Improving Data Quality for Web Services Composition

Xitong Li, Stuart Madnick, Hongwei Zhu, Yushun Fan

Stuart E. Madnick
MIT Sloan School of Management

smadnick@mit.edu

The 7th International Workshop on Quality in Databases (QDB’09)
August 24th, 2009, Lyon, France
Agenda

• **Research Motivation** – *focused on data quality*
  – Problem of **data misinterpretation**: Different Web services have different assumptions about the interpretation of the exchanged data among them in the composition

• **Our Solution Approach**
  – Representation of Ontology and Context
  – Semantic Annotation
  – Reconciliation Approach
  – Prototype: *CMT*
  – Summary and Future Work
Real-world Scenarios:
GetQuote operation of the StockQuote service
Response for IBM

1. What currency it uses?
2. What does “B” mean?

(Even if you knew “B” meant Billion – is it US billion or UK billion?)
Another Example: Response for ITWO

1. What currency does it use?
2. What does “M” mean?
Different Example: Xignite Web Service

GetTotalAssets
What is this date 04/23/2009?
What if it was 04/05/06?

MSFT Total Assets: 68,853
GetTotalAssets for ITWO

MSFT Total Assets: 68,853 vs. ITWO Total Assets: 313,776

(ITWO has Five Times more assets than MSFT?)
### Balance Sheets (in millions)

<table>
<thead>
<tr>
<th>Assets</th>
<th>March 1, 2009 (Unaudited)</th>
<th>June 30, 2009 (Audited)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Current assets</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cash and cash equivalents</td>
<td>$ 7,285</td>
<td>$ 10,339</td>
</tr>
<tr>
<td>Short-term investments</td>
<td>18,055</td>
<td>13,323</td>
</tr>
<tr>
<td>Total cash, cash equivalents</td>
<td>25,340</td>
<td>23,682</td>
</tr>
<tr>
<td>Inventories</td>
<td>5,182</td>
<td>13,589</td>
</tr>
<tr>
<td>Deferred income taxes</td>
<td>657</td>
<td>985</td>
</tr>
<tr>
<td>Other</td>
<td>1,526</td>
<td>2,017</td>
</tr>
<tr>
<td>Total current assets</td>
<td>40,724</td>
<td>43,242</td>
</tr>
<tr>
<td>Property and equipment</td>
<td>7,412</td>
<td>8,542</td>
</tr>
<tr>
<td>Equity and other investments</td>
<td>4,117</td>
<td>6,588</td>
</tr>
<tr>
<td>Goodwill</td>
<td>12,554</td>
<td>12,100</td>
</tr>
<tr>
<td>Intangible assets, net</td>
<td>1,756</td>
<td>1,573</td>
</tr>
<tr>
<td>Deferred income taxes</td>
<td>556</td>
<td>549</td>
</tr>
<tr>
<td>Total assets</td>
<td>$ 68,853</td>
<td>$ 72,793</td>
</tr>
</tbody>
</table>
# Table of Contents

## PART I. FINANCIAL INFORMATION

### ITEM 1. FINANCIAL STATEMENTS

**Condensed Consolidated Balance Sheets**

*In thousands, except par value (unaudited)*

<table>
<thead>
<tr>
<th></th>
<th>March 31, 2009</th>
<th>December 31, 2008</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ASSETS</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Current assets</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Cash and cash equivalents</td>
<td>$157,679</td>
<td>$229,039</td>
</tr>
<tr>
<td>Restricted cash</td>
<td>8,901</td>
<td>5,777</td>
</tr>
<tr>
<td>Accounts receivable, net</td>
<td>22,980</td>
<td>25,846</td>
</tr>
<tr>
<td>Other current assets</td>
<td>5,418</td>
<td>9,577</td>
</tr>
<tr>
<td>Total current assets</td>
<td>195,978</td>
<td>275,362</td>
</tr>
<tr>
<td>Premises and equipment, net</td>
<td>4,003</td>
<td>4,915</td>
</tr>
<tr>
<td>Goodwill</td>
<td>16,664</td>
<td>16,994</td>
</tr>
<tr>
<td>Non-current deferred tax asset</td>
<td>5,836</td>
<td>7,289</td>
</tr>
<tr>
<td>Other non-current assets</td>
<td>3,311</td>
<td>5,024</td>
</tr>
<tr>
<td><strong>Total assets</strong></td>
<td><strong>226,914</strong></td>
<td><strong>313,025</strong></td>
</tr>
</tbody>
</table>

| **LIABILITIES AND STOCKHOLDERS’ EQUITY** |                |                   |
| Current liabilities    |                |                   |
| Accounts payable       | $3,878         | $4,855            |
| Accounts payable       | 16,499         | 15,316            |
| Accrued compensation and related expenses | 11,489        | 18,079            |
| Deferred revenue       | 28,297         | 30,228            |
| Total current liabilities | 90,463       | 91,978            |
| Total long-term debt, net |              |                   |
| Taxes payable          | 5,292          | 8,348             |
| **Total liabilities**  | **95,755**     | **163,316**       |
| Commitments and contingencies |            |                   |
| Stockholders’ equity   |                |                   |
| Preferred Stock, 50,001 par value, 5,000 shares authorized, none issued and outstanding | —             | —                 |
| Series A junior participating preferred stock, 50,001 par value, 2,000 shares authorized, none issued and outstanding | —             | —                 |

---
Research Motivation

• **Goal of Using Web Services**
  – Web services composition/integration/mashup
    • **Combine multiple web services to create a composite web service**
    – Standards: WSDL (single WS), BPEL (composition process)

• **Long-Standing Challenge for Data Quality**
  – Problems of **data misinterpretation**
    • Different Web services have different assumptions about the interpretation of the exchanged data among them in the composition
    • Root cause: inconsistent data representation, unit, precision, scaling, and meaning, etc.
Example Used in our Demonstration: Illustrating Creation of Composite Web Service

CS: QuoteOpeningPriceWS
Operation: getOpeningPrice

S1: StockIPOWS
Operation: getDateofIPO

S2: HistoricalStockQuoteWS
Operation: getDailyOpenPrice

Get Opening Price on 1st day of Issue (IPO) of a stock

Provide date of 1st day of Issue (IPO)

Provide Stock Price on any specified date

Notes: 1. Assumptions of data interpretation are not explicitly represented in WSDLs.
2. Often parallel asynchronous processing (omitted from example)
Context-enriched Ontology Model

• Ontology with Context Modifiers

<table>
<thead>
<tr>
<th>Service</th>
<th>Context Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>CS, S1</td>
<td>$ctxtUK = {&lt;\text{format}, \text{dd-mm-yyyy}&gt;, &lt;\text{currency}, \text{GBP}&gt;}$</td>
</tr>
<tr>
<td>S2</td>
<td>$ctxtUS = {&lt;\text{format}, \text{mm/dd/yyyy}&gt;, &lt;\text{currency}, \text{USD}&gt;}$</td>
</tr>
</tbody>
</table>
Specification of Conversion Among Modifier Types (generic conversions – not specific to example)

• Conversion Representation

\[
cvt(C, m, ctxt_s, ctxt_t, mvs, mvt, vs, vt)
\]

\[
cvtformat(date, format, ctxtUK, ctxtUS, "dd-mm-yyyy", "mm/dd/yyyy", "09-08-2009", vt)
\]

\[
vt = "08/09/2009"
\]

• Conversion Implementation

– Xpath functions: \( cvtformat \)

\[
Vt = Concat(substring-before( substring-after(Vs,"-"),"-"),"/",
    substring-before(Vs,"-"),"/", substring-after(substring-after(Vs,"-"),"-"))
\]

– External services: \( cvtcurrency \)
  • CurrencyExchangeService
Semantic Annotation of WSDL (using SAWSDL)

<wsdl:definitions targetNamespace="http://stockQuote.coin.mit"
xmlns:stkOntology="http://stockQuote.coin.mit/ontologies/stockOntology#"
xmlns:sawsdl="http://www.w3.org/ns/sawsdl" ... sawsdl:modelReference="stkOntology#ctxtUK">
<wsdl:types>
<schema elementFormDefault="qualified" targetNamespace="http://stockQuote.coin.mit" xmlns="http://www.w3.org/2001/XMLSchema">
<element name="getDateofIPO">
<complexType>
<sequence>
<element name="tickerSymbol" type="xsd:string" sawsdl:modelReference="stkOntology#stockSymbol"/>
</sequence>
</complexType>
</element>
<element name="getDateofIPOResponse">
<complexType>
<sequence>
<element name="getDateofIPOReturn" type="impl:IPOBean"/>
</sequence>
</complexType>
</element>
<complexType name="IPOBean">
<sequence>
<element name="dateofIPO" nillable="true" type="xsd:string" sawsdl:modelReference="stkOntology#date stkOntology#ctxtUK"/>
<element name="tickerSymbol" nillable="true" type="xsd:string" sawsdl:modelReference="stkOntology#stockSymbol"/>
</sequence>
</complexType>
</schema>
</wsdl:type>

Both global and local context specification possible.
Reconciliation Approach for Any Web Service Composition

Generic Ontology/SAWSDL context definitions of web services and Conversion Specifications set up

Our Reconciliation Approach

1. Translating WSDL/BPEL to LOTOS NT
2. Detecting Context Conflicts
3. Incorporating Conversions into LOTOS NT
4. Generating Mediated BPEL
1. Translating WSDL/BPEL to LOTOS NT

• Benefits of LOTOS NT
  – Formalism for verifying composition processes, e.g., deadlock-freeness
  – Independence of any composition languages, e.g., BPEL, OWL-S process

• Translating Static Aspect to Types
  – WSDL: data types, messages, operations
  – BPEL: variables, partner links

```xml
(* @ tickerSymbol:mdlRef="stkOntology#stockSymbol"
 * @ dateofIPO:mdlRef="stkOntology#date stkOntology#ctxtUK"
 *)
type IPOBeanComplexType is
  IPOBeanComplexType (tickerSymbol:string, dateofIPO:string)
end type
```
2. Detecting Context Conflicts

• 2.1 Identify Data Transfers
  – Explicit data transfers: `<assign>` (BPEL), `:=` (LOTOS NT)
  – Implicit data transfers through variables in the process

Implicit data transfers refer to the situations that a piece of data is written to a variable and then directly read from the variable, without assignments between the write and read operations.
2. Detecting Context Conflicts

- Implicit Data Transfers

QuoteOpeningPricePL(\(?\)quoteOpeningPriceRole
\(\langle\)invoke\(\rangle\)
\(\langle\)invoke\(\rangle\)
\(\langle\)assign\(\rangle\)
\(\langle\)reply\(\rangle\)
\(\langle\)receive\(\rangle\)
2. Detecting Context Conflicts

• 2.2 Examine Each Data Transfer for Conflicts

Typical Scenario of Context Conflicts within a Data Transfer

ctxtUK={“dd-mm-yyyy”}  ctxtUS={“mm/dd/yyyy”}
2. Detecting Context Conflicts

- **Detected Context Conflicts**

<table>
<thead>
<tr>
<th>Source</th>
<th>Target</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOTOS NT</td>
<td>dateofIPO</td>
</tr>
<tr>
<td>Web service</td>
<td>S1</td>
</tr>
<tr>
<td>Context</td>
<td>ctxtUK</td>
</tr>
<tr>
<td>date.format</td>
<td>“dd-mm-yyyy”</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Source</th>
<th>Target</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOTOS NT</td>
<td>getDailyOpenPriceResponse</td>
</tr>
<tr>
<td>Web service</td>
<td>S2</td>
</tr>
<tr>
<td>Context</td>
<td>ctxtUS</td>
</tr>
<tr>
<td>moneyValue.currency</td>
<td>“USD”</td>
</tr>
</tbody>
</table>
3. Incorporating Conversions into LOTOS NT

• 3.1 Make implicit data transfers with conflicts explicit

3. Incorporating Conversions into LOTOS NT


var getOpeningPriceResponse

3. Incorporating Conversions into LOTOS NT

• 3.2 Incorporate conversions and generate mediated process

\[S1: \text{StockIPOWS} \quad \text{Operation: getDateofIPO}\]

\[\text{Input: tickerSymbol} \quad \text{Output: dateofIPO, tickerSymbol}\]

\[S2: \text{StockQuoteWS} \quad \text{Operation: getOpenPrice}\]

\[\text{Input: dateofQuote, tickerSymbol} \quad \text{Output: openPrice}\]

\[ES: \text{CurrencyConvertorWS} \quad \text{Operation: convertCurrency}\]

\[\text{Input: priceInUSD} \quad \text{Output: PriceInGBP}\]

\[CS: \text{QuoteofOpeningWS} \quad \text{Operation: getOpeningPrice}\]

\[\text{Input: tickerSymbol} \quad \text{Output: openingPrice}\]
Prototype: CMT – Illustrating Conflict Detection

Number of Detected Context Conflicts: 2

No. 1

service: StockIPO vs. HistoricalStockQuote
element: dateofIPO vs. dateofQuote
data transfer type: Explicit
concept: date vs. date
context: ctxUK vs. ctxUS
modifier: format
modifier value: dd-mm-yyyy vs. mm/dd/yyyy
conversion: Vt = Concat(substring-before(substring-after(Vs,""),""),"","",substring-before(Vs,""),""),substring-after(substring-after(Vs,""),""))

No. 2

service: HistoricalStockQuote vs. QuoteofOpening
element: getDailyOpenPriceReturn vs. getOpeningPriceReturn
data transfer type: Implicit
concept: stockPrice vs. stockPrice
context: ctxUS vs. ctxUK
modifier: currency
modifier value: USD vs. GBP
conversion:
<wsdloperation name="cvtOP">
  <wsdl:input message="msgType_s" name="msgName_s"/>
  <wsdl:output message="msgType_t" name="msgName_t"/>
</wsdl:operation>
Summary and Future Work

• **Strengths**
  – Address data misinterpretation problems to improve quality
  – Separation of concern between service descriptions (e.g., BPEL) and ontology/context descriptions
  – Use the standard-compliant method (i.e., SAWSDL) for semantic annotation
  – Use LOTOS, a kind of process algebra, which can be integrated with behavioral compatibility verification
  – Automates the incorporation of conversions into composition (can assemble complex conversions)

• **Ongoing Work**
  – Extend to deal with semantic heterogeneity at both schematic and data instance level
  – (Semi-)automatic construction of ontology, contexts and annotations
Maybe Web Services were in use during the Napoleonic Wars? *(Data Quality Consequences)*

In 1805, the Austrian and Russian Emperors agreed to join forces against Napoleon. The Russians promised that their forces would be in the field in Bavaria by **Oct. 20**.

The Austrian staff planned its campaign based on that date in the **Gregorian calendar**. Russia, however, still used the ancient **Julian calendar**, which lagged 10 days behind.

The calendar difference allowed Napoleon to surround Austrian General Mack's army at Ulm and force its surrender on Oct. 21, well before the Russian forces could reach him, ultimately setting the stage for Austerlitz.

Thanks again for your interest in this work. Please refer to the paper for more details.

Any questions?
Backup Slides
1. Translating WSDL/BPEL to LOTOS NT

- **Translating Dynamic Aspect to Process**

<table>
<thead>
<tr>
<th>BPEL</th>
<th>LOTOS NT</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>&lt;receive variable=&quot;v&quot; .../&gt;</code></td>
<td><code>(c?v)</code></td>
</tr>
<tr>
<td><code>&lt;reply variable=&quot;v&quot; .../&gt;</code></td>
<td><code>(c!v)</code></td>
</tr>
<tr>
<td><code>&lt;invoke inputVariable=&quot;v1&quot;</code></td>
<td><code>(c!v1 ; c?v2)</code></td>
</tr>
<tr>
<td><code>outputVariable=&quot;v2&quot;.../&gt;</code></td>
<td></td>
</tr>
<tr>
<td><code>&lt;assign ...&gt;</code></td>
<td></td>
</tr>
<tr>
<td><code>&lt;copy&gt;</code></td>
<td><code>(v2 := v1; v4 := v3)</code></td>
</tr>
<tr>
<td><code>&lt;from variable=&quot;v1&quot;</code></td>
<td></td>
</tr>
<tr>
<td><code>&lt;to variable=&quot;v2&quot;/&gt;</code></td>
<td></td>
</tr>
<tr>
<td><code>&lt;/copy&gt;</code></td>
<td></td>
</tr>
<tr>
<td><code>&lt;copy&gt;</code></td>
<td></td>
</tr>
<tr>
<td><code>&lt;from variable=&quot;v3&quot;</code></td>
<td></td>
</tr>
<tr>
<td><code>&lt;to variable=&quot;v4&quot;/&gt;</code></td>
<td></td>
</tr>
<tr>
<td><code>&lt;/copy&gt;</code></td>
<td></td>
</tr>
<tr>
<td><code>&lt;sequence ...&gt;</code></td>
<td><em>(action1 ; action2)</em></td>
</tr>
<tr>
<td><code>&lt;...activity1.../&gt;</code></td>
<td></td>
</tr>
<tr>
<td><code>&lt;...activity2.../&gt;</code></td>
<td></td>
</tr>
<tr>
<td><code>&lt;/sequence&gt;</code></td>
<td></td>
</tr>
</tbody>
</table>
1. Translating WSDL/BPEL to LOTOS NT

```plaintext
QuoteOpeningPricePL(?quoteOpeningPriceRole
(getOpeningPriceInput(getOpeningPriceRequestMessage
(getDateofIPO)))) ;
StockIPOPL(!stockIPOServiceRole(getDateofIPOInput
(getDateofIPORequestMessage(getDateofIPOType
(getDateofIPO)))))) ;
StockIPOPL(?stockIPOServiceRole(getDateofIPOOutput
(getDateofIPOResponseMessage(getDateofIPOResponse)))) ;
getDailyOpenPrice := getDailyOpenPriceType (getDateofIPOResponse.getDateofIPOReturn.tickerSymbol,
getDateofIPOResponse.getDateofIPOReturn.dateofIPO) ;
HistoricalStockQuotePL(!historicalStockQuoteServiceRole
(getDailyOpenPriceInput(getDailyOpenPriceRequestMessage (getDailyOpenPrice))) ) ;
HistoricalStockQuotePL(?historicalStockQuoteServiceRole
(getDailyOpenPriceOutput(getDailyOpenPriceResponseMessage (getDailyOpenPriceResponse)))) ;
QuoteOpeningPricePL(!quoteOpeningPriceRole
(getOpeningPriceOutput(getOpeningPriceResponseMessage (getDailyOpenPriceResponse))))
```