AdaptLoad: effective balancing in clustered web servers under transient load conditions

Alma Riska, Wei Sun, Evgenia Smirni, Gianfranco Ciardo

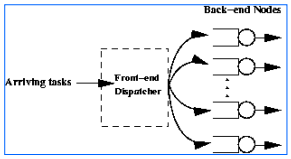
Computer Science Department
College of William & Mary

Motivation

- Internet
  - Too big, too variable, too heavy-tailed
- Issues
  - Flash crowd effect
  - Sudden fluctuations in arrivals and demands
  - $$$
- Capacity planning
  - Critical for E-commerce sites
- High performance & availability
- But mostly cost effective!

Clustered Web Servers

- Clustering with a single system image
- Front-end: level-7 switch
- Back-end: multiple identical nodes



This Talk: Load Balancing

- Classic solutions
  - Random
  - Round-robin
  - Join the Shortest Queue (JSQ)
- Not effective in such environments!

Why not effective?

- Too diverse workloads
- Heavy-tailed behavior
- Short jobs may get stuck behind extra-long ones in the queue
- Effect: HUGE slowdown
- Idea:
  - Separate long from short jobs!
- Size-based policies then…

What is our workload?

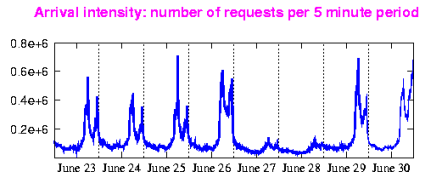
- Must be realistic!!!
- Trace data rather than artificial
- 1998 World Soccer Club
  - 30-low latency platforms
  - 92 days (April 26, 1998 to July 1998)
  - Date & time, size of transferred data
  - Static content

Workload Characterization

- Arrival intensities
  - How often
  - How variable
  - Possible periodic behavior
- Service intensities
  - Assumption: service time linear to file size
  - How variable

Workload Characterization: Arrival Process

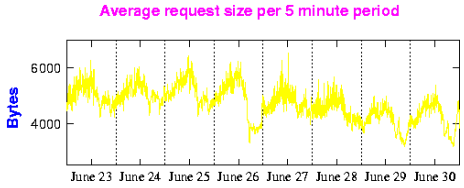
- Look at one busy week…



- High variability
- Clear periodicity

Workload Characterization: Service Process

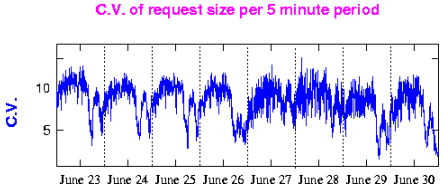
- Keep looking at the same week…



- High variability

Workload Characterization: Service Process (cont.)

- C.V. (i.e., standard deviation / mean)

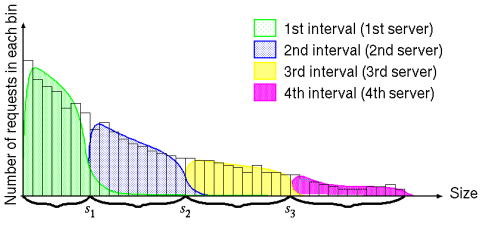


Workload Summary

- Rapidly changing environment
- Very wide variability in both arrivals and service times
- Load balancing more tricky!
- Challenges
  - Adapt balancing parameters
  - Ensure equal load on all servers

Our solution: AdaptLoad

- Basic idea: tasks of similar sizes to the same server



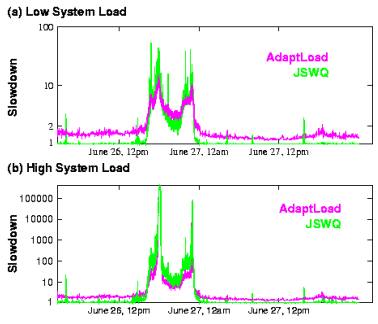
AdaptLoad (cont.)

- Problem
  - A priori knowledge of the workload
- Solution
  - Use past to predict future
  - On-line observations of limited number of requests
- Evaluation
  - Trace driven simulation

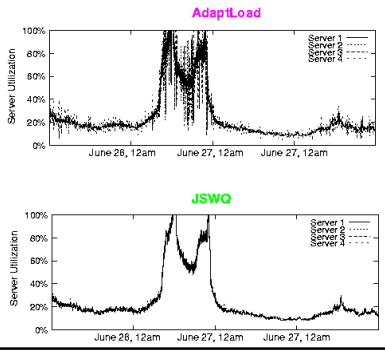
Performance Issues

- Transient overloads
- Fast and slow servers
- Equal load (utilization)
- Fairness
- Scalable
- Improvements?

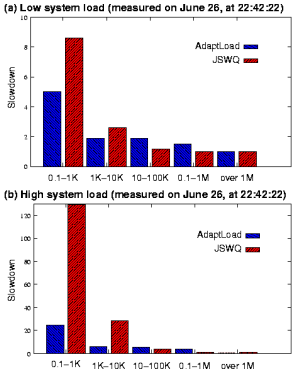
Transient Overloads



Server Utilization



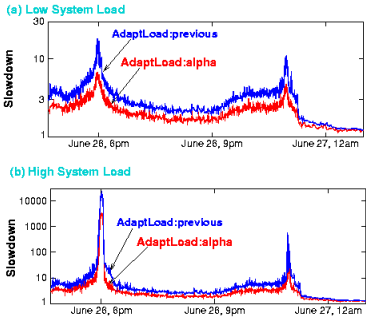
Fairness: per class slowdown



Improvements?

- Better knowledge?
- Exponentially discounted history
- Two parameters
  - Batch size K
  - Coefficient 0 <= a < =1
- Exhaustive search for (K,a)
- Sensitivity?
- Robustness?

Improved AdaptLoad



Summary

- New policy: AdaptLoad
- Simulation-based evaluation
- Use history for future prediction
  - –Previous K requests
  - –Exponentially discounted history
- Works great!

Future Directions?

- Service differentiation
- Dynamic content
- Time-series analysis
- On-line analytic models
- Prototype implementation