Towards Process-Oriented Recommender Capabilities in Flexible Process Environments - State of the Art

Thomas Burkhart
German Research Center for Artificial Intelligence (DFKI)
Thomas.Burkhart@dfki.de

Benjamin Weis
German Research Center for Artificial Intelligence (DFKI)
Benjamin.Weis@dfki.de

Dirk Werth
German Research Center for Artificial Intelligence (DFKI)
Dirk.Werth@dfki.de

Peter Loos
German Research Center for Artificial Intelligence (DFKI)
Peter.Loos@dfki.de

Abstract
The lack of flexibility during process execution has been commonly identified as a major drawback of classical process-aware information systems. Hence, in the last decade, extensive research has been done and multiple approaches concerning process flexibility have emerged. Since an increasing degree of freedom implies a decreasing inbuilt guidance regarding the way business processes are handled, the decision making about their execution is delegated to the personnel at runtime. Obviously, there is an imperative necessity for functionalities that support users in finding the right decisions during flexible runtime procedures. Concerning this matter, the following paper investigates the application of recommender systems as they are widely known from the e-commerce domain. Its main contribution is a profound state of the art analysis and an identification of current research gaps.

1. Introduction
In the globalized world with its ever-changing economic conditions [1], today’s enterprises are compelled to organize their business processes in a highly flexible manner [2]. As widely recognized by extensive research activities in the last decade, the solely separation of functional and procedural logic serves only insufficiently to meet these requirements [3]. Thus, classical process-aware information systems (PAIS) are considered to be far too inflexible [4-6], which has motivated the development of several paradigms concerning the flexibilization of process execution or the adaption of running business processes [2;3].

However, none of these approaches has prevailed so far, which may be affiliated to a major drawback of process flexibility concerning the sacrifice of inbuilt process guidance [2-4;6]. As a matter of fact, there is a trade-off relationship between flexibility and the offered support regarding the decision about the process execution [4;7-9]. Whereas classical solutions rigidly prescribe the way business processes are handled, flexible systems feature a higher degree of freedom, but confer the decision about the exact process composition upon the executing personnel [4;5;7-9]. With increasing runtime flexibility, employees are confronted with an expanding decision space and they are needed to possess more expertise in dealing with the processes they are involved in. Therefore, future solutions have to provide flexibility and guidance at the same time [6]. In doing so, finding the right decisions should be assisted instead of relieving responsibilities and restricting the process execution [4;9]. Hence, an intelligent user assistance is required that offers process-oriented support during flexible runtime procedures, and thereby compensates the stated drawback of process flexibility [5;7;8;10;11].

The information overload of the internet and the associated vast space of options have motivated the development of recommender systems (RS) [12-14]. Through making recommendations for possibly interesting objects, RS provide a personalized decision support. Due to obvious analogies between the purpose of these systems and the described requirements, a directed technology transfer seems to be a possible problem-solving approach. As already recognized by some research publications [5;7;8], the application of recommender functionalities could reduce the decision room in flexible runtime procedures.

This paper identifies the particular potentials of a technological adoption and possible design options. On that basis, a profound survey of the state of the art (SOTA) is given and current research gaps are highlighted. Thus, the remainder is structured as follows: section two outlines the main functionality of RS, which provides the basis for some fundamental conceptual considerations in section three. In section four an evaluation framework for a subsequent analysis of flexible PAIS is derived, whose findings are stated in section five. Section six concludes with a short summary and depicts the need of further research.
2. The Recommender System Technology

In the World Wide Web's myriad universe of alternatives, which could impossibly be evaluated by a single person, RS serve as personalized information agents that recommend objects based on a user's needs, preferences and interests [12]. Especially the successful application in e-commerce platforms made these systems very popular, where they suggest potentially interesting products to customers [15]. As an example, the recommendation service of Amazon recommends (among others) similar products to already bought ones [16]. But besides that, there are countless other application areas in which a decision support in terms of a recommending guidance is needed [14;17]. As a result, RS constitute an own comprehensive field of research [14;18]. Accordingly, the originated solutions feature a variety of functionalities [18], whereas the following provides a short overview on essential characteristics and typical recommendation methods. Of particular importance, it is to remark initially that RS generally offer personalized suggestions in an active manner [19;20]. Hence, these systems have to be differentiated from regular search engines, which require the user to express a query the inferred results depend on.

2.1. Typical Functionality

Basically, RS succeed to the task of evaluating items, which are unknown to the user so far, and recommending objects with a maximizing utility [14;18]. In doing so, the item-specific usefulness is typically expressed through numerical ratings on some metric scale. In principle, the major challenge is to extrapolate the recommendation space (users x objects) and to reason unknown ratings from existing ones. The latter may be given explicitly by the user or inferred from his or her system usage in an implicit way [18;21].

As mentioned before, there are various methods to implement such recommender mechanisms [18]. Although this overview focuses on typical concepts, there is an essential extension to the traditional functionality. Since it seems quite obvious that the utility of an object depends on the current situation of a user, the context sensitivity of a respective solution appears indispensable [14]. Consequently, the system's recommendation space would be multidimensional.

2.2. System Classification

A widely used differentiation of traditional approaches is generally based on the procedure of deriving recommendations [13;14;22]. Thereby, the system types collaborative filtering (CF), content-based filtering (CB) and hybrid filtering (HF) are typically distinguished. In CF the whole recommendation space is used to determine unknown utilities [18]. Based on known ratings, either the similarities between the examined user and others are measured to weight their ratings or the similarities between the focused object and others are determined to do the same [22]. CB approaches derive user profiles from known utilities and the properties of the corresponding objects [14;19;22]. When estimating a rating for a certain object, the gathered knowledge about the user's attribute specific preferences is applied to the object's characteristics. Hybrid methods combine CF and CB to deal with certain problems of their single usage. In case of non-traditional RS even the combination of multiple approaches may be considered as well [19;20].

Even though knowledge-based (KB) approaches differ from classical methods [18], this paper furthermore distinguishes them as they are fundamental for the underlying topic. In contrast to the above described system types, these RS make use of functional knowledge about the user's needs and their possible satisfaction [19;20]. As a matter of fact, such approaches require a respective knowledge basis, which may be available and used in different ways [14;18;19]. One possible way to do so is the application of case-based reasoning (CBR) techniques [14].

3. Technological Adoption

Based on the motivation of this paper and the described functionalities of original recommender solutions, this section identifies possible applications and design options for respective mechanisms in flexible process environments. Note that these conceptual considerations focus on the intended usage of the technology to improve the process-oriented guidance.

Whereas advanced flexibility increasingly motivates the application of compensating recommender functionalities, the resulting potentials and the possible field of application also depend on the given degree of freedom. For instance, a flexible PAIS that supports the structural adaption of running business processes in exceptional situations (adaptive Workflow Management [3]) would not require a recommender mechanism for the regular process execution. Therefore, it is necessary to determine different kinds of flexible runtime procedures in order to identify concrete application possibilities. The taxonomy of [2] basically differs three types of process flexibility during runtime:

- Flexibility by deviation: The execution sequence of tasks in a process may deviate from a predefined procedure (use of change primitives).
Flexibility by underspecification: The process model may hold unspecified parts, which have to be defined individually during process runtime (late binding or late modeling).

Flexibility by change: The PAIS allows for instance-specific changes on the process model (ad-hoc change) or enables modifications on type level, which affect future cases or even corresponding running instances (evolutionary change).

Even though the authors in [9] associate declarative approaches for process modeling with "flexibility by underspecification", it seems obvious that the selection of single activities differs from the definition of an underspecified process structure. The former actually appears quite similar to the runtime specification in ad-hoc workflow management systems (WFMS) [23], which this paper considers as highly flexible concepts with an inherent need for more process-oriented guidance. Since the application of simple change primitives may not require a recommender component, there are three basic flexible runtime procedures that could be supported through a respective solution:

A recommender mechanism for the purpose of process navigation could suggest tasks in order to support the selection of activities.

Through the utilization as specification support, a recommender functionality may serve as a process-oriented guidance during the runtime specification.

With the purpose of modification support, a RS could offer guidance during instance-specific modification procedures.

Based on these potential applications, there are various design options for the implementation of a respective RS. In general, there is no limit to the technological adoption in terms of an adapted application of original functionalities. However, as RS typically do not suggest objects, which have already been consumed by a user [14], and a certain kind of a business process may appear multiple times, there is a major abstract that has to be made. In other words, instead of suggesting unknown objects the system should recommend the best way of solving a business case without limiting this procedure to unknown practices.

Besides offering guidance, a respective recommender mechanism may additionally support the personal working habits of its users [5]. This could be achieved by considering individual preferences on the process execution and personalizing the suggestions provided by the system. As a matter of fact, a well-directed application of that kind could be a major organizational advantage, especially in highly flexible and knowledge-intense process environments.

However, it is also imaginable that basic principles of the respective PAIS imply some restrictions concerning such dynamics in the recommendation behavior. In particular, the application in less flexible solutions may not involve the explicit support of personal working habits for the purpose of standardizing the process structures as far as possible. Consequently, it is questionable in which case and to which extent personalization should be applied at all. Nevertheless, personalization is a fundamental characteristic of RS as it is their spirit and purpose [18-20].

When allowing such far reaching abstracts of the original technology, a dissociation of non-recommender related functionalities has to be defined. In other circumstances, it would be possible to argue that regular PAIS already feature rigid recommender capabilities. Therefore, the SOTA analysis of this paper demands the following requirements regardless of further implementation issues:

The derivation of suggestions should at least be based on a 2-dimensional recommendation space.

Recommendations should be shown proactively as an assortment of options or the user should at least have the opportunity to decide on accepting a prompting best choice (implicit suggestion).

During suggestion development, the user's individual preferences or at least situation-related contextual information should be taken into account.

As it fundamental for a process-oriented recommender service, the recommendation basis has to contain respective process knowledge. The latter may be specified explicitly or, in the best case, implied from the system usage [9;10]. Because like in classical PAIS it is not always possible to anticipate all occurring circumstances at some configuration time [5;24]. Moreover, a non-dynamic acquisition or administrative definition of knowledge lacks in actuality, which causes a reduction of the systems effectiveness in terms of the recommendation quality. Therefore, the systems suggestion behavior has to be continuously enhanced with the business processes of an organization [5] to circumvent the contrasts between reality and a centralized capturing of process knowledge [25;26].

This requirement has already been recognized in the field of modification support, in which a user-driven acquisition offers the possibility of reusing existing knowledge for solving exceptional situations [11;24;27]. Thus, former successful decisions can support future procedures and valuable experiences persist in the respective organization. The outcome of a dy-

1 Note that evolutionary changes are typically executed by administrators, which is why these procedures are no longer considered.
dynamic knowledge acquisition and usage constitutes a crucial advantage: the inbuilt continuous development and utilization of best-practices.

4. Survey of Current Approaches

In the following, the discussed conceptual principles are concretized through the definition of specific evaluation criteria and corresponding feature characteristics. The resulting evaluation framework serves as an objective basis for the subsequent analysis of important research prototypes and some commercial solutions to process flexibility. However, the following subsection initially outlines the general methodology of both steps in the presented survey.

4.1. Methodology

The selection of appropriate evaluation criteria is primarily based on an extensive literature analysis in the underlying subject area and the field of research concerning recommender technology. Since an adoption has already been proposed, but not discussed in detail, this paper is a first approach to deal with the topic in a general manner (as it is to the knowledge of the authors). Hence, the obtained insights have been enhanced by profound factually logical considerations. Furthermore, a preceding analysis of existing concepts had to be taken into account in order to cover all implemented functionalities. As a basic principle, the selection of relevant approaches regarding process flexibility is focused on concepts that already have been realized, and which are thus indeed available. For the purpose of offering a comprehensive survey on the current SOTA, the respective systems have been chosen in consideration of actuality, importance (by means of citation frequencies) and differentiation. Nevertheless, some solutions—especially commercial products—had to be omitted from the analysis due to the fact that the respective documentation lacks in important information. The measurement of parameter values in regard to the appliance of the evaluation framework is generally based on subjective considerations.

4.2. Evaluation Framework

As is apparent from Table 1, the specified criteria have been classified to content-related categories. Thereby, the application context describes some essential characteristics of the PAIS, in which the recommender mechanism is integrated. The main functionality of the latter is coined by its user interaction as well as the actual derivation of suggestions on a certain recommendation basis. As argued before, the adequate usage of the recommender technology allows the dynamic development and utilization of best-practices, which motivates a respective investigation of relevant features. Despite of the intended liberalization of the process execution, a successful implementation demands certain mechanisms for an administrative exertion of control and influence. Similarly the end-users of the system should be able to influence its recommendation behavior especially in the case of dynamic knowledge acquisition.

| Table 1. The evaluation framework of the SOTA analysis |
| System Type | This criterion basically differentiates between research prototypes and commercial solutions. |
| Flexible Runtime Procedure | The second criterion determines which of the above identified flexible runtime procedures are supported by the system: selection of activities, specification and/or modification. |
| Degree of Freedom | Through the degree of freedom the offered runtime flexibility is measured on a three-point rating scale: low, medium or high. |
| Recommender Mechanism |
| Purpose | This criterion determines the recommender mechanism’s actual purpose and, therefore, differs the three main application possibilities process navigation, specification support and modification support. Note that a comparison to the offered Flexible Runtime Procedures allows a simplified reasoning with regard to the question, whether the system makes use of intelligent user assistance to compensate drawbacks of its flexibility. |
| Provision of Information | As it is a major distinctive criterion for the delimitation of recommender-related functionality, the provision of information investigates the user interaction of the mechanism. Besides a pure search-based information retrieval, the user may enter into a dialogue (conversation-oriented) with the system to specify his problem [11;24]. In contrast an automated generation of suggestions provides its results in a proactive manner or directly chooses a certain solution and recommends it implicitly. |
| Recommendation Acceptance | When speaking of suggestions and runtime flexibility, it is mandatory for the user to have the freedom on deciding about the recommendation acceptance (arbitrary instead of imperative). |
| Explanation of Recommendations | Due to the fact that the acceptance and effectiveness of a RS depends on the transparency of its suggestion development [22;28], this criterion determines whether the offered recommendations are justified (none if no explanation is given) and if the offered explanation is adequate or rather insufficient. Note that this characteristic is fundamental for obtaining the user’s trust and enabling them in controlling the system. |
Recommendation Basis

<table>
<thead>
<tr>
<th>Recommendation Basis</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Derivation Procedure</strong></td>
<td>Within the scope of this criterion the typical procedures of deriving recommendations are differentiated, whereas no specific demands on the actual implementation are made. In other words, whenever users are compared the system makes use of a collaborative approach. Similarly the comparison of recommendation objects indicates a content-based functionality, while the utilization of knowledge coins a knowledge-based system. An explicit separation of hybrid approaches will be omitted.</td>
</tr>
<tr>
<td><strong>Recommendation Dimension</strong></td>
<td>This criterion refers on the dimension of the recommendation space and determines whether it is one-, two- or multidimensional. Note that for reasons of simplicity the progression in the process execution in terms of basic contextual information won’t be seen as a concept that constitutes a separate dimension.</td>
</tr>
<tr>
<td><strong>User Individuality</strong></td>
<td>If the recommendation mechanism distinguishes between users (none if not), the degree of personalization in the offered recommendations is measured on a three-point rating scale: low, medium or high.</td>
</tr>
<tr>
<td><strong>Context Sensitivity</strong></td>
<td>Similarly, this criterion estimates the systems degree of context sensitivity. If available (none if not), it is measured, again, on a three-point rating scale: low, medium or high.</td>
</tr>
<tr>
<td><strong>Process Type Dependency</strong></td>
<td>Through this criterion it is investigated, if the process type of a specific process instance affects the given suggestions (influencing instead of independent) or if the recommendation room is even restricted. Note that the latter may lead to a limited distribution of best-practices (only within the process type).</td>
</tr>
</tbody>
</table>

Development and Utilization of Best-Practices

<table>
<thead>
<tr>
<th>Development and Utilization of Best-Practices</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Inclusion of Experiences</strong></td>
<td>As it has been identified as the requirement for allowing the continuous development and utilization of best-practices, this criterion determines whether or not the experiences of an organization are included into the recommendation space in a dynamic manner (adynamic if otherwise).</td>
</tr>
<tr>
<td><strong>Recommendation Evolution</strong></td>
<td>The continuous development of business processes and a respective suggestion behavior demand mechanisms for the evolution of the recommendation basis. Thus, this criterion investigates whether (not considered if not) newer knowledge is preferred (weighting) or outdated know-how is even dropped (dispersion).</td>
</tr>
<tr>
<td><strong>Actuality of the Recommendation Basis</strong></td>
<td>Likewise, this criterion determines the actuality of the recommendation space by means of the differentiation between real-time and periodical updating.</td>
</tr>
</tbody>
</table>

Control and Influence

<table>
<thead>
<tr>
<th>Control and Influence</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Controllability through End-User</strong></td>
<td>Especially in the case of dynamic knowledge acquisition it is questionable how the system gets to know the user's preferences and situational coherences in order to evaluate the recommendation objects. However, this criterion doesn't investigate implicit user ratings, but rather the explicit exertion of influence like the rejection of recommendations (runtime feedback) [19;29]. Besides a performance evaluation after the process execution, it also may necessary to adjust misperceptions of the system (correction) [22].</td>
</tr>
<tr>
<td><strong>Administrative Influence</strong></td>
<td>Also the administration of the system demands certain influence mechanisms to adjust the recommendation behavior in order to correct misperceptions or rather to propagate new insights (correction). Moreover, in some situations it may become necessary to restrict the systems autonomy (specification of restrictions).</td>
</tr>
</tbody>
</table>

4.3. System Analysis

While Table 2 briefly describes the individual systems, the actual evaluation is illustrated in the subsequent Figure 1. Note that an extensive discussion and justification of the evaluation is omitted for reasons of complexity. As the given literature lacks in focus on the underlying topic, it is important to remark that all assumptions and implications are solely based on the cited sources.

Table 2. Description of the analyzed flexible PAIS

<table>
<thead>
<tr>
<th>@enterprise [30]</th>
<th>The commercial WFMS „@enterprise“ is a fully web-based solution that supports user-driven ad-hoc processes in addition to a regular process execution on the basis of predefined structures. While the latter does not feature any flexibility besides the application of simple change primitives, the former is totally detached from the coordination of the system. As a result, only during the rigid process execution, a process-oriented guidance is offered.</th>
</tr>
</thead>
<tbody>
<tr>
<td>AristaFlow® BPM Suite [31]</td>
<td>The commercial PAIS „AristaFlow® BPM Suite“ is an extensive adaptive WFM solution that emerged from the research project „ADEPT“. In addition to the structural modification of running processes, the system also supports evolutionary changes with an automatic schema migration. Although the developers aspire to implement an adequate user assistance for supporting runtime changes—besides some consistency checks—the system does not offer any guidance so far. The only process-oriented assistance that users obtain is the regular suggestion of single activities.</td>
</tr>
<tr>
<td>CAKE [27;32]</td>
<td>The „Collaborative Agent-based Knowledge Engine“ (CAKE) is an adaptive WFMS that provides novel suspension mechanisms to modify certain parts of a process without halting the whole execution. Though only simple change primitives may be applied, CAKE allows for the reuse of former modification cases through CBR. For this purpose, the research prototype performs a similarity search with regard to a textual problem description, the process structure and the names of the involved activities. Subsequently, the user may select a similar case to apply its modifications to the current process specification. The system also supports the underspecification of the underlying structures. However, the corresponding flexible runtime procedure doesn’t find a respective assistance.</td>
</tr>
</tbody>
</table>
In the commercial collaboration system „Caramba“ typical WFMS, groupware and project management functionalities are combined to a novel hybrid solution, which intends a process-oriented support especially for highly knowledge intense procedures. Besides the execution of ad-hoc processes, the system also allows a generic definition of process structures that may be modified during process runtime. Despite the high degree of flexibility, Caramba only offers process-oriented guidance when following a certain process schema, whereas flexible runtime procedures like the activity selection do not achieve any respective assistance.

The adaptive WFMS „CBRFlow“ is a research prototype that utilizes conversational CBR (CCBR) functionalities to assist runtime modifications through the possible reuse of existing change know-how. Hence, the users are required to document their treatment of exceptional situations by defining problem-related question-answer pairs. If the execution of a process requires any changes, the user is enabled to search for former cases in the respective „case-base“. For this purpose the assortment of cases could be concretized by answering some proactively displayed questions.

The flexible PAIS „Chameleon“ implements the approach „Pockets of Flexibility“. As a consequence, the research prototype focuses primarily on the underspecification of process structures. This is achieved through the possible specification of placeholder activities that, in each case, comprise a set of suitable process fragments and some constraints that have to be satisfied. In the course of runtime specification, an authorized user is charged in selecting a certain fragment, which is handled as a regular task within the process. Since this procedure isn’t supported by any guidance, the system only suggests single process activities based on respective process models.

The research prototype „Collaborative Task Manager“ (CTM) implements a novel approach for supporting ad-hoc processes through an email-driven delegation of single tasks. By monitoring the communication structures, the system offers the possibility to reuse so-called „task patterns“ that comprise descriptions to necessary activities, documents and possible delegations of sub-tasks to respective co-workers. Nevertheless, the required selection of existing patterns is search-based and the user does not obtain any help in finding the appropriate template.

In the European research project „Community-based Interoperability Utility for SMEs“ (COMMIUS) a novel concept has been developed that enables small- and medium sized enterprises (SMEs) in realizing process-oriented interoperability in a simple and affordable manner. For this purpose, the COMMIUS prototype analyzes the organizations email traffic and automatically generates the outlined process models. As a result, the system provides task-related information when handling single activities. Furthermore, the email-driven composition of process steps is assisted by a recommender mechanism that suggests possible activity sequences. This is based on the possible detection of a certain process type through analyzing the initial email that initiates a process-related dialog. Consequently, the system suggests the detected process type and the corresponding activity sequence, which evolves permanently with differing process executions. Besides organization-wide practices, COMMIUS also considers the working habits of its users by personalizing the offered assistance. Therefore, the individuality in the suggested activity sequences is controlled through a user classification with regard to the observable self-reliance in process execution.

The WFMS „Declare“ is a prototypical implementation of a declarative approach that allows for an highly flexible process execution. As it is the main functionality of declarative systems, the sequencing of single activities specified for a certain process type is only limited to the satisfaction of predetermined constraints. Similarly to a missing assistance during possible modification procedures, the resulting degree of freedom in choosing from available activities doesn’t find any guidance. Interestingly, there are two possible extensions to Declare, which are analyzed separately.

This first extension to the Declare system prototypically demonstrates the possible integration with the process mining framework „ProM“ in order to accommodate the inherent lack of process-oriented guidance. For this purpose, a novel recommendation service evaluates the utility of alternative activities during process execution. Based on the comparison between the partial activity sequence and former procedures of handling a certain process type, the mechanism infers the value of benefit with regard to a specific target function. As a result, the system displays a ranking of all alternatives in a proactive manner, whereby the user obtains a target-oriented assistance in selecting activities.

While this second approach also incorporates Declare and ProM, the actual intension fundamentally differs from the above. Instead of a process execution based on a certain process specification, the product based workflow design (PBWD) focuses on a data-oriented support for information-driven business processes. Therefore, a product data model (PDM) determines possibly multiple compositions of single activities that may lead to a successful process completion. In order to support the corresponding decision making, a special recommendation service evaluates occurring alternatives with regard to a specific optimization strategy. The latter either specifies the overall target-oriented optimization of a process or simply the isolated best selection of the respective upcoming activity. Note that in this prototypical implementation the main functionality of Declare is omitted.

Through integrating the adaptive WFMS „ADEPT2“ (see AriaFlows) with an extension of the CBRFlow methodology, the research prototype „ProCycle“ introduces a WFM solution that supports the process lifecycle in a whole. In doing so, the system assists possible modification procedures of the ADEPT2-component by offering former cases to reuse their problem-solving knowledge. Besides the specification and respective manual answering of non formal questions, the CCBR-component also supports the automatic recognition of relevant contextual information in order to find similar cases. Furthermore, the process-type-specific assortment of cases is reduced to currently applicable modifications and enriched with relevant information like the usage frequency or its overall user evaluation. As the latter indicates, ProCycle provides the feasibility to write and read reviews to the existing modification know-how.

The „TIBCO iProcess™ Suite“ is a commercial solution that applies various methods to improve the offered flexibility. Besides an easy administrative adaption of the process structures through the separation of process and business logic, the system allows for the selection of concrete sub processes at designated positions during runtime (late binding). Furthermore, a special component makes it possible to manage the execution of several sub processes in a target-oriented manner. However, this takes place at a higher level and the regular handling of the business cases does not feature any flexibility but the specification procedures, which in turn is not assisted by the system.

The adaptive WFMS „WASA2“ is a prototypical implementation of an object-oriented concept that allows for the flexible association of process models. Thereby, it is possible to assign different specifications to a certain instance at different points of time in order to compensate exceptional situations. Even though such procedures are highly knowledge intense, the system does not assist the user during the modification of the process structure or the selection of alternative existing models. The only runtime guidance WASA2 offers is the regular assignment of single activities.
The implementation of the „Worklets“ approach into the open source WFMS „YAWL“ introduces two different functionalities to make the process execution more flexible. On the one hand, „Worklets“ allow for the individual selection of specific sub processes at designated positions (late binding). Thereby, the respective placeholder element holds an extendable set of sub processes from which a rule engine automatically selects the appropriate fragment. On the other hand, „Exlets“ prescribe defined procedures of handling exceptional situations in a certain process type either through the application of simple changes or the execution of compensating processes. Since in both mechanisms only administrators may intervene in the selection procedure, the given degrees of freedom do not result in flexible runtime procedures. As usual, the regular process execution is based on the assignment of activities.

**Figure 1: Evaluation of the analyzed flexible PAIS**

---

For reasons of clarity, the approaches have been clustered as follows: Firstly, the approaches were separated according to the purpose of their recommender component in question. Secondly, the first cluster has been differentiated further into highly and less flexible systems.
5. Evaluation and Inference

This section presents the main findings of an analysis regarding the preceding evaluation and, thereby, assesses the SOTA. In the course of this, it is initially determined to what extent the necessity of intelligent user assistance has been recognized so far. Subsequently, the utilization of recommender-related functionalities is delimited and the actual exploitation of the RS technology is implied.

In order to analyze the prevailing awareness concerning the requirement of adequate user assistance, the supported flexible runtime procedures and the purpose of the recommender mechanism in question were compared (cf. M1 in Figure 1). However, it is important to remark initially that the systems @enterprise, Caramba and Declare do not assist their users in the given flexible runtime procedures, which is why they could already be excluded from all further examinations. By matching the respective criteria as mentioned, it could be observed that the need for assisting users has only been recognized in some research prototypes and mostly in very flexible systems (cf. M2). While the greater part supports its users in selecting activities, a few concepts focus on the assistance of modification procedures.

Based on the delimitation criteria defined in section three, it becomes apparent that only four of the fifteen analyzed systems possess capabilities that are related to the original recommender technology: COMMIUS, Declare & ProM A, Declare & ProM B and ProCycle. In fact, three of these systems, again, focus on guidance through process navigation and only ProCycle concerns about modification procedures. Similarly to the observations described above, first and foremost very flexible solutions apply the RS technology. With regard to the actual implementation of the recommender functionalities (cf. M3), the delimited group of systems mostly provides suggestions in a proactive manner. The acceptance of the given recommendations is basically within the user’s discretion. However, the necessity of explaining suggestions is mostly unrecognized, since the systems with an adequate explanation just do not have the need for further justification, but at the same time do not implement comprehensive functionalities. In general, the derivation procedures are knowledge-based, whereas only Declare & ProM A additionally compares recommendation objects in order to determine utility values. Even though all systems operate on a multidimensional recommendation basis, this does not imply any strengths or weaknesses of the respective systems, as typically at least single activities in certain process types are suggested. In contrast, it appears as a crucial fact to determine that only COMMIUS personalizes its recommendations and, thereby, supports the individual working habits of its users. Basically, all systems take further information into account than only the advancement of the process execution. Although, the recommendation basis of all approaches is restricted to a certain process type.

The latter could lead to a delimited dispersion of best-practices, whose dynamic development and utilization has mostly been recognized as a certain potential of the technology. However, for the greater part, the acquired knowledge is not always up to date and only COMMIUS supports the evolution of its recommendations. Even though especially the dynamic acquisition of knowledge demands the possibility to control the systems behavior, users basically do not have any scope of explicit influence. In contrast, there are mostly implicit opportunities to influence the system from an administrative perspective.

6. Conclusion and Summary

This paper investigated profoundly the systematic application of RS in order to increase the process-oriented guidance offered in flexible PAIS. For this purpose, the work in hand has identified various fields of application that correspond to certain flexible runtime procedures. As a result of an extensive discussion about the technological adoption, concrete design options and application advantages have been determined. Besides the intended increase of guidance, especially the possible inbuilt continuous development and utilization of best-practices appears to be a crucial benefit of using the recommender technology.

Through the comprehensive SOTA analysis given by this paper, it could be observed that the necessity of assisting users has only been recognized within some research prototypes. In particular, only a few and mainly very flexible systems apply recommender mechanisms that are somehow related to the original technology. While some concepts have already discovered the potentials of a dynamic knowledge acquisition, most of the corresponding design aspects are disregarded so far. The implemented mechanisms possess only marginal analogies to the functionality of typical RS. In this respect, the existing application

---

3 Note that this is not directly implied by the evaluation criteria in the applied framework. While @enterprise and Caramba allow an adhoc composition of business processes, their recommender component in question solely focuses on the execution of structured processes. In turn, Declare “suggests” all possible activities and does not support the user in a respective decision making.

4 In general, all other approaches do not fulfill all requirements (cf. the respective criteria in Figure 1).
potentials of the recommender technology are only exploited to a minor degree.

Therefore, further research activities are required in order to develop novel concepts that incorporate intelligent user assistance in terms of recommender capabilities. In doing so, first and foremost profound methods are needed to gather, maintain and evolve the recommendation basis. Moreover, mature derivation procedures for the suggestion development and comprehensive functionalities for the administrative controllability are required.

7. References


