WOPR 12

Resource Monitoring During Performance Testing

A layered approach to metrics....

Date : April 2009

Raymond Rivest

CRIM

Your technology accelerator

Abstract

- Does performance testing requires the big artillery ?
- Do we need tons of measures and graphs to figure out a monster architecture or just a few that demonstrate design inefficiency ?

- Think big ! Think fast and cheap !
  What kind of server is needed to achieve proper response time when the application generates a 1.2Mbps signature per page and the client point has only a 56Kbps aircard ?

Abstract

- Are we looking the metrics at the proper end ? If your application does intensive usage of LDAP and DNS, could the response times of these have an influence over your server size and metrics ?
- What if the technology on which the system relies is simply not made for the business objectives and SLAs ?
- What metrics will be usefull then ?
- These are some of the questions that are often forgotten because focus is set on the Server size and metrics attached to the server... But are these the right metrics ?

Abstract

- **Relevance to theme:**
- Metrics are metrics are metrics... but are the metrics relevant and usefull... or missleading.
- Yeah... the server can deliver... but client will never be able to process anyway, but that's out of our scope, our SLA's are ending on our firewall...

LAN or WAN

- Key differences
  - Bandwidth
  - Latency
  - Packet loss
  - Packet reordering
- Key services
  - DNS
  - LDAP

Some facts

- **How to Calculate TCP throughput for long distance WAN links**
- December 19th, 2008 • Related • Filed Under
- Tags: tcp • waas
- So you just lit up your new high-speed link between Data Centers but are unpleasantly surprised to see relatively slow file transfers across this high speed, long distance link — Bummer! Before you call Cisco TAC and start trouble shooting your network, do a quick calculation of what you should realistically expect in terms of TCP throughput from a one host to another over this long distance link.
- When using TCP to transfer data the two most important factors are the TCP window size and the round trip latency. If you know the TCP window size and the round trip latency you can calculate the maximum possible throughput of a data transfer between two hosts, regardless of how much bandwidth you have.
- Here is how you make the calculation:

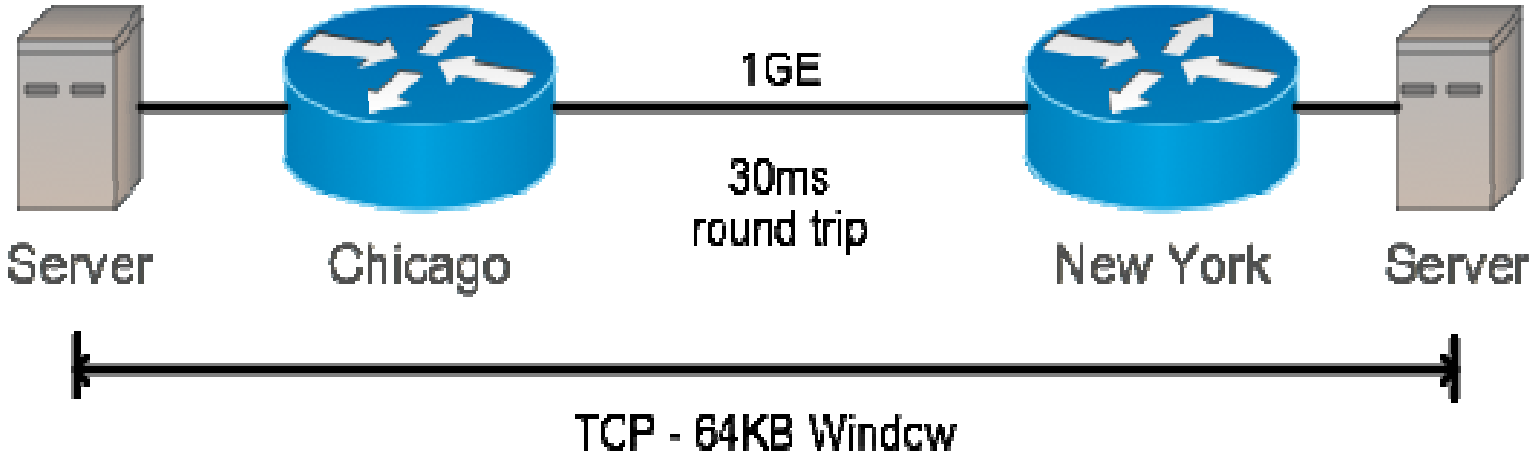
Sources :

http://www.internetworkexpert.org/2008/12/19/how-to-calculate-tcp-throughput-for-long-distance-links/

http://en.wikipedia.org/wiki/Measuring_network_throughput

Big world…

- **TCP-Window-Size-in-bits / Latency-in-seconds = Bits-per-second-throughput**
- So lets work through a simple example. I have a 1Gig Ethernet link from Chicago to New York with a round trip latency of 30 milliseconds. If I try to transfer a large file from a server in Chicago to a server in New York using FTP, what is the best throughput I can expect?



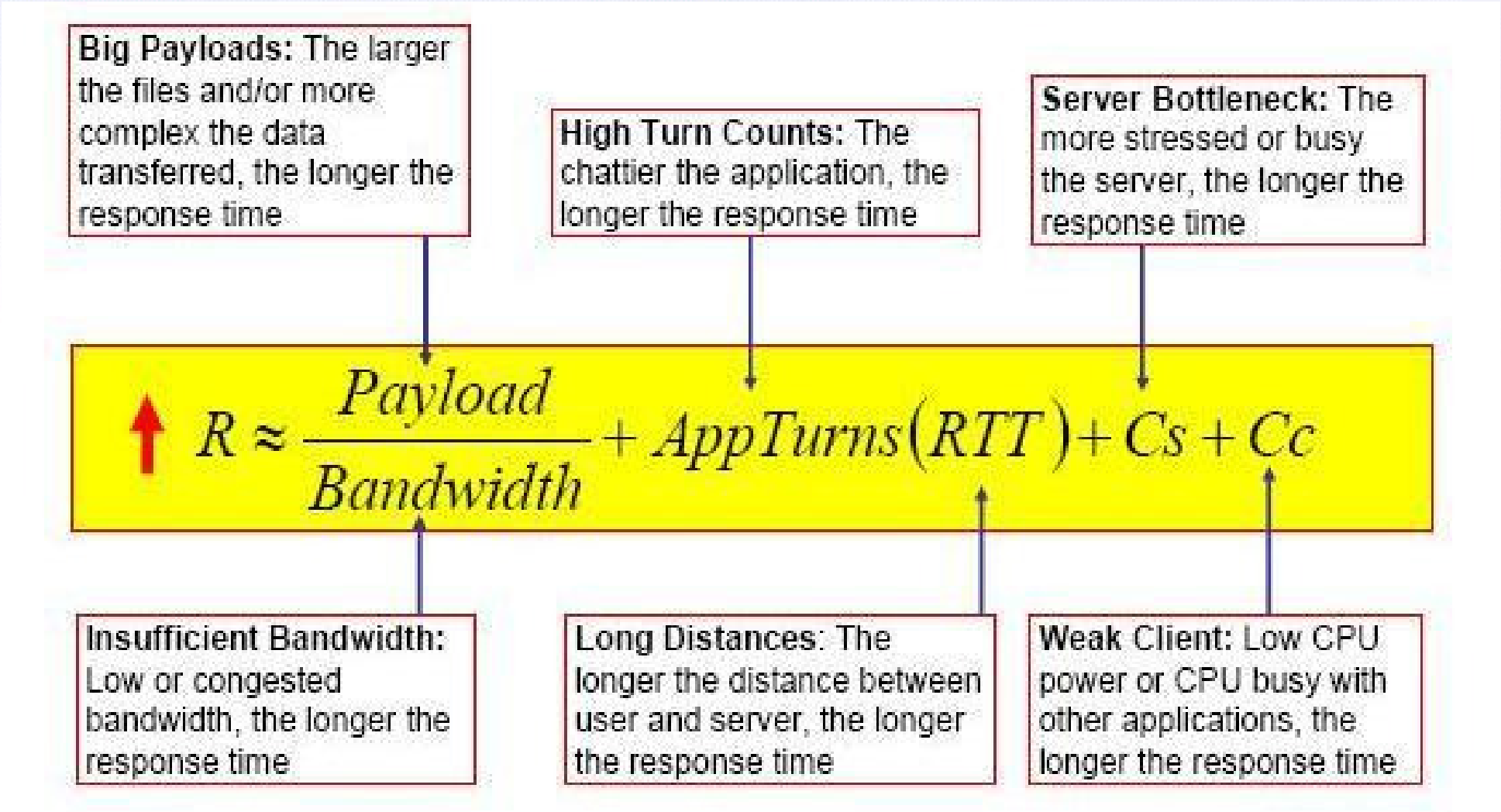
More

- First lets convert the TCP window size from bytes to bits. In this case we are using the standard 64KB TCP window size of a Windows machine.
- 64KB = 65536 Bytes. 65536 * 8 = **524288 bits**
- Next, lets take the TCP window in bits and divide it by the round trip latency of our link in seconds. So if our latency is 30 milliseconds we will use 0.030 in our calculation.
- 524288 bits / 0.030 seconds = **17476266** bits per second throughput = **17.4 Mbps maximum possible throughput.**

Some more

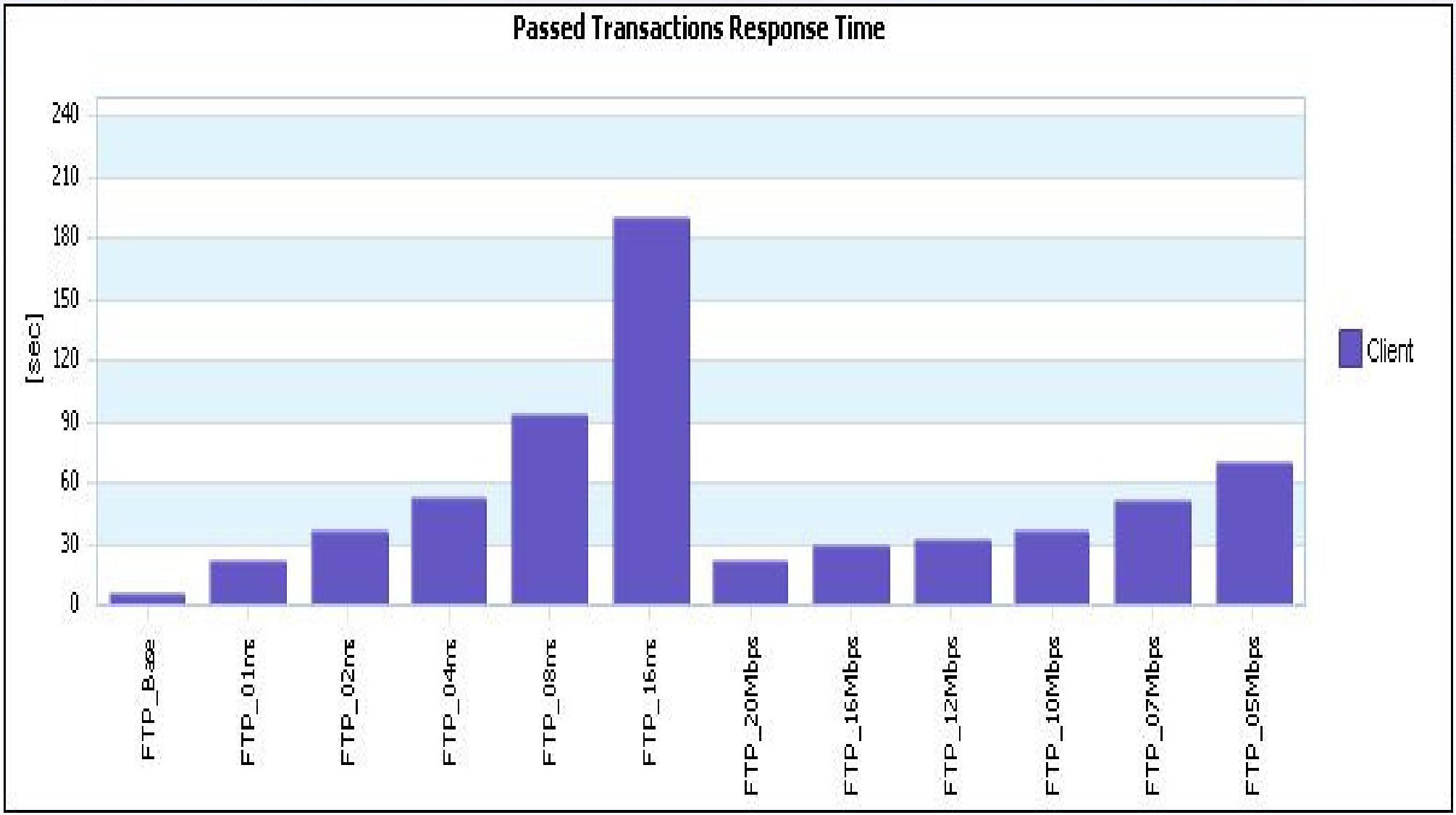
- So, although I may have a 1GE link between these Data Centers I should not expect any more than 17Mbps when transferring a file between two servers, given the TCP window size and latency.
- What can you do to make it faster? Increase the TCP window size and/or reduce latency.
- To increase the TCP window size you can make manual adjustments on each individual server to negotiate a larger window size. This leads to the obvious question: What size TCP window should you use? We can use the reverse of the calculation above to determine optimal TCP window size.

One more

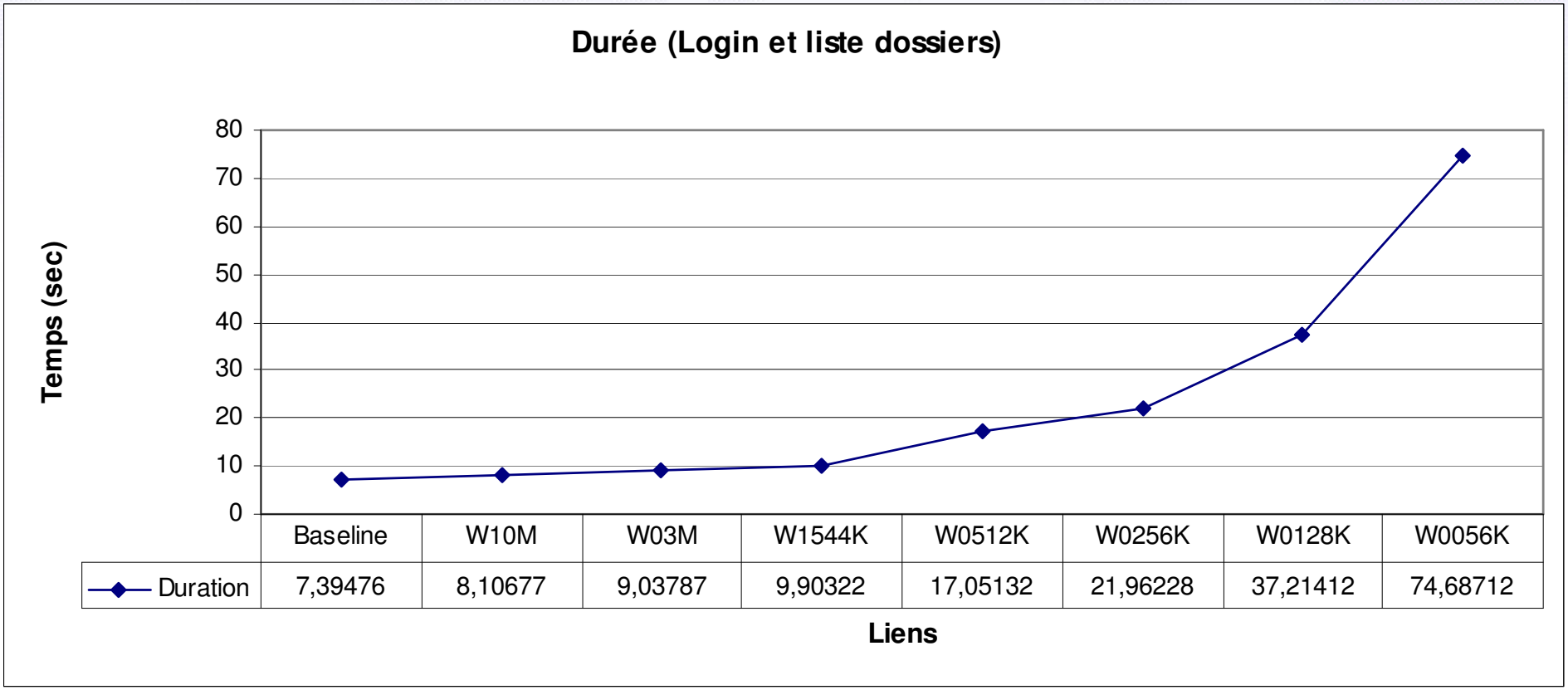


Source : http://www.webperformancematters.com/journal/2007/7/24/latency-bandwidth-and-response-times.html

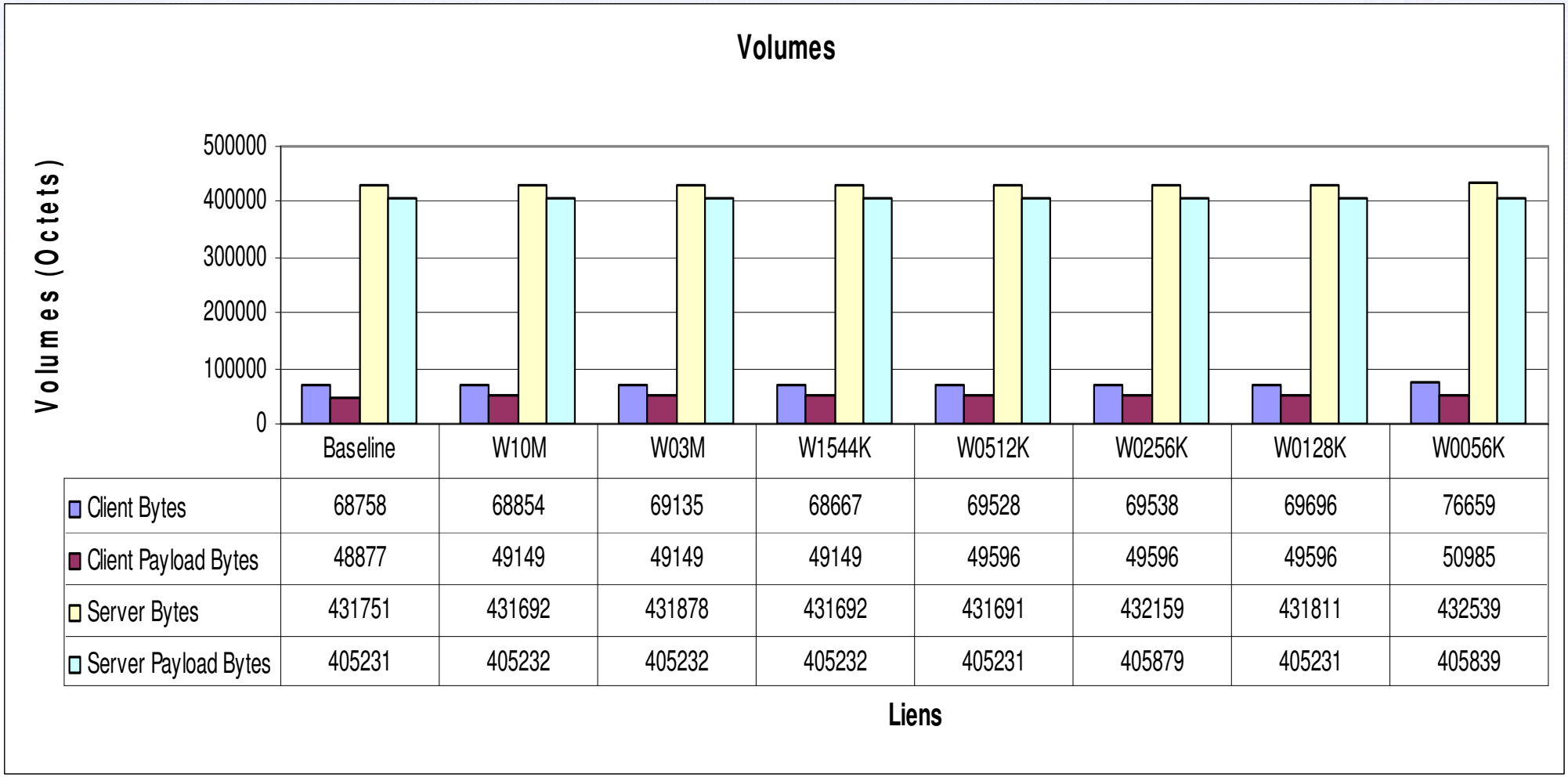
Straight talk



Think big – Client side

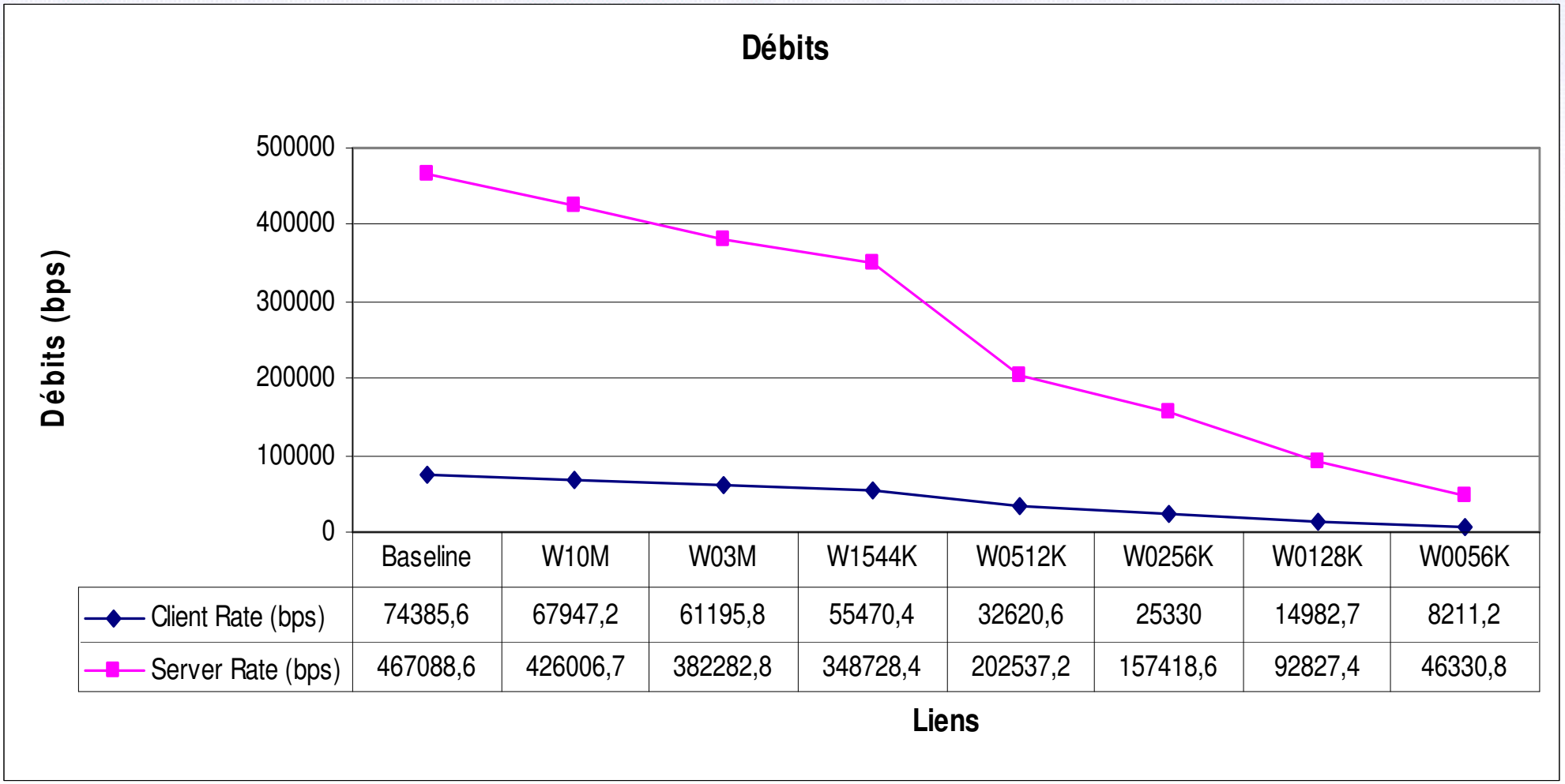


Size matter ?



	Baseline	W10M	W03M	W1544K	W0512K	W0256K	W0128K	W0056K
Client Bytes	68758	68854	69135	68667	69528	69538	69696	76659
Client Payload Bytes	48877	49149	49149	49149	49596	49596	49596	50985
Server Bytes	431751	431692	431878	431692	431691	432159	431811	432539
Server Payload Bytes	405231	405232	405232	405232	405231	405879	405231	405839

Throughput anyone ?



	Baseline	W10M	W03M	W1544K	W0512K	W0256K	W0128K	W0056K
Client Rate (bps)	74385,6	67947,2	61195,8	55470,4	32620,6	25330	14982,7	8211,2
Server Rate (bps)	467088,6	426006,7	382282,8	348728,4	202537,2	157418,6	92827,4	46330,8

A bit more…

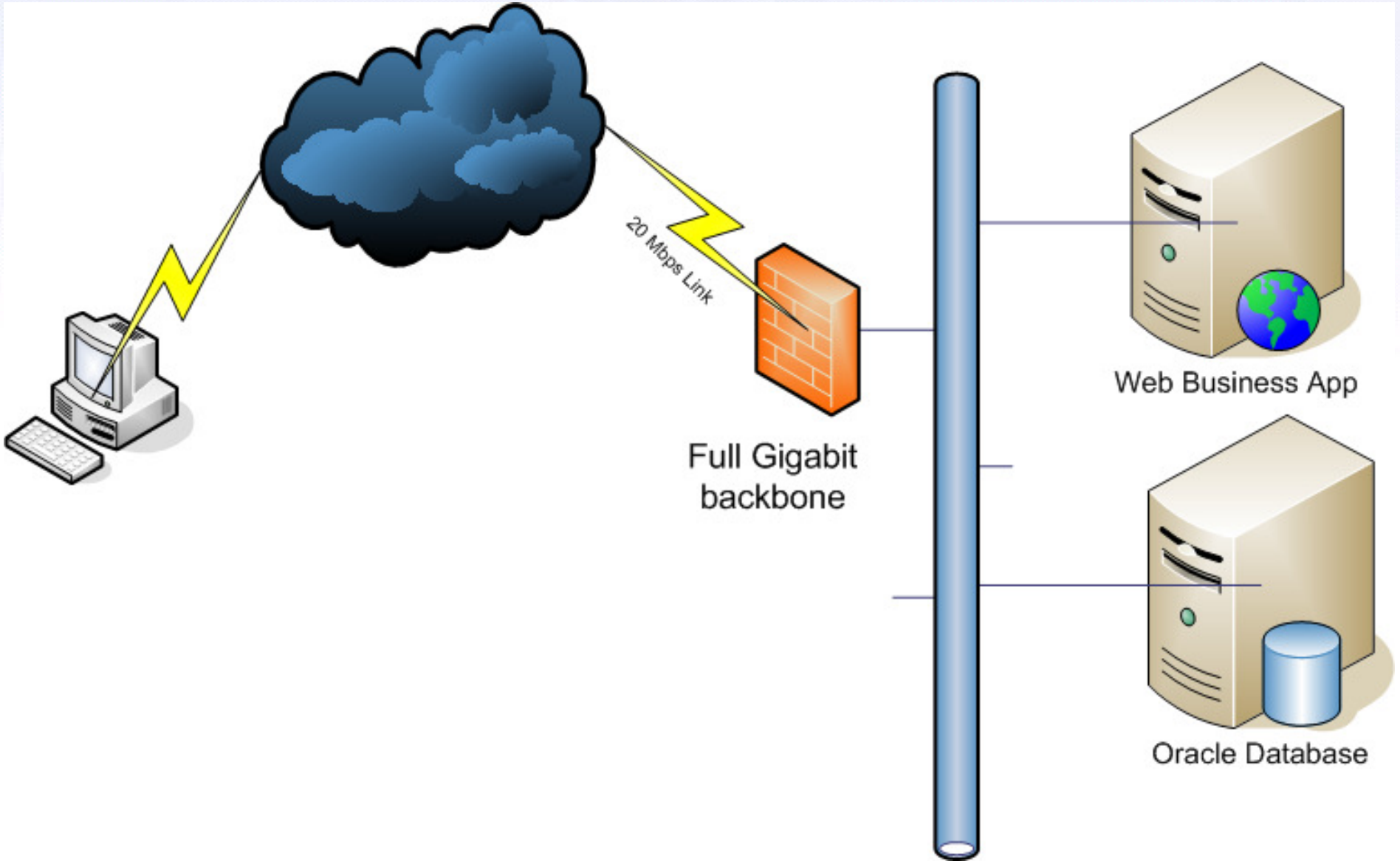
Downstream speed (download)	Upstream speed (upload)	Traffic (bps) entering (to server)	Traffic (bps) outgoing (to user)	Average Length O/ Average Length E	Transmit Time/ Server & Client Time
3000000	3000000	**72824,9**	**2881,1**	31,0	1%

	New UTILIZATION levels					
	New average utilization during **Input**					
Input	**Previous Input Utilization**	Number of users 1	5	10	15	
1	1%	3%	13%	25%	37%	
2	3%	5%	15%	27%	39%	
3	10%	12%	22%	34%	46%	
4	30%	32%	42%	54%	66%	
5	60%	62%	72%	84%	s.o.	
	New average utilization during **Output**					
Output	**Previous Output Utilization**	Number of users 1	5	10	15	
1	1%	1%	1%	2%	2%	
2	3%	3%	3%	4%	4%	
3	10%	10%	10%	11%	11%	
4	30%	30%	30%	31%	31%	
5	60%	60%	60%	61%	61%	

Odds of making it…

	DEGRADATION of TRANSMISSION per Channel				
	Degradation relative to **Input** transmission time				
Input / choice	**Previous Input Utilization**	Number of users 1	5	10	15
	1%	3%	14%	32%	58%
	3%	3%	14%	33%	60%
	10%	3%	16%	37%	68%
	30%	4%	21%	53%	**108%**
	60%	6%	44%	154%	s.o.
	Degradation relative to **Output** transmission time				
Output / choice	**Previous Output Utilization**	Number of users 1	5	10	15
	1%	0%	0%	1%	1%
	3%	0%	0%	1%	2%
	10%	0%	1%	1%	2%
	30%	0%	1%	1%	2%
	60%	0%	1%	2%	4%
	% Global DEGRADATION of TRANSMISSION				
Previous Input Utilization	**Previous Output Utilization**	Number of users 1	5	10	15
4	4	0%	1%	3%	5%
Ponderated Utilisation (input + output)		30%	30%	31%	32%
New Input Utilization		32%	42%	54%	66%
New Output Utilization		30%	30%	31%	31%
Previous Input Utilization		30%	30%	30%	30%
Previous Output Utilization		30%	30%	30%	30%

So given a model



Question now is….

- Which metrics are usefull ?
- The answer
  - Hardware matters !
  - But do you know what's under the hood ?