A Unified Framework for Modeling
Cooperative Design Processes and Cooperative Business Processes

Colette Rolland, Selmin Nurcan, Georges Grosz
Université Paris 1 - Panthéon - Sorbonne

The 31st Hawaii International Conference on System Sciences
Big Island, Hawaii
6-9 January 1998

Introduction and context of work

Old focus of methods : Product - desired result

New focus of methods : Process - the route followed to reach the result

Methods need to propose PROCESS GUIDANCE
Provide advice on what to do, why and how to do it

Method EKD : Enterprise Knowledge Development
- a set of models (Goal, Process, Object and Rule)
- A way-of-working
- A set of tools supporting the way-of-working

ESPRIT Project ELEKTRA (EEC founding) : Designing a
tool set for re-organising electricity companies
(e.g. towards EEC deregulation rules)
**A meta-model for cooperative processes**

Meta-model: *basis for process model definition*

<table>
<thead>
<tr>
<th>Knowledge required to design ways-of-working</th>
<th>Process meta-model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ways-of-working</td>
<td>Process model</td>
</tr>
<tr>
<td>Traces</td>
<td>Process execution</td>
</tr>
<tr>
<td></td>
<td>Instance of procedure</td>
</tr>
</tbody>
</table>

Output: product

---

**The concept of context**

- A **situation** is a part of the product it makes sense to make a decision on.
- A decision reflects a choice that a user can make at a given moment in the process. A decision refers to an intention.
- An **intention** expresses what the user wants to achieve, it is a goal.
**Different types of contexts**

- Decisions have consequences which differ from one granularity level to another.
- The different contexts are classified according to their consequences.

**The concept of role**

In a given situation, a user has an intention (according to his/her role in this process), that makes him/her progress in the cooperative process.
**Executable context**

At the most detailed level, the execution of any process can be seen as a set of transformations performed on the product.

< (Situation); Intention ; Role >

< (Reservation request+resources) ; Create reservation ; Reservation_service_clerk >

An **executable context** implements a decision, its intention is realised by an action.

---

**The concept of action**

Performing an action changes the product and may generate **new situations**.
The choice context

- A **choice context** corresponds to a situation which requires the exploration of alternatives in decision making.

- A choice context offers a choice among a set of strategies, all of them achieving the same purpose.

\[
\text{<(Request + Resources); Adapt_request ; Reservation_Service_Manager>}
\]

C1: resort and date can not be changed

\[
\text{<(Request + Resources); Change_hotel_category ; Reservation_Service_Manager>}
\]

C2: resort and hotel category can not be changed

\[
\text{<(Request + Resources); Change_dates ; Reservation_Service_Manager>}
\]

C3: hotel category and date can not be changed

\[
\text{<(Request + Resources); Change_resort ; Reservation_Service_Manager>}
\]
The plan context

A plan context is an abstraction mechanism by which a context viewed as a complex issue can be decomposed in a number of sub-issues.

\begin{align*}
\langle \text{Request} + \text{Resources}; \text{Satisfy}_{\text{request}} ; \text{Reservation}_{\text{Service Team}} \rangle \\
\langle \text{Request} + \text{Resources}; \text{Analyse}_{\text{Request}} ; \\
\quad \text{Reservation}_{\text{Service Clerk}} \rangle & \quad \langle \text{Analysed/Adapted Request} + \text{Resources}; \\
\quad \text{Create}_{\text{Reservation}} ; \\
\quad \text{Reservation}_{\text{Service Clerk}} \rangle \\
\langle \text{Analysed Request} + \text{Resources}; \\
\quad \text{Adapt}_{\text{Request}} ; \\
\quad \text{Reservation}_{\text{Service Manager}} \rangle
\end{align*}

The ordering of the contexts, within a plan, is defined by a precedence graph.

A1: the request can be immediately satisfied
A2: the request is not appropriate
**The concept of way-of-working**

- The basic building block of a way-of-working is an instance of context.

- Contexts in the meta-model have hierarchical relationships of two different types: decomposition and refinement.

- A way-of-working, is a grouping of contexts based upon these links. The modules resulting from this grouping are hierarchies of contexts called *trees*.

- A way of working can be composed of several trees: a *forest* of trees.

**The EKD way-of-working for participative design**

A participative design process is a *decision making process*  
*i.e.* a non deterministic process

A participative design process cannot be an ad-hoc and chaotic process

A participative design process can follow a *pattern for decision making*

The EKD process is guided using a repository of contexts which acts as guidelines

A guideline supports decision making, proposing different alternatives and argumented ways to fulfil a given intention in a given situation
The EKD process is guided

An intention and situation driven process

A context : \( \langle \text{situation, intention, role} \rangle \) is either a plan, a choice or an executable

Guidance in the EKD process

The repository of contexts is structured in three layers, corresponding to three types of guidance

Generic guidance

EKD guidance

Domain specific guidance
**Domain specific guidance**
- Grounded on experience, to reuse and/or to adapt concrete and tested solutions in the same domain

<situation G1 : minimize risk of befalling aircrafts
Intention : reduce goal>

CC1

<situation G1 : minimize risk of befalling aircrafts
Intention : Perform AND reduction
with G2 : decrease risk of human errors
and G3 : maintain separation standards>

CC1 : the rate of human errors is below
the international norm N° 1234

---

**EKD guidance**
- based on EKD knowledge used specifically for models construction:

<table>
<thead>
<tr>
<th>Context</th>
<th>Technique</th>
<th>(1) Model the current enterprise state</th>
<th>(2) Acquire goal</th>
<th>(3) Operationalize goal</th>
<th>(4) Generate design model</th>
<th>(5) Validate design model</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>EM construction strategy</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>SWOT analysis</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>Categorizer</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>Goal template</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>Pair-Wise comparison</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>Theorem proof</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>Do/Choose/Pan strategies</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>Goal template</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>Brainstorming strategies</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>Graph construction strategies</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>View models</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>Scenario</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>Argument based reasoning</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>Cost/resource evaluation</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>Agent Dependency model</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
<tr>
<td></td>
<td>Process trace</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
<td>*</td>
</tr>
</tbody>
</table>

- to represent the current enterprise state,
- the goals for change
- the future enterprise state implementing the goals for change
**EKD guidance**

Example of a guideline supporting the process of goal operationalization

<situation>: goal G1;  **Intention**: reduce G1; EKD engineer

- Do exist cases for the achievement of G1?
- Are there milestones to achieve G1?
- Are there different agents involved in the achievement of G1?

<situation>: goal G1; intention: Use an OR reduction of G1 based on each case>

<situation>: goal G1; intention: Use an AND reduction based on agent roles>

<situation>: goal G1; intention: Use an AND reduction based on each milestone>

---

**Generic guidance**

<situation>: "Input context", intention: progress, EKD engineer

<situation>: "Input context", intention: use the do strategy>

<situation>: "Input context", intention: use the choose strategy>

<situation>: "Input context", intention: use the plan strategy>

<situation>: "Input context", intention: use the brainstorm strategy>

- **Pro argument for CC1**: Fulfilling the input context’s intention is possible by specifying design action(s) to be performed on the product
- **Pro arguments for CC2**: It exists several alternative ways to fulfil the input context’s intention
- **Cons arguments for CC2**: Fulfilling the input context’s intention is possible by specifying design action(s) to be performed on the product

- **Pro arguments for CC3**: The achievement of the input context’s intention requires a composite decision making process to take place
- **Cons arguments for CC3**: It exists several alternative ways to fulfil the input context’s intention

- **Pro arguments for CC4**: ...

It is the «default» option used when no over type of guidance applies...
An example of EKD guidance

- (Goal), Reduce goal
  - (Goal), use a milestone-driven strategy
    - (Sub-goals identified), perform AND reduction
      - (Milestones identified), associate a sub-goal with each of them
  - (Goal), use an agent-driven strategy
    - (Sub-goals identified), perform AND reduction
      - (Roles identified), associate a sub-goal to each role

- (Goal), use an agent-driven strategy based on scheduled cooperation
  - (Schedule identified), perform AND reduction between sub-goals
    - (Sub-goals identified), identify schedule
  - (Goal), use a case-driven strategy
    - (Goal), use a case-driven strategy based on dependent situations
      - (Sub-goals defined), perform OR reduction with sub-goals
        - (dependent situations identified), define sub-goals for each situation
      - (Goal), identify dependent situations
    - (Goal), use a case-driven strategy based on independent situations
      - (independent situations identified), define sub-goals for each situation
      - (Sub-goals defined), perform OR reduction with sub-goals
    - (Goal), identify independent situations
    - (Goal), use a case-driven strategy based on dependent intentions
      - (dependent intentions identified), define sub-goals for each intention
      - (Sub-goals defined), perform AND reduction with sub-goals
    - (Goal), identify dependent intentions
    - (Goal), use a case-driven strategy based on independent intentions
      - (independent intentions identified), define sub-goals for each intention
      - (Sub-goals defined), perform AND reduction with sub-goals
    - (Goal), identify independent intentions

Adapted from Dardenne et al. 1993 [12]
**Other properties of the EKD process**

The EKD process is **incremental** and **dynamic** in nature.

The EKD process is **supported by software tools**.

---

**Conclusion**

*The EKD decision making pattern is*:  
- a reasoning mechanism  
- supporting decision making  
- using a repository of guidelines  
- for guiding the participative design process

*The repository contains*:  
- domain specific guidelines  
- EKD specific guidelines  
- One generic guideline

*The cooperative process meta-model allows*:  
- to represent cooperative design processes  
- to represent cooperative business processes  
  using a single notation

*Current work*:  
- Development of a WEB based tool allowing:  
  - to use the guidelines  
  - to trace the process