

Santa Clara Marriott, Santa Clara, CA

SMI-S Over WS-Management: A Progress Report

Josh Cohen Jim Davis

Agenda

- ❑ WS-Management
 - ❑ The origin of the beast
 - ❑ What problem was it intended to solve?
- ❑ WS-Management Protocol
 - ❑ SOAP Overview
 - ❑ WS-Management Protocol
 - ❑ WS-CIM Translation
- ❑ WS-Management Community and beyond
 - ❑ Heterogeneous implementations
 - ❑ Other areas of Management
- ❑ Q&A

Where we came from

- ❑ 1990s Desktop Systems suffered from a lack of a rich unified instrumentation standard
 - ❑ SNMP was not pervasive on Desktops and MIB structure was limiting
- ❑ Microsoft joined industry vendors in forming the DMTF to address this problem. The output was Common Information Model and CIM Server (CIMOM)
- ❑ Windows implemented and shipped WMI starting with Windows 2000
 - ❑ Data modeled with CIM
 - ❑ Remote access via DCOM
- ❑ CIM Implementations also shipped in Unix

What Happened (for us)

- ❑ Applications did not create providers as aggressively as we hoped
 - ❑ CIM and WMI providers had a high learning curve
 - ❑ Lack of ubiquitous tools
 - ❑ Developers and Administrators had difficulties
- ❑ Management Consoles did not support monitoring of Windows based CIM systems
 - ❑ Even within Windows environments, DCOM presented problems for firewalls
- ❑ Difficulty interoperating due to our wonderful DCOM protocols

What about CIM-XML?

- ❑ An HTTP+XML proposal (CIM-XML) was developed
- ❑ Shortly after inception, XML Web Services gained momentum
 - ❑ SOAP and Web Services defined a superset of the CIM-XML functionality but did so in a way that was not specific to CIM, Management and was broadly re-usable across many communities.
- ❑ CIM-XML was domain specific
 - ❑ Its formats and operations were unique to management
 - ❑ CIM-XML libraries did not get the reach compared to WS*beyond management
 - ❑ Along with CIM, a high learning curve was presented to would-be developers



A problem has been detected and windows has been shut down to prevent damage to your computer.

NO_MORE_IRP_STACK_LOCATIONS

If this is the first time you've seen this Stop error screen, restart your computer. If this screen appears again, follow these steps:

Check to make sure that any new hardware or software is properly installed. If this is a new installation, ask your hardware or software manufacturer for any windows updates you might need.

If problems continue, disable or remove any newly installed hardware or software. Disable BIOS memory options such as caching or shadowing. If you need to use Safe Mode to remove or disable components, restart your computer, press F8 to select Advanced Startup options, and then select Safe Mode.

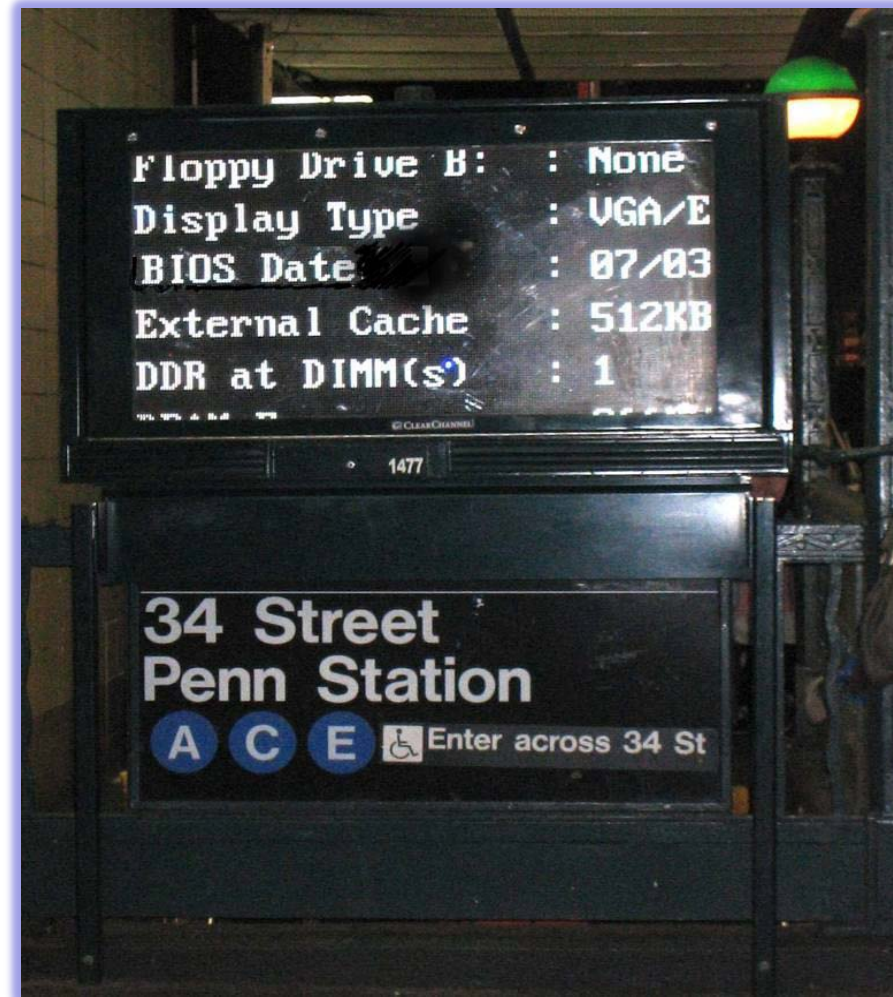
Technical information:

*** STOP: 0x00000035 (0x00000000,0x8C200F31,0x00000008,0xC0000000)

*** win32k.sys - Address 8C200F31 base at 8C200000, DateStamp 36B0245B

The Catalyst

- ❑ What if?
 - ❑ No Operating System
 - ❑ Dead Operating System
 - ❑ Unbootable System
- ❑ Enter SMASH
 - ❑ DMTF charters work to address these problems with
 - ❑ SMWVG: Server Management
 - ❑ DMWVG: Desktop and Mobile



- ❑ **Goal: Use the same protocol for In band and Out of band**

- Web Services Access to CIM/WMI Instrumentation
 - Evolve from DCOM and CIM/XML
 - Transform CIM schemas to XML Schemas via WS-CIM
- Expose Hardware Instrumentation
 - Web Services in your CPU
 - Expose SMASH data
- Leverage Web Services tools and infrastructure
 - SOAP clients and server exposure already exceeds that of CIM and CIMOMs
 - XML and XML Schema tools ubiquitous
 - Large industry wide cross platform investment

- ❑ DSP 226: WS-Management Protocol
- ❑ DSP 227: WS-Management CIM Binding
- ❑ DSP 230: WS-CIM Schema
- ❑ 1.0 Versions 2008
- ❑ 1.1 Versions ~Q1/2010 for ISO
- ❑ 2.x Versions WSRA



Broaden the consistency

- ❑ Why stop at just OS and Hardware?
 - ❑ Other management domains, especially where CIM is used can leverage the same protocols.
 - ❑ Virtualization
 - ❑ Storage
- ❑ Non-Management domains
 - ❑ The line between management data and business logic is blurry.
 - ❑ Leverage a common set of technology investments for “protocol infrastructure”
 - ❑ IT Pro / LOB app builders can reuse their skillsets and tools
 - ❑ This is one area where CIMXML doesn't do it.

WS-Management Protocol Overview

WS-CIM Schema Translation (DSP230)

Integration

WS-Man WSDL Binding for CIM (DSP 227)

Description

WS-Management (DSP 226)

Application

Resource
Addressing
Models

WS-Transfer

WS-Enum

WS-Eventing

Data Transfer

Security profiles

Security

XML, SOAP, WS-Addressing

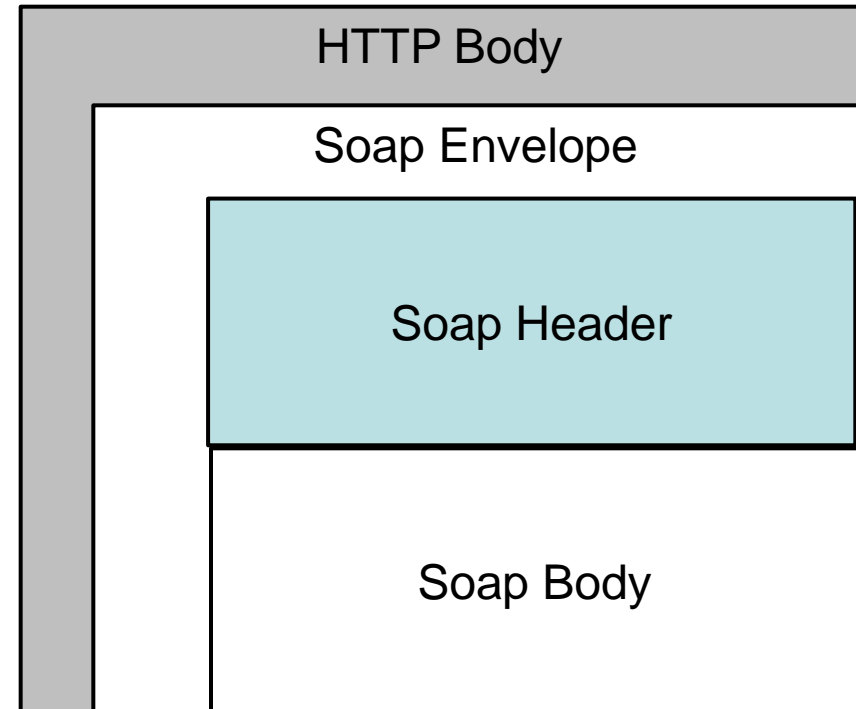
Messaging

HTTPS, TCP

Transports

Soap Background

- ❑ Transport neutral – though we use a SOAP HTTP binding
- ❑ SOAP Envelope has 2 parts
- ❑ SOAP Header
 - ❑ Contains addressing information, “headers”, verb/or method
- ❑ SOAP Body
 - ❑ Payload or data



Basic Soap Envelope

```
<?xml version='1.0' ?>  
  <env:Envelope xmlns:env="http://www.w3.org/2003/05/soap-  
envelope">  
  <env:Header>  
    Header information  
  </env:Header>  
  <env:Body>  
    Body Payload  
  </env:Body>  
</env:Envelope>
```

- WS-Addressing builds upon SOAP to provide addressing and Verb

- Addressing block called “EndPoint Reference” or EPR

```
<wsa:Action>GET</wsa:Action>
```

```
<wsa:EndpointReference>
```

```
  <wsa:Address>xs:anyURI</wsa:Address>
```

```
  <wsa:ReferenceParameters>
```

```
    subaddress
```

```
  </wsa:ReferenceParameters>
```

```
</wsa:EndpointReference>
```

- Note: EPR Wacky-ness

- Its different as a reference and an address! Yay!

EPR Resource Addressing

- ❑ WS-Management “default” EPR addressing
 - ❑ ResourceURI is additional SOAP header
 - ❑ Selectors (required only to act as ‘keys’ for multi-instanced resources)

```
<wsa:To>  
http://myserver/wsman</wsa:To>  
<wsman:ResourceURI>  
  http://schemas.dmtf.org/wbem/wsman/1/wmi/root/cimv2/CIM_Service  
</wsman:ResourceURI>  
<wsman:SelectorSet>  
  <wsman:Selector Name="Name">winrm</wsman:Selector>  
</wsman:SelectorSet>
```

- ❑ Alternative addressing models are used for integration or where appropriate
- ❑ “__cimnamespace” selector can identify CIMOM namespace

Example

```
<s:Envelope>  
  <s:Header>  
    <wsa:To>http://1.2.3.4/wsman/</wsa:To>  
    <wsman:ResourceURI>http://microsoft.com/.../Win32_Service  
  </wsman:ResourceURI>  
    <wsa:Action>http://schemas.xmlsoap.org/ws/2004/09/transfer/Get  
  </wsa:Action>  
    <wsman:SelectorSet>  
      <wsman:Selector Name="Name">winrm</wsman:Selector>  
    </wsman:SelectorSet>  
  </s:Header>  
  <s:Body/>  
</s:Envelope>
```

Modeling Data

CIM Native MOF

```
class Win32_Service : CIM_Service
{
    string Caption;
    string ErrorControl;
    string Name;
    string PathName;
    boolean Started;
}
```

At your WS-CIM_Service

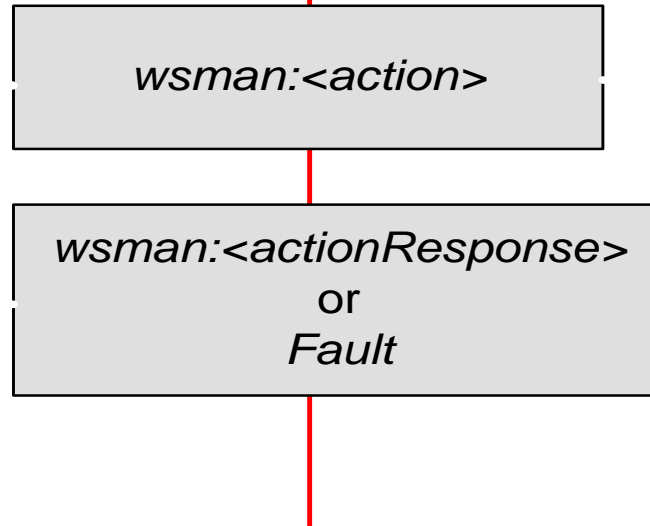
```
<p:CIM_Service xsi:type="p:CIM_Service_Type"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:p="http://schemas.dmtf.org/wbem/wsman/1/wmi/root/cimv2/Win32_Service"
  xmlns:cim="http://schemas.dmtf.org/wbem/wscim/1/common">
  <p:Caption>Windows Remote Management (WS-Management)</p:Caption>
  <p:DisplayName>Windows Remote Management (WS-
    Management)</p:DisplayName>
  <p:ErrorControl>Normal</p:ErrorControl>
  <p>Name>winrm</p>Name>
  <p:PathName>C:\Windows\System32\svchost.exe -k
    NetworkService</p:PathName>
  <p:Started>>false</p:Started>
</p:CIM_Service>
```

Defined by WS-Transfer

- Get
- Put
- Create
- Delete

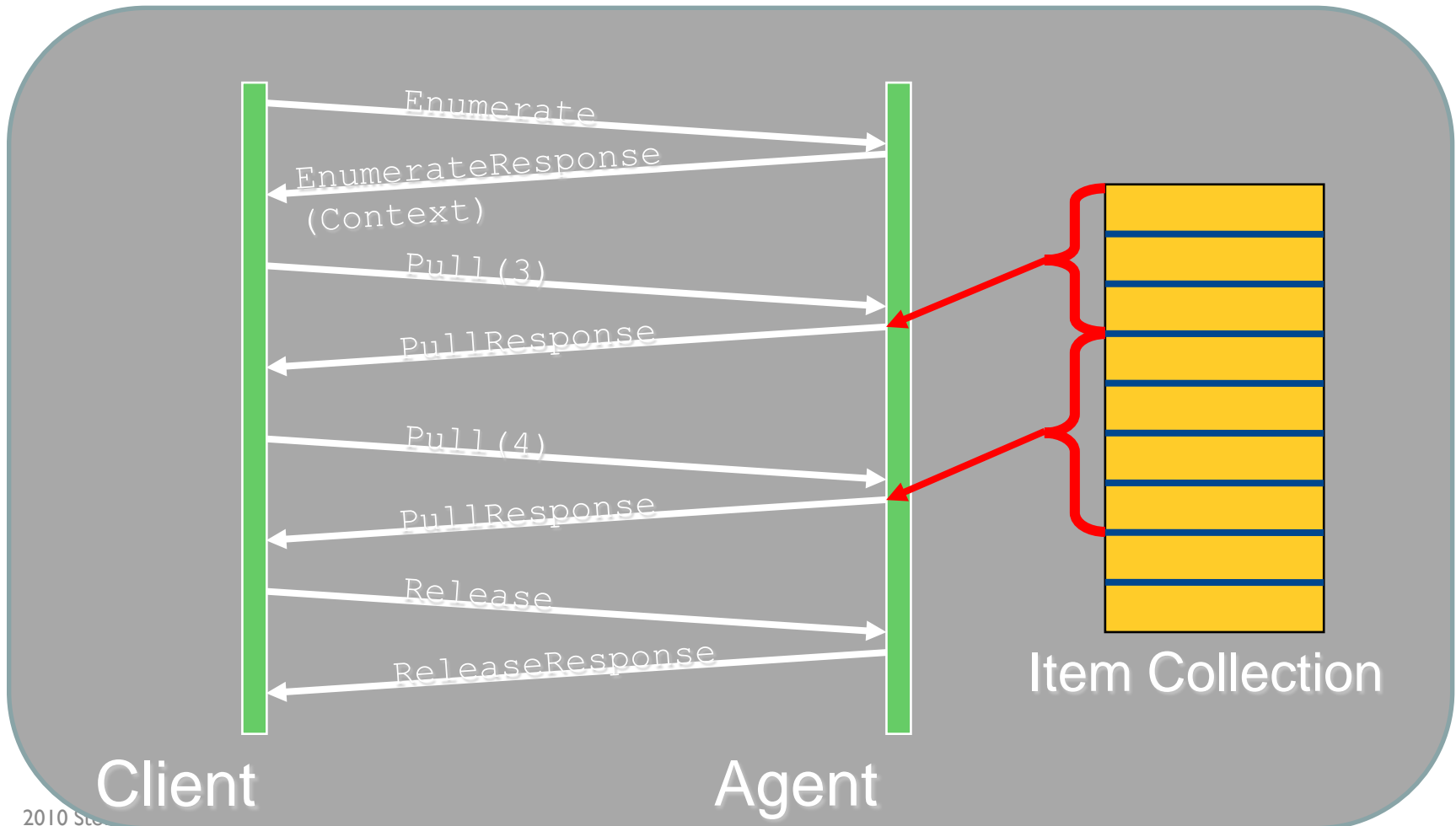
Defined by WS-Mgmt

- Partial Get
- Partial Put



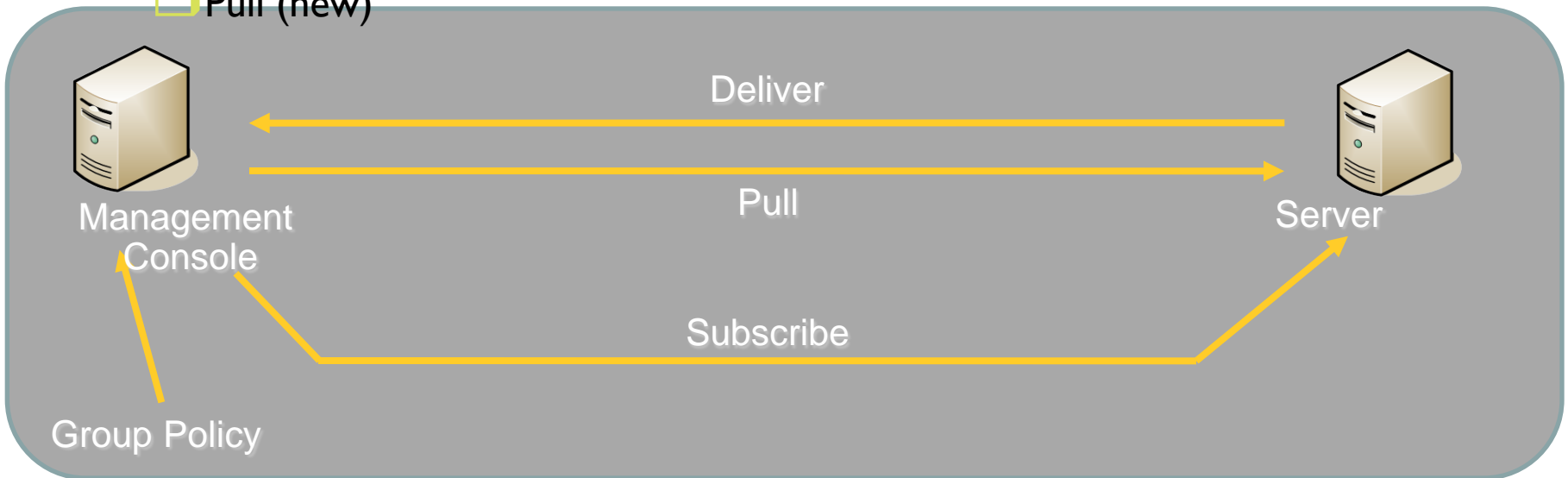
WS-Enumeration

- WS-man allows using partial results and retrieving EPRs



WS-Eventing

- Events can be delivered to a log or to a specific subscriber
- Delivery modes:
 - Push (defined by WS-eventing)
 - PushWithAck (new)
 - Batched (new)
 - Pull (new)



Hardware and systems

- ❑ AMD OPMA based Hardware
- ❑ Dell DRAC management card
- ❑ HP ILO
- ❑ Intel vPro Hardware
- ❑ Broadcom devices
- ❑ Microsoft Windows Vista, Windows XP, Windows Server 2003 R2
- ❑ Microsoft System Center
- ❑ SAP NetWeaver Administrator
- ❑ Oracle Enterprise Manager

Industry Adoption - Community

- ❑ Sun “WISEMAN” open source Java
 - ❑ Used by Java Application Vendors
 - ❑ <https://wiseman.dev.java.net/>
- ❑ Microsoft Embedded WS-Management Stack
 - ❑ Available to embedded vendors
- ❑ OpenWSMAN
 - ❑ Open source C implementation Founded by Intel
 - ❑ Compatible with many CIMOMs
 - ❑ OpenWBEM, Pegasus, others
 - ❑ <http://www.openwsman.org/>
- ❑ OpenPegasus WS-Management Connector

What you can do...

- ❑ What you can do:
 - ❑ Windows is my favorite platform:
 - ❑ Deploy Win7, Vista, XP or 2003 R2
 - ❑ Experiment with WWS-Management Scripting and PowerShell
 - ❑ Prototype Linux Integrations
 - ❑ Windows ***is not*** my favorite platform:
 - ❑ Switch to Windows (just kidding)
 - ❑ Check out the OSS WWS-Man community
 - ❑ WISEMAN for Java
 - ❑ OpenWSMAN for C and CIMOMS
 - ❑ Watch for WWS-Management Enabled Hardware

Storage Management

- ❑ EMC
- ❑ PMC Sierra
- ❑ System Center support to manage SMIS based storage as well as other forthcoming platform support

- ❑ Demo
 - ❑ Jim Davis SMIS WSMAN vs CIMXML example

- ❑ Real and perceived complexity of the VWS protocol
 - ❑ SOAP Stock ticker killed the CORBA
 - ❑ Unrealized complexity of the underlying problem
 - ❑ The complexity of the underlying problem forces designers to end up rebuilding a complicated solution
- ❑ There are some assumptions within VWS design which turned out to not come true
 - ❑ Well intentioned over engineering to remain flexible
 - ❑ Protocol independence (HTTP, SMTP, MSMQ, etc)
 - ❑ Created forced separation between addressing aqnd header/control in SOAP and in HTTP.
- ❑ Example Enumeration

Questions?