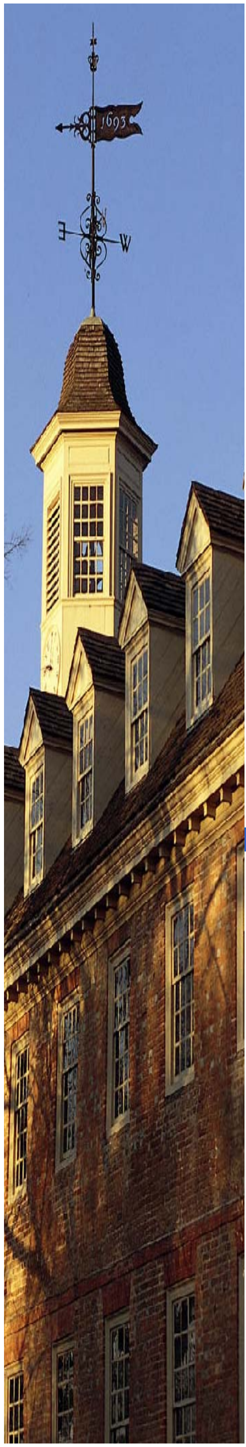


New Results on the Performance Impacts of Autocorrelated Flows in Systems

Evgenia Smirni

The College Of WILLIAM & MARY

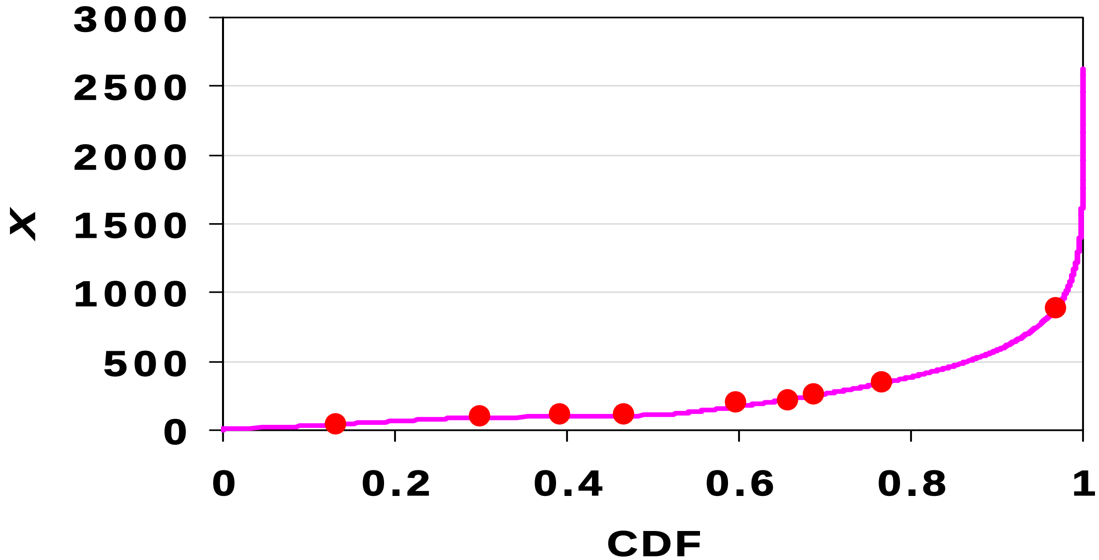
What is autocorrelation?

- Burstiness
- Self-similarity
- Dependence (short-range, long-range)
- Well-studied in networking

- Systems?
  - Some early studies in storage systems (HP traces from the early 90s)
  - Recently (USENIX'06) from Seagate

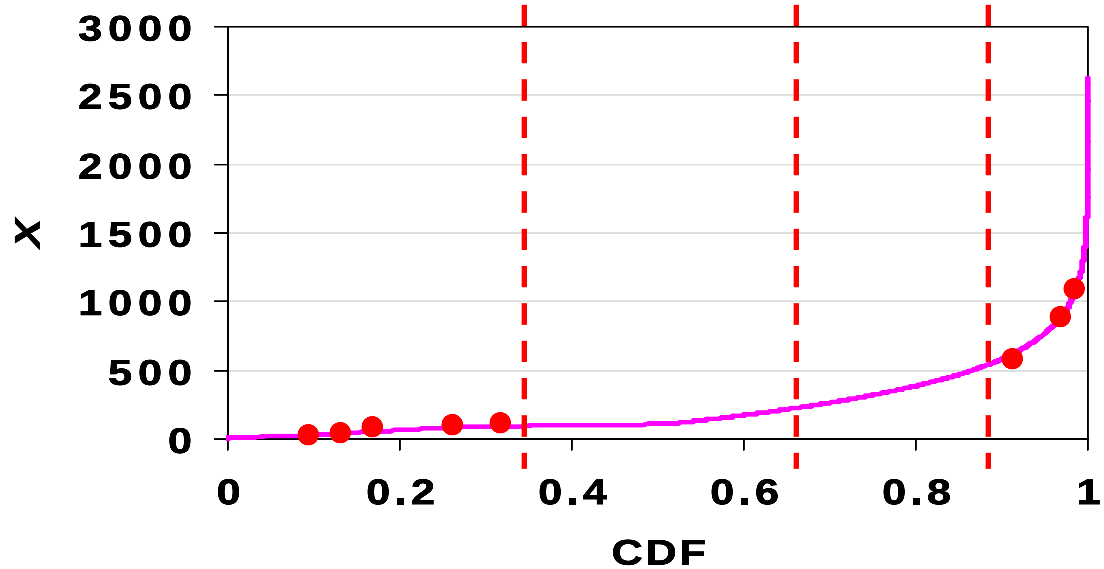
WILLIAM & MARY

Dependent process (example)



- Independent process

Dependent process (example)



- Independent process

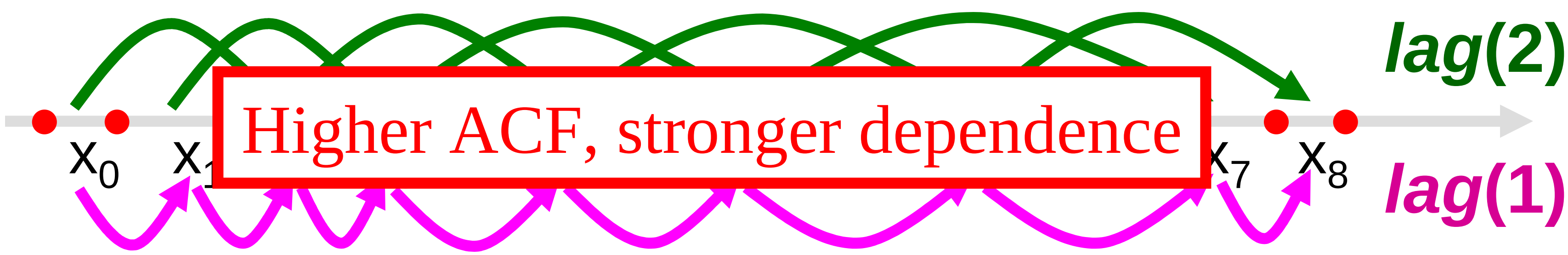
- Dependent process

WILLIAM & MARY

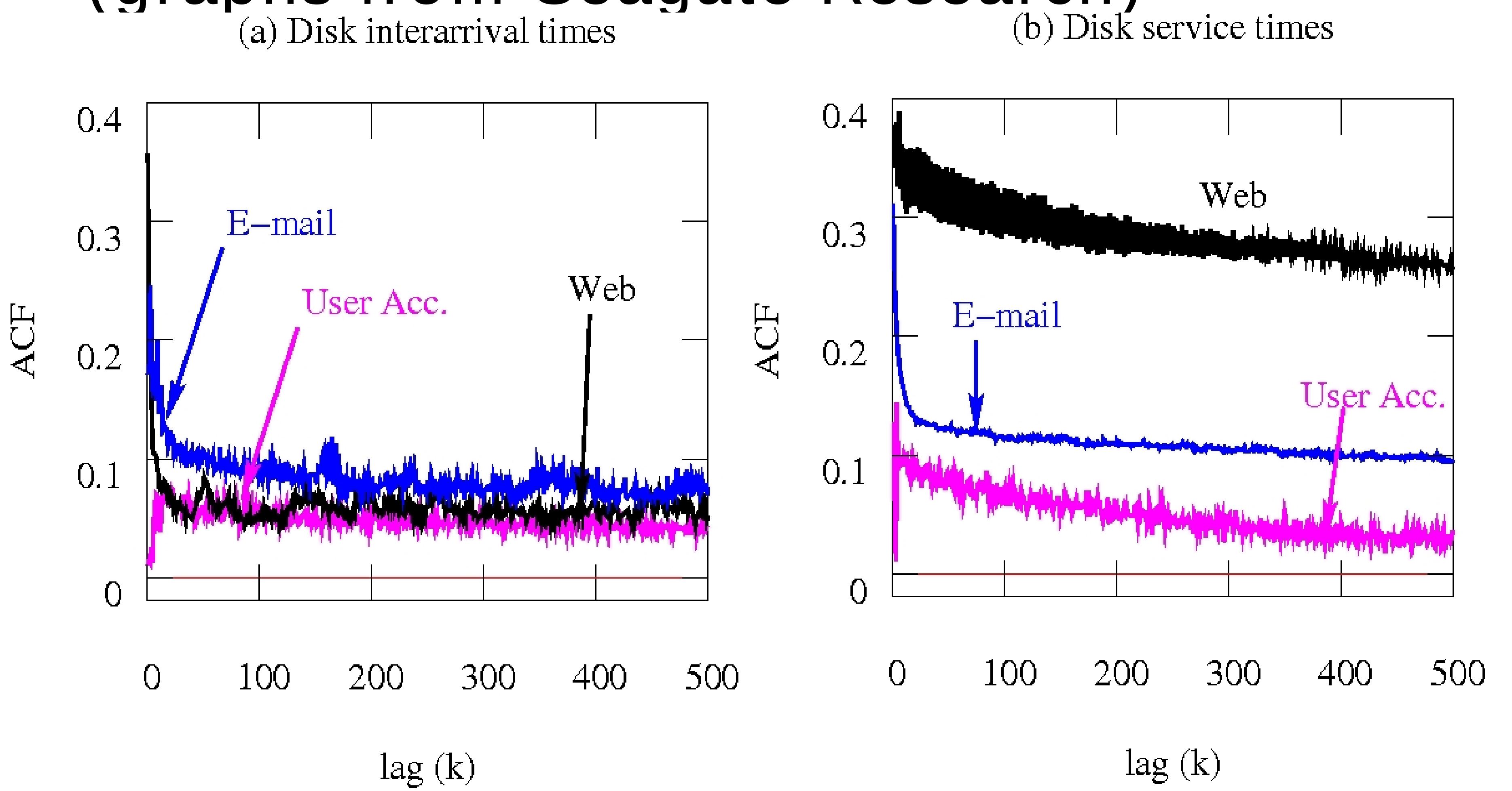
Dependence Metrics

- Autocorrelation function (ACF) of a process $\{X_0, X_1, X_2, X_3, \ldots\}$ with lag k

$$\mathrm{corr}[X_0, X_k] = \frac{E[(X_0 - E[X])(X_k - E[X])]}{\mathrm{Var}[X]}$$



Dependence in Storage Systems (graphs from Seagate Research)



WILLIAM & MARY

Summary of Results

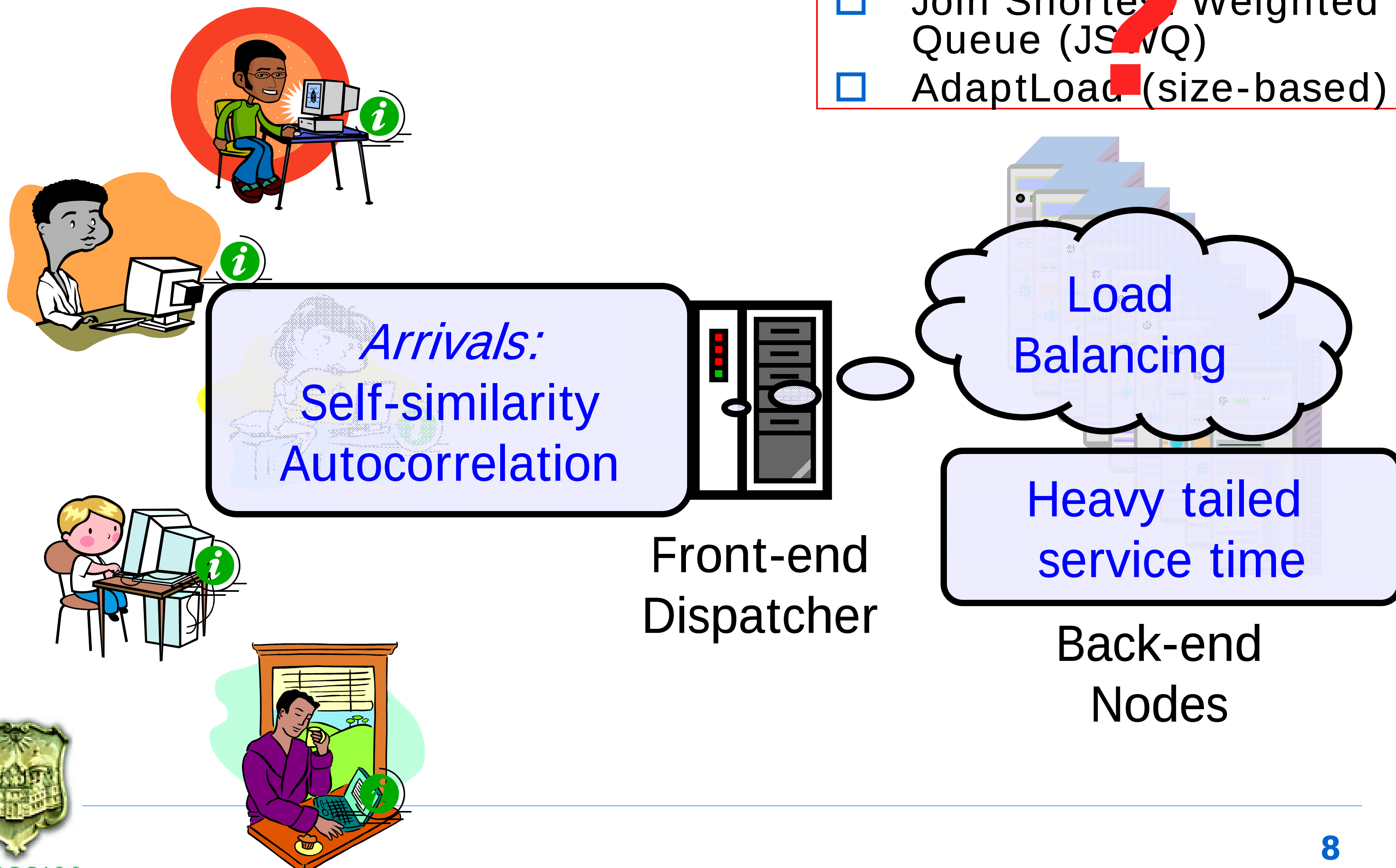
- Open Systems
  - Load balancing under autocorrelated **arrivals**
    - Load unbalancing as a solution
- Closed Systems
  - Multi-tiered systems (TPC-W)
    - **Service process** can be autocorrelated
    - Autocorrelation propagation
  - Impact of autocorrelation
- On-going work

WILLIAM & MARY

Clustered Servers

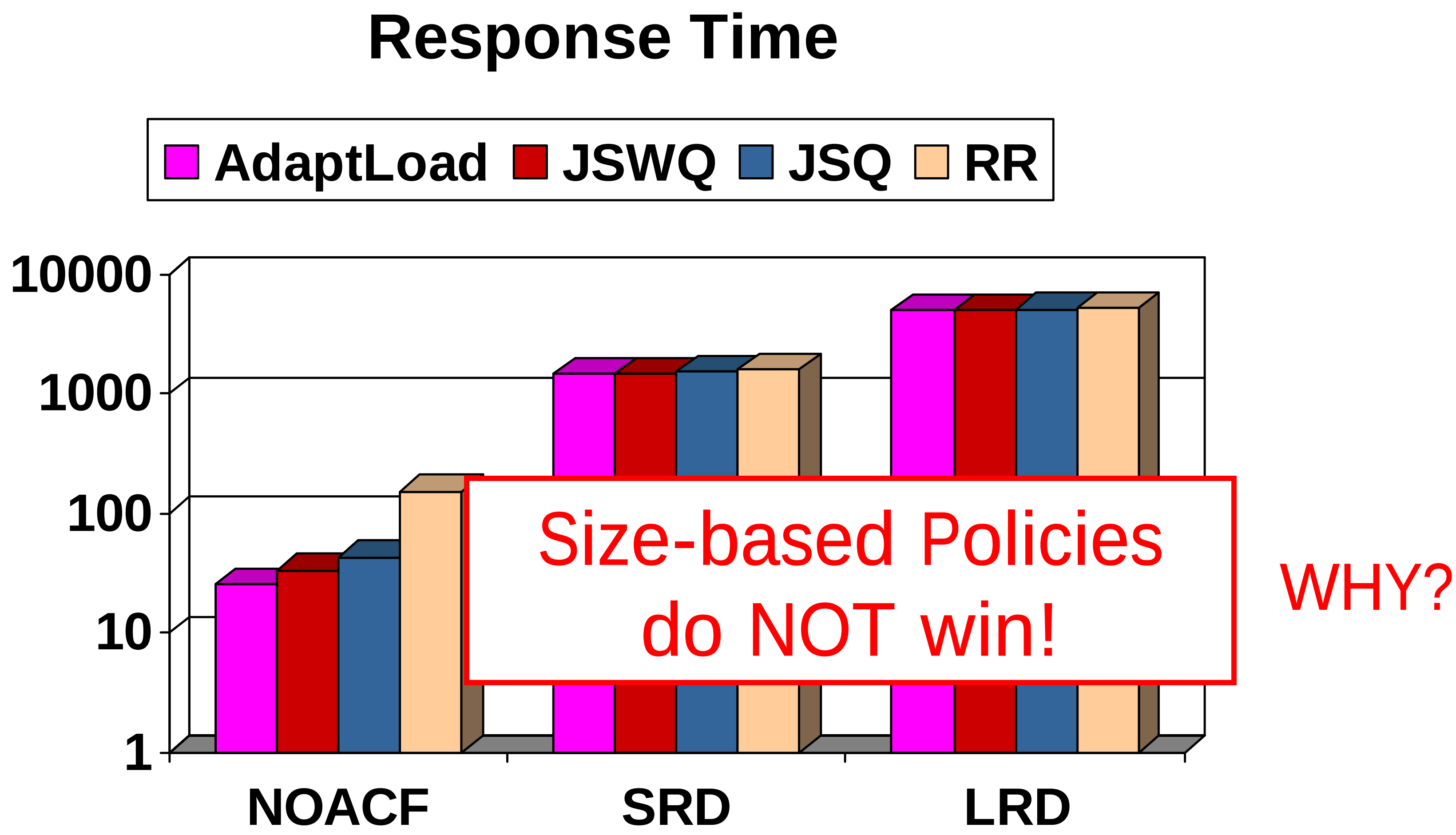
- Round Robin (RR)
- Random
- Join Shortest Queue (JSQ)
- Join Shortest Weighted Queue (JSWQ)
- AdaptLoad (size-based)

Performance ?

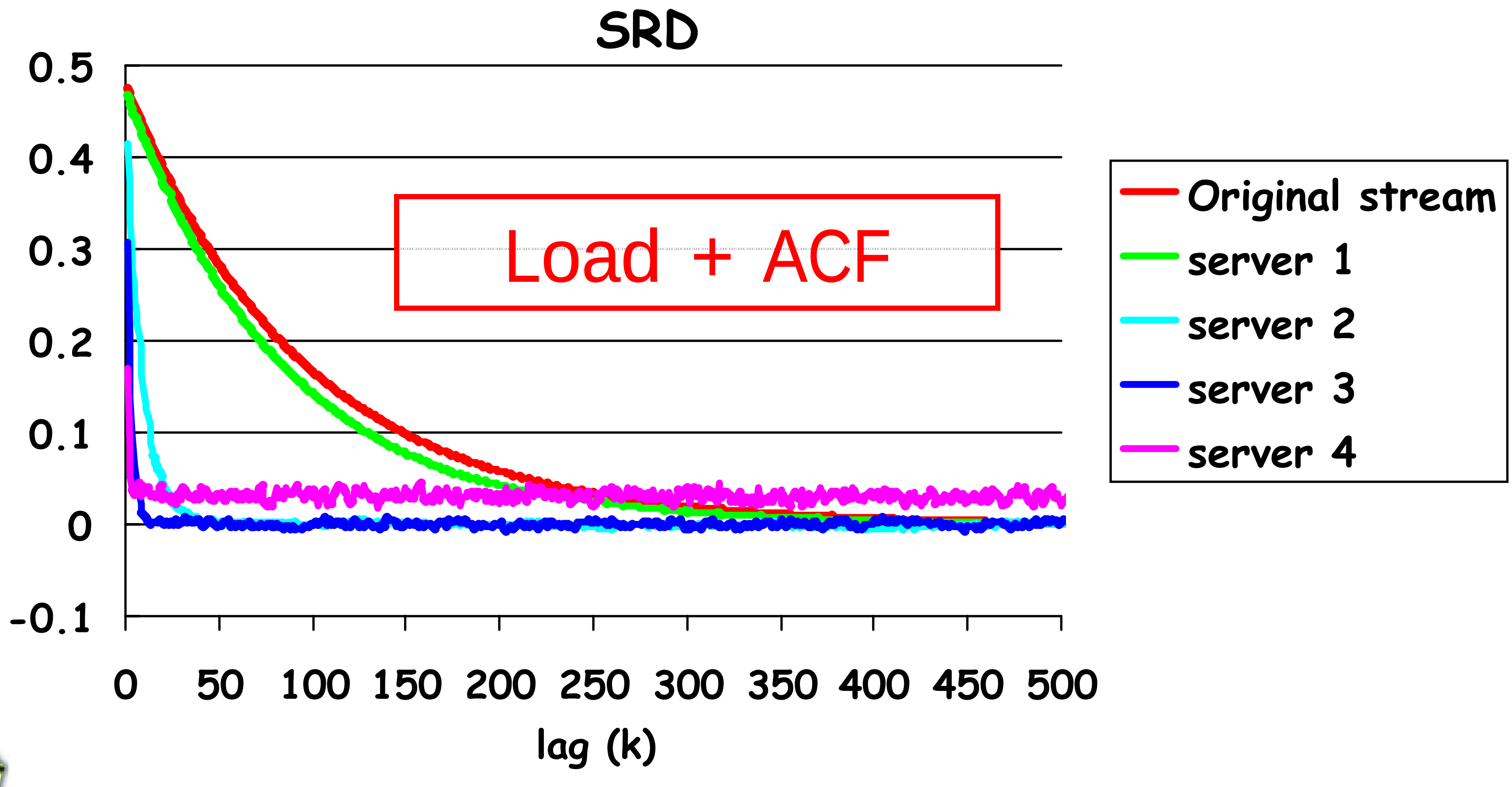


WILLIAM & MARY

Effect of ACF on Load Balancing



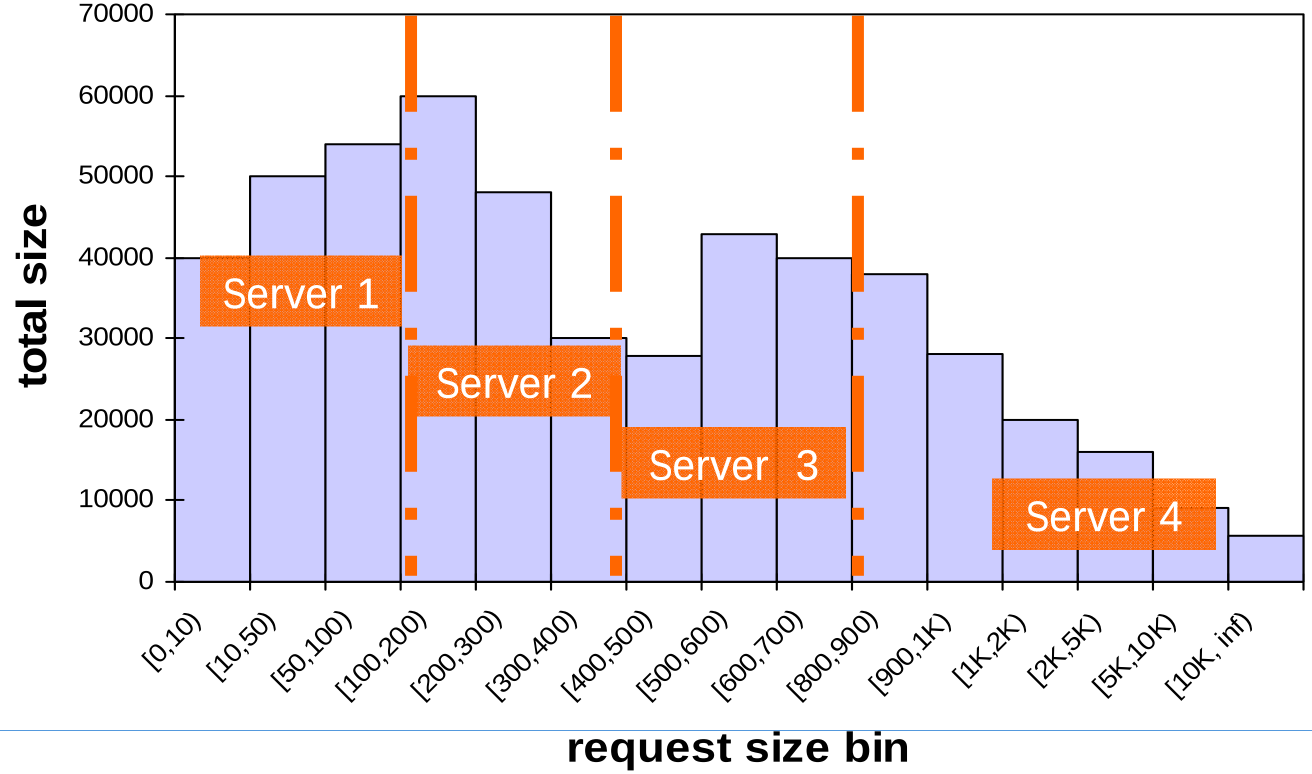
Possible Reason …
What is ACF in Each Node?



WILLIAM & MARY

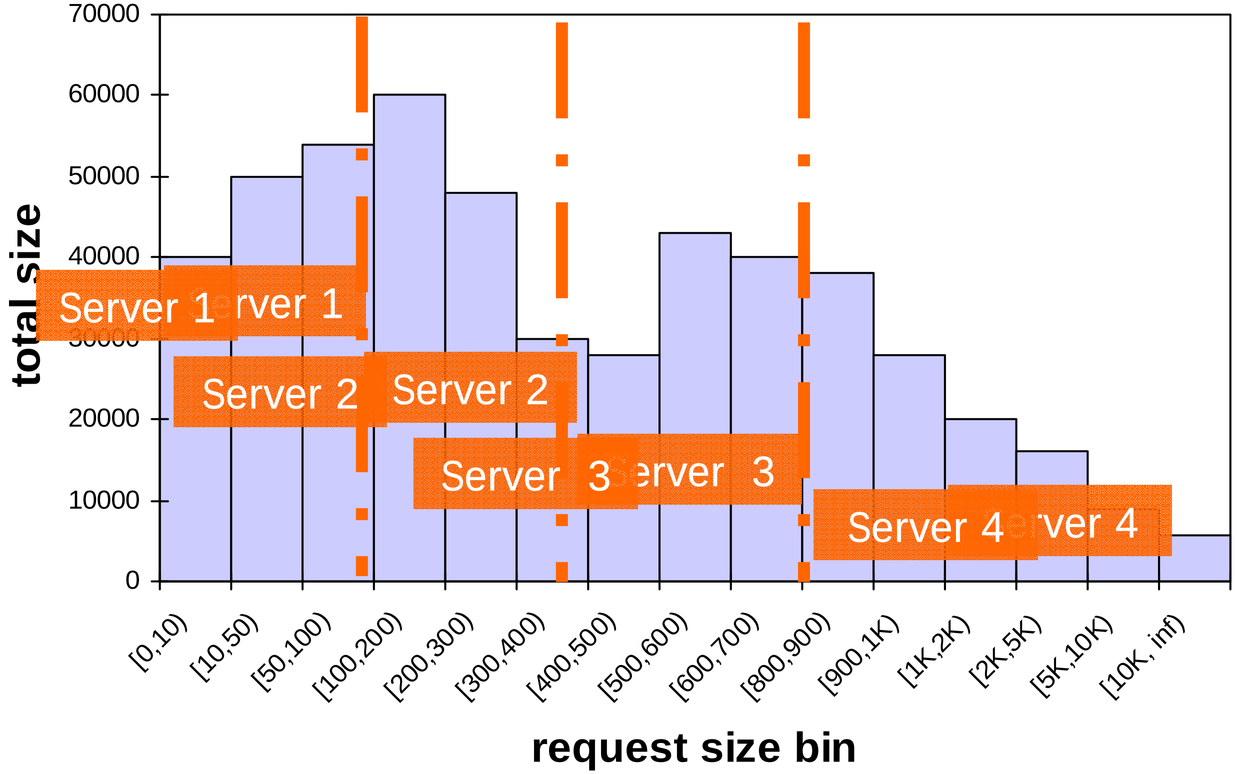
Review: AdaptLoad (Size-based)

Step 1: Build histogram on-line
Step 2: At the end of monitoring window, find the boundaries to partition the total work (area) equally

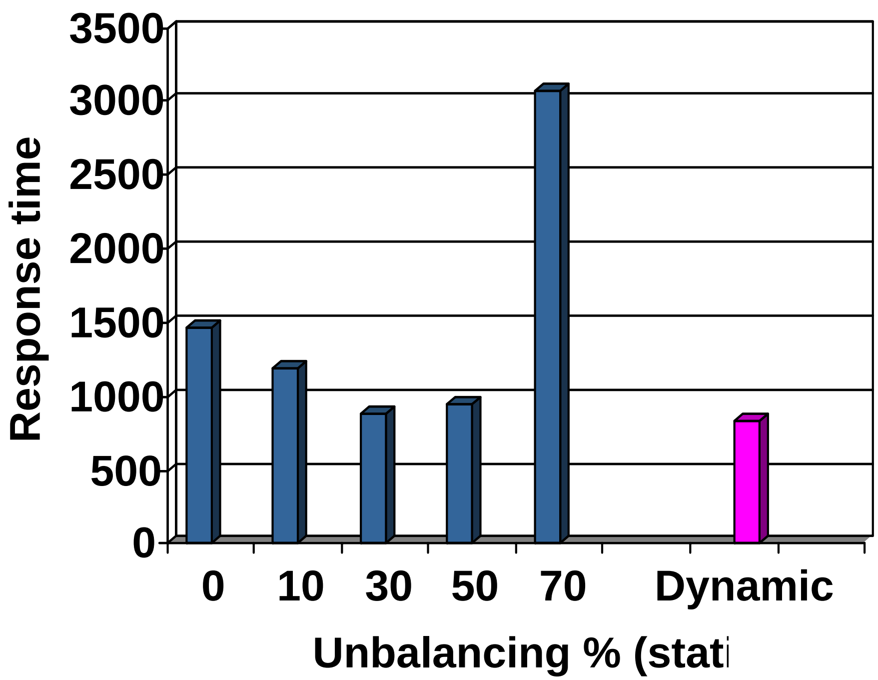


Load Unbalancing

- Server *i* increase p_i of its work
  - Static version
  - Dynamic version



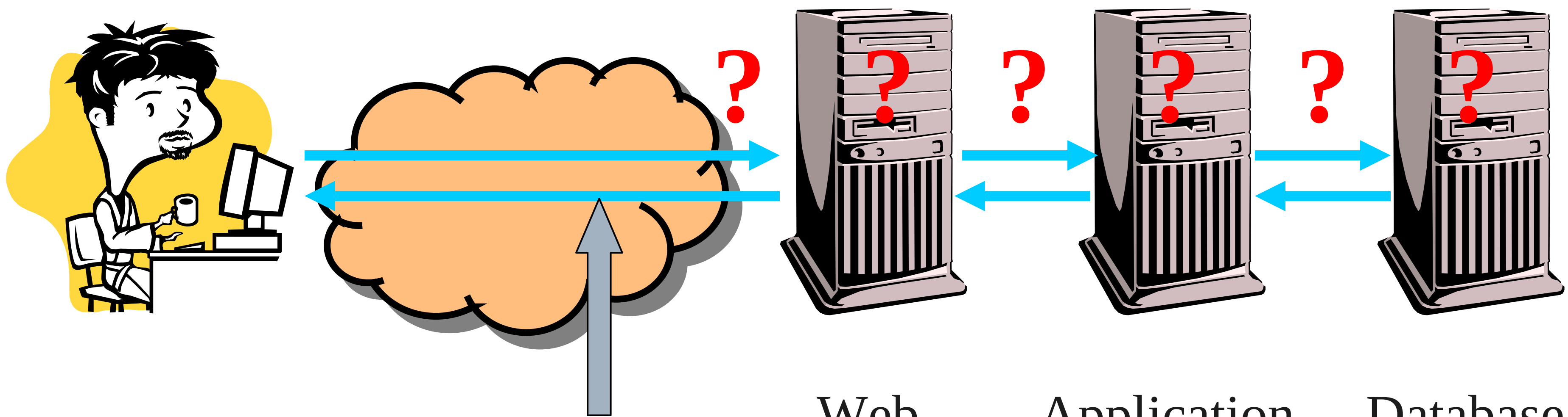
Performance of Unbalancing



Closed Systems: Three-tiered architecture

- Experiments
- Analytic models
- Policy Development

Autorrelation



Web Server

Application Server

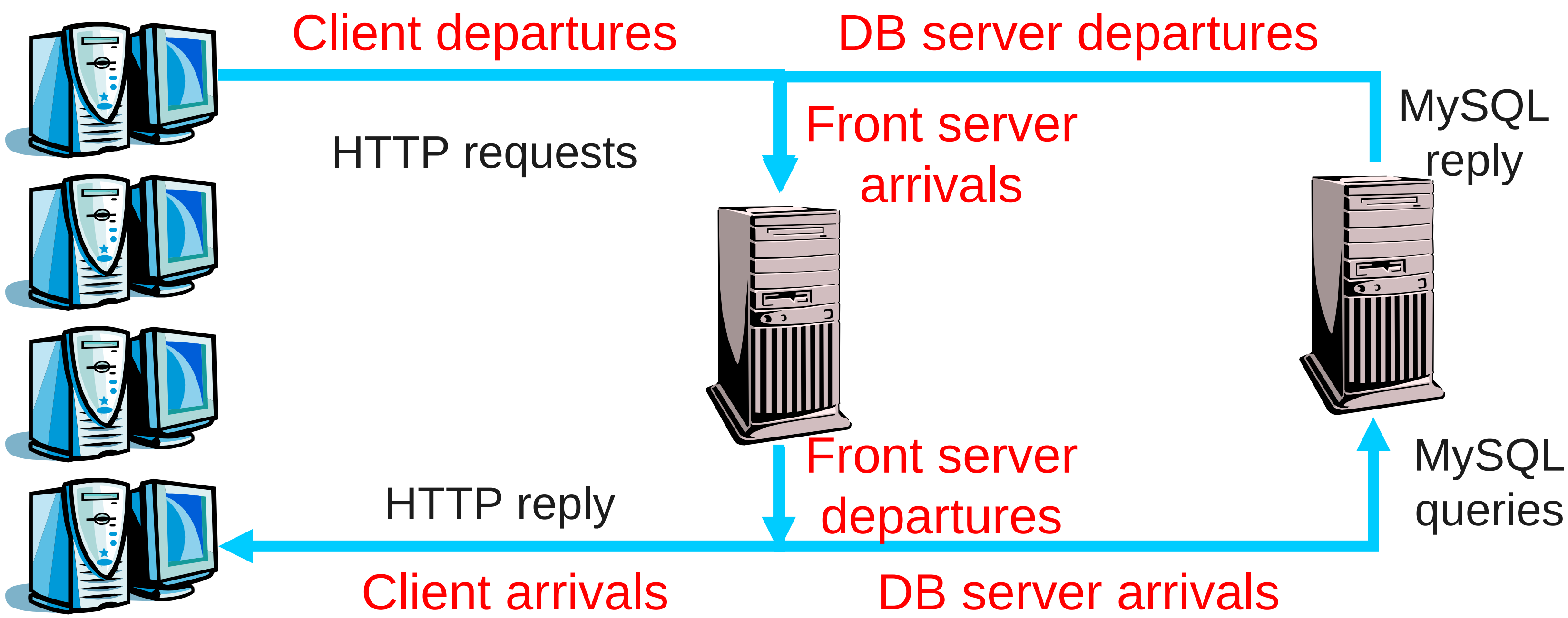
Database Server

Highly Correlated (Bursty) Arrivals

[related work]

WILLIAM & MARY

Multi-tiered E-commerce Site Set-up



Clients
(EB)
P4/2GHz
256MB

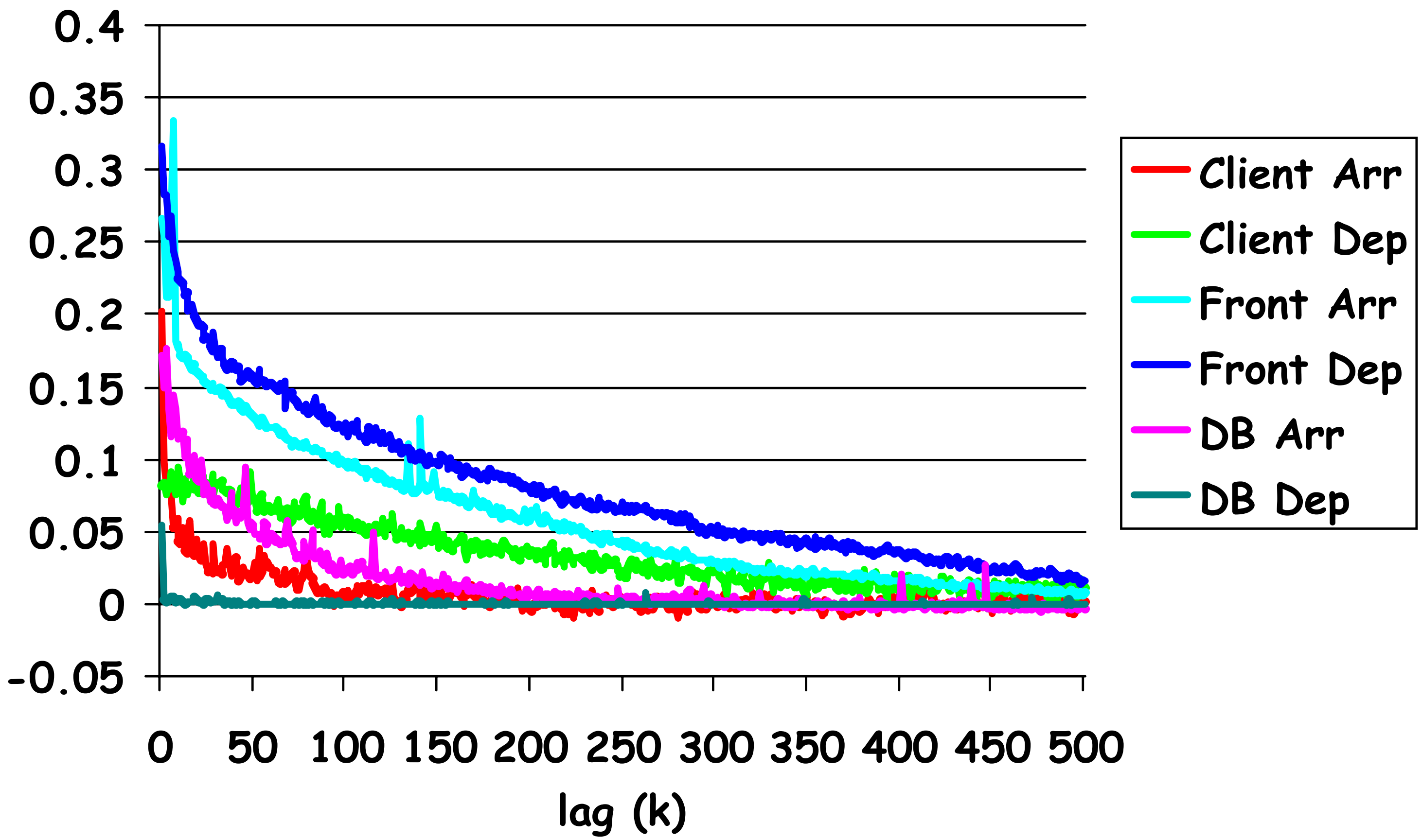
Front Server
(Apache/Tomcat)
P3/1.3GHz
2GB

DB Server
(MySQL 4.0)
Dual Xeon/1.5GHz
768MB

WILLIAM & MARY

ACF Propagation

Browsing mix, 10K DB, 384 EBs



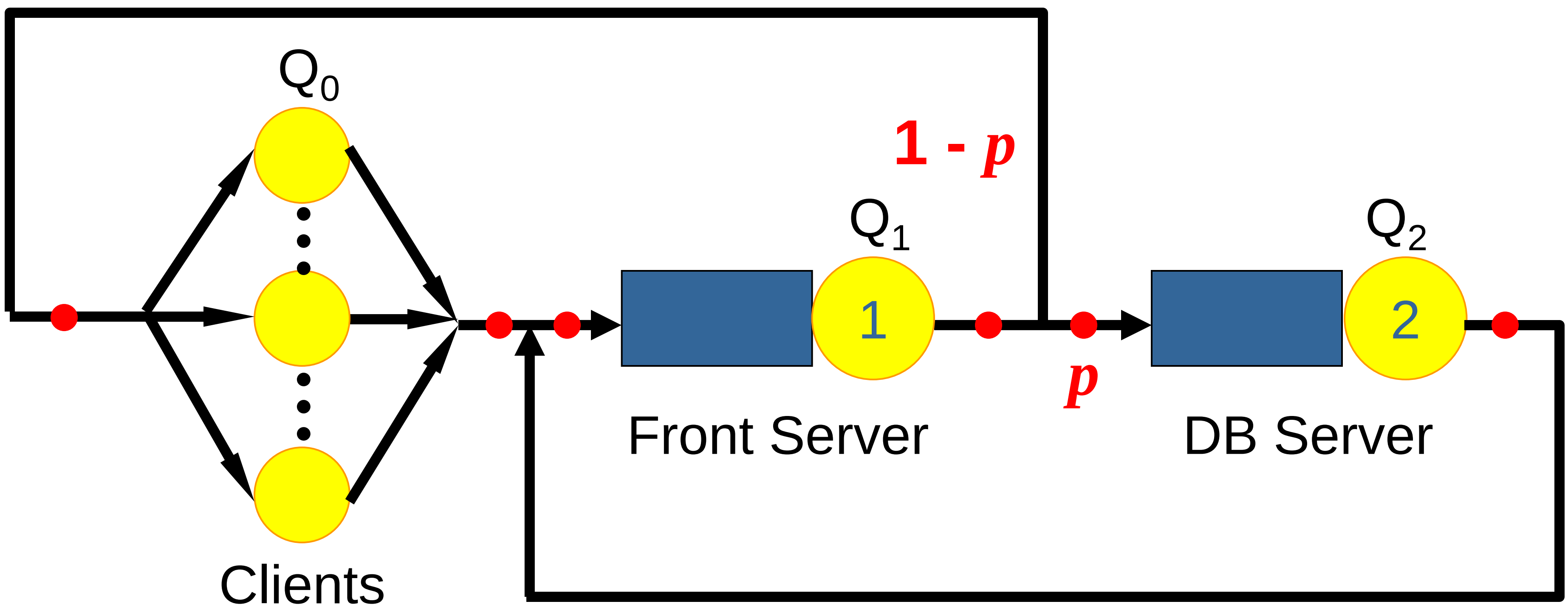
Observations

- Dependence in service processes
- Dependence in lower tiers affects the arrival process to the higher tiers
  - but no dependence in the process of session generation !
- ACF propagation in all tiers

Confirm the observations by an analytic model

WILLIAM & MARY

TPCW Model



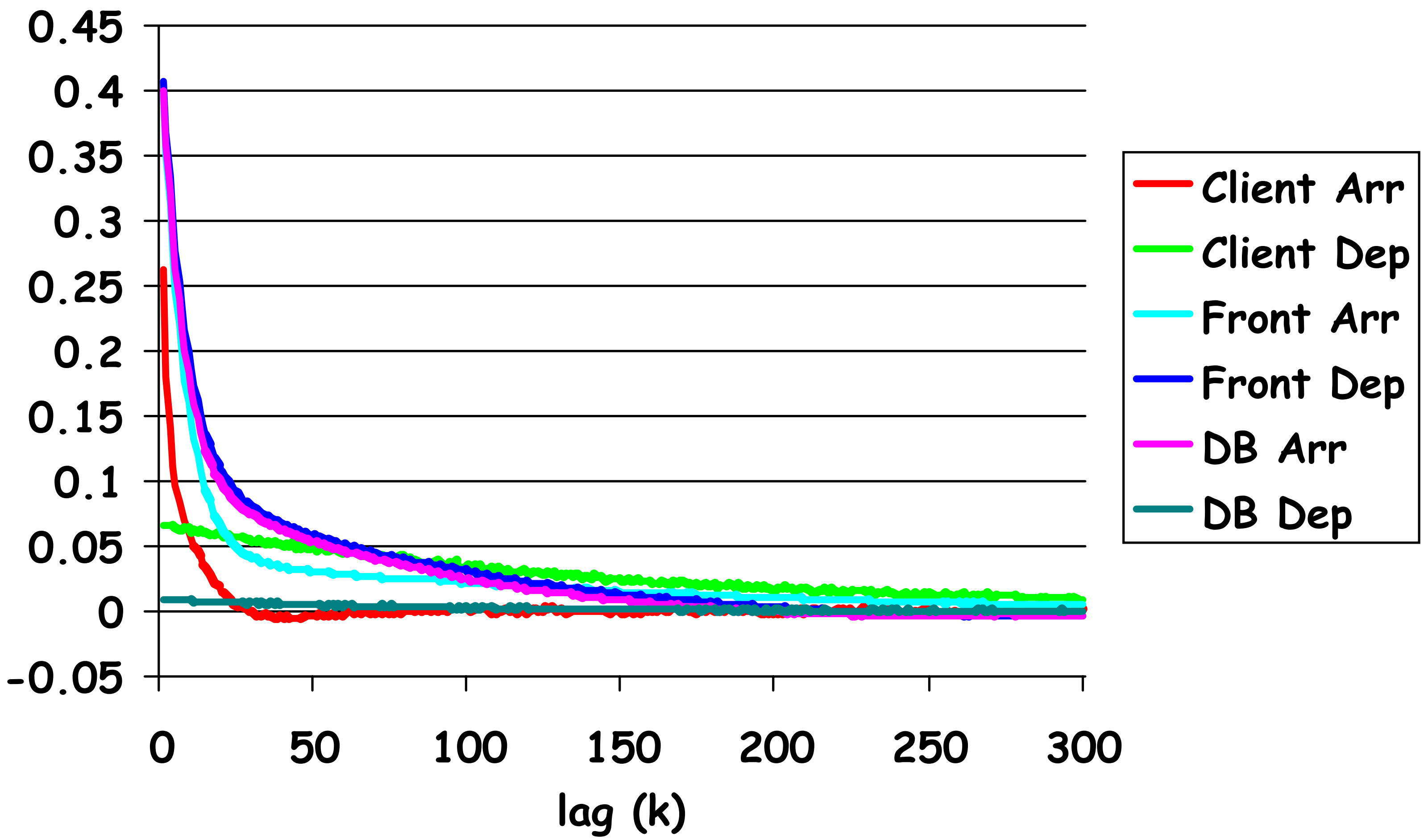
Q_0: Exponential distribution

Q_1: Correlated MMPP process, high variance

Q_2: Non-correlated Hyperexponential, high variance

WILLIAM & MARY

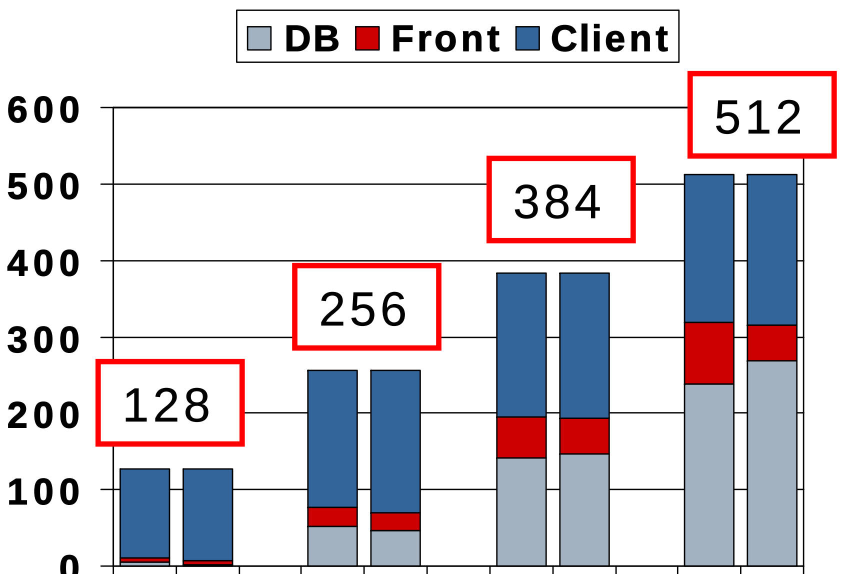
ACF Propagation -- 384 MPL



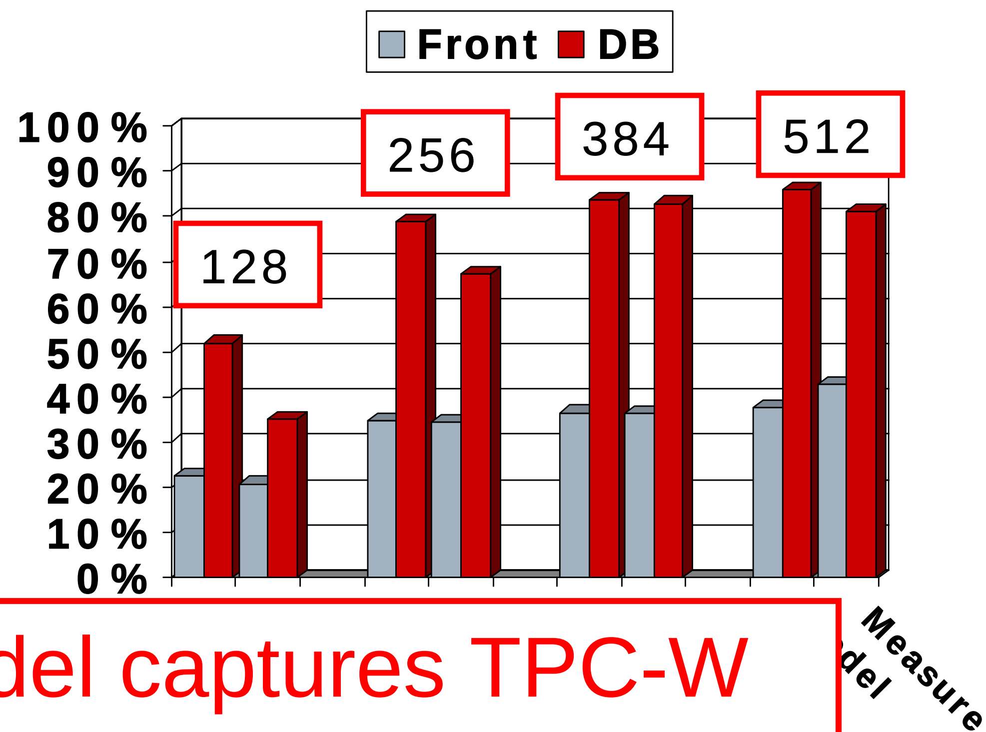
WILLIAM & MARY

Performance Comparison

(a) Average queue length



(b) Average utilization



This simple model captures TPC-W behavior qualitatively.

WILLIAM & MARY

Detailed analysis: counter-intuitive results

- Autocorrelation propagates in the entire system and has serious performance impact
  - **"Balances" the load of queues**
  - **Bottleneck utilizations decrease**
  - **System throughput decreases**
  - **Cyclic bottleneck switch**
- Overload (*VERY long response times)* can happen under medium load if dependence exists
  - Dependence should be considered in capacity planning
  - Tails do not necessarily come from the bottleneck server

WILLIAM & MARY

On-going work

- Use autocorrelation to model caching/locks/memory hierarchy: very compact model
  - Trace fitting into processes that capture autocorrelation
- Theory
  - Closed systems (i.e., multi-tiered example)
    - New analytic models for non-product form networks that can support autocorrelated processes
    - Approximation methods
    - Capacity planning
  - Open systems: Departure process
- Policy development/Scheduling
  - QoS policies
  - Storage systems to schedule foreground/background jobs
  - General scheduling policies with minimum information

Acknowledgements

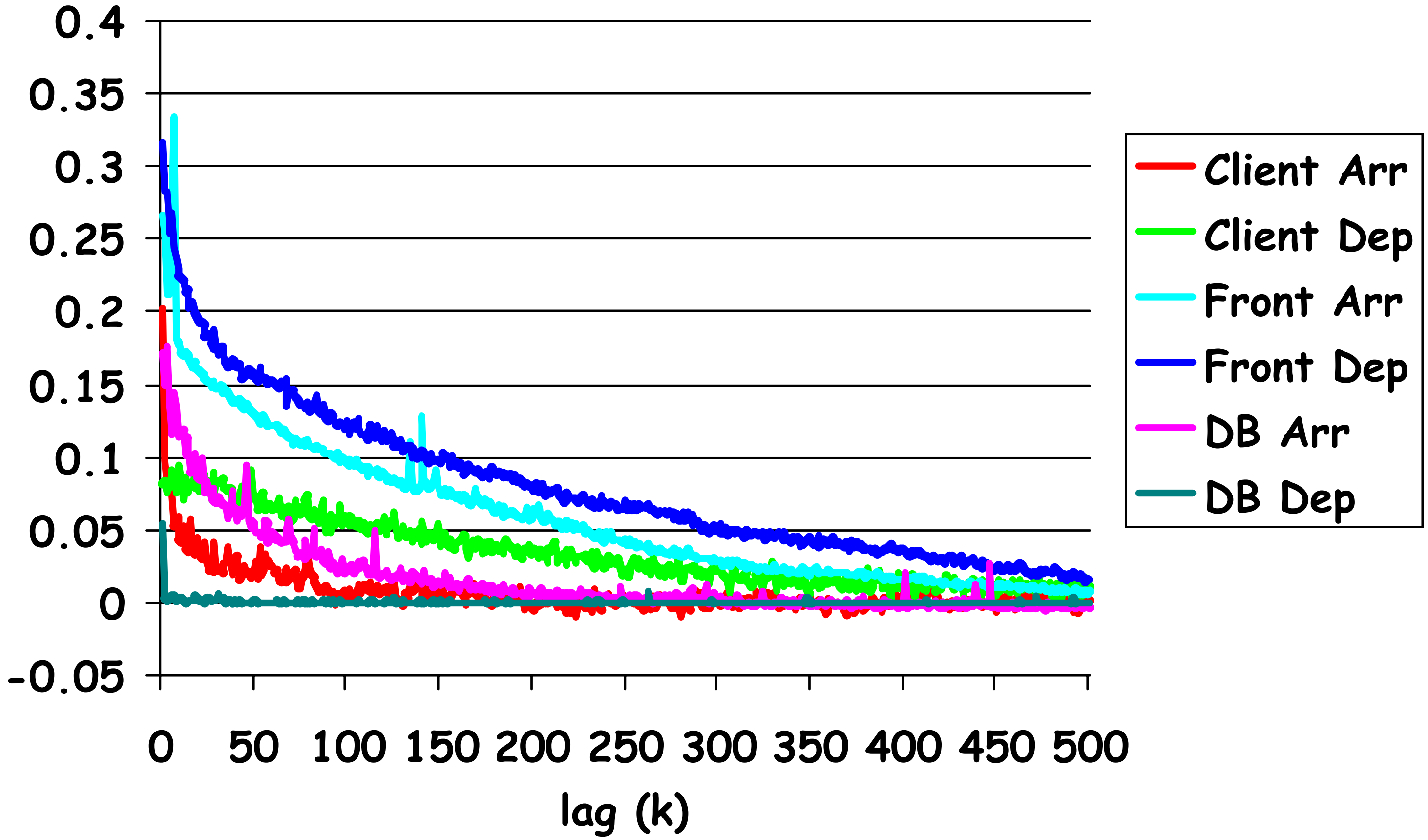
- Students
  - Qi Zhang (just graduated, now at Microsoft)
  - Ningfang Mi
  - Zheng Zhang
- Collaborators
  - Alma Riska and Erik Riedel (Seagate Research)
  - Lucy Cherkasova (HP Labs)
  - Giuliano Casale (postdoctoral associate)

- More information (several papers)
  http://www.cs.wm.edu/~esmirni

WILLIAM & MARY

ACF Propagation

Browsing mix, 10K DB, 384 EBs

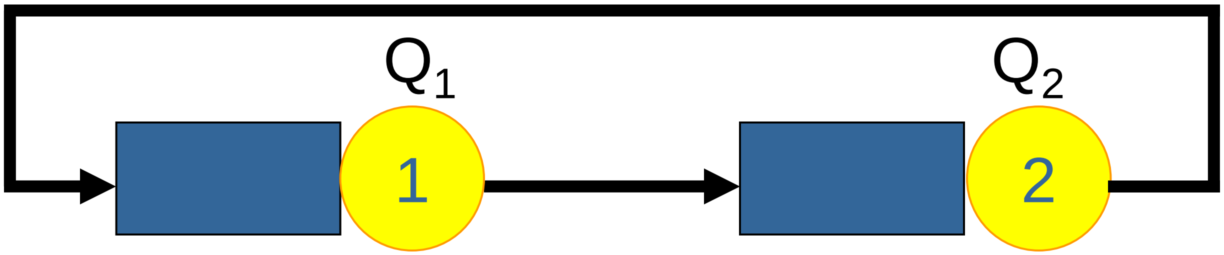


Dependence in Service Process

- Collected traces across tiers
  - Calculate ACF off-line
- Thinking time – exponential distribution
  - No ACF
- Service process in each server
  - Hard to obtain by measurements
  - Observing dependence in arrival and departure processes
  - Existence of dependence in service process
    - Increase of ACF for small lags
- DB server is the bottleneck

WILLIAM & MARY

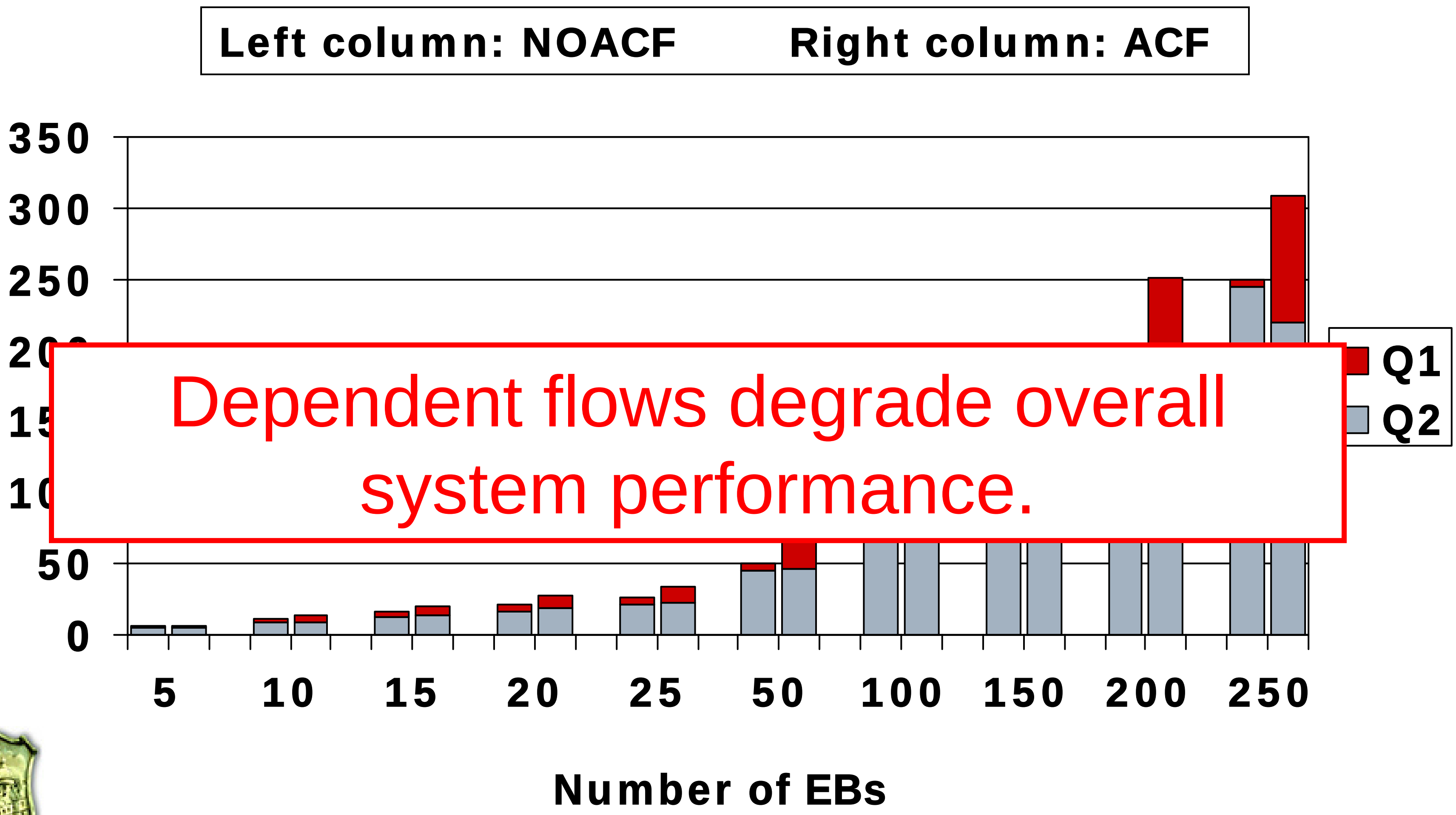
Comparison



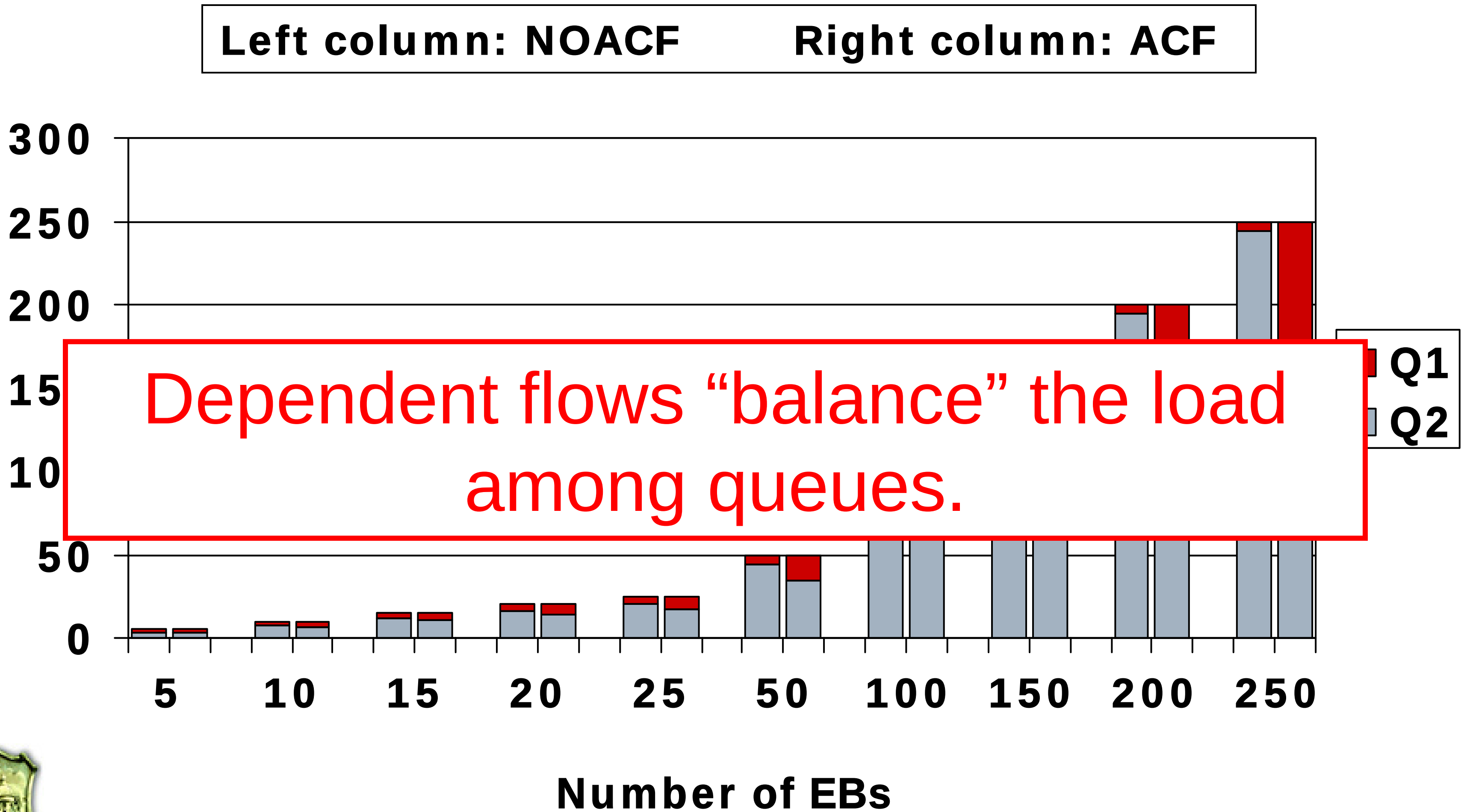
- Comparison with independent services
- same moments
  - mean, cv and higher moments

	Q_1	Q_2 (bottleneck)
ACF	Dependent	Independent
NOACF	Independent	Independent

Performance Comparison
(a) Average round-trip time

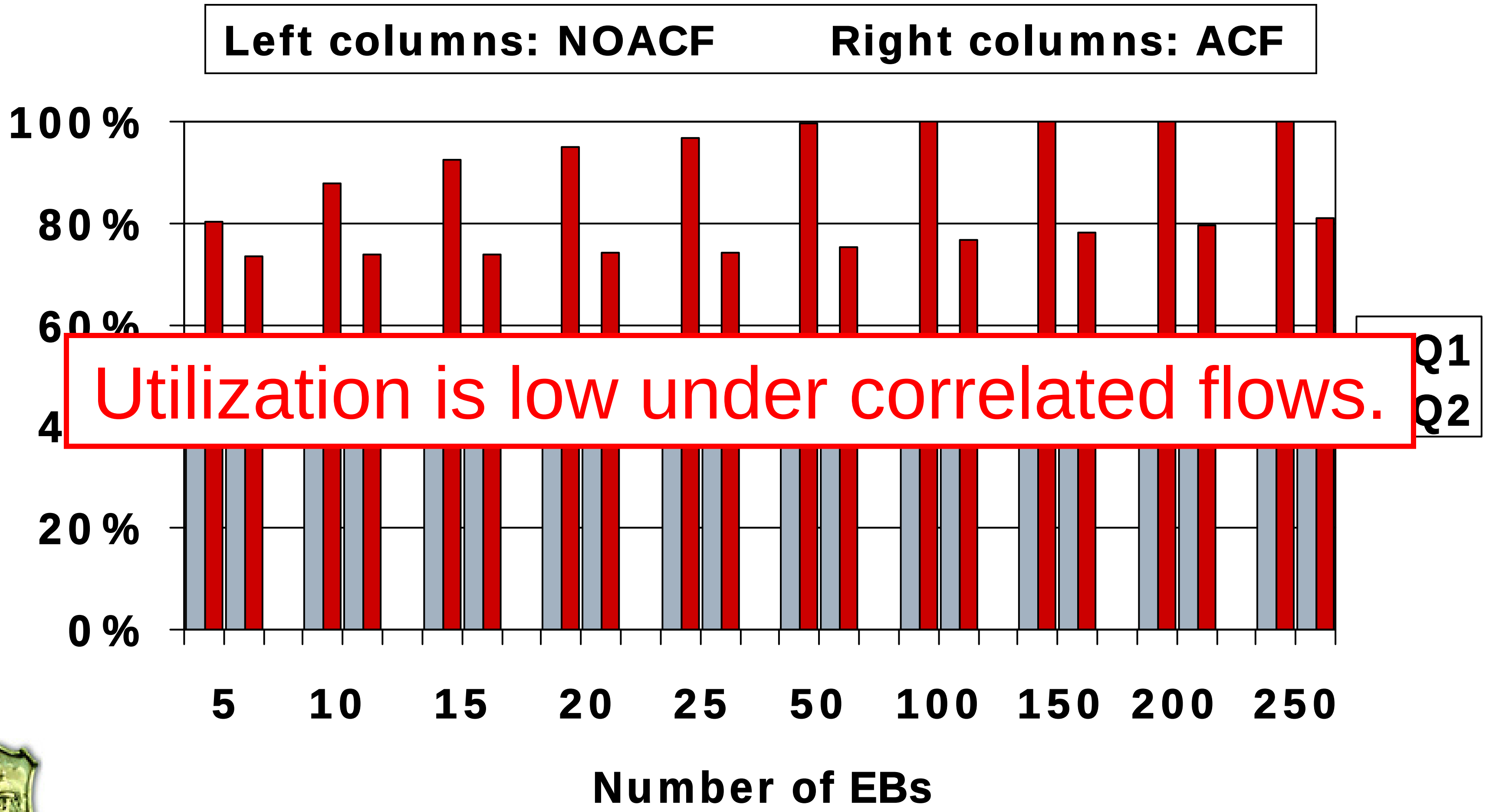


Performance Comparison
(b) Average queue length



WILLIAM & MARY

Performance Comparison
(c) Average utilization



Observations

- Dependence has significant effect on system performance
- ACF propagates into *all* tiers
- Overload (*VERY long response times)* can happen under medium load if dependence exists
  - Dependence should be considered in capacity planning
  - Tails do not necessarily come from the bottleneck device

WILLIAM & MARY

Summary

- Workload characterization in multi-tiered closed systems
  - ACF propagates into all the tiers
    - Exists in storage systems
    - But also other parts (e.g., cache behavior, memory pressure)
  - Overload can happen under medium load if dependence exists
    - Tier with ACF affects performance a lot (although not bottleneck)
    - Cyclic bottleneck switch (very tricky!)
  - Classic analytic modeling techniques do not apply
    - e.g., MVA or approximation methods
  - Yet, simple models that capture ACF in service process capture trends

- Policy development
  - ACF-aware load balancing policy for cluster with dependent flows

WILLIAM & MARY

Dynamic Policy: D_EQAL

- *R* is initialized as 0
- Adjust *R* for a small value *Adj* at the end of each monitoring window
- The adjustment should improve both slowdown and response time
- If not, wrong direction

WILLIAM & MARY

TPC-W Specifications

- On-line book store Web site
- 14 Interactions

  (browsing-based vs. ordering-based)
  - Browsing mix (95% vs. 5%)
  - Shopping mix (80% vs. 20%)
  - Ordering mix (50% vs. 50%)
- Databases (different number of items)

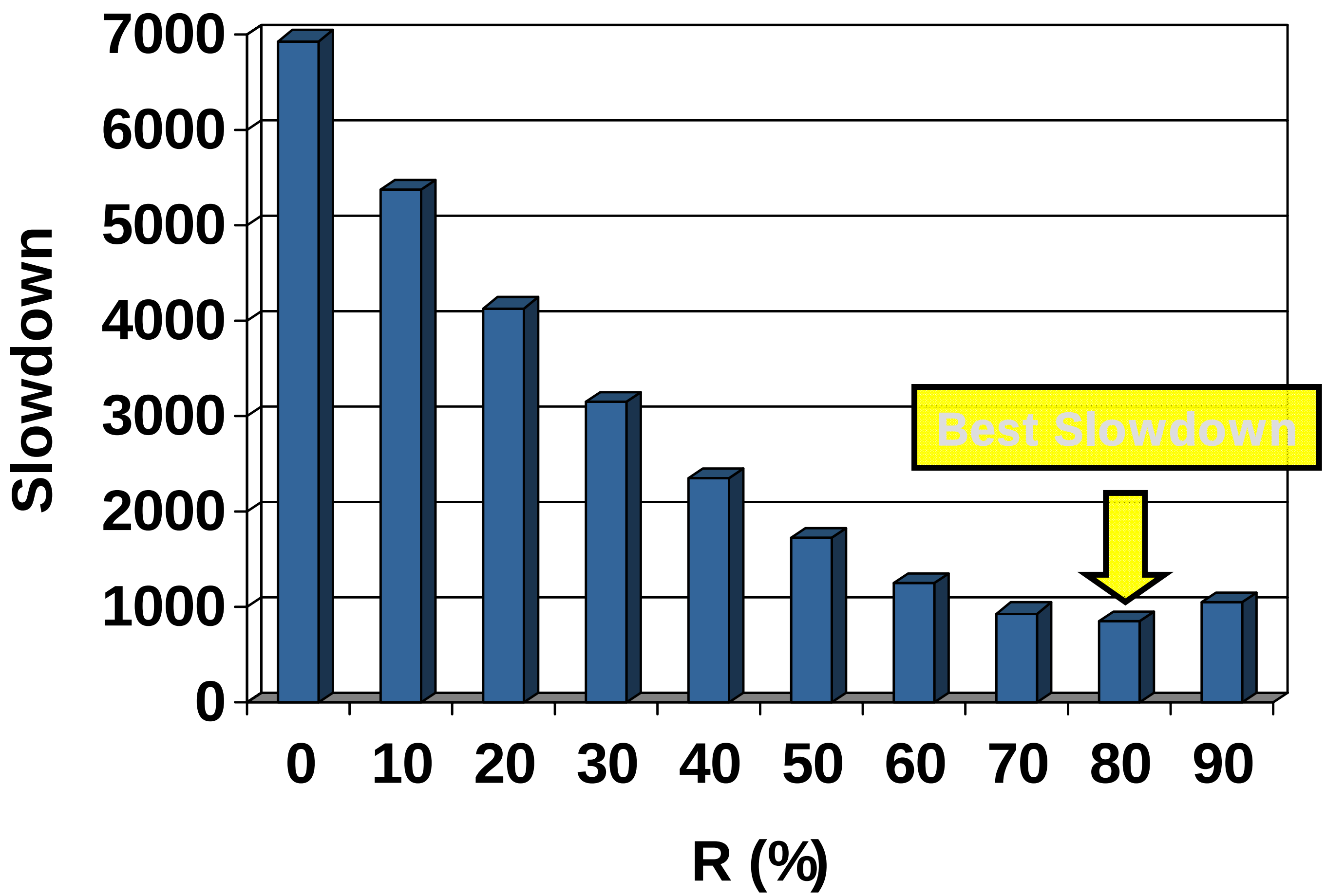
# Items	10K	100K	500K	1M
DB size	1.5GB	1.5GB	1.9GB	2.1GB

WILLIAM & MARY

Performance of S_EQAL

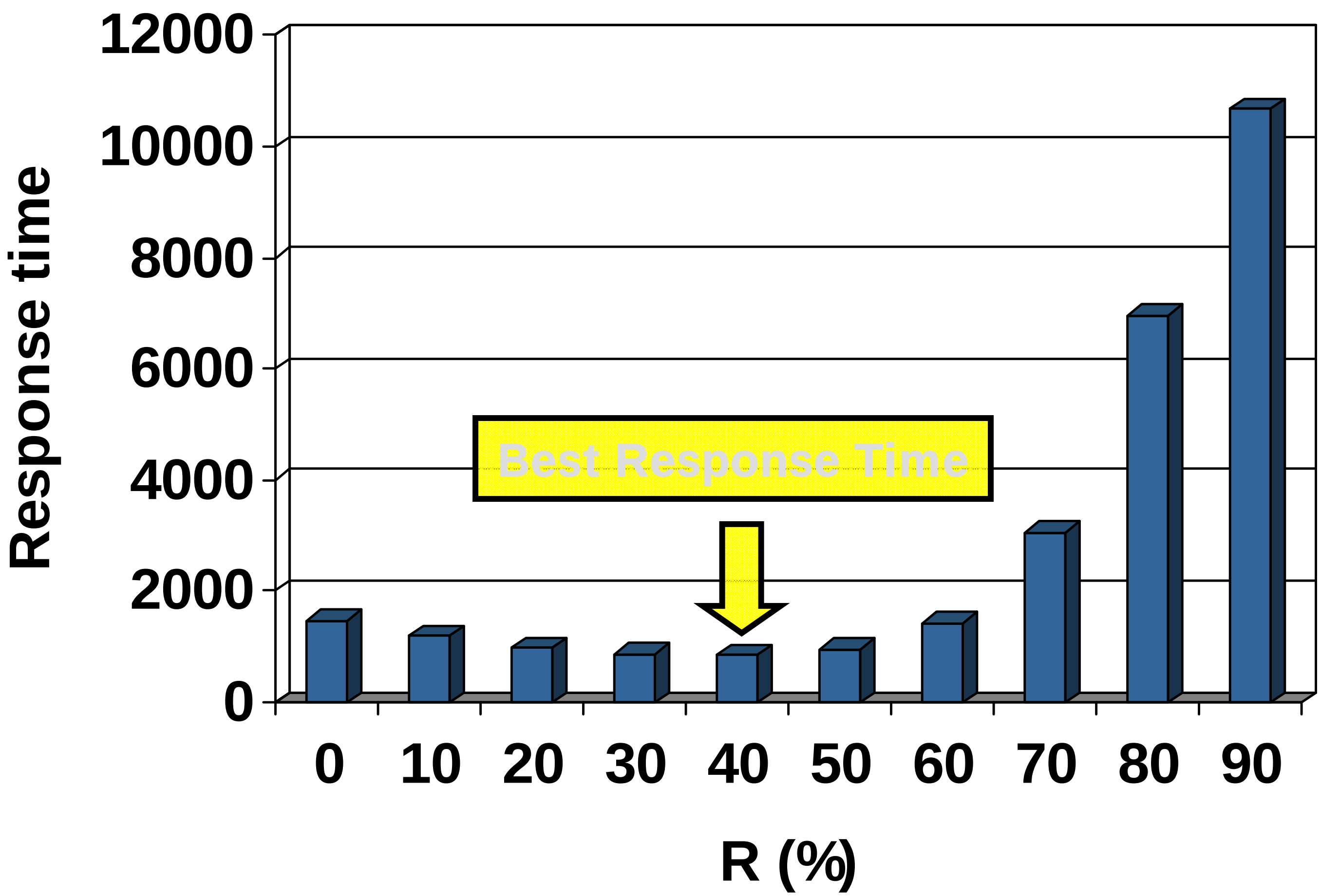
- Service time: WorldCup 1998 Trace
- Inter-arrival time: MMPP(2)
  - Same moments
  - With short range dependence (SRD)
- 4 servers in the cluster
- Average utilization per server: 62%

Average Slowdown by *R*

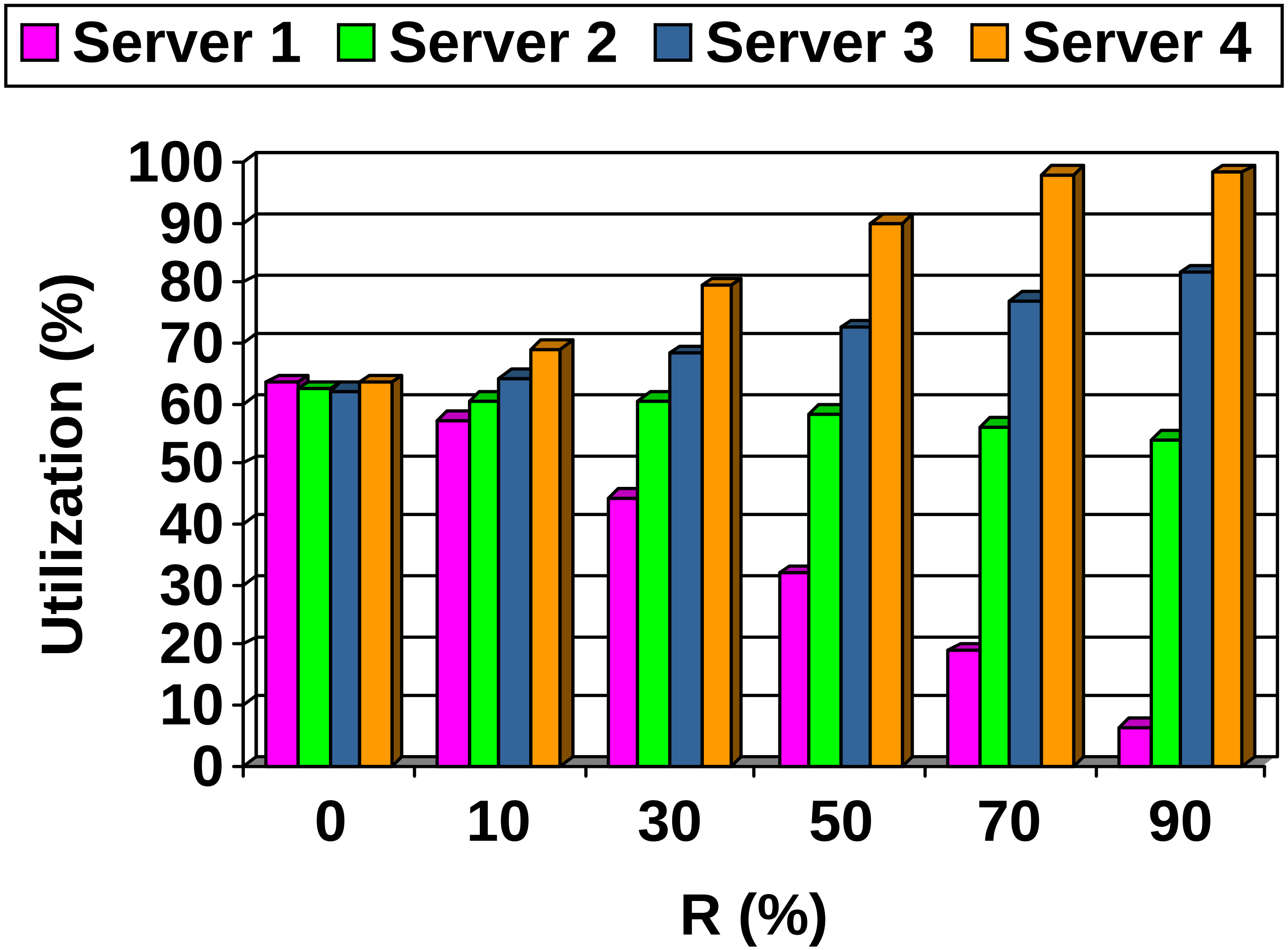


WILLIAM & MARY

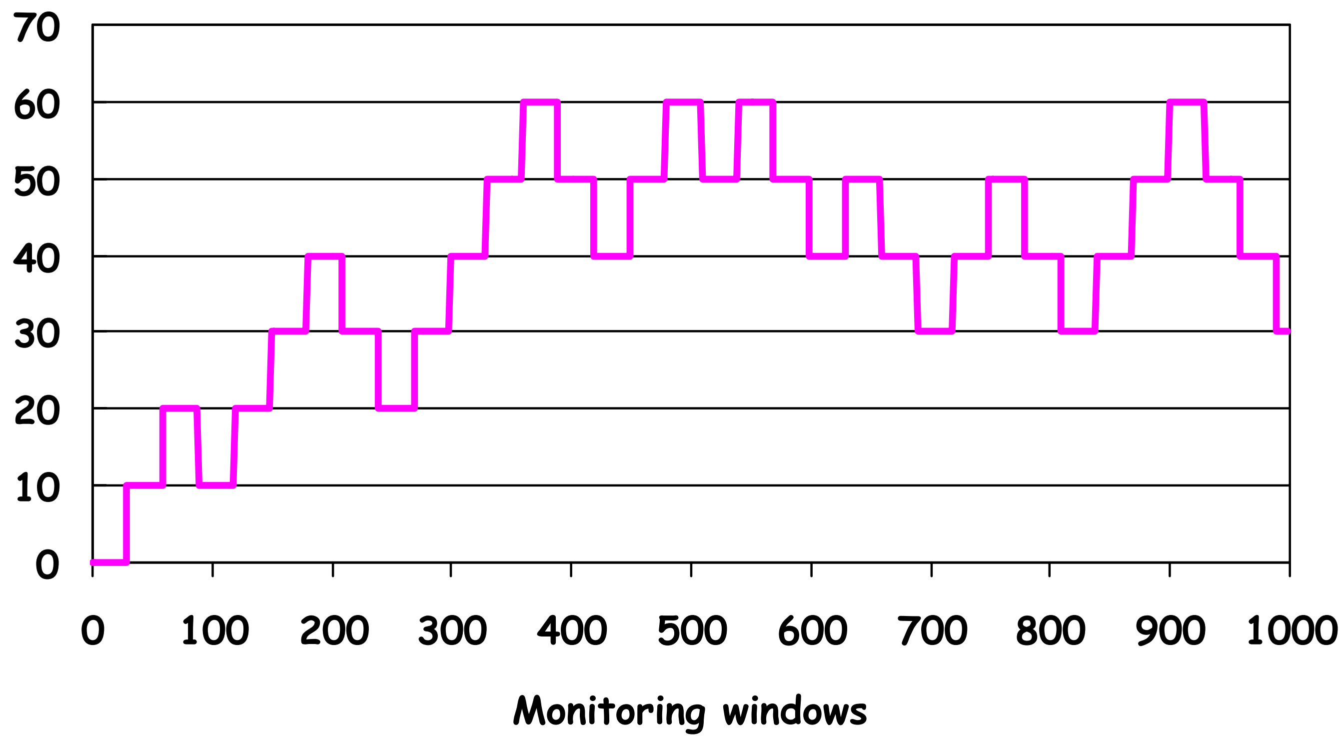
Average Response Time by *R*



Inside Each Server



Effectiveness of D_EQAL



Outline

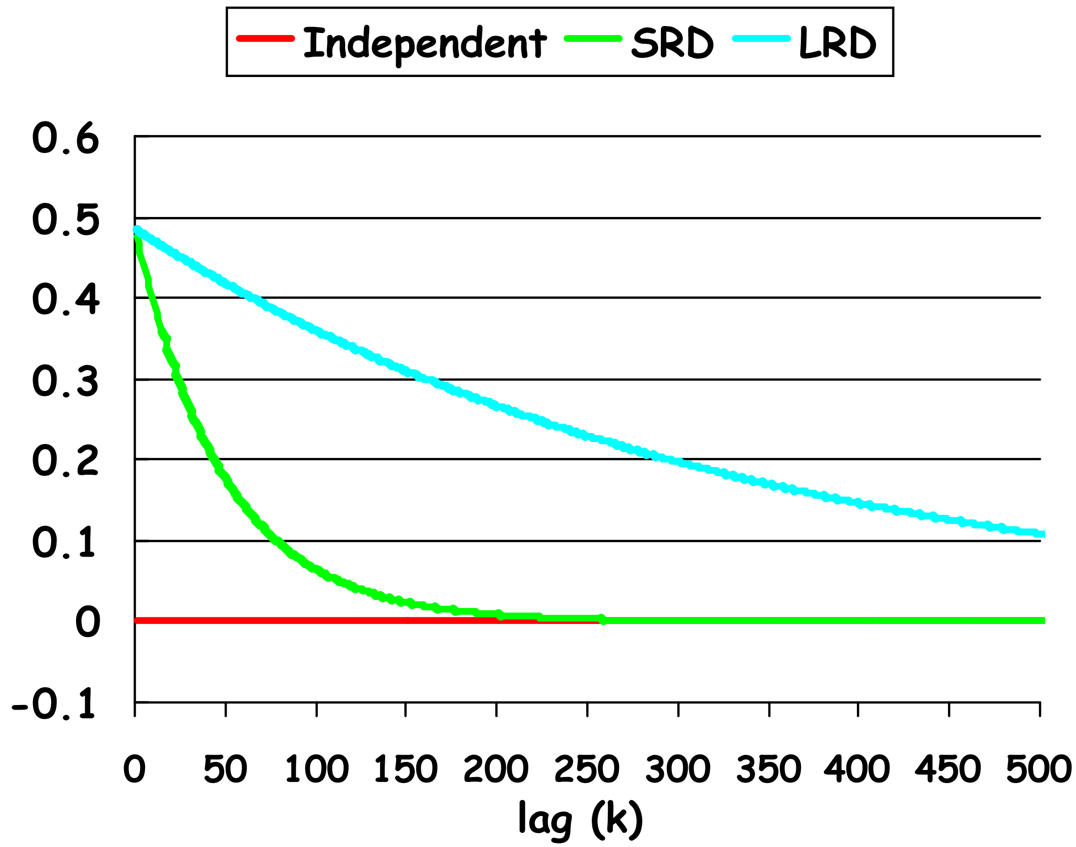
- Closed System
  - Experimental Evaluation using TPC-W
    - Autocorrelation propagation
  - Impact of autocorrelation
  - Two-queue system

- Policy Development
  - Load balancing under autocorrelated flows

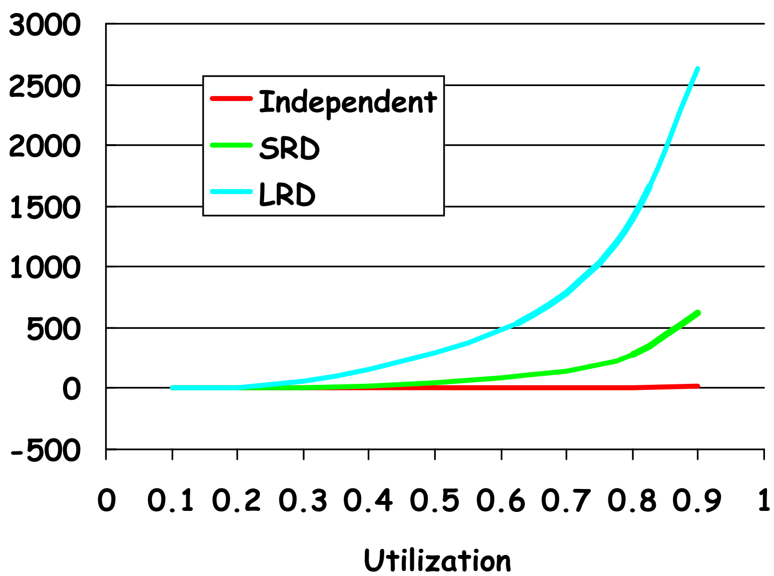
- On-going work

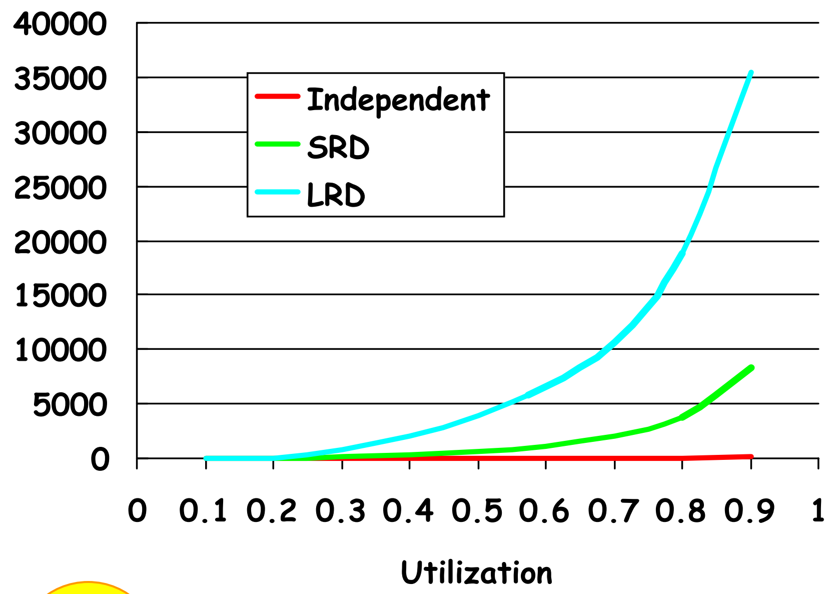
WILLIAM & MARY

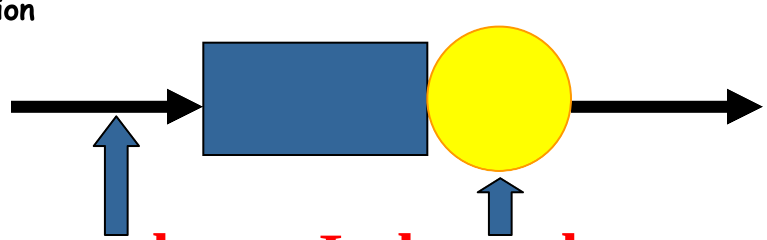
Examples of ACF



Impact of Correlated Arrivals







Dependent Arrivals

Independent Service (Exponential)