

# A Combined LIFO-Priority Scheme for Overload Control of E-commerce Web Servers

Naresh Singhmar   Vipul Mathur   Varsha Apte  
D. Manjunath

Indian Institute of Technology - Bombay  
Powai, Mumbai, 400 076, India

International Infrastructure Survivability Workshop, 2004



# Overload at E-commerce Web Sites

## UK E-tailers 'lose £300m' in Xmas sales

"Top UK E-tailers are estimated to have lost more than £300 million over the busy Christmas shopping period because of flaky website performance."

—The Register, January 15, 2004

## E-tail sites failing the Xmas test

"Empirix monitored the websites of 10 of the UKs biggest and best retailers and found many were failing to take all the hassle out of Christmas shopping."

—silicon.com, December 19 2003

## Online retail sites strain under 'Black Friday'

"Online retailers failed to complete 1 in 5 transactions during peak hours of the biggest shopping day of Xmas season"

—InternetWeek.com, December 4, 2003

## Iraq conflict hits Web sites hard

"... traffic to the site has already almost tripled and is expected to grow further."

"... the top 15 news sites have seen traffic jump by more than 40%."

— BBC News Online, March 20, 2003



# Overload at E-commerce Web Sites

## UK E-tailers 'lose £300m' in Xmas sales

"Top UK E-tailers are estimated to have lost more than £300 million over the busy Christmas shopping period because of flaky website performance."

—The Register, January 15, 2004

## E-tail sites failing the Xmas test

"Empirix monitored the websites of 10 of the UKs biggest and best retailers and found many were failing to take all the hassle out of Christmas shopping."

—silicon.com, December 19 2003

## Online retail sites strain under 'Black Friday'

"Online retailers failed to complete 1 in 5 transactions during peak hours of the biggest shopping day of Xmas season"

—InternetWeek.com, December 4, 2003

## Iraq conflict hits Web sites hard

"... traffic to the site has already almost tripled and is expected to grow further."

"... the top 15 news sites have seen traffic jump by more than 40%."

—BBC News Online, March 20, 2003

# Overload at E-commerce Web Sites

## UK E-tailers 'lose £300m' in Xmas sales

"Top UK E-tailers are estimated to have lost more than £300 million over the busy Christmas shopping period because of flaky website performance."

—The Register, January 15, 2004

## E-tail sites failing the Xmas test

"Empirix monitored the websites of 10 of the UKs biggest and best retailers and found many were failing to take all the hassle out of Christmas shopping."

—silicon.com, December 19 2003

## Online retail sites strain under 'Black Friday'

"Online retailers failed to complete 1 in 5 transactions during peak hours of the biggest shopping day of Xmas season"

—InternetWeek.com, December 4, 2003

## Iraq conflict hits Web sites hard

"... traffic to the site has already almost tripled and is expected to grow further.

... the top 15 news sites have seen traffic jump by more than 40%."

— BBC News Online, March 20, 2003



# Overload at E-commerce Web Sites

## UK E-tailers 'lose £300m' in Xmas sales

"Top UK E-tailers are estimated to have lost more than £300 million over the busy Christmas shopping period because of flaky website performance."

—The Register, January 15, 2004

## E-tail sites failing the Xmas test

"Empirix monitored the websites of 10 of the UKs biggest and best retailers and found many were failing to take all the hassle out of Christmas shopping."

—silicon.com, December 19 2003

## Online retail sites strain under 'Black Friday'

"Online retailers failed to complete 1 in 5 transactions during peak hours of the biggest shopping day of Xmas season"

—InternetWeek.com, December 4, 2003

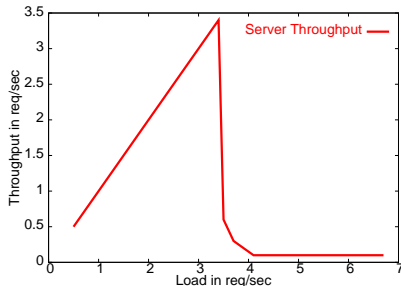
## Iraq conflict hits Web sites hard

"... traffic to the site has already almost tripled and is expected to grow further. ... the top 15 news sites have seen traffic jump by more than 40%."

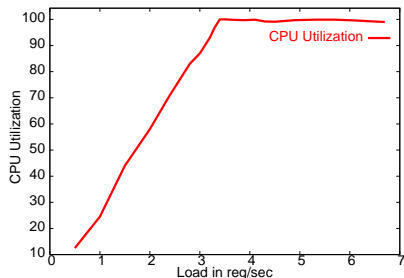
—BBC News Online, March 20, 2003

# Overload and its Effects

- Overload: Offered load  $>$  system capacity
- Cause of Overload: sales, big shopping days, server failures, breaking news



Throughput vs. Load



CPU Utilization vs. Load



# Overload and its Effects

## Effects of Overload

- Increased response time
- Abandonment due to timeouts
- Retries  $\Rightarrow$  increase in load
- Dramatically deteriorated throughput
- E-commerce Web sites lose revenue
- Customer experience deteriorates at times of peak usage

## Objective of Overload Control

Reduce the amount of lost requests and increase throughput



# Overload and its Effects

## Effects of Overload

- Increased response time
- Abandonment due to timeouts
- Retries  $\Rightarrow$  increase in load
- Dramatically deteriorated throughput
- E-commerce Web sites lose revenue
- Customer experience deteriorates at times of peak usage

## Objective of Overload Control

Reduce the amount of lost requests and increase throughput





# Overload and its Effects

## Effects of Overload

- Increased response time
- Abandonment due to timeouts
- Retries  $\Rightarrow$  increase in load
- Dramatically deteriorated throughput
- E-commerce Web sites lose revenue
- Customer experience deteriorates at times of peak usage

## Objective of Overload Control

Reduce the amount of lost requests and increase throughput



# Overload Control

## Previous Work

Focusses mainly on sophisticated techniques which may be difficult to implement, or are too generic to be effective for E-commerce Web-servers with dynamic content

## Our Work

Focus on simplicity, ease of implementation, and on E-commerce Web-servers



# Overload Control

## Previous Work

Focusses mainly on sophisticated techniques which may be difficult to implement, or are too generic to be effective for E-commerce Web-servers with dynamic content

## Our Work

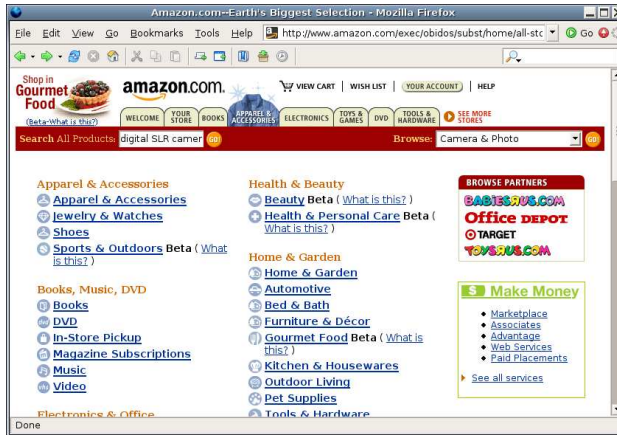
Focus on simplicity, ease of implementation, and on E-commerce Web-servers



# Online Retail Store

Possible Activities on an On-line Store (Screen shots courtesy Amazon.com)

Main, Browse, Search, Details, Login, Shipping, Payment, Confirmation



# Online Retail Store

Possible Activities on an On-line Store (Screen shots courtesy Amazon.com)

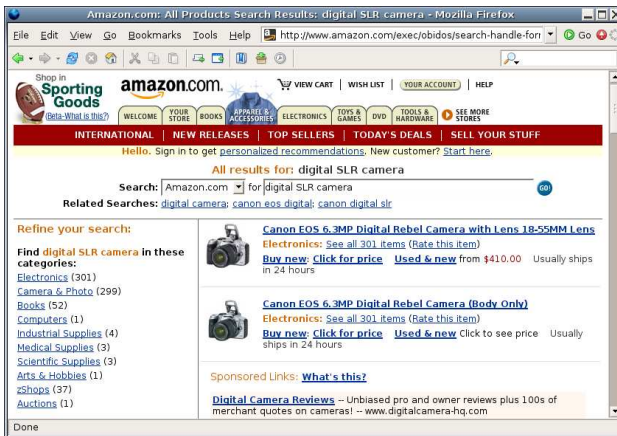
Main, **Browse**, Search, Details, Login, Shipping, Payment, Confi rm



# Online Retail Store

Possible Activities on an On-line Store (Screen shots courtesy Amazon.com)

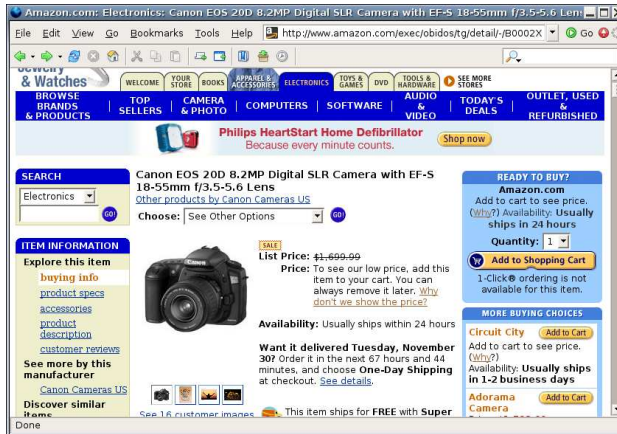
Main, Browse, **Search**, Details, Login, Shipping, Payment, Confi rm



# Online Retail Store

Possible Activities on an On-line Store (Screen shots courtesy Amazon.com)

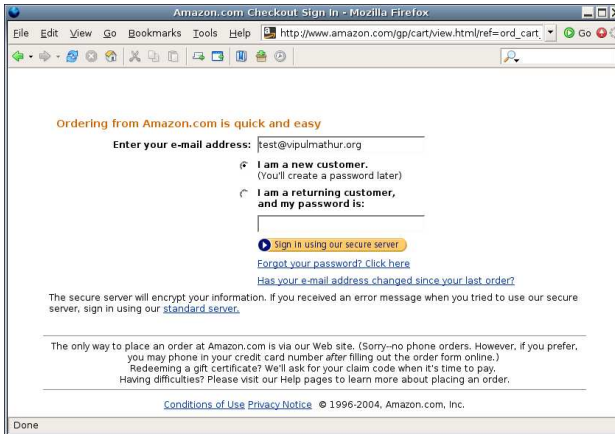
Main, Browse, Search, **Details**, Login, Shipping, Payment, Confi rm



# Online Retail Store

Possible Activities on an On-line Store (Screen shots courtesy Amazon.com)

Main, Browse, Search, Details, **Login**, Shipping, Payment, Confi rm





# Online Retail Store

Possible Activities on an On-line Store (Screen shots courtesy Amazon.com)

Main, Browse, Search, Details, Login, **Shipping**, Payment, Confirmation

The screenshot shows the Amazon.com checkout page titled "Amazon.com Checkout: Select Address - Mozilla Firefox". The browser's address bar displays the URL "https://www.amazon.com/gp/fix/checkout/sign-in/select-address". The page features the Amazon logo and navigation links: "SIGN IN", "SHIPPING & PAYMENT" (highlighted), "GIFT-WRAP", and "PLACE ORDER".

**Enter the shipping address for this order:**  
Please enter a shipping address for this order. When finished, click the "Continue" button. Or, if you're sending items to more than one address, click the "Add another address" button to enter additional addresses.

The form includes the following fields:

- Full Name:
- Address Line1:  (Street address (PO Boxes not acceptable))
- Address Line2:  (Apartment, suite, unit, building, floor, etc.)
- City:
- State/Province/Region:
- ZIP/Postal Code:
- Country:  (United States)
- Phone Number:

A yellow "Continue" button with a right arrow is located at the bottom of the form. The status bar at the bottom of the browser window shows "Done" and the Amazon website URL.



# Online Retail Store

Possible Activities on an On-line Store (Screen shots courtesy Amazon.com)

Main, Browse, Search, Details, Login, Shipping, **Payment**, Confir m

Amazon.com Checkout: Payment - Mozilla Firefox

File Edit View Go Bookmarks Tools Help <https://www.amazon.com/gp/checkout/ship/select.htm> Go

amazon.com. SIGN IN SHIPPING & PAYMENT GIFT-WRAP PLACE ORDER

**Please select a payment method and create a password**  
You're almost done! Please enter your credit card number below. You may also pay using a [Purchasing Card](#), Check Card, or Amazon.com gift certificate. (All purchases are safe and secure—[guaranteed](#).)  
If you prefer to give the number to us by phone, enter only the card's last five digits. After you have completed your order, we'll e-mail you the phone number to call to provide your full credit card number. You may also pay by check ([why this takes longer](#)).

**Paying with a credit card?**

Payment Method	Credit Card No.	Expiration Date	Cardholder's name
<input checked="" type="radio"/> Amazon.com Visa	<input type="text"/>	01 2004	<input type="text"/>
<input type="radio"/> Amazon Credit Account <a href="#">Learn more</a>	<input type="text"/>	Does not expire	<input type="text"/>

Note: Using an Amazon.com Visa Card? Select Amazon.com Visa. Using a Visa Check Card? Select Visa. Using a Eurocard or MasterMoney card? Select MasterCard.

☐ Pay by check or money order  
(or check funds on account)

Want to pay by purchase order? Learn more about [our Corporate Accounts program](#).

Have any gift cards, gift certificates or promotional claim codes?

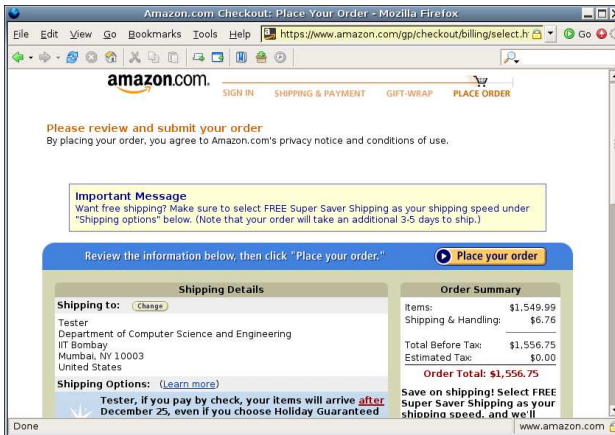
Done [www.amazon.com](http://www.amazon.com)



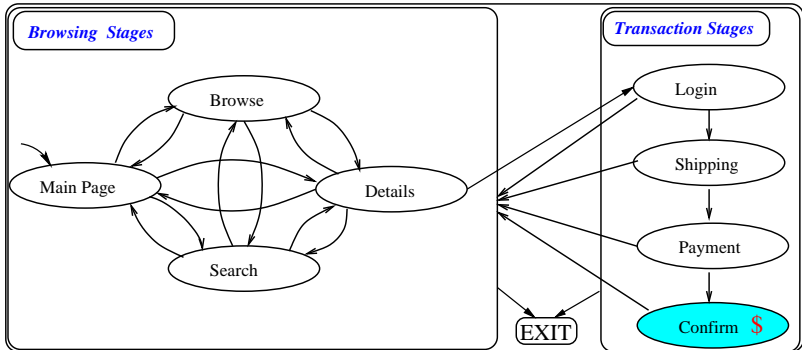
# Online Retail Store

Possible Activities on an On-line Store (Screen shots courtesy Amazon.com)

Main, Browse, Search, Details, Login, Shipping, Payment, **Confirm**



# E-commerce Workload Model



- Most users go only through *Browsing stages*
- Very few proceed to revenue generating *Transaction stages*



# Proposed Scheme and Architecture



# Key Ideas of Proposed Solution

## Increase completion rate of revenue generating requests

- Separate queues for each type of request
- *Transaction queues* have strictly higher priority than *browsing queues*
- Relative priority within *transaction* and *browsing* based on “utility” of the queue

## Increase the overall throughput of Web-server during overload

- Using LIFO for *browsing queues* during overload
- Switch between LIFO and FIFO based on thresholds
- Always FIFO for *transaction queues*



# Key Ideas of Proposed Solution

## Increase completion rate of revenue generating requests

- Separate queues for each type of request
- *Transaction queues* have strictly higher priority than *browsing queues*
- Relative priority within *transaction* and *browsing* based on “utility” of the queue

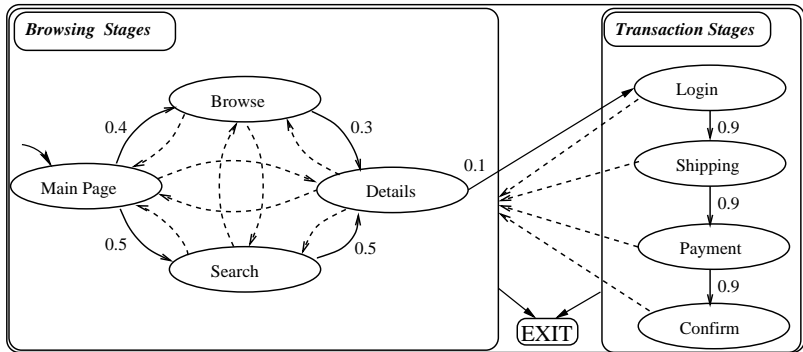
## Increase the overall throughput of Web-server during overload

- Using LIFO for *browsing queues* **during overload**
- Switch between LIFO and FIFO **based on thresholds**
- Always FIFO for *transaction queues*



# E-commerce Workload Model

## Represented as a Markov Chain



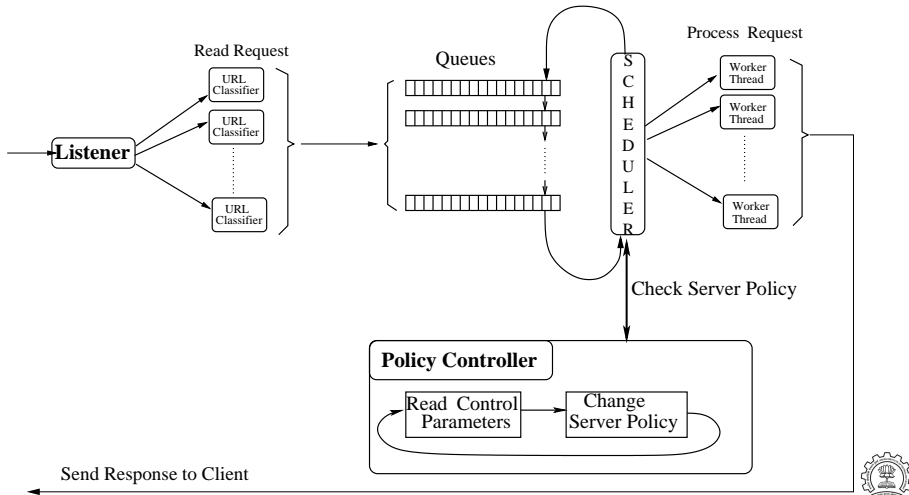
Probability of generating revenue can be used as 'utility' value





# Proposed Web-server Architecture

A prototype Web-server with this architecture has been implemented



# LIFO-Pri Scheme

## Set Service Discipline of Browsing Queues

- 1 Measure CPU Utilization over an interval
- 2 If utilization is more than **upper threshold**, then set browsing queue discipline to LIFO
- 3 If utilization is less than **lower threshold**, then set browsing queue discipline to FIFO



# LIFO-Pri Scheme

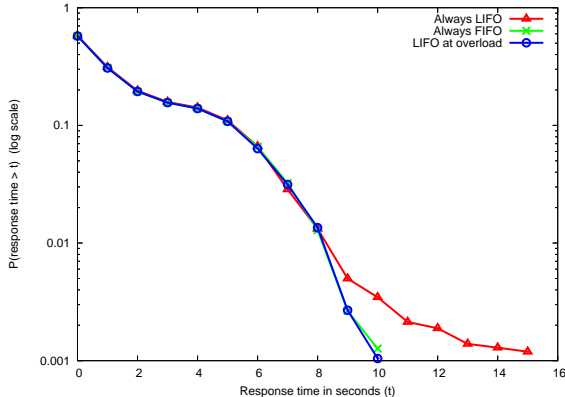
## Dynamic Priority

- 1 When a worker thread is available and at least one queue has a pending request,
- 2 Calculate **dynamic priority** of each queue  
$$= \text{queue length} \times \text{utility}$$
- 3 Select the queue with highest dynamic priority
- 4 Read a request from this queue according to current service discipline
- 5 Assign worker thread to request.



# LIFO vs. FIFO: Response Time

Response time distribution at  $\rho = 0.941$  with a timeout of 20 seconds.

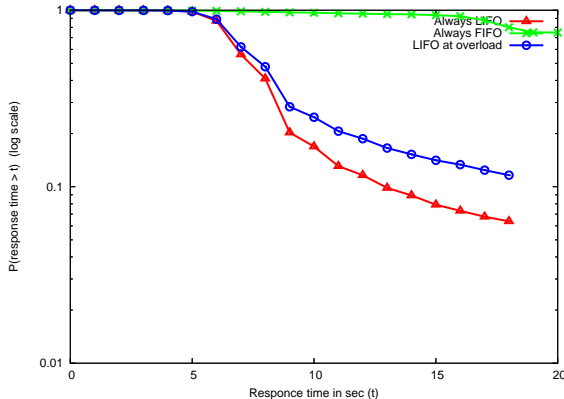


LIFO always has longer tail in non-overload conditions



# LIFO vs. FIFO: Response Time

Response time distribution at  $\rho = 1.47$  with a timeout of 20 seconds.

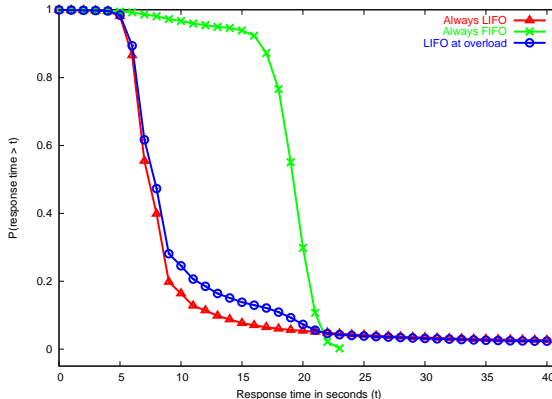


$$P[R_{LIFO} > 15] = 0.1 \text{ whereas } P[R_{FIFO} > 15] = 0.95$$



# LIFO vs. FIFO: Response Time

Response time distribution at  $\rho = 1.47$  with a timeout of 40 seconds.



For longer timeout, long tail of LIFO is seen again



# LIFO vs. FIFO: Throughput

Timeout of 40 seconds ( $\rho = 1.47$ )			
Percentage	Always-FIFO	Always-LIFO	LIFO-at-overload
Completed	86.7	84.4	84.6
Timed-out	00.0	02.3	02.0
Dropped	13.3	13.4	13.4
Timeout of 20 seconds ( $\rho = 1.47$ )			
Completed	21.9	81.0	76.8
Timed-out	64.9	05.4	09.7
Dropped	13.3	13.6	13.4

Large rate of abandonment in FIFO with a shorter timeout



# Observations

## Summary of Observations for LIFO vs. FIFO

- Longer tail for LIFO  $\Rightarrow$  using LIFO not appropriate when offered load  $<$  capacity
- Larger timeout value favors FIFO (no long tail)
- Success rate is higher for LIFO policies in overload (with small timeouts)
- LIFO-at-overload gives higher throughput and better response time distribution in overload





# Experiments and Results



# Experimental Setup

- Emulate an E-commerce Web site
- Eight stages represented by Perl CGI scripts
- Modified version of `httperf` for workload generation
- Exponentially distributed timeouts
- Retries for requests abandoned due to timeouts
- Session abandonments
- Separate priority queues for each type of request:  
4 browsing, 4 transaction



# Experiments Performed

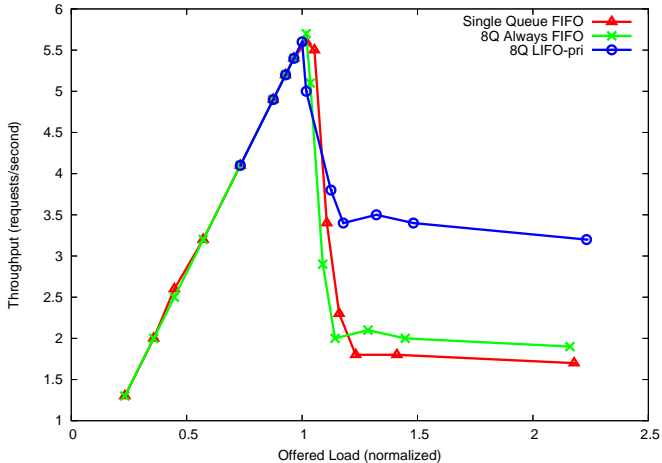
Three sets of experiments were done.

- Single Queue: FIFO order. Capacity: 100.
- 8Q Always FIFO: All 8 queues always in FIFO order. Capacity: 50 for browsing queues, 25 for transaction queues.
- 8Q LIFO-Pri: LIFO at overload for browsing queues. Always FIFO for transaction queues.

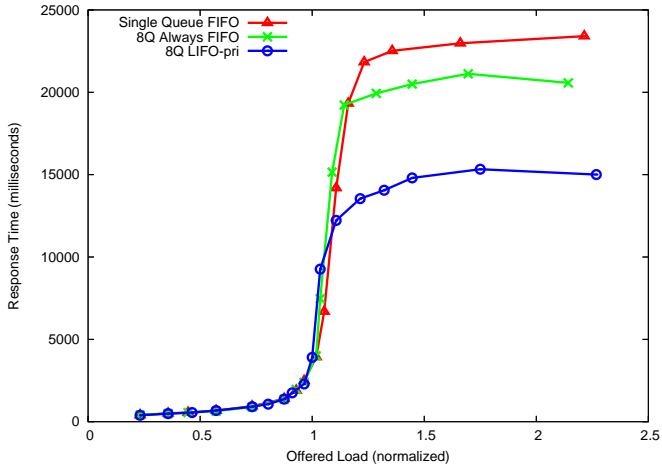
Dynamic priority is used for multi-queue setups. Utility of a queue is assigned in proportion to probability of a request in that queue resulting in a final 'confirm' transaction.



# Overall Throughput vs. Offered Load



# Average Response Time vs. Offered Load



# Looking at Request Types

Throughput data for different types of requests at  $\rho = 1.4$

Case	Requests	Browsing	Tr-1	Tr-2	Tr-3	Tr-4
SQ	<b>Generated Completed Timed out Dropped</b>	42029				
		16170	20	15	9	8
		20029	18	5	1	1
		5753				
8Q-AF	<b>Generated Completed Timed out Dropped</b>	43324	24	20	19	15
		19852	23	19	19	15
		16305	1	1	0	0
		7167	0	0	0	0
8Q-LIFO-Pri	<b>Generated Completed Timed out Dropped</b>	44826	195	137	99	53
		30851	187	127	87	50
		4075	8	10	12	3
		9900	0	0	0	0



# Looking at Request Types

## Requests Completed at $\rho = 1.4$

Case	Browsing	Tr-1	Tr-2	Tr-3	Tr-4
SQ	16170	20	15	9	8
8Q-AF	19852	23	19	19	15
8Q-LIFO-Pri	30851	187	127	87	50

6-7 fold increase in 'confirm' requests from SQ to 8Q-LIFO-Pri



# Overall Throughput Data

At  $\rho = 1.4$  (percentages)

Case	SQ	8Q-AF	8Q-LIFO-Pri
Completed	29.9	36.6	57.5
Timed out	36.8	29.9	07.5
Dropped	10.6	13.1	18.2
Not Generated	22.8	20.4	16.8





# Summary

- Presented a reasonably realistic model of E-commerce workload
- LIFO-Pri scheme for overload control: experimentally verified
  - Server could do productive work at 60% of its capacity
  - Upto a 7-fold increase in number of successful 'confirm' requests when compared to single queue model
  - Minimal overheads
- Outlook
  - Need to look at better indicators of overload
  - More appropriate user behavior models
  - Analytical models for further insight



# Thank You!

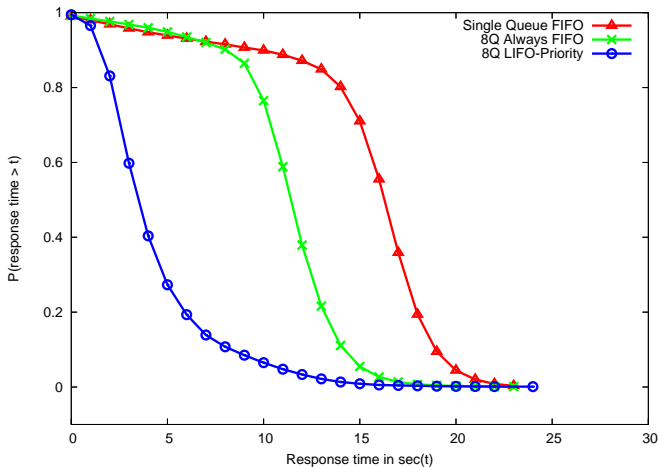


<http://www.cse.iitb.ac.in/perfnet>



# Response Time Distribution

Response time distribution for 'main' page requests for  $\rho = 1.4$



# Previous Work

## Previous Work

- Session-based admission control. (*Cherkasova and Phaal*)
- Dynamic Weighted Fair Sharing. (*Chen and Mohapatra*)
- Admission control with request scheduling. (*Elnikety et al*)
- Control theory based approach. (*Abdelzaher et al.*)
- Improving user-perceived performance at a Web server. (*Dalal and Jordan*)



# Sample 'Utility' Values for Queues

Request Queue	Utility
Main Page (Br-1)	27
Browsing (Br-2)	22
Searching (Br-3)	36
Details (Br-4)	73
Login (Tr-1)	3650
Shipping (Tr-2)	4050
Payment (Tr-3)	4500
Confirm (Tr-4)	5000

