



Trends in Global Application Development

Alberto Pace - January 2002

alberto.pace@cern.ch

<http://cern.ch/alberto.pace>





The web as an application platform

- ◆ The web as the backbone of distributed applications
- ◆ The vertical integration of E-commerce has shown its limits: A single company cannot do it all
 - ◆ Production, Marketing, catalogue, web portal, Sales, Distribution, After sales support, etc
- ◆ Going back to the real segmented world: Companies do only what they are good at. This requires improvements and **automation** in the business-to-business (B2B) technology.



What we have already

- ◆ The “old” web was targeted to humans, not computers.
- ◆ We have a world-wide file system, read/write (webdav)
- ◆ XML seems to be “the” way to represents information
- ◆ Automation of b2b relations is done only with expensive, custom, non-reusable, developments
 - ◆ **SOAP addresses these issues**
 - ◆ **Goes well beyond the b2b case, several technologies can be leveraged for CERN needs**



Ideas behind SOAP

- ◆ The web as the backbone of distributed applications
- ◆ SOAP is a set of well defined XML schemas used for remote procedure calls, and Object Oriented programming
 - ◆ **Allows Method invocation, property modifications, event triggering, callback notifications, etc.**
- ◆ By using HTTP/XML, it is possible to reach any OS (not only Windows and Unix) and any device (not only computers)



SOAP

- ◆ **Simple Object Access Protocol**
 - ◆ **Fully based on HTTP (or Secure HTTP)**
 - ◆ **Remote Procedure call (RPC) model inspired by CORBA Internet Inter Object Protocol (IIOP)**
 - ◆ **Supported by W3C and by all major names in the computing industry (Compaq, HP, IBM, Microsoft, SAP Sun, Oracle, and other companies)**
 - ◆ <http://www.w3.org/2000/xp/>



SOAP Example

(simplified)

```
POST /path/foo.pl HTTP/1.1
Content-Type: text/xml
Content-Length: nnnn
```

```
<soap:Envelope>
  <soap:Body>
    <Add>
      <arg1>24</arg1>
      <arg2>53.2</arg2>
    </Add>
  </soap:Body>
</soap:Envelope>
```

```
200 OK
Content-Type: text/xml
Content-Length: nnnn
```

```
<soap:Envelope>
  <soap:Body>
    <Response>
      <sum>77.2</sum>
    </Response>
  </soap:Body>
</soap:Envelope>
```



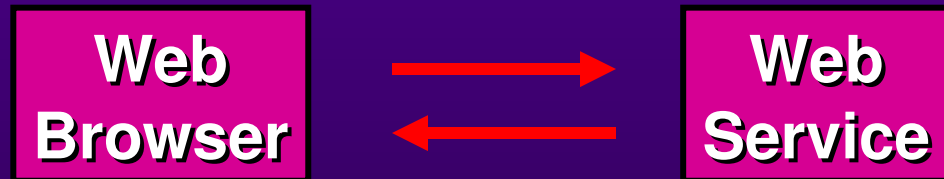
SOAP terminology

- ◆ **WebMethod**
 - ◆ **The URL used to invoke a function implemented in a WebService**
 - ◆ **A WebService can expose an unlimited number of WebMethods**
- ◆ **WebService (or SOAP interface)**
 - ◆ **An URL (accessible using HTTP) that exposes a set of callable methods that conform to the SOAP XML schema**
 - ◆ **A device on the network (internet), can expose an unlimited number of WebServices**
- ◆ **WSDL (Web Service Definition Language) Contract**
 - ◆ **An XML document that describes the WebService for programmatical use. Every Web Service should expose a WSDL contract that allow the automated interface discovery by SOAP clients.**



Demo of SOAP

- ◆ Normally you never use telnet to connect to a Web server, unless you really want to see how http is implemented
- ◆ Similarly, you never use a web browser to connect to a webservice, unless you really want to see the XML that implements SOAP



Standard SOAP
(XML / HTTP)



Slides for the DEMO no 1

The screenshot displays two overlapping windows from a Microsoft Internet Explorer browser. The background window, titled "Service1 Web Service - Microsoft Internet Explorer provided by CERN", shows the "Service1" web service interface. It includes a "SetMagnetCurrent" section with a "Test" button and a form for "NewCurrent:". Below this is a "SOAP" section with a sample request and response. The foreground window, titled "http://pcap5/2002-01-23-1/Service1.asmx/SetMagnetCurrent?NewCurrent=33 - Micr...", shows the XML response: `<?xml version="1.0" encoding="utf-8" ?>
<string xmlns="http://tempuri.org/">Current has been set to 33</string>`. A "Performance" window is also open, showing a line graph of a counter over time. The graph shows a step change from approximately 33 to 55. Below the graph, the following statistics are displayed:

Last	33	Average	42	Minimum	33
		Maximum	55	Duration	1:40

At the bottom of the Performance window, a table lists the counter details:

Color	Scale	Counter	Instance	Parent	Object	Computer
Blue	1.000	test1	---	---	2001-0...	\\PCAPS



Demo of SOAP

- ◆ SOAP is used only for inter-object communications
- ◆ One or more objects implement the user interface.
- ◆ The user interface is in this example, based on HTML





DEMO No 2

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click
Dim ws As New pcap5.Service1()

ws.SetMagnetCurrent(TextBox1.Text)

End Sub

The screenshot shows a Microsoft Internet Explorer browser window titled "WebForm1 - Microsoft Internet Explorer provided by CERN". The browser address bar shows the URL "http://localhost:23-2/WebForm1.aspx". The page content displays "New Current: 21" next to a text input field containing the value "21". Below the input field is a button labeled "Set Current".

Overlaid on the browser is the Windows Performance Monitor window. The left pane shows the tree view with "System Monitor" selected. The right pane shows a performance graph with a vertical red line at approximately 56. Below the graph, the following statistics are displayed:

Last	21	Average	56	Minimum	21
		Maximum	80	Duration	1:40

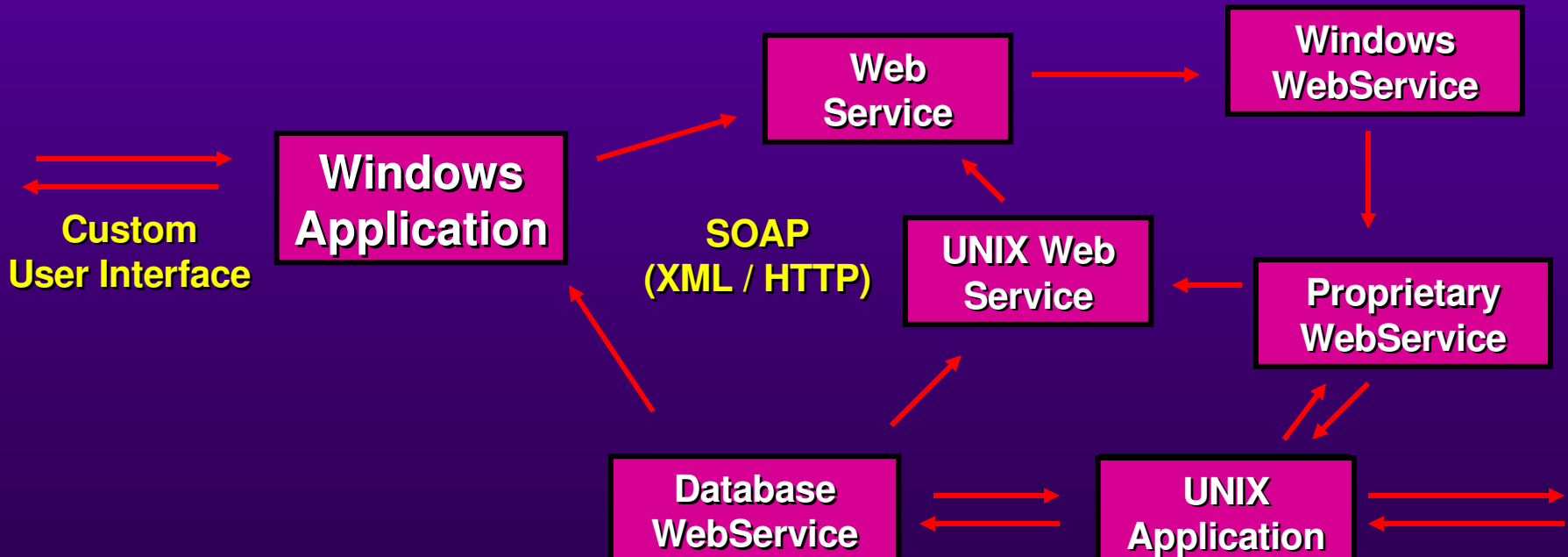
Below the statistics is a table with the following columns: Color, Scale, Counter, Instance, Parent, Object, Computer.

Color	Scale	Counter	Instance	Parent	Object	Computer
Blue	1.000	test1	---	---	2001-0...	\\PCAPS



User interface

- ◆ SOAP represents ONLY how objects communicates between them
- ◆ There is STRICTLY no implications on the user interface. Any SOAP agent can have a user interface and it can be a native Windows or Unix application





Demo of SOAP

- ◆ The usage of web services is completely hidden to the user that cannot be aware that he is using the web





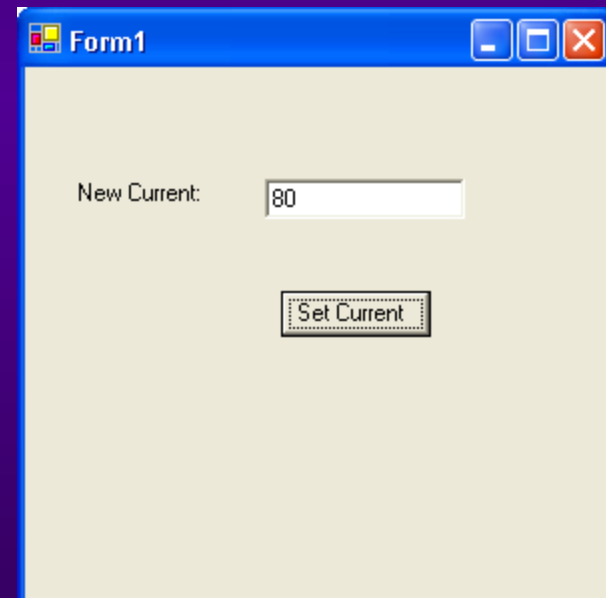
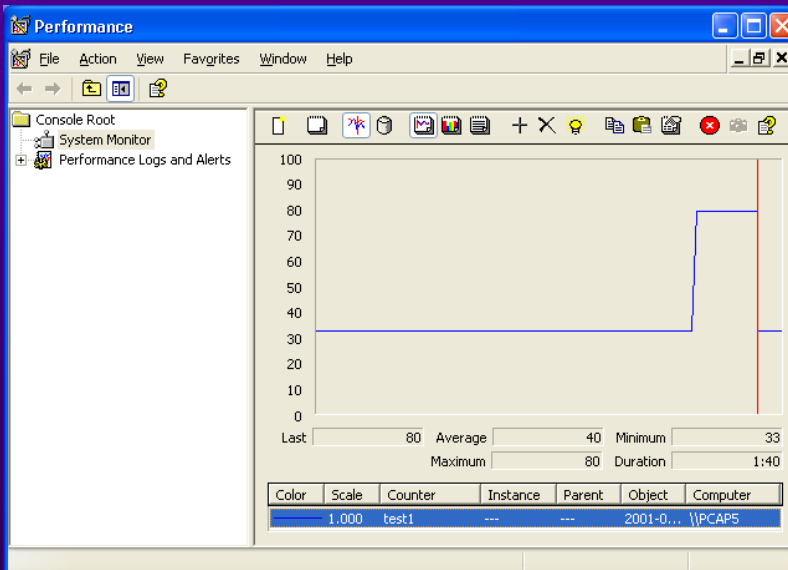
Demo No 3

Private Sub Button1_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles Button1.Click

```
Dim ws As New pcap5.Service1()
```

```
ws.SetMagnetCurrent( textbox1.Text)
```

End Sub





Security of SOAP

- ◆ When running on the internet every object of your distributed application is accessible by anyone
- ◆ Security can be implemented at the **web service** level, but typically is done at the **web server** (Apache, IIS) level (see next slides).
- ◆ In a distributed application using SOAP, security must be thought from the beginning.
 - ◆ **Don't implement everything without security and then expect the "network administrator" to fix the security for you. Especially if you are planning to use SSL.**



Security and authentication

- ◆ On the Web, security has been traditionally been implemented at the “(web) server level”
 - ◆ **Local accounts, local password, local authentication, local authorization.**
 - ◆ **A different account for every service you access on the internet**
- ◆ Kerberos has been one technology that did not reach the web immediately
 - ◆ **Typically limited by the necessity of a web server to be able to forward authentication credentials to a database server**
- ◆ Certificates have been effective to identify Computers (servers) but not people
- ◆ ... but things are changing ...



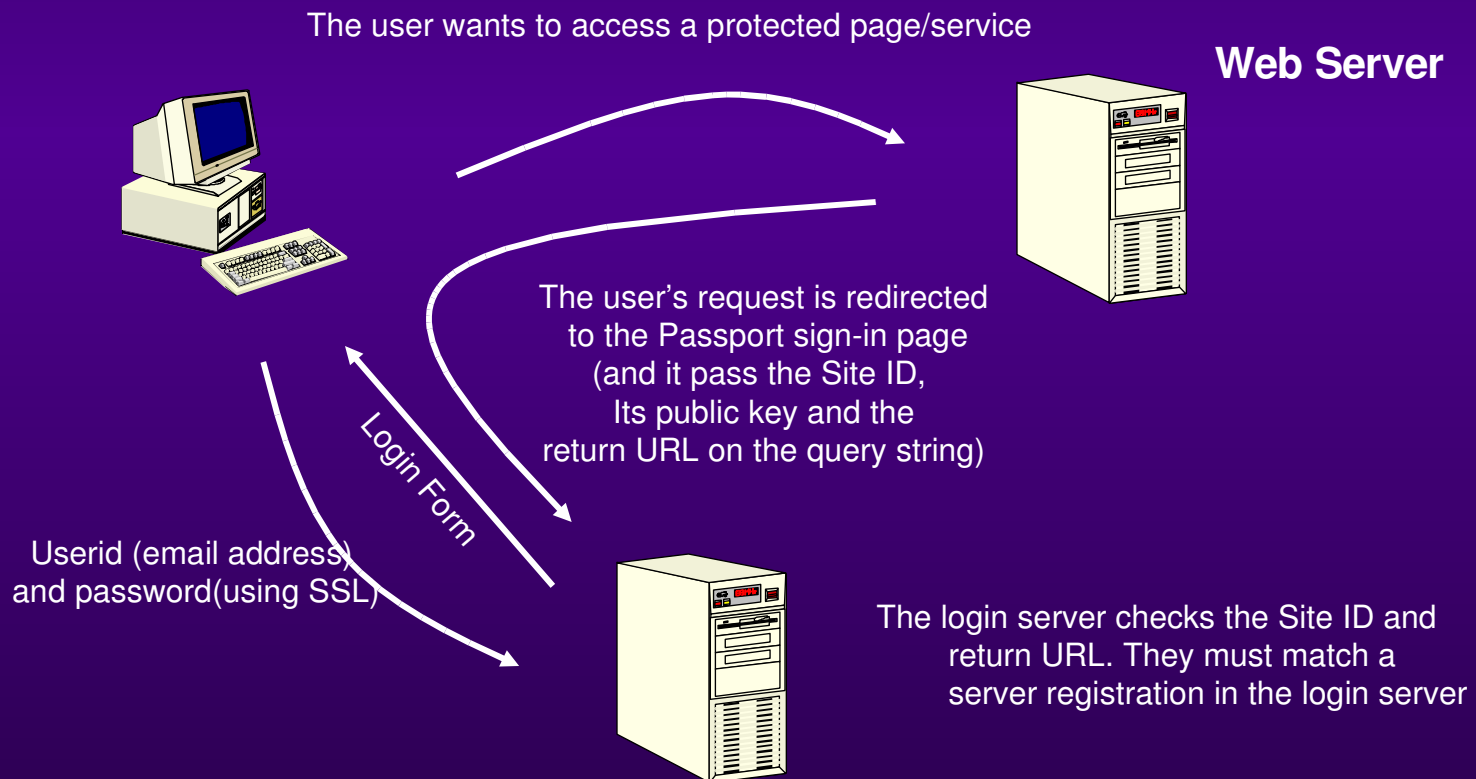
Passport.com and Kerberos

- ◆ “Endorse” Kerberos and “Extend”
- ◆ Solves 2 major problems
 - ◆ 1) forwarding credentials between “trusted servers” (ticket forwarding)
 - ◆ 2) share user-defined “profiles” among independent servers (hailstorm, myservices)
- ◆ But there is one additional issue
 - ◆ Passport.com is owned by Microsoft



World-Wide kerberos-based services

◆ Example: passport.com





World-Wide kerberos-based services

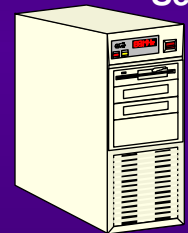
◆ Example: passport.com



The ticket can be forwarded to another server that shares the Site-ID for which the ticket is valid
Impersonation is possible !



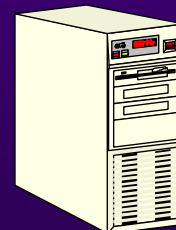
Database Server



Web Server

The user is redirected to the participating site, and the encrypted ticket and profile data are passed to the site as query string parameters

The server can verify at any moment
The validity of the ticket



If the user's sign-in name and password match in the login database, the user is authenticated and the Passport Unique Identifier (PUID) and Passport Profile are extracted from the Passport database

- Three encrypted cookies are set on the client
- Ticket cookie (includes the PUID and a time stamp)
 - Profile cookie (stores the profile information)
 - Visited Sites cookie (stores a list of sites where the user has signed in)



Demo

- ◆ Site with authentication using passport.com
 - ◆ <http://www.argosytech.com/>
 - ◆ <http://www.doristheflorist.com/>
 - ◆ <http://www.ebay.com/>
 - ◆ <http://www.edine.com/>
 - ◆ <http://www.expedia.com/>
 - ◆ <http://www.fye.com/>
 - ◆ <http://lastminute.com/>

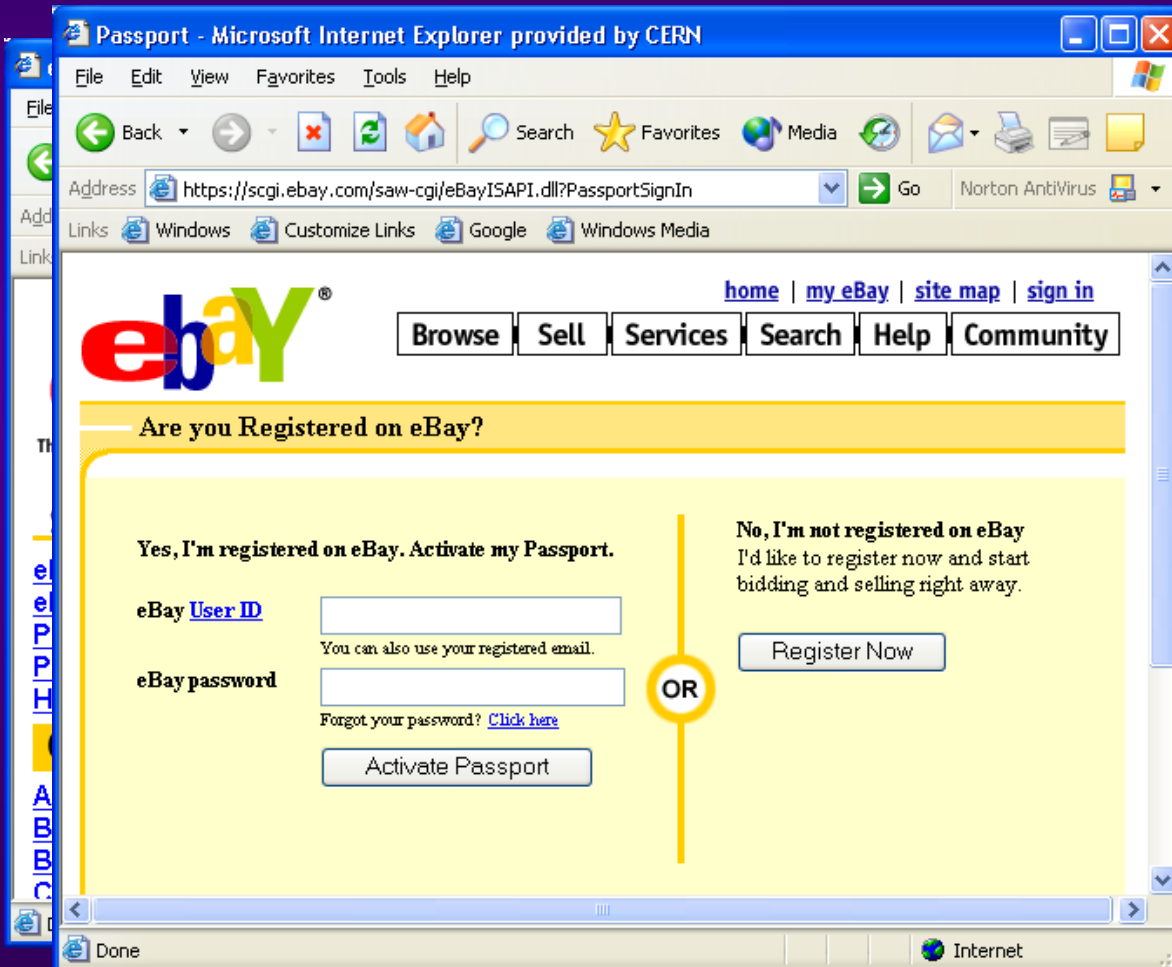


Slides for DEMO No 4

The image shows a screenshot of the Expedia.com website. The browser window title is "Expedia Travel -- discount airfare, flights, hotels, cars, vacation packages, cruises, maps". The address bar shows "http://www.expedia.com/daily/home/default.asp?rfr=-1065". The website header includes the Expedia.com logo and a navigation menu with options: home, flights, hotels, cars, vacation packages, cruises, deals, guides, maps, business. A promotional banner at the top right says "Save 25% on vacation rentals in Hawaii and Orlando." Below the navigation menu, there are links for Site Map, My Trips, My Profile, and Customer Support. A red oval highlights the user's name "Welcome, alberto.pace." and a "Sign Out" link. Below this is a "BUSINESS SEARCH" section with radio buttons for Flights, Hotels, Cars, Packages, and Cruises. The "Flights" option is selected. The search form includes fields for "Departing from:", "Depart:" (with value "2/13/02" and a calendar icon), "Morning" (with a dropdown arrow), "Return:", and "Adults:" (with value "1" and a dropdown arrow). On the right side, there are sections for "TOP DESTINATIONS" (Hawaii, Las Vegas, Los Angeles, Miami, Orlando, Ski vacation) and "TODAY'S DEALS" (Hawaii hotels: From \$52, Flights: up to 60% off, Vegas air/hotel: \$152). At the bottom right, there is a section for "EVERYDAY DEALS".



Demo No 4 (missing profile)





Passport.com owned by Microsoft

- ◆ The technology is simple and works
- ◆ Anybody can setup an identical service:
 - ◆ **Passport.org is still free, anybody can take his chance ...**
 - ◆ **We have identical services already at CERN**
 - ◆ AIS login for EDH, HRT, BHT, PIE, ...
 - ◆ WinServices/WebServices authentication
 - ◆ Missing “SITE ID concept” that allows independent user authentication per site (with same login/password) – but straightforward to implement
- ◆ But there is more ...



Is the owner of passport.com an issue ?

- ◆ The Passport.com ticket only guarantees you that the “entity” who claims to be “myname@mycompany.com” has access to the “myname@mycompany.com” mailbox.
- ◆ In other words this means that the “real authentication” of the user is DELEGATED to the mail administrator of the mycompany.com site. If you hack the mailbox, you hack the passport.
- ◆ Basically the passport initiative creates a worldwide web account repository based on email addresses where authentication is delegated to the existing mail servers and authorization is done by the web servers (based on email addresses)
- ◆ No need to have user certificates (nightmare to manage)



Role of the WSDL interface

- ◆ Allows “Discovery” of your webservices by other software (or human being).
- ◆ It always contains the XML formal definitions of all callable webmethods
 - ◆ **i.e. the programmatical interface can be discovered**
- ◆ It can contain documentation in human-readable format (XML + DTD)
 - ◆ **i.e. functionalities can be discovered**
- ◆ In the demos, we have seen how Visual Studio can, given the URL of a webservice, discover all its methods and their exact semantics



The UDDI initiative

- ◆ UDDI = Universal Description Discovery and Integration – See <http://www.uddi.org>
- ◆ UBR (UDDI Business Registry), based on the WSDL contract
- ◆ It is a set of specifications to describe and discover WebServices
 - ◆ **Mainly targeted at the b2b communities**
- ◆ Applications like Office and internet Explorer will be integrated within UDDI. Services can be plugged in your desktop effortless
- ◆ Great opportunities for services distributions, including application distributions, File and Web Storage, authentication, authorization, payment, ...
 - ◆ **Basically almost all what we are doing in-house today**



Demo

◆ A company makes Routers and Network devices

◆ <http://www.linksys.com>

A screenshot of a Microsoft Internet Explorer browser window displaying the Linksys website. The browser's address bar shows the URL <http://www.computers4sure.com/store/linksys.asp?affid=5778>. The website header includes the Linksys logo and the tagline "Making connectivity easier". Below the header, there is a navigation menu with categories like "PDA's", "Memory", "Networking", "Modems", "Storage", "Accessories", "Supplies", "Projectors", "Software", "Computers", "Notebooks", "Servers", "Printers", "Monitors", "Peripherals", "Cameras", "Scanners", "Office Products", and "Clearance". A search bar is visible with the text "Enter a keyword:". The main content area features a diagram of network devices connected by cables, with labels for "Adapters/NIC Cards", "Cables & Accessories", and "KVM Switches". A "SHOPPING CART" section indicates "CART IS EMPTY". The browser's taskbar shows several open windows, including "Links", "Amazon.com: Electro", and "buy...".



Creating SOAP applications

- ◆ You don't need to learn SOAP
- ◆ SOAP can be buried in your ORB product
 - ◆ A "CORBA version" using SOAP instead of IIOP
 - ◆ Your existing object can use soap without modification
- ◆ SOAP can be buried in your Web Server
 - ◆ Apache, ASP/ISAPI, JSP/Servlets
 - ◆ The server maps SOAP calls into COM+, Corba, RMI, ...
- ◆ SOAP can roll from your own components
 - ◆ Written in any language (Java, Perl, Basic, C/C++/C#...)
- ◆ Can be automatically generated from your development environment
 - ◆ Visual Studio .NET has built-in SOAP support. This includes Visual Basic, C++, C#.



SOAP implementations (webservices)

- ◆ Too many for a single page
- ◆ Links to some “partial lists”
 - ◆ <http://www.software.org/directory/4/implementations>
 - ◆ <http://www.soaplite.com/#TOOLKITS>
- ◆ If you search on Google for “**SOAP+Linux+implementations**” you get back 4300 pages
- ◆ Linux Magazine claims
 - ◆ “
there are already more than 60 implementations of SOAP available for jus
”



Conclusion

- ◆ Put the WEB at the “core” of your computing services
- ◆ Understand security implications
- ◆ Tools exists to build distributed, global services and applications that can be targeted to any platform (virtually anything that can acces the internet, including mobile phones, webTv, etc ...).
- ◆ Your applications will cross all firewalls
- ◆ It is no longer vaporware, but real implementations exist, are used and have real companies behind
 - ◆ It has never been easier to build web applications using the current generation of tools
 - ◆ We can even expect a migration of Win32 applications to the Web (as a platform)



<http://cern.ch/web>

<http://cern.ch/win>

<http://cern.ch/mail>

<http://cern.ch/it-div-is-techmeet>



CERN
Web Services



NICE
2000



NICE 2000
Server Infrastructure



NICE xp