Case Management Model and Notation - CMMN

Knut Hinkelmann
Case Management and Case Modeling

- Any individual Case may be resolved in a completely ad-hoc manner.

- But as experience grows in resolving similar Cases over time, *a set of common practices* can be defined for Cases.

- This becomes the practice of Case Management.

- Case management requires modeling which can express the essential flexibility that human Case workers require for:
  - run-time planning for the selection of Tasks for a Case,
  - run-time ordering of the sequence in which the Tasks are executed,
  - ad-hoc collaboration with other knowledge workers on the Tasks.
Ad hoc Processes in BPMN

- Adhoc (sub)processes – marked with a tilde (~) - are a way to represent cases in BPMN.

- There is no specific ordering or obvious decisions. The activities in an ad hoc process can occur:
  - in any order
  - In any frequency

- Typically, the activities in an ad hoc process involve human performers to make decisions as to what activities to perform, at which time and how many times.

- It is possible, however, to use occasional sequence flow between some activities, but sequence flow does not imply that there are explicit start and end events.
Example of an Ad hoc Process
CMMN - Case Management Model and Notation

- OMG defined a Modeling Standard for Case Modeling
  - Case Management Model and Notation (CMMN)

- In January 2013 OMG published the first beta version
  - http://www.omg.org/spec/CMMN/1.0/
  - It is referenced in this presentation as *(CMMN 1.0)*

- CMMN is specialized notation to model cases. It is independent from BPMN
Classification of Processes

structured process
- structured process flow
- activities can be defined in advance
- many repetitive elements
- no degrees of freedom for people with respect to process flow

case
- process flow can partly be structured
- activities can partly be defined in advance
- some repetitive elements
- degrees of freedom for people with respect to process flow

ad hoc process
- process flow cannot be structured
- activities can partly be defined in advance
- marginal portion of repetitive elements
- very high degrees of freedom for people with respect to process flow

Process logic
- can be modelled

Business logic
- cannot be modelled

partly translated from (Gadatsch 2005, S. 44)
Design Time vs Run Time = Modeling vs Planning

- A Case has two distinct phases: design-time and run-time
  - During the **design-time** phase, **business analysts** engage in modeling, which includes defining
    - Tasks that are always part of pre-defined segments in the Case model, and
    - “discretionary” Tasks that are available to the Case worker, to be applied in addition, to his/her discretion.
  - In the **run-time** phase, **Case workers** execute the plan, particularly by
    - performing Tasks as planned,
    - adding discretionary Tasks to the Case plan instance in run-time.

<table>
<thead>
<tr>
<th>Design-time phase</th>
<th>Run-time phase</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Modeling</strong></td>
<td><strong>Plan</strong></td>
</tr>
<tr>
<td><img src="image" alt="Diagram" /></td>
<td><img src="image" alt="Diagram" /></td>
</tr>
</tbody>
</table>

(CMMN 1.0, p. 5f)
A Case Model in CMMN

- A Case consists of a case Plan Model, a Case File Model, and a set of case Roles

(CMMN 1.0, p. 12)
The information model of a Case comprises of classes for the management of the information (data) aspects of a Case.

All information, or references to information, that is required as context for managing a Case, is defined by a Case File.

A Case File consists of Case File Items.

A Case File Item is depicted by a “Document” shape

A Case File Item may represent a piece of information of any nature, ranging from unstructured to structured, and from simple to complex.

A Case File Item can be anything from a folder or document, an entire folder hierarchy referring or containing other Case File Items or simply an XML document.

(CMMN 1.0, p. 15)
**Case Plan Models**

- The complete behavior model of a Case is captured in a case Plan Model.

- For a particular Case model, a case Plan model comprises:
  - all elements that represent the initial plan of the case, and
  - all elements that support the further evolution of the plan through run-time planning by case workers.

- There are four types of Plan Items:
  - **Tasks**
  - **Plan Fragments / Stages**
  - **Event Listeners**
  - **Milestones**

(CMMN 1.0, p. 17)
Example of a Case Plan Model

A case Plan Model is depicted using a “Folder” shape.

The name of the Case can be enclosed into the upper left rectangle.

The various elements of a case Plan Model are depicted within the boundary of the case Plan Model shape.

The diagram shows an example of a case Plan Model.
Tasks

- A Task is a unit of work. There are three types of tasks
  - **Human Task** - a Task that is performed by a Case worker, they can be
    - Blocking: Task is waiting until the work associated with the Task is completed
    - Non-Blocking: the Task is not waiting for the work to complete and completes immediately, upon instantiation.
  - **Process Task** - can be used in the Case to call a Business Process
  - **Case Tasks** - can be used to call another Case

- Task Items are depicted by rectangle shape with rounded corners (cf. tasks/activities in BPMN),

  - [Non-Blocking Human Task](#)
  - [Blocking Human Task](#)
  - [Process Task](#)
  - [CaseTask](#)

(CMMN 1.0, 48ff)
Discretionary Tasks

- Tasks are always part of pre-defined segments in the Case model.
- In addition to tasks there are Discretionary Tasks which are available to the Case worker, to be applied in addition, to his/her discretion.
- A discretionary Task is depicted by a rectangle shape with dashed lines and rounded corners.
- Any task type can be discretionary.

(CMMN 1.0, p48f)
Event Listeners

- An event is something that “happens” during the course of a Case. CMMN predefines many events, and their causes:
  - Anything that can happen to information in the CaseFile.
  - Anything that can happen to Stages, Tasks and Milestones.

- Event Listeners are used to model events that do not happen to plan items.

- An EventListener is depicted by a double line circle shape

- Event Listeners are specialized to
  - Timer Event Listener
  - User Event Listener

(CMMN 1.0, p. 18f, 52f)
Sentries - Entry and Exit Criterion

- Plan Items may have associated Sentries.
- Sentries define the criteria according to which the Plan Items are enabled (or entered) and terminated (or exited).
- A Sentry “watches out” for important situations to occur which influence the further proceedings of a Case.
  - a Sentry used as an entry criterion is depicted by a shallow “Diamond”
  - a Sentry used as an exit criterion it is depicted by a solid “Diamond”
- Discretionary tasks must not have Sentries.

(CMMN 1.0, p. 23f)
Sentry

■ A Sentry is a combination of an event and/or a condition.
  ♦ An On-Part specifies the event that serves as trigger.
  ♦ The If-Part specifies a condition, as expression that evaluates over the Case File.

■ When the event is received, the condition might be applied to evaluate whether the event has effect or not.
  ♦ If all On-Parts of a Sentry have occurred, and its If-Part (if existent) evaluates to “true”, the Sentry is “satisfied”.

■ A Sentry that is satisfied triggers the Plan Item that refers to it:
  ♦ When the Sentry is referenced by one of the Plan Item’s entry Criteria: a Task or Stage will be enabled, and a Milestone will be achieved.
  ♦ When the Sentry is referenced by one of the Plan Item’s exit Criteria: a Task or Stage will be terminated (exited).
Sentries and Rules

- Sentries may take one of the following forms:
  1. An event part and a condition part in the form
     
     on <event>
     if <condition>
  2. An event part in the form
     on <event>
  3. Just a condition part in the form
     if <condition>

- An Sentry and the task correspond to an ECA (Event-Condition-Action) rule.

(CMMN 1.0, p. 23)
Plan Fragment

- A Plan Fragment is a container of Plan Items and the Sentries.
- A PlanFragment is depicted by a rectangle shape with dashed lines and softly rounded corners and a marker in the form of a “+” sign in a small box at its bottom center.
- When the PlanFragment is expanded it has a marker in the form of a “-” sign in a small box at its bottom center.
- When a PlanFragment is expanded, elements contained in it become visible.

(CMMN 1.0, p. 61)
Plan Fragment

- A Plan Fragment is a container of Plan Items and the Sentries.

- Simple examples of Plan Fragments are:
  - A combination of two Tasks, whereby, the completion of one Task satisfies the Sentry that enables the start of the other (sequence flow).
  - A combination of an Event Listener and a Task, whereby the occurrence of the event satisfies the Sentry that enables the start of the Task.

(CMMN 1.0, p. 21)
Stage

- Stages are Plan Fragments that can be tracked.
- Stages maybe considered “episodes” of a Case. They can be regarded as sub-cases (cf. sub-processes in BPMN)
- A Stage is depicted by a rectangle shape with angled corners and a marker in the form of a “+” or "-" sign in a small box at its bottom center (“+” or "-" designate expanded or collapsed stages).

Collapsed stage with two entry and one exit criterion

Expanded versions of the Stage with one sub Stage and three Tasks
Milestones

- A Milestone is a Plan Item Definition that represents an achievable target, defined to enable evaluation of progress of the Case.

- No work is directly associated with a Milestone, but completion of set of tasks or the availability of key deliverables (information in the CaseFile) typically leads to achieving a Milestone.

- A Milestone is depicted by a rectangle shape with half-rounded ends.

- A Milestone may have zero or more entry criteria, which define, when a milestone is reached

(CMMN 1.0, p. 21, 52)
Plan Item Controls

- Plan Item Controls define aspects of control of instances of Tasks, Stages, Event Listeners and Milestones.

- Plan Item Controls may specify the following:

  ✦ Under which conditions will Tasks and Stages, once enabled, start manually or automatically

  ✦ Under which conditions will Tasks, Stages and Milestones be “required” to complete or terminate before their containing Stage can complete.

  ✦ Under which conditions will Tasks, Stages and Milestones need to be repeated.

(CMMN 1.0, p. 39)
Connectors

- Connectors can be used to visualize dependencies between Plan Items
  - One such depicted dependency is the On-Part of a Sentry
  - The other type of dependency is between a Human Task and Discretionary Items in its Planning Table (see later)

- The shape of the connector object is a dotted line.

(CMMN 1.0, p. 53ff)
**Connector Usage**

- Connectors that represent Sentry On-Parts can be used to visualize dependencies between Plan Items.
- The following pictures illustrates situations where Task C can be activated only
  1. if Task A is complete
  2. if Task A and Task B are complete
  3. if Task A or Task B are complete

(CMMN 1.0, p. 54)
**Connector Usage**

- Stage B depends on the exit criterion of Stage A
- Task A depends on the achievement of Milestone A.
- Task A depends on a TimerEventListener
- Task A depends on a CaseFileItem

(CMMN 1.0, p. 55)
Planning Table

- Users (Case workers) are said to “plan” (at run-time), when they select Discretionary Items.

- A Planning Table defines the scope of planning.

- Planning Tables can be assigned to a Stage or a HumanTask.
  - Stages: The Planning Table can be used to plan instances of Tasks and Stages into that Stage instance.
  - Human Tasks: The Planning Table can be used to plan instances of Tasks and Stages into the Stage that contains the Human Task.

- A Planning Table is depicted by a “Table” shape.

- A Planning Table can have several Table Items (i.e. Discretionary Items). Planning Table and Table Items can have applicability rules.

(CMMN 1.0, p. 56f)
**Planning Tables**

- The maker at the center bottom cell of the Planning Table indicates if the Discretionary Items are visualized (+) or not (-).
  - When a user “expands” a Planning Table, its contained Discretionary Items become visible within the Stage.
  - When the Planning Table of a Human Task is expanded, its contained Discretionary Items are visualized outside the Human Task shape.

(CMMN 1.0, p. 56f)
**Expansion of Planning Tables**

- These four figures illustrate expansion of Planning Tables

1. Collapsed Stage with Collapsed Planning Table
2. Expanded Stage with Collapsed Planning Table
3. Expanded Stage with Expanded Planning Table
4. Expanded Stage with Expanded Planning Table and Expanded HumanTask Planning Table

(CMMN 1.0, p. 57f)
Planning at Run Time: Applicability Rules

- With planning tables it is possible to make Discretionary Items dynamically applicable for planning.

- Applicability Rules are used to specify, whether a Table Item is “applicable” (“eligible”, “available”) for planning, based on conditions that are evaluated over information in the Case File.

- If the condition of the ApplicabilityRule evaluates to “true”, then the TableItem is applicable for planning.

- During planning only Discretionary Items, for which the ApplicabilityRule evaluates to “true”, must be shown to the Case Worker.

(CMMN 1.0, p. 29, 56)
## Decorators

- Case Plan Models, Stages, Tasks and Milestones can have decorators

<table>
<thead>
<tr>
<th>Decorator Applicability</th>
<th>Planning Table</th>
<th>Entry Criterion</th>
<th>Exit Criterion</th>
<th>AutoComplete</th>
<th>Manual Activation</th>
<th>Required</th>
<th>Repetition</th>
</tr>
</thead>
<tbody>
<tr>
<td>CasePlanModel</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>#</td>
</tr>
<tr>
<td>Stage</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
</tr>
<tr>
<td>Task</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark] #</td>
</tr>
<tr>
<td>Milestone</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark]</td>
<td>![Checkmark] #</td>
</tr>
</tbody>
</table>

(CMMN 1.0, p. 62)
Model Elements with all possible Decorators

(CMMN 1.0, p. 63f)
What is the meaning of this model?

(CMMN 1.0, p. 46)
What is the meaning of this model?

Application arrived

Check land-use plan

Check environmental sustainability

Check historical preservation

Publish application

Specify stipulations

Assessment finished

Inform applicant