

# Web Services

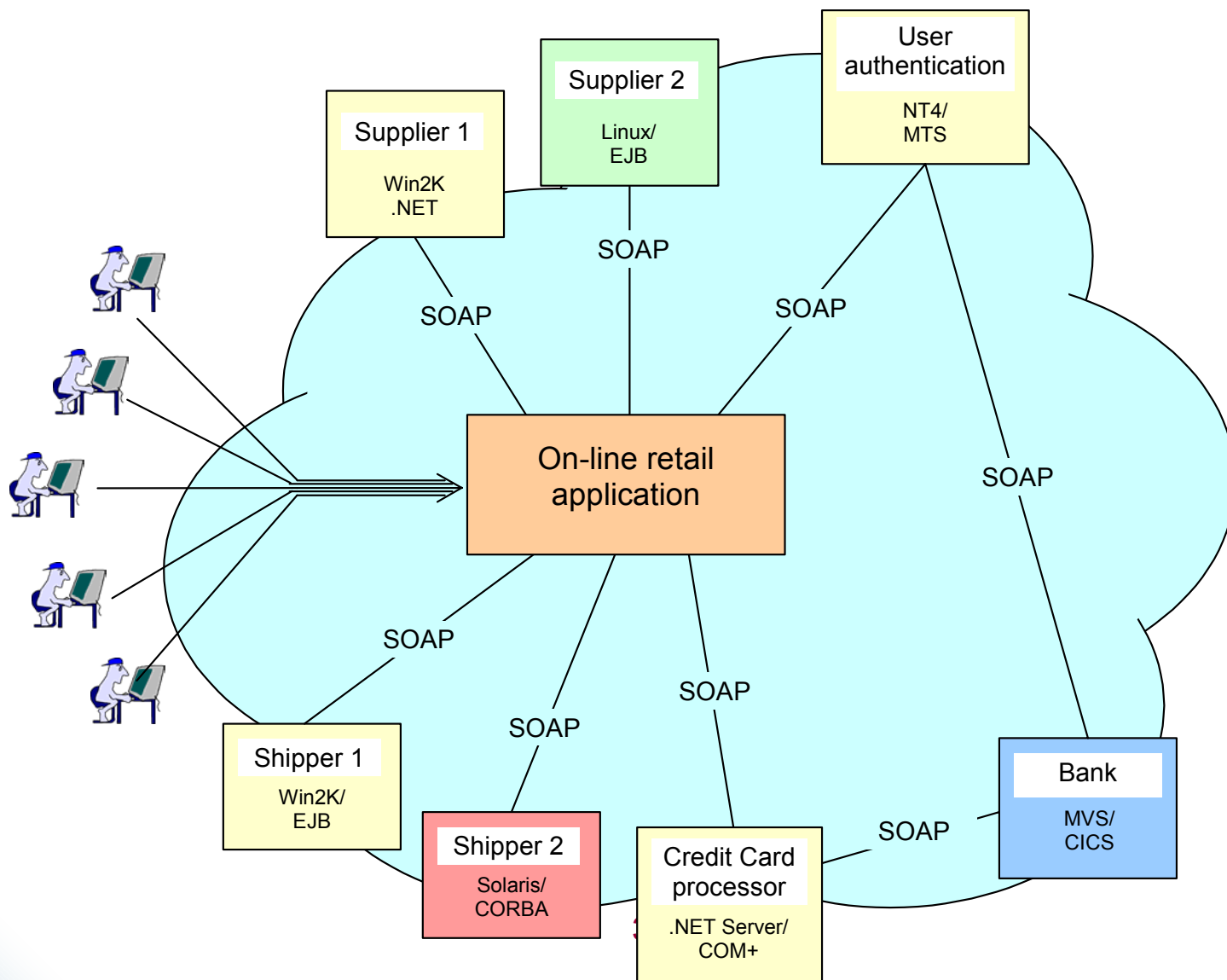
## Facts, Fables and Future

Paul Greenfield  
March 2003

# A Vision...

- A peaceful, harmonious services-based world...
  - Applications everywhere exporting services and working together
  - Seamless, easy integration between applications, across departments, across organisations
  - One common set of standards
  - A programmable Web...

# Web Services Vision



# A Programmable Web?

- What's wrong with the Web today?
  - It's just pretty & universal terminals
    - Better 3270s or VT100s
  - Really just same old dumb client model
- The services-based Web
  - Servers offer services not just pages
  - Smart clients weave new applications by combining and integrating services
  - As significant a change as the introduction of the personal computer!
  - A web made for computers, not people!



# Alternatives?

- Screen-scraping analogues
  - Programming against human interfaces
  - Divining the semantics of Web pages?
  - Fragile, clumsy applications
- Web Services is simpler solution
  - Service to return available seats, place order, update account, ...
  - No semantics, no untangling formatting

# Distributed Computing?

- Web Services are just XML-based distributed computing protocols
  - Nothing inherently new
  - Similar to EDI, CORBA, DCOM, RMI, CICS, MQ, ...
  - More verbose and slower
  - Perhaps somewhat simpler
    - This may change as standards evolve...
  - Currently compatible
    - One set of common standards
    - Despite best efforts of some vendors...

# Previous Attempts?

- *CORBA standards*
  - Object-oriented distributed computing
  - Complex OO programming model
    - Supports multiple programming languages
  - Major changes needed to existing applications and infrastructure
  - Now a niche technology
  - Killed by Sun's J2EE

# Proprietary Standards?

- Microsoft's COM+
  - Only works on MS platforms
  - Some application rewriting or wrapping
- Sun's J2EE
  - Any language you like, as long as it's Java
  - Application rewriting or adaptors
- Limited interoperability
- Not designed to work on the public Internet

# Standards?

- Mostly self-declared standards groups
  - Largely committees of industry players representing vendor interests
  - *OMG, OASIS, OSF, ...*
  - Temporary alliances against a common enemy (often Microsoft or IBM)
- De facto standards (proprietary)
  - Owned/controlled by one vendor
    - Microsoft products, IBM products, Sun's J2EE specs
    - Can be more focussed - need for product

# Standards

- Formal standards bodies
  - W3C
  - IETF
  - ECMA
- All types of standards processes active in Web Services
  - Especially now that the core proposals from MS and IBM have won through
  - Heavily commercial & competitive

# Standards...

- Software is too soft...
  - No laws of physics to constrain standards
  - Software standards can specify anything
    - And may not be easily implemented
    - And can suffer from committee syndrome
  - Software standards can expand forever
    - Already too many WS standards?
    - Already too complex?
    - Hard to argue against standardisation though - obviously a 'good thing'

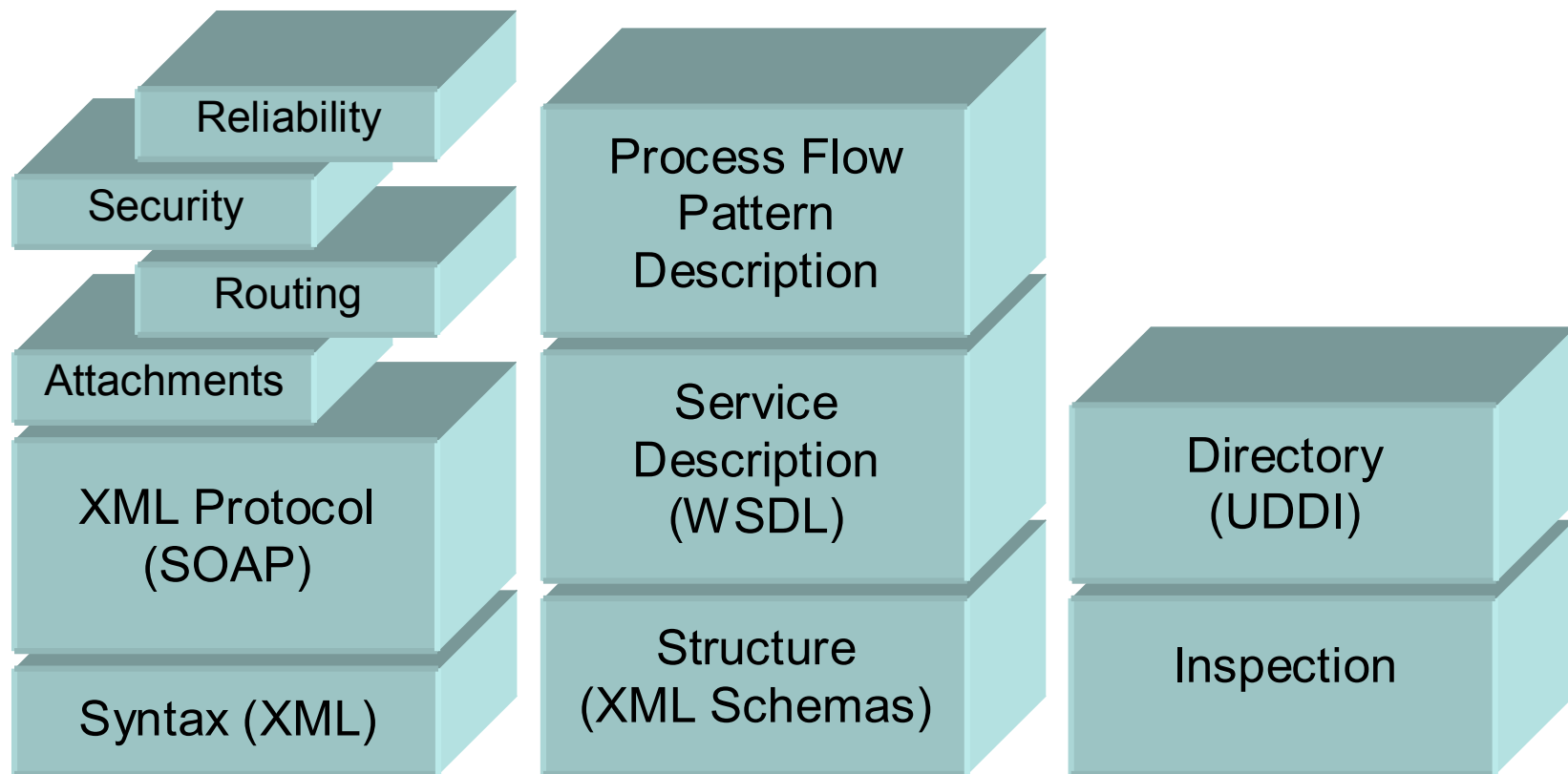
# Distributed Computing 101

- Find something
  - Object, component, service
- Find out how to call it
  - What information it needs
  - Where it is located
- Ask it to do something for you
  - Call a method, invoke a service
- And add a few useful services

# XML Web Services

- XML-based Distributed Computing
  - XML messages in the protocol
  - XML definitions of services
  - Pass across XML documents
  - Building on other XML standards
  - All 'human readable'
- Possibly running over HTTP...
  - And possibly not as well

# Web Services Stack



# Finding Things

- UDDI - a directory of services
  - Universal Description, Discovery and Integration
  - A Web Service in its own right!
- Vision of a single universal directory?
  - Reality is probably many directories
  - Hasn't taken off...
  - Included with Windows Server 2003
    - Intended for internal Web Services

# Finding About Things

- WSDL - Web Services Description Language
- XML description of a service
  - Interfaces, methods, parameters
  - Request, response
  - How to bind to the service
  - Widely accepted standard
- Generated and consumed by tools
  - Microsoft Visual Studio, IBM WebSphere

# WSDL

```
<?xml version="1.0"?>
<definitions name="StockExchange">
  <types>
    ... ..
  </types>
  <message name="getStockQuoteRequest">
    <part name="symbol" type="xsd:string"/>
  </message>
  <message name="getStockQuoteResponse">
    <part name="Result" type="xsd:float"/>
  </message>
  <portType name="StockQuoteService">
    <operation name="buyRequest" parameterOrder="symbol">
      <input message="wsdl:ns:getTradePriceRequest"/>
      <output message="wsdl:ns:getTradePriceResponse"/>
    </operation>
  </portType>
  <binding>
    ... ..
  </binding>
</definitions>
```



# Calling Things

- *SOAP - Simple Object Access Protocol*
  - Originally simple remote procedure exchange using XML
    - XML-RPC is a subset of SOAP
    - Now can pass messages as well as RPC
  - Envelope, header, body
  - Extensible protocol
    - Add new header elements to access additional semantics
  - Very widely accepted standard
    - Up to SOAP 1.2 now

# SOAP

Envelope (Mandatory) -  
Marks the start and end of a message

Header (Optional) -  
General information about message – e.g. authentication and transaction management

Body (Mandatory) -  
Data for the actual message or document being sent

# SOAP

```
POST /StockQuote HTTP/1.1
Host: www.stockquoteserver.com
Content-Type: text/xml
Content-Length: nnnn
SOAPMethodName: Stock-Namespace-URI#GetLastTradePrice
<SOAP:Envelope xmlns:SOAP="urn:schemas-xmlsoap-org:soap.v1">
  <SOAP:Body>
    <m:GetLastTradePrice xmlns:m="Stock-Namespace-URI">
      <symbol>DIS</symbol>
    </m:GetLastTradePrice>
  </SOAP:Body>
</SOAP:Envelope>
```

```
HTTP/1.1 200 OK
Content-Type: text/xml
Content-Length: nnnn
<SOAP:Envelope xmlns:SOAP="urn:schemas-xmlsoap-org:soap.v1">
  <SOAP:Body>
    <m:GetLastTradePriceResponse xmlns:m="Stock-NS-URI">
      <return>34.5</return>
    </m:GetLastTradePriceResponse>
  </SOAP:Body>
</SOAP:Envelope>
```



# Everything Else

- SOAP, WSDL (& UDDI) only meet simplest requirements
  - Still usable for many internal or non-critical applications
    - Google searches, MapPoint maps
- Other standards emerging for everything else
  - Security, transactions, routing, reliability, attachments, ...
  - Current standards battleground...

# Security!

- “SOAP is firewall friendly”
  - Normally transported over HTTP
  - Firewalls expect HTTP to be Web requests, not procedure calls
    - Fetch the brochure...
    - Not update my bank account...
  - This is probably a good thing!
    - Have to rely on strong security rather than weaker physical security

# WS-Security

- Proposal from IBM & MS
- SOAP header extensions
  - Protect integrity and confidentiality of messages
  - Attach encoded security tokens
    - X509, Kerberos
- Leverage XML standards
  - XML Signature
  - XML Encryption

# More WS-Security

- **WS-Policy**
  - Specifying requirements & capabilities
  - Policies including privacy, encoding security tokens, QoS, ...
  - First published Dec 2002
- **WS-Trust**
  - Establishing trust relationships
  - Trusted security tokens
  - First published Dec 2002

# More WS-Security

- WS-Privacy
- WS-SecureConversation
- Ws-Federation
- Ws-Authorization

And probably a few more...

# More Security Standards

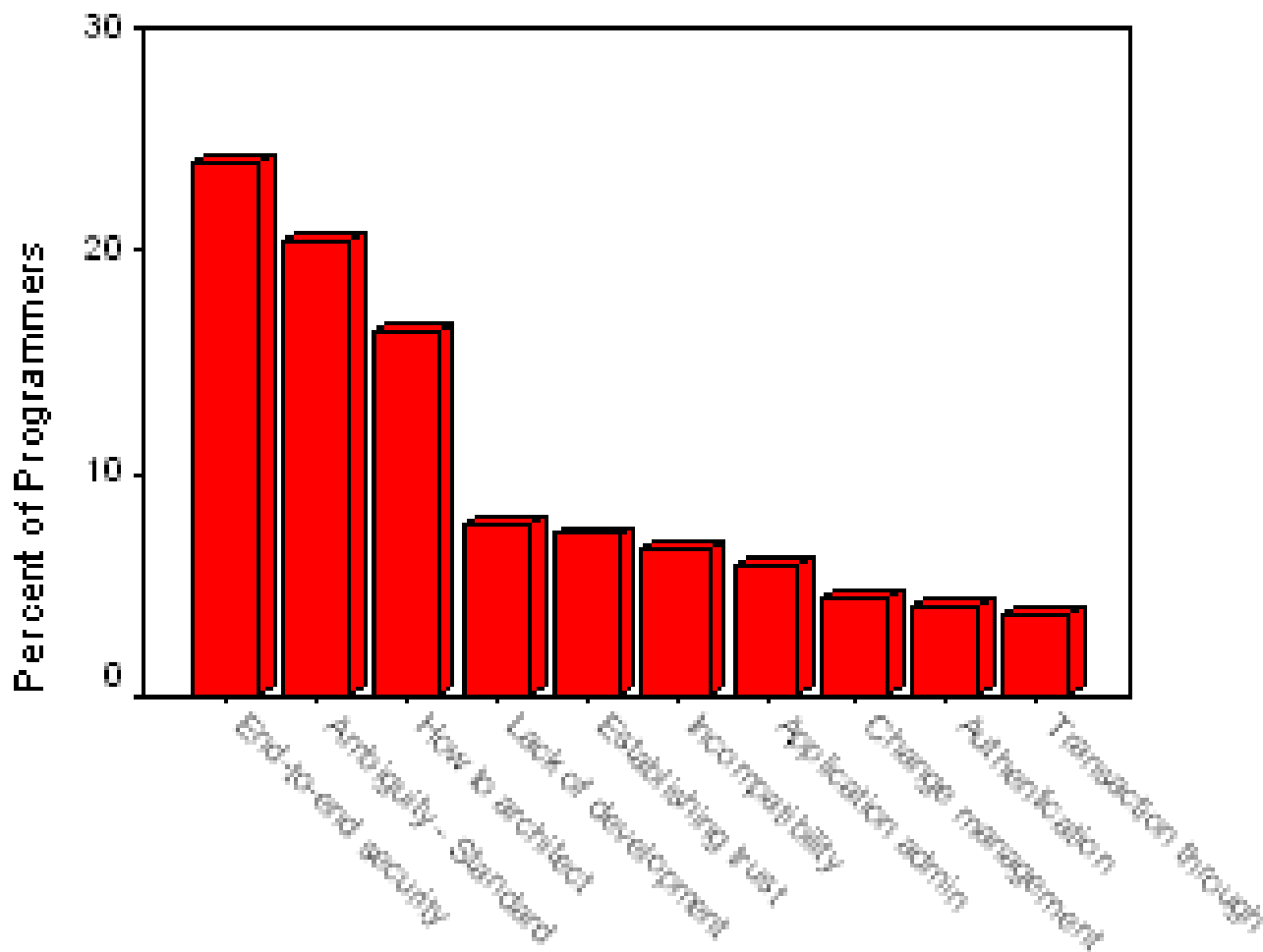
- **SAML**
  - Representing authentication and authorisation information
- **XACML**
  - Access Control
- **XrML**
  - Digital Rights

# Yet More Security

- All these standards based on existing strong security standards
  - PKI
  - Kerberos
  - Digital signatures
- And why so much interest?

# Security Concerns?

Most Significant Barrier to Web Services Implementation



# WS Transactions

- Ensuring consistency and integrity
  - WS-Coordination & WS-Transactions
    - Proposals from MS and IBM (& BEA)
    - ACID and long-running transactions
  - BTP
    - Proposal from OASIS (Oracle, Sun, BEA,...)
  - No consensus yet

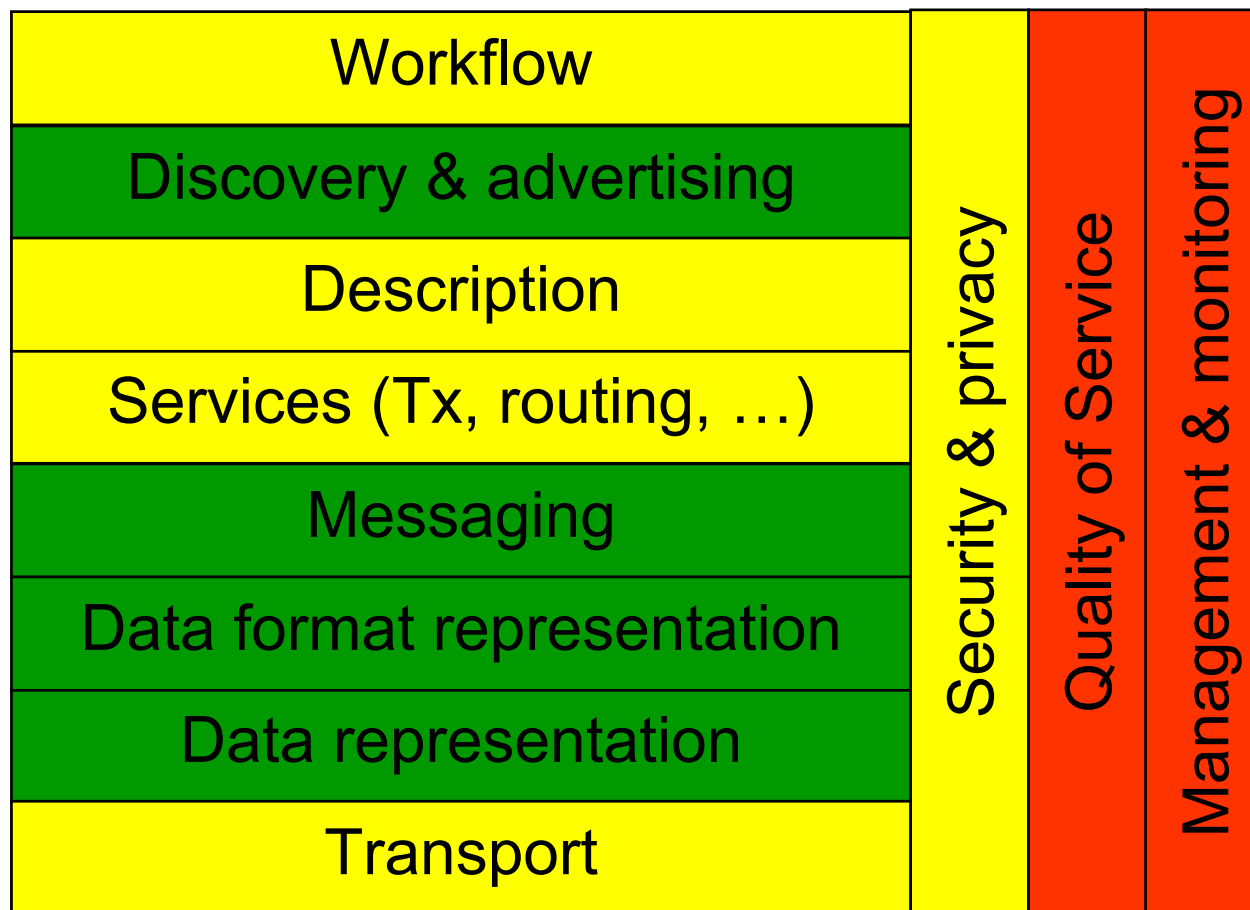
# Business Processes

- WS standards for how services interact and work together
  - Workflow languages
  - BPEL4WS (IBM & MS)
    - Merging of WSFL and XLang
  - Competing standards from Oracle, Sun, HP through OASIS

# Transports

- **Web Services independent of HTTP**
  - HTTP only offers partial solution
    - Unreliable, point-to-point
- **WS-Routing**
  - Path a message should take
- **WS-Referral**
  - Relaying/delegating service requests
- **WS-Reliability & HTTPR**
  - Guaranteed exactly-once delivery

# Current State



# Immaturity

- Core standards in place
  - SOAP, UDDI, WSDL
- Competing standards or nothing
- Standards without wide industry support
  - Very Darwinian standards process
  - IBM & MS vs Sun & Oracle
- Can connect systems
  - But not much more today

# Vendor Support

- **Microsoft**
  - Web Services and XML at core of .NET
    - .NET remoting based on SOAP
    - UDDI as local service directory
    - Transparent use of Web Services from Visual Studio (SOAP and WSDL)
    - Early release of proposed standards
- **IBM**
  - Extensive tools support in WebSphere
  - Alpha and beta for proposed standards

# Vendor Support

- Sun and Java
  - Behind the game..
    - Standard support for Web Services mid-2003 in J2EE 1.4
    - Proprietary support by app server vendors
  - Aggressively trying to get back in..
    - Trying for unclaimed territory...
- And support for Perl and ...

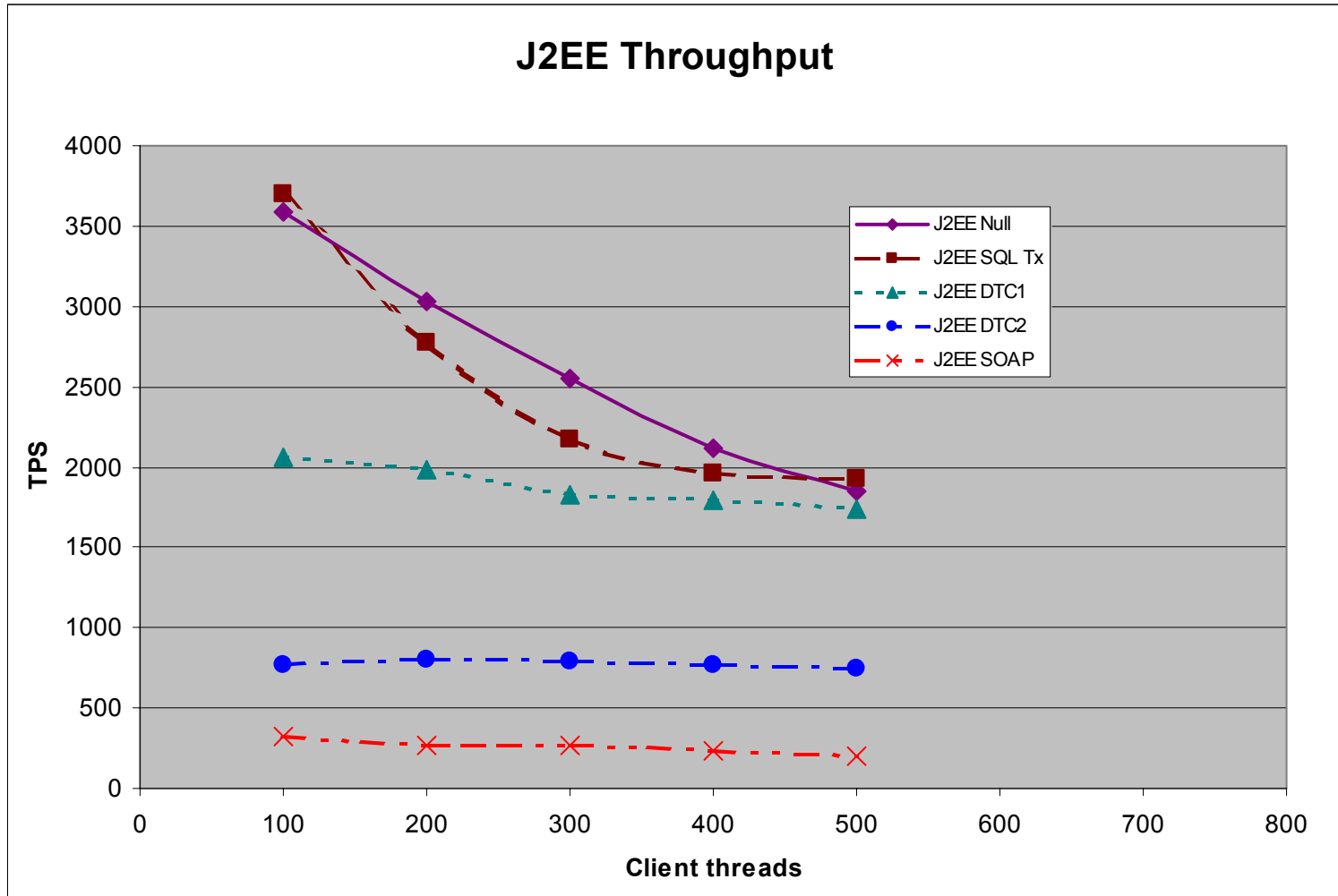
# WS and the Grid

- Next generation Grid standards based on Web Services
  - OGSA using SOAP, WSDL, ...
- Recasting existing OGSA APIs and protocols to use Web Services

# Performance

- Bound to be excellent with all that XML and HTTP
  - Machines and networks are getting faster...
  - Designed for interoperability rather than raw speed
  - And will get better over time
    - And has a lot of improving to do...

# Performance



# Back to the Vision

- Vision is about rich, seamless interoperability
  - My client application talking to your server application, regardless of platforms, languages, ...
- Not a reality today but close
  - Pretty good for simple tasks
  - Other standards too immature
  - Critical role for WS-Interop

# WS-Interop

- Industry body formed to ensure interoperability vision is achieved
  - Standard profiles of current standards
    - Simplifying, ensuring common subsets
  - Compatibility testing
  - All major vendors participating
    - Eventually even Sun... after some politics
  - But Ws is getting to be a big set of standards...

# Conclusion

- Is Web Services fact or fable
  - It's a good story - and the only one
  - Major industry investment should make the technology work
  - Huge returns for customers in making application integration easier
    - And everyone needs application integration
  - Potential for a new Web and new kinds of applications