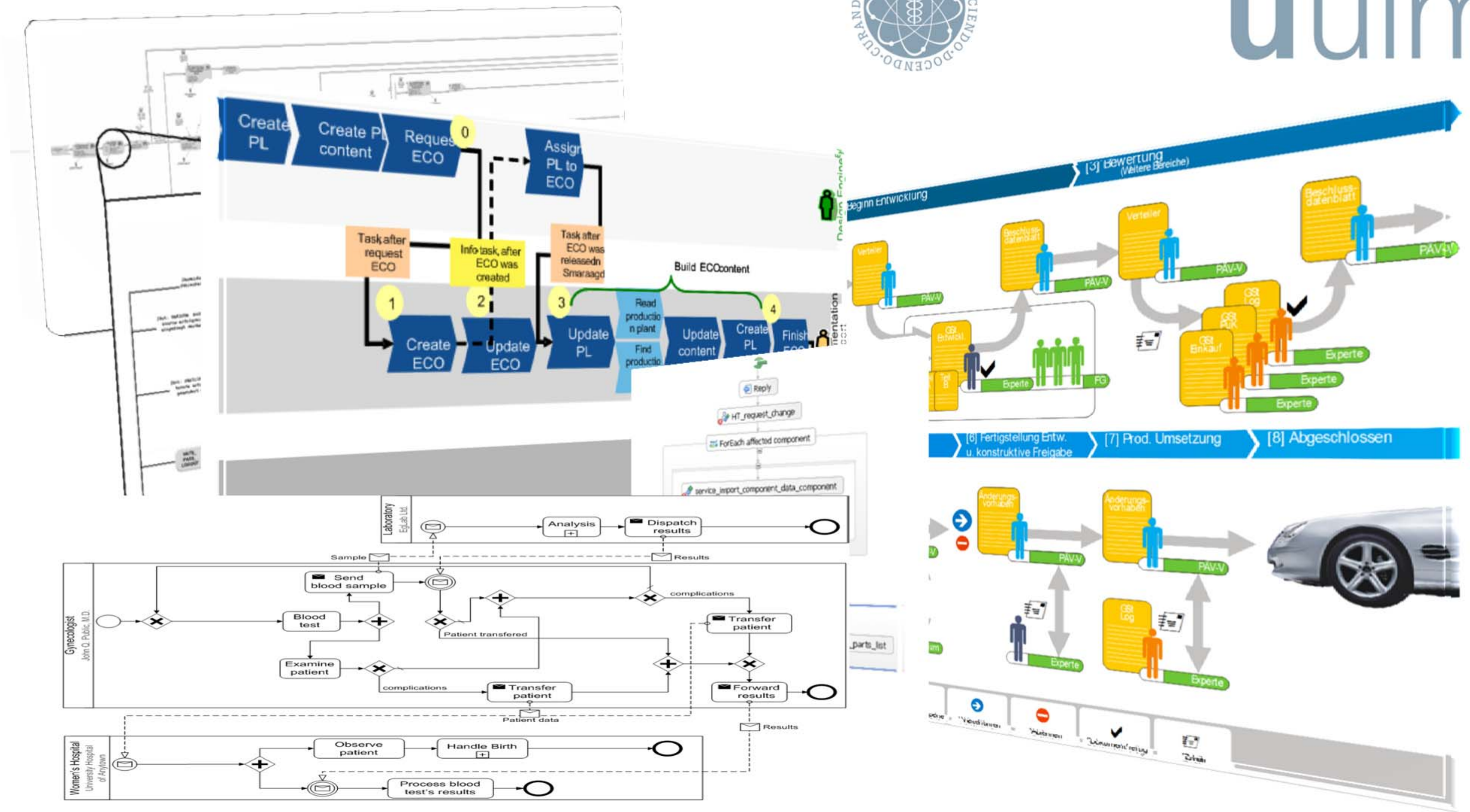
ulm university universität ulm



From Theory to Practice – Transferring Innovative BPM Research to Industrial Practice

Manfred Reichert

Introduction

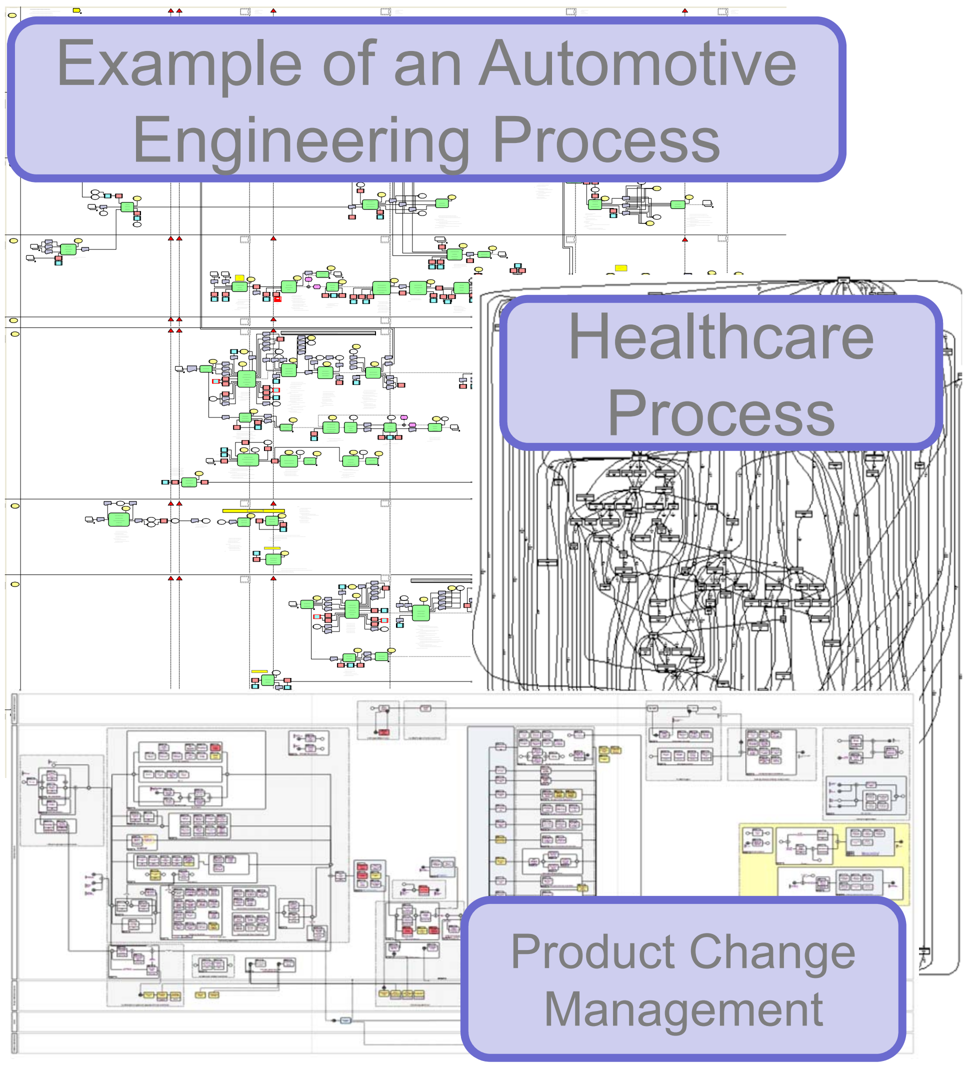
ADEPT – A Decade of Research on Flexible Process-Aware Information Systems and its Transfer to Practice

Large Processes in the Automotive Industry and Technologies to Support them

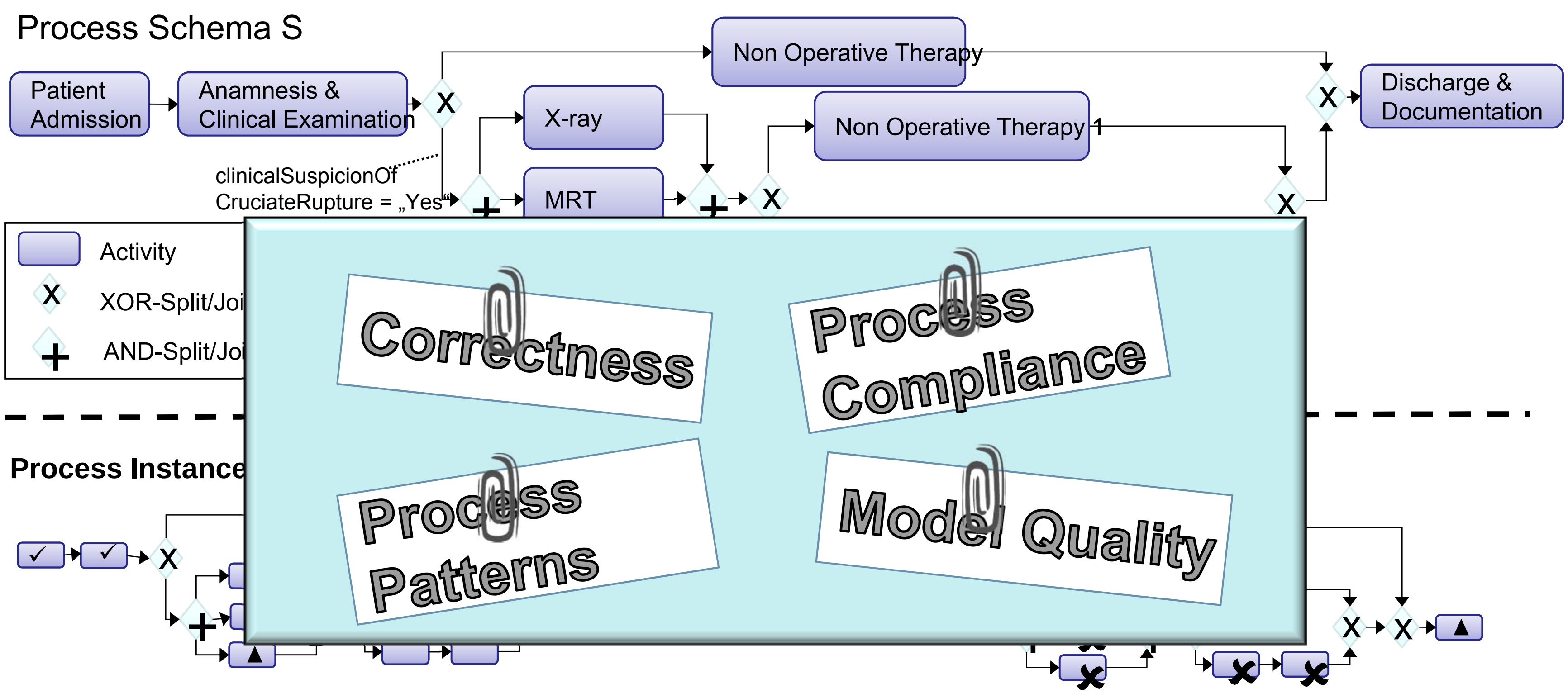
Conclusions

Introduction

- Processes can become very large and complex
- Thousands of concurrently executed process instances
- High need for flexibility in all phases of the process lifecycle
- Support for application integration is fundamental
- Correctness and robustness are crucial features of any process-aware information systems
- Integrated support of all phases of the process lifecycle required



Introduction: PAIS Build-Time

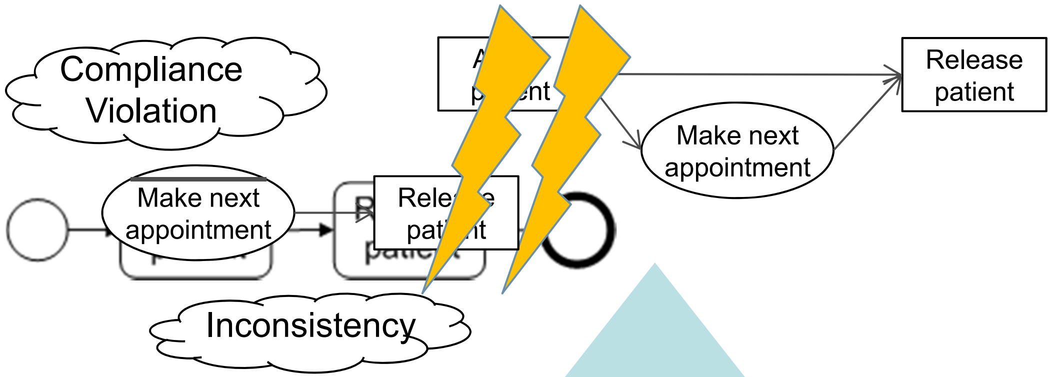


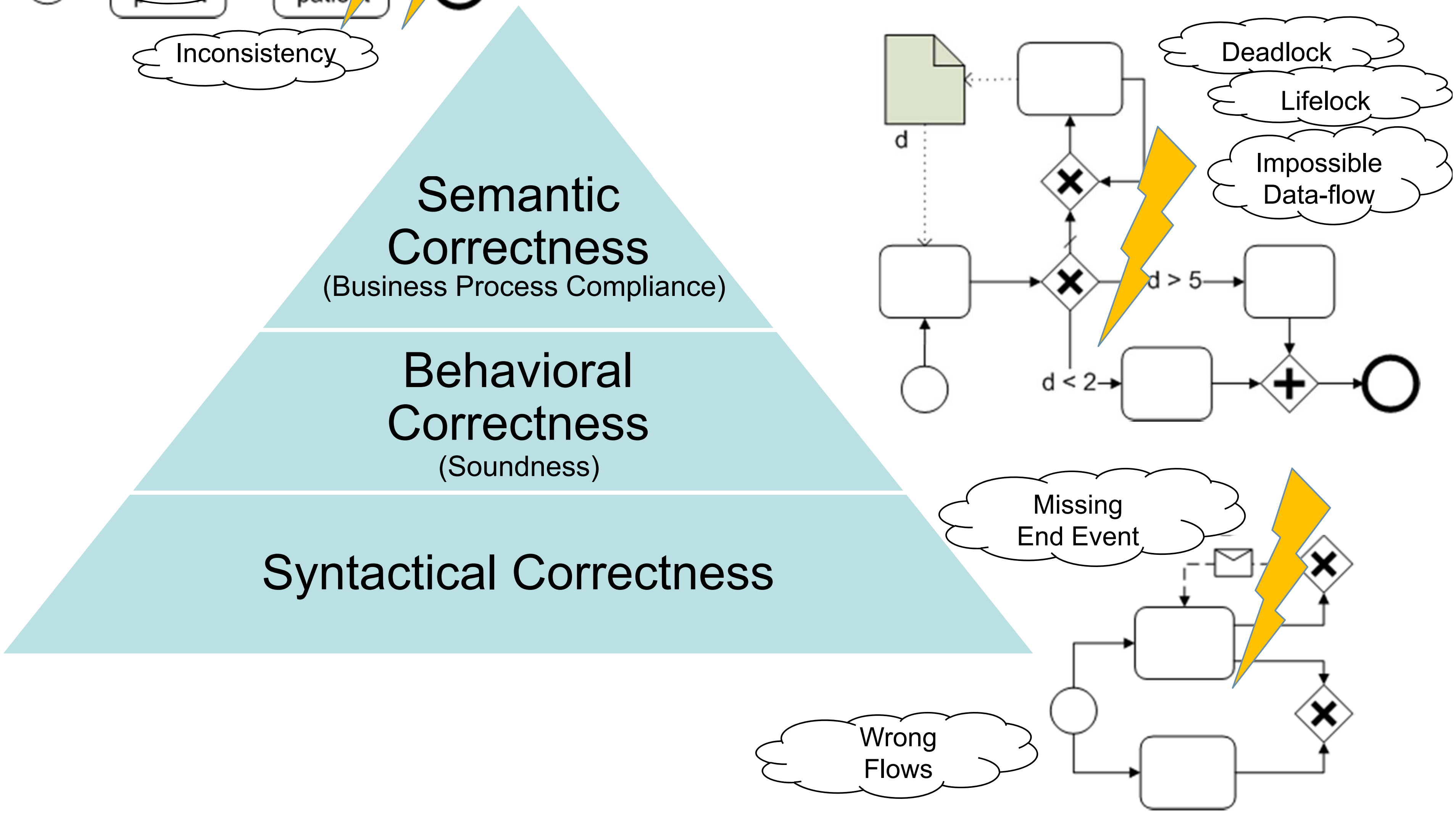
Execution Trace:
σ_1 = < „Patient Admission“, „Anamnesis & Clinical Examination“, „X-ray“>

Execution Trace:
σ_2 = < „Patient Admission“, „Anamnesis & Clinical Examination“, „Non Operative Therapy“>

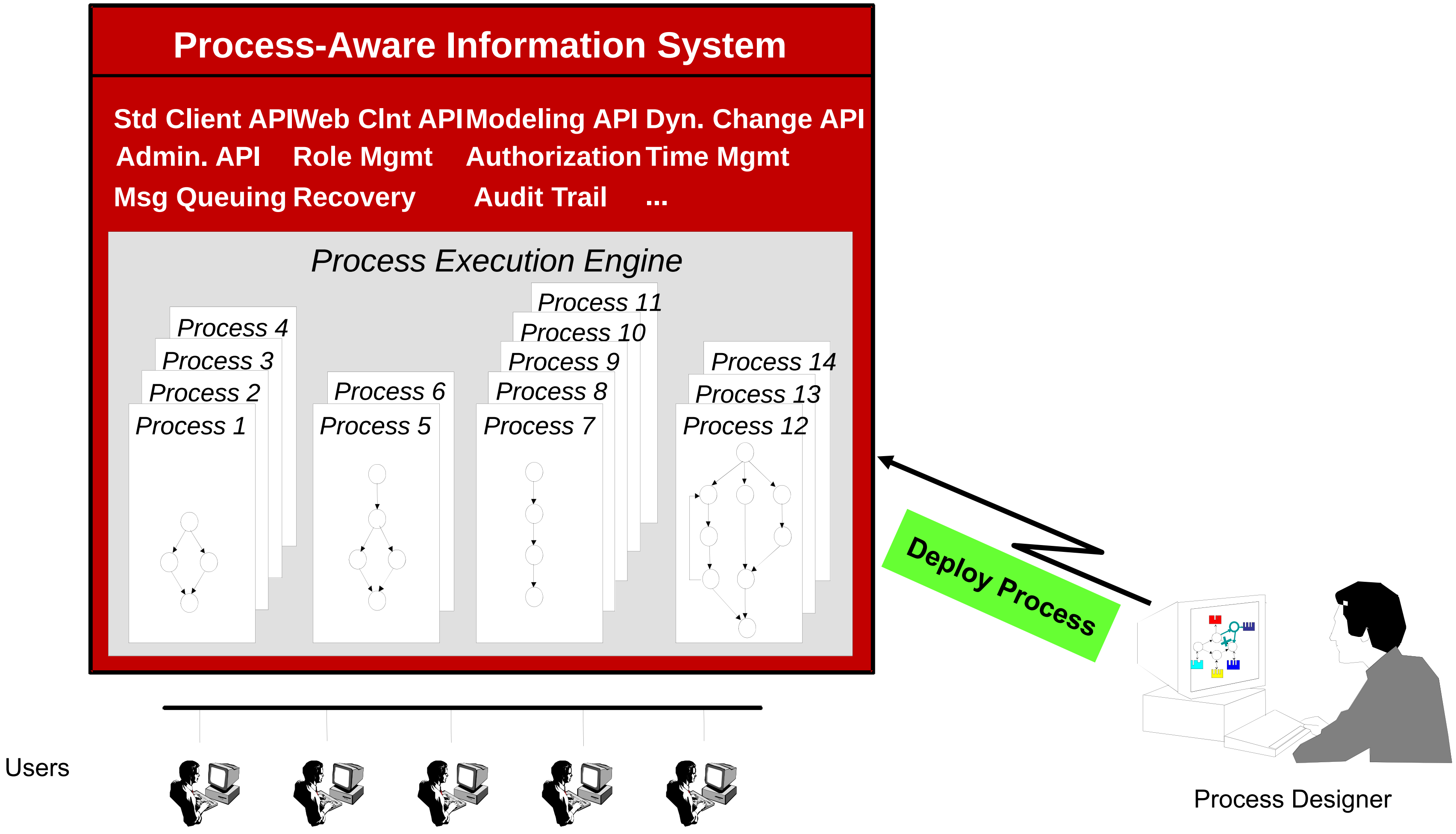
Activity States:	▲ Activated	✓ Completed	Skipped

PAIS: Levels of Correctness

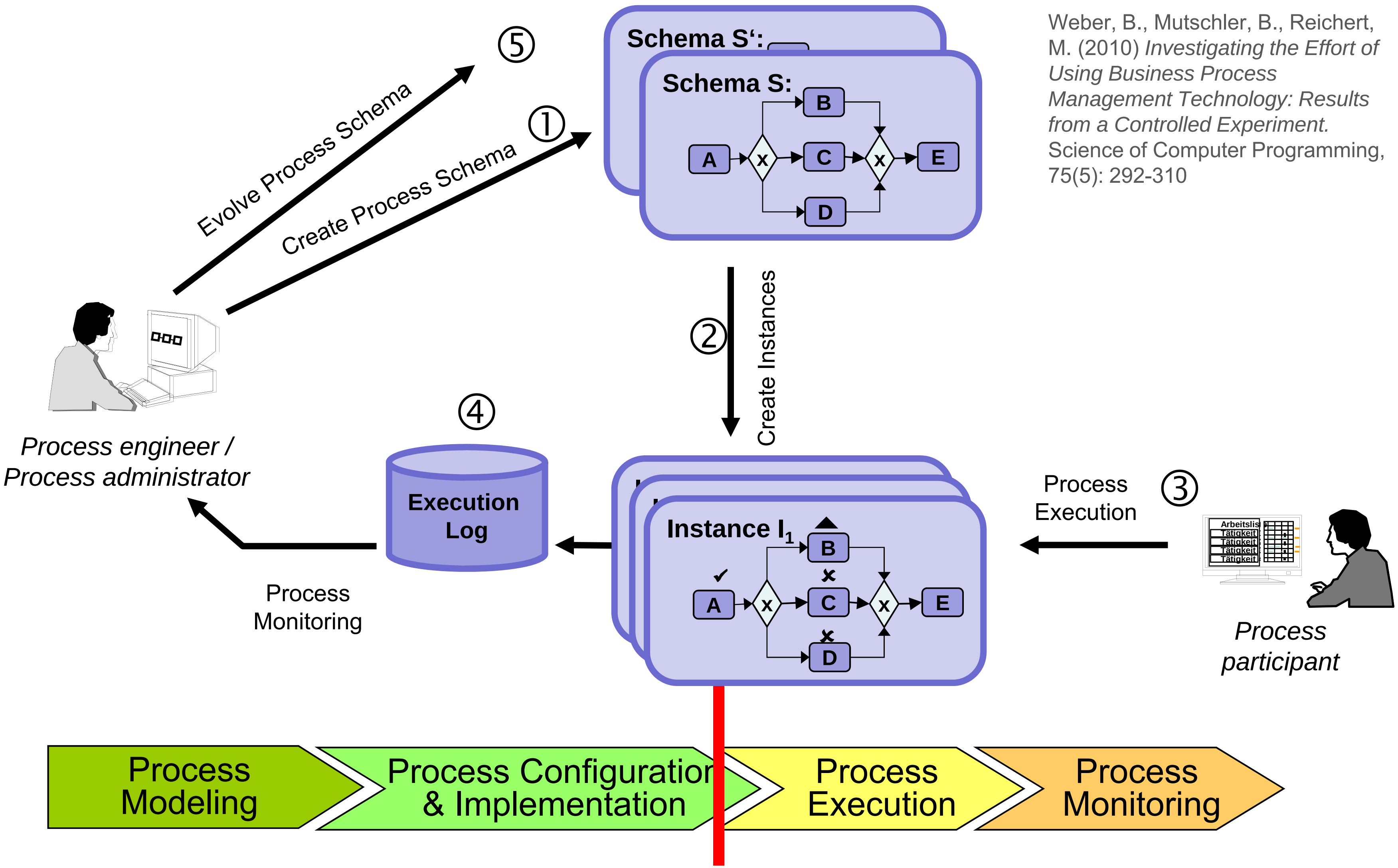




Introduction: PAIS Run-Time



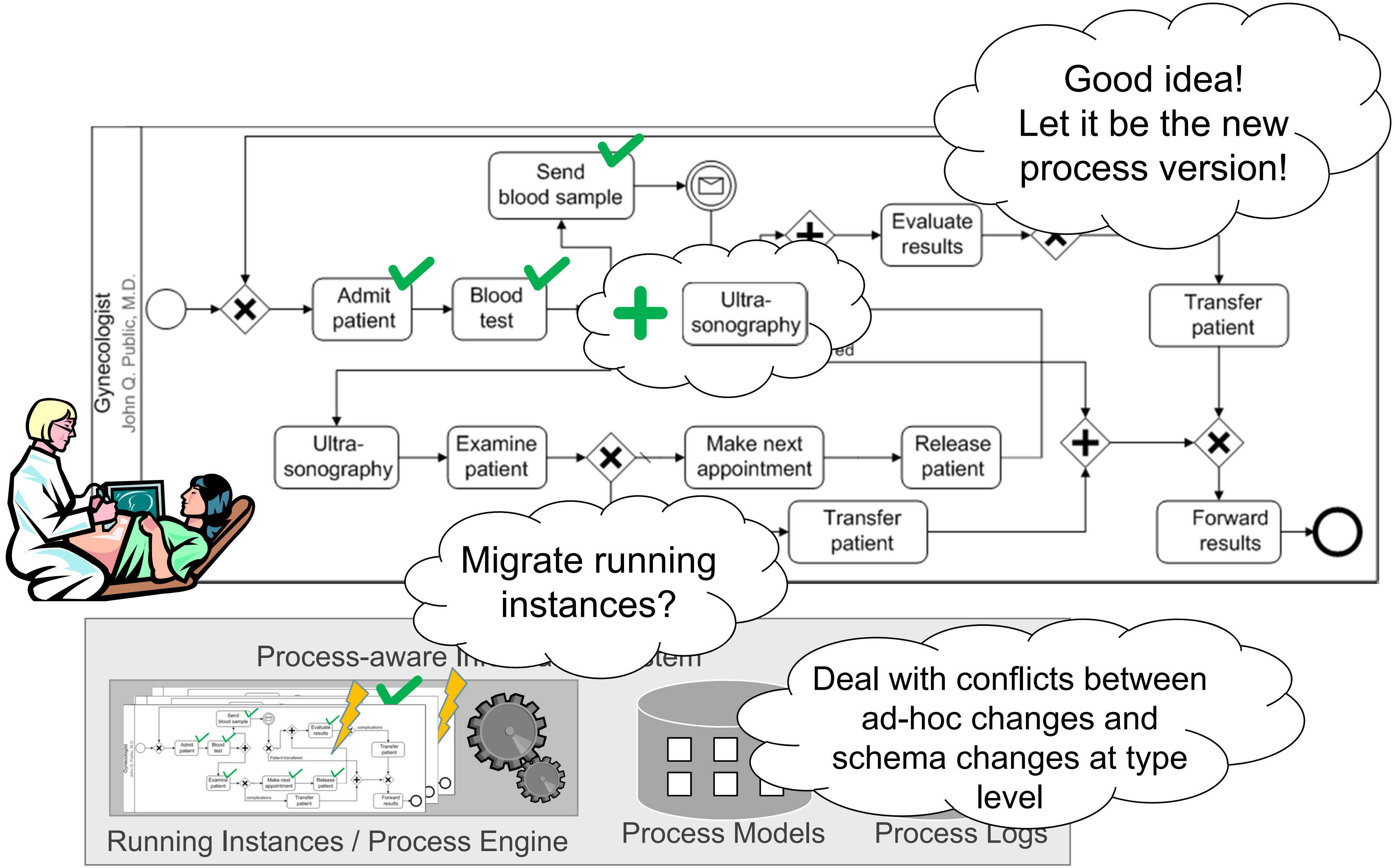
Introduction: PAIS Lifecycle



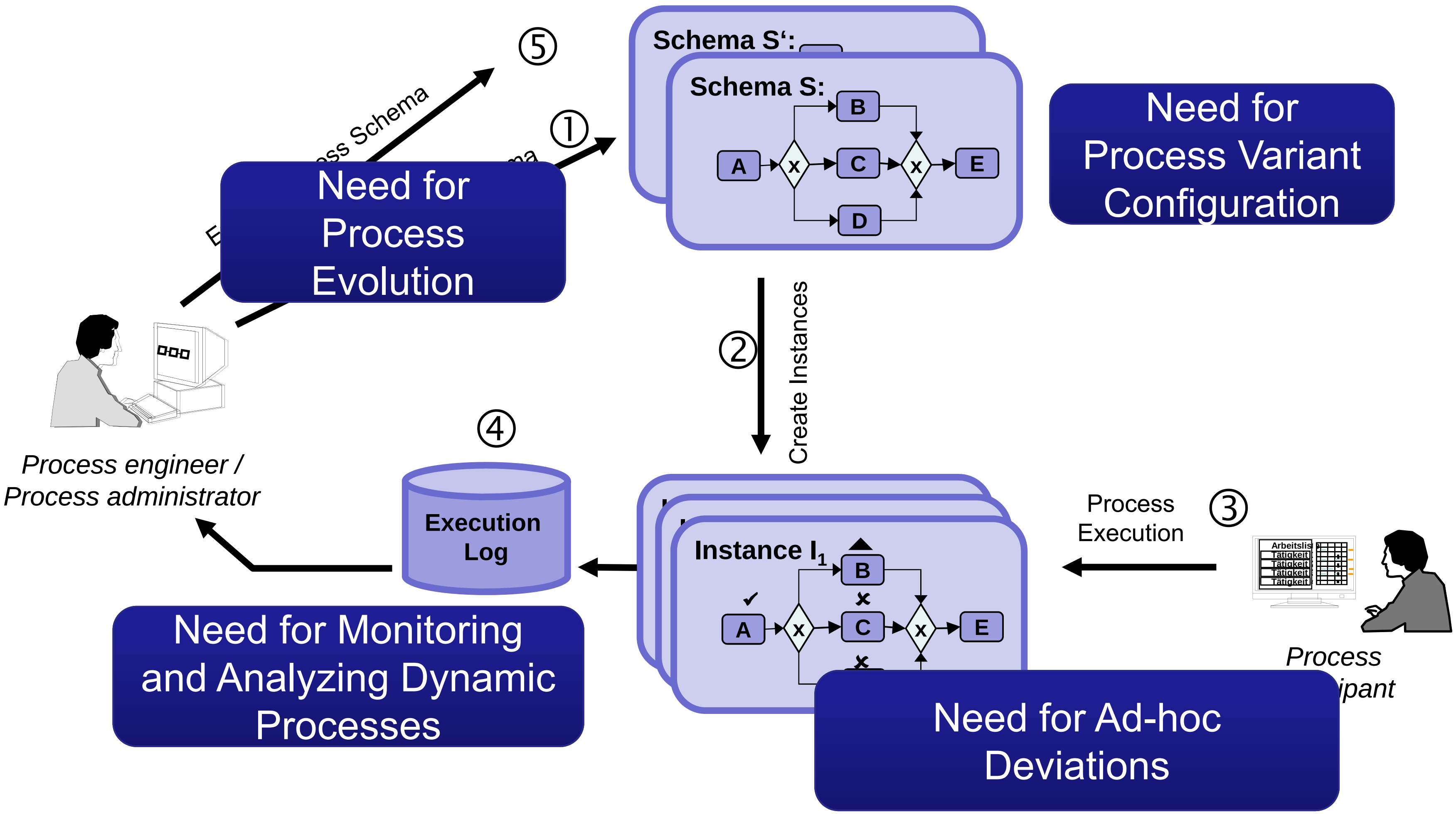
Weber, B., Mutschler, B., Reichert, M. (2010) *Investigating the Effort of Using Business Process Management Technology: Results from a Controlled Experiment.* Science of Computer Programming, 75(5): 292-310

- Introduction
- **ADEPT – A Decade of Research on Flexible Process-Aware Information Systems and its Transfer to Practice**
- Large Processes in the Automotive Industry and Technologies to Support them
- Conclusions

ADEPT: Challenges

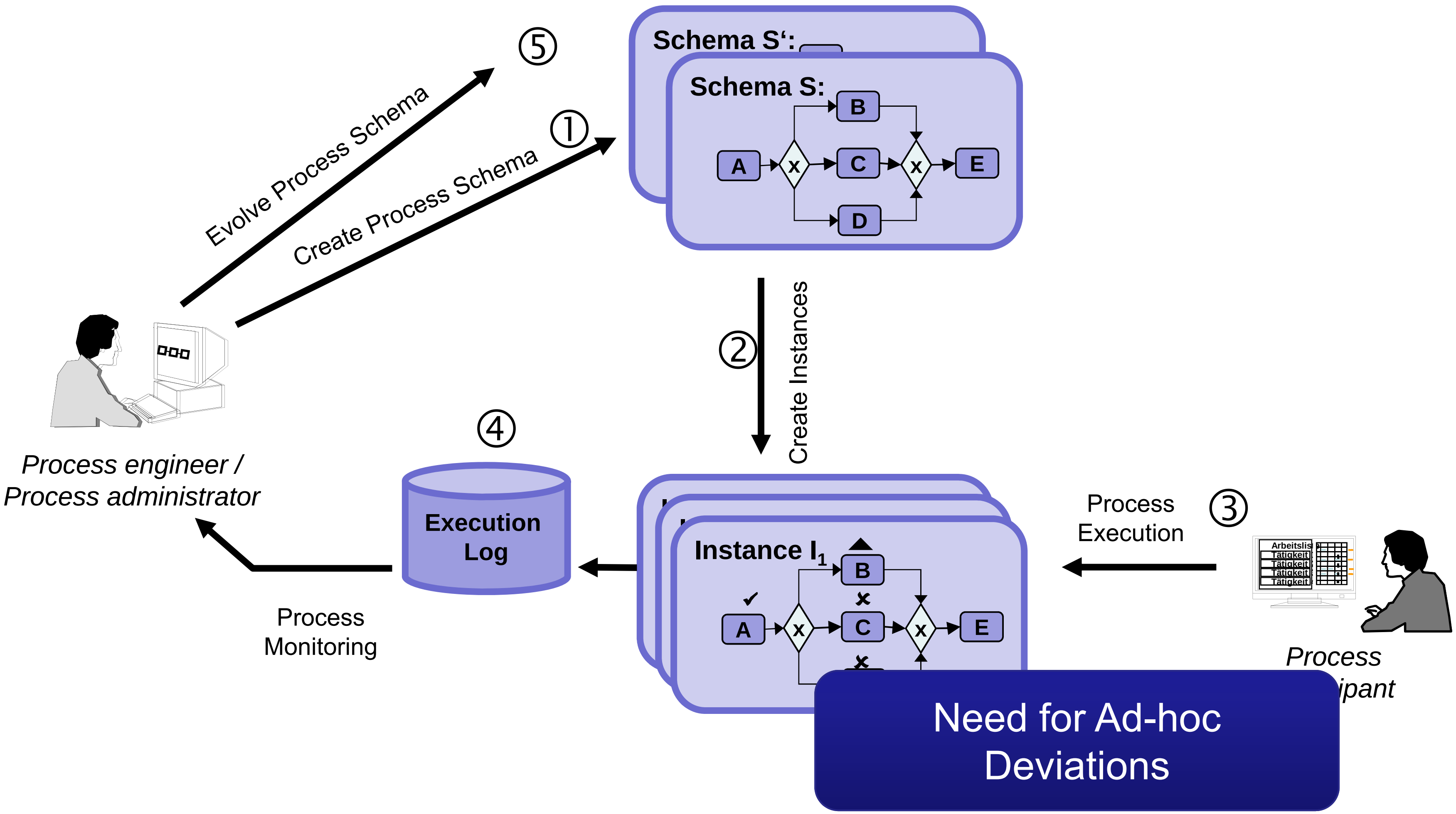


ADEPT: Challenges

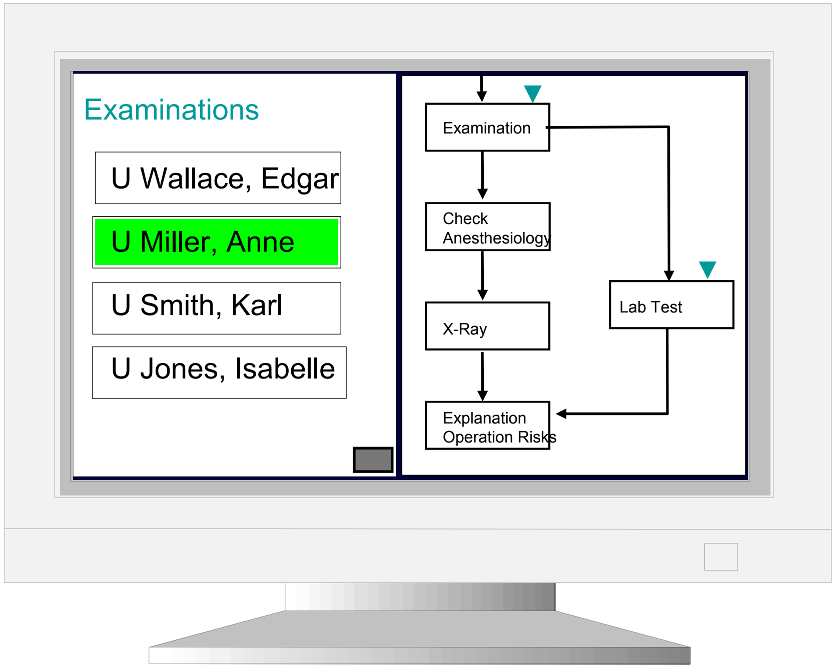


Reichert, M., Rinderle-Ma, S., Dadam, P. (2010) *Flexibility in Process-aware Information Systems.* LNCS Transactions on Petri Nets and Other Models of Concurrency, Vol. 2, pp. 115-135

ADEPT: Challenges



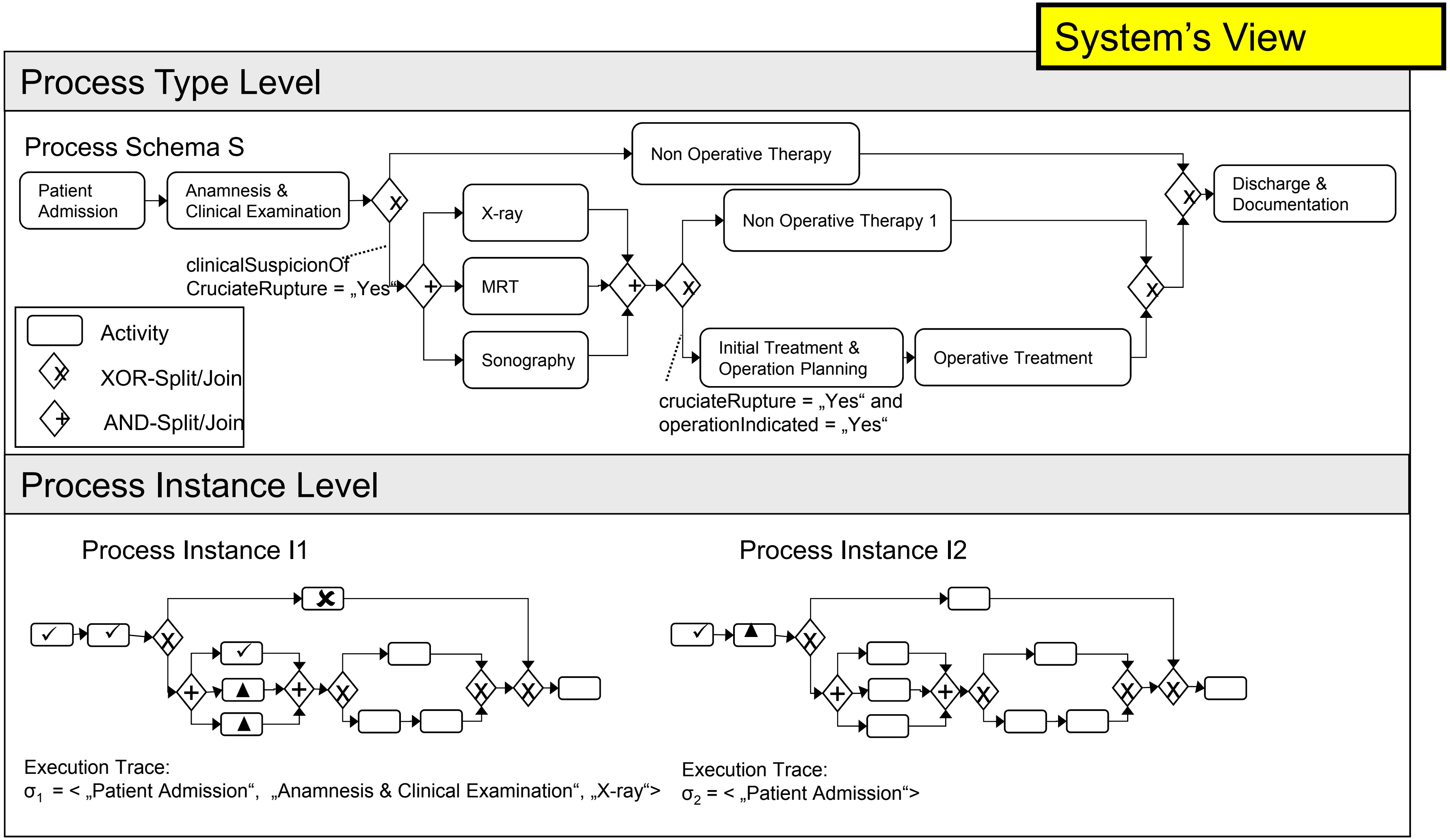
ADEPT: Ad-hoc Changes



The Users' View

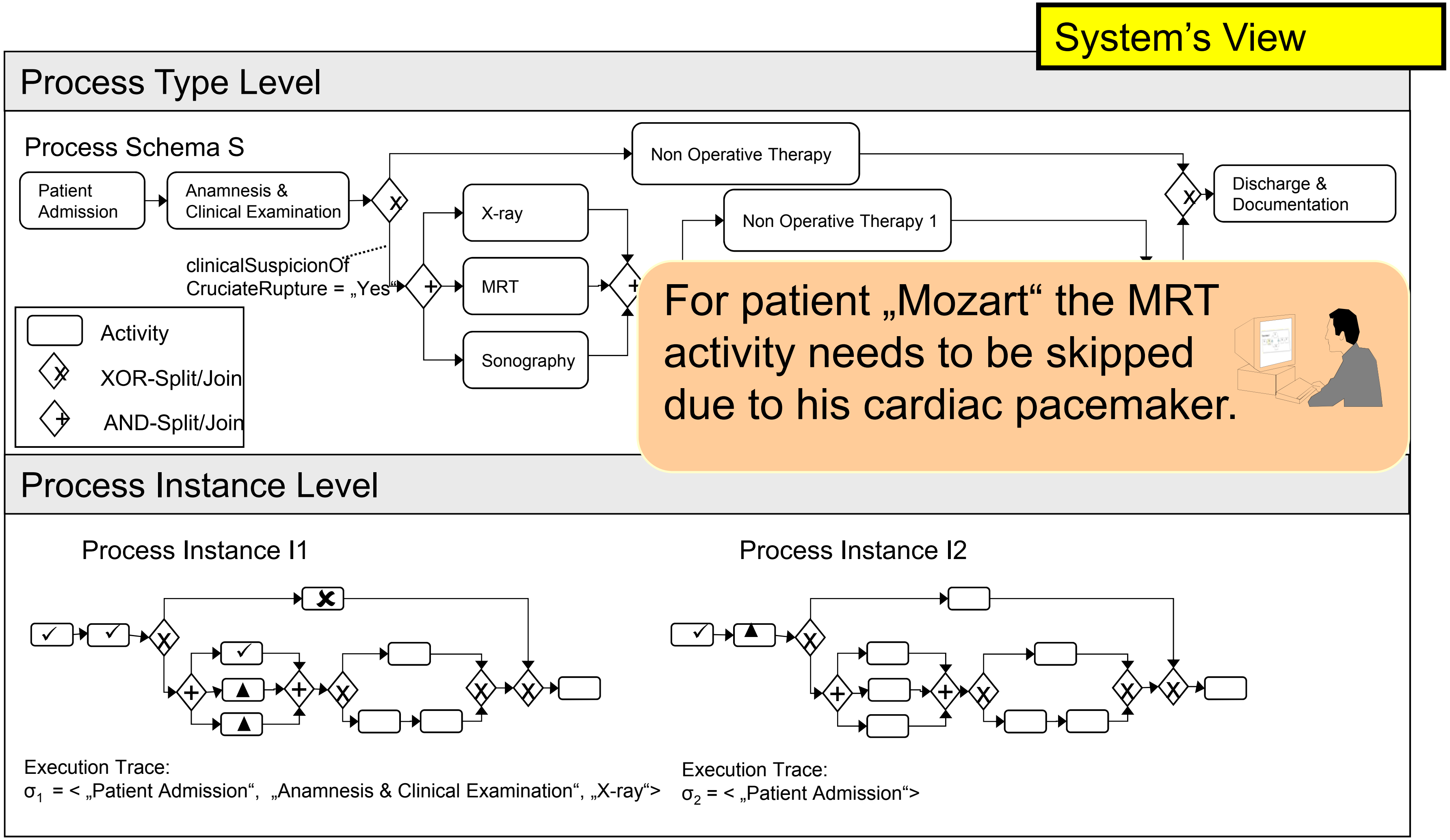
Reichert, Manfred and Dadam, Peter (1998) *ADEPTflex-Supporting Dynamic Changes of Workflows Without Losing Control.* Journal of Intelligent Information Systems, Special Issue on Workflow Management Systems, 10(2): 93-129, Kluwer

ADEPT: Ad-hoc Changes



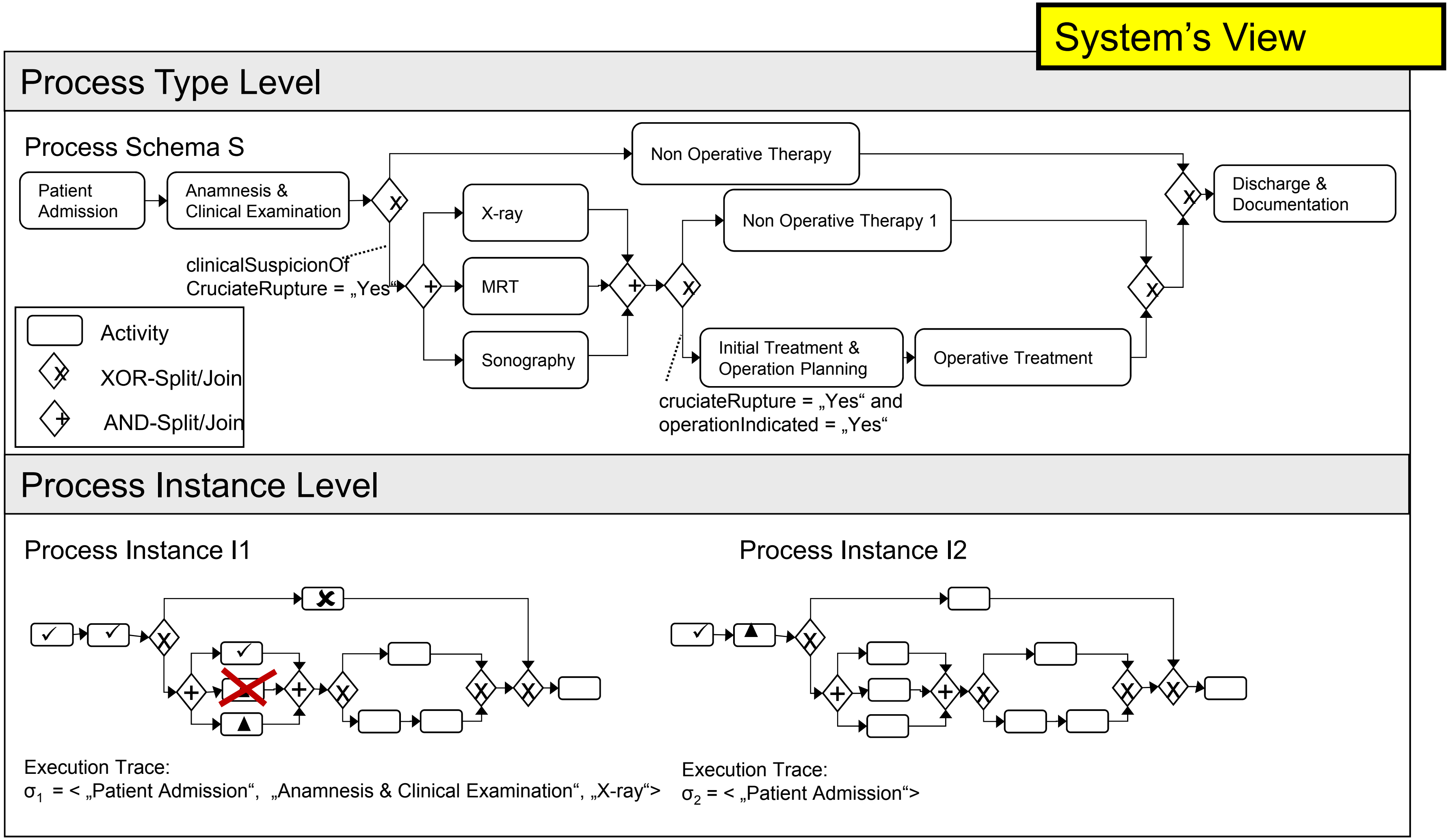
Reichert, Manfred and Dadam, Peter (1998) *ADEPTflex-Supporting Dynamic Changes of Workflows Without Losing Control.* Journal of Intelligent Information Systems, Special Issue on Workflow Management Systems, Vol. 10, No. 2, pp. 93-129

ADEPT: Ad-hoc Changes



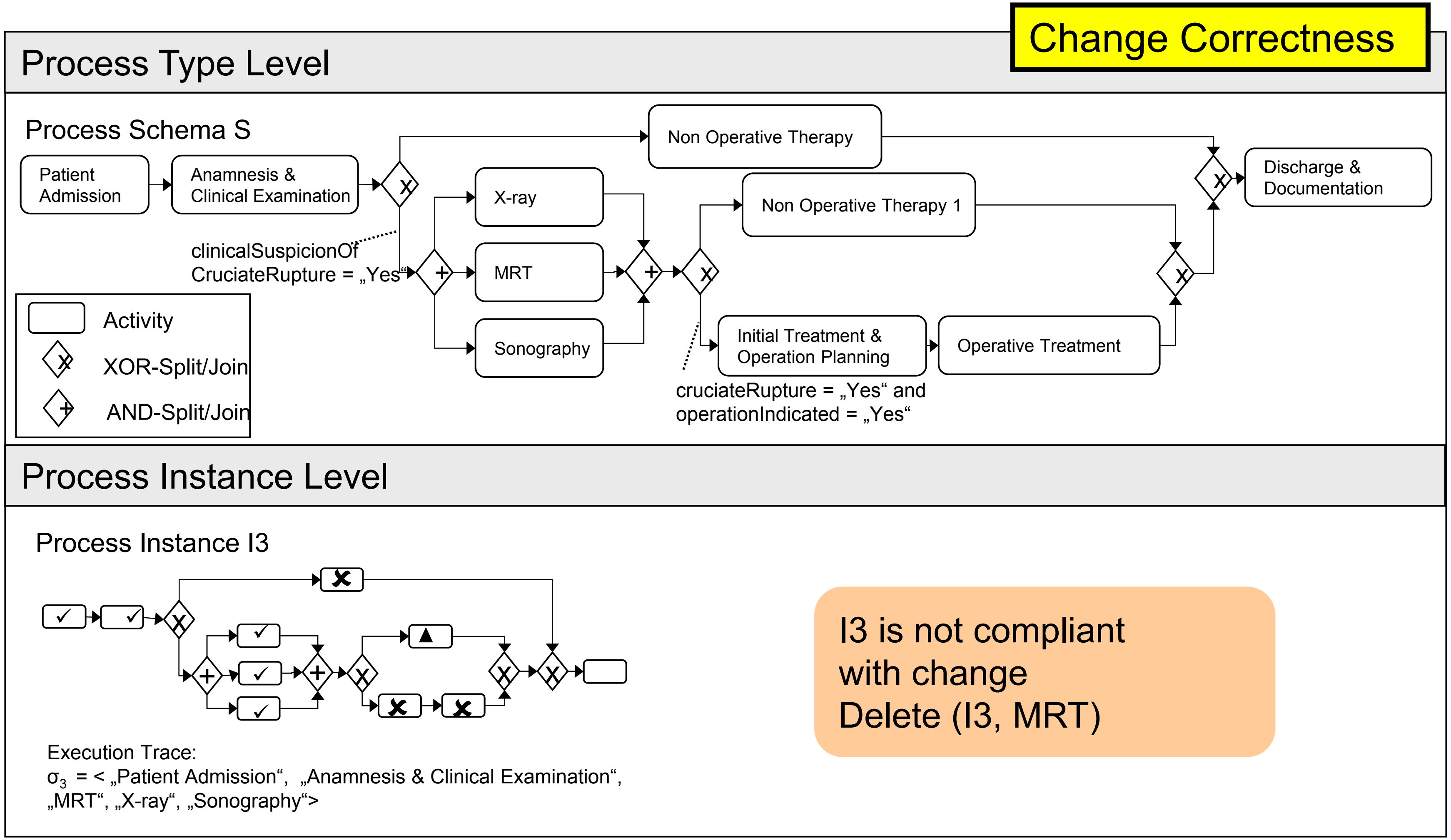
Reichert, Manfred and Dadam, Peter (1998) *ADEPTflex-Supporting Dynamic Changes of Workflows Without Losing Control.* Journal of Intelligent Information Systems, Special Issue on Workflow Management Systems, Vol. 10, No. 2, pp. 93-129

ADEPT: Ad-hoc Changes



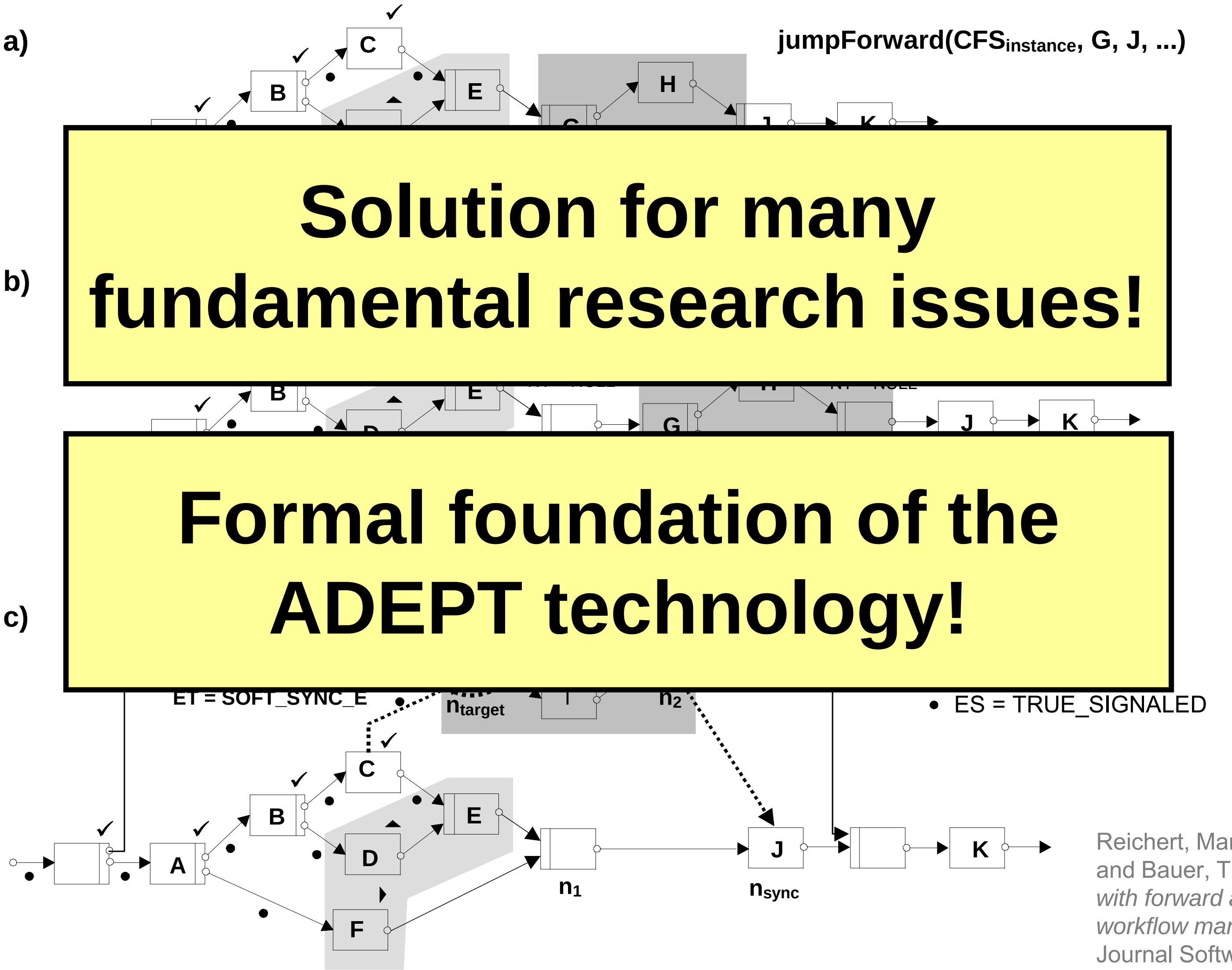
Reichert, Manfred and Dadam, Peter (1998) *ADEPTflex-Supporting Dynamic Changes of Workflows Without Losing Control.* Journal of Intelligent Information Systems, Special Issue on Workflow Management Systems, Vol. 10, No. 2, pp. 93-129

ADEPT: Ad-hoc Changes



Rinderle, Stefanie and Reichert, Manfred (2004) *Correctness Criteria for Dynamic Changes in Workflow Systems: A Survey.* Data & Knowledge Engineering, Vol. 50, No. 1, pp. 9-34.

ADEPT: Ad-hoc Changes



Solution for many fundamental research issues!

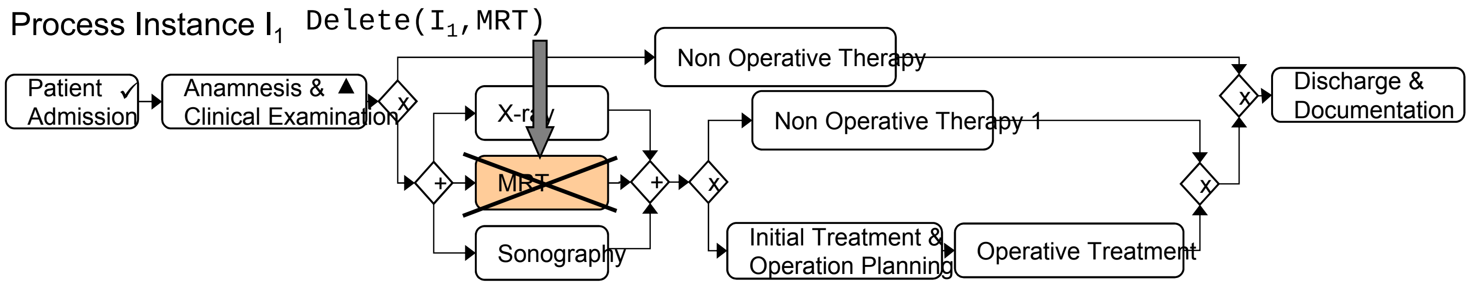
Formal foundation of the ADEPT technology!

Reichert, Manfred and Dadam, Peter and Bauer, Thomas (2003) *Dealing with forward and backward jumps in workflow management systems.* Int'l Journal Software and Systems Modeling (SOSYM), 2(1): 37-58

ADEPT: Ad-hoc Changes

- Annotating changes with information about the reasons for the change
- Retrieval of similar past changes based on context information
- Reuse of changes through PAIS

User Assistance

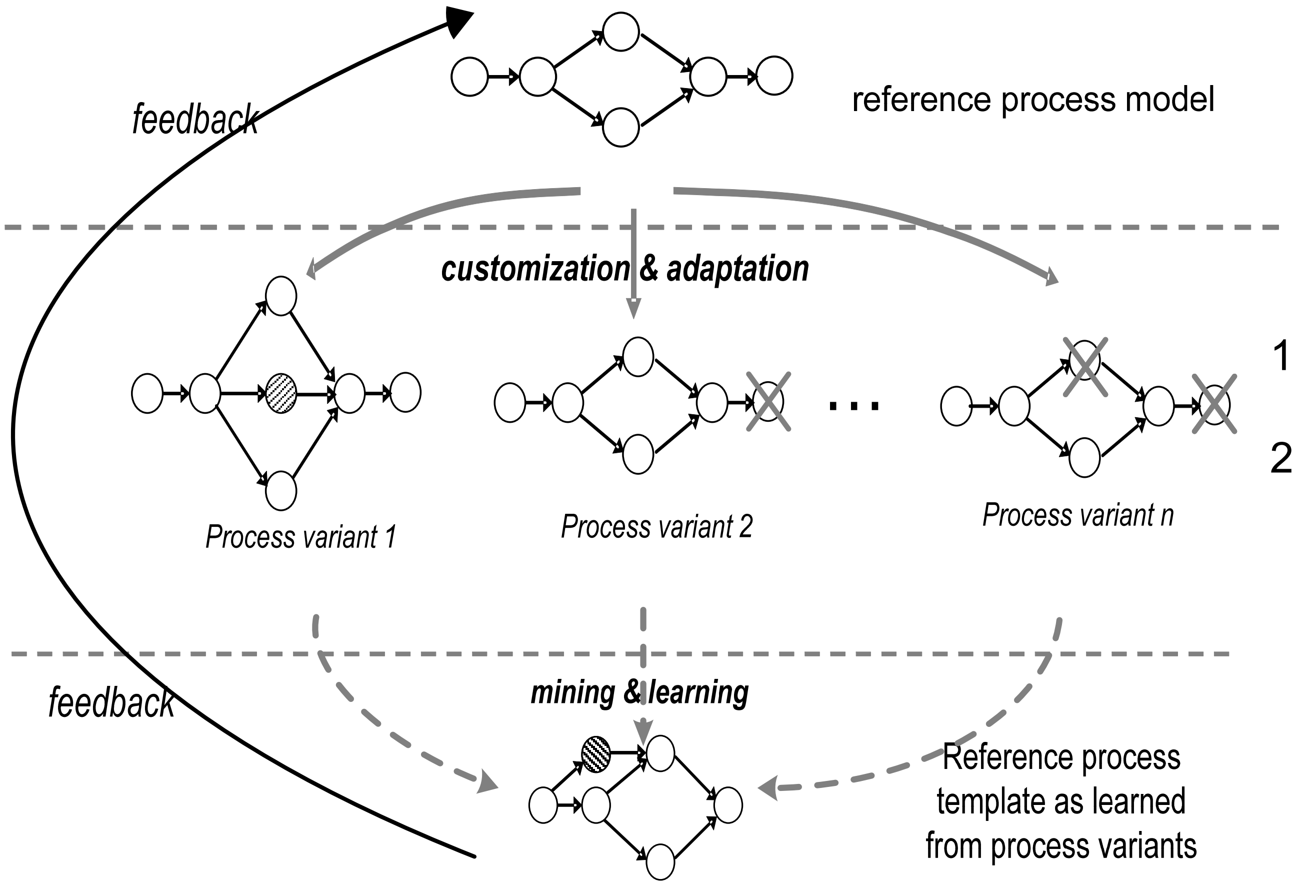



```
pdc1    =  The treatment of cruciate ruptures routinely includes a magnetic
resonance
           tomography (MRT), an X-ray and a sonography. However, for a particular
           patient the MRT may have to be skipped as the respective patient has a
           cardiac pacemaker.
qaSetc1 = {(Does the patient have a cardiac pacemaker?,
              Patient.problemList.hasPacemaker = 'Yes')}
solc1   = <Delete(SI,MRT)>
freqc1  = 1
```

Case c_1

Weber, Barbara and Reichert, Manfred and Wild, Werner and Rinderle-Ma, Stefanie (2009) *Providing Integrated Life Cycle Support in Process-Aware Information Systems.* Int'l Journal of Cooperative Information Systems, Vol. 18, No. 1, pp. 115-165

ADEPT: Change Mining and Learning

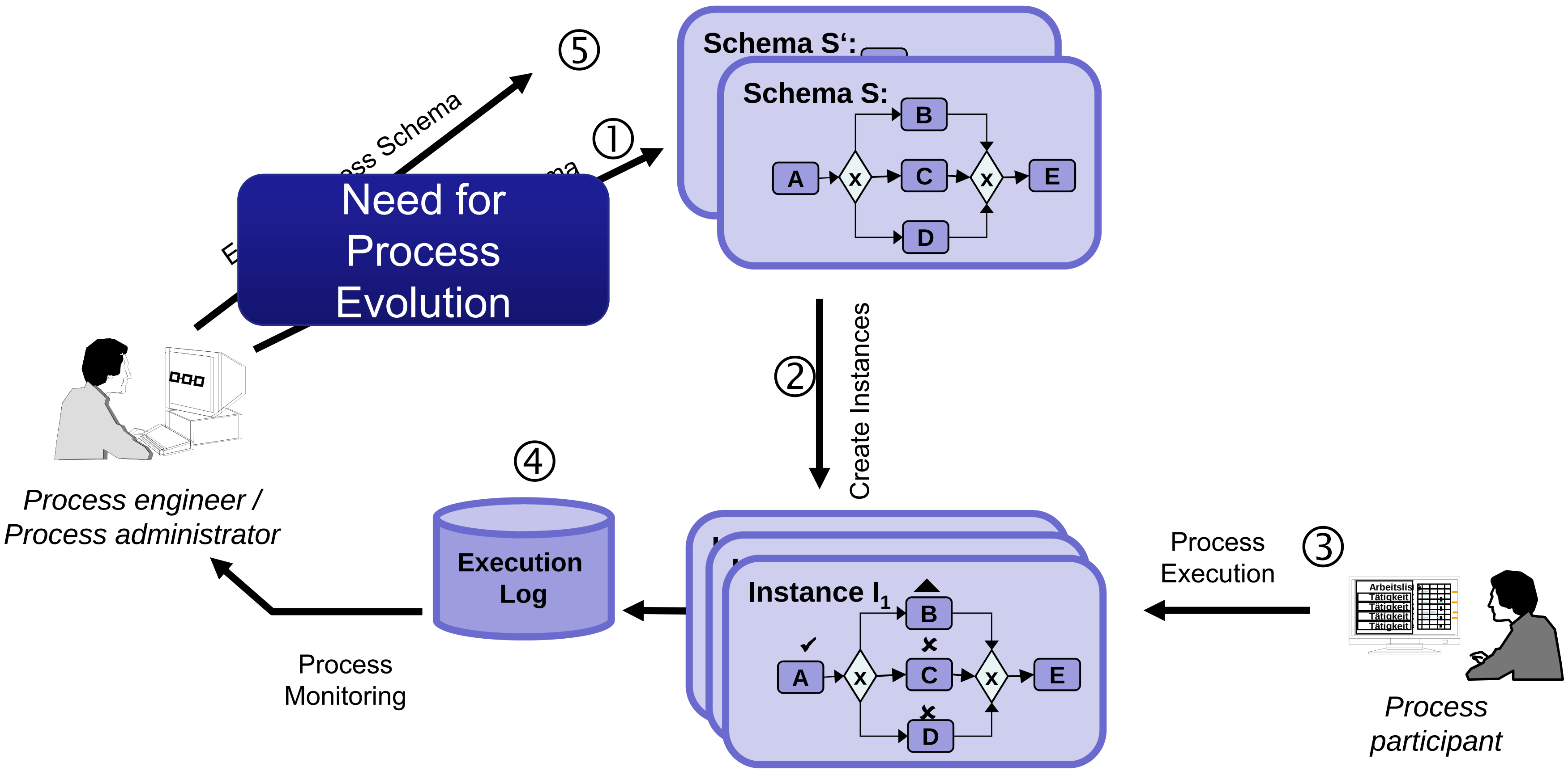


1. Changes are rather costly
2. Variants are difficult to maintain

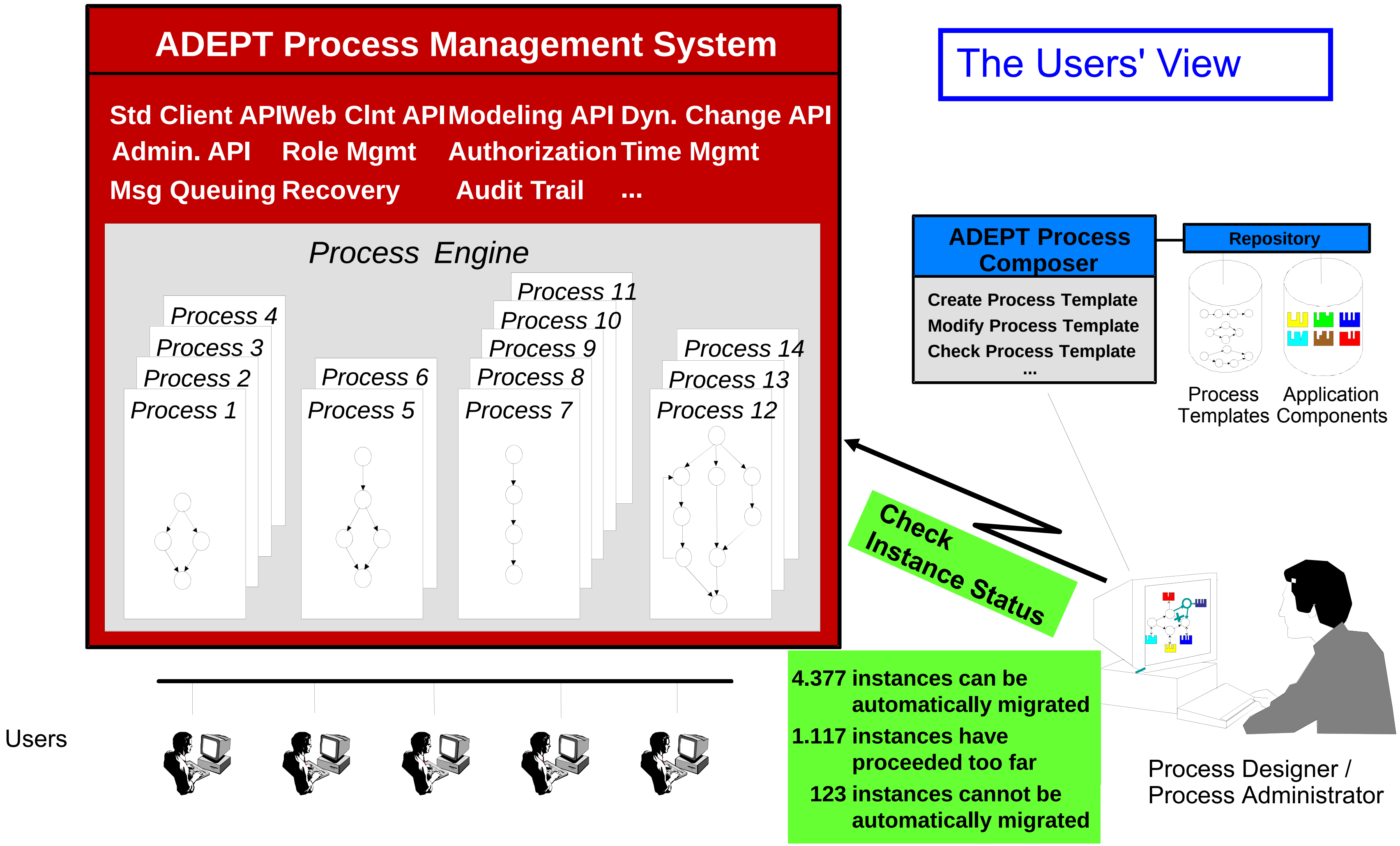
Derive a new reference process model from the the variants such that:
Less adaptations are needed in future!

Li, C., Reichert, M., Wombacher, A. (2011) *Mining Business Process Variants: Challenges, Scenarios, Algorithms.* Data & Knowledge Engineering, Vol. 70, No. 5, pp. 409-434,

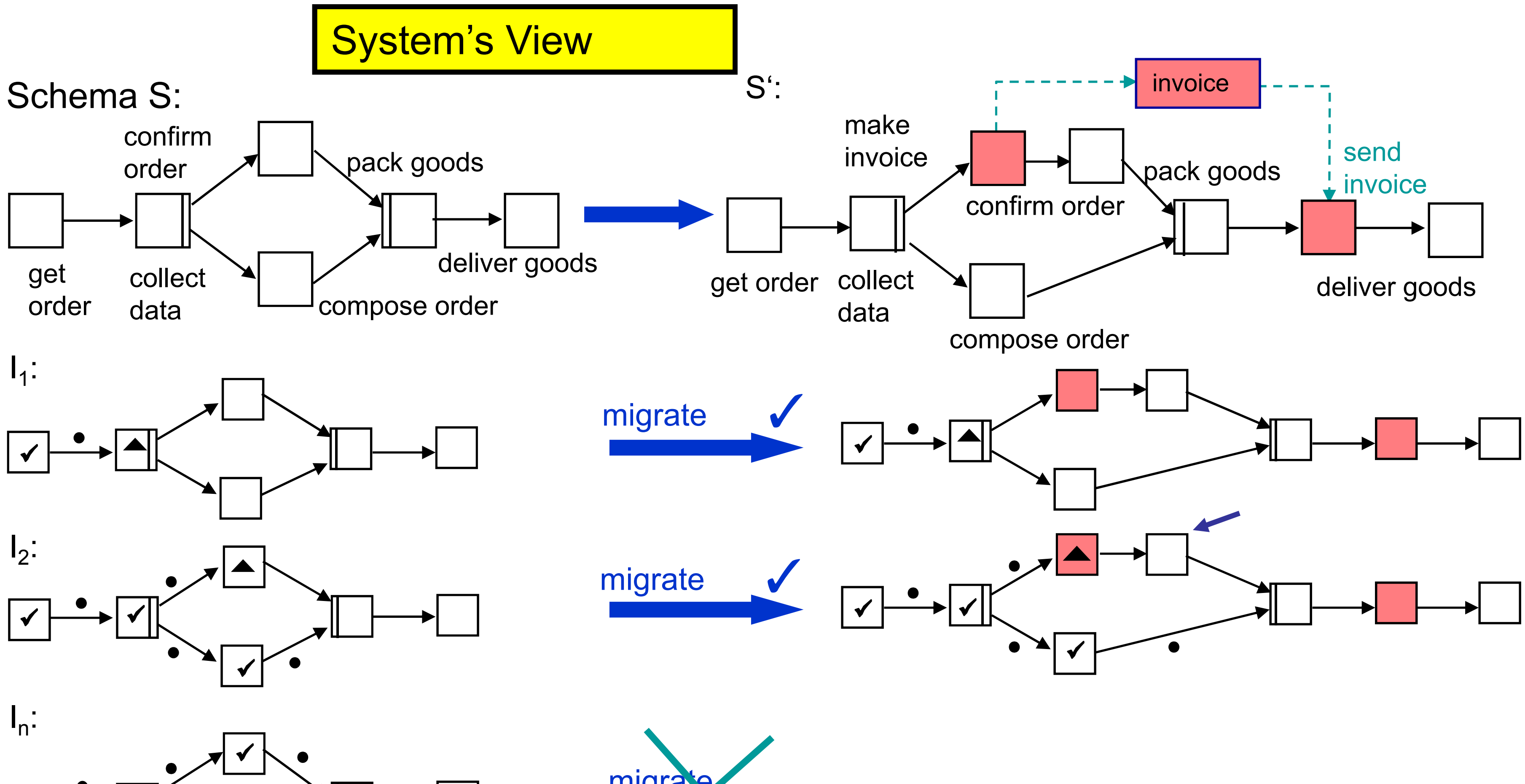
ADEPT: Process Schema Evolution



ADEPT: Process Schema Evolution



ADEPT: Process Schema Evolution

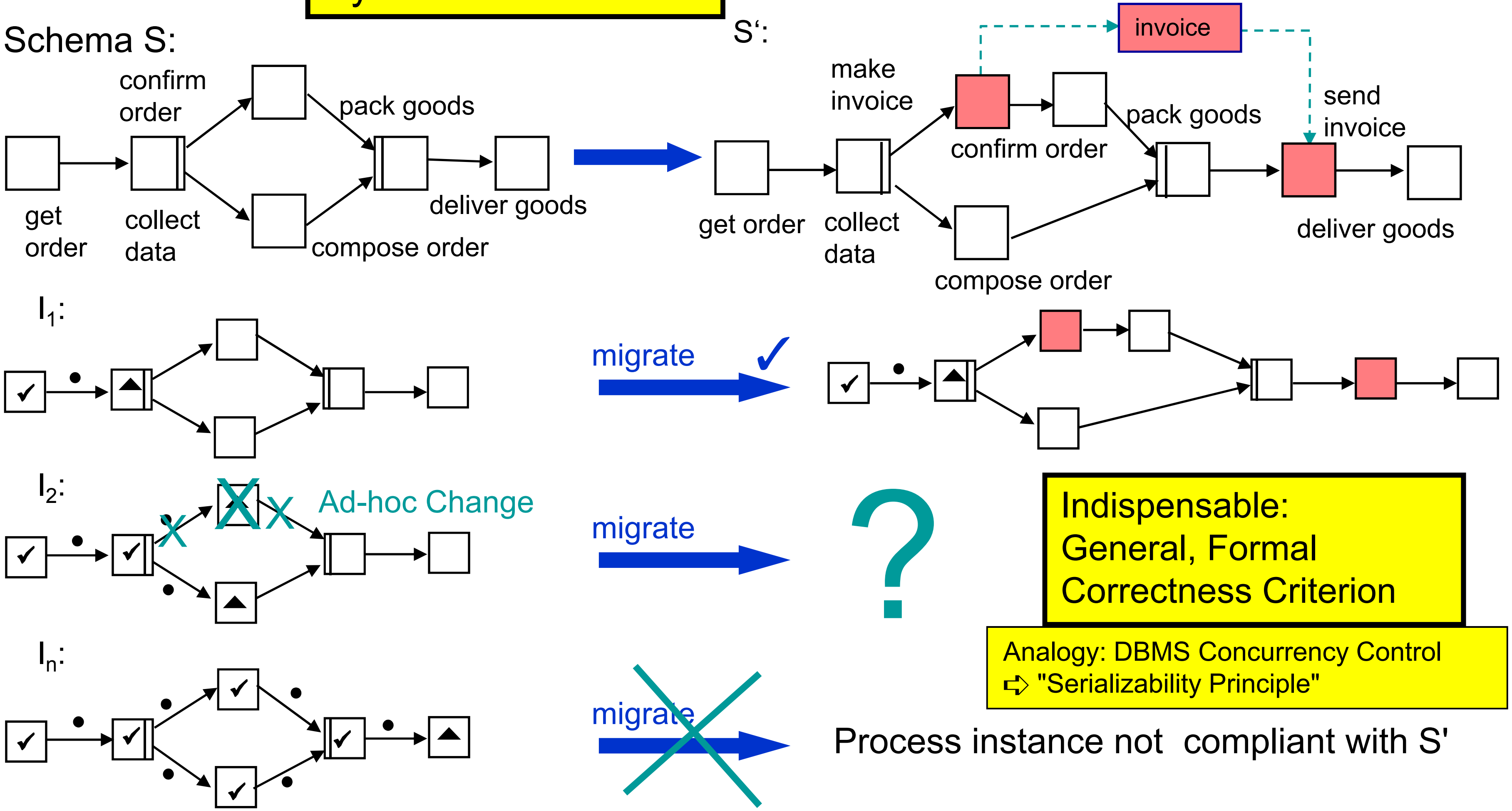


Process instance not state compliant with S' (remains being executed according to S)

Rinderle, S., Reichert, M., Dadam, Peter (2004) *Flexible Support of Team Processes by Adaptive Workflow Systems.* Distributed and Parallel Databases, 16(1): 91-116

ADEPT: Process Schema Evolution

System's View



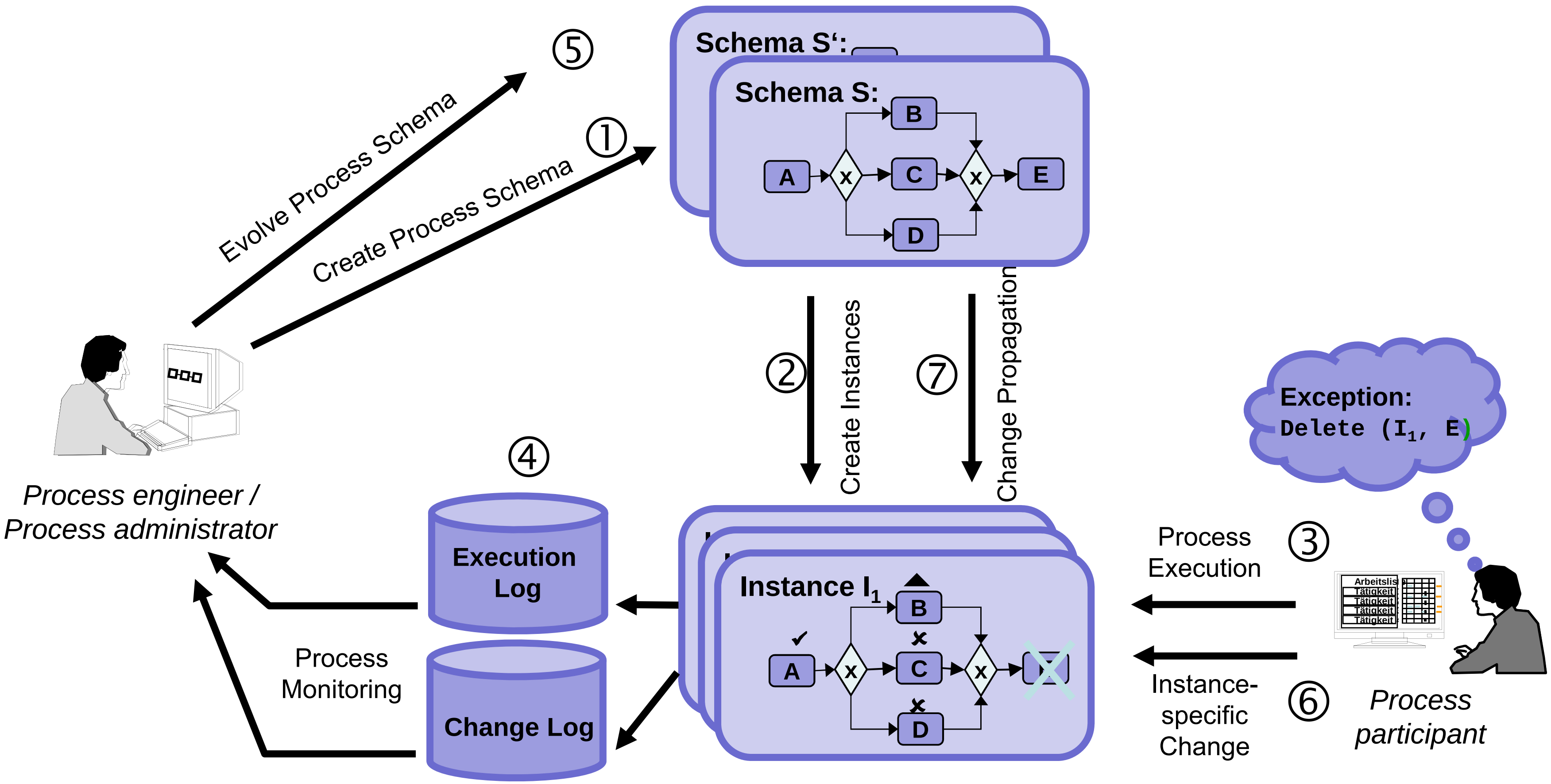
Indispensable: General, Formal Correctness Criterion

Analogy: DBMS Concurrency Control
➪ "Serializability Principle"

Process instance not compliant with S'

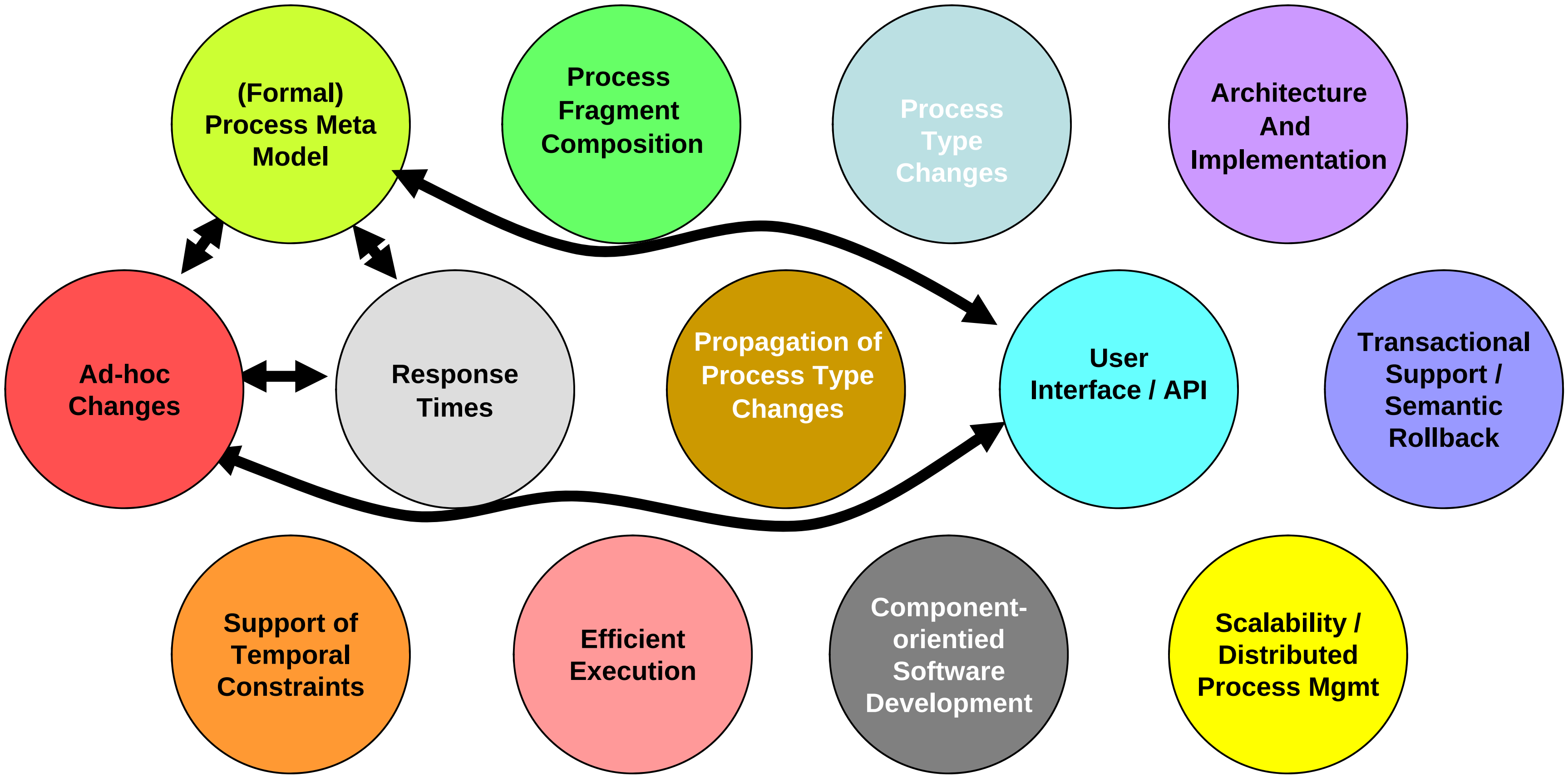
Rinderle, S., Reichert, M., Dadam, P. (2004) *Disjoint and Overlapping Process Changes: Challenges, Solutions, Applications.* In: Proc. CoopIS'04, Agia Napa, LNCS 3290,, pp. 101-121

ADEPT: Extended Process Lifecycle Support



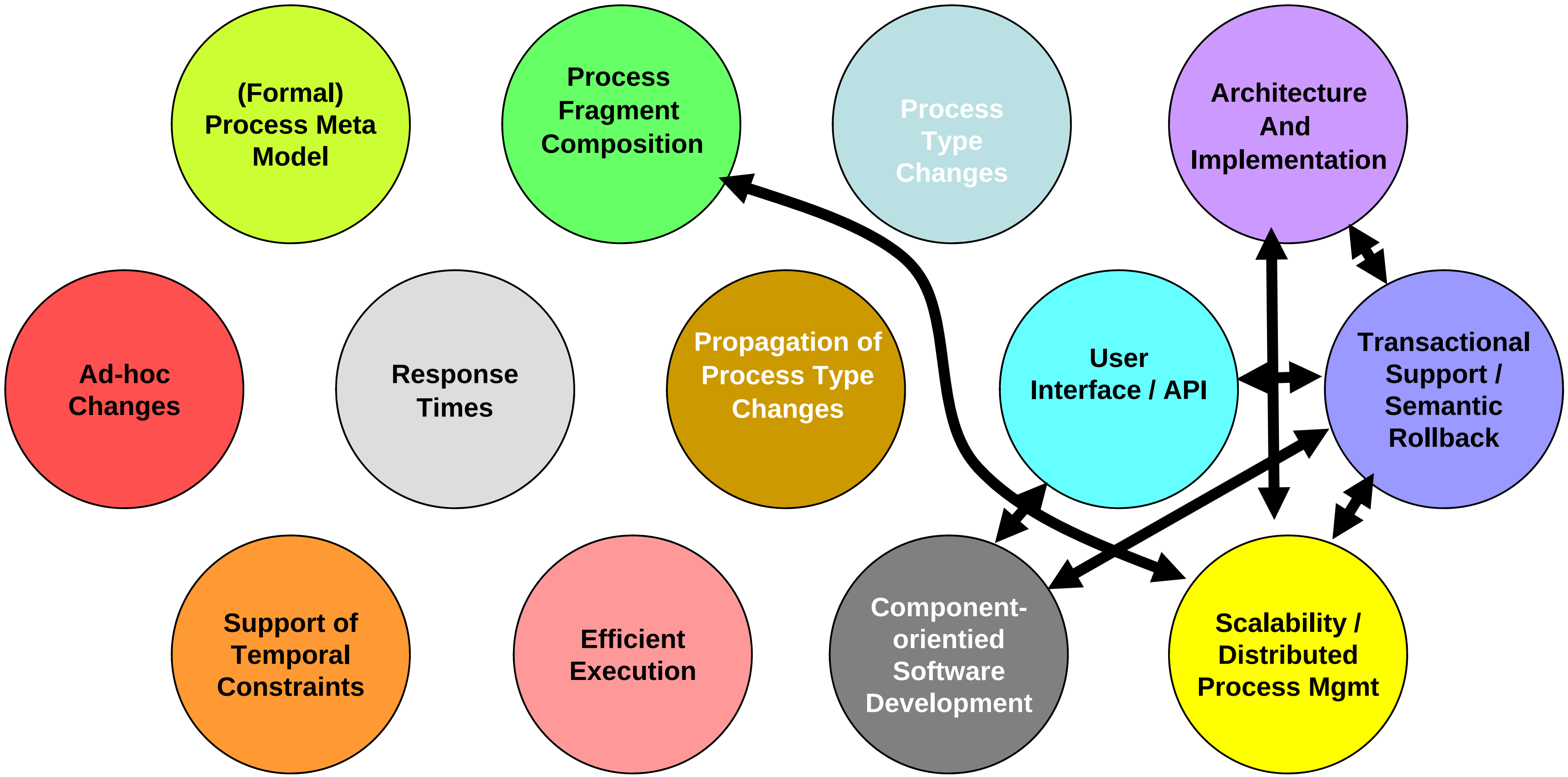
Weber, B., Sadiq, S., Reichert, M. (2009) *Beyond Rigidity - Dynamic Process Lifecycle Support: A Survey on Dynamic Changes in Process-aware Information Systems.* Computer Science - Research and Development, 23(2): 47-65, Springer.

ADEPT: Implementing the Framework



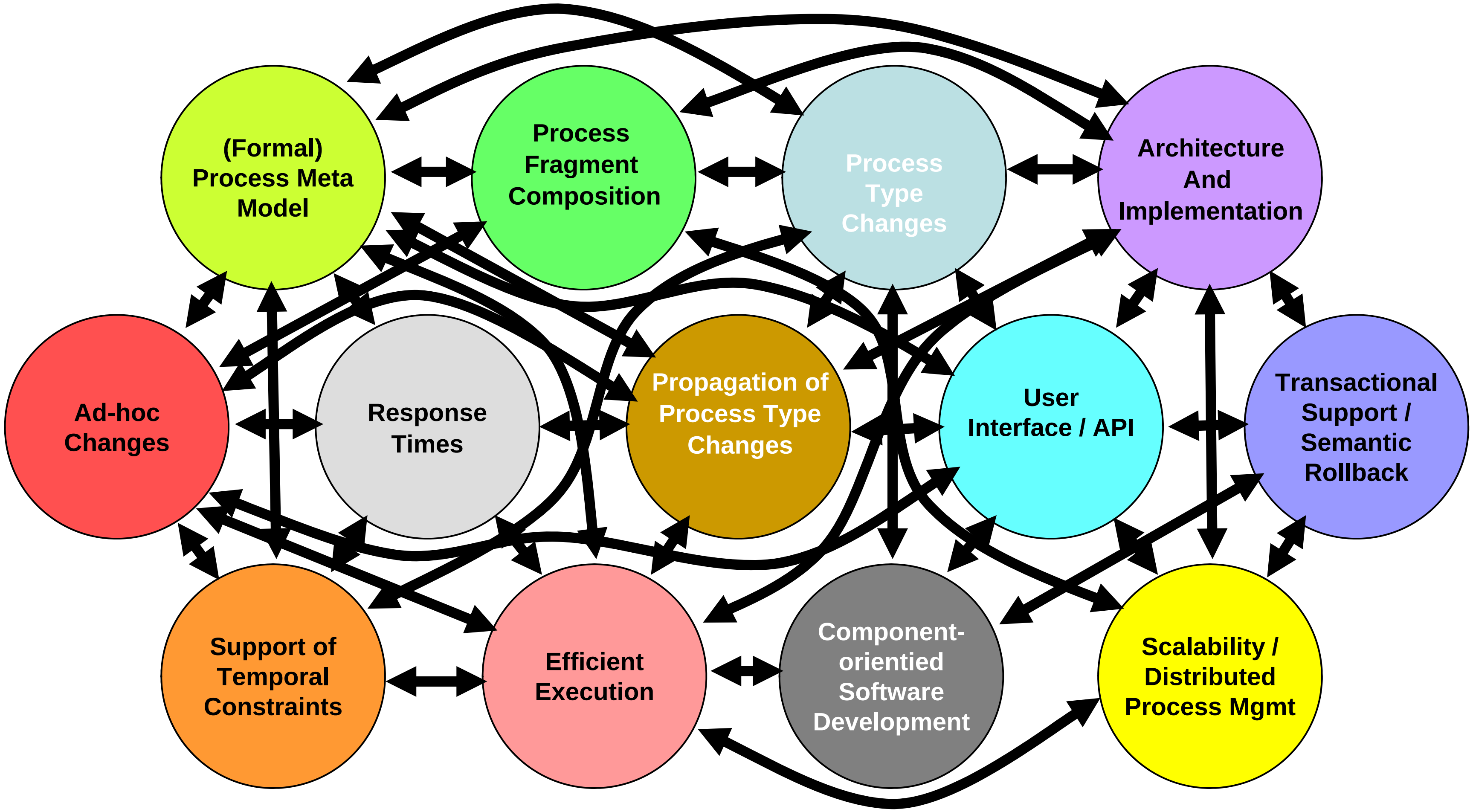
Dadam, Peter and Reichert, Manfred (2009) *The ADEPT Project: A Decade of Research and Development for Robust and Flexible Process Support - Challenges and Achievements.* Computer Science - Research and Development, Vol. 23, No. 2, pp. 81-97.

ADEPT: Implementing the Framework



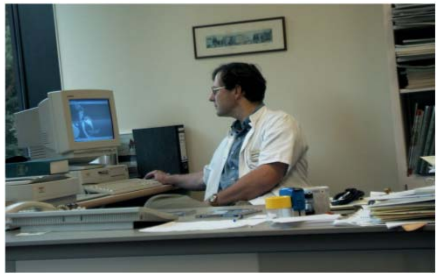
Dadam, Peter and Reichert, Manfred (2009) *The ADEPT Project: A Decade of Research and Development for Robust and Flexible Process Support - Challenges and Achievements.* Computer Science - Research and Development, Vol. 23, No. 2, pp. 81-97.

ADEPT: Implementing the Framework



Dadam, Peter and Reichert, Manfred (2009) *The ADEPT Project: A Decade of Research and Development for Robust and Flexible Process Support - Challenges and Achievements.* Computer Science - Research and Development, Vol. 23, No. 2, pp. 81-97.

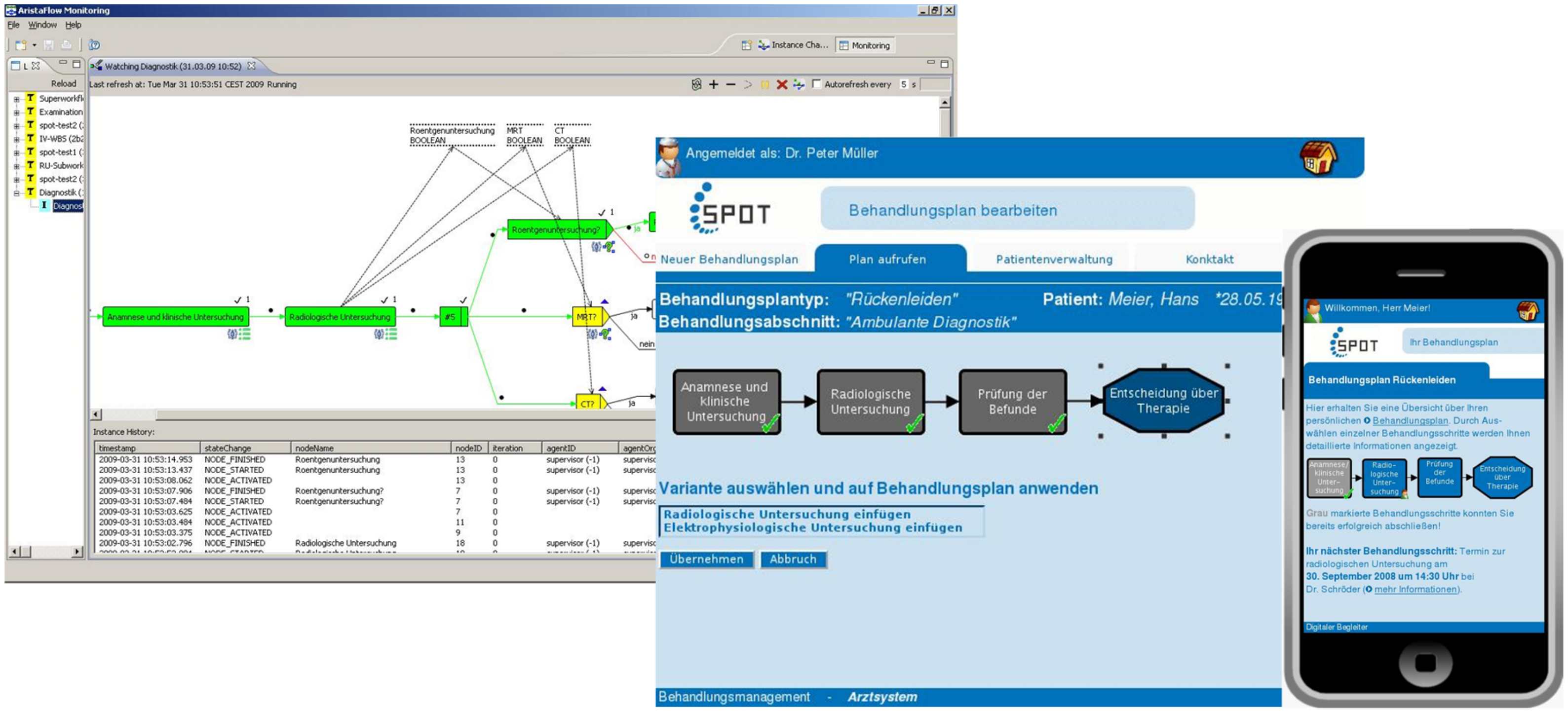
ADEPT: Clinical Pathway Support



Flexible Support of Clinical Pathways with ADEPT

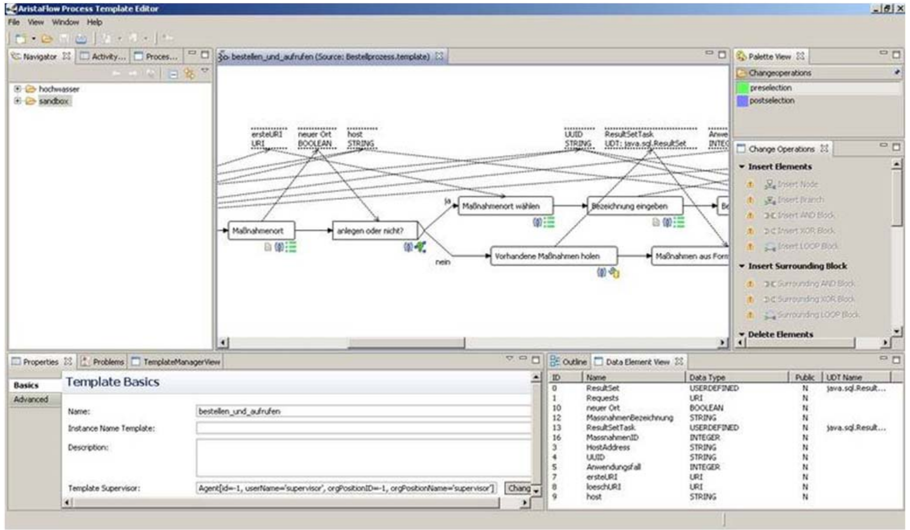
Partners:

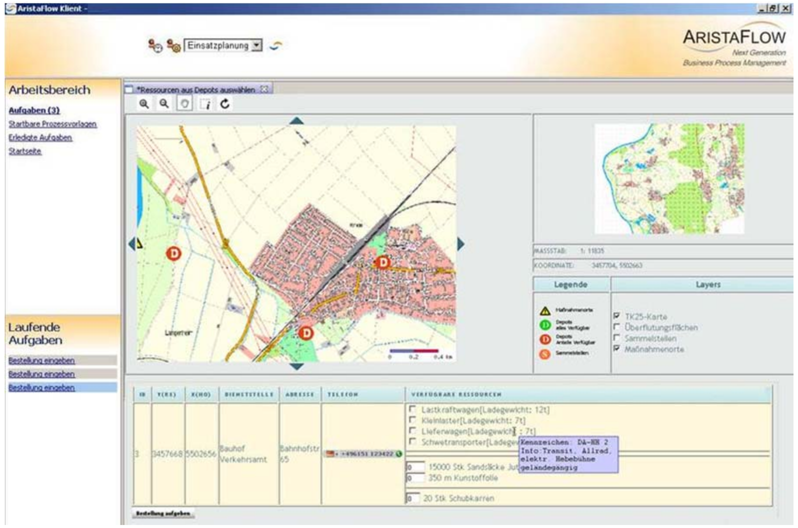
Jan Neuhaus, Claudia Reuter
Fraunhoferinstitut Dortmund



ADEPT: Disaster Management

Process-aware, Cooperative Emergency Management for Water Infrastructures
Partner: TU Darmstadt



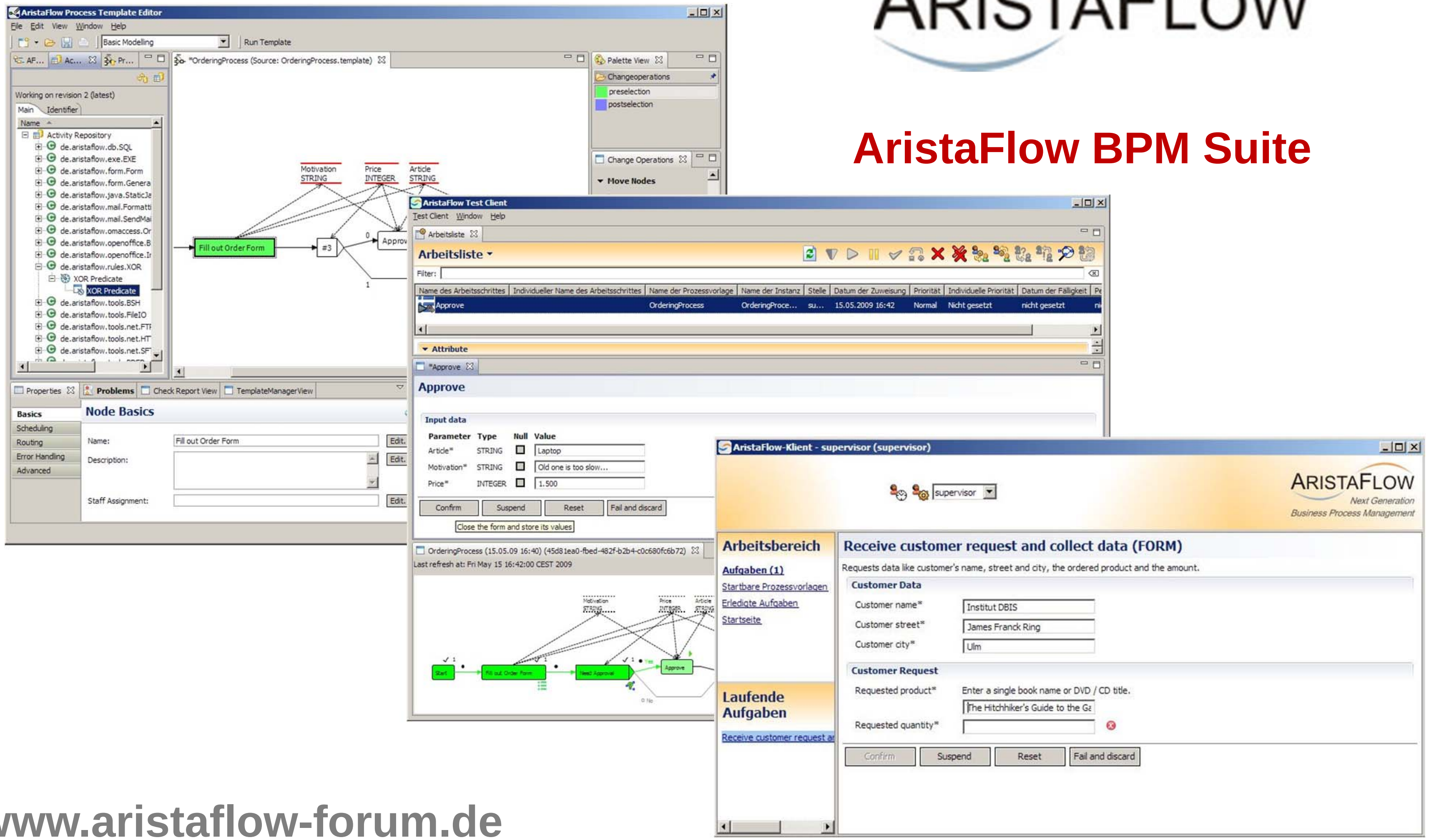


A. Wagenknecht; U. Rüppel: Improving Resource Management In Flood Response With Process Models and Web GIS. In: 16th TIEMS Conf., 2009

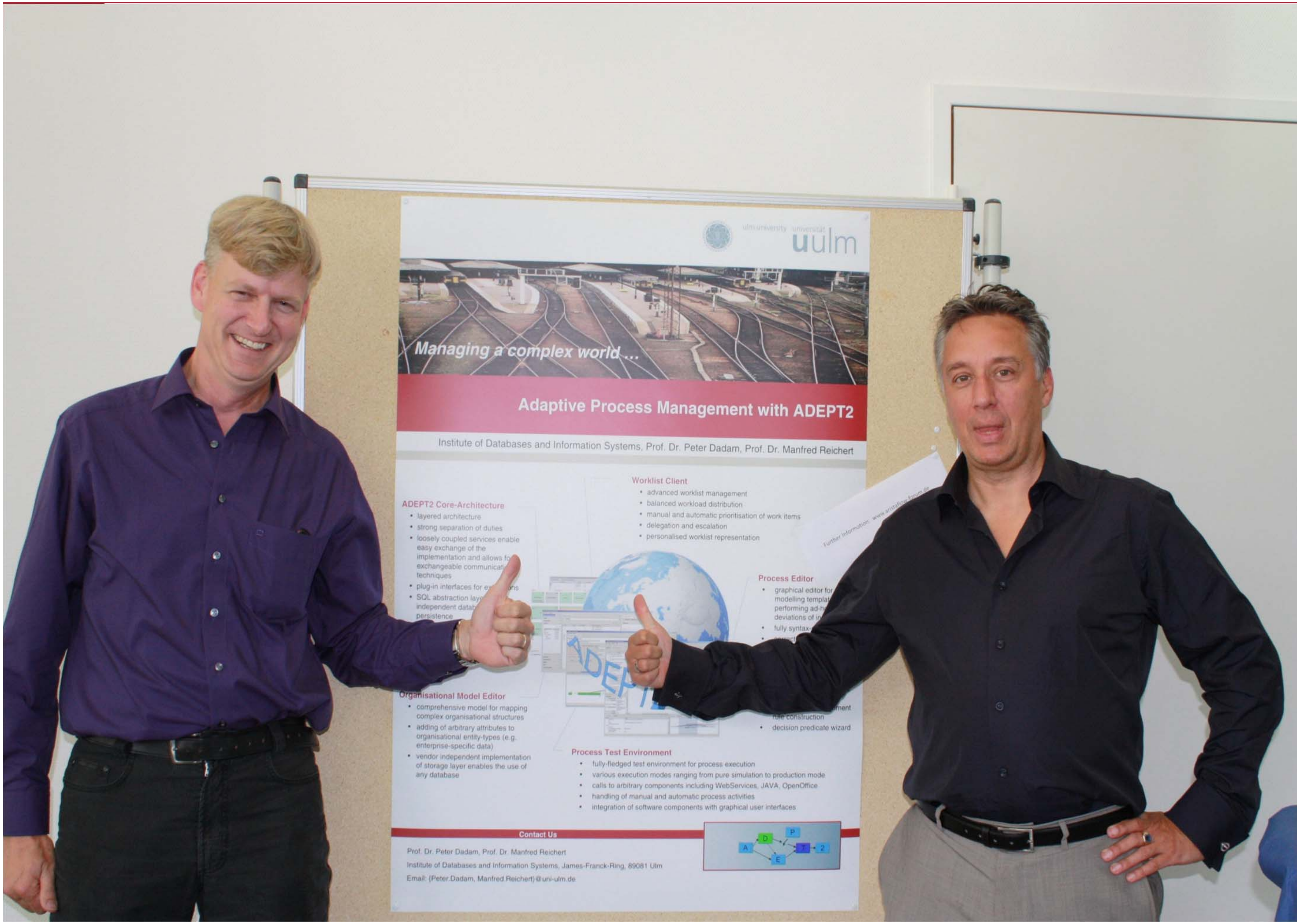
ADEPT: Transfering ADEPT to Practice
The AristaFlow BPM Suite

ARISTAFLOW

AristaFlow BPM Suite

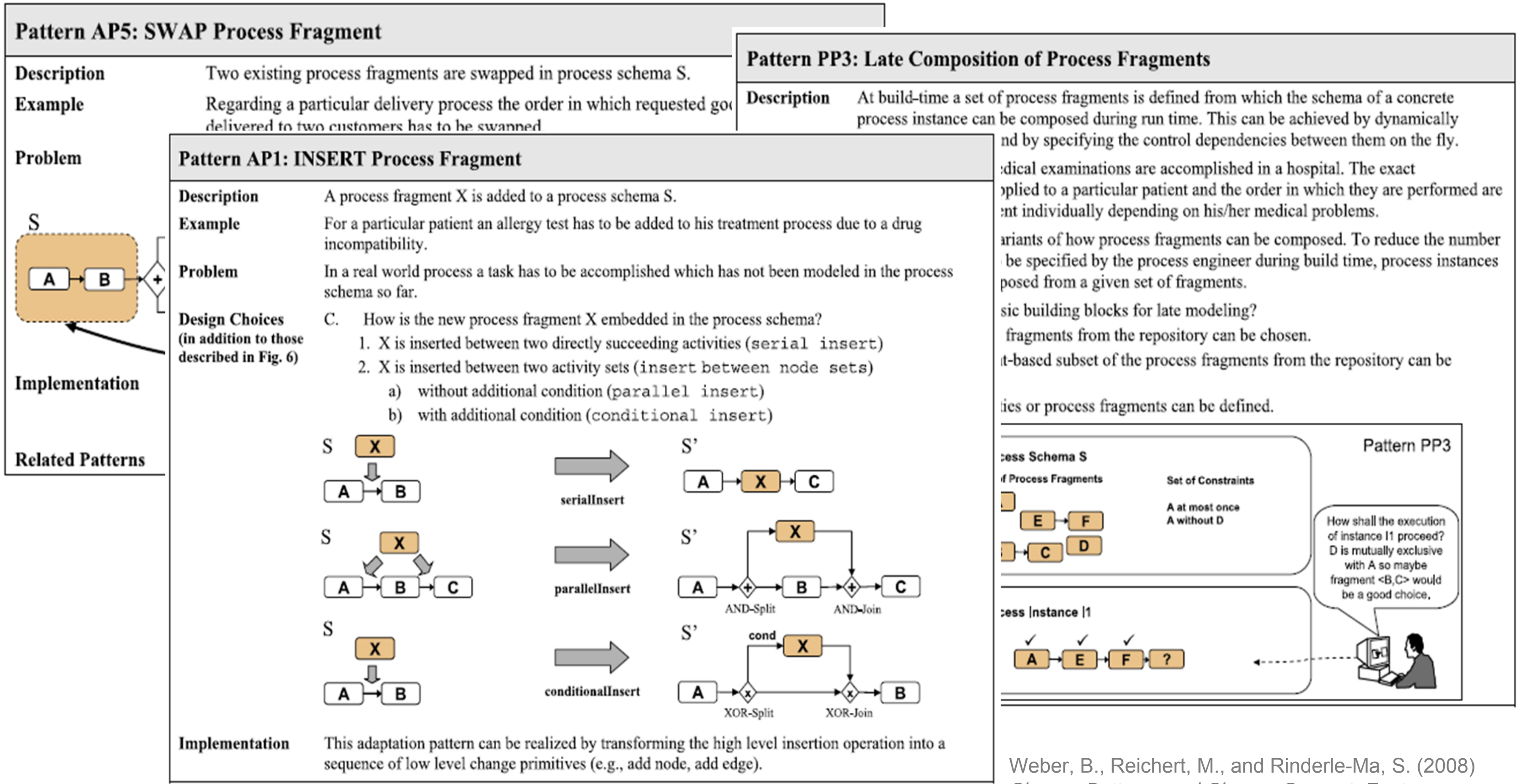


www.aristaflow-forum.de


uulm
Managing a complex world ...
Adaptive Process Management with ADEPT2
Institute of Databases and Information Systems, Prof. Dr. Peter Dadam, Prof. Dr. Manfred Reichert
Worklist Client
advanced worklist management
balanced workload distribution
manual and automatic prioritisation of work items
delegation and escalation
personalised worklist representation
ADEPT2 Core-Architecture
layered architecture
strong separation of duties
Process Editor
comprehensive model for mapping complex organisational structures
adding of arbitrary attributes to organisational entity-types (e.g. enterprise-specific data)
vendor independent implementation of storage layer enables the use of any database
decision predicate wizard
Process Test Environment
fully-fledged test environment for process execution
various execution modes ranging from pure simulation to production mode
calls to arbitrary components including WebServices, JAVA, OpenOffice
handling of manual and automatic process activities
integration of software components with graphical user interfaces
Contact Us
Prof. Dr. Peter Dadam, Prof. Dr. Manfred Reichert
Institute of Databases and Information Systems, James-Franck-Ring, 89081 Ulm
Email: {Peter.Dadam, Manfred.Reichert}@uni-ulm.de

Another Contribution: Comparing PAIS Flexibility Frameworks (1)

Change Patterns



Pattern AP5: SWAP Process Fragment

Description Two existing process fragments are swapped in process schema S.

Example Regarding a particular delivery process the order in which requested go[ods are] delivered to two customers has to be swapped

Problem

Implementation

Related Patterns

Pattern PP3: Late Composition of Process Fragments

Description At build-time a set of process fragments is defined from which the schema of a concrete process instance can be composed during run time. This can be achieved by dynamically [...]nd by specifying the control dependencies between them on the fly.

[...]dical examinations are accomplished in a hospital. The exact [...]plied to a particular patient and the order in which they are performed are [...]nt individually depending on his/her medical problems.

[...]ariants of how process fragments can be composed. To reduce the number [...] be specified by the process engineer during build time, process instances [...]posed from a given set of fragments.

[...]sic building blocks for late modeling?

[...] fragments from the repository can be chosen.

[...]t-based subset of the process fragments from the repository can be

[...]ies or process fragments can be defined.

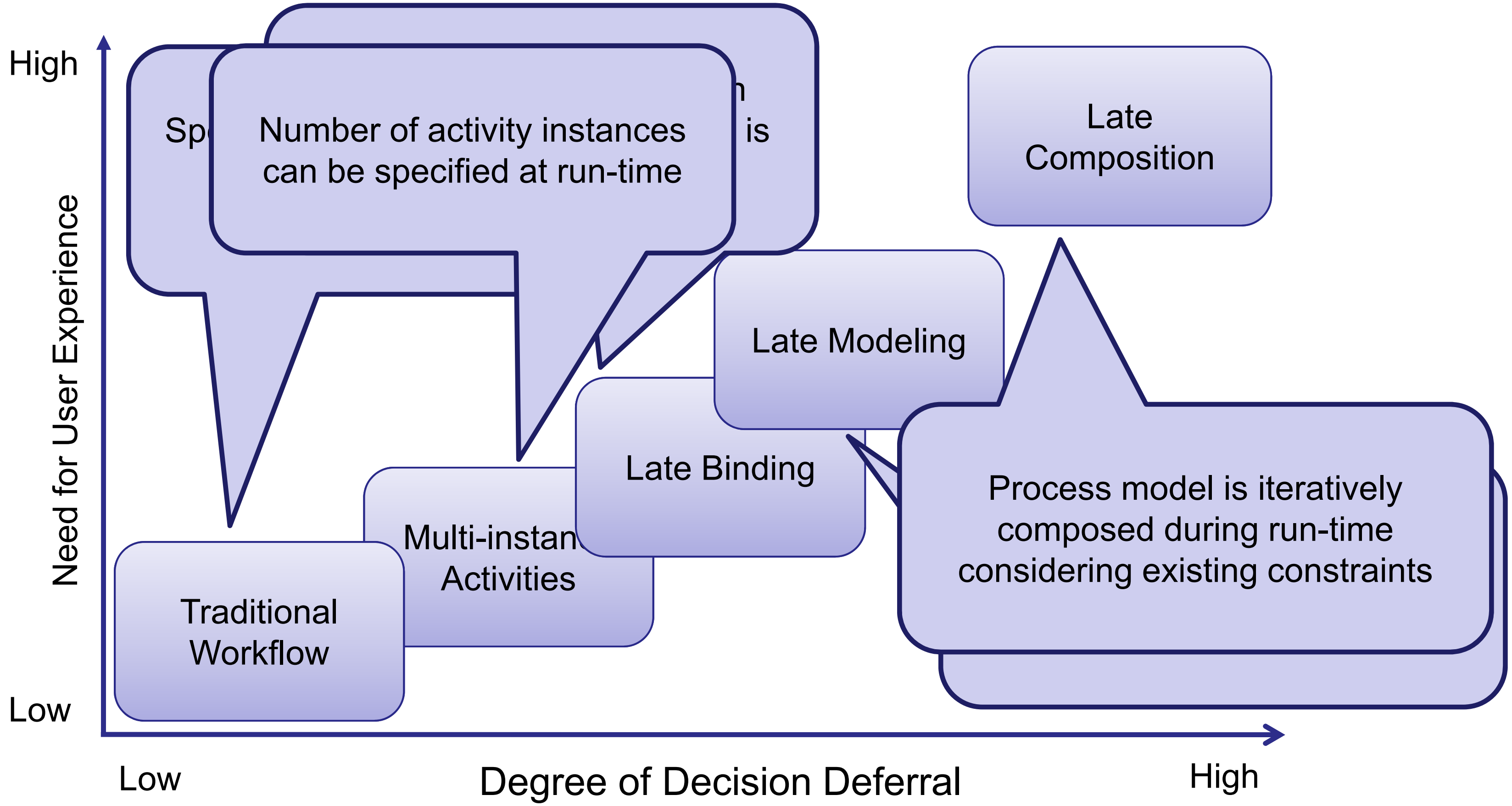
Pattern AP1: INSERT Process Fragment

Description	A process fragment X is added to a process schema S.
Example	For a particular patient an allergy test has to be added to his treatment process due to a drug incompatibility.
Problem	In a real world process a task has to be accomplished which has not been modeled in the process schema so far.
Design Choices (in addition to those described in Fig. 6)	C. How is the new process fragment X embedded in the process schema? 1. X is inserted between two directly succeeding activities (`serial insert`) 2. X is inserted between two activity sets (`insert between node sets`) a) without additional condition (`parallel insert`) b) with additional condition (`conditional insert`)
Implementation	This adaptation pattern can be realized by transforming the high level insertion operation into a sequence of low level change primitives (e.g., add node, add edge).

Weber, B., Reichert, M., and Rinderle-Ma, S. (2008)
Change Patterns and Change Support Features – Enhancing Flexibility in Process-Aware Information Systems.
Data & Knowledge Eng, 66(3): 438-466,

Another Contribution: Comparing PAIS Flexibility Frameworks (2)

Patterns for Decision Deferral



Another Contribution: Comparing PAIS Flexibility Frameworks (3)

Change Support Features

- Schema Evolution, Version Control and Instance Migration
- Support for Instance-Specific Changes
- Correctness of Changes
- Traceability and Analysis of Changes
- Access Control of Changes
- Change Reuse
- Change Concurrency Control
- Refactoring Support for Process Models

Another Contribution: Comparing PAIS Flexibility Frameworks (4)

Primitive / Pattern	Academic								Commercial	
	ADEPT2 / CBRFlow	CAKE 2	HOON	MOVE	PoF	WASA2	WIDE	YAWL + Worklets / Exlets	Flower	Staffware
Change Primitives										
PR1 – Add Node	–	+	+	+	+	+	+	+	+	+
PR2 – Remove Node	–	+	+	+	+	+	+	+	+	+
PR3 – Add Edge	–	+	+	+	+	+	+	+	+	+
PR4 – Remove Edge	–	+	+	+	+	+	+	+	+	+
PR5 – Move Edge	–	+	–	–	–	–	–	+	–	–
Adaptation Patterns										
AP1 – Insert Fragment	A[1, 2], B[1,2,3], C [1, 2]	–	–	–	–	–	A[2], B[1], C[1,2]	–	–	–
AP2 – Delete Fragment	A[1, 2], B[1,2,3]	–	–	–	–	–	A[2], B[1]	–	–	–
AP3 - Move Fragment	A[1, 2], B[1,2,3], C[1,2]	–	–	–	–	–	–	–	–	–
AP4 – Replace Fragment	–	–	–	–	–	–	A[2], B[1]	–	–	–
AP5 – Swap Fragment	–	–	–	–	–	–	–	–	–	–
AP6 – Extract Fragment	A[1,2], B[3]	–	–	–	–	–	–	–	–	–
AP7 – Inline Fragment	A[1,2], B[2]	–	–	–	–	–	–	–	–	–
AP8 – Embed Fragment in	A[1,2], B[1,2,3]	–	–	–	–	–	–	–	–	–
AP9 – Parallelize Activities	A[1,2], B[1,2,3]	–	–	–	–	–	–	–	–	–
AP10 - Embed Fragment in Conditional Branch	–	–	–	–	–	–	A[2]	–	–	–
AP11 – Add Control Dependency	A[1,2]	–	–	–	–	–	–	–	–	–
AP12 – Remove Control Dependencies	A[1,2]	–	–	–	–	–	–	–	–	–
AP13 – Update Condition	A[1,2]	–	–	–	–	–	A[2]	–	–	–
AP14 – Copy Fragment	–	–	–	–	–	–	–	–	–	–

Weber, Barbara and Rinderle, Stefanie and Reichert, Manfred (2007) *Change Patterns and Change Support Features in Process-Aware Information Systems.* In: CAiSE'07, Trondheim, Norway, LNCS 4495, pp. 574-588.

Introduction

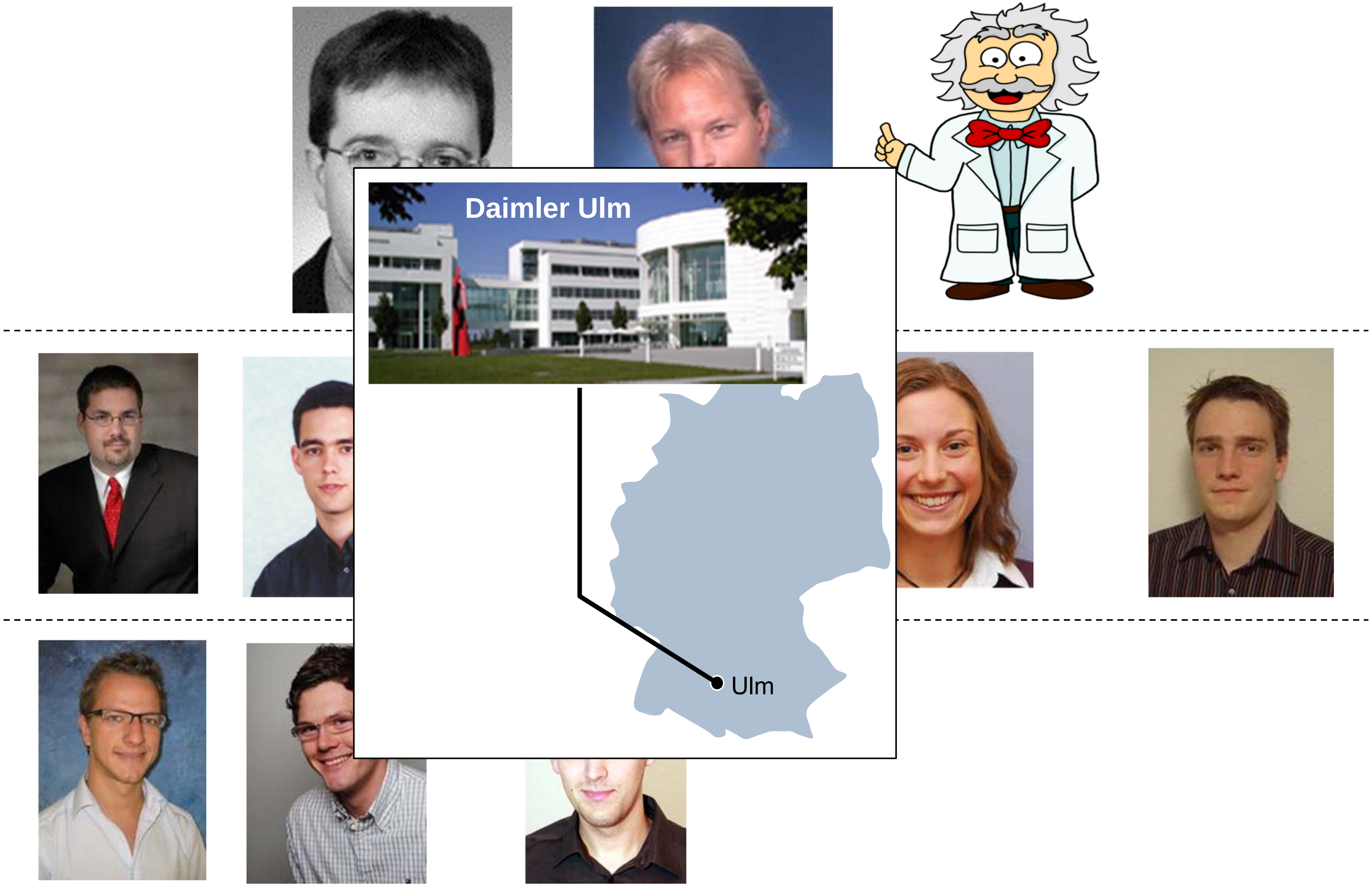
ADEPT – A Decade of Research on Flexible Process-Aware Information Systems and its Transfer to Practice

Large Processes in the Automotive Industry and Technologies to Support them

- Large Process Models
- Large Process Collections
- Large Process Structures

Conclusions

The Daimler BPM Round Table



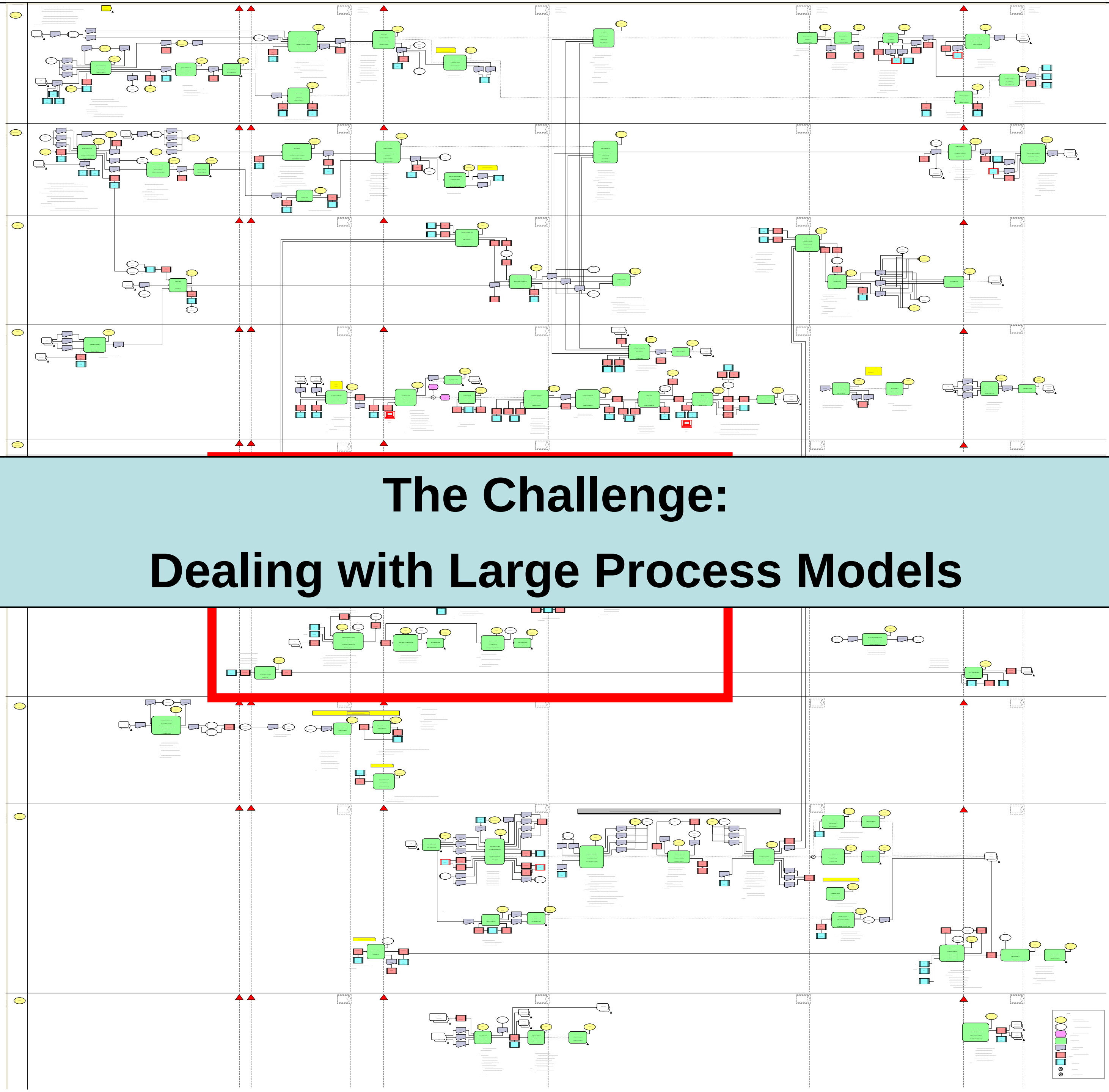
Introduction

ADEPT – A Decade of Research on Flexible Process-Aware Information Systems and its Transfer to Practice

Large Processes in the Automotive Industry and Technologies to Support them

- Large Process Models
- Large Process Collections
- Large Process Structures

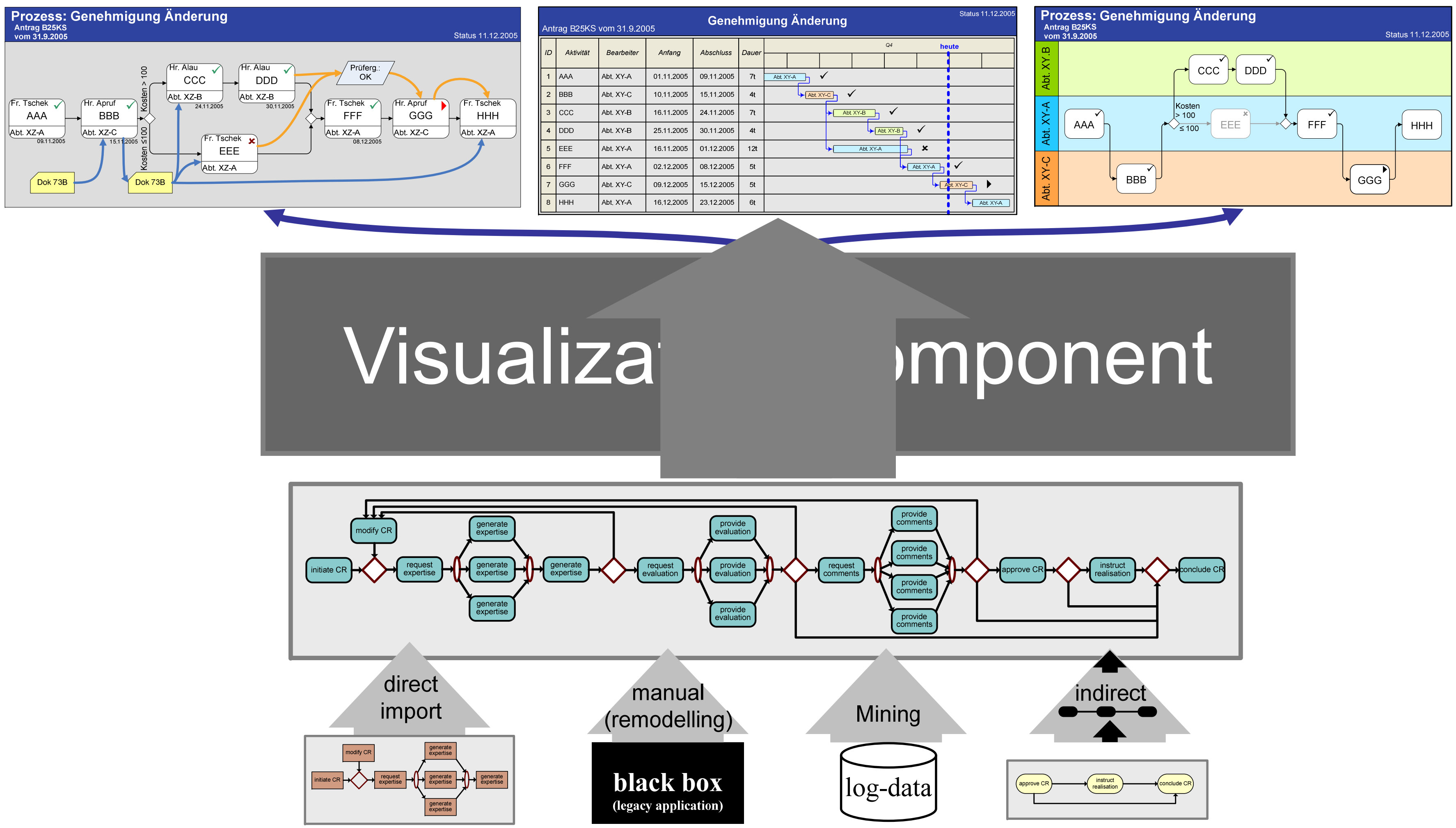
Conclusions



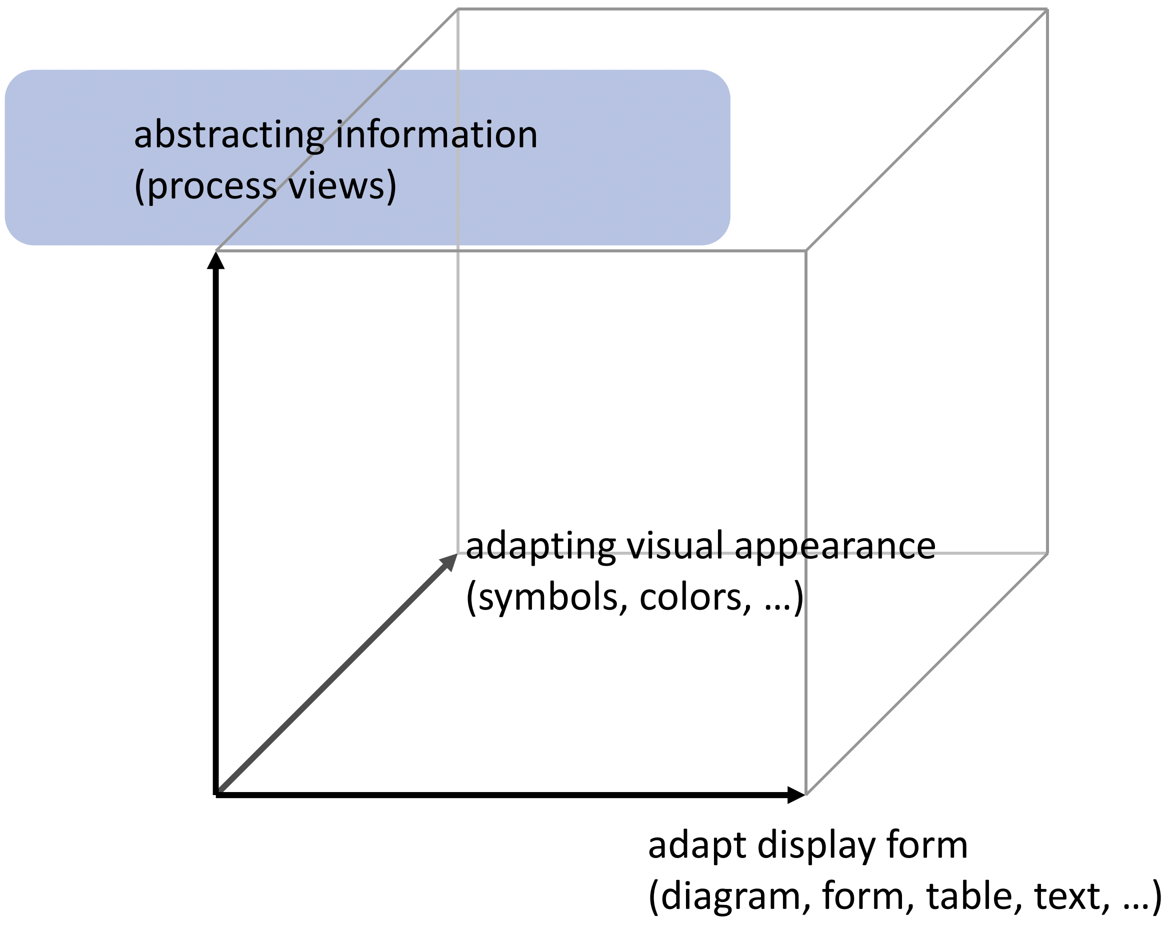
The Challenge:

Dealing with Large Process Models

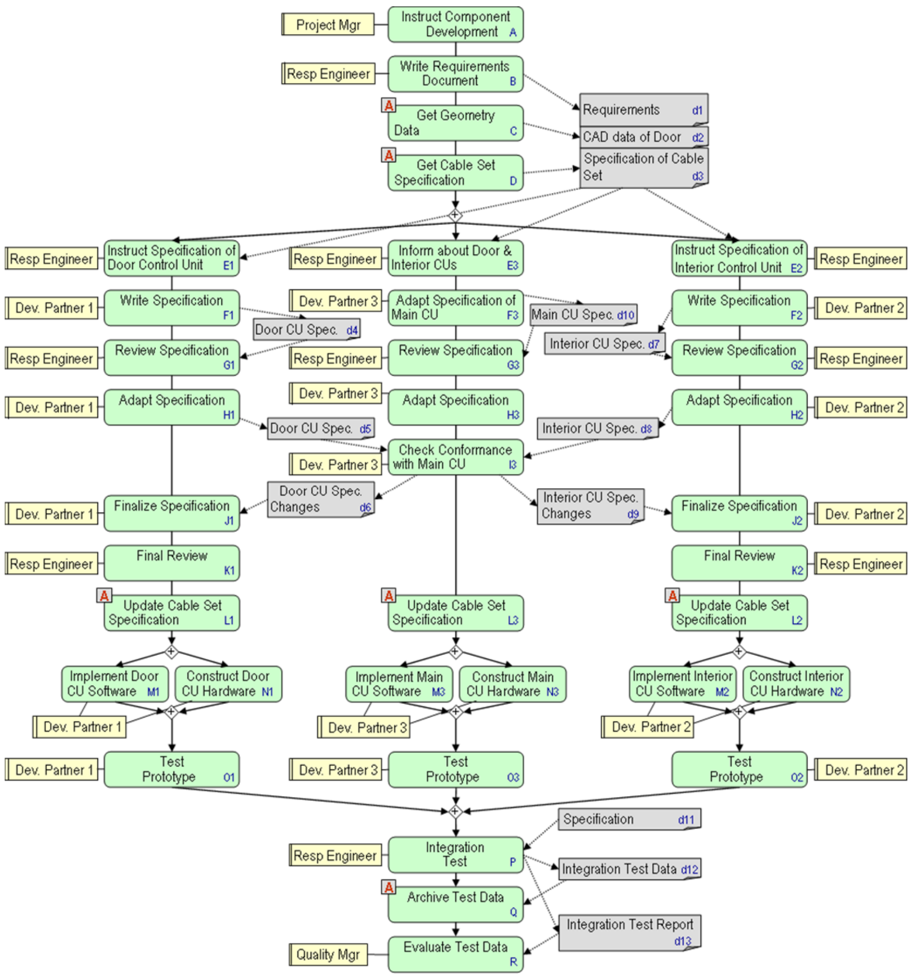
Dealing with Large Process Models: Need for an Advanced Visualization Framework

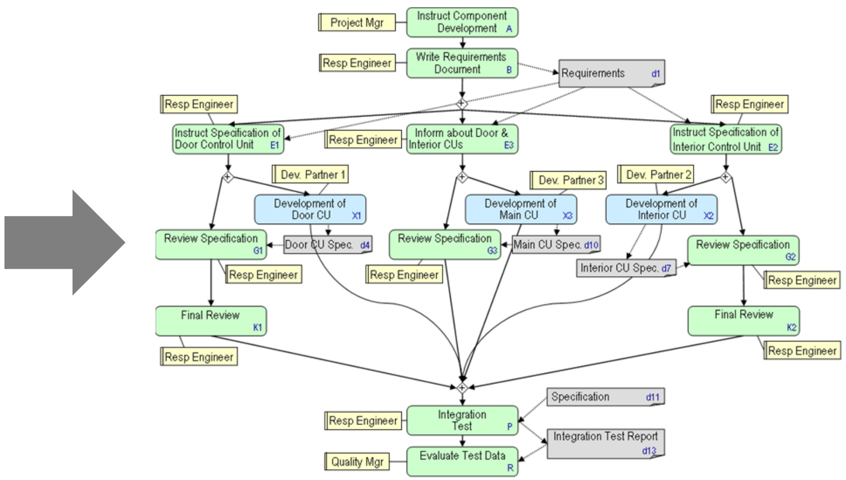


The Proviado Visualization Framework

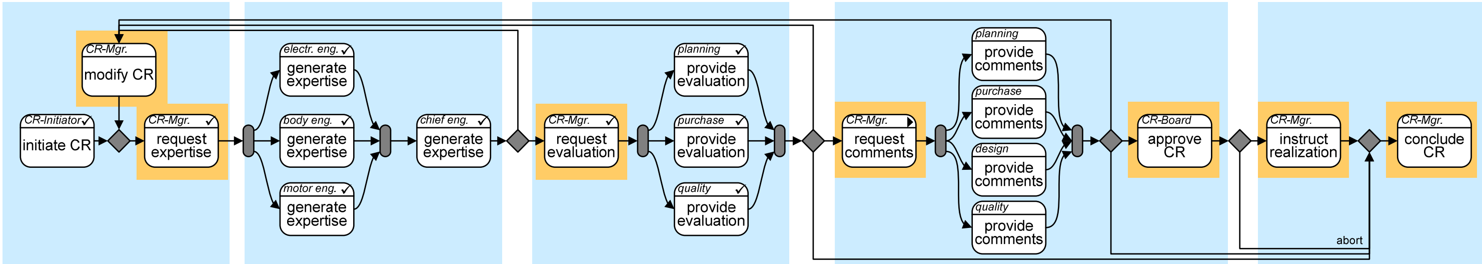


Proviado: Process Model Abstraction - Example



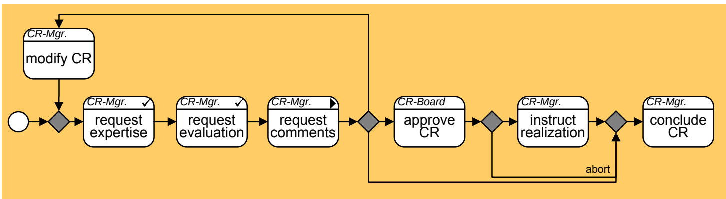


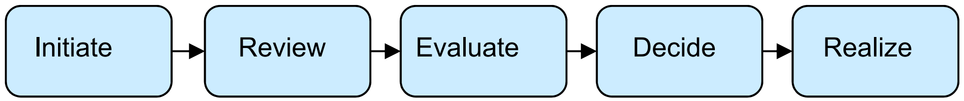
Proviado: Process Model Abstraction – Basic Operations (1)



CREATE REDUCED VIEW cr-manager AS
SELECT FROM cr-process p
WHERE p.activity.actor = „CR-Mgr."

CREATE VIEW cr-overview AS
AGGREGATE(‚Initiate CR',...) AS ‚Initiierung'
...
FROM cr-process



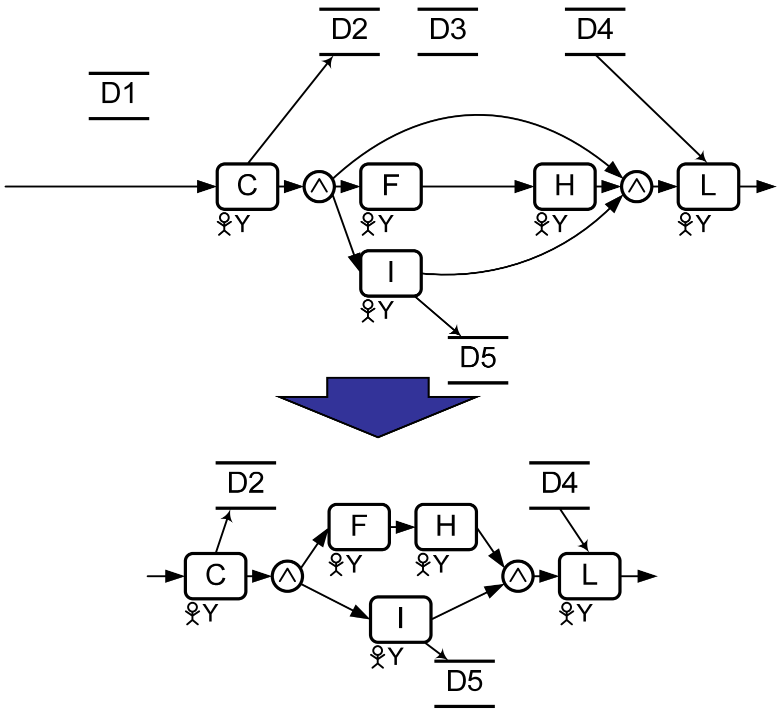


Some Requirements:

- Reduce complexity of (large) process models
- Aggregate or eliminate certain process information in a given application context
- Cover all process perspectives: behavior, data, …

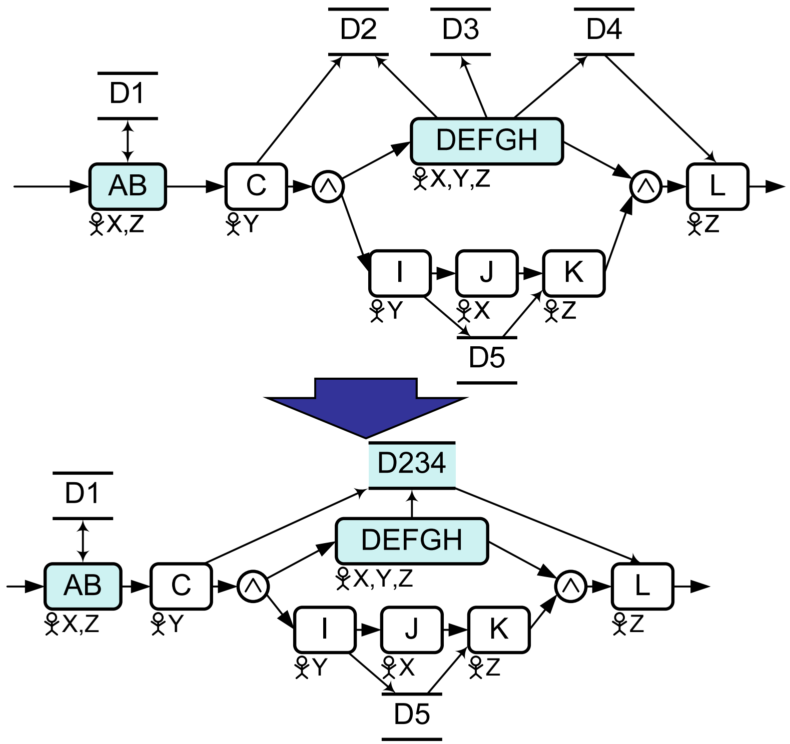
Proviado: Process Model Abstraction – Basic Operations (2)

Reduction



- Eliminate activities
- Simplify the resulting schema
- Remove adjacent satellite objects

Aggregation

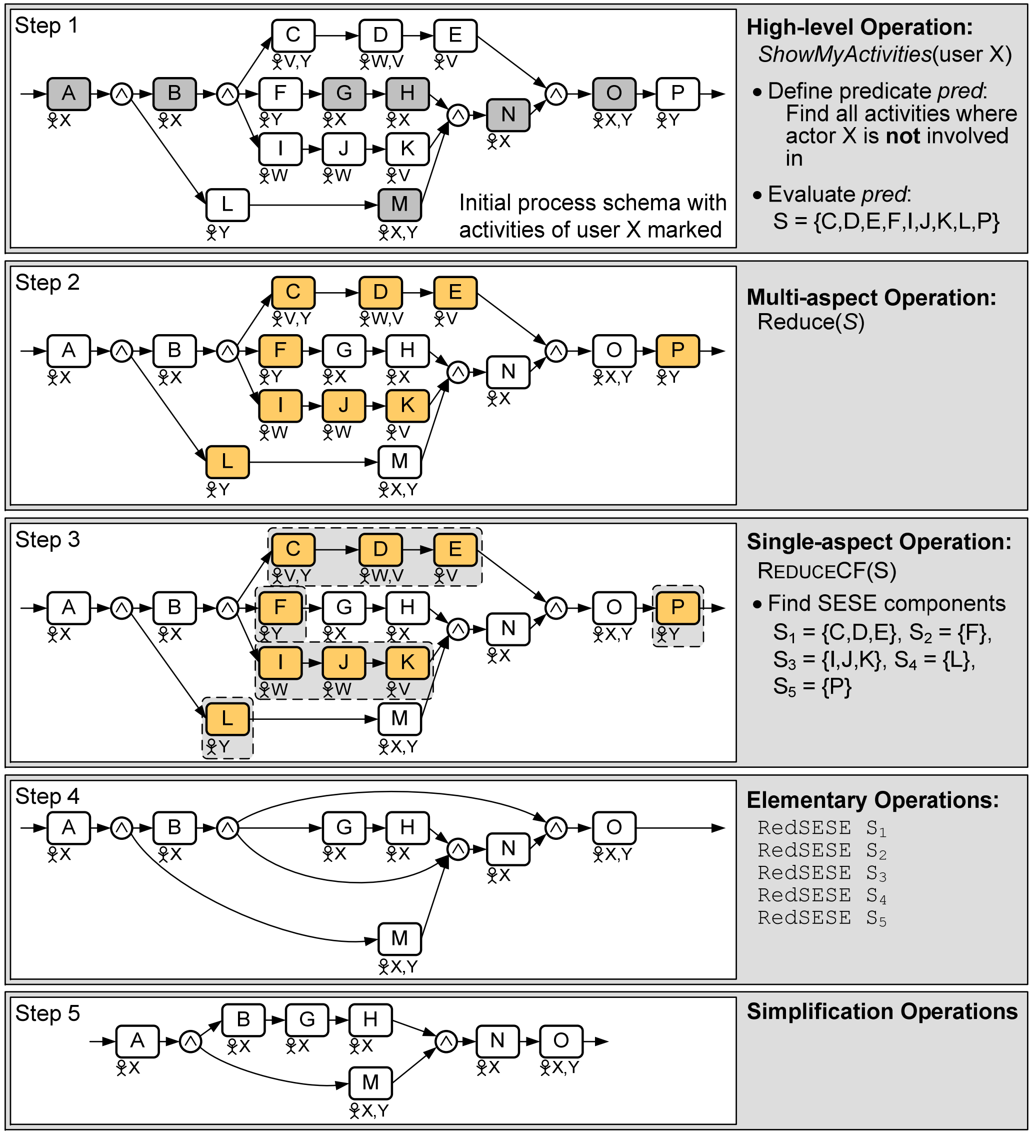


- Aggregate activities
- Aggregate adjacent objects if required

Proviado: Process Model Abstraction – High-Level Operations

Example:

ShowMyActivities

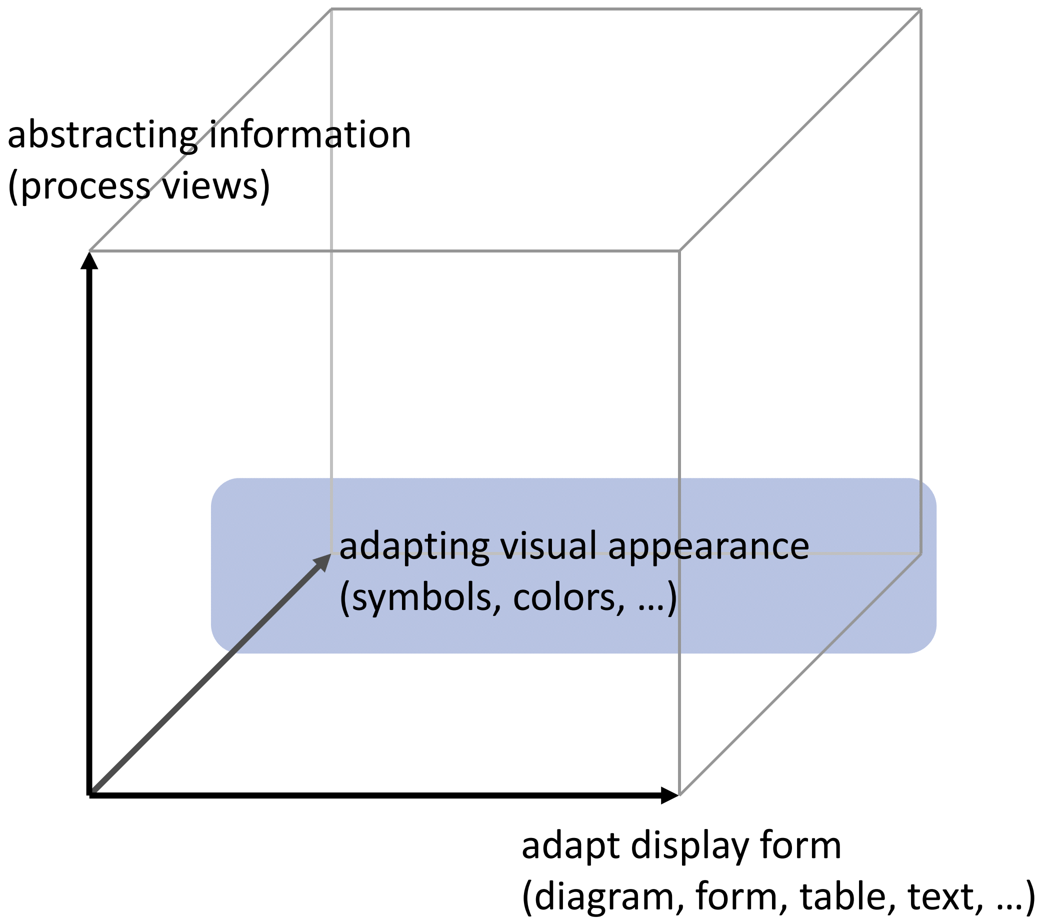


Proviado: Process Model Abstraction – Summary

Proviado …

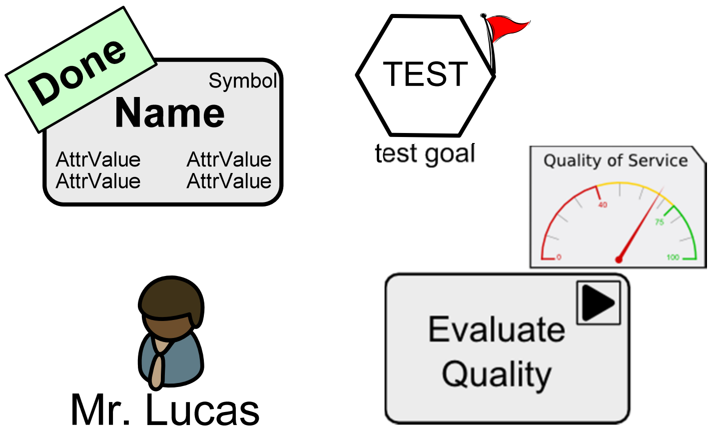
- offers a powerful mechanism for creating and visualizing process model abstractions (i.e., process views)
- enables a high degree of flexibility in respect to the artefacts created (based on parameterizable view-building operations)
- considers all process perspectives, e.g., control and data flow, process attributes, process logs
- has a well-defined formal foundation

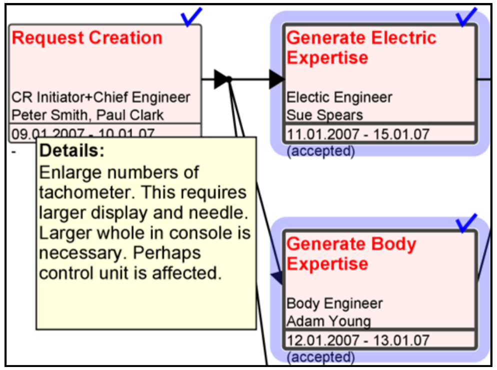
The Proviado Visualization Framework

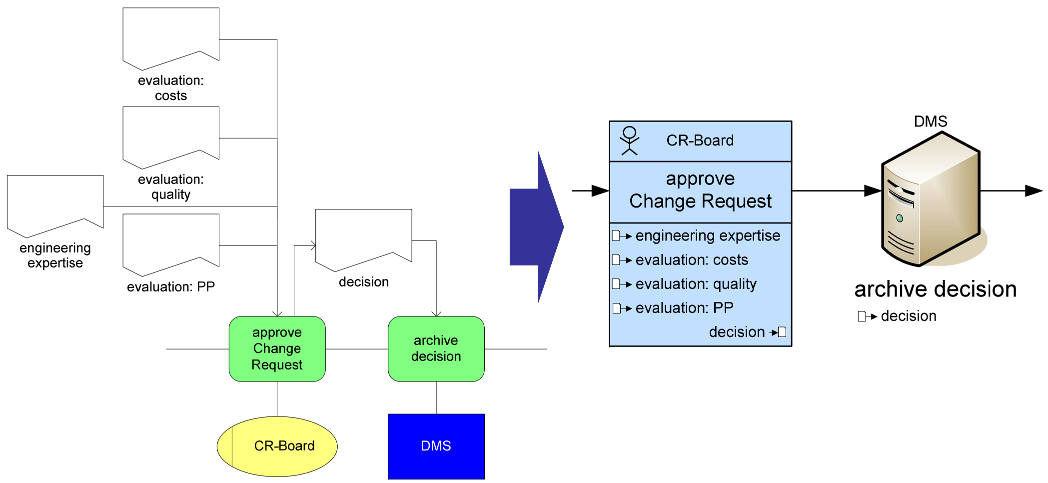


Proviado: Adjusting the Visual Appearance of Process Models

Visualization templates





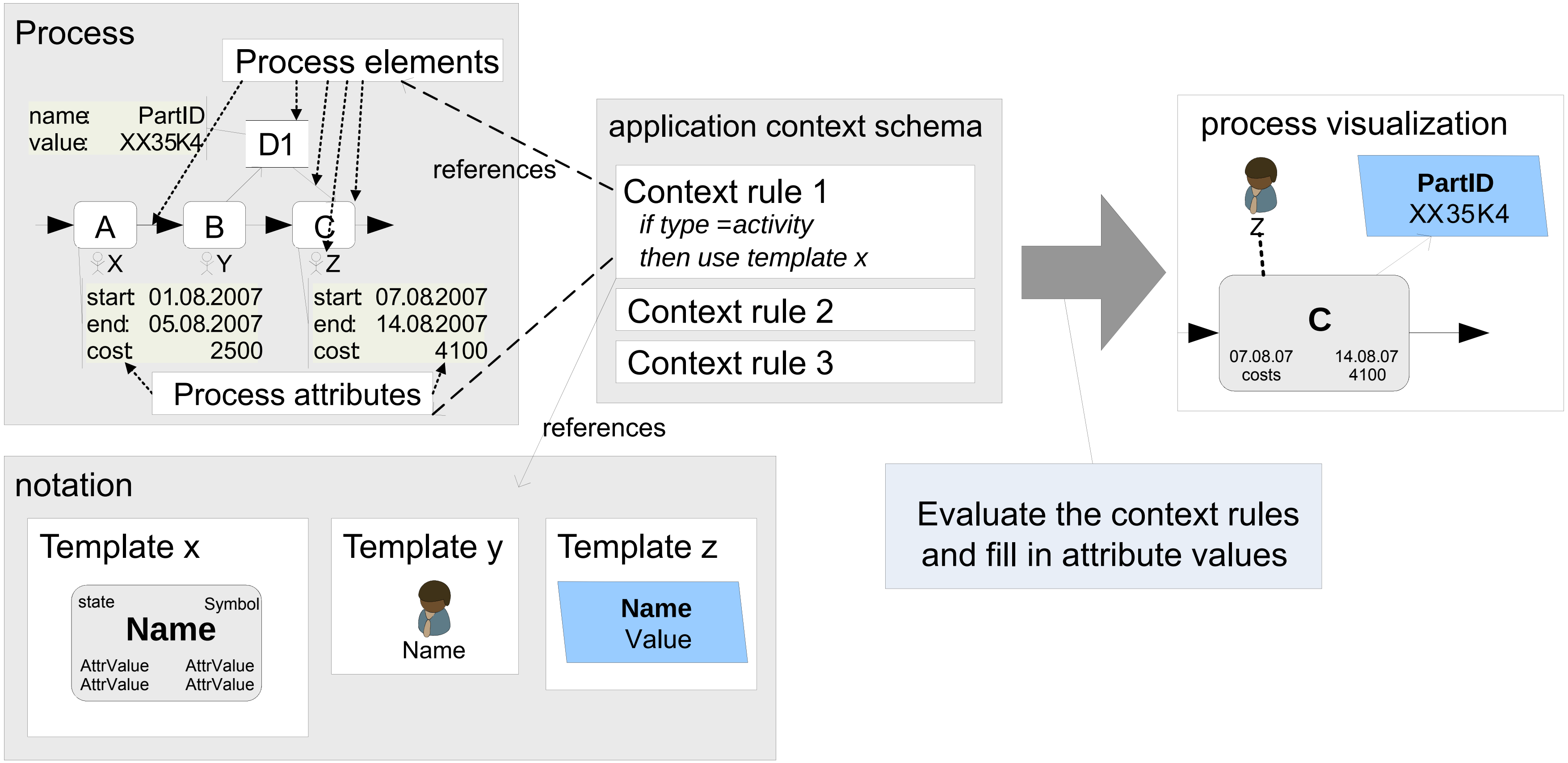


Visualization template defines

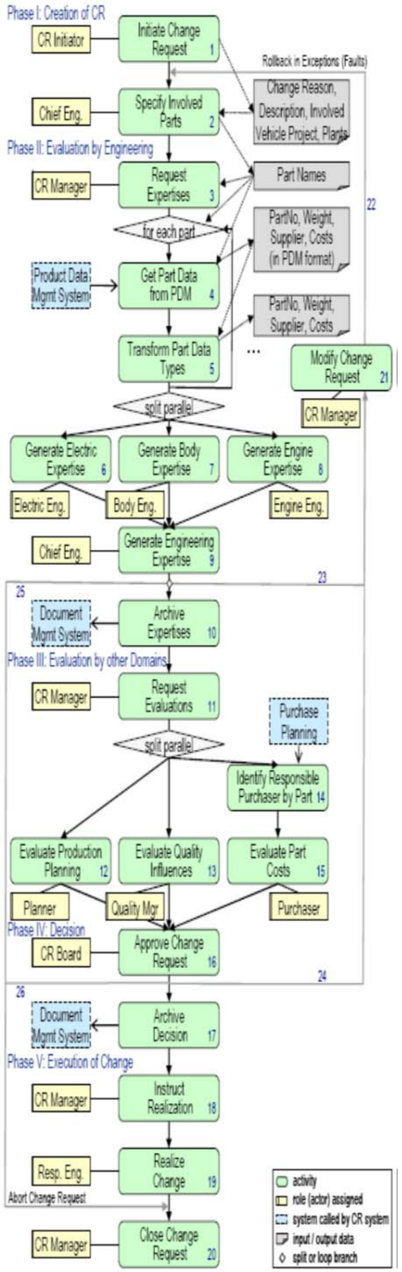
1. symbol to be used
2. data to be displayed
3. application context

Proviado: Adjusting the Visual Appearance of Process Models

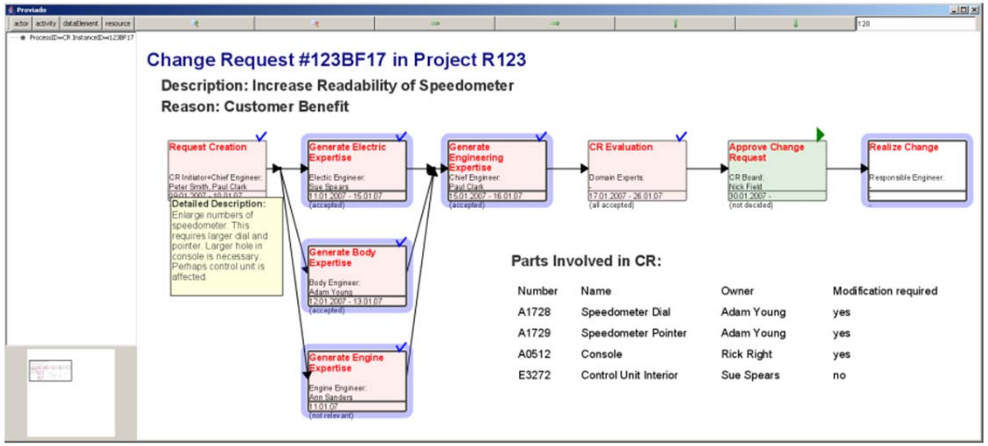
Creating a process visualization



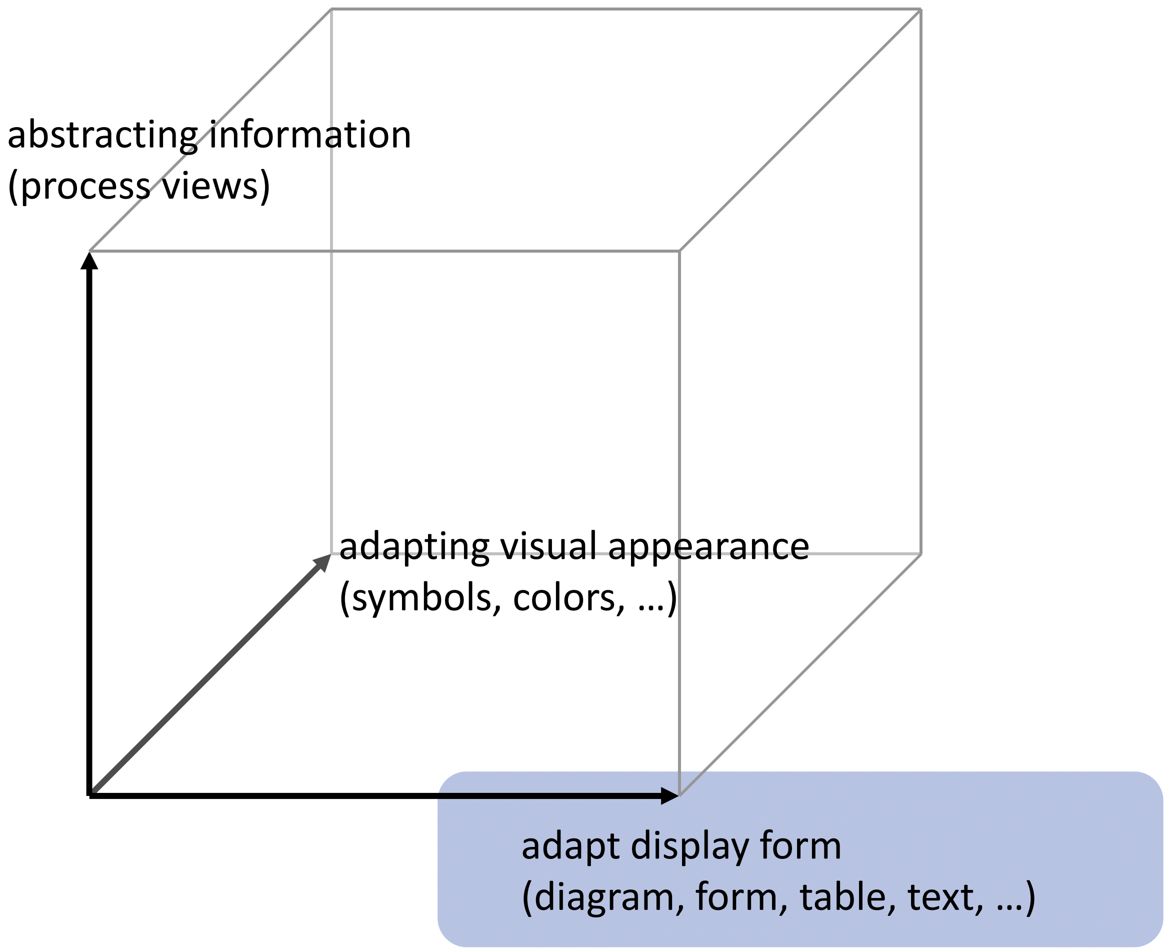
Proviado: Abstraction + Visual Configuration



Personalized Visualization

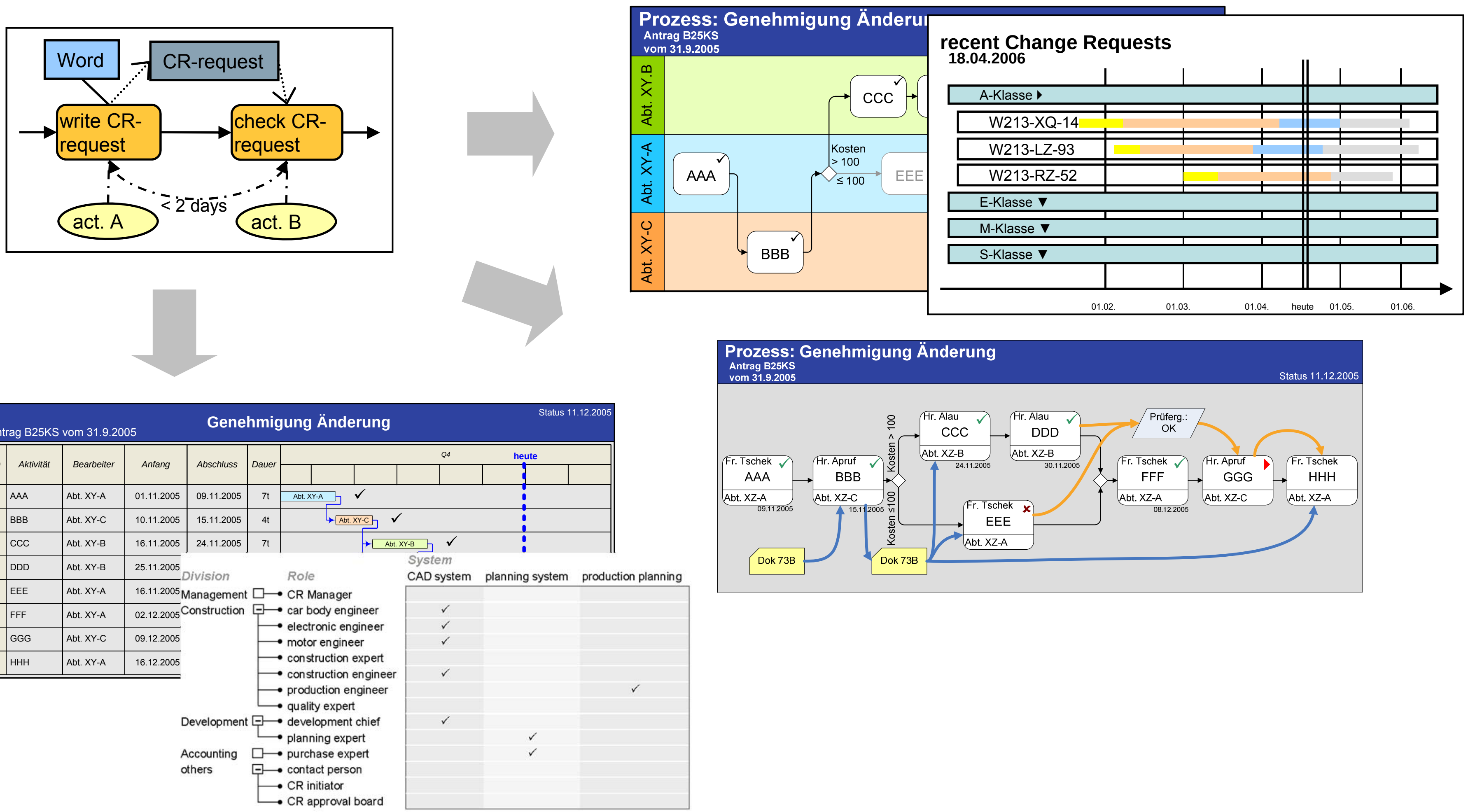


The Proviado Visualization Framework

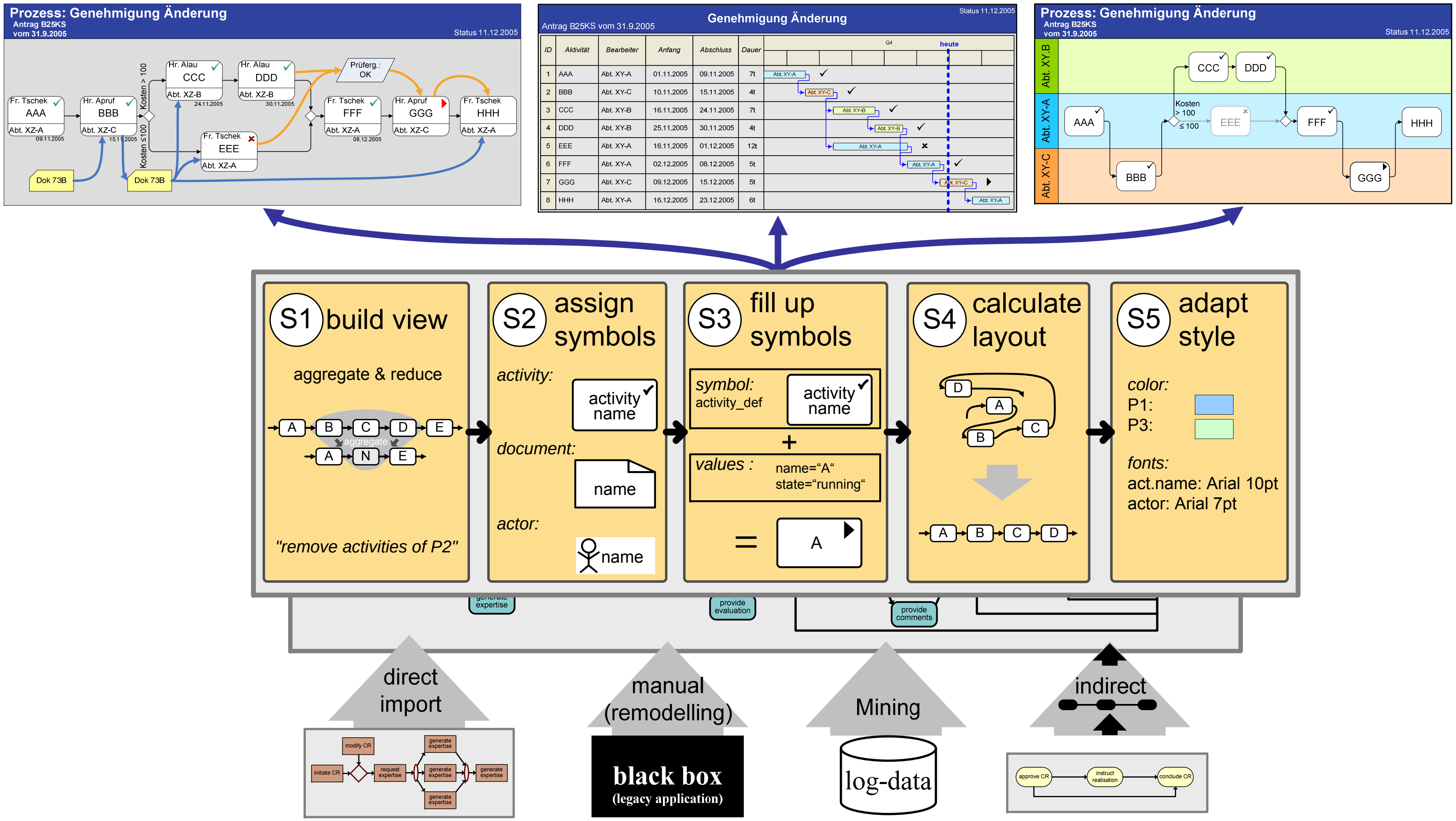


Proviado: Supporting Different Display Forms for Process Models

Proviado: Supporting Different Display Forms for Process Models



The Proviado Visualization Framework: Achievements



Introduction

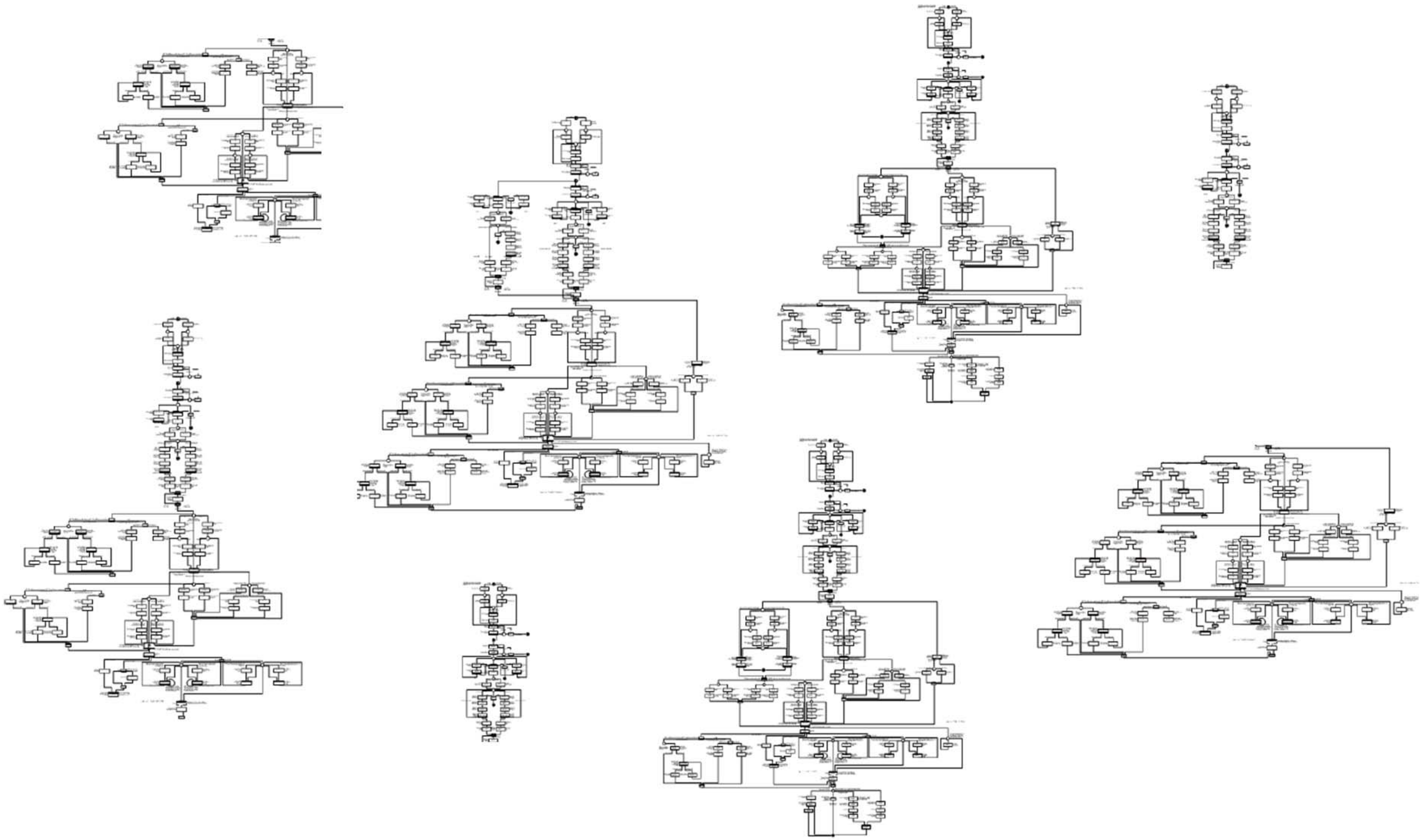
ADEPT – A Decade of Research on Flexible Process-Aware Information Systems and its Transfer to Practice

Large Processes in the Automotive Industry and Technologies to Support them

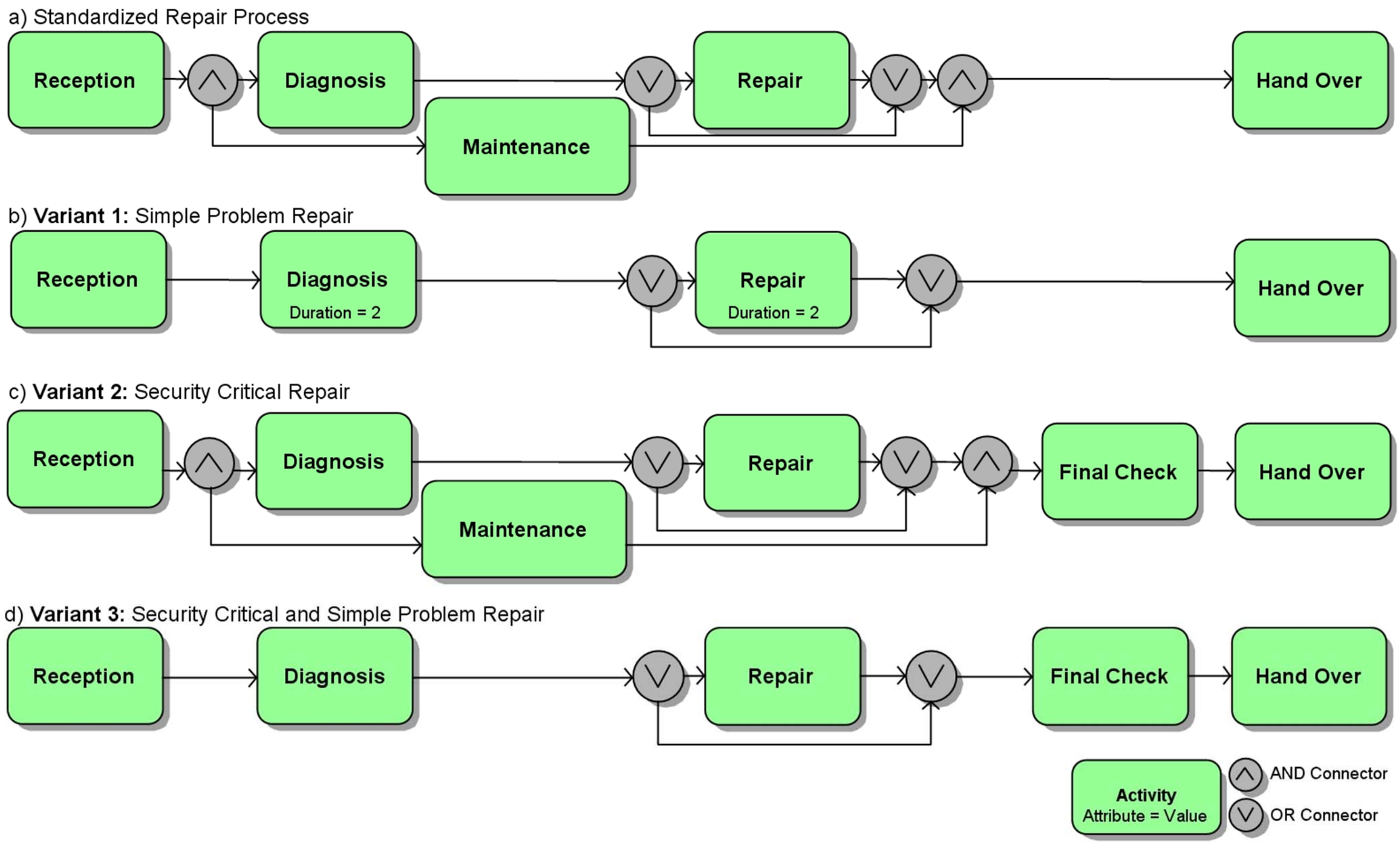
- Large Process Models
- Large Process Collections
- Large Process Structures

Conclusions

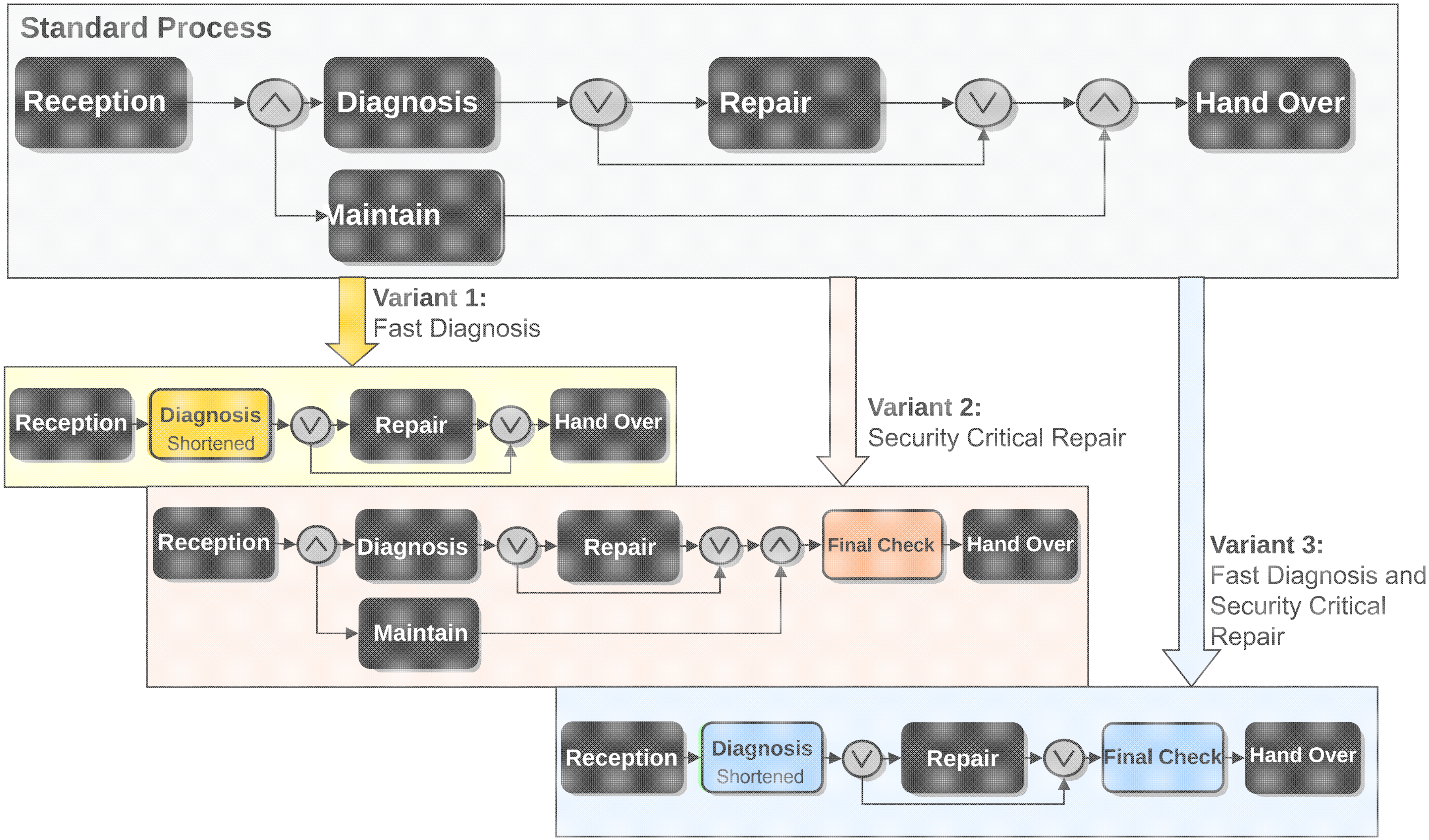
The Challenge: Dealing with Large Process Model Collections



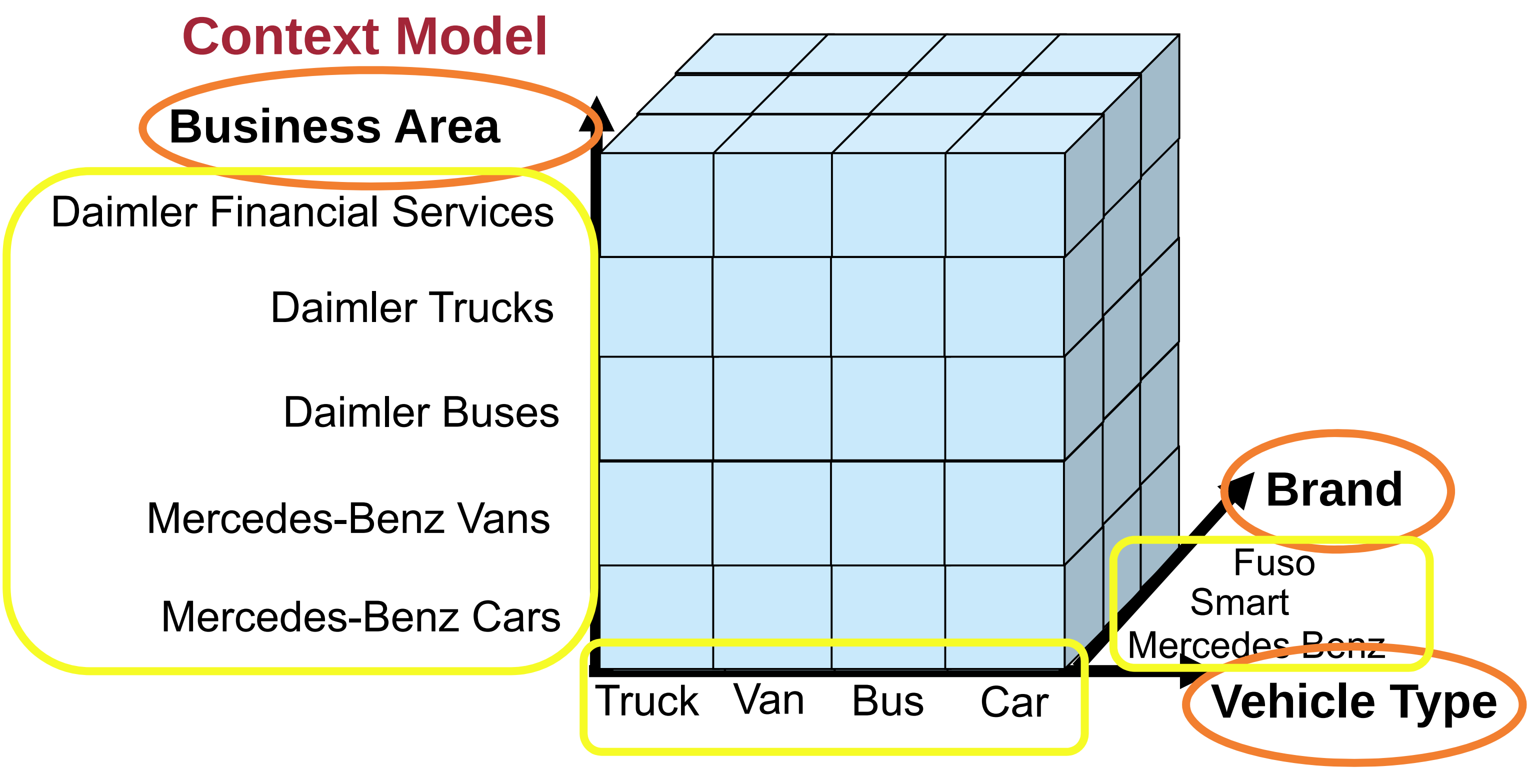
… and a Particular Challenge: Managing Process Variants



… and a Particular Challenge: Managing Process Variants

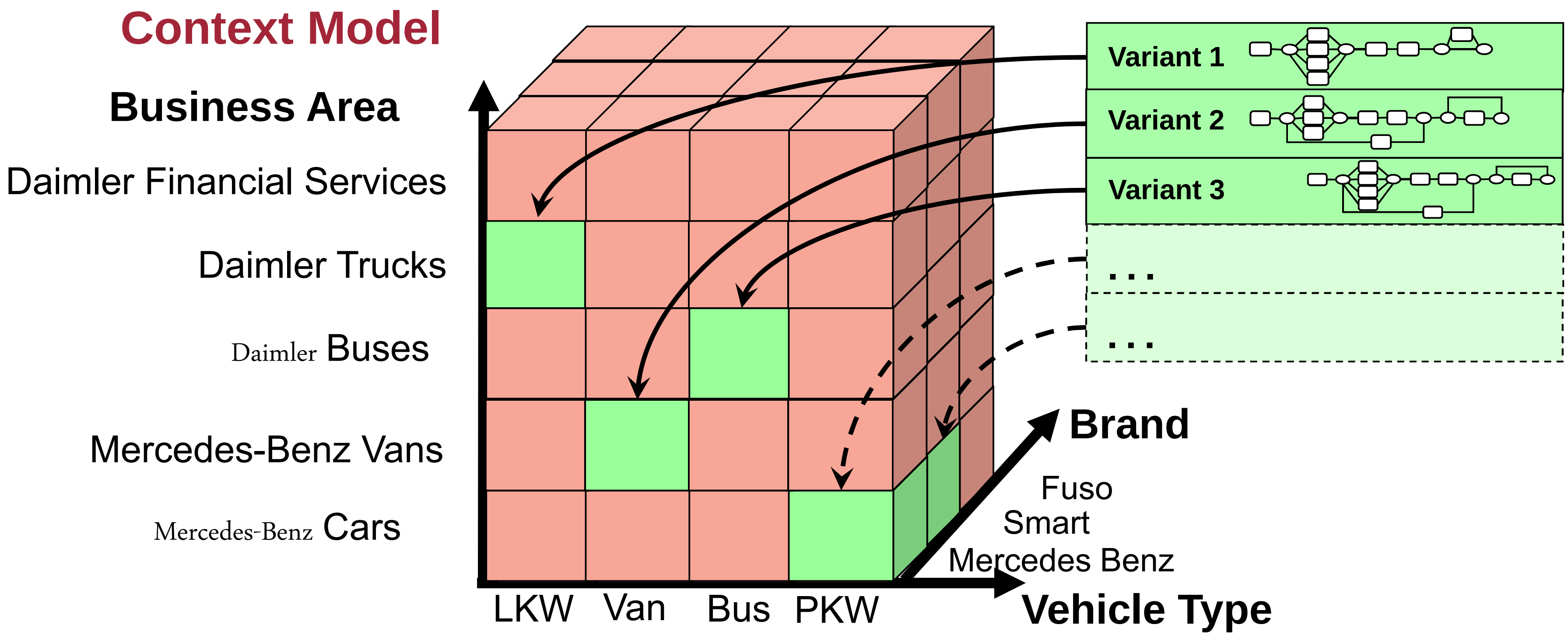


… and a Particular Challenge: Managing Process Variants



Problem: Not all value combinations make sense!

… and a Particular Challenge: Managing Process Variants

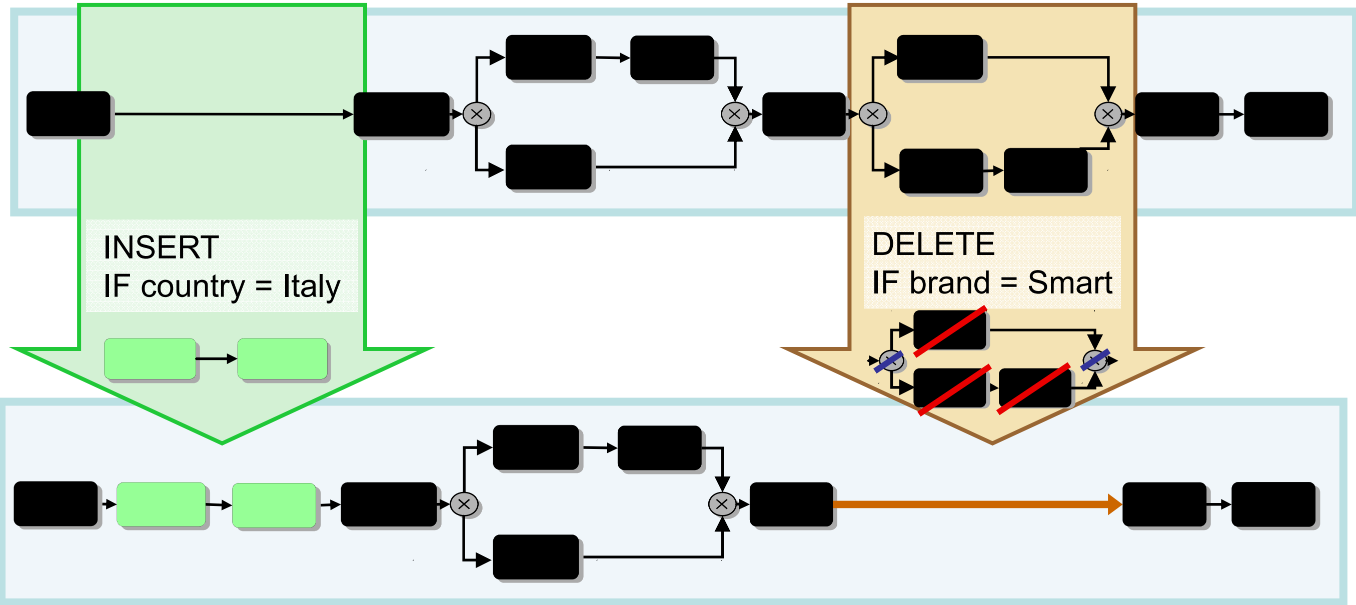


The Provop Approach for Managing Process Variants

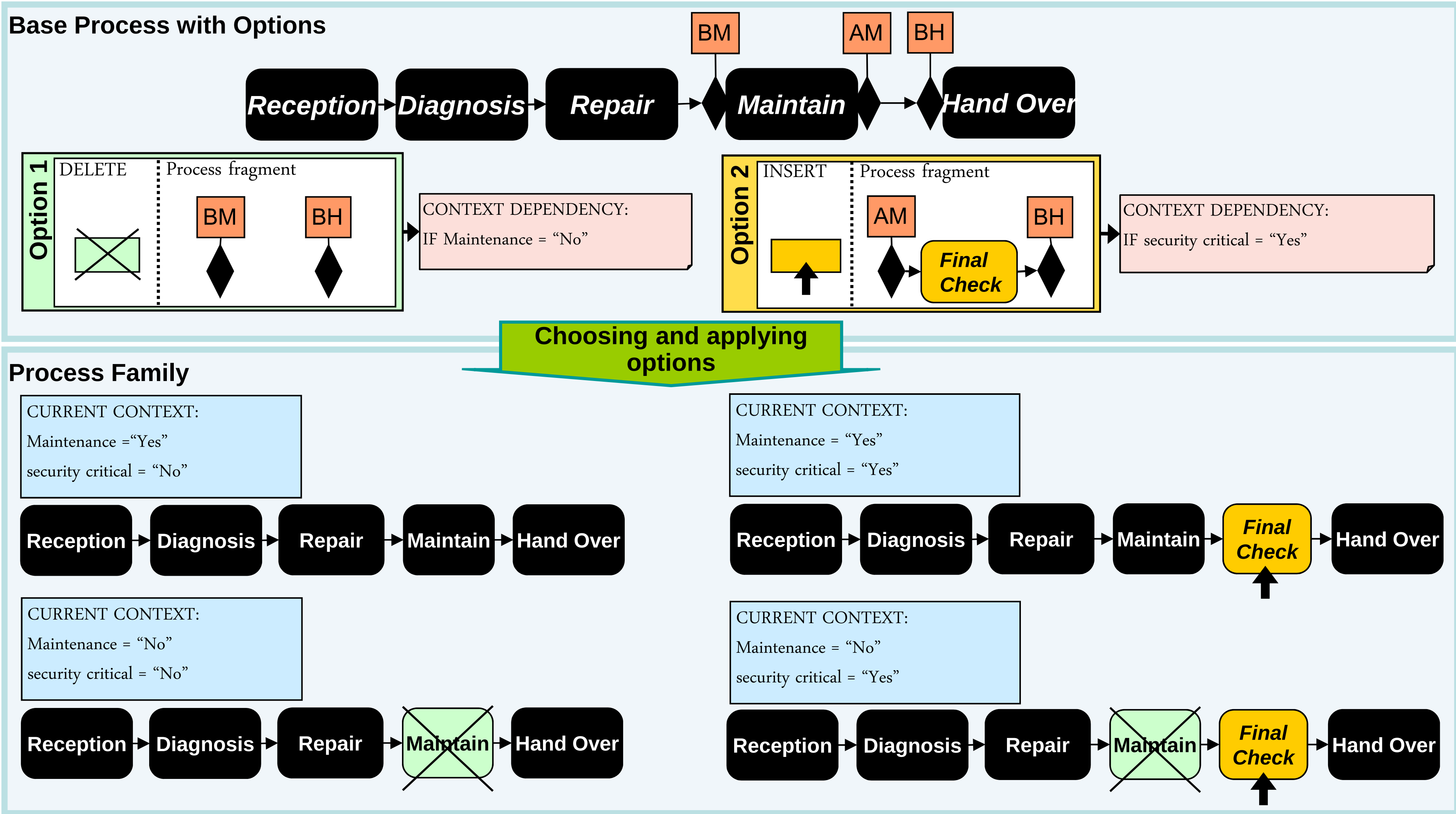
Base process

Variant specific adjustments

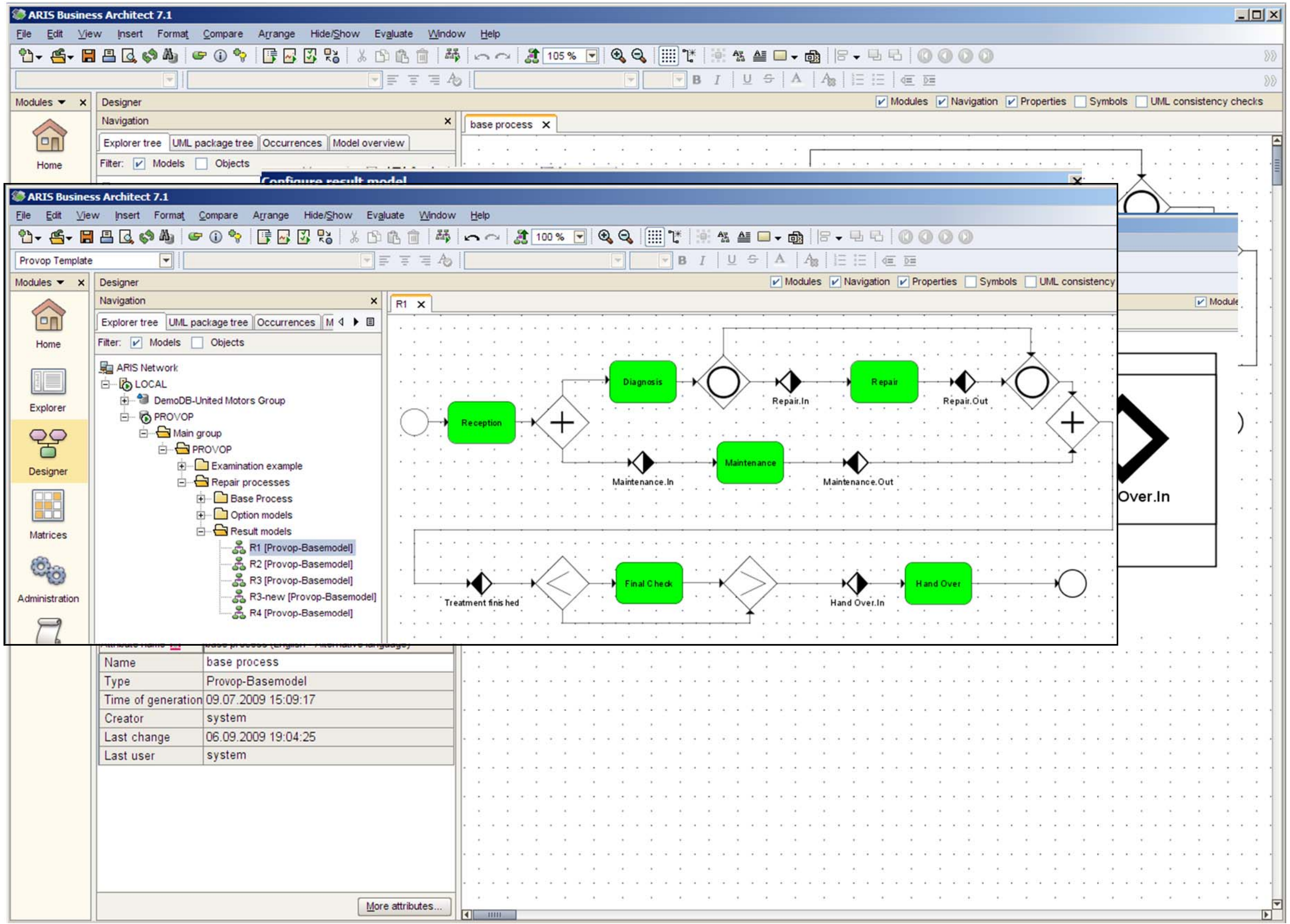
Configured process variant



The Provop Approach for Managing Process Variants



The Provop Approach for Managing Process Variants



Introduction

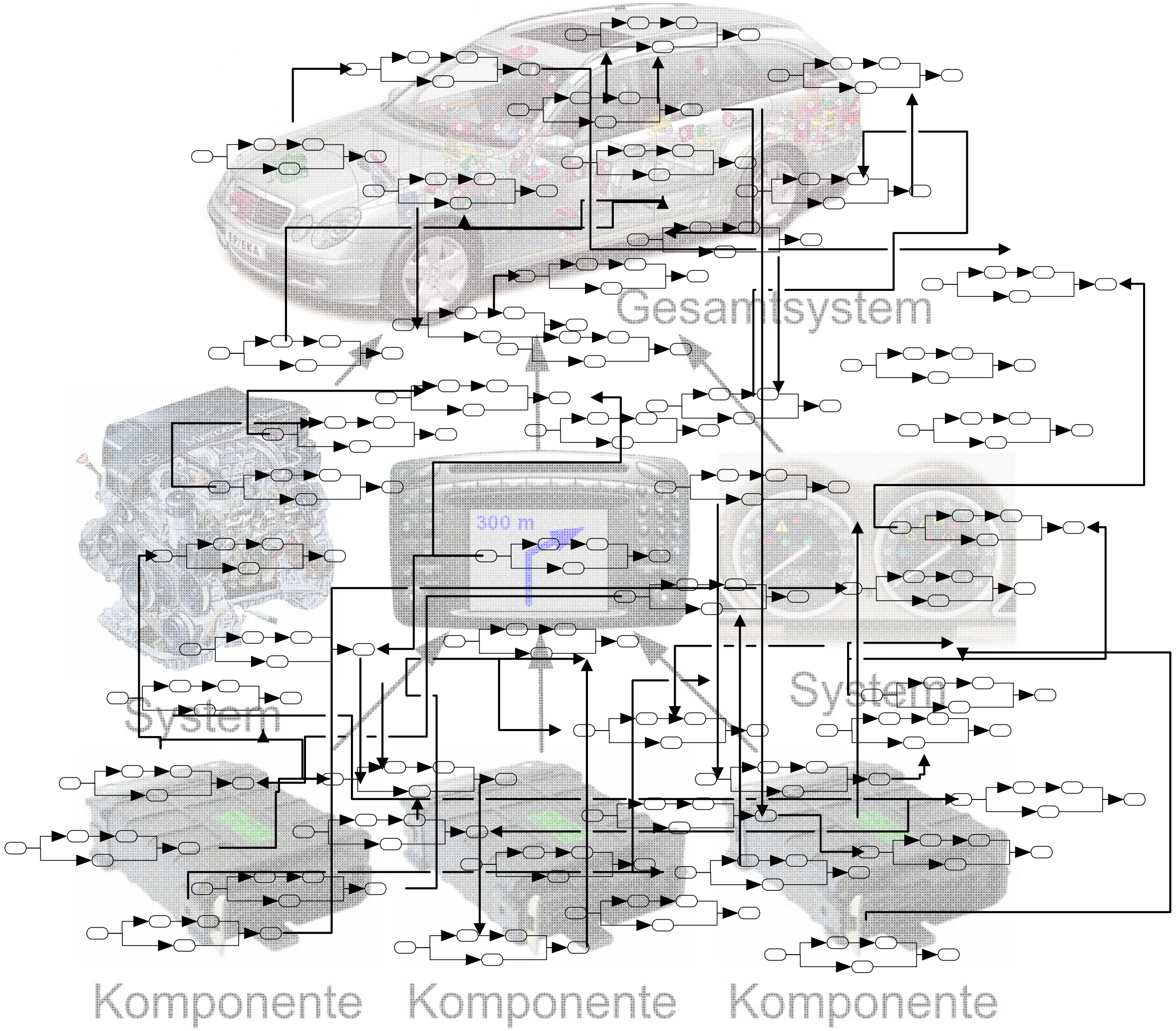
ADEPT – A Decade of Research on Flexible Process-Aware Information Systems and its Transfer to Practice

Large Processes in the Automotive Industry and Technologies to Support them

- Large Process Models
- Large Process Collections
- Large Process Structures

Conclusions

The Challenge: Dealing with Large and Complex Process Structures



The Challenge: Dealing with Large and Complex Process Structures

Automotive Engineering:

- ❑ Electrical control units (ECUs) become more and more important:
  - provide many safety-critical functions
  - fast implementation of changes: adjustments and bug fixes by flashing new software onto the ECU
- ❑ Modern cars comprise up to 70 ECUs; >10.000.000 LoC
- ❑ ECUs interconnected by up to 10 buses with 2 kilometers of wires
- ❑ 90% of car innovations enabled by E/E systems

Example:
Electronics in side door

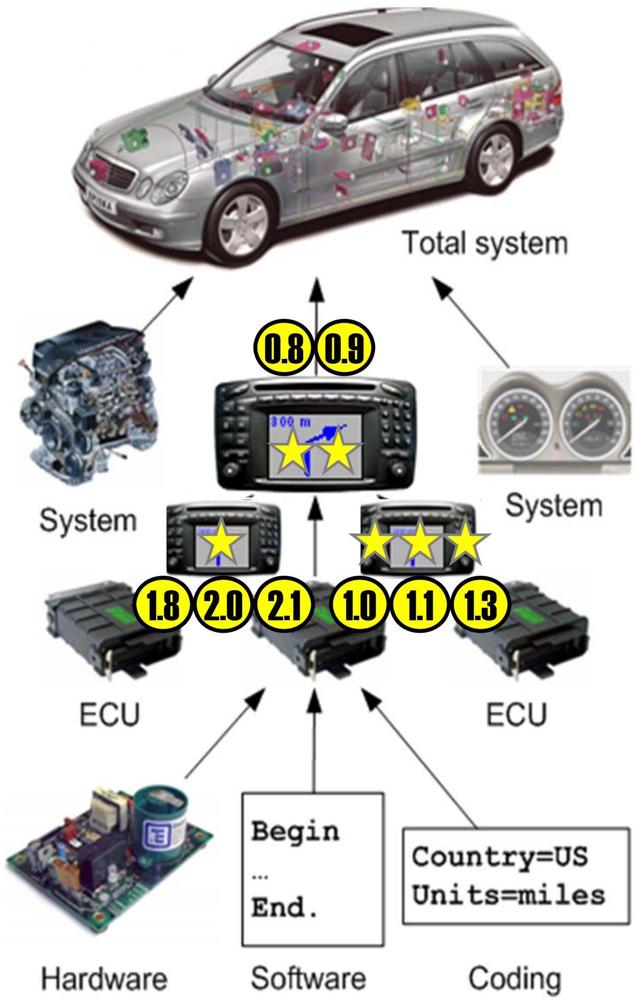
- **Power window**
  - Safety stop
  - Close with central locking system
  - Safety functions (Presafe)
  - Communication with air condition
- **Electrical side mirrors**
  - Electrical adjustment
  - Electrical heating
  - Memory function
  - Retractable side mirror
  - Automatic fading out
  - Ambient illumination
  - Turn indicator
- **Door lock**
  - Open / Close with central locking system
  - Sensors for alarm system
  - Power closing
- **Sidebags**
  - Side impact sensors
- **Active surround speakers**
- **Control unit for**
  - Power windows
  - Mirror adjustment
  - Seat adjustment
  - Memory function
  - Child safety lock
  - Central locking system

The Challenge: Dealing with Large and Complex Process Structures

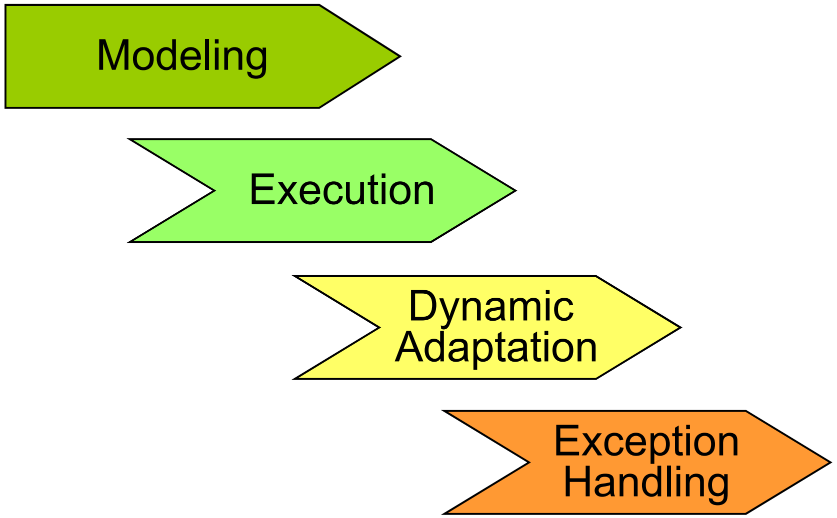
Current Problems in Automotive Engineering

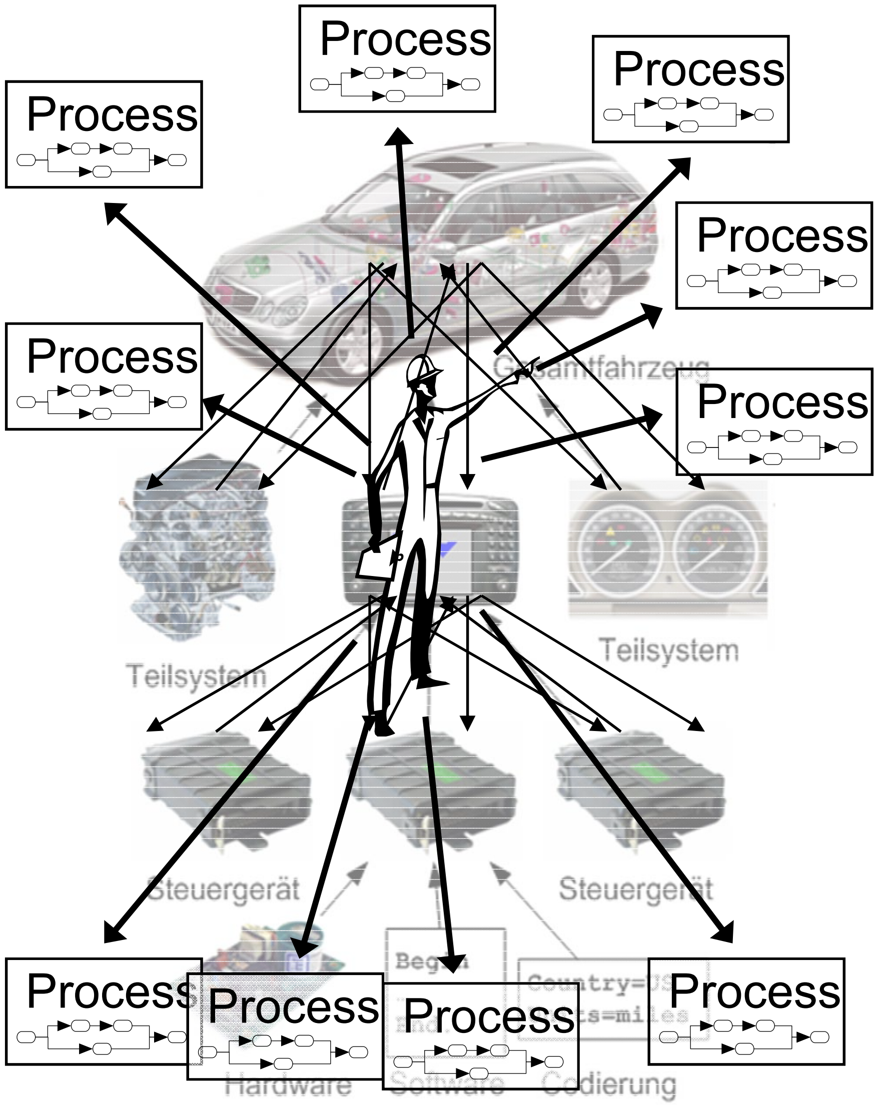
- ❑ Up to 50% of all car breakdowns due to electrical / electronic problems
- ❑ Some facts
  - Many non-obvious dependencies between ECUs
  - Different life and development cycles of mechanics, hardware and software
  - Numerous ECU variants and versions

☞ **Systematic verification and release management required**

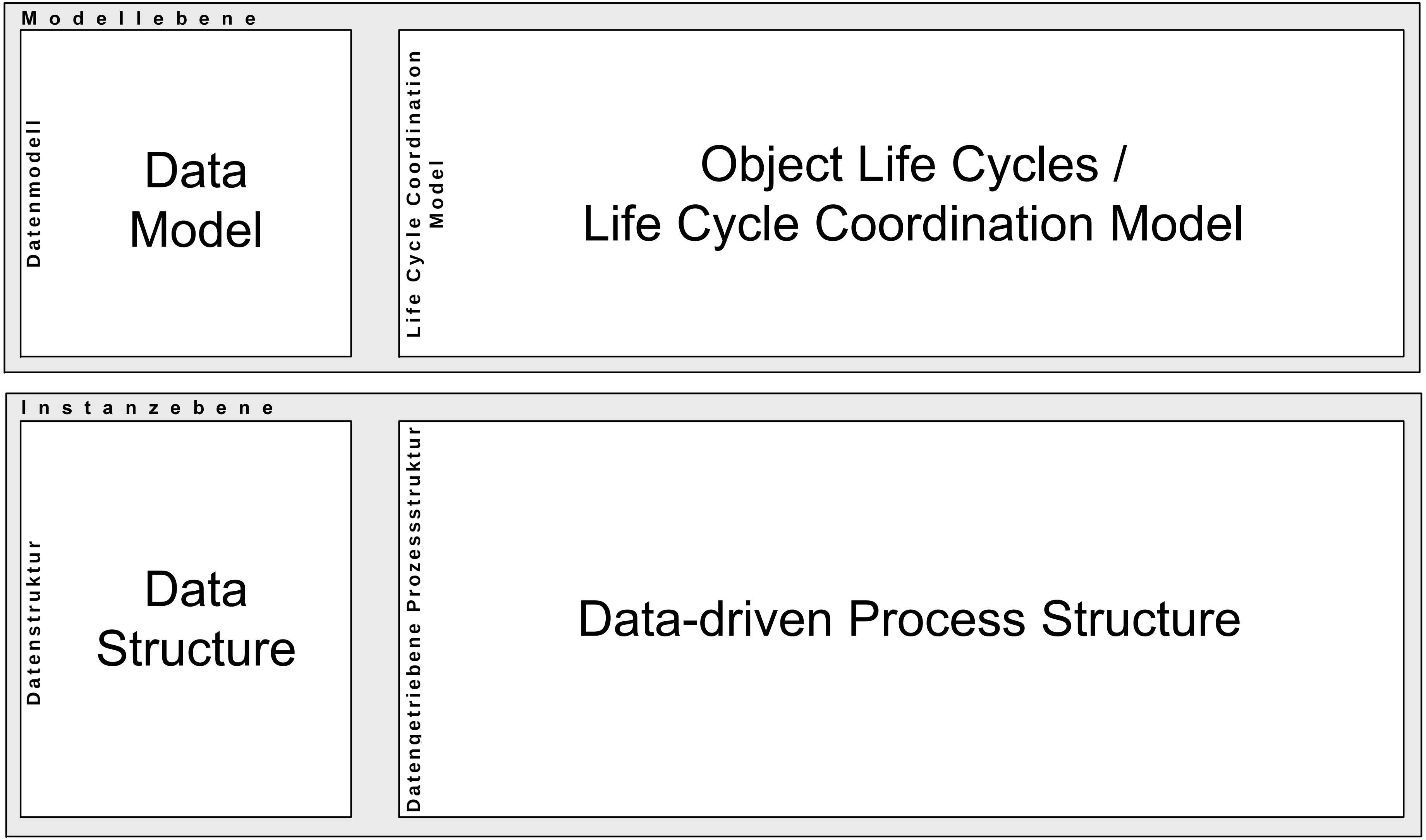


The Challenge: Dealing with Large and Complex Process Structures

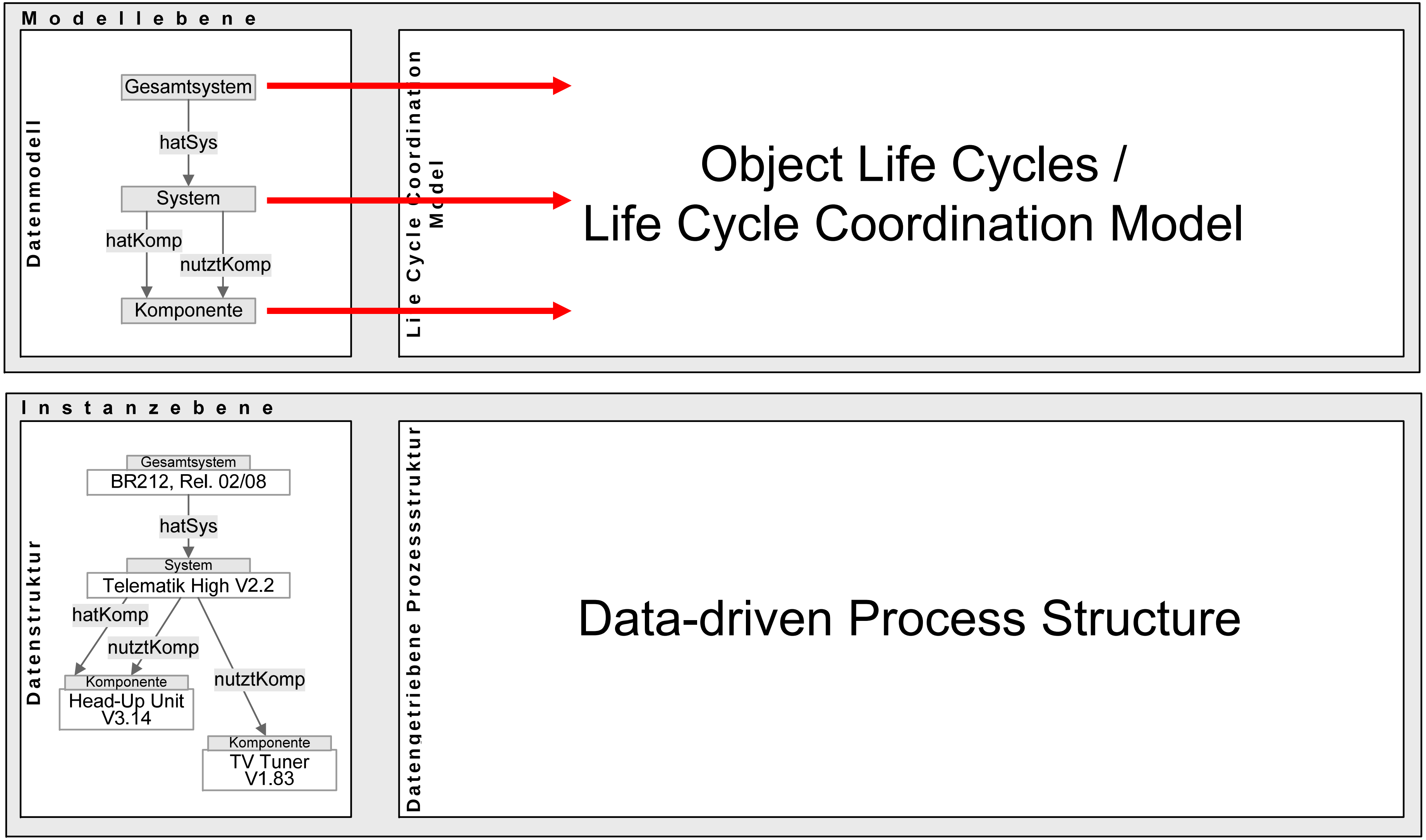




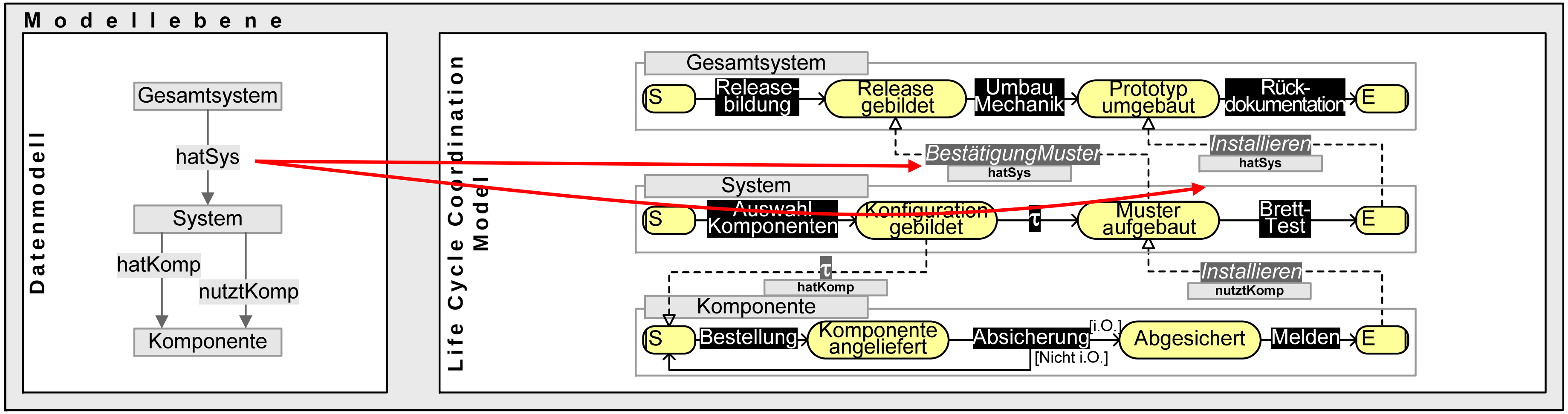
The Corepro Project – Basic Approach

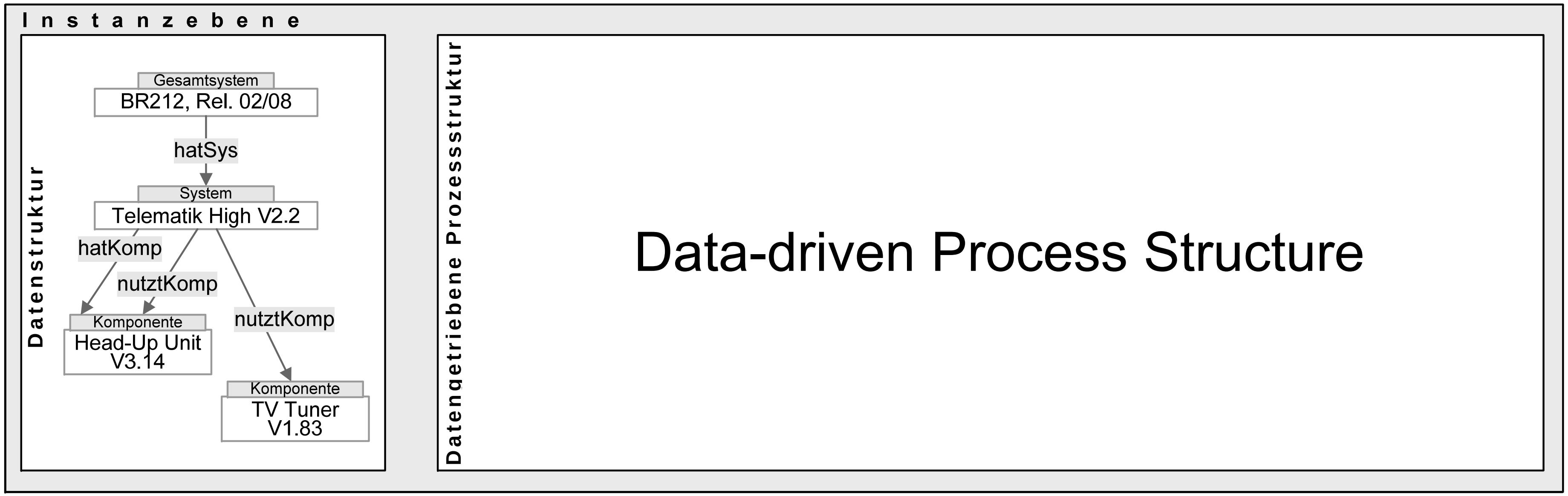


The Corepro Project – Basic Approach

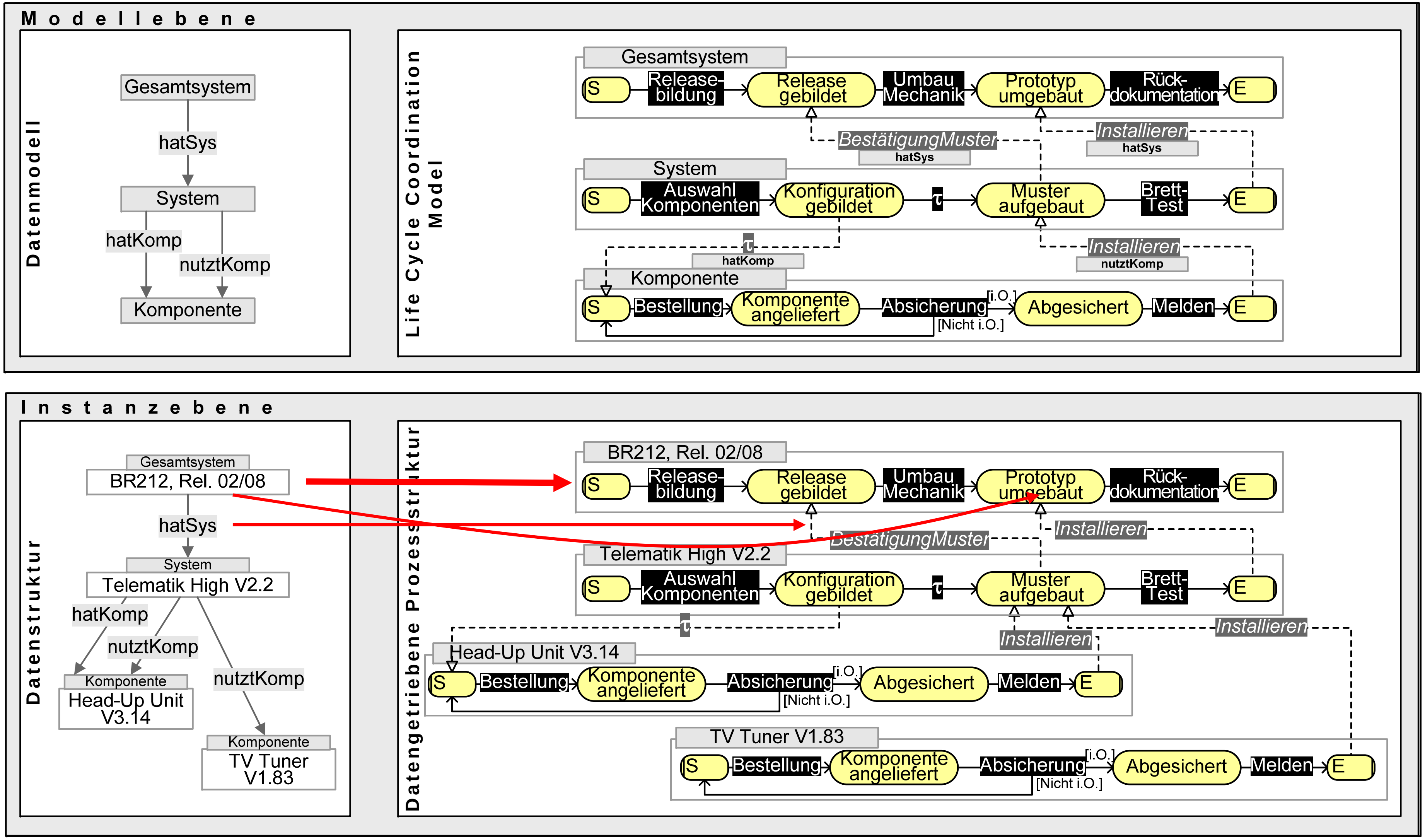


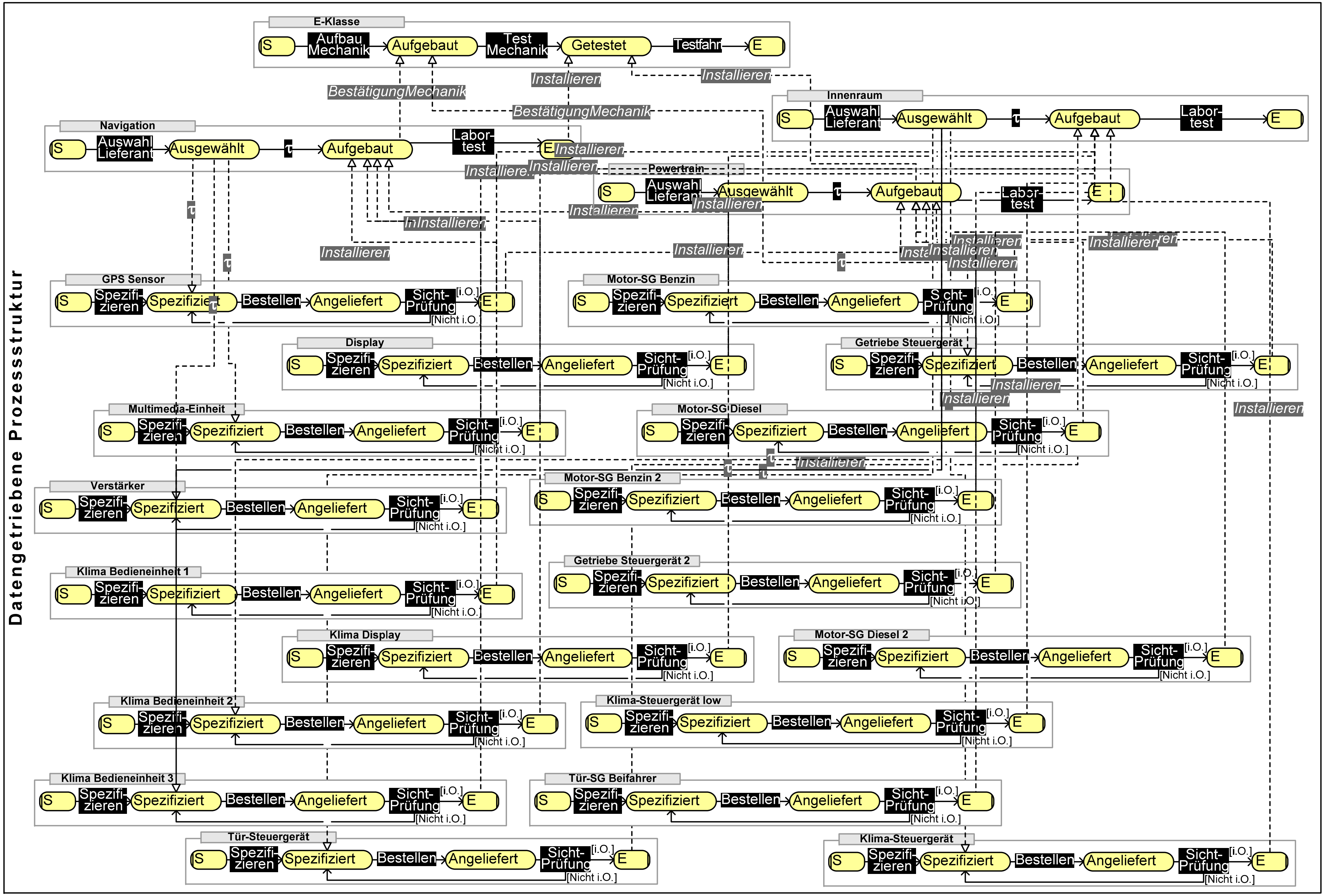
The Corepro Project – Basic Approach

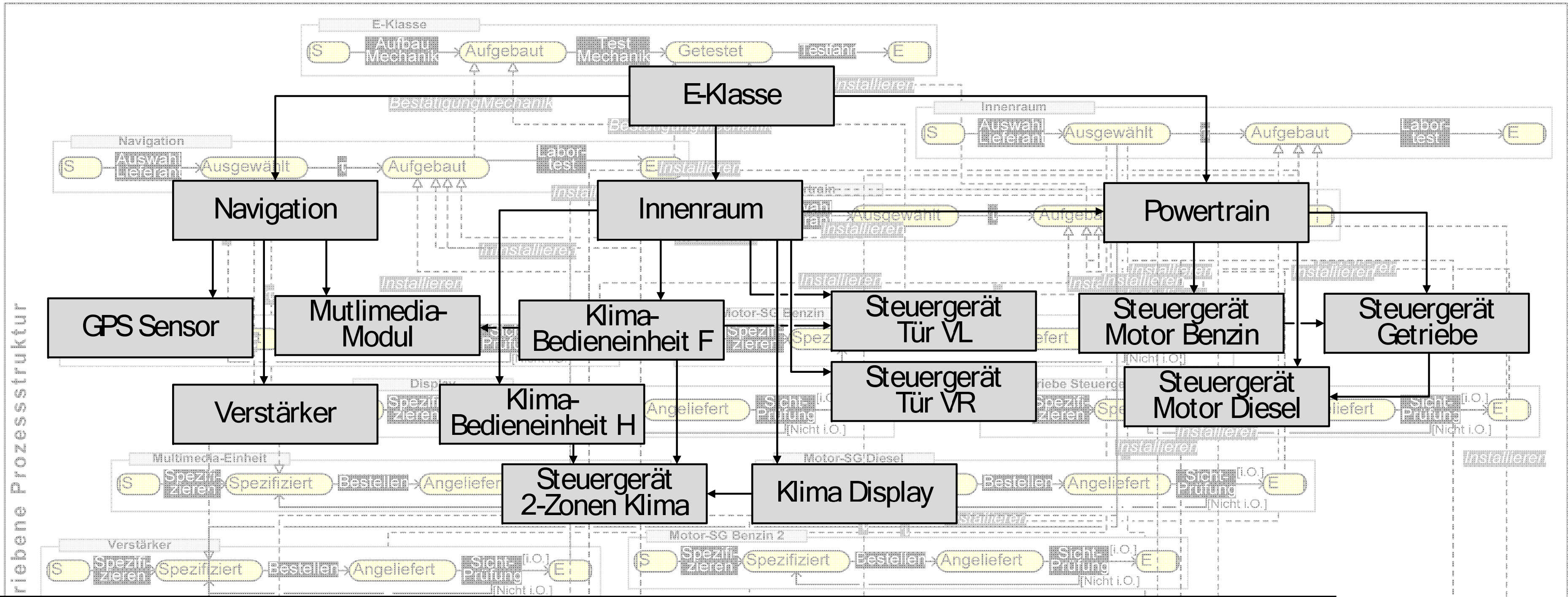




The Corepro Project – Basic Approach

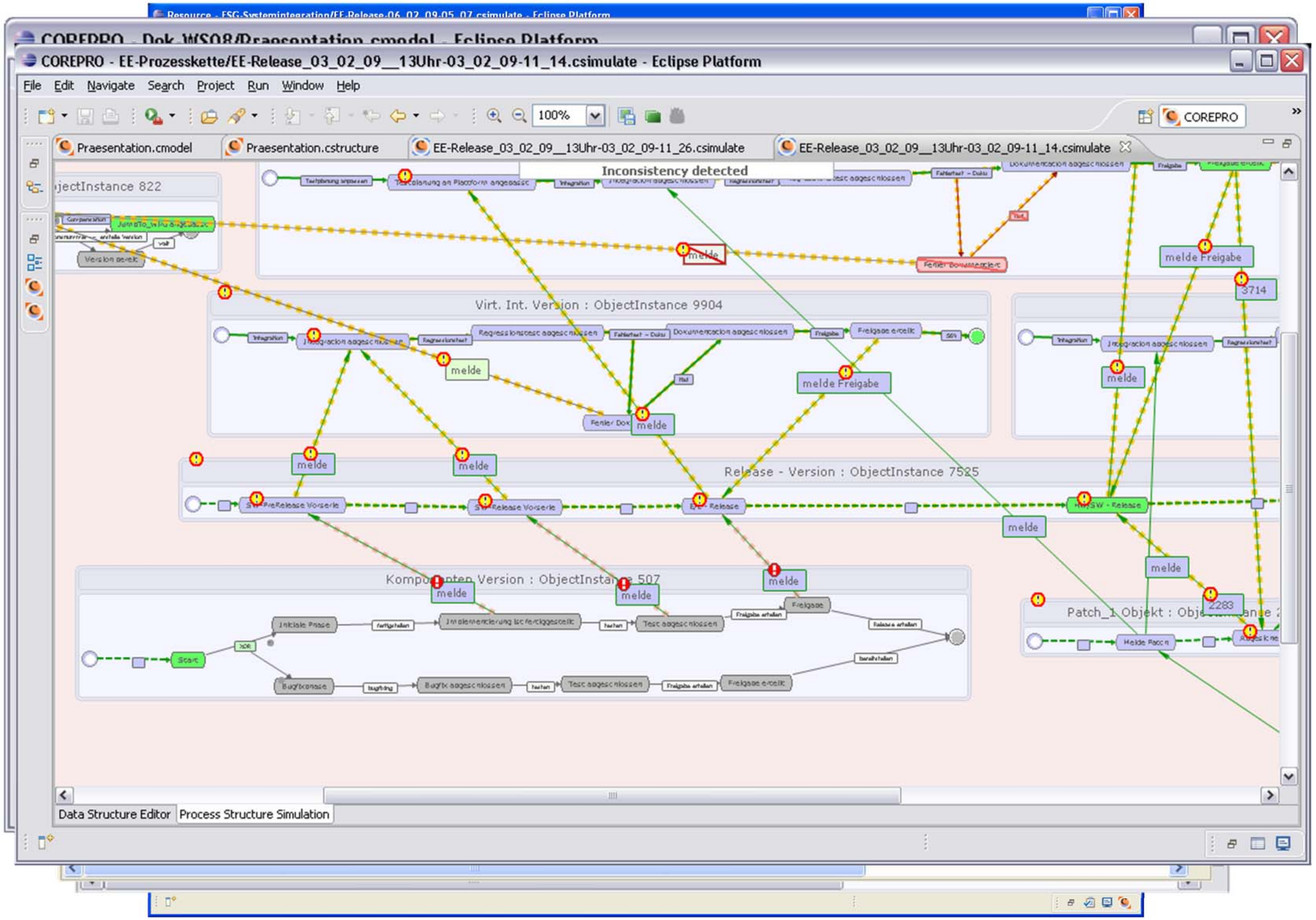



Datengetriebene Prozessstruktur
E-Klasse
Aufbau Mechanik
Aufgebaut
Test Mechanik
Getestet
Testfahrt
BestätigungMechanik
Installieren
Innenraum
Auswahl Lieferant
Ausgewählt
Labor-test
Navigation
Powertrain
GPS Sensor
Spezifi-zieren
Spezifiziert
Bestellen
Angeliefert
Sicht-Prüfung
[i.O.]
[Nicht i.O.]
Motor-SG Benzin
Display
Getriebe Steuergerät
Multimedia-Einheit
Motor-SG Diesel
Verstärker
Motor-SG Benzin 2
Klima Bedieneinheit 1
Getriebe Steuergerät 2
Klima Display
Motor-SG Diesel 2
Klima Bedieneinheit 2
Klima-Steuergerät low
Klima Bedieneinheit 3
Tür-SG Beifahrer
Tür-Steuergerät
Klima-Steuergerät



- Significant reduction of modeling efforts for process engineers
- Formal operational semantics allows for correct executability
- Soundness can be guaranteed on an abstracted level

The Corepro Project – Exception Handling



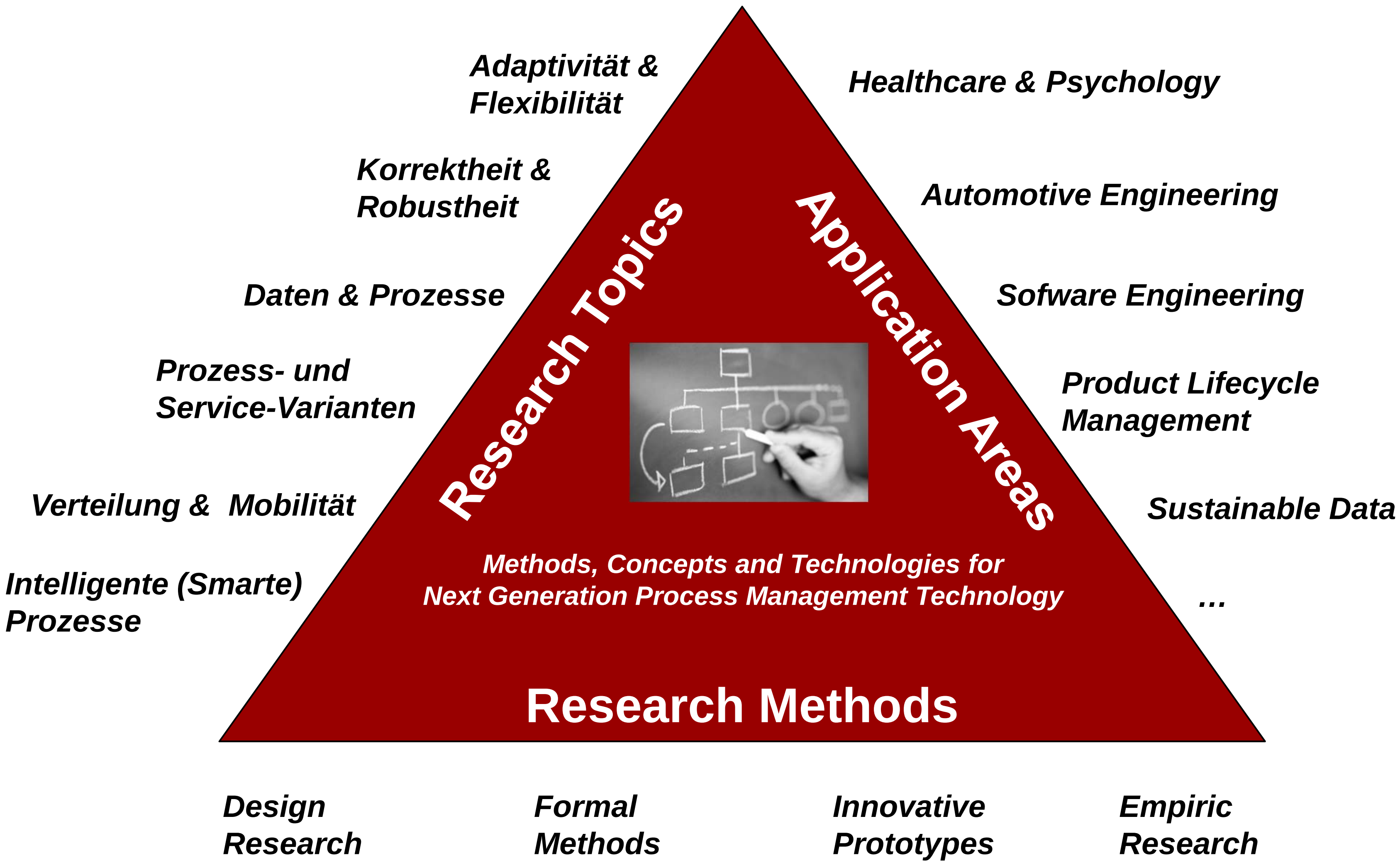
Introduction

ADEPT – A Decade of Research on Flexible Process-Aware Information Systems and its Transfer to Practice

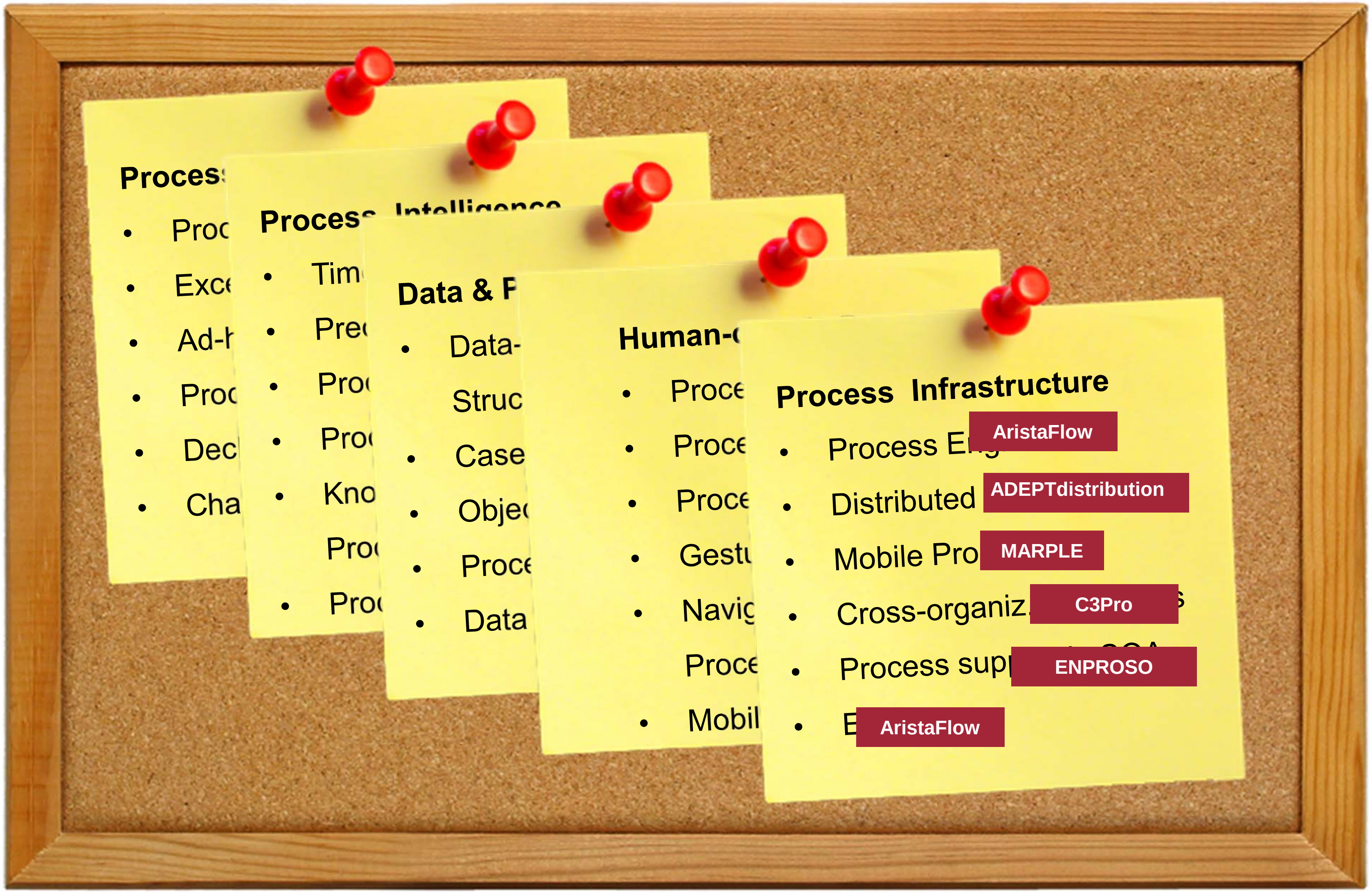
Large Processes in the Automotive Industry and Technologies to Support them

Conclusions

Research of my Team



Research Projects



www.process-flexibility.com