Enabling Business Experts to Discover Web Services for Business Process Automation

Emerging Web Service Technologies

Jan-Felix Schwarz

3 December 2009
Agenda

- Problem & Background
- Approach
- Evaluation
- Future Work
- Summary
Agenda

- **Problem & Background**
  - Business Process Automation
  - The Gap Between Business & IT
  - Theoretical Foundations of Service Discovery
- Approach
- Evaluation
- Future Work
- Summary
Enabling Business Experts to Discover Web Services ... | Jan-Felix Schwarz | 3 Dec 2009
How can a business expert discover web services while designing a business process?
Approaches for Service Discovery

1. Structural
   - Use syntactical information (e.g. operations, messages)
   - Very technical

2. Lexical
   - Use natural language descriptions (e.g. operation names, documentation descriptions)
   - NLP, lexical databases

3. Semantic
   - Use ontologies to describe capabilities and properties
   - Semantic Web Services
Most discovery algorithms combine different approaches to achieve a better result.

1. **Mixed strategy**

2. **Cascading strategy**

3. **Switching strategy**
Major Phases of Web Service Discovery

- Matching Phase
  - All Web Services
  - Service Request
  - Criteria Set
- Assessment Phase
  - Matching Web Services
- Selection Phase
  - Assessed Web Service
  - Selected Web Service

[Kokash et al.]
Agenda

- Problem & Background
- **Approach**
  - Overview of the Solution
  - Structural Matching
  - Semantic Matching
  - Assessment & Selection
- Evaluation
- Future Work
- Summary
Solution

- structural matching
- lightweight semantic matching

mixed strategy

manual lexical matching
Structural Matching

Enabling Business Experts to Discover Web Services... | Jan-Felix Schwarz | 3 Dec 2009
Structural Matching

From Business Objects to Message Types

- Business objects are on a conceptual modeling level
- Web Services use message types to define their input and output
- Mapping from business objects to technical data structures needed

[Diagram showing the relationship between business objects and message types: customer -> address ->-xsd:string, xsd:integer]
Structural Matching

1. Business Object
   - Input
   - Business Activity
   - Business Object
   - Output

2. Traverse Mapping
   - Abstract
   - Customer B.O.
   - Address B.O.
   - Data Type (XSD)
   - xsd:String

3. Match message types with operation parameters (Input/Output)
   - Message Type
   - Input
   - Output

4. Extract matching operation for each matched parameter (Input/Output)
   - Operation
   - Parameter
   - Operation
   - Parameter
   - Operation

5. Get owning service for each operation
   - Service
   - Input
   - Output

6. Match message types with operation parameters (Input/Output)
   - Service
Assessment & Selection

- Unite results of both matching algorithms (→ mixed strategy)
- User assesses results based on:
  - Service name and description
  - Supported business objects
  - Other contexts in which the service is used
- Finally selects one service
Agenda

- Problem & Background
- Approach
- **Evaluation**
  - Matching: Structural vs. Semantic
  - Strength & Weaknesses
- Future Work
- Summary
Matching: Structural vs. Semantic

**Structural Matching**
- Fascinates technical oriented users
- Requires an information architecture

**Semantic Matching**
- Easy to understand for non-technical users
- Requires a taxonomy
- Imported services need to be tagged
Strengths & Weaknesses

**Strengths**

- Involves business experts in process implementation
- Leads to a better common understanding between business and IT
- Alternative matching approaches for improved acceptance

**Weaknesses**

- Only works for a managed repository of web services
- Requires that users stick precisely to the taxonomy
Agenda

- Problem & Background
- Approach
- Evaluation
- Future Work
- Summary
Future work proposed in the paper

- Improve performance & scalability
- Allow more sophisticated taxonomies
- Generalize approach to support any kind of service
- Conduct a user study
Possible Enhancements

Make matching more flexible

- Suggest services that require additional inputs / outputs
  → Modeling auto-completion
Agenda

- Problem & Background
- Approach
- Evaluation
- Future Work
- **Summary**
Summary

- Bridge gap between business and IT by letting business experts bind business process activities to web services
- Two alternative matching approaches for service discovery
  - Structural matching based on a mapping between business object and message types
  - Lightweight semantic matching using tags
- Integrated in ARIS SOA Architect

Questions?