

# SOAP und WSDL in der Praxis

## Lernziele

- ✓ Prinzipielle Aufbau von WSDL-Beschreibungen
- ✓ Protokoll-Bindungen in WSDL
- ✓ Google-WSDL lesen können
- ✓ Vor- und Nachteile von WSDL

- Wie wird WSDL verwendet?
- Anforderungen der Praxis
- Erweiterungen von SOAP/WSDL

heute

## Wie wird WSDL verwendet?

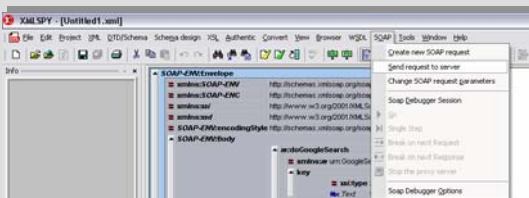
## Wie wird WSDL verwendet?



.net und Apache Axis

- Aus einer WSDL-Beschreibung wird *automatisch* eine Schnittstelle zur Anwendung (Stubs) generiert.
- Stubs abstrahieren von SOAP und vom konkreten Übertragungsprotokoll.
- Web-Dienst erscheint als *lokale* Bibliothek.
- umgekehrt kann auch aus Anwendungsprogramm automatisch eine WSDL-Beschreibung erzeugt werden

## Wie wird WSDL verwendet?



XML Spy

- aus WSDL-Beschreibung SOAP-Nachricht generieren
- Parameter der SOAP-Anfrage anpassen
- SOAP-Nachricht an Web-Dienst senden
- sehr gut geeignet zum schnellen testen

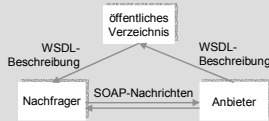
## Web-Dienste: Die Vision

### Beispiel:

- personalisierte Informations-Webseite benötigt Echtzeitkurse von bestimmten Aktien einer bestimmten Börse
- Informations-Webseite sucht entsprechenden Web-Dienst für Echtzeitkurse und ruft diesen automatisch auf.

dynamische Einbindung zur Laufzeit

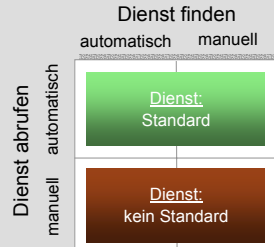
## Dynamische Einbindung zur Laufzeit



1. Anwendung sucht in einem Verzeichnis einen Web-Dienst, z.B. für Echtzeitkurse von Aktien
2. Suchergebnis: WSDL-Beschreibung
3. Anwendung erzeugt aus WSDL-Beschreibung Stubs
4. Anwendung ruft Web-Dienst als lokale Bibliothek auf

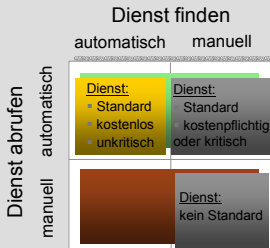
Dienst muss automatisch gefunden und aufgerufen werden.

## Aufruf des Dienstes



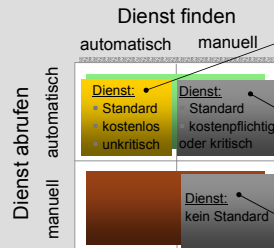
- gegeben: WSDL-Beschreibung
- Um den Web-Dienst *automatisch* abzurufen, muss er standardisiert sein:
- WSDL beschreibt zwar die *Syntax* der Schnittstelle, nicht aber die *Bedeutung* der Prozedur/Parameter.
- Bedeutung muss außerhalb von WSDL festgelegt (standardisiert) sein
- Beispiel: Float Aktienkurs(Integer WKN, String Boersenplatz)

## Dienst finden



- gegeben: standardisierter Web-Dienst
- gesucht: Anbieter des Web-Dienstes (seine WSDL-Beschreibung)
- Um einen Anbieter *automatisch* zu finden, muss der Dienst kostenlos und unkritisch sein.
- andernfalls Rücksprache und Verhandlungen nötig

## Die Realität



- standardisierte Informationsdienste (kostenlos und unkritisch)
- standardisierte Informationsdienste (kostenpflichtig und/oder kritisch)
- Google-Web-Dienst, E-Business

## Schlussfolgerung



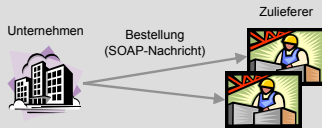
### Einbindung eines Web-Dienstes

- zur Laufzeit: Vollautomatisches Einbinden prinzipiell mit WSDL möglich, wird aber die *Ausnahme* bleiben.
- zur Entwicklungszeit: kein vollautomatisches Einbinden, aber WSDL *erleichtert* das Einbinden ganz erheblich

## Anforderungen der Praxis



## Anforderung des Praxis



- Absender eindeutig identifizierbar?
- Nachricht unversehrt?
- Nachricht *genau einmal* übermittelt?
- Antwort nach bestimmter Zeit garantiert?
- Passt die Nachricht in den Geschäftsprozess (z.B. keine Bestellung ohne vorherige Bestellanfrage)?

## Realisierung mit SOAP

- Absender eindeutig identifizierbar?
- Nachricht unversehrt?
- Nachricht *genau einmal* übermittelt?
- Antwort nach bestimmter Zeit garantiert?
- Passt die Nachricht in den Geschäftsprozess?

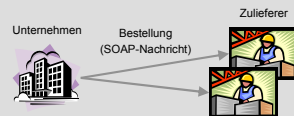
- *keine* der Anforderungen von SOAP direkt unterstützt
- können aber mit Zusatzinformationen im Briefkopf der SOAP-Nachricht realisiert werden

## Zusatzinformationen im SOAP-Briefkopf

- digitale Signatur
  - Identifikation des Absenders
  - Unversehrtheit der Nachricht
- Anforderung einer Empfangsbestätigung
  - Nachricht mindestens einmal zugestellt
- eindeutige Nachrichtenreferenz
  - Erkennung von Duplikaten
- Verweis auf vorherige Nachrichten, z.B. Bestellanfrage
  - Berücksichtigung des Workflows

- Absender eindeutig identifizierbar?
- Nachricht unversehrt?
- Nachricht *genau einmal* übermittelt?
- Antwort nach bestimmter Zeit garantiert?
- Passt die Nachricht in den Geschäftsprozess?

## So einfach ist es dann doch nicht...



Problem der Interoperabilität (Zusammenarbeitsfähigkeit):

- Zulieferer soll ersetzt werden.
- Neuer Zulieferer bekommt WSDL-Beschreibung.
- WSDL beschreibt zwar die *Syntax* der Schnittstelle, einschl. der Zusatzinformationen im Briefkopf.
- Was die *Bedeutung* der Zusatzinformationen ist, muss zwischen Unternehmen und Zulieferer geklärt werden.

## Beispiel Fax

Fax ist hingegen uneingeschränkt interoperabel:

- Ein Faxgerät kann durch ein anderes ersetzt werden, *ohne* mit potentiellen Sendern Protokolle auszuhandeln: wie z.B. der Eingang eines Fax bestätigt wird
- Grund: es gibt einen allgemein akzeptierten internationalen Fax-Standard

## Interoperabilität von Web-Diensten

- SOAP/WSDL zwar akzeptierte internationale Standards.
- SOAP ist ein Format, mit dem Web-Dienste Nachrichten austauschen.
- WSDL beschreibt die Syntax der Schnittstelle eines Web-Dienstes.
- Wie aber z.B. ein Sender authentifiziert werden kann, wird weder in SOAP noch WSDL festgelegt.
- Ein Web-Dienst kann *nicht* ohne weiteres durch einen anderen ersetzt werden, selbst wenn beide die *gleiche* abstrakte Schnittstelle haben.

## Lösung: Erweiterungen von SOAP/WSDL

- Sicherheit:  
WS Security
- Zuverlässigkeit  
WS Reliable Messaging
- qualitativen Aspekten (*quality of service*):  
WS Policy
- geschäftlicher Workflow:  
BPEL
- sehr gute Übersicht:  
<http://www-306.ibm.com/software/solutions/webservices/pdf/SecureReliableTransactedWSAction.pdf>

## Beispiel

```
<?xml version="1.0" encoding="UTF-8" ?>
<Envelope xmlns="http://schemas.xmlsoap.org/soap/envelope/">
  <Header>
    <wsa:Security xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/soap/envelope/">
      <wsu:BinarySecurityToken
        xmlns:wsu="http://schemas.xmlsoap.org/ws/2004/08/soap/envelope/"
        ValueType="http://www.w3.org/2001/04/xmldsig#sha1"
        EncodedType="Base64Binary"
        Id="WS-Security-Token"
        wsu:Value="Fw0wMTEwMTAwMD"
        wsu:DigestValue="Fw0wMTEwMTAwMD"
      />
    </wsa:Security>
  </Header>
  <Body>
    <wsr:Sequence
      xmlns:wsr="http://schemas.xmlsoap.org/ws/2004/08/soap/envelope/"
      MessageID="http://schemas.xmlsoap.org/soap/envelope/123456789"
      Version="1.0"
    />
  </Body>
</Envelope>
```

WS Reliable Messaging wird exemplarisch betrachtet

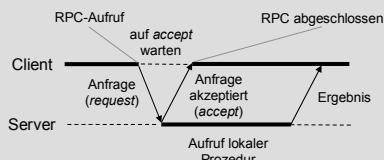
## Zuverlässige Kommunikation

## Beispiel

Buchungsanfrage:

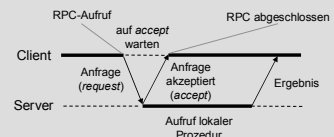
- Unternehmen ruft Buchungsprozedur eines Reiseunternehmens auf
- Buchung nimmt häufig längere Zeit in Anspruch.
- deshalb *asynchroner* Aufruf der Buchungsprozedur

## Asynchrone Buchungsprozedur



- Buchungsanforderung wird vom Client aufgerufen
- Client wartet bis er vom Server Bestätigung der Annahme bekommen hat.
- Server schickt später das endgültige Ergebnis der Buchungsanfrage

## Zuverlässige Buchungsanfrage



- Anfrage (*request*): genau einmal zustellen
- Annahme (*accept*): mindestens einmal zustellen
- Ergebnis: genau einmal zustellen
- Annahme immer vor Ergebnis, ansonsten wird evtl. eine zweite Buchungsanfrage gestellt.

➔ Annahme und Ergebnis bilden eine Sequenz

## WS Reliable Messaging

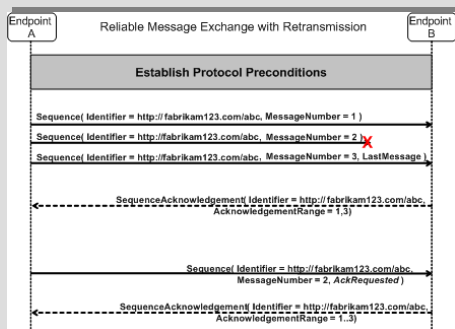
- XML-Protokoll für zuverlässige Kommunikation
  - kann im Briefkopf von SOAP-Nachrichten oder in WSDL-Beschreibungen verwendet werden
  - gemeinsamer Vorschlag von BEA, IBM, Microsoft und TIBCO
  - von März 2003
  - noch kein etablierter Standard
- <http://www-106.ibm.com/developerworks/library/ws-rm/>

## Beispiel

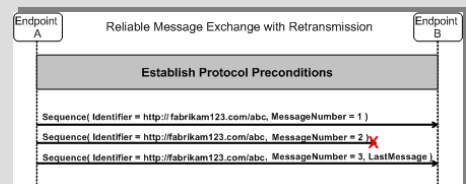


- Sequenz von drei Nachrichten
- jede Nachricht genau einmal zustellen
- Reihenfolge beachten
- Empfang bestätigen

## Beispiel



## Sequenz von SOAP-Nachrichten senden



## Erste SOAP-Nachricht

```
<S:Envelope xmlns:wsm="http://schemas.xmlsoap.org/ws/2003/03/rm"
  xmlns:wsa="http://schemas.xmlsoap.org/ws/2003/03/addressing">
  <S:Header>
    <wsa:MessageID>
      http://Business456.com/guid/71e0654e-5ce8-477b-bb9d-34f05cfc9e
    </wsa:MessageID>
    <wsa:To>http://fabrikam123.com/serviceB/123</wsa:To>
    <wsa:ReplyTo>
      <wsa:Address>http://Business456.com/serviceA/789</wsa:Address>
    </wsa:ReplyTo>
    <wsm:Sequence>
      <wsu:Identifier>http://Business456.com/RM/ABC</wsu:Identifier>
      <wsm:MessageNumber>1</wsm:MessageNumber>
    </wsm:Sequence>
  </S:Header> ...
```

- WS RM benutzt WS Addressing
- Nachricht hat eindeutige Referenz (MessageID)

## Erste SOAP-Nachricht

```
<S:Envelope xmlns:wsm="http://schemas.xmlsoap.org/ws/2003/03/rm"
  xmlns:wsa="http://schemas.xmlsoap.org/ws/2003/03/addressing">
  <S:Header>
    <wsa:MessageID>
      http://Business456.com/guid/71e0654e-5ce8-477b-bb9d-34f05cfc9e
    </wsa:MessageID>
    <wsa:To>http://fabrikam123.com/serviceB/123</wsa:To>
    <wsa:ReplyTo>
      <wsa:Address>http://Business456.com/serviceA/789</wsa:Address>
    </wsa:ReplyTo>
    <wsm:Sequence>
      <wsu:Identifier>http://Business456.com/RM/ABC</wsu:Identifier>
      <wsm:MessageNumber>1</wsm:MessageNumber>
    </wsm:Sequence>
  </S:Header> ...
```

- WS RM benutzt WS Utilities
- Sequenz hat eindeutige Referenz (Identifier)
- erste Nachricht der Sequenz (MessageNumber)

## Zweite SOAP-Nachricht

```
<S:Envelope ...>
<S:Header>
  <wsa:MessageID>
    http://Business456.com/guid/daa7d0b2-c8e0-476e-a9a4-d164154e38de
  </wsa:MessageID>
  <wsa:To>http://fabrikam123.com/serviceB/123</wsa:To>
  <wsa:ReplyTo>
    <wsa:Address>http://Business456.com/serviceA/789</wsa:Address>
  </wsa:ReplyTo>
  <wsrm:Sequence>
    <wsu:Identifier>http://Business456.com/RM/ABC</wsu:Identifier>
    <wsrm:MessageNumber>2</wsrm:MessageNumber>
  </wsrm:Sequence>
</S:Header> ...
```

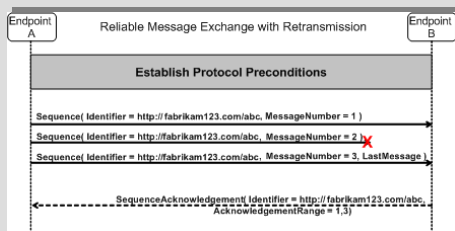
- neue Nachrichten-Referenz
- alte Sequenz-Referenz

## Dritte und letzte SOAP-Nachricht

```
<S:Envelope ...>
<S:Header>
  <wsa:MessageID>
    http://Business456.com/guid/0baaf88d-483b-4ecf-a6d8-a7c2eb546817
  </wsa:MessageID>
  <wsa:To>http://fabrikam123.com/serviceB/123</wsa:To>
  <wsa:ReplyTo>
    <wsa:Address>http://Business456.com/serviceA/789</wsa:Address>
  </wsa:ReplyTo>
  <wsrm:Sequence>
    <wsu:Identifier>http://Business456.com/RM/ABC</wsu:Identifier>
    <wsrm:MessageNumber>3</wsrm:MessageNumber>
    <wsrm:LastMessage>
      <wsrm:Sequence>
        <wsu:Identifier>http://Business456.com/RM/ABC</wsu:Identifier>
        <wsrm:MessageNumber>2</wsrm:MessageNumber>
      </wsrm:Sequence>
    </wsrm:LastMessage>
  </wsrm:Sequence>
</S:Header> ...
```

- neue Nachrichten-Referenz
- alte Sequenz-Referenz

## Bestätigung

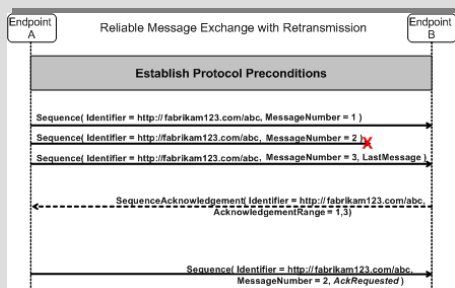


## Bestätigung

```
<S:Envelope ...>
<S:Header>
  <wsa:MessageID>
    http://fabrikam123.com/guid/0baaf88d-483b-4ecf-a6d8-a7c2eb546817
  </wsa:MessageID>
  <wsa:To>http://Business456.com/serviceA/789</wsa:To>
  <wsa:ReplyTo>
    <wsa:Address>http://fabrikam123.com/serviceB/123</wsa:Address>
  </wsa:ReplyTo>
  <wsrm:SequenceAcknowledgment>
    <wsu:Identifier>http://Business456.com/RM/ABC</wsu:Identifier>
    <wsrm:AcknowledgmentRange Upper="1" Lower="1"/>
    <wsrm:AcknowledgmentRange Upper="3" Lower="3"/>
  </wsrm:SequenceAcknowledgment>
</S:Header> ...
```

- erste und dritte Nachricht angekommen

## Zweite Nachricht nochmals senden

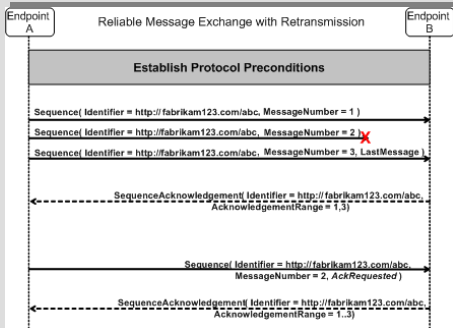


## Zweite Nachricht nochmals senden

```
<S:Envelope ...>
<S:Header>
  <wsa:MessageID>
    http://Business456.com/guid/0baaf88d-483b-4ecf-a6d8-a7c2eb546817
  </wsa:MessageID>
  ...
  <wsrm:Sequence>
    <wsu:Identifier>http://Business456.com/RM/ABC</wsu:Identifier>
    <wsrm:MessageNumber>2</wsrm:MessageNumber>
  </wsrm:Sequence>
  <wsrm:AckRequested>
    <wsu:Identifier>http://Business456.com/RM/ABC</wsu:Identifier>
  </wsrm:AckRequested>
</S:Header> ...
```

- zweite Nachricht der Sequenz
- Bestätigung vom Empfänger verlangt

## Endgültige Bestätigung



© Klaus Schild, 2004

37

## Endgültige Bestätigung



```
<S:Envelope ...>
<S:Header>
  <wsa:MessageID>
    http://fabrikam123.com/guid/0baaf88d-483b-4ecf-a6d8-a7c2eb546817
  </wsa:MessageID>
  <wsa:To>http://Business456.com/serviceA/789</wsa:To>
  <wsa:ReplyTo>
    <wsa:Address>http://fabrikam123.com/serviceB/123</wsa:Address>
  </wsa:ReplyTo>
  <wsrm:SequenceAcknowledgment>
    <wsu:Identifier>http://Business456.com/RM/ABC</wsu:Identifier>
    <wsrm:AcknowledgmentRange Upper="3" Lower="1"/>
  </wsrm:SequenceAcknowledgment>
</S:Header> ...
```

Sequenz vollständig empfangen

© Klaus Schild, 2004

38

## Was gibt es noch?



- **wsu:Expires:** Verfallsdatum und -zeit
- **wsrm:SequenceFault:** Fehlermeldung bzgl. Sequenzen, wie z.B. `wsrm:SequenceTerminated` oder `wsrm:InvalidAcknowledgement`
- **wsrm:DeliveryAssurance:** verlangte Zuverlässigkeit, z.B. `AtLeastOnce` oder `InOrder`
- **wsrm:InactivityTimeout:** wie lange ein Empfänger höchstens auf die nächste Nachricht warten muss

© Klaus Schild, 2004

39

## Integration in SOAP oder WSDL?



- WS-RM-Elemente können im Briefkopf einer SOAP-Nachricht erscheinen.
- Elemente wie z.B. `<wsrm:DeliveryAssurance Value="wsrm:AtLeastOnce"/>` machen in SOAP-Nachrichten allerdings keinen Sinn.
- Solche Elemente werden in einer WSDL-Beschreibung integriert.

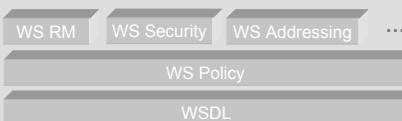
© Klaus Schild, 2004

40

## Zweistufige Integration in WSDL



1. WS-RM-Elemente werden als Kommunikationsregeln (*policy*) formuliert – und zwar mit Hilfe von WS Policy.
  2. Diese Kommunikationsregeln werden dann in WSDL integriert.
- Gleiches Vorgehen auch bei anderen Erweiterungen:



© Klaus Schild, 2004

41

## Beispiel



```
<wsdl:definitions name="PurchaseOrder" ...>
  <wsdl:message name="OrderRequest">
    <wsp:Policy>
      <wsrm:DeliveryAssurance Value="wsrm:ExactlyOnce"
        wsp:Usage="wsp:Required"/>
    </wsp:Policy>
  </wsdl:message>
  <wsdl:message name="OrderAccept">
    ...
  </wsdl:message>
  ...
</wsdl:definitions>
```

- hier `wsp:Policy` als Erweiterungselement
- tatsächlich erlaubt aber WSDL für message und portType *keine* Erweiterungselemente
- daher etwas andere Kodierung mit Attributen

© Klaus Schild, 2004

42

# Bewertung der Erweiterungen

## Bewertung der Erweiterungen

- + einzelne Erweiterungen *unabhängig* voneinander
- + meist gemeinsame Vorschläge von Microsoft und IBM
- noch *keine* etablierten Standards
- WS Policy Grammatik zur Festlegung von qualitativen Aspekten, einzelne qualitative Aspekte müssen noch standardisiert werden

Keine *prinzipiellen* Hürden, jedoch noch ein langer Standardisierungsweg zu gehen!

## Lernziele

- ☑ Wie wird WSDL verwendet?
- ☑ Anforderungen der Praxis
- ☑ Erweiterungen von SOAP/WSDL
- Kurze Wiederholung von Themen, die bei Rücksprachen Probleme bereiteten.
- Diskussionswürdiges

nächste  
Woche