

# MTOM en ambiente heterogéneo



Guzmán Llambías

# Agenda

- ❑ Introducción
  - Motivación
- ❑ ¿Qué es MTOM?
- ❑ Soporte en plataformas empresariales
- ❑ Caso de estudio
- ❑ Implementación JEE/.NET
- ❑ Problemas encontrados
- ❑ Trabajos futuros tentativos
- ❑ Conclusiones



# Introducción

- Año 2000: Surgen los primeros WS por acuerdos entre MS e IBM de alcanzar un estándar de comunicación entre aplicaciones tecnológicamente heterogéneas
  - Desarrollan 3 estándares
    - SOAP
    - WSDL
    - UDDI



# SOAP

- ❑ Provee una forma estándar de estructurar mensajes utilizando XML
- ❑ Define mecanismos para utilizar distintos protocolos de transporte para el envío de mensajes
- ❑ Especifica un modelo de procesamiento que indica cómo se deben procesar los mensajes
- ❑ Una forma de adjuntar datos no-XML a los mensajes.



# SOAP

- ❑ Provee una forma estándar de estructurar mensajes utilizando XML
- ❑ Define mecanismos para utilizar distintos protocolos de transporte para el envío de mensajes
- ❑ Especifica un modelo de procesamiento que indica cómo se deben procesar los mensajes
- ❑ Una forma de adjuntar datos no-XML a los mensajes.



# Ventajas de serialización a XML

- Poder reutilizar estándares y herramientas ya existentes en datos no-XML:
  - XML Schema
  - XPath
  - XSLT
  - XQuery
  - XML Encryption
  - XML Signature



# Cómo lo hace?

- ❑ SOAP utiliza la codificación a Base64 para convertir datos binarios a XML
  
- ❑ XMLSchema soporta el uso de este tipo de serialización
  - tipo de datos *xsd:base64binary*



# Codificación base64

- ❑ Base 64 toma los datos binarios y los convierte en una serie de caracteres ASCII
- ❑ Toma 3 octetos de 8 bits y los convierte en 4 caracteres del estándar ASCII
  - Se usan solo 64 caracteres del estándar
  - a-z, A-Z, 0-9, + y /
- ❑ Ejemplo
  - Mary had a little lamb...
  - TWFyeSBoYWQgYSBsaXR0bGUgbGFtYi4uLiA=





# Codificación base64

Texto ASCII	Mary had
Representación hexadecimal	4D 61 72 79 20 68 61 64
Representación binaria agrupada en bytes	01001101 01100001 01110010 01111001 00100000 01101000 01100001 01100100
Representación binaria agrupada de a 6 bits	010011 010110 000101 110010 011110 010010 000001 101000 011000 010110
Representación decimal de los bloques de 6 bits	19 22 05 50 30 18 01 40 24 22 16=
Codificación base64	TWFyeSBoYWQ=



# Debilidades

- ❑ Tamaño del texto aumenta un 25%
- ❑ En base64 6 bits son codificados con 8 bits
  - Base64(010011) = T (8 bits)
- ❑ Gran impacto en datos binarios de gran tamaño!



# MTOM (Message Transmission Optimization Mechanism)

- ❑ MTOM es un estándar de la W3C con el propósito de optimizar la transferencia de archivos binarios entre Web Services.
- ❑ MIME gran protagonista!



- Compuesto por:
  - Abstract SOAP Transmission Optimization Feature
    - mecanismo de optimización abstracto para mensajes SOAP
  - Optimized MIME multipart/Related Serialization of SOAP messages
    - Implementación basada en XOP e indep. del medio de transporte
  - HTTP SOAP Transmission Optimization Feature
    - extensión para ser usada con HTTP



# Ejemplo: Serialización original

HTTP/1.1 200 OK

Content-Length: 4542508

Content-Type: text/xml; charset=utf-8

```
<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetDataResponse xmlns="http://tempuri.org/">
      <GetDataResult>
        UesDBAoAAAAAAGRoATsAAAAAAAAA.../Cj8KBxAEAG9xMgAHAFBBQ0syMDA=
      </GetDataResult>
    </GetDataResponse>
  </s:Body>
</s:Envelope>
```



# Ejemplo: Serialización MTOM

```
HTTP/1.1 200 OK
Content-Length: 3407478
Content-Type: multipart/related; type="application/xop+xml";
start="<http://tempuri.org/0>"; boundary="uuid:774aed72-9fbd-43ba-;
24a5a025899e+id=2"; start-info="text/xml"
Server: Microsoft-HTTPAPI/1.0
MIME-Version: 1.0
Date: Fri, 03 Sep 2010 18:33:41 GMT

--uuid:774aed72-9fbd-43ba-a09a-24a5a025899e+id=2
Content-ID: <http://tempuri.org/0>
Content-Transfer-Encoding: 8bit
Content-Type: application/xop+xml;charset=utf-8;type="text/xml"

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetDataResponse xmlns="http://tempuri.org/">
      <GetDataResult>
        <xop:Include
          href="cid:http%3A%2F%2Ftempuri.org%2F1%2F634191248212968750"
          xmlns:xop="http://www.w3.org/2004/08/xop/include" />
        </GetDataResult>
      </GetDataResponse>
    </s:Body>
  </s:Envelope>

--uuid:774aed72-9fbd-43ba-a09a-24a5a025899e+id=2
Content-ID: <http://tempuri.org/1/634191248212968750>
Content-Transfer-Encoding: binary
Content-Type: application/octet-stream

PK##
##### META-INF/PK##
#####
```



# Soporte

	.NET WSE2	.NET WSE3	WCF 3.5	AXIS1.3	AXIS2	Metro (WSIT)	CXF (XFire)	gSOAP (C++)	SpringWS	JBossWS
Base64Binary	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓
SwA (MIME)	✗	✗	✗	✓	✓	✓	?	✓	?	✗
DIME	✓	✗	✗	✓	✗	✗	✗	✓	?	✗
MTOM	✗	✓	✓	?	✓	✓	✓	✓	✓	✓
WSI Attachment Profile	✗	✗	✗	?	?	?	✗	✗	?	✓



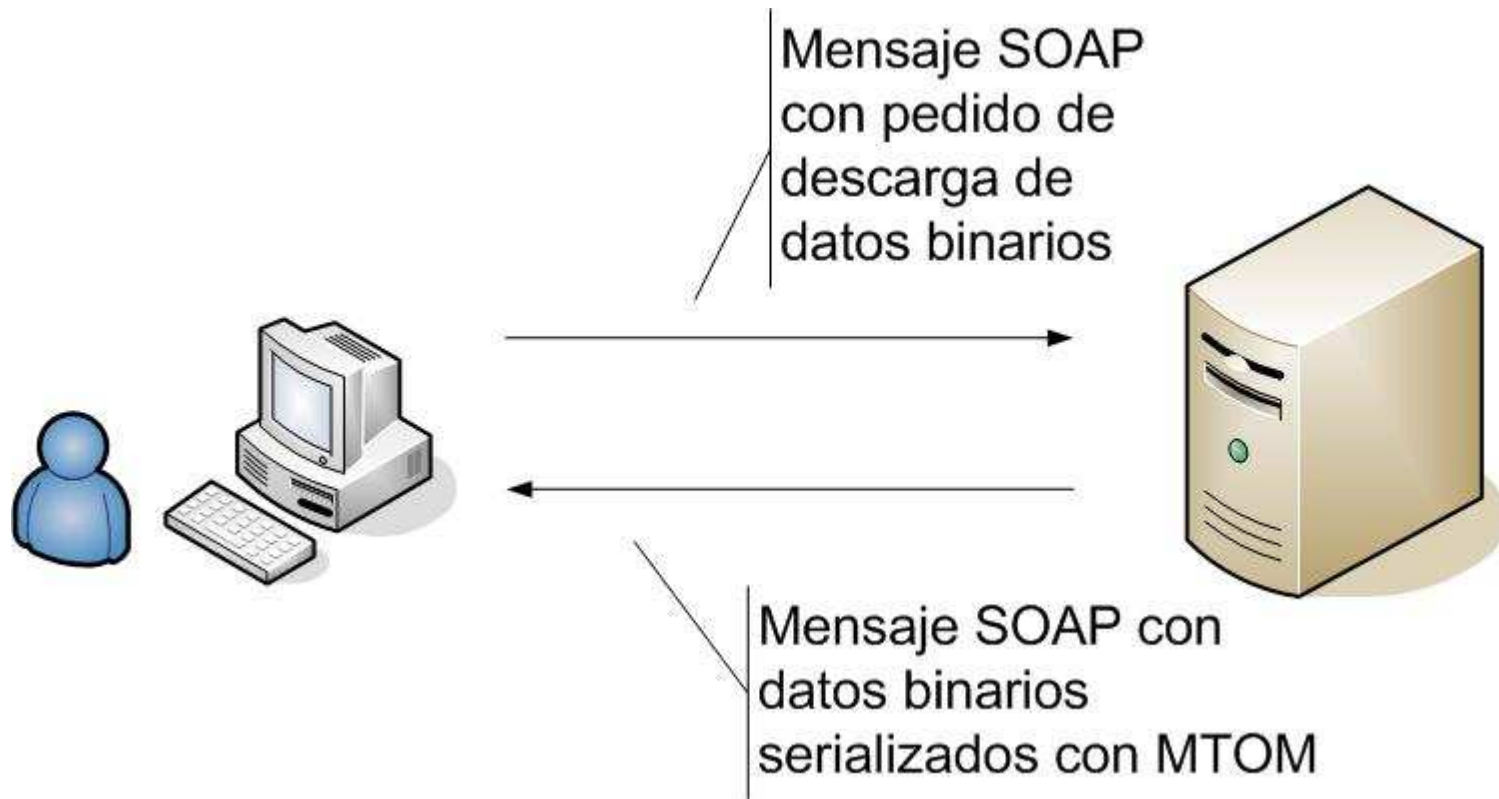
# Consideraciones

- ❑ MTOM fue diseñado para optimizar la serialización de mensajes SOAP que involucren datos binarios de gran tamaño
  - Poco útil con datos binarios pequeños
  
- ❑ Limitado al protocolo HTTP
  - No sirve con binding smtp o jms





# Escenario



# Implementación

## □ .NET 3.5

- Modificar archivos de configuración
  - MessageEncoding = "Mtom"
- Tipos de datos: byte[]

## □ JBossWS – Native 3.2.1

- Agregar anotaciones al código del servicio
  - @MTOM
  - @BindingType(value="http://schemas.xmlsoap.org/wsdl/soap/http?mtom=true")
- Anotaciones y clases especiales para los tipos de datos
  - DataHandler
  - @XmlMimeType("image/png")



# Problemas encontrados

- ❑ WCF exige la inclusión del valor charset dentro del parámetro Content-type
  - charset determina el tipo de codificación
    - UTF-8, UTF-16
  - WS-I Basic Profile 1.2 presenta la restricción
  
- ❑ Out of Memory en cliente JBoss con datos binarios “grandes” (40Mb) al no usar MTOM



# Ejemplo: Serialización MTOM

```
HTTP/1.1 200 OK
Content-Length: 3407478
Content-Type: multipart/related; type="application/xop+
start="<http://tempuri.org/0>"; boundary="uuid:774aed72-9fbd-43ba-
24a5a025899e+id=2"; start-info="text/xml"
Server: Microsoft-HTTPAPI/1.0
MIME-Version: 1.0
Date: Fri, 03 Sep 2010 18:33:41 GMT

--uuid:774aed72-9fbd-43ba-a09a-24a5a025899e+id=2
Content-ID: <http://tempuri.org/0>
Content-Transfer-Encoding: 8bit
Content-Type: application/xop+xml;charset=utf-8;type="text/xml"

<s:Envelope xmlns:s="http://schemas.xmlsoap.org/soap/envelope/">
  <s:Body>
    <GetDataResponse xmlns="http://tempuri.org/">
      <GetDataResult>
        <xop:Include
          href="cid:http%3A%2F%2Ftempuri.org%2F1%2F634191248212968750"
          xmlns:xop="http://www.w3.org/2004/08/xop/include" />
        </GetDataResult>
      </GetDataResponse>
    </s:Body>
  </s:Envelope>

--uuid:774aed72-9fbd-43ba-a09a-24a5a025899e+id=2
Content-ID: <http://tempuri.org/1/634191248212968750>
Content-Transfer-Encoding: binary
Content-Type: application/octet-stream

PK##
##### META-INF/PK##
#####
```



# Trabajos futuros tentativos

- ❑ WS-Security compuesto con MTOM
- ❑ Cómo garantizar confidencialidad/integridad en mensajes serializados con MTOM?
- ❑ Es interoperable?



# Conclusiones

- ❑ MTOM es una solución efectiva ante el requerimiento de datos binarios de gran porte
- ❑ Verificar también tiempos de respuesta
- ❑ Es viable el uso de MTOM en ambientes heterogéneos para los escenarios planteados



# Preguntas



# Contacto



---

[gllambi@fing.edu.uy](mailto:gllambi@fing.edu.uy)