Making Processes from Best Practice Frameworks Actionable

Sven Graupner, Hamid R Motahari-Nezhad, Sharad Singhal, Sujoy Basu

HP Laboratories
HPL-2009-196

Keyword(s):
Best practice frameworks, Business processes, Process-aware Knowledge Repository

Abstract:

Best-practice frameworks provide guidance for organizing work in business. They enable reuse of experience within a domain. However, best practice frameworks are general and usually cover broad domains. Their guidance thus is often offered at an abstract level rather than as details of actionable tasks and processes to accomplish work. This paper presents an approach to bridge the gap between the abstractions available in best practice framework and actions that have to be performed by people or systems in a repeatable manner. We identify knowledge from best practices frameworks, categorize it and represent it in the form of reusable, interpretable templates. Template interpretation guides the refinement process from general concepts of best practices frameworks into actionable concepts such as specific tasks to be performed by assigned roles. A prototype implemented to validate the approach is also described.

External Posting Date: September 6, 2009 [Fulltext] Approved for External Publication
Internal Posting Date: September 6, 2009 [Fulltext]

Published in the 3rd Business-driven IT Management (BDIM 2009), Hofstra University, Long Island, NY, June 1-5,2009.

Making Processes from Best Practice Frameworks Actionable

Sven Graupner, Hamid R. Motahari-Nezhad, Sharad Singhal, Sujoy Basu
Hewlett-Packard Laboratories
1501 Page Mill Road, Palo Alto, CA 94304, USA
{sven.graupner, hamid.motahari, sharad.singhal, sujoy.basu}@hp.com

***Abstract*—Best-practice frameworks provide guidance for organizing work in business. They enable reuse of experience within a domain. However, best practice frameworks are general and usually cover broad domains. Their guidance thus is often offered at an abstract level rather than as details of actionable tasks and processes to accomplish work. This paper presents an approach to bridge the gap between the abstractions available in best practice framework and actions that have to be performed by people or systems in a repeatable manner. We identify knowledge from best practices frameworks, categorize it and represent it in the form of reusable, interpretable templates. Template interpretation guides the refinement process from general concepts of best practices frameworks into actionable concepts such as specific tasks to be performed by assigned roles. A prototype implemented to validate the approach is also described.**

Keywords-Best practice frameworks, Business processes, Process-aware Knowledge Repository

I. INTRODUCTION

A best practice framework is a collection of well-established and documented "techniques, methods, processes, activities, incentives or rewards that are more effective at delivering a particular outcome than any other technique, method, process, etc. … best practices can also be defined as the most efficient (least amount of effort) and effective (best results) way of accomplishing a task, based on repeatable procedures that have proven themselves over time for large numbers of people" [1]. Best practices are employed in various domains including business, governance, quality assurance, software development, performance management, risk management, and IT management (e.g. see [2, 3, 4, 5, 6, 7]). Some focus more on technical aspects, others more on business factors.

Examples of best practices frameworks are Six Sigma [8] and ISO 9000 [9] for quality assurance. Balance score card [10] is a framework for performance management using business metrics for decision making. Enterprise Architecture frameworks such as TOGAF [6] and Zachman [7] aim to link business functions to information technology (IT). COBIT [11] is a best practices framework for IT governance. The IT Infrastructure Library (ITIL) [2, 3], eTOM [5] for telecom companies and ISO/IEC 20000 [12] are best practices frameworks for IT Service Management (ITSM).

Many best practice frameworks refer to *processes* as the main abstraction to define scoped work (e.g. ITIL, COBIT and eTOM refer to processes). The term "process" in these frameworks often refers to the scope of work (the "what") rather than how this work is organized and performed. This scope is subject to subsequent specialization and refinement.

A common feature of best practice frameworks is that they summarize a large amount of experience gathered and refined over many years by domain experts, who define and describe organization, structure and context of how work should be organized among people in a particular domain. "Implementing" processes defined in such frameworks involves 1) choosing and understanding a framework appropriate to the domain; 2) refining and specializing the abstractions presented in the framework; 3) mapping those abstractions into the target environment by creating the organizations, roles and processes described in the framework, and 4) assigning responsibilities and tasks to the defined roles. Assigning responsibility means that individuals or organizations are being charged with a specified scope of work. While the scope of work is defined in the framework as well as the relationships among roles, it is often not defined how this work should be organized and carried out by the assigned roles.

To improve organizational efficiency, and repeatability of work in the enterprise, it is desirable that best practice frameworks be followed by knowledge workers and professionals in their daily work. However, interactions between such workers are often facilitated through conversations, collaboration and ad-hoc decision making. It is questionable whether traditional control-flow oriented process specifications are appropriate to implement processes identified in best practice frameworks, especially in domains where high-skilled professionals work in contexts requiring creativity and flexibility. Business processes, in their traditional definition [13, 14] are rigid, precisely-defined and expensive, and hence do not meet the flexibility and dynamicity requirements. Traditional business processes automation and support systems such as SAP systems [25] are not appropriate in these contexts. Indeed, interactions among people can be spontaneous, unforeseen and may not be followed in a strict order making them difficult to support through systems with pre-defined and fixed processes. Lighter-weight and more flexible approaches are required to support the enactment of best practice processes.

It may be argued that collaboration environments that enable informal and non-structured interaction and knowledge sharing among people such as Wiki [15] or Microsoft Sharepoint [16] may offer a viable option for best practice

processes. They provide flexibility, but are entirely unaware of the work context in which they are used and hence can only serve as passive information repositories through which people exchange information.

Motivated by the goal of bridging the gap between the high level abstractions available in best practice processes, deriving actionable tasks and activities that can be automated, and retaining the flexibility of ad-hoc interactions among people while improving repeatability, we propose a system which offers the following capabilities:

- It supports process definitions from best practice frameworks in form of templates which guide the overall organization of work in business. A template is a semi-structured construct, representing a best practice process, defined using a set of concepts and relationships taken from the best practice framework.
- It incorporates the concepts of organizations, individuals and roles (who), scope (what), time (when), activities (how), resources (documents), and permits integration with existing systems to support these concepts.
- It supports the creation of new templates using the built-in concepts and permits dynamic adjustments to existing template definitions.
- The system provides workspaces through which conversations are mediated between people (or organizations) working together in the context of a business practice process. Workspaces are also used to share electronic work material.
- It drives the interactions between people based on the templates, and enables extension of the templates as part of normal interactions as the knowledge workers carry out their tasks.

We describe in this paper how business interactions can be derived and executed from best practice frameworks. Our goal is to demonstrate how actionable tasks can be identified in best practice frameworks, refined and formally described in templates. Templates are annotated with semantic information allowing them to be interpreted by the system, and be used for triggering actionable tasks, that can be assigned to people (or organizations) to be performed by them. The system uses an event-based mechanism to track the definition of tasks as well as their state of execution (progress). This enables flexible execution of activities not necessarily in a fixed order.

The remainder of paper is structured as follows. Section II presents a motivating example from the ITIL Service Design best practices framework ("Supplier Management Process"). We describe the proposed abstractions and approach in section III. Section IV presents the architecture of the proposed system and presents a methodology for using the proposed approach. Section V describes the implementation of our prototype, followed by a discussion of related work in section VI. Finally, we conclude the paper and present future work in Section VII.

II. MOTIVATING EXAMPLE

Let us consider high-skilled professionals working in the IT services business solving customer problems as our target audience with the goal of making their work more efficient. For efficiency, leverage is critical. Well-defined approaches to complex problems are needed, which includes the ability to reuse experiences from the past, bringing to bear knowledge obtained in specific industries as well as implementing best practice frameworks.

We refer to the IT Infrastructure Library (ITIL) [2, 3] as an example of a best practice framework that is used within the context of IT services and their management. ITIL broadly covers aspects of service strategy, service design, transition and operation as well as ongoing service improvement. The documentation encompasses five volumes of text [3].

In this paper, we select one part of ITIL framework (4.7 "Supplier Management", ITIL Vol. 2 Service Design). This part deals with guidance on how external IT services should be provisioned (procured, integrated, maintained). The section takes the view of the service consumer. Examples of such services are basic information services such as networking, email or archiving services, but can also be business services such as payroll or accounting services. Since it is assumed that these services are delivered by external service providers, which are separate legal entities (external companies), business aspects must be considered in addition to the technical aspects. Business aspects identified in ITIL include establishing and maintaining legally valid service provider/customer relationships, establishing and monitoring contracts as well as the delivered services, maintaining a contract database, etc.

The guidance in ITIL's Supplier Management section is described in about 20 pages of text. The following is a fragment from this text describing the scope of the Supplier Management Process (quote):

"The Supplier Management process should include:

- Implementation and enforcement of the supplier policy
- Maintenance of a Supplier and Contract Database (SCD)
- Supplier and contract categorization and risk assessment
- Supplier and contract evaluation and selection
- Development, negotiation and agreement of contracts
- Contract review, renewal and termination
- Management of suppliers and supplier performance
- Agreement and implementation of service and supplier improvement plans
- Maintenance of standard contracts, terms and conditions
- Management of contractual dispute resolution
- Management of sub-contracted suppliers."

Process descriptions at this level cannot be implemented directly, and must be refined and supplemented with more detailed contextual information in order to arrive at a level of specificity that can actually be implemented to structure and perform work among people. For example, the ITIL description includes additional information about activities that comprise the "supplier and contract evaluation and selection" process. In addition, the description does not provide detailed guidance on *how* the activities described should be performed, and is subject to interpretation by domain experts within a given context.

III. A TEMPLATE-BASED APPROACH TO MAKING BEST PRACTICE PROCESSES ACTIONABLE

The topics listed above for the Supplier Management Process describe the scope of supplier management in context of the ITIL framework. Although this scope is referred to as Supplier Management *Process*, it is clear that the listed topics cannot be understood as steps of a flow of activities. Modeling those topics in a process modeling tool will not lead to actionable information. Rather, the topics must be categorized, refined and interpreted within the context in which they are being used.

We propose a template-based approach to represent these processes so that it can be enriched with actionable information. Figure 1 contrasts our template-based approach with best practices frameworks and process frameworks based on their definition, support for definition refinement, and their purpose.

Best Practices Framework e.g. ITIL	**Template-based Framework** (our approach)	**Process Framework** e.g. BPEL, BPEL4People (WS-HumanTask), Workflows
Definition: a comprehensive set of informal documentation.	**Definition**: through finite sets of concepts which can be interpreted by a system by applying *interpretation patterns.*	**Definition**: through finite set of syntactical elements which can be executed.
Refinement: interpretation by domain experts, result is refined informal documentation.	**Refinement**: guided by the system by invoking refinement patterns on more generic templates leading to more specific templates through a dialog with a person.	**Refinement**: is not supported. A level of executable specificity is assumed.
Purpose: to structure work of groups of people. Processes are human-activated, performed and tracked leaving room for flexibility and creativity. Lack of formality prevents support through systems.	**Purpose**: to provide a *work environment for people* who can trigger activities, perform and track their execution and provide the electronic work material in workspaces.	**Purpose**: to define exact, planned flows of activities among systems and / or people to enable systems drive processes. Strict formality often prevents flexibility needed by people.

Fig. 1. Template-based Framework versus Best Practices and Process Frameworks

Best practice frameworks (left column) define a broad domain using high level informal descriptions (text) which are manually refined by domain experts into more targeted, but still remain informal descriptions to be followed by people. Typical outputs of refinement are documents that describe processes and provide guidance on how processes should be followed. There have been some efforts to use business process modeling tools for modeling best practice processes described at a high level [40]. However, creating such process models is difficult due to the lack of details and formality of best practices and preciseness needed by the tools. Indeed, definition, refinement, and actuation of activities identified by best practice frameworks are usually performed by people and depend on the context of the project.

The process frameworks (right column) include languages such as BPEL[1], which focus on machine-executable definitions of processes and their steps. Consequently, they define a finite set of syntactical elements which can be executed by a process execution engine. Workflow management systems such as SAP [25] also assume a pre-defined, fixed set of processes. It is assumed that process definitions have gone through refinement before execution. The process framework itself thus does not provide means for refinement, and its purpose is to enable systems to execute processes that have been pre-defined at a level of detail necessary for automation.

Efforts to map best practices frameworks into process frameworks often fail because of lack of alignment between the definition, the need for refinement, and precision needed for execution through a system. Definitions from best practices frameworks are "too broad" and "too high-level" to be implemented using process frameworks. They need to be artificially forced into the precision needed by process frameworks making them rigid, fragile and not readily usable in different contexts.

The proposed template-based approach (center column) aims to combine the benefits of the two other approaches.

Definition. We introduce formality as well as flexibility by using templates as opposed to informal text of best practices or the strict syntactical elements of process frameworks. Templates offer flexibility while providing the structure necessary for interpretation by the system.

Refinement. We acknowledge the need for process refinement as part of work among people. The set of tasks necessary for the same process may change (some are performed, some not) from one project to another. Therefore, templates need to be refined to add a task to a set of pre-existing tasks or remove some. Process templates are also continuously refined as people interact during the project. For example, refinement may occur when a task is performed and its status is updated in the template, or when a task is assigned by a manager to a person. We support refinement of templates by providing patterns which can be applied on templates and interpreted by our system. For example, a pattern may instruct the system to drive a dialog with a person to capture further information. This information is added to the template, thus refining it.

Note that there is a contrast between process refinement and process customization. Process customization is needed to adapt process definitions usually in engagements with large organizations that have specific requirements (e.g., SAP processes are customized for each customer). However, in this context we consider process refinement in the context of SMBs where little specialization is necessary, but processes still need to be adapted to the specific project context.

The two fundamental abstractions used in our approach are *concept templates* and *interpretation patterns*. These are described in more detail next.

[1] *www.oasis-open.org/committees/wsbpel*

Concept	Sub-concepts	Properties	Examples
Thing ("an identifiable entity")			Supplier, Contract.
Context ("Identifiable set of connections between things")	**Relationship**, **Organization**, **Project**	Lifecycle (plan -> existence -> remembrance)	a business, customer relationship, company, project.
Activity ("Identifiable motion of something over time")	**Task** ("A thing to do"), **Process** ("planned & sync., multi-task activity"), **Step** ("atomic unit of a process"), **Event** ("notification that something happened"), **Conversation** ("interaction between actors for purpose of information gathering").	Lifecycle, spontaneous, planned, unplanned, synchronized has trigger, owner, actor, goal, state, Transitions over states.	organize a project, manage a team, prepare a report, a dialog (as a conversation).
State ("Identifiable condition of something at a point in time")	**Goal** ("end state of an activity"), **Lifecycle State** ("stages over: plan -> existence -> remembrance")	Enumeration of conditions.	Lifecycle state, KPI, desired state.
Item ("An identifiable thing to work on or work with")	**Document**, **Schedule**	Lifecycle.	plan, schedule, reminder item.
Actor ("Someone or something that carries activity forward").	**Person** ("a human being in a role"), **System** ("system performing activity") **Service** ("something or someone doing something for someone else"),	Lifecycle, has role(s), activities.	Person, Team, Organization, Process, Salesforce.com.
Role ("function assumed by an actor in a context").		Lifecycle, Assigned to Actor(s).	Owner, editor, creator, manager.

Table 1: Top-level concepts to model processes in best practice frameworks

A. Concept Templates

Our approach to templates is knowledge-oriented. A template consists of a formally defined set of concepts with their relationships bounded by a scope. Concepts are words which are uniquely identified and carry meaning for a person. Concepts can be sub-categorized meaning that abstract concepts can be broken into more concrete concepts.

Our system is built around certain concepts (Table 1) that are often used in human activities and found in process of best practice frameworks. Two examples of top-most concept categories are "thing" and "activity", which can often be associated with nouns and verbs (or phrases with nouns and verbs) in informal text. Other top-level concepts are "context", "actor", "role" and "relationship". Relationships among concepts lead to graphs, which are used within our system as a way of formalizing and representing domain knowledge.

General templates about a domain can be created centrally by domain experts and shared as part of the system. These templates can then be copied and subsequently refined for a particular purpose. As templates are used, we assume that they go through a lifecycle, from more general to more refined. As experience is gained within a domain, these refined templates can again be shared and reused.

The following shows a conceptualization of the fragment about ITIL's Supplier Management Process with a concept graph. Important concepts are highlighted first and then categorized into "things" and "activities".

"*The **Supplier Management process** should include:*

- *Implementation and enforcement of the **supplier policy***
- *Maintenance of a **Supplier and Contract Database** (SCD)*
- *Supplier and **contract categorization** and **risk assessment***
- ***Supplier** and **contract evaluation** and **selection***
- ***Development**, **negotiation** and **agreement** of contracts*
- ***Contract review**, **renewal** and **termination***
- ***Management** of **suppliers** and **supplier performance***
- *Agreement and implementation of service and **supplier improvement plans***
- ***Maintenance** of standard contracts, terms and conditions*
- *Management of **contractual dispute resolution***
- *Management of **sub-contracted suppliers**."*

These concepts can be represented as a knowledge graph. The domain expert decides which concepts are considered relevant for inclusion in the knowledge graph.

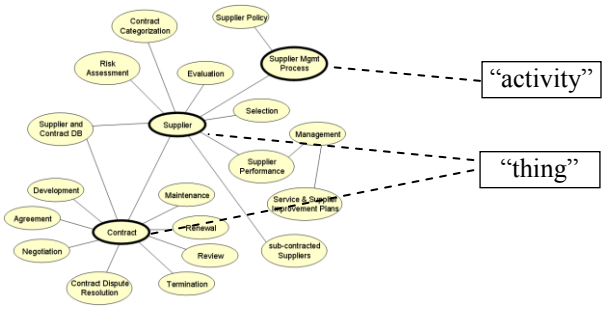

Fig. 2. Top-level view of the knowledge-graph for the Supplier Management Process with categorization into concepts known to the system with two refinement patterns

Figure 2 shows an example of knowledge graph for the supplier management process scenario. In this example, the main concepts of "Contract" and "Supplier" are categorized as "things", while "Supplier Management Process" is an "activity". When combined with the built-in categories shown in Table 1 and interpretation patterns (introduced later), the system can generate a dialog that asks questions about details such as the steps needed for a particular process. Answers to questions in this dialog leads to further refinement of the template.

The top-level concepts the system uses to define templates and represent the relevant information around best practice processes (Table 1) are represented as RDF [17] ontologies. We have chosen to use Notation3 (N3) [18] language for concept representation. N3 is a human readable language for RDF and an alternative to XML-based format of RDF ontologies. An RDF statement in N3 consists of a triplet of "subject, verb and object". For instance, the following shows some of the concepts from Table 1 defined in N3 language:

```
:Concept a rdfs:Resource.
:Context a :Concept.
:Activity a :Concept.
:Actor a :Concept.
:Role a :Concept.
….
```

Based on this representation, it is possible to build the knowledge graph in Figure 2. For simplicity, the verb which identifies the relationships of concepts in Figure 2 is not shown. It should be noted also that the concepts of task, step and conversation have additional properties such as start-date, end-date (deadline), actor, depends-on (a task, step or conversation) and state that are not shown in Table 1.

B. Interpretation Patterns

The templates need to be actionable to enable the system to drive activities between people. We achieve this by augmenting the concepts in templates by *interpretation patterns* and interpreting the concepts in templates by applying those interpretation patterns. While the concepts in the template describe information about a domain, interpretation patterns describe actions that can be performed on this information.

An interpretation pattern is a formally defined structure of particular concepts for which actions can be performed by the system. The actions prescribed in an interpretation pattern may be executed directly by our system, e.g., by driving a dialog with a user to refine a concept, can lead to actions being performed by other systems (e.g., process engines), or be assigned to people (or groups of people).

Examples of general interpretation patterns are "refinement", "creation" and "execution", for concepts such as "thing" or "activity". Top-level concept categories with associated top-level interpretation patterns are built into our system up-front. As top-level concepts are refined into more detail, more concepts with specific interpretation patterns can be added. The notion of interpretation pattern is a powerful notion for two reasons: (i) it allows separation of the actionable aspect of the template from the concepts, and (ii) it enables association of actions to any concept in the template at any level of detail.

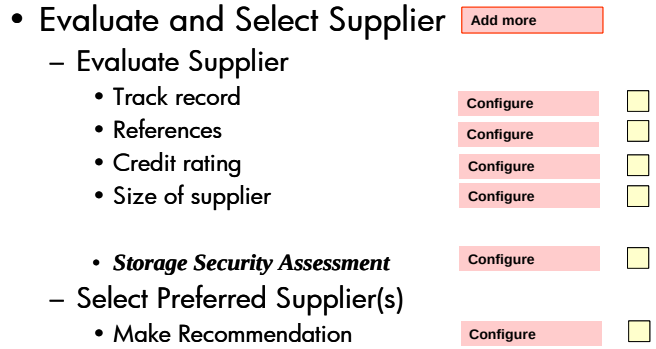

Fig. 3. Illustration of refinement patterns for "Evaluate and Select Supplier" Process.

Refinement Pattern. As discussed, the processes in best practice frameworks are ad-hoc and may change at runtime depending on the context of projects. To support this dynamicity, we provide refinement interpretation patterns on process templates including adding task/step, removing task/step, task/step assignment and task/step status update. Most refinement interpretation patterns in our systems come with a dialog asking questions to capture further information or update information in the template.

As an example, consider the supplier evaluation and selection process from the supplier management scenario. Using a process creation interpretation pattern, the process illustrated in Figure 3 can be created by a domain expert from ITIL documentation (part shown in normal font). This process consists of two tasks "Evaluate Supplier" and "Select Preferred Suppliers". Each task has a number of steps. For example, the Evaluate Supplier involves 4 work items (represented as steps in the template). Assume that a project manager copies this template into her environment for a project to find a data storage outsourcing supplier. In this example, the project manager adds a step called "Storage Security Assessment" to the template (using dialogs in our system), which is not part of ITIL documentation but is needed in the context of this project (the added step is shown in italic-bold font). The system

applies the interpretation pattern of Add Task/Step pattern and therefore refines the template.

Execution Pattern. To support the execution of activities and the interactions among people, we define a set of built-in execution interpretation patterns for activity-related concepts. In particular, we provide a mapping from activity-related concepts to abstractions in a process definition and execution language so that we can use the benefits offered by process engines to enact best practice processes. In other words, an execution pattern uses the mapping to generate a process definition for an activity template, and updates it as the template gets refined.

We have chosen to use jPDL (jBPM Process Definition Language) [20] as the process definition language in our approach. The processes that are defined in jPDL can be deployed and executed using JBoss jBPM[2] [21]. We map a "process" concept to the abstraction of "process", a "task", "conversation" and "step" to "task" (with a "task-node") in jPDL. Note that the process definition is a living entity because as the activity definition gets refined in the template, the corresponding process definition is also updated. Some of the refinement happens during the time that the process is running. We discuss the implementation aspect of this mapping in Section V.

IV. Solution Components and Methodology

Figure 4 shows the building blocks of the proposed system. It consists of three main components: an information repository, a logic layer, and a portal as web access layer.

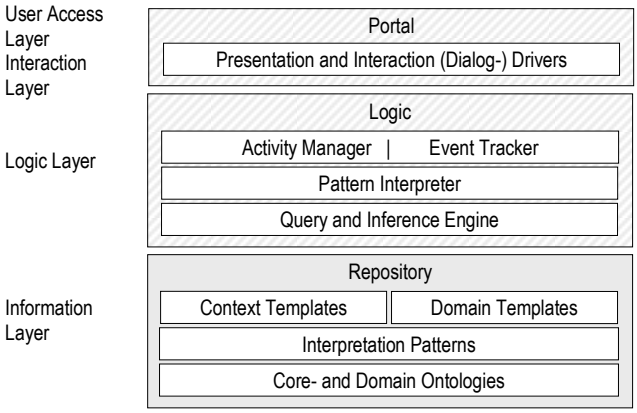

Fig. 4. System Architecture with Information Layer, Logic Layer, and User Access Layer.

A. Components

The information layer stores the core information used by the system. Most fundamental are the core and domain ontologies, which define the concepts that are understood by the logic and the interpretation patterns that refer to them. The interpretation patterns in this layer refer to concepts defined in the ontologies and associate them with concepts for which functionality in the logic layer exists. Templates link concepts from ontologies and interpretation patterns together. They are used to capture domain information. Templates fall into two categories, general domain templates and context templates, which are specific to projects.

The logic layer accesses the information layer for executing queries and generating inferences over the information. Ontologies, interpretation patterns and templates are all represented in RDF using N3 language. This layer provides a generic access layer, which is used by a pattern interpreter to load and interpret patterns. Pattern interpretation is triggered from the Event Tracker and the Activity Manager. Activities are triggered either as result of user interaction via the user interface portal (e.g. a manager assigns a task to a person), or as the result of some condition becoming true (e.g. the due date for a deliverable being reached). Events are associated with interpretation patterns, which describe the reaction to the event. The Activity Manager is an interface between our system and existing process execution engines. It enables transfer of events captured by the Event Tracker regarding the progress of an activity to the corresponding process instance in the process engine, and to return changes in the process execution (e.g., creation of a new task in the workspace of another person) to the user access layer in our system. The activity manager is also responsible for updating the process definition and process instance when an activity template (context template) is refined in our system.

The user access layer is a web-based portal which mediates the user interactions with the system. It performs two major functions. One is the presentation of concepts (information), and the other is to creation and refinement of templates, initiate activities and present information regarding the progress of activities to the user.

B. Methodology for using the template-based approach to making best practice processes actionable

The overall methodology for mapping process descriptions from best practice frameworks into actionable steps consists of four steps.

(1.) The first step is to identify the concepts that are relevant in the best practice framework and relate those concepts in a graph. This work is done by a domain expert.

(2.) The second step is to map the domain concepts to pre-defined categories that are understood in our system. For example, the Supplier and Contract Database mentioned in the domain topic list falls into the category of a "thing", while Supplier and Contract Evaluation and Selection refers to a "process" that needs to be established and performed. This step is also performed by the domain expert.

(3.) The third step is to refine the domain concepts. Concept categories have associated semantics, which means they have properties such as name, definition, etc. The other aspect of a template is that the actions are represented in form of interpretation patterns. For instance, in our example, Supplier and Contract Evaluation and Selection had been categorized as an activity. One of interpretation patterns on activities is the process creation pattern. Applying this pattern on Supplier and Contract Evaluation and Selection leads our system to initiate a dialog with the domain expert asking for more detail about this particular process such as tasks and individual steps, the

[2] *jboss.com/products/jbpm*

required inputs and outputs, relationships with other concepts (steps and conversations), etc. The domain expert answers this dialog to define the activity template as part of domain templates.

Note that the template can be further refined after being created to be used in context of a new domain example to create context templates. The result is a definition of a refined Supplier and Contract Evaluation and Selection process. Answers provided by the domain expert in this context lead to more detailed concepts and supplemental information which is now known to the system.

(4.) Step four refers to the use of the refined process definition, which includes the creation of instances of this process and the application of further interpretation patterns that are defined for the process category, such as the configuration pattern (e.g. assignment) or the execution pattern.

Invoking the assignment pattern means that an instance of the refined Supplier Evaluation and Selection process will be taken through a dialog in which individual roles and responsibilities are assigned to the tasks of this process. In contrast to the definition pattern, the assignment pattern will not be invoked by the ITIL domain expert, rather by an end user who is in charge of a particular Supplier Evaluation and Selection engagement.

Invoking the execution pattern means that actual actions are triggered for the assigned roles. Individuals behind those roles receive actionable items about expected deliverables and timelines as well as the input material they need to perform the task assigned to them.

V. IMPLEMENTATION

We have implemented an initial prototype to demonstrate the proposed approach. We use RDF to represent the information for ontologies, interpretation patterns and templates. We use the Jena toolkit [19] which includes a variety of model stores for the repository as well as libraries for query and inference. The pattern interpreter was implemented in Java. It reads the patterns encoded in RDF (N3 notations) and creates HTML pages for the user interface or invokes operations from the activity manager. We use a Java-based Wiki implementation [22] as the user interface portal. The wiki's language was extended to provide programmable control dialog elements.

Figures 5 and 6 show screens from the ITIL Supplier Management example presented earlier. The scenario is that a person (Bill) is tasked with a new supplier management project. He creates a new workspace for his new project, which triggers the interpretation of a number of creation patterns starting with a general pattern asking in a dialog about the domain. Choosing ITIL triggers a more specific pattern asking for a domain within ITIL. If Supplier Management is chosen, the creation pattern for Supplier Management is activated. Figure 5 shows the end state of the creation patterns with the choices made.

Figure 6 shows the workspace of the new project which is structured based on the Supplier Management process listing its parts in the task column. Bill (the project creator) is at this point the only person involved in this project. The resources column shows work material that has been included by the system, also as result of the creation pattern. Each object on the page has further interpretation patterns associated, which are activated when clicked. People can be added or removed, as well as tasks and resources. It should be noted that this layout of the workspace is rendered based on templates, so it would be different for another project with a different template.

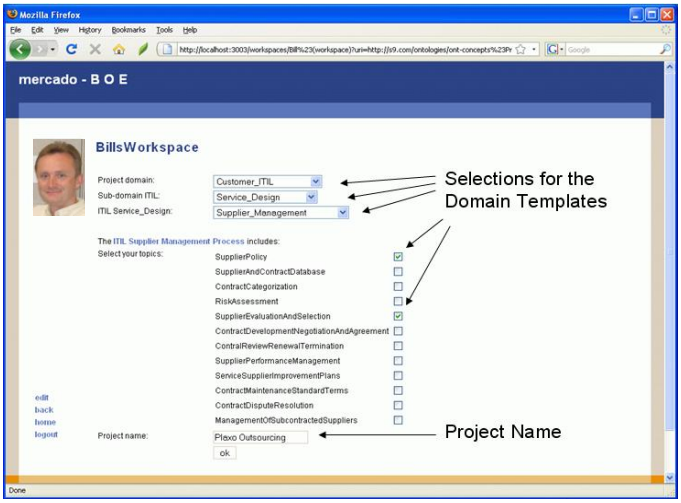

Fig. 5. Making choices during the "Creation Dialog" for ITIL Supplier Management Project.

The dialogs from figures 5 and 6 have been created from templates stored as RDF graphs in the system. For each dialog, a copy of a graph is made and supplemented and extended with the answers returned from a user interaction.

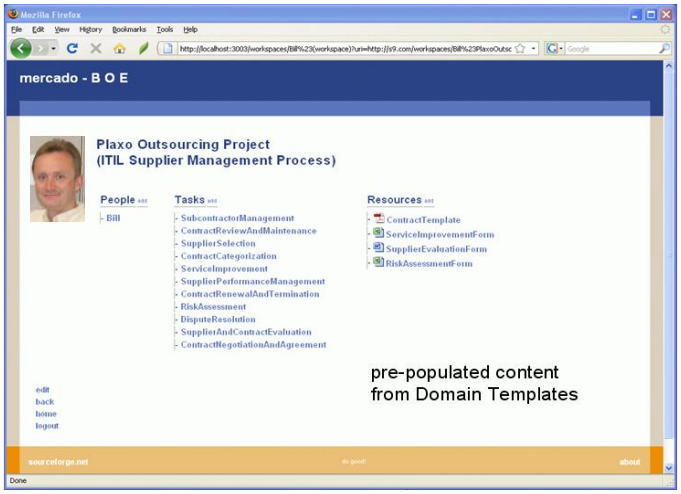

Fig. 6. New workspace is populated with concepts from ITIL Supplier Management.

The following figure shows a fragment in N3 showing the top-level structure of Bill's workspace and the structure for generating the dialog for choosing the top-level project domain generating the form presented in figure 5.

```
@prefix : <http://s9.com/workspaces/Bill#> .
@prefix ont: <http://s9.com/ontologies/ont-concepts#> .
@prefix ows: <http://s9.com/ontologies/ont-workspace#> .
@prefix q: <http://s9.com/ontologies/ont-graphquery#> .
@prefix ia: <http://s9.com/ontologies/ont-interactor#> .
```

```
:BillsWorkspace a ont:Workspace ;
  ont:content (
        ( :Projects   ont:members ( ont:Project ))
        ( :People     ont:members ( ont:Person ont:Team ))
        ( :Actions    ont:members ( ont:Action ))
        ( :Tasks      ont:members ( ont:Process ont:Task ))
        ( :Resources                ont:members ( ont:Resource ))
  ) .

:DForm_CreateProject_0 a ia:DForm ;
    ia:createNew ont:Project ;
    ia:formTable ( ( ( "Project domain:" )
( ( :DFCreateProject_Type_0      ia:WT_DropDownMenu)
( ia:value (              :Customer
                          :Customer_ITIL
                          :Customer_BPO
                          :Customer_ServiceSale
                          :Internal_Service
                          :Internal_ITIL
        ) ) ) )
( ( "" )   ( ( :DFCreateProject_OK ia:WT_Button) ( ia:text "ok" ) ) ) ) ;
  ia:continueWith     :DForm_CreateProject_1 .
```

Fig. 7. Template for driving the user dialog for the project domain.

The activity manager and event tracker components of the logic layer are implemented in Java. As mentioned in Section III.B, the activity manager is implemented on top of JBoss jBPM [21]. When a process template is created (using the creation pattern), the corresponding process definition is deployed in jPBM process engine. When the activity template is refined (e.g., a task is added), the corresponding process definition is updated by the activity manager and a new version of the process is deployed to the process engine. If the refinement happens while the process is running (there is an associated process instance), then the process instance is migrated to use the new process definition in the jPBM process engine. There is no strict ordering between tasks of an activity in best practice processes, unless dependencies are used in templates to make the ordering explicit. Thus all tasks are created as child elements of a task node in jPDL. It allows achieving the goal of not requiring a strict order, while other constraints such as actor, start-date and due-date are enforced for each task.

VI. Related Work

We categorize the related work into four areas: best practice frameworks; business processes; knowledge and document management systems; and collaboration environments.

Best practice frameworks. Processes from best practice frameworks such as ITIL [2, 3] and eTOM [5] has been often described as textual descriptions. There have been efforts to support people in formalizing and following best practice frameworks. For example, some approaches propose using Semantic Wiki [34] and also ontologies [23] to represent ITIL processes. However, these efforts only look at this representation as a knowledge base rather than actionable processes.

Business processes. Business process modeling and management tools such ARIS [24] or SAP [25] allow definition of well-defined, rigidly-structured business processes. However, many processes in the enterprise, especially in the context of best practice processes, involve human interactions that are semi-structured or ad-hoc. In the same line of work, [40] formalizes ITIL processes as precise business process models expressed in process modeling languages such as BPMN. However, such processes inherit the same limitation and they are over-precise and cannot be followed as designed in the context of ad-hoc processes of enacting ITIL processes. While our solution takes advantage of existing business process management, by building on top of them in driving the interactions, it allows best practice processes to be defined and refined in a more "lightweight" approach and provides supports of a collaborative environment rather than executing hard-coded processes.

BPEL4People [26] (as extension to WS-BPEL) addresses the need to capture human interaction in business processes. It is complementary to our framework, and our activity manager can be built on top of a BPEL4People engine to enact processes that include human involvements. However, a BPEL4People engine has to handle complex human interaction patterns, such as manual nomination of a task by a supervisor to an employee, escalation and independent decision-making by 2 humans (4-eye principle). We have not (yet) observed the need for this level of complexity in our work.

Definition of ad-hoc and flexible processes has also gained attentions recently [14, 25]. For instance, Caramba [25] enables definition of ad-hoc processes in the context of virtual team. In this work, the process definition has to be explicitly defined by the team members using graphical process modeling tools. However, in the context of best practices: (i) process are not well-specified to enable formal definitions directly, (ii) process users are knowledge workers that are only familiar with environments such as Wiki and MS Office; they find it difficult to work with formal modeling tools, and (iii) our proposed template-based approach provides a living knowledge-base for best practice processes. It enables making the domain and context knowledge of processes available to knowledge workers. This is not supported by approaches enabling the definition of ad-hoc processes such as Caramba.

Change management for adaptable and dynamic workflows is also studied [41, 42, 43] in the literature. Adaptable workflows address changes that affect the workflow definition (structure, type, etc), while dynamic workflows are concerned with changes to runtime instances of workflows. ADEPTflex [42] enables operators to (manually) change the running instances of a statically defined workflow, while ensuring the correctness. In [43] a high level definition of a workflow is assumed and the concept of *worklet* is introduced to attach concrete activities from a library to realize tasks in the high level workflow. In our context, the changes to the workflow are of both types: workflow definition (before collaboration between users starts) and workflow instances (as work is performed). The changes to the ad-hoc processes and instances are made as the user works with process-level concepts in collaborative environment. In other words, users are not exposed to (formal) process definitions. We do not assume availability of a library of tasks that could be used to realize high level tasks, as most often changes to tasks in process templates are specific to the project context. The change management is offered by activity manager, which is built as a layer on top of jBPM as it allows defining tasks in task nodes without imposing a particular execution order.

Knowledge and document management systems, and collaboration environments. Many existing requirement capture and management tools [28] and business process analysis tools such as ProVison [29] simplify the tasks of gathering, documenting, tracking and managing requirements and process definitions in an enterprise. Typically these tools help document requirements and processes, and in some instances simulate the impact of changes. They are geared towards implementing and executing projects and processes in IT systems, not among people.

A major hurdle to wide-spread adoption of knowledge management tools [32] today is the poor linkage between them and the surrounding human processes: they are only repositories of information. Our work addresses this hurdle by providing the link so that information in the knowledge base is now actionable. That is they can be interpreted and executed. The other key differentiator of our work from knowledge management systems or business process management tools is that our system does not have to be used "after the fact" (after processes are thoroughly defined). Often knowledge bases or process design tools are used to craft processes, but people rarely go back to update information in them. This reduces the ability of enterprises to "reuse" information and processes as a result of ad-hoc changes needed by people. Our solution differs from document management systems [30] such as Documentum [31] in that it is targeted at not just managing documents or document workflows, but business interactions between participants.

Collaboration environments. The proposed Wiki-based platform differs from Semantic Wikis (e.g. Semantic Media Wiki [33], IkeWiki [34], OntoWiki [35] or KaukoluWiki [36]) because we incorporate domain-independent semantics into the Wiki to organize people's activities by defining concepts in the RDF model, which relate to functionality implemented in the Wiki back end. We also differ in the ability to upload RDF models templates with domain knowledge into the Wiki.

The social computing concepts, approaches and tooling for knowledge sharing and collaboration incorporated in the enterprise systems, collectively known as Enterprise 2.0 [37], are complementary to our work. Our platform benefits from leveraging abstractions and methods from this area for knowledge capture and sharing between business users.

VII. Conclusion and Future Work

In this paper, we have presented a novel template-based framework to enable making processes from best practice frameworks, which are dynamic, ad-hoc and not-structured, actionable. We have implemented a prototype system that demonstrates the viability of the proposed approach.

As future work one of our goals is to assist generating concept graphs from best practice frameworks by populating the RDF database with concepts and relationships extracted the documents for these frameworks. The user should be able to search the database, and include some of the concepts and relationships found as a result into the concept graphs.

This involves knowledge acquisition from text based on natural language processing and machine learning techniques. A survey of such techniques for learning ontologies from text is provided in [38]. The taxonomic relationships leading to the hierarchy of classes in the ontology have received a lot of attention. These are the transitive 'is a' relationships. The non-taxonomic relationships express the properties of classes and instances and are more challenging. The common approach in the past has been finding anonymous relationships from the text that are frequent or important and then labeling them in a subsequent phase. Generating the relationship label can be done with a chunk parser that maps sentences to structures containing noun phrase (NP) and verb (V) with optional preposition (P). Common patterns can be NP-V-NP and NP-V-P-NP. The verb with optional preposition then becomes the relationship label. When domain knowledge is available, it can guide the generation of an RDF graph. As an example, in [39], the Unified Medical Language System (UMLS) is used to determine the relationships that should be extracted from abstracts of biomedical publications available in PubMed.

We also plan to observe and learn from people's interactions in order to enrich domain templates by learning from context templates and also for the purpose of provenance (identifying who did what and when). We are in process of beginning to test our prototype in context of HP's outsourcing services business. We are specifically considering smaller, standardized engagement processes which are performed repeatedly.

References

[1] Wikipedia, "Best Practice," http://en.wikipedia.org/wiki/Best_practice, visited May 2009.

[2] L. Hendriks and M. Carr, "ITIL: Best Practice in IT Service Management," in: Van Bon, J. (Hrsg.): The Guide to IT Service Management, Band 1, London u. a. 2002, pp 131-150.

[3] Information Technology Infrastructure Library (ITIL), Version 3, 5 Volumes, http://www.itil-officialsite.com.

[4] OMG. "Business Motivation Model (BMM)," www.omg.org/technology/documents/br_pm_spec_catalog.htm.

[5] TeleManagement Forum, "Enhanced Telecom Operations Map (eTOM) – The Business Process Framework," http://www.tmforum.org/BestPracticesStandards/BusinessProcessFramework/1647/Home.html, visited May 2009.

[6] The Open Group, "The Open Group Architecture Framework (TOGAF)," V 9, http://www.opengroup.org/togaf/, visited May 2009.

[7] ZIFA, "The Zachman Framework for Enterprise Architecture," www.zifa.com, visited May 2009.

[8] Thomas Pyzdek, "The Six Sigma Handbook: A Complete Guide for Green Belts, Black Belts, and Managers at All Levels," McGraw-Hill, 2001.

[9] RW Peach, "The ISO 9000 handbook," McGraw-Hill, 2003.

[10] Maris Martinsons, Robert Davison, Dennis Tse, "The balanced scorecard: a foundation for the strategic management of information systems," Decision Support Systems, Volume 25, Issue 1, February 1999, pp 71-88.

[11] ISACA, "The Control Objectives for Information and related Technology (COBIT)," www.isaca.org/cobit/, visited May 2009.

[12] Jenny Dugmore, "Achieving ISO/IEC 20000 - The Differences Between BS 15000 and ISO/IEC 20000," BSI Standards. ISBN 0580473481, 2006.

[13] Mathias Weske, "Business Process Management : Concepts, Languages, Architectures," Springer-Verlag, November 2007.

[14] Marlon Dumas, Wil M. van der Aalst, Arthur H. ter Hofstede, "Process Aware Information Systems: Bridging People and Software Through Process Technology," WileyBlackwell, 2005.

[15] Wikipedia, Wiki, http://en.wikipedia.org/wiki/Wiki, visisted May 2009.

[16] Microsoft, "Microsoft SharePoint," www.microsoft.com/Sharepoint/default.mspx, visited May 2009.

[17] W3C: Resource Description Framework (RFD), http://www.w3.org/RDF.

[18] W3C, "Notation 3 (N3): A Readable RDF Syntax," http://www.w3.org/DesignIssues/Notation3.html.

[19] HP, "Jena–A Semantic Web Framework for Java," http://jena.sourceforge.net.

[20] JBoss, "jBPM Process Definition Language (jPDL)," docs.jboss.com/jbpm/v3/userguide/jpdl.html, visited May 2009.

[21] JBoss, "jBPM Workflow Engine," www.jboss.com/pdf/jb_jbpm_04_07.pdf, visited May 2009.

[22] Bliki in a JAR, http://www.matheclipse.org/en/Bliki_in_a_JAR.

[23] *Zhenning Shangguan, Zhipeng Gao, Kai Zhu,* "Ontology-Based Process Modeling Using eTOM and ITIL," In CONFENIS (2) pp. 1001-1010, 2007.

[24] IDS Scheer, Aris, http://www.ids-scheer.com.

[25] SAP, "SAP NetWeaver Capabilities - Business Process Management," https://www.sdn.sap.com/irj/sdn/nw-bpm, Visited May 2009.

[26] WS-BPEL Extensions for People—BPEL4People: A Joint White Paper by IBM and SAP, July 2005 Accessed on May 14, 2009 from: http://www.sdn.sap.com/irj/servlet/prt/portal/prtroot/docs/library/uuid/cfab6fdd-0501-0010-bc82-f5c2414080ed

[27] S. Dustdar, "Caramba — A Process-Aware Collaboration System Supporting Ad hoc and Collaborative Processes in Virtual Teams," Distrib. Parallel Databases 15, 1 (Jan. 2004), pp. 45-66.

[28] Volere. "List of Requirement Managemnt Tools," http://www.volere.co.uk/tools.htm.

[29] Metastorm, ProVision, http://www.metastorm.com

[30] HCi Journal, "List of Document Management Software," http://www.hci.com.au/hcisite3/journal/List of document management software.htm

[31] EMC. "EMC Documentum," http://www.emc.com/products/category/subcategory/collaboration-and-document-management.htm

[32] M. Philippe, "Knowledge Representation and Translation in RDF+OWL, N3, KIF, UML and the WebKB-2 languages," http://www.webkb.org/doc/model/comparisons.html,

[33] University of Karlsruhe, "Semantic Media Wiki," http://semantic-mediawiki.org

[34] Ike Wiki, http://ikewiki.salzburgresearch.at.

[35] OntoWiki, http://ontowiki.net.

[36] KaukoluWiki, http://kaukoluwiki.opendfki.de.

[37] T. Davenport, "Enterprise 2.0: The New, New Knowledge Management?" Harvard Business Online, Feb. 19, 2008.

[38] Philipp Cimiano, Johanna Völker and Rudi Studer, "Ontologies on Demand? A Description of the State-of-the-Art,Applications,Challenges and Trends for Ontology Learning from Text," Available at www.aifb.uni-karlsruhe.de/WBS/pci/Publications/iwp06.pdf

[39] Cartic Ramakrishnan, Krys J. Kochut, and Amit P. Sheth, "A Framework for Schema-Driven Relationship Discovery from Unstructured Text". 5th International Semantic Web Conference (ISWC 2006).

[40] Orbus Software, The iServer ITIL Solution, http://www.orbussoftware.com/business-process-analysis/products/itil-solution/

[41] Mülle, J. A., Böhm, K., Röper, N., and Sünder, T. 2006. Building conference proceedings requires adaptable workflow and content management. In *Proceedings of the 32nd international Conference on Very Large Data Bases* (VLDB), 1129-1139, 2006.

[42] Reichert, M.; Dadam, P.: ADEPT flex Supporting Dynamic Changes of Workflows Without Loosing Control. Journal of Intelligent Information Systems, 10 (1998) 2, pp. 93-129.

[43] M. Adams, A.H.M. ter Hofstede, D. Edmond, and W.M.P. van der Aalst. Implementing dynamic flexibility in workflows using worklets. In *BPMCenter Report*, volume BPM-06-06. BPMCenter. org, 2006.