

KISS – Knowledge-Intensive Service Support: An approach for agile process management

Daniela Feldkamp, Knut Hinkelmann, Barbara Thönssen

University of Applied Sciences Northwestern Switzerland, School of Business,
Riggenbachstr. 16, 4600 Olten, Switzerland
{daniela.feldkamp, knut.hinkelmann, barbara.thoenssen}@fhnw.ch

Abstract. Automating business processes especially in the tertiary sector is still a challenge as they are normally knowledge intensive, little automated but compliance relevant. To meet these requirements the paper at hand introduces the KISS approach: modeling knowledge intensive services by enriching business rules semantically and linking rules to processes. We defined four types of rules, each focusing on a specific problem (resource allocation, constraints checking, intelligent branching and variable process planning and execution). For knowledge formalization we provide a 3-phase-procedure starting with a semi-formal representation (business model), followed by a formal representation (interchange model), leading to a machine executable representation (execution model).

Keywords: Agile Business Process, Process Modelling, Business Rules, Business Rules Formalization, Variable Process

1 Introduction

Business process management has been very successful for structured processes with the objectives of process optimization, quality management, implementation of business information systems, or workflow management. In actual applications, however, we still face various problems: Often process documentations are not in line with the real work in the organization, e.g. because the processes are not implemented as documented or because processes have changed and the documentation is not adjusted. Also, process definition often lack the right level of granularity, i.e. they are very detailed forcing participants to follow a rigid regime and prohibiting flexibility in process execution.

The situation is even worse for knowledge-intensive or dynamic processes as they have to deal with

- exceptional situations
- unforeseeable events
- unpredictable situations, high variability
- highly complex tasks.

As Adams, Edmond and ter Hofstede [2] state are WfMSs designed to support rigidly structured business processes but not knowledge intensive processes which are

by nature weakly structured and do not match at least the one crucial condition for process automation: A high repeatability rate, i.e. doing the same thing in the same way many times. Consider for example a claim process in a company: Approving a claim for damages may vary from simply checking whether the loss is covered by the insurance contract to complex evaluation including locally inspection, verification by an expert, examination of the damaged goods, checking on beneficiaries to avoid fraud etc.

If possible at all, a process model covering all possible cases would be too complex to be manageable. Typically, decisions on those actions like who has to perform the task, in which order by when, with what result etc. requires knowledge and experience and therefore are taken by a domain expert. As the process varies very much, it is normally not automated and often not even documented in detail.

Disadvantages of that situation are lack of transparency and traceability of work, inconsistent decisions and not taking into account company's regulations. Especially the increasing demands on governance and compliance have been forcing companies as well as public administrations in the last few years to review these kinds of processes for improvement.

That brings the subject of 'Business Rules' into the picture. Business rules allow for an explicit and consistent specification of regulations [16]. They provide an excellent means of encapsulating knowledge about the intentions of business and their implementation in business processes and information systems [13].

In this paper we present a new approach for agile business process management combining the strengths of both business processes modelling and business rules. Business process models are kept as simple as possible but complex enough to allow for process optimization and automation. To cope with the requirements for flexibility, consistency and compliance we integrate process models with business rules. Making consistent business rules available for all stakeholders of a business process (that is suppliers, employees, partners, clients etc.) reduces the drawback of process and workflow management when applied to knowledge intensive tasks.

2 The KISS approach

The approach introduced in this paper combines business rules and process models in order to automate knowledge intensive services by taking advantage of both fields:

- process models are used for
 - explicit documentation and visualisation
 - execution automation.
- rules are used for
 - resource allocation
 - constraint checking
 - branching (of processes) and decision making
 - process identification and selection

To allow for flexible process execution we introduced a new modelling construct that we call *variable activity*. A variable activity is closely related to a knowledge-intensive task as introduced by Abecker et al. [1]. A variable activity corresponds to a

subprocess with the particularity that the activities of the subprocess are determined at run-time instead of strictly modeling them.

The following example introduces the approach: In an insurance company the process for claim checking consists of five activities:

1. check claim for formal correctness
2. check claim for consistency with contract and decide whether to
3. rejected claim or
4. evaluated claim in detail
5. decide on claim.

Note that activity 4 (evaluate claim) is a variable activity. Depending on the specific claim one or more activities or sub-processes have to be executed. It can range from simply checking whether the loss is covered by the insurance contract to complex evaluation including locally inspection, verification by an expert, examination of the damaged goods, checking on beneficiaries to avoid fraud etc. Thus, the process model consists of activities that are fixed (1, 2, 3 and 5) and one activity (4) that is variable and will be determined not until runtime.

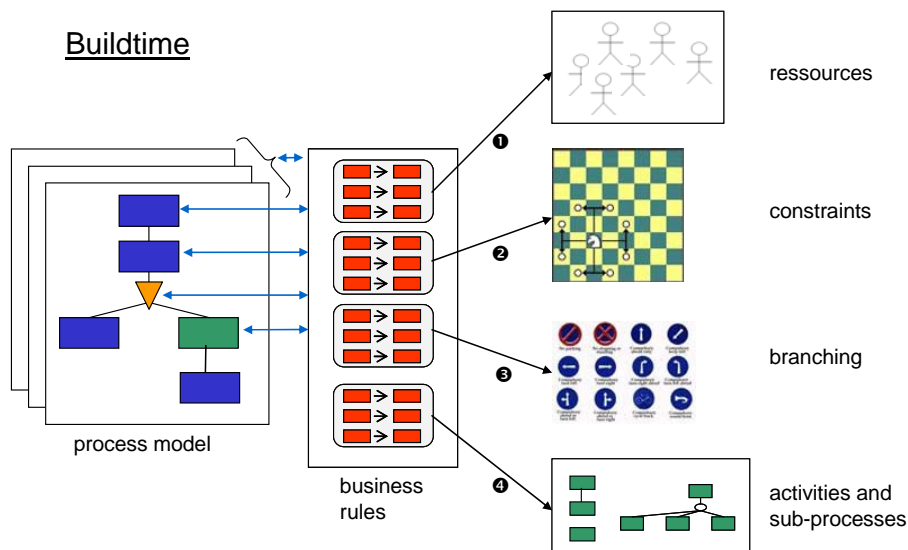


Fig. 1. KISS Buildtime Activities

Fig. 1 depicts the build time activities for knowledge intensive claim checking process. The process model consists of the five activities. The blue coloured are the pre-defined activities whereas the green one indicates the variable activity. The business rules are grouped for resource allocation (1), constraint checking (2), branching decisions (3) and activity or sub-process identification and selection (4).

Business rules are linked to the process model. Note that same business rules can be used by several activities and processes.

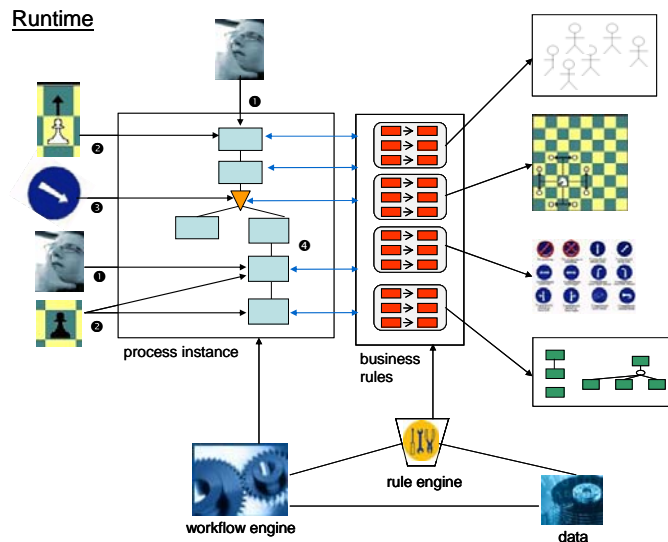


Fig. 2: KISS Runtime Activities

Fig. 2 illustrates an instance of the above process model—the variable activity (4) is expanded to two activities. The process instance is executed by a workflow engine that triggers the rule engine. Both workflow and rule engine have access to application, process and domain data. During runtime the linked rule sets are executed and

- (1) the appropriate resources are allocated (the example just shows one domain expert allocated to two activities)
- (2) constraints are checked (for example whether the accepted claim does not exceed the insurance sum or whether the client belongs to the platinum, gold or silver category)
- (3) branching is fixed (e.g. as it is a platinum client the tolerance is higher and the claim is accepted) and
- (4) the variable activity is expanded by determining the concrete activities or sub-processes (e.g. examination of the damaged goods and computing the amount of coverage).

This small example illustrates how knowledge intensive services can be modeled in a way that is flexible but is still eligible for automation.

In the following chapters we describe how to model knowledge intensive processes in the outlined way and how to execute these processes automatically.

3 KISS in depth

Business process management is one approach to make business agile. In their seminal book on Business Reengineering, Hammer and Champy [6] mentioned adaptivity and changes as important challenges of business. This in the end led to workflow management systems. However, it was realized quite soon, that workflow management systems are mainly useful for production oriented business processes, because these processes are well-structured. But, they lack flexibility in process execution. To achieve more flexibility also for weakly structured processes, the workflow management systems were advanced toward adhoc workflows, in which the participants can modify the workflow at run time. The flexibility is achieved by an intervention of a human. There are only a few approaches toward system-supported agility [20], [10]. But these approaches are limited in the sense that they rely on predefined workflow or activities. They mainly deal with modifying the logical structure of the workflow neglecting aspects like flexible resource allocation or decision making.

An alternative approach towards agility takes into account that every business application is based on rules to conduct the business logic. [19],[15],[17] When compliance requirements increased, along with the demand for business flexibility the business rules approach emerged.

To combine the advantages of well-structured processes with the demand for adaptivity and flexibility, the integration of business rules with business process management can be regarded as promising.

Roger T. Burlton expressed: “Separate the *know* from the *flow*.” (cited by Ross [15]) The implication is that the know part and the flow part are different. While business processes specify a process flow that has to be followed, e.g. for reasons of efficiency and traceability, business rules support flexibility and customization. In addition to intelligent branching and deriving values for application data (see Xpert.ivy [14] as an example) rules can be utilized to allow for context-specific adaptation of process steps. For instance, instead of assigning workflow participants because of very general roles, a rule-based approach would be able to select participants because of individual skills (e.g. select the person that had formerly solved a similar task). Additionally, separating the know from the flow has the advantage to achieve adaptivity and flexibility. In this chapter we first introduce business rules and the KISS approach how to relate rules to activities. After it, we describe a 3-phase procedure to model agile processes combining business process modelling (the “flow”) and business rules (the “know”).

3.1 Business rules

The Business Rules Group [4] defines a business rule as “a statement that defines or constraints some aspect of the business. It is intended to assert business structure or to control or influence the behavior of the business.”

There are several classification schemata for Business rules formalization, like Endl[5], Herbst [7], Ross [15], [16] and von Halle[19]. Since it is well understood and

comprehensive for KISS we use the classification given by Barbara von Halle [19]. The rule classification depicts business rules into terms, facts and rules.

A term is a phrase which has a specific meaning for the business in some context; a fact is a statement which expresses relationships between two or more terms. Rules are a declarative statement that applies logic or computation to information values.

Following the definitions of von Halle, rules are split into five sub-classes:

- **Mandatory constraints**

Mandatory constraints are statements which have to be kept at any time. To express this kind of rules the auxiliary verb “must” is used.

- **Guidelines**

Guidelines are rules, which could be followed but must not. The auxiliary verb “should” is used to express guidelines.

- **Action enabling**

Action enabling rules trigger another rule or a process (step), if the condition holds.

- **Computations**

Computations are rules to perform calculations, like sum and difference.

- **Inferences**

Inferences are statements which establish the truth of a new fact, if the condition holds.

3.2 Relation between rules and activities

Separating business rules from activities and storing the rules independent from the processes in a repository implements the separation of know and flow (see above). It has the advantages, that rules can be modified independent from the business logic, thus leading to more flexible and adaptable business processes. Additionally, since a single rule can be associated to several activities, business logic is more reusable.

We identified four different ways of relating rules to activities:

- Variable process execution for flexibility
- Intelligent resource allocation at run time
- Constraint checking
- Intelligent branching and decision making

This means, that an activity of a process can have relations to four different rule sets, one rule set for each of the above mentioned relation types.

As the benefits of separating but linking business process and business rules are obvious, leading software producers like ILOG and IBM already implemented an interface for example for constraint checking and decision making [9]. However, not resource allocation and least of all variable process execution have been investigated in depth. In the following the four types of relations are explained, but focussed on the variability of processes..

3.2.1 Variable process execution for flexibility

Each variable activity is related to a set of action-enabling rules (maybe together with inference and computation rules), that select the appropriate activities and sub-processes. *This rule set is invoked first* to dynamically determine and instantiate the

appropriate actions. They trigger planning and refinement of a process by selecting from predefined activities e.g. for requesting additional documents or notifying other administration units.

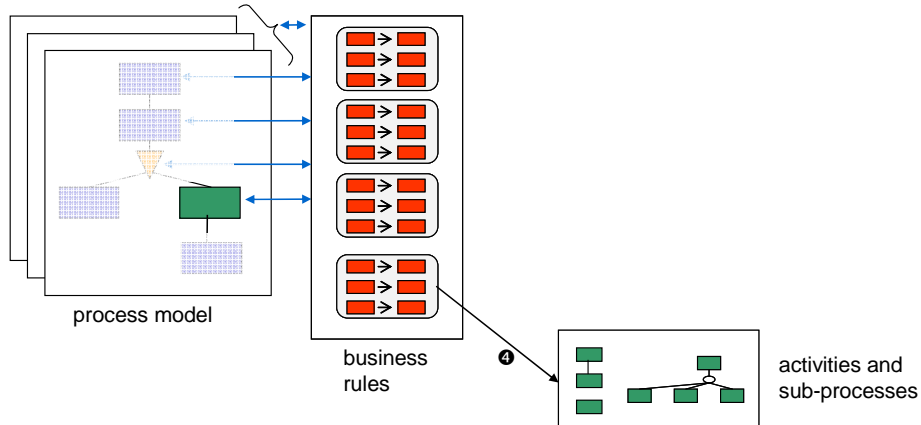


Fig. 3: Variable process execution

Regarding the claim example introduced in chapter 2, the fourth activity “evaluate claim” is a variable process, containing the following four activities:

- (A) Request documents
- (B) CallInsuranceSurveyor
- (C) Conduct Interview
- (D) FileWitnessReport

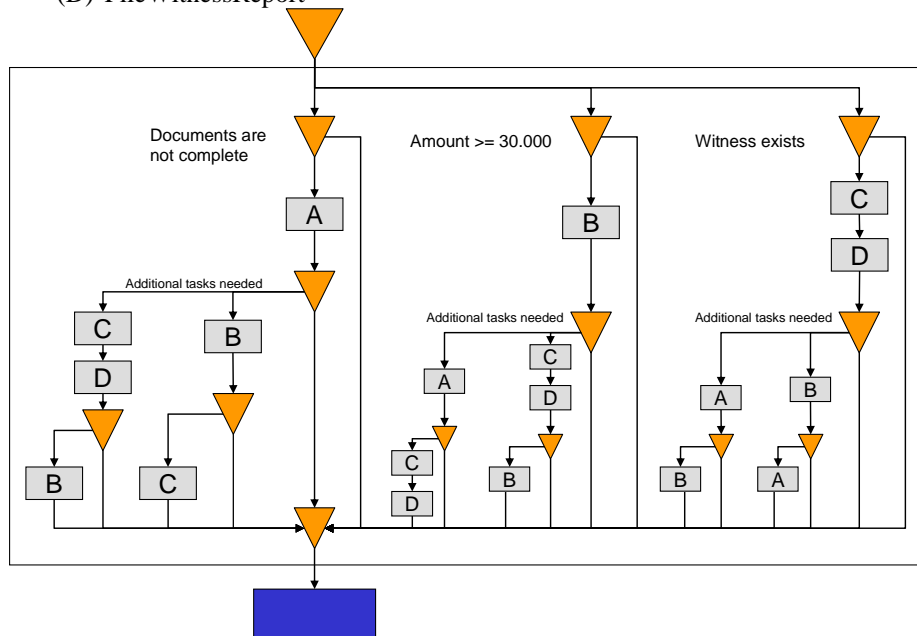


Fig. 4: Traditional process models

Fig. 4 illustrates the process in the traditional way. For example to approve a application it may be necessary to conduct several checks including requesting documents (activity A), interviewing witnesses (sub process containing a sequence of the two activities C and D) and calling insurance surveyor (activity B). All tasks depend on each other. The control flow expresses that the outcome of one task (e.g. A) may require the involvement of the other task (e.g. B and/or C, D). However, the outcome of one task also may be that the application will fail; then no further tests are required. Incorporating this behaviour in the control flow would increase the complexity enormously. In fact not all aspects of the control can be modeled in an appropriate way.

This variable process is replaced in the traditional model by a new object type, labeled with "KIT" (Knowledge-intensive task). This object is related to a pool of activities, which are linked to action-enabling rules. At run time the associated action-enabling rules select the activities that have to be executed depending on the actual context of the process instance.

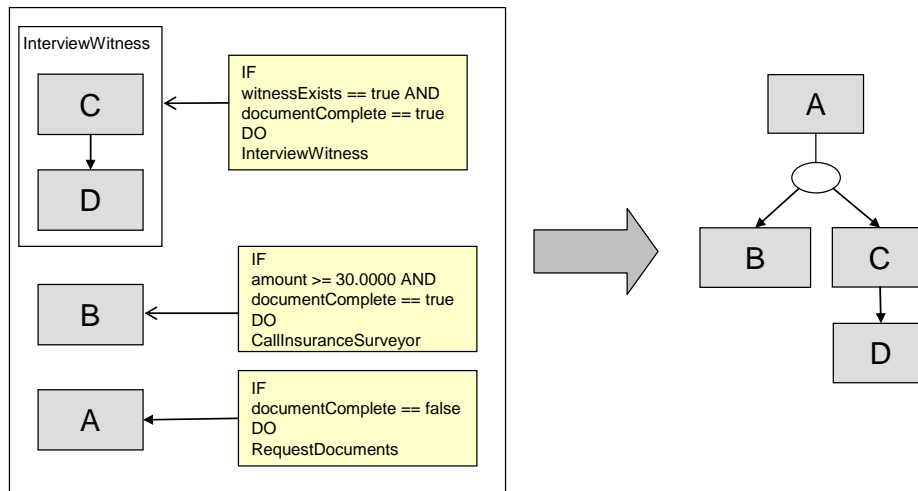


Fig. 5: Pool of sub-processes and activities

Fig. 5 illustrates the service model created using the KISS approach. Each of the three activities and sub processes is related to an action-enabling rule.

That the outcome of an activity may require the involvement of another activity is not express explicitly. As an effect of the invocation the information can be modified and stored for example in a database. As a "side effect" further rules can be fired probably resulting in the instantiation of further activities. For example, if the client did not have sent all required documents, a request must be sent to him before executing the processes "Interview Witness" and "CallInsuranceSurveyor". If all required documents are sent and the amount is less than 30.000, but witnesses exist, the process "InterviewWitness" will be invoked. Otherwise, if both conditions hold, the processes will be executed in parallel.

3.2.2 Intelligent resource allocation at run time

An activity can reference a rule set with inference and computation rules for context-specific selection of

- human resources (e.g. based on specific skills),
- information sources (e.g. using appropriate forms or adapt information presentation depending on user category) or
- IT resources (e.g. selection of a particular web service)

This rule set is applied *before invocation of the activity*.

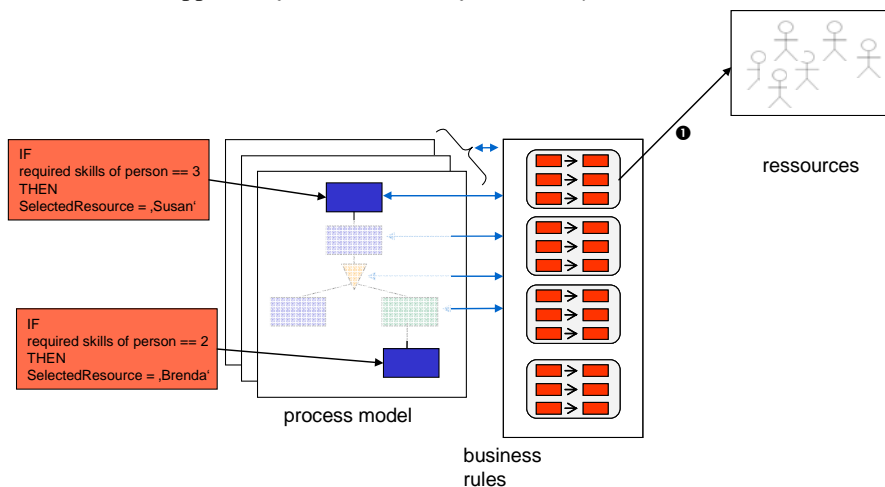


Fig. 6: Resource allocation

In the claim example, the first activity is executed by Susan, if the required skills are 3 and the last activity will be assigned to Brenda, because her skill is 2.

Even though resource allocation is a standard workflow management system functionality the KISS approach offers the opportunity of much more complex queries than for selecting appropriate roles and skills. Because of the expressiveness of business rules other sources than only the WfMS can be evaluated, e.g. yellow pages, human resource management systems and case or project databases etc. Including specific information or knowledge management systems with the help of business rules is a smooth way to increase the adequateness of allocated resources.

3.2.3 Constraint checking

Another reference relates an activity to a rule set of mandatory constraints and guidelines (maybe together with inference and computation rules). These are checked *during run-time of the activity*.

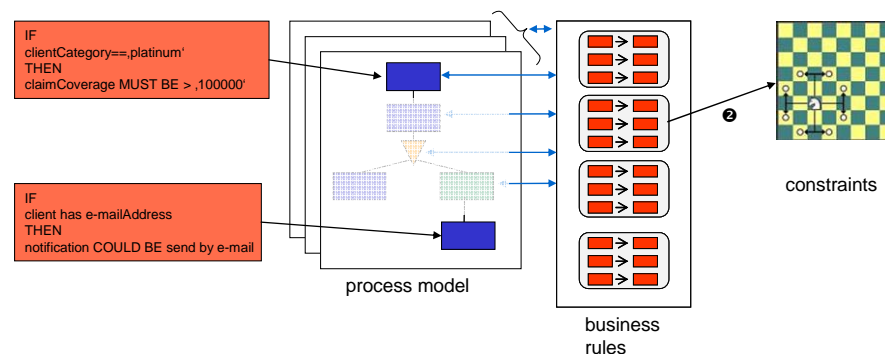


Fig. 7: Constraint checking

Regarding the example the first activity is related to the constraint, if the category of a client is “platinum” then the claim coverage must be higher than 100.000. If this constraint is violated, an exception must be thrown.

3.2.4 Intelligent branching and decision making

Inference rules and computation rules for decision making can be associated to both activities and gateways. They derive workflow-relevant data on which the branching depends.

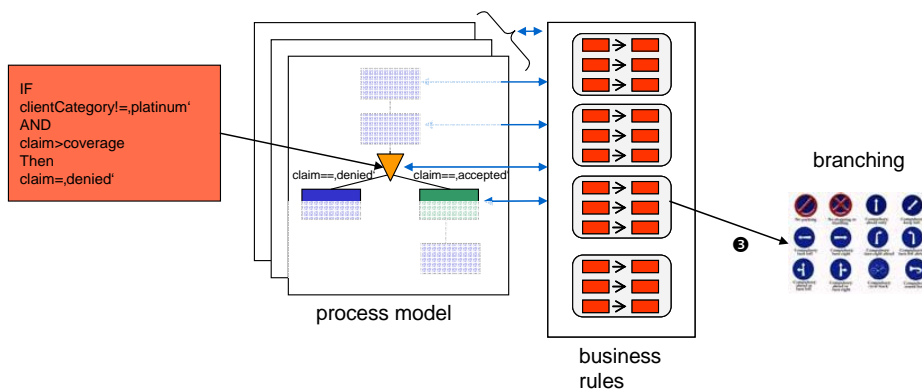


Fig. 8: Branching

Regarding the example, a gateway is related to an inference rule. If a client category is not platinum but the claim is higher than the coverage the claim is denied. If the claim is accepted, the next activity “evaluate claim in detail” is invoked. Otherwise, if the claim is denied, the claim will be rejected.

3.3 Agile Process Modelling Approach

To make the “know” explicit we introduce a 3-phase procedure to model adaptive processes and business rules. The first phase deals with capturing rules, drafting a process model and linking rules and processes. In the second the rules and process model are transformed into an interchange form and in the last phase the rules and the process model are transformed into a machine executable form.

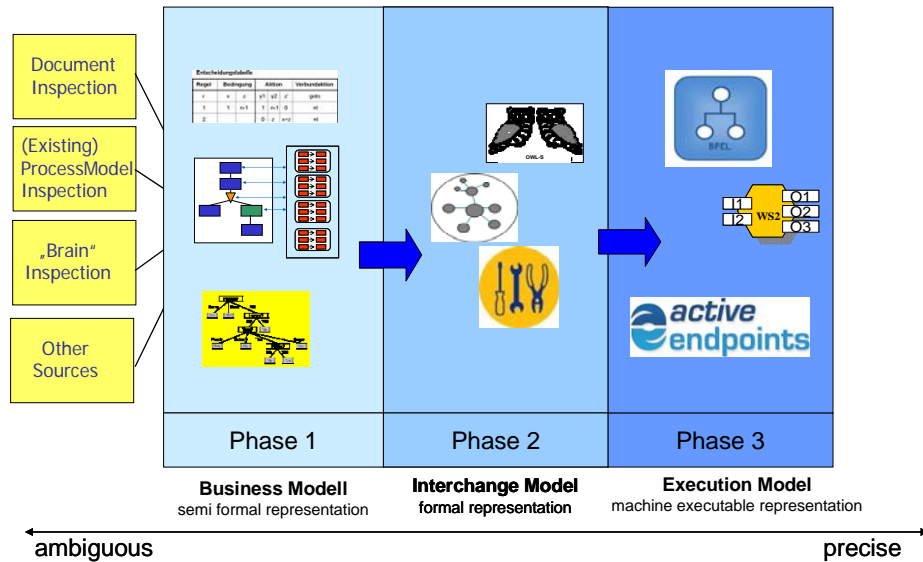


Fig. 9. Agile Process Modelling Approach

3.3.1 Phase 1

The first phase of designing agile processes focuses on knowledge capturing from business users. In general, it is easier for business people to sketch a process model – how business should be done – so it is recommended to start with a ‘process skeleton’ (q.v. Step 1). Business people normally express regulations in natural language that is by nature ambiguous, often not very precise, may be inconsistent, incomplete etc. Therefore the focal point of rule capturing is to phrase rules in a way that overcomes the drawback of natural language problems but is still well understandable.

Capturing rules and modelling the process is regarded as an iterative procedure as in the beginning it is often not very clear what should be modeled as an activity (atomic process) and which part should be better modeled as business rules.

3.4 Phase 2

The second phase of designing adaptive processes is focused on transformation of the models into a precise, machine understandable form.

The purpose of this interchange model derived in this phase is twofold: First, the rule of the business models are in a format that can be easily understood by business people. Therefore a semi-structured representation was chosen that contains natural language text. This, however, has the disadvantage that the rules can be ambiguous. In order to be validated and executed, the business rules and processes have to be represented in a language with well-defined semantics. Second, there can be different run-time environments for the execution of business processes. The interchange

format shall serve as a common language from which the execution formats can be derived unambiguously, if possible even automatic.

To fulfill these purposes the interchange format must have a clear and precise semantics. We have pointed out, that business rules systems only have simple formalism with weak semantic for representing facts and terms. Extending the expressiveness towards ontologies has the advantage of higher expressiveness and the chance to use inferences like inheritance and consistency checking. As consequence, since the procedural knowledge must be highly integrated with declarative knowledge, a rule language must be available in which all rule types can be expressed and which can be integrated with ontologies. Therefore, we use OWL [14] and SWRL [8] to express terms, facts and business rules, while the process models are represented in OWL-S [12]. Because of the partially ambiguous business models this procedure is not automated: the process model has to be transferred into OWL-S and the rules into OWL respectively into SWRL manually. However, the semi-structured representation of the business models the development of a semantic representation is straightforward.

3.5 Phase 3

In the last phase the interchange model created in second phase will be automatically migrated into machine executable forms if necessary. For example OWL-S must be transformed into BPEL to execute the workflow. Rules can be exported into e.g. Java or JavaScript.

4 Conclusion

In this paper we presented the KISS approach for agile process management. It leads to more flexible and agile business processes by integrating business process management with business rules in three different ways:

Variable activities are linked to action-enabling rules. This leads to flexible process execution, because the final process flow is determined at run-time. In contrast to ad-hoc workflows, where a human participant decides to deviate from a predefined process flow, traceability and compliance can be controlled by a set of rules.

Resource allocation is more flexible. For example, participant can be selected based on individual skills instead of general role specifications.

Intelligent branching is supported by business rules representing expert knowledge.

In addition, violation of constraints and guidelines are determined immediately at run-time avoiding expensive repair actions.

With the KISS approach, process models can be kept small and simple (following the slogan “KISS = Keep It Simple and Small”) representing only the process flow while the knowledge is separated in the business rules. Separating business rules from activities and storing the rules independent from the processes in a repository has the advantages, that rules can be modified independent from the business logic. Additionally, one rule can relate to several activities, so the business logic is more reusable.

An additional advantage of our approach is achieved in that the expressiveness of business rules is extended towards ontologies allowing for inferences like inheritance and consistency checking.

The KISS approach has been developed in the FIT project¹ and is applied for e-government services of the Austrian city of Voecklabruck.

5 References

1. Abecker, A.; Bernardi, A.; Hinkelmann, K.; Kühn, O.; Sintek, M.: Toward a Well-Founded Technology for Organizational Memories. IEEE Intelligent Systems and their Applications, Vol. 13, Nummer 3, S. 40 - 48, May/June 1998. Reprint in: James W. Cortada, John A. Woods (Eds.), The Knowledge Management Yearbook 1999-2000, Butterworth-Heinemann, (1998), pp. 185-199
2. Adams, M.; Edmond D. and ter Hofstede, A. H. M.: The Application of Activity Theory to Dynamic Workflow Adaptation Issues, In The Seventh Pacific Asia Conference on Information Systems (PACIS-2003), Adelaide, Australia, (2003), p 1836 – 1852.
3. Andrews, T.; Curbera, F.; Dholakia, H.; Golland, Y.; Klein, J.; Leymann, F.; Liu, K.; Roller, D.; Smith, D.; Thatte, S.; Trickovic, I.; Weerawarana, S.: Business Process Execution Language for Web Services. (2002) Available at: <http://www-128.ibm.com/developerworks/library/specification/ws-bpel/>, Retrieved April 4, 2007.
4. Business Rules Group: Defining Business Rules ~ What Are They Really? (2000) Available at: http://www.businessrulesgroup.org/first_paper/BRG-whatisBR_3ed.pdf, p. 4., Retrieved: 19.03.2007
5. Endl, R.: Regelbasierte Entwicklung betrieblicher Informationssysteme, Gestaltung flexibler Informationssysteme durch explizite Modellierung der Geschäftslogik. Wirtschaftsinformatik, Band 45, Josef Eul Verlag, Bern (2004)
6. Hammer, M. and Champy, J. 1993, *Reengineering the Corporation*, Harper Collins Publishers, New York.
7. Herbst, H.: Business Rule-Oriented Conceptual Modeling. Physica-Verlag, Heidelberg (1997)
8. Horrocks, I.; Patel-Schneider, P.; Boley, H.; Tabet, S.; Grosz, B. and Dean, M.: SWRL - A Semantic Web Rule Language, Combining OWL and RuleML. (2004) Available at: <http://www.w3.org/Submission/SWRL/>, Retrieved: June 12, 2006.
9. ILOG, <http://www.ilog.com/>
10. Karagiannis, D. and Hinkelmann, K.: Context-sensitive office tasks – a generative approach. Decision Support Systems. Vol. 8, North-Holland, Elsevier Science Publishers B.V. (1992) 255-267
11. Lienhardt, H. and Künzi U.: Workflow and Business Rules: a Common Approach. (2006) Available at: <http://www.bptrends.com/>, Retrieved: March 18, 2007
12. Martin, D.; Burstein, M.; Hobbs, J.; Lassila, O.; McDermott, D.; McIlraith, S.; Narayanan, S.; Paolucci, M.; Parsia, B.; Payne, T.; Sirin, E.; Srinivasan, N. and Sycara, K.: OWL-S: Semantic Markup for Web Services. (2004) Available at: <http://www.w3.org/Submission/2004/SUBM-OWL-S-20041122/>, Retrieved: March 22, 2007
13. Morgan, T.: Business Rules and Information Systems: Aligning IT with Business Goals. Addison-Wesley, Boston (2002)
14. McGuinness, D. L. and von Harmelen, F.: OWL Web Ontology Language Overview. (2004) Available at: <http://www.w3.org/TR/owl-features/>, Retrieved: March 19, 2007

15. Ross, R. G.: Principles of the business rules approach, Addison-Wesley Professional, Boston (2003)
16. Ross, R. G.: Business Rules Concepts- - Getting to the Point of Knowledge. Business Rule Solutions, LLC, Houston (1998)
17. Schacher, M. and Grässle, P.: Agile Unternehmen durch Business Rules, Springer-Verlag Berlin Heidelberg (2006).
18. Soreco Group, Was ist Xpert.ivy? Available at: http://www.soreco.ch/ivy/pro/soreco/WebSite/index.jsp?navId=Products/xpertivy/was_ist_xivy, Retrieved April 3, 2007
19. Von Halle, B.: Business Rules Applied, Building Better Systems Using the Business Rules Approach. Wiley and Sons, New York (2002)
20. Wargitsch, C., Wewers, T., and Theisinger, F.: WorkBrain- Merging Organizational Memory and Workflow Management Systems. In: Abecker, A., Decker, S., Hinkelmann, K., and Reimer U. (Ed.) Knowledge-Based Systems for Knowledge Management in Enterprises, Workshop held at the 21st Annual German Conference on AI, Document D-97-03, DFKI Kaiserslautern (1997)

¹ FIT (Fostering self-adaptive e-government service improvement using semantic Technologies) is a project funded by the European Commission within the IST programme, IST-2004-27090