Electron Commer Res (2009) 9: 269–316
DOI 10.1007/s10660-009-9039-z

Business process management with the user requirements notation

Alireza Pourshahid · Daniel Amyot · Liam Peyton · Sepideh Ghanavati · Pengfei Chen · Michael Weiss · Alan J. Forster

Published online: 6 August 2009

Based on "Toward an Integrated User Requirements Notation Framework and Tool for Business Process Management", by A. Pourshahid, D. Amyot, L. Peyton, S. Ghanavati, P. Chen, M. Weiss, and A.J. Forster, which appeared in the *2008 International MCETECH Conference on e-Technologies*, pp. 3–15. ©2008 IEEE.

A. Pourshahid (✉) · D. Amyot · L. Peyton · S. Ghanavati
School of Information Technology and Engineering (SITE), University of Ottawa, 800 King Edward Avenue, Ottawa, ON, Canada K1N 6N5
e-mail: apour024@uottawa.ca

D. Amyot
e-mail: damyot@site.uottawa.ca

L. Peyton
e-mail: lpeyton@site.uottawa.ca

S. Ghanavati
e-mail: sghanava@site.uottawa.ca

Present address:
A. Pourshahid
IBM, 3755 Riverside Drive, P.O. Box 9707, Station T, Ottawa, ON, Canada K1G 4K9
e-mail: alireza.pourshahid@ca.ibm.com

P. Chen · M. Weiss
Department of Systems and Computer Engineering, Carleton University, 1125 Colonel By Drive, Ottawa, ON, Canada K1S 5B6

P. Chen
e-mail: pchen@connect.carleton.ca

M. Weiss
e-mail: weiss@scs.carleton.ca

Present address:
P. Chen
Research In Motion, 3026 Solandt Drive, Ottawa, ON, Canada K2K 3K2

Abstract A number of recent initiatives in both academia and industry have sought to achieve improvements in e-businesses through the utilization of Business Process Management (BPM) methodologies and tools. However there are still some inadequacies that need to be addressed when it comes to achieving alignment between business goals and business processes. The User Requirements Notation (URN), recently standardized by ITU-T, has some unique features and capabilities beyond what is available in other notations that can help address alignment issues. In this paper, a URN-based framework and its supporting toolset are introduced which provide business process monitoring and performance management capabilities integrated across the BPM lifecycle. The framework extends the URN notation with Key Performance Indicators (KPIs) and other concepts to measure and align processes and goals. An example process for controlling access to a healthcare data warehouse is used to illustrate and evaluate the framework. Early results indicate the feasibility of the approach.

Keywords Business process management · Business process model · Goal-oriented business process monitoring · Performance management · User requirement notation

1 Introduction

Business processes and their management have always represented challenges for organizations. These difficulties are now amplified by processes that are often cross-functional in the organization or that are crossing the organization's boundaries. Gathering the information related to processes from different sources, monitoring these processes, and aligning them with corporate strategies and high-level goals is now a major issue for decision makers in enterprises and in the public sector.

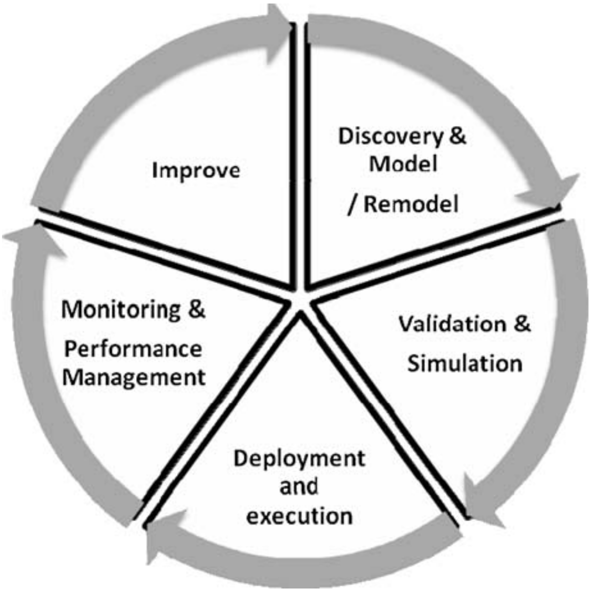
A Business Process Management (BPM) lifecycle often consists of several iterative steps that aim to improve the quality of business processes in an incremental way [9, 28, 58]. Figure 1 gives an overview of a typical model-driven lifecycle, starting with the discovery and modeling of business processes. Using superficial or outdated process models at this point in the lifecycle can prevent the next steps of the BPM project from providing optimal results and value [32].

An integrated and tool-supported methodology is essential for BPM projects to help users who are *modeling* business processes and *validating* them. In addition, allowing the integration of business goals and performance models, in a traceable way to and from business models, greatly contributes to ensuring goal satisfaction and process adaptation [52, 55, 61]. The integration of goals, processes, and performance in a BPM framework enables many useful capabilities, some of which are explored in this paper. As an example, the traceability between business process models and

A.J. Forster
Department of Medicine, University of Ottawa, Ottawa, Canada
e-mail: aforster@ohri.ca

A.J. Forster
Ottawa Health Research Institute, Clinical Epidemiology Program, Ottawa Hospital, Civic Campus,
Administrative Services Bldg, Room 1-019 1053 Carling Avenue, Ottawa, ON, Canada K1Y 4E9

Fig. 1 BPM lifecycle


Improve
Discovery & Model / Remodel
Validation & Simulation
Deployment and execution
Monitoring & Performance Management

business goal models plays a significant role in the successful and practical definition of process-oriented *Key Performance Indicators* (KPI). KPIs are a common way of evaluating different aspects of a business by qualitative measurement [31].

Often, the focus of the *deployment* phase of a BPM project is to automate the processes modeled [28]. However, we believe this phase also offers opportunities for improving the subsequent phases, if it is integrated with information systems that produce data useful for the monitoring and performance management phase. Deploying a data warehouse (DW) that collects performance measurement data related to business processes is one approach. The DW can be used as a data source for a Business Intelligence (BI) tool integrated to the BPM environment. This enables one to investigate business processes in their associated context along different dimensions [52].

The *improvement* phase can use performance measurement results from the *monitoring* phase in combination with a repository of process redesign patterns like those defined in [53]. In other words, redesign patterns to improve the target process can be suggested based on a problem observed during the monitoring phase in order to better align the process with goals defined in the modeling phase. If a process performs well, then its goals can also be revised to become more ambitious. These are some of the main reasons to bring in a performance measurement framework [39].

In this paper, we propose many elements of an integrated BPM framework. Our modeling approach is based on the User Requirements Notation (URN), which combines complementary goal and scenario views. The notation is extended to support KPI and performance modeling, process portfolio monitoring, and scenario-based performance and impact analysis. Our URN-based approach also provides conformance and compliance capabilities [15, 55], which support traceability from goals and scenarios to requirements and policies from the organization or external legislation. Such traceability enables impact analysis as goals, processes, and external requirements evolve.

In addition, we elaborate on the required extensions to the current URN metamodel as well as development and integration efforts done to prototype such enhancements (using tools like the jUCMNav URN editor [54], Cognos' Business Intelligence tools [8], and Telelogic's requirements management system DOORS [22]). Our framework will be discussed and illustrated with an example process for controlling access to a healthcare data warehouse.

This paper is organized as follows. Section 2 provides background information on business process management and modeling, the User Requirements Notation, and data warehousing. We introduce our framework and methodology for BPM in Sect. 3. Tool support for performance monitoring is detailed in Sect. 4, followed by the application of the approach to a healthcare data warehouse access process in Sect. 5. A discussion and conclusions follow in Sects. 6 and 7 respectively.

2 Background

2.1 Business process management (BPM)

A *business process* is a "coordinated chain of activities intended to produce a business result" [4] or a "repeating cycle that reaches a business goal" [9]. People from different units and organization are usually involved to complete an end-to-end process.

Processes can be simple and restricted to a functional unit of an organization or complex and cutting across several business partners. Today's customer-focused business environment requires much business-to-business cooperation to complete a process and often massive integration between different information systems [42]. However, legacy software applications are usually built based on different functional units of businesses, hindering integration [4].

Business Process Management (BPM) is the management of diverse and cross-organizational processes using methods and tools to support the design, execution, management, and analysis of business processes. *Business Process Management Systems* (BPMS) are integrated tools that enable businesses to perform the required steps in a BPM project.

A BPMS is one of the most recommended investments for process improvement [56]. It adds value to the business, enables the reuse of IT investments and addresses the aforementioned integration issues. In addition, a BPMS helps businesses automate and manage business rules and processes. As a key component of BPMS, *Business Activity Monitoring* (BAM) is "the real-time reporting, analysis and alerting of significant business events, accomplished by gathering data, key performance indicators and business events from multiple applications" [12]. According to Kronz [31], the principal outcomes of continuous monitoring, controlling and analysis of processes are improved decision making and process optimization.

A new term that has been recently introduced in the industry is *Business Process Intelligence* (BPI). BPI integrates BPMS and *Business Intelligence* (BI) systems, the latter being used to describe and leverage the organizations' internal and external information assets for making better business decisions [29]. BPI usually includes a DW and a BI tool and is used by both business and information systems users [20].

BPI tools reduce the traditional gap between process execution and performance monitoring [40] and enable continuous process improvement. "BI 2.0" is the name given to this new generation of BI tools. In [46], BI 2.0 and its ability to track ongoing business behaviour are elaborated. There has also been movement towards the integration of BPMS and Corporate Performance Management (CPM) to align corporate goals with business processes [17]. However, neither BPI, nor integration with CPM provides sufficient support in dealing with the business goal models and business process models in an integrated manner.

2.2 Business process modeling

Graphically documenting and displaying business processes is called *Business Process Modeling*. This is a structural method that helps stakeholders analyze processes and find possible points of improvements. While modeling a process, one usually specifies the defined activities performed by different parties involved in the process [61]. According to Mili et al. [42], a business process and its surrounding area consist of activities, events, resources, roles and actors, functions, organization and hierarchy. In addition, as indicated and modeled by List and Korherr [35], a business process context includes process goals, enterprise goals, and measures, as well as process deliverables. Therefore, business process modeling notations should be able to model these aspects of a process. A business process modeling language should also be able to answer the famous W5 questions—Why, What, Who, Where and When [60, 61]. While answering the last four questions helps with defining the process itself, answering the first question—why—helps with specifying goals and requirements behind a process. Although goal modeling and traceability between the initial requirements and the implemented process is an important feature for process management, there exists few process modeling notations that support goal modeling [36].

To support alignment of business processes with business process goals, a process modeling notation should support process modeling, goal modeling, traceability between goal models and processes, and goal model evaluation mechanisms. Otherwise, one would not be able to demonstrate the impact of processes on organizational goals. Also, supporting roles (or actors/organizations), activities (or functions) and events will greatly enhance the capabilities of business process modeling, and enrich the meaning of links between business processes and goals. URN is a notation that supports these features and additional ones introduced in this paper. Table 1 is a summary of popular notations that support process modeling and their features that were considered for our work. More details are available in Pourshahid's thesis [51].

One of the most widely used process modeling notations is the Business Process Modeling Notation (BPMN) [47], developed by the OMG Business Modeling & Integration Domain Task Force. Its main objective is to provide an easy-to-understand language for all business users with different roles and levels of expertise. Based on [49], so far 53 implementations of BPMN exist. According to White [62], the four basic categories of elements supported by BPMN are Flow objects, Connecting objects, Swim lanes, and Artifacts. These basic elements provide support for modeling sequence flow, roles, activities, events, and process hierarchies. Despite its good support for process modeling and user-friendly nature, BPMN does not support any kinds

Table 1 Comparison of notations and the features they support for process modeling

	BPMN	UML	EPC	YAWL	IDEF3	*i**	NFR	EEML	URN
Sequence Flow	✓	✓	✓	✓	✓	+/−	+/−	✓	✓
Roles	✓	✓	✓	✓	✗	✓	✗	✓	✓
Activities	✓	✓	✓	✓	✓	✗	✗	✓	✓
Events	✓	✓	✓	✓	✗	✗	✗	✓	✓
Process Hierarchies	✓	✓	✓	✓	✓	✗	✗	✓	✓
Goal modeling	✗	✗	✗	✗	✗	✓	✓	+/−	✓
Goal model Evaluation	✗	✗	✗	✗	✗	✗	✓	✗	✓
Goal/Process Traceability	✗	✗	✗	✗	✗	✗	✗	✗	✓

of goal modeling or traceability between goal models and process models. However, OMG's Business Motivation Model (BMM) [48], a companion standard from the same task force, defines the notion of goal and related concepts. BMM focuses on business plans and defines how courses of actions (means) can be connected to goals (ends), with some assessments that can be used to document decisions. A business process then realizes a course of action. Visions, assets, and directives, together with their relationships, are also defined. Nevertheless, this standard currently does not offer any notation, semantics, or analysis techniques for goals and their relationship to BPMN processes.

The Unified Modeling Language (UML) is another widely used modeling language. The second version of UML includes 13 notations. According to [42], UML does not explicitly support process modeling. Practitioners, who are familiar with UML, use its activity diagrams (AD) for this purpose. This modeling notation, however, is not as user-friendly as BPMN and has been designed mostly with software engineers as its target audience. Activity diagrams support the modeling of sequence flow, roles, activities, events, and process hierarchies. Again, AD does not provide any capability for goal modeling and traceability between goal models and process models [36].

Event Driven Process Chain (EPC) has been developed by Sheer, Keller and Nuttgens within the framework of the Architecture of Integrated Information System (ARIS) [14] with the same goal as BPMN—to make the process modeling simple for business users [36]. It essentially suffers from the same benefits and limitations.

Yet Another Workflow Language (YAWL) is another business process modeling language, which addresses some of the inadequacies of other existing languages for modeling workflow patterns [57]. YAWL supports most of the requirements of a process modeling language through its extensive support for workflow patterns, but support for goals is absent.

Integrated Definition Method 3 (IDEF3) is another process modeling language developed with the intent of being easy to use [41]. In addition, this language targets both individuals and teams. Using process descriptions and the ability to generate multiple views of a process, IDEF3 gives users more flexibility in process documentation and lets them deal with uncertainties during the modeling process. However, it falls short of supporting many important features for business process modeling.

In contrast with the above languages that offer no support for goal modeling, $i*$ and the Non-Functional Requirements framework (NFR) are two modeling languages that support goal modeling but have very limited support for process modeling. $i*$ has been introduced to address the organizational context and rationales for requirements and to answer the why aspects of the requirements [64]. $i*$ is capable of modeling goals, tasks, and resources and demonstrates their dependencies and mutual contributions using various types of links. Furthermore, it allows users to model roles as organizational actors. NFR, on the other hand, deals with *softgoals* or the goals that are hard to express (e.g. flexibility or security) [7]. NFR allows users to model softgoals in a graph structure and specifies their levels of mutual contribution. In addition, NFR allows users to specify if root goals are satisfied by the selected goals on the leaf by using an evaluation mechanism.

The Extended Enterprise Modeling Language (EEML) was developed to provide comprehensive process modeling across a number of layers. This modeling language consists of four modeling domains—process modeling, resource modeling, goal modeling, and data modeling [30]. Although it supports goal modeling, it is not as comprehensive as URN (Sect. 2.3) in this regard. For example, it has only one type of goal unlike URN, which supports hard goals and soft goals. Furthermore, the relationship provided between goal model components is only a logical decomposition (and, or, xor) and does not provide contribution and dependency relationships. In addition, it does not have a goal propagation mechanism enabling analysis.

To conclude, the process modeling languages investigated so far each have their own strengths and weaknesses. Most process modeling languages do not provide suitable capabilities for goal modeling. On the other hand, goal modeling languages have limited support for process modeling. The only modeling language that supports both is EEML, whose goal modeling capabilities are weaker than common goal modeling notations. Furthermore, URN has the advantage of being a standard of the International Telecommunication Union (ITU-T) [25].

Based on our investigation none of the existing process and goal modeling languages provides the same capability as the User Requirements Notation (URN). Although URN needs some improvements to support more workflow pattern [43], its overall support for process and goal modeling, and its meta-model extensibility make it a suitable choice for our application. In the next section, we introduce URN in more detail. A more thorough comparison of URN and other workflow languages is available in [43, 44].

2.3 User requirements notation

The User Requirements Notation (URN) was designed for modeling and analyzing requirements in the form of goals and scenarios prior to design [25, 61]. It can be used to model most kinds of reactive and distributed systems, as well as business processes.

The URN combines two complementary notations: the Goal-oriented Requirement Language (GRL) and Use Case Maps (UCM) which are used for modeling goals and processes respectively. Figure 2 and Fig. 3 show brief summaries of these two notations.

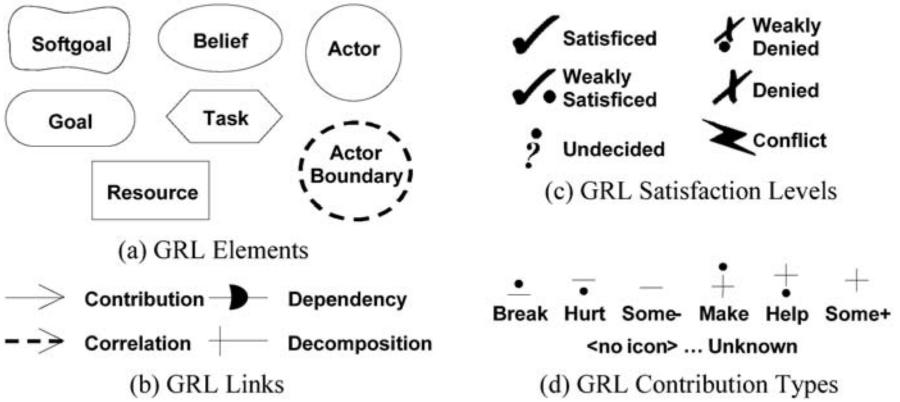


Fig. 2 Subset of GRL notation

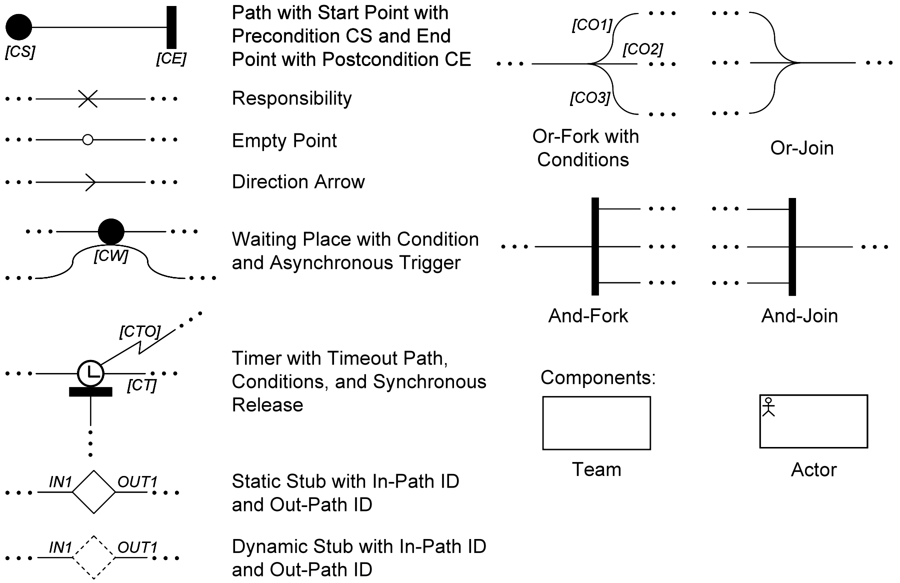

[CS]
[CE]
Path with Start Point with Precondition CS and End Point with Postcondition CE
Responsibility
Empty Point
Direction Arrow
[CW]
Waiting Place with Condition and Asynchronous Trigger
[CTO]
[CT]
Timer with Timeout Path, Conditions, and Synchronous Release
IN1
OUT1
Static Stub with In-Path ID and Out-Path ID
IN1
OUT1
Dynamic Stub with In-Path ID and Out-Path ID
[CO1]
[CO2]
[CO3]
Or-Fork with Conditions
Or-Join
And-Fork
And-Join
Components:
Team
Actor

Fig. 3 Subset of UCM notation

More complex business process modeling languages exist, but a combined view of goals and processes and traceability features between them are unique capabilities of URN. Moreover, URN provides the ability to align business goals and business processes by adopting design experts' business knowledge and experience [37].

GRL supports an evaluation mechanism that lets users define sets of initial satisfaction values (Fig. 2c) on chosen intentional elements (Fig. 2a) in a GRL model (called *strategies*). Those values are propagated to the other intentional elements in the model via their contribution, correlation, decomposition, and dependency links (Fig. 2b), up to the highest level goals [54, 55]. Contribution and correlation links can be positive or negative, with various weights (Fig. 2d). This capability can be used for evaluating the effect of different tasks and processes on goal models, enabling global evaluations of alternatives and trade-off analysis [52]. Finding realistic contribution weights and initial satisfaction values to define a GRL strategy can however be difficult, but our framework proposes novel solutions to this issue.

The UCM process view specifies the responsibilities to be performed (the *what* aspects) by *whom*, *when*, and *where*. The GRL goal view provides a rationale (*why*) for the business process elements, together with an explanation of why alternative solutions were chosen or not. More details on URN are provided in [1, 25]. A detailed analysis of the capabilities of URN in comparison with other well-known business process modeling languages is given in [43].

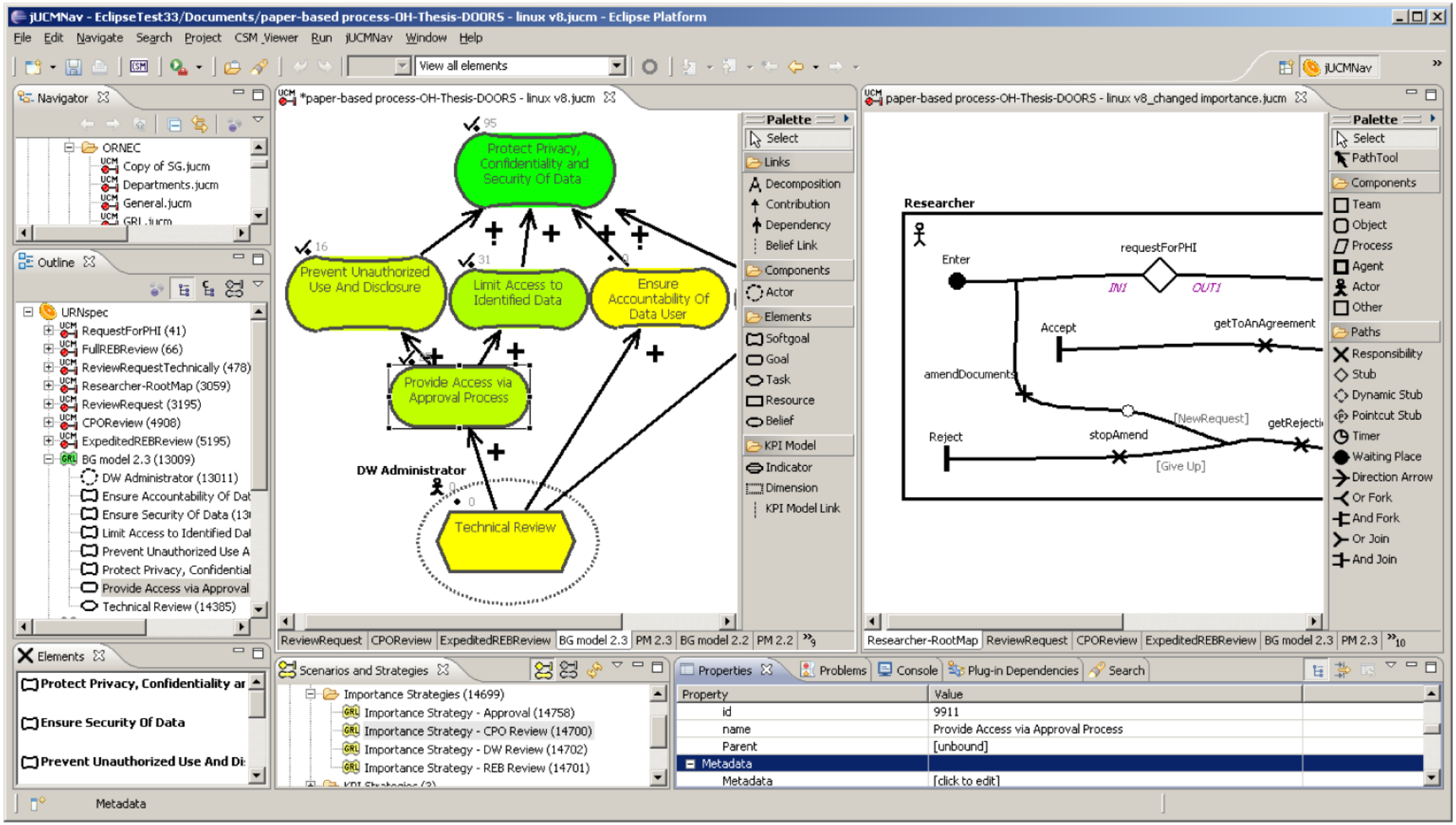
Work on the framework described in this paper was initiated in [60, 61], which introduces business process modeling using URN. The framework's core tool is *jUCM-Nav*, shown in Fig. 4, an open URN modeling, analysis and transformation tool based on the Eclipse platform [54].

The data exchange layer and integration with external tools was formalized in [27]. Integration of goals/scenarios with a Requirement Management System (Telelogic DOORS) [22] was used as a basis for process validation and compliance verification against requirements, goals, and policies, as elaborated in [15]. Business process monitoring and performance management was introduced in [52]. This paper builds on top of this work to help answer a new "*how well*" question with BPM.

2.4 Data warehouse

A *data warehouse* is described as a subject-oriented, integrated, time varying, non-volatile collection of data that is used primarily in organizational decision making [23]. A data warehouse plays an important role in process measurement, which typically provides statistical process data that combines with business data to determine the performance of business processes [32]. In our work, the data warehouse is mainly used as a query-able source of business information, to either provide data directly to monitoring services or feed data to a BI engine for further refinement. The dimensional modeling, which is mainly a data mart design, is one of the main concerns in preparing business information providers in this approach.

As a logical subset (view) or a physical subset (extract) of the complete data warehouse [23], a *data mart* is built to meet a specific, predefined need for a certain grouping and configuration of selected data. A star schema or a dimensional model is a popular way to design data marts, but a snowflake schema is also used for some particular considerations. As a special case of the snowflake schema, the star schema allows a single level of dimension tables, while the snowflake schema may contain a set of constituent dimension tables which can be further broken up into sub-dimension tables and thus is better normalized but more complex to query [34]. For dimension tables, it is common to consolidate redundant data and be in second normal form, while fact tables are usually in third normal form, i.e. without redundant data.


jUCMNav - EclipseTest33/Documents/paper-based process-OH-Thesis-DOORS - linux v8.jucm - Eclipse Platform
File Edit Navigate Search Project CSM_Viewer Run jUCMNav Window Help
View all elements
jUCMNav
Navigator
ORNEC
Copy of SG.jucm
Departments.jucm
General.jucm
GRL.jucm
Outline
URNspec
RequestForPHI (41)
FullREBReview (66)
ReviewRequestTechnically (478)
Researcher-RootMap (3059)
ReviewRequest (3195)
CPOReview (4908)
ExpeditedREBReview (5195)
BG model 2.3 (13009)
DW Administrator (13011)
Ensure Accountability Of Dat
Ensure Security Of Data (13
Limit Access to Identified Dat
Prevent Unauthorized Use A
Protect Privacy, Confidential
Provide Access via Approval
Technical Review (14385)
Elements
Protect Privacy, Confidentiality ar
Ensure Security Of Data
Prevent Unauthorized Use And Di
Metadata
*paper-based process-OH-Thesis-DOORS - linux v8.jucm
95
Protect Privacy, Confidentiality and Security Of Data
16
Prevent Unauthorized Use And Disclosure
31
Limit Access to Identified Data
0
Ensure Accountability Of Data User
Provide Access via Approval Process
DW Administrator
0
Technical Review
+
Palette
Select
Links
Decomposition
Contribution
Dependency
Belief Link
Components
Actor
Elements
Softgoal
Goal
Task
Resource
Belief
KPI Model
Indicator
Dimension
KPI Model Link
ReviewRequest CPOReview ExpeditedREBReview BG model 2.3 PM 2.3 BG model 2.2 PM 2.2 »9
paper-based process-OH-Thesis-DOORS - linux v8_changed importance.jucm
Researcher
Enter
requestForPHI
IN1
OUT1
Accept
getToAnAgreement
amendDocuments
[NewRequest]
getRejecti
Reject
stopAmend
[Give Up]
Palette
Select
PathTool
Components
Team
Object
Process
Agent
Actor
Other
Paths
Responsibility
Stub
Dynamic Stub
Pointcut Stub
Timer
Waiting Place
Direction Arrow
Or Fork
And Fork
Or Join
And Join
Researcher-RootMap ReviewRequest CPOReview ExpeditedREBReview BG model 2.3 PM 2.3 »10
Scenarios and Strategies
Importance Strategies (14699)
Importance Strategy - Approval (14758)
Importance Strategy - CPO Review (14700)
Importance Strategy - DW Review (14702)
Importance Strategy - REB Review (14701)
KPI Strategies (2)
Properties Problems Console Plug-in Dependencies Search

Property	Value
id	9911
name	Provide Access via Approval Process
Parent	[unbound]
⊟ Metadata	
Metadata	[click to edit]

Fig. 4 jUCMNav tool for URN models

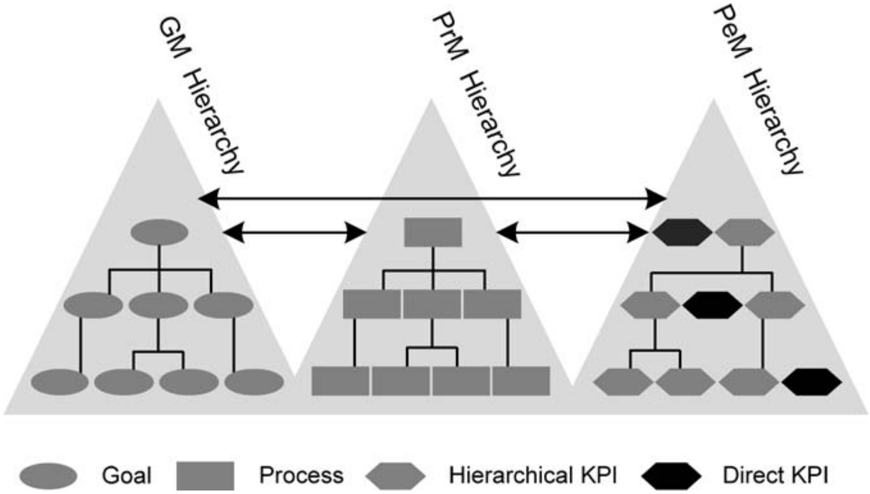


Fig. 5 Organization of framework models

3 An integrated framework and methodology for BPM

This section presents our integrated framework, including its three types of core models, together with a supporting tool infrastructure and a methodology for aligning business processes with business goals.

3.1 Framework core models

The three core artefacts of our proposed framework are models for goals, processes, and performance. Figure 5 and Fig. 6 describe their associations and mutual effects.

Figure 5 shows that all three main artefacts are modeled in a hierarchical manner. The hierarchy in the goal model ("GM Hierarchy" in Fig. 5) is based on the decomposition of high-level goals into operational goals that can be used to decide about the atomic elements of the process models (i.e. tasks, responsibilities, and components). The goal hierarchy can also reflect the process hierarchy. After defining the processes at different levels of abstractions, one can create a goal model for each level. Goal hierarchies are demonstrated using GRL contribution links and diagrams. On the other hand, process hierarchies ("PrM Hierarchy" in Fig. 5) can be motivated by the organizational structure of a business or by abstraction layers defined by process authors. The process hierarchies are modeled using UCM stubs and plug-ins (sub-maps).

In the performance model ("PeM Hiearchy" in Fig. 5), both hierarchical KPIs and direct KPIs are defined. While hierarchical KPIs at higher levels of the hierarchy are the aggregation of the same type of KPI at lower levels, direct KPIs only evaluate one particular process or sub-process and do not affect other layers.

Figure 6 defines these artefacts and the relationships between them. Goals and processes are defined to fulfill specified requirements. Each process model is a use

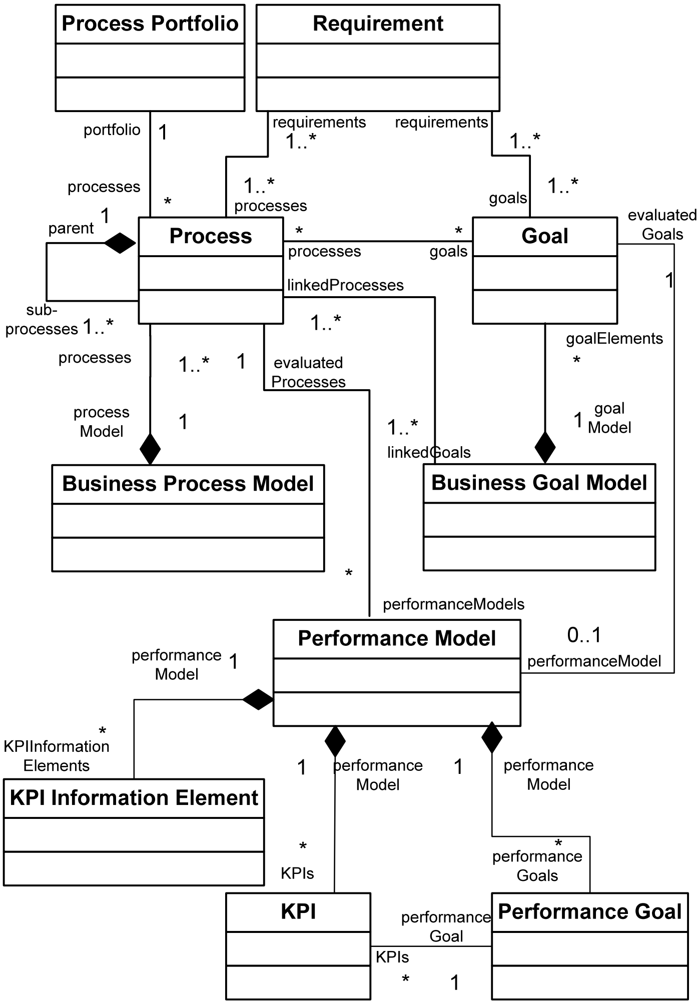


Fig. 6 Framework modeling components

case map constructed by one or multiple processes. A process portfolio is a collection of related processes analyzed and monitored together. Each process model could have single or multiple associated goal models and each goal model can describe the objectives of one or multiple processes. Furthermore, process models contain processes, which can be further composed from sub-processes in a hierarchical manner. Each process can have multiple performance models that evaluate the processes from different aspects of the business or from the perspectives of multiple stakeholders. In addition, performance models are always associated with a goal, and each evaluated process can have multiple performance models. Finally, a performance model consists of KPI information elements, KPIs and performance goals. Performance goals

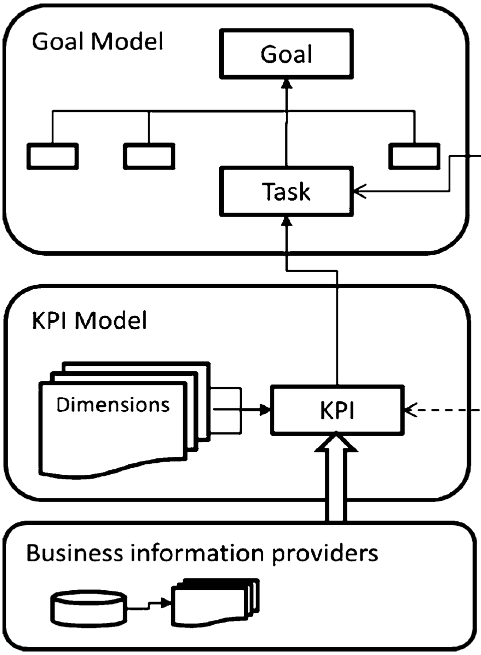


Fig. 7 Overview of the KPI model evaluation mechanism

are used to specify the objectives of the KPIs defined by the performance model, and to aggregate the evaluation results of related KPIs.

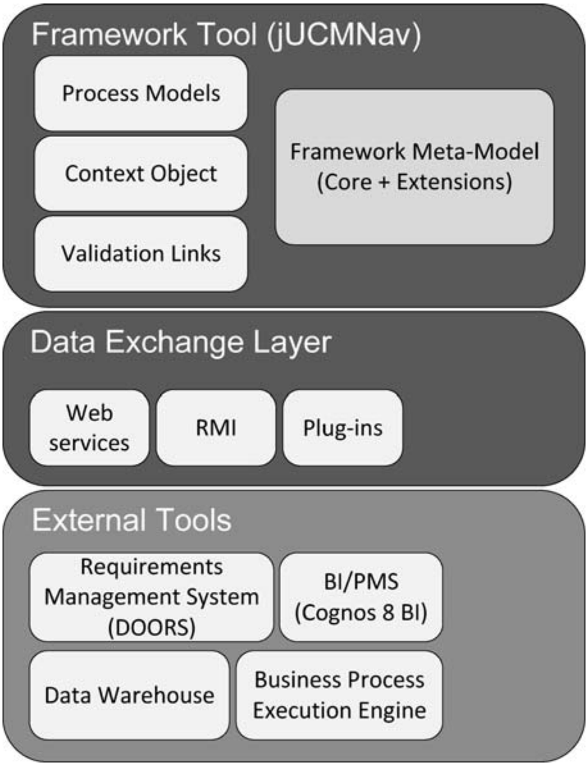
Figure 7 provides an overview of the KPI model evaluation mechanism. Evaluation context information for the KPI, including the linked process and dimensions, can be provided to business information providers for the building of data models and generation of KPI reports. Next, the KPI values generated in the reports will be retrieved, e.g. from a data warehouse, and transferred to the KPI model. Then, through the contribution links and the links between goals and processes, the satisfaction levels of goals and the performance status of the processes can be calculated and presented to users. This evaluation mechanism will be more formally described in Sect. 4.

3.2 Framework supporting tools and infrastructure

Our proposed framework consists of different layers of components in an open, extensible architecture. As illustrated in Fig. 8, the core component is the framework meta-model, which describes the semantic concepts for creating process, goal, and performance models. A subset of this meta-model is shown in Fig. 11. The core framework tool, jUCMNav, is built on top of this enhanced meta-model.

The framework allows one to model processes, goals, and context objects (e.g. KPIs and compliance constraints). Validation links allow one to connect models to external tools through the data exchange layer. Although the framework is generic enough to support different types of technologies (e.g. Eclipse plug-in, RMI, etc.),

Fig. 8 Framework components



we adopted a Service Oriented Architecture (SOA) solution in the current implementation. SOA provides flexibility, scalability, and small development and deployment cycle times and cost [13].

jUCMNav is an Eclipse-based URN modeling tool. Our new KPI module in jUCMNav requests KPI values from the Web services in the data exchange layer and then evaluates goal models in GRL as well as the linked process models in UCM. jUCMNav also includes a scenario traversal mechanism which highlights traversed paths in UCM diagrams, and a GRL strategy mechanism to analyze GRL models using a set of user-defined evaluations [54, 55]. The web services located in the data exchange layer, where information from the miscellaneous business information providers is extracted, formatted and organized, offer an agile and adaptable interface to facilitate the tool's generation of KPI values.

Miscellaneous business information providers including operational databases, data warehouses, BI servers, and other external business applications can provide data for the evaluation process. Multiple information providers can coexist in the same system and provide information from different sources and perspectives for KPI evaluation.

3.3 Methodology

A common multi-phase and iterative set of tasks is shown in Table 2 and is detailed for our methodology for aligning business processes with business goals. The aim

Table 2 Framework steps and iterations

Step	Objectives	Tasks	Artifacts	Roles	Tools
Target Selection	Based on mission statement or strategy, pick the right target for the first iteration of the project.	• Identify Re, BG and Pr • Specify the relationship between above artifacts • Indicate the target process	RD GM PrM Iteration Plan	Process Expert Business Analyst Requirements Engineer Process Modeler Software Engineer DW & BI Expert	RMS jUCMNav
Artifact Modeling	Complete the modeling of main three artifacts and specify their relationship.	• Model GM till task level • Model PrM till task level • Link GM and PrM models • Specify performance objectives and designing PeM • Link PeM with PrM and GM	GM PrM PeM Strategies	Process Expert Business Analyst Process Modeler	jUCMNav
Data Source Preparation	Provide the required sources of information for monitoring the process from available resources across the organization.	• Find main sources of data • Design a DW as central repository • Extract a data mart from DW • Design dimensional model • Provide required KPI reports • Provide the required WS	DW BI Model BI Reports WS	Process Modeler DW & BI Expert Software Engineer/ Developer	Database BI tool Application Server
Monitoring	Observe the performance of processes based on defined performance model and their impact on the business goals.	• Performance analysis • Impact analysis • Select the improvement candidates	Altered Strategies New Strategies Analysis Reports	Business Analyst Process Expert Process Modeler	jUCMNav BI Tool

is to include all processes in the framework, execute them, monitor their performance continuously and improve them incrementally. Monitoring and performance management is an ongoing task as there is always room for improvement in both processes and business goals. In addition, changes in the business environment can lead to changes in processes that require validation and adaptations in performance models.

Table 2 (*Continued*)

Step	Objectives	Tasks	Artifacts	Roles	Tools
Alignment	Suggest improvement to processes or goals.	• Use redesign patterns to improve the processes • Adjust business goals	Remodeled GM Remodeled PrM Remodeled PeM Altered Strategies	Process Expert Business Analyst Process redesign expert	jUCMNav Documentation Redesign Patterns

App: Application
BG: Business Goals
BPM: Business Process Management System
DES: Data Exchange Services
GM: Business Goals Models
Re: Requirements
WS: Web Services

RD: Requirements Documents
Pr: Processes
PrM: Process Models
PeM: Performance Models
PMR: Performance Management Reports
PMS: Performance Management System
RMS: Requirements Management System

3.3.1 Target selection

A common best practice among quality methods is to develop projects in an incremental and iterative manner. This best practice has been adopted in our methodology [51].

In the first iteration, based on the priority of requirements and relative importance of goals, the corresponding process, which satisfies the high priority requirements and goals, is selected as the target. If the selected process is still too large for available resources and project timeline, then one or multiple sub-processes of that process are selected and modeled. In parallel, goal models are also decomposed to reach to the same level as the process.

After a full cycle of the project, when we have a complete environment with all important processes modeled and monitored, the selection of target processes is easier. At this point, we can select the targets based on monitoring results, the performance of the processes and their impact on business goals. In Sect. 4.3, we propose an extension to the existing tool to assist process experts and business analysts in selecting targets.

Note that the amount of effort required for our methodology, and the potential benefits that can be accrued are based on this critical step of selecting what are the important processes to be modeled and monitored. It is not necessary for all processes to be modeled and monitored in order to achieve benefits from this methodology.

3.3.2 Artifact modeling

In the second step of the methodology, process experts, business analysts, and process modelers create the goal, process, and performance models for each selected process accurately and in detail. jUCMNav is used extensively to model all the artifacts and to link them together. Additionally, they specify the performance objectives along the way and use GRL strategies to reflect them into the models. Furthermore, they define monitoring dimensions and map them to performance models. At this point in time, the models are ready and, with the artifacts produced so far, it becomes possible to experiment with "what if" scenarios by assigning evaluation values manually and observing the results of process performance impact on the goal models. We must go to the next step of the methodology only if we need to perform the monitoring automatically and based on information received from external sources on an ongoing basis. Otherwise, we already have a complete working model that integrates goals, processes, and performance.

3.3.3 Data source preparation

In this third step, the information sources required to monitor the process are provided [5]. Software engineers first identify the operational information systems generating the information related to the target process. Depending on the target process and operational information systems, this could be as simple as finding one source of data. In more complex environments, this might involve going beyond the boundaries of the organization.

Next, the operational data is gathered in a data warehouse. If a DW already exists, then it is customized as necessary to provide the required data. Using the data stored in the DW, a data mart, a corresponding OLAP model, and KPI reports are generated. Detailed instructions on how to provide these artifacts are summarized below.

A data mart contains the required information related to multiple processes and performance models. Performance models created in jUCMNav in the previous step are used to help DW and BI experts come up with the data mart model. Dimensions, the associated process, the granularity of the data mart schema, and the facts are defined using Table 3. This table shows the mapping between information presented in the performance models and required information for designing a data mart.

After preparing the sources required for calculating the value of KPIs, we should also provide the infrastructure in the data exchange layer for communicating the data with the process monitoring tool, as suggested in Fig. 9. Software developers play a significant role at this stage for customizing or implementing a suitable data exchange layer. If done properly, this is a one-time task in the first iteration that does not need to be repeated in subsequent iterations.

Table 3 Mappings between performance model and required data mart

Data mart design elements	Performance model information
Dimensions	Information elements
The associated process	Linked process to the KPIs
Granularity of data mart schema (the smallest grain of data mart)	The lowest level of detail in all performance strategies defined on all related KPIs; or, the lowest possible level of detail.
Facts (what is being measured)	The essential metrics required to calculate KPIs

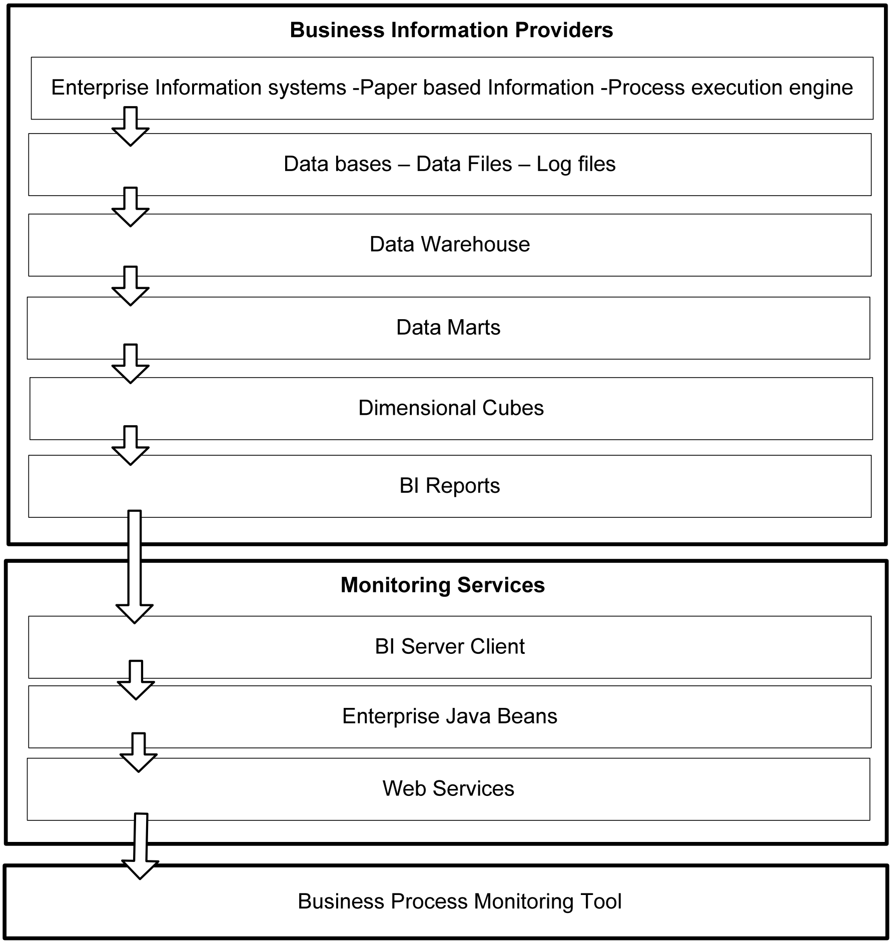

Business Information Providers
Enterprise Information systems -Paper based Information -Process execution engine
Data bases – Data Files – Log files
Data Warehouse
Data Marts
Dimensional Cubes
BI Reports
Monitoring Services
BI Server Client
Enterprise Java Beans
Web Services
Business Process Monitoring Tool

Fig. 9 Architecture and information flow of the monitoring system

3.3.4 Monitoring

The objective of the monitoring step is to perform business performance analysis, business goal impact analysis, and improved candidate selection. Performance analysis involves triggering the defined performance strategies and observing the performance models and evaluation values as they are propagated from KPIs up to the task or sub-process attached to the performance model. Impact analysis involves observing the goal models to see which goals or sub-goals are impacted by the current performance of the process. In addition, the existing strategies can be altered or new strategies may be defined to monitor the process in other dimensions or based on different value sets. This can be done through further investigation or by exploring "what if" scenarios.

3.3.5 *Alignment*

The last step, alignment, consists of business processes and business goal improvement. In case of processes with poor performance not aligned with business goals, process redesign patterns introduced in [53] can be used to improve business processes. On the other hand, in case of healthy processes, which meet or surpass the specified target values, the goals may need to be revisited and the expectation bar might be raised. Goal improvement, however, may only be considered when there are no other priorities (e.g. other critical processes).

After the alignment step, the project moves forward to start a new iteration. In the process modeling step of the next iteration, the suggested redesign patterns are applied to the processes. In addition, if there are still processes that need to be added to the process portfolio, one of them is selected based on its priority.

4 Performance monitoring with URN

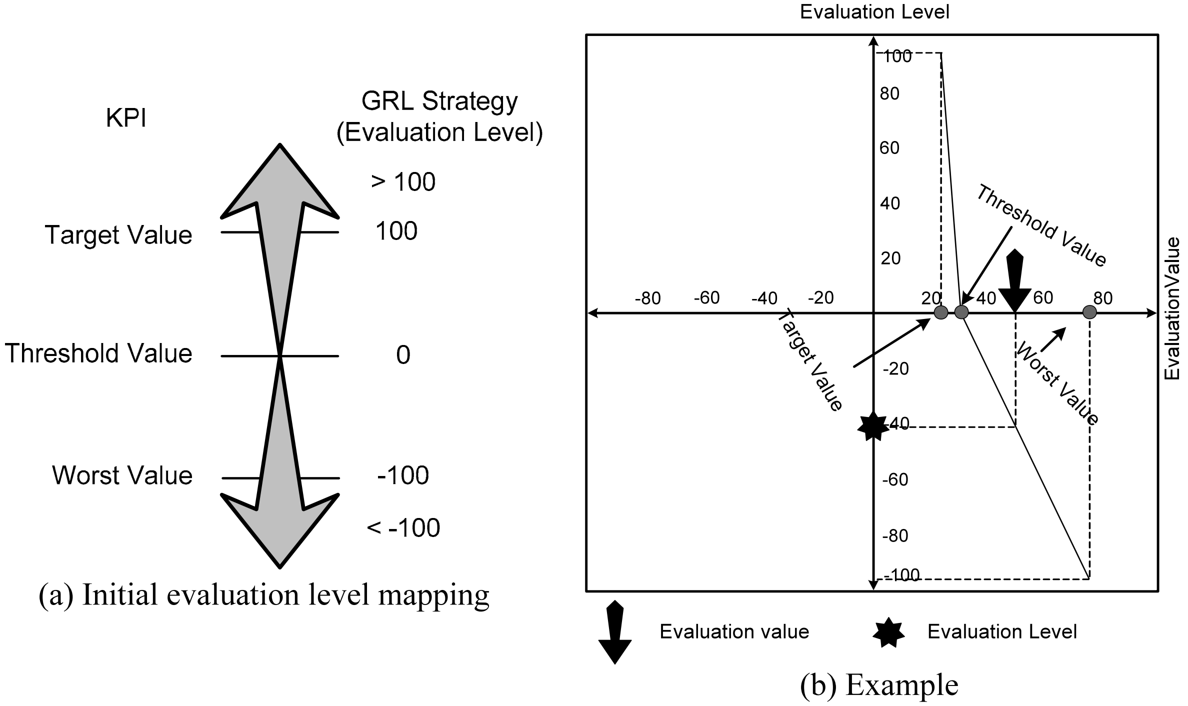
This section presents the extensions we have done to the User Requirements Notation and to the jUCMNav tool to support performance monitoring. Performance, impact, and portfolio analysis techniques are then described that take advantage of these new additions.

4.1 URN meta-model extensions for performance monitoring

Mapping of KPI evaluation values to GRL evaluation levels is done through four value sets associated to each KPI: Target Value, Threshold Value, Worst Value and Evaluation Value. Target Value is the expected performance of the process under evaluation. Threshold Value is used to separate acceptable from unacceptable values, while Worst Value is used to specify the most serious condition from a users' perspective. Although any value between Threshold Value and Worst Value is unacceptable, the definition of a Worst Value helps specify the level of dissatisfaction. These three values are adjustable as required. The Evaluation Value is the KPI's *actual* value retrieved from back-end business data sources at run-time or defined by users for test purposes at design time (for what-if scenarios). Figure 10a illustrates how these four values are mapped to GRL evaluation levels. As jUCMNav supports a range of evaluation levels from -100 (totally dissatisfied) to 100 (totally satisfied), we map these to the Worst and Target Values respectively, while the Threshold Value is mapped to 0. We use linear functions to interpolate satisfaction levels for Evaluation Values, as illustrated in the example of Fig. 10b. These concepts are elaborated further in [52].

Business process performance modeling is a new application area for URN. As a result, the original URN meta-model needed to be extended to support the required functionalities. The extensions are shown as shaded classes in Fig. 11.

In Fig. 11, the Indicator class inherits all features of GRL intentional elements. Thus, a KPI can be used for evaluation in a way similar to tasks and goals in a GRL model. Indicators also have some specific features such as groups that can be used



(a) Initial evaluation level mapping

(b) Example

Fig. 10 KPI evaluation value mapped to GRL evaluation level

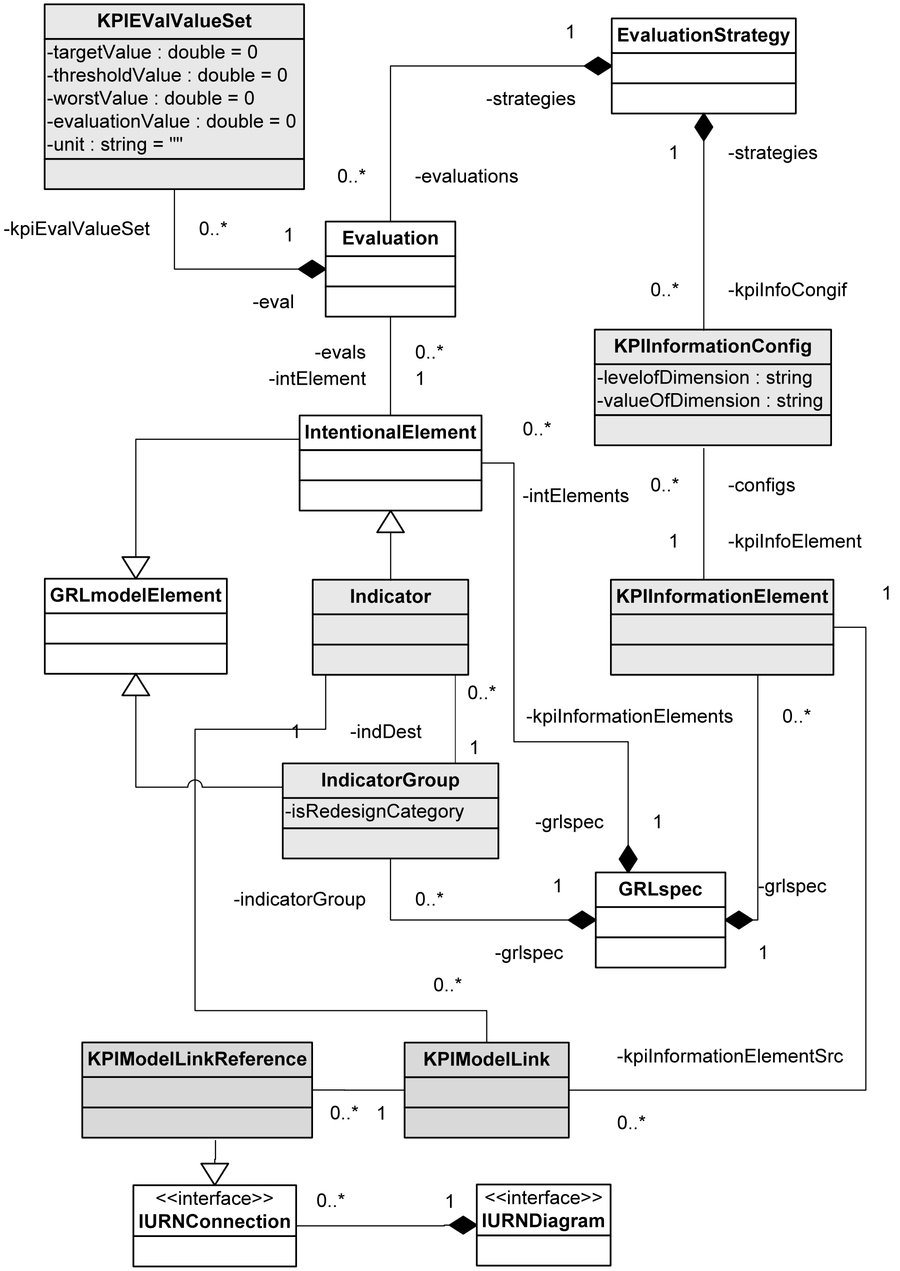


Fig. 11 Meta-model extensions for performance models and KPI evaluation

to categorize KPIs for redesigning, filtering and other user-defined purposes. In addition, performance model links are used to link indicators to KPI information elements. These elements allow users to add detailed information (e.g. dimensions) to performance models. Dimensions can specify data categories and hierarchies used for aggregating and summarizing data [18, 29].

In addition, Fig. 11 illustrates the extensions to GRL evaluation strategies. Unlike the initial design for indicators in [52], the KPI value sets in the new meta-model are defined as values of KPI strategies. This new design allows users to define multiple strategies and corresponding evaluation value sets for a KPI. This feature also helps users define and compare different possibilities. The KPIInformationConfig class contains a set of specific dimension information that can be used for retrieving KPI values under each strategy. That dimension information includes the level of dimension and the value of dimension. The dimension settings define a specific context for KPI evaluation and the setting information will be used to build data models and KPI reports on the BI server. For example, for the KPI "number of complaints" on the time dimension in Fig. 19, users may want to evaluate it on a monthly and daily basis in different evaluation strategies, so the level of dimension could be month and day.

When building the data model, the smallest or lowest level will be chosen to be the granularity of the data model. On the other hand, when creating KPI reports, the value of dimensions in different evaluation strategies can be used to select ranges of values from the data model, such as the year 2006 or February 2007.

4.2 Performance and impact analysis

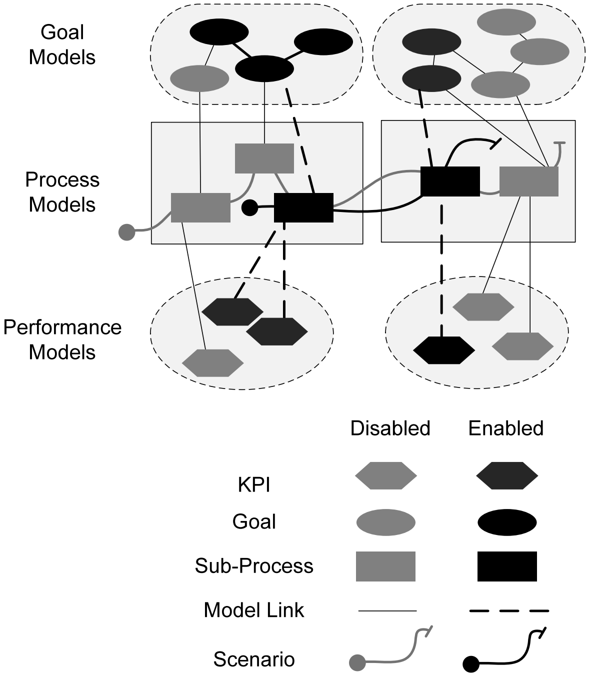
Since business processes usually span organizational structures, their monitoring and analysis introduces specific requirements that cannot be fulfilled by ordinary BI tools [46]. Combining business process monitoring with UCM scenarios [27] enables business users to do performance analysis and impact analysis on specific parts of the processes of interest.

As shown in Fig. 12, scenarios are used to highlight parts of processes as they appear in different models. The process models are connected to performance models from one side and to goal models from the other side. After a scenario definition is enabled by the user, our tool activates all the KPIs that have been defined in that path. Then, the performance of that part of process is shown. This displays the impact of the highlighted part of the process on the related goals of the organization.

Monitoring and performance analysis of processes in an automated and integrated manner is a new concept, based on work for integrating scenarios and goals in [55], that demonstrates the real value of URN in the context of BPM. Figure 13 shows the workflow of scenario-based impact analysis among several modules of the tool.

First, the user chooses a scenario and a performance evaluation strategy to give the evaluation a context (1), then the scenario and strategy view sends a request to the scenario manager (2). The scenario manager interacts with the performance/goal model component to find the relevant performance models and map them to their associated processes and tasks (3). Subsequently, by scenario manager's request (4), the KPI manager retrieves KPI values from the monitoring services (via the data exchange layer) and maps them to evaluation levels (5), as seen in Fig. 10. The strategy is then triggered by the scenario manager (6) and the evaluation results are presented to the users on the new jUCMNav KPI View, with their impact simultaneously shown on the goal model (7).

Fig. 12 Scenario-based process analysis



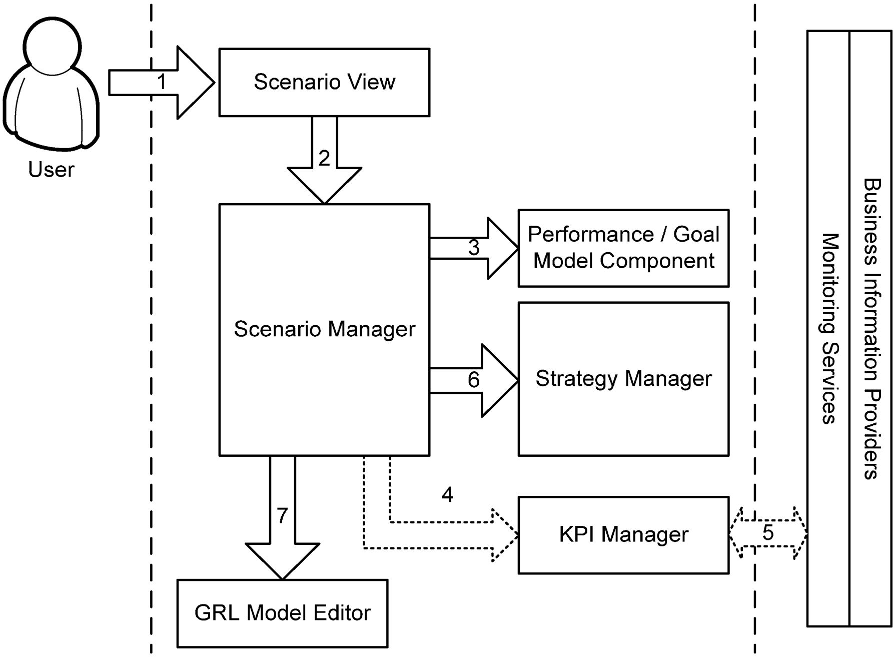


Fig. 13 Scenario-based performance and impact analysis—run-time sequence model

Figure 14 highlights the extensions made to jUCMNav's user interface to support the new framework. jUCMNav extends standard Eclipse views (Outline, Properties, and Problems) with graphical editors for UCM (process model) and GRL (goal model). In our framework, the GRL editor was further extended to support the definition of KPIs and dimensions. The Scenarios and Strategies view is used to define and execute GRL strategies and UCM scenarios, which results in color feedback in the models.

This view was extended to support KPI strategies and evaluation. In the middle of Fig. 14, an evaluated performance model is illustrated. This model is evaluated based on the strategy selected on the left side in the scenarios and strategies view. On the right side, the Key Performance Indicators view shows detailed information related to all the KPIs defined in the performance model. This view is synchronized with the diagram currently studied by the user. In other words, if we switch to another process or performance diagram of this model, the KPI view will be updated accordingly. On the left hand side, we also have the list of Key Performance Indicators with their statuses.

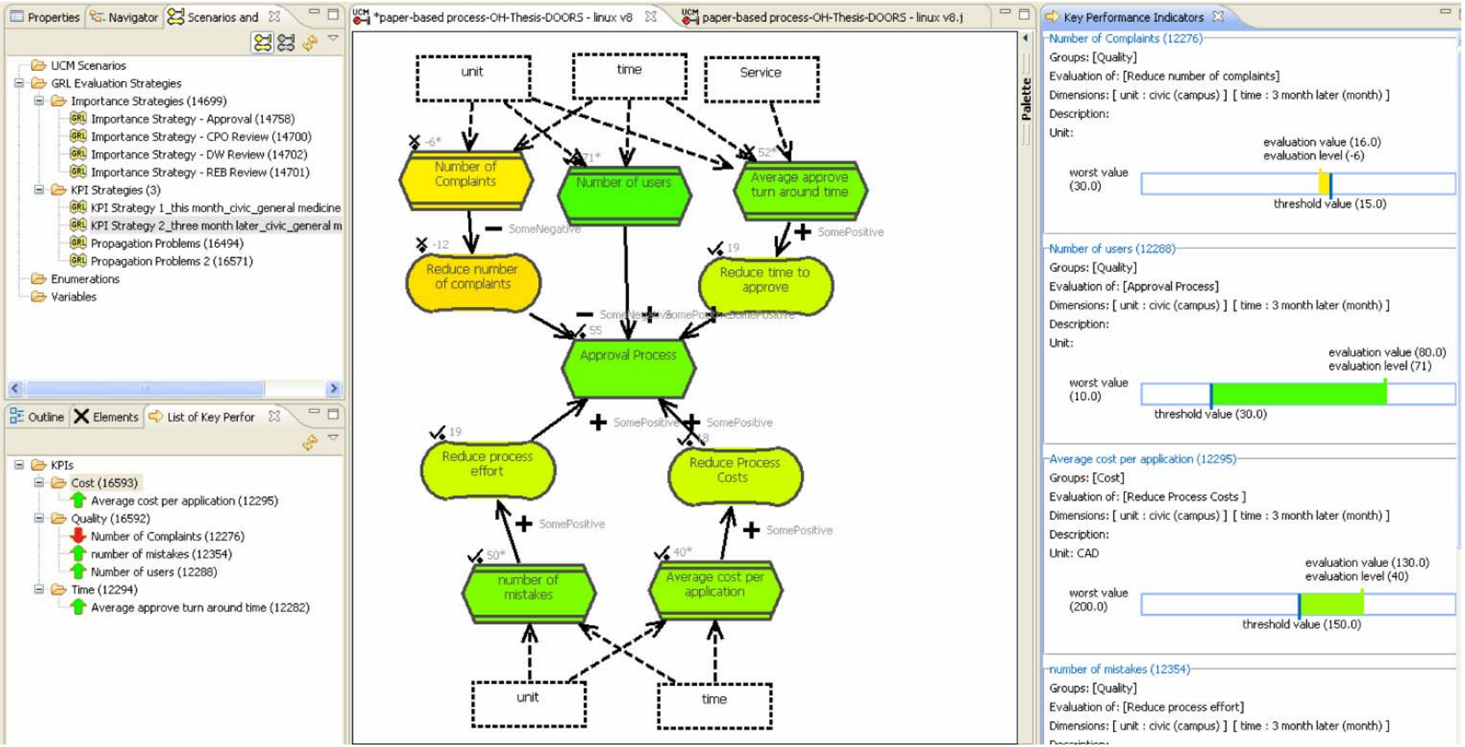
4.3 Process portfolio analysis

A *process portfolio* consists of multiple processes modeled in an organization. This view is a representation of all processes in a hierarchical quadrant with drill-down and drill-up capabilities. It gives users an overall understanding of the current status of existing processes and helps them decide the next step in a BPM project (e.g., prioritization of the next targets for improvement or the next candidate process for outsourcing [21]). In addition, it provides the capability for users to pinpoint a malfunctioning process and drill down to the next levels of abstraction to find out the causes of problems (Fig. 15).

Although using quadrants for analyzing process portfolios has been already introduced in [21], using process models (UCM), goal models (GRL), and performance models (GRL+KPI) to provide this capability in a hierarchical structure is a novel method. Goal and performance models are used respectively to calculate the *importance* and *performance* values of the processes that are the two axes of the quadrant (Fig. 15) and process models are used to specify the hierarchical structure.

In our definition, the *importance value* is the average satisfaction level of the top-level business goals when a process performs at its 100% capacity. In other words, importance is the impact of the business process on the business goal model. The *performance value* is calculated using the KPIs and performance model defined for the process as described in Sect. 4.1. Each layer of a process portfolio view includes all the responsibilities and sub-processes (stubs) of a UCM map. This view is synchronized with the process view and the KPI view. Hence, while users browse the processes (i.e. individual UCM maps) this view and the KPI view are updated based on the current process being observed.

The workflow among the relevant modules of the framework to support the process portfolio view, shown in Fig. 16, is very similar to the one in Fig. 13. First, the user chooses a process from the outline view or drills-down (using stubs) to a sub-process in the UCM view (1). A request is then sent to the process portfolio manager (2).



UCM Scenarios
GRL Evaluation Strategies
Importance Strategies (14699)
Importance Strategy - Approval (14758)
Importance Strategy - CPO Review (14700)
Importance Strategy - DW Review (14702)
Importance Strategy - REB Review (14701)
KPI Strategies (3)
KPI Strategy 1_this month_civic_general medicine
KPI Strategy 2_three month later_civic_general m
Propagation Problems (16494)
Propagation Problems 2 (16571)
Enumerations
Variables

KPIs
Cost (16593)
Average cost per application (12295)
Quality (16592)
Number of Complaints (12276)
number of mistakes (12354)
Number of users (12288)
Time (12294)
Average approve turn around time (12282)

Number of Complaints (12276)
Groups: [Quality]
Evaluation of: [Reduce number of complaints]
Dimensions: [unit : civic (campus)] [time : 3 month later (month)]
Description:
Unit:
evaluation value (16.0)
evaluation level (-6)
worst value (30.0)
threshold value (15.0)

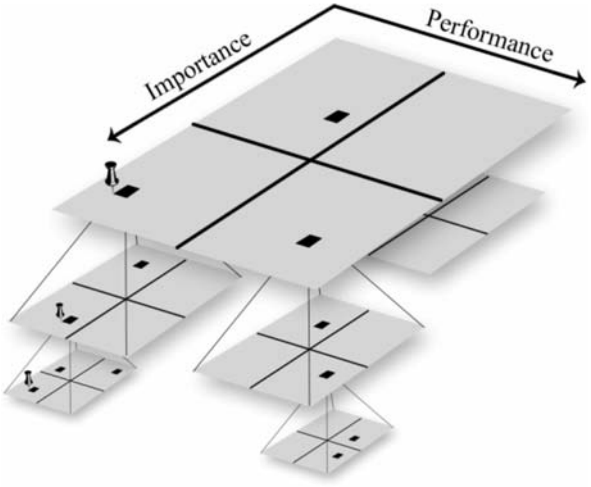
Number of users (12288)
Groups: [Quality]
Evaluation of: [Approval Process]
Dimensions: [unit : civic (campus)] [time : 3 month later (month)]
Description:
Unit:
evaluation value (80.0)
evaluation level (71)
worst value (10.0)
threshold value (30.0)

Average cost per application (12295)
Groups: [Cost]
Evaluation of: [Reduce Process Costs]
Dimensions: [unit : civic (campus)] [time : 3 month later (month)]
Description:
Unit: CAD
evaluation value (130.0)
evaluation level (40)
worst value (200.0)
threshold value (150.0)

number of mistakes (12354)
Groups: [Quality]
Evaluation of: [Reduce process effort]
Dimensions: [unit : civic (campus)] [time : 3 month later (month)]

Fig. 14 jUCMNav views for KPI and BPM

Fig. 15 Process portfolio—drill down scenario



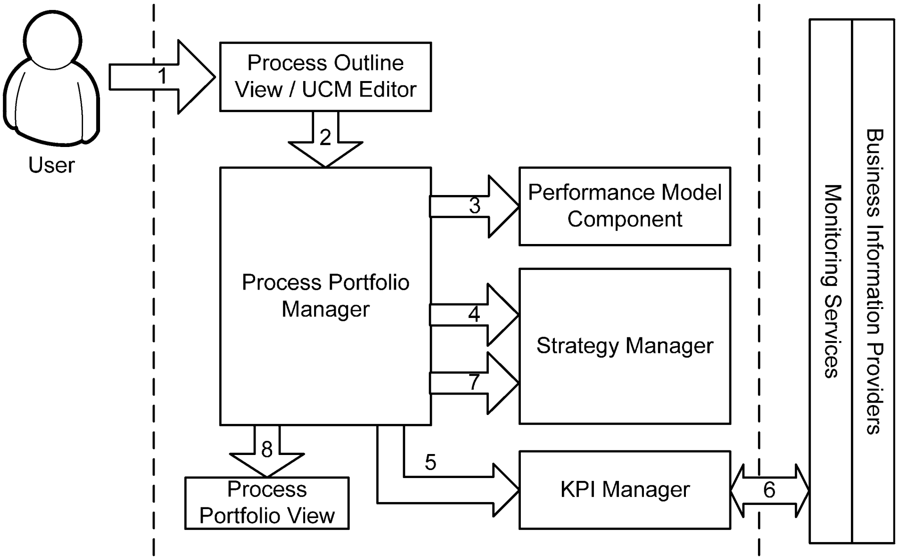

User
1
Process Outline View / UCM Editor
2
Process Portfolio Manager
3
Performance Model Component
4
Strategy Manager
7
8
5
Process Portfolio View
KPI Manager
6
Monitoring Services
Business Information Providers

Fig. 16 Process portfolio quadrant: run-time sequence model

Subsequently, the process portfolio manager finds all the performance and goal models that are related to the selected map/process (3). Next, an importance strategy is created, initialized, and triggered dynamically to calculate the importance value for each process (4). The process portfolio manager then retrieves KPI values through the KPI manager (5 and 6), creates a performance strategy, and initializes the evaluation levels of KPIs (7) to calculate the performance values. One could also, however, use this feature without being connected to external sources, i.e., simply by initializing the evaluation values internally for simulation purposes. Finally, importance values and performance values for each sub-process or responsibility in the selected process are used by the portfolio manager to calculate their locations on the quadrant and show them on the new process portfolio view (8).

5 Validation example

Given the extensions to URN, supported by an enhanced jUCMNav tool, and the new analysis capabilities it supports as seen in Sect. 4, we will now illustrate how our methodology and framework (seen in Sect. 3) can be applied in a case study.

5.1 Example overview, target selection, and artefact modeling

To improve its quality of care, a large research hospital in Ontario aims to facilitate the access to its data warehouse (DW) to researchers and administration staff. However, in granting access to a large DW there are many issues which need to be addressed, including compliance with personal health information protection regulations in Ontario (PHIPA) [50]. To address these issues, the hospital has put in place a business process called *DW Approval Process*. Figure 17 illustrates the top-level view of this process, whereas Fig. 18 and Fig. 19 describe the business goals and performance models respectively.

The most important part of this process is the *review request* sub-process, shown in Fig. 20, which we selected as our first target for this case study. In a nutshell, the CPOReview sub-process is performed by the chief privacy officer (CPO) to ensure that the requested data will not violate the relevant privacy laws. For REBApproval, the research ethics board (REB) of the hospital evaluates the intended usage of the data. The data warehouse administrator reviews the request (technicalReview) to make sure the requested data can technically be obtained from the warehouse and that the users' access rights are defined properly. However, this review request process takes time (months actually) to be completed, and therefore it needs some improvement.

As illustrated in Fig. 19, several KPIs have been defined which can help improve this process. In the initial step, the business goals and performance models of these three sub-processes are created using jUCMNav. For example, Fig. 21 and Fig. 22 define goal and performance models for the technical review sub-process.

5.2 Data source preparation

We replicated the hospital's DW schema and infrastructure in our lab for this experiment. During our work on this validation example however, we did not have access to any existing computer-based data sources related to this process, and the information required to evaluate the KPIs was not available. Therefore, based on initial statistical information received from process experts, we have generated artificial KPI evaluation values to continue with the rest of the case study and exercise all the steps of the methodology. The data was generated artificially using the tool built by Zhan [65] to produce data looking similar (in size and trends) to what is usually seen in reality.

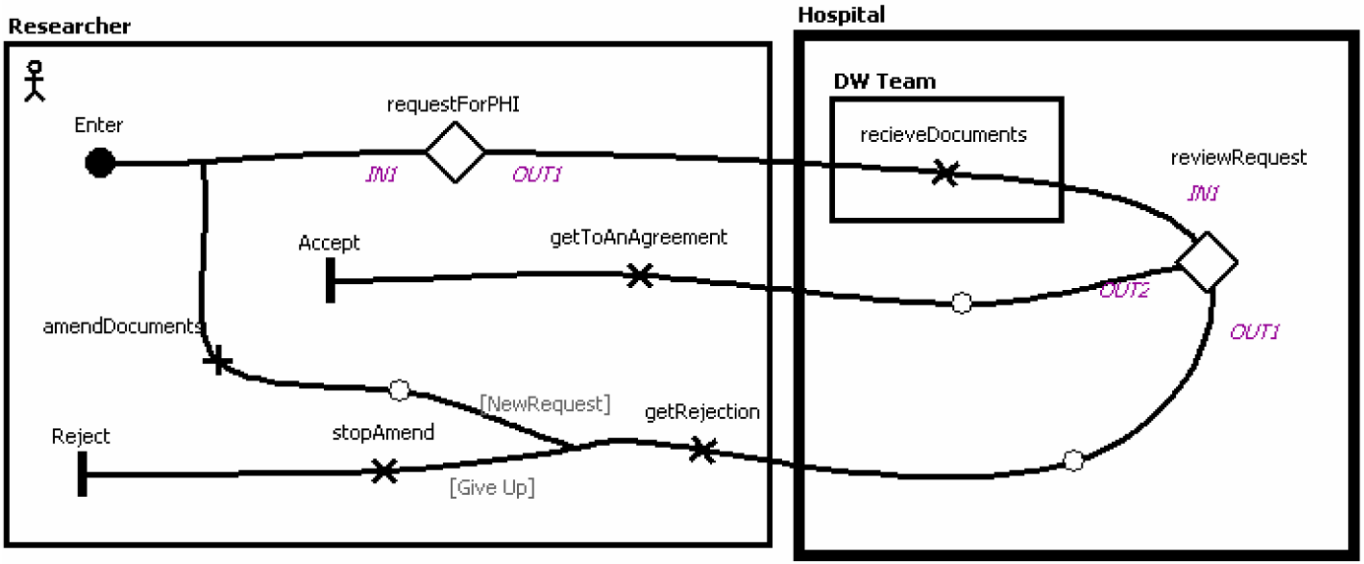
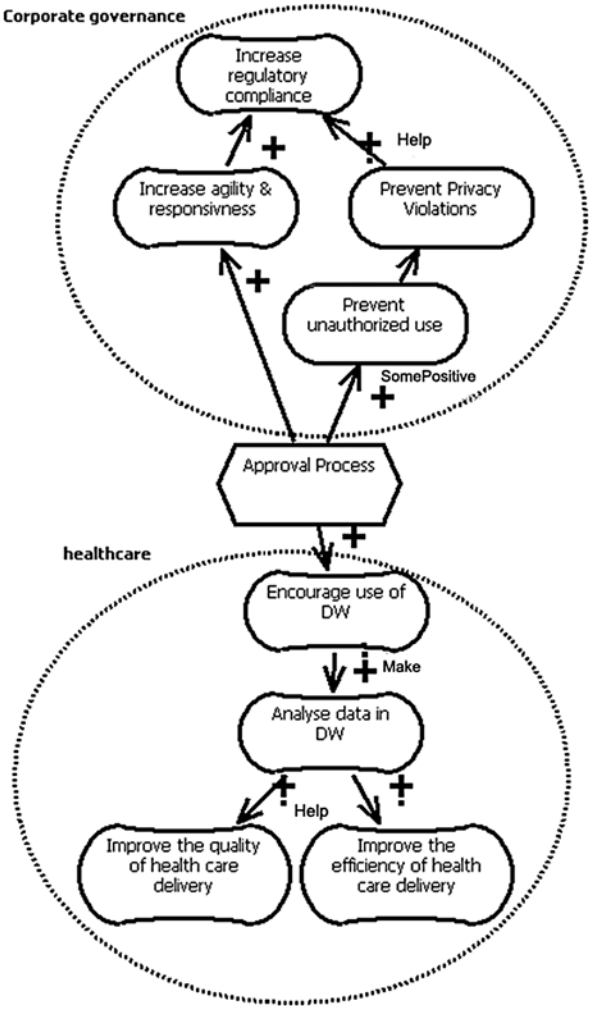


Fig. 17 DW access approval process—high-level view

Fig. 18 Goal model for the DW access approval process


Corporate governance
Increase regulatory compliance
+
+ Help
Increase agility & responsivness
Prevent Privacy Violations
+
Prevent unauthorized use
SomePositive
+
Approval Process
+
healthcare
Encourage use of DW
+ Make
Analyse data in DW
+ Help
+
Improve the quality of health care delivery
Improve the efficiency of health care delivery

In this section, we briefly explain how the mapping between the performance model and a dimensional data source designed for this process works. We can use this dimensional data source to extract the real KPI values from the appropriate sources of information after they become available in the hospital.

Data source preparation, including the dimensional data source design method, was elaborated in [5]. We use the same principles here to design our data sources. Table 4 and Table 5 show the information required for designing the dimensional data source. These tables include dimension levels and values as well as KPI units. Based on this initial information, we can extract the required data mart from the DW or from operational data sources. Then, we generate a dimensional model similar to Fig. 23.

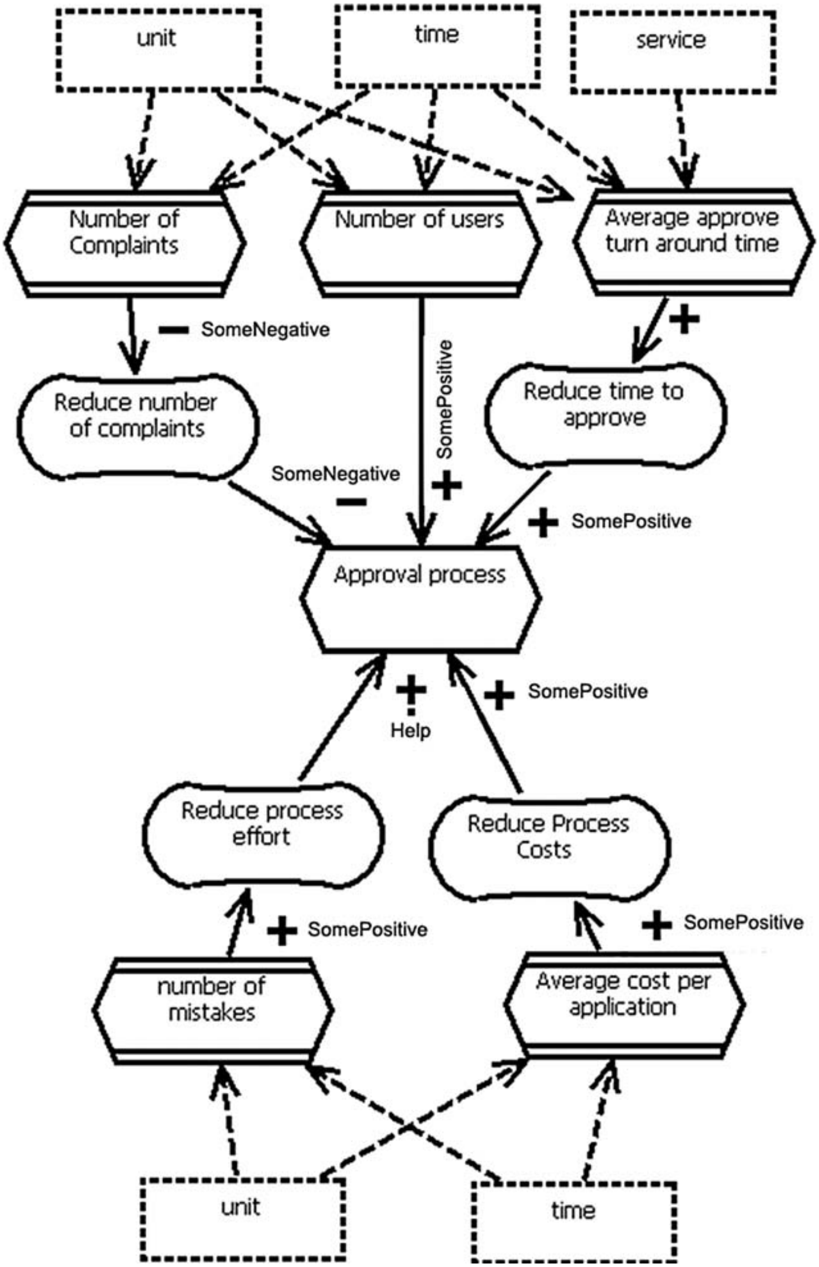


Fig. 19 Performance model for the DW access approval process

5.3 Monitoring

Based on the models presented previously and on the KPI values we generated (Table 6), we obtained the result reported in the last two columns of Table 6 and Table 7.

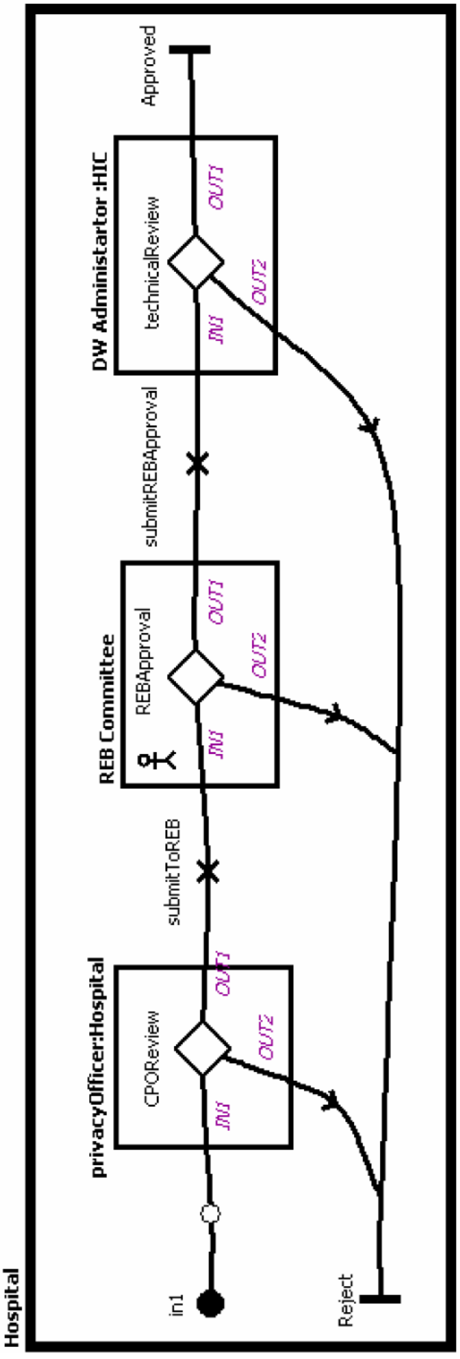


Fig. 20 Review request sub-process

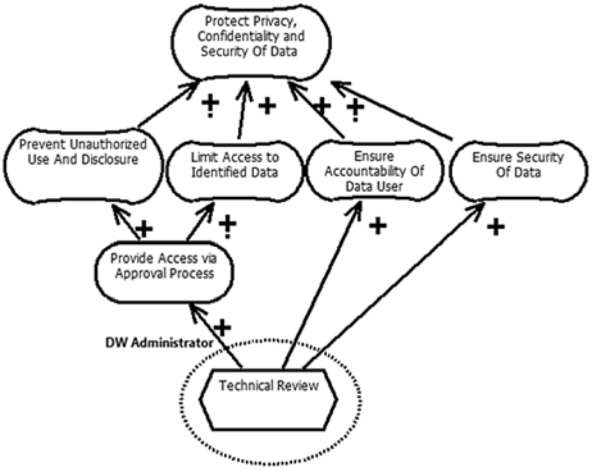

Protect Privacy, Confidentiality and Security Of Data
Prevent Unauthorized Use And Disclosure
Limit Access to Identified Data
Ensure Accountability Of Data User
Ensure Security Of Data
Provide Access via Approval Process
DW Administrator
Technical Review
+

Fig. 21 Goal model for technical review sub-process

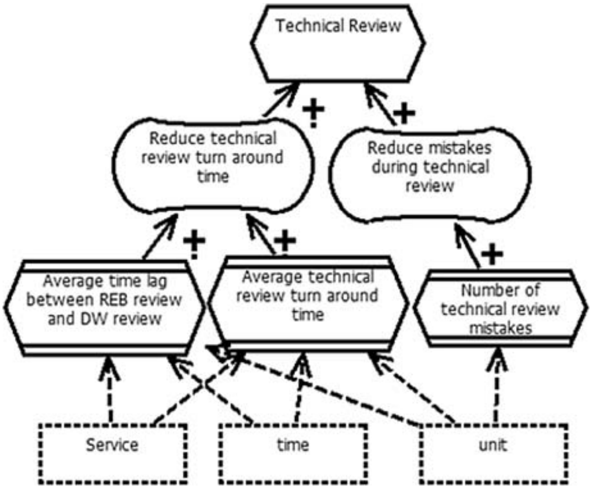

Technical Review
Reduce technical review turn around time
Reduce mistakes during technical review
Average time lag between REB review and DW review
Average technical review turn around time
Number of technical review mistakes
Service
time
unit
+

Fig. 22 Performance model for technical review sub-process

Note that all evaluation levels are positive because evaluation values for all KPIs are between their threshold and target values. Figure 24 shows the corresponding process portfolio view. So far, all sub-processes have almost similar levels of importance, but the REB review has the highest performance.

5.4 Alignment

To further showcase the properties of our BPM framework's methods and tool, we consider the following scenario. After a three-month period, the hospital substantially increases the number of DW users. Presumably, the results observed in this new context would be updated as shown in Table 8 and Table 9.

Table 4 Approval process performance model—dimensions levels and values

	Level enumeration	Value enumeration
Unit	Campus	Civic
		General
Time	Year	2005
	Month	January
	Day	Monday, December 24, 2007
Service	Service	General Medicine
		Emergency Department

Table 5 Approval process performance—model KPI unit

	Unit	Metrics
Number of complaints	Integer	Complaints
Average approve turnaround time	Day	Application submission time
		Application response time
Number of mistakes	Integer	Mistakes
Number of users	Integer	Users
Average CPO review turnaround time	Day	CPO review start time
		CPO review finish time
Number of privacy related complaints	Integer	Complaints
Average time between CPO review and REB review	Day	CPO review finish time
		REB review start time
Average REB review turnaround time	Day	REB review start time
		REB review finish time
Number of review mistakes	Integer	REB review mistakes
Average time lag between REB review and Technical review	Day	REB review finish time
		Technical review start time
Average technical review turnaround time	Day	Technical review start time
		Technical review finish time
Number of technical review mistakes	Integer	Technical review mistakes

DW stakeholders start to send in various types of complaints. A new *type* dimension defined for the complaints KPI allows one to categorize the complaints based on their type. In our example, the number of complaints about DW efficiency is the highest, but we also have some privacy complaints.

After having experienced inefficiencies in the DW, the number of users starts decreasing once more and the hospital realizes that efficiency contributes significantly in encouraging users to use the DW.

As a result, the goal model of Fig. 18 is updated to Fig. 25, and the performance model of Fig. 22 is updated to Fig. 26. The process portfolio view changes to Fig. 27, where issues with the technical review sub-process are highlighted. Such modifications to the goals inevitably lead to changes in the business processes in order to bring

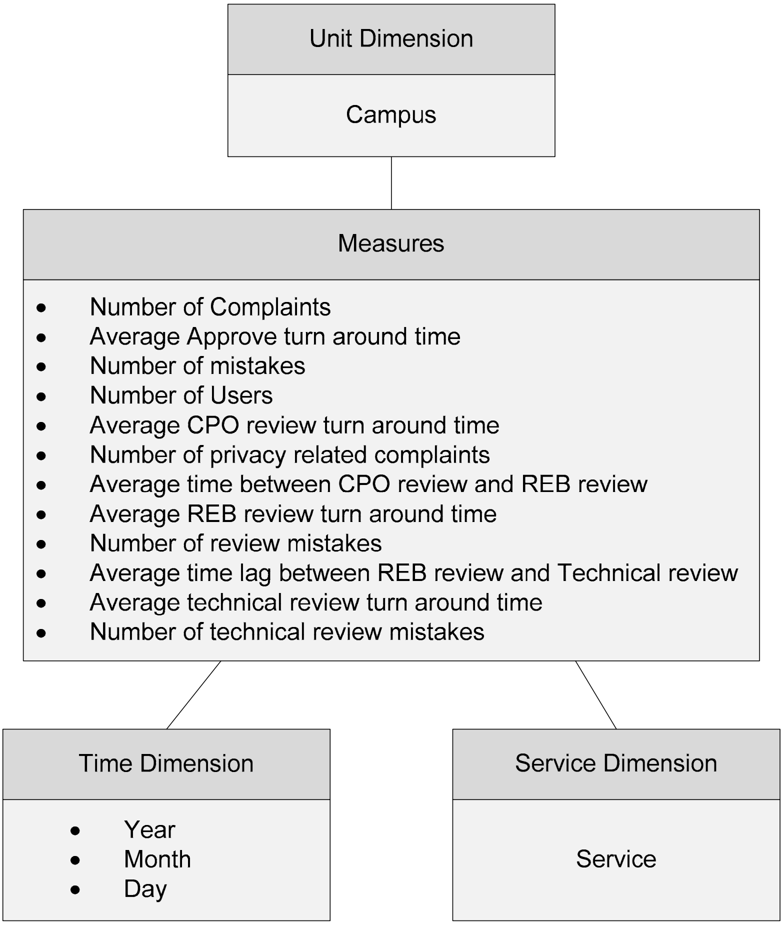

Unit Dimension
Campus
Measures
Number of Complaints
Average Approve turn around time
Number of mistakes
Number of Users
Average CPO review turn around time
Number of privacy related complaints
Average time between CPO review and REB review
Average REB review turn around time
Number of review mistakes
Average time lag between REB review and Technical review
Average technical review turn around time
Number of technical review mistakes
Time Dimension
Year
Month
Day
Service Dimension
Service

Fig. 23 Dimensional data source for DW access approval process

the process portfolio back to a sweet spot performance-wise and satisfy stakeholders within the organization.

Based on the monitoring results, including the number of complaints, the main current problem is the impact of the approval process on the DW performance. To solve this issue, we should improve the technical review sub-process by adding proper tasks to prevent further negative impact on its performance. Figure 28 illustrates the suggested improved technical review sub-process. To address its problems, we added a new task to the process to test performance impacts before finalizing the delivery method. If the initial delivery method has a negative performance impact, DW administrators will modify it to meet the performance requirement. This additional task might however have a negative impact on the technical review turnaround time. We can address this issue by observing the technical review turnaround time in the next monitoring cycle and make further adjustments.

Table 6 KPI value sets

KPIs	T	Th	W	EV	EL
Average CPO review turnaround time (days)	3	15	30	7	66
Number of privacy related complaints	2	7	15	4	60
Average time lag between CPO and REB reviews (days)	1	5	15	3	50
Average turnaround time of viewing REB (days)	2	8	15	6	33
Number of review mistakes	5	12	20	8	57
Average time lag between REB and DW reviews (days)	2	6	16	3	75
Average technical review turnaround time (days)	2	7	15	5	40
Number of technical review mistakes	5	8	15	6	66
Number of complaints	5	15	30	9	60
Number of users	100	30	10	35	7
Average approve turnaround time (days)	7	30	60	18	52
Number of mistakes	10	20	35	12	80

T: Target Value; Th: Threshold Value; W: Worst Value EV: Evaluation Value; EL: Evaluation Level (GRL)

Table 7 Initial portfolio results

	CPO Review	REB Review	Technical Review	Approval Process
Importance	37.6	36.7	37.6	32.67
Performance	71	78	60	89

Table 8 Modified KPI value sets

KPIs	T	Th	W	EV	EL
Number of privacy related complaints	2	7	15	**6**	**20**
Number of review mistakes	5	12	20	**10**	**28**
Average number of DW performance complaints	3	8	15	**10**	**−28**
Number of complaints	5	15	30	**16**	**−6**
Number of users	100	30	10	**50**	**28**
Number of mistakes	10	20	35	**15**	**50**

T: Target Value; Th: Threshold Value; W: Worst Value EV: Evaluation Value; EL: Evaluation Level (GRL)

6 Discussion

The framework and its supporting tool can be compared with other approaches from three perspectives: modeling notation, methodology and integration with tools. We also discuss our experiences and results from applying the methodology to health care processes.

Fig. 24 Initial process portfolio view

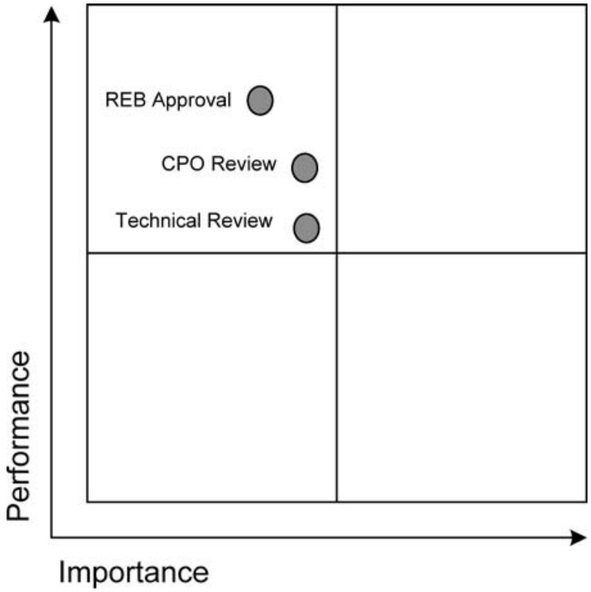


Table 9 Modified portfolio results after changing KPI values

	CPO Review	REB Review	Technical Review	Approval Process
Importance	28.25	27.5	57.75	40
Performance	50	76	49	55

6.1 Modeling notation

One of the most important competitive advantages of the suggested framework is its modeling notation—URN. Using the original standard URN, we are able to model goals and processes. With the proposed extensions, we can also illustrate performance models. Other recent worked has targeted the combination of goals with scenarios for business process monitoring. For instance, Ghose et al. [16] suggested to combine $i*$ with BPMN for the co-evolution of operational and organizational models. Their $i*$ model intentional elements are annotated with *effects*, similar to our evaluation levels, and rules for checking consistency between the goal view and the process view are discussed. However, they do not support our KPI and performance model concepts, nor do they have a mechanism to gather external data, define strategies, and evaluate alignment of processes and goals.

In [19], Greenwood and Rimassa also combine a simple goal notation with BPMN for business process management. They do so however in a context where processes are selected adaptively and automatically, so the goal model is used more for planning purpose than for measuring organizational goal satisfaction. This is more in line with the automatic selection of business process redesign patterns we discussed in [52],

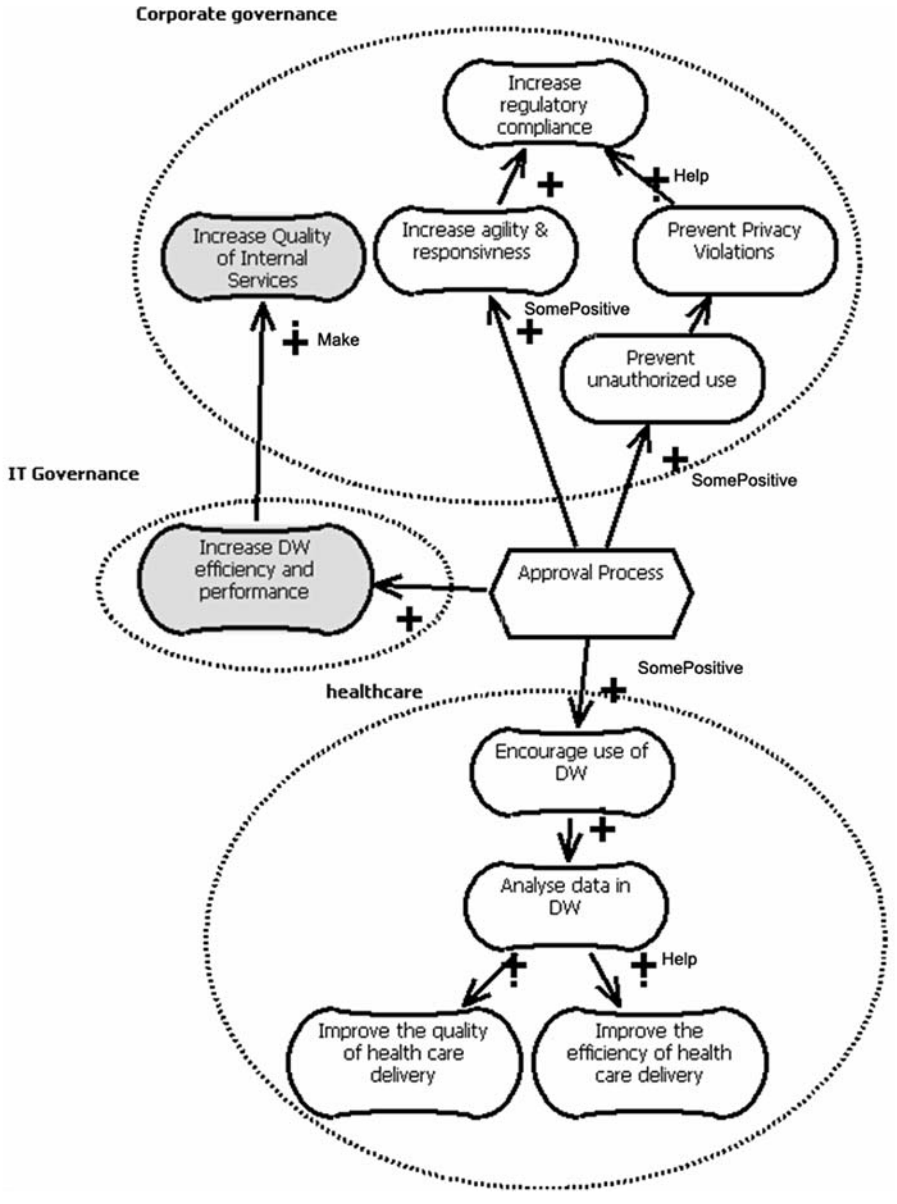

Corporate governance
Increase regulatory compliance
Help
Increase Quality of Internal Services
Increase agility & responsivness
Prevent Privacy Violations
SomePositive
Make
Prevent unauthorized use
IT Governance
SomePositive
Increase DW efficiency and performance
Approval Process
SomePositive
healthcare
Encourage use of DW
Analyse data in DW
Help
Improve the quality of health care delivery
Improve the efficiency of health care delivery

Fig. 25 Revised goal model for technical review sub-process

and with URN-based modeling and monitoring of adaptive telecommunication services developed in [2].

González and Díaz [10] argue that good domain knowledge is key to successful requirements elicitation. They also employ *i** goal models to derive requirements. However, unlike our approach, which defines business goals in a manner that is independent from the business process, their goal models are derived from BPMN busi-

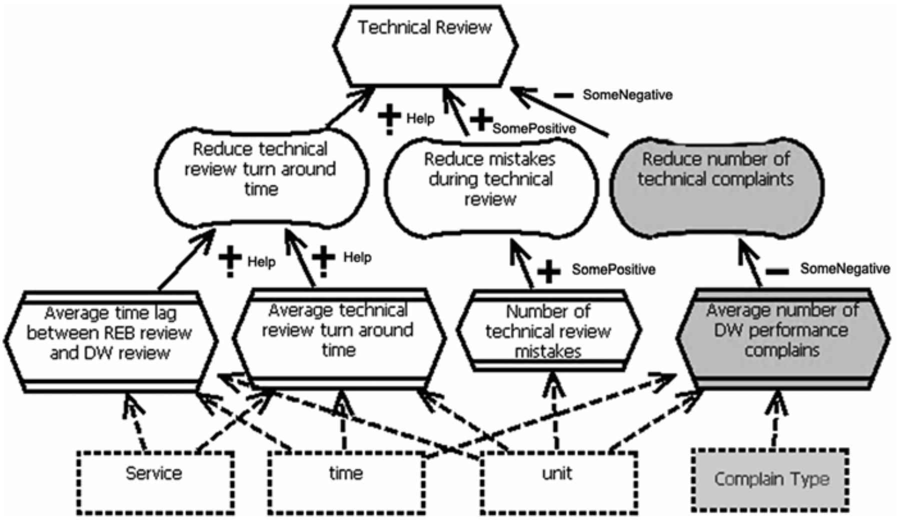

Technical Review
Help
SomePositive
SomeNegative
Reduce technical review turn around time
Reduce mistakes during technical review
Reduce number of technical complaints
Help
Help
SomePositive
SomeNegative
Average time lag between REB review and DW review
Average technical review turn around time
Number of technical review mistakes
Average number of DW performance complains
Service
time
unit
Complain Type

Fig. 26 Revised performance model for technical review sub-process

ness process models. A goal tree containing subgoals for tasks in the business process is then mapped to a use case model for the IT infrastructure. While their work also outlines the idea of linking business processes to strategic goals through measures, no details are provided, and as the approach is purely a modeling approach, there are no provisions for gathering data from executing processes.

6.2 Methodology

In terms of methodology steps, the proposed approach uses steps that are common in process improvement methods but it also provides some enhancements. First, unlike business process reengineering, we do not suggest radical and revolutionary improvements in all processes. However, at the same time, the methodology does not suggest to focus on small-grained details that cause the big picture to be lost, which is often the case for statistical-oriented improvement methods like six sigma [26]. Instead, our proposed methodology suggests a spiral approach of improvement. After creating the global model, each subsequent iteration deals with the most important processes. Also, unlike classical improvement methods that have trouble gathering information about processes scattered across an organization [63], our framework provides a data exchange layer capable of obtaining data from different sources including data warehouses and business intelligence systems.

Recent work by Chowdhary et al. [6] led to a model-driven methodology used to create IBM's Business Performance Management. Their methodology shares many steps with ours and is also taking advantage of both data warehouses and event-based and data-based business KPI definitions (with dimensions), with a focus on automation and adaptive management. However, their modeling targets the implementation of metrics and event collection rather than goals and business processes.

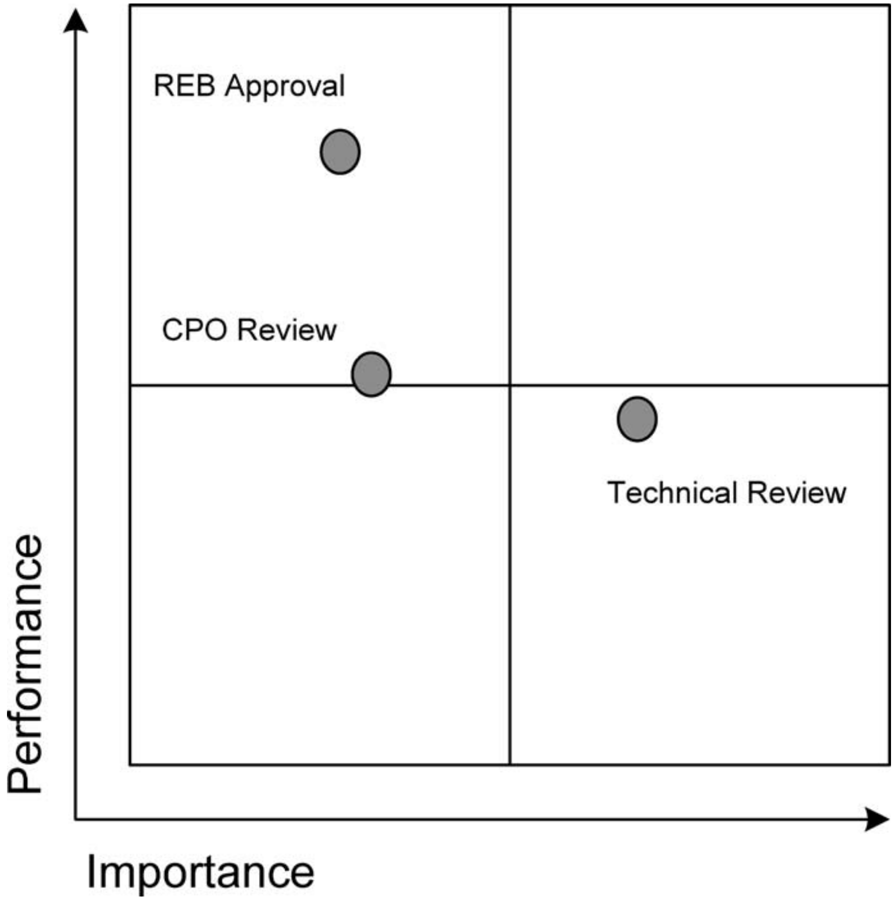


Fig. 27 Updated process portfolio view

Similar to our work, Lapouchnian et al. [33] are also concerned with evaluating alternative ways of structuring business processes (different configurations of a business process) to achieve business goals. They also provide traceability links between goal and business process models. These are accomplished by annotating goal models to indicate control flows and resource dependencies. However, their approach is limited to planning, and does not make provisions for business process monitoring, as they do not have the concept of KPI.

6.3 Integration with tools

Table 10 briefly compares our URN-based BPMS with other major BPMS in the industry. The relevant criteria include support for process modeling, goal modeling, goal evaluation, links between processes and goals, and KPI modeling and evaluation.

The table shows that by providing an integrated system based on URN and jUCMNav, our solution can provide stronger support for modeling and evaluation across

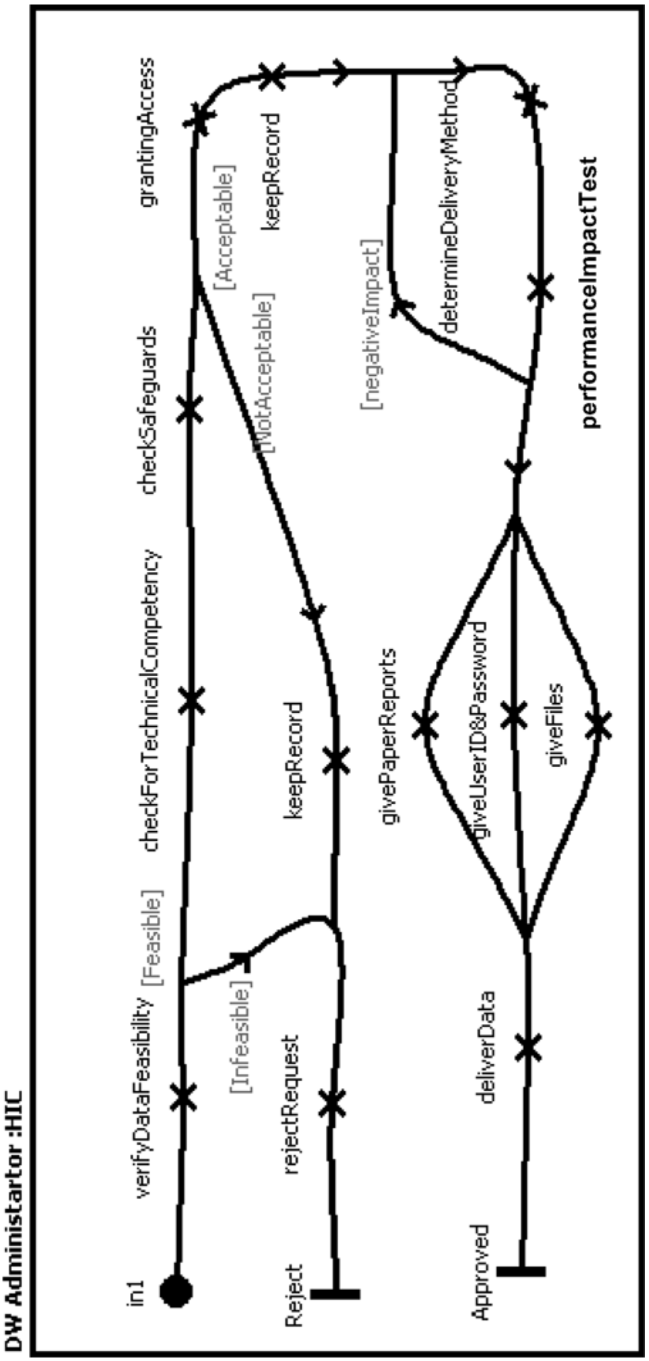


Fig. 28 Updated technical review sub-process

Table 10 Comparison between leading BPMS tools

	Process modeling	Goal modeling	Goal evaluation	Links between goals and processes	KPI modeling and evaluation
The URN-based BPMS	√	√	√	√	√
IBM WebSphere Process Integration [59]	√	×	×	×	–
Intalio BPMS [24]	√	×	×	×	×
Appian Enterprise BPM Suite [3]	√	×	×	×	–
Tibco iProcess Suite [11]	√	–	×	–	–
G360 Enterprise BPM Suite [17]	√	×	×	×	–
Lombardi Teamworks BPMS [38]	√	×	×	×	–
EMC BPM Suite	√	×	×	×	–
Pegasystems SmartBPM Suite	√	×	×	×	–
Savvion BusinessManager Platform	√	×	×	×	–
FileNet BP Manager	√	×	×	×	–
Fujitsu Interstage BPM Suite	√	×	×	×	–
BEA AquaLogic BPM Suite (Fuego)	√	×	×	×	–
webMethods Fabric BP integration platform	√	×	×	×	–
SeeWhy real time BI platform	×	×	×	×	–

Legend: √: Supported, –: Partial supported, ×: Not supported, Covered, Uncovered

processes, KPIs and goals. Moreover, in our solution, URN links connecting business processes and business goals in different hierarchies enable traceability among business processes, business goals and performance models. Note however that many of the other tools provide more detailed descriptions of business processes and many support process automation.

6.4 Experiences with health care business processes

Our methodology was applied in part to several health care business processes at a major teaching hospital in Ontario (Canada). Many positive points have been observed:

- We have documented the process for accessing (confidential) patient information in a data warehouse (the example used to illustrate our methodology in this paper). This was done with a particular emphasis on *compliance*. The analysis of compliance relationships between the organizational processes and goals and provincial privacy legislation [50], supported by a requirements management system, is explored in detail in [15]. This case study demonstrated that changes in processes and laws can also be monitored, and that their links can help maintain compliance in such evolution context. The presence of models also helped reduce the effort required to manually establish compliance and traceability links.
- We have modeled a complex patient *discharge process*, a key process that starts with the very first point of patient entry at the hospital and ends with patient discharge [52]. Within that process, there are issues to be addressed with respect to delays, adverse events and undesirable outcomes. Continuous traceability and monitoring of activities in this context can help reduce variations in practices and detect, measure, and prevent adverse events occurring during healthcare delivery. Our URN model, composed of over 50 UCM and GRL diagrams, helped demonstrate the scalability of the approach but also some limitation in terms of ease of modeling, which led to potential improvements to the notations based on workflow patterns discussed in [43, 44]. We expect many of them to be incorporated in a future release of jUCMNav and of the URN language. A replicated version of the hospital's data warehouse populated with simulated but realistic data was again used for the experiments. The monitoring results were reported through a commercial business intelligence tool (Cognos 8, [8]), demonstrating the feasibility and usefulness of such integration.
- Goals and KPIs were defined for a hospital *infection control* process, again with integration to the Cognos business intelligence tool. The details of the process workflow are not yet modeled. This is ongoing work, and we expect this process to be documented in two different hospitals located in two different Canadian provinces (to highlight commonalities and differences, especially in terms of cultures and legislation).
- We are also currently modeling a *palliative care* process within a Local Health Integration Network, also in Ontario.

We have observed several operational challenges in implementing this approach. The first step is often to get organizational consensus on goals, and identify champions at all levels in the organization. Workflow descriptions and appropriate legislation must then be collected in order to describe business processes as they are and as we would like them to be. Then, we have to obtain the data necessary for measurement. This is particularly difficult when data quality is poor. For example, in our health care context:

- Several different medical codes are used for the same medical diagnostic;
- Different people use the same medical code for different diagnostics;
- Text fields allow for ambiguous and inconsistent input when precise input is necessary in order to quantify results (e.g. prescription amounts);
- Data for computing KPIs is not necessarily collected by existing operational systems and hence can be missing from the data warehouse.

The monitoring part requires the buy-in of IT support specialists, especially as we need to get access to infrastructure elements that cross the boundaries of different business units in the organization. Finally, in order for alignment to become possible, getting the buy-in of upper management contributes positively to the actual implementation and acceptance of potential improvements, not just their identification.

These problems are not unique to health care and can be observed in other business domains. They are not insurmountable but being aware of them in advance will improve the feasibility of business process management projects where our URN-based approach is intended to be used.

7 Conclusions and future work

In this paper, we have defined a goal-oriented process management methodology and framework based on the User Requirements Notation as our modeling language, jUCMNav as the supporting tool for modeling and monitoring processes, and data warehouses and business intelligence systems as our information sources. The semantics of the framework is provided by an extensible meta-model designed to support more capabilities as requirements arise.

The focus of this paper was on the business process monitoring and performance management aspects of the framework (the compliance and conformance aspects being detailed in [15]). First, we introduced the concepts of KPIs and performance models, derived from business goal models, which enable organizations to improve both their processes and goals. Second, we have discussed a data exchange layer that enables the KPIs to be populated with values from operational systems, including data warehouses. Third, a scenario-based technique for performance and impact analysis was proposed and implemented, allowing users to only analyze the part of the processes covered according to UCM scenario definitions. Finally, we presented process portfolio analysis, where we elaborate how a quadrant view of the modeled processes allows one to see the performance of processes and their level of importance for the business. In addition, this allows users to drill down into a process hierarchy and find out the root causes of problems. The methodology and the framework were illustrated and validated with a real business process from a Canadian research hospital.

Our contributions address several problems and limitations observed in existing process improvement methodologies, in Business Process Management Systems, and in the current User Requirements Notation. Table 11 gives a summary of these problems and of the solutions we provided. Most aspects of this approach have been compared with related work throughout the paper.

Table 11 Summary of problems and solutions

Problem	Solution
Since improvement methodologies and process monitoring tools do not utilize a goal modeling language, it is very difficult to visualize the impact of process performance on business goals.	We have suggested a novel process monitoring approach using the User Requirements Notation, a language with the ability to model both goals and business processes.
Using tools traditionally used by management and statistics-oriented methodologies does not allow us to have access to the process-related information dispersed inside and outside the organizations.	We have suggested an approach based on an open layer of data access that can be used to provide access to any kind of information related to the processes and using them for process monitoring purposes.
Methodologies with supporting software have integration issues with different information systems generating related information.	A supporting tool has been developed that uses DW and BI as sources of information and passes the information to jUCMNav by communicating through a layer of web services.
The main URN supporting tool, jUCMNav, cannot support the use of data dispersed across different information systems as source for process monitoring.	
URN does not offer explicit capabilities for defining metrics and indicators to perform process monitoring.	The URN meta-model has been extended to provide the required capabilities for performance monitoring. Additional editing capabilities and views taking advantage of this new information have been added to jUCMNav.
There are few guidelines and methodologies available on how to use goals and scenarios with a set of integrated tools to perform process monitoring.	We suggested a methodology that one can use with the extended URN and supporting tools for business process monitoring.

For future work, we plan to enrich the framework in several different ways. To complete the improvement cycle of our methodology, business process reengineering patterns should be used and detailed guidelines on how to use them in different situations based on monitoring results should be provided. Preliminary work in that direction is suggested in [51, 52]. In addition, the tool support required to recommend patterns to users automatically and to apply these patterns to the appropriate parts of the process should be developed. Moreover, detailed guidelines on how to make the best use of the performance strategy definition capability will be provided. This feature can be used for simulation and validation of the suggested improvements. Furthermore, to provide better process analysis capabilities to end users, advanced features defined in Sects. 4.2 and 4.3 including process portfolio analysis and scenario based performance and impact analysis will soon be implemented and validated.

Another area of focus will be improvements to the current evaluation mechanism by providing suitable GRL propagation algorithms specific to business process monitoring applications. In addition, the KPIValueSet part of the meta-model that currently has four hardcoded values could be made more flexible by allowing users to define their own customized set of boundaries and value sets.

In terms of integration with external tools, two major improvements are required. First, a better integration with the data source layer to allow users to add a new KPI automatically should be provided. Based on our observations, this enhancement could improve the performance and shorten the life cycle time of the suggested methodology. Second, our supporting tool should be integrated with process execution engines to be able to automate the modeled processes and workflow as part of our Business Process Management Framework.

Finally, we could take advantage of recent developments in Aspect-oriented URN [45] to enable the modeling of business goals and processes in an aspect-oriented way. This could potentially allow modelers to compose processes and their related concerns in a more flexible manner.

Acknowledgements This research was supported by the Ontario Research Network on e-Commerce (ORNEC) and by NSERC and CIHR through their joint Collaborative Health Research Projects program. We are grateful to Telelogic and Cognos (now both part of IBM) for providing us with their tools, and to J. Kealey, G. Mussbacher, B. Zhan, and J. Sincennes for their help with the various tools used in this framework.

References

1. Amyot, D. (2003). Introduction to the user requirements notation: Learning by example. *Computer Networks*, *42*(3), 285–301.
2. Amyot, D., Becha, H., Bræk, R., & Rossebø, J. E. Y. (2008). Next generation service engineering. In *ITU-T innovations in NGN kaleidoscope conference* (pp. 195–202), Geneva, Switzerland, May 2008. IEEE CS.
3. Appian (2007). http://www.appian.com/.
4. Bruce Silver Associates (2006). *The 2006 BPMS report: Understanding and evaluating BPM suite*. Published in collaboration with BPMInstitute.org. 2006.
5. Chen, P. (2007). *Goal-oriented business process monitoring: An approach based on user requirement notation combined with business intelligence and Web services*. M.Sc. thesis, Carleton University, Canada, December 2007. http://jucmnav.softwareengineering.ca/twiki/bin/view/UCM/VirLibPChenThesis07.
6. Chowdhary, P. et al. (2006). Model driven development for business performance management. *IBM Systems Journal*, *45*(3), 587–605.
7. Chung, L., Nixon, B. A., Yu, E., & Mylopoulos, J. (2000). *Non-functional requirements in software engineering*. Dordrecht: Kluwer Academic Publishers.
8. Cognos, an IBM Company (2006). *Cognos 8 business intelligence*. http://www.cognos.com/businessintelligence/.
9. Debevoise, T. (2005). *Business process management with a business rules approach*. Business Knowledge Architects, 2005.
10. De la Vara González, J.L. & Sánchez Díaz, J. (2007). Business process-driven requirements engineering: a goal-based approach. In *8th workshop on business process modeling, development, and support (BPMDS'07)*, Trondheim, Norway, June 2007.
11. DiToro, L., & Schaffhauser, D. (2006). *BPM software report: Tibco iProcess suite 10.5*. BPMEnterprise.com, CTQ Media LLC.
12. Dresner, H. (2003). Business activity monitoring. In *Gartner symposium*, Cannes, France, 4–7 November 2003.
13. Enix Consulting (2006). *Issues and best practices for the BPM and SOA journey* (White paper).
14. Ferdian, A. (2001). *Comparison of event-driven process chains and UML activity diagram for denoting business processes*. Masters thesis, Technische Universität Hamburg-Harburg, Germany, April 2001.
15. Ghanavati, S. (2007). *A compliance framework for business processes based on URN*. M.Sc. thesis, University of Ottawa, Canada, May 2007. http://jucmnav.softwareengineering.ca/twiki/bin/view/UCM/VirLibGhanavatiMScThesis.
16. Ghose, A. K., Koliadis, G., Vranesevic, A., Bhuiyan, M., & Krishna, A. (2006). Combining $i*$ and BPMN for business process model lifecycle management. In *LNCS: Vol. 4103. Proc. BPM-2006 workshop on grid and peer-to-peer based workflows* (pp. 416–427). Berlin: Springer.

17. Global 360 (2007). http://www.global360.com.
18. Gong, L. et al. (2005). Deliver an effective and flexible data warehouse solution, Part 2: Develop a warehouse data model. IBM, Jul. 2005.
19. Greenwood, D., & Rimassa, G. (2007). Autonomic goal-oriented business process management. In *Third international conference on autonomic and autonomous systems (ICAS'07)*, Athens, Greece. IEEE CS.
20. Grigori, D. et al. (2004). Business process intelligence. *Computers in Industry*, *53*(3).
21. Heß, H. (2006). From corporate strategy to process performance—what comes after business intelligence? In *Corporate performance management* (pp. 7–29). Berlin: Springer.
22. IBM (2008). *Telelogic DOORS*. http://www.telelogic.com/products/doors/doors/index.cfm.
23. Inmon, W. H. (2005). *Building the data warehouse*. New York: Wiley.
24. Intalio (2007). http://www.intalio.com/.
25. ITU-T (2008). *Recommendation Z. 151 (11/08): User requirements notation (URN)—language definition*. Geneva, Switzerland.
26. Jeston, J., & Nelis, J. (2006). Why six sigma is not complete without BPM. In *Business process management: Practical guidelines to successful implementations*. London: Butterworth-Heinemann.
27. Kealey, J. (2007). *Enhanced use case map analysis and transformation tooling*. M.Sc. thesis, University of Ottawa, Canada, October 2007. http://jucmnav.softwareengineering.ca/twiki/bin/view/UCM/VirLibKealeyThesis07.
28. Keen, M. et al. (2006). *Patterns: SOA foundation—business process management scenario*. IBM Redbooks.
29. Kimball, R., & Ross, M. (2002). *The data warehouse toolkit: The complete guide to dimensional modeling*. New York: Wiley.
30. Krogstie, J. (2005). *EEML2005: Extended enterprise modeling language* (Technical report). Norwegian University of Science and Technology, Norway.
31. Kronz, A. (2006). Managing of process key performance indicators as part of the ARIS methodology. In *Corporate performance management* (pp. 31–44). Berlin: Springer.
32. Küng, P., Hagen, C., Rodel, M., & Seifert, S. (2005). Business process monitoring & measurement in a large bank: Challenges and selected approaches. In *Proc. 16th int. workshop on database and expert systems applications (DEXA'05)* (pp. 955–961), August 2005.
33. Lapouchnian, A., Yu, Y., & Mylopoulos, J. (2007). Requirements-driven design and configuration management of business processes. In *LNCS: Vol. 4714*. *Business process management* (pp. 246–261). Berlin: Springer.
34. Levene, M., & Loizou, G. (2003). Why is the snowflake schema a good data warehouse design? *Information Systems*, *28*(3), 225–240.
35. List, B., & Korherr, B. (2005). A UML 2 profile for business process modeling. In *LNCS: Vol. 3770*. *1st int. workshop on best practices of UML*. Klagenfurt, Austria (pp. 85–96). Berlin: Springer.
36. List, B., & Korherr, B. (2006). An evaluation of conceptual business process modelling languages. In *21st ACM symposium on applied computing (SAC'06)* (pp. 1532–1539), Dijon, France.
37. Liu, L., & Yu, E. (2004). Designing information systems in social context: A goal and scenario modelling approach. *Information Systems*, *29*(2), 187–203.
38. Lombardi Software (2007). http://www.lombardisoftware.com/.
39. Longo, A., & Motta, G. (2006). Design processes for sustainable performances: A model and a method. In *LNCS: Vol. 3812*. *BPM 2005 workshops* (pp. 399–407). Berlin: Springer.
40. Marketos, G., & Theodoridis, Y. (2006). Measuring performance in the retail industry. In *LNCS: Vol. 4103*. *BPM 2006 workshops* (pp. 129–140). Berlin: Springer.
41. Mayer, R., Menzel, C., Painter, M., Witte, P., Blinn, T., & Perakath, B. (2005). *Information integration for concurrent engineering—IDEF3 process description capture method report*. Texas: Knowledge Based Systems Inc.
42. Mili, H., Jaodue, G. B., Lefebvre, E., Tremlay, G., & Petrenko, A. (2003). Business process modeling languages: Sorting through the alphabet soup (TR LATECE). UQAM, 55 pages, November 2003.
43. Mussbacher, G. (2007). Evolving use case maps as a scenario and workflow description language. In *10th workshop of requirement engineering (WER'07)* (pp. 56–67), Toronto, Canada, May 2007.
44. Mussbacher, G., & Amyot, D. (2008). Assessing the applicability of use case maps for business process and workflow description. In *2008 international MCETECH conference on e-technologies* (pp. 219–222), Montréal, Canada, January 2008. IEEE CS.
45. Mussbacher, G., Amyot, D., & Weiss, M. (2007). Visualizing early aspects with use case maps. In *Transactions on aspect-oriented software development III* (pp. 105–143). Berlin: Springer.

46. Nicholls, C. (2006). In search of insight. See Why Software Limited. August 2006.
47. OMG (2008). *Business process modeling notation (BPMN)*, version 1.1, January 2008. http://www.omg.org/spec/BPMN/1.1/.
48. OMG (2008). *Business motivation model (BMM)*, version 1.0, January 2008. http://www.omg.org/spec/BMM/1.0/.
49. OMG (2008). *Business process modeling notation (BPMN) information*. http://www.bpmn.org/, accessed December 7, 2008.
50. PHIPA (2004). *Personal health information protection act*. Government of Ontario, Canada, 2004. http://www.region.peel.on.ca/corpserv/phipa/index.htm.
51. Pourshahid, A. (2008). *A URN-based methodology for business process monitoring*. M.Sc. thesis, University of Ottawa, Canada, March 2008. http://jucmnav.softwareengineering.ca/twiki/bin/view/UCM/VirLibPourshahidMScThesis.
52. Pourshahid, A., Chen, P., Amyot, D., Weiss, M., & Forster, A. J. (2007). Business process monitoring and alignment: An approach based on the user requirements notation and business intelligence tools. In *10th workshop of requirement engineering (WER'07)* (pp. 80–91), Toronto, Canada, May 2007.
53. Reijers, H. A. (2005). *Process-aware information systems*. New York: Wiley.
54. Roy, J.-F., Kealey, J., & Amyot, D. (2006). Towards integrated tool support for the user requirements notation. In *LNCS : Vol. 4320. SAM 2006: Fifth workshop on system analysis and modelling* (pp. 198–215). Berlin: Springer. http://jucmnav.softwareengineering.ca/.
55. Roy, J.-F. (2007). *Requirement engineering with URN: Integrating goals and scenarios*. M.Sc. thesis, University of Ottawa, Canada, March 2007. http://jucmnav.softwareengineering.ca/twiki/bin/view/UCM/VirLibRoyMScThesis.
56. Rudden, J. (2007). Making the case for BPM—a benefits checklist. In *BPTrends*, January 2007.
57. van der Aalst, W. M. P., & ter. Hofstede, A. H. M. (2005). YAWL: Yet another workflow language. *Information Systems*, *30*(4), 245–275.
58. Vonderheide-Liem, D. N., & Pate, B. (2004). *Applying quality methodologies to improve healthcare: Six sigma, lean thinking, balanced scorecard, and more*. HCPro, Inc., November 2004.
59. Wahli, U., Avula, W., Macleod, H., Saeed, M., & Vinther, A. (2007). *Business process management: Modeling through monitoring using WebSphere V6.0.2 products*. IBM Redbooks.
60. Weiss, M., & Amyot, D. (2005). Designing and evolving business models with URN. In *Montreal Conf. on eTechnologies (MCeTech)* (pp. 149–162). Montreal, Canada, Jan. 2005.
61. Weiss, M., & Amyot, D. (2006). Business process modeling with URN. *International Journal of E-Business Research*, *1*(3), 63–90.
62. White, S. A. (2004). *Introduction to BPMN*. May 2004. Available online at http://www.bpmn.org/Documents/Introduction%20to%20BPMN.pdf.
63. Williams, B. (2007). *BPM: The next stage for continuous process improvement*. Software AG, webMethods.
64. Yu, E. (1997). Towards modelling and reasoning support for early-phase requirements engineering. In *3rd IEEE int. symp. on requirements engineering* (pp. 226–235), Washington, USA. IEEE CS.
65. Zhan, B. (2007). *An integrated quality assurance framework for enterprise performance management systems*. M.Sc. thesis, University of Ottawa, Canada.

Alireza Pourshahid received his M.Sc. degree in E-Business Technologies from the University of Ottawa in 2008 and is now working at IBM. He is also a Ph.D. student at the University of Ottawa. His main research interests are Business Process and Performance Management, Process Modeling, Trust Modeling, and Software Development Methodologies.

Daniel Amyot is Associate Professor at the University of Ottawa, which he joined in 2002 after working for Mitel Networks as a senior researcher in software engineering. His research interests include scenario-based software engineering, requirements engineering, business process modeling, aspect-oriented modeling, and feature interactions in emerging applications. Daniel is Rapporteur for requirements languages at the International Telecommunication Union, where he leads the development of the User Requirements Notation. He has a Ph.D. and a M.Sc. from the University of Ottawa (2001 and 1994), as well as a B.Sc. from Laval University (1992).

Liam Peyton Ph.D., P.Eng., is the principal investigator for the Intelligent Data Warehouse laboratory and Associate Professor at the University of Ottawa which he joined in 2002 after spending 10 years as an industry consultant specializing in business process automation, performance management, and software development methodologies. His current research focus is the securing, monitoring and enabling of data sharing within business to business networks based on model-driven, service oriented architecture in compliance with government regulations. He has degrees from Aalborg Universitet (Ph.D. 1996), Stanford University (M.Sc. 1989), and McGill University (B.Sc. 1984).

Sepideh Ghanavati is a Ph.D. student and research assistant at the University of Ottawa. Her research interests include requirements engineering, business process modeling, and privacy. She has an M.Sc. in Systems Science from the University of Ottawa (2007) and a B.Sc. in Industrial Engineering from Amirkabir University of Technology (2003).

Pengfei Chen received his M.Sc. degree in Computer Science from Carleton University in 2008 and is now working at Research In Motion. His main research interests include goal-oriented business process monitoring and modeling.

Michael Weiss is Associate Professor in the Department of Systems and Computer Engineering at Carleton University in Ottawa and a member of the Technology Innovation Management (TIM) program. His research interests include open source ecosystems, service-oriented architectures, mashups/Web 2.0, business process modeling, product architecture and design, and pattern languages. Between 2000 and 2007, he was a professor of Computer Science at Carleton University. From 1994 to 1999, he was a member of the Strategic Technology group at Mitel. Michael obtained his Ph.D. (Dr.rer.nat.) from the University of Mannheim in 1993. He is author of over 70 peer-reviewed publications in leading journals and conferences.

Alan J. Forster is a general internist, Co-Director of the Ottawa Hospital Center for Patient Safety, and Scientist with the Ontario Ministry of Health and Long Term Care. He is also an Associate Professor of Medicine at the University of Ottawa and Scientist in the Clinical Epidemiology Program at the Ottawa Health Research Institute, where his research focuses on patient safety and quality improvement. He has performed seminal work evaluating the incidence of adverse events following discharge from hospital. He is also leading the development of a hospital data warehouse, which serves as a supporting infrastructure for a research program in patient safety and quality of care. Dr. Forster received his M.D. (1994) as well as his M.Sc. in Epidemiology (1998) from the University of Ottawa.