A Thing Called “Fluid Process”
Beyond Rigidity in Business Process Support

Manfred Reichert
Motivation

Permanent new "trends" – require new or adapted services
... which must be integrated

Issues:
- How quickly can healthcare processes be implemented?
- At which costs? – With which error risk?
- How expensive will later process changes be?
- How to avoid the "maintenance trap"?

Need for Process-awareness
Motivation

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 Execution Engine

Process 1  Process 2  Process 3  Process 4
Process 5  Process 6  Process 7  Process 8
Process 13  Process 14

Deploy Process

Users

Process Designer
Motivation

Schema S:

A → B → C → D → E

Process designer / Process administrator

Create Process Schema

Create Instance

Process Log

Process Monitoring

Process Configuration & Implementation

Process Execution

Process Monitoring
Motivation

- Today’s BPM tools are ill equipped to meet the aforementioned challenges due of their inherent brittleness and inflexibility

- Current tool generation implicitly embraces the “engineer – use “ dichotomy inherited from traditional SE approaches; i.e., systems are first “engineered” and then “used” (or “operated”)

- Maintenance and evolution activities are not regarded as part of operation, but rather as interruptions to the “in use” state

- Role of end users and process actors is not well understood!
Real-world processes are “fluid”!

Why process instances need to be dynamically adaptable?
Why Process Instances Need to be Dynamically Adaptable?

Do we believe …

- that processes in the transportation domain can be completely pre-modeled?
Why Process Instances Need to be Dynamically Adaptable?

Or do we really believe …

- that process-aware information systems (PAIS) can prescribe to a physician how to treat his or her patients?
Why Process Instances Need to be Dynamically Adaptable?

Or do we really believe …

- that long-running engineering processes can be completely pre-modeled?

- Example: Release management for E/E-systems in a car
  - 200 - 300 control devices to be systematically tested and released
  - Requires the execution of hundreds up to thousands of processes
  - Concurrent engineering complex dependencies have to be considered
Why Process Instances Need to be Dynamically Adaptable?

- The only feasible way to cope with these challenges is to dissolve the fundamental distinction between “engineering” and “use”; i.e., end users must be empowered to dynamically adapt or evolve processes.

- This will lead us to a new class of processes – so called fluid processes whose “engineering” and “use” is intertwined.

- Fluid processes are continually being adapted and reformed to fit the actual needs and constraints of the situation in hand and to fulfill the overall goals of the involved organizations in the best possible way.
Adaptive Process Management Technology for Enabling Fluid Processes at Runtime
Ad-hoc Process Change

### Process Type Level

**Process Schema S**

- **Patient Admission** → **Anamnesis & Clinical Examination**
- **X-ray** → **Non Operative Therapy**
- **MRT** → **Non Operative Therapy 1**
- **Sonography** → **Initial Treatment & Operation Planning**
- **Operative Treatment** → **Discharge & Documentation**

- **clinicalSuspicionOfCruciateRupture = „Yes“**
- **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: $\sigma_2 = \langle \text{„Patient Admission“} \rangle$
Ad-hoc Process Change

Process Type Level

Process Schema S

Patient Admission → Anamnesis & Clinical Examination → X-ray → Non Operative Therapy

clinicalSuspicionOfCruciateRupture = "Yes" → MRT → Sonography → Non Operative Therapy 1

Process Instance Level

Process Instance I1

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

Ad-hoc Flexibility: Deviations, Change

For patient "Oberweis" the MRT activity needs to be skipped due to his cardiac pacemaker.

Execution Trace:
\[ \sigma_2 = \text{"Patient Admission"} \]
Ad-hoc Process Change

Ad-hoc Flexibility: Deviations, Change

Process Type Level

Process Schema S

Patient Admission → Anamnesis & Clinical Examination → X-ray → Non Operative Therapy → Non Operative Therapy 1

clinicalSuspicionOfCruciateRupture = "Yes"

Patient Admission → MRT → Non Operative Therapy

Patient Admission → Sonography

X-ray → Non Operative Therapy

MRT

Non Operative Therapy 1

Discharge & Documentation

Initial Treatment & Operation Planning

Operative Treatment

cruciateRupture = "Yes" and operationIndicated = "Yes"

Process Instance Level

Process Instance I1

Process Instance I2

Execution Trace:

$\sigma_1 = \langle \text{Patient Admission}, \text{Anamnesis & Clinical Examination}, \text{X-ray} \rangle$

$\sigma_2 = \langle \text{Patient Admission} \rangle$
Ad-hoc Process Change (User’s View)

Exceptional case – we need an additional lab test!
Ad-hoc Process Change (User‘s View)

Examinations
- U Wallace, Edgar
- U Miller, Anne
- U Smith, Karl
- U Jones, Isabelle

Select Activity
- Schedule counsel examination
- Lab Test
  - Prepare patient for operation
  - Inform patient
  - Wash patient
  - Schedule examination date

Exceptional case – we need an additional lab test!
Ad-hoc Process Change (User‘s View)

The Users' View

Start immediately, results are needed before explanation of operation risks
Ad-hoc Process Change (User‘s View)

Great!!

The Users' View
Ad-hoc Process Change (User‘s View)

The Users' View

Examinations
U Wallace, Edgar
U Miller, Anne
U Smith, Karl
U Jones, Isabelle

OK, now let us continue with the examination!
Ad-hoc Process Change: Correctness

Process Type Level

Process Instance Level

Execution Trace:
\( \sigma_3 = < \text{Patient Admission}, \text{Anamnesis & Clinical Examination}, \text{MRT}, \text{X-ray}, \text{Sonography}> \)
Ad-hoc Process Change: User Assistance

- 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 $I_1$ Delete($I_1$, MRT)

$pd_{c1}$ - 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.
$qSetc1 = \{(\text{Does the patient have a cardiac pacemaker?}, \text{Patient.problemList.hasPacemaker = 'Yes'})\}$
$sol_{c1} = \langle \text{Delete}(S_1, \text{MRT}) \rangle$
$freq_{c1} = 1$
The ADEPT Approach

- Ad-hoc Changes
- Response Times
- Propagation of Process Type Changes
- User Interface / API

- (Formal) Process Meta Model
- Process Fragment Composition
- Process Type Changes

- Architecture And Implementation
- Transactional Support / Semantic Rollback

- Support of Temporal Constraints
- Efficient Execution
- Component-oriented Software Development
- Scalability / Distributed Process Mgmt

- Transactional Support / Semantic Rollback
- Scalability / Distributed Process Mgmt
- Component-oriented Software Development
- Efficient Execution

- Support of Temporal Constraints
- Efficient Execution
- Component-oriented Software Development
- Scalability / Distributed Process Mgmt

- Architecture And Implementation
- Transactional Support / Semantic Rollback
- Scalability / Distributed Process Mgmt
- Component-oriented Software Development

- Support of Temporal Constraints
- Efficient Execution
- Component-oriented Software Development
- Scalability / Distributed Process Mgmt

- Architecture And Implementation
- Transactional Support / Semantic Rollback
- Scalability / Distributed Process Mgmt
- Component-oriented Software Development

- Support of Temporal Constraints
- Efficient Execution
- Component-oriented Software Development
- Scalability / Distributed Process Mgmt
The ADEPT Approach
The ADEPT Approach

- (Formal) Process Meta Model
- Process Fragment Composition
- Process Type Changes
- Architecture And Implementation
- Ad-hoc Changes
- Response Times
- Propagation of Process Type Changes
- User Interface / API
- Transactional Support / Semantic Rollback
- Support of Temporal Constraints
- Efficient Execution
- Component-orientied Software Development
- Scalability / Distributed Process Mgmt
The ADEPT Approach

ADEPT:

Individually adaptable Process Instances

Process Instance

= (individual) "Process Program"
The ADEPT Approach

ADEPT:
Individually adaptable Process Instances
Process Instance
= (individual) "Process Program"

Achievements:
- Formal process meta model (expressive + restricted enough)
- Formal Criteria for Change Correctness (incl. "Theorems & Proofs")
- Efficient, build-in consistency checks ("no bad surprise")
- Support of a high number of change patterns
- API for accomplishing ad-hoc changes
The ADEPT Approach

AristaFlow BPM Suite

www.aristaflow-forum.de
Applications

Applying the ADEPT / AristaFlow Technology in Practice
Enabling “Fluid Processes” with ADEPT: The Spot Project

Flexible Support of Clinical Pathways with ADEPT

Partners:
Jan Neuhaus, Claudia Reuter
Fraunhoferinstitut Dortmund

Diagram:

1. Lab specimen delivery
2. Anamnesis
3. Examination
4. Sonography
5. Lab test 1
6. Lab test 2
7. Consider findings
8. Susicion for urinary stone?
9. Abdominal X-ray
10. End

Conditions:
- AND-Split
- AND-Join
- OR-Split
- OR-Join

Urinary_Stones_Diagnostics
Enabling “Fluid Processes” with ADEPT: The Spot Project

Clinical pathways constitute “Fluid Processes” which need to be statically and/or dynamically configured to fit to the patient’s current situation!

Patient is pregnant; process variant „pregnancy“ is chosen

Increased blood sugar level; process variant „Diabetes“ is chosen

New process fragments are inserted in parallel!
Enabling “Fluid Processes” with ADEPT: The Spot Project

The Process Structure Tree - Providing abstraction to end users
Enabling “Fluid Processes” with ADEPT: The Spot Project

The *Process Structure Tree* representing the patient-specific pathway!
Enabling “Fluid Processes” with ADEPT: The Spot Project

Proof-of-Concept Implementation Based on the ADEPT System
Enabling “Fluid Processes” with ADEPT: The MTCT Project

Possible unexpected events: new client request arrival, delayed vehicles, technical problems

The MTCT System for Managing Container Transportation

Partners: University of Montreal
Enabling “Fluid Processes” with ADEPT: The MTCT Project

The MTCT System for the Flexible Management of Container Transportation

- Based on a transportation system framework

- Functionality:
  - Modeling and enacting the processing of client requests for container transportation (i.e., complex processes)
  - Tracking and monitoring the progress of the processes accomplishing these client requests
  - Identifying the activities to be (dynamically) composed and executed; e.g., attach/detach container to/from vehicle, move vehicle to location, load/unload container, wait at location

- Processing of client requests for container transportation
  - Dynamic creation and adaptation of specific sequences of interdependent activities; Use of ADEPT technology
  - Dynamic structural modifications of process instances, e.g., adding a transfer to an already planned client request processing
Enabling “Fluid Processes” with ADEPT: The MTCT Project

Proof-of-Concept Implementation Based on the ADEPT System

Enabling “Fluid Processes” with ADEPT: The Consensus Project

The Consensus Approach for Supporting E-Negotiations

Partners: University of Montreal
Enabling “Fluid Processes” with ADEPT: The Consensus Project

Proof-of-Concept Implementation Based on the ADEPT System

- Supports dynamism in (combined) e-negotiations; it is possible to dynamically …
  - insert a new negotiation
  - move a negotiation
  - remove already scheduled activities

Enabling “Fluid Processes” with ADEPT: Disaster Management

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

Semantically Constraining Possible Adaptations in Fluid Processes

Linh Thao Ly
thao.ly@uni-ulm.de
How to model semantic constraints, such as compliance rules, adequately?

How to ensure that process models meet compliance requirements?

How to ensure that flexible ad-hoc process adaptations do not introduce incompliance?

How to trace compliance violations and benefit from past process executions?

Constraint modeling

Intuitive modeling of semantic constraints using a visual logic-based formalism

Process design

Enrichment of process models with semantic constraints
Verification of process models against imposed constraints using graph algorithms and model checking (static verification)
Helpful validation reports and visualization of counterexamples

Process execution

Runtime verification of dynamic constraints (dynamic verification)
Integrated verification of ad-hoc process changes with regard to compliance
Flexible overriding of constraints if necessary

Process evaluation

Documentation of constraint violations and constraint overriding enables traceability and process mining

some impressions

SeaFlows Project – Some Impressions

Condition

After a surgery, an appointment is necessary before the patient is discharged.

Condition

Prior to a CT, the patient has to be informed and after the CT the results have to be reported to the patient.

Condition

Prior to an examination of a patient aged beyond 75, an additional tolerance test must be performed.
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

Prior to an examination of a patient aged beyond 75, an additional tolerance test must be performed.
Changes in the Jamuna river (a branch of the Brahmaputra) in Bangladesh between March 1987 (shown in dark blue) and March 1989 (shown in light blue) and superimposed on a SPOT satellite basemap. Change monitoring made it possible to model the river's course and behaviour and to undertake preliminary studies to control flooding.
Monitoring and Mining “Fluid Processes”

1. Create Process Schema
2. Create Instances
3. Process Execution
4. Instance-specific Change
Execution and Change Logs of “Fluid Processes”

Original Schema S

<table>
<thead>
<tr>
<th>Activity</th>
<th>Event</th>
<th>User</th>
<th>Timestamp</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Started</td>
<td>Garry</td>
<td>2007/09/08 15:00</td>
</tr>
<tr>
<td>A</td>
<td>Started</td>
<td>Garry</td>
<td>2007/09/08 15:30</td>
</tr>
<tr>
<td>A</td>
<td>Completed</td>
<td>Garry</td>
<td>2007/09/08 15:45</td>
</tr>
<tr>
<td>B</td>
<td>Started</td>
<td>Helen</td>
<td>2007/09/10 11:00</td>
</tr>
<tr>
<td>X</td>
<td>Started</td>
<td>Fritz</td>
<td>2007/09/11 09:01</td>
</tr>
</tbody>
</table>

Change Log Instance 4711 on Schema S

Change TX Applied Changes : User:Timestamp

001 InsertFragment[S;X,A,C]:Helen:2007/09/10 12:02
002 ReplaceFragment(S;C,Z):Jim:2007/09/11 09:31

Process Instance 4711

2007/09/10 11:00

2007/09/10 13:00

2007/09/11 10:00
Change Analysis – A Simple Approach

Change Logs

Phase I

Change Processes
Change Analysis – A Simple Approach

The discovered meta change process covers all changes applied to at least one of the given fluid process instances.


Change Analysis – A Simple Approach (Proof-of-Concept Prototype)

Change Mining Plugin in ProM

Change Logs Imported from ADEPT
A More Advanced Approach: Process Variants Mining

1. Configurations are very costly
2. Variants are difficult to maintain

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

Less adaptations are needed in future!

Process Variants Mining: Basic Goal

How to discover a reference process model by mining a collection of process (instance) variants in order to reduce the need of future process adaptations?
Process Variants Mining: Bias and Distance

- **Process Bias**: Minimal set of high-level change operations needed to transform a given process model S into another model S'.

- **Process Distance**: # change operations of any bias between S and S'; can be used to measure the complexity for process change.

---

Process Variants Mining: Reformulated Basic Goal

How to derive a reference process model by mining the a collection of process (instance) variants which has minimal average distance to the process variants?

When representing a process (instance) variant as a “dot” in a 2-dimensional space, discovering a reference model logically corresponds to finding the “center” (for which the average distance is minimal to all “dots”).
**Example: Clinical Pathways**

1. **Formalization**
   - *Formalized medical guideline G*

2. **Dissemination**
   - *Disseminated medical guideline G*

3. **Adaptation**
   - *Medical pathway schema P’*
   - *Process instantiation*

4. **Process instantiation**
   - *Actual treatment process for patient p1*

5. **Process instance change**

6. **Process execution**
   - *Medical Staff*
   - *Notify process engineer*

7. **Feedback**
   - *Change pathway schema*

8. **Create new pathway schema**

9. **Process engineer**

10. **Site-specific**
    - *Process engineer*
    - *Notify process engineer*

11. **Patient treatment-specific**
    - *Individual treatment plan for patient p1*

12. **Domain-specific**
    - *Medical Guideline (narrative)*

Diagram:
- Process instantiation from process engineer to medical pathway schema P.
- Process execution from medical staff to actual treatment process for patient p1.
- Feedback from process instances to change logs.
- Notify process engineer from patient treatment-specific to domain-specific.
- Site-specific feedback to adaptation.
Enabling Data-driven Process Structures with COREPRO
**The COREPRO Approach: Motivation**

Process structure needs to be adapted when product structure changes!
The COREPRO Approach: Motivation
The COREPRO Approach: Motivation

Corepro: Integrated Support of Data-driven Process Structures
The COREPRO Approach: Data-driven Process Structures

Modellebene

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

Instanzebene

- Datenstruktur
  - Data Structure
- Datengetriebene Prozessstruktur
  - Data-driven Process Structure
The COREPRO Approach: Data-driven Process Structures

Object Life Cycles / Life Cycle Coordination Model

Data-driven Process Structure
The COREPRO Approach: Data-driven Process Structures
The COREPRO Approach: Data-driven Process Structures
The COREPRO Approach: Data-driven Process Adaptations

**Change Operation (Data Structure)**
1) `removeRelation(Telematik High V2.2, TV Tuner V1.83, nutztKomp);`
2) `removeObject(TV Tuner V1.83);`

**Change Operation (Process Structure)**
1) `removeExtTrans(Telematik High V2.2, Muster Aufgebaut, Installieren, TV Tuner V1.83 . E);`
2) `removeOLC(Tuner V1.83);`
The COREPRO Approach: Proof-of-Concept Implementation

Automatically creating and executing a process structure

[Diagram of COREPRO interface showing process structures and interactions]
Corepro: Case Study  ISO 26262 (Road Vehicles – Functional Safety)

Instance Level: Data Structure and Automatically Created Process Structure

PD = Product Development, HW = Hardware, SW = Software
Philharmonic Flows

Object-aware Processes
Data Handling in Existing WfMS

1. access on data only during the execution of activities
2. missing or incomplete context information
3. no optional activities
4. no control whether the semantic goals are reached
5. each process instance is executed in isolation
6. inadequate asynchronous execution of sub-processes

- generic functions
- comprehensive lifecycle support
- missing data-oriented view
Philharmonic Flows: Object-aware Process Management

data-oriented view

<table>
<thead>
<tr>
<th>application</th>
</tr>
</thead>
<tbody>
<tr>
<td>applicant</td>
</tr>
<tr>
<td>CV</td>
</tr>
<tr>
<td>cover letter</td>
</tr>
<tr>
<td>send</td>
</tr>
<tr>
<td>evaluation</td>
</tr>
<tr>
<td>decision</td>
</tr>
<tr>
<td>confirmation</td>
</tr>
<tr>
<td>location</td>
</tr>
<tr>
<td>comment</td>
</tr>
<tr>
<td>...</td>
</tr>
</tbody>
</table>

process-oriented view

```
application
CV != null
!
submit != null
checked = true

CV != null
!
submit != null
checked = true

decision = 'accept'
confirm. = true

decision = 'reject'

make reviews

proposal != null
submitted = true
```
Philharmonic Flows: Object-aware Process Management

Challenge 1: Integrated View

Challenge 2: Clear Granularity!

Challenge 3: Synchronization

Challenge 4: Data-centered Paradigm

Challenge 5: Flexibility
Summary & Outlook
Summary & Outlook

- Business conditions vary with innovation pressure
- Business objectives vary with business conditions
- Business processes vary with business objectives
- Changing business processes will be a common business process in the future
Contributors

Hilmar Acker
Colin Atkinson
Thomas Bauer
Sarita Bassil
Reinhold Bentele
Marco Berroth
Florian Berle
Ulrich Bestleisch
Reiner Blaser
Ralph Bobrik
Lianne Bodenstaff
Elmar Braig
Florian Breier
Stephan Buchwald
Carolina Chiao
Peter Dadam
Ralph Deininger
Christian Ergenscheifer
Alexander Forschner
Thomas Fries
Kevin Göser

Martin Grimm
Christian Günther
Alena Hallerbach
Christian Heinlein
Clemens Hensinger
Joachim Herbst
Carolin Hitzler
Martin Jurisch
Ralf Kießling
Martin Kirch
Ashim Klotz
David Knupfel
Jens Kolb
Ingred Konyan
Utzih Kreher
Markus Kubicek
Vera Küpper
Andreas Lanz
Markus Lauer
Chao Li
Ahmet Macar
Udo Martschat
Joachim Meyer
Ralf Michel
Florian Micheler
Tiberius Mihai
Max Moldmann
Dominik Müller
Nicolas Mundbrot
Bela Mutschler
Michael Räuber
Andreas Neubert
Romy Spitz
Jakob Pinggara
Stanislav Pokrov
Michael Predeschil
Rüdiger Pryss
Manfred Rechert
Stefanie Rühle-Ma
Ulrich Sauer
Birgit Schulmei

Reiner Siebert
Sonja Sparr
Martin Steinle
Dietmar Stoll
Johannes Stöhr
Lucineia Thom
Julian Tiedeken
Wil van der Aalst
Marco Waim
Barbara Weber
Peter Weißbach
Ursula Wiedemuth
Warner Wild
Jonas Wolz
Andreas Wombacher
Jochen Zeitler
Xiaodong Zhao
Stefan Zugal
…
Summary & Outlook

Flexibility Support in most existing PAIS is like Teenager Sex!!

It’s on everyone’s mind all the time.
Everyone’s talking about it all the time
Everyone’s thinks everyone is doing it.
Almost no one is really doing it.
The few who are doing it:
   Do it poorly
   Think “sure it will be better the next time”.
   Are not practicing it safely

Everyone is bragging about their successes all the time, although very few have actually had any success

Anonymous