Using Reo for Compliance-driven Design of Service-Oriented Applications
Introduction

COMPAS overview
- Motivation and goals
- Partners, work packages and case studies
- WP3: Behavioral models for service compositions and their verification

Business Process Modeling
- Mapping BPMN to Reo – events, gateways, message exchange, exception handling, compensation activities, transactions

Compliancne Modeling
- Expressing compliance constraints using Reo – some ideas and examples

Related Work

Conclusions and Future Work
COMPAS goals

🎯 COMPAS = Compliance-driven Models, Languages, and Architectures for Services

🌐 http://www.compas-ict.eu/

🔍 Ensure dynamic and on-going compliance of software services to business regulations and user requirements

🔍 Help organizations to develop business compliance solutions easier and faster

🔍 Use model-driven techniques, domain-specific languages, and service-oriented computing
Compliance is

- a multi-faceted concept that encompasses the capability of an organization to meet requirements coming from
  - Regulatory/legislative documents
    - Basel II, Sarbanes-Oxley, IFRS, MiFID, LSF, HIPAA, Tabaksblat, etc.
  - Internal movements towards quality of service
  - User-defined constraints…

- a state of “adherence of one set of rules (source rules) against another set of rules (target rules)”

- a process, which is about “ensuring that business processes, operations and practice are in accordance with a prescribed set of norms”
COMPAS lifecycle
COMPAS partners
## COMPAS

### work packages

<table>
<thead>
<tr>
<th>WP</th>
<th>Description</th>
<th>Leaders</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Core concepts and model-driven compliance framework</td>
<td>VITALAB, CWI, Universität Stuttgart</td>
</tr>
<tr>
<td>2</td>
<td>Expressive languages for compliance concerns</td>
<td>Tilberg University, Telcordia, CWI</td>
</tr>
<tr>
<td>3</td>
<td>Behavioral models for service compositions and their verification</td>
<td>CWI, LIRIS</td>
</tr>
<tr>
<td>4</td>
<td>Reusable business process fragments augmented with compliance concerns</td>
<td>Universität Stuttgart, VITALAB</td>
</tr>
<tr>
<td>5</td>
<td>Compliance governance</td>
<td>LIRIS, Universität Trento</td>
</tr>
<tr>
<td>6</td>
<td>Exploitation, dissemination, standardization</td>
<td>Thales, PricewaterhouseCoopers</td>
</tr>
<tr>
<td>7</td>
<td>Validation of the results through case studies</td>
<td>COMPAS, CWI</td>
</tr>
<tr>
<td>8</td>
<td>Project management</td>
<td>VITALAB</td>
</tr>
</tbody>
</table>

17/06/2008   ACG Meeting, CWI, Amsterdam
Case studies
 (> 5 scenarios)

Telcordia

- Unified Incident Command System
  - Coordination compliance:
    - Role/task assignment to involved organizations
    - Obligations to add/remove activities, controls or data
    - Constraints on activities and corresponding service ordering and data flow
    - Time constraints
  - Information exchange compliance
  - Access control compliance
  - Quality compliance
    - Dynamic (runtime) process and service adaptation to meet process metrics
  - Resource compliance

- Advanced Telecom Services
  - Contract and legal compliance
  - Licensing compliance…

THALES

- Loan origination process
  - Focus on security
WP3 (led by CWI)

<table>
<thead>
<tr>
<th>Task</th>
<th>Description</th>
<th>CWI expected contribution</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Formal behavioural models for services</td>
<td>Develop a (Reo / constraint automata based) model for composite services able to support multi-partner processes with compliance concerns</td>
</tr>
<tr>
<td>2</td>
<td>Visual environment for service description</td>
<td>Develop a graphical language and a GUI editor in Eclipse for specification of service interfaces.</td>
</tr>
<tr>
<td>3</td>
<td>Tools for service description verification</td>
<td>Develop a suitable tool</td>
</tr>
<tr>
<td>4</td>
<td>Tools for service compliance behavior management</td>
<td>Develop a suitable tool</td>
</tr>
<tr>
<td>5</td>
<td>Service model interpreter/simulation engine</td>
<td>Develop a suitable interpretor engine</td>
</tr>
</tbody>
</table>
WP3 initial work plan

**Business Process Lifecycle**
- Graphical Modeling Tools (GMT) (BPMN, UML ADs, BPEL)
- Reo/Constraint Automata Modeling Tools
- BPEL, WS-CDL, WSDL

**Compliance Concerns**
- DSLs, GMT extensions
- Constraints, Temporal Logic Formulae, Automata
- Web Services, WS-Policies, XACML, etc.

Model checking, refinement

Modeling

Implementation
BPMN

EVENT
- Start
- Intermediate
- End

ACTIVITY
- Task
- Sub-process

MACROS
- Activity looping
- Multiple instances

GATEWAY
- Parallel fork
- Parallel join
- XOR Decision
- OR Decision
- OR Merge

SEQUENCE FLOW
- Normal
- Exception

MESSAGE FLOW
- Conditional
- Default

ADVANCED CONSTRUCTS
- Compensation association
- Transaction
Purchase-to-pay: process model

[Sadiq et al, BPM'07]
Reo

- FIFO1 channel
- synchronous channel
- lossy synchronous channel
- filter channel
- P-producer
- \( \leq \tau \)
- synchronous drain
- asynchronous drain
- synchronous spout
- asynchronous spout
- timer channel

Exclusive choice (deferred XOR)

Valve connector: controls flow from A to B

Exclusive choice with priority

open C (close B)
BPMN2Reo: basic gateways

- XOR merge
- Parallel fork
- Parallel join
- Complex Merge (e.g., m out of n)

Data-based XOR decision

- $g_1$
- $g_2$

Event-based XOR decision

- $\leq \tau$

17/06/2008    ACG Meeting, CWI, Amsterdam
BPMN2Reo: events and messages

Message

Complex

Valve

Send order

Receive order

Synchronous message exchange

Asynchronous message exchange

Outgoing messages

Blocking

Non-blocking lossy

Non-blocking waiting
Purchase-to-pay: Reo model
BPMN2Reo: exception handling in sequential processes

Sequential atomic tasks

Sequential sub-processes
BPMN2Reo: exception handling in parallel processes (1)

Parallel atomic tasks
BPMN2Reo: exception handling in parallel processes (2)

Parallel sub-processes
Reo2BPMN: compensation activities
BPMN2Reo: transaction sub-processes
Reo2BPMN: transaction with hazard

![Diagram of Reo2BPMN transaction with hazard]

17/06/2008 ACG Meeting, CWI, Amsterdam
BPMN Macros

- Number of activities may depend on data flow
- Run-time reconfiguration by graph transformation

Activity looping

Multiple instances

Cancel sub-process T

Exception in sub-process T
Workflow analysis

[Diagram of workflow processes]

- The process may be completed incorrectly: If credit card checking fails the products remain prepared.

- The process never completes.

- The process contains unreachable activities.

[Dijkman et al., 2008]
Synchronization of concurrent flows

Constraint: $A_4$ must be executed after $A_3$
Using Reo for compliance modeling

[Sadiq et al, BPM’07]

<table>
<thead>
<tr>
<th>Category</th>
<th>Description</th>
<th>Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>Flow</td>
<td>Impact on BP activities and their coordination</td>
<td>Reo circuits, connector reconfiguration, graph transformations</td>
</tr>
<tr>
<td>Data</td>
<td>Constraints on data retention and transfer</td>
<td>Constraint automata operating on special data domains (e.g., privacy)</td>
</tr>
<tr>
<td>Resource</td>
<td>Controls related to the access and management of resources</td>
<td>Resource-sensitive CA [Meng &amp; Arbab, FMOODS’07], Quantitative CA</td>
</tr>
<tr>
<td></td>
<td></td>
<td>[Arbab et al., Coordination’07]</td>
</tr>
<tr>
<td>Time</td>
<td>Time constraints (activity start-ups, deadlines, durations, etc.)</td>
<td>Timed CA [Arbab et al., JSSM’07]</td>
</tr>
</tbody>
</table>
Purchase-to-pay: compliance concerns

- Creation and approval of purchase requests must be undertaken by 2 different purchase officers

1. Supplier can be charged a penalty if a customer did not receive goods within $k$ days after goods shipment notice
2. Purchase requests which are not closed within $k$ days should raise an alert to purchasing manager
Related work on BPMN analysis


Related work on compliance checking

- Liu, Y., Muller, S., Xu, K.: A static compliance-checking framework for business process models, pp. 335-361

- Many works dedicated to particular areas as privacy, trust, security, QoS
- ISO 27001:2005, ISO 27002:2007, SOX, CobiT (e.g., firewall placement and secure connection)
Related work on transaction modelling


Conclusions

Reo can be used as a language for business process modeling

Expressive

- Capable of expressing complex BPMN constructs
- There exists a repository of Reo workflow patterns

Compositional

- The structure of BPMN diagrams is preserved – important for manual process refinement

Reo mathematical abstractions can be useful for compliance-aware process development

- Time-, data-, resources-, QoS-related constraints
Future work

Implementation

- Converters from graphical models to Reo/CA and from Reo/CA to executable models (EMF/BPEL/Java)

Languages for expressing compliance concerns

- Expressiveness (e.g., must deal with time constraints, parameterized roles, permissions/obligations/violations)
- Formal reasoning

Transaction modeling

- Various compensation strategies
  - compensation activity ordering,
  - compensation in tricky patterns such as discriminator,
  - definition of compensation scope,
  - nested transactions, etc.
- Selective/alternative compensation (e.g., some activities must be compensated while others may remain uncompensated depending on the nature of the exception)
- Exception handling (failures of compensation activities, default exception handlers, etc.)