Evaluating the Compatibility of Conversational Service Interactions

Sam Guinea and Paola Spoletini
Agenda

• Introduction
• Interaction Sequence Charts
  – Describing BPEL interactions
• Compatibility Evaluation
• The Future of ISC
Introduction

- Open World Assumption
  - System depends on the functional and qualitative evolution of partner services
- Self-adaptation is a way to cope with these evolutions
  - Substitute a service with a compatible alternative
- Conversational services impose:
  - a specific interaction protocol and
  - specific data types
Innovative Contributions

• Interaction Sequence Charts
  – derived from Message Sequence Charts
  – describe an isolated service’s interaction with the outside world (protocol and data types)

• Compatibility Evaluation Algorithm
  – uses ISCs to determine protocol compatibility
  – uses Fuzzy techniques for data types (FXPath)
  – provides a report indicating where the mismatches are

Tool support:
Graphical Design-time tool
Automatic ISC extraction for BPEL 2.0 processes

Tool support:
Algorithm fully implemented
FXPath manually integrated
Approach overview

Data type adapter
Process adaptation
Interaction Sequence Charts

- Seven basic constructs
  - Receive
  - Send
  - Or
  - And
  - Loop
  - Parallel Loop
  - Terminate

Partner name
Data type
Importance level
Preference
Expressing BPEL Interactions

Sequence  
Receive - Invoke - Reply

Switch  
If else if  
While - RepeatUntil - For Each (sequential)

Flow  
ForEach (parallel)  
Pick

Event handler

Fault handler

Compensation handler
Compatibility evaluation

• Measures:
  – Structural coherence
  – Data type similarity

• Algorithm structure
  1. ISC pruning: isolate a partner’s interactions
  2. Graph construction: build a minimal structure
  3. Iterative comparison:
     a. Graph structure compatibility
     b. Fuzzy analysis of data similarities
Isolating P’s interactions

- Delete basic interactions that do not involve P
- Remove sequential structures that do not contain basic interactions
- Omit And branches that contain no interactions
- Reduce And structures with a single branch to a path
- Collapse Or branches into one (highest preference level)

Flatten out Or structures, and generate all the possible combinations of Or branches
Graph construction

intPoint\textsubscript{S} \rightarrow \text{intPoint}_{C_1} \rightarrow \text{initialAnd} \rightarrow \text{initialBranch}_1 \rightarrow \text{lastBranch}_1 \rightarrow \text{finalAnd} \rightarrow \text{finalLoop}

intPoint\textsubscript{S} \leftarrow \text{intPoint}_{C_1} \rightarrow \text{initialBranch}_n \rightarrow \text{lastBranch}_n \rightarrow \text{finalLoop}
Graph construction

- The nodes are enumerated following the temporal sequence and a specific ordering of And branches
- The degree of importance is spread onto the structural arcs
- Redundant nodes are removed
- The candidate’s graph is built using the same procedure but reversing the direction of interaction edges
Analysis procedure

a. Structural comparison
   - Graphs with the same size have to be identical
   - If the sizes do not match, nodes that are not strictly necessary can be deleted
Tele-Radiology
Tele-Radiology

Privacy => minimize the number of partners managing sensitive data

Get rid of partner EM

Strong vs. weak interpretation for Loops

Data analysis
The Future of ISC

• Continue to validate the algorithm
• Close the loop
  – Process adaptation and Data type adapters
• Extend beyond BPEL

• Refine ISCs with properties inferred from history of interactions
  – Refined meaning of compatibility with actual partner
  – What am I looking for in a new partner?