

# Adaptive Case Management Book Project Proposal

Keith Swenson  
Nov 23, 2009

## ***Executive Overview***

**Proposal:** Write and produce a book to define the subject of “Adaptive Case Management” with a focus on needs and use cases. Involve 6 to 9 authors to produce this book on a very short schedule.

**Audience:** CEO and CTO. Target the CEO because Case Management is ultimately about the ability to support knowledge workers in an organization, and so it strategic for organization that need to improve knowledge worker efficiency. Target the CTO because it is an overview to help keep up on the latest strategic technology trends. Thus it need to avoid being deeply technical in details of the technology, but is intended to be a complete treatment of the advantages and benefits of this product area. Most notably, we do not target the CIO since this is for end users, not development teams.

**Timeframe:** three months for author contribution, another two months refinement after that. Publication target end of March 2010.

**Size:** Approximately 300 pages divided more or less evenly among authors.

## ***Overview***

This proposal is following on the heels of the completed Case Management Thought Leader Summit held in Maidenhead on Nov 3. The presentations were surprisingly consistent, and there was a strong feeling of needing to get the message out. Various attempts have been made to communicate these concepts in shorter works: white papers, presentations, and academic papers. None of the existing efforts target this audience of decision makers with a not-so-technical approach with the kind of completeness necessary. We recognize that this is a new market that needs to be differentiated from the BPM market. Bringing a number of noted experts together to write the book covering the topic from all sides is what is needed to help to form this nascent market.

Concurrent with the writing of a book is the plan to found a “Case Management Consortium” (CMC, working title) as well as a conference in the spring. The April timeframe for conference and book launch might be good timing.

Criteria for authors are that they are a noted expert on the subject and have some experience writing. A small number of vendor (estimated at 6) have offering in this area, so probably the noted expert that those vendors would want to participate. We anticipate two or three noted experts who are not with vendors may be interested in participating as

well. The reason for specifically including vendors, besides the fact that they have experience, is that the purpose of the CMC is to make the market safe for consumers with accurate description of what really exists

Author could bring in more collaborators if they desire to, but the number of main authors has to be kept to a manageable number. Each author will be responsible for 30 to 50 pages of writing, as well as reviewing all other contributions.

The goal is to have a collection articles on subject contributed by knowledgeable people. While each section will be the viewpoint of the author of that section, we will also try to get full coverage of all the subjects listed below. Each author will need to declare the subject that will be covered in their section. It is OK if there is some duplication, but primary goal is to avoid gaps.

Goal is to get a Foreword from a notable B School futurist.

## **Schedule**

- End Nov 2009: Subject Selection
- Dec 15 2009: Outline of chapter
- End Dec 2009: Draft Due
- Jan 15 2010: Feedback to authors, first round of reviews finished
- End Jan 2010: updated “final draft” due.
- End Feb 2010: formatted

## **Subject Outline**

| Title & Sections  | Explanation  |
|---|--|
| 1. Knowledge Workers & Supporting their work. <ul style="list-style-type: none"> <li>• The nature of knowledge work, including specific <b>use cases</b></li> <li>• Current technology misses important aspects of knowledge work.</li> <li>• Potential for much more.</li> </ul> | Summary of the current situation of knowledge workers. Like “Thinking for a Living” but updated with more recent information. Need to clearly lay out the need for technological support, and the promise. Might touch briefly upon how BPM does not do the job, but only briefly. In general we want to position this as a separate market with its own needs. This should identify who should be concerned about this, and what the possible value might be.                                     |
| 2. Emergent vs. Predefined – <ul style="list-style-type: none"> <li>• Exploring what this means.</li> <li>• Examples from <b>real life cases</b> that illustrate the need for emergence.</li> </ul>   | This is the primary requirement that makes most existing BPM approaches unsuitable. It is this requirement that, more than anything else, defines the marketplace for support of knowledge workers. This will clarify form the start that users are themselves doing all the work at runtime. There for the approach will emphasize ease of use over complex completeness. There is no value in extensive work ahead of time, if every instance will be different. Solid example to motivate this. |

|  |   |
|--|---|
| <p>3. Data Subject Orientation –</p> <ul style="list-style-type: none"> <li>• <b>Use cases</b> showing how the case data is first in consideration, not the process.</li> <li>• Information Aggregation.</li> <li>• Document Attachments</li> <li>• Examples of how the cases <i>become</i> the record.</li> <li>• Archiving &amp; disposal policies may require consideration.</li> </ul>   | <p>BPM technology differs because it focuses on the process itself. The focus on the data at the center of the case is a dramatically different approach. BPM work <i>ON</i> the data, while case management tends to <i>BE</i> the data. This is not a requirement, but it is an approach that is common across the market, and influences many of the capabilities of the technology.</p> |
| <p>4. Goal oriented.</p> <ul style="list-style-type: none"> <li>• <b>Use cases</b> showing how the desired outcome is specified first, and tasks elaborated backward from that.</li> <li>• Means and Ends and Context.</li> <li>• Examples of “pull based”.</li> <li>• <b>Real life examples</b> of how goal orientation is necessary.</li> </ul>  | <p>This chapter helps the CEO understand how the technology is used, and how knowledge workers get value. Now we start getting a picture of how knowledge workers work with it, and real example of use cases as they unfold will be helpful.</p>   |
| <p>5. Tasks are planned, not modeled.</p> <ul style="list-style-type: none"> <li>• <b>Use cases showing</b> how planning is part of the job of the user (like project planning).</li> <li>• Reusable sets of tasks are a “procedure” not “process diagram”.</li> <li>• Cases often created without any procedure included at the start.</li> <li>• No limit on the number of procedures used in one case.</li> <li>• Tasks represent a human activity, not a service call.</li> <li>• Ad-hoc routing. Quasi-sequential routing.</li> <li>• All about Follow-up &amp; communicating the current status</li> </ul> | <p>Continuation of the last chapter in helping CEO to understand how it is used, with some additional details on the task planning. Parallels are drawn here to project management. For the CTO, a clear distinction is made with the kinds of “process” which you find in BPM and SOA technology. Focus on the requirements that workers have.</p>   |
| <p>6. End user, non technical, oriented.</p> <ul style="list-style-type: none"> <li>• <b>Use cases</b> showing that the limitation of the administrator being the only one to control or configure would be</li> </ul>   | <p>Stories about how knowledge workers use this directly. It is at this point we make it clear that programming is unnecessary and unwanted. They may have assumed up to this point the same story that BPM always gives about business users using it, but really we mean</p>  |

|  |   |
|--|---|
| <p>unacceptable.</p> <ul style="list-style-type: none"> <li>• <b>Use cases</b> showing how it needs to be transparent to (usable by) management. No white lab coats.</li> <li>• <b>Examples</b> of how there can be no finality or determinism, and that state can in reality be changed by entirely external events. Requirement that the user remains in complete ability to change any aspect of a case.</li> </ul>   | <p>programmers. This has to make it clear that programmers are not needed or wanted. Also a good example story should be presented to cover the idea that everything is under the control of the knowledge worker because changes in the external real world can require a corresponding internal change to the case, which could never have been predicted or even sensed.</p>   |
| <p>7. Life cycle</p> <ul style="list-style-type: none"> <li>• <b>Use cases</b> showing that typical application development lifecycle would be a problem.</li> <li>• <b>Examples</b> of in-process optimization of the process itself.</li> </ul>  | <p>In the Thought Leader Summit we spoke a lot about lifecycle, and I felt there was probably enough detail to form another chapter on that. This is really further highlighting that a developer is not needed or even allowed for, and how things are created and made available in a very different mode.</p>  |
| <p>8. Guiding vs. Constraining –</p> <ul style="list-style-type: none"> <li>• <b>Examples</b> of rules which are soft to guide the user without restricting,</li> <li>• <b>Examples</b> of rules that might at other times be strictly enforced.</li> <li>• <b>Examples</b> of the kinds of rules that might be used.</li> </ul>   | <p>We have built to this point all about the flexibility, but this chapter introduces the concept that the knowledge worker is not all alone and not completely manual. They can make templates with rules. The rules can either be completely enforced (no way around) or they can be guiding and allowing the user to go around them after getting a warning. Specific examples of how this might be used, and how it would be helpful should be included.</p>                      |
| <p>9. Access control policies</p> <ul style="list-style-type: none"> <li>• <b>Use cases</b> of sensitive knowledge and how important to control access</li> <li>• <b>Examples</b> of how this needs to be specified by the user as the case proceeds. Part of the job.</li> <li>• <b>Use cases</b> showing difficulty if access is controlled by predefined application or by separate administrator.</li> <li>• <b>Examples</b> of how access control changes tracked in history, just like other changes.</li> </ul> | <p>Flexible working patterns do not mean that all the information is open to the public. Clearly real work must take place in a controlled environment, and ACM allows for strong access controls. But this approach is somewhat different from others since it is directly added/changed by the user. Other approaches rely on a programmer to write an application to enforce such. Since no application is developed, this has to be a feature that the user can use directly.</p> |
| <p>10. Templates, not programs.</p>  | <p>Another view on how this is not programming</p>  |

|  |  |
|--|--|
| <ul style="list-style-type: none"> <li>• <b>Examples</b> showing show case data, tasks, access control would be used by direct manipulation, and not programmed.</li> <li>• <b>Use cases</b> showing the tracking of potential tasks &amp; artifacts that are not there yet.</li> <li>• <b>Use cases</b> showing a procedure might be invoked in order to produce a necessary artifact.</li> <li>• Details?</li> </ul> | <p>for the CTO. Templates work in a very different way from programs (and most BPM is programming). Templates can be created by any user, and then exchanged with any other user in a completely decentralized way. Knowledge workers can leverage each other to discover or develop best practices.</p> |
| <p>11. Temporal by Nature:</p> <ul style="list-style-type: none"> <li>• Demonstrations that there is a requirement that everything be versioned.</li> <li>• Use cases of all changes being recorded in history.</li> <li>• Ability to roll back any change &amp; restore previous state. “undo” Not ACID. No guaranteed consistency with the world, but not needed.</li> </ul>   | <p>Document versioning won't be a surprise, but access control versioning, task versioning, and task assignment versioning will be. For the CTO, explanation should be included on how these are not “application like” transactions, but more “people-like” transactions.</p>                           |
| <p>12. Community</p> <ul style="list-style-type: none"> <li>• <b>Examples</b> of sharing of data structures, tasks, roles, procedures, templates on individual basis or publish / subscribe.</li> <li>• Possibly collected by team or organization, but not required.</li> <li>• Searching for, tagging, rating, and commenting on templates.</li> </ul>   | <p>Data sharing is a strong element of supporting knowledge workers. Also, Templates are shared. Templates themselves become a kind of knowledge which can be searched for as well.</p>  |
| <p>13. Assigning tasks to people.</p> <ul style="list-style-type: none"> <li>• <b>Use case</b> showing how assigning a task is part of job</li> <li>• Procedure templates may come without assignee pre-assigned.</li> <li>• <b>Examples</b> of how this might be integrated with organization / social network. for looking up people</li> </ul>  | <p>For the CTO, this is a more detailed view on task planning, especially how tasks are handed out, and how someone might find the right person to give a task to. Integration with organizational information should be discussed.</p>  |

|  |  |
|--|--|
| <ul style="list-style-type: none"> <li>• <b>Use case</b> for personal roles (as opposed to “official” or global).</li> </ul>   |  |
| <p>14. Integrated to knowledge management tools,</p> <ul style="list-style-type: none"> <li>• <b>Use cases</b> showing the need for integration to document mgmt systems, wikis, blogs , IM</li> <li>• Enterprise search.</li> <li>• Tagging</li> <li>• Cases <i>become</i> knowledge</li> </ul>   | <p>Clarifying that this is not another island of information on its own, but instead integrated into everything the knowledge worker does. Description about how cases are available to everyone as well. This also helps stress how integration with SOA is not critical because knowledge workers are accessing knowledge, not making web service calls.</p>   |
| <p>15. History</p> <ul style="list-style-type: none"> <li>• <b>Use cases</b> for all actions being tracked in history: data changes, task changes, task assignment, access control changes.</li> <li>• Conformance of behavior can be enforced through audit logs. Surprisingly, such conformance rules do not need to be specified in advance.</li> <li>• Explain how a standard audit log format would be used.</li> <li>• Examples of analytic reports across many cases.</li> <li>• Mining of procedures from history</li> </ul> | <p>There are two key things to present here. The first is the rather obvious point that all changes need to be tracked, and that you need a report of how many times things were changed, or how long different actions take. The second key thing is that this is how conformance is might be enforced. Traditional process approaches have tried to force a particular path, and allow only a conforming path, but that does not work with knowledge workers who need the flexibility. Even though flexibility is given, there is no reason to give up on proving that everything was in the end done correctly. Example of police investigations or court cases would be good.</p>              |
| <p>16. Non – Features</p> <ul style="list-style-type: none"> <li>• Simulation not really possible, and not necessary.</li> <li>• SOA? – what is the relationship? Web service calls do not matter?</li> <li>• BPMN? Might not be useful.</li> <li>• Event processing? ESB? Show how this is a different subject.</li> </ul>  | <p>Here is a place to address misconceptions about the technology, and to head off wrong assumptions. This chapter will help to drive clear differentiation between case management and BPM/SOA. Assumptions may be based on things that are “always” done (like ACID transactions) so readers can get a full appreciation of why ACM is separate from BPM. Maybe a clear statement on how BPMN is focused in an entirely different direction (give examples) that attempting to do too much with one language would only confuse things. For example, tasks are not “atomic” in ACM. Instead, they are started at one point, and then much later, in a separate action by the use, completed.</p> |
| <p>17. Glossary and Definitions</p>  |  |

