Modelling Sound Conflict Management for Virtual-Enterprise Collaboration

Alex Norta, PhD., Department of Informatics, TTÜ@IEEE SCC'14, Anchorage, Alaska, USA
Agenda

- Introduction
  - Virtual Enterprise (VE) status quo
  - Gap and research questions
  - Method

- Context of VE
  - Running example
  - Conflict Scenario

- VE-architecture
  - Architecture Overview
  - Business Rule Modelling

- Conflict-management approach
  - Exception Detection & Classification
  - Conflict Classification & Resolution

- Evaluation
  - VE-Collaboration Specification
  - Conflict-Resolution Protocol

- Conclusion and future work
Introduction

- New enabling concepts and technologies
  - service-oriented cloud computing (SOCC)
  - Business Process as a Service (BPaaS)
  - Cross-organizational eSourcing framework
Introduction

- **Gap:** Conflict management requires
  - a governance structure
  - to ensure soundness of the collaboration,
  - even when reconciliation triggers changes in service interfaces

- **Research question:** How to develop a VE service ecosystem with conflict management- and resolution mechanisms?
  - What are the key features of a VE service ecosystem?
  - What are the specifics of business rules that guarantee soundness of the VE?
  - What conflict types exist in a VE and what are the appropriate conflict resolution approaches?
Context of VE-collaboration

- Running example:

Context of VE-collaboration

Goal model: left/top → right/bottom
Context of VE-collaboration

- VE-architecture
Context of VE-collaboration

- Monitoring package.
## Context of VE-collaboration

- Semantics of some [Declare](https://www.declare.org/) templates

<table>
<thead>
<tr>
<th>Constraint</th>
<th>LTL semantics</th>
<th>Constraint</th>
<th>LTL semantics</th>
</tr>
</thead>
<tbody>
<tr>
<td>responded existence</td>
<td>◊A → ◊B</td>
<td>co-existence</td>
<td>◊A ↔ ◊B</td>
</tr>
<tr>
<td>response</td>
<td>□(A → ◊B)</td>
<td>precedence</td>
<td>¬B ⋀ A</td>
</tr>
<tr>
<td>alternate response</td>
<td>□(A → ◌(¬A ∨ B))</td>
<td>alternate precedence</td>
<td>(¬B ⋀ A) ∧ □(B → ◌(¬B ⋀ A))</td>
</tr>
<tr>
<td>chain response</td>
<td>□(A → ◌B)</td>
<td>chain precedence</td>
<td>□(◎B → A)</td>
</tr>
<tr>
<td>not co-existence</td>
<td>◊A → ¬◊B</td>
<td>not succession</td>
<td>□(A → ¬◊B)</td>
</tr>
</tbody>
</table>

---

**Declarative workflows:**
Balancing between flexibility and support

W. M. P. van der Aalst, M. Pesic, H. Schonenberg
Context of VE-collaboration

- **Business rules for**: \( \forall x. \Box((A \land \Phi(x)) \rightarrow \Diamond \exists y. (B \land \Psi(x, y))) \)
- **Rule R1**: 
  - “If order amount for any shipment is greater than $1 million, choose the lowest cost shipper; else choose Ship”
- **Rule R2**: 
  - “Choose Ship for all shipments”
- **Rule R3**: 
  - “If order amount is between $100,000 and $250,000, choose Ship; else choose the lowest cost shipper”

**Ship cases**

- for R1: \( \Box((\text{ReceiveOrder} \land (\text{amount} > \$1 \text{ million})) \rightarrow \Diamond \text{ChooseShip}) \),
- for R2: \( \Box(\text{ReceiveOrder} \rightarrow \Diamond \text{ChooseShip}) \),
- for R3: \( \Box((\text{ReceiveOrder} \land (\text{amount} \geq \$100,000) \land (\text{amount} \leq \$250,000)) \rightarrow \Diamond \text{ChooseShip}) \).

**Low-cost shipper**

- for R1: \( \Box((\text{ReceiveOrder} \land (\text{amount} \leq \$1 \text{ million})) \rightarrow \Diamond \text{ChooseLowestCostShipper}) \),
- for R3: \( \Box((\text{ReceiveOrder} \land (\text{amount} < \$100,000) \lor (\text{amount} > \$250,000)) \rightarrow \Diamond \text{ChooseLowestCostShipper}) \).

- \( \Diamond \text{ChooseLowestCostShipper} \rightarrow \neg \Diamond \text{ChooseShip} \)
Context of VE-collaboration

R1  R2  R3

100t

250t

1 mill

ship

low cost
Conflict Management Approach

- Conflict management lifecycle

![Conflict Management Lifecycle Diagram]
## Conflict Management Approach

### Conflict types

<table>
<thead>
<tr>
<th>Local</th>
<th>Service</th>
<th>Party</th>
<th>Global</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>disruptive</strong></td>
<td></td>
<td></td>
<td>eCommunity</td>
</tr>
<tr>
<td>rule</td>
<td>service</td>
<td>party</td>
<td></td>
</tr>
<tr>
<td><strong>non-disruptive</strong></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>x</td>
<td>x</td>
<td>x</td>
<td></td>
</tr>
<tr>
<td>effect</td>
<td>epoch change</td>
<td></td>
<td>termination</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>ignore</td>
</tr>
</tbody>
</table>
Conflict Management Approach

- Epoch-change types: [10.1109/EDOC.2012.14](10.1109/EDOC.2012.14)
Evaluation

- Exemplary mapping of rule R3 to StandardML-statement

![Diagram]

http://cpntools.org/

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>p</td>
<td>Process Instantiation ID</td>
</tr>
<tr>
<td>b</td>
<td>bill of SupTr</td>
</tr>
<tr>
<td>c</td>
<td>cost</td>
</tr>
<tr>
<td>ts</td>
<td>tyre specification with price</td>
</tr>
<tr>
<td>ws</td>
<td>steering wheel specification with price</td>
</tr>
<tr>
<td>trc</td>
<td>tyre prices and b aggregated</td>
</tr>
<tr>
<td>ag</td>
<td>aggregated prices of steering wheels, tyres, bills</td>
</tr>
<tr>
<td>sb</td>
<td>bill of wheel supplier</td>
</tr>
<tr>
<td>s</td>
<td>shipping specifications</td>
</tr>
<tr>
<td>a</td>
<td>delivery address</td>
</tr>
<tr>
<td>shcm{1</td>
<td>2}</td>
</tr>
<tr>
<td>shtr{1}</td>
<td>shipper choice of tyres manufacturer</td>
</tr>
<tr>
<td>shst{1}</td>
<td>shipper choice of steering wheels manufacturer</td>
</tr>
<tr>
<td>wse</td>
<td>extra specification details</td>
</tr>
</tbody>
</table>
Evaluation

- Conflict-management by agents of collaborating parties
Conclusion

- How to develop a VE service ecosystem with conflict management- and resolution mechanisms

Sub-questions

- Key features of a VE service ecosystem?
  - External-level process-view matching
  - Conceptual-level larger imperative processes
  - Orchestration of legacy systems
  - System replication for each party

- Business rules specifics?
  - Declare statements with Linear Temporal Logics
  - Event-Condition-Action rules for conflict detection, e.g., LTL rules conflicts
  - Exception classifications

- Conflict types and conflict resolutions?
  - Disruptive when conflict so severe that collaboration collapses
  - Non-disruptive when conflict can be reconciled

Future work

- Prototype and evaluating our approach
- Integrate negotiation-based automated conflict resolution strategies
Thank you for listening!

Q&A

Study IT in .ee

sponsored by Skype™