A Decade of Research on a Next Generation Process Management Technology

Challenges, Projects, Achievements

Manfred Reichert
Process-Aware Information Systems

A Decade of Research on Flexible Process-Aware Information Systems and its Achievements

A Decade of Research on Large Processes in the Automotive Industry and Enabling Technologies

Other Running Projects
Process-Aware Information Systems

- 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
Process-Aware Information Systems

- Registration
- Patient information
- Orders Placed
- Examination orders
- Orders Filled
- RIS
- HIS
- Acquisitions in-progress/Acquisitions completed
- Acquisition Modality
- PACS
- Acquisition Modality stored
- Acquisition completed
- Image Manager & Archive
- Images retrieved
- Image Lightbox
- Film Lightbox
- Film Folder
- Film
- Report Repository
- Report
- Prior studies
- Prefetch any relevant prior studies
- Worklist
Process-Aware Information Systems: Buildtime

Process Schema S

Process Instance I1

Execution Trace:
\( \sigma_1 = \langle \text{Patient Admission}, \text{Anamnesis & Clinical Examination}, \text{X-ray} \rangle \)

Activity States: ▲ Activated ✓ Completed ✗ Skipped

Process Instance I2

Execution Trace:
\( \sigma_2 = \langle \text{Patient Admission}, \text{Anamnesis & Clinical Examination}, \text{Non Operative Therapy} \rangle \)
Process-Aware Information Systems: Buildtime

Process Compliance
Process-Aware Information Systems: Buildtime

The activities CT and Inform patient are on different branches of an XOR-Block.

Process Compliance

The activities CT and Inform patient are on different branches of an XOR-Block.
Process model to be checked

Generated counterexample:
Execution path and corresponding process context violating the constraint
Process-Aware Information Systems: Runtime

### Process-aware Information System (PAIS)

- Late Modeling
- Admin. API
- Validation
- Authorization
- Dynamic Change API
- Time Mgmt
- Web Client API
- Exceptions
- Audit Trail
- Modeler API
- Administration API
- Exceptions Audit Trail

### Process Execution Engine

- Instance 1
- Instance 2
- Instance 3
- Instance 4
- Instance 5
- Instance 6
- Instance 7
- Instance 8
- Instance 9
- Instance 10
- Instance 11
- Instance 12
- Instance 13
- Instance 14

### Process Composer

- Create Process Schema
- Modify Process Schema
- Check Process Schema

### Process Repository

- Process Schemas
- Application Components

### Users

- Process Engineer

- Process Composer

- Process Repository
Process-Aware Information Systems: Traditional Process Lifecycle

1. Create Process Schema
2. Create Instances
3. Process Execution
4. Execution Log
5. Evolve Process Schema

Process-Aware Information Systems

A Decade of Research on Flexible Process-Aware Information Systems and its Achievements

A Decade of Research on Large Processes in the Automotive Industry and Enabling Technologies

Other Running Projects
Enabling Process Flexibility: Challenges

Need for Process Evolution

Need for Monitoring and Analyzing Dynamic Processes

Need for Ad-hoc Deviations

Execution Log

Process engineer / Process administrator

Schema S:

Schema S:

Instance I:

Need for Process Variant Configuration

Enabling Process Flexibility: Challenges

1. Create Process Schema
2. Create Instances
3. Process Execution
4. Execution Log
5. Evolve Process Schema

Process engineer / Process administrator

Need for Ad-hoc Deviations
ADEPT: Ad-hoc Changes

The Users' View

Exceptional case – we need an additional lab test!

ADEPT: Ad-hoc Changes

**System’s View**

### Process Type Level

**Process Schema S**

- Patient Admission
- Anamnesis & Clinical Examination
- X-ray
- Non Operative Therapy
- MRT
- Sonography
- Initial Treatment & Operation Planning
- Operative Treatment
- Discharge & Documentation

**Activities**
- Patient Admission
- Anamnesis & Clinical Examination
- X-ray
- Non Operative Therapy
- MRT
- Sonography
- Initial Treatment & Operation Planning
- Operative Treatment
- Discharge & Documentation

**Decision Points**
- XOR-Split/Join
- AND-Split/Join

**Workflow Rules**
- **XOR Rule:**
  - clinicalSuspicionOfCruciateRupture = "Yes"
- **AND Rule:**
  - cruciateRupture = "Yes" and operationIndicated = "Yes"

### Process Instance Level

**Process Instance I1**

**Execution Trace:**
\[ \sigma_1 = \langle \text{Patient Admission}, \text{Anamnesis & Clinical Examination}, \text{X-ray} \rangle \]

**Process Instance I2**

**Execution Trace:**
\[ \sigma_2 = \langle \text{Patient Admission} \rangle \]

ADEPT: Ad-hoc Changes

**System’s View**

For patient „Mozart“ the MRT activity needs to be skipped due to his cardiac pacemaker.

ADEPT: Ad-hoc Changes

Process Type Level

Process Schema S

- Patient Admission
- Anamnesis & Clinical Examination
- X-ray
- Non Operative Therapy
- MRT
- Sonography
- Initial Treatment & Operation Planning
- Operative Treatment
- Discharge & Documentation

XOR-Split/Join
AND-Split/Join

Process Instance Level

Process Instance I1

Process Instance I2

Execution Trace:
\[ \sigma_1 = \langle \text{Patient Admission}, \text{Anamnesis & Clinical Examination}, \text{X-ray} \rangle \]

Execution Trace:
\[ \sigma_2 = \langle \text{Patient Admission} \rangle \]

**ADEPT: Ad-hoc Changes**

### Process Type Level

#### Process Schema S

- **Patient Admission**
- **Anamnesis & Clinical Examination**
- **X-ray**
- **MRT**
- **Sonography**
- **Non Operative Therapy**
- **Operative Treatment**
- **Discharge & Documentation**

**Activities:**
- Patient Admission
- Anamnesis & Clinical Examination
- X-ray
- MRT
- Sonography
- Non Operative Therapy
- Operative Treatment
- Discharge & Documentation

**Conditions:**
- clinicalSuspicionOfCruciateRupture = „Yes“
- cruciateRupture = “Yes“ and operationIndicated = “Yes”

### Process Instance Level

#### Process Instance I3

**Execution Trace:**

\[ \sigma_3 = \langle \text{„Patient Admission“}, \text{„Anamnesis & Clinical Examination“}, \text{„MRT“}, \text{„X-ray“}, \text{„Sonography“} \rangle \]

- I3 is not compliant with change Delete (I3, MRT)

---

ADEPT: Ad-hoc Changes

Solution for many fundamental research issues!

Formal foundation of the ADEPT technology!

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

Process Instance $I_1$, \texttt{Delete($I_1$, MRT)}

$p_{d_{1}} = \text{The treatment of cruciate ruptures routinely includes a magnetic resonance tomography (MRT), an X-ray and a sonography. However, for a particular patient the MRT may have to be skipped as the respective patient has a cardiac pacemaker.}$

$q_{aSetc1} = \{(\text{Does the patient have a cardiac pacemaker?}, \text{Patient.problemList.hasPacemaker} = \text{"Yes"})\}$

$s_{ol_{c1}} = \text{<Delete($S_i$, MRT)>}$

$f_{req_{c1}} = 1$

Case $c_1$
Derive a new reference process model from the variants such that:

Less adaptations are needed in future!
Process Variants Mining: Supported Scenarios

Scenario 1: No original reference process model available

- Discovered reference process model $S'$

- Mining & learning

- Process variant $S_1$
- Process variant $S_2$
- Process variant $S_3$
- Process variant $S_4$
- Process variant $S_5$
- Process variant $S_n$ ...

Process Repository

Goal: Discover a (new) reference process model which requires less configuration efforts

Scenario 2: Original reference process model known

- Customization & adaptation

- Original reference process model $S$

- Process improvement

- Discovered reference process model $S'$
ADEPT: Process Schema Evolution

1. Need for Process Evolution
2. Create Instances
3. Process Execution
4. Execution Log
5. Process Schema Change
ADEPT: Process Schema Evolution

ADEPT Process Management System

| Std Client API | Web Clnt API | Modeling API | Dyn. Change API | Admin. API | Role Mgmt | Authorization | Time Mgmt | Msg Queuing | Recovery | Audit Trail | ...

Process Designer / Process Administrator

Repository

ADEPT Process Composer

Create Process Template
Modify Process Template
Check Process Template

ADEPT: Extended Process Lifecycle Support

ADEPT: Implementing the Framework

ADEPT: Implementing the Framework

ADEPT: Implementing the Framework

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

ADEPT: Transferring ADEPT to Practice
The AristaFlow BPM Suite

AristaFlow BPM Suite

www.aristaflow-forum.de
Process Flexibility: A Framework

Change Patterns

Process Flexibility: A Framework

Patterns for Decision Deferral

- **Traditional Workflow**: Low Need for User Experience, Low Degree of Decision Deferral. Process model is specified before execution.
- **Multi-instance Activities**: Low Need for User Experience, Low Degree of Decision Deferral. Process model can contain placeholders whose content is specified at run-time.
- **Late Binding**: Low Need for User Experience, High Degree of Decision Deferral. Process model can contain placeholders whose content is selected during run-time.
- **Late Modeling**: High Need for User Experience, Low Degree of Decision Deferral. Process model contains placeholders whose content is modeled during run-time.
- **Late Composition**: High Need for User Experience, High Degree of Decision Deferral. Process model is iteratively composed during run-time considering existing constraints.
- **Number of activity instances can be specified at run-time**: High Degree of Decision Deferral.
Process Flexibility: A Framework

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
## Process Flexibility: A Framework

<table>
<thead>
<tr>
<th>Primitive / Pattern</th>
<th>Academic</th>
<th>Commercial</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>ADEPT2 / CBRFlow</td>
<td>CAKE 2</td>
</tr>
<tr>
<td><strong>Change Primitives</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>PR1 – Add Node</td>
<td>–</td>
<td>+</td>
</tr>
<tr>
<td>PR2 – Remove Node</td>
<td>–</td>
<td>+</td>
</tr>
<tr>
<td>PR3 – Add Edge</td>
<td>–</td>
<td>+</td>
</tr>
<tr>
<td>PR4 – Remove Edge</td>
<td>–</td>
<td>+</td>
</tr>
<tr>
<td>PR5 – Move Edge</td>
<td>–</td>
<td>+</td>
</tr>
<tr>
<td><strong>Adaptation Patterns</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AP1 – Insert Fragment</td>
<td>A[1, 2], B[1,2,3], C[1,2]</td>
<td>–</td>
</tr>
<tr>
<td>AP2 – Delete Fragment</td>
<td>A[1, 2], B[1,2,3]</td>
<td>–</td>
</tr>
<tr>
<td>AP3 – Move Fragment</td>
<td>A[1, 2], B[1,2,3], C[1,2]</td>
<td>–</td>
</tr>
<tr>
<td>AP4 – Replace Fragment</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>AP5 – Swap Fragment</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>AP6 – Extract Fragment</td>
<td>A[1,2], B[3]</td>
<td>–</td>
</tr>
<tr>
<td>AP7 – Inline Fragment</td>
<td>A[1,2], B[2]</td>
<td>–</td>
</tr>
<tr>
<td>AP8 – Embed Fragment in</td>
<td>A[1,2], B[1,2,3]</td>
<td>–</td>
</tr>
<tr>
<td>AP9 – Parallelize Activities</td>
<td>A[1,2], B[1,2,3]</td>
<td>–</td>
</tr>
<tr>
<td>AP10 - Embed Fragment in Conditional Branch</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>AP11 – Add Control Dependency</td>
<td>A[1,2]</td>
<td>–</td>
</tr>
<tr>
<td>AP12 – Remove Control Dependencies</td>
<td>A[1,2]</td>
<td>–</td>
</tr>
<tr>
<td>AP14 – Copy Fragment</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

A current project: C3Pro
A current project: C3Pro
A current project: C3Pro
A current project: C3Pro
A current project: C3Pro
A current project: C3Pro
A current project: C3Pro

- **Soundness** of private processes
- **Consistency** of private processes with public processes
- **Realizability** of choreographies
- **Compatibility** between public processes
Consistency of private processes with public processes

Soundness of private processes

Heterogeneity of Meta-Models

Compatibility Criteria

Realizability of choreographies
A current project: C3Pro

**Rule 1:** After a blood-test the bloodsample has to be destroyed.

**Rule 2:** If complications are detected the patient has to be transferred to the hospital and be observed.
C³Pro
Change and Compliance for Collaborative Processes

Transitivity
Impacts on the soundness of the choreography

Propagation of changes
Negociation?
Process-Aware Information Systems

A Decade of Research on Flexible Process-Aware Information Systems and its Achievements

A Decade of Research on Large Processes in the Automotive Industry and Enabling Technologies

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

Other Running Projects
The Daimler BPM Round Table
Process-Aware Information Systems

A Decade of Research on Flexible Process-Aware Information Systems and its Achievements

A Decade of Research on Large Processes in the Automotive Industry and Enabling Technologies

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

Other Running Projects
The Challenge:
Dealing with Large Process Models
Dealing with Large Process Models: Need for an Advanced Visualization Framework
The Proviado Visualization Framework

abstracting information (process views)

adapt display form (diagram, form, table, text, ...)

adapting visual appearance (symbols, colors, ...)

The Proviado Visualization Framework
Proviado: Process Model Abstraction - Example
Proviado: Process Model Abstraction – Basic Operations (1)

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, …
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
The Proviado Visualization Framework

abstracting information (process views)
adapting visual appearance (symbols, colors, ...)
adapt display form (diagram, form, table, text, ...)
Proviado: Abstraction + Visual Configuration

Personalized Visualization
The Proviado Visualization Framework

abstracting information (process views)

adapting visual appearance (symbols, colors, ...)

adapt display form (diagram, form, table, text, ...)
Proviado: Supporting Different Display Forms for Process Models
The Proviado Visualization Framework: Achievements
Topics we are currently working on …

Process Navigation Support (niPRO project)

(a) geographic navigation dimension

(b) semantic navigation dimension

(c) view navigation dimension "logic-based"

"time-based"
Topics we are currently working on …

Updatable Process Model Abstractions (Process Views)

Core Process Model CPM → A → B → C → D → E →

AggrSESE(V1, {D, E})
RedActivity(V1, B)

Process View V1 on CPM → A → C → DE →

View V1:
InsertSerial(V1, C, DE, Y)
InsertSerial(V1, A, C, X)

CPM:
InsertNode(CPM, A, B, X)
InsertNode(CPM, C, D, Y)
InsertNode(CPM, B, C, X)

proVie
Topics we are currently working on …

Gesture-based Interaction with Process Models
Process-Aware Information Systems

A Decade of Research on Flexible Process-Aware Information Systems and its Achievements

A Decade of Research on Large Processes in the Automotive Industry and Enabling Technologies

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

Other Running Projects
The Challenge: Dealing with Large Process Model Collections
… and a Particular Challenge: Managing Process Variants

a) Standardized Repair Process

b) Variant 1: Simple Problem Repair

c) Variant 2: Security Critical Repair

d) Variant 3: Security Critical and Simple Problem Repair
... and a Particular Challenge: Managing Process Variants

Standard Process:
1. Reception → Diagnosis → Repair → Hand Over
   - Maintain

Variants:
1. Variant 1: Fast Diagnosis
   - Reception → [Diagnosis Shortened] → Repair → Hand Over

2. Variant 2: Security Critical Repair
   - Reception → Diagnosis → Repair → Hand Over

3. Variant 3: Fast Diagnosis and Security Critical Repair
   - Reception → Diagnosis → Repair → Final Check → Hand Over
   - Maintain

... and a Particular Challenge: Managing Process Variants
... and a Particular Challenge: Managing Process Variants

Context Model

Business Area

Daimler Financial Services
Daimler Trucks
Daimler Buses
Mercedes-Benz Vans
Mercedes-Benz Cars

Brand

Fuso
Smart
Mercedes-Benz

Vehicle Type

Truck
Van
Bus
Car

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

INSERT IF country = Italy

Configured process variant

DELETE IF brand = Smart
The Provop Approach for Managing Process Variants
The Provop Approach for Managing Process Variants
Topics we are currently working on, e.g.

VIVACE:


Clara Ayora, Victoria Torres, Barbara Weber, Manfred Reichert
Process-Aware Information Systems

A Decade of Research on Flexible Process-Aware Information Systems and its Achievements

A Decade of Research on Large Processes in the Automotive Industry and Enabling Technologies

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

Other Running Projects
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

**Modellebene**

- **Datenmodell**
  - Data Model
- **Life Cycle Coordination Model**
  - Object Life Cycles / Life Cycle Coordination Model

**Instanzebene**

- **Datensstruktur**
  - Data Structure
- **Datengetriebene Prozessstruktur**
  - Data-driven Process Structure
The Corepro Project – Basic Approach

Object Life Cycles / Life Cycle Coordination Model

Data-driven Process Structure
The Corepro Project – Basic Approach

Data-driven Process Structure
The Corepro Project – Basic Approach

Modellniveau

Datenmodell

- Gesamtsystem
  - hatSys
- System
  - hatKomp
  - nutztKomp
- Komponente

Life Cycle Coordination Model

- Gesamtsystem
  - Release bildung
  - Release gebildet
  - Umbau
  - Mechanik
  - Prototyp umgebaut
  - Rücksprache dokumentation
- System
  - Auswahl
  - Konfiguration gebildet
  - Mustar aufgebaut
  - Test
- Komponente
  - Bestellung
  - Komponente angeliefert
  - Absicherung
  - Abgesichert
  - Fehlermeldungen

Instanzenbene

Datenstruktur

- Gesamtsystem BR212, Rel. 02/08
  - hatSys
- System
  - Telematik High V2.2
  - hatKomp
  - nutztKomp
- Komponente
  - Head-Up Unit V3.14
  - TV Tuner V1.83

Datengetriebene Prozessstruktur

- BR212, Rel. 02/08
  - Release bildung
  - Release gebildet
  - Umbau
  - Mechanik
  - Prototyp umgebaut
  - Rücksprache dokumentation
- Telematik High V2.2
  - Auswahl
  - Konfiguration gebildet
  - Mustar aufgebaut
  - Test
- Head-Up Unit V3.14
  - Bestellung
  - Komponente angeliefert
  - Absicherung
  - Abgesichert
  - Fehlermeldungen
- TV Tuner V1.83
  - Bestellung
  - Komponente angeliefert
  - Absicherung
  - Abgesichert
  - Fehlermeldungen
- 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
Topics we are currently working on, e.g.

The PHILharmonicFlows Project

Object-Aware Process Management

Vera Künzle, Manfred Reichert & Persis GmbH
Object-Aware Process Management

- Modeling object behavior and interactions
- Runtime interpretation based on a precise operational semantics
- Automated generation of end-user components (e.g. forms)
Object-Aware Process Management

Data Model
- Job Offer
  - identifier
  - description
  - vacant from
  - vacant until
- Application
  - name
  - e-mail
  - appl. letter
  - decision
  - evaluation
- Review
  - priority
  - return date
  - remark
  - proposal
  - evaluation
  - reason
  - comment
  - committed
- Interview
  - date
  - time
  - location

Process Model
- Job Offer
  - create → publish → analyze → fill
- Application
  - fill in → send → check
  - accept
  - reject
- Review
  - initiate → fill in → complete
- Interview
  - initiate → perform
Object-Aware Process Management: Object Behavior

User Integration

- **staff member**
  - department

- **personnel officer**
  - human resources

Form Logic

<table>
<thead>
<tr>
<th>Priority</th>
<th>Remark</th>
<th>Proposal</th>
<th>Evaluation</th>
<th>Reason</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Object Instance

<table>
<thead>
<tr>
<th>Review</th>
</tr>
</thead>
<tbody>
<tr>
<td>remark</td>
</tr>
<tr>
<td>return date</td>
</tr>
<tr>
<td>priority</td>
</tr>
<tr>
<td>proposal</td>
</tr>
<tr>
<td>evaluation</td>
</tr>
<tr>
<td>reason</td>
</tr>
<tr>
<td>comment</td>
</tr>
<tr>
<td>committed</td>
</tr>
</tbody>
</table>

Process Instance

- **initiate**
- **fill in**
- **complete**

- **staff member**
  - department

Object-Aware Process Management: Object Behavior
Object-Aware Process Management: Object Interactions

Data Model

- Job Offer
  - identifier
  - description
  - vacant from
  - vacant until

- Application
  - name
  - e-mail
  - appl. letter
  - decision
  - evaluation

- Review
  - priority
  - return date
  - remark
  - proposal
  - evaluation
  - reason
  - comment
  - committed

- Interview
  - date
  - time
  - location

Process Model

Job Offer
- create
- publish
- analyze
- fill

Application
- fill in
- send
- check
- accept
- reject

Review
- initiate
- fill in
- complete

Interview
- initiate
- perform
Object-Aware Process Management: Flexible Activity Execution

User Integration

Staff member
Department

Context-sensitive activity

Job Offer
- identifier: salesman
- description:
- vacant from: 10.09.2011
- vacant until: 01.01.2012

Application
- decision: reject

Batch activity

Review
- priority: high
- return date: 12.12.2011
- remark: complete soon
- proposal: invite
- evaluation: very good
- reason:
- comment: many competencies

Process Model

Job Offer
- create
- publish
- analyze
- fill

Application
- fill in
- send
- check
- accept
- reject

Review
- initiate
- fill in
- complete

Interview
- initiate
- perform
Object-Aware Process Management System

- **Overview Lists**
- **Worklists**

Activities
- optional
- mandatory

**Program Code**
- Function Logic

**Data Models**
- Relations
- Object Types
- Attributes

**Process Models**
- Object Interactions
- Object Behavior
- Activities
Object-Aware Process Management

- integrated access
- object interactions
- data-driven execution
- flexible activity granularity
- object behavior
Process-Aware Information Systems

A Decade of Research on Flexible Process-Aware Information Systems and its Achievements

A Decade of Research on Large Processes in the Automotive Industry and Enabling Technologies

Other Running Projects
Other Running Projects

- ATAPIS (Adaptive Time- and Process-aware Information Systems)
- C³Pro (Enabling Change and Compliance for Collaborative Processes)
- MARPLE (Managing Robust Mobile Processes in a Complex World)
- niPRO (Personalized and Intelligent Process Portals)
- PHILharmonic Flows (Linking Processes, Humans and Information)
- PROCEED (PROactive Consistency for EE product Data management)
- proCollab (Process-aware Support for Collaborative Knowledge Workers)
- proView (Personalized and Updatable Process Visualizations)
- Qube (Quality Assessment and Management in Business Process Design, Implementation and Enactment)
- QuestionSys (A Generic and Flexible Questionnaire System Enabling Process-Driven Mobile Data Collection)
- SOPHINA (Software Products and Processes with Integrated Variability Support)
- SustainHub (Sustainability Data Exchange Hub)
Structuring of our Research

Research Topics

- Adapativity & Flexibility
- Healthcare & Psychology
- Mobile Processes
- Automotive Engineering
- Business Process Variability
- Software Engineering
- Data- and Object-Aware Processes
- Product Lifecycle Management
- Human-Centric Processes
- Sustainable Data
- Smart Processes

Application Areas

- Methods, Concepts and Technologies for Next Generation Process Management Technology
- ...