

# **Programmable Web Services and SOAP Architectural Overview**

***Kent Sharkey***  
***Technical Evangelist***  
***ksharkey@microsoft.com***

# Agenda

- ❖ **The evolution to Web Services**
- ❖ **SOAP**
- ❖ **Future of Web Services**

# Web Services are ...

- ❖ **Components over the web**
- ❖ **“Best of breed” tools**

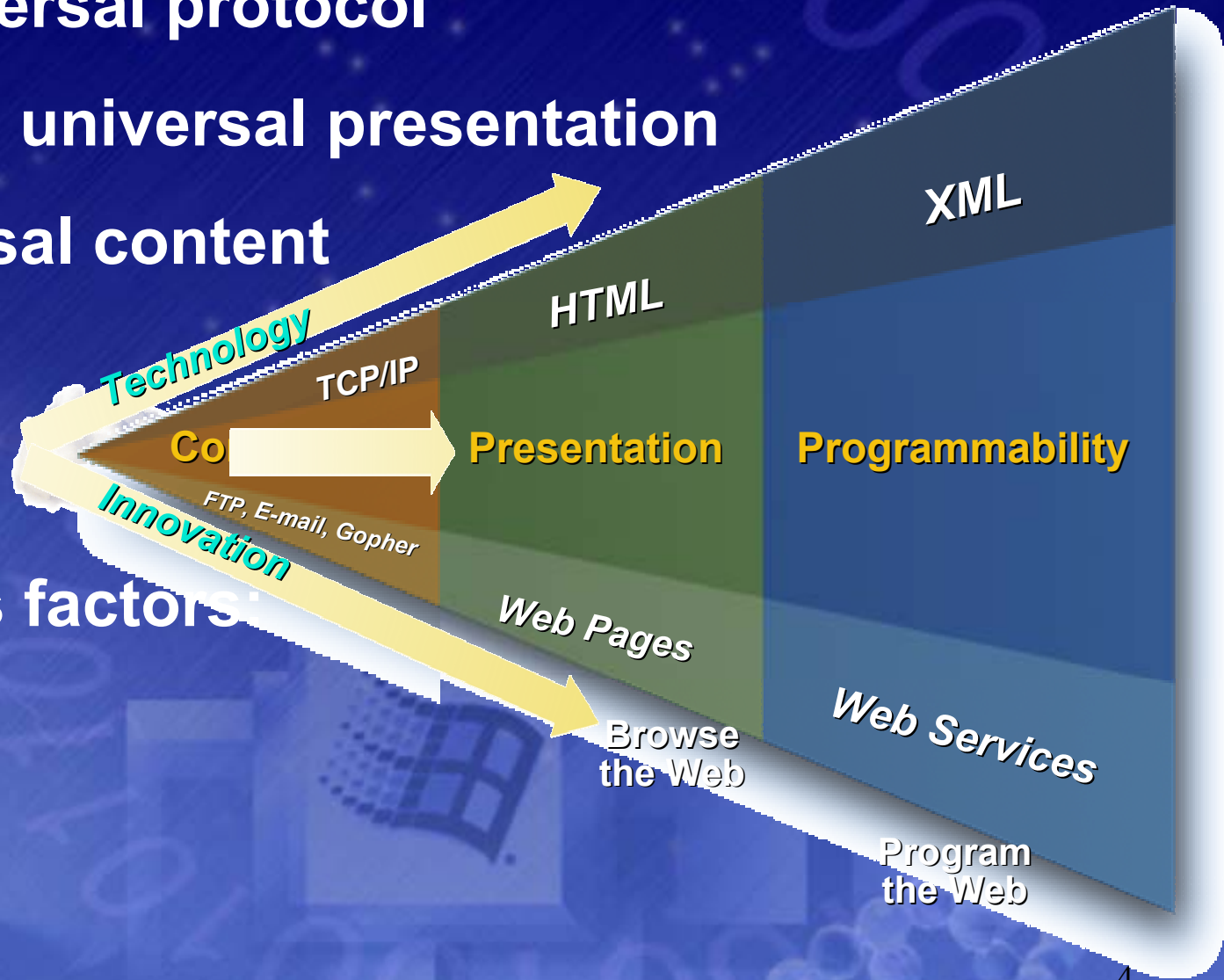


# The path to Web Services

TCP/IP: universal protocol

HTTP/HTML: universal presentation

XML: universal content



Key success factors:

- Ubiquity
- Simplicity
- Standards

# Why do we need the Programmable Web?

Can't I just do all this with Java?

Doesn't the Web do everything you need?

Doesn't SOAP compromise security?

Aren't DCOM and CORBA IIOP enough?

# ASP/JSP/Servlets/CGI

## ❖ Pros

- Simple solution
- Works well with firewalls
- Somewhat portable

## ❖ Cons

- Output is typically HTML (requires parsing to retrieve data)



# DCOM/Corba (IIOP)/RMI

## ❖ Pros

- Most direct way of accessing component functionality
- Best performance

## ❖ Cons

- Complexity
- Not good for talking to “the other ones”
- Don’t pass through firewalls well

# SOAP

## ❖ Pros

- Works well with firewalls
- Great for communicating between different platforms
- Uses XML, therefore allows rich, extensible messages

## ❖ Cons

- Limited tools (for now...)



# Uses for Programmable Web Services

- ❖ **Inside the firewall**
  - **Enterprise Application Integration (EAI)**
  - **Intranet**
- ❖ **Between partners**
  - **Business to Business (B2B)**
  - **Application to Application (A2A)**
- ❖ **On the Internet**

# Benefits

- ❖ **Standard communication format**
- ❖ **Use existing infrastructure**
- ❖ **Connecting diverse systems**
- ❖ **New revenue streams**

# Agenda

- ❖ The evolution to Web Services
- ❖ **SOAP**
- ❖ Future of Web Services



# Standards and SOAP

- ❖ **Builds on**
  - W3C XML standards
  - IETF HTTP standard
- ❖ **SOAP Spec v1.0 to IETF**
  - <http://www.ietf.org/internet-drafts/draft-box-http-soap-01.txt>
- ❖ **SOAP Spec v1.1 to W3C**
  - <http://msdn.microsoft.com/workshop/xml/general/soapspec.asp>

# Logical structure of SOAP

## Applications

Routing protocols (BizTalk, ebXML)

XML RPC protocols (SOAP, XML-RPC)

W3C Standards (XML, Namespaces)

Internet Standards (TCP, UDP, IP)

# SOAP message types

- ❖ Request
- ❖ Response
- ❖ Fault



# SOAP message structure: Request

```
POST /StockQuote HTTP/1.1
Host: www.stockquoteserver.com
Content-Type: text/xml
Content-Length: nnnn
SOAPAction: Some-Namespace-URI#GetLastTradePrice
```

```
<SOAP-ENV:Envelope
```

```
  xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
```

```
  SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
```

```
>
```

```
  <SOAP-ENV:Body>
```

```
    <m:GetLastTradePrice xmlns:m="Some-URI">
```

```
      <symbol>DIS</symbol>
```

```
    </m:GetLastTradePrice>
```

```
  </SOAP-ENV:Body>
```

```
</SOAP-ENV:Envelope>
```

# SOAP message structure: Response

```
HTTP/1.1 200 OK
Content-Type: text/xml;
charset="utf-8"
Content-Length: nnnn
```

## <SOAP-ENV:Envelope

```
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/"
SOAP-ENV:encodingStyle="http://schemas.xmlsoap.org/soap/encoding/"
```

```
>
```

## <SOAP-ENV:Body>

```
<m:GetLastTradePriceResponse xmlns:m="Some-URI">
```

```
<Price>34.5</Price>
```

```
</m:GetLastTradePriceResponse>
```

```
</SOAP-ENV:Body>
```

```
</SOAP-ENV:Envelope>
```

# SOAP/HTTP and Security

- ❖ **HTTP Header Filtering**
  - **M-POST verb**
- ❖ **HTTP**
  - **Basic**
  - **SSL**
- ❖ **Header filtering (on SOAPAction)**
- ❖ **Application Level Restriction**
  - **Only allow access to “Web Service” registered components**



# Industry support

- ❖ **DevelopMentor**
- ❖ **IBM/Lotus**
- ❖ **Intel**
- ❖ **IONA Technologies**
- ❖ **Jetform**
- ❖ **ObjectSpace**
- ❖ **Rockwell Software**
- ❖ **Rogue Wave Software**
- ❖ **UserLand Software**
- ❖ **And many more...**

# Agenda

- ❖ **The evolution to Web Services**
- ❖ **SOAP**
- ❖ **Future of Web Services**

# SOAP Toolkit

- ❖ **Designed to make using SOAP over HTTP easier**
- ❖ **A reference implementation of a proxy and listeners**
- ❖ **Due out RSN**



# Agenda

- ❖ **The evolution to Web Services**
- ❖ **SOAP**
- ❖ **Future of Web Services**



# Towards the future

- ❖ **Increasing industry support**
  - Iona, IBM/Lotus now, more soon
- ❖ **Better tool support**
  - Visual Studio
  - Object Brokers (ORBs)
- ❖ **Implementations on other protocols**
  - SMTP, FTP, MSMQ, MQ Series...

# Summary

- ❖ **SOAP = XML + HTTP (eventually other protocols)**
- ❖ **Cross platform**
- ❖ **Language independent**
- ❖ **Works with firewalls**

# Call to action

- ❖ **Read the SOAP spec:**  
<http://msdn.microsoft.com/workshop/xml/general/soapspec.asp>
- ❖ **Identify potential WebServices**
- ❖ **Implement SOAP listeners for service**
- ❖ **Create SOAP clients for other services**



# Questions?

