



JAX-WS mélyvíz

Viczián István

<http://jtechlog.blogspot.com>

JUM XII.

2009. november 18.

Web szolgáltatások

- Web szolgáltatás
- W3C definíció: hálózaton keresztüli gép-gép együttműködést támogató szoftverrendszer
- Általában: HTTP(S)/SOAP
- SOAP
- WSDL
- UDDI
- SOA

SOAP

- Stílus
 - RPC-stílus
 - Document stílus
- Kódolás
 - Literal-kódolás
 - SOAP-kódolás = encoded
- Leggyakrabban használt
 - Document/literal

WS-*

- WS-I
 - WS-I Basic Profile
 - WS-I Attachments Profile
 - WS-I Simple SOAP Binding Profile
- WS-
 - WS-Security
 - WS-Reliability
 - WS-Transaction

Implementációs megközelítések

- Bottom up: kód alapján
- Top down: WSDL alapján
- Meet in the middle

Java API-k

- JAXP (1.0 – JSR 5, 1.1 és 1.2 JSR 63, 1.3 és 1.4 JSR 206)
 - DOM
 - SAX
 - StAX
 - Validation
 - XSLT (XPath)
- JAXB (JSR 222)
- SAAJ
- JAX-RPC 1.1 (JSR 101)
- JAX-WS 2.0 (JSR 224)

Implementációk

- GlassFish > JAXP Reference Implementation
 - JDK 5 – JAXP 1.3
 - JDK 6 – JAXP 1.4.0 (StAX)
- (Apache Xerces)
- GlassFish > Metro > JAXB Reference Implementation
- GlassFish > Metro > SAAJ Standard Implementation
- GlassFish > Metro > JAX-WS Reference Implementation

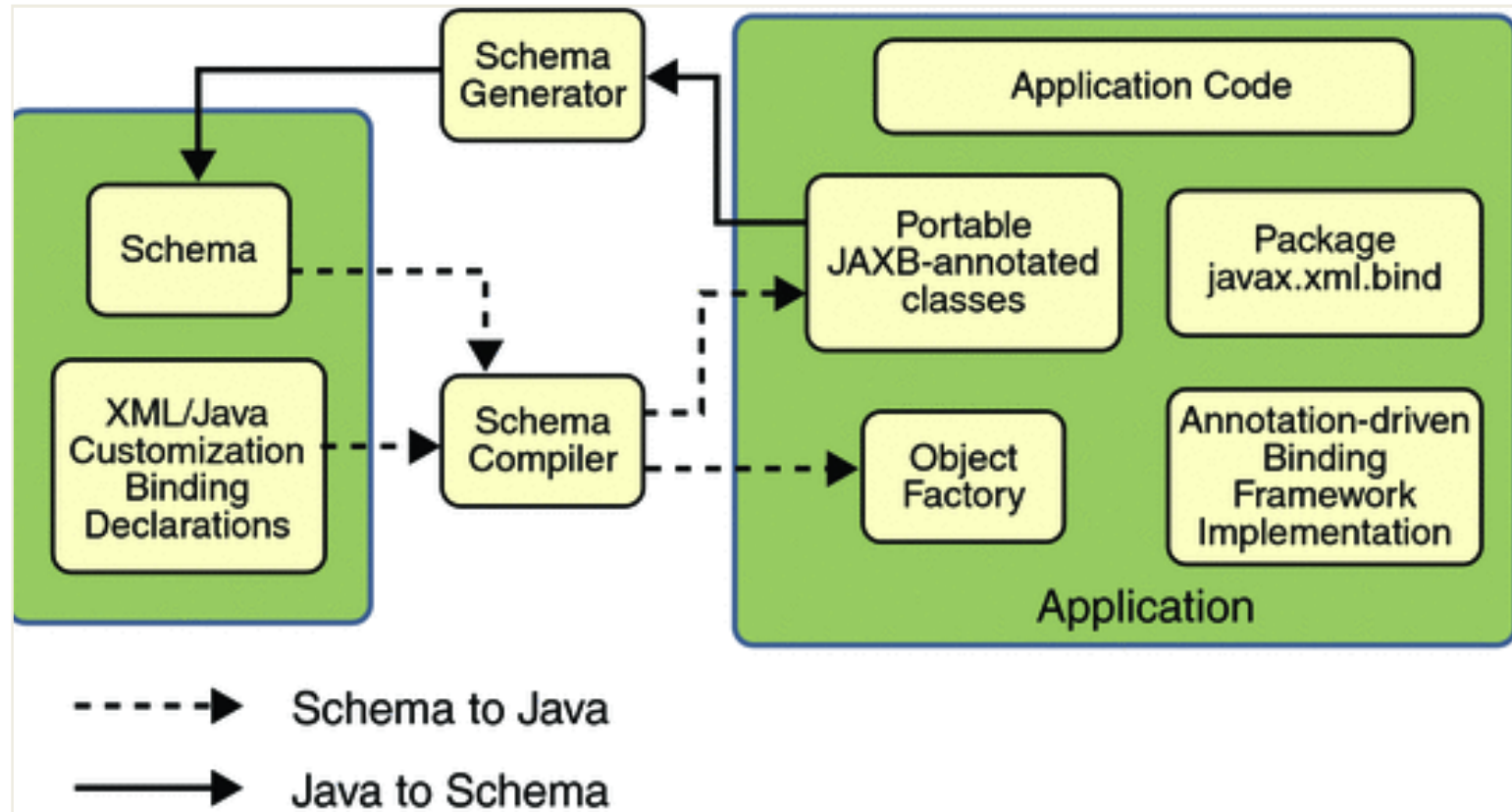
Verziók

- `jdk1.6.0_17`
- `schemagen.exe -version`
 - JAXB 2.1.10 in JDK 6
- `wsimport -version`
 - JAX-WS RI 2.1.6 in JDK 6
- JDK váltásnál verzió váltás
- Pl. előző verzió elnyelte a kivételt
- JDK-ban lévő nem ette meg az ANY-t

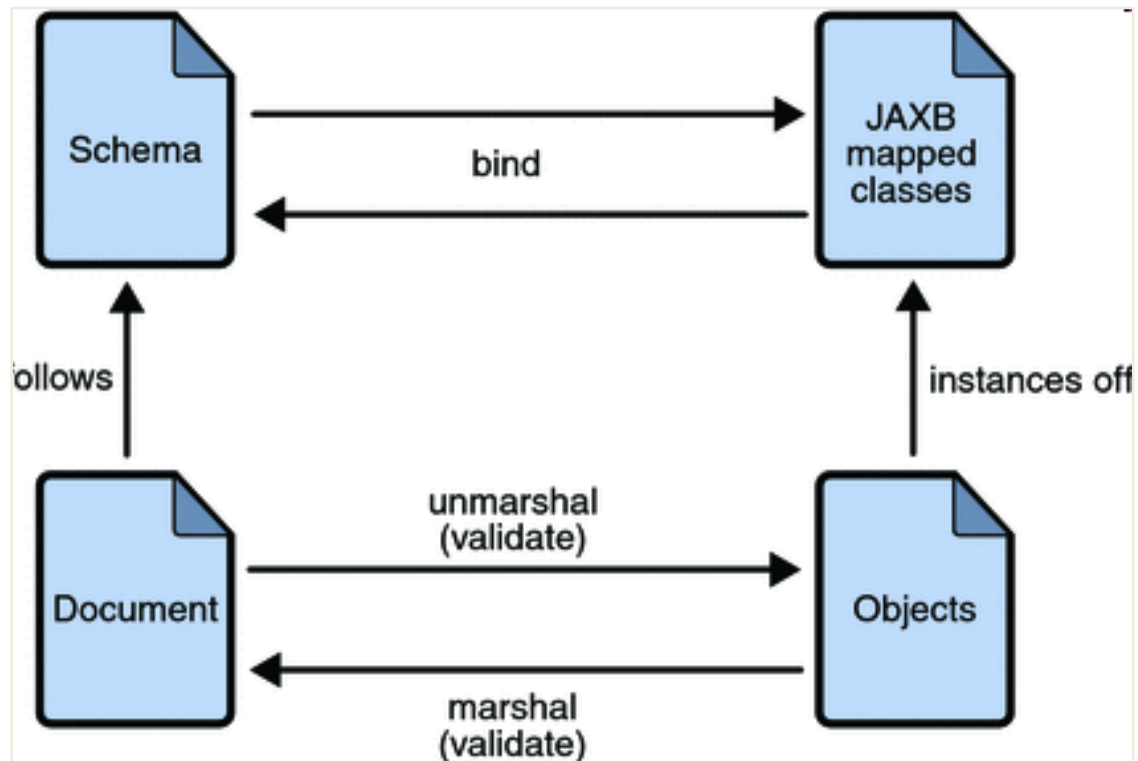
JAXB működése

- Binding <-> mapping
- Fejlesztés egyszerűsítése
- Annotációk
- Default binding
- Inline annotations
- External binding customization file
- Ant

JAXB



JAXB binding process



JAX-WS

- JAX-RPC 1.1 továbbfejlesztése
- RPC stílus és üzenet alapú
- Fejlesztés egyszerűsítése
- Annotációk
- SEI
- Kliens oldalon: proxy
- Handler
- Ant

SOAPUI

- SOA teszt eszköz
- Ingyenes és kereskedelmi verzió
- Web szolgáltatások felderítése, hívása, szimulálás és mock
- Funkcionális és terheléses tesztelés
- IDE-plugins
- SOAPUI – nightly build – jó support

JAXB - Körkörös referencia

- "A cycle is detected in the object graph. This will cause infinitely deep XML,,
- @Transient
- afterUnmarshal metódus
- CycleRecoverable interfész

JAXB - Ős és leszármazott

- Leszármazott: `@XmlRootElement`
- Hivatkozó attribútum: `@XmlElementRef`

JAXB - ANY

- Marshall/unmarshall
 - File, InputStream, URL, Source, Node, XMLStreamReader, XMLEventReader
- any-ben másik névtér, setPrefix

```
<xs:complexType name="AnyType">  
  <xs:sequence>  
    <xs:any minOccurs="0" maxOccurs="unbounded"  
      namespace="##any" processContents="skip" />  
  </xs:sequence>  
</xs:complexType>
```


JAXB - Type és JAXBElement<T>

- JAXB 2.0 – JDK 5 – generikusok megjelenésével
- XJC
- @XmlRootElement, ha a típus nem használható más néven
- JAXBElement<T>, ha használható
- Megoldás
 - Simple-minded binding mode: <xjc:simple />
 - Példányosítás: `new JAXBElement(new QName("uri", "local"), MessageType.class, messageType)`

JAX-WS - BARE/Wrapper

- Return type: List
 - Error starting wsgen:
- `verbose="true" fork="true"`
- `java.util.List` is an interface, and JAXB can't handle interfaces.
- Amennyiben egy metódusnak több paramétere van, az is hiba.
 - Exception in thread "main"
com.sun.xml.ws.model.RuntimeModelerException:
runtime modeler error: SEI [osztály neve] has method [metódus neve]
annotated as
BARE but it has more than one parameter bound to body. This is
invalid.
Please annotate the method with annotation:
`@SOAPBinding(parameterStyle=SOAPBinding.ParameterStyle.WRAPPE
D)`

JAX-WS - JAXB binding nélkül

- implements `Provider<T>`
- `SOAPMessage`
- `javax.xml.transform.Source`
- `invoke(T)`
- Payload mode

JAX-WS - Attachment kezelés

- SwA
- MTOM
 - SOAP Message Transmission Optimization Mechanism
 - XOP XML-binary Optimized Packaging
 - Multipart <-> BASE64 (33% méret növekedés)
 - xop:Include hivatkozás
- Kötelező az attachment, nem lehet opcionális:
0 hosszú bájtömb

Tippek

- JAXBContext cache
- Inout paraméter: Holder<T>
- Aszinkron hívás
- Integrációs réteg
- com.sun.xml.ws.developer.SchemaValidation
- Cserélhető encoding: JSON
- Cserélhető transport, binding: In-VM, JMS, SMTP
- StAX a hatékonyságért

Problémák

- Hiba: név ugyanaz, mint a portType name attribútuma
- java.dev.net rendelkezésre állás (catalog)
- Nem szimmetrikus – explicit WSDL generálás
- Nem adható meg kötelező paraméter
- SEI-ben nem lehet konstans - no support
- wsdlLocation does not resolve if it is a relative URI
- TODO: `jaxws-ri\src\com\sun\xml\ws\api\message\Message.java`

Eszköz problémák

- NetBeans minőség, ad-hoc, nem megtervezett képernyők
- Pl. validate: NetBeans validate:
<mime:multipartRelated>
"ERROR: At least one <soap:body> element is required per input/ouput message in a soap operation."

Spring integráció

- GlassFish > Metro > JAX-WS commons > Spring support
- NetBeans generálás
- Több metadata paraméter megadás

Naplózás

- Handler
 - Logical handler
 - Protocol handler: SOAPHandler
- NetBeans deadlock: Handler, Tomcat, Log4J
 - Thread.sleep(100);
- DOM
 - JAX-WS RI com.sun.xml.ws.api.handler.MessageHandler
- Eltérő működés
 - message.readEnvelopeAsSource(); - nem ad vissza semmit
 - message.readPayloadAsSource(); - jót ad vissza
- Eltérő működés
 - XMLOutputFactory
 - com.sun.xml.ws.api.streaming.XMLStreamWriterFactory
- Aszinkron

Alternatíva

- RMI
- Apache Axis
- Apache CXF
- REST (JSR 311)

Forrás

<http://www.w3.org>

<http://ws-i.org>

<http://jcp.org>

<https://jaxp.dev.java.net/>

<https://jaxb.dev.java.net/>

<https://saaj.dev.java.net/>

<https://jax-ws.dev.java.net/>

Mark D. Hansen – SOA Using Java Web Services

The Java EE 5 Tutorial

