Aligning People to Business Processes
Experience Report

Tomas Andersson, Ilia Bider, Rogier Svensson

IbisSoft AB, Box 19567, SE 10432 Stockholm, Sweden

{tomas|ilia|rogier}@ibissoft.se

Abstract. Business process orientation requires some kind of a business processes support system. However, building a proper system is not enough for introducing process orientation, the organization should ensure that the system is routinely used by all people involved in the processes. The latter task may be more complicated than the one of building the system. This paper summarizes an experience of building business process support systems and introducing them into operational practice acquired over some period of time. The paper describes and explains difficulties of marketing such systems and introducing them into the operational practice, as well as shows examples of how these difficulties could be overcome. In conclusion, it suggests an approach to introducing business process orientation, called a "reverse" evolutionary approach, that is suitable for some business contexts.

1. Introduction: basic assumptions and structure of the paper

This paper represents an experience report from a small consulting company that considers its mission as promoting business process orientation. The material is written having practitioners in mind. The number of references was intentionally kept to the minimum to make the paper easier to read for the intended audience. Despite the practical orientation of the paper, some parts of it might be also of interest for the researchers as a raw material to support/disprove certain theories, or serve as an impulse for creating new ones.

In the paper, we assume that a process-oriented organization of work is better than the traditional functional or project-oriented organization; for argumentation on this topic see classical publications, e.g., [Hammer & Champy, 1994]. Under process orientation, we understand not the way of "thinking process oriented", but the way of "working process oriented" [Bider, 2005]. The latter includes genuine cooperation between all process's participants independently of to which department they belong, and whether a particular process instance follows the standard pattern, or deviates from it. It also means motivated involvement of process participants who understand their own roles in the process and the roles of others (including the management). Business process orientation as a way of working also means that the experience gathered from previously completed processes is directly used in operational practice.

Business process-orientation (as a way of working) is difficult (if ever possible) to achieve without a specially constructed Business Process Support (BPS) system. Such system among other things should:

- Give to each process participant easy access to the state of the affairs in any particular process instance. This includes information on what has been achieved so far (process state), how it has been achieved (process history), and what is going to be done in the nearest future (process plan).
- Give to each process participant easy access to all process instance he/she is participating, including information on what he/she is supposed to do in the frame of each process instance and when.
- Provide participants with effective communication channels along the process instance lines.
- Provide easy accesses to the organization's experience, e.g., already finished processes, so that it can be analyzed, and participants can learn by example.

Naturally, a BPS system should also help processes participants to complete activities within the frame of each process instance. This function of a BPS system is the same as in traditional business applications, while the above listed properties are specific for BPS systems.

Due to its specific functionality, a BPS system is often regarded not just as a system that automates part of human operations, but also as a tool of organizational change. Introduction of such a system in operational practice should be considered as part of the task of creating the fit between business processes and support system. This part consists of adjusting the people to business processes by making them routinely use the support system in all every day operations. This task is not a trivial one, because, usually, most of the workers have no practical (and/or theoretical) experience of working in the process-oriented way.

This paper describes our experience in building and introducing BPS systems in operational practice, with all its setbacks and achievements. The experience is divided in 2 periods, the early period from 1989 to 1992, and the later one from 1998 up to now. We did not work much with BPS systems in between these two periods. The discussions are focused mostly on the introduction of BPS systems into operational practice; the issues related to the technical development can be found in other works that we refer to in the main text.

The paper is structured in the following way. In Section 2, we give a brief description of the state-oriented view on business processes that was and still is used in our BPS systems, and describe the main ideas behind the system architecture. Section 3 describes our early experience in building and marketing BPS systems. Section 4 presents the experience of the period from1998 to Spring 2004, while Section 5 presents our latest strategy that is in place at the moment of the writing. Section 6 contains concluding remarks that suggest a "reverse" evolutionally approach to building and introducing BPS systems. "Reverse" means that the goal of evolution is not to develop a system that satisfy the customers needs, but rather to "develop" people so that they could fully exploit the possibilities built in the BPS system architecture.

2. State oriented view on business processes

The BPS systems that we discuss in the paper were built based on the state-oriented view on business processes as described in [Khomyakov & Bider, 2000, Bider 2002]. This view focuses on changes produced by activities executed in the frame of a given process instance over its lifespan. The main concept of the state-oriented view is the process's *state.* The process's state is aimed to show how much has been done to achieve the operational *goal* of the process instance and how much is still to be done. The state does not show what activities have been executed to reach it; it only shows the results achieved so far.

A *state* of a business process is defined by a "construct" that reflects the relevant part of the "business world" at a given moment of time. The internal structure of the state construct depends on the business process type to which the current instance belongs. An example, of such structure for a business process related to a customer order is represented on Fig. 1. The state structure includes: (a) attributes (variables), like *To pay*, *Paid, Ordered,* etc., and (b) references to various human and non-human participants of the process, like customer, product, etc.

A *goal* of a business process can be defined as a set of conditions that have to be fulfilled before a process instance can be considered as finished (end of the process instance trajectory in the space state). A state that satisfies these conditions is called a *final state* of the process. The set of final states for the process in Fig. 1 can be defined as follows: (a) for each ordered item *Ordered = Delivered*; (b) *To pay = Total + Freight + Tax*; (c) *Invoiced = To pay*; (d) *Paid = Invoiced.* These conditions define a "surface" in the state space of this process type.

The process is driven forward through *activities* executed either automatically or with a human assistance. Activities can be planned first and executed later. A *planned activity* records such information as type of action (goods shipment, compiling a program, sending a letter), planned date and time, deadline, name of a person responsible for an action, etc.

The process's state is used as a primary tool in deciding on what should be done to reach the process's goal from the current state. However, in some cases, a *history* of the process's evolution in time is important when deciding on actions. The history is defined as a time-ordered sequence of all previous states.

All activities planned and executed in the frame of the process should be aimed diminishes the *distance* between the current state of the process instance and the *nearest* final state. The meaning of the term distance depends on the business process in question. Here, we use this term informally. For example, activities to plan for the process in Fig.1 can be defined in the following manner:

- If for some item *Ordered > Delivered*, *shipment* should be performed, or
- If *To pay > Invoiced*, an *invoice* should be sent, etc.

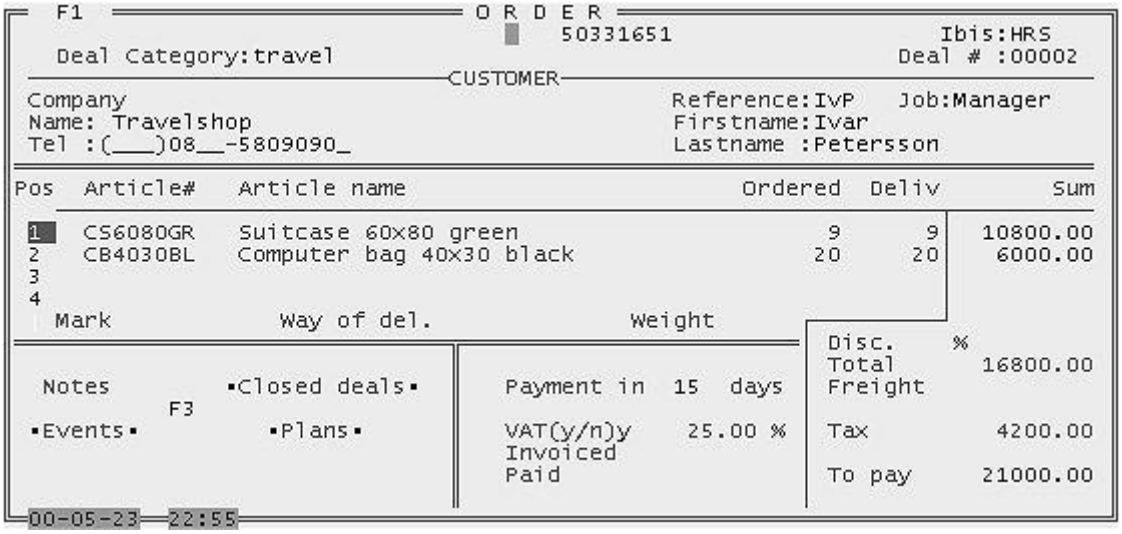

Figure1. State representation of order processing.

All activities currently planned for a process instance make up the process *plan*. The plan together with the "passive" state (attributes and references) constitute a so-called generalized state of the process, the plan being an "active" part of it (engine). The process plan on Fig. 2 corresponds to the process instance state shown on Fig. 1.

When an activity is executed, a process changes its generalized state. Changes may concern the passive and/or active parts of the state. At the minimum, the executed activity disappears from the plan. In addition, changes are introduced in attributes and references and/or new activities are planned to drive the process forward.

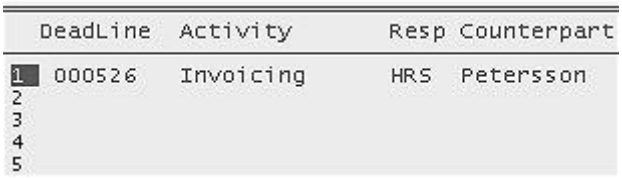

Figure 2. Process's plan that complements the state from Fig. 1.

When an activity is executed in the frame of a process instance, an event is registered. A *registered event* is a record that links the change in the state of a process to the reality outside the process. For example, it can record the date-time when the event happened and/or was registered, name the responsible for the event, register comments on the event at the moment of registration (or even later), etc. A list of all events that happened within the frame of a given process constitutes the *chronicle* of the process, i.e. its written history.

The heart of a BPS system based on the state-oriented view consists of:

- Historical database that automatically stores information on all events and all past states of all processes, documents, and other business objects.
- Principle of dynamic and distributed planning. Dynamic means planning when needed, distributed means planning to each other. Planning for each other constitute a communication channel between process participants along business process instances. Planning can be partly manual, partly automatic.

- Navigational system that allows the end user to freely navigate through the space of interconnected processes in the present and past.

When using a state-oriented BPS system, executing of an activity in the frame of a process instance conceptually consists of (at maximum) three steps: (1) introduce changes in the process state, (2) plan new activities, (3) register an event of current execution.

3. First period

Our first experience of dealing with BPS systems goes back to 1989-1990, when we built a system to support sales and marketing activities of a trading company [Bider, 1997]. The system was called *DealDriver* to highlight that it helped the workers to "drive" their deals from the beginning (e.g., getting an order) to the end (e.g., receiving payment). The order processing process had automatic planning programmed in it; Fig.1,2 represent screen snapshots from this part of *DealDriver*. Sales and marketing activities were planned manually. The system was developed for the character-based environment and run under DOS over a PC LAN.

This system has been used internally at *IbisSoft* for 13 years. The main advantage of the system for us was that it gave full control over all communication with the external world in the past and future (plans). First, the system was used only by the people who developed it, but later, new sales persons were trained to use the system. We encountered no serious problems in teaching new people how to work with the system.

The efforts of marketing our BPS ideas at that period were not successful. In our view, the main reasons for that were as follows:

- Our inexperience in sales and marketing in general.
- Low readiness of the market: the ideas of business process orientation were unknown to the majority of companies, and, definitely, BP was not a buzzword.
- Wrong focus of marketing and sales. The system was marketed as a system not as new way of working. As a result, marketing efforts were wasted on IT personnel who could not see what was special with the system in comparison to all other systems in the given application domain.

Still, our efforts lead to getting commissions for building a couple of prototypes based on our ideas, but the recession of early nineties forced us to abandon the BPS business activity for some time.

In parallel, our colleagues in Moscow built another BPS system based on the *DealDriver* ideas and its source code. The system, called *SoftMotors*, fully supported business processes at a car service station (service and repair). The system had all features incorporated in *DealDriver* but one, it had no planning component (neither automatic, nor manual). Communication between process participants was along changes in the states of process instances. When urgent attention was required, non-computer means were used, like banging on the wall that separated the neighboring department.

The system was successfully installed and used at about 20 customer sites. Marketing and sells were relatively easy because of lack of competition at that time in this part of the world. When competing products did arrive on the market, our colleagues had got the same situation as we had from the very beginning.

Training the personnel for using *SoftMotors* was not an exceptionally big problem. Most of the workers did not have any previous experience with computer systems, and accepted the system "look and feel" as something normal. They could not compare it with other systems, and thus discover its unusual nature. It did not mean that all users fully understood how the system worked. At least in one case, the workers tried to cheat the system by arbitrarily changing the process state. The fraud was easily discovered due to the historical nature of the underlying database.

4. Second Period

4.1 Marketing strategy

The second period of our experience with BPS systems started in 1998. The following changes happened by that time. First, the ideas of business process orientation became relatively popular, almost on the level of "buzzword". Second, we understood that we should not market our ideas as a system, but rather as a new, more effective way of working. At that time, our working example (*DealDriver*) was still character-based, and it could not be used even for demonstration. The decision was made to start marketing business process analysis instead of developing a new system in hope that some customers would wish to develop a BPS system afterwards.

The strategy we chose proved to be relatively successful. In the end, we got a number of business process analysis projects. During the first projects, we developed a practical methodology of business process analysis based on the state oriented view on business processes, see [Andersson, et al., 2002, Bider 2002] for details. A typical analysis was done in the form of meetings with a group of domain expert that included people actually working in the business process under analysis. None of the projects was of the "management-only" kind. During each of the analysis projects, a full understanding was reached with the group of what their processes were and how they could be driven in a more structured and effective way provided a proper support system had been obtained. This gave us a hope that introduction of a support system would not constitute a major problem.

Our marketing and sales efforts were not directed to any special segment of the market, we tried to go to the Industry, Non-profit organizations, and Public sector. However, the latter two proved to be more accessible, at least for a small consulting company as ours. Therefore, the analysis projects completed so far was conducted in a somewhat special environment, namely, Swedish public sector (e.g., Swedish municipalities), and non-profit organizations (e.g., association for tenants). Both these sectors have their peculiarities that concern the nature of their business processes as well as their internal environment.

Most of business processes in the above sectors are of administrative nature, such as preparation of decisions, inquiries/investigations, processing of complaints, lobbying,

etc. We call such processes loosely structured to stress that for these processes it is difficult to pre-determine the order of activities. This term has connotation with the concept of ill-defined problems in AI [Simon, 1973], fuzzy tasks in GSS, [Zigurd & Buckland, 1998], and ad-hoc, emergent and dynamic workflows, see for example [Jørgensen et al., 1999, Bernstein, 2000].

The internal environment in the "chosen" sectors is in a way more "democratic" than in industry. The management is reluctant to dictate its will or/and do not know how to do it in a proper way, and it is more wiling to give the initiative to the employees. The latter has both positive and negative sides, e.g. requires more internal marketing.

Our marketing strategy worked well also in the area of getting orders for building BPS systems. After completing a number of analysis projects, we started to get commissions on delivering support systems.

4.2 Systems introduced and under introduction

The *first system* we built and introduced was a support system for recruiting of new members to the Association of Tenants, Region West Sweden. The system, called *ReKo,* was built based on the analysis of the recruiting process overviewed in [Andersson, et al., 2002]. The system had a reduced functionality because most of the participants of the process were working outside their office (going from one apartment to another); the planning capability was not built in the first version of *ReKo*.

It took about half a year to build the system. It took one year before the organization started to use it in their operational practice and fully substituted the previously used technique based on Microsoft Excel sheets. The main reason for a one-year delay was (natural) resistance to organizational changes. To overcome the resistance, a competition was used at one point to show the advantages of the new system to the administrative personal. The goal of the competition was to test who can complete certain tasks quicker: an experience office worker with the old technique or non-experienced person (our own member of staff) with the new technique. Winning the competition was one of the many steps to overcome the natural resistance.

After one more year, the recruiting staff became fully acquainted with the system and understood advantages of working in a more structured way. It is worthwhile to mention that new members of recruiting staff, employed after *ReKo* had already been introduced in operational practice, had no problems in adjusting themselves to the established way of working.

By the end of the second year (i.e. one year after introduction), the statistics showed growing numbers of new members, which was attributed to better order in recruiting activity imposed by the system. Now, we are considering the next step of the system development, which is going from paper lists to palm computers. In parallel, we negotiate with other regions of Association of Tenants on introduction of *ReKo* in their operational practices.

The *second system* we built was a support system for processing applications for child adoption. The system, called *Emma*, was built on commission from one of the Swedish municipalities. It supports inquiry/investigation of families applying for child adoption. Initially, the system supported full featured manual planning and detailed registration of

all events happened in the course of an inquiry. The first stages of introduction, however, showed that the workers were reluctant to use exact planning and detailed registering. To go further with the introduction, a compromise adjustment was done to the system. Manual planning was substituted by automatic planning without deadlines. Registration of events was simplified to just making comments.

Emma is in full use, but the volume of adoption applications is quite small and we do not have enough data so far to analyze the results of its introduction. Our original plan was to extend *Emma* to other activities in the same department of the municipality. Unfortunately, the manager that commissioned us the development of *Emma* had left her office in the middle of *Emma*'s introduction. The new manager so far has not showed any enthusiasm to extend the system beyond its current functionality.

The *third system* we built on commission was a support system for a number of administrative processes at the main office of Association of Tenants, Region West Sweden. The processes included: service to their field (grass root) organizations, processing of feedback (e.g., complaints), lobbying (influence of decision of others), and some others. The system was planned as a full-featured system with manual planning, detailed event registration and structured state definition for each process.

The system, called *ProBis*, was delivered during 2003 in the module-by-module fashion. Modularization was envisaged to make the introduction easier. All modules were built on the same principles and they shared common user-interface. We believed that by the time the next module was introduced, the workers would already know how to deal with the system by learning the previous module. Due to organizational difficulties, our plan had failed, and the operational test of the system started only after practically all modules had been delivered and installed.

Having previous experience of introducing a support system in this organization, we expected some delays in the introduction process. However, the difficulties of introducing a system aimed at functioning through the whole organization showed to be much greater than in case of introduction of a system in a single department.

4.3 Problems encountered

We hoped that it would be easy to introduce a BPS system built on the results of an analysis project. This hope has never materialized. Each introduction took a lot of time and efforts, to which we were not prepared.

Some of the problems were purely organizational; they are not unique to BPS systems, and they are often encountered during introduction of new information systems. To this class of problems, for example, belong:

- Bad planning of the introduction, e.g. absence of a formal responsible for introduction of each module
- Unwillingness from the management to press their people to use the system
- Assigning people that were not part of the analysis project to test the system
- Bad training programs

Another not-unique problem was users expectations. Part of the personnel had very little experience of using computers in general, and business systems in particular. This situation, in itself, was predictable. What was not predictable was lack of willingness to learn. Current users expect a system to be so intuitively clear that even an inexperienced user can "mouse click" him/herself to understanding the system. We consider this as a new trend, as we have not experienced such situations in our practice before. A scientific evolution of this trend see in [Heijden, 2004]. Our original user-interface did not meet the users' expectation, especially for a situation with a greater number of occasional users, i.e. the users who did not use the system each day.

In addition, we discovered a number of conceptual problems that are specific to BPS systems, and our kind of BPS systems in particular. Understanding business process orientation in theory does not automatically imply understanding of how it will look like in practice. The following problems are examples in this category:

- Use of planning in general. People are usually not accustomed to plan their operative work in details, unless absolutely necessary, such as in cases where other people are involved.
- Use of planning as a way of communication. People are not accustomed to plan tasks for their colleagues or managers.
- Level of details on the computer screens. Though the level of details was always agreed during analysis, the members of the analysis team did not fully understand that this would be the level of details to work with after the system had been introduced.

4.4 Solutions

After analysis of the problems encountered, a number of measures were undertaken to speed up the introduction of *ProBis* in HGF's operational practice. These measures concern both adjusting the system, e.g. changing the user-interface, and improving the organizational structure of the introduction project at the customer site.

Up to now the following measures has already been completed:

- **User-interface** has been completely rebuilt. Particular care has been taken to better visualize both most attractive and most difficult features of the system. To the attractive to the users features belongs, for example, easy access to the information associated with each object. To visualize this feature, an icon bar was placed to the left of each object, as shown on Fig. 3 that represents an organization. Through the icons on the bar, the user can access all changes made to the organization, all processes connected to it (now and in the past), and all activities directed at this organization, planned as well as completed. In the initial version, this information was available through a pop-up menu the existence of which was not known for the occasional users.

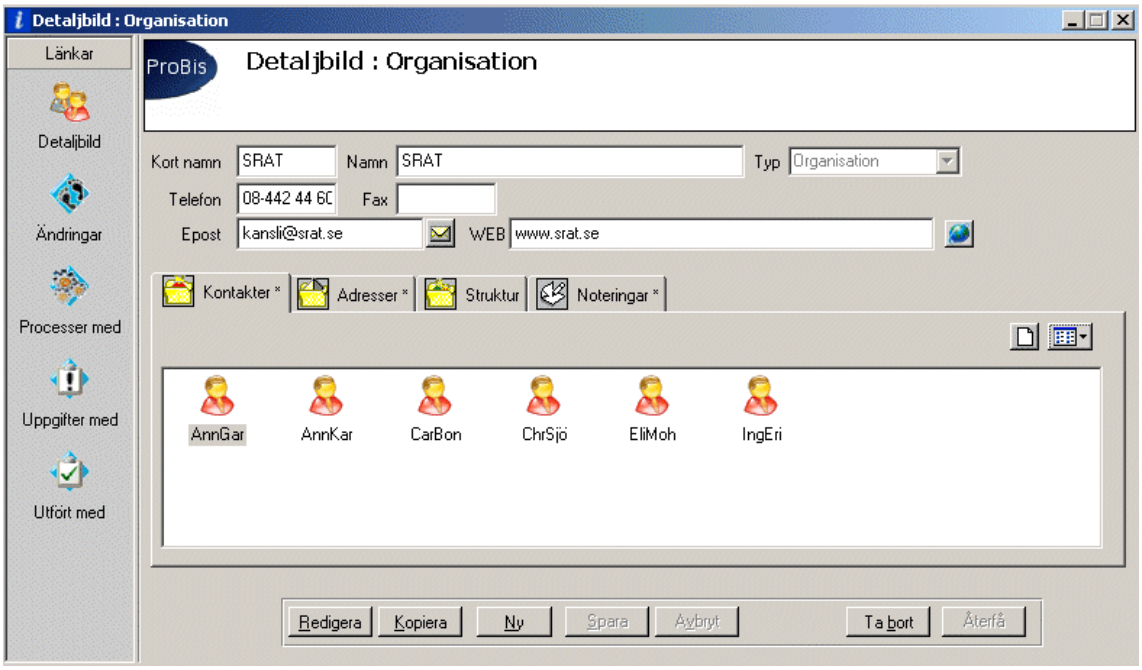

Figure 3. Representing Organization.

To the conceptually difficult features, belongs, for example, execution of a planned activity in the frame of a process instance. According to our model, such execution includes three steps:

1. introducing changes in the process state
2. planning new activities instead of the executed one
3. registering an event of current execution

We found that the end-users had difficulties to understand what actually happens with the planned activity under execution. To show this explicitly, a following visualization has been used, see Figure 4. Both, the activities planned for a process instance, and the events already registered in the frame of this process are represented as two list boxes adjacent to each other. The left box represents planned activities, while the right one represents completed ones. To start the execution of a planned activity, the end user presses the “>” button placed between the boxes, alternatively drags this activity from the left box to the right one. The planned activity disappears from the “plan” box and appears in the event box with a specially designated icon (lightening). After all changes in the process state are completed, the user presses the “save” button (“Spara”). At this moment the “lightening” icon is substituted by the “green tick” icon for events/completed activities. This visualization gives the end-user clear understanding as to what happens with the planned activity.

- **Planning and registering requirements** have been eased so that the system does not force the end-users to introduce new activities in the process plan, or add comments to the history records. He/she can just move planned activity to the right box and press “save”. These requirements can be introduced on the later stage.
- **The complexity of the screens** that represent the states of various business processes has been significantly reduced. This was done by temporarily hiding part of the

information fields. These fields will be opened later when the end-users are more ready for extended complexity.

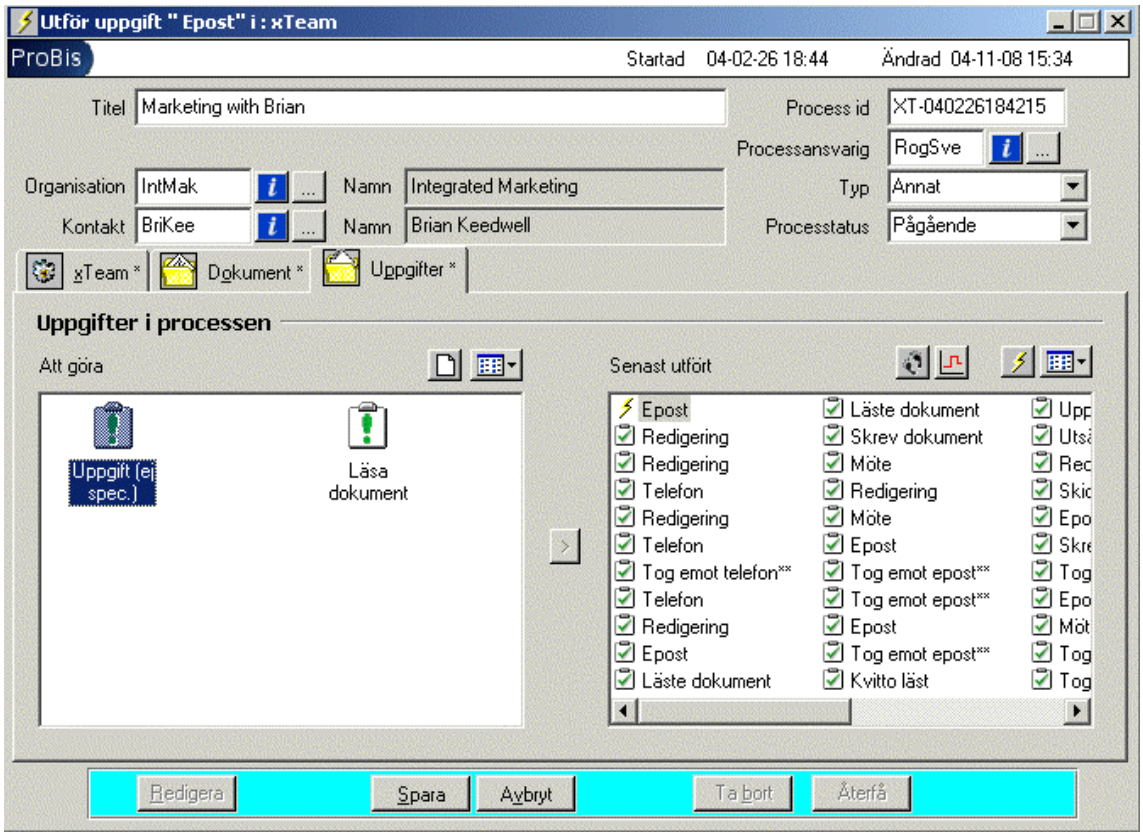

Figure 4. Execution of activities

- The refaced system was reinstalled at the customer site, and tested by the end-users, most of whom greatly appreciated the new user-interface. Now, there is a mutual agreement that there is not much more we can do as far as the system is concerned. The next task is to start using the system in full in all business processes it is meant to support.

5. New Strategy

Our development approach as described in the previous sections more or less followed the “classical” development paradigm. First, conduct a detailed business analysis/build a model, and then develop a system. Our system development always followed the evolutionary model, i.e. building a prototype and successfully enhancing it based on the customers feedback. However, our latest experience showed that this approach needed some modification. While part of our latest development could be considered as evolution, e.g., building a better user interface, the other part looks more like “devolution”. To this part, for example, belongs reducing complexity of the screens or easing requirements on registering/planning.

The underlying reason for changing the development approach lies in the nature of BPS systems. In the most business environments of today, we have a kind of “catch 22”. On one hand, the workers are not accustomed to work in a process-oriented way,

because this way cannot be introduced without a BPS system. On the hand, the workers have difficulties to learn how to work with such a system because they are not accustomed to the way of working it supports. The main challenge here, for the type of customers we currently deal with, is to go over from quite unstructured way of working to a relatively structured one.

The catch is difficult to resolve through the standard training programs, even good ones. The problem is not in teaching how to use the system technically, but how to use it in a particular business context. Therefore, it is difficult (if possible) to have "teaching" in the class, the workers need to use the system directly in their own operational environment. In order to resolve the "catch 22", the following strategy has been adopted for the future projects:

- Create a basic version of *ProBis* that supports all essential process-oriented features but that is not specialized for any particular business processes.
- Approach new customers with a "new" system development method. Instead of doing full-featured business analysis, requirement specification, etc., suggest to put the basic system directly into operational practice. This, naturally, requires some minimum of analysis to find an appropriate (for the test) part of the business. It might also require making some adjustments to the system, which could be limited to changing coded values of attributes. The test should be made with a group of volunteers for a limited period, of about 6 months, under supervision from one of our consultants.
- After completion the test period, the customer will have three choices, (1) discontinue using the system, (2) continue using it as it is, (3) develop modules that are specialized for particular processes. In any case, the decision will be made on the customer's own experience of actual using the process oriented way of work, with all its hardships and advantages.
- In case, the customer chooses to adjust the system to specialized processes, a full analysis will be done before enhancing the system. The analysis will be facilitated by the experience gained by the customer, and processes histories automatically saved by the system.

So far, the following has been accomplished for implementing the new strategy:

- The basic version of *ProBis* aimed at supporting administrative/management type of business activities has been created. It includes three modules that correspond to a three generalized types of processes called *iTeam, xTeam,* and *iProject.* All modules support planning and execution of activities and document processing. *iTeam* represent a minimally structured generalized process suitable for internal distribution of information, preparing documents, meetings, etc. *xTeam* differs from *iTeam* in that it has one designated external partner in form of an organization and/or person. It is suitable for supporting selling/buying, CRM, service, etc. *iProject* differs from *iTeam* in that it has a fixed number of participants who may be internal (the system users) or external. This module is aimed at supporting administration of projects.
- We have found the first customer to test the new strategy. It is a medium size Swedish union for academic employees called SRAT. The basic system was installed two month ago, and five people volunteered to use it. They got basic training and

decided to use the system when providing service to the union members. The *xTeam* module was chosen for this end. Though it became apparent quite soon that the generalized *xTeam* process did not fully match the service process at SRAT, the customer agreed to postpone any adjustments until the end of the test period.

After getting the introduction, the volunteers started to register their service processes in the system. They have regular meeting with our assigned consultant to follow up the progress, which was slow in the beginning. This time the slowness has not been attributed to the system, but to the difficulties of getting used to a more structured way of work. Currently, the frequency of the system use is growing and we hope that SRAT will get enough experience to take a qualified decision on in what way to proceed by the end of the 6th months term.

6. Conclusion

As follows from this experience report, we encountered and overcame a number of obstacles in our "missionary" work of promoting business process orientation. We needed conceptual understanding of what business processes were, and we created a state-oriented view on business-processes [Khomyakov & Bider, 2000]. We found it difficult to market and sell systems, and we changed the focus of marketing and sell activities from system to business. We needed a practical approach to business process analysis, and we invented a "State-Flow" technique [Andersson et.al, 2002].

Our greatest challenge for the moment is making people use BPS systems. This problem is quite essential in the environment of public/non-profit organizations sector in which we currently work. Though we do not know the extent of this problem in the industry, we suspect that it does exist in this sector as well.

To solve the usage problem, we have modified the evolutionary system development method as described in Section 5. This modification can be called a "reverse" evolutionally approach. Under "reverse", we mean that before setting an evolutionary process for the system, we need to complete the evolutionary process for its users. The goal of this evolution is to achieve better understanding of what business process oriented way of working is, what are its benefits, and what will be required from the users in order to obtain these benefits.

It is too early to say whether the reversed evolutionary approach can help in solving our current problem or not. Nevertheless, we believe that it is worth trying, because it is built upon a "win-win" business approach. In none of the three outcomes of the reverse evolution listed in section 5, there is any lost for a customer or a consultancy:

1. If the customer decides to discontinue using the system, it is done based on their own experience. The consultancy gets its money for supervising the experiment and it avoids getting an unsatisfied customer, the one that, because of some internal reasons, will not be able to use the system properly.
2. If the customer decides to continue using the system as it is, the consultancy gets a new satisfied customer who is willing to pay for the system licenses and technical support.

3. If the customer decides to develop specialized system, the chances of success of the "forward" evolution will be much greater then in the traditional evolutionary approach. The customer will have enough understanding of the business process oriented principles to be able to formulate what they actually need. In addition, system logs could be used as an extra material for detail analysis of the customer's business processes. Here, one could use process mining techniques of the sort being developed in [Aalst & Weiters, 2004].

As with any other system development method, the reverse evolutionary approach has its own limitations. It presumes that it is possible to find a "place" in the given customer's business where one of the basic generalized processes fits well. The approach won't work if there is no fit at all. The end-users will get an extra job of recording in the system without getting any immediate benefits from their actions, which can negatively affect their judgment of the system, and the way of working it supports.

From the IS research perspective, our experience represent a typical example of the action research theory in "action", see overview in [Baskerville & Myers, 2004]. According to the classical definition [Lewin, 1947], action research includes iteration of six phases (1) analysis, (2) fact finding, 3) conceptualization, (4) planning, (5) implementation of action, and (6) evaluation. Our experience shows that phase (5), implementation of action, is quite difficult, at least in the given social/business context.

On the whole, our experience confirms currently well-known fact that introducing a new technology that requires changing the way of working is quite a difficult task; we do not claim the discovery of this fact. However, our experience concerns a particular kind of business context, and a particular kind of technology, which lead to some specific problems and solutions. We hope that both might have been of some interest for the reader.

Acknowledgements: Writing of this report was supported by the Swedish Agency for Innovation Systems (Vinnova) under the grant for a project on "Integration Business Process Support with Knowledge Management". We would also like to thank all people who participated in the projects described in this paper, our colleagues, as well as experts in the field that worked for our customers.

References

Aalst, W.M.P van der, and Weiters A.J.M.M.. Process mining: a research agenda, *Computers in Industry* 53, pp 231-244, 2004.

Andersson, T., Andersson-Ceder, A., and Bider, I., 2002. State Flow as a Way of Analyzing Business Processes - Case Studies, *Logistics Information Management*, Emerald, Vol 15, No 1, pp. 34-45. Available at: http://www.ibissoft.se/English/Cases.pdf

Baskerville, R. and Myers, M.D., Special Issue on Action Research in Information Systems: Making IS Research Relevant to Practice—Foreword, *MIS Quarterly*, 28(3), pp. 329-335, 2004

Bernstein A. How can Cooperative Work Tools Support Dynamic Group Processes? Bridging the Specificity Frontier. *CSCW 2000*, Philadelphia, USA, ACM, 2000.

Bider, I. Developing Tool Support for Process Oriented Management. Data Base Management. 26-01-30, Auerbach, 1997.

Bider, I. *State-oriented business process modeling: principles, theory and practice*. PhD thesis, KTH (Royal Institute of Technology), Stockholm, 2002.

Bider I. Business Process Orientation: a Way of Thinking or a Way of Working? Industrial Editorial. *Business Process Management Journal* 11(1), 2005.

Hammer, M., and Champy, J. *Reengineering the Corporation – A Manifesto for Business Revolution,* Nicholas Brealey Publishing, London, 1994.

Heijden,H. van der. User Acceptance of Hedonic Information Systems. *MIS Quaterly*, Vol 28(4), pp.695-704, 2004.

Jørgensen H.D., and Carlsen S. Emergent Workflow, Integrated Planning and Performance of Process Instances. *Workflow Management '99*, Münster, Germany, 1999.

Khomyakov, M., and Bider, I., Achieving Workflow Flexibility through Taming the Chaos. *OOIS - 6th international conference on object oriented information systems*, Springer, 2000, pp. 85-92. Reprinted in the Journal of Conceptual Modeling: http://www.inconcept.com/JCM/August2001/bider.html.

Lewin, K. Frontiers in Group Dynamics II. *Human Relations* (1:2), pp. 143-153, 1947.

Simon, H. A. The structure of ill-structured problems. *Artificial Intelligence* 4, 1973, pp.181-201.

Ilze Zigurs, Bonnie K. Buckland: A Theory of Task/Technology Fit and Group Support Systems Effectiveness. MIS Quarterly 22(3): (1998)