

How to Make Use of Rea Questions from CSCW and HCI when operationalizing the REA ontology

Yvonne Dittrich

Abstract

Few studies from CSCW or HCI address ERP systems or other software used to integrate and monitor production processes. However, as ERP systems are software to be used to support and monitor highly co-operative efforts of producing and selling goods, and as the management and control of these processes is also a co-operative activity, results from this area might give indications for improvements.

This position paper explores a set of questions regarding the embedment of ERP systems and especially a yet fictive one based on the REA ontology into the dynamics of everyday work in.

Introduction

This position paper is about operationalizing REA not only for analysis and knowledge management [7] but also as a meta-model for a fictive enterprise resource management system. ERP systems are highly adaptable and tailorable software systems. However, in the ERP literature, the subject of configuring and tailoring is normally discussed as a problem. The more the system has to be adapted to the specific organization the greater the risk for failure for the return of investment. [5, 2] On the other hand the implementation of ERP is understood and treated as streamlining the organization under development to fit with the practices supported by the specific ERP.

One of the reasons for this mismatch between the implemented flexibility and the needs of different organization might be the fact that ERP systems historically developed 'inside-out' integrating different modules that in themselves were developed as stand alone software, addressing the specific sub-domain instead of implementing a more abstract model. [9]

Building an ERP system on a simple but powerful meta-model for accountancy like the REA ontology might allow to support a variety of business practices with the same basic idea about accounting and management. If the model is powerful enough to accommodate different business areas and developments within the business as well as each company, it might be a good base to accommodate fitting degrees and dimensions of variability.

Designing what should be adaptable by whom, and what kinds of interfaces are suitable to support the tailoring is not a trivial task. It should be based on thorough knowledge of the application domains and the work practices that should be supported, and the design of the variability is tightly coupled with the underlying architecture decisions. [4]

Research on tailoring or end-user development has so far not addressed the adaptation of enterprise resource planning systems. (Though it is mentioned in [13] as a very promising application area.) The reason for this might have been that the de-facto

possibility to adapt ERP systems prior and after introduction was rather none-existent. [2] The first article I read that takes up this issue in a constructive way, that takes up the issue had been published only recently. [15]

This position paper does not aim to solve the related problems, but to discuss question that will become relevant when addressing the issue. It is clustered around a set of questions and relates to relevant literature from the respective discourses. The questions are partly orthogonal, so their order does not imply either importance or priority. The last section discusses research methodology to address the questions discussed in the article.

What are the dimensions of variability needed?

I did not find any article that evaluated the dimensions of variability needed and evaluated the development of ERP system implementations over time. One can expect a number of layers of variability needed.

Where existing ERP systems are flexible enough, or even provide unnecessary possibilities of adaptations and what deviations from the standard model that companies are willing to pay for, could when systematized give an idea for the overall needs of variability in relation to different branches and cultures.

Different installations and modification within the branches can give an idea what kind of variability is required to accommodate different business and work practices.

War stories about work-arounds during everyday use and in order to handle exceptions and the adaptations of their ERP implementations undertaken by companies might indicate the necessary variability to be able to accommodate development within one company over time.

To map out the different needs for flexibility, methods from domain analysis (see e.g. [3]) and specific case studies could be combined. The result would yield a base to discuss the following question and might be useful to evaluate the design of different ERP systems or a new system under construction [10].

Who might be doing what kind of change?

When introducing an ERP system normally a consultancy is hired to adapt a of the shelf product to the specific needs of the company. This in itself is a major development project. Not only the basic software has to be configured, also specific solutions – like interfaces to other systems in the company, specific functionality to accommodate just this business – have to be developed. The customer is often considered as just end users.

However, the use of this kind of systems often includes changes of aspects of the system. Using a ‘normal’ order form is one thing. Adapting it to fit an exceptional agreement requires an understanding of the implications of that change to the handling of the order in the system. Developing a new pattern contract will require a different set of skills and probably a different interface.

Research on end-user development and tailoring, e.g. of CAD-systems [6, 12], or even text editors [17], has shown that the user community differentiates into different groups that are more or less competent in higher level use and adaptation of the system. So-called ‘gurus’, or ‘super users’ are central figures for the adaptation of the software and the distribution of tailoring artifacts and knowledge.

Medium size or bigger companies might be able to effort own system administration competencies. These people then are responsible for the operation of besides others

the ERP system. Some companies might invest into education of their IT personnel to be able to configure and maintain the implementation.

Who of these actors is doing what kind of adaptations might be different from company to company. To open up for a diversity of smaller and larger adaptations by different kinds of user-designer and IT professionals requires designing different kind of tailoring/adaptation interfaces.

How to represent tailoring functionality?

Based on analysis and design of what is the normal use, what is the normal change and what is a reconfiguration of the system, the different user and tailoring interfaces can be designed.

The interfaces for end-user tailoring have to represent the possibilities for change from a user point of view. However, it does not make sense to hide the ontology the system is built on. This basic model can here serve to identify the aspects that should be possible to change.

The model then can be used to check the consistency of changes and adaptations by users. The duality of events in REA is one example for what can be used to check whether a supply chain does not contain a problem.

Often adaptations of the system can be classified in different levels of complexity. Users who start tailoring normally start with adaptations on a lower level. To promote a smooth learning process the tailoring interface can cluster different levels of the tailoring functionality so that a novice tailor can start to implement changes without getting confused too much by higher level tailoring features. The tailoring interface should provide a 'gentle slope of complexity' [18] to the tailors.

How should the model become active?

In CSCW a long discussion had taken place of how to support distributed co-operation. Special criticism has been on workflow management systems. [the printing] Systems that prescribe a certain order of doing things tend to become problematic. Even when a normal order of things exists, the specific case might require another order. Unnecessary enforced sequentiality will result in cumbersome work-arounds.

An agent that checks the fulfillment of contracts by matching events and contracts reduction grammars is a good example for a mechanism not depending on a certain order of events. [Fritz' student's text] Similarly the ontology can be used to check the formal correctness of contracts or events.

If a certain order of events is needed, e.g. for sending around a task between different work places, and because a logical order between different task exist in a normal case, a tailorable work-flow would give enough support for the normal case and still allow flexibility for exceptions. [11]

Distinguishing between tailoring of the common infrastructure and what is tailoring of the individual work environment

As a salesperson I am responsible for this special customer. They get special conditions and they often border a related set of products. To make life easier I assemble a specific pattern contract for this company where I already fill in the customer field, and the normal set of orders.

Such a pattern contract might make life easier for me, but it might not be useful as a company wide pattern contract. However, I might want to share it with my colleague, who is responsible for a similar customer. So I might want to annotate it so that she can understand what she might want to change to accommodate her case.

Contracts might be a case where one could think to have both, tailoring for individual preferences and tailoring to adjust the pattern contracts for the whole company. Both activities are similar, but require different kinds of support. The adjustment of the common infrastructure or artifacts requires documentation and negotiation support. Tailoring artifacts that enhance the individual environment require support for documentation and sharing. [18]

How to handle exceptions and negotiate changes?

One of the result of detailed studies of office work is that there is hardly any routine that is not a day to day achievement. A routine case often is a construction, a decision to regard a certain event as an instance of a general rule. [16] In a more complex organization the decisions of which cases to treat under which rules often is a question of maintaining and developing the business policy. Such ‘articulation work’ [8] on the other hand may even result in the change of the routine work.

Especially if the software does not prescribe one way of doing things, the organization might want to decide in a specific forum on changes especially when related to business policies. [17] To prepare, support and communicate such organization-plus-technique development projects, the provision of a possibility to communicate about the application and its usage had been discussed. Providing the possibility to communicate about the application and its configuration in a ‘second level’ [14] could be a means to support these processes.

How to support tailoring practices?

Whether people tailor their individual work environment or their common infrastructure, they would probably want to document what they have done for future changes and use by others. And they also are interested in whether their adaptations are doing what they are supposed to do. The possibility to document and to test their adaptations in a ‘save’ environment is one of the features users asked for when starting to use tailoring features. [13]

Beyond this kind of support a formal model like the REA ontology can be used to check the adaptation for formal correctness, thus preventing adaptations that cannot or should not be handled by the system.

Ways to explore the questions

To explore the above-mentioned questions the normal approach in CSCW and participatory design might be difficult. The already mentioned domain analysis methods and case studies can yield a map of variability needs, Studies of existing work practices of use and configuration of ERP systems can provide a starting point to understand the complexity involved in the implementation of such systems into an organization. But neither will be able to provide a final set of requirements: Either companies use a traditional ERP system and are used to work around the system if things have to change. Their picture of the situation will be to a high degree

influenced by today's ERP design. Other companies do not use an ERP system and so might not be able to imagine a highly tailorable one either. Prototypes based on the simplified implementation of the REA ontology might indicate possibilities for a more flexible design and indicate suitable tailoring dimensions that can be evaluated by domain experts in relation to their experience and discussed with pilot users. Through such evolutionary cycles new technical possibilities and new practices can be developed and evaluated hand in hand.

References

1. J. Bowers, G. Button and W. Sharrock Workflow From Within and Without: Technology and Cooperative Work on the Print Industry Shopfloor. In *Proceedings of the Fourth European Conference on Computer-Supported Cooperative Work. ECSCW '95*. Y.Sundblad, K. Schmidt and H. Marmolin. Kluwer Academic Publishers 1995, pp. 51-66.
2. L. Brehms, A. Heinzl, M. Lynne Markus Tailoring ERP Systems: A Spectrum of Choices and their Implications. *Proceedings of the 34th Hawaii International Conference on System Sciences 2001*.
3. K. Czarnecki and U. Eisenecker. *Generative programming, methods, tools and applications*. Addison Wesley, 2000.
4. Y. Dittrich, O. Lindeberg Designing for Changing Work and Business Practices. In: N. Patel *Adaptive Evolutionary Information Systems*. Idea group Inc. 2003.
5. F. Fui Hoon Nah J. Lee-Shang Lau Critical factors for successful implementation of enterprise systems. *Business Process Management Journal*, Vol. 7, No. 3, pp 285-296.
6. M. Gantt, B.A. Nardi Gardeners and gurus: Patterns of co-operation among CAD users. In: *Proceedings of the CHI '92*, ACM 1992, pp. 107-117.
7. G.L. Geerts, W. E. McCarthy The Ontology Foundation of REA Enterprise Information Systems. xxx
8. E. M. Gerson, S. L. Star Analyzing Due Process in the Workplace. *ACM Transactions on Office Information Systems* 1986, Vol. 4, No. 3, pp 257-270.
9. K. Kumar, J. Van Hillebersberg (guest editors) ERP Experiences and Evolution. *CACM* 2000, Vol. 43, No.4, special issue on ERP systems, pp 23 ff.
10. O. Lindeberg, W. Diestelkamp: How Much Adaptability do You need? Evaluating Meta-modeling Techniques for Adaptable Special Purpose Systems. *Proceedings of the Fifth IASTED International Conference on Software Engineering and Applications, SEA 2001*.
11. A. Morch, N. Mehandjiev Tailoring as Collaboration: The mediating Role of Multiple Representations and Application Units *JCSCW* Nov 2000
12. B. A. Nardi *A small Matter Of Programming. Perspectives on End-User Computing*. MIT Press 1993.
13. F. Paternó, M Klann, V. Wulf Research Agenda and Roadmap for EUD. Deliverables of the Network of Excellence on End-User Development December 2003. (<http://giove.cnuce.cnr.it/eud-net.htm>)
14. M. Robinson, Mike Design for Unanticipated Use. In: G. DeMichaelis, C. Simone and K. Schmidt (Eds.) *Proceedings of the Third European Conference on Computer Supported Cooperative Work*, Milan, 1993.

15. C. Soh, S. Siew Kien, J. Tay-Yap Cultural Fits and Misfits: Is ERP a Universal Solution? *CACM* 2000, Vol. 43, No.4, special issue on ERP systems, pp 27 ff.
16. L. A. Suchman Office Procedure as Practical Action: Models of Work and System Design. *ACM Transactions on Office Information Systems* 1983, Vol. 1, No. 4 pp 320-328.
17. R. Trigg, S. Bødker From Implementation to Design: Tailoring and the Emergence of Systematization. In *CSCW, Proceedings of the CSCW '94*, ACM-Press, New York, pp. 45-55.
18. V. Wulf, M. Rohde Towards an Integrated Organisation and Technology Development. In: *Proceedings of the Symposium on Designing Interactive Systems 23. – 25. 8. 1995*, ann Arbor Michigan, ACM Press, New York 1995, pp. 55 – 64.