

The application-driven enterprise network

Planning networks
according to the
applications they handle

Alistair A. Croll
Chief Strategy Officer
Coradiant, inc.

**NETWORLD
+ INTEROP**



Summary

- **Five silos in the data center**
- **Real user transactions are the killer app**
 - Networks, servers, and apps *work for the user*
 - We seldom watch them today
- **But applications are dynamic; platforms must adapt**
 - Infrastructure capacity
 - Throughput and responsiveness
 - Functionality and protocols
- **How can we hit a moving target?**
 - Understand the business drivers
 - Watch what matters
 - Start with solid methodologies
 - Know how load relates to performance
 - Plan the network accordingly

The data center tomorrow

- **Five main apps**

- Interactive transactional (HTTP)
- Isochronous media (VOIP)
- Store-and-forward messaging
- Interactive messaging (Groupware)
- Media-on-demand (Streaming)

- **Plus management traffic**

Inside the pipe

- **Virtually separate networks for each**
- **Management traffic always gets through**
- **Within each app silo, build to tech and business needs**

Your total network profile

Management traffic (SNMP, alerts, some SSH, etc.)

Production traffic

Interactive transactional
(Total transaction time, human or machine audience)

Isochronous media (VOIP)
(Peak circuit volume * data per circuit + margin of error, jitter, RTT)

Store-and-forward messaging
(Data volume, longest allowable delivery, spam/antivirus overhead)

Interactive messaging
(RTT, spam/antivirus overhead, number of users)

Media-on-demand
(Throughput, multicast approaches, allowable buffering, loss)

Let's look at one in depth

Interactive transactional
(Total transaction time, human or machine audience)

Real user transactions
are the killer app

- **Applications and sites don't make money**
 - *Real users do*
- **We seldom watch real transactions today**
 - This is behind much of the IT-business battle
- **Suits don't want to hear “Server 10 had a 15% increase in network flight time”**
 - They hear “Bob's blaming the order app for missing his sales quota this quarter”

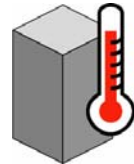
If we seldom watch them today,
what *do* we watch?

- **We look at equipment**
 - Assume if it's OK, users are fine
- **We test it ourselves**
 - Assume that users do what we do
- **We wait for helpdesk complaints**
 - But can't reproduce or diagnose

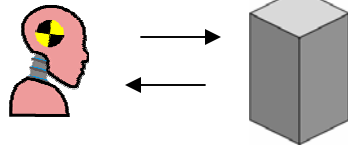
Consider a typical performance profile

- **The tech support site needs a network time of 500 milliseconds!**
- **Not measurable, not accountable, and not useful for network design**
 - **Which users?**
 - **Doing what?**
 - **At what times?**
 - **Coming from where?**
 - **With what margin of error?**

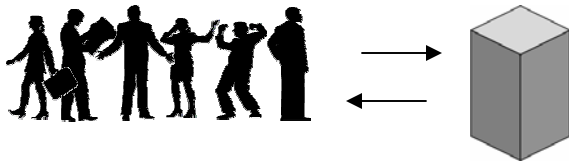
How we measure: Perception is reality



- **Device measurements**
 - Health of the infrastructure

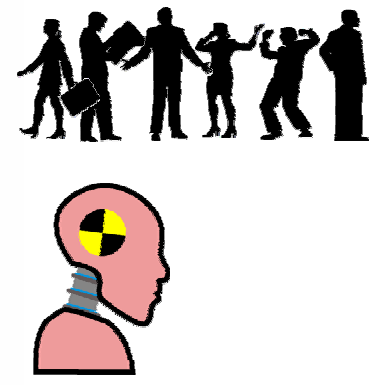
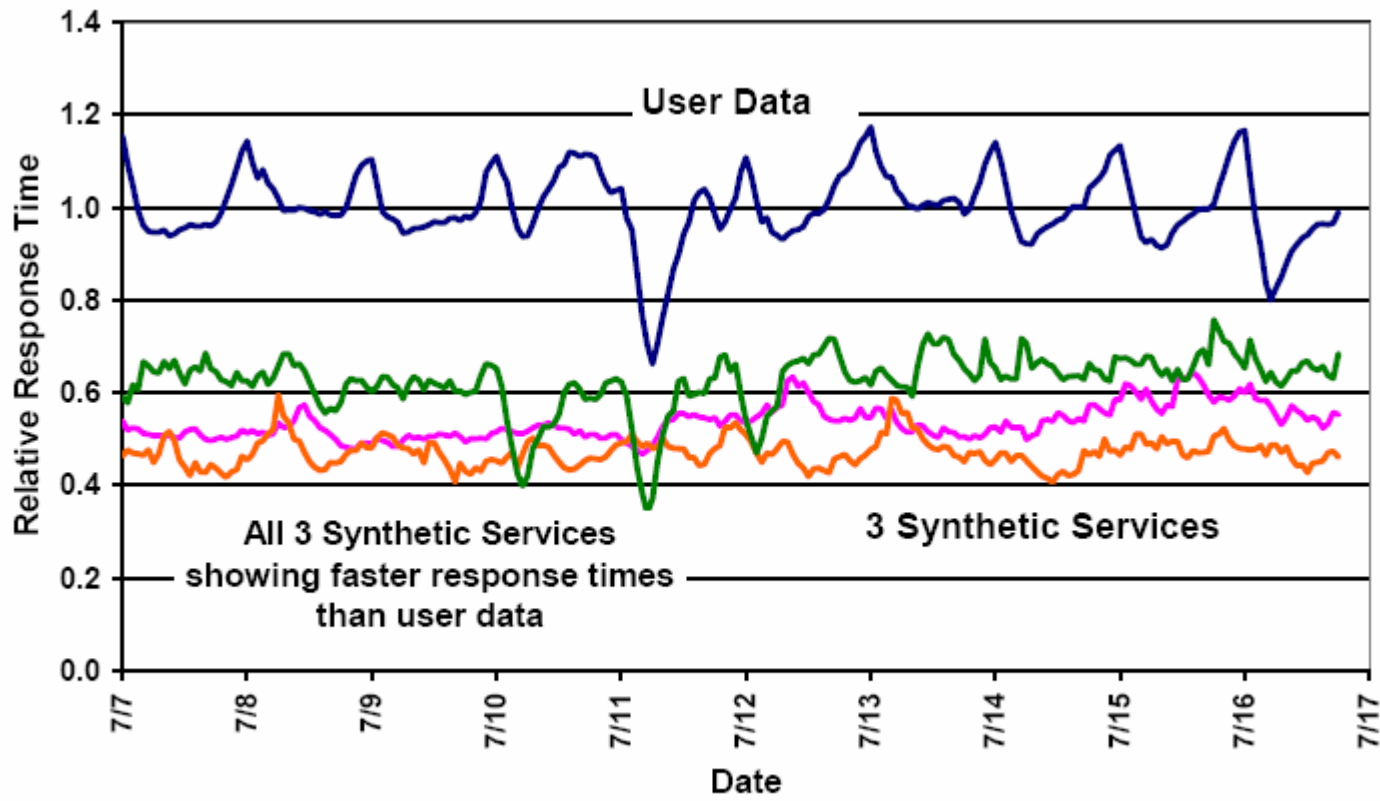


- **Synthetic (service) testing**
 - Correct functionality of a narrowly defined, well-known set of steps
 - Whether a system can handle a volume of traffic



- **Per-user (session) measurements**
 - What users are really doing
 - Where errors are occurring
 - What actual performance is like
 - How load relates to performance on a production system

Bad data means bad profiles— and underbuilt networks



Watch the wrong thing,
get the wrong impression

- **“Synthetic measurements randomly and frequently understate the actual response time seen by real users by as much as 50%, and overstate the response time of other applications by as much as 350%.**
- **Synthetic agents usually interact with less than 10% of all Web site URLs and less than 15% of the ISPs connecting real users to the site.”**

• Netforecast, January 2005

So how do we hit a moving target?

- **Start with solid methodologies**
- **Understand the business drivers**
 - Technology needs
 - Productivity, satisfaction, and adoption
 - Avoiding failures at bottlenecks
- **Watch what matters**
 - Profile the target application
 - Watch real sessions on production systems
 - Watch the percentiles
 - Know how load relates to performance

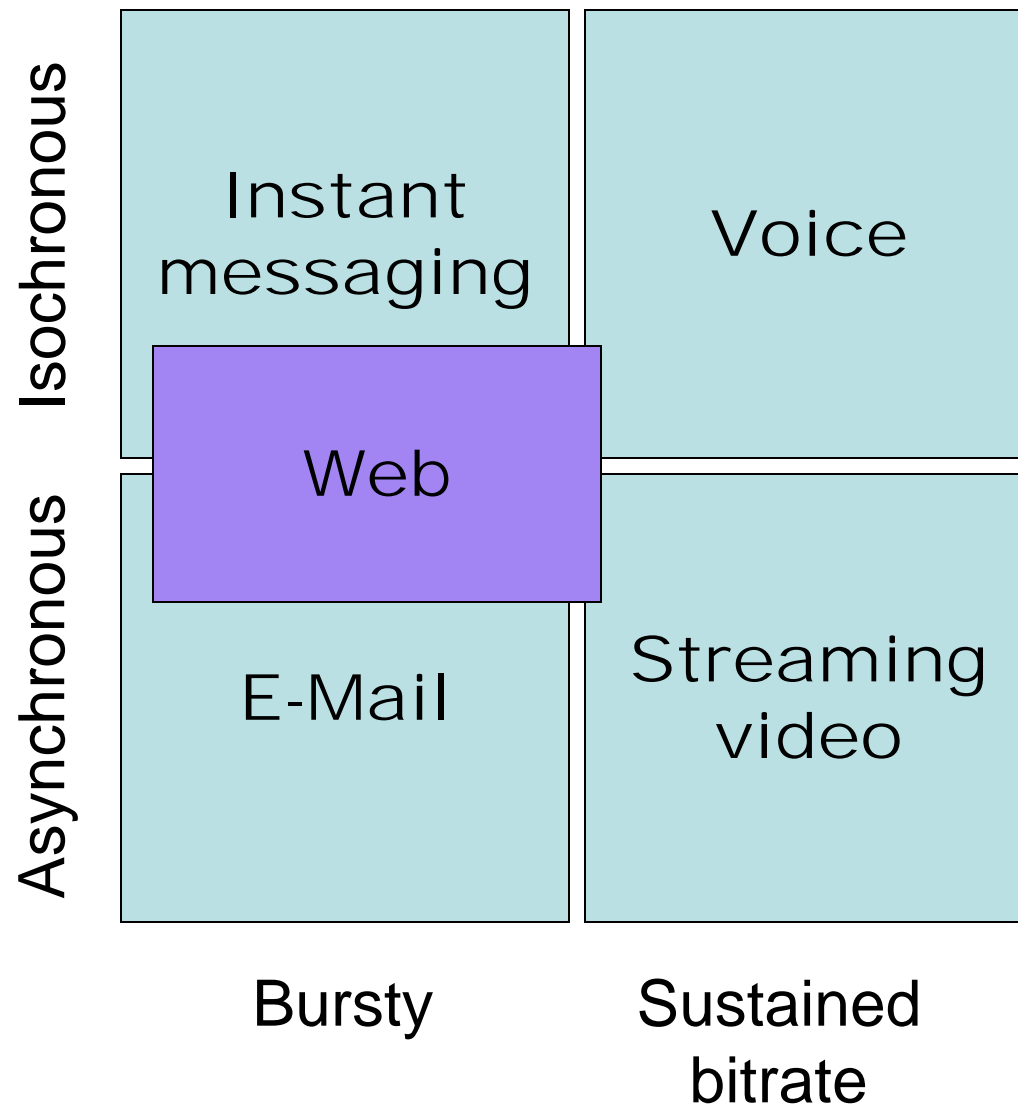
Start with solid methodologies

- **Define performance goals**
 - What return do you hope to achieve?
 - Productivity
 - Revenue
 - Cost avoidance
 - What investment are you making?
 - Fixed costs
 - Recurring costs
 - How much should the system handle?
- **Baseline from the best possible data**
- **Understand trends and adoption profiles**
- **Develop simple models**
 - The few things that use the most resources

So how do we hit a moving target?

- **Start with solid methodologies**
- **Understand the business drivers**
 - Technology needs
 - Productivity, satisfaction, and adoption
 - Avoiding failures at bottlenecks
- **Watch what matters**
 - Profile the target application
 - Watch real sessions on production systems
 - Watch the percentiles
 - Know how load relates to performance

Business drivers: Technology needs



Business drivers: Customer satisfaction

- **Loyalty**
- **Revenue attainment**
- **Brand erosion**
- **Poorly understood causality and limited empirical proof**

Jakob Nielsen defines “reasonably fast operations, taking between 2 and 10 seconds” as the range between tolerating and frustrated.

“Usability Engineering,” by Jakob Nielsen

In 1997, when the typical Web page loaded in 10 seconds, users significantly changed their perception of how interesting the content was when they had to wait 41 seconds and longer.

“A Psychological Investigation of Long Retrieval Times on the World Wide Web,” Judith Ramsay, Alessandro Barbasi, Jenny Preece

Definite shift of ratings from good to poor at 10 seconds, and from poor to unacceptable at 39 seconds, when doing online PC configuration.

“Integrating User-Perceived Quality Into Web Server Design,” by Nina Bhatti, Anna Bouch, Allan Kuchinsky

Business drivers: End-user productivity

- **To be productive, workers need speed**
 - Under 0.1 second feels “instant”
 - Under 1 second allows uninterrupted thought
 - Over 10 seconds loses attention

Robert B. Miller, in 1968 paper at Fall Joint Computer Conference

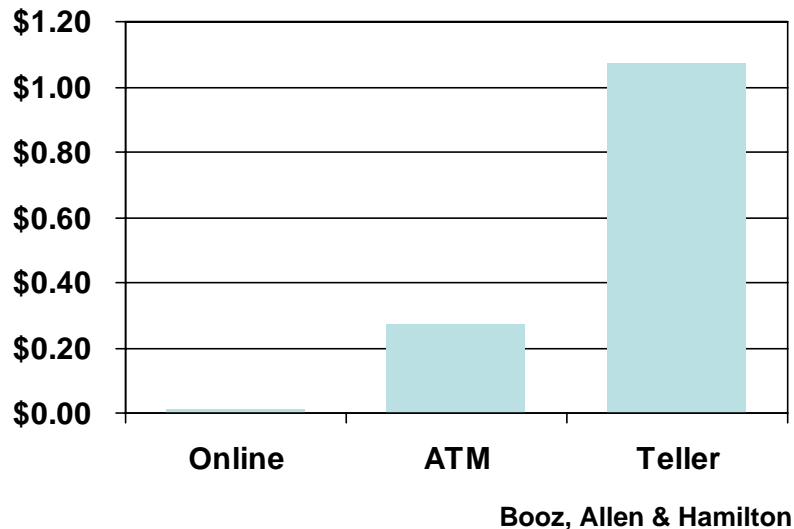
- **Studies show information workers are more productive in “flow state”**
- **Relies heavily on several factors**
 - Concentration
 - Interruption avoidance
 - Real-time feedback levels

Mihaly Csikszentmihalyi

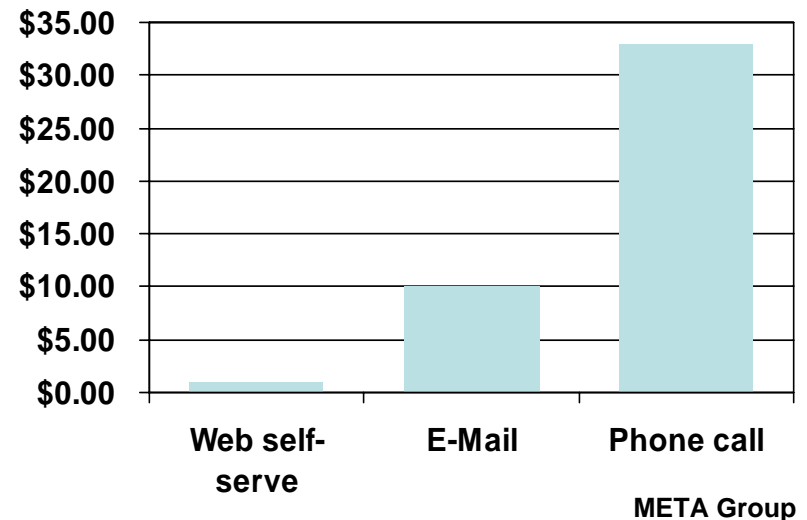
- **The coffee break effect**
- **Strongly linked to failure of major deployments**

Business drivers: Adoption of cheaper technologies

Relative cost of banking transaction



Relative cost of customer support

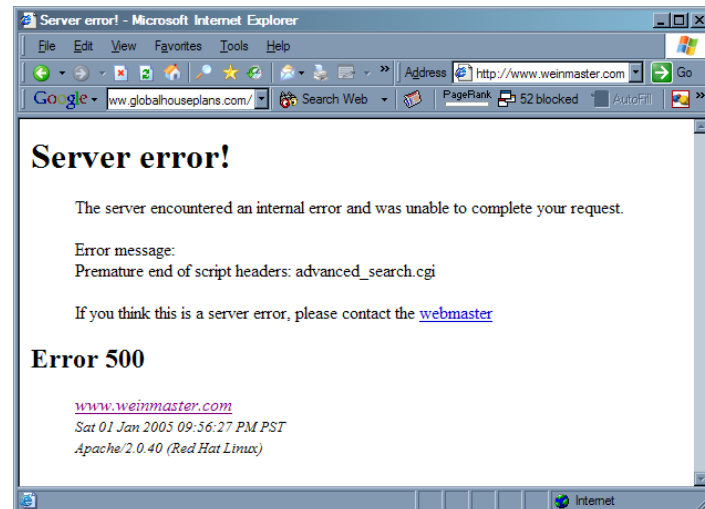
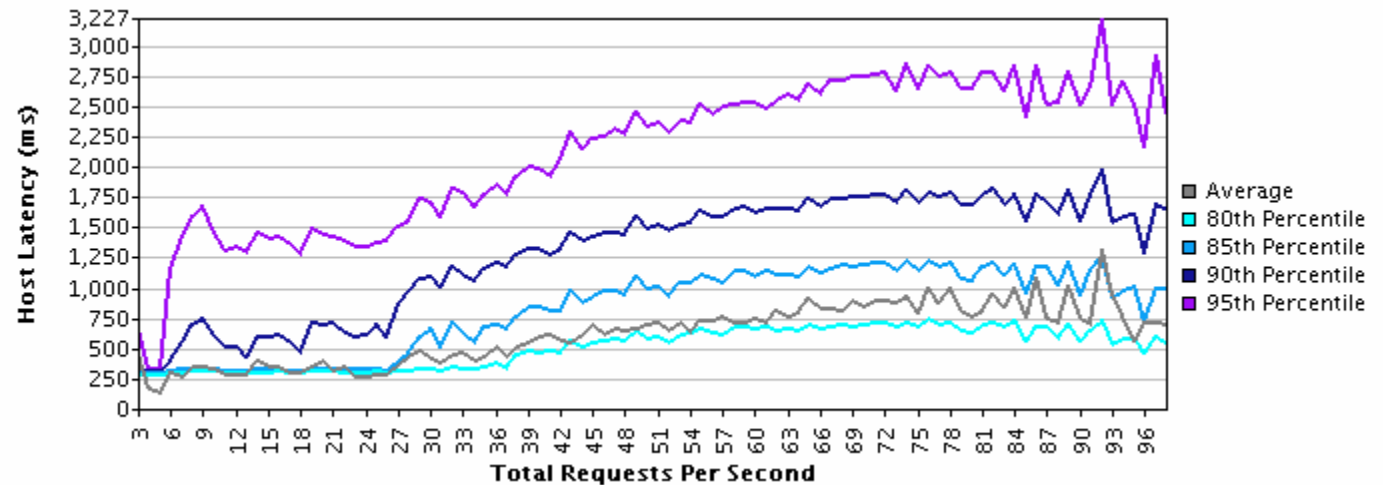


- **24.3M households banked online in 2002**
- **Grew 37.6% in 2002**
- **Internet-based billing and payment from 22% in 2002 to 45% in 2005**

Jupiter, Gartner

Business drivers: Failures at bottlenecks

- Performance is generally related to resource shortage
- When resources are missing, systems have no cycles to do maintenance and housekeeping
- If a single element is the bottleneck, it becomes the point of failure



So how do we hit a moving target?

- **Start with solid methodologies**
- **Understand the business drivers**
 - Technology needs
 - Productivity, satisfaction, and adoption
 - Avoiding failures at bottlenecks
- **Watch what matters**
 - Profile the target application
 - Watch real sessions on production systems
 - Watch the percentiles
 - Know how load relates to performance

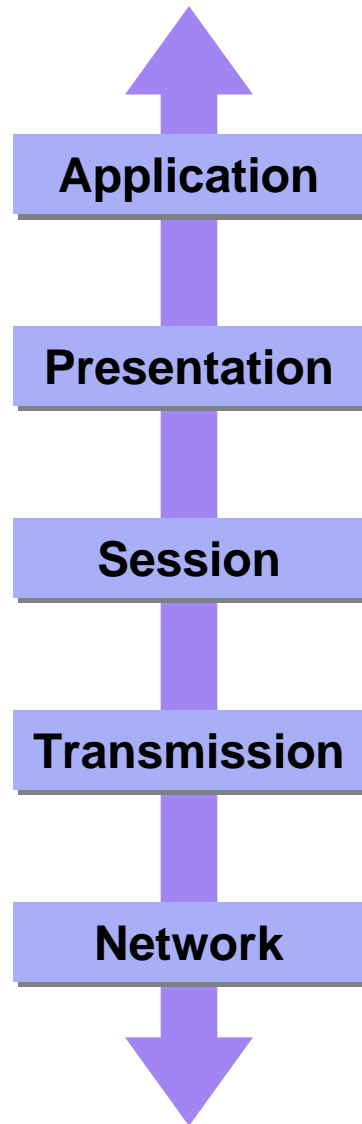
Watch what matters

- **Profile the target application**
 - Define which functions to watch
 - Capture data from all layers of OSI
- **Watch real sessions on production systems**
 - Transparently
 - Over busy and light periods
- **Ignore averages: Watch the percentiles**
 - Store frequency distributions
- **Know how load relates to performance**
 - To generate acceptable load profiles

Define which functions to watch


- **Which would you rather test?**
- **The employee expense site**
- **The time it takes the server to handle 95% of the expense report submissions from 9AM to 4PM on weekdays**
- **...but be sure it's relevant!**

Get data from all layers



- End-to-end transaction completion
- Individual steps (i.e. a page load)
- Page rendering
- Components of delay (host, network, SSL)
- Encryption overhead
- Session establishment (TCP SYN/ACK)
- TCP round-trip time
- TCP retransmissions, segment reordering
- Service discovery (DNS)
- Throughput
- Packet loss

You'll need them because it's all related

 Report for Average TCP Roundtrip Time on All-Objects

From: Aug 14 2004 19:00 UTC
To: Sep 13 2004 18:59 UTC

Avg TCP RTT Time (ms)



 Report for Retransmits on All-Objects

From: Aug 14 2004 19:00 UTC
To: Sep 13 2004 18:59 UTC

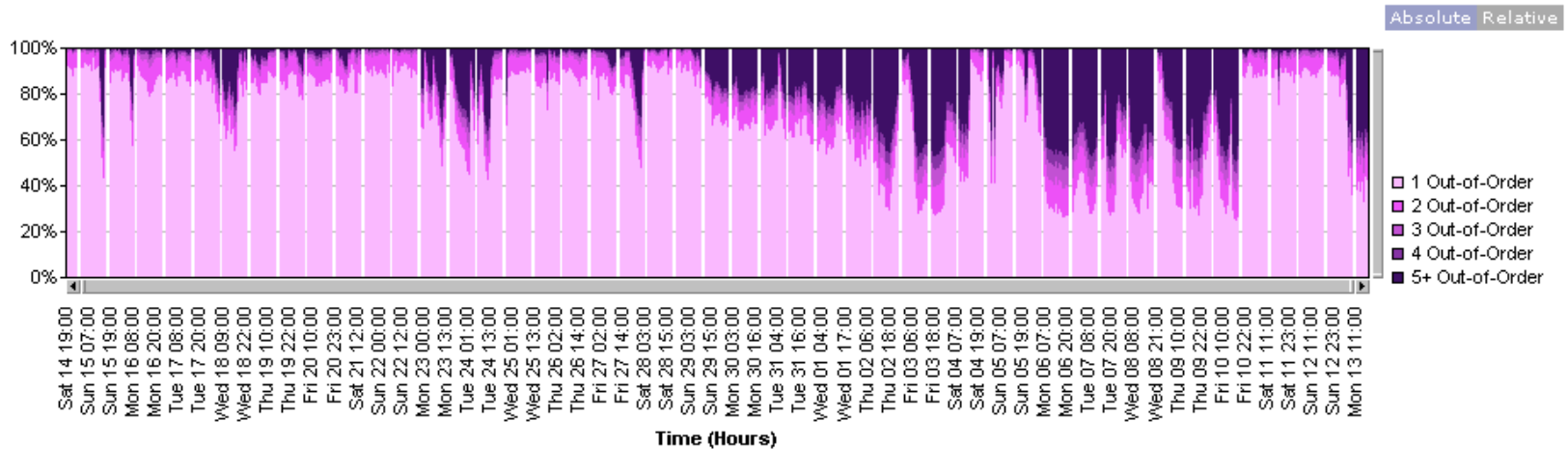


 Report for Out of Order Segments on All-Objects

From: Aug 14 2004 19:00 UTC
To: Sep 13 2004 18:59 UTC

% of Requests

% of Requests



Sat 14 19:00
Sun 15 07:00
Sun 15 19:00
Mon 16 08:00
Mon 16 20:00
Tue 17 08:00
Tue 17 20:00
Wed 18 09:00
Wed 18 22:00
Thu 19 10:00
Thu 19 22:00
Fri 20 10:00
Fri 20 23:00
Sat 21 12:00
Sun 22 00:00
Sun 22 12:00
Mon 23 00:00
Mon 23 13:00
Tue 24 01:00
Tue 24 13:00
Wed 25 01:00
Wed 25 13:00
Thu 26 02:00
Thu 26 14:00
Fri 27 02:00
Fri 27 14:00
Sat 28 03:00
Sat 28 15:00
Sun 29 03:00
Sun 29 15:00
Mon 30 03:00
Mon 30 16:00
Tue 31 04:00
Tue 31 16:00
Wed 01 04:00
Wed 01 17:00
Thu 02 06:00
Thu 02 18:00
Fri 03 06:00
Fri 03 18:00
Sat 04 07:00
Sat 04 19:00
Sun 05 07:00
Sun 05 19:00
Mon 06 07:00
Mon 06 20:00
Tue 07 08:00
Tue 07 20:00
Wed 08 08:00
Wed 08 21:00
Thu 09 10:00
Thu 09 22:00
Fri 10 10:00
Fri 10 22:00
Sat 11 11:00
Sat 11 23:00
Sun 12 11:00
Sun 12 23:00
Mon 13 11:00

Time (Hours)

Out of Order Segments summary for entire period

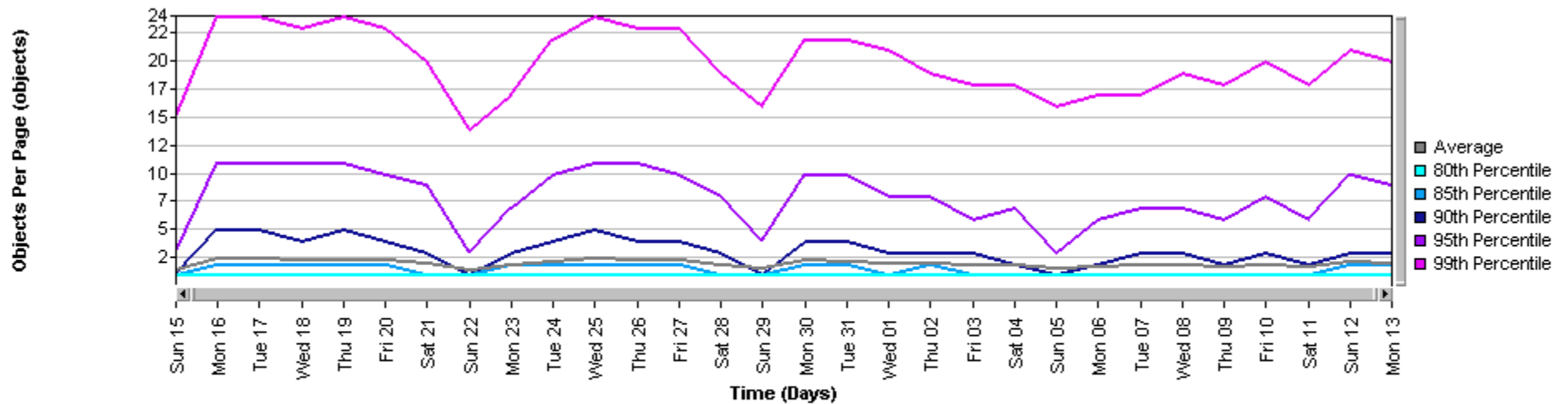
Total OOO Requests	% with 1 OOO	% with 2 OOO	% with 3 OOO	% with 4 OOO	% with 5+ OOO
7456707	60.22%	9.96%	4.49%	3.01%	22.32%

You'll need them because it's all related



Report for Objects Per Page on All-Pages

From: Aug 15 2004 00:00 UTC
To: Sep 13 2004 23:59 UTC



Average 80th %ile 85th %ile 90th %ile 95th %ile 99th %ile

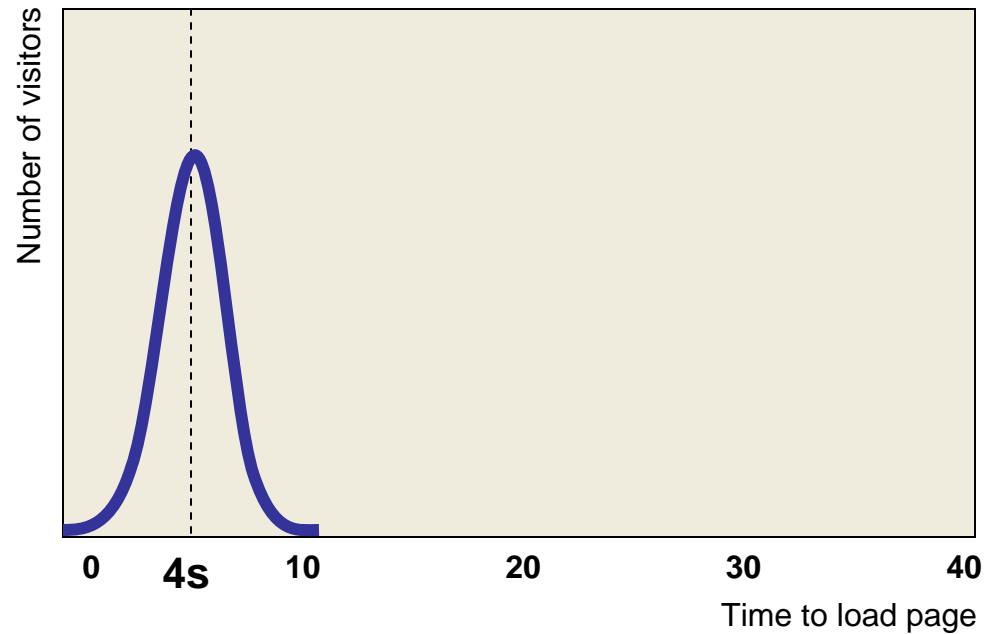
Objects Per Page summary for entire period (objects)

Minimum	Mode	Average	Normalized Average	Std Dev	80 %ile	85 %ile	90 %ile	95 %ile	99 %ile	Maximum
1	1	2.1	1	6	1	2	3	9	21	265

- More objects per page multiplies TCP RTT

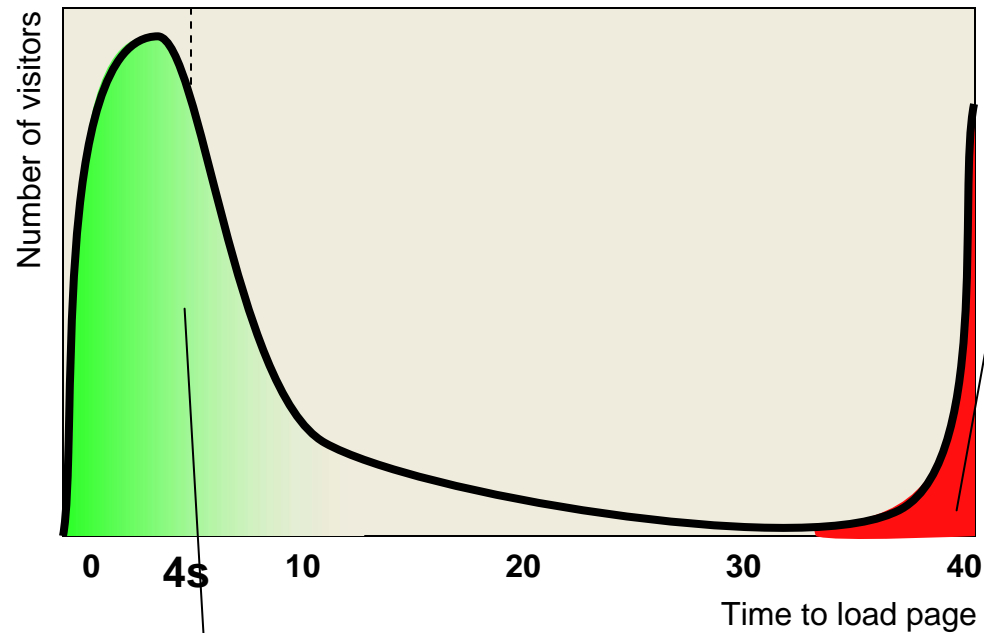
Watch real sessions on production systems

Most people assume that users are evenly distributed around an "average" page load time.



Watch real sessions on production systems

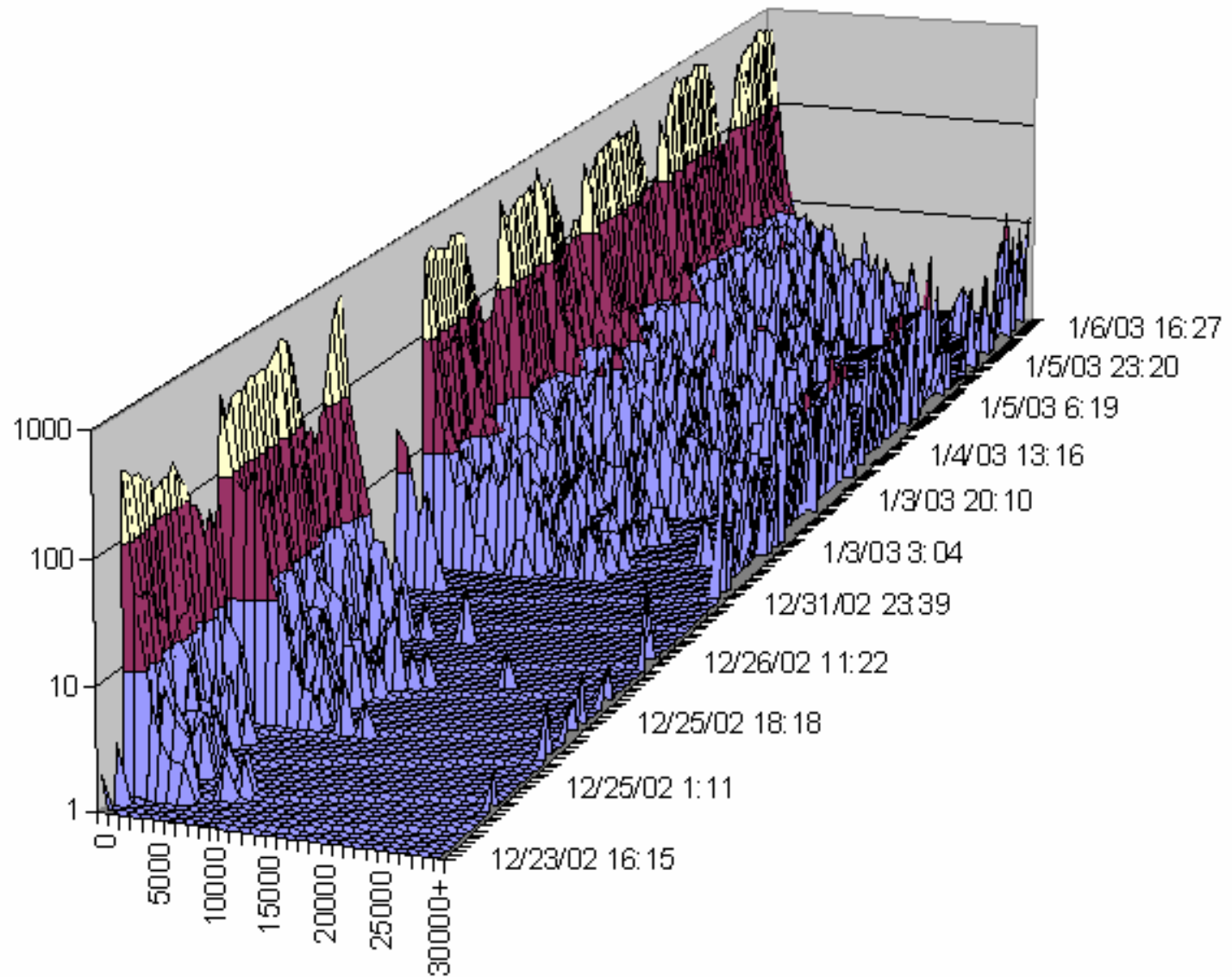
In reality, the distribution of users is a “tailed” curve in which many users are suffering despite an acceptable average, as shown at right.



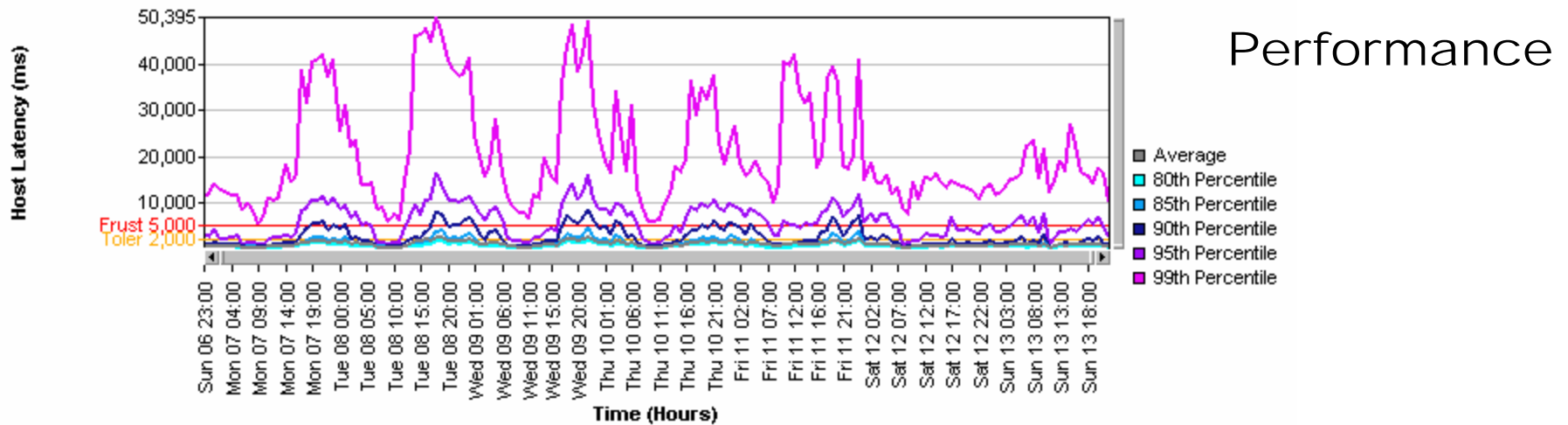
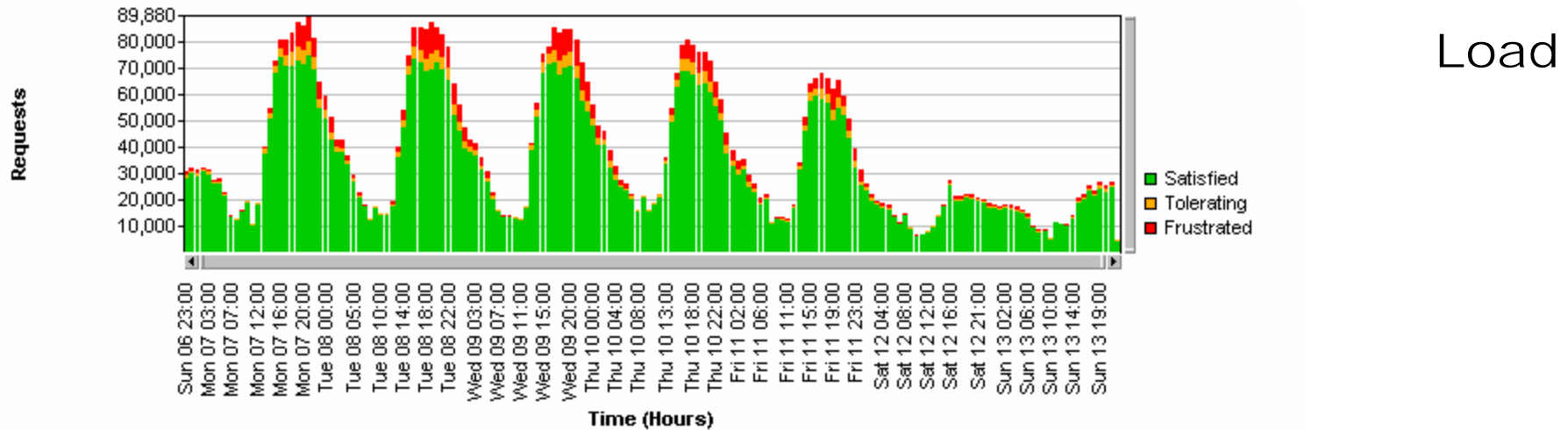
Average page load time is ~4s

6% of users get worse than 40s delay!

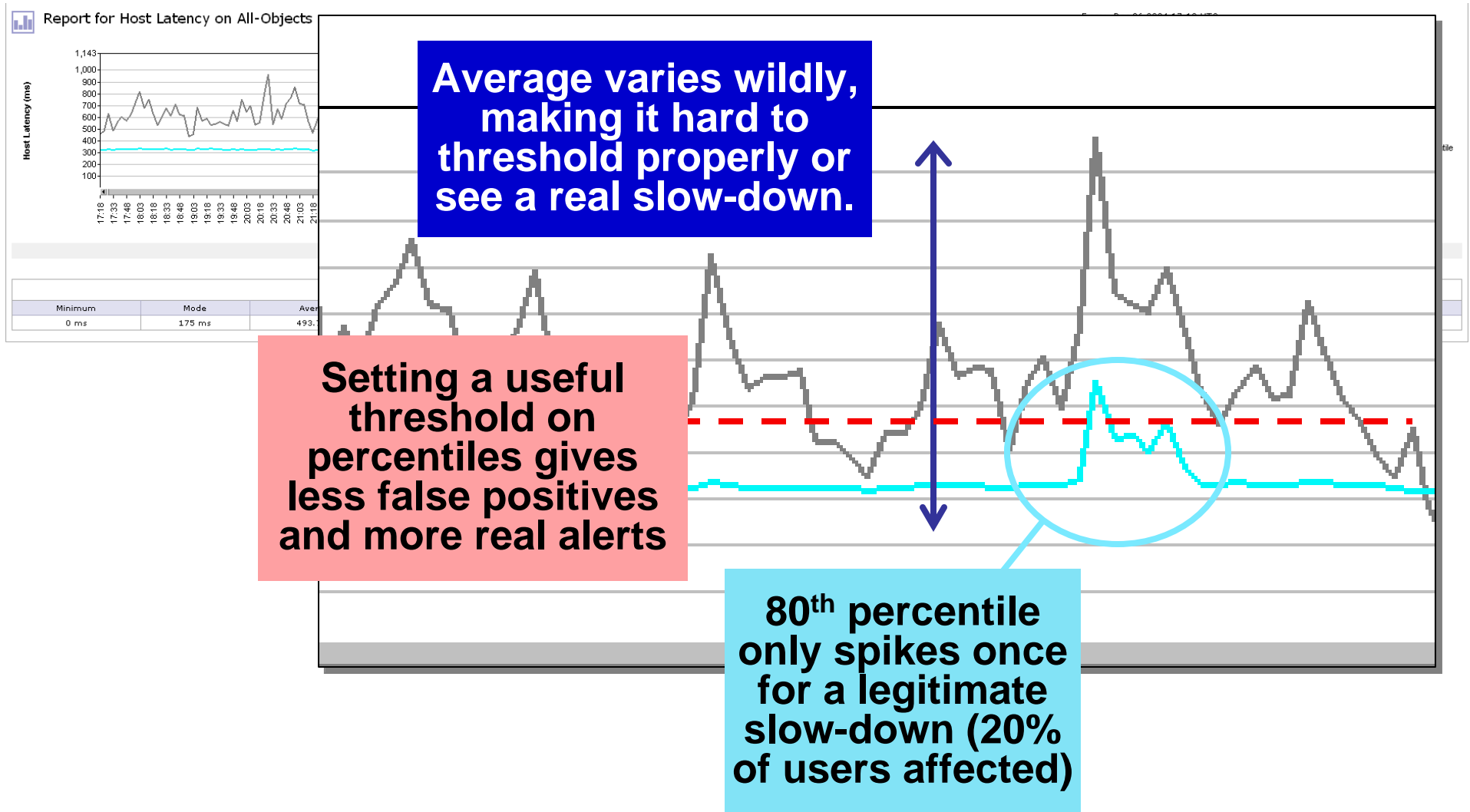
Watch real sessions on production systems



Over busy and light periods: Does the profile show load factors?

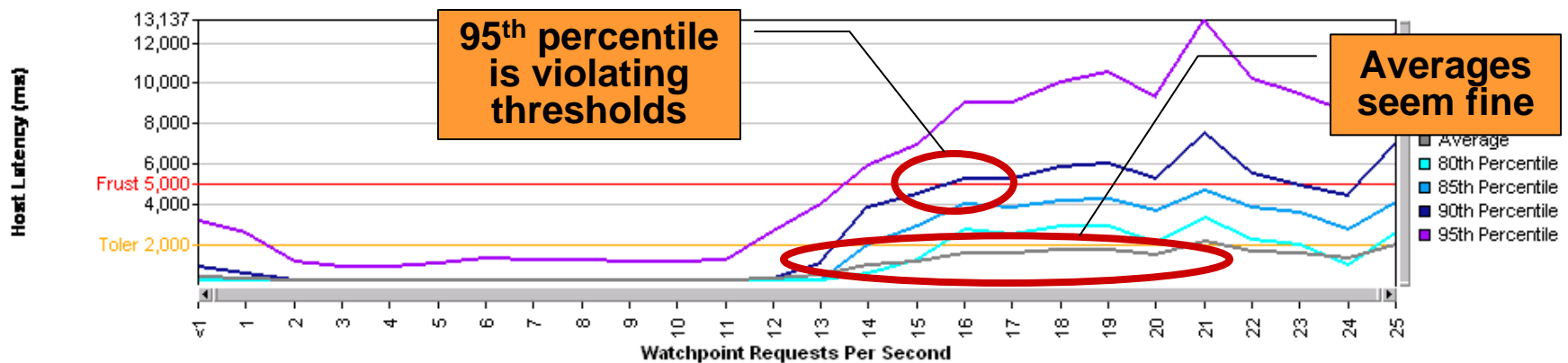


Ignore averages: Watch the percentiles

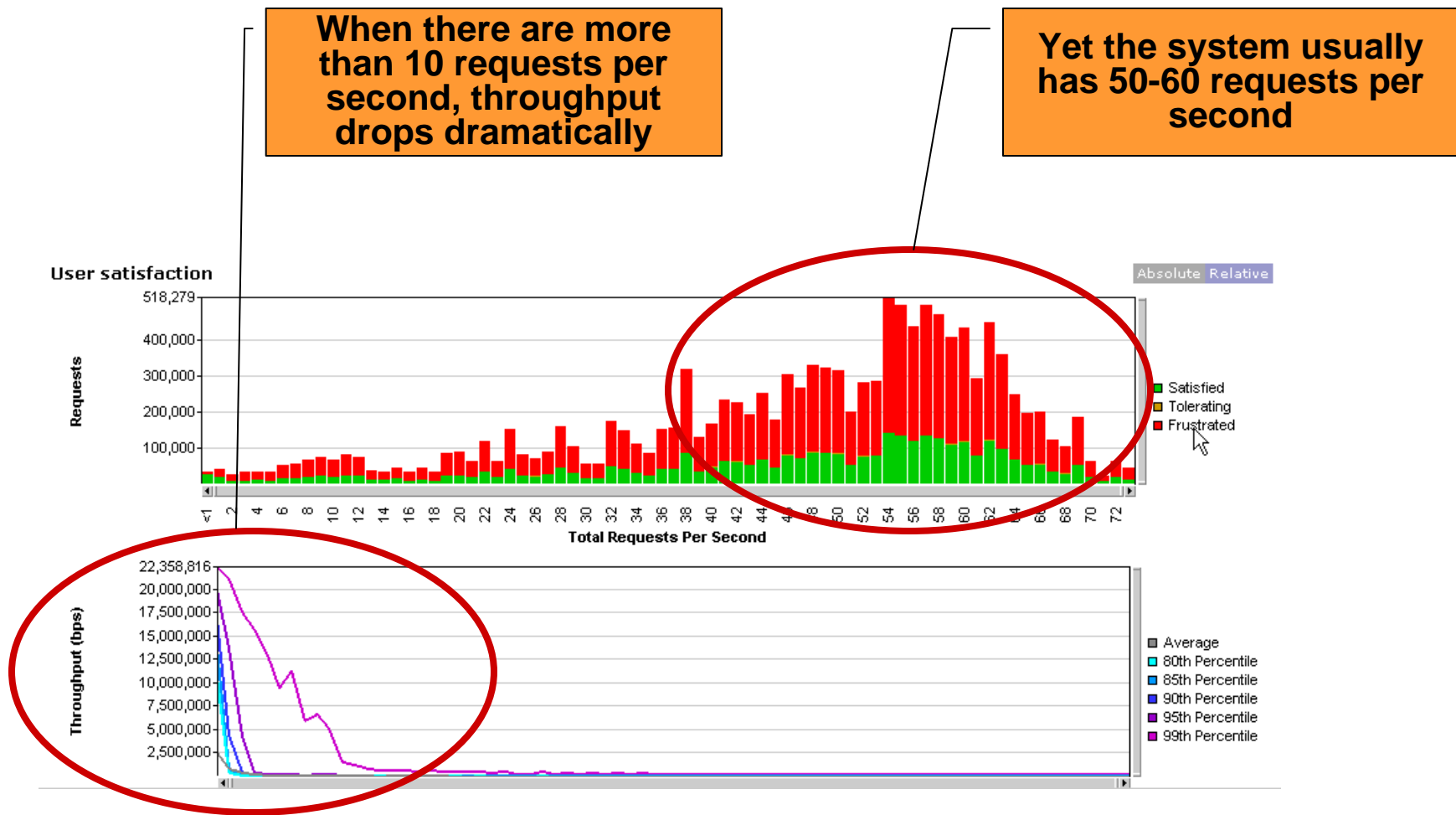


How load relates to performance: Host latency profiling

- **Check percentiles, not just averages**



How load relates to performance: Throughput profiling



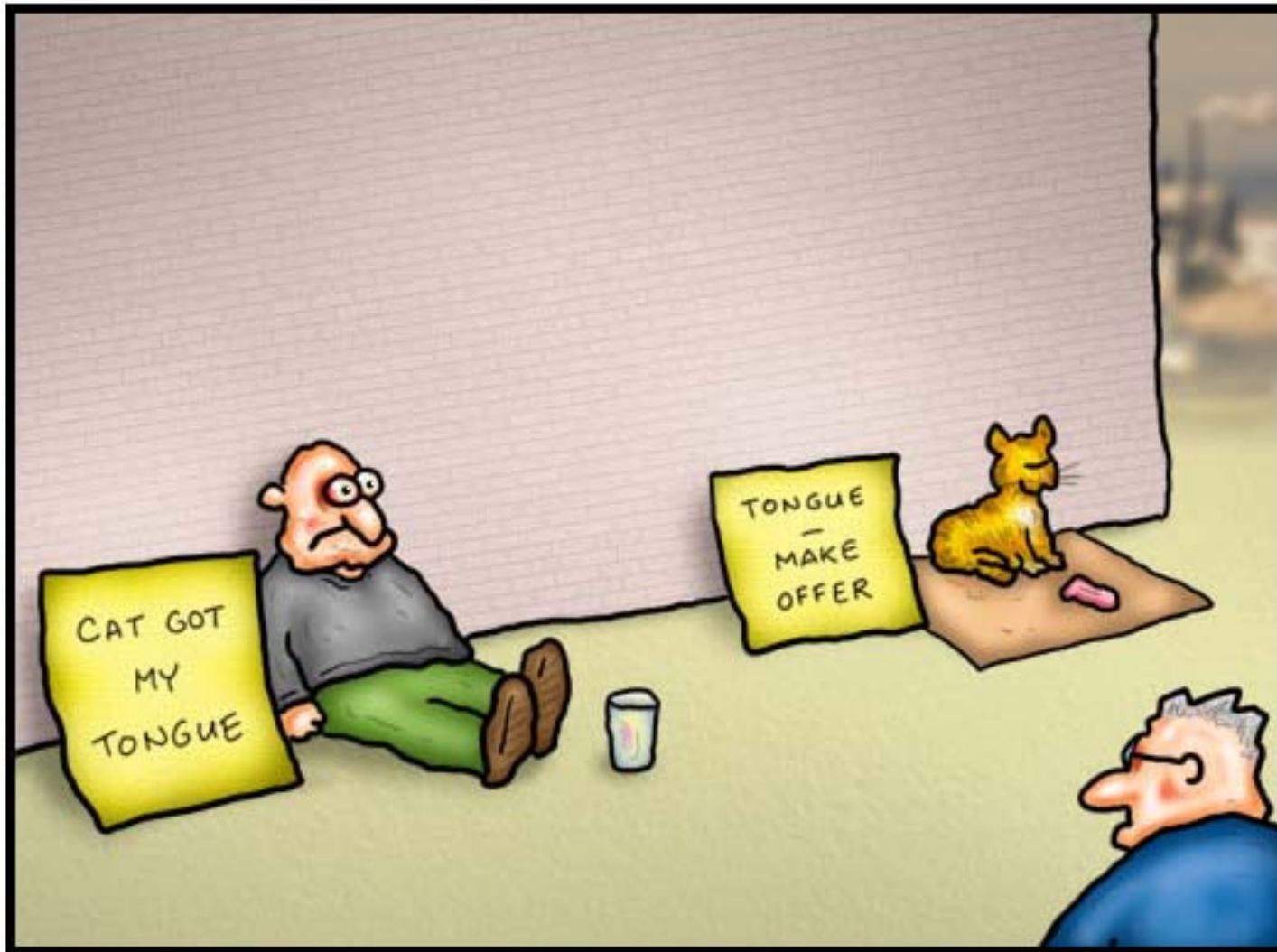
Create a target performance profile

- **The tech support site needs a network time of 500 milliseconds!**
- **The function**
 - Entry of a customer complaint
- **The users**
 - By support teams on the private network
- **The production window**
 - During regular business hours
- **The performance metric**
 - Should have a network time no greater than
- **The value**
 - 500 milliseconds
- **The percentile**
 - 99.9% of the time
- **The capacity**
 - For up to 20 requests per second

Conclusions

- **The old days**
 - How much we can send (throughput)
 - How fast we send it (round-trip time)
 - How often we lose things (packet loss)
- **New factors**
 - Where we put the content (CDNs)
 - How efficiently we tune stacks
 - The impact of QoS and shaping
- **Greater precision is critical**

Questions?



Doctor Fun, <http://www.ibiblio.org/Dave/>