

# Web Services Performance

Paul Greenfield  
CSIRO ICT Centre

# Web Services Preconceptions

- Performance...
  - Bound to be excellent with all that XML and HTTP ☺
  - Machines and networks are getting faster anyway...
- Anyway... WS protocols designed for interoperability and capabilities, not raw speed...
- Straw poll
  - Are Web Services slow??
  - Are Web Services verbose network hogs??

# Web Services Performance

- XML-based SOAP protocol
  - XML messages being sent around the network
  - XML being handled on clients and servers
  - Protocols designed for interoperability and capabilities rather than raw speed...
- Normally runs on top of HTTP
  - Messages pass through Web Servers too?
- Cost of other services
  - Encryption, signatures, policy
- Is performance really a problem?
  - Machines and networks are getting faster anyway...

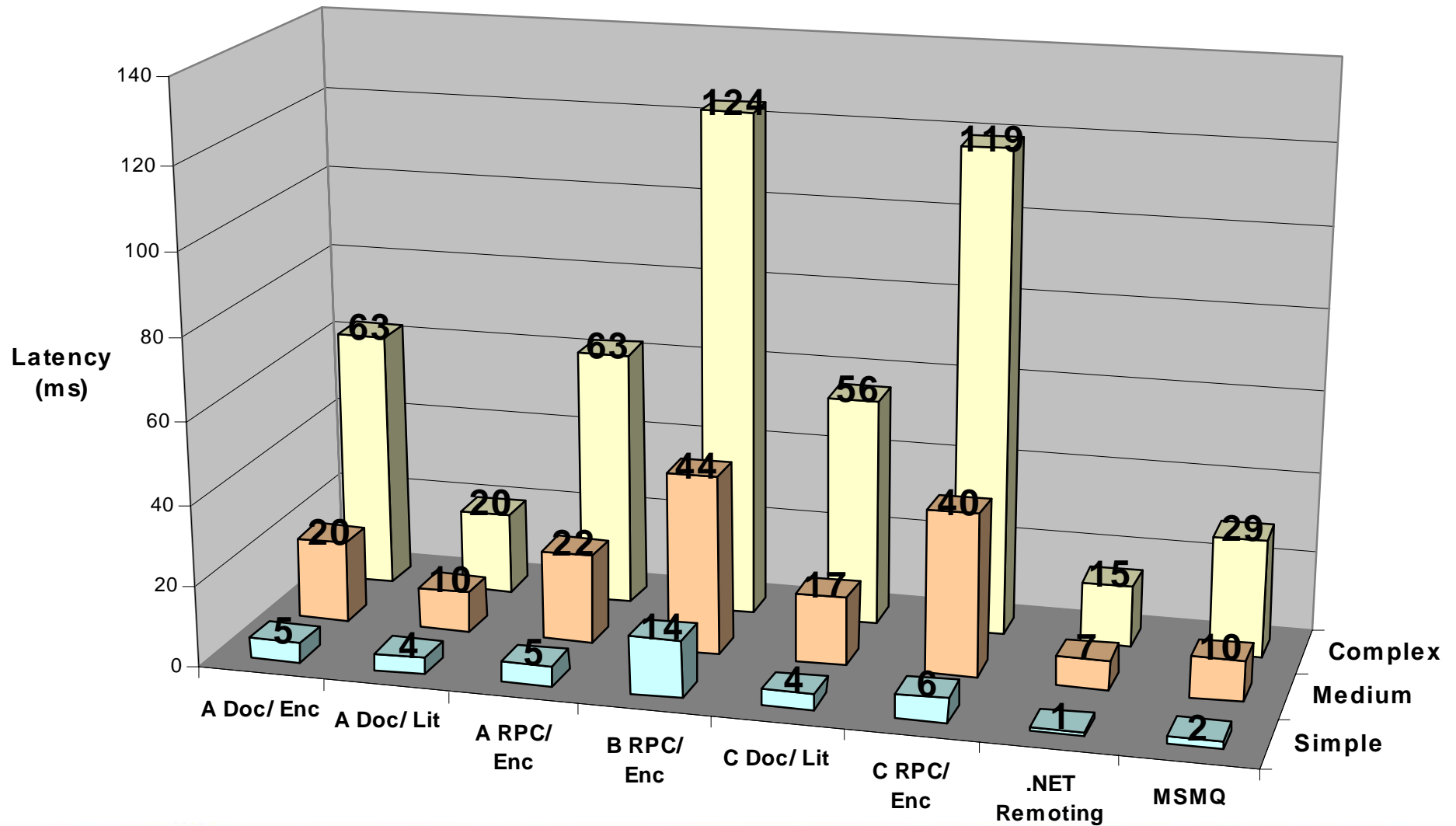
# Performance Factors

- Processor time for XML encoding/decoding
  - Is XML slower than binary alternatives?
- Number and size of messages passed around
  - Does verbosity cost performance?
- Processor time taken for transport protocols
  - TCP/IP and HTTP
- Network delays
  - Switches, routers, ...
- Speed of light delays
  - Caused by synchronous message exchanges
  - Takes 1.5mS from Sydney to Canberra in glass

# Are All SOAPs Equal?

- Took three commercial SOAP products (2003)
- Compared to two non-SOAP alternatives
- Effects of different SOAP encoding styles?
- Measured single-thread call latency...
  - Client call to response received
  - Small, medium and complex messages
    - 1.5KB, 7.5KB, 13KB (call & response)
  - Quad processor Xeon servers
  - Connected over 100M LAN
- Let's us focus on what's possible
  - And not generalise from poor implementations

# Are All SOAPs Equal?

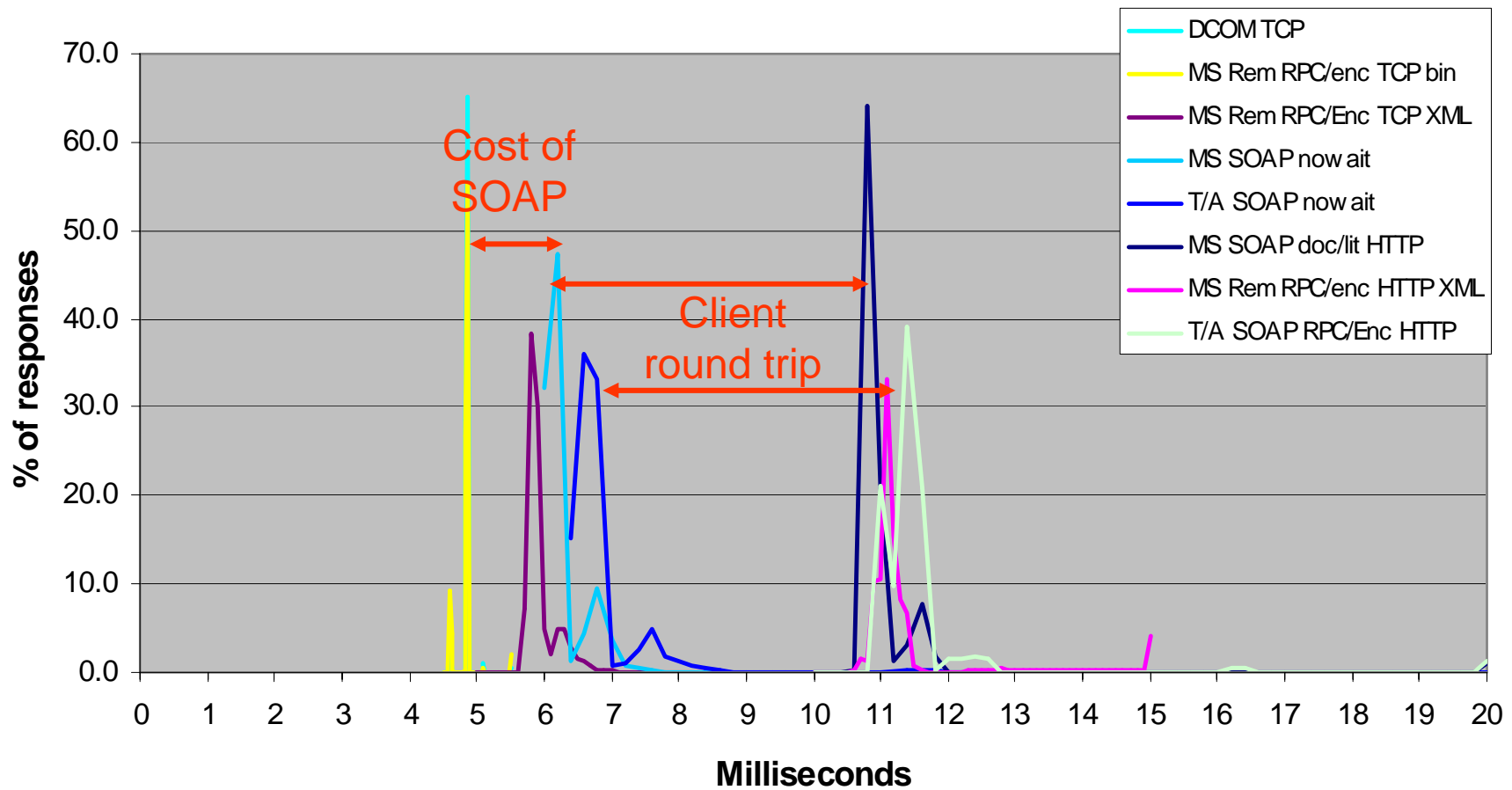


# SOAP Over WAN

- Sydney-Canberra over CeNTIE network
  - 100M locally, 1G for core network
  - Lightly-loaded, low contention network
- Single-threaded clients
  - 2.8GHz P4 Dell desktop systems (HT turned off)
  - Windows 2003 Server, .NET 1.1 client
- Initial tests
  - SOAP (Microsoft ASMX & Apache)
  - Alternatives (MS DCOM, .Net remoting)
  - Very simple messages + simple application call

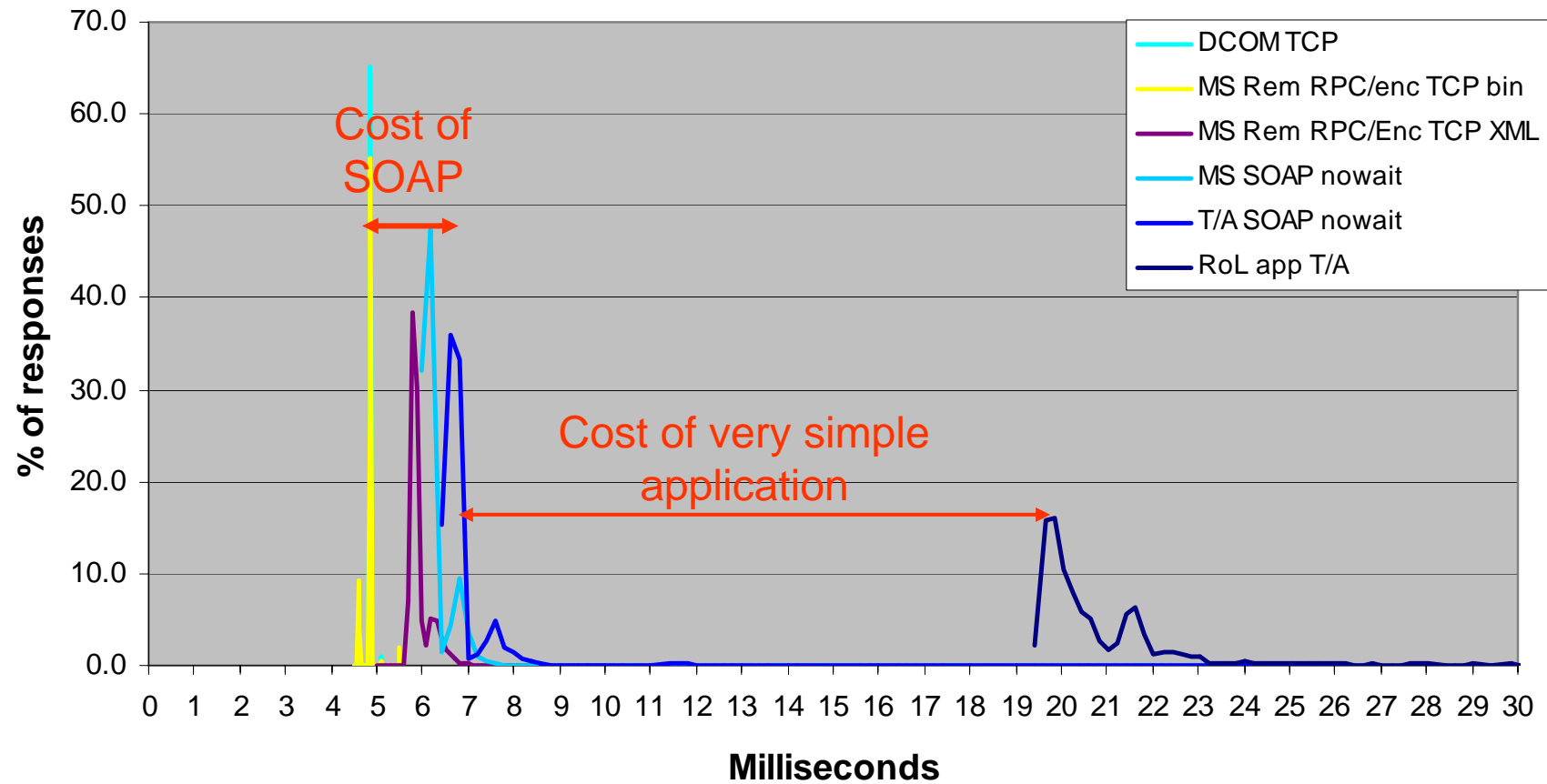
# SOAP Over WAN (Tuned)

**SOAP Latencies**  
Syd-Cbr-Syd over 1Gpbs CeNTIE WAN



# Performance in Perspective

## SOAP Latencies Syd-Cbr-Syd over 1Gpbs CeNTIE WAN

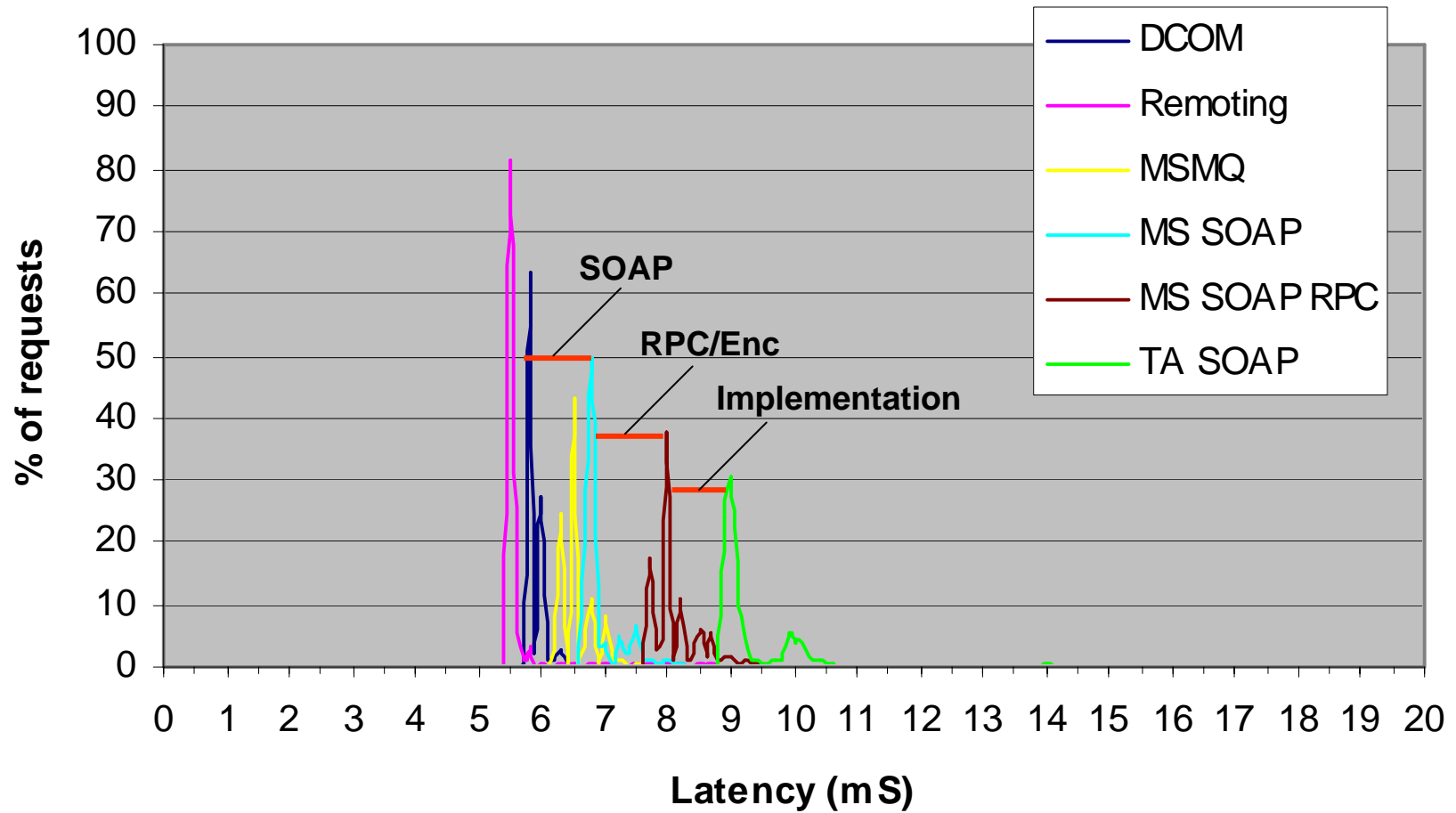


# Scalability

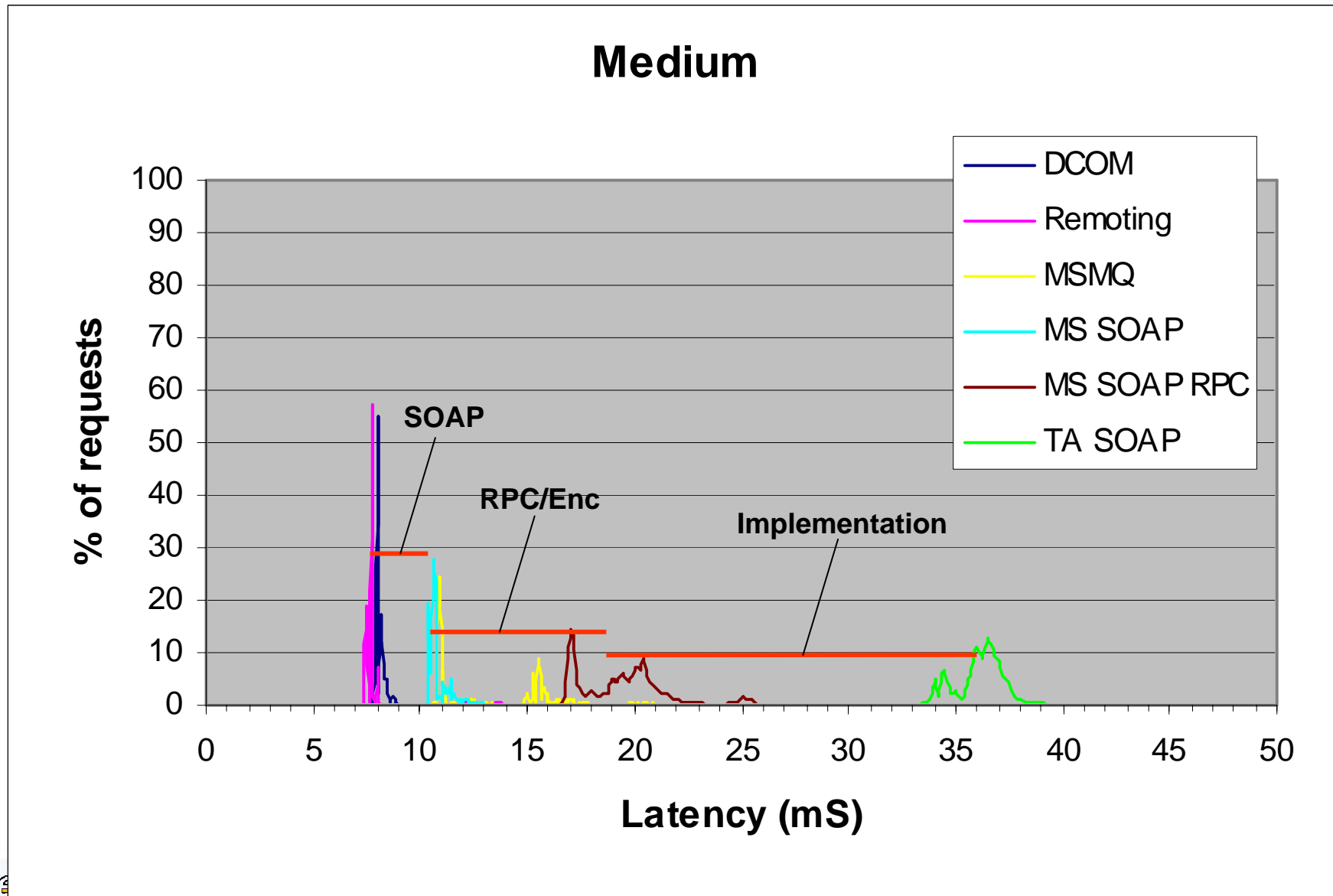
- OK – performance fair for small messages...
  - About 25% slower than fast binary alternatives
  - Overheads easily swamped by transaction time
  - High processor and network usage observed
- What happens as messages get bigger?
  - Does performance drop off rapidly?
- What are limiting factors on scalability
  - Processor time parsing XML?
  - Network time sending all those bytes?
  - How would you build highly-scalable SOAP servers?

# Scalability

## Simple

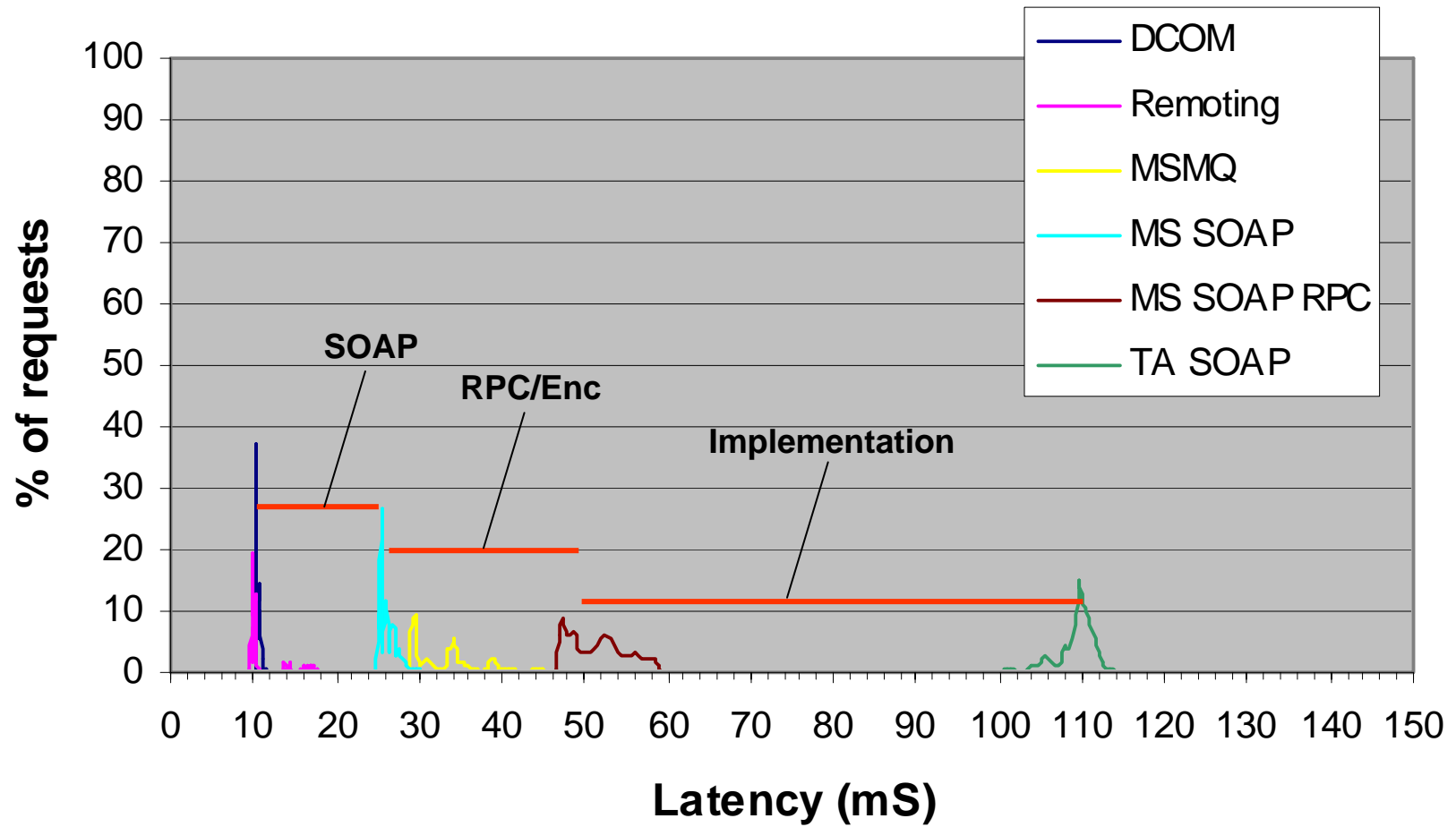


# Scalability

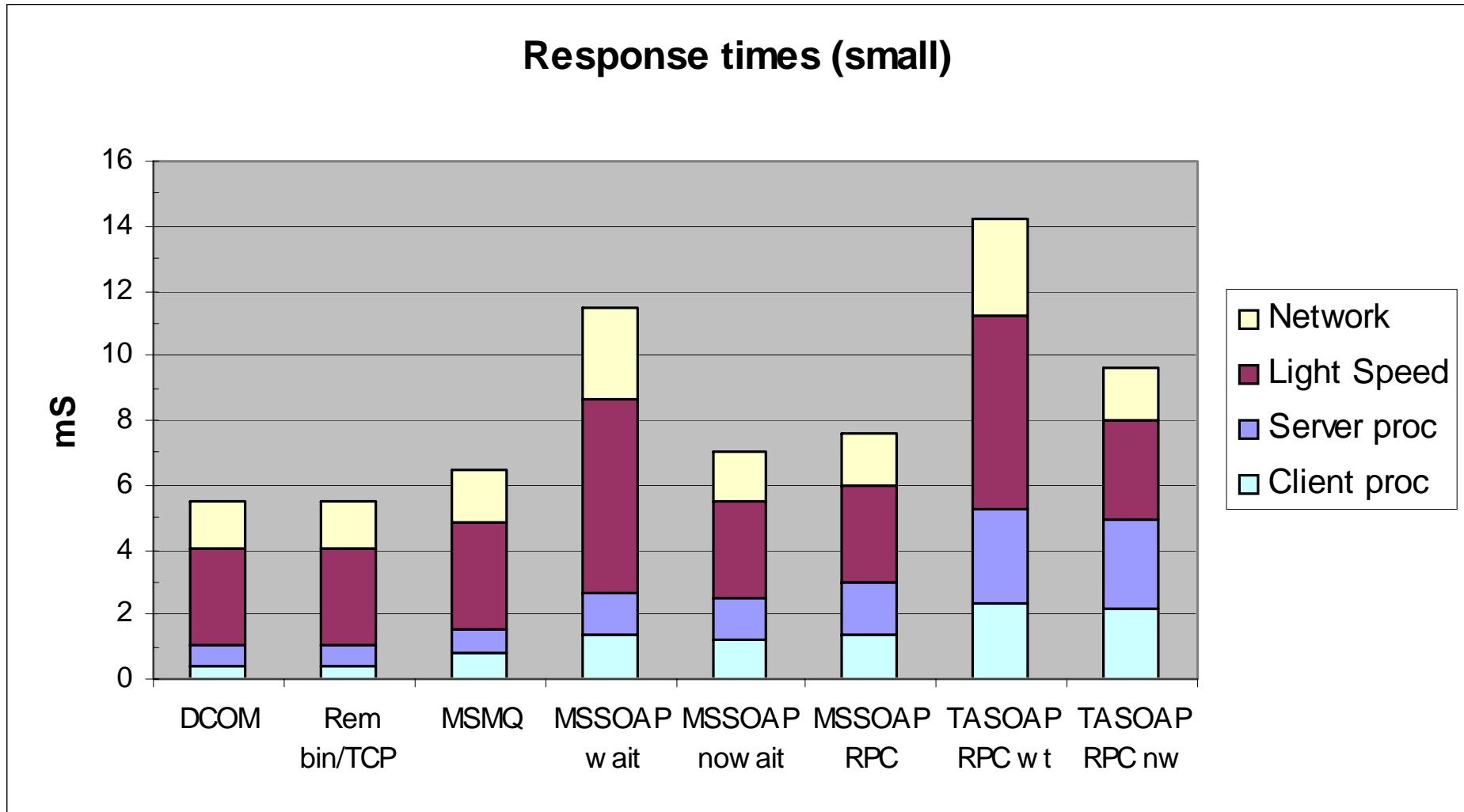


# Scalability

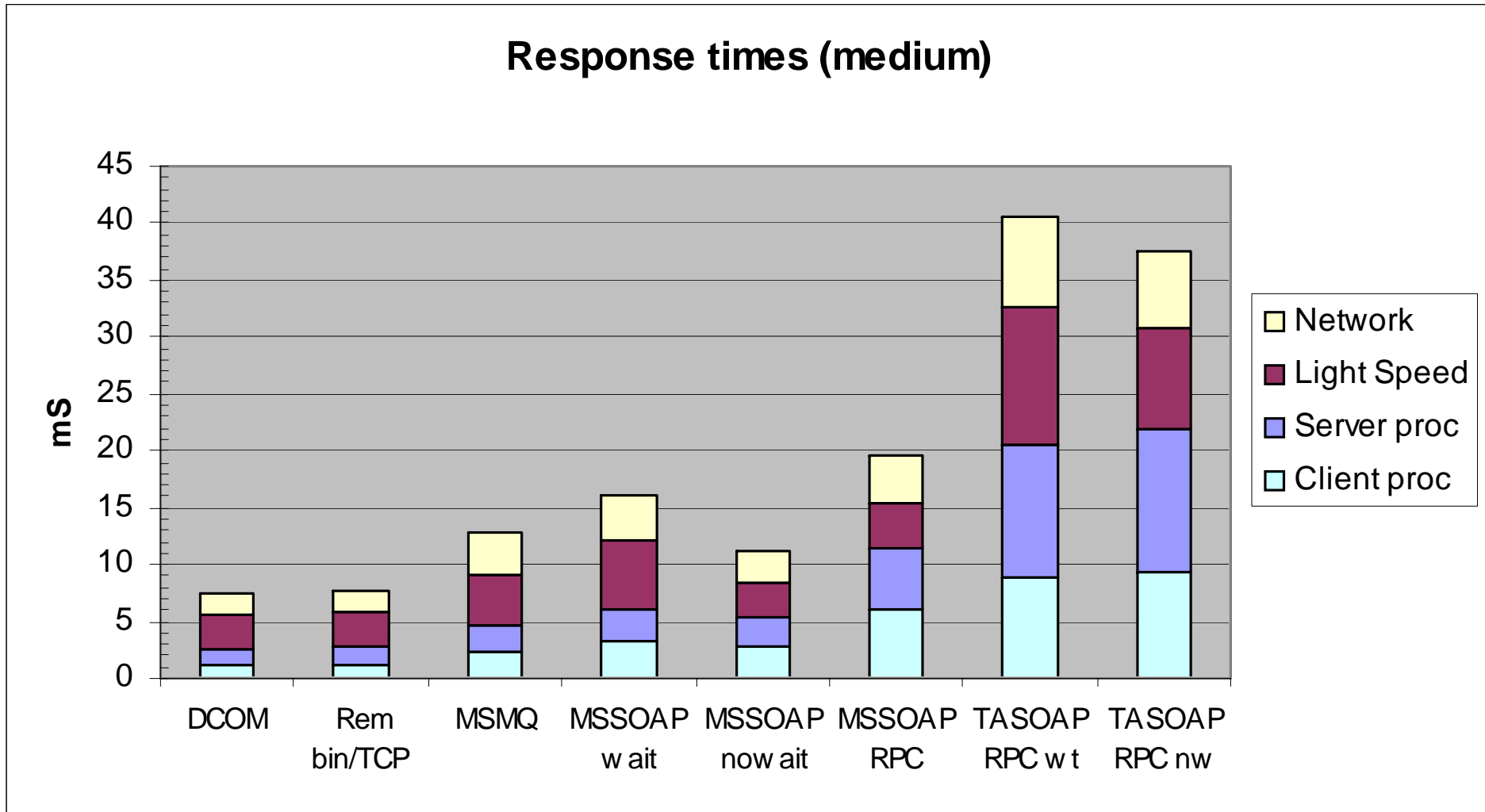
## Complex



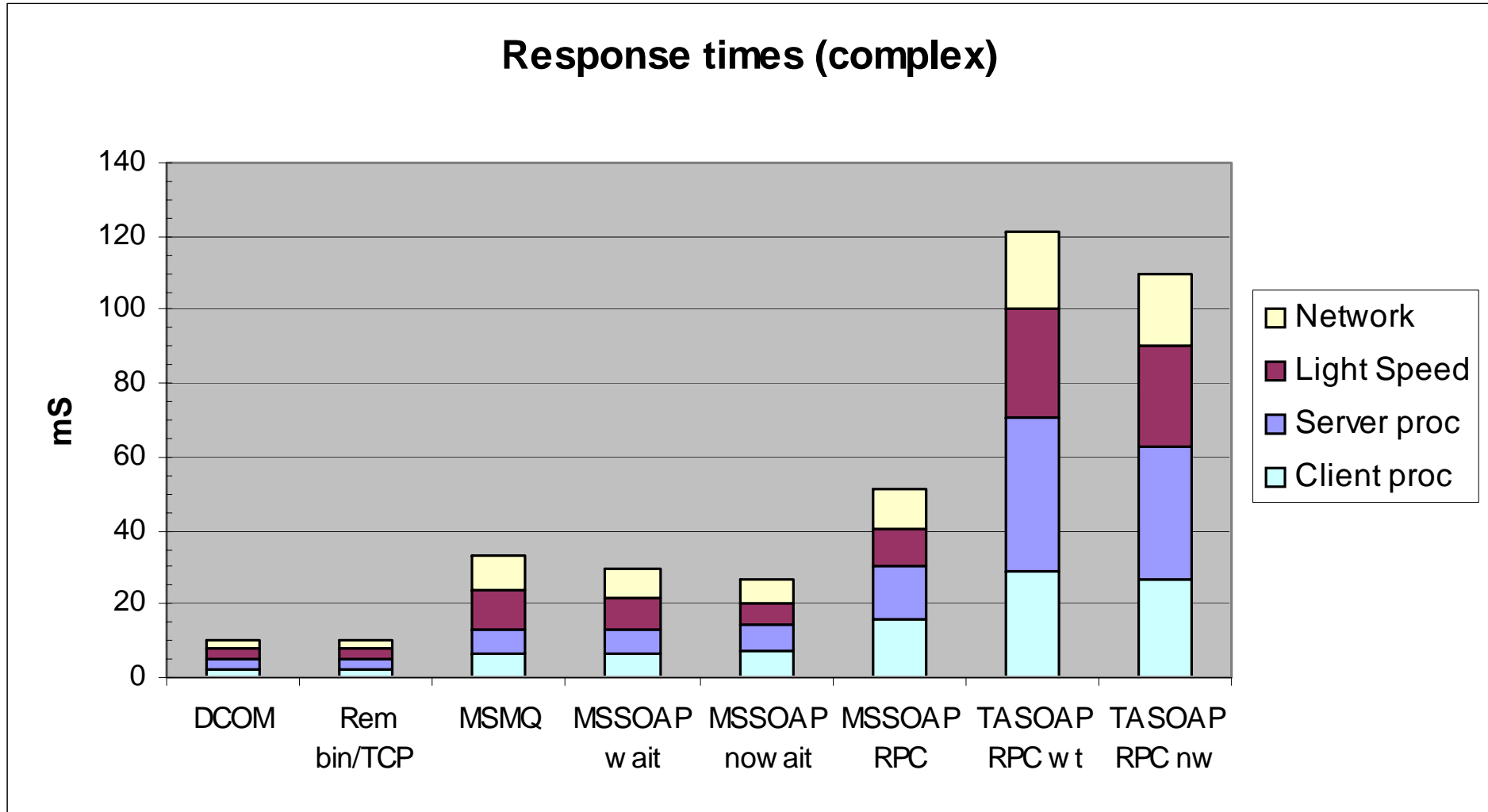
# What's Happening?



# What's Happening?



# What's Happening?

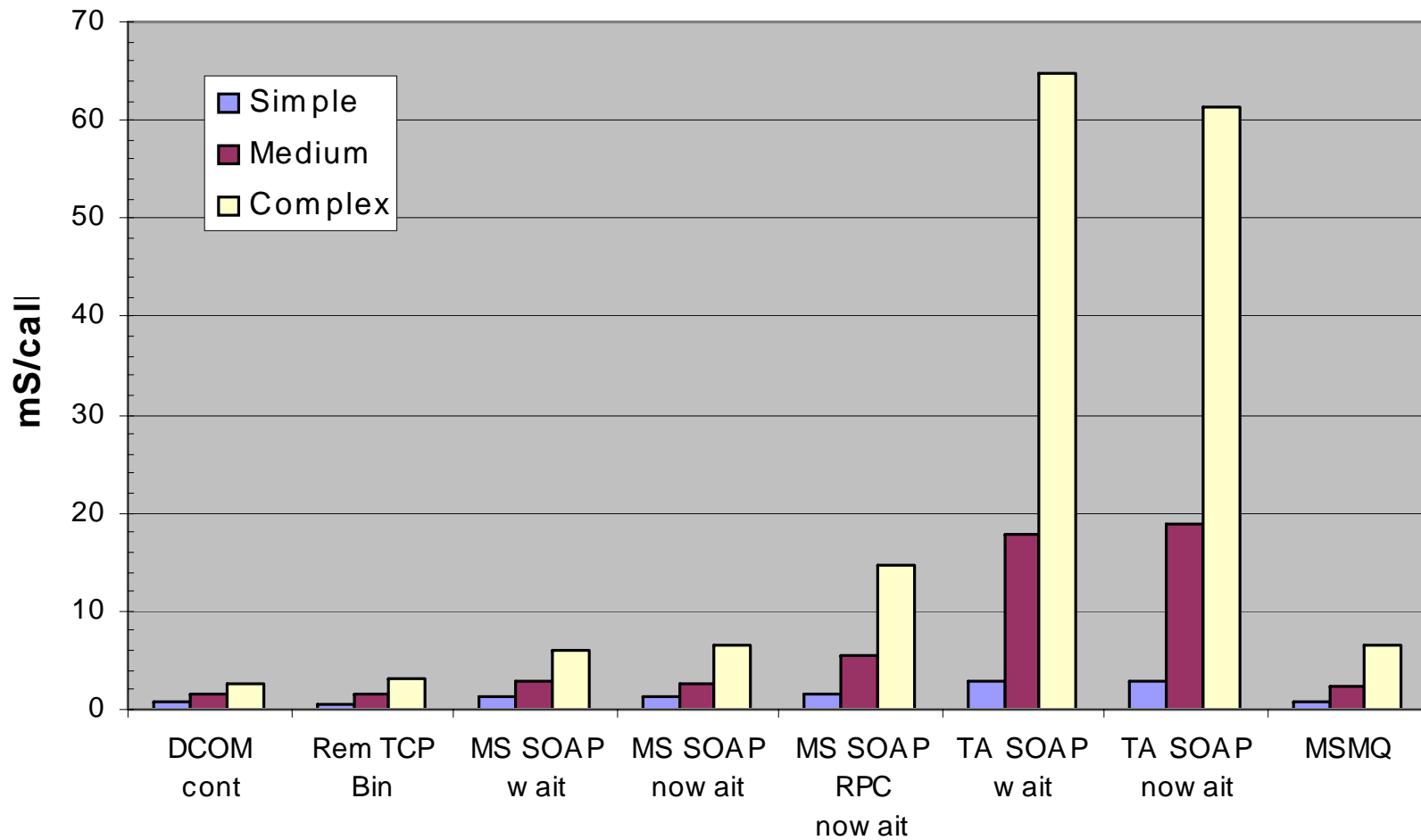


# Network View

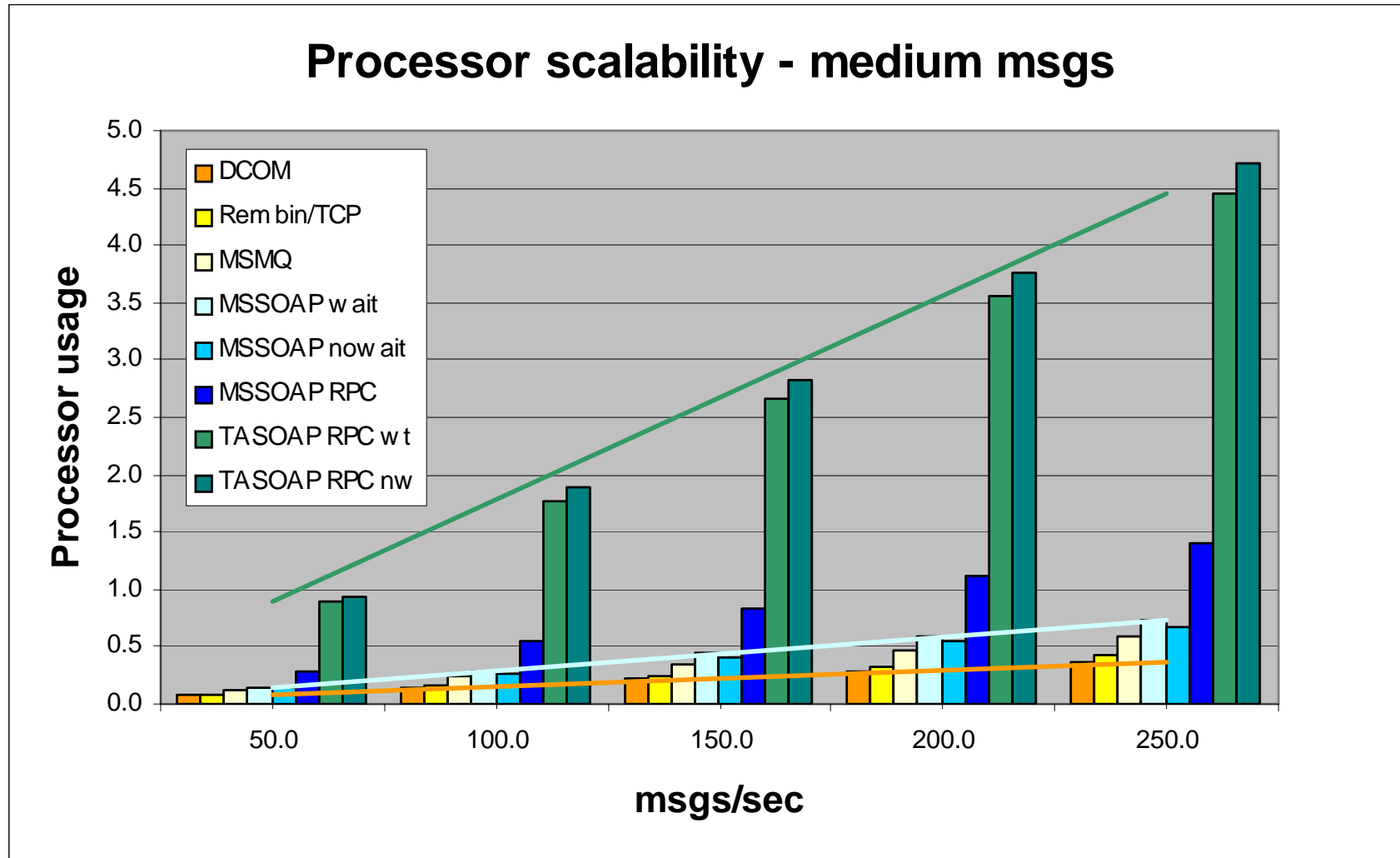
	Simple			Medium			Complex		
	Bytes	Packets	Waits	Bytes	Packets	Waits	Bytes	Packets	Waits
DCOM	1660	2	1	7402	6	1	13134	11	1
.Net remoting	1819	6	1	7511	9	1	13174	14	1
MS SOAP wait	2482	4	2	16516	15	2	44979	30	3
MS SOAP nowait	2514	5	1	16434	14	1	45005	31	2
MS SOAP RPC nowait	4010	7	1	29440	21	1	80922	56	3
Apache SOAP wait	4911	12	2	35521	40	4	97196	99	10
Apache SOAP nowait	4810	11	1	35318	37	3	95449	95	9
MSMQ	1968	2	1.1	19782	14	1.5	56086	40	3.6

# Processor View

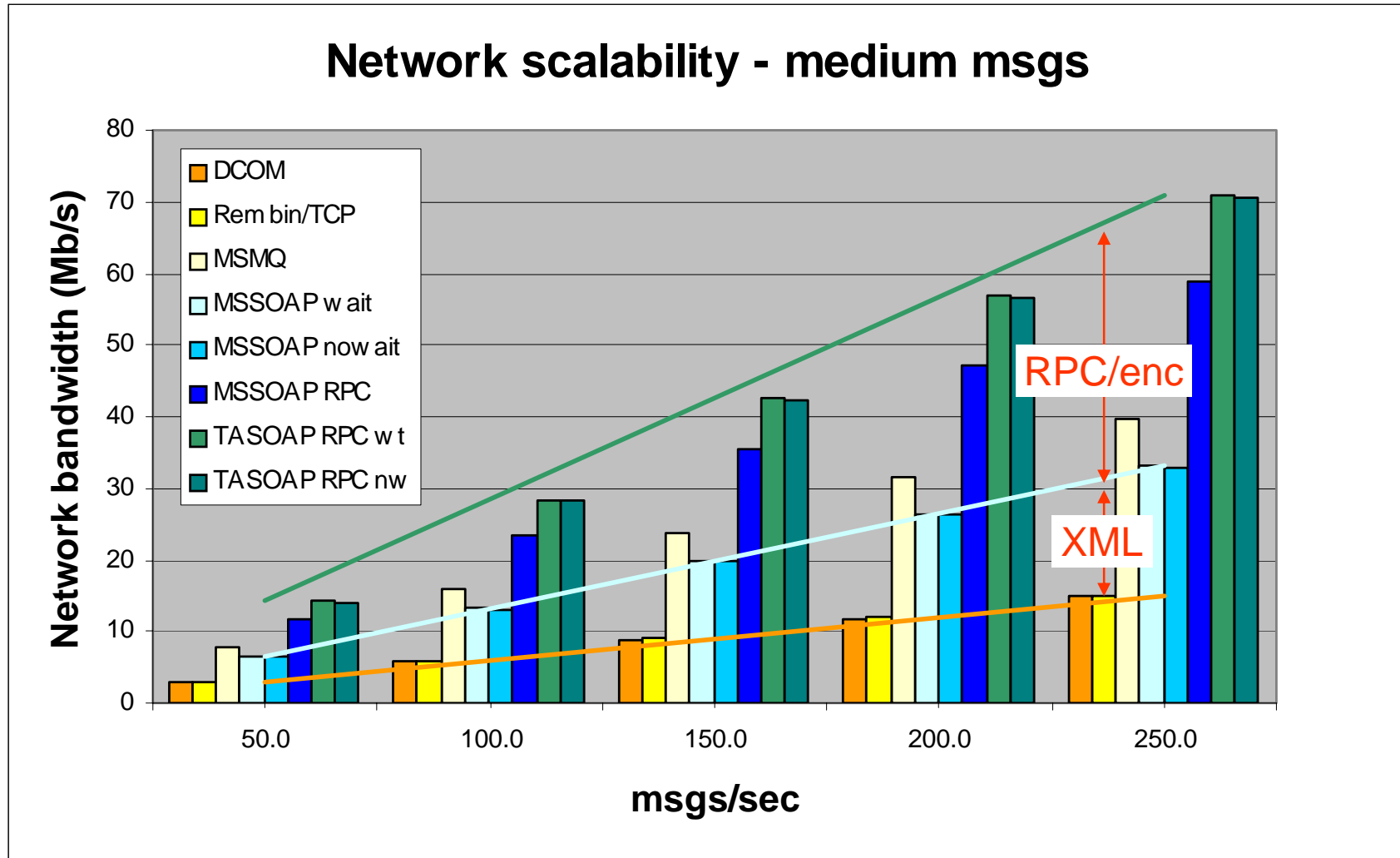
## Server Processor Utilisation



# Scaling to a Work Load



# Scaling to a Work Load



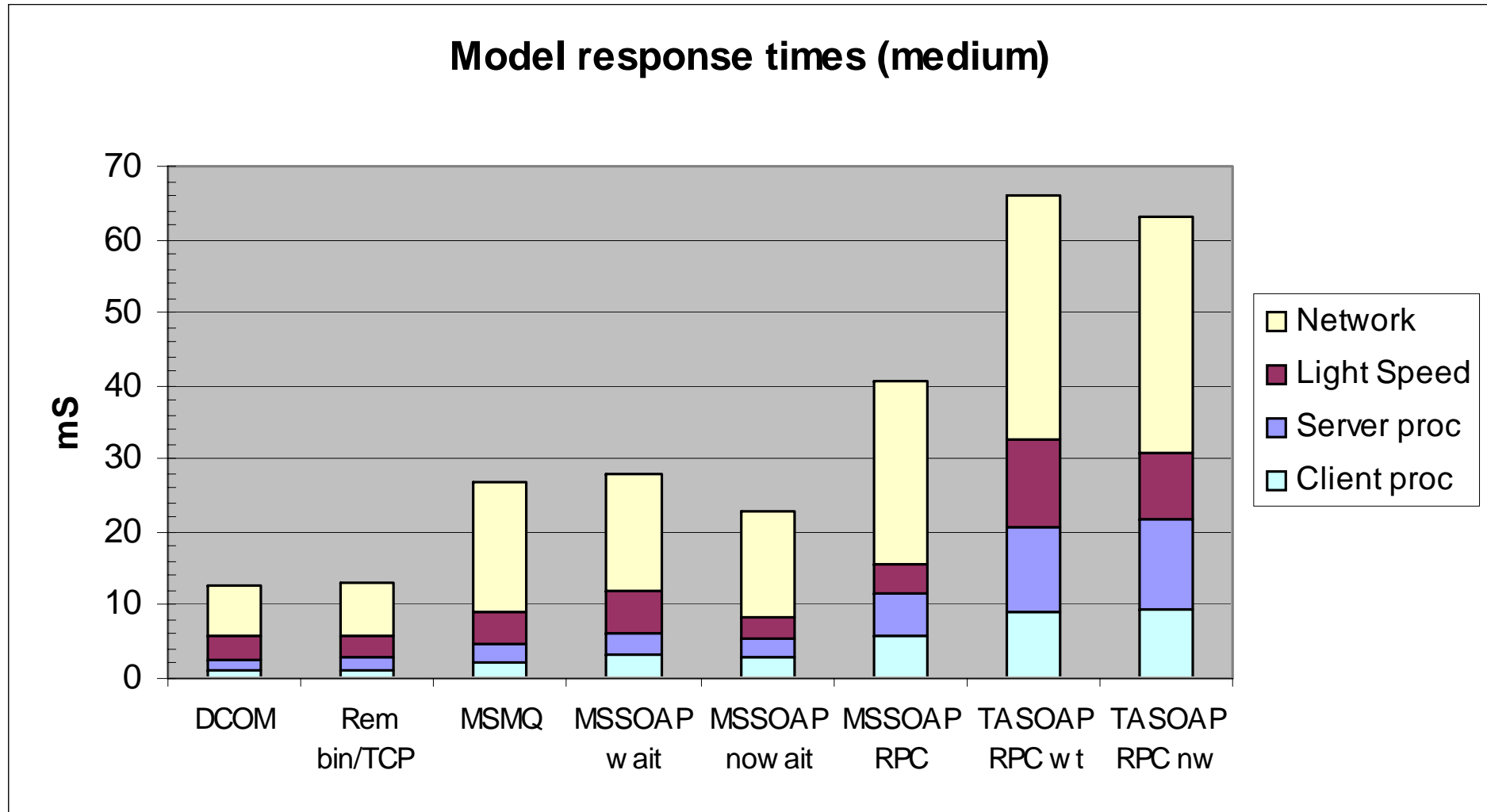
# Scalability Lessons

- XML message size may cause...
  - Latency problems on slower networks
  - Higher network loads
    - And message size increases with new features such as security and reliability
    - What happens with slower networks?
  - Encoding style has impact on performance
- Processor utilisation
  - Comparable to alternatives for best implementations
    - And others have room to improve
    - Plenty of cycles for other uses
  - Amenable to server farms?
    - SOAP stateless servers with backend transactions

# Scalability Reality Check

- Tests done on 1Gigabit, lightly loaded WAN...
- Limiting factor is 100M LAN edge
- May not be atypical in practice
  - Telco core network is high bandwidth
  - You just pay for low bandwidth at entry point
- So... modelled slower speed networks
  - Assumed linear network behaviour
  - Network transfer times start to dominate as bandwidth decreases
  - Expected behaviour given sizes of messages
  - Aggregate bandwidth required will limit scalability

# Model for 10M network



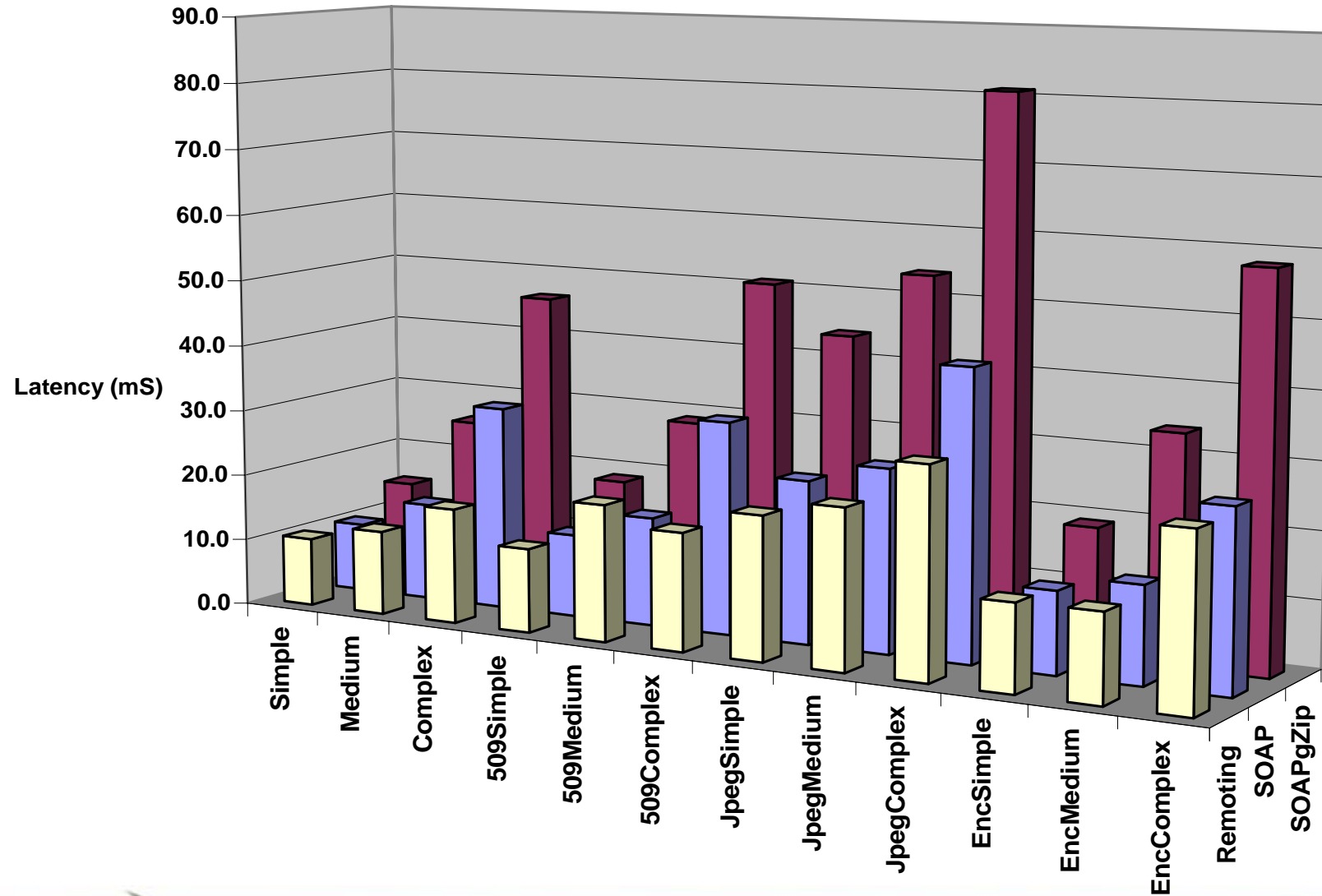
# Reducing Bandwidth Usage

- Trade bandwidth for processor time?
  - Low processor but high bandwidth requirements
  - Use compression to reduce bandwidth at expense of higher processor usage??
- Use binary rather than text encoding?
  - Sun's FastInfoset – ASN.1 proposal
  - MTOM for binary content (rather than Base64)
- Tests using...
  - GZip, .Net remoting (binary) and MTOM
  - Same msgs + JPEG, signatures, encryption
    - (Larger msgs use arrays of records)
    - Binary content

# Bandwidth Requirements

	Medium Msg			
	Base	+JPEG	+Signed	+Encrypt
SOAP	7599	27935	9149	12455
SOAP+ gZip	2278 70%	13860 50%	3316 64%	10068 19%
Remoting (binary)	3476 54%	18403 34%	5275 42%	10168 18%
MTOM		23321 17%	10012 -9%	10666 14%

# Impacts on Latency (100Mbps)



# Modelled Latency

Medium	Predicted Latency (ms)		
	SOAP	gZip	Bin
Bandwidth			
100Mbps	16	26	8
10Mbps	28	30	13
1Mbps	147	72	67
500Kbps	279	118	127

Complex	Predicted Latency (ms)		
	SOAP	gZip	Bin
Bandwidth			
100Mbps	29	46	11
10Mbps	62	55	20
1Mbps	386	109	115
500Kbps	745	169	220

# WS Performance

- Best implementations comparable
  - Reasonable processor usage (and getting better)
    - XML parsing now quite fast
  - Unavoidable XML encoding tax
    - Even using doc/lit encoding style
- Compression works well in some cases
  - Reduces latency for low-medium bandwidth
  - On regular, repetitive msgs with compressible data
- Binary works well but...
  - Interoperability costs only worthwhile for low bandwidth networks?