Intention-mining: A solution to process participant support in process aware information systems

Elena V. Epure
Charlotte Hug
Rébecca Deneckère
Sjaak Brinkkemper

Technical Report 2013-020
December, 2013

Department of Information and Computing Sciences
Utrecht University, Utrecht, The Netherlands
www.cs.uu.nl

ISSN: 0924-3275

Department of Information and Computing Sciences
Utrecht University
P.O. Box 80.089
3508 TB Utrecht
The Netherlands

INTENTION-MINING: A SOLUTION TO PROCESS PARTICIPANT SUPPORT IN PROCESS AWARE INFORMATION SYSTEMS

Elena V. Epure
Department of Information and Computing Sciences, Utrecht University
E.V.Epure@students.uu.nl

Charlotte Hug
Centre de Recherche en Informatique, Université Paris 1 Panthéon-Sorbonne
Charlotte.Hug@univ-paris1.fr

Rébecca Deneckère
Centre de Recherche en Informatique, Université Paris 1 Panthéon-Sorbonne
Rebecca.Deneckere@univ-paris1.fr

Sjaak Brinkkemper
Department of Information and Computing Sciences, Utrecht University
S.Brinkkemper@uu.nl

Abstract. *While the advantages of flexible business processes have been highly recognized by the academia and organizations, the research focus has recently shifted to its trade-offs and how the negative consequences could be minimized. This research addressed the problem consisting of the difficulties encountered by process participants when interacting with flexible process aware information systems. In order to overcome these difficulties, several approaches for guiding or supporting the process participants during enactment, based on process mining, have been proposed. However, these solutions lacked the suitable semantics for humans reasoning and decision making during enactment as they provided recommendations at a low granularity level. Consequently, the objective of this research was twofold. First, the implications for agents, process participant and process administrator, of integrating flexible processes into process aware information systems were analyzed through a systematic literature study. Secondly, using design science, two artifacts were created to solve the problematic situation: 1) an innovative process mining technique that discovers the intentional model of the executable process in an unsupervised manner, and 2) a recommendation tool that formulates recommendations as intentions and confidence factors, based on partial traces and probabilistic calculus. The artifacts were evaluated in a case study with a Childcare application supporting flexible process enactment with a data-driven approach. The experiments revealed that the intention mining technique had a precision of 0.69 in discovering the correct intentions. Regarding the recommendation tool, the majority of the participants agreed on the improved support for decision making, offered by the recommendations given as intentions in comparison to recommendations given as activities, while a majority disagreed on the utility of the confidence factors attached to each recommendation.*

Keywords: intention mining, process mining, flexible processes, process aware information systems, process recommendations

1 INTRODUCTION

It was a typical morning during my browse of the web for news, when suddenly a line on the screen caught my eyes: "*Information Eats the Enterprise*". I was intrigued as I was trying not only to realize what it meant but also if it was a positive conclusion about the link between information and enterprise, or, by contrary, it revealed a shortcoming. I knew the answer to my questions was just one click away, but in those minutes delaying reading the article I went even further with my reflections on this peculiar statement: Why information and not information technology? Why not enterprise eats the information? And, what is the meaning of "eats" ultimately?
I read all the text in one gulp. It was an article entitled "How to Compete When IT Is Abundant", published by Harvard Business Review (HBR) on June 26, 2013. It started by presenting the prediction made by Nicholas Carr in another article published by HBR ten years ago that claimed that information technology (IT) would not provide any competitive advantage in the future. Starting from this idea, Aaron Levie, the author of the current article, highlighted that the prediction was partially true as IT shifted indeed from supporting the core business to "*becoming intrinsic to the very products and services that every company offers*". Consequently, IT has become even more important for organizations and what truly transforms IT in a competitive advantage is the information:

> *"In this transition from a world of IT scarcity to abundance, competitive advantage has little to do with unique access to technology, and everything to do with unique access to — and use of — information. When technology is near-ubiquitous, it's the connection between people and information that drives business forward. Organizations that capitalize on this trend will ensure that as information eats the enterprise, they'll be the ones satiated."*

The idea that the role of technology for an organization has changed considerably over the last decades has been extensively discussed by Ward and Pepper (2002) too. While the first adoption of technology aimed at supporting the operational level of an organization, technology has become a strategic enabler more recently (Ward and Peppard, 2002). Moreover, the support of flexible business processes in strategic information systems has been defined as critical to properly handling the complexity of business requirements and the fast-paced environment. Nevertheless, how could flexible business processes be integrated with information systems and technology for offering competitive advantage? Is indeed information a key resource for enterprise nowadays? Could the information improve the integration of flexible business processes, and information systems and technology? In this thesis, I will show that information could substantially contribute to the effective adoption of flexible business processes. The reasons why and the manner how it does are progressively presented throughout the following chapters.

1.1 Problem Statement

A category of information systems, highly adopted by organizations, is Process Aware Information Systems (PAISs) defined as "*software systems that manage and execute operational processes involving people, applications, and/or information sources on the basis of process models*" (van der Aalst, 2011). Information systems such as workflow management systems, business process management systems and enterprise resource management systems fall into this category. Consequently, PAISs are a special type of information systems which incorporate business process schemas explicitly or implicitly.

By ensuring flexible processes in PAIS, a greater importance is given to human power of making decisions, but also to human ability to understand the process context and to use the process knowledge. Flexible processes imply that humans can adapt the processes when they want, according to their reasoning, or they can make different process-related decisions when required by PAIS. As the quality of decisions impacts the outcomes of flexible processes, humans become central to the flexible PAIS. An experienced process participant who is highly aware of the business process is more able to make a decision about which activity to execute next or how to model a process fragment at run-time, considering the situation at hand. In contrast, this can be very challenging for a less knowledgeable process participant who might be used to rely on the prescriptive definition of processes, or for a process participant who faces a very dynamic process environment (Schonenberg et al., 2008; van der Aalst et al., 2009).

Consequently, the problem addressed throughout this thesis consists of the difficulties encountered by process participants when interacting with flexible PAISs.

Obtaining the benefits promised by the adoption of flexible processes depends significantly on how process participants enact the processes. If this problem is ignored, the adoption of flexible processes can negatively impact the organizations because of the complexity it introduces in the enactment. Nevertheless, a proper support of process participants can relieve their interaction with flexible PAISs, thus minimizing the negative consequences produced by the complexity of flexible processes enactment.

1.2 Research Objectives

One of the characteristics of PAISs is that they could offer support for discovering the processes in a bottom-up manner by collecting event logs during enactment. This is enabled by process mining, a discipline which continuously evolved in the last years and its main goal is "*to discover, monitor and improve real processes*" by transforming the event logs data into valuable knowledge (van der Aalst, 2011). Besides its main goal, process mining has been used as a starting point in several approaches to supporting process participants such as suggesting them possible future activities based on the extracted process knowledge (Schonenberg et al., 2008; Van Der Aalst et al., 2009; Sun, Huang, and Meng, 2011).

While these solutions successfully use process mining to support process participants during enactment, I consider the given recommendations semantically not rich enough to support effective decision making. The recommendations are formulated based on the mined process model which is frequently represented as a control flow of activities. Therefore, in order to semantically enrich the recommendations, the mined process model must be enriched.

The first objective of the current research aims at developing a process mining technique for discovering the intentional facet of a process: the intentions behind a process enactment. The second objective is to create a tool integrated with the intention mining technique which offers recommendations based on the discovered intentional process model and to investigate to what extent the process participants' interaction with flexible PAISs is improved through its usage.

1.3 Research Questions

The current research project will be guided by the central research question:

To what extent does a recommendation tool based on intention mining improve the process participant's interaction with flexible process aware information systems?

The main research question is answered through the investigation of three sub-questions:

1. How does the implementation of flexible processes in process aware information systems impact the process participants?
2. How can the intentional process, behind the interaction of process participants with process aware information systems, be mined?
3. How can the process participant's activities impacted by the implementation of flexible processes be supported through a recommendation tool based on intention mining?

The ultimate goal of this research project is to investigate if the problem context (the difficulties encountered by the process participants when interacting with flexible PAISs) can be solved by introducing a recommendation tool based on intention mining. This goal is achieved:

- First, by understanding the implications of implementing flexible processes;

- Then, by investigating how the intentions behind the enacted process can be mined (the creation of the process mining technique);
- Finally, by creating the recommendation tool based on intention mining and by analyzing how it could support the activities of process participants that are impacted by the implementation of flexible processes.

1.4 Research Relevance

The topic of flexible business processes has been extensively discussed in the literature from both an organizational and a technical perspective. Moreover, while the advantages of flexible processes have been highly recognized by academia and organizations, the research focus has recently shifted on its trade-offs and how their negative consequences could be minimized. Complementary to this top-down evolution within business process management area, process mining has gathered increasing interest by providing a bottom-up analysis and support.

This current research will benefit the research community by proposing a new process mining technique which aims at discovering the intentional facet of a process. Furthermore, it will benefit the organizations by systematically analyzing the impact of flexible processes' implementation on process participants. Finally, these research directions are integrated in a recommendation tool based on the intention mining technique which aims to improve the process participants' interaction with flexible PAISs.

1.4.1 Scientific Contribution

The scientific contribution of this research project is twofold. Firstly, I will contribute to the scientific body of knowledge with a systematic, human-centered analysis of flexible PAISs. The results of this systematic literature review are further used as a foundation for developing a support tool for process enactment.

Secondly, this research is intended to the creation of an intention mining technique, incorporating unsupervised machine learning. Intention mining is in an early stage of research and the only related research was conducted by Khodabandelou et al. (2013) who used a supervised learning approach, namely Hidden Markov Model. The difference between the two approaches is: in supervised learning the process model is known in the beginning and it is used for training the technique, while in unsupervised learning the process model is not known being identified from the data according to some heuristics. Most of the mining techniques capture the control-flow view of a process, at a very low granularity level (Aalst, 2011). A high level, intentional perspective, could improve the expressiveness of the as-is process model, thus offering a better support for process analysis and support.

1.4.2 *Societal Contribution*

The direct beneficiaries of this research are the process participants of flexible process aware information systems who could have a better support in working with flexible processes. This is achieved by more effectively supporting the decision making process, by the instrumentality of the created recommendation tool. Moreover, this research will benefit the organization itself by enabling a more effective adoption of flexible business processes trough the support of the process participants. An effective decision making can lead to a more effective and efficient process enactment.

Finally, intention mining could give very important insights into the behavior of process participants during process enactment. This knowledge can be used by the process administrator and the process owners to improve the executable processes by understanding how processes are enacted in reality, compared to the conceptual, pre-defined models.

1.5 Thesis Overview

The thesis is structured as it follows:

Chapter 1 "Introduction" provided a synthesized background of the problem, defined the problem statement, the research objectives which were further formulated as research questions, and presented the relevance of the current research project.

Chapter 2 "Research Method" describes the research method and the research approach for answering the main research question and its sub-questions.

Chapter 3 "State of the Art" provides a theoretical background of the process aware information systems, intentionality and process mining. It ends with a brief description of the previous work regarding the support and guidance solutions for process enactment based on process mining.

Chapter 4 "Agent-centered Analysis in Flexible PAIS" corresponds to the first research sub-question analyzing how the process flexibility is supported in PAIS and how this influences the activities of humans interacting with the system. The chapter ends with the formulation of several requirements regarding the recommendation tool.

Chapter 5 "Intention-mining Technique" corresponds to the second research sub-question and it describes how the intention mining technique was created: the design and the development. The discussion is sustained by a case study example, presented in the beginning.

Chapter 6 "Intention-based Recommendation Tool" focuses on the third research sub-question, the creation of the recommendation tool. The structure of the chapter is similar to the previous one, starting with an example and then continuing with the presentation of the tool's design and development.

Chapter 7 "Artifacts' Evaluation" presents the evaluation of the two created artifacts: the evaluation method, the evaluation process and the results.

Chapter 8 and Chapter 9 conclude this research by presenting the closing remarks regarding different aspects of the project and the directions for future research.

2 RESEARCH METHOD

The current research project followed the problem solving cycle proposed by Mitroff (1974). A real-world problem was identified in its natural environment. The problem situation was further abstracted in a descriptive model which was the basis for defining the solution. The formalization of the solution allowed me to project the prescriptive model of the improved situation (Figure 1). Nevertheless, through its implementation the solution was evaluated.

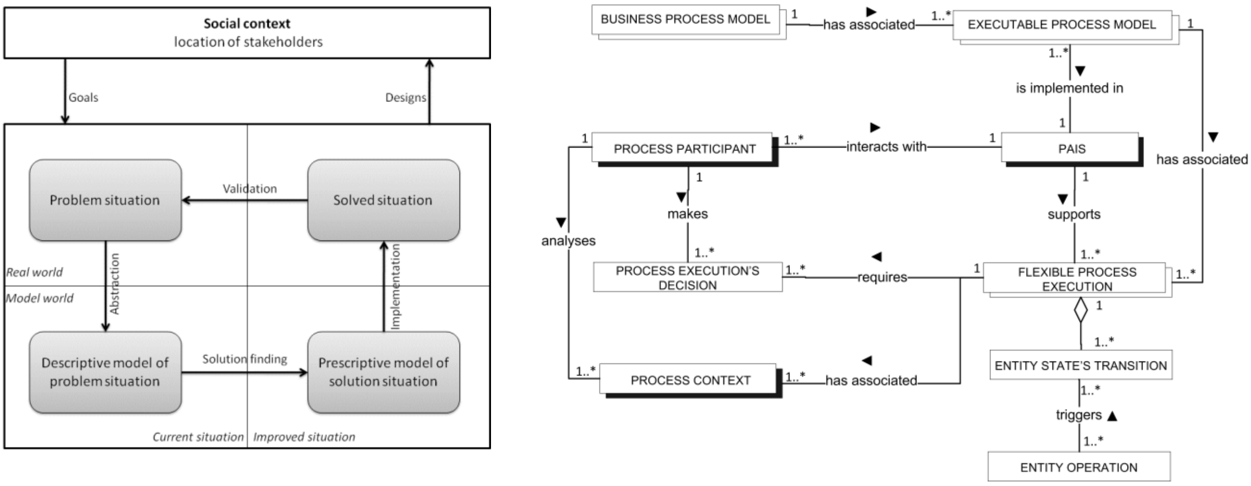


Figure 1. The problem solving cycle (Mitroff, 1974) | The descriptive model of the problem situation

Formally, I chose design science research as the research method guiding the development the thesis. Design-science research is a solution-oriented approach consisting in the design of artifacts aimed to solve a problem context and the investigation of the designed artifacts in this context. While Hevner, March, Park, and Ram (2004) limit the context to organization's boundaries, Wierienga (2009, 2010) expands it to a social dimension marked by the goals of the internal and external stakeholders.

2.1 Design Science Research

The design science research paradigm is especially valuable because it addresses the role of IT/IS artifacts and their relevance in the domain of application. Therefore it is considered a pragmatic approach of solving real-world business problems (Hevner & Chatterjee, 2010; Hevner et al., 2004).

2.1.1 Design Science Research in Theory

Hevner et al. (2004) identifies three research cycles during a design science project (Figure 2):

1. The relevance cycle consists of the analysis of the application domain (people, organizational systems and technical systems) in order to identify potential areas of improvements. The result of this cycle is the definition of the solution's requirements and the acceptance criteria. Furthermore, the design processes (search heuristics) for building the artifacts are also defined.
2. The rigor cycle grounds the design in the existing theories and methods. Moreover, the state of the art of the application domain is created based on the experience and expertise (the body of knowledge) and the existing artifacts and processes relevant to the application domain. The knowledge extracted from the design cycle and presented as research contributions is finally added to the knowledge base.
3. The design cycle is the central element having two sub-components: the "build" and "evaluate" processes. For building the artifacts, it uses as input the solution's requirements defined during the relevance cycle and the theoretical background described during the rigor cycle. The evaluation can trigger a new execution of the design cycle by exploring different design options if the artifact under construction does not reach the required level of acceptance.

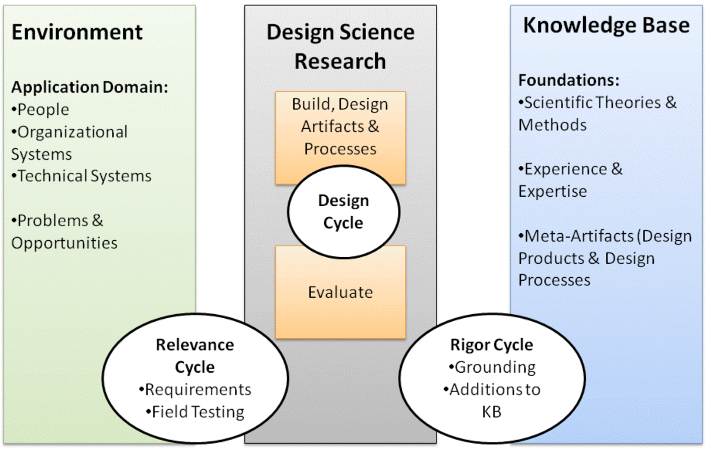


Figure 2. The design science research cycles (Hevner et al., 2004)

The knowledge base is founded on mathematics, social science, natural science, design science, design specifications, useful facts, practical knowledge, common sense and other beliefs (Wieringa, 2009). In general the theories are used for description (conceptual frameworks to describe a phenomenon), explanation (generalizations to admit causes of a phenomenon) and prediction (generalizations to predict effects of a phenomenon). The theories are not prescriptive but rather they can be useful and usable with regard to the artifact's design (Wieringa, 2010).

The design science research development process proposed by Peffers, Tuunanen, Gengler, Rossi, Hui, Virtanen, and Bragge (2006) was used during the project. This is a conceptual process consisting of six activities in a nominal sequence:

- During the research problem identification phase, the outcomes of solving this problem are stated by justifying the value of such a solution. The complexity of the problem is handled by dividing the main problem in sub-problems that could be solved separately.
- The solution's objectives are inferred based on the definition of the problem and they can be qualitative and/or quantitative.
- The design and development of the artifact start with the identification of the artifact's functionality and architecture through the collection and analysis of the relevant data. The motivation of the design decisions are usually sustained by the existing knowledge.
- The demonstration of the artifacts' efficacy in solving the problem has to be done in a suitable context, following a rigorous approach.
- The evaluation of the artifact has as a reference the solution's objectives established in advance. The artifact is observed and its outcomes are measured during the demonstration phase. First the functionality of the artifact is compared to the solution's objectives, and then the satisfaction is evaluated by processing the feedback of the involved stakeholders.
- The communication starts with outlining the problem's importance. Then it continues with the presentation of the solution, discussing the utility of the artifacts and the stringency of their design processes.

2.1.2 Design Science Research in Practice

Hevner et al. (2004) postulate seven guidelines for capturing the fundamentals of design science research and for assisting the researchers during the research process. The adaptation of these guidelines to the current project is described in the following table:

Guideline	**General description**	**Research project description**
Design an artifact	To produce practicable artifacts such as constructs, models, methods, and instantiations.	**Artifacts**: the intention-based recommendation tool and the intention-mining technique.
Problem relevance	To provide solutions to relevant and significant problems	**Problem**: the support of process participants in enacting flexible processes has been recognized as a necessity but the current solutions based on process mining do not provide an effective support for decision making.
Design evaluation	To evaluate methods to prove the utility, quality, and efficacy of the produced artifacts	**Evaluation method:** the artifacts are evaluated in a case study through interviews organized with stakeholders (process participants, process owner);

		the acceptance evaluation is preceded by a functional evaluation.
Research contribution	To state clearly and in a verifiable manner the contributions of the designed artifact, methodologies and foundations.	**Contributions**: a new process mining technique which discovers the intentional facet of a process, a systematic analysis of the flexible processes' impact on process participants, and an intention-based recommendation tool.
Research rigor	To ensure the methods for constructing and evaluating the artifacts are rigorous	**Evaluation's rigor**: the construction of the artifacts is grounded in the existing body of experimental knowledge and theories; a case study protocol is followed.
Design as a search process	To use all the available means to ensure that the search for an accepted artifact satisfies the problem's laws.	**Means**: various sources of knowledge are utilized for the creation of the artifact such as the existing literature, the case study analysis.
Communication of the research	To ensure the communication is effectively done for both the technical and management audience	**Communication**: several presentations were given and a scientific article was submitted to the International Conference on Information Systems (ICIS 2013)

Table 1. Design science research guidelines

In summer 2012, Centre de Recherche en Informatique, Paris 1, proposed a collaborative project whose aim was to develop a tool for offering recommendations to the users of various information systems. The tool is based on the intentional process which must be automatically and in advance discovered using a process mining technique. The project was positioned at the intersection of three areas of interest for the Business Informatics research group of Utrecht University: Method Engineering, Process Engineering and Algorithmic. Since intentional processes are acknowledged to be more powerful to support human-computer interaction, this appeared like a very good opportunity to explore process mining from a completely new and innovative angle.

Problem identification and motivation: the initial scanning of the literature regarding the process aware information systems revealed that incorporating flexibility has become mandatory for maintaining the alignment with the organization's goals. Furthermore, the main tradeoff introduced by the flexibility has fallen upon the process participants who must be experienced and knowledgeable to make decisions during the process enactment, and to better understand the process context. The training of process participants cannot cover all the possible situations especially when dealing with highly flexible, knowledge intensive processes. Hence, the addressed problem was the necessity of a method/tool for supporting the process participants in making decisions during the enactment of flexible processes based on the intentional process model.

Objectives of the solution: the objectives of the solution are mapped on the three research sub-questions. The first objective was to understand how flexible processes were implemented in practice and how they impacted the involved parties. The second objective was focused on understanding what intentionality meant and how it could be captured from the systems, for mining the intentional process. The third objective was the creation of a recommendation tool that could minimize the negative consequences of flexibility, discovered in the first research sub-question. The prototype has as input the intentional process model, discovered with the created process mining technique, and it produces as output recommendations representing the next intentions that could be achieved, each of them accompanied by a confidence factor.

Design and development: the first research sub-question was answered based on the analysis of the existing literature regarding the implementation of flexible processes in process aware information systems. The design decisions were formulated after the discussion of the first sub-question. Moreover, the motivation of using an intentional approach is given in the following chapter through an analysis of intentionality as a standalone topic, and the current process support and guidance solutions. The other design decisions were made after exploring and thoroughly understanding various options of mining a process (process mining techniques), of capturing intentionality (text mining, natural language processing, ontologies), and of giving recommendations (machine learning algorithms for prediction). While the conceptual design was partially shaped in advance for both artifacts, the final design and development was targeted for a specific flexible PAIS, during a case study which will be discussed in a following section.

Demonstration and Evaluation: these phases focused on two aspects of the created artifacts: functionality and the extent to which they solved the problem or achieved their objectives. The functionality of intention mining technique was evaluated through an experiment with process participants. The functionality of the recommendation tool was evaluated based on the artifact's specifications. The stakeholder's acceptance of the recommendation tool was analyzed during individual semi-structured interviews with the process owner and process participants.

Communication: the complete project description is presented in this thesis. Intermediary communication was done through presentations, company's meetings and scientific articles.

2.2 Literature Review

Literature review was used to establish the theoretical background of the current project and to answer the first research sub-question. Levy and Ellis (2006) summarize that an effective literature review should include a systematic analysis of the existing literature, should establish firm foundations of the research topic and of the research methodology, and, finally, should outline what are the contributions of the research to the existing body of knowledge. Doing so, the researchers ensure that the theoretical foundation of the research is built and that the current research is related to the literature (Levy & Ellis, 2006).

The process proposed by Levy and Ellis (2006) for the literature review was followed. The process has a main part, consisting in a processing unit which receives an input and produces an output. The sequential steps, within the processing unit are: know the literature, comprehend the literature, apply the literature, analyze the literature, synthesize the literature and evaluate the literature.

Both peer-reviewed and non peer-reviewed publications were searched in the electronic databases such as IEEE, Elsevier Science Direct, ACM Digital Library, Ebsco, JStor and Springer Link using Omega and Google Scholar search engines. Except of these, the publications of “Business Process Modeling, Development, and Support” and “Business Process Management” conferences were scanned too. Backward references search was also applied. Several pre-search criteria were imposed: the articles must have been written in English, the publication date should have been at least 2000 and the journals or conferences where the articles were published must have been related to the areas of interests. The keywords used in the searching process depended on the scope of the literature review. In the next phase, the abstract of the selected articles was screened and an initial selection was made to be included in the review. The final selection was made based on the content of the articles. Some aspects of the literature review process are presented below:

Reviewed topic	**Review’s goal**	**Search keywords**
Business process management	To create the theoretical foundation	business process management
Business / executable processes	To create the theoretical foundation	business / operational / executable process
Process aware information systems	To create the theoretical foundation	process aware information system
Intentionality	To create the theoretical foundation To motivate the design	intentional process, intentional model, users intentions, belief desire intention, human-centered system
Process mining	To create the theoretical foundation To motivate the design	process mining, process mining technique
Process support	To analyze the existing solutions	process guidance/recommendations

Flexible processes	To answer knowledge questions To motivate the design	flexible process/workflow, flexible business process

Table 2. Literature review approach: topics, goals and keywords

2.3 Case Study

Case studies can provide a systematic and rigorous manner of analyzing events and information, collecting data and reporting the results. Furthermore, through the study of a single case, researchers are able to investigate a phenomenon in depth, getting closer to the phenomenon, enabling a rich description and revealing its deep structure (Yin, 2009). Additionally, three different categories of case studies are identified by Yin, namely: exploratory, descriptive and explanatory. This case study might be considered explanatory since it aims at explaining a causal, difficult relationship between the treatment with the designed artifacts of the problem context and the expected effects.

Case study research method was chosen for:

- Providing real-life knowledge that influenced the decisions of the artifacts design;
- Providing the resources necessary for the demonstration of the artifacts;
- Providing a real-life context for evaluating the artifacts.

Two categories of knowledge questions were investigated during the evaluation: artifact-oriented and problem-oriented (Wierenga, 2009). The questions regarding the artifact and the problem are presented in Table 3 and they are revisited when reporting the evaluation.

Artifact-oriented knowledge questions	• Effect question: what are the effects of treating the problematic context with the artifact? • Requirements satisfaction question: do the effects satisfy the functional and non-functional requirements of the artifact? • Trade-off question: what are the effects of treating the same problem context with another artifact?
Problem-oriented knowledge questions	• Stakeholder question: who are the stakeholders? • Goal question: what are the stakeholders' goals? • Requirements question: what requirements should the artifacts satisfy? • Contribution question: do the designed artifacts contribute to the goals satisfaction?

Table 3. Knowledge questions used in the evaluation

2.3.1 Case Study Selection

The selection was made based on the suitability of the case organization (the support of flexible processes through its software product) and its availability (the willingness to give access to the resources for interviews, documents and records scanning, and observations).

The case company, 42windmills, is a software company located in the Netherlands. The main product of the company is a platform which supports model driven development in an agile manner. The platform together with a Web-based application designer enables the customers to design, preview, re-design and deploy a wide variety of business applications. Providing a system to customers that can be adjusted to the changing organizational context is a key aspect of the company's vision. Business flexibility is ensured in two ways. On one hand, the customer business applications can be generated, reducing substantially the development time, and the further extensions are realized according to the agile principles. On the other hand, even if some of the processes of the resulting business application can be automated, most of them incorporate flexibility by providing a data-centered, human-driven enactment. In a data-centered approach, the elements that influence the process enactment are entities, entity attributes and entity relationships. A transition during the process execution is triggered by a change in the entity state through user forms (Reichert & Weber, 2012).

The current research project is focused on a specific software system created by 42windmills and used by several child care centers in the Netherlands. The Childcare system supports all the main processes and consists of two Web-based software components: a back-office application and a Parents portal. The main process modules (Figure 3) are: the customer relationship module including the registration of the parents, locations and children; the contracts administration module including the recording, storage and automatic handling of the contracts between parents and day care centers; the planning module including the child planning based on groups, locations and time periods; and the accounting module including invoicing, billing, credit management functionalities and integration with a bookkeeping system.

The most complex and flexible process is the child planning. It is triggered by the registration process and it communicates with the other modules too. Its complexity is represented by the variety of existing scenarios regarding child planning and by the fact that, even if context information is provided by the system, the process participant decides on how to enact the process accordingly. The planning depends on the age of the child, the locations and the preference of the parents for having their child in a horizontal group targeting the same age, a vertical group consisting of children of different ages but in the same proportion, and a variable group. Except of the standard planning, there are also other non-standard types: alternating planning when the groups, the locations, or/and the time slots are alternated for a child, occasional planning which is an incidental, non-repetitive care, and holiday planning. The process of planning includes also the management of a waiting list and the management of the groups' occupation. Depending on the chosen planning, the other processes can also change such as the financial flow requires specific actions of the process participant in the case of incidental planning or holiday planning, but not necessarily always the same.

Child Registration
- Portal registration
- Back-office registration

Child Care Planning
- Standard child care planning
- Alternating child care planning
- Incidental child care planning
- Holidays child care planning

Automatic Contract Handling

Invoicing

Figure 3. The main Childcare process

While the Childcare system supports a high degree of process flexibility, it has been noticed during its usage so far that this can also create problems. Inexperienced process participants often enact inefficiently the processes or make mistakes during enactment because of the scenario complexity and the lack of experience. Moreover, the process owners have noticed that even if a process was expected to be enacted in a specific way, the reality was significantly different. Consequently, process mining can offer very useful information about how the system is used and also could represent the basis for recommendations during enactment. Further, one of the leading design decisions was that offering support at a low level of granularity will not improve substantially the process enactment. For example, considering that the process participant has to make decisions, the system could suggest several options on how to execute the process at a specific moment based on the aggregated process experience. However, if the system suggests activities such as "change the value of attribute A of entity E" or "create a link between entity E1 and E2", this would be too vague for actually helping the process participant in making the decision. However recommending the possible intentions such as "Update the contract" or "Create a new group for location" and the possible strategies for achieving an already chosen intention can improve the decision making. Deciding in terms of intentions and strategies is closer to human reasoning and understanding, as it will be presented in the next chapter. After the decision is made at the intentional level, the system can indeed suggest to the process participant the activity to be executed next.

2.3.2 Case Study Protocol

In order to ensure the validity and reliability of the gathered data regarding the problem and the results, a case study protocol was defined beforehand. First the **data**

collection was driven by the questions regarding the artifact and the problem stated above. Several assumptions that influenced the design of the artifacts were made such as: the process mining technique and the recommendation tool could improve the problem context because they are intention-centered and not activity-centered; the recommendation tool is more reliable because it incorporates process experience without relying on a prescribed process model. Further, the **units of analysis** were defined: the process participants of the Childcare system. Finally, the **interpretation criteria** for the findings, and the logic linking between the collected data and the made assumptions were established

The case study started with the investigation of the product and various internal documents about the company. In this way, I got familiarized with some technical concepts behind the product and could identify the product's components that were of interest for this research. Additionally, during the first meeting with the company, an on-site demonstration of the product was given. After the first meeting, the Childcare system was chosen for deeper investigation. Access to the product and to its source code was provided during the whole research period.

Several interview sessions were held too in order to deepen the understanding of the problem company was facing, to study more thoroughly the technical aspects of the product and to validate the suitability of the chosen solution.

The first interview was conducted with one of the company's founders, the company's Chief Technology Officer (CTO). During this first interview open questions were asked regarding topics such as the company's profile, the product's functionalities and its software architecture. The information collected during this interview was presented in the previous section.

Two interviews were held with the consultant responsible for Childcare and the goal was to understand the processes, how the system is used for the enactment, what is the entity model and what were the problems identified so far in the organizations where the system was already deployed. Based on these interviews, the main Childcare process was described (in the previous section) and a formal definition was created (in the Intention Mining Technique chapter). The other information such as the usage of the Childcare system and the entity model were used in the design of the artifacts and the evaluation of the artifacts, namely the experiment's definition.

Two interviews, followed by multiple discussions during the artifacts' development, were organized with the platform architect and they were focused on understanding the technical and architectural design of the product, what were the possibilities of extending the system for supporting logging and what kind of information could be logged. The technical details of the Childcare system which were extracted from the discussions with the platform architect are documented in the chapters presenting the artifacts.

Finally the last interviews focused on validating the chosen solution and they will be described in details in the chapter presenting the artifacts' evaluation.

2.3.3 Research Validity

To ensure the quality of the research and the obtained results, the following validity factors were considered: construct, internal, external and reliability (Yin, 2009).

The **construct validity** ensures the correctness of the measures involved in the investigation, the relevance of the used concepts and the proper chain of evidence. In order to secure this, a first scanning of the company's public information and an exploratory interview was conducted to gain a better understanding of the case company and to be able to select the right information system and the right people for the following interviews. Additionally, several control actions were taken: the interviewees were informed beforehand about the interview's topic and purpose, the interviews were documented in real time and a research database was maintained. Multiple sources of evidence were used in the data triangulation for ensuring the construct validity: primary and secondary documents, artifacts, direct observations of the events, interviews with the involved parties.

The **internal validity** ensures that the causal relations used in research are properly motivated; for instance the results obtained and the assumptions made are correctly based on the collected data. In order to be sufficiently objective in creating the right assumptions and building the explanation, multiple sources of data were used in this research: literature study, interviews, technical product documentation and source code.

The **external validity** is concerned with the generalization of the results and thus more difficult to be guaranteed in a single-case study. Due to time constraint, the author decided to focus only on this company, but future work aims at applying this research for other case companies as well.

The **reliability** was respected by the use of a case study protocol and carefully documenting the collected evidence in a case study database. By doing so, the same results will be generated if the research is performed by a different team, following the same research approach.

2.4 Research Process in a Nutshell

To sum up, the research method (Figure 4) is depicted by applying the meta-modeling technique proposed by Weerd & Brinkkemper (2008) which is composed of two parts: the meta-process modeling and the meta-data modeling. Each of these two phases produces a specific input in the process deliverable diagram (PDD), respectively in the process component represented as an UML activity diagram and in the deliverable component represented as an UML class diagram (Weerd & Brinkkemper, 2008).

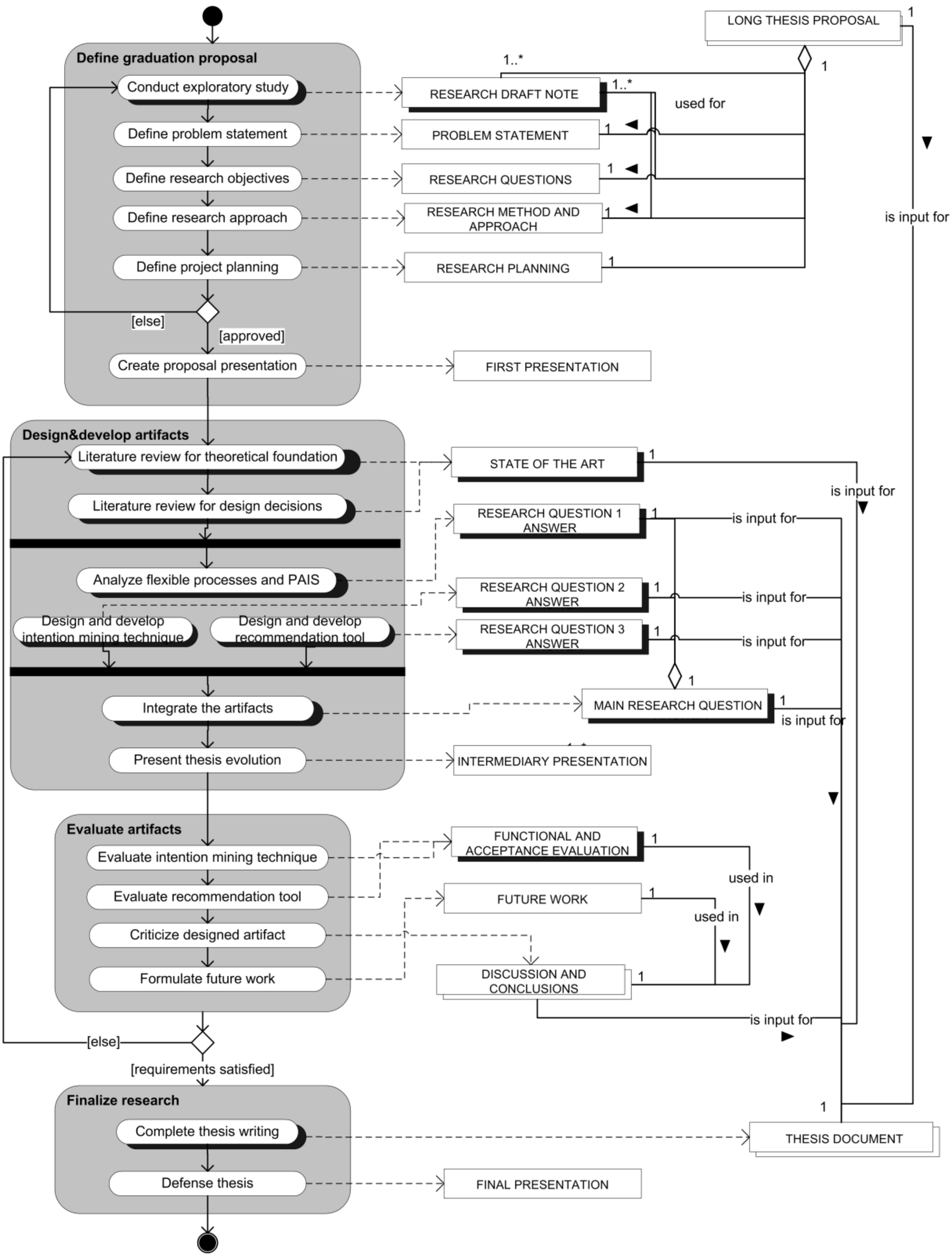


Figure 4. The process deliverable diagram depicting the current research process

3 STATE OF THE ART

The theoretical background regarding the current research is presented by exploring several areas of interest: process aware information systems, intentionality and process mining. Some existing approaches for process enactment support based on process mining are discussed too.

3.1 Process Aware Information Systems

In this section, the fundamental concepts of business process management and process aware information systems are presented in an evolutionary discussion. Firstly, business process management is introduced. Then the transition from business processes to system-ready, executable processes is described in order to explain the integration with enterprise systems. Finally, a presentation of process aware information systems from a historical and general perspective is provided.

3.1.1 Business Process Management

Business process management (BPM) provides a systematic governance of the end-to-end business processes and their environment, in order to control and improve the business performance. In other words, business process management represents a different approach of managing an organization whose core functions are implemented as business processes with a central focus on the customers (Hammer, 2010). According to Weber, Sadiq, and Reichert (2009) the definition of business process management has been extended in the last years to explicitly encompass other aspects, such as human involvement in process lifecycle and inter-organization collaboration. Among the main advantages yielded by BPM, a better overall quality of services and products, reflected in both the organization's performance (lower costs, faster speed, better consistency) and customer's satisfaction, is mentioned. Furthermore, a faster and more effective reaction of organizations to the changing environment has also been reported (Hammer, 2010).

Depicting in more details, Rosemann and Vom Brocke (2010) outline the core elements of business process management (Figure 5): strategic alignment, governance, methods, information technology, people and culture. The BPM requirements should be driven by the organization strategy so as to the strategic priorities of the organization are aligned to the process lifecycle needs. The alignment relationship must also be supported in reverse: the process capabilities and prospects could serve as input for the design of the organization's strategy. Governance comprises of the roles and the descriptions of their responsibilities at different BPM levels as follows from top to bottom: portfolio, program, project and operations. The decision making process and the reward process are the main components of the governance. Methods consist in the set of tools and techniques which support process lifecycle phases such as process modeling and process analysis. Information Technology is one of the BPM enablers making possible a more efficient support of business processes within organizations. However, most of the business processes cannot be fully supported by information systems through their lifecycle phases. Therefore people, as individuals or groups, are one core element, being actively and significantly involved. Finally, the organizational culture consisting of values and beliefs should be adapted to a process-orientation environment in order to foster a more efficient BPM adoption and implementation.

Figure 5. The BPM core components (Rosemann & Vom Brocke, 2010)

The central element of BPM is the business process which is defined as "*a set of business events that together enable the creation and delivery of an organization's products or services to its customers*" (Gelinas, Sutton, & Fedorowicz, 2004). A business process is described by Gelinas et al. (2004) from two perspectives: horizontal and vertical. The horizontal perspective is a lower level translation of the business process in a set of business operations which are further segmented in sequences of events. The vertical perspective implies the business process is composed of three logical components: an information process, an operations process and a management process. The information process is the concrete, low-level implementation of the business process supported by the information systems while the operations process is the high level organizational process involving human interaction. The management process has a support purpose and is focused on planning and controlling the organizational aspects. Similar to the vertical perspective introduced by Gelinas et al. (2004), Hammer (2010) introduces three views of the business processes, but identified in relation to the customer types. The core processes focus on the external customers, enabling/support processes focus on the internal customers, and governance processes focus on the process management, risk and performance at the organizational level.

The management of business processes could be decomposed in the following sub-activities: the formal definition, the ongoing assessment of the performance indicators established in advance such as customer needs and organizational requirements, and the definition of the intervention plans in case of performance inefficiency (Hammer, 2010). Nevertheless, the business process management activities are very often mapped on and evaluated in connection to the process lifecycle phases (Rosemann & Vom Brocke, 2010; Weber et al., 2009):

1. *Design*: a management top-down analysis of the processes driven by the business goals and organization' strategy. Several approaches have been proposed such as Capability Maturity Matrix (Software Engineering Institute, 2008), Porter's value chain (Porter, 1985), Process Handbook (Malone, Crowston, & Herman, 2003), Reference Models (Scheer, 1994).
2. *Model*: the modeling of the business and operational processes using a specific meta-model such as Petri Nets (Petri, 1962), Business Process Modeling Notation (Object Management Group, 2004), MAP (Rolland, Prakash, & Benjamen, 1999), Process Delivery Diagram (Brinkkemper, 1996; van der Weerd & Brinkkemper, 2008)
3. *Execute*: the accomplishment of the process tasks, translated in operations, involving organization's resources and information systems support.
4. *Monitor*: the post-enactment analysis of the processes based on the collected traces. It provides a real overview of the executed processes and it could assist the process administrator in diagnosis.

A business process management approach of the organization does not necessarily results in an improved performance. In order to achieve the established performance targets, a process design is mandatory because it enables a better alignment with the organization strategy from an early stage of the business process' adoption. Other performance enablers rely on the human roles. The process owner is responsible for managing the end-to-end processes and process performers. The process participant should adapt his skills to the new activities involved in the process enactment. A process infrastructure consisting in the integration of information systems and human resources can lead to a better performance too. Finally, another enabler is the definition of the process performance metrics in advance, before the processes are implemented and deployed (Hammer, 2010).

3.1.2 Towards Executable Business Processes

Another more practice-oriented definition introduced by Russel (2007) states that business process management represents a means of supporting business processes lifecycle through methods, techniques and software, used for designing, enacting, controlling and analyzing operational processes. A difference is made between the business processes and the processes that are actually enacted named operational or executable processes. The translation of the business process in an executable process is realized during the model phase of the process lifecycle which captures a high-level, global view of the business and offers support for simulation and visualization by enabling the business-technology dialogue (Reichert & Weber, 2012). The executable process could also serve as input for the business process design as result of the monitor phase. A complete mapping of the business process model on the executable process is rarely achievable. Except from the automated activities, a process could also contain manual or system-supported activities which require a full or partial human involvement. Even if the mapping of the business process on the executable process is not complete, an ongoing optimization and reengineering process should be in place, to maintain their partial synchronization (Reichert & Weber, 2012).

Reichert and Weber (2012) classify the executable processes in activity-centered and data-centered. In a data-centered approach, the elements that influence the process enactment are objects, objects attributes and objects relationships. A transition during the execution is triggered by a change in the object states. Activity-centered processes are represented by two categories according to the degree of the system automation: pre-specified processes and knowledge intensive processes. Pre-specified processes are repetitive and known in advance; therefore they can be modeled and highly automated. The models of this type of processes include the activity description, the control flow and the data flow. Knowledge intensive processes are dynamic and cannot be completely specified and automated. The human component is very dominant during their enactment as the participants decide on the activities they want to execute and the execution order. Being often loosely specified processes, the process participant's decisions have a significant impact on the completion of the process definition (Reichert & Weber, 2012).

Russel (2007) characterizes an executable process from the control flow, data and resource perspectives. In a control flow view, the process is described as a set of activities, simple or complex, and their interconnections. The data perspective focuses on how the process data is defined and used during enactment. The resource view describes the organizational context of the process such as the manner how resources are linked to process items during enactment. Moreover, Russel (2007) claims that despite the significant evolution of the business processes and information systems, a formal and globally accepted definition of the business process is missing. Consequently, it results in a gap between how various business process concepts are described at design time through modeling and how they are practically enacted at runtime through a specific technology. In order to undertake these problems, Russel (2007) proposes a formalization of the business process core constructs, presented as a pattern catalogue with four main categories mapped on the aforementioned perspectives: control flow, data, resource, and, additionally, exception handling.

Dumas, Aalst, and Hofstede (2005) analyze the executable processes based on the type of the organizational resources involved. The first category of processes – "*Person to Person*" or P2P, consists of processes where human intervention is dominant and the system tools have only a support role in human-driven process enactment. The second category, "*Application to Application*" (A2A), assembles all those processes that can be completely automated and enacted by the system. The last category, "*Person to Application*" (P2A), is found at the intersection between the other two categories, as both humans and applications are involved in the process enactment to different extents. The extent to which the process enactment is more system or human influenced is very dependent on the structure and predictability of the process. Ad-hoc processes have partially pre-identified models; therefore they are mainly influenced by the human decisions. Contrary, administrative and production processes, which are highly predictable and composed of repetitive, tasks, are enacted almost entirely by the system (Dumas et al., 2005).

3.1.3 Enterprise System – Business Process Integration

The enterprise systems are integrated software packages whose purpose is to ensure the integration of the existing organizational systems and data. The understanding and integration of business processes with the enterprise systems is doubly crucial because these systems are the basis for integrated management and operational support. Processes frequently exceed the organization's boundaries, fostered by e-Business (Gelinas et al., 2008). Electronic networks enable the interaction between the internal processes (also known as back-office processes) and the external processes (also known as front-office processes).

A shift from systems that incorporate over-specialized functional elements to systems capable of supporting business processes has been widely noticed in the last decades (Weber et al., 2009). This transition has been reflected in the development of enterprise systems and in their adoption by organizations. Workflow systems have been the most widespread process-centered systems. However, this solution was focused on working with predefined and repetitive processes which proved to be insufficient when process flexibility became essential for improving the organization's performance (Weber et al., 2009).

While the first integration of enterprise systems with business processes targeted the enactment phase, enabled by the workflow technology, with the wide dissemination of the business process management approach, the IT solutions focused on covering all the process lifecycle phases (Rosemann & Vom Brocke, 2010). The following solutions are identified (Figure 6):

- IT solutions for process design and modeling: two types of tools are identified of which the first category is used for process modeling, simulation and animation, and the second category focuses on mining a process model based on the system logs.

- IT solutions for process implementation and execution: focus on transforming the process models and specifications in operational processes ,incorporated in information systems.
- IT solutions for process control and measurement: consist in exception handling and process controlling, performance analysis and visualization.
- IT solutions for process improvement and innovation: consist in tools capable of adapting the processes according to the contextual changes and adopting the improved processes in an automatic manner.
- IT solutions for project and program management: represent tools that support the process owners in the BPM activities and they are very often decision support systems.

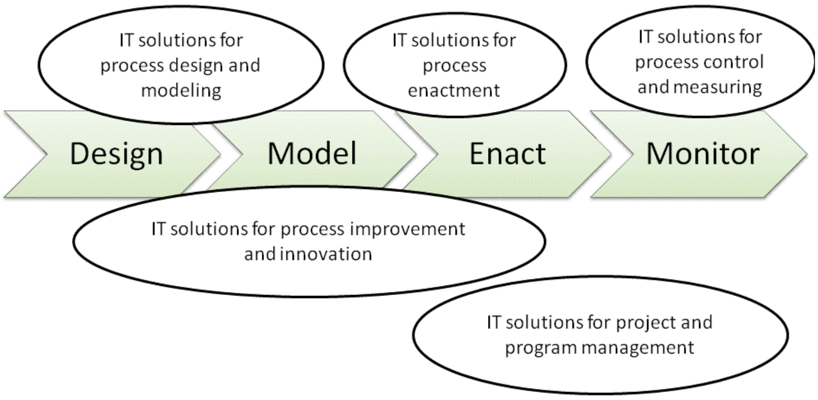


Figure 6. The integration of enterprise systems and business processes lifecycle

While business processes are integrated with the enterprise applications and their interaction is enabled through the e-business concept, information technology is the actual enabler of both intra-organization and inter-organization integration (Gelinas et al., 2004). The influence of the information technology with respect to business process – enterprise system integration is analyzed by Dumas et al. (2005) in connection with the trends in information systems development. While in the beginning, the trend was focused on developing hardware infrastructure and operating systems, the trend shifted to building generic enterprise applications afterwards, which were used by multiple departments such as document editors or database management systems. Further, an intensive development of domain specific information systems, such as accounting software or call centre software, was observed. Finally the focus switched on the development of tailor-made applications created for specific organizations. This period culminated with a shift from the application development to the application integration ("*assembling vs. programming*") (Dumas et al. 2005).
In parallel with the information technology trend, Dumas et al. (2005) highlight also a shift from the data-centered to process-centered information systems and another transition from the very specifically planned system design to redesign and organic growth as a result of fast-paced changes in the organizational environment. All these trends ultimately emerged in a new category of information systems known as process

aware information systems (PAIS) and defined as "*software system that manages and executes operational processes involving people, applications, and/or information sources on the basis of process models*".

Data-centered information systems, widespread initially, incorporated processes too. However because the process management was not placed in the foreground, it proved very cumbersome for organizations to adapt their information systems according to the occurred changes in an optimal manner. The main principle of PAIS is to separate the management of the processes from their implementation, in order to encourage a more efficient usage of the organization's resources and to provide support for an easier process redesign. The information systems that are process aware bring together the participants, information and technology, by organizing the work and resources, for achieving the established strategic objectives (Dumas et al., 2005).

3.1.3.1 An Evolutionary View of Process Aware Information Systems

The early ancestors of PAIS are the office information systems namely BDL (Hammer, Howe, Kruskal, & Wladawsky, 1977), SCOOP (Zisman, 1977), POISE (Croft & Lefkowitz, 1984), Officetalk-Zero (Ellis & Nutt, 1980), whose purpose was to automate the office functions such as document editing or communication. There were two research approaches, one relying on a procedural prescription of the tasks using Petri nets (Zisman, 1977; Holt, 1985; Hammer et al., 1977) and another being data-centered and focusing on the office's objects manipulation. The lifespan of these systems was rather short, the reasons being related to the context of that period. The technology was not enough developed, the modeling techniques and methods were in very early stages and the organizations were focused on very specific, individual tasks instead of organizational processes (Dumas et al., 2005).

Workflow technology, another influential factor of PAIS evolution, has been behind a large number of information systems in the last three decades (Russel, 2007). While these systems have had initially a specific scope – enabling office operations such as document processing and email support, they evolved in more complex support systems for organizations (Muehlen, 2004; Georgakopoulos, Hornick, & Sheth, 1995). In the overview conducted by Georgeakopoulos et al. (1995), different categories of workflows were identified including commercial transaction processing systems, commercial workflow management systems etc.

The next significant evolution in this direction was business process engineering research, initiated by Hammer and Champy (1993) and Davenport (1993), which focused on structuring the individual tasks of an organization in processes and on re-defining the role of IT from task support to intercommunication and coordination support. Starting from this critical period, the PAIS development and adoption grew substantially encouraged simultaneously by a parallel development of the business process modeling techniques and tools (Dumas et al., 2005). If the classical PAIS supported only process enactment and relied completely on workflow technology, nowadays there could be found a wide variety of tools and systems covering various process lifecycle phases (Table 4):

PAIS	**Examples**
Process-aware collaboration tools	CoWord, WebEx Meeting Center, IPMM, Caramba
Project management tools	AMS Realtime, Microsoft Project
Tracking tools	JobPro Central
Enterprise resource planning and customer relationship management systems	SAP, PeopleSoft
Case handling systems	FLOWer
Business process design and engineering tools	ARIS, Protos
Enterprise Application Integration systems	TIBCO ActiveEnterprise, Microsoft BizTalk
Extended Web integration servers	BEA WebLogic Integration, IBM Websphere MQ

Table 4. Examples of contemporary PAISs (Dumas et al., 2005)

3.1.3.2 Process Aware Information Systems Overview

Reichert and Weber (2012) propose an analysis of PAIS from different perspectives which provides insights into the alignment of the executable processes with the business goals.

The ***function perspective*** consists in the business functions which serve as input for defining the activities of the executable processes. Simple functions are translated in atomic activities while complex ones in sub-processes. Human or machine resources might be necessary in process execution, depending on the automation degree of the activities (Reichert & Weber, 2012).

The ***behavior perspective*** is represented by the dynamic aspects of the process, namely the control flow. In general, a control flow specification includes the order of activities and the pre-conditions for executing a specific activity or constraints in the case of the declarative-based approach. In the data-centered processes, the behavior is treated differently, depending on the object’s behaviors and object’s interactions. Object’s behavior describes the order of reading and writing the object’s attributes, the roles responsible for the operations and the consistency of the attributes’ values. Both object’s behavior and interactions depend on the evaluation of the objects states (composed of object’s attributes). The object states can have a validation role or they could be triggers for conditions (Reichert & Weber, 2012).

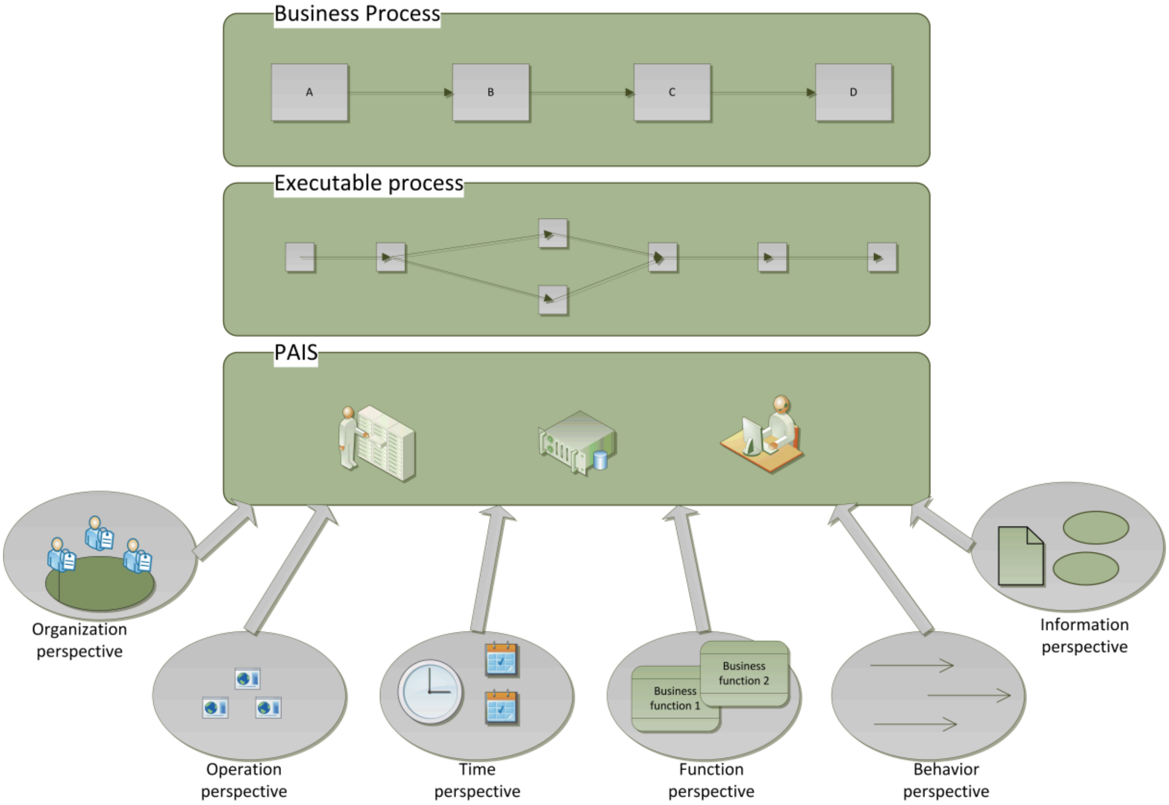


Figure 7. Process aware information systems overview

The **information perspective** consists of the data flow that describes which activities can manipulate or modify specific data objects, and the available set of data objects. Three types of data are identified in PAIS: application data, process-relevant data and process control data. In the data-driven processes the information perspective is integrated with the behavior perspective (Reichert & Weber, 2012).

The **organization perspective** consists of the management of organizational resources for human activities. It is an integration of the process model with the organizational model defined by actors, roles, organizational units and their relationships (Reichert & Weber, 2012).

The **operation perspective** deals with the implementation of activities and their mapping on the business functions, in the case of activity-centered process models. The implementation of activities does not necessarily need to be known; only their sequence is manipulated by PAIS, based on the input and output. Activities are considered as black-boxes inside which various application services are composed to accomplish the activity's function. In data-centered process models, the operation perspective is represented by user forms (Reichert & Weber, 2012).

The **time perspective** is the aspect of PAIS describing and handling temporal constraints. If an activity deadline is about to be reached, escalation or notification procedures are executed by the system to inform the process participant and the process owners about it (Reichert & Weber, 2012).

The analysis ground offered by these perspectives could also be used in the decisions regarding the development of PAIS. Two approaches are outlined by Dumas (2005): the development of process aware information systems from the scratch or the configuration of a generic system. In the first category, there are included those systems developed in house by organizations. They vary from hard coded process execution systems to process platforms offering support for the whole process lifecycle (Reichert & Weber, 2012). Even if they represent an integrated solution, process platforms imply significant investments in the beginning while they may prove not scalable for keeping the pace with the changing organizational requirements and customer needs. Generic process aware information systems are not developed for a specific domain or organization, and they rely on neutral technologies such as workflows.

Process aware information systems incorporate process models implicitly or explicitly (Figure 7). This represents a comprehensive method of communication between the business side – business managers, business analysts, and the IT side – IT architects, system administrators and developers. Moreover, being a model-driven information system, PAIS could integrate the process changes in a top-down manner by changing the model, offering in this way a better management support. The process model integration also permits automatic enactment of the repetitive, less human-intensive processes (Dumas et al., 2005). However, despite the advantages associated with PAIS, these systems had not the expected adoption rate because of the lack of support for flexible process enactment (Weber et al., 2009).

3.2 Intentionality Exploration

As discussed in the previous section, people are one of the core components of Business Process Management (Rosemann & Vom Brocke, 2010) and one of the key perspectives of process aware information systems (Reichert & Weber, 2012). Moreover, it has been stated that a complete translation of business processes in operational automatic processes is very often impossible as human involvement during enactment is most of the time mandatory. The degree of human involvement is even higher and dominant in the case of flexible, knowledge intensive processes.

3.2.1 An Overview of Humans and Processes

Rosemann and Vom Brocke (2010) discuss various aspects of humans in business process management and outline the focus areas for a successful adoption. **Process skills and expertise** become mandatory assets especially for process owners and all the individuals holding process management positions. They consist of a deep understanding of the business process requirements, extracted from the customers and stakeholders needs. **Process education and learning** focus on the dissemination of the process knowledge among individuals within the organization through trainings and certification programs. **Process collaboration and communication** refer to how the individuals within the organization work together during the process lifecycle. It also refers to the mechanisms for discovery and sharing the process knowledge, in both tacit and explicit forms (Rosemann & Vom Brocke, 2010).

While the former analysis highlights the general aspects of human involvement in business process management, Gelinas et al. (2004) define three process-related organizational roles: process designer, process participant and process owner. The **process designer** has a thorough knowledge of the business processes, information systems and development methods that he uses during the process definition and implementation. The **process participant** enacts processes by interacting directly with the information systems. The **process owner** is responsible for controlling the business processes by rigorously evaluating their effectiveness and efficiency.

The primacy of humans in process aware information has been highly acknowledged (Ellis & Kim, 2007; Lee et al., 2011; Aalst, 2011). Ellis and Kim (2007) outline the importance of incorporating the human perspective into PAIS as organizational processes have been becoming more complex and highly dynamic. They analyze the collective human endeavor when interacting with PAIS considering factors such as the organizational structures, roles, and the social and cultural settings. Lee et al. (2011) discuss the support for human interaction and innovative involvement in knowledge intensive processes, and propose a solution based on condition-based process patterns for facilitating this.

Consequently, the nature of activities during the process lifecycle requiring a high degree of human involvement is related to reasoning, decision-making and learning by self experience or collaboration. Humans are able to analyze the process environment, deliberate or create new solutions when none of the existing ones fulfills the contextual requirements (Pohl, Dmges, & Jarke, 1996; Yu, 1997; Yu, 2009). Moreover, humans should not only have a reactive behavior, such as making a decision when requested by the system, but also act proactively when influenced by the contextual information. As a result of a deep understanding of processes, humans are able to criticize process definition leading to potential process improvements (Pohl et al., 1994; Kueng & Kawalek, 1997). All these inherent human features can be aggregated in two concepts: agency and intentionality which are discussed further.

3.2.2 Intentionality and Agency

Human behavior is intentional by nature and this topic has been thoroughly discussed in philosophy (Bratman, 1987) and further adopted by the artificial intelligence community (Cohen & Levesque, 1990; Lacey, Hexmoor, & Beavers, 2002). Generally, when discussing about intentionality, the term "agent" is used to refer to humans, and to any entity able to reason over the context and inner knowledge, make decisions, act according to its goals and learn from experience. Extending this idea, the agency can be then considered a characteristic of any entity that is described by all these agent-related properties.

Bratman (1987) analyzes the rationality of an agent through a model representing its mental state, composed of beliefs – what the agent knows about the world, desires – what the agent might want to accomplish but not necessarily to act upon it, and intentions – what the agent decided to accomplish and to act towards it. The process of intention formulation is called procedural reasoning and it has two phases: deliberation when the agent adopts some intentions; means-ends reasoning when the agent acts in order to complete its intentions. Hence, the intentions are not studied only in relation to agent deliberation but also considering the agent's acts, defining its behavior. Considering all these aspects, agent rationality implies the agent behaves in a certain manner under the motivation of achieving some intentions and these intentions should be under rational norms when forming, reconsidering and revising them.

While Bratman makes a clear distinction between intentions and beliefs, Fishbein and Ajzen (1975) sustain that an intention could be considered a belief itself. They define the belief as information that agent has about an object where an object is a generic term for naming any entity belonging to the agent's environment, either active or passive. Moreover, intentions or desires, described in these terms, are the attitude of the agent to changing the objects' states. Therefore, an intention becomes a belief for which the object is the agent itself while the agent's behavior is an attribute. Going further in explaining the intention, Fishbein and Ajzen (1975) define four basic elements: the agent's behavior consisting of the set of observable acts, the objects targeted by the behavior, the situation where and the time when the behavior is applied.

An intention could be the result of two different attitudes of the agent. So far, the intention has been presented as a driver of the action "*intending to do something*". However, the term "*intention*" is also used when the agent does something intentionally (Bratman, 1987). In the later case, the consequences of achieving a certain intention are highlighted instead of the intention itself. The side effects are created intentionally but they do not represent the agent's goal. Two types of intentions are identified by Cohen and Levesque (1990), namely future-directed and present-directed intentions. In general the immediate, present-directed intentions are not very common but rather they are transformed in plans for future accomplishment.

Even if the term of intention is often used as synonym for goal, objective or purpose, it differs conceptually. An intention involves commitment of the agent to actions and bounded rationality (Bratman 1987). The commitment is generally reasoning-centered and includes the following regulations: an intention is not revised and reconsidered except for the situation where the agent becomes aware of new and relevant information; once an intention is adopted, the agent must reason about the means and methods to achieve it; the agent must pro-actively exclude from his future reasoning any options that do not comply with the adopted intention (Bratman, 1987). In general, humans have changing desires and there is a continuous flux of information that can alter their beliefs. Nevertheless, humans should not reconsider the intentions every time when new information is known or the desires change, but rather to commit in a limited wait to a specific intention, and follow the plans to its satisfaction. If an intention becomes non-feasible – meaning cannot be achieved through any strategy, the agent quits the commitment and formulates another intention (Cohen & Levesque, 1990). Building around the concept of intention persistency, Cohen and Levesque (1990) proposed a theoretical framework entailing the principles of intention commitment. This formed the basis for creating artificial intelligent agents and a philosophical formalization of the human action theory.

Furthermore, intention reasoning has other advantages too, as outlined by Bratman (1987). Intentions are problem oriented, thus leading the agent to a thorough understanding of the context, guiding him through various achievement strategies. Moreover, the agent must understand that choosing an intention imposes constraints on the possible future intentions. While desires can be contradictory, an agent will only formulate and consider future intentions which are consistent with the intentions he's committed to in the present.

Finally, intentions enable the agent to learn from the past by tracking the intermediary strategies and their impact on the intention's achievement (Bratman 1987, Cohen & Levesque 1990). Agents keep track of all the attempts for achieving an intention, and, in the case of failure, the next plan is created considering the agent's previous experience.

3.2.3 Intentionality in Information Systems

While the difference between "intention" and "goal" is very relevant when discussing about agents, these terms have been used interchangeably in the context of non-autonomous entities such as most of the information systems. In addition to artificial intelligence domain, where intentionality has been thoroughly discussed and applied in the development of agents, this topic has been a source of inspiration for various research areas in information systems as requirements engineering, enterprise engineering and data mining.

Requirements engineering is the sub-domain of software engineering consisting of the following activities: requirements elicitation, requirements modeling and requirements analysis. Goal oriented requirements engineering (GORE) implies a different approach of treating the requirements, focusing on what the system needs to do (the intentional view), in addition to how to do it (the prescriptive view). By focusing on goals, it positions the rationale of software system development in the foreground (Maria, Werneck, Padua, Oliveira, Cesar, Leite, & São, 2009).

In a GORE approach, the organization's functional and non-functional goals are transformed in requirements. Instead of modeling the requirements, GORE focuses on modeling the goals which trigger the existence of the requirements. Several reasons behind a goal centered meta-model for requirements engineering were invoked (Yu, 1997). The non-functional goal, focusing on meta-models, has limited scope offering support only for the software specifications but without considering the composite systems including the environment analysis. Non-functional goals are poorly captured and there is a lack of support for the exploration of alternatives. Different frameworks have been proposed in this respect, two of which, i* and KAOS, are discussed further. These frameworks are top-down/bottom-up structures or AND/OR graphs.

i* framework was created by Eric Yu (1997, 2009) and its principles are built on two concepts: intentional actors and intentional dependencies among actors. Two types of goals are introduced in i*: concrete goals which can be satisfied, and softgoals which can be satisficied denoting a "*lack of precision in the perception of satisfaction*" (Yu, 1997; Maria et al., 2009).

The central element of i* (Figure 8) is the actor which depends on other actors in achieving a goal or a softgoal. Behind formulating a goal satisfaction plan, the actor depends on the others actors for providing the necessary resources and for performing certain tasks. Consequently, an actor is characterized as strategic, trying to satisfy its intentions while dealing with its opportunities and vulnerabilities (Maria et al., 2009). Actors are a general category including agents (the concrete, active entities), roles (the set of agent's responsibilities) and positions (the roles played by the agents). According to Yu (1997, 2009), the actors are intentional so they have beliefs, aptitudes and commitments. Generally, the i* framework focuses on the social environment by incorporating the actor's autonomy (the behavior cannot be fully controllable or knowable), intentionality (the behavior is not random but rather lead by intents), sociality (the actor depends on social relationships in reaching his goals), rationality (the reasons behind the behavior can be understood), strategic reflectivity (the actor reflects on the relations that can position him strategically with regard to his goals).

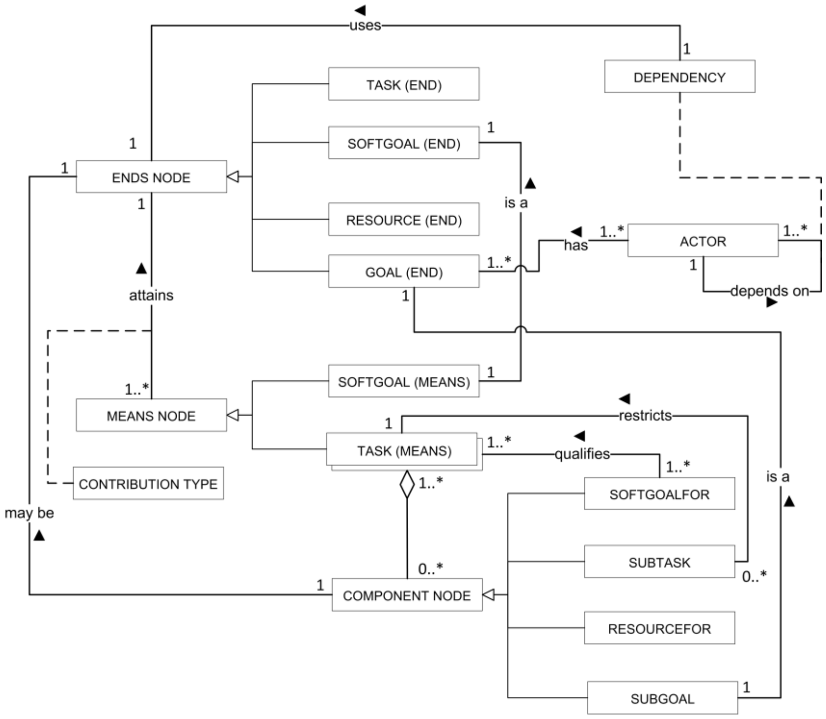


Figure 8. i* meta-model (Maria et al., 2009)

While i* focuses on capturing intentionality behind the agent's behavior by modeling the humans as entities seeking for goal satisfaction, KAOS focuses directly on goals. KAOS is characterized as a multi-paradigm framework which exposes different degrees of expression: semi-formal for goal modeling, qualitative for alternatives selection and formal for reasoning. It is composed of an external representation using concepts, attributes and relationships, and an internal layer capturing the temporality (Van Lamsweerde & Letier, 2004).

In the KAOS framework (Figure 9), goals can be approached through requisites. The requisites are further classified in requirements referring to the software products, and assumptions referring to the behavior of the external agents. The objects used in the goal achievement represent the structural perspective of the framework and they can be active or passive. The goal model is represented as an AND/OR graph where a goal is recursively reduced to sub-goals. Both functional goals (services) and non-functional goals (qualities of the services) are considered (Van Lamsweerde & Letier, 2004; Maria et al., 2009). The process of building the goal model includes several steps: goal identification, goal formalization, the detection and handling of goal-related conflicts, goal refining, the allocation of responsibilities to agents, and the detection and handling of operations. The goals are classified in relation to the agents as follows: satisfaction goals are what the agents want to achieve, information goals have the purpose to inform agents and accuracy goals captures the non-functional aspects (Van Lamsweerde & Letier, 2004).

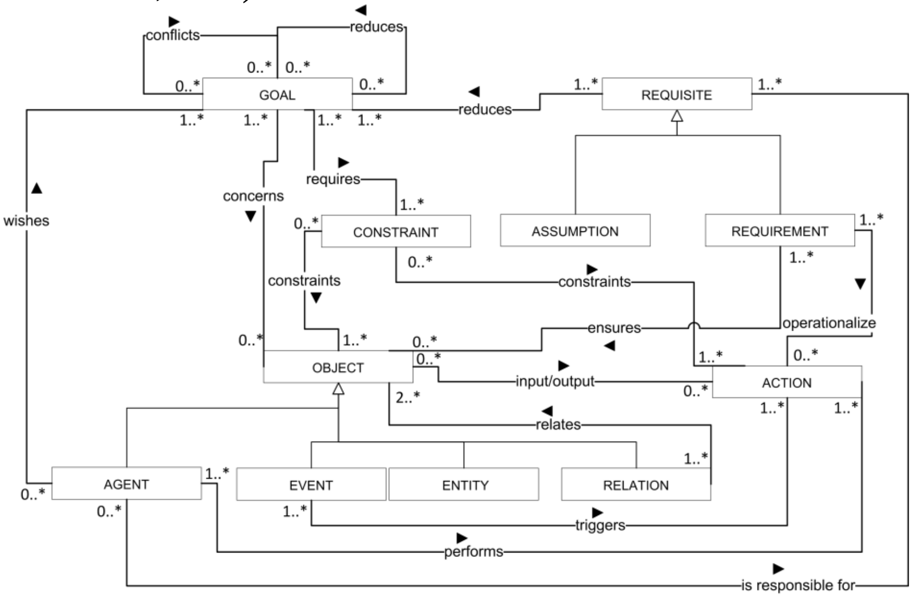


Figure 9. KAOS meta-model (Maria et al., 2009)

Enterprise engineering, especially business process engineering is another domain where goal-based approaches have been used. Various methods of analyzing and designing a process (Kueng & Kawalek, 1997; Yu, 2009; Lee et al., 2011) are goal-centered especially due to the usage of the same term in the business vocabulary. The intention-based definition of processes can lead to a better alignment of the activities with what is meant to be achieved. The best alternative, depending on the context could be identified using an intentional facet of the process (Kueng & Kawalek, 1997, Lee et al., 2011). Furthermore, the validation of the process conceptual design could be done in advance, before its actual implementation and execution.

Except for its applicability in requirements engineering, the i* framework is also used for modeling business processes and enterprise architecture (Yu, 2009). It links the business goals to business processes, outlining the dependencies of stakeholders (customers, employees, regulators, investors) and the goals impact on the organizational environment. Yu (2009) argued that this approach could represent a better basis for business innovation and technology's adoption.

Lee (1993) proposes a goal-centered method for process analysis and design, offering support for the systematic identification and implementation of goals, the identification of the non-functional goals and the achievement alternatives exploration. The alignment of business goals to business processes, implicitly of sub-goals with sub-processes, is ensured in this way. The method consists of several steps: enumerate goals, relate goals by identifying the super-goals and sub-goals, check the goals completeness, identify non-functional processes and explore alternatives (Lee, 1993).

Kueng and Kawalek (1997) propose another method of modeling business processes with four levels of representation: goal, activity, roles and objects. The authors claim that a goal-based modeling technique would benefit the stakeholders by structuring and evaluating the process design, evaluating the operational process and providing analysis support for the broader implications. Thereby, challenges such as capturing the process participant goals, evaluating goals compatibility, resolving inconsistencies and aligning business goals with the processes could be overcome (Kueng & Kawalek, 1997).

Koliadis and Ghose (2006) link two popular modeling approaches, KAOS and BPMN (business process modeling notation), creating GoalBPM. It is a method for modeling the stakeholder goals, their dependencies and their evolution, traced in connection to the business processes. They make an explicit mapping of the goals to the process activities through traceability and satisfaction links. Traceability link is a weak relationship between an activity and a goal meaning the execution of that activity does not necessarily lead to the achievement of the goal. Satisfaction link marks a strong relationship translated into: the execution of that activity is a mandatory step for satisfying the goal (Koliadis & Ghose, 2006).

A newer approach for business goals modeling is based on ontologies (Markov, & Kowalkiewicz, 2008). The ontology framework contains functions which define pair relationships of the following concepts: Business Goal, Business Function, Business Role and Process Resource. Goals are modeled using AND/OR relationships at different levels: strategic and operational. Strategic goals are long term and are achieved through the satisfaction of the operational goals. For exploring various alternatives, measurability is introduced as a goal property estimating the degree of goal satisfaction (Markov, & Kowalkiewicz, 2008).

Finally, another area where intentionality plays a significant role is data mining. Understanding user behavior for improving both the information systems and the quality of services has been a highly researched topic in the last years. Chen et al. (2002) proposed a method for identifying user intentions in Web applications to predict their future behavior. They focused only on predicting low level intentions (action intentions) by using a Naïve Bayesian classifier. Furthermore, Song and Diederich (2010) extracted the users intentions from text messages by applying text mining techniques inspired from the Speech act theory.

3.2.4 Intentional Process Modeling with MAP

Modeling is necessary for representing the professionals' knowledge in a systematic formal manner which makes it ready for understanding and usage (Yu, 2009). As outlined in the previous, section, the goal-oriented perspective has the potential to further improve the utility of these models as it provides a representation closer to human reasoning. The intentional meta-model chosen for modeling operational processes in the current research method is MAP. MAP has been successfully used and validated in several domains as: requirements engineering (Prakash & Rolland, 2006; Zoukar & Salinesi, 2004), method engineering (Kornyshova, Deneckère, & Salinesi, 2007; Kornyshova, Deneckère, & Rolland, 2011), and process modeling (Ralyte, 2012). Additionally, an enhanced version of MAP with qualitative criteria for exploration, inspired from the theory of graphs, was proposed by Deneckere, Kornyshova and Rolland (2009).

MAP (Rolland et al., 1997) is a labeled directed graph with intentions as nodes and strategies as edges. The intention incorporates the process rational (what and why is aimed to be accomplished) while the strategy represents an alternative of achieving an intention. A MAP model has two default intentions "Start" and "Stop" corresponding to the beginning and the end of the process. A section in a MAP is defined by the triplet <source intention, strategy, target intention> that captures a step in the intentional process (Figure 10).

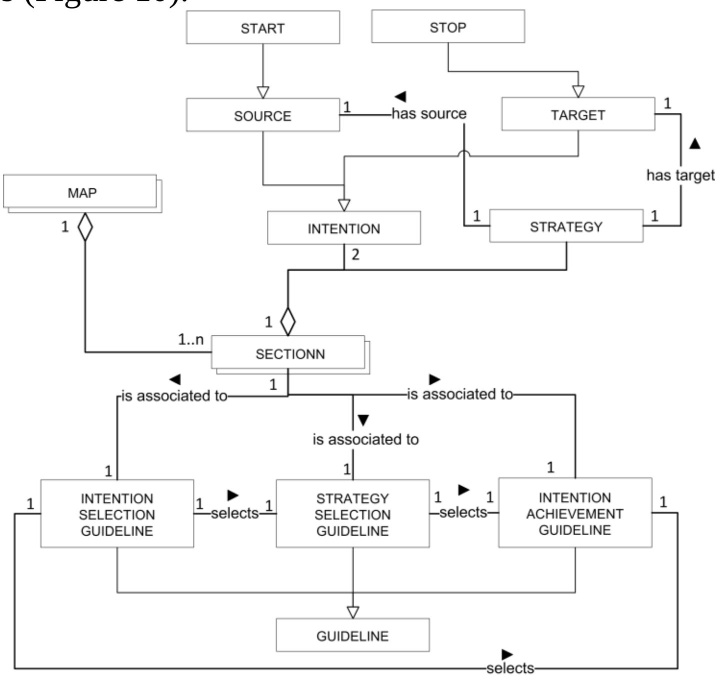


Figure 10. MAP meta-model (Rolland et al., 1997)

Multiple ways of satisfying an intention are represented in a MAP model through multiple edges that could link two nodes. Consequently, this model provides a synthetic and comprehensive view of process variability. Variations are revealed twofold: by the gradual exploration of the directed graph from the top of the MAP to the terminal nodes, and by the alternative paths available at a given level. In the exploration, there are several types of guidelines (Rolland et al., 1997):

- Intention Selection Guideline for identifying all the intentions that can be reached from a given node in the graph;

- Strategy Selection Guidelines for identifying all the strategies that can lead to the achievement of a given intention;
- Intention Achievement Guidelines for specifying the achievement strategy for a given, specific situation

A guideline has a signature in the form <sit, I>, where "*I*" stands for the intention that needs to be achieved in the situation "*sit*" (Rolland et al., 1997). The signature of the intention is more complex, being developed according to the Fillmore's theory of case grammar.

3.3 Process Mining

Organizations want to understand and learn how their business processes are executed in practice. As described in the first section, business processes are translated in executable processes by means of models. These models are further used for the specification, configuration and implementation of the process aware information systems. Whereas these systems capture large amounts of information, a new opportunity upraised for organizations – to gain knowledge about their business processes in a bottom-up manner by using the recorded event logs (Figure 11). An event log contains detailed information about the process activities being executed.

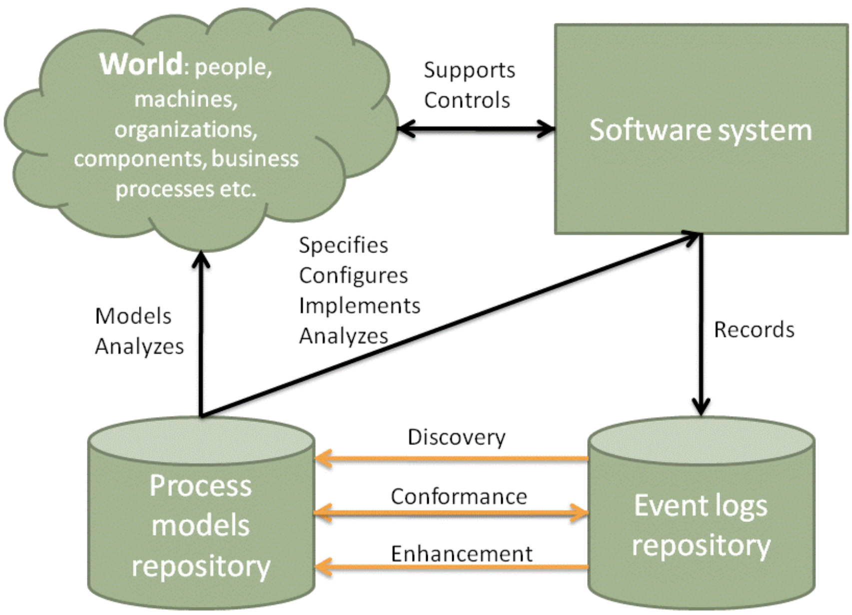


Figure 11. Process mining overview (Aalst, 2011)

As a result of these favorable circumstances, process mining has emerged as a standalone discipline, found at the intersection of data mining and process modeling and analysis. The goal of process mining is to discover, monitor and improve the executable processes based on the knowledge extracted from event logs (Aalst, 2011). The predecessor of process mining is data mining which refers to transforming large data into knowledge, by identifying patterns relevant for the organization's behavior and structure. The approaches used in data mining have been adapted for process mining, and further extended with techniques borrowed from computer science areas (Tiwari, Turner, & Majeed, 2008).

The pioneers of process mining are Agrawal, Gunopulos and Leymann (1998) who focused on creating process flow graphs from workflow systems' logs, and Cook and Wolf (1998) who synthesized software process models from event logs. Cook and Wolf (1998) used three methods from statistics for analyzing the tasks: RNet, Markov and KTail. The difference between them is the input for defining the current state of a process. In RNet a process state is defined based on the past behavior, in the Markovian approach based on both past and future behavior while in KTail only the future behavior is analyzed for defining the process state (Cook & Wolf, 1998). Moreover, Aalst (2011) identifies two types of process model categories that have been employed in process mining over time: graph-based models (e.g. Agrawal et al. (1998) used directed graph for modeling activities as nodes and edges as activity flows) and block-based models. Nevertheless, Aguilar-Saven (2004) makes an additional and specific distinction between the graph-based models and the net-based models (e.g. Petri nets).

There are several types of process mining depending on their scope: discovery, conformance and enhancement (Aalst, 2011).

- **Discovery process mining** when a process model is extracted from event logs without having any external information.
- **Conformance process mining** when a process model is known in advance and the knowledge extracted from the event logs is used for detecting, locating and explaining process deviations while offering support for measuring their impact.
- **Enhancement** when the business process is changed based on the process knowledge discovered in the event logs. Two types of enhancement are identified: *repair* when the business process is changed to reflect the executable process and *extension* when the business process is extended with a new perspective identified in the executable process.

Process mining can be performed from different perspectives, depending on the discriminator event data used as input to the process mining technique (Table 5). The resulting process model is also aligned with the mined perspective.

Process mining perspective	**Description**
Control flow perspective	Focuses on capturing the order of activities and all possible paths of a process model

Organizational flow perspective	Focuses on the identification of resources such as actors (people, roles, units, systems) and the relations among them (social networks)
Case perspective	Focuses on identifying characteristics of the cases such as the corresponding path in the process model, originator, and data values. Decision mining is a concrete example of case perspective and its goal is to discover the rules influencing the choices during enactment.
Time perspective	Focuses on capturing the timing and frequency of events based on timestamps. This knowledge is used for the identification of bottlenecks or the degree of resource utilization.

Table 5. The process mining perspectives (Aalst, 2011)

Van der Aalst (2011) outlines several challenges regarding process mining. The first challenge is to prepare the event logs. Even if events are recorded and stored, these data is rather scattered over different tables and databases and the extraction requires first the identification of the data relevant for process mining, and, second, the further localization of this data in databases/tables. Additionally, the data needs to be structured in a proper format and filtered out.

The second challenge is to create a process mining technique meaning an algorithm which having as input event logs produces a model which captures a process perspective. Frequently, these techniques are based on the artificial intelligence discipline, namely machine learning, and some of the most common are genetic algorithms, neural networks, Markov chains and clustering (Tiwari et al., 2004). Finally, these two challenges can become even more complex because of issues interfering with a correct process model discovery (Aalst, 2011):

- Noise when event logs contain events that are rather exceptions, not representative for the standard behavior or structure of the process
- Incompleteness when the event log data is not enough for describing the process

In the remaining of this section a concrete example of discovering a process model from event logs inspired from the book "*Process Mining: Discovery, Conformance and Enhancement of Business Processes*" (Aalst, 2011) is given. However before moving to its presentation, the terminology used in process mining needs clarification. A process consists of cases, also named traces. A case consists of an ordered set of events such that each event appears only once. Events can have attributes such as the activity name, the timestamp or the resource (Figure 12).

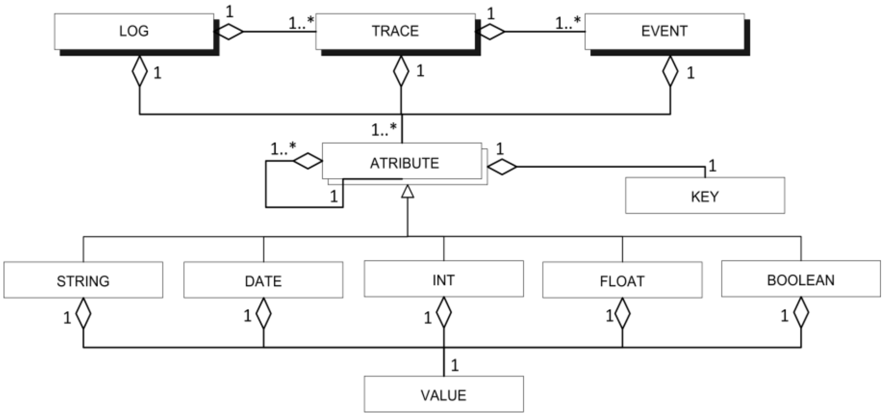


Figure 12. Process log meta-model (Aalst, 2011)

In Table 6, an event log is presented. Each entry of the table represents an event which is associated to a specific case. In this situation, the event is identified by four properties: the event id which is a unique identifier, the timestamp when the event is recorded, the process activity and resource associated to the event. The case id and the timestamp are the minimum information for mining the control flow perspective of this process.

Case id	Event id	Properties		
		Timestamp	Activity	Resource
1	35654423	30-12-2010:11.02	Register request	Pete
	35654424	31-12-2010:10.06	Examine thoroughly	Sue
	35654425	05-01-2011:15.12	Check ticket	Mike
	35654426	06-01-2011:11.18	Decide	Sara
	35654427	07-01-2011:14.24	Reject request	Pete
2	35654483	30-12-2010:11.32	Register request	Mike
	35654484	30-12-2010:12.12	Examine casually	Mile
	35654485	30-12-2010:14.16	Check ticket	Pete
	35654486	05-01-2011:11.22	Decide	Sara
	35654487	08-01-2011:12.05	Pay compensation	Ellen

Table 6. The example of an event log

For the sake of the example, assuming that the data contained in this event log is complete and it does not contain noise, the discovered control-flow process model is:

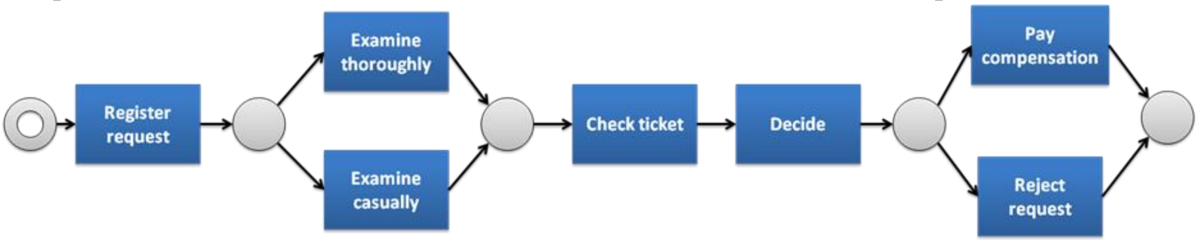


Figure 13. The mined process model based on the example

The algorithm identifies that the process start always with “Register request”, then two possible activities can follow “Examine thoroughly” or ”Examine casually”. Further, the process paths are combined in a single flow with the activities “Check ticket” and “Decide”. Finally the model ends with a decision point separating the process flow in two paths corresponding to “Pay compensation” and “Reject request”.

Having the discovered process, other event logs or cases can be used for checking the conformance, and for its enhancement. If discrepancies are identified when analyzing event logs this could lead to the following conclusions: the model does not reflect the reality anymore or these are deviations from the wanted model. If the first conclusion is reached then the current model can evolve in an improved one. If the second conclusion is reached, further investigation is necessary to discover the causes of this behavior, such as an inexperienced process participant or a very particular context.

3.4 Processes Enactment Support

In the context of processes, the knowledge extracted through process mining could influence the implementation of and serve as the input to recommendation solutions (Schonenberg et al., 2008; Aalst, 2008, Petrusel & Stanciu, 2012; Petrusel, 2012).

The process guidance is needed to support the humans or systems in making decisions over different phases of the process lifecycle: design, configuration, implementation and enactment. Van der Aalst (2008) discusses thoroughly this topic mapped on the following types of decisions: design-time, configuration-time, control-time and run-time. Design-time decisions focus on defining the model, configuration-time decisions focus on customizing the model, control-time decisions focus on modifying all the process instances at run-time, and run-time decisions focus on the enactment of a particular process instance. Conformance checking can be used to support the process re-design as it allows the identification of the existing problems and the evolved process versions. Furthermore, the discovered process models can provide feedback regarding the usability of the process configurations (for example, by identifying features that are never used) or can conduct the migration of process instances to the model emerged from the real world execution. Finally, the mined cases could be used for building a recommendation system aimed to help the process participant during the process enactment based on the past experience and the current case (Aalst, 2008).

Petrusel and Stanciu (2012) promote the idea that the decision making process and its results are concretized by a set of activities. These activities can be identified in the event logs. The authors argue that, because of the fuzziness of the decision making process, it is more suitable to offer recommendations based on process mining instead of following a top down approach based on the process definition. Two algorithms to support the decision making process are proposed: a Greedy approach when the activity that was the most frequently executed by other process participants for a specific situation is recommended, and a second approach based on the A* path finding algorithm (Petrusel, 2012).

Schonenberg et al. (2008) propose a declarative-based solution for offering recommendations considering the process mining knowledge and the process participant case. A case is considered to be the partial process execution and the corresponding non-functional goals. The recommendation service returns several possible steps to be followed and a list of attributes such as: the probability to achieve the non-functional goal if a certain path is followed and the number of previous cases used as input knowledge for the prediction.

Driven by several pre-identified requirements of an effective guidance in ad-hoc workflows, Dorn, Burkhart, Werth and Dustdar (2010) propose a recommendation solution adapted to the user's behavior while considering the best practices of the crowd. The prototype is built to support the work habits of each individual process participant, to explore the crowd process knowledge and to offer support for the automatic adaption to changes. The last requirement is based on the claim that the flexible processes need to evolve over time, to reflect the behavior of the process participants in real-life. The recommendation is incorporated in every step of the process execution (Dorn et al., 2010).

Even if the above mentioned solutions represent an important step in this direction, they are limited in their usage when dealing with the lack of process participant's experience or knowledge-intensive processes. In order to support decision making with helpful and meaningful information, the recommendation of a possible next intention instead of an activity, and the recommendation of possible strategies in achieving that intention could prove more effective. Until now, only one research work on process support based on the intentional process model is known. Khodabandelou, Hug, Deneckere, and Salinesi (2013) propose an approach for discovering the intentions behind process participant's activities using Hidden Markov Models. The probabilistic model is trained for the prediction of the next intention and its corresponding set of activities.

4 AGENT-CENTERED ANALYSIS IN FLEXIBLE PAIS

This chapter includes a detailed presentation and discussion of the first research sub-question:

How does the implementation of flexible processes in process aware information systems impact the process participants?

A flexible PAIS is a process aware information system that implements flexible business processes. In this context, we define flexibility as the ability of the information system to support process changes as result of the changes in the external and internal environment. In the following discussion, we consider an agent to be a human or an artificial entity, able to reason over context and inner knowledge, make decisions, act according to its goals and learn from experience. Two roles in flexible PAIS, process administrator and process participant, are analyzed in the following section, followed by an analysis of their agency ability.

4.1 Literature Review on Flexible PAIS

The research sub-question is answered in two steps. First an analysis of how flexible processes are implemented in PAIS is conducted. Then, the implications for agents of implementing flexible processes are outlined. Based on this the initial design decisions translated in artifact's requirements or objectives are formulated.

The research method used for answering the first research question was a literature review. As mentioned before, the process proposed by Levy and Ellis (2006) was followed. Through Omega and Google Scholar search engines, several databases were interrogated such as IEEE, Elsevier Science Direct, ACM Digital Library, Ebsco, JStor and Springer Link. Furthermore, the articles presented to the "*Business Process Management*" and "*Business Process Modeling, Development, and Support*" conferences were scanned too. The minimum accepted publication date was established to year 2000 inclusive. The groups of keywords provided to the search engines were "*flexible process/workflow*" (search query 1) and "*flexible business process*" (search query 2).

After the first search, 212 publications were found for search query 1 and 126 for search query 2. All the documents were scanned, first based on the abstract, and then based on the content. The final selection contained 26 documents. The decision criteria for the selection of the documents for the final analysis were:

- The articles must contain an implicit or explicit discussion of flexible processes in the context of process aware information system.
- The articles related to implementing process flexibility based on service-oriented computing (service composition, choreography and orchestration) and to business to business collaboration were not considered. The reason is that when dealing with distributed

architectures, exceeding the organization's boundaries, it is harder to monitor the users' activities and meaningfully aggregate them through process mining.

Of the 26 final articles, 20 were primary sources of data and 6 secondary. The primary sources were articles mostly delivered at conferences and several published in journals, containing the original research and their results. These articles presented new solutions regarding one or more phases of the flexible business process lifecycle either as artifacts (tools, algorithms) or methods.

The analysis of these articles started with the scanning of the secondary sources of data in order to identify the categories of process flexibility already defined by the business process management community. These categories were proposed by different authors, though their naming was not consistent even if they were referring to the same thing. Four categories of process flexibility were extracted from the secondary sources of data under the following names: variability, adaptation, under-specification and evolution. Besides these, I finally introduced the fifth category, data-driven approach, which was deducted from multiple articles without being officially recognized.

The next step in the analysis was the classification of the primary sources of data on these categories. One article could belong to one or more categories as I noticed that the process flexibility classes overlapped to some extent. However, the central element of the analysis was the agent and his activities which depended on the process flexibility category. Therefore, the key activities were extracted and further mapped on the process lifecycle phases based on the article's text. Finally, I created Table 6 with all this extracted information, which was further used in the identification of the agent's needs, dictated by the implementation of flexible processes.

4.2 Flexibility Implications

Based on a literature study, the following categories of process flexibility are identified: **variability**, **adaptation**, **under-specification**, **evolution** and **data-driven approach** (Schonenberg et al., 2008; Weber et al., 2009, Reichert & Weber 2012; Burkhart & Loos, 2010). These categories are not mutually exclusive and many of the methods and technologies underlying the process aware information systems provide multiple types of flexibility (Weber et al. 2009, Burkhart & Loos, 2010). However, specific implications are associated with each of these categories and they are going to be discussed in this section.

The analysis of the agent roles is discussed in relation to the process lifecycle phases: design, configuration and enactment. A traditional approach of implementing flexible processes requires the process administrator to identify the flexibility needs and options beforehand, during the build-time. However, recent approaches (Aalst, Pesic,

& Schonenberg, 2009; Dadam & Reichert, 2009) extend the idea of flexibility with a necessity of supporting it during run-time. Consequently, some of the process administrator responsibilities such as modeling are adopted by the process participant too.

4.2.1 Process Variability

Process variability (Schonenberg et al., 2008; Weber et al., 2009; Hallerbach, Bauer, & Reichert, 2010; Lee at al. 2011; Reichert & Weber, 2012) channels the main effort on the process design phase. The responsibilities of the process administrator during the process design phase include the creation of a reference model which could contain multiple execution paths, adjustment points or configurable nodes. Furthermore, the process administrator is also responsible for the creation of a process component repository. Depending on the definition of the reference model, various process components must be created and stored such as activities for each configurable node, prescribed change options for each adjustment point or process variant schemas. During the configuration phase, the process administrator analyzes the process context, and chooses or creates a process variant. During the enactment phase, the process participant is involved in making decisions. The decisions focus either on the process structure as selecting a process path in a conditional branch or on the process behavior whether to iterate an activity, to execute multiple activities in parallel or to cancel an activity.

Another approach for process variability involves the use of patterns (Russell, Hofstede, & Edmond, 2004; Weber, Rinderle, & Reichert, 2008). Russel et al. (2004) created a list of forty three workflow control-flow patterns which were identified in the existing workflow systems and in multiple process models at that time. Weber et al. (2008) complemented this list with other twenty five patterns focusing on process change and change support features. The change patterns are grouped in the following categories: adaptation patterns (modify the structure of the process by using high-level operations such as insert process fragment, delete process fragment, move process fragment and update condition), and patterns for changes in predefined regions (add unspecified parts to the process definition that will be specified during run-time through operations such as late selection of the process fragments, late modeling of the process fragments, late composition of the process fragments). The change support features are defined for ensuring the correctness and consistency when applying a specific pattern in practice. They are focused on process schema's evolution and version control, traceability and analysis of changes, and access control (Weber et al., 2007). A similar approach using workflow and interaction patterns for collaborative, innovative and human-centered processes were reported by Lee et al. (2011).

An approach for capturing process variability using MAP model was also proposed (Bentellis & Boufaida, 2009). During the design phase the process administrators and the relevant stakeholders analyze the process by specifying its objectives and sub-objectives. Further, the process and its objectives are modeled using MAP. In the end, the model is translated in an executable-ready version using a business process execution language (Bentellis & Boufaida, 2009).

A challenge regarding the process variability is the management of the process variants. Either the variants are represented as separate models or as conditional branches in a super-model, the maintenance of the repository is time consuming and error-prone (Hallerbach, Bauer, & Reichert, 2010; Lu, Sadiq, & Governatori, 2009). Hallerbach et al. (2010) proposed a tool (Provop) for defining and managing the variants. The variant definition starts from a reference model containing adjustment points to which a set of change operations is applied. The management of the variants focuses on the evolution of the process family, the constraints regarding the configuration of the variants and the correctness of the obtained models (Hallerbach et al., 2010). Lu et al. (2009) presented a solution for the discovery of the most suitable variants from the repository according to a set of specific requirements. Process variants were checked for similarity based on quantitative measures (Lu et al., 2009).

4.2.2 Process Under-specification

Process under-specification (Pesic, Schonenberg, & Aalst, 2007; Schonenberg et al., 2008; Weber et al., 2009; Reichert & Weber 2012) consists in a partial definition of the process model at build-time which is completed at run-time. Technically speaking, it is reliant on the mechanism of late binding. During the design phase, the process administrator creates the base process model which contains placeholders and defines the repository with process fragments and process activities. The process participant will complete the definition of the process model during the enactment phase by either late selection or late modeling mechanisms. Process under-specification could also impose a complete process modeling during the run-time as is the case with the declarative programming paradigm (Pesic et al., 2007; Aalst et al., 2009).

Lu et al. (2009) ensure the under-specification through loosely coupled activities which have attached a plan, a list of conditions for activating the plan and possible constraints. The process administrator defines a process template at build-time containing a prescribed part and a constrained, unspecified part. Then, he or other agents are responsible for the correct concretization of the process model before the execution (Lu et al., 2009). A similar approach aimed to support dynamic processes is proposed by Li and Du (2009). The process is defined by a frame and flexible activities which incorporate unknown factors through a set of constraints and an optional set of sub-activities. The work of Sun, Huang and Meng (2011) focuses on the integration of security constraints (restrictions of the roles performing certain activities) and business constraints (restrictions regarding the dependency between activities) in the definition of the process models. They propose algorithms for checking the constraints satisfaction and for planning the process execution based on constraints (Sun et al., 2011). Stefansen and Borch (2008) motivate the usage of soft constraints instead of hard constraints as the latter category does not properly provide support to the process participants. The soft constraints contain rules that can be violated. The impact of the rule violation is documented in advance by the process administrator (Stefansen & Borch, 2008).

DECLARE is a declarative-based tool that offers support for the under-specified model development, verification, automatic execution, run-time adaptation and analysis of the executed processes (Pesic et al., 2007). The constraint template is used for customizing the relations between activities in the process model. Before the execution, the model is verified for dead activities, conflicting constraints and history-based violation. Besides this, DECLARE support the process participant at design time (the syntactical and semantic verification of the defined model, the performance analysis through simulation) and at run-time (enforcing correct execution, giving recommendation regarding the most effective paths, monitoring process instances, learning from instances and enforcing correct changes) (Pesic et al., 2007; Aalst et al., 2009).

4.2.3 Process Adaptability

Process adaptability could be ensured in several ways (Reichert & Dadam, 1998; Dadam & Reichert, 2009; Schonenberg et al., 2008; Polyvyanyy & Weske, 2009; Weber et al., 2009; Reichert & Weber, 2012). One method consists in the anticipation of the possible exceptions (Kim et al., 2011) and the specification of their corresponding handlers during the design phase. Another method implies the creation of a repository with adaptation patterns (Weber et al., 2007) by the process administrator. Furthermore, if planned deviations are implemented, then the exceptions are usually handled completely or partially by the system. Contrary to this mechanism, unplanned deviations are instantiated by the process participant who decides when and how to invoke contextual changes of the process structure or behavior. Marrella and Mecella (2011) propose a prototype, built after a declarative approach that allows unplanned deviations. Every time when an unexpected deviation is detected, a recovery plan is run to handle it while the stable parts of the process are still executed (Marrella & Mecella, 2011).

The dynamic adaptability of the process is supported in the modeling phase by several works (Polyvyanyy & Weske, 2009; Fernandes, Ciarlini, Furtado, Hinchey, Casanova, & Breitman, 2007). Polyvyanyy and Weske (2009) created a method using hypergraphs to model ad-hoc processes which are more restrictive than the traditional ad-hoc processes (described further) and less restrictive than the control flow processes. The authors outline as disadvantage the lack of support for intuitive process visualization. Fernandes et al. (2007) propose a solution based on incremental planning and a library of pre-defined simple workflows. The planner can create new complex workflows based on the existing ones, having as input the description of the current scenario defined by the initial and final conditions (Fernandes et al., 2007).

The most cited project dealing with ad-hoc processes is ADEPT (Reichert & Dadam, 1998; Dadam & Reichert, 2009). The ADEPT technology support ad-hoc deviations, the migration of process instances to new process models and the correct process execution. It supports the agents in different activities of process lifecycle such as modeling, modifying, and deploying processes. Driven by the assumption the model cannot be too adaptable because this can raise problems, the authors created a minimal set of change operations that could be used by the process participants in modifying the process structure at run-time (Reichert & Dadam, 1998; Dadam & Reichert, 2009). Using ADEPT, the process administrator can create new process instances without being obligated to know the technical details of the process implementation. The activities composing the process are seen as black boxes with input and output. The process administrator must ensure the correct composition of these activities. The process participant can define ad-hoc changes during enactment and can use an integrated knowledge management system to assist him in triggering deviations (Dadam & Reichert, 2009).

Adams, Hofstede, Edmond, and Aalst (2005) explore a solution using worklets where a worklet is an action or an entire activity (set of actions) belonging to a process. The process is created of placeholders that have attached a repertoire of worklets and a rule tree guiding the worklet selection during instantiation. The selection depends on the case's dependent and independent context information (Adams et al., 2005).

One of the most discussed methods for adapting dynamically the processes is exception handling (Adams et al., 2005; Adams, Hofstede, Aalst, & Edmond, 2007; Kim, Choi, & Park, 2011). In general the solutions for exception handling consist of anticipating possible exceptions in advance. However, unexpected exceptions are frequently met and the traditional approach imposes the suspension of the workflow while manually treating them. The mechanisms of handling the exceptions remains very often in the head of the process participants and are not collected as an organizational knowledge (Adams et al., 2007). Adams et al. (2007) extends the worklets solution with a repertoire of exlets which are routines for handling various exceptions. Initially the process administrator defines the repertoire which automatically grows after, as new exceptions or new mechanisms of handling existing exceptions are registered. Different types of exceptions (constraints, timeout, resource

unavailable etc.) and exception handling primitives (remove, suspend, continue, restart, force completion, force fail etc.) are considered (Adams et al., 2007).
Compared to these reactive approaches, Kim et al. (2011) militate for proactive exception handling mechanisms consisting in exception prediction and exception prevention. They define a set of behavioral, functional, and informational requirements for proactive exception handling and a rule language based on these requirements. The process administrator or participant is responsible for manually handling the exceptions. If an exception is expected then it is predicted and after prevented by the system. If the prevention fails, a handling method is proposed by the tool to the agent, who decides if he applies it as it is, or modifies it. A knowledge base is also maintained after every exception's prediction and handling (Kim et al., 2011).

4.2.4 Process Evolution

Process evolution (Schonenberg et al., 2008; Weber et al., 2009; Reichert & Weber, 2012) consists in managing the migration of process instances, to comply with the latest process model. Compared to the other flexibility mechanisms, which deal with momentary changes, this approach encompasses evolutionary changes as a result of permanent variations in the organizational environment. Additionally, depending on the adopted migration strategy, the evolution can be radical (all the process instances are migrated) or incremental (only the process instances compatible with the new model are chosen) (Schonenberg et al., 2008). The decisions involved in process evolution are usually high level, organization-driven. Various stakeholders could be involved: internal stakeholders such as business owners, business unit managers, and external stakeholders such as suppliers and investors.

4.2.5 Data-driven Approach

The case handling paradigm and the corresponding tool, FLOWer, proposed by Aalst, Weske, and Grunbauer (2005) focuses on what can be done to achieve a certain business goal. A case is a process definition composed of a set of activities and their ordering. Each activity has associated a role that is responsible for its execution. The system creates a work item for each activity that's enabled, and assigns it to the workers having the correct role. The knowledge worker decides on how to achieve the goal during the case, assisted by the system. The system provides the relevant contextual information (the values of data objects related to the case's activities). The case approach is both data flow and control flow paradigm (Aalst et al., 2005).
An artifact driven enactment is described by Eckermann and Weidlich (2011) too. These processes are enacted based on the state of the objects (the values of object's properties). First a high level description of the process is realized. Further, the object lifecycle is modeled as transitions of objects states. Finally the workflow model is

created based on the object lifecycle model by specifying the execution order in terms of object states (Eckermann & Weidlich, 2011).

Vanderfeesten, Reijers and Aalst (2008) propose an enactment that is realized according to the product data model. The artifacts guide the decisions regarding the execution's order of the activities. The informational product for a specific process instance is created from the product data model which specifies the elements that need to be assembled. A product is created as a set of operations that have associated attributes such as the execution cost, processing time, execution conditions, failure probability, and resource class. Based on the values of these attributes and a set of criteria defined in advance, PAIS guides the users in assembling the elements to deliver the final product (Vanderfeesten et al., 2008).

4.3 Agency Implications

In Table 7, I created a summary of the flexibility requirements for each category (process variability, process under-specification, process adaptation, process evolution and data-driven approach) mapped on each phase of the process lifecycle (design, configuration and enactment). Moreover, since the recommendation tool will be designed to help the **process participants** during **enactment**, the implications of implementing flexible processes for them are highlighted in the circles marked in the table, and they are discussed in this section. The role of the process administrator was discussed too because of the overlapping activities with the process participant in several circumstances, as mentioned above.

	Design Phase	Configuration Phase	Enactment Phase
Process Variability	Reference model design - All possible paths (flexibility by enumeration) - With configurable nodes - With adjustment points Repository with process components: - Activities for each configurable node - Change options for each adjustment point - Process variant schemas - Workflow patterns Constraints and	Process variant selection by: - Selecting a value for the configurable node - Selecting a change option for the adjustment point - Selecting a process variant - Selecting a workflow pattern	Structure-based selection: - Selecting a process path in a conditional branch - Selecting a process variant Behavior-based selection: - see Workflow patterns Process participant guidance for: - analyzing the context - making decisions during enactment

	Guidelines for creating the process variant		
Process Under-specification	Reference model design: - With placeholders solved at runtime (*late binding*) - Template with loosely specified activities - Frame with flexible activities Repository with: - Process fragments - Loosely specified activities (plans, conditions, constraints) - Flexible activities (sub-activities, constraints) - Soft constraints (rules) Constraints and guidelines for process component selection, integration and enactment		Late selection of process component (activity, fragment, operation) Late modeling: - Partial/Complete modeling - Imperative/Declarative modeling - Iterative/Ad-hoc modeling Process model verification (constraints satisfaction) Process participant guidance for: - analyzing and defining the context - making decisions about selection / modeling - model verification
Process Adaptation	Reference model design: - As hypergraph - With placeholders Repository with: - Adaptation patterns - Worklets and Exlets - Exception handlers Constraints and guidelines for planned and unplanned deviation		Planned and unplanned deviation by exception handling/prevention Unplanned deviation (ad-hoc changes): - Structure-based changes - Behavior-based changes Process participant guidance for: - analyzing the context - making decisions about deviations / exception handling

Process Evolution	Process re-modeling Migration of process instances to a new model: - All of them (radical evolution strategy) - Only those compliant with the new model (incremental evolution strategy) Flexibility by change – modify the model at runtime: - Momentary change - Evolutionary change		
Data-driven Approach	Cases: - sequence of activities and their attached roles High-level process model / Object lifecycle model / Workflow model Product data model		Process participant guidance for: - analyzing the context - making decisions about artifacts transitions Artifact driven enactment based on: - object states (attributes) - external criteria

Table 7. The impact of flexible processes on the process participant

Flexibility highlights the agency characteristic of the process participant entailing the freedom of agents in making choices during the process enactment. The decision is founded in the context reasoning and experience. Additionally, the process participant should be able to validate the made decision, against the contextual constraints. Consequently, three key activities could be defined: **context interpretation**, **decision making** and **decision validation**.

The **decision** can be system or agent triggered. In PAISs incorporating flexibility by variability and by under-specification, the decision is system triggered. When this happens, the agent must choose between multiple branches, actions, activities, process fragments etc. When interacting with adaptive or data-driven PAISs, the agent can change the process instance at run-time. The agent’s decision to deviate the process execution path in an unplanned manner is based on the contextual knowledge. A clear comprehension of the context becomes thus mandatory.

The **context** is defined by the states of those objects relevant to the process execution. The context could be also interpreted in terms of events, linked to the changes in the object state. Furthermore, the contextual objects could belong to the system or to the real world (Figure 14). The agent is generally supported by the system in interpreting the system objects as, for example, in the situation of choosing a process fragment for filling a placeholder. However, when dealing with the interpretation of the real-world objects, an informed agent is not sufficient; he needs to be knowledgeable. This means that apart from being aware of the contextual information, the agent has also to rely on experience and skills.

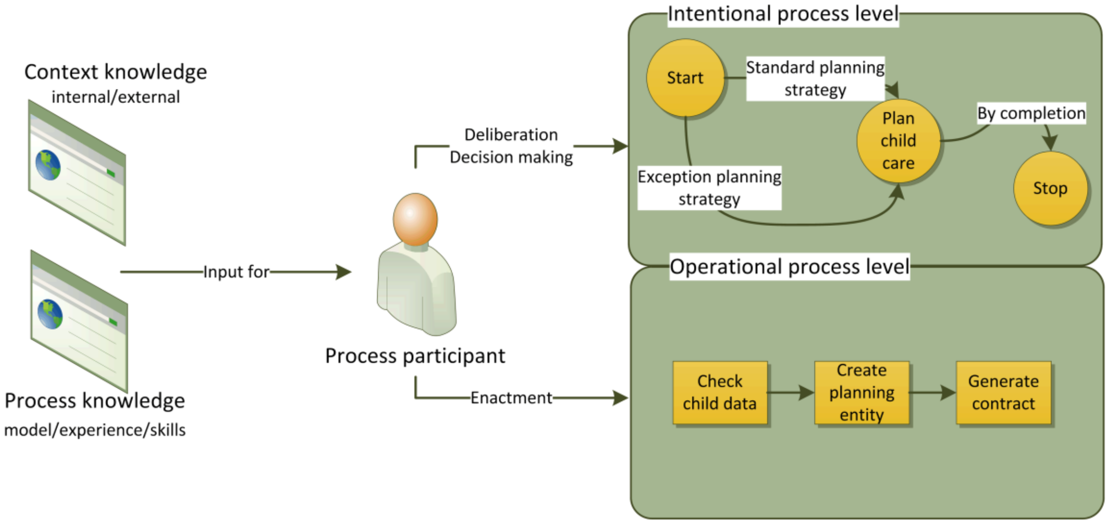


Figure 14. The process participant's activities requiring support

Finally, decisions could vary in complexity from simple scenarios such as conditional branches, to complex ones leading to modeling of process fragments at run-time. Hence, the **validation** of decisions can also have multiple levels of complexity. Frequently, constraints (Polyvyanyy & Weske, 2009; Sun et al., 2011) are used for validating decisions. However, it has been shown that, by imposing hard constraints, flexibility is actually restricted. Soft constraints (Stefansen & Borch, 2008) have proved to be a more effective mechanism. Nevertheless, system support becomes a necessity in order to ensure a correct process enactment in this context.

4.4 Artifact's Design Decisions

According to the previous section, three key activities of the process participants during enactment are impacted (and triggered) by the implementation of the flexible processes: context interpretation, decision making and decision validation. Therefore, in order to ensure a correct and effective enactment of the flexible processes, the process participant should be effectively supported in these activities.
The current research investigates mainly the decision-making support based on the process enactment history, focusing less on the context interpretation and validation. Having as input the mined intentional process model and the activity traces, the system could assist the process participant in making decisions by proposing possible intentions or strategies at a specific point in time (Figure 14). The process support in

this situation can be reduced to a well known problem: the plan recognition. There are various techniques, for plan or goal inference from user actions, based on machine learning algorithms (Carberry, 2001). Until now, only one research work on process guidance based on an intentional process model is known by the authors. Khodabandelou et al. (2013) propose a solution for discovering the intentions behind process participant activities using Hidden Markov Models. Nevertheless, other plan recognition techniques should be analyzed too and the most suitable and effective solution be integrated by the case company.

The design decisions regarding the recommendation tool are defined in accordance with the fact that the decision making is the central activity of the process participant that significantly influences the outcomes of the process. Therefore, the recommendation tool should:

- Support effectively the process participants in making decisions during process enactment;
- Consider the internal context knowledge (the process participant's trace) and the process knowledge (the process enactment experience, the process model);
- Provide recommendations correctly, by satisfying the process model, constraints and/or guidelines.

The effective support in decision making could be further formulated as a set of sub-requirements (Harris, 2012). The effective support in decision making is influenced by:

1. The effectiveness in identifying the decision criteria
2. The effectiveness in developing the decision alternatives
3. The effectiveness in analyzing the decision alternatives

5 INTENTION-MINING TECHNIQUE

This chapter illustrates the analysis of the second research sub-question:

How can the intentional process, behind the interaction of process participants with process aware information systems, be mined?

The chapter starts with the introduction of an example from the case company. Then it continues with the formulation of the design decisions behind the artifact: the intention-mining technique. Finally, the development of the artifact following the established design is presented.

5.1 Case Study Example

In Figure 15, the main process supported by the Childcare system is modeled using MAP meta-model (Rolland et al., 1999). A MAP process model always starts with a default intention "Start" and ends with a default intention "Stop". The other intentions that were identified through the analysis of the system (back-office and portal applications) and interviews with the stakeholders are: “Request child care”, “Plan child care”, “Manage contract” and “Manage invoice”. All these sections incorporate complex processes, thus they can be further decomposed in other maps. Strategies are identified by labels and they could be decomposed in sets of activities.

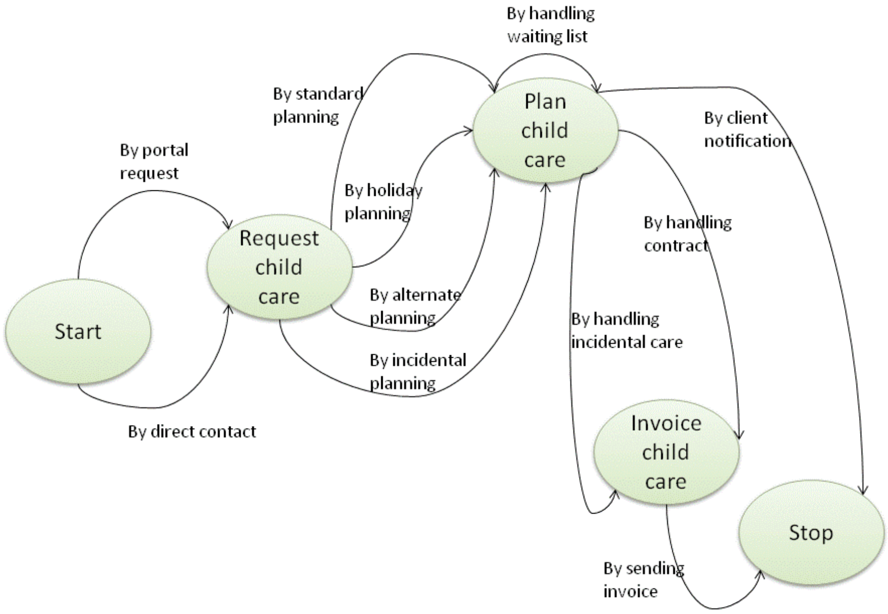


Figure 15. The high-level Childcare process map

In order to achieve the intention “Request child care” two possible strategies exist: “By portal request” (the parent will submit a child care request using the portal) and “By direct contact” (the parent contacts directly the Childcare organization and makes a request, the process participant responsible for the communication will directly create the request in the back-office application).

As already mentioned in the presentation of the case study, the achievement of the intention “Plan child care” is the most complex and flexible. After the child and the parent are registered in the system and a request have been submitted, the child planning will be made by the process participant using the back-office application. One or more strategies depending on the criteria specified in the request are applied for achieving this intention: “By standard planning”, “By holiday planning” and “By alternate planning”. If the required planning could not be realized respecting all the parent’s criteria, the request is put in a waiting list and the parent is contacted. Later on, it will be handled either when the parent makes a new request with other criteria or when a free slot is identified (a new group is created in a child care location or another child leaves an existing group). The parent could also contact the organization after the permanent planning has been made, asking for incidental planning (for several specific days or a period).

The first time when the request for permanent planning is made, the process participant generates a contract. Then, he sends it through the back-office application to the parents who are able to agree or not with it, using the portal application. To achieve this intention, only one strategy exists “By analyzing planning”.

The invoice is managed for the first time in case of the permanent planning “By contract analysis”, when a planning is changed and so the contract, or when incidental child care is requested (a special invoice is issued besides the permanent one).

The Stop intention is reached from “Plan child care” intention through the strategy “By client notification” and/or from “Manage invoice” intention through the strategy “By sending invoice”.

Further, the “Request child care” intention with the strategy “By direct contact” is zoomed in, using MAP (Figure 16):

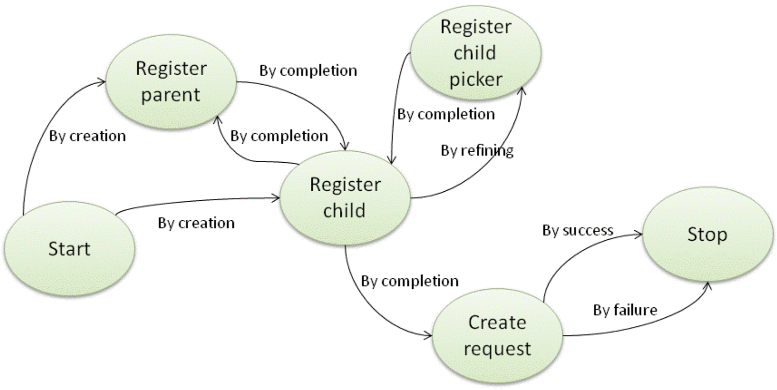


Figure 16. The map for the section <“Request child care”, “By direct contact”>

When the request is made through direct contact, all the operations could be made from back-office application and it depends on the preference of the process participant: register parent, then child, then child picker (optional) or register parent, then child picker, then child or register child, then parent and then child picker (optional) etc. After the registration of the child is achieved a request for child care is made from the portal (by the parent) or from the back-office application (by the process participant). Finally depending on the mentioned criteria and the technical details, the request for child care is ended with success or failure yielding for different activities incorporated in the corresponding strategies “By success” and “By failure”.

The intention mining technique is demonstrated for the “Request child care” process because it incorporates flexibility by the existence of multiple strategies and multiple ordering of the intentions. However, being less complex than the planning process, it can be easier analyzed in depth. Therefore, a detailed description of the executable process is further provided.

The manner of enacting processes in the Childcare system is driven by artifacts. More specifically, the transitions of their states composed by the values of the attributes define the control flow. Thus, an analysis of the entities involved in the "Request child care" process becomes mandatory.

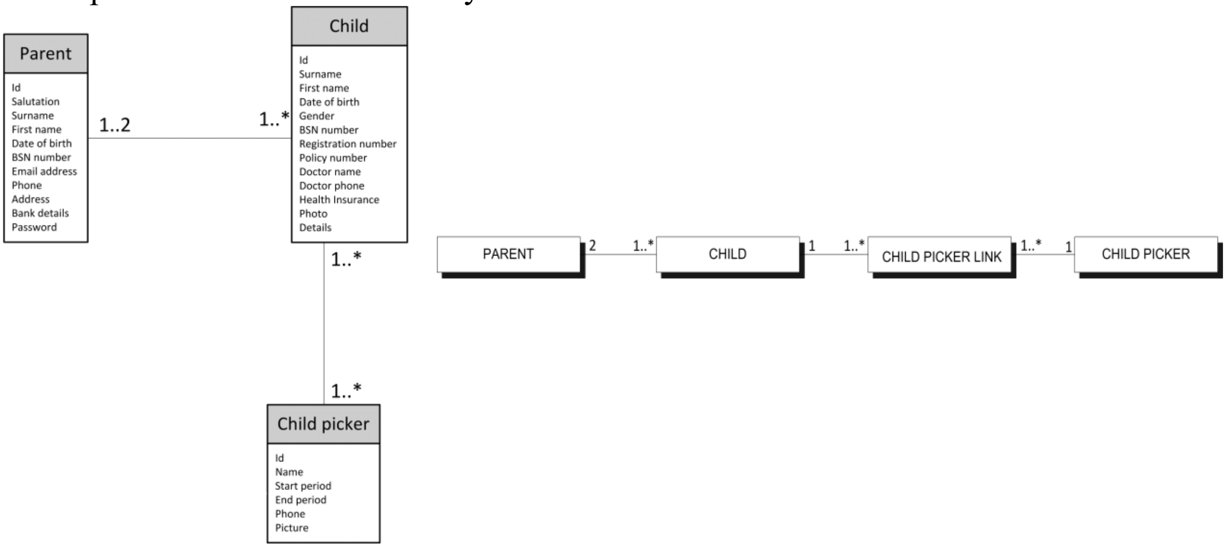


Figure 17. The entities involved in the registration process

In Figure 17, it is presented a snapshot of the entities relevant to the registration part. Some attributes and relationships to other external entities are omitted for the simplicity's sake. The conceptual model defines three entities of type Child, Child Picker and Parent and their related relationships. However, this model is not the one used when generating the application because of a constraint imposed by the generator: many-to-many relationships are not supported. Therefore, the Child entity has attached two relationships "*Parent 1*" and "*Parent 2*" to the same entity type, Parent. With Child and Child Picker, the transformation is different because of the unknown multiplicity of the association relation. To solve this, a new entity of type Child Picker Link is introduced.

A part of the scenario regarding the registration that a process participant might follow in the back-office application could be: Register Parent 1, Register Child, Register Parent 2 and Register Child Picker. The manipulation of the entities through CRUD operations ("*Create*", "*Read*", "*Update*" and "*Delete*") will compose the low level intentional process (represented as a control flow in Figure 18):

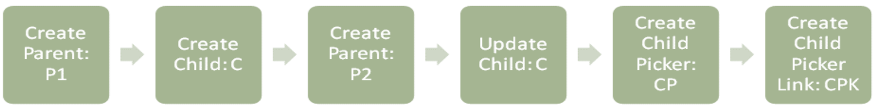


Figure 18. The low-level intentional registration process

First an entity of type Parent, P1, is created. Then, an entity of type Child, C, is created and, even if not explicitly represented in the image, the relationship "*to Parent 1*" is also created. After parent P2 is created, the child C needs to be updated with the value of the second parent. Finally the Child Picker, CP, is created and used in the creation of the Child Picker Link, CPK, for the corresponding relationship.

While the intentions described in the maps are high level, the example in Figure 18 represents the model of the lowest intentional process. A decomposition of these low level intentions results in a control flow model consisting in activities (collected through the application's event logs). For example, one strategy for achieving the intention "Update Child: C" could be composed of the following actions: navigate to the list of children, search child C by querying his family name, scan results, navigate to child C, create relationship to the second parent and save the modifications for child C. An interesting remark is that an activity does not exclusively belong to one single intention. The activity "*navigate to child*" could belong to the intention "*Update Child*" but it could also belong to the intention "*Read Child*" in the scenario when the process participant is only interested in opening the form to check some details.

The first version of the intention mining technique aims at discovering the low level intentional process while future extensions will be focused on the aggregation of these models in higher level maps. The example with the low-level intentional registration process, introduced in this section, will be used in the presentation of the artifact's design and demonstration.

5.2 Artifact's Design

The main design goal of the intention-mining technique is to mine the low-level intentional process which means: 1) mining the intentions behind the process participant's activities, and 2) mining the flow between these intentions. Several challenges are identified and could be grouped in the following categories:

1. The *input* of the intention mining technique. The first challenge is to identify what data could be collected from the information system during the process enactment and could be the base for extracting intentionality-related knowledge. The second challenge consists in identifying a suitable mechanism of collecting this data, as event logs, from the information system.
2. The *algorithm* behind the intention mining technique. The challenge here is to discover an algorithm which is able to correctly mine the low level intentions from the event logs that are given as input. This implies the formulation of rules that guides the extraction of intentions, strategies and the intentional flow, from event logs.
3. The *output* of the intention mining technique. In the previous statement it was mentioned that the intention-mining technique must provide a correct result. The challenge for this category is to define what correct means considering that intentionality is very subjective, and what would be a suitable method for the artifact's evaluation.

The design decisions regarding each of the above presented categories and their materialization in the final artifact's design are discussed in the following sections.

5.2.1 Input-related Design

In order to identify the data, that could be transformed in knowledge for mining the intentional process, an analysis of this topic for other process mining techniques applied in practice was conducted (Aalst, 2011). Aalst (2011) covers three areas of interest regarding the collection of data for process mining: data sources, event logs and the XES standard for storing and exchanging logs.

Data sources can be very diverse depending on the nature of the log collection mechanism of the information system in question such as database tables, simple text or binary files, Excel spreadsheets or transaction logs. The data contained by these sources might be structured (described by clear meta-data) or completely unstructured (Aalst, 2011). Therefore, the first decision is to identify the data sources that could be relevant. The following phase is inspired from data mining and business intelligence and it consists of three steps: extract the data, transform the data for the operational needs and load the transformed data into a storage system such as a relational database (Aalst, 2011). The next step introduced by the process mining requires a mechanism for converting the log data from such a repository in event logs, that are further used as input in different techniques.

The second interview with the case company revealed that the Childcare system did not log any data regarding the interaction of process participants with the system. However, the existing data in the database tables could provide some relevant information such as the creation of an entity or the deletion of an entity. The first issue was that only high level activities could have been extracted from this data. The second issue was that the recorded data was not comprehensive to represent a good support for mining the process as only some activities could have been identified even if others were acknowledged to be relevant to the process (for example a read operation was not detected). Lastly, the case company declared that they were also very interested in a fine grained analysis of the process participant's behavior. All these observations lead to the conclusion that a logging mechanism had to be integrated in the Childcare system. The company completely agreed and provided all the resources for putting the plan in execution.

Considering all the theoretical and practical aspects introduced above, the following design decisions regarding the data sources were made:

- The data source will be a relational database.
- A logging mechanism for collecting the relevant data in database will be created. Thus, the data does not need transformation as it will be registered in the required form from the beginning.

- A mechanism for extracting the data from its corresponding database table and for converting it in event logs will be created.

Event log is formed from the assumption that it contains events only related to a single process. Each event must belong to a process instance, also called “case” or “trace”. Integrating this with what was presented in the section about process mining, a log has associated a single process, and consequently a process consists of cases, a case consists of events and events can have multiple attributes (Aalst, 2011). After analyzing other process mining techniques and the Childcare system the attributes presented in Table 8 were chosen to be included in an event’s description.

	Attribute	Description
General information	Event Id	The unique identifier of the event (the primary key for the database table where logs are saved).
	Originator	The username of the process participant that triggered the event. Each process participant logins the system before the process enactment with a specific username and password.
	Operation	The name of the operation. The list is generated based on the CRUD operations and other possible operations identified when analyzing the back-office application.
	Timestamp	Date and time information describing the moment when the event was triggered.
	Entity Type	The name of the entity’s type involved in the event, if any. For example: Childcare.Child, Childcare.Parent etc.
	Trace Id	The trace is identified by a child’s id. All events that include entities relating to a specific child are considered as belonging to the same trace.
Auxiliary information	Lifecycle Transition State	In some systems, there could be events that do not end in the moment they are started. They have different states such as “Start”, “Schedule” etc. For the Childcare system the events are momentary. This attribute will not be used in intention-mining but it is included in case of future extensions.
Context information	List of <Key, Value>	The context information is considered very relevant for mining intentions. The logged data depends on the operation and it is detailed in “Artifact’s development” section.

Table 8. The event log structure for Childcare

XES standard for storing and exchanging logs specifies a XML based syntax which could be easily extendible. A XES has as a root element a log which can contain multiple traces. The core types of the attributes for describing an event are the standard XML types: xs:string, xs:dateTime, xs:long, xs:double, and xs:boolean. An event does not have a prescribed list of attributes except from the mandatory ones, described as global attributes in the schema. Attributes are defined depending on the operational needs using extensions. For instance, the Time extension defines a timestamp attribute of type xs:dateTime and an Organizational extension defines an originator attribute of type xs:string (Aalst, 2011).

This leads to a new requirement of the mechanism for extracting the data from its corresponding database table. The data will be converted in event logs generated in accordance with the XES schema. An example of a resulting event log is given further:


```xml
<?xml version="1.0" encoding="utf-8"?>
<log   xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"   xmlns:xsd="http://www.w3.org/2001/XMLSchema"
xes.version="1"  xes.features="nested-attributes" openxes.version="1.0RC7" xmlns="http://www.xes-standard.org/">
        <string key="source" value="Childcare.mxml" />
        <string key="description" value="CPN Tools simulation" />
        <extension name="Lifecycle" prefix="lifecycle" uri="http://www.xes-standard.org/lifecycle.xesext" />
        <extension name="Organizational" prefix="org" uri="http://www.xes-standard.org/org.xesext" />
        <extension name="Time" prefix="time" uri="http://www.xes-standard.org/time.xesext" />
        <extension name="Concept" prefix="concept" uri="http://www.xes-standard.org/concept.xesext" />
        <extension name="Semantic" prefix="semantic" uri="http://www.xes-standard.org/semantic.xesext" />
        <global scope="trace">
                <string key="concept:name" value="UNKNOWN" />
        </global>
        <global scope="event">
                <string key="concept:name" value="UNKNOWN" />
                <string key="lifecycle:transition" value="complete" />
        </global>
        <trace>
                <event>
                        <string key="org:resource" value="System" />
                        <date key="time:timestamp" value="2013-04-10T18:08:11.140432+02:00" />
                        <string key="concept:name" value="Register" />
                        <id key="concept:id" value="26362345"/>
                        <string key="concept:entityType" value="Childcare.Child"/>
                        <string key="lifecycle:transition" />
```

```
                    </event>
          </trace>
</log>
```

Figure 19. The XES-compliant event log

5.2.2 Algorithm-related Design

The design of the algorithm started from two assumptions about the intentions:

1. Intentions are mined per process participant. Consequently, the collective intentions are not identified. Considering that the mining algorithm focuses on low-level intentions which span over a short period of time, it is natural to assume that they belong to the same process participant. This is the trivial situation, deducted from the statement that humans are intentional: the process participant enacts a process driven by some low-level intentions which he achieves in the end. However, these low level intentions belong to higher level intentions which incorporate the collective intentionality.
2. A low-level intention consists of a sequence of consecutive activities. This is deducted from the previous assumption that the low-level intentions are time-bounded and span over a short period. Once more, it is assumed that if a process participant starts to act to achieve a low-level intention, he does not interpose other intentions in between. The interposition means changing the intention in the context of low level intentional perspective of the process. However, this does not hold for high level intentions.

The steps of the created algorithm are showed in Figure 20. The rationale behind the design of each step is presented together with the step's description further.

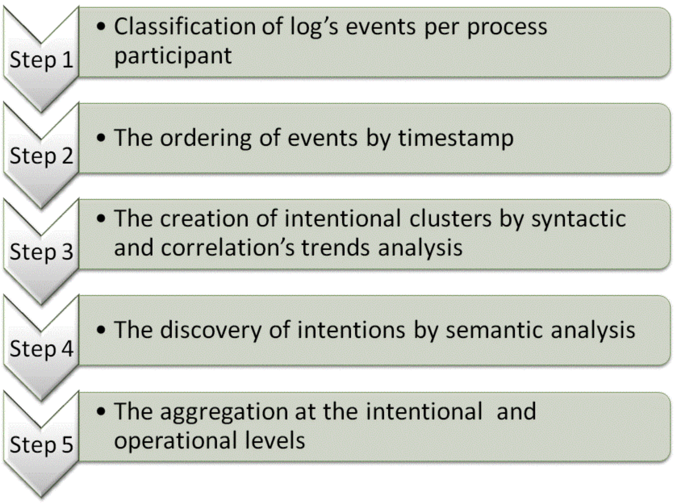


Figure 20. The intention mining algorithm

The first assumption implies that the algorithm should start with **classifying the events per process participant**. Then, for each set of events belonging to a process participant, the events should be **ordered by timestamp**. According to the second

assumption, this implies that the intentions will be also ordered in time (which is nevertheless a simplified case of the reality).
What is known at this moment is that the transformed events (classified per process participant and ordered by timestamp) could be grouped in some intentional clusters and these clusters are also ordered (Figure 21).

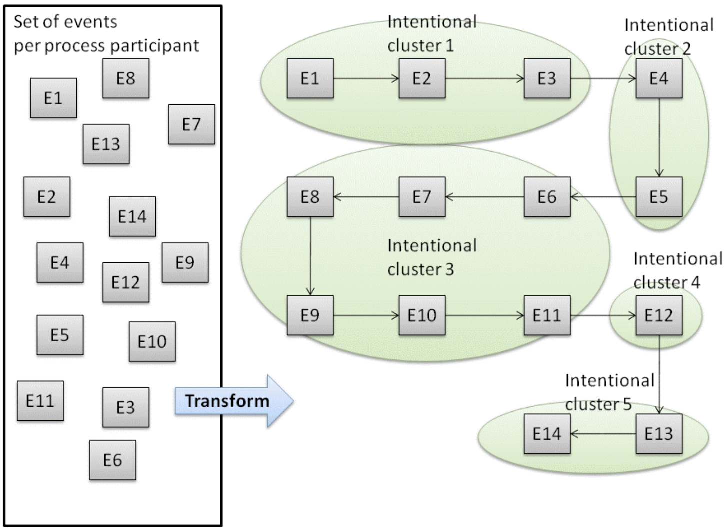


Figure 21. The transformation of events in intentional clusters

What is not known yet is how to discover these intentional clusters. Accordingly, the next challenge is the identification of a cost function that could measure the intentional dissimilarity of two events, (also known as distance in clustering algorithms). However, comparing to the traditional clustering algorithms where the cost function is applied for each pair belonging to the whole space of events, in this mining algorithm the function is applied only for two consecutive events. This design decision is in accordance with the second assumption presented in the beginning of the section.
Therefore, this function quantifies the intentional correlation of two consecutive events. The intentional correlation depends on the information known about and incorporated in the events: the trace id, the entity type and the specific entity instance identified by its id. Moreover these values should not equally contribute to the function. For example, two consecutive events that refer to the same entity instance should have a much stronger correlation than two events that refer to the same entity type. The intentional correlation can also be considered a **syntactic analysis** of the events as it quantifies structural information of the activities. The function that calculates the intentional (syntactic) correlation could be defined as it follows:

Intentional_corr(E1, E2) = EntityType_corr (E1,E2) + EntityId_corr(E1,E2) + TraceId_corr(E1, E2)

EntityType_corr (E1, E2) = entityTypeCorrelationFactor * ReferenceUnit EntityId_corr(E1, E2) = entityInstanceCorrelationFactor * ReferenceUnit TraceId_corr(E1, E2) = traceCorrelationFactor * ReferenceUnit
where: ➢ entityTypeCorrelationFactor, entityInstanceCorrelationFactor and traceCorrelationFactor are configured by the process administrator and entityInstanceCorrelationFactor > entityTypeCorrelationFactor > traceCorrelationFactor ➢ ReferenceUnit is a constant which equals 1

The next step is the **normalization** of the correlation values calculated by the function. For example, if the sequence of events belongs to the same trace, then the minimum correlation value found in this set is "*traceCorrelationFactor* *** *ReferenceUnit*". But, since this correlation compared to the other correlation values is the lowest, from all the values is subtracted this minimum so as to the minimum correlation value becomes null.

For the deduction of the next step of the algorithm, the following sequence of ordered events belonging to the same trace and originator is presented as an example:

E1: Read the list of Child entities -> E2: Read the Child entity with the id C1 -> E3: Update the Child entity with the id C1 -> E4: Read the list of Child entities -> E5: Read the Child entity with the id C2 -> E6: Read the list of Parent entities -> E7: Read the list of Child Picker entities.

Interpreting this subset, the most probable and logical flow of process participant's intentions is:

- Update the Child C1: he first navigated to the list of children, after he opened the form with the child he was looking for and finally he updated it;
- Read the Child C2: he first navigated to the list of children and he opened the form with the searched child;
- Read the list of parents;
- Read the list of child pickers.

The first observation that could be made out from this example is that the intentionality is transmitted progressively: an event or a sequence of events contributes increasingly to the intention's construction.

Further, the correlation values are calculated considering that the traceCorrelationFactor equals X, entityTypeCorrelationFactor equals X+1 and entityInstanceCorrelationFactor equals X+2, where X is a certain numerical value:

*Intentional_corr(E1, E2) = 2*X + 1 -> Intentional_corr(E2, E3) = 3*X + 3 -> Intentional_corr(E3, E4) = 2*X + 1 -> Intentional_corr(E4, E5) = 2*X + 1 -> Intentional_corr(E5, E6) = X -> Intentional_corr(E6, E7) = X*

After normalization the correlation values become:

*Intentional_corr(E1, E2) = X + 1 -> Intentional_corr(E2, E3) = 2*X + 3 -> Intentional_corr(E3, E4) = X + 1 -> Intentional_corr(E4, E5) = X + 1 -> Intentional_corr(E5, E6) = 0 -> Intentional_corr(E6, E7) = 0*

Analyzing these values, the second observation is that the progressive transmission of intentionality is captured by the trend in the correlation's values as follows:

- An increasing trend marks the progressive construction of an intention.
- A change in trend from increasing to decreasing and a null correlation value delineates two intentional clusters.

According to these rules the intentional clusters formed, based on this example, are: C1={E1, E2, E3}, C2={E4, E5}, C3={E6}, C4={E7}. E1 and E2 have a correlation higher than 0, then they are grouped in the same cluster. The correlation of E3 with E2 is higher than its correlation with E4, thus it is added to the first cluster. At this point, the first change in trend is identified (the decrease from 2*X+3 to X+1) so the second intentional cluster is formed to which E4 is added. Further, the correlation of E5 with E4 is higher than the correlation with E6 so it is added to the second cluster. Moreover, the change in trend (the decrease from X+1 to 0) marks the creation of the third cluster consisting of E6. Finally, because the correlation of E6 with E7 is zero (they do not correlate at all), the forth cluster consisting of E7 is built.

Now that the intentional clusters are identified, the next step is the identification of the intention associated with each cluster. In order to obtain this, the **semantic analysis** of the intentional clusters is conducted. The semantic analysis consists in understanding behind the structure of the process activities, their meaning by interpreting the fields "*Operation*" and "*Event Type*". As presented in the description of the case company, the development or better said the generation of Childcare follows a model driven paradigm. The implications are that the entity model is known in advance (it is defined using the web-based Application Designer and it is provided as input to the Generator) and the set of operations is finite and could be also identified. After the analysis of the Childcare back-office and having as reference the definition of CRUD, the following operations and intentions were identified: Create entity, Update entity, Read entity, Delete entity, Update entity's field, Create entity's relation, Update entity's relation, Read entities, Search entity by keyword(s), Search entities by keyword(s), Show popup, Show report. Moreover, the intention composition that is the pre-defined knowledge base for the semantic analysis is:

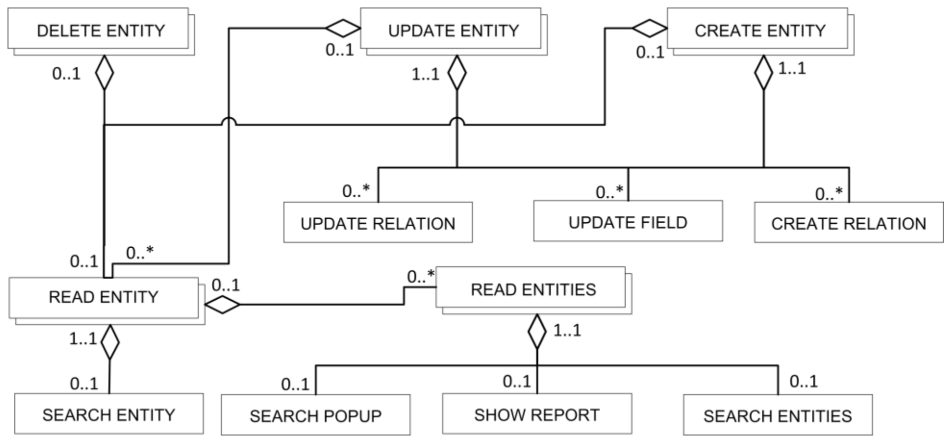


Figure 22. Intention composition for semantic analysis

Figure 22 shows that there are four types of intentions that could be identified in advance for each entity type: Create entity, Update entity, Delete entity, Read entity and Read entities. Moreover, it identifies a list of intentional activities which could not be considered as stand-alone low level intentions but as activities belonging to certain intentions: Update relation, Update field, Create relation, Search entity, Search entities, Show popup and Show report. There are two types of composition relations: one for marking that an activity belongs to an intention and another one for marking that an intention could belong to another intention. This means that some intentions are the final goal of the process participant or they are achieved just as a mean to other intention's achievement: Read entities, Read entity. The intention mining will incorporate this knowledge for extracting the intention from an intentional cluster by applying semantic analysis. Once new intentions or new activities are identified, the internal knowledge of the intention mining technique needs to be updated too.

The final step is the **aggregation** which takes place at two different levels:

- The aggregation of the intentional process instance in the global process model. If new intentions or new connections among intentions are discovered, they are added to the MAP corresponding to the process.
- The aggregation of the intentional clusters corresponding to a specific intention in one control flow model composed of events.

In order to construct the complete map process, the strategies should be identified too. The strategies define multiple possibilities of achieving an intention. Therefore, they could be extracted from the intentional cluster by the identification of the paths in the control flow model consisting of events. However, the first problem is that multiple paths do not necessarily represent different strategies and the second problem is the naming of the strategy from the known information considering that this activity is very subjective in real-life too. For the scope of the current project, it is considered that only one strategy links two intentions while the problems identified earlier are left for future exploration (with ontologies and natural language processing).

5.2.3 Output-related Design

The discovery of the intentional process model has some parts that could be formally verified such as the sorting algorithm or the aggregation algorithm. However the validation of the correctness regarding the intentions discovery could be realized only with the involvement of the process participant. The intention is in process participant's mind and he is the only person that could articulate it and thus validate the intention mining technique.

The precision of the algorithm will be measured by calculating the ratio between the correct identified intentions by the algorithm and the number of total trials. An intention is considered as correctly discovered if the process participant confirms that he acted in order to achieve that. More about the evaluation will be presented in a following chapter.

5.3 Artifact's Development

The artifact was developed using Microsoft .NET framework, C# programming language, Visual Studio 2012 as integrated environment and Microsoft SQL Server 2005. The decision to choose these technologies was influenced by the case company as these were also used in the development of the Childcare system. Therefore, an easier integration of the artifacts with the company's product was possible.

During the discussions with the senior developer, it has been decided that the most effective and efficient way to store and manipulate the events is by generating them as entities of the Childcare system. Using the Application Designer, two entity types were defined: EventTraceInformation and ItemInformation (Figure 23). The fields of the entities followed the specification of the artifact's design, but their types were adapted to the application type system consisting of Text, Bit, Decimal, Date and time, Integer and Numeric. A relation of one-to-many was defined between EventTraceInformation and ItemInformation entities.

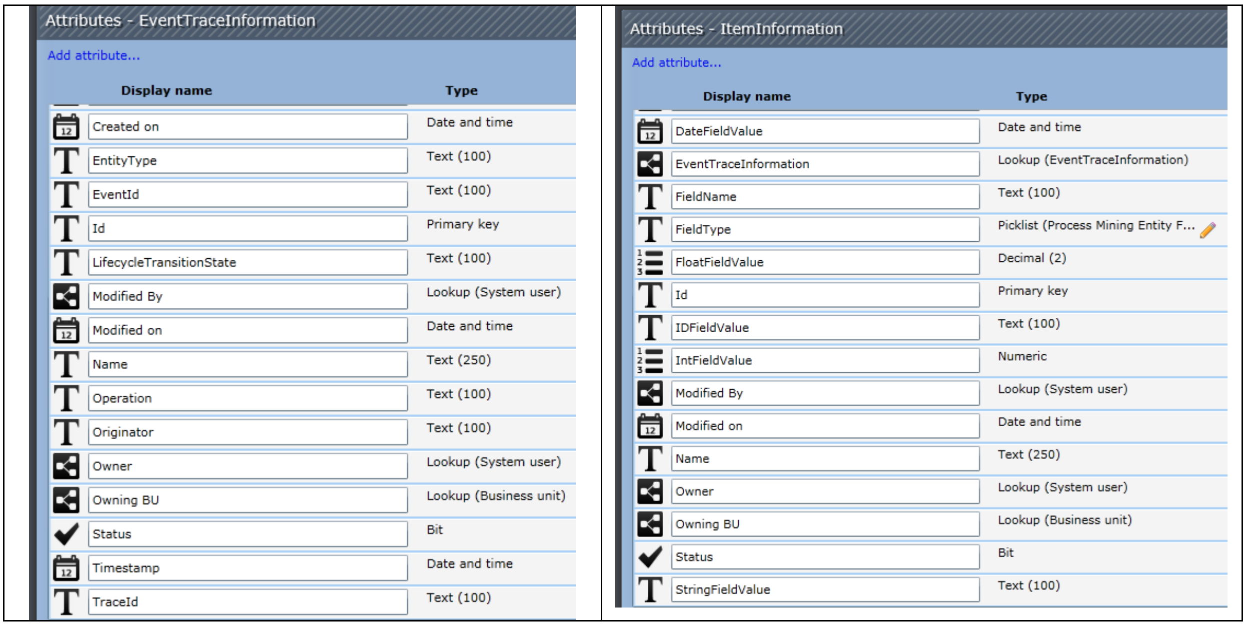


Figure 23. The definition of log-related entities in Application Designer

The logging mechanism was implemented using seven extension points in the Childcare back-office:

Class	Method	Description
BusinessEntityFactory	SaveEntityPartial	Each entity has a factory which handles the save operations. “Create entity” and “Update entity” are logged here.
BusinessEntity	OnChangedPartial	All entities inherit the class BusinessEntity. “Create relation”, “Update relation” and “Update field” are logged here.
BasePage	DoNavigatePartial	The web pages inherit the class BasePage. “Read entity” and “Read entities” are logged here.
Master	PerformActionPartial	All web pages inherit the class Master. Various actions on the page are logged.
Master	ShowPopupPartial	“Show popup” action is logged here.
GridBaseControl	SetSearchConditionPartial	This is the method where a search is logged.
Report	ShowReportPartial	The application offers the possibility to generate reports. “Show report” action is

		logged here.

Table 9. Extension points for logging mechanism

A separate tool for the conversion of traces in XES compliant log was created. The first phaseconsisted in the definition of a XES library consisting of C# classes mapped on the XES schema (Figure 24). The classes were annotated with XML attributes so that the serialization of their objects would result in an XML file that is compliant with the XES standard.

```
[XmlType(Namespace = "http://www.xes-standard.org/")]
[XmlRootAttribute("log", Namespace = "http://www.xes-standard.org/", IsNullable = false)]
public class LogType: ElementType
{
    [XmlElementAttribute("extension")]
    public List<ExtensionType> Extensions { get; set; }

    [XmlElementAttribute("global")]
    public List<GlobalsType> Globals { get; set; }

    [XmlElementAttribute("classifier")]
    public List<ClassifierType> Classifiers { get; set; }

    [XmlElementAttribute("trace")]
    public List<TraceType> Traces { get; set; }

    [XmlAttribute("xes.version")]
    public Decimal Version { get; set; }

    [XmlAttribute("xes.features", DataType = "token")]
    public String Features { get; set; }

    [XmlAttribute("openxes.version")]
    public String OpenXesVersion { get; set; }
}
}
```

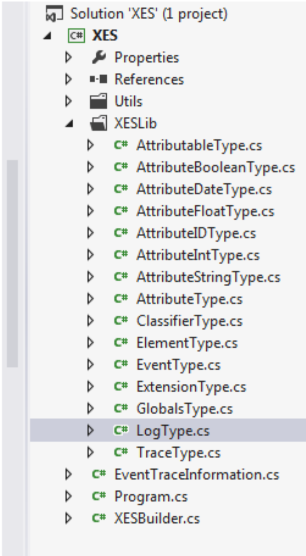


Figure 24. The XES conversion tool

The second phase was the creation of the XESBuilder class which main role was to convert a list of EventTraceInformation entities in a XES log file (Figure 25). The public method of the class is CreateXESFile which could be found in two versions depending on the parameters given as input. The other private methods are used as helpers for the main public functionality.

```
namespace XES
{
    public class XESBuilder
    {
        public void CreateXESFile(List<EventTraceInformation> events, List<ItemInformation> logInformationList)...

        public void CreateXESFile(string fileAbsolutePath, List<EventTraceInformation> events, List<ItemInformation> logInformationList)...

        private void SerializeLogInFile(string fileAbsolutePath, LogType logType)...

        private void CreateTraceEvent(ref LogType logType, List<EventTraceInformation> events)...

        private void CreateLogGeneralInformation(ref LogType logType, List<ItemInformation> logInformationList)...

        private void CreateDefaultLogAttributes(ref LogType logType)...

        private void CreateLogGlobals(ref LogType logType)...

        private void CreateLogExtensions(ref LogType logType)...

        private ExtensionType CreateExtensionType(string extensionName, string extensionPrefix, string extensionUri)...
    }
}
```

Figure 25. The XESBuilder class

The intention-mining algorithm is presented in Figure 26. The first method is the configuration of the parameters: traceCorrelationFactor, entityInstanceCorrelationFactor and entityTypeCorrelationFactor. Then, it has two methods for input/output operations: the loading and writing of a list of events from and to a file. The main method is ExtractIntentionsFromEvents which is implemented based on the helper methods. The methods are mapped on the steps presented in the algorithm's design: the sorting based on timestamp, the extraction of intentional clusters based on syntactic analysis (including the normalization) etc. The aggregation stores the output as a Graph structure which will be presented in the following chapter.

```
public static void ConfigureParams()...

public static bool LoadEventsFromLogFile(string logFullPath, out List<EventTraceInformation> events)...

public static void WriteEventsInLog(string logFullPath, List<EventTraceInformation> events)...

public static List<Intention> ExtractIntentionsFromEvents(List<EventTraceInformation> events)...

private static void ExtractIntentionsFromIntentionalClusters(List<List<EventTraceInformation>> intentionalClusters)...

private static void SortEventsByTimestamp(ref EventTraceInformation[] events)...

private static List<List<EventTraceInformation>> GetIntentionalClusters(List<List<SyntacticCorrelation>> intentionalPairClusters,
    List<EventTraceInformation> events)...

private static List<List<SyntacticCorrelation>> GetIntentionalPairClusters(List<SyntacticCorrelation> eventsSyntactiCor)...

private static List<SyntacticCorrelation> GetNormalizedSyntacticCorrelationForEvents(List<EventTraceInformation> events)...

private static SyntacticCorrelation GetSyntacticCorrelation(EventTraceInformation event1, EventTraceInformation event2)...
```

Figure 26. The implementation of the intention mining technique

To conclude with, the second sub-question was answered by proposing an approach to discovering the intentions behind the process enactment focused on three key areas: the input, the algorithm and the output. The intention mining technique was created in compliance with several design decision made in advance. The design decisions were influenced by the case organization and the existing theories.

6 INTENTION-BASED RECOMMENDATION TOOL

In this chapter, the third research sub-question is discussed:

How can the process participant's activities impacted by the implementation of flexible processes be supported through a recommendation tool based on intention mining?

First, the usage of the recommendation tool is presented in the settings of the case study. Further the artifact's design is created, having as a starting point the decisions made after the discussion of the first research sub-question. Finally, the details regarding the development of the artifact are exposed.

6.1 Case Study Example

While flexible processes are ensured though artifact-driven implementation and enactment, the interaction of process participants with the system becomes much more reliant on their agency capacity. Consequently, a correct and efficient enactment depends substantially on a correct reasoning and an efficient decision making.

This section starts by considering the following scenario in the context of Childcare system: a child was registered and started to benefit of childcare when he was three year old. After he turns four year old and he starts the school, its care planning needs to change because he needs a different type of care (from KDV – "*little children care*" to BSO – "*after school care*"). What does this mean in terms of process enactment? Several steps must be taken but the decision on the exact set of steps and their execution order can raise the next issues for the process participant:

- Should the current child's planning be updated or should a new planning be created?
- If the school does not exist in the system, it must be created together with all the requested entities related to it. When and by whom should it be introduced in the system?
- If a new planning is created then should it be generated through a new request entity or should it be directly created, without being generated through a request?
- If the request must be created then who is responsible for its creation: the process participant from the back-office or the parents from the portal? Should the process participant contact the parents or should they have the initiative for change?
- Should the initial contract be updated or should a new contract be created?
- If the new planning is not possible, should the request go in the waiting list or should the parent agree with the modification of the request?
- What should be the order of these steps?

This scenario is just one example of complex decision making situations and it outlines clearly the necessity of support for the process participant. The recommendation tool should help the process participant to make an informed decision about what he could do further. An informed decision means on one hand that all the possible options are presented together with a confidence factor, on the other hand these options are not restrictive and the process participant could decide to follow his particular approach. Another aspect of the recommendation tool is that the recommendation should be provided first at the intentional level (the natural abstraction for agent's reasoning) and after at the executable level.

Further, a generic example (a possible abstraction from the Childcare' scenarios) is presented for which the names of the intentions are simplified. This will be used in supporting the functional description of the recommendation tool.

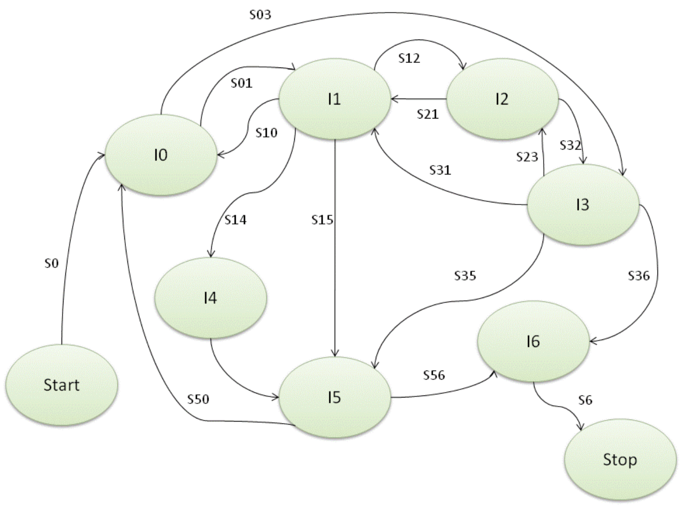


Figure 27. Case study example for the recommendation tool

Considering that this map could be discovered using the intention-mining technique, the model inherits its limitations: between the source intention and the target intention there is only one strategy. However, a second strategy might exist between two different intentions but in the opposite direction (the source and target roles are inversed as it is the case with the pairs (I0, I1), (I1, I2) etc.).

6.2 Artifact's Design

Several design decisions have been outlined as a concluding part of the first research sub-question's discussion. The first artifact's design decision concerned the decision making during process enactment: the recommendation tool should provide effective support for making decisions where the effectiveness could be defined by three criteria: effectiveness in identifying the decision criteria, effectiveness in developing the decision alternatives and effectiveness in analyzing the decision alternatives.

The recommendation tool is built so as to the decision criterion is defined at the intentional level. As outlined in the section regarding intentionality by the existence of theories in several areas such as philosophy, artificial intelligence and information system, the human reasoning is driven by intentions. A tool that articulates the intentions offers a better support to the process participant to make a decision from two perspectives: the intention can be realized by the process participant if it was not before and the intention represents the criterion for choosing a specific path for its achievement.

The decision alternatives could be provided at different levels as it follows: first at the intentional level (the intention that has to be achieved is chosen), then at the plan level (the strategy how to realize a certain intention is chosen) and finally at the executable level (the action that needs to be taken to follow a certain strategy is chosen).

Consequently, the analysis of alternatives starts at the intentional level having the context and the process participant's tacit knowledge as criterion. After, the chosen intention could become the criterion in deciding upon a strategy.

The second design decision of the artifact was that the recommendations are formulated considering the internal context knowledge – the process participant's trace, and the process knowledge – the process enactment experience and the process model. The process participant's trace offers information about the historical flow of intentions that he achieved. Combining this knowledge with the collective process experience gathered as the mined process model, the recommendation tool provides recommendations in terms of intentions. However, because the input is generated by the intention mining technique and the strategies are not yet identified, it incorporates only the first step – the recommendation of intentions, while the recommendation of strategies is left as future functionality.

The last requirements was that the recommendations should be correct, satisfying the process model, constraints and/or guidelines. This will be partially satisfied as the tool will follow the mined process model but the map cannot be formally verified for its correctness. However, each recommendation is given with a confidence factor which is calculated based on the scenario's frequency inferred from the tool's knowledge base. A high confidence factor reveals that usually the crowd follows that specific path in the enactment. However, a low confidence factor does not necessarily mean an incorrect decision. It could also represent an exceptional adaptation of the process instance under certain circumstances. Therefore, the process participant should reason about which alternative he will follow.

To this point, it is known that the recommendation tool receives as initial input a map represented as a graph of intentions with its corresponding semantics. Then, every time it is invoked by the process participant, a trace of intentions is provided as input and it will create a set of recommendations as output (a set of intentions with their corresponding confidence factors). Moreover, these partial traces of the process participants will be used for updating the map after the recommendation is given. The next step is the design of a mechanism that calculates the confidence factors for each recommendation.

A Bayesian approach is adopted as it offers support for statistical inference based on known prior information and conditional probability calculus. The Bayesian approach is built around the mathematical theorem discovered by Reverend Thomas Bayes (1701-1761) which is formulated as it follows (Bovens & Hartmann, 2003):

$P(H|E) = P(H) * P(E|H) / P(E)$, where:
P(H|E) is the conditional probability that a certain hypothesis H is true when evidence E is observed P(H) is the unconditional probability that a certain hypothesis H is true without having any evidence P(E) is the probability the a certain evidence E is observed (its corresponding event occurs)

P(E|H) is the conditional probability that a certain evidence E is observed when hypothesis H is true

Moreover, from conditional probabilistic calculus, the chain rule specifies that:

$\mathbf{P(H_n, \ldots, H_2, H_1) = P(H_n \mid H_{n-1}, H_{n-2}, \ldots, H_2, H_1) * P(H_{n-1}, H_{n-2}, \ldots, H_2, H_1)}$, where:
$P(H_n, \ldots, H_2, H_1)$ is the probability that all hypothesis $H_n, \ldots, H_2, H_1$ are true $P(H_n \mid H_{n-1}, H_{n-2}, \ldots, H_2, H_1)$ is the conditional probability as defined above but with a set of evidence

The Bayesian approach can be used to predict with a certain probability that a certain sequence happens and the probability is calculated using the chain rule and Bayes' Theorem. Nonetheless, for being able to do so, it requires the formulation of a probabilistic model consisting of hypothesis and evidence which represents the base for the statistical inference. There is not one single way for modeling the environment but for this situation Occam's razor is applied which specifies that the simplest model consistent with the environment is chosen (RefHutter, 2007).
The implications of following a Bayesian epistemology for the recommendation tool are:

- Providing a recommendation based on a partial trace is a matter of prediction. A recommendation is the next predicted intention together with the probability that this intention is true, conditioned by the fact that the input sequence of intentions occurs.
- All the intentions that could be reached by following a certain sequence are predicted with their probabilities. This helps the process participant to make an informed decision.
- The initial traces used for the map's construction and the partial traces will form the prior information for making prediction. This approach aggregates historical information by updating the known probabilities as soon as this information is revealed.

An important remark is that not only the fact that some intentions occur is important but also the order in which they occur (the sequence's order). Therefore, the probabilistic model used for the recommendation tool will evaluate the occurrence of a set of sections (a section is a pair of intentions, one of which is source and the other target).
In order to present how the recommendation tool works, a running example based on the map from the beginning of the chapter is followed step by step. It is assumed that the knowledge base of the recommendation tool consists of this map and also other information regarding the frequencies of intentions. When the process participant invokes the tool, the following trace is extracted and given as input: I0 -> I1 -> I2 -> I3. From the knowledge base, the tool could identify that the intentions I1, I2, I5 and I6 could be reached having I3 as source. Therefore, the subsequent step is to calculate the probabilities of each of the following sequences:

I0 -> I1 -> I2 -> I3 -> I1
I0 -> I1 -> I2 -> I3 -> I2
I0 -> I1 -> I2 -> I3 -> I5
I0 -> I1 -> I2 -> I3 -> I6

For example, the calculus of the probability for the last sequence is:

P(I0 -> I1 -> I2 -> I3 -> I6) = P(S01, S12, S23, S36) where S01 is the section corresponding to the transition I0 -> I1, S12 is the section corresponding to the transition I1 -> I2 and so on.

Consequently the probability that the certain sequence of intentions occurs could be written as the probability that all the corresponding sections occur. Applying recursively the chain rule, the formula becomes:

*P(S01, S12, S23, S36) = P(S36 | S01, S12, S23) * P(S01, S12, S23) = P(S36 | S01, S12, S23) * P(S23 | S01, S12) * P(S01, S12) = P(S36 | S01, S12, S23) * P(S23 | S01, S12) * P(S12 | S01) * P(S01)*

This formula can be simplified by using the Bayesian reasoning which looks into how the existence of new evidence impacts the hypothesis. For example, the probability that S36 occurs (I3 -> I6) is only conditioned by the existence of the evidence S23 (I2 -> I3). With other words, a transition from I3 to I6 is possible only if before there was a transition that had as target the intention I3. However, S36 is not conditioned directly by the occurrence of S01 and S12. Considering that a section is only conditioned by its immediate predecessor, the formula becomes:

*P(S01, S12, S23, S36) = P(S36 | S23) * P(S23 | S12) * P(S12 | S01) * P(S01)*

The general formula used for estimating the probability of a sequence's occurrence having as the target intention I is:

P(Sij ^ Sjk ^ Skl ... ^ Smn) = P (Sij | Sjk) * P(Sjk | Skl) * ... * P(Smn)
P(S) = Frequency (S in sequence with target I) / Frequency (sequences with target I) the ratio between the occurrence frequency of S in a sequence having as target the intention I and the total number of sequences that have as target the intention I
P(Sij | Sjk) = P(Sij, Sjk) / P(Sjk) = Min(Frequency (Sij in sequence with target I), Frequency (Sjk in sequence with target I)) / Frequency (Sjk in sequence with target I) the ratio between the minimum of the frequencies of occurrence of Sij and Sjk in a sequence having as target the intention I and the frequency of occurrence of Sjk in a sequence having as target the intention I

In the presented example, a special situation is identified for the first sequence: I0 -> I1 -> I2 -> I3 -> **I1** as the target intention I1 appears also in its list of predecessors. The question that arises is whether to consider that an intention could be preceded by itself in a trace or this trace represents in fact two ways of achieving an intention – one through the sequence I0 -> I1 and the other through the sequence I2 -> I3 -> I1.

Considering the constraint of the Bayesian approach (a hypothesis cannot be an evidence for itself) (Hautaniemi, Korpisaari, & Saarinen, 2000) and invoking Occam's razor (the model with the simple assumptions should be selected), the second explanation is chosen: each intention in the sequence of predecessors must be unique and different from the target intention. However, Dynamic Bayesian Nets (Hautaniemi et al., 2010) are proposed as a future work for supporting the first explanation, which is most probable in real world and compliant with the MAP formalism. In the case of Dynamic Bayesian approach, states are introduced for each variable of the environment. For example, it could be assumed that I1 in the moment of time T1 could be a predecessor of I1 in a future moment of time T2 (T1 < T2) so as to the state is a time's measure.

The implication of the situation exposed above is that the trace must be processed before it is given as input to the recommendation tool so that the intentions found in a sequence are unique. Each intention will have attached a graph used for the conditional probabilistic inference which is created in the moment the map is created, from the initial input, and updated whenever the map is updated based on the partial traces. The algorithm for the creation or update of the graph consists of the following steps:

1. For each intention found in the trace of the ordered intentions given as input to the recommendation tool, extract the sequences of predecessors until the beginning of the trace is reached or one of the predecessors or the target intention appears twice.
2. Apply a transformation function to the set of predecessors to obtain a set of sections.
3. For each section in the set of sections:
   a. If the section already exists in the intention's graph then increment the variable counting its frequency;
   b. If the section is new then add it to the intention's graph and initialize its frequency to 1.

As it can be noticed, the algorithm handles also the situation when a new section is discovered. For instance, the following trace might be provided as input when the process participant uses the recommendation tool: I5 -> I6 -> I3. The current intention being the same as in the previous example, the list of target intentions is I1, I2, I5 and I6. For calculating the confidence factor for the intention I1, the list of sections used as input is {S56, S63, S31}. The graph of I1 is synchronized with the map thus it does not recognize the section S63 which is new. In this case, the longest sequence of sections that could be identified in the graph is used for computing the probability: {S31}. Furthermore, the confidence factor aggregates the obtained probability but also the length of the longest sequence of sections found in the corresponding graph:

ConfidenceFactor = discriminatorProbFactor * Probability + + discriminatorSeqFactor * MaxSequenceLength/InputSequenceLength

discriminatorProbFactor and **discriminatorSeqFactor** are used for the tuning of the function depending on what is more important in the recommendation: the frequency or the length of the sequence.

MaxSequenceLength is the length of the maximal sequence of sections found in the graph.
InputSequenceLength is the length of the sequence of sections given as input.

Finally, summarizing what has been discussed, the algorithm for creating the recommendations is:

1. Discover the current intention based on the trace of intentions given as input.
2. Create the list of intentions that can be directly reached from the current intention in the map.
3. For each target intention that could be reached from the current intention:
    a. Extract the sequences of predecessors until the beginning of the trace is reached or one of the predecessors or the target intention appears for the second time.
    b. Transform the list of intentions in sections.
    c. Search in its corresponding graph the longest sequence of sections that maps on the sequence of sections extracted in the previous step.
    d. Calculate the probability that the longest sequence of sections occurs.
    e. Calculate the confidence factor by considering the probability found in the previous step and the length of the longest sequence of sections that was discovered in the graph.
    f. Update the graph by applying the third step of the previous algorithm. Accordingly, the partial traces used in formulating the recommendations are integrated in the original map influencing the future predictions.

6.3 Artifact's Development

Similar to the intention mining technique, the recommendation tool was developed using Microsoft .NET framework, C# programming language and Visual Studio 2012 as integrated environment.

The map was developed using the following data structures: a generic list with objects of type Intention and a generic dictionary having as key the intention's id and as value a generic list of objects of type Section (the sections for which the intention is the source). The Section class consists of a source intention, target intention and frequency. Besides these data structures, another dictionary is used for storing the graphs for each intention. The class Graph was defined using adjacency lists, maintaining the reachable neighbors, for each node.

The methods that the Map class exposes are presented in Figure 28. Given a trace of intentions first the recommendations are formulated; then the map is updated by updating the list of intentions and the dictionary storing the sections for each intention; finally the Bayesian graphs for each intention are also updated. The confidence factors are calculated based on each graph, the list of sections and the index where the longest sequence of sections that can be found in the graph starts. This index is calculated using the method ExtractMaximalSequence. Several other helper methods are defined such as the extraction of predecessor for a specified intention from the trace and the generation of sections.

```
public void GiveRecommendation(List<Intention> trace)[...]

public void CreateOrUpdateMap(List<Intention> trace)[...]

private void CreateOrUpdateBayesianNets(List<Intention> trace)[...]

private double CalculateRecommendationFactor(Graph graph, List<Section> sections, int beginSeqIndex)[...]

private int ExtractMaximalSequence(Graph graph, List<Section> sections)[...]

private List<List<Intention>> ExtractPredecessorSequences(Intention intention, List<Intention> trace)[...]

private List<Section> GenerateSections(List<Intention> intentions)[...]

private List<Section> GenerateSections(Intention intention, List<Intention> intentions)[...]
```

Figure 28. The implementation of Map class

The parameters, “discriminatorProbabilityFactor” and “discriminatorSequenceFactor”, used in the computation of the confidence factors, are defined in the configuration file. Their values for the testing phase were set to 0.5 but they could be changed by the process administrator as desired with values between 0 and 1.

During this chapter, the third research question was answered. The main implication of implementing flexible processes for agents is the complexity of decision making during process enactment. The created recommendation tool could support the agents in making decisions by offering recommendations at the intentional level, accompanied by a confidence factor reflecting other agents’ behavior. Moreover, the recommendations are created based on the process knowledge, discovered through intention mining. In conclusion, the recommendation tool provides the agents with information that could help them during enactment, thus facilitating their interaction with flexible PAIS.

7 ARTIFACTS’ EVALUATION

In this chapter, I answer the main research question which regards the non-functional evaluation of the proposed artifacts:

To what extent does a recommendation tool based on intention mining improve the process participant’s interaction with flexible process aware information systems?

Several knowledge questions were formulated in the beginning of the thesis with regard to the artifacts and the problem. The majority of the questions regarding the problem were already answered during the previous chapters: the identification of the stakeholders (process participants, process owners), the identification of their goals (the research project's objectives) and the requirements of the artifacts based on the problem (the artifacts' design decisions). The questions focusing on the artifacts - their effects, their requirements satisfaction, their trade-offs, and one question focusing on the problem – the artifacts' contribution, which overlaps with the main research question, will be answered during the following sections.

The evaluation of the artifacts took place in the settings of a case study and it followed the protocol defined in Appendix A. I conducted an experiment with 10 process participants of whom 6 were women and 4 men with age range from 24 to 54, interacting with the Childcare application.

7.1 Intention Mining Technique's Evaluation

The intention mining technique is evaluated following the Confusion matrix approach (Kohavi and Provost, 1998). The Confusion matrix is built around the idea of instance's classification (also known as prediction) made by a classifier system.

In this context, the classifier system is the recommendation tool and the instance is the discovery/existence of an intention. The instance's discovery can be classified as:

- Positive: a certain intention is discovered from a given trace.
- Negative: a certain intention is not discovered from a given trace.

These predictions are compared to the reality (the actual classification). Similar to what it has been said before, the instance's existence can be classified in reality as:

- Positive: the process participant has a certain intention.
- Negative: the process participant does not have a certain intention. There are two types of negative instances in this situation:
    - An intention does not exist and it cannot be recorded;
    - An intention does not exist but it is recorded.

Four types of predictions are further identified (Table 10):

- **True positive**: a correct prediction that the instance's existence is positive meaning that the discovered intention using the intention mining technique is verbalized directly or indirectly by the process participant. An indirect verbalization of the intention is when the process participant does not state it clearly, but the interviewer identifies it from the participant' speech.
- **False positive**: an incorrect prediction that the instance's existence is positive meaning that the discovered intention using the intention mining technique is not verbalized by the process participant.
- **False negative**: an incorrect prediction that the instance's existence is negative meaning that an intention is not discovered using the intention mining technique even if it is directly or indirectly verbalized by the process participant.
- **True negative**: a correct prediction that the instance's existence is negative meaning that an intention is not discovered using the intention mining technique and it is not verbalized directly or indirectly by the process participant. Since the process participant does not have

an intention and does not act according to it, there are no traces based on which the intention can be discovered by the intention mining technique. Therefore, the number of true negative instances is always **0**.

		Predicted	
		Negative: a certain intention is not discovered	**Positive**: a certain intention is discovered
Actual	**Negative**: the process participant does not have a certain intention	**True negative** #TN = number of true negative predictions	**False positive** #FP = number of false positive predictions
	Positive: the process participant has a certain intention	**False negative** #FN = number of false negative predictions	**True positive** #TP = number of true positive predictions

Table 10. Confusion matrix (Kohavi and Provost, 1998)

Based on the confusion matrix, the following measures are calculated and used in the evaluation:

1. **Accuracy** is the proportion of the total number of predictions that were correct and it is determined using the equation:

$$\text{Accuracy} = (\#TN + \#TP) / (\#TN + \#TP + \#FP + \#FN)$$

2. **Precision** is the proportion of the predicted positive cases that were correct and it is determined using the equation:

$$\text{Precision} = \#TP / (\#TP + \#FP)$$

In Appendix D, the data for the intention mining evaluation is presented. For each trace, there is a table consisting of the intentions discovered by using the intention mining technique and the intentions declared by the process participants during the interviews. Moreover, the cardinals of the classes belonging to the Confusion matrix are also calculated per trace (Table 11).

Trace number	#TP	#FP	#TN	#FN
1	10	5	0	0
2	11	4	0	0
3	10	6	0	0
4	11	4	0	0
5	12	3	0	0

6	12	7	0	1
7	7	5	0	1
8	10	4	0	0
9	13	4	0	1
10	9	5	0	0

Table 9. The cardinal of the Confusion matrix classes per trace

Accuracy and precision are calculated per trace further (Table 12). The average of each measure is also computed to give a global evaluation of the intention mining technique.

Trace number	Accuracy	Precision
1	0.66	0.66
2	0.73	0.73
3	0.62	0.62
4	0.73	0.73
5	0.8	0.8
6	0.6	0.63
7	0.53	0.58
8	0.71	0.71
9	0.72	0.76
10	0.64	0.64
	Average Accuracy: 0.67	**Average precision: 0.69**

Table 10. The accuracy and precision per trace

These results show that the average accuracy and precision of the intention mining technique are satisfactory but their values per trace vary quite significantly: from 0.53 to 0.8 for the accuracy, and from 0.58 to 0.8 for the precision.

For getting more insights in what can be improved regarding the intention mining technique, I thoroughly analyzed each trace and I noticed several recurring issues:

- Some intentions were discovered by the technique even if the activities behind them were not intended for that discovered intention. For example, every time when the experiment started and the process participant clicked on the tab **Familie** (Appendix B), a list of children was printed. This mined intention was "*Read children*", but the participant had in fact another intention: to start the experiment. The same thing happened in the situation when the process participant was curious about the application's functionalities and he started to navigate into tabs but without the actual intention of reading an entity or an entity list. The observation here is that some intentions might not be in the mind of the process participant but they belong to a higher level intention, usually a non-functional one ("Explore the application").

- Some intentions were discovered by the technique but in reality they were triggered by the system when another intention was realized, and not by the process participant. Therefore, these intentions should be incorporated in the intention that triggered them. This is the situation with the intention "*Create child*" which frequently was followed by "*Create child picker*" (an empty entity was instantiated), "*Create child – child picker link*" (an empty entity was instantiated) and "*Read child*".

In conclusion, the functional requirements of the intention mining technique were completely satisfied as it was proved by its usage without errors during the experiment. The non-functional requirements of the artifact were satisfied but the precision of the results had an average of 0.69. The intention mining technique can be used for mining the intentional process but a supplementary review is required considering that the results are not completely accurate (the average for this set of traces is 0.67).

7.2 Recommendation Tool's Evaluation

The functional and non-functional requirements of the recommendation tool were evaluated. Unit tests were used for validating the artifact's functionalities. The setup for the non-functional evaluation was problematic because the tool was not integrated in the Childcare application. Consequently, recommendations could not be provided live, during the enactment.

However, the motivation for creating the tool was that it could improve the support for decision making because it formulates recommendations based on the experience of other process participants at the intentional level. Therefore, the non-functional evaluation of the recommendation tool was reduced to the following parts:

- The non-functional evaluation of the intention mining technique as the quality of the produced output (which is the input for the recommendation tool) influences the quality of the recommendations. This was already described in the previous section.
- The analysis of the perceived contribution of the recommendations given as intentions and confidence factors on decision making support by the process participants. This was realized based on a questionnaire having various conceptual scenarios inspired from the Childcare application, requiring the process participants to make decisions (Appendix C).

The first part of the questionnaire focused on identifying the background of the participants in working with software and especially with software that provides support for recommendations. Only 3 of 10 participants mentioned that the tools they are working with provide support for recommendations. Nevertheless, the majority of the participants (7 of 10) agreed that they would consider the recommendations provided by tools in performing their work activities if they were provided, while the others did not express their opinion (Appendix E).

The first experiment's hypothesis guiding the evaluation of the recommendation tool was:

Hypothesis 1	**H0:** The recommendations given as intentions do no improve the support for decision making by supporting criteria identification in comparison to the recommendations given as activities.
	H1: The recommendations given as intentions improve the support for decision making by supporting the criteria identification.

In the beginning, a scenario was presented to the participants and they were asked to identify what they should do next. In general, the participants identified the high level intention – to update the planning for the child, without problems. Most of them gave details about the steps of the process but only with regard to the update of the child's profile but not to the planning's update (except of 3 participants who mentioned this). After the first set of recommendations was given, most of the participants chose the option that was aligned with their previously identified intention except from two: one changed his intention from updating the child's profile to updating the planning and the other stated that his decision was based on the confidence factor's value. When the process participants were asked if it would help them to know the intention behind these actions before making the decision, 9 of 10 participants agreed motivating the answer as follows:

> "The previous recommendations are very general and they lack in information as one can't see the actions or the purpose of following them" (Participant no. 8)

- The recommendation as intention would help him to understand what he exactly needs to do.
- The recommendation as intention would help him to clarify the actions he should take and thus helping him to make the decision.
- The recommendation as intention could help him to validate an intention adopted in advance with regard to a certain situation.

> "The intention can guide me to do the right action"
> "It helps to make the right decision"
> "It consolidates my opinion"

One participant disagreed with the added value of intention by invoking the efficiency in following actions without reasoning about intentions (step by step guidance).
After the intention behind this scenario was revealed, the same proportion of participants (9/10) agreed that the decision is easier to be made if the intention is known. While motivating their answer, several other reasons were added to the previous list:

- The recommendation as intention leaves enough flexibility to enact the process in the desired way while also providing some guidance.
- The recommendation as intention gives information about the context.

Consequently, the first hypothesis was verified and the recommendation as intention proved to improve the decision making by supporting the criteria selection. The majority of the experiment's participants (9/10) agreed that the intention is helpful when deciding among several activities. Thus, the recommendation as intention supports the criteria selection either by its realization (when the process participants adopts the suggested intention and make the decision according to it) or by its validation (when the process participant checks if the suggested intention is the same as the one he already formulated in his head).
The second and third experiment's hypotheses guiding the evaluation of the recommendation tool (Appendix A) are:

Hypothesis 2	**H0:** The recommendations given as intentions do no improve the support for decision making by supporting the alternatives formulation in comparison to the recommendations given as activities.
	H1: The recommendations given as intentions improve the support for decision making by supporting the alternatives formulation.
Hypothesis 3	**H0**: The recommendations given as intentions do no improve the support for decision making by supporting the alternatives analysis in comparison to the recommendations given as activities.
	H1: The recommendations given as intentions improve the support for decision making by supporting the alternatives analysis.

The aim of the next scenario was to compare the decision making support when recommendations were given as intentions and then as activities.
7 of 10 participants found the set of recommendations given as intentions helpful for supporting the decision making while 3 disagreed with it, stating the following reasons: two preferred a step by step recommendation without being necessary to reason about the intentions and one said he found it hard to make the decision because there were too many recommendations in the set. The same proportion of participants (7/10) disagreed that the set of recommendations given as activities was helpful for

supporting the decision making. The main reason was that they did not provide enough information and only an experienced process participant could choose easier in this situation.

Analyzing all the data, it was noticed that the majority of the participants prefer to have support in interacting with the application as they admitted there were many possible ways to perform the tasks which was confusing. Therefore, the formulation of the alternatives improves the support for decision making by identifying possible options. Moreover, the recommendations as intentions are preferred to those as activities. This depends on the behavior of the process participants too because there were some that preferred a step by step guidance. However, this is not applicable for flexible process aware information systems where decision making is a mandatory part of the process enactment.

"The intention can guide me to do the right action" (Participant no. 1)
"It helps me to make the right decision" (Participant no. 2)
"It consolidates my opinion" (Participant no. 7)

Finally, the last guiding hypothesis of the experiment was with regard to the confidence factors as part of the recommendations (Appendix A):

Hypothesis 4	**H0:** The confidence factors included in the recommendations do not improve the support for decision making.
	H1: The confidence factors included in the recommendations improve the support for decision making.

The participants were informed about the meaning of the confidence factors that they aggregated the other participants' behavior for the same situation. 6 of 10 participants disagreed with the numerical values attached to each recommendation influenced their decision. The main invoked reason was that there was no re-assurance that other participants enacted the process more efficient or more effectively, in order to follow their behavior. Nevertheless, the other 4 participants that agreed with the usefulness of the confidence factors mentioned that their decision was influenced completely by this (following the crowd) or partially (in checking if the crowd reasoned similarly). Consequently, this hypothesis cannot be verified based on the existing data.

"I consider the second set of recommendations lacks in information, there are only simple technical steps without explaining their purpose." (Participant no. 8)
"You have to know well the system to choose this set of recommendations." (Participant no. 9)

In conclusion, the evaluation of this artifact showed that a recommendation tool incorporating the intentional process can support the decision making by supporting the criteria selection, alternatives formulation and alternatives analysis. Accordingly, the main research question and the contribution question were answered: the interaction of process participants with flexible process aware information system is improved as the key activity implied by the flexible process enactment – decision making, is supported in an effective manner. The artifact's effects (the effect of recommendations given as intentions on decision making support) and the artifact's trade-off were discussed during this section, too.
Furthermore, a couple of critical notes were taken and considered for future integration in the recommendation tool: the provided set of recommendations should be limited to smaller number in order to ease the reasoning or the meaning of confidence factors should be explained better in order to be used in the reasoning. A better understanding of the confidence factors could help the process participant to select for reasoning a smaller set of recommendations from a large one. This solution might be preferred to excluding possibilities of process execution as it gives an informed and more complete picture of the flexible process (with all its exceptions).

8 CONCLUSIONS

This research project was set out to investigate how the interaction of process participants with flexible process aware information systems could be improved. The implementation of flexible processes within the information systems has significant implications for process participants requiring them to have a better ability of interpreting the process' context, making decision and validating decisions, in comparison to the traditional workflow systems. Consequently, an efficient and effective flexible process enactment depends on the interaction of process participants with PAIS. While this problem was already discussed in the literature, the solutions consisting in support/guidance tools integrating process mining have proved useful to supporting decision making to a certain extent. Most of the approaches provide recommendations /guidelines based on a very low level description of the process model, which is mined with the available techniques.
Therefore, the goal of this research project was to create an improved tool for supporting process participants during flexible process enactment which offered recommendation on a higher level description of the process model: the intentional facet. As the knowledge gathered through process mining captures more accurately the process model as it is in the real life, it was decided that the intentional process model should be discovered automatically from event logs. This leaded to the sub-goal of this research project: to create an intention-mining technique that discovers the intentional

facet of a process in terms of intentions and strategies modeled with the MAP formalism.

The research project sought to answer progressively the main research question: "***To what extent does a recommendation tool based on intention mining improve the process participant's interaction with flexible process aware information systems?***" through the investigation of three sub-questions:

1. How does the implementation of flexible processes in process aware information systems impact the process participants?
2. How can the intentional process, behind the interaction of process participants with process aware information systems, be mined?
3. How can the process participant's activities impacted by the implementation of flexible processes be supported through a recommendation tool based on intention mining?

Design science research was the main research method supported also by case study research (for evaluation) and literature review. The findings for each research question were presented in their specific chapter: the first research sub-question – "*Chapter 4. Agent-centered Analysis in Flexible PAIS*"; the second research sub-question – "*Chapter 5. Intention-mining technique*"; the third research sub-question – "*Chapter 6. Intention-based Recommendation Tool*". The findings are synthesized further to answer the three research sub-questions:

1. Several categories of process flexibility were identified in a systematic literature study: process variability, process under-specification, process adaptability, process evolution and data driven approach. The implications of ensuring process flexibility trough each of these categories were discussed for both process participant and process administrator as their responsibilities overlap in some circumstances such as late-modeling. The agency characteristic of process participants is prominently emphasized during the enactment of flexible processes and implies three key activities: context interpretation, decision making and decision validation.
2. An unsupervised intention mining technique was created and validated in a case study using the Childcare system. The design and development of the artifact were focused on three categories:
    a. **Input:** the identification of the relevant data and the integration of a mechanism for collecting the event logs in the case system.
    b. **Algorithm:** by analyzing the traces and the theory regarding intentionality and process mining, several design decisions were adopted and applied resulting in a five-steps algorithm (step 1 – the classification of events per process participant; step 2 – the ordering of events by timestamp; step 3 – the application of syntactic and correlation's trends analysis to form the intentional clusters; step 4 – the application of semantic analysis for extracting the intention from each intentional cluster; step 5 – the aggregation at the intentional and operational levels to obtain the process models).

c. **Output:** the evaluation of the mined intentions was conducted in an experiment with 10 participants interacting with the Childcare system (Appendix A, Appendix B). An artifact's precision of 0.69 was obtained for the collected traces. Several issues were also identified such as the classification of some actions under certain intentions while they were belonging to a higher level (non-functional) intention and the discovery of certain intentions that were in reality parts of another intention.

3. A recommendation tool incorporating the results of intention mining and following the Bayesian epistemology was created to support process participants in making decisions during enactment. Recommendations were formulated as intentions and confidence factors (a numerical value aggregating the behavior of other process participants being in the same or similar situation during enactment). The evaluation revealed that the recommendations provided at the intentional level improved the support for decision making by improving the support for criteria selection, alternatives formulation and alternatives analysis in comparison to recommendation given as activities. However, no hard statement could be made about the confidence factors with regard to the support for decision making as 4 of 10 participants admitted they considered them in their reasoning while the others did not.

The main contribution of this research project was the innovation brought to the process mining area – mining the intentional process model in an unsupervised manner, and the innovation brought to the flexible process aware information systems area – creating a recommendation tool prototype that gives recommendation based on the intentional process model discovered automatically from event logs.

9 FUTURE WORK

As a direct consequence of the project's novelty, several limitations were encountered and needs to be considered:

- Some design decisions of the intention mining technique are general while others are case specific. In order to validate the artifact in other situations, its implementation must be partially modified according to the new case study (the data that is collected, the function that calculates the syntactic correlation, the ontology used for semantic analysis).
- The recommendation tool was created based on the assumption that an intention cannot influence its realization in the same trace (an intention cannot be in its list of predecessors when creating the probability graphs). However, in reality this can happen but it was a simplification to comply with the adopted probabilistic model.
- The evaluation of the artifacts was realized for one case study with 10 participants. According to Yin (2009) a more accurate evaluation should include at least 3 case studies.

These limitations lead to the formulation of the future research directions. Firstly, more case study research should be conducted to allow further assessment of the created artifacts.

Secondly, the intention mining technique could be improved, as demonstrated in the evaluation phase, to mine more accurately the intentions. Then, the semantic analysis should be supported by ontology and the semantic annotation of the event logs which could also enable the mining of strategies, not only intentions, and the mining of non-functional intentions. Moreover, other machine learning algorithms for clustering could be explored, such as self organizing maps or genetic algorithms.
The intention mining technique in its current form requires several adaptations for being re-used with another application. The adaptations are: the selection of the logged data relevant for the syntactic analysis, the correlation function must be redefined according to the selected data and the hierarchy of intentions for semantic analysis must be adapted as well. Therefore, these activities and changes of the intention-mining technique triggered by specific cases should be formalized in a method and supported by a tool so as to the next adaptation is easier.
Thirdly, the recommendation tools should be extended with an inference mechanism based on Dynamic Bayesian Network which is a more suitable probabilistic method for processes. This would allow an intention to be in its list of predecessors when calculating the confidence factors and predicting the next intentions in a trace. The prototype should be released in a stable version and integrated in a process aware information system to allow its run-time evaluation.
Driven by the problematic support of process participants when interacting with flexible process aware information systems, the research made a step further in process mining by proposing the discovery of a process from a new perspective using unsupervised machine learning. The intentional process model was integrated in a recommendation tool, which after the evaluation in a case study, demonstrated its contribution to problem solving. To sum up with, I considered that the largest contribution of this research is the thorough study of the intentionality in the process enactment and its integration with process mining.

ACKNOWLEDGEMENTS

First, I would like to express my deep gratitude to my supervisor, Sjaak Brinkkemper who made it possible for me to conduct the thesis research at the Université Paris 1 Panthéon-Sorbonne.
Then, I would like to express my very great appreciation to my daily coordinators, Rébecca Deneckère and Charlotte Hug for their patient guidance, enthusiastic encouragement and useful feedback of this research work. I would also like to thank to the other members of Centre de Recherche en Informatique for their constructive suggestions and support during these months.
Special thanks should be given to Kevin Vlaanderen for his insightful feedback and the help with the organizational aspects of my graduation.

I would also like to thank to Marc Remmers and Vincent van Nimwegen for believing in this project and accepting that 42windmills was the case company for my research. Finally, I would like to thank to my friends and family for their great support and patience during the last 10 months.

REFERENCES

Aalst, W. M. P. van der (2008). Decision support based on process mining. In *Decision Support Systems* (Vol. 1, pp. 637-657). Springer Berlin Heidelberg.

Aalst, W. M. P. van der (2011). *Process Mining: Discovery, Conformance and Enhancement of Business Processes*. Springer Berlin Heidelberg.

Aalst, W. M. P. van der, Pesic, M., & Schonenberg, H. (2009). Declarative workflows: Balancing between flexibility and support. *Computer Science Research and Development*, *23*(2), 99-113. Springer. Retrieved from http://www.springerlink.com/index/10.1007/s00450-009-0057-9

Aalst, W. M. P. van der, Weske, M., & Grunbauer, D. (2005). Case handling: a new paradigm for business process support. (Elsevier, Eds.)*Data & Knowledge Engineering*, *53*(2), 129-162. Elsevier. Retrieved from http://linkinghub.elsevier.com/retrieve/pii/S0169023X04001296

Adams, M., Hofstede, A. H. M., Aalst, W. M. P. van der, & Edmond, D. (2007). Dynamic, Extensible and Context-Aware Exception Handling for Workflows. (R. Meersman & Z. Tari, Eds.)*International Conference on Cooperative Information Systems LNCS 4803*. Springer. Retrieved from http://www.springerlink.com/content/y386117j42352620

Adams, M., Hofstede, A. H. M., Edmond, D., & Aalst, W. M. P. van der (2005). Facilitating Flexibility and Dynamic Exception Handling in Workflows through Worklets. (O. Belo, J. Eder, O. Pastor, & J. Falcao E Cunha, Eds.)*Knowledge Acquisition*, *161*(May), 45-50. FEUP. Retrieved from http://eprints.qut.edu.au/7984/

Agrawal, R., Gunopulos, D., & Leymann, F. (1998). Mining Process Models from Workflow Logs. In H.-J. Schek, G. Alonso, F. Saltor, & I. Ramos (Eds.),*Advances in Database Technology — EDBT98* (Vol. 1377, pp. 467-483). Springer. Retrieved from http://www.springerlink.com/index/0836165X01666L40.pdf

Aguilar-Saven, R. S. (2004). Business process modelling: review and framework. *International Journal of Production Economics*, *90*(1), 129-49.

Bentellis, A., & Boufaida, Z. (2009). Objective Based Flexible Business Process Management Using the Map Model. *Information Technology Journal*. *8*(1), 495-503.

Bovens, L., & Hartmann, S. (2003). *Bayesian Epistemology*. In Zalta, E. N. (Ed.)*The Stanford Encyclopedia of Philosophy* (Vol. 114, pp. 159-178). Clarendon Press. Retrieved from http://plato.stanford.edu/entries/epistemology-bayesian

Bratman, M. E. (1999). *Intention, Plans, and Practical Reason*. Cambridge University Press.

Brinkkemper, S. (1996). Method engineering : engineering of information methods and tools. *Information and Software Technology*, *38*(4), 275-280. Retrieved from http://doc.utwente.nl/18012/1/Brinkkemper96method.pdf

Burkhart, T., & Loos, P. (2010). Flexible business processes - evaluation of current approaches. In *Proceedings: Multikonferenz Wirtschaftsinformatik (MKWI-2010)*. Gttingen, Germany.

Carberry, S. (2001). Techniques for Plan Recognition. *User Modeling and UserAdapted Interaction*, *11*(1-2), 31-48. Kluwer Academic Publishers. Retrieved from http://dx.doi.org/10.1023/A:1011118925938

Chen, Z., Lin, F., Liu, H., Liu, Y., Ma, W.-Y., & Wenyin, L. (2002). User Intention Modeling in Web Applications Using Data Mining. *World Wide Web Internet And Web Information Systems*, *5*(3), 181-191. Springer. Retrieved from http://www.springerlink.com/index/UW00R45713480U07.pdf

Cohen, P. R., & Levesque, H. (1990). Intention is choice with commitment. *Journal of Artificial Intelligence, 42*(2), 213-261.

Cook, J. E., & Wolf, A. L. (1998). Discovering models of software processes from event-based data. *ACM Transactions on Software Engineering and Methodology*, 7(3), 215-49.

Croft, W. B., & Lefkowitz, L. (1984). Task support in an office system. *ACM Transactions on Office Information Systems*, *2*(3), 197-212.

Dadam, P., & Reichert, M. (2009). The ADEPT project: a decade of research and development for robust and flexible process support. *Computer Science Research and Development*, *23*(2), 81-97. Springer. Retrieved from http://www.springerlink.com/index/10.1007/s00450-009-0068-6

Davenport, T. H. (1993). *Process innovation: reengineering work through information technology*. (H. B. S. Press, Ed.)*Journal of Engineering and Technology Management* (Vol. 11, p. 337). Harvard Business School Press.

Deneckere, R., Kornyshova, E., & Rolland, C. (2009). Enhancing the guidance of the intentional model “MAP”: Graph theory application. *2009 Third International Conference on Research Challenges in Information Science*.

Dorn, C., Burkhart, T., Werth, D., & Dustdar, S. (2010). Self-adjusting Recommendations for People-Driven Ad-Hoc Processes. *Proceedings of the 8th International Conference on Business Process Management BPM 2010*(pp. 327-342). Retrieved from http://link.springer.com/chapter/10.1007/978-3-642-15618-2_23

Dumas, M., Aalst, W. M. van der, & Hofstede, A. H. ter (2005). *Process Aware Information Systems: Bridging People and Software Through Process Technology*. Wiley-Interscience.

Eckermann, O., & Weidlich, M. (2011). Flexible Artifact-Driven Automation of Product Design Processes Enterprise. In Halpin, T., Nurcan, S., Krogstie, J., Soffer, P., Proper, E., Schmidt, R., & Bider, I. (Eds.) *Business-Process and Information Systems Modeling* (Vol. 81, pp. 103-117). Springer Berlin Heidelberg.

Ellis, C. A., & Kim, K. (2007). Process aware information systems: a human centered perspective. *AsiaPacific Web Conference International Conference on WebAge Information Management APWebWAIM ’07* (Vol. 4505, pp. 39-49). LNCS 4505, Springer-Verlag. Retrieved from

Ellis, C. A., & Nutt, G. J. (1980). Office Information Systems and Computer Science. *ACM Computing Surveys*, *12*(1), 27-60. ACM New York, NY, USA. Retrieved from http://portal.acm.org/citation.cfm?doid=356802.356805.

Fernandes, A., Ciarlini, A. E. M., Furtado, A. L., Hinchey, M. G., Casanova, M. A., & Breitman, K. K. (2007). Adding flexibility to workflows through incremental planning. *Innovations in Systems and Software Engineering*, *3*(4), 291-302. Springer. Retrieved from http://dx.doi.org/10.1007/s11334-007-0035-y

Fishbein, M., & Ajzen, I. (1975). *Belief, Attitude, Intention, and Behavior: An Introduction to Theory and Research*. Reading, MA: Addison-Wesley.

Gelinas, U. J., Sutton, S. G., & Fedorowicz, J. (2004). *Business processes and information technology*. Cincinnati, Ohio : South-Western/Thomson Learning.

Georgakopoulos, D., Hornick, M., & Sheth, A. (1995). An overview of workflow management: From process modeling to workflow automation infrastructure.*Distributed and Parallel Databases*, *3*(2), 119-153. Springer. Retrieved from http://www.springerlink.com/index/10.1007/BF01277643

Hallerbach, A., Bauer, T., & Reichert, M. (2010). Capturing variability in business process models: the Provop approach. (R. Messnarz & M. Biro, Eds.)*Journal of Software Maintenance and Evolution Research and Practice*, *22*(6-7), 519-546. John Wiley & Sons, Ltd. Retrieved from http://doi.wiley.com/10.1002/smr.491

Hammer, M. (2010). What is Business Process Management? . In Brocke, J., & Rosemann, M. (Eds.), *Handbook on Business Process Management* (Vol. 1, pp 3-16). Springer Berlin Heidelberg.

Hammer, M., & Champy, J. (1993). *Reengineering the Corporation: A Manifesto for Business Revolution*. (M. Hammer, Ed.)*Business Horizons* (Vol. 36, pp. 90-91). Harper Business. Retrieved from http://www.amazon.co.uk/dp/0887306403

Hammer, M., Howe, W. G., Kruskal, V. J., & Wladawsky, I. (1977). A very high level programming language for data processing applications.*Communications of the ACM*, *20*(11), 832-840. Retrieved from http://portal.acm.org/citation.cfm?id=359863.359886

Harris, R. (2012). *Introduction to decision making*. Retrieved from http://virtualsalt.com/crebook5.htm

Hautaniemi, S. K., Korpisaari, P. T., & Saarinen, J. P. P. (2000). Target identification with dynamic hybrid Bayesian networks. In Serpico, S. B. (Ed.) *Image and Signal Processing for Remote Sensing VI* (Vol. 4170, pp. 92-103). Bellingham, WA.

Hevner, A., & Chatterjee, S. (2010). Design Science Research Frameworks. In A. Hevner & S. Chatterjee (Eds.), *Design Research in Information Systems* (Vol. 22, pp. 23-31). Springer US. Retrieved from http://www.springerlink.com/index/10.1007/978-1-4419-5653-8.

Hevner, A. R., March, S. T., Park, J., & Ram, S. (2004). Design Science in Information Systems Research. (A. R. Hevner & S. Chatterjee, Eds.) *MIS Quarterly*, *28*(1), 75-105. JSTOR. Retrieved from http://www.jstor.org/stable/25148625.

Holt, A. W. (1985). *Coordination technology and Petri nets*. In Rozenberg, G. (Ed.), *Advances in Petri Nets*. Paper presented at the 6th European Workshop on Applications and Theory in Petri Nets (pp. 278-296). Springer-Verlag, London, UK.

Hutter, M. (2007). On Universal Prediction and Bayesian Confirmation.*07091516*. Retrieved from http://arxiv.org/abs/0709.1516

Khodabandelou, G., Hug, C., Deneckere, R., and Salinesi, C. (2013). Supervised Intentional Process Models Discovery using Hidden Markov Models. In *Proceedings of the 7th Conference on Research Challenges in Information Science.* IEEE.

Kim, K., Choi, I., & Park, C. (2011). A rule-based approach to proactive exception handling in business processes. *Expert Systems with Applications*,*38*(1), 394-409. Elsevier Ltd. Retrieved from http://linkinghub.elsevier.com/retrieve/pii/S0957417410005828

Kohavi, R. &, Provost, F. (1998). On Applied Research in Machine Learning. *Applications of Machine Learning and the Knowledge Discovery Process, 30*(2), 127-132.

Koliadis, G., & Ghose, A. (2006). Relating Business Process Models to Goal-Oriented Requirements Models in KAOS. *Proceedings of the PacificRim Knowledge Acquisition Workshop PKAW2006*, *4303*(8), 25-39. Springer Berlin Heidelberg. Retrieved from http://www.springerlink.com/content/g8154846h5410418.

Kornyshova, E., Deneckère, R., & Rolland, C. (2011). Method Families Concept: Application to Decision-Making Methods. In Halpin, T. A., Nurcan, S., Krogstie, J., Soffer, P., Proper, E., Schmidt, R., & Bider, I. (Eds.) *Enterprise, Business-Process and Information Systems Modeling* (Vol. 1, pp.413-427). Springer.

Kornyshova, E., Deneckère, R., & Salinesi, C. (2007). Method Chunks Selection by Multicriteria Techniques: an Extension of the Assembly-based Approach. In Ralite, J., et al. (Eds) *Situational Method Engineering: Fundamentals and Experiences* (Vol. 244, pp. 64-78). Springer Boston.

Kueng, P., & Kawalek, P. (1997). Goal-Based Business Process Models: Creation and Evaluation. *Business Process Management*, *3*(1), 17-38.

Lacey, N., Hexmoor, H., and Beavers, G. (2002). Planning at the intention level. In *Proceedings of the 15th International Florida Artificial Intelligence Research Society Conference* (pp. 8-12). AAAI Press.

Lee, J. (1993). Goal-based process analysis: a method for systematic process redesign. *SciencesNew York* (pp. 196-201). ACM. Retrieved from http://portal.acm.org/citation.cfm?id=168577.

Lee, J., Kim, H., Seo, W., Kim, K., & Kim, C. (2011). Condition-based process patterns for modeling of human processes in knowledge-intensive business services. *Expert Systems with Applications*, *38*(4), 4025-4038.

Levy, Y., & Ellis, T. J. (2006). A Systems Approach to Conduct an Effective Literature Review in Support of Information Systems Research. *Science Journal*, *9*(1), 181-212. Informing Science Institute, 131 Brookhill Court, Santa Rosa, CA, 95409, USA. Retrieved from http://www.scs.ryerson.ca/aferworn/courses/CP8101/CLASSES/ConductingLiteratureReview.pdf.

Li, P., & Du, Y. (2009). Modeling and Design for Dynamic Workflows Based on Flexible Activities. *Information Technology Journal*, *8*(5), 750-756.

Lu, R., Sadiq, S., & Governatori, G. (2009). On managing business processes variants. *Data & Knowledge Engineering*, *68*(7), 642-664. Elsevier B.V. Retrieved from http://linkinghub.elsevier.com/retrieve/pii/S0169023X09000196

Malone, T. W., Crowston, K., & Herman, G. A. (2003). *Organizing Business Knowledge: The MIT Process Handbook*. (T. W. Malone, K. Crowston, & G. A. Herman, Eds.)*Journal of Product Innovation Management* (Vol. 22, pp. 218-219). The MIT Press. Retrieved from http://doi.wiley.com/10.1111/j.0737-6782.2005.116_4.x.

Maria, V., Werneck, B., Padua, A. D., Oliveira, A., Cesar, J., Leite, P., São, R., et al. (2009). Comparing GORE Frameworks : i-star and KAOS. In C. P. Ayala, C. T. L. L. Silva, & H. Astudillo (Eds.), *Framework*.

Markov, I., & Kowalkiewicz, M. (2008). Linking Business Goals to Process Models in Semantic Business Process Modeling. *2008 12th International IEEE Enterprise Distributed Object Computing Conference*. Ieee. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4634785

Marrella, A. & Mecella, M. (2011). Continuous Planning for Solving Business Process Adaptivity. In Halpin, T., Nurcan, S., Krogstie, J., Soffer, P., Proper, E., Schmidt, R. & Bider, I. (Eds.) *Enterprise, Business-Process and Information Systems Modeling* (Vol. 81, pp. 118-132). Springer Berlin Heidelberg.

Mitroff, I. (1974). *The Subjective Side of Science: A Philosophical Inquiry into the Psychology of the Apollo Moon Scientists*. Elsevier, Amsterdam (reissued in 1984 by Intersystems Publishers, Seaside, California).

Muehlen, M. zur (2004). *Workflow-based Process Controlling: Foundation, Design and Application of workflow-driven Process Information Systems*. Logos, Berlin.

Object Management Group (2004). Business Process Modeling Notation (BPMN) Version 1.0. *Business*. O.M.G. Retrieved from http://www.omg.org/spec/BPMN/2.0/

Peffers, K., Tuunanen, T., Gengler, C. E., Rossi, M., Hui, W., Virtanen, V., & Bragge, J. (2006). The design science research process: a model for producing and presenting information systems research. In S. Chatterjee & A. Hevner (Eds.), *DESRIST 2006* (Vol. 24, p. 83–106). Claremont Graduate University.

Pesic, M., Schonenberg, H., & Aalst, W. M. P. van der (2007). DECLARE: Full Support for Loosely-Structured Processes. In Spies, M., & Blake, M. B. (Eds.) *11th IEEE International Enterprise Distributed Object Computing Conference EDOC 2007*. Ieee. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=4384001

Petri, C. A. (1962). *Kommunikation mit Automaten*. Unpublished doctoral dissertation, Institut fur instrumentelle Mathematik, Bonn, Germany.

Petrusel, R. (2012). Aggregating Individual Models of Decision-Making Processes. In Ralyté, J., Franch, X., Brinkkemper, S., & Wrycza, S. (Eds.) *Advanced Information Systems Engineering* (Vol. 7328, pp. 47-63). Springer Berlin / Heidelberg. Retrieved from http://dx.doi.org/10.1007/978-3-642-31095-9_4.

Petrusel, R., & Stanciu, P. L. (2012). Making Recommendations for Decision Processes Based on Aggregated Decision Data Models. In *Business Information Systems* (Vol. 117, pp. 272-283). Springer Berlin / Heidelberg.

Pohl, K., Dmges, R., and Jarke, M. (1994). Decision oriented process modeling. In *Proceedings of the 9th Workshop on Intelilgent Software Process* (pp. 124-128). IEEE Computer Society Press.

Polyvyanyy, A., & Weske, M. (2009). Hypergraph-based Modeling of Ad-Hoc Business Processes. (W. M. P. Van Der Aalst, J. Mylopoulos, N. M. Sadeh, M. J. Shaw, C. Szyperski, D. Ardagna, M. Mecella, et al., Eds.)*Business*, 278-289. Springer. Retrieved from http://www.springerlink.com/index/LMJP77656228871V.pdf

Porter, M. E. (1985). *Competitive Advantage: Creating and Sustaining Superior Performance*. (Anonymous, Eds.)*New York* (Vol. 1, p. xviii, 557). Free Press. Retrieved from http://books.google.com/books?id=_Fu1AAAAIAAJ&pgis=1.

Prakash, N., & Rolland, C. (2006). Systems Design for requirements expressed as a map. *Emerging Trends and Challenges in Information Technology Management* (Vol. 1, pp. 501-504). Mehdi Khosrow-Pour.

Ralyte, J. (2012). Applying transdisciplinarity principles in the information services co-creation process. In *Research Challenges in Information Science (RCIS), 2012 Sixth International Conference on* (pp. 1-11). IEEE.

Reichert, M., & Dadam, P. (1998). ADEPTflex—Supporting Dynamic Changes of Workflows Without Losing Control. *Journal of Intelligent Information Systems*, *10*(2), 93-129. Springer Netherlands. Retrieved from http://www.springerlink.com/content/w852302k53v81700/

Reichert, M., & Weber, B. 2012. *Enabling Flexibility in Process-Aware Information Systems - Challenges, Methods, Technologies*. Springer Berlin-Heidelberg.

Rolland, C., Prakash, N., & Benjamen, A. (1999). A Multi-Model View of Process Modelling. *Requirements Engineering*, *4*(1), 169-187. Springer-Verlag London Limited.

Rosemann, M., & Vom Brocke, J. (2010). The Six Core Elements of Business Process Management. In J. Vom Brocke & M. Rosemann (Eds.), Handbook on Business Process Management 1 (pp. 107-122). Springer Berlin Heidelberg.

Russell, N. C. (2007). Foundations of Process-Aware Information Systems. (M. Dumas, W. M. P. Van Der Aalst, & A. H. M. Ter Hofstede, Eds.)Language. Queensland University of Technology. Retrieved from http://www.yawlfoundation.org/documents/RussellThesisFinal.pdf.

Russell, N., Hofstede, A. H. M., & Edmond, D. (2004). Workflow resource patterns. *Business*, *3520*(5446), 13–17. Citeseer. Retrieved from http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.77.8969&rep=rep1&type=pdf

Scheer, A-W (1994). Business Process Engineering: Models for Industrial Enterprises. Springer, Berlin Heidelberg New York Tokyo.

Schonenberg, H., Weber, B., Dongen, B., & Aalst, W. M. P. van der (2008). Supporting flexible processes through recommendations based on history. In Dumas, M., Reichert, M., & Shan, M. C. (Eds.) *Proceedings of the 6th International Conference on Business Process Management (BPM '08)* (pp. 51-66). Springer Berlin-Heidelberg.

Software Engineering Institute (2008). *Capability maturity model*. Retrieved from www.sei.cmu.edu/cmmi.

Song, I., & Diederich, J. (2010). Intention extraction from text messages. In Wong, K. W., Mendis, B. S. U., & Bouzerdoum, A.(Eds.) *Neural information processing: theory and algorithms* (Vol. 1, pp. 330-337). Springer-Verlag, Berlin, Heidelberg.

Stefansen, C., & Borch, S. E. (2008). Using Soft Constraints to Guide Users in Flexible Business Process Management Systems. *International journal of business process integration and management*, *3*(1), 26-35.

Sun, Y., Huang, J. Z., & Meng, X. (2011). Integrating constraints to support legally flexible business processes. *Information Systems Frontiers* (Vol. 13, pp. 171-189). Retrieved from http://dx.doi.org/10.1007/s10796-009-9190-z

Tiwari, A., Turner, C. J., & Majeed, B. (2008). A review of business process mining: state-of-the-art and future trends. *Business Process Management Journal*, *14*(1), 5-22. Retrieved from http://www.emeraldinsight.com/10.1108/14637150810849373

Vanderfeesten, I., Reijers, H. A., & Aalst, W. M. P. van der (2008). Product Based Workflow Support : A Recommendation Service for Dynamic Workflow Execution. *Design, 5074 {LNCS*, 571-574. BPMcenter.org. Retrieved from http://bpmcenter.org/wp-content/uploads/reports/2008/BPM-08-03.pdf

Van Lamsweerde, A., & Letier, E. (2004). From object orientation to goal orientation: A paradigm shift for requirements engineering. (M. Wirsing, A. Knapp, & S. Balsamo, Eds.)*Radical Innovations of Software and Systems Engineering in the Future*, *2941*(I), 325-340. SPRINGER-VERLAG BERLIN. Retrieved from http://discovery.ucl.ac.uk/97080/

Weber, B., Rinderle, S., & Reichert, M. (2008). Change Patterns and Change Support Features in Process-Aware Information Systems. (J. Krogstie, A. Opdahl, & G. Sindre, Eds.)*Change*, *4495*(May 2008), 574-588. Springer. Retrieved from http://portal.acm.org/citation.cfm?id=1768029.1768084

Weber, B., Sadiq, S., & Reichert, M. (2009). Beyond Rigidity - Dynamic Process Lifecycle Support. *Computer Science - Research and Development*, *23*(2), 47–65. Springer. Retrieved from http://www.springerlink.com/index/H2NT15J657N46413.pdf

Weerd, I. van de, & Brinkkemper, S. (2008). Meta-modeling for situational analysis and design methods. In M. R. Syed, & S. N. Syed (Eds.), *Handbook of Research on Modern Systems Analysis and Design Technologies and Applications* (pp. 38-58). Hershey: Idea Group Publishing.

Wieringa, R. (2009). Design science as nested problem solving. *Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology DESRIST 09* (p. 1). ACM Press. Retrieved from http://portal.acm.org/citation.cfm?doid=1555619.1555630.

Wieringa, R. (2010). Relevance and Problem Choice in Design Science. In R. Winter, J. L. Zhao, & S. Aier (Eds.), *Global Perspectives on Design Science Research* (Vol. 6105, pp. 61-76). Springer. Retrieved from http://www.springerlink.com/index/6M55774MW8880235.pdf.

Yin, R. K. (2009). *Case Study Research: Design and Methods*. In Bickman, L., & Rog, D. J. (Eds.) *Essential guide to qualitative methods in organizational research* (Vol. 5, p. 219). Sage Publications. Retrieved from http://books.google.com/books?id=FzawIAdilHkC&pgis=1.

Yu, E. S. K. (1997). Towards modelling and reasoning support for early-phase requirements engineering. *Proceedings of ISRE 97 3rd IEEE International Symposium on Requirements Engineering*. IEEE Comput. Soc. Press. Retrieved from http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=566873.

Yu, E. (2009). Social Modeling and i. (A. Borgida, V. Chaudhri, P. Giorgini, & E. Yu, Eds.)*Conceptual Modeling Foundations and Applications*, *5600*(c), 99-121. Retrieved from http://www.springerlink.com/content/682l1wx71752170t

Zisman, M. D. (1977). *Representation, Specification and Automation of Office Procedures*. Unpublished doctoral dissertation, Wharton School of Business, University of Pennsylvania, PA, USA.

Zoukar, I, & Salinesi, C. (2004). Using goal/strategy/maps to reduce the language disparity issue in ERP projects. In Grundspenkis, J., & Kirikova, M. (Eds.) *Knowledge and Model Driven Information Systems Engineering for Networked Organisations* (Vol. 2, pp. 325-339). Riga Technical University, Riga, Latvia.

APPENDICES

Appendix A: Case Study Protocol for the Artifacts' Evaluation

The case study protocol follows the template proposed by Yin (2009) and it contains the following sections:

- The introductory section of the case study where its purpose is outlined
- The field procedures regarding the data collection
- The case study questions
- The data analysis procedures.

Introduction to the Case Study and Purpose of the Protocol

The goal of this case study is to evaluate the created artifacts. The evaluation is conducted twofold: the first part focuses on evaluating the functional requirements of the artifacts, the second part focuses on evaluating the non-functional requirements of the artifacts. In this phase of the project the main and the knowledge questions are answered:

The main research questions: • To what extent does a recommendation tool based on intention mining improve the process participant's interaction with flexible process aware information systems?
The artifact-oriented knowledge questions: • Effect question: what are the effects of treating the problematic context with the artifact? • Requirements satisfaction question: do the effects satisfy the functional and non-functional requirements of the artifact? • Trade-off question: what are the effects of treating the same problem context with another artifact?
The problem-oriented knowledge questions: • Contribution question: do the designed artifacts contribute to the satisfaction of goals?

The functional requirements are formulated as design decisions of the intention mining technique and the recommendation tool.
The non-functional requirements are formulated as follows:

- For the intention mining technique:
  - **Correctness** meaning that the artifact discovers the same intentions as those reported by the process participant.
  - **Completeness** meaning that the artifact discovers all the intentions reported by the process participant.
- For the recommendation tool:
  - **Effectiveness** in supporting decision making by supporting **criteria identification** during process enactment.
  - **Effectiveness** in supporting decision making by supporting the **formulation of alternatives** during process enactment.
  - **Effectiveness** in supporting decision making by supporting the **analysis of alternatives** during process enactment.

Data Collection

The data is collected in the settings of an experiment (Annex 2), defined in advance. The experiment aims at testing the intention mining technique during the first exercise and the recommendation tool during the second exercise.

Experiment's Subjects

The subjects of the experiment are process participants interacting with the Childcare application. Previous experience is not required especially because the problems regarding the decision making during enactment occur mainly when there is a lack of experience. However, a tutorial about how to use the application is provided in advance. The first requirement regarding the experiment's participants is to comprehend and to be able to express themselves in English. The second requirement is to have basic computer skills.

Guiding Hypotheses

The hypothesis guiding the evaluation of the intention mining technique is that the mined intentions are correct and complete. In order to collect the necessary data, semi-structured interviews are held with the experiment's participants while they are accomplishing the tasks. During their performance, they are asked about their intentions. Besides identifying the intentions, the participants are asked to notify the interviewer when they start to proceed with a certain intention and when they realized the intention.

The evaluation of the recommendation tool is realized at a conceptual level through a questionnaire. Its aim is to analyze if recommendations given as intentions and confidence factors improve the effectiveness of supporting decision making by supporting the criteria selection, alternatives formulation and alternatives analysis. The questionnaire starts with general questions for understanding the background of the participants regarding the recommendations given by software.

The first hypothesis guiding the evaluation of the recommendation tool is that the recommendations given as intentions improve the support for decision making by supporting the criteria identification. During the enactment, if the process participant asks for recommendations, several next intentions are provided. At this point, the criteria leading the decision making process is in the mind of the process participant and is based on his knowledge and on his perception of the environment. This is a fuzzy situation for evaluation and probably the accuracy of the results would be affected through the subjectivism introduced by agency's characteristic of humans. However, continuing with the flow of events after the intention is chosen, several possibilities regarding what to do next are provided as activities. This is the second decision that the process participant must make, but having as criterion the intention previously chosen. The first part of the questionnaire aims at evaluating if the decision making at the executable level of process (activities) is improved by supporting the criteria identification. In order to do so, a starting scenario is presented and the process participant is asked to identify what he should do. The same scenario is after enriched with a set of recommendation given as activities and the process participant is asked to make a decision, to motivate it and to evaluate the difficulty of the process. Finally, the previous scenario is enriched not only with the set of recommendations but also with the intention behind them. The process participant is asked to answer the same questions as in the previous scenario.

The next hypotheses guiding the evaluation of the recommendation tool is that the recommendations given as intentions improve the support for decision making by supporting the alternatives formulation and by supporting the alternatives analysis. These are validated by comparing the opinions of process participants with respect to decision making in three different scenarios: one without any support, one with support at the executable level (alternatives are given as activities with or without the intention behind them) and one with support at the intentional level (alternatives given as intentions).
Finally, the last hypothesis is that the confidence factors included in the recommendations improve the support for decision making. This is tested by analyzing their influence on the process participants in making decisions and their perceived usefulness.

Data collection procedure

The data is collected only in the presence of the main researcher, which can be on-site or remote using various tools for ensuring this such as Skype or chatting clients. To ensure the rigor of the evaluation process, the research supervisors evaluate the case study protocol for compliance with validity criteria, in advance. Moreover, they participate in a pilot experiment, launched before the actual evaluation.
An experiment lasts around two hours starting with the training of the process participants regarding the application and the first exercise. Then, it continues with the collection of the data for the evaluation of the intention mining technique through semi-structured interviews. During these interviews, clarifications about the tasks are given if needed or required, but without influencing the participant's behavior in realizing the intentions. The experiment ends with the questionnaire regarding the evaluation of the recommendation tool. The collected data is recorded in a case study database consisting of interviews logs and process participants traces (recorded in the SQL Server database used by the Childcare application).

Case Study Questions

The questions regarding the first exercise of the experiment focus on making the process participant to verbalize the intentions such as:

- What you are doing now? Why are you doing this now?
- What do you want to do next? If the process participant verbalizes a composed intention, then the researcher asks: with what do you want to start?

Moreover, the process participants must acknowledge directly or indirectly the time when they start a certain intention and the time when they achieved it or dropped it.
The questions regarding the second exercise of the experiment follow the hypotheses outlined in the previous section. The full questionnaire is presented in Annex C.

Data Analysis

The evaluation of the collected data is conducted by the main researcher under the supervision of the project coordinators. The results are reported in the evaluation chapter of the thesis.

For the evaluation of the intention mining technique, the intentions are discovered from the process participants' traces recorded in the Childcare system's database. The results are then compared with the verbalized intentions of the process participants gathered during the interviews using the Confusion matrix (Kohavi and Provost, 1998). In the evaluation, two measures, calculated based on the confusion matrix, are used: accuracy and precision.

The evaluation of the recommendation tool is driven by the formulated hypotheses. Once the data is collected, groups with relevant data for each hypothesis are formed. The next step is to visualize the data per each group in a proper format such as tables, charts or diagrams in order to have good support for data analysis. Finally, the analysis of data is conducted in correspondence with the guiding hypotheses and conclusions are drawn.

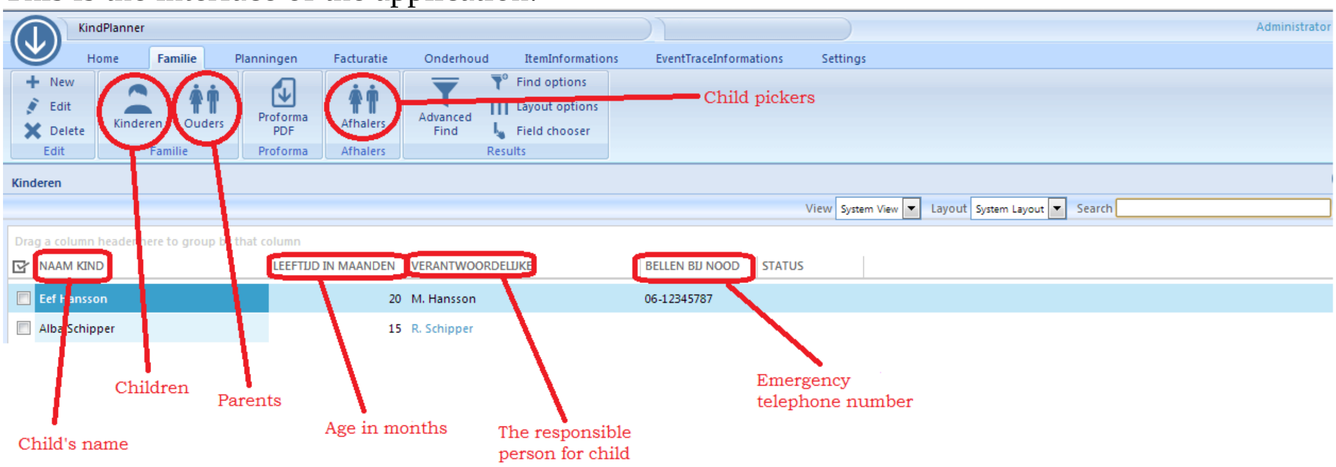
Appendix B: Experiment's Description

You just started your new job at **Balloons and Butterflies**, a Dutch child care organization. Here, you use a back-office application to perform the daily work activities. The activities are grouped in the following categories: "*Registration*", "*Child care's planning*", "*Contract handling*" and "*Invoicing*".

For starters, you are responsible only for "*Registration*".

In the first day of work, your manager provided you a tutorial about how to use the Childcare back-office application which is presented further.

This is the interface of the application:



- The application’s interface is partly in English, partly in Dutch. The English translation is provided for the words in Dutch (in red font).

- All the tasks regarding the registration for child care are grouped under the tab **Familie**. Consequently, your activities are carried out only in this section of the application.

- When you click on the tab **Familie** you can see a ribbon with several buttons:
  - **Kinderen** (Children) used to navigate to the existing list of registered children
  - **Ouders** (Parents) used to navigate to the existing list of parents
  - **Afhalers** (Child pickers) used to navigate to the existing list of child pickers. A child picker is the person responsible for picking up the child from the childcare organization and bringing him/her home.

- Besides the ribbon, the existing list of children is printed on the screen, by default. For each child, several fields are showed: his/her name, his/her age in months, the person responsible for the child (usually one of the parents) and the emergency telephone number.

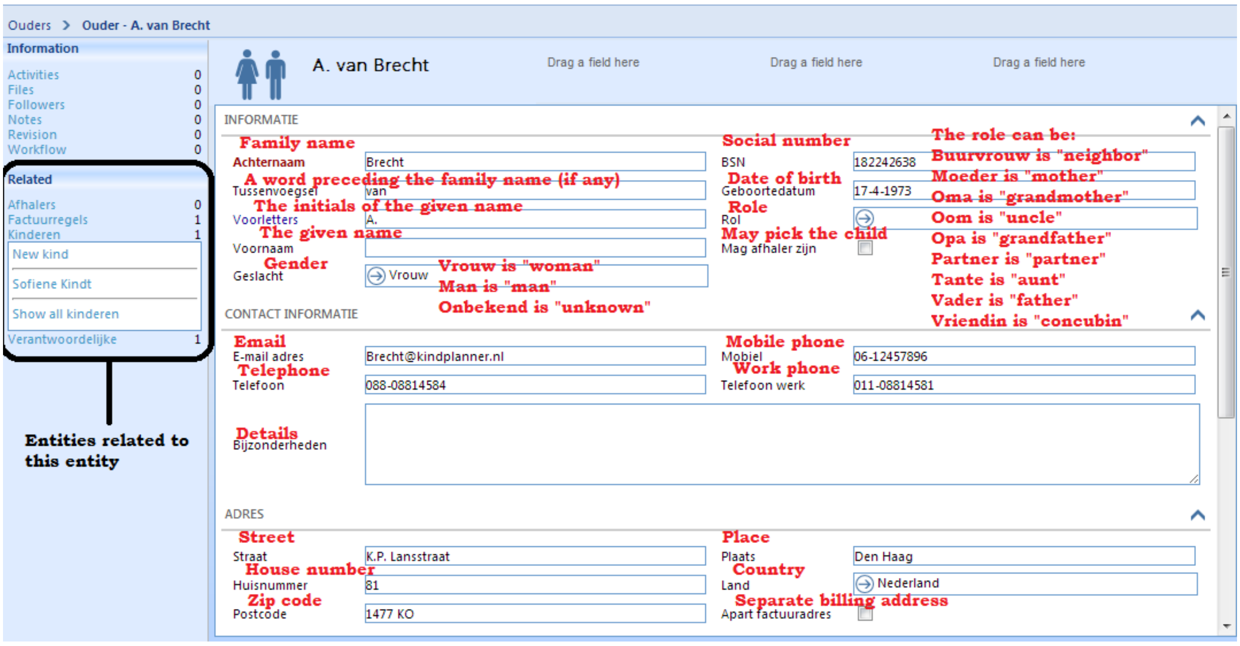
You manipulate three types of entities: **Kind** (Child), **Ouder** (Parent) and **Afhaler** (Child picker). For each of these entities you can:

- **Read** the entities registered in the system:
  - By clicking on the corresponding button in the ribbon (**Kinderen**, **Ouders**, **Afhalers**);
  - By clicking on a specific entity.
- **Create** a new entity:
  - By clicking on the **New** button (up-left corner) after you are in the right window. For example, to create a new parent you first navigate to the window with the list of parents and after you click on the **New** button.
- **Update** an existing entity :
  - By selecting an entity and after clicking on the **Edit** button (up-left corner).
  - By opening directly the entity’s form (click on the entity)
- **Delete** an existing entity:
  - By selecting an entity and after clicking on the **Delete** button (up-left corner).
- **Search** in the list of entity using a keyword. The keyword can be written above the list of entities, in the right.

The creation and update of an entity is ended after the entity is saved (the **Save** button in the up-left corner of each entity’s form).

Ouder Form

The translated field for an entity of type **Ouder** (Parent) is presented below:



The form contains also other fields than those showed in this picture, but they are ignored for now.

In the left side of the form there is an area named **Related** which shows all the entities related to the entity that is read. By clicking on one of those links (which are entity types) several options appear. The case of clicking on **Kinderen** is exposed in the picture but it is similar for all the others:

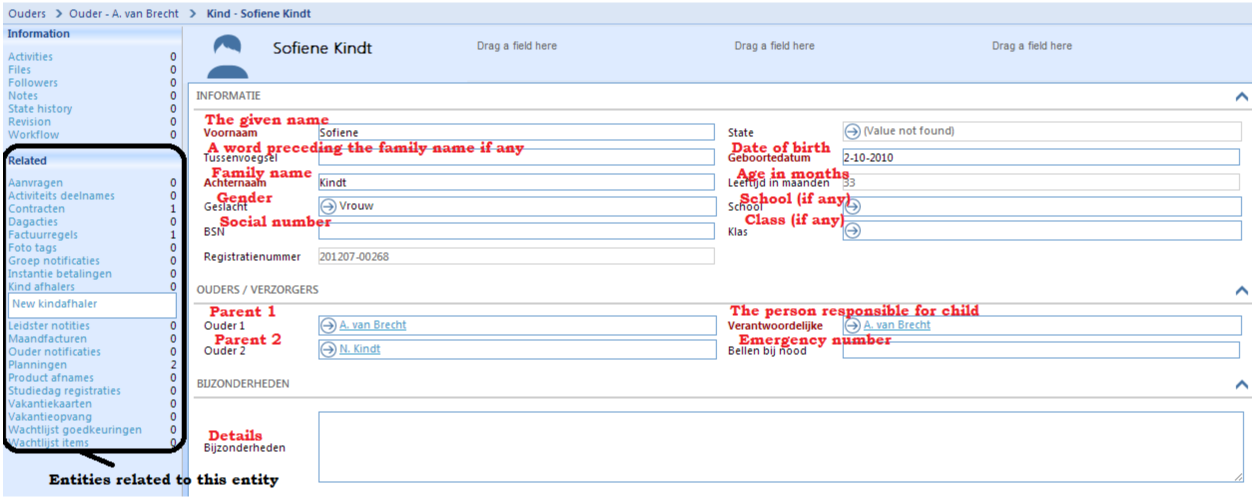
- "*New kind*": create a new child related to this parent
- "*Sofiene Kindt*": is the name of a specific child that is related to this parent. By clicking on this name you can navigate to its form
- "*Show all kinderen*": read the list of children related to this parent

Child Form

The translated form for an entity of type **Kind** (Child) is presented below.

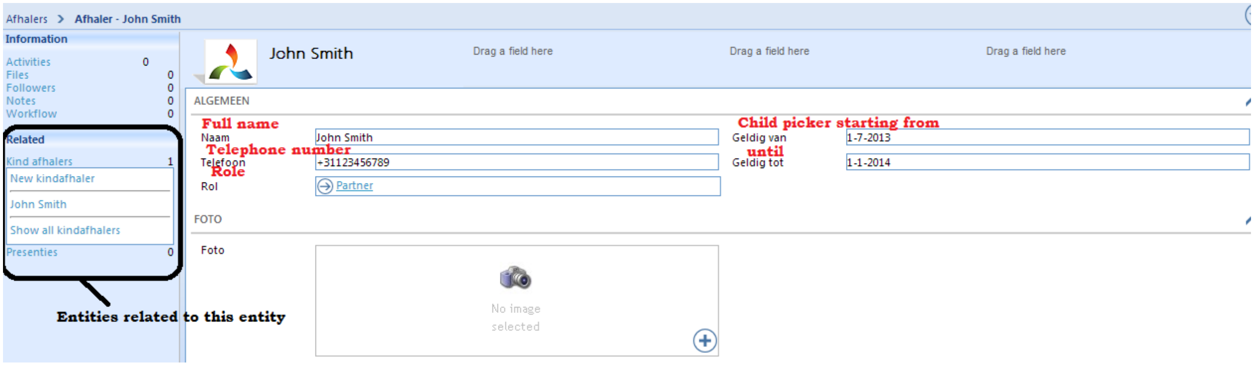
The form contains also other fields than those showed in this picture but they are ignored for now.

In the left side of the form there is an area named **Related** which shows all the entities related to the entity that is read.



Afhaler Form

The translated form for an entity of type **Afhaler** (Child picker) is presented below.
In the left side of the form there is an area named **Related** which shows all the entities related to the entity that is read.



Kindafhaler Form

In order to define a child picker, the creation of an entity of type **Afhaler** is not enough.
It is also necessary to create an entity of type **Kindafhaler** which represents the relation between the child entity and the child picker entity. There are two possibilities to do so:

- From the related entities of the child (see the section **Related** of the **Child** form above), click on "*New kindafhaler*".
- From the related entities of the child picker (see the section **Related** of the **Afhaler** form above), click on "*New kindafhaler*".

A form will appear with two fields: **Afhaler** (Child picker) and **Kind** (Child).

Final Remark: There is no button to go back or to cancel something. In order to do so, you click on **Home** button in the ribbon.

Exercise no. 1

The training period ended and you are able now to start the work.

REQUIREMENTS:

The manager wants to check if your training was successful. Therefore, he asks you to perform several tasks, planned for today. In parallel with this, he asks you to declare all the time:

- What is your intention; the list of possible intentions is
  - **Read a list of entities** (such as read the list of parents, read the list of children etc.)
  - **Read a specific entity** (such as read the child with the name "*X*" etc.)
  - **Create a new entity** (such as create a new child, create a new parent etc.)
  - **Update an existing entity** (such as update the child with the name "*X*" etc.)
  - **Delete an existing entity** (such as delete the child with the name "*X*" etc.)
- When your intention starts
- When your intention ends

In conclusion, during the interaction with the application, you will complete a separate form where you mark: your intention, the start time of the intention, and the end time of the intention.

The following tasks were planned:

- A new registration for child care. A parent called and he filled out the following form:

Child	Family name	Johan
	Given name	Sebastian
	Gender	Man
	Social number	3443567678
	Date of birth	12/11/2010
	The person responsible for child	the registered parent
Parent	Family name	Johan
	Given name(s)	Marcela
	Genre	Woman
	Social number	2345134534
	Date of birth	25/10/1970
	Role	Mother
	May pick the child	Yes
	Email	Spruit.marcela@mail.nl
	Telephone	+31 668 83 35 24
	Street	Kriekenpitplein

	House number	123
	Zip code	2354 EH
	Place	Utrecht
	Country	Nederland
Child picker	Name	Tolbiac Cedric
	Telephone	+31 123 37 45 84
	Role	Grandfather
	Starting from	01/08/2013
	Until	01/01/2014

- The child "*Kim Hermans*" will have a child picker starting from August. The parent declared the details of the new child picker:

Child picker	Name	Turner Maria
	Telephone	+31 321 73 45 67
	Role	Aunt
	Starting from	01/08/2013
	Until	01/12/2013

- The parent "*J. Kalma*" called:
  - For updating the details regarding his child who started to go to school. The school is "*John F. Kennedyschool*"
  - For requesting child care for her second child who has the following details:

Child	Family name	Kalma
	Given name	Marco
	Gender	Man
	Date of birth	06/05/2011
	The person responsible for child	J. Kalma

Remark 1: the tasks could be accomplished in any order.
Remark 2: do not forget to write down the intentions while you are performing the tasks.
The back office is a web-based application so you can access it by following this link: http://deployments.test1.42windmills.com/ChildCare/ChildCare/testElena/Default.aspx
The credentials are:

- Username: **testuser1**
- Password: **password**

Exercise no. 2

Now, that you gained experience in working with the Childcare application, your activities expended to the "*Child's care planning*" too.

The care planning is triggered by the registration process. The standard planning depends on the age of the child, the locations and the preference of the parents for having their child in a horizontal group targeting the same age, a vertical group consisting of children of different ages but in the same proportion or a variable group. The process of planning includes also the management of a waiting list and the management of the group occupation.

Being over-cautious, your manager wants to evaluate your understanding and preparation in advance. The evaluation is in the form of a questionnaire which can be accessed online via this link:

https://docs.google.com/forms/d/1fZ53ZewnnGS6usKTZqalJHjFQtCzwJynmudASoUay_k/viewform

Remark: this exercise is offline. You **do not** need to use the back office application for this part.

Appendix C: Experiment's Questionnaire

1. In the context of your previous jobs, what kind of software did you use?
2. Did any of these tools you used support your work activities by providing suggestions / recommendations while you were interacting with them?
3. I usually considered the recommendations provided by the tool(s) in performing my work activities. (If applicable)
    a. Strongly disagree
    b. Disagree
    c. Agree
    d. Strongly agree

Scenario no. 1

You have the following situation: A child **ChildX** was registered and started to benefit of childcare when he was three years old. This month, the responsible parent, **ParentX**, called announcing that **ChildX** turned four years old and he started school. His care planning must change because he needs a different type of care: from KDV – "*little children care*" to BSO – "*after school care*".

4. Having this information available, could you describe what you believe you should do?
5. I find easy to identify what I should do.
    a. Strongly disagree
    b. Disagree

    c. Agree
    d. Strongly agree
6. Motivate your previous answer.

Scenario no. 2

You have the following situation as before: A child **ChildX** was registered and started to benefit of childcare when he was three years old. This month, the responsible parent, **ParentX**, called announcing that **ChildX** turned four years old and he started school. His care planning must change because he needs a different type of care: from KDV – "*little children care*" to BSO – "*after school care*".

You just logged in the back office application and you navigated in the tab **Familie**. The following recommendations about what to do next are given:

R1: Navigate to the list of parents, 0.3
R2: Navigate to the list of children, 0.2
R3: Search the child ChildX, 0.6
R4: Click on the child ChildX, 0.6
R5: Navigate to the tab Onderhoud (Administration), 0.6

The numbers associated to recommendations aggregate knowledge about what other process participants chose, being in the same or an approximate scenario as you.

7. Which recommendation would you choose to follow?
    a. R1: Navigate to the list of parents
    b. R2: Navigate to the list of children
    c. R3: Search the child ChildX
    d. R4: Click on the child ChildX
    e. R5: Navigate to the tab Onderhoud (Administration)
    f. None
8. Motivate your decision.
9. I find these recommendations useful to decide what I want to do next.
    a. Strongly disagree
    b. Disagree
    c. Agree
    d. Strongly agree
10. It would help me to know the intention behind these recommendations before making the decision.
    a. Strongly disagree
    b. Disagree
    c. Agree
    d. Strongly agree
11. Motivate your previous answer.
12. For this set of recommendations, which intention do you believe is behind them?

Scenario no. 3

You have the following situation as before: A child **ChildX** was registered and started to benefit of childcare when he was three years old. This month, the responsible parent, **ParentX**, called announcing that **ChildX** turned four years old and he started school. His care planning must change because he needs a different type of care: from KDV – "*little children care*" to BSO – "*after school care*".

You just logged in the back office application and you navigated in the tab **Familie**. The following recommendations about what to do next are given:

R1: Navigate to the list of parents, 0.3
R2: Navigate to the list of children, 0.2
R3: Search the child ChildX, 0.4
R4: Click on the child ChildX, 0.4
R5: Navigate to the tab Onderhoud (Administration), 0.4

However, this time you know the intention behind the recommendations: **Update the school of ChildX**.

13. Which recommendation would you choose to follow?
    a. R1: Navigate to the list of parents
    b. R2: Navigate to the list of children
    c. R3: Search the child ChildX
    d. R4: Click on the child ChildX
    e. R5: Navigate to the tab Onderhoud (Administration)
    f. None
14. Motivate your decision.
15. Knowing the intention, I find it easier to choose a specific recommendation.
    a. Strongly disagree
    b. Disagree
    c. Agree
    d. Strongly agree
16. Motivate your previous answer.

Scenario no. 4

You have the following situation as before: A child **ChildX** was registered and started to benefit of childcare when he was three years old. This month, the responsible parent, **ParentX**, called announcing that **ChildX** turned four years old and he started school. His care planning must change because he needs a different type of care: from KDV – "*little children care*" to BSO – "*after school care*".

You just logged in the back office application. Two sets of recommendations are given.

SetA	SetA-R1: Update the planning of ChildX, 0.3
	SetA-R2: Create a new planning for ChildX, 0.4
	SetA-R3: Register the school of ChildX if it does not exist, 0.2
	SetA-R4: Update the school of ChildX, 0.3

	SetA-R5: Update the contract of ChildX, 0.1
	SetA-R6: Check if the new planning is possible, 0.4

SetB	SetB-R1: Click on the tab Planningen (Planning), 0.4
	SetB-R2: Click on the tab Familie (Family), 0.4
	SetB-R3: Click on the tab Onderhoud (Administration), 0.2

Which recommendation would you choose to follow from SetA?

a. SetA-R1: Update the planning of ChildX
b. SetA-R2: Create a new planning for ChildX
c. SetA-R3: Register the school of ChildX if it does not exist
d. SetA-R4: Update the school of ChildX
e. SetA-R5: Update the contract of ChildX
f. SetA-R6: Check if the new planning is possible
g. None

17. Motivate your decision.
18. I find the first set of recommendations (**SetA**) useful to decide what I want to do next.
    a. Strongly disagree
    b. Disagree
    c. Agree
    d. Strongly agree
19. Motivate your previous answer.
20. Which recommendation would you choose to follow from **SetB**?
    a. SetB-R1: Click on the tab Planningen (Planning)
    b. SetB-R2: Click on the tab Familie (Family)
    c. SetB-R3: Click on the tab Onderhoud (Administration)
    d. None
21. Motivate your decision.
22. I find the second set of recommendations (**SetB**) useful to decide what I want to do next.
    a. Strongly disagree
    b. Disagree
    c. Agree
    d. Strongly agree
23. Motivate your previous answer.
24. Which set of recommendations do you prefer?
    a. SetA
    b. SetB
25. Motivate your previous answer.

Confidence factors in recommendations

26. The numerical values attached to the each recommendation influenced my decision.
    a. Strongly disagree
    b. Disagree

c. Agree
d. Strongly agree

27. Motivate your answer.

About you

28. What is your age?
29. What is your gender?

Appendix D: Data for the Intention Mining Technique's Evaluation

The intentions discovered by the intention mining technique and those verbalized directly or indirectly by the participants are presented further. The intentions are reported in the order they were realized.
Besides this, the cardinal of the Confusion matrix classes is calculated for each trace as follows:
True positive predictions: the discovered intention using the intention mining technique is verbalized directly or indirectly by the process participant. The matching table's entries are counted.
False positive predictions: the discovered intention using the intention mining technique is not verbalized by the process participant. The table's entries that are found in the left column (intentions discovered by the intention mining technique) but not found in the right column (intentions verbalized by the participant) are counted.
False negative predictions: an intention is not discovered using the intention mining technique even if it is verbalized directly or indirectly by the process participant. The table's entries that are found in the right column (intentions verbalized by the participant) but not found in the left column (intentions discovered by the intention mining technique) are counted.
True negative predictions: an intention is not discovered using the intention mining technique and it is not verbalized directly or indirectly by the process participant. The number of true negative predictions is considered always 0 as the existence/discovery of the intention is always false. An intention that does not exist in the mind of the process participant cannot be discovered.
During the experiment, 2 participants (Test User7 and Test User10) encountered the same bug in the application while creating a new child and the intention was abandoned in the end. That part of trace was excluded from classification.

Trace 1:

Trace 1	Intentions discovered using the intention mining technique	Intentions verbalized directly or indirectly by the process participant
	Read children	
	Create parent	Create parent

Task 1	Create child	Create child
	Create child picker	
	Create child – child picker link	
	Read child	
	Create child picker	Create child picker
Task 2	Read child picker	Search child picker
	Read child	Search child to create the link to child picker
	Create child – child picker link	Try to create child –child picker link (it fails because the child picker does not exist)
	Create child picker	Create child picker
	Create child – child picker link	Create child – child picker link
Task 3	Update child	Update child
	Create child	Create a new child
	Read children	

True positive predictions	**False positive predictions**	**True negative predictions**	**False negative predictions**
10	5	0	0

Trace 2:

Trace 2	Intentions discovered using the intention mining technique	Intentions verbalized directly or indirectly by the process participant
Task 1	Read children	
	Create parent	Create parent
	Create child	Create child
	Create child picker	
	Create child – child picker link	
	Update child	
	Create child picker	Create child picker
Task 2	Create child picker	Create child picker
	Read child	Search child to create the link to child picker
	Read child - child picker links	Read for deleting the existing child picker

	Read child	Read child to add the new child picker
	Create child – child picker link	Create child – child picker link
Task 3	Read parent	Search the parent
	Update child	Update child
	Create child	Create a new child

True positive predictions	False positive predictions	True negative predictions	False negative predictions
11	4	0	0

Trace 3

Trace 3	Intentions discovered using the intention mining technique	Intentions verbalized directly or indirectly by the process participant
Task 1	Read children	
	Create parent	Create parent
	Create child	Create child
	Create child picker	
	Create child – child picker link	
	Read child	
	Create child picker	Create child picker
Task 2	Read child	
	Create child picker	Create child picker
	Read child	Search child to create the link to child picker
	Create child – child picker link	Create child – child picker link
Task 3	Read children	
	Read parent	Search parent
	Update child	Update child
	Read parent	Search parent
	Create child	Create child

True positive predictions	False positive predictions	True negative predictions	False negative predictions
10	6	0	0

Trace 4

Trace 4	Intentions discovered using the intention mining technique	Intentions verbalized directly or indirectly by the process participant

Task 1	Read children	
	Create parent	Create parent
	Create child	Create child
	Create child picker	
	Create child – child picker link	
	Read child	
	Create child picker	Create child picker
	Read child	Search child to create the link to child picker
	Create child – child picker link	Create child – child picker link
Task 2	Create child picker	Create child picker
	Read child	Search child to create the link to child picker
	Create child - child picker link	Create child – child picker link
Task 3	Read parent	Search parent
	Update child	Update child
	Create child	Create child

True positive predictions	**False positive predictions**	**True negative predictions**	**False negative predictions**
11	4	0	0

Trace 5

Trace 5	Intentions discovered using the intention mining technique	Intentions verbalized directly or indirectly by the process participant
Task 1	Create child	Create child (without success, the responsible parent was missing)
	Create parent	Create parent
	Create child	Create child
	Create child picker	
	Create child – child picker link	
	Read child	
	Create child picker	Create child picker
	Read child	Search child

Task 2	Create child - child picker link	Create child - child picker link (without success because the child picker was missing)
	Create child picker	Create child picker
	Read child	Search child to create the link to child picker
	Create child - child picker link	Create child - child picker link
Task 3	Read parent	Search the parent
	Update child	Update child
	Create child	Create child

True positive predictions	**False positive predictions**	**True negative predictions**	**False negative predictions**
12	3	0	0

Trace 6

Trace 6	Intentions discovered using the intention mining technique	Intentions verbalized directly or indirectly by the process participant
Task 1	Create child	Create child (the child is saved with another parent)
	Create child picker	
	Create child – child picker link	
	Read child	
	Create parent	Create parent
	Update child	Update child
	Create child picker	
	Create child – child picker link	
	Update child	
	Create child picker	Create child picker
	Create child – child picker link	Create child – child picker link
Task 2	Create child picker	Create child picker
	Read child	Search child
	Create child - child picker link	Create child - child picker link
	Read children	

Task 3	Read parent	Search parent
	Update child	Update child
	Read parent	Search parent again
	Create child	Create child
		Read children

True positive predictions	**False positive predictions**	**True negative predictions**	**False negative predictions**
12	7	0	1

Trace 7

Trace 7	Intentions discovered using the intention mining technique	Intentions verbalized directly or indirectly by the process participant
Task 1	Read children	
	Create parent	Create parent
	Create child**	Try to create a child several times (it was an error in the application). It is excluded from counting the predictions.
Task 2	Read child	Search child
	Create child - child picker link	
	Create child picker	Create child picker
	Read child	Read child
	Update child - child picker link	Update child - child picker link
Task 3	Read parent	Search parent
	Update child	Update child
	Read parent	
	Read child	
	Read parent	
		Create child

True positive predictions	**False positive predictions**	**True negative predictions**	**False negative predictions**
7	5	0	1

Trace 8

Trace 8	Intentions discovered using the intention mining technique	Intentions verbalized directly or indirectly by the process participant
Task 1	Read children	
	Create parent	Create parent
	Create child	Create child
	Create child picker	
	Create child – child picker link	
	Read child	
	Create child picker	Create child picker
	Create child – child picker link	Create child – child picker link
Task 2	Read child	Read child
	Create child - child picker link*	Create child - child picker link without success as the child picker does not exist
	Create child picker	Create child picker
	Create child - child picker link	Create child – child picker link
Task 3	Update child	Update child
	Create child	Create child

True positive predictions	**False positive predictions**	**True negative predictions**	**False negative predictions**
10	4	0	0

Trace 9

Trace 9	Intentions discovered using the intention mining technique	Intentions verbalized directly or indirectly by the process participant
Task 1	Read children	
	Create parent	Create parent
	Create child	Create child without success because the parent was not saved
	Create parent	Create parent
	Create child	Create child
	Create child picker	

	Create child – child picker link	
	Read child	
	Create child picker	Create child picker
	Read child	Read child
	Create child – child picker link	Create child – child picker link
Task 2	Read child	Search child
	Create child picker	Create child picker
	Read child	
	Create child - child picker link	Create child - child picker link
Task 3		Search child
	Read parent	Search parent
	Update child	Update child
	Create child	Create child

True positive predictions	False positive predictions	True negative predictions	False negative predictions
13	4	0	1

Trace 10

Trace 10	Intentions discovered using the intention mining technique	Intentions verbalized directly or indirectly by the process participant
Task 1	Read children	
	Create parent	Create parent
	Create child**	Try to create a child several times (it was an error in the application)
	Create child picker	Create child picker
	Create child - child picker link	Create child - child picker link
Task 2	Read child	
	Read parent	Search parent
	Read child	
	Create child picker	Create child picker
	Create child - child picker link	Create child - child picker link

	Read parent	Search parent
	Read child	
	Read parent	
	Update child	Update child
Task 3	Create child	Create child

True positive predictions	**False positive predictions**	**True negative predictions**	**False negative predictions**
9	5	0	0