Scaling similarity-based retrieval of semantic workflows

Ralph Bergmann, Mirjam Minor, Mohd. Siblee Islam, Pol Schumacher, and Alexander Stromer

University of Trier - Department of Business Information Systems II
D-54286 Trier, Germany
[bergmann|minor|islam|pol.schumacher|s4alstro]@uni-trier.de

Abstract. This paper presents an approach for scaling the retrieval of semantic workflow cases. Similarity-based graph-matching approaches have been used in our previous work for the retrieval of semantic workflows. However, their high computational complexity makes it difficult to scale the approach to case bases with a size of more than a few hundred cases. However, many application areas of semantic workflows like scientific workflows or cookery workflows involve a large amount of semantic workflows to be reused. We propose a novel two-step retrieval method for workflows, inspired by the MAC/FAC (many are called, but few are chosen) approach proposed by Forbus et al. An additional computationally efficient retrieval step (MAC stage) is introduced prior to the graph-based retrieval (FAC stage) to perform a pre-selection of potentially relevant cases. It is based on a feature representation of the workflows automatically derived from the original graph-based representation. In the paper, we briefly introduce our previous work on the semantic workflow retrieval and then we describe the pre-selection step in more detail. An evaluation with case bases from the cooking domain has been performed. It demonstrates scalability towards case bases of up to 15000 cases.

1 Introduction

In the past few years, Case Base Reasoning (CBR) systems have significantly improved their ability to deal with large numbers of cases. Indexing techniques like decision trees [1, 2], kd-trees [3], or case retrieval nets [4] are often applied for a large case base to achieve retrieval results within milliseconds. However, if the representation of cases is complex, the retrieval time can be very high, because a complex representation requires complex similarity measures, which are computationally expensive. Workflows require a complex case representation for retrieval. Bergmann and Gil [5] propose a graph-based approach to represent and retrieve workflow cases. The graph-based retrieval is computationally expensive as the similarity computation involves a kind of graph matching. Current experiments have shown that it works sufficiently fast only for cases bases containing less than 200 cases. Today, websites are a source of procedural knowledge, which can be represented as workflows [6]. We can create a large case base using these workflows for CBR applications. The graph-based retrieval is approaching its limits with respect to computational complexity. However, indexing is hard to apply to the graph-based retrieval as discussed in [5], because sets and mapping functions are used in many similarity functions.

As a consequence, we developed a novel, two-step retrieval for workflows inspired by the MAC/FAC ("Many are called, but few are chosen") approach proposed by Forbus, Gentner and Law [7]. An additional retrieval step (MAC stage) prior to the graph-based retrieval mentioned above (FAC stage) is the solution to our problem. Features are extracted from the original graph representation in order to pre-select cases in a computationally cheap retrieval step during the MAC stage. For our experiments, we implemented this approach with the CAKE framework [8].

The remainder of the paper is organized as follows: The MAC/FAC approach for workflows is described in Section 2 including a brief recall of the graph-based retrieval. Experimental results are discussed in Section 3. A conclusion is drawn in Section 4.

2 A MAC/FAC Approach to Workflow Retrieval

Traditionally, workflows are "the automation of a business process, in whole or part, during which documents, information or tasks are passed from one participant to another for action, according to a set of procedural rules" [9]. In addition, tasks exchange certain *products*, which can be of physical matter (such as ingredients for cooking tasks) or information. Tasks, products and relationships between the two of them form the *data flow*. Broadly speaking, workflows consist of a set of *activities* (also called *tasks*) combined with *control-flow structures* like sequences, parallel or alternative branches, and loops. Tasks and control-flow structures form the *control-flow*. Today, graph representations for workflows are widely used in process-oriented CBR. In this paper we build upon the workflow representation using semantically labeled graphs developed by Bergmann & Gil [5], which is briefly summarized in section 2.1. This graph representation enables modeling related semantic similarity measures which are well inline with experts assessment. Specific heuristic search algorithms for computing the semantic similarity for graphs have been developed, but their scalability with growing case bases is quite limited. This is caused by the inherent computational complexity of graph similarity.

To overcome this problem, we investigate a retrieval method based on the MAC/FAC [7] idea, similar to what was proposed by Leake and Kendall-Morwick [10, 11]. The basic idea behind MAC/FAC is very simple: it is a two-step retrieval approach that first performs a rough pre-selection of a small subset of cases from a large case base. This pre-selection is the MAC stage ("Many Are Called"), which is performed using a selection method which is computationally efficient even for large case bases. For example, cases may be stored in a relational data base and the pre-selection can be performed by an SQL query [12]. Then, the second step called FAC phase ("Few Are Choosen") is executed, which only uses the pre-selected cases to perform the computationally expensive similarity computation. This method improves the retrieval performance, if the MAC stage can be performed efficiently and if it results in a sufficiently small number of pre-selected cases that allows applying the complex similarity measure for retrieval.

The major difficulty with MAC/FAC retrieval in general is the definition of the filter condition of the MAC stage. Since cases that are not selected by the MAC stage will not occur in the overall retrieval result, the completeness of the retrieval can be easily violated if the filter condition is too restrictive. Hence, retrieval errors, i.e., missing cases will occur. On the other hand, if the filter condition is less restrictive, the number of pre-selected cases may become too large, resulting in a low retrieval performance. To balance retrieval error and performance, the filter condition should be a good approximation of the similarity measure used in the FAC stage, while at the same time it must be efficiently computable to be applicable to a large case base in the MAC stage.

We address this problem by proposing an additional feature-based representation of workflows, which is automatically derived from the original graph-based representation. This representation thus simplifies the original representation while maintaining the most important properties relevant for similarity assessment. The MAC stage then selects cases by performing a similarity-based retrieval using a feature-based similarity measure. This similarity measure will partially use the local similarity functions of the graph-based retrieval but in a more simple manner, ignoring the structural properties of the workflow graph. The resulting feature-based retrieval method is thus more efficient. A further important property of this realization of the MAC stage is that the number of selected cases can be easily controlled. Therefore, we introduce a parameter we call *filter size* s, which specifies the number of cases resulting from the MAC stage. Hence, the MAC stage retrieves the s-most similar cases using feature-based retrieval. The choice of the filter size determines the behavior of the overall retrieval method with respect to retrieval speed and error in the following manner: the smaller the filter size, the faster the retrieval but the larger the retrieval error will become. Hence, an appropriate choice of the filter size is important.

We now introduce our approach in more detail. Next, the basic ideas and the notation used in the graph-based retrieval described by Bergmann and Gil [5] are revisited. Then, the feature-based workflow representation and the related similarity measure used in the MAC phase are described.

2.1 Graph-based Retrieval

We represent a workflow as a directed graph $W = (N, E, S, T)$ where N is a set of nodes and $E \subseteq N \times N$ is a set of edges. Nodes and edges are annotated by a type from a set Ω and a semantic description from a set Σ. Type and semantic description are computed by the two mapping functions $T\colon N \cup E \rightarrow \Omega$ and $S\colon N \cup E \rightarrow \Sigma$, respectively. The set Ω consists of the types: *workflow node, data node, task node, control-flow node, control-flow edge, part-of edge* and *data-flow edge*. Each workflow W has exactly one workflow node. The task nodes and data nodes represent tasks and data items, respectively. The control-flow nodes stand for control-flow elements. The data-flow edge is used to describe the linking of the data items consumed and produced by the tasks. The control-flow edge is used to represent the control flow of the workflow, i.e., it links tasks with successor tasks or control-flow elements. The part-of edge represents a relation between the workflow node and all other nodes. Σ is a semantic meta data language that is used for the semantic annotation of nodes and edges. In our work we treat the semantic descriptions in an object-oriented fashion to allow the application of well-established similarity measures.

Based on this representation, Bergmann & Gil [5] introduced a framework for modeling semantic workflow similarity. It is based on a local similarity measure for semantic descriptions $sim_\Sigma : \Sigma^2 \rightarrow [0, 1]$ that must be formulated for nodes and edges.

The similarity of a query workflow QW and a case workflow CW is then computed by means of a legal mapping $m : N_q \cup E_q \rightarrow N_c \cup E_c$, which is a type-preserving, partial, injective mapping function of the nodes and edges of the query workflow to those of the case workflow. For a particular mapping m the overall workflow similarity $sim_m(QW, CW)$ is computed by a particular aggregation of the local similarity values modeled using sim_Σ (for details see [5]). The overall workflow similarity $sim(QW, CW)$ is then determined by the best possible mapping of that kind, i.e.,

$$sim(QW, CW) = \max\{sim_m(QW, CW) | \text{ legal mapping } m\}.$$

As a consequence of this definition, the computation of the similarity requires the systematic construction of such mappings m, which is the cause for the computational complexity of this approach.

2.2 Feature-based Retrieval

The MAC phase for the proposed retrieval approach is based on a feature-based representation of workflows. A feature-based case base $CB' = \{CW'_1, \ldots, CW'_n\}$ is computed offline, i.e., prior to performing the retrieval. Therefore, each case CW'_i is derived from the corresponding case CW_i of the original graph-based case base CB. In the representation of a feature-based case CW', two types of features are considered: *semantic features* and *syntactic features*. A vector V_{sem} represents the semantic features derived from the workflow graph, while a vector V_{syn} represents the syntactic features, thus $CW' = (V_{sem}, V_{syn})$.

Currently, two semantic features are considered. The first feature is related to the data nodes and is represented by a set $D \subseteq N$. The second feature is related to the task nodes and is represented by a set $A \subseteq N$. Hence, $V_{sem} = (D, A)$ with

$$D = \{n \in N | T(n) = DataNode\}$$

$$A = \{n \in N | T(n) = TaskNode\}$$

These features (together with the related semantic description of the nodes in D and A) can be considered an abstraction of the overall graph, as the linking of the nodes is completely ignored.

The syntactic features, however, are simple numerical features that together build a kind of profile reflecting the size of the graph. Hence, V_{syn} is defined as $V_{syn} \in \mathbb{R}^f$, with f being the number of features. These features reflect the number of the various components the graph consists of. Currently, the derived features are: the number of data flow nodes, number of task nodes, number of control flow nodes, the number of data flow edges and the number of control flow edges.

To perform the MAC/FAC retrieval for a given query workflow QW the related feature-based representation QW' of the query is derived in the same manner as for cases in the case base. The similarity measure sim' that compares a query $QW' = (V_{sem_q}, V_{syn_q})$ with a case $CW' = (V_{sem_c}, V_{syn_c})$ is further specified as follows: For both vectors, separate similarity functions are specified. The computed similarity values are then aggregated into the overall similarity. For the two semantic features D and A, the local similarity measure sim_Σ modeled for the graph-based retrieval is used again, but without applying any mapping. Let's assume, $D_q = \{d_{q_1}, d_{q_2}, ..., d_{q_n}\}$ and $D_c = \{d_{c_1}, d_{c_2}, ..., d_{c_m}\}$. The measure sim_Σ is used to assess the similarity between each pair of nodes (d_{q_i}, d_{c_j}). Based on this, a local similarity measure for D is specified as follows:

$$sim'_\Sigma(D_q, D_c) = \Phi\left(\begin{pmatrix} sim_\Sigma(S_q(d_{q_1}), S_c(d_{c_1})) & \cdots & sim_\Sigma(S_q(d_{q_1}), S_c(d_{c_m})) \\ \vdots & \ddots & \vdots \\ sim_\Sigma(S_q(d_{q_n}), S_c(d_{c_1})) & \cdots & sim_\Sigma(S_q(d_{q_n}), S_c(d_{c_m})) \end{pmatrix}\right)$$

Here, Φ is an aggregation function specified for the matrix (s_{ij}) as follows:

$$\Phi((s_{ij})) = \frac{1}{n} \cdot \sum_{i=1}^{n} max\{s_{ij} \,|\, j = 1..m\}$$

Hence, for each data node in the query, the best matching data node in the case is selected. Their similarity is aggregated into the overall similarity for D. This is obviously still a kind of mapping, but it is less constraint with respect to the mapping m computed in the graph-based approach, because each node is mapped independent of the mapping of the other nodes and independent of any linking. Thus, the computed similarity is an upper bound for the similarity of the nodes in D that can be achieved by the best mapping m in the graph-based retrieval. The local similarity measure $sim'_\Sigma(A_q, A_c)$ for A, the set of task nodes, is specified analogously. Again, the computed similarity can be considered an upper bound for the similarity of the task nodes.

In addition, the similarity of the syntactic features is considered. Here, we apply a standard similarity measure $sim' : \mathbb{R}^2 \rightarrow [0, 1]$. In order to aggregate the local similarity values into the global similarity, feature weights are considered for the features in V and for the semantic features D and A. Lets assume, $W = (w_1, \ldots, w_f)$ is a vector of feature weights for the corresponding features in

$V = (v_1, \ldots, v_f)$ and w_d and w_a are the feature weights for D and A, respectively. Then, the global similarity between the query and the case for feature-based retrieval is specified as follows:

$$sim'(QW', CW') = \frac{w_d \cdot sim'_{\Sigma}(D_q, D_c) + w_a \cdot sim'_{\Sigma}(A_q, A_c) + \sum_{i=1}^{f}(w_i \cdot sim'(v_{q_i}, v_{c_i}))}{w_d + w_a + \sum_{i=1}^{f} w_i}$$

The selection of cases $CW_1, ..., CW_s$ during the MAC phase is performed by a similarity-based retrieval from CB' using the similarity measure $sim'(QW', CW'_i)$. Thereby, the s most-similar cases are retrieved (s is the filter size).

3 Experimental Results

The benefits of our approach are demonstrated by means of some experiments in the cooking domain. Cooking recipes are represented in the form of workflows created by an automated extraction procedure [6]. We measure the retrieval time and the retrieval error for the MAC/FAC approach parameterized by

- the size of the case base,
- the filter size s, and
- the number of retrieval phases (the entire MAC/FAC retrieval vs. the MAC stage only vs. the FAC stage only).

The retrieval error is measured by the percentage of cases missing in the retrieval result of the MAC/FAC approach with the retrieval result of the un-filtered graph-based approach (the FAC stage only) as a base line. Broadly speaking, the retrieval error measures how many cases we are loosing during the MAC phase that are considered to be relevant by the more sophisticated retrieval method of the FAC phase.

We investigated the following hypotheses:

H1. The retrieval time of the feature-based retrieval (MAC stage only) is significantly shorter than the retrieval time of the graph-based retrieval (FAC stage only).

H2. For a certain filter size s, the retrieval time of the MAC/MAC approach is sufficiently shorter compared to the graph-based retrieval (FAC stage only) while the retrieval error is significantly low.

Three case bases with the sizes of 200 cases, 2000 cases, and 15000 cases have been extracted from various cooking recipe websites by using the procedure, described in [6]. Complete recipes are considered and all recipes are chosen randomly. Most of the recipes consist of few tasks and few ingredients used as input for the tasks. Currently, we are only extracting cooking workflows with sequential control-flow.

The experiments are performed for two hundred queries, which are identical for the three case bases.

The extracted features have been specified as introduced in Section 2.2: Only the names of ingredients and tasks have been extracted as semantic features. Furthermore, we do not yet consider an ontology to derive similarity functions for this domain, which can cover the huge number of tasks and ingredients in our large case base. Hence, for this work, the Levenshtein distance measure and standard similarity measures are used for the local similarity computations for the semantic syntactic features.

We implemented our MAC/FAC retrieval approach and run the experiment in the CAKE framework [8]. We run the experiment on Windows 7 Enterprise 64-bit, using an Intel i7 CPU 870 @ 2.93GHz and 8.00 GB ram.

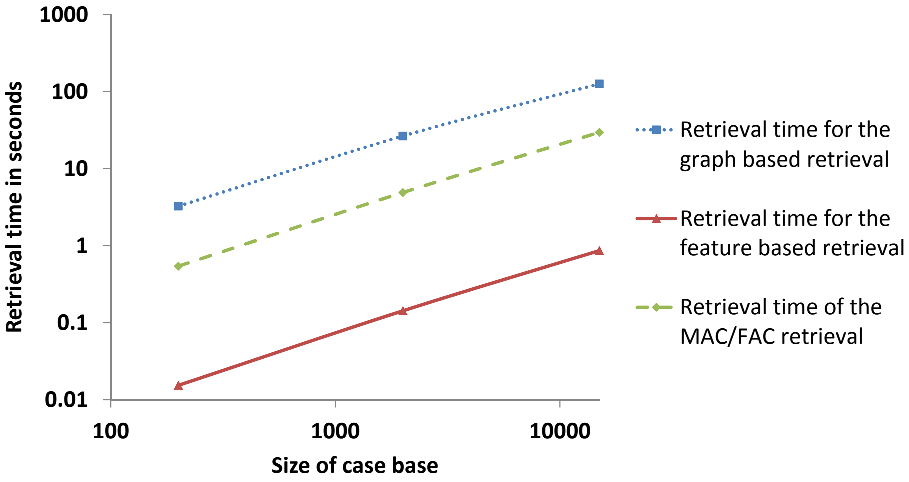

Fig. 1: The average retrieval time of the graph-based, the feature-based, and the MAC/FAC-retrieval for the filter size s set to 12 % of the size of the case base.

The experimental results are shown in Figure 1 to Figure 4[1]. In the first experiment (see Figure 1), the filter size s has been specified without loss of generality by 12 percent of the size of the particular case base.

We can see from the squares and triangles depicted in Figure 1 that the retrieval time for the feature-based retrieval is significantly shorter than the retrieval time for the graph-based retrieval for the three case bases considered. This clearly confirms hypothesis **H1**.

[1] This pdf paper version shows corrected figures compared to the printed publication.

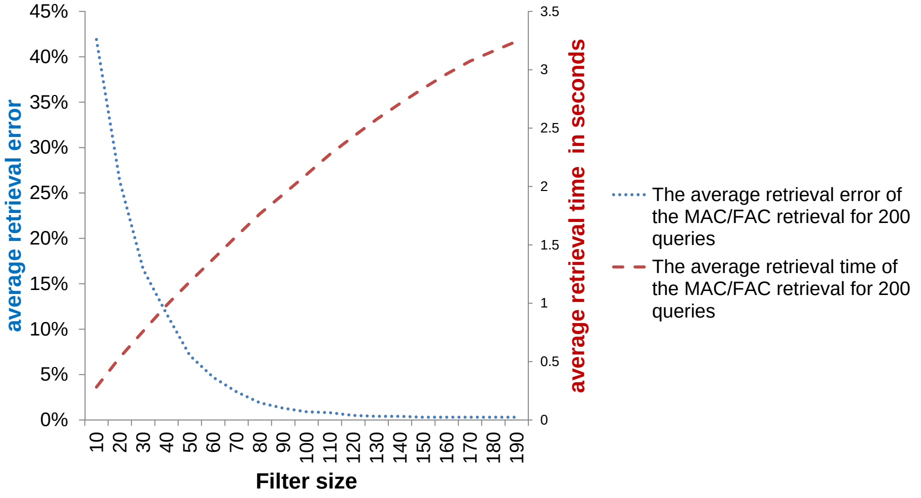

Fig. 2: The average retrieval error and average retrieval time for the case base with 200 cases with different filter sizes s.

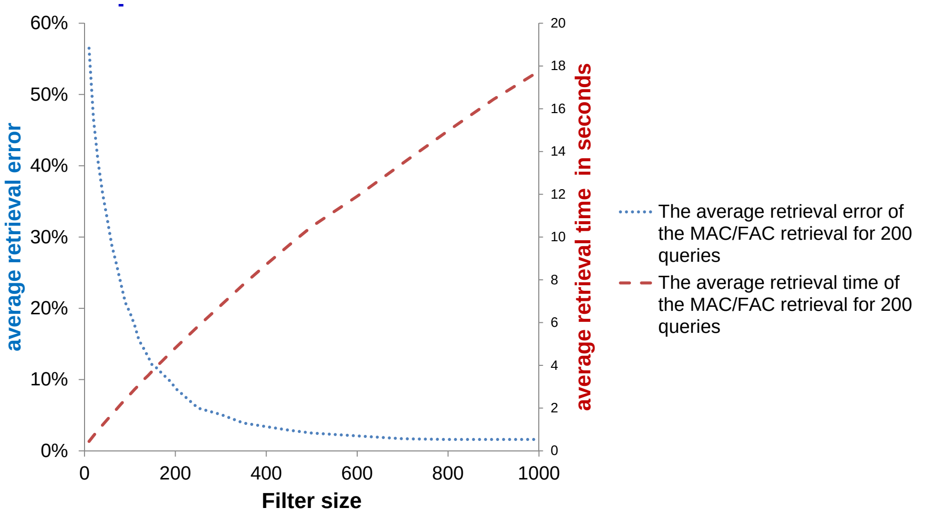

Fig. 3: The average retrieval error and average retrieval time for the case base with 2000 cases with different filter sizes s.

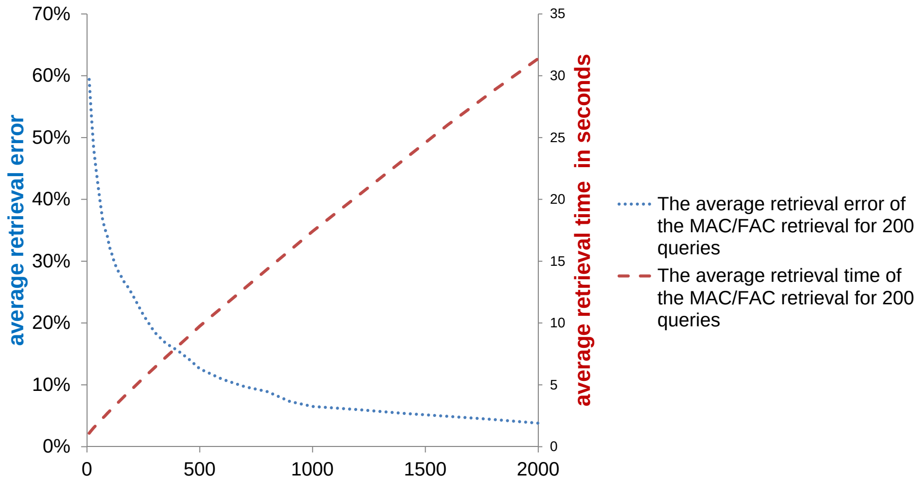

Fig. 4: The average retrieval error and average retrieval time for the case base with 15000 cases with different filter sizes s.

Furthermore, the figure shows that the retrieval time for the MAC/FAC-retrieval (depicted by diamonds) is also significantly shorter than the retrieval time of the graph-based retrieval. We conducted further experiments with a variable filter size s for each of the three case bases. Figure 2 to Figure 4 depict the average retrieval errors and average retrieval times for these runs. The results illustrate that for a certain filter size the retrieval time is significantly shorter while the retrieval error is sufficiently low. This confirms our hypothesis **H2**.

4 Conclusion

We presented a new MAC/FAC approach to scale the similarity-based retrieval of semantic workflows. A similar method was proposed by Leake and Kendall-Morwick [10, 11], but they use a different filter method in the MAC phase. Our approach is based on a feature-based representation of workflows, which includes properties that are relevant for the similarity assessment. In our current experiments, we just considered data and task nodes represented as sets as well as some ad-hoc features representing the size of the workflow. Even with this representation we were able to show that the retrieval time can be significantly reduced without introducing a very high error rate. A more elaborated definition of features with related local similarity measures will probably lead to a better performance. Currently, we don't apply any indexing of the features to improve the retrieval speed of the MAC stage. Due to the use of the two set features A and D, the straight-forward application of an existing indexing method is not possible. However, we feel that case-retrieval nets could be extended to be able to cover those features as well. Also methods for optimizing the feature weights and the filter size would be useful. Both issues will be addressed in our future work. Also more detailed empirical evaluations are necessary, involving other domains and more sophisticated ontologies and similarity measures.

5 Acknowledgements

This work was funded by the German Research Foundation, project number BE 1373/3-1.

References

1. Jarmulak, J., Craw, S., Rowe, R.: Self-optimising cbr retrieval. In: Proceedings of the 12th IEEE International Conference on Tools with Artificial Intelligence. (2000) 376–376
2. Jarmulak, J., Craw, S., Rowe, R.: Genetic algorithms to optimise cbr retrieval. In: Lecture Notes in Computer Science. (2000) 159–194
3. Wess, S., Althoff, K.D., Derwand, G.: Using k-d trees to improve the retrieval step in case-based reasoning. In: Lecture Notes in Computer Science. (1994) 167–181
4. Burkhard, H.D.: Extending some concepts of CBR – foundations of case retrieval nets. In: Case-Based Reasoning Technology. Springer (1998) 17 – 50
5. Bergmann, R., Gil, Y.: Retrieval of semantic workfows with knowledge intensive similarity measures. In: Case-Based Reasoning. Research and Development, 19th International Conference on Case-Based Reasoning, ICCBR 2011, Springer (2011) 17–31
6. Schumacher, P., Minor, M., Walter, K., Bergmann, R.: Extraction of procedural knowledge from the web. In: WWW'12 Workshop Proceedings, ACM (2012)
7. Forbus, K.D., Gentner, D., K., L.: Mac/fac: A model of similarity-based retrieval. Cognitive Science **19**(2) (1995) 141 – 205
8. Bergmann, R., Freßmann, A., Maximini, K., Maximini, R., Sauer, T.: Case-Based support for collaborative business. In: Advances in Case-Based Reasoning, 8th European Conference, ECCBR 2006, Fethiye, Turkey, September 4-7, 2006, Proceedings. Volume 4106., Springer (2006) 519–533
9. Workflow Management Coalition: Workflow management coalition glossary & terminology. http://www.wfmc.org/standars/docs/TC-1011_term_glossary_v3.pdf (1999) last access on 05-23-2007.
10. Leake, D.B., Kendall-Morwick, J.: Towards Case-Based support for e-Science workflow generation by mining provenance. In: Advances in Case-Based Reasoning, 9th European Conference, ECCBR 2008, Trier, Germany, September 1-4, 2008. Proceedings. (2008) 269–283
11. Kendall-Morwick, J., Leake, D.: A toolkit for representation and retrieval of structured cases. In: Proceedings of the ICCBR 2011 Workshops. (2011) 111–120
12. Schumacher, J., Bergmann, R.: An efficient approach to Similarity-Based retrieval on top of relational databases. In Blanzieri, E., Portinale, L., eds.: Advances in Case-Based Reasoning, 5th European Workshop, EWCBR 2000, Trento, Italy, September 6-9, 2000, Proceedings. Volume 1898 of Lecture Notes in Computer Science., Springer (2000) 273–284 The original publication is available at www.springerlink.com.