Towards a generic Intelligent Monitoring Platform for business processes.

Stelios Kapetanakis, Miltos Petridis, Jixin Ma and Brian Knight

School of Computing and Mathematical Sciences, University of Greenwich, Maritime Greenwich Campus, Old Royal Naval College, Park Row, Greenwich, London SE10 9LS, UK
(s.kapetanakis, m.petridis, j.ma, b.knight}@gre.ac.uk

Abstract. Modern business processes have a key role in operating, controlling, and managing large organizations. The management and monitoring of business processes can be problematic since their structural complexity and large volume of involved data makes efficient monitoring and decision making hard. This paper presents a platform for an intelligent monitoring business processes platform. An approach to using CBR for the reuse of knowledge to the monitoring of business workflows is presented. The CBR-WIMS platform and its architecture is presented. An overview of the evaluation of the approach and the platform is presented as applied to a real business workflow case study. This shows that CBR-WIMS can assist business workflow managers in the monitoring and intelligent decision support of real business workflows.

Keywords: Case Based Reasoning, Business Workflows, Temporal Reasoning, Graph Similarity.

1 Introduction

In modern world we often come across Business Processes and an increasing demand for them. Most organisations are using business processes for a variety of purposes: formalising their product assembly procedures, defining their internal hierarchy structure; formulating their relationships with various internal and external stakeholders [1]. Business processes offer the opportunity to observe an organisation's structure, at different levels of abstraction. Levels start from top, in such way you can have an overview of the whole operation to middle or bottom level where a manager can focus on sections of mighty operational significance.

Business processes can be represented as a set of activities with temporal relationships and constraints imposed on them. Over the last years software systems are used increasingly to manage and automate the operation of business processes. This caused a need for standard formalism to represent business processes. As a consequence, new standards have emerged to fulfil this need. The Business Process Modelling Notation (BPMN) developed by the Business Process Management Initiative (BPMI) and Object Management Group (OMG) provides a standard for the graphical representation of workflow based business processes [2]. Standards produced for business process representation aim to cover the definition, orchestration and choreography of business process. Over the last few years, a number of standards have emerged and are widely accepted and supported by mainly Service Oriented Architecture (SOA) based enterprise technologies and systems. The OASIS Business Process Execution Language (BPEL), short for Web Services BPEL (WS-BPEL) is a key orchestration technology [3]. The Workflow Management Coalition (WfMC) backed XML Process Definition Language (XPDL) is a format standardised to interchange Business Process definitions between different workflow products and systems [4].

The real challenge that a Business Process faces is the effective monitoring and interpretation of the events that take place. Usually when a business process runs, every event that takes place leaves a trace which can be represented in various ways. This could just be in the form (Event Name, Time Stamp) but it can also involve a great deal of contextual information, such as (Event Name, Event Nature, Origin, Actors Involved, Actions generated, Communications generated, Time Stamp). The nature and frequency of events creation differs from system to system. We can have hundreds of events fired simultaneously in some systems during normal operation whereas just there could be just a few in other systems may need raising an automatic alert.

Systems with a small, finite number of different states are assumed to be easier to monitoring by a manager. However, when a system expands and becomes more complex, the time required on monitoring is considerably increased. The difficulty in monitoring a workflow is raised even more when uncertainty is present in some actions. Business processes when in use usually keep log records of events that happened during the operation of the process. However, business processes that involve human resources cannot log information about actions that took place outside the scope of the system (such as unofficial chats or phone calls). In such cases the person monitoring the process will be probably confused by seemingly unjustified actions in the system with no obvious reasoning. An interesting phenomenon is also when some strictly defined processes have to be overridden in order to deal with unanticipated situations. In such cases the monitoring cannot be efficient since an action occurred out of the current context and the system cannot identify the correlation with existing logged actions.

Working towards addressing these issues, this paper presents a CBR approach to intelligent monitoring of business processes based on past knowledge experience. The system cooperates with an existing business process and identifies all associated information available to them. Furthermore it extracts information both from the definition of the business process and from the data that were produced during its operation, typically in the form of event logs). From this information the system can attempt to establish what may have happened and notify appropriately the business workflow monitoring manager.

This paper presents CBR-WIMS, a Case-Based Reasoning (CBR) based platform for the intelligent monitoring of business workflows. Section 2 gives some background on business workflow monitoring and section 3 presents the business process case study that is used as an experimental vehicle for this research. Section 4 presents the CBR-WIMS architecture and adaptor layer used and section 5 shows the application and use of CBR-WIMS on the exams case study and presents an overview of the evaluation done so far showing the efficiency of the CBR engine and the usability of the system. The Conclusion summarises the work done and indicates future work currently planned.

2 Approaches to the Monitoring Business Process Workflows

There are various approaches to the problem of monitoring a business process. A business process is tightly dependent on its workflow representation. When monitoring information about a business process, the current workflow state must be analysed and compared using domain/model knowledge and knowledge gained from past experience. As problems usually recur, if similar cases are found this can provide the context for reasoning about the workflow or, if no such precedent can be found, new knowledge can be derived in the form of a new case that can be stored in the system for later use. This approach matches the behaviour and process of Case-Based Reasoning (CBR) systems. The standard CBR process cycle follows the Retrieve, Reuse, Revise, Retain model [5]. CBR based systems can be used for this purpose. Literature shows several examples of the effective use of CBR to the management of business workflows. An approach to reuse and adaptation of workflows was proposed by Minor et al [6] where the workflows were represented in terms of graphs and structural similarity measures were applied. Kyong Joo Oh and Tae Yoon Kim [7] have proposed CBR for financial market monitoring and examine whether they can build efficiently the daily financial condition indicator. Dijkman et al[9] have investigated algorithms for defining similarities between business processes focused on tasks and control flow relationships between tasks. Van der Aalst et al [10] compare process models based on observed behaviour in the context of Petri nets. The definition of similarity measures for structured representations of cases in CBR has been proposed [11] and applied to many real life applications requiring reuse of domain knowledge associated with rich structure based cases [12],[13].

Although CBR seems to be an effective way of monitoring business processes, there is lack of a generic platform which could be abstract enough to host monitoring for an existing business processes and adapt its environment according to the investigated process's needs. An interesting approach which tries to generalize towards implementation of processes using Case Base Reasoning is jColibri [14]; an open-source CBR framework towards integrated applications that specific case knowledge is needed and contain models of general domain knowledge. Another worth mentionable approach is myCBR [15], also open – source CBR tool for rapid prototyping of CBR applications and more specialized on case-based product recommender systems. Both tools work well towards CBR modelling of an application but do not offer the possibility of working with business processes defined in terms of a workflow and deal with uncertainty in both definition and operational data.

A CBR approach for the intelligent monitoring of business process workflows has been proposed and has shown able to monitor effectively real business workflows when compared to human domain experts[8],[16]. This approach can deal effectively with the workflow monitoring problem if similarity measures have been defined and known problems from the past have been used in order to form a knowledge case base. Cases based on business process's attributes (events, actions and their temporal relationship) are being represented in terms of a simple graph which is used for estimating similarity.

3 The Exam Moderation Case Study Business Process

In order to evaluate the approach proposed in this research, we used the University of Greenwich, School of Computing and Mathematical Science exam moderation system. This is an automated web enabled secure system that allows various actors to interact with the system as well as among them. These actors can be course (module) coordinators, course moderators, exam drafters (typically senior managers), admin staff and external examiners and can upload, modify, approve and lock student exam papers. The system automates the whole process and provides an audit trail of events generated by workflow stakeholders and the system. The system orchestrates a formal process made up of workflows. The process can be defined and displayed formally in terms of a UML activity diagram (**Fig. 1**). The system tracks most workflow actions in terms of timed events. Most of these generate targeted email communications to workflow stakeholders, some for information and others requiring specific further actions from these stakeholders.

For example, the action of a new exam version upload from a course coordinator is notified to the moderator, drafter and admin staff. This can prompt the moderator to approve the uploaded version or upload a new version. However, the coordinator can also upload a new version and admin staff may also decide to format the uploaded version and upload it as a newer version. The system captures all versions, workflow actions, emails sent and there is a facility to record free form comments to document versions and/or workflow actions.

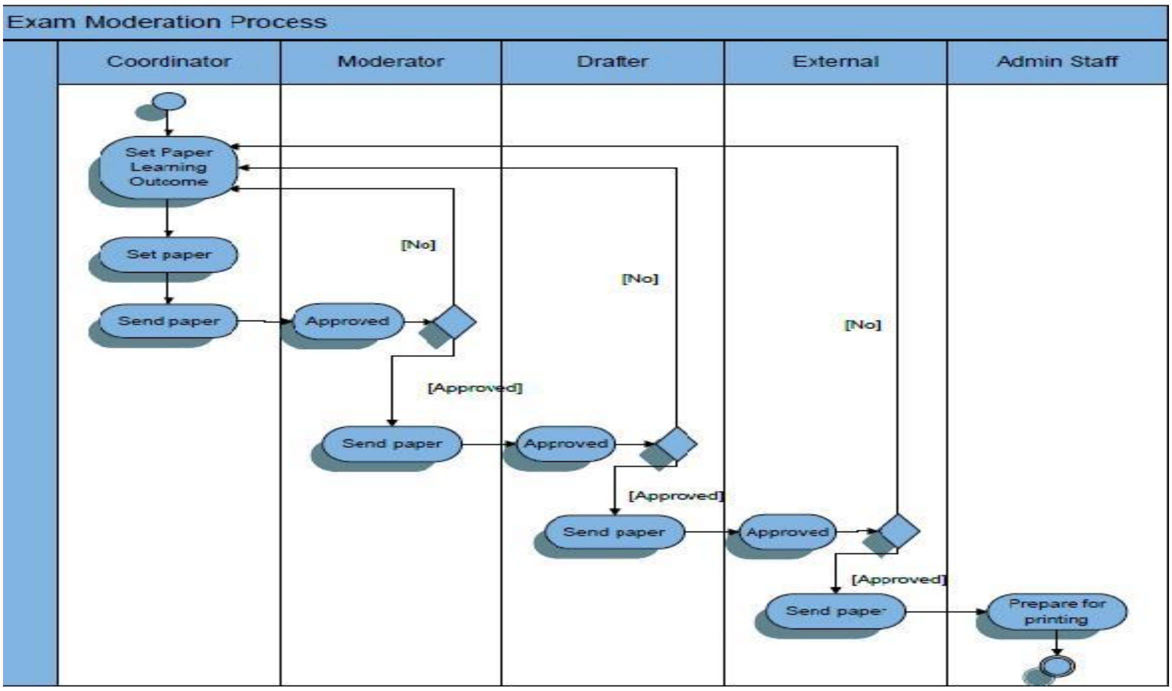

Fig. 1. The exam moderation process activities and workflows (simplified)

4 The CBR-WIMS Platform

CBR-WIMS is a generic platform for intelligent monitoring of Business Processes' Workflows. The idea behind its implementation is to have a flexible and robust system which is able first to understand/adapt to the business process workflow, identify rules and correlation among data, plus actors' roles in a given business process workflow.

4.1 Event trace similarity

A monitoring tool should have a way of measuring similarity efficiently. Measuring similarity among business processes could be defined if the processes are represented in terms of a graph. Then similarity measures could be applied using an exhaustive graph similarity search algorithm based on Maximum Common Subgraph [13]. Events extracted from investigated workflow event log could be represented using a general time theory, based on intervals [17]. For the investigated business process workflow the temporal relationships required have been reduced from the ones proposed by Allen [18] to the "meets" relationship. This approach is

4.2 The CBR-WIMS Architecture

CBR-WIMS is based on agile component oriented architecture. The overall architecture can be seen in **Fig. 3** below. The system contains a core API; we will refer to that as kernel from now on, which can be regarded as the core of the system. The kernel can serve to a number of different roles. The first role of the kernel is to identify the definition of external business process components. In order for this to be done effectively the kernel follows a set of procedural identification rules. This set is internally pre-defined but offers the possibility to be updated. This set's mode can be *special request mode* where the system makes an exception just for this particular input or *learn mode* where the system is always on an alert, absorb state. The latter is necessary if the system has to adapt to a new environment where new components are likely to arrive. A third option is also available (*strict mode*) where system expects only components of already known definition.

A business process in CBR-WIMS can be defined using a BPMN or XPDL format. In this way tasks, actors, their connections pre and post prerequisites plus any other constraint can be depicted graphically. CBR-WIMS offers the option to incorporate the business process representation in one of the above formats. The system extracts information contained inside the represented model and "sketches" the formal business process graph. This graph will be the starting point for successive iterations where each of them will lead to a new rule creation. In this way the system can generate all rules and constraints associated with the investigated business process. If the imported model is missing some information additional rules can be added in a manual way.

Part of a component's identification in the system is the simulation of its behavioural needs. Since CBR-WIMS is a generic API, it expects from any component to share information in order to proceed in an accurate, efficient way. A number of checks have to be done before the engine can accept a component. This is practically translated as a number of preliminary communication exchanges when a component requests a service from CBR-WIMS. The kernel's role is to estimate the number of necessary checks, initiate communication transactions, process results and categorise the component and its needs. After the successful completion of this stage, the system can initiate communications with the investigated component.

After the preliminary investigation of the business process component by the engine, the engine knows the schema of component and can proceed requesting more information about the various inner sections and subsections of the second. If a component's response to the preliminary engine inquiries is appraised successfully, the engine marks the business component as safe to proceed with and can establish on request imminent monitoring hooks. Otherwise, the engine will try to estimate up to which extent the communication is safe and will gives portioned permissions to components requests accordingly. The engine always tries to simulate at least once an action that might take place with an investigated component. If the simulation fails it will not repeat it or agree to a component's request with same content. The engine's priority is the enforcement of the component's data integrity and pre-verification helps with achieving this objective. On any abnormal request or transaction, the engine always tries to ensure no data loss and stability from the component's perspective.

A vital part of engine's architecture is the incorporated CBR component. The CBR component contains methods fundamental for the definition, orchestration and handling of the CBR process. This component works with a set of rules which mainly have the role of an adaptor. The adaptor's role is to adapt existing CBR component and actually create a new CBR layer for the imported business process. The produced new layer combines functionality of the core CBR component with the history log from the imported business process. However, history log cannot be used directly since its format is usually unknown. In order to make it readable to the CBR component is has to be parsed via the Parser Component whose functionality is beyond the scope of this paper. The read log will be separated into appropriate isolated cases that will serve as past cases history. This is shown in **Fig. 2** below that illustrates the structure of the adaptor and the way it adapts to an imported business workflow.

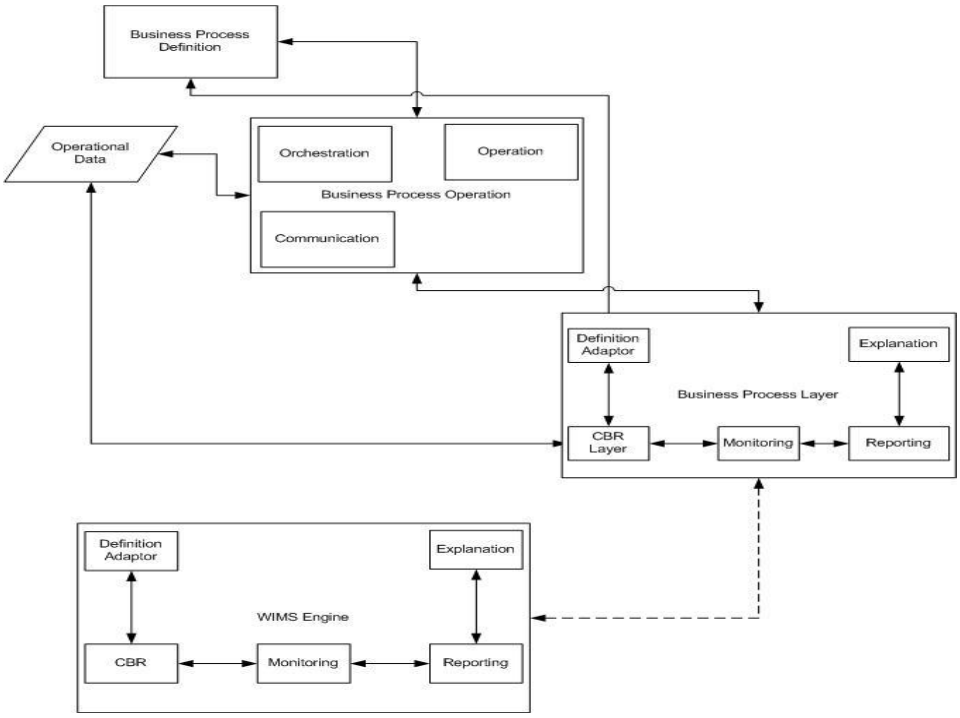

Fig. 2. Simplified WIMS layer adapted to imported business process

The CBR component is heavily used by the monitoring component [8] which can retrieve useful experience about problems occurred in the past. Monitoring component scans recursively current operations - actions of a workflow and attempts to identify whether the existing actions sequence is part of a previously “blacklisted”/“whitelisted” pattern or a “greylisted” pattern. In the case of any pattern identification, extracted information is being transferred to both the Results and Explanation Components (**Fig. 3**).

The CBR-WIMS engine contains a variety of procedures to handle business process data and extract useful information from them. Its methods used to estimate case similarity, workflow coherence and correlation are combined to produce the final judgment on current patterns. In order to present the analysis made effectively a Results Component is being provided which offers options for quick view of current workflow actions and the system’s view on whether the action(s) are expected or raise worries. A manager can resort to this view in order to identify on the spot what the system’s state indicates. In complex cases, especially when faulty pattern is detected, the manager can resort to the engine’s Explanation Component, which is delegated by the engine to provide information on how the engine was convinced to judge situation as faulty. The identified past patterns which have been found to have high similarity with the requested case are being provided to the Explanation component as well as addressed environment information different from one workflow to another.

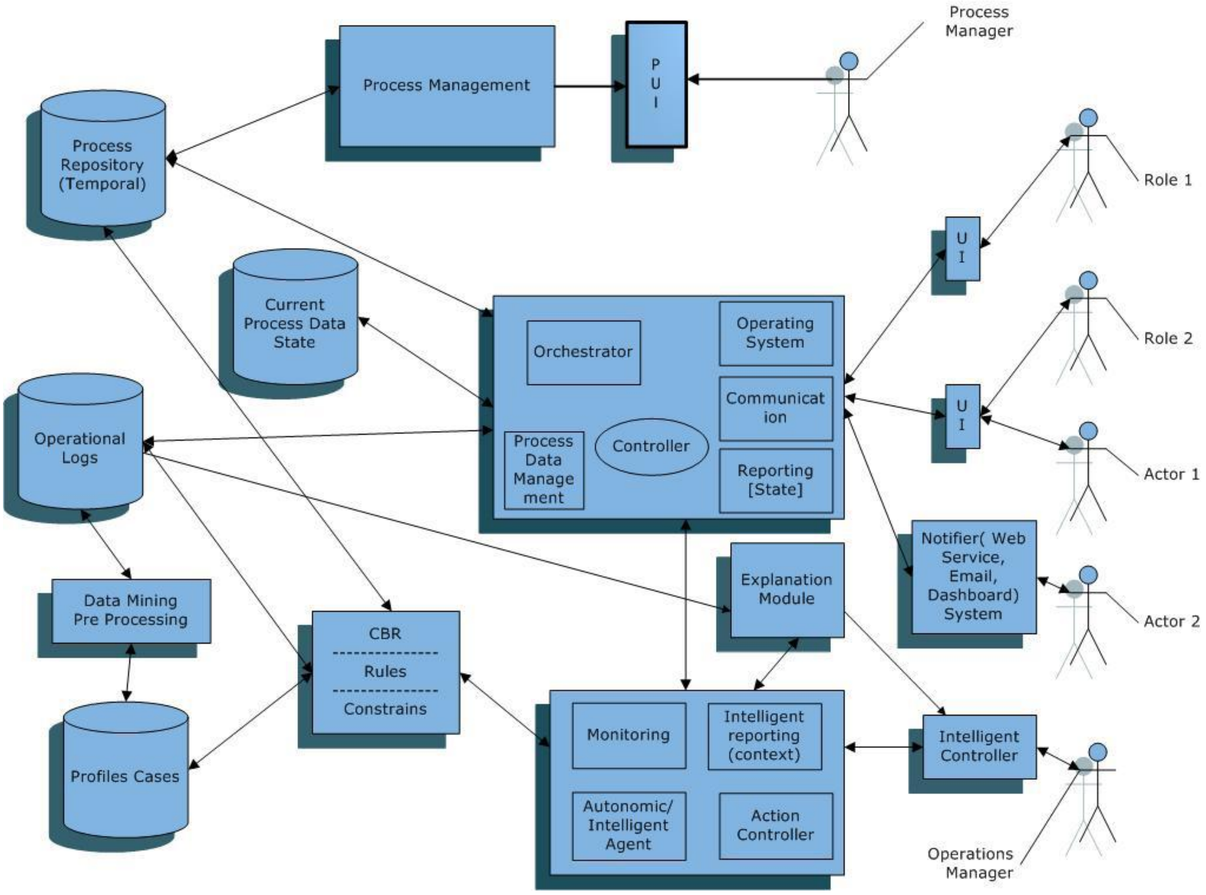

Fig. 3. The WIMS CBR overall architecture

5 Applying the CBR-WIMS platform and selected experiments for evaluation

In order to evaluate whether the proposed system can monitor effectively business processes a set of experiments with scalable complexity have been conducted. As a preliminary test the system was called to prove its effectiveness with simulated workflow data [8]. Following encouraging results from this, further evaluation with real data from the exams moderation process showed that the system could fai well when compared to a human expert workflow manager[21].

Fig. 4 taken from this series of evaluation experiments shows that CBR-WIMS can classify correctly problems occurring in a real workflow, even when only using a subset of available event trace information. This evaluation shows that in most cases the system predicts well the cases where there are substantial problems currently with the exam moderation process (C). It also generally predicts well the cases where the process has gone smoothly (B). The cases where some problems occurred at some past point of the process (A) are less well predicted, although it is evident that "filtering" of similar consecutive events (smoothing) and removing irrelevant admin and reporting events improves the performance of the classification process. This is especially the case when using the MCSG similarity measure. The details of that evaluation and full analysis of the results can be seen in Kapetanakis et al [21]

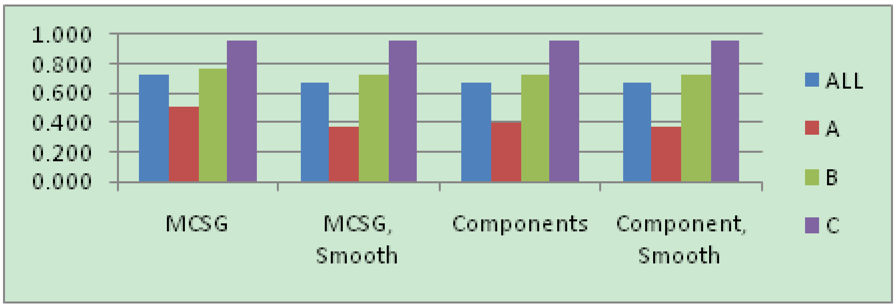

Fig. 4. Results for 3NN with no admin and reporting events filters applied.

However, a more important aspect of the use and evaluation of CBR-WIMS is related to the ease of incorporation and integration of new workflow processes and the ability to adapt to changes in the business processes. This is very important as in most cases, business processes change and evolve to suit the ever changing needs of an organisation.

Furthermore, the evaluation has shown that CBR-WIMS can provide a useful tool for managers to "drill down" into the workflow operation data and understand any issues that arise. The explanation module in WIMS-CBR has proven to be particularly useful to workflow managers as it does not only provide a warning of problems, but it can provide the context and further information that can be used to make an informed action. The table below (table 1) summarises the results of the simple evaluation of the explanation capabilities of WIMS-CBR. The experts replied to the questions using a scale of 1 (disagree) to 5 (agree strongly). The results were averaged over the 20 target cases [22].

	WIMS-CBR no explanation	WIMS-CBR with explanation
Correct classification is clear	3.2	4.2
Similarity is obvious to the 3NN	2.8	3.9
Advice clarity	3.3	4.5

Table 1. Evaluation of the explanation and advice

Fig. 5 below shows the CBR-WIMS UI module that allows users to "drill down" into the workflow execution logs and identify particular patterns of workflow operation that are flagged as problematic. The user can compare these to real past cases in the case base. The similarity measures are shown, allowing the user to see why a particular workflow problem diagnosis has been made by the system. By examining the relevant parts of the matched cases and browsing all available contextual information, the user can see more clearly any issues identified. The workflow manager can then take appropriate action enabled from a clearer view of the workflow process that is being monitored.

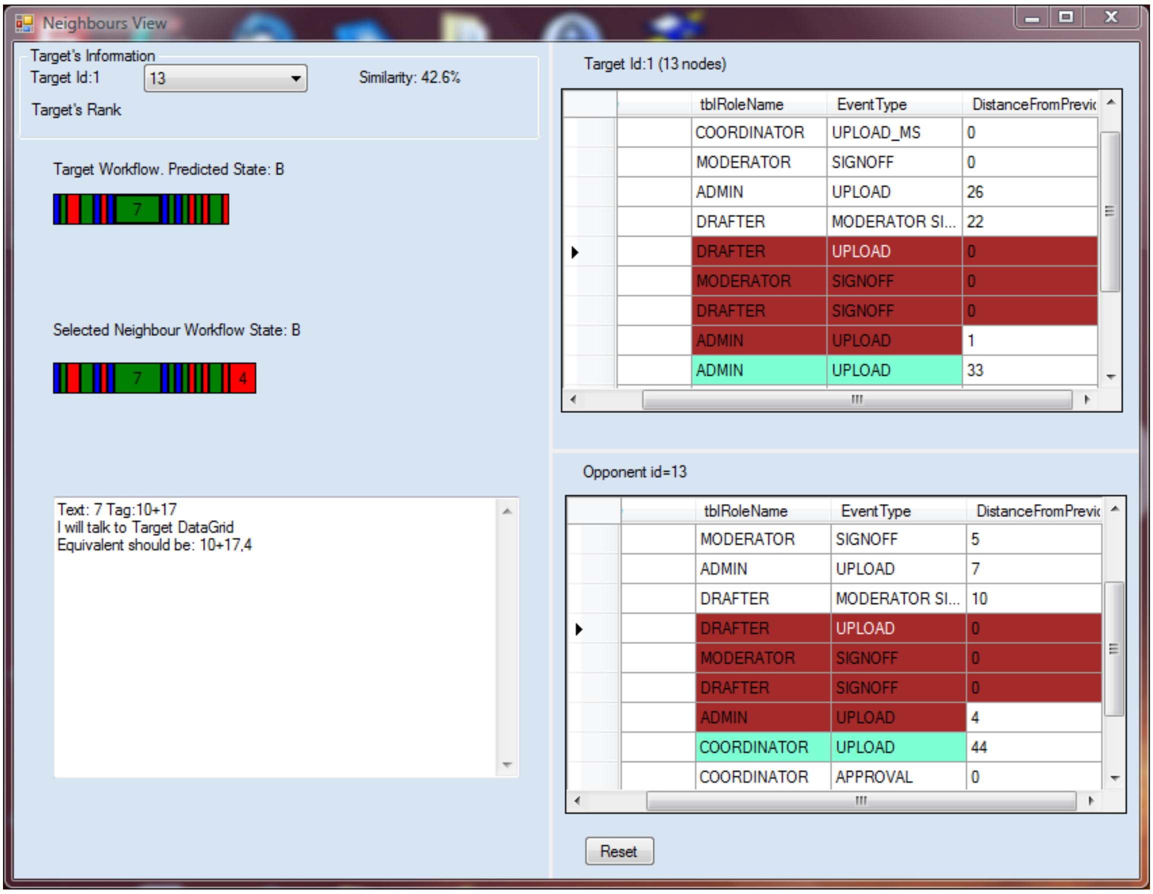

Fig. 5. The Similarity investigation screen in CBR-WIMS

6. Conclusion

This paper discusses an approach towards intelligent monitoring of business processes workflows. The CBR-WIMS platform has been developed, which has shown that it can monitor workflows efficiently when compared to human business workflow management experts. Key advantage of proposed system is its ability to adapt and integrate itself into a new and evolving business process in a non-intrusive way. The CBR-WIMS architecture is generic to be able to deal with different business process specifications and implementations but can specialize on demand by adapting and retaining workflow domain specific knowledge in the form of a case-base and observed patterns. This has shown to be a useful and usable tool for human workflow management experts specialising on the monitoring of workflows. Future work will concentrate in more depth integration of the engine to business processes as well as investigate on how the suggested platform can work efficiently according a variety of different workflows. Finally, further tests are planned to evaluate the ability of CBR-WIMS to adapt to changing business processes with minimum loss of past useful experience.

References

1. Gil Regev and Alain Wegmann, Regulation Based Linking of Strategic Goals and Business Processes, Workshop on Goal-Oriented Business Process Modeling, London, September 2002
2. Business Process Management Initiative (BPMI): BPMN 1.1: OMG Specification, February, 2008, http://www.bpmn.org/, accessed October 2010.
3. OASIS: BPEL, The Web Services Business Process Execution Language Version 2.0. http://www.oasis-open.org/apps/org/workgroup/wsbpel, May 2006.
4. Workflow Management Coalition (WfMC): XPDL 2.1 Complete Specification (Updated Oct 10, 2008), http://www.wfmc.org/xpdl.html, accessed October 2010.

5. A. Aamodt and E. Plaza, Case-based reasoning; Foundational issues, methodological variations, and system approaches. AI Communications, vol. 7, no. 1, pp. 39-59, 1994.
6. Minor, M., Tartakovski, A. and Bergmann, R.: Representation and Structure-Based Similarity Assessment for Agile Workflows, in Weber, R., O. and Richter, M., M.(Eds) CBR Research and Development, Proceedings of the 7th international conference on Case-Based Reasoning, ICCBR 2007, Belfast, NI, UK, August 2007, LNAI 4626, pp 224-238, Springer-Verlag, (2007)
7. Kyong Joo Oh, Tae Yoon Kim, Financial market monitoring by case-based reasoning, Expert Systems with Applications, Volume 32, Issue 3, Pages 789-800, April 2007
8. Kapetanakis, S., Petridis, M., Ma, J., Bacon, L. "CBR-WIMS, an Architecture for the Intelligent Monitoring and Diagnosis of Workflows" in proceedings of the 14th UK CBR workshop, Peterhouse, December 2009,(Ed M. Petridis), CMS press, pp. 67-78
9. Dijkman, R.M., Dumas, M., Garcia-Banuelos, L. Graph matching algorithms for business process model similarity search. In U. Dayal, J. Eder (Eds.), Proc. of the 7th Int. conference on business process management. (LNCS, Vol. 5701, pp. 48-63). Berlin: Springer. (2009)
10. Van der Aalst, W., Alves de Medeiros, A. K., Weijters, A : Process Equivalence: Comparing two Process Models Based on Observed Behavior, In Proc. Of BPM 2006, vol 4102 of LNCS, pp 129-144, Springer, (2006)
11. Bunke, H., Messmer, B.T.: Similarity Measures for Structured Representations. In: Wess, S., Richter, M., Althoff, K.-D. (eds) Topics is Case-Based Reasoning. LNCS, vol. 837, pp 106-118, Springer, Heidelberg (1994)
12. Mileman, T., Knight, B., Petridis, M., Cowell, D., Ewer, J. (2002): Case-Based Retrieval of 3-D shapes for the design of metal castings in Journal of Intelligent Manufacturing, Kluwer. 13(1): 39-45; Feb 2002.
13. Wolf, M., Petridis, M.: Measuring Similarity of Software Designs using Graph Matching for CBR, In workshop proceedings of AISEW 2008 at ECAI 2008, Patras, Greece (2008)
14. Juan A. Recio-García, Antonio A. Sánchez-Ruiz, Belén Díaz-Agudo and Pedro A. González-Calero, jCOLIBRI 1.0 in a nutshell. A software tool for designing CBR systems, in: Proccedings of the 10th UK Workshop on Case Based Reasoning, CMS Press,University of Greenwich, 2005
15. Stahl, A., Roth-Berghofer, Th.: Rapid prototyping of CBR Applications with the Open Source Tool myCBR. Künstliche Intelligenz, 23(1):34–37, March 2009.
16. Kapetanakis, S., Petridis, M., Ma, J., Bacon, L. : Workflow Monitoring and Diagnosis Using Case-Based Reasoning on Incomplete Temporal Log Data, in Workshop proceedings of the 8th International Conference on Case Based Reasoning, Seattle, USA, (2009)
17. Ma, J., Knight, B.: A General Temporal Theory, the Computer Journal, 37(2), 114-123 (1994).
18. Allen, J., Hayes, P.: Moments and Points in an Interval-based Temporal-based Logic, Computational Intelligence, 5, 225-238 (1989).
19. Ma, J., Knight, B.: A Framework for Historical Case-Based Reasoning, in K.D. Ashley and D.G. Bridge (Eds.): ICCBR 2003, LNAI 2689, pp. 246–260 (2003)
20. Ma, J., Knight, B., Petridis, M.: Deriving Explanations from Partial Temporal Information, in workshop proceedings of ExACT-08 at ECAI 2008, Patras, Greece (2008)
21. Kapetanakis, S., Petridis, M., Knight, B., Ma, J.,Bacon, L. : A Case Based Reasoning Approach for the Monitoring of Business Workflows, 18th International Conference on Case-Based Reasoning, ICCBR 2010, Alessandria, Italy, LNAI (2010)
22. Kapetanakis, S., Petridis, M., Knight, B., Ma, J.,Bacon, L.: Providing Explanations for the Intelligent Monitoring of Business Workflows Using Case-Based Reasoning, in workshop proceedings of ExACT-10 at ECAI 2010, Lisbon, Portugal (2010)