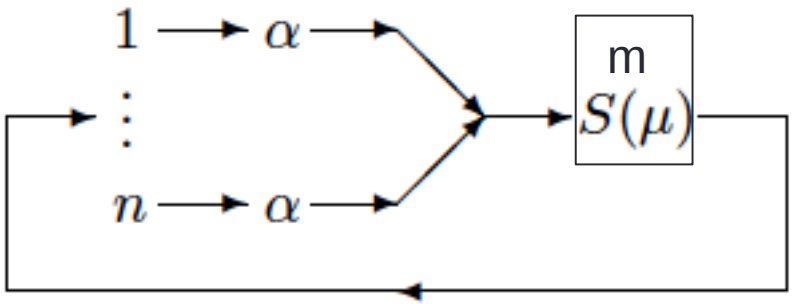
Performance modeling

So far only 1 server considered
Client request rate α mainly triggers response time
„Put“ more parallel servers into the model...
Changed server model:

Constraints:

Simplified view

no Database modeled



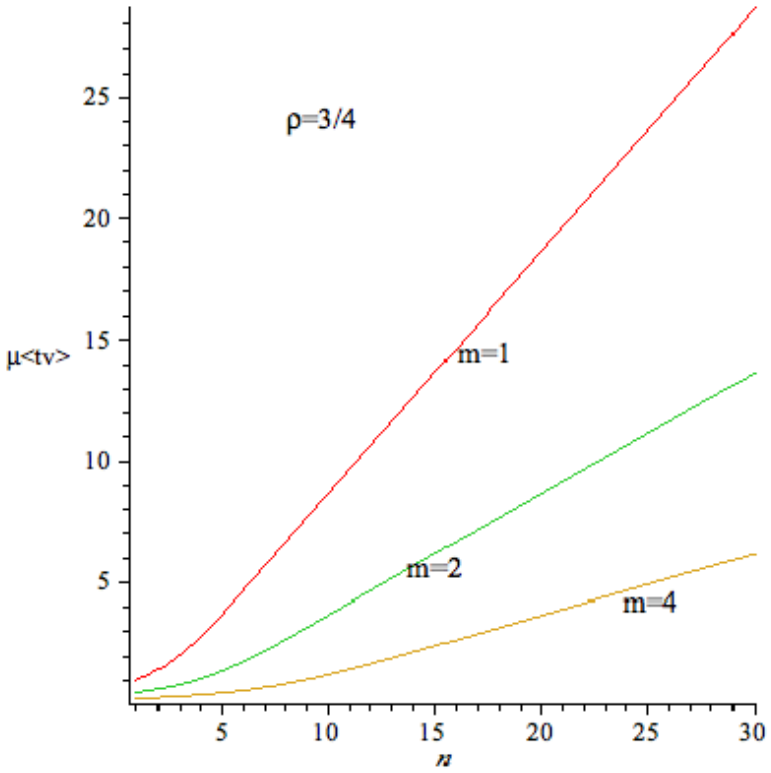
- Point of Interest:
  - Probability of waiting pw(m)
  - Average response time μ<tv>

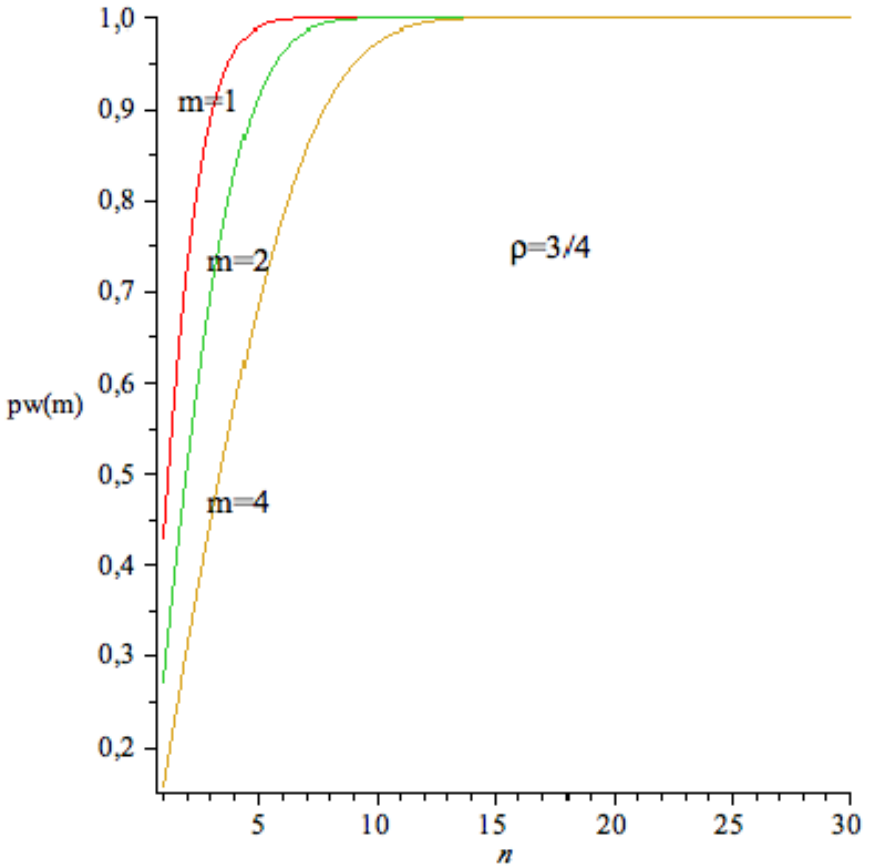
Performance modeling

After solving equations…

$$\mu < t_v > \mapsto \mu < t_v^m > = \frac{(n/m)}{p_w(n,m,\rho)} - \frac{1}{\rho} \qquad p_w(n,\rho) \mapsto p_w(n,m,\rho) = 1 - S_n^{-1}(\frac{\rho}{m})$$

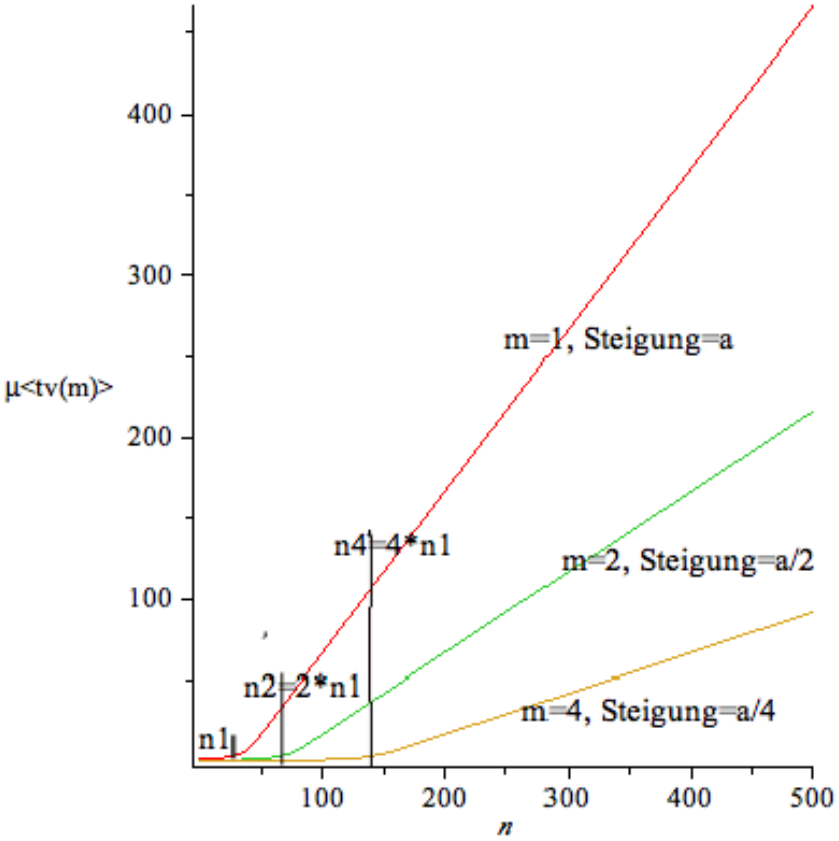
Load: ρ = α/μ = 3/4

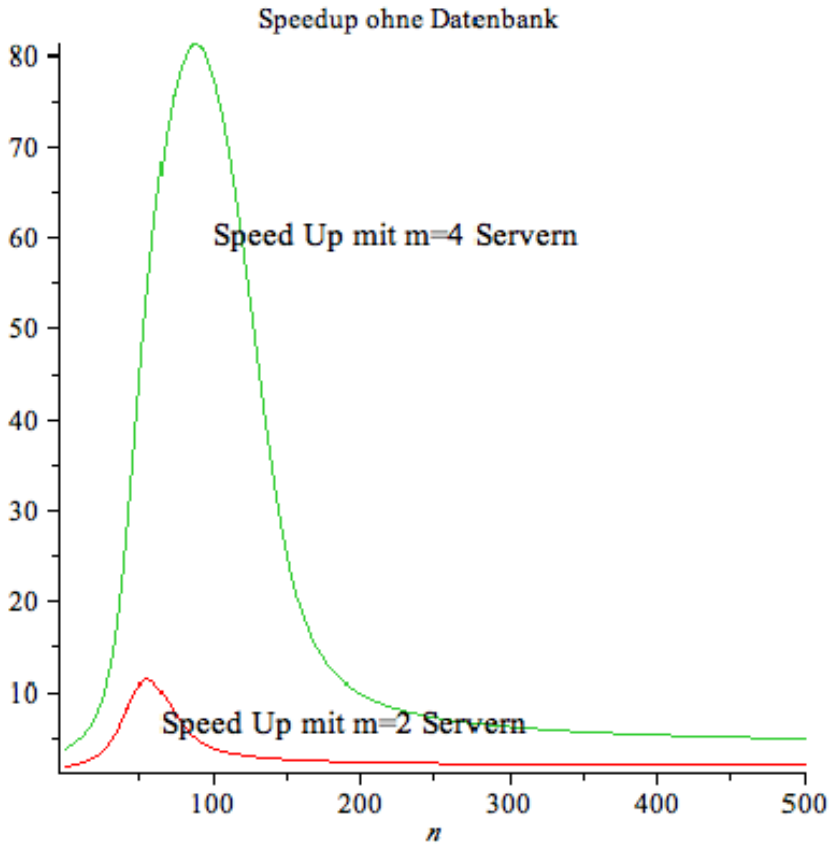




Performance modeling

Speedup matches observed data
Model adjusted (α and μ)





Performance modeling

Qualitative aspects of model matches empirical result

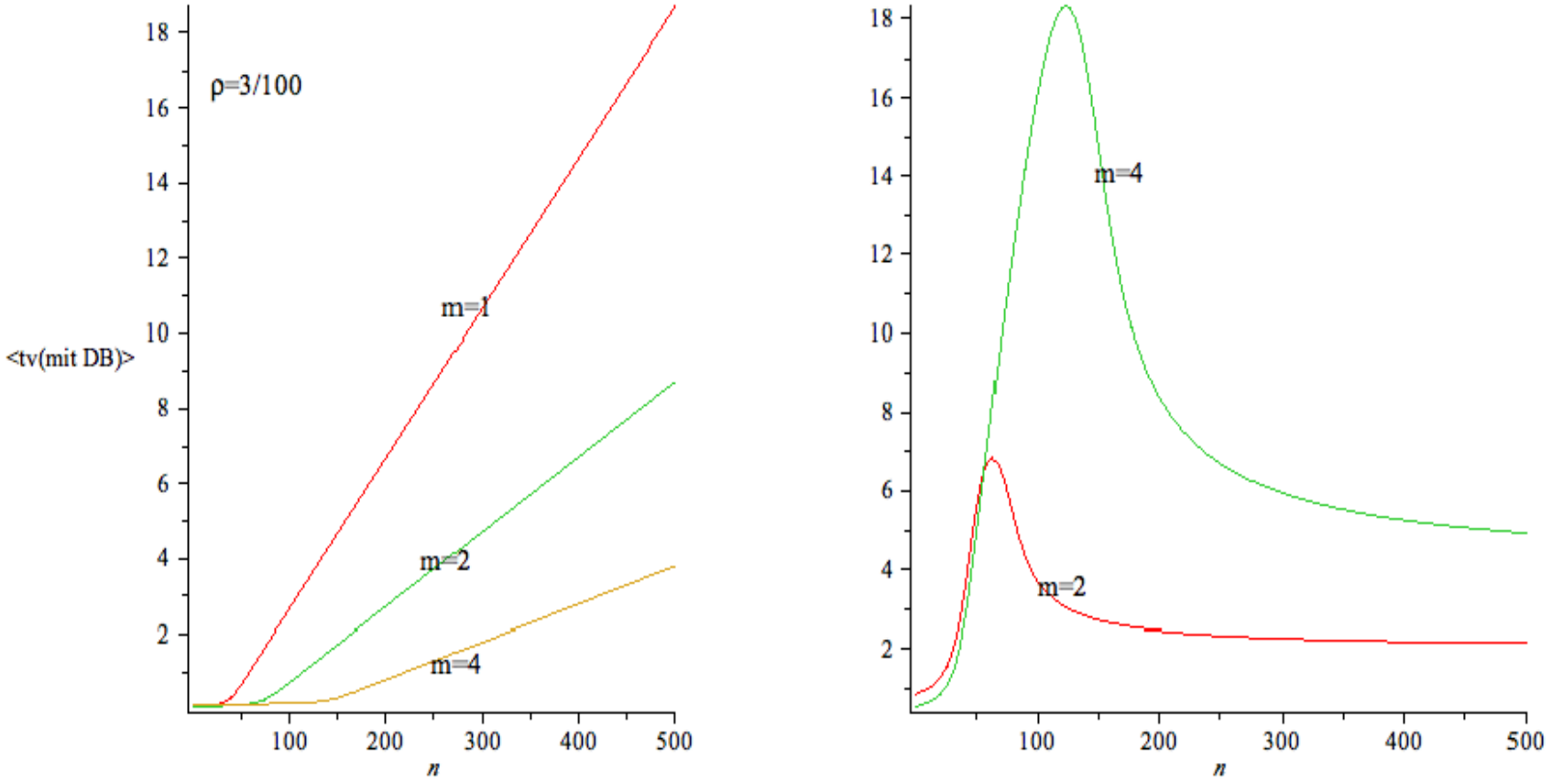
Database missing in model

Extended model:

- Database with own service rate
- Markov models like M/M/1/∞ or M/G/1/∞ are appropriate
- Load of database is fixed by observation to (μ/μDB)=0.9
- Not in all requests the database is needed

Performance modeling

- Average response time μ<tv> with Database and
- Corresponding Speedup
- Taking the database into account hardly changes shapes



Summary

Performance analysis with rebuilding infrastructure
- detailed and accurate insights of a system
- expensive

Performance modeling
- saves costs
- Good prediction of necessary infrastructure
- Reusable, but adjustment necessary

Thank you

Questions?