From Compliant Business Process Specifications to Code

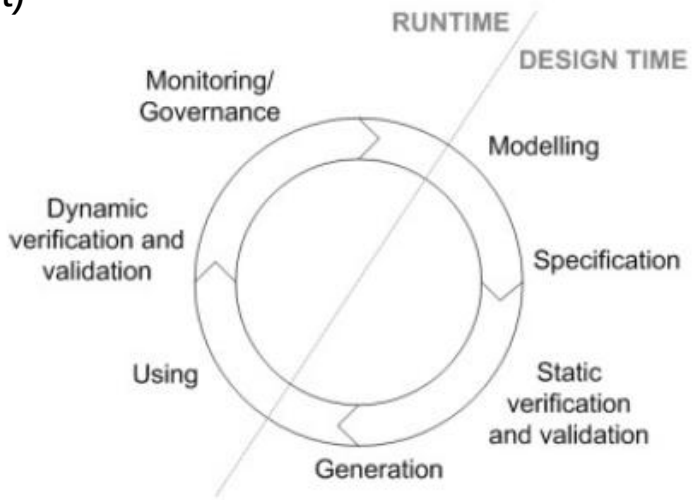
Natallia Kokash

Introduction

- COMPAS project overview
- Compliance requirements
  - Definition, categories, specification formats
- Business process modeling and formalization of compliance requirements
  - Graphical notations + informal annotations
  - Formal process models + formally-specified compliance rules
- Demo: compliance-aware business process development using ECT
- Future work

COMPAS project

- COMPAS = Compliance-driven Models, Languages, and Architectures for Services http://www.compas-ict.eu/
- Goals:
  - *Ensure dynamic and on-going compliance of software services to business regulations and user requirements*
  - *Help organizations to develope business compliance solutions easier and faster*
- Directions:
  - Infrastructure (SOA, model-driven development)
  - Domain Specification Languages (DSLs) and tools for describing compliance requirements
  - Repository of reusable process fragments, request languages
  - Formal models for process/service description, process fragments composition, automated analysis (design time compliance)
  - Monitoring tools, logs mining, dashboard (runtime compliance)



Compliance requirements

- Any explicitly stated rule or regulation that prescribes any aspect of an internal or cross-organizational business process
- Sources of compliance requirements:
  - Internal polices (e.g., technical instructions, regulations aimed at improving Quality-of-Service (QoS))
  - External polices (e.g., privacy regulations, fraud prevention acts, laws)
  - Contracts and mutually acceptable agreements (e.g., Service Level Agreements (SLAs))
- Compliance policy is a logical grouping of a set of coherent rules that realizes a specific goal (e.g., data access control for fraud prevention).

Specification of compliance requirements

1. Goals: “To be compliant with SOX and/or BASELII”
2. Policies: “Investment process - segregation or duties”
3. Rules: Formally specified compliance rules like “investment and authorization operations must be performed by different people”

- Logic-based approaches
  - First-Order Logic [DF07, HW03]
  - LTL [LMX07]
  - CTL [MDK+03, KTK02]
  - Deontic logic [SGN06, CCD+07]
  - Temporal deontic assignments [GV06]
  - Concurrent transaction logic [MDK03]

- Particular compliance categories
  - Control flow and temporal constraints [GMS06, GK07]
  - Security requirements [BCC+07]
  - Privacy policies [BDM+06, HBP07, MBS+08]

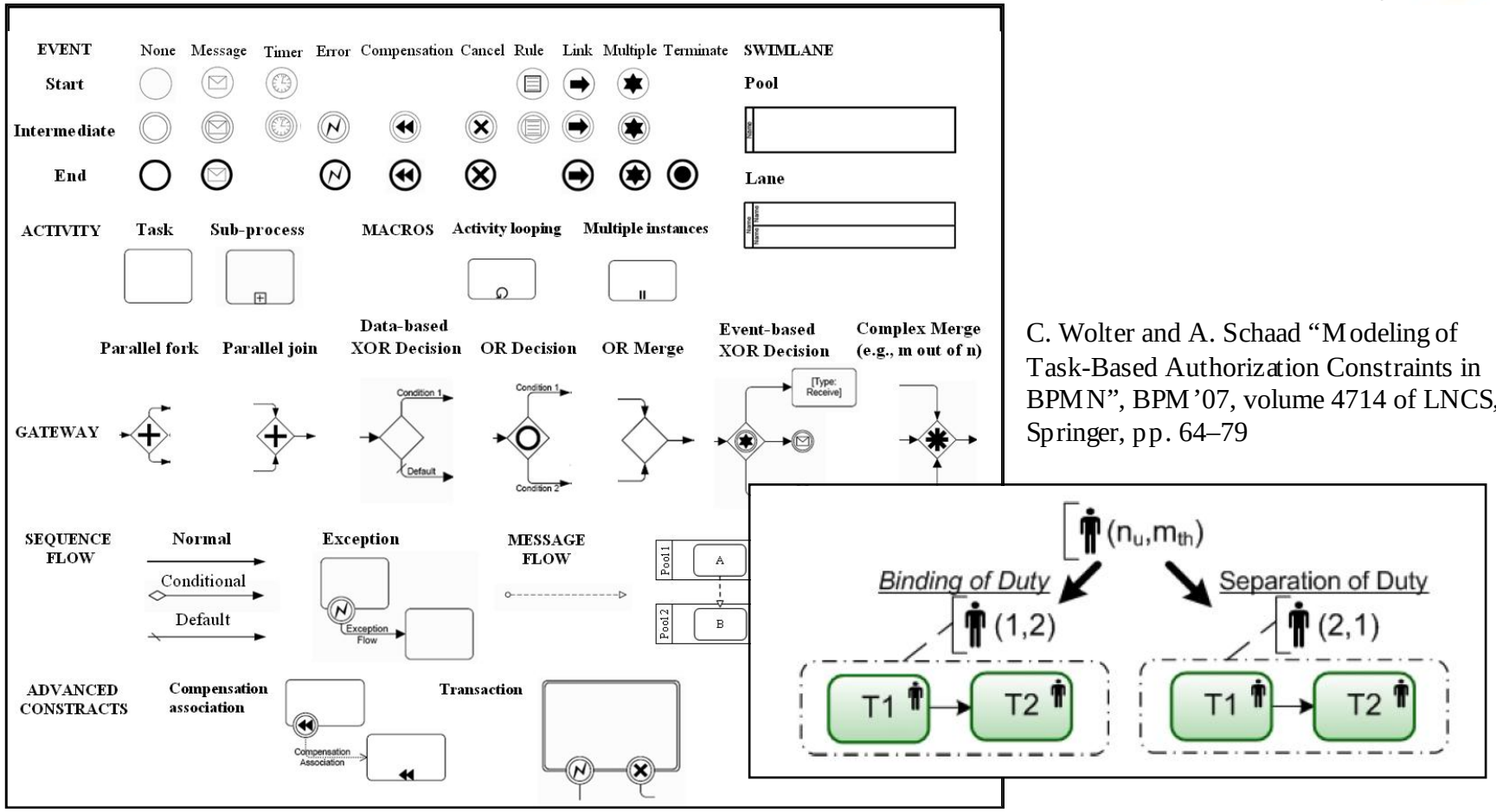
Formal specification of compliance requirements

- Internet Reseller Scenario:
  - R1: Computer-generated sales order confirmations are sent to customers for order acknowledgement at the end of each day or on the next working day.
  - R2: Appropriate segregation of duties should be maintained. Specifically whether the *credit*, *shipping* and *invoicing* functions are segregated from accounts receivable, general ledger and cash functions.
- LTL (COMPAS Deliverable 2.2):
  - R1:
    - G (Action = SalesOrder & paralist = {y, x} à F ((Action = SendCustomerConf & paralist = {c, x, k}) v (Action = SendCustomerConf & paralist = {c, x, 24})))
  - R2:
    - R2.1: G(Action = credit& paralist = {x} à G (! Action = cash & paralist = {x}))
    - R2.2: G(Action = shipping& paralist = {x} à G (! Action = cash & paralist = {x}))
    - R2.3: G(Action = invoicing & paralist = {x} à G (! Action = cash & paralist = {x}))

Business process modeling

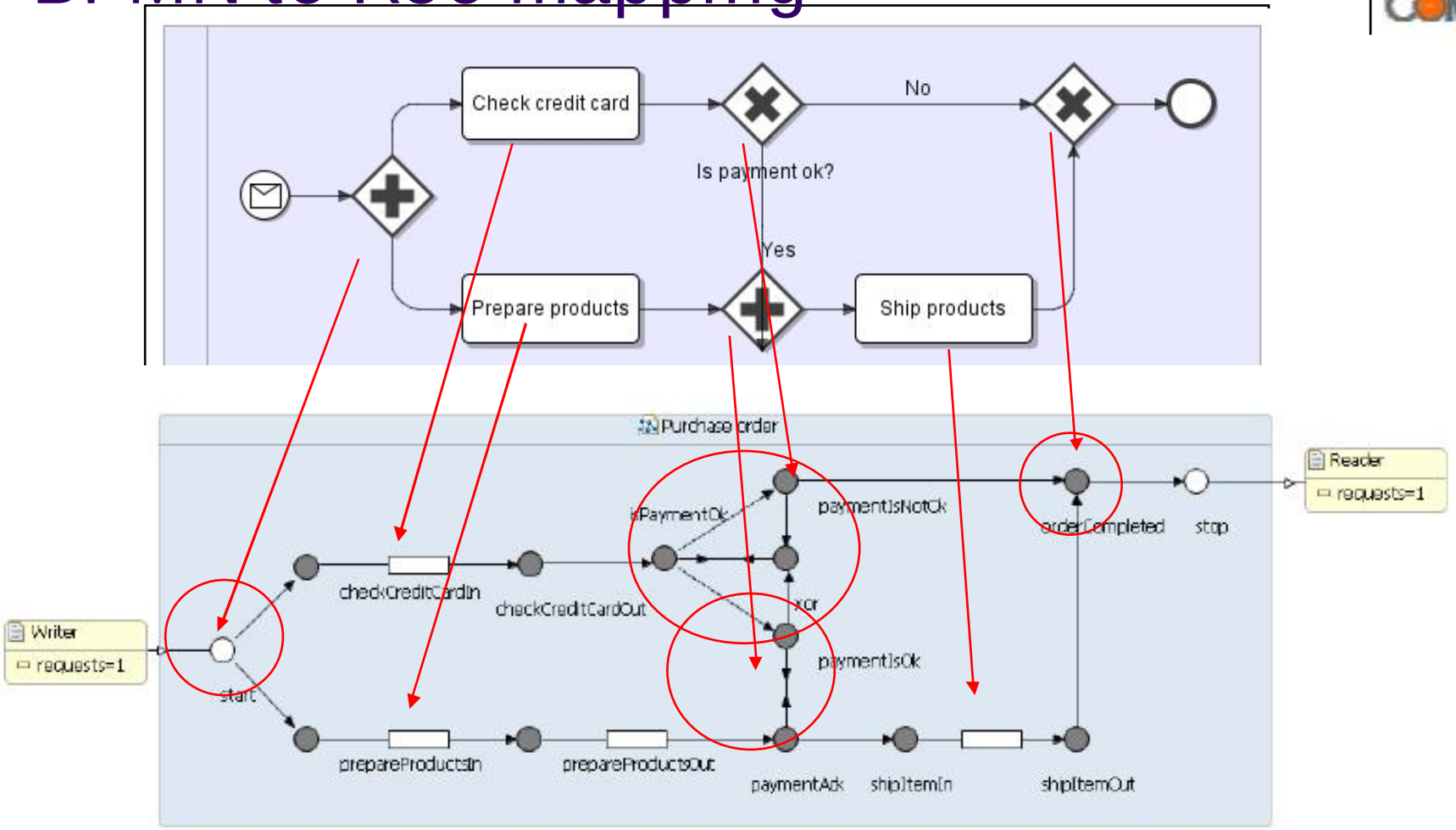
- Graphical notations:
  - Business Process Modeling Notation (BPMN),
  - UML2 Activity Diagrams,
  - UML2 Sequence Diagrams
- Business Process Execution Language (BPEL)
- Formal models for business process modeling and web service composition:
  - Petri-nets [HB03, YTX05, DDO08]
  - Transition systems [KPP06]
  - Process algebras [WG08, WG08a]
  - Logic-based approaches [MDK+01]

BPMN



C. Wolter and A. Schaad "Modeling of Task-Based Authorization Constraints in BPMN", BPM'07, volume 4714 of LNCS, Springer, pp. 64–79

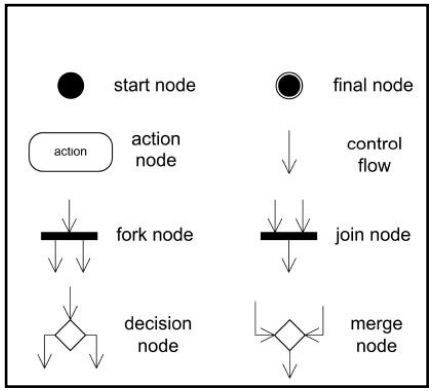
BPMN to Reo mapping



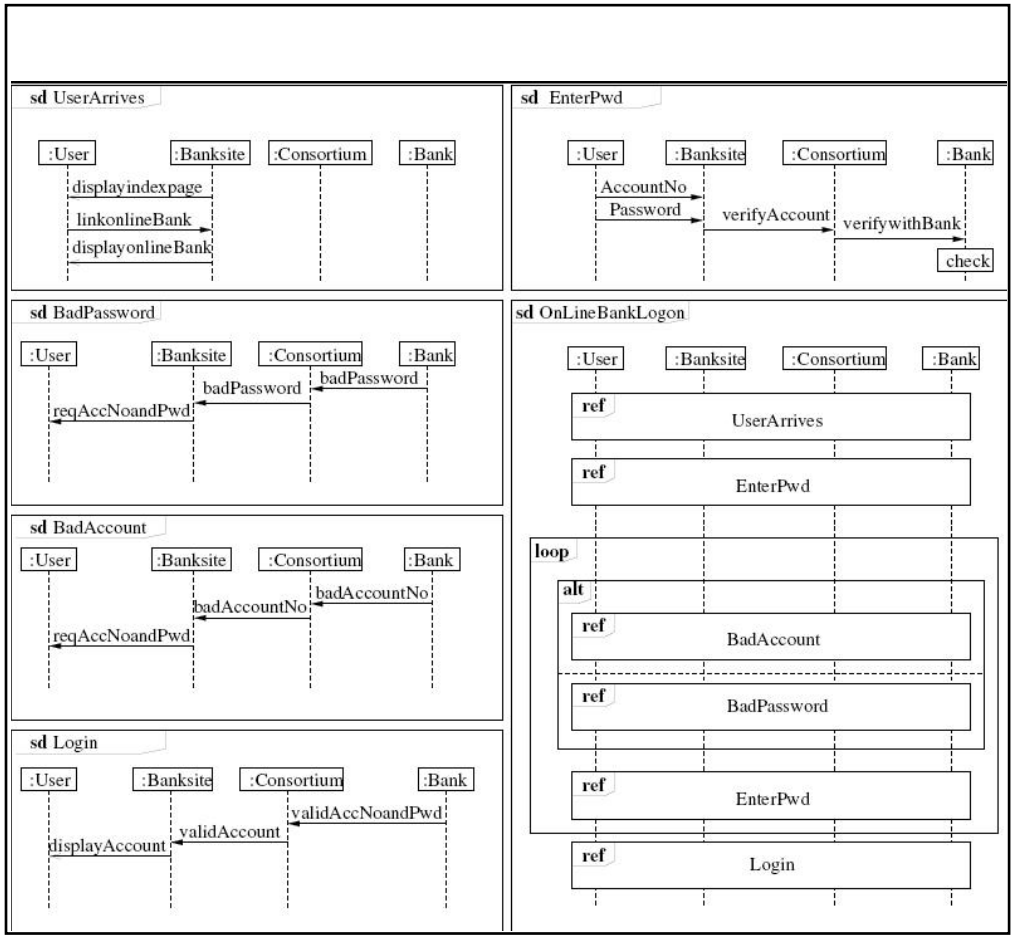
F. Arbab, N. Kokash and M. Sun: "Towards Using Reo for Compliance-aware Business Process Modelling." In: ISOLA'08, vol. 17 of CCIS, Springer, 2008, pp. 108-123.

UML2

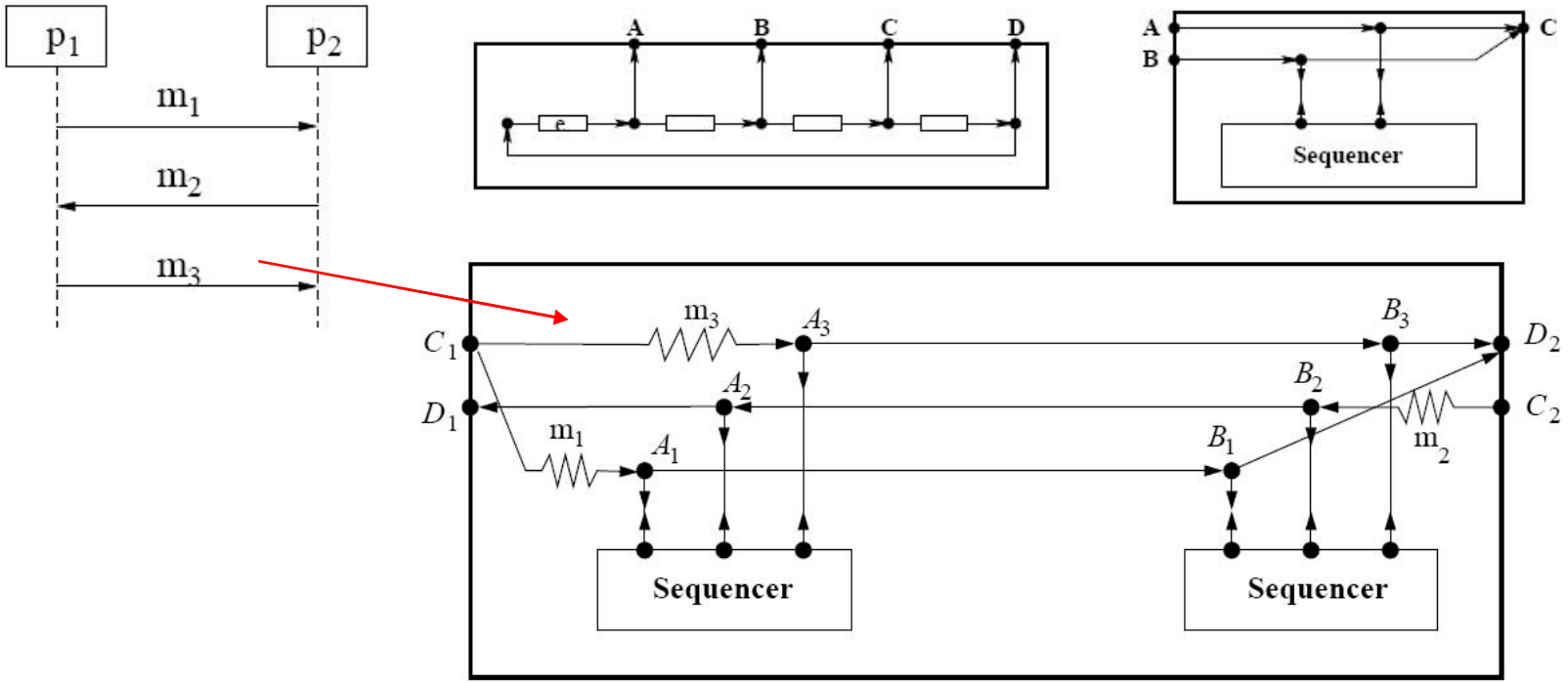
Activity Diagrams



Sequence Diagrams



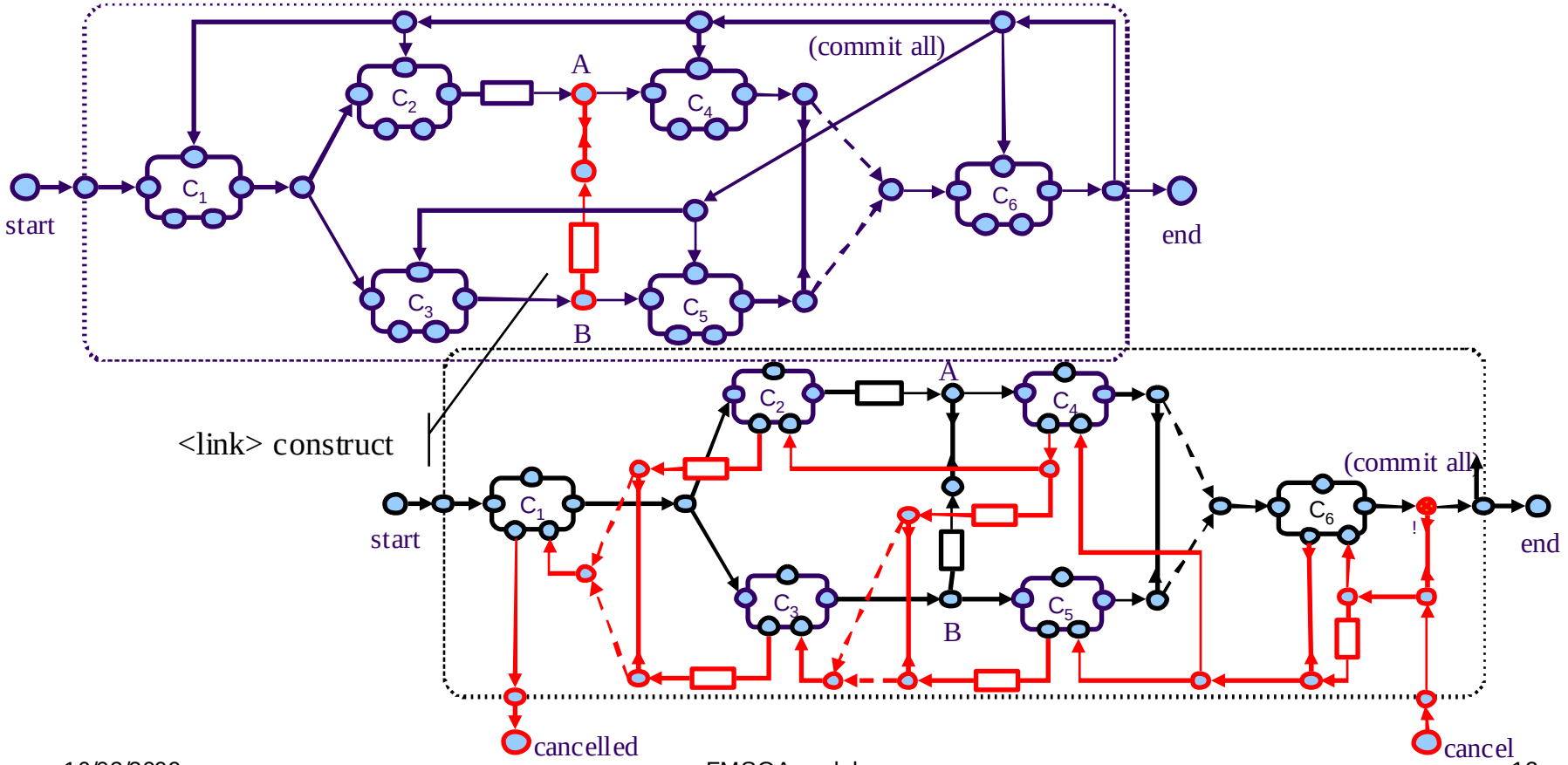
UML2 to Reo mapping



F. Arbab and M. Sun: “Synthesis of Connectors from Scenario-based Interaction Specifications.” In: Proceedings of the International Symposium on Component Based Software Engineering (CBSE'08), 2008.

BPEL to Reo mapping

S. Tasharofi, M. Vakilian, R. Z. Moghaddam and M. Sirjani, “Modeling Web Service Interactions Using the Coordination Language Reo”, Proc. of the Int. Workshop on Web Services and Formal Methods, 2008, volume 4937 of LNCS, Springer, pp. 108-123



Vereofy model checker

- Developed at TU Dresden as part of the EU project CREDO and NWO/DFG bilateral project SYANCO
- Command line tool and integrated into Eclipse environment
- Input format:
  - Reo Scripting Language (RSL) – syntactic version of Reo
  - Constraint Automata Reactive Module Language (CARML) – syntactic version of CA
- Specifications:
  - Linear Temporal Logic (LTL)
  - Alternating-time Stream Logic (ASL)

ASL

- ASL is a CTL-like logic which combines features of BTSL and ATL
  - S. Klüppelholz and C. Baier. Alternating-Time Stream Logic for Multi-Agent Systems. Proc. of the Int. Conf. on Coordination Models and Languages, 2008.
- Branching Time Stream Logic (BTSL) – is a logic specially designed for Reo. It extends CTL with the ability to express conditions on data flow in channel nodes using regular expressions
  - S. Klüppelholz and C. Baier. “Symbolic Model Checking for Channel-based Component Connectors”. Proc. of the Int. Workshop on the Foundations of Coordination Languages and Software Architectures, volume 175(2) of ENTCS, pp. 19–37, 2007.
- Alternating-time Temporal Logic (ATL) – reasoning about existence or absence of a coalition's strategy to achieve or avoid a specific temporal goal given the behavioral specification of each component

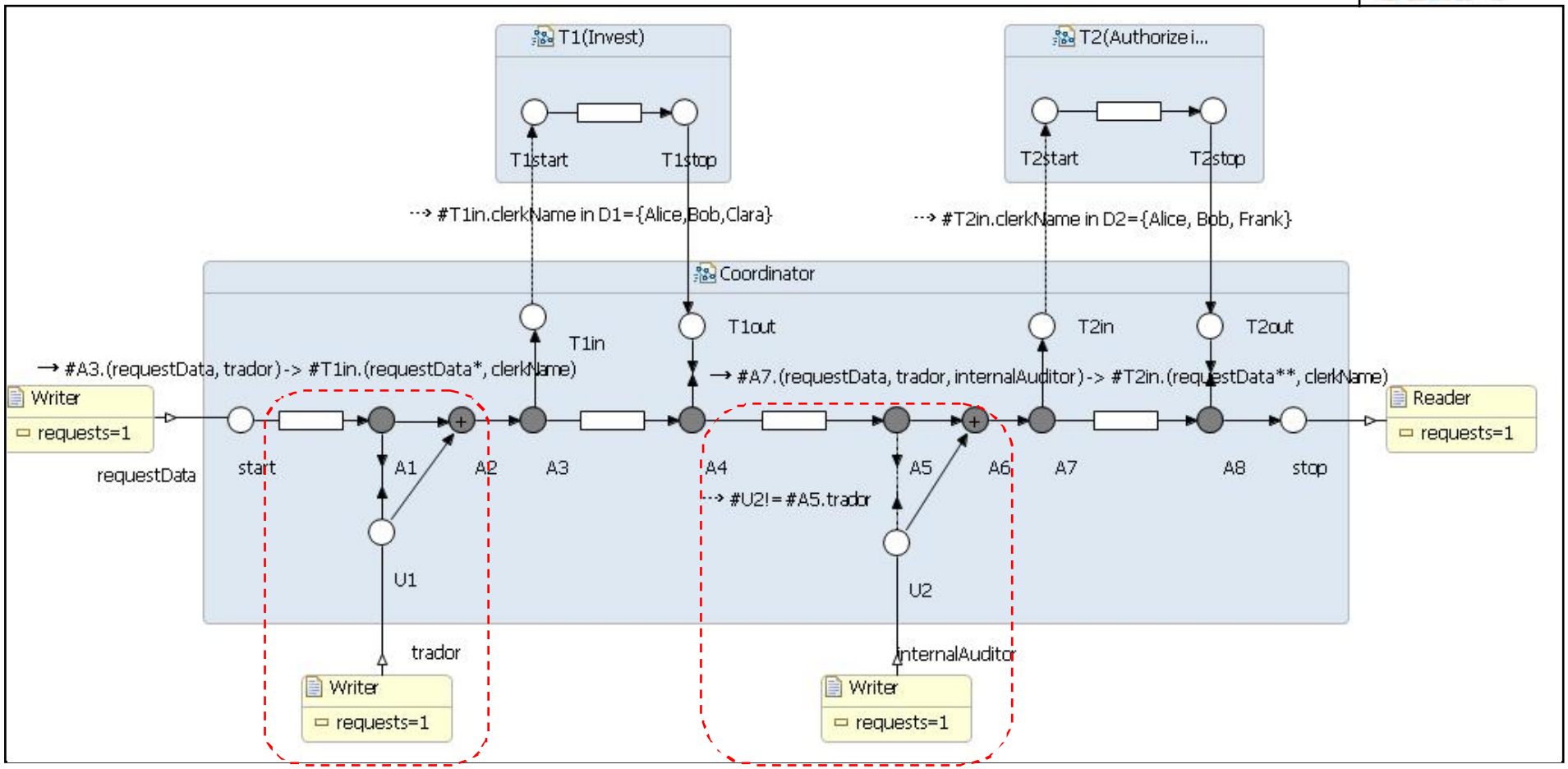
ASL syntax

```
1  E<alpha>Phi | E[alpha]Phi | A<alpha>Phi | A[alpha]Phi
2
3  alpha := io_guards |
4           (alpha)* | (alpha)+ |
5           (alpha1;alpha2) | (alpha1|alpha2)
6
7  io_guards := epsilon | dataflow | io_guard |
8               io_guard & io_guards
9
10 io_guard := p |  !p | {p1,..,pk} |
11             #p==d | #p!=d | #p==X | #p!=X |
12             #p1==#p2 | #p1!=#p2
```

```
1  E{N1,..,Nk}X[Phi] | E{N1,..,Nk}U[Phi1,Phi2] | E{N1,..,Nk}F[Phi]
2  E{N1,..,Nk}G[Phi] | E{N1,..,Nk}R[Phi1,Phi2]
3  E{N1,..,Nk}<alpha>Phi | E{N1,..,Nk}[alpha]Phi
4
5  A{N1,..,Nk}X[Phi] | A{N1,..,Nk}U[Phi1,Phi2] | A{N1,..,Nk}F[Phi]
6  A{N1,..,Nk}G[Phi] | A{N1,..,Nk}R[Phi1,Phi2]
7  A{N1,..,Nk}<alpha>Phi | A{N1,..,Nk}[alpha]Phi
```

Segregation of duties

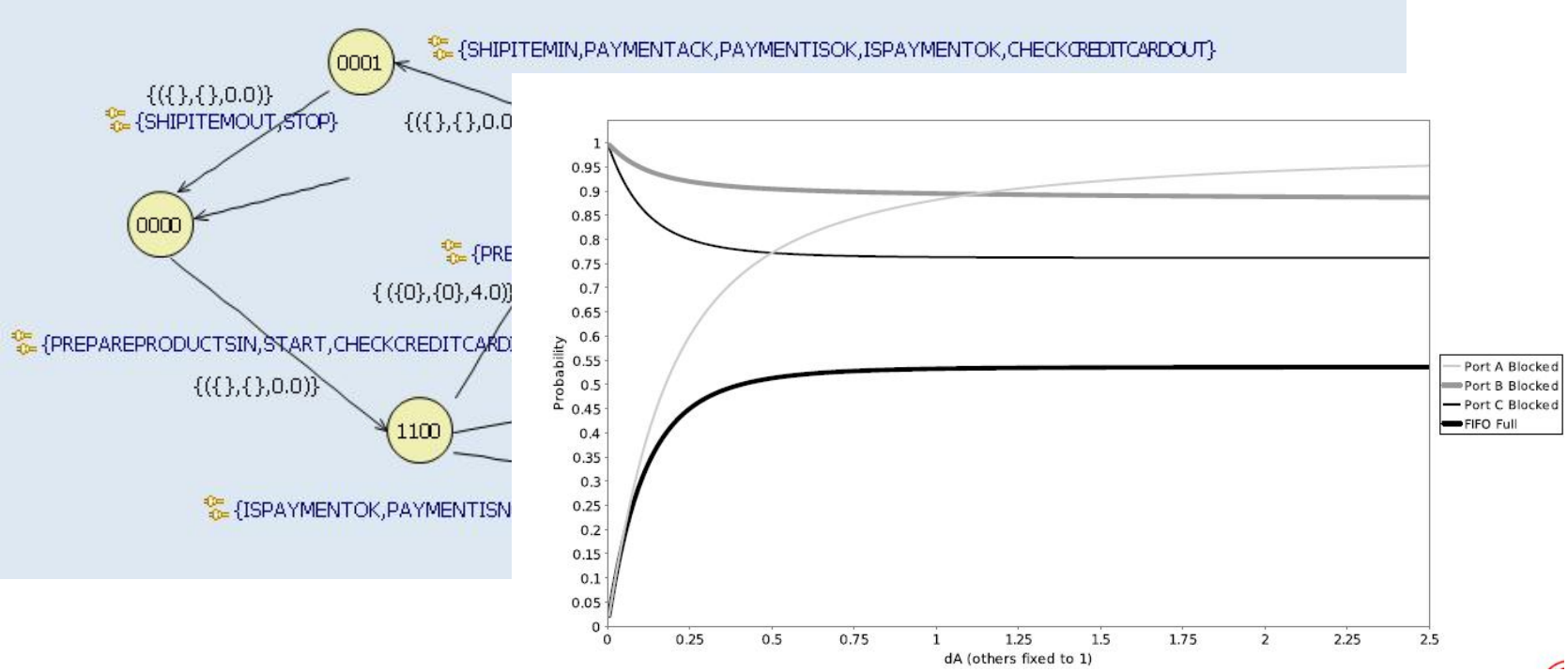
Investment banking scenario



- ASL: A[#T1start.clerkName!=T2start.clerkName]true

QoS Analysis with PRISM

- Part of the CooPer project
- Translation of the quantitative Reo to a stochastic model
- Integration of the PRISM model checker to the ECT



Demo: Compliance-aware business process development with ECT

ECT = Eclipse Coordination Tools http://reo.project.cwi.nl/

- Reo editor
- Animation view
- Reo to Constraint Automata converter
- Model checking
  - Vereofy (Control/data flow analysis)
  - PRISM (QoS analysis)
- Code Generation
- Conversion Tools
  - BPEL to Reo
  - BPMN to Reo
  - UMLSDs to Reo

- Example: Purchase order scenario

Future work

- Modeling of real-world scenarios and compliance requirements (COMPAS case studies)
- How can we deal with different compliance categories (data flow, resources, security, privacy, etc.)
- Dynamic process adaptation
  - Scripts for connector reconfiguration
- Dynamic service discovery
  - Integration of syntactic/semantic matching algorithms for matching port names and algorithms for CA bisimulation equivalence checking
  - Convert service specifications to CA (e.g., WSRF)

Related Work

- BPMN semantics
  - [DDO08] Dijkman, R.M., Dumas, M., Ouyang, C.: Formal semantics and analysis of BPMN process models. In: Information and Software Technology (IST). (2008)
  - Wong, P., Gibbons, J.: A process semantics for BPMN. Technical report, Queensland University of Technology (2007)
  - Wong, P., Gibbons, J.: A relative timed semantics for BPMN. Technical report, Queensland University of Technology (2007)
- BPEL semantics
  - [Loh08] Lohmann, N.: A feature-complete Petri net semantics for WS-BPEL 2.0. In: Proc. of the Int.Workshop on Web Services and Formal Methods. Volume 4937 of LNCS., Springer (2008) 77-91
  - [LM07] Lucchia, R., Mazzara, M.: A pi-calculus based semantics for WS-BPEL. Journal of Logic and Algebraic Programming 70(1) (2007) 96-118
- UML sematics
  - [SH05] H. Störrle, J. H. Hausmann: “Towards a Formal Semantics of UML 2.0 Activities”. Software Engineering, 2005, pp. 117-128.

Related Work

- Formal Methods for Compliance-aware Business Process Design
  - [MLX07] Liu, Y., Muller, S., Xu, K.: A static compliance-checking framework for business process models. IBM Systems Journal 46(2) (2007) 335-361
  - [GK07] Ghose, A.K., Koliadis, G.: Auditing business process compliance. In: Proc. of the Int. Conf. on Service-Oriented Architectures (ICSOC'07). Volume 4749 of LNCS., Springer (2007) 169-180
  - [GMS06] Governatori, G., Milosevic, Z., Sadiq, S.: Compliance checking between business processes and business contracts. In: Proc. of the Int. Enterprize Distributed Object Computing Conf. (EDOC'06), IEEE Computer Society (2006) 221-232
  - [BCC+07] Brunel, J., Cuppens, F., Cuppens, N., Sans, T., Bodeveix, J.P.: Security policy compliance with violation management. In: Proc. of the Workshop on Formal Methods in Security Engineering (FMSE'07), ACM Press (2007) 31-40
  - [ADW08] A. Awad, G. Decker and M. Weske, “Efficient Compliance Checking Using BPMN-Q and Temporal Logic”, Proc. of the Int. Conf. on Business Process Management (BPM), 2008
  - [KPP06] R. Kazhamiakin, P. K. Pandya, and M. Pistore. Representation, Verification, and Computation of Timed Properties in Web Service Compositions. In Proc. ICWS, 2006.
  - [SLS06] A. Schaad, V. Lotz, K. Sohr: “A Model-checking Approach to Analysing Organisational Controls in a Loan Origination Process”. In: Proceedings of the eleventh ACM symposium on Access Control Models and Technologies (SACMAT), 2006.

The end

- Thank you!