A Case Based Approach to Serve Information Needs in Knowledge Intensive Processes

Debdoot Mukherjee[1], Jeanette Blomberg[2], Rama Akkiraju[2], Dinesh Raghu[1], Monika Gupta[1], Sugata Ghosal[1], Mu Qiao[2], and Taiga Nakamura[2]

[1] IBM Research – India
{debdomuk,diraghu1,monikgup,gsugata}@in.ibm.com
[2] IBM Almaden Research Center, USA
{blomberg,akkiraju,taiga,mqiao}@us.ibm.com

Abstract. Case workers who are involved in knowledge intensive business processes have critical information needs. When dealing with a case, they often need to check how similar case(s) were handled and what best practices, methods and tools proved useful. In this paper, we present our Solution Information Management (SIM) system developed to assist case workers by retrieving and offering targeted and contextual content recommendations to them. In particular, we present a novel method for intelligently weighing different fields in a case when they are used as context to derive recommendations. Experimental results indicate that our approach can yield recommendations that are approximately 15% more precise than those obtained through a baseline approach where the fields in the context have equal weights. SIM is being actively used by case workers in a large IT services company.

1 Introduction

Case Management [23] has emerged as the discipline for supporting flexible and knowledge intensive business processes, which may require significant human judgment and decision making. Unlike traditional Business Process Management (BPM), which has focused on automating process workflows, Case Management is aimed at equipping *knowledge workers*[1] efficiently steer processes toward completion. Since, knowledge workers add significant economic value to an enterprise and their contributions are especially critical in growing the services economy, the demand for Case Management tools has been growing [1]—especially in domains such as customer relationship management, IT service management, healthcare, legal, insurance and citizen services. When knowledge workers begin to work with a case they often ask—*Did we handle such a case before? If so, how? What best practices are available to solve similar cases?* To get answers to such questions, they often search in enterprise repositories. However, knowledge workers are frustrated with the inability of the available knowledge management tools in finding the information they need, when they need it, due to the poor state of the art of enterprise search. Studies report that they may spend 15% to 35% of their time

[1] The term, *knowledge worker*, was first coined by Peter Drucker to denote those who develop or apply knowledge in the workplace. [16] discusses different roles of knowledge workers.

S. Basu et al. (Eds.): ICSOC 2013, LNCS 8274, pp. 541–549, 2013.

Key Case Fields (by Type)	
Categorical	Win/Loss Outcome, Industry, Country, Offerings, Contract Type, Service Line, Risk Rating, Delivery Center
Numeric	Total Contract Value, Governance Cost, Transition Cost, Onsite / Offshore ratio, Resource Mix
Text	Client Name, Opportunity title & overview, Scope, Solution Summary, Competition Analysis, Win Themes, Value Proposition, Governance, Transition methodology, Risks, Assumptions, Cost Case, Architecture, Legal Terms & Conditions, Project plan, SLA - Support Model

Fig. 1. Sample Fields from SIM's Case Model

searching for information and are successful in finding relevant information less than 50% of the time [7,5]. In practice, what works better is reaching out to subject matter experts in the organization through informal networks. But, identifying the right person often requires numerous phone calls and email exchanges, which takes up precious, productive time of knowledge workers. Our study of knowledge workers at a large IT services organization reveals that a multitude of technical and organizational challenges currently make it extremely difficult for case workers to find the information necessary for their daily work. Clearly, developing technologies for effectively aggregating and disseminating case knowledge is a strong business imperative for next generation Case Management [13,20,22] and Social BPM [21] products.

In this paper, we describe how information retrieval guided by the context of the case-at-hand and the semantics of the case domain generates useful content recommendations for the knowledge workers. We discuss a knowledge management application called Solution Information Management (SIM), which was developed to serve information needs arising in the *Opportunity-To-Order* process (*i.e.,* the sales lifecycle) at an IT services company. SIM mines contextual and targeted information by searching a federated set of repositories. The repositories store solution design documents created during past opportunities as well as best-practice reference materials about offerings, delivery capabilities, lessons learned and engagement processes. A *case* in SIM uniquely identifies an IT service deal that was pursued in the past. For each case, we catalog all information related to the deal as fields in a *case model*; Figure 1 shows a sample of fields in SIM's case model. We apply an array of information extractors to resolve contents of different case fields from the unstructured documents created in a deal. Then, the richly fielded case models are indexed such that one can execute targeted semantic queries and not just full text keyword search. Suppose, a case worker is looking for existing prior assets or lessons learned on "low cost data center consolidation solutions in financial service industry in Western Europe". In SIM, one can create a complex query to address such a requirement—*Geography* : "Western Europe", *Offering* : "data center consolidation", *Win-Theme* : "Low Cost", *Industry* : "Financial Services". The results obtained from a such a query are much more precise than what a keyword search would yield. Further, the SIM system can generate content recommendations for the information needs in different process steps based on the already known fields in the case or the *context* of the case. An interesting question that arises is how to weigh the affect of different fields in the context. In the above example, suppose the case worker is now interested in recommendations for potential risks underlying the solution. How do we weigh the four query clauses as we look to retrieve cases with *Risks* that can be of interest? Do we weigh the clause on *Offering* more than the others or is a match of the *Industry* more important to fetch relevant *Risks*? Resolving an appropriate

weighting of the different query clauses is crucial in order to maximize the relevance of recommendations. It is a complex problem since there are hundreds of fields in industrial case repositories and manually specifying weighting schemes is infeasible. We propose an automated approach, named *Correspondence Analysis*, which infers how one case field can influence recommendations for another case field. Correspondence scores dictate the weights of different query clauses in generating recommendations.

We conduct experiments where we assess the relevance of recommendations on two different case fields, obtained from a corpus of 715 cases. The relevance of recommendations obtained through our approach is significantly better than that from a baseline approach which assumes equal weights for all fields in context. Improvement observed in standard IR metrics like *Precision@K* and *nDCG* is as high as 15%.

2 System Overview: Solution Information Management

In this section, we describe Solution Information Management (SIM), a tool that assembles content from a variety of data sources relevant to case domain, indexes the information after converting it into a semantic format, and then delivers relevant information to the case worker depending upon the context of a case. Figure 3 shows the different stages in the knowledge engineering pipeline in SIM. Here, we briefly outline the function of each stage. Refer to our technical report [25] for a detailed overview.

Crawl & Parse: We configure crawlers in SIM that periodically download the contents of the different repositories. The crawlers output files in their native, binary formats, *e.g.,* .pdf, .ppt, .xls, .doc. The next step is to *parse* formatted text from such files. Also, we export pages and slides as images; these images show up alongside search results to enable a preview feature.

Annotate: The *Annotate* stage creates semi-structured case models with information extracted from dense, unstructured documents associated with historical cases. The *Segmenter* module takes as input the formatted text parsed from the documents. It distinguishes the headings in the documents from any other text based on their special formatting or font-styles. Next, it feeds the words in an inferred heading to a trained text classifier model which predicts the case semantic implied by that heading. Once we determine the case semantic for a heading, we extract the text from the region following the heading into a field in a case model. Additionally, we compute a *Quality Score* and a *Summary* for every text field. The *Quality Score* assesses the amount of information present in the case field relative to that present in the same field in other cases in the corpus. Such a score helps penalize sparse fields as we generate recommendations. A *Summary* for a case field is obtained by applying the Maximal Marginal Relevance technique [4] to choose a small number of sentences that convey the maximum information. Summaries help users do a quick evaluation of case fields.

Index: The case models created in the *Annotate* stage are imported into a full text index of a search engine. SIM uses Apache Solr as the foundation for indexing and search.

Query & Search: To generate recommendations for a field of a case being worked upon, SIM creates an OR-ed construct of query clauses, where each clause is generated from the contents of a known field in the case. A key question that arises is how do we

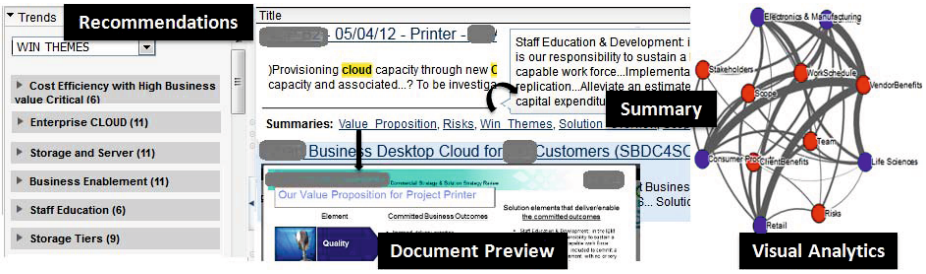

Fig. 2. Visualizing Results in SIM

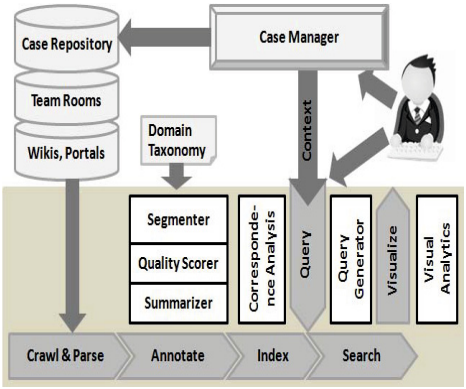

Fig. 3. Knowledge Engineering Pipeline in SIM

1: **function** CORRESPONDENCE-ANALYSIS
Input: Case Corpus, C; Set of Fields, F
Output: $Corr$ - $|F| \times |F|$ matrix
2: Initialize Σ - $|C|^2 \times |F|$ observation matrix
3: **for all** $(c_i, c_j) \in C \times C, i \neq j$ **do**
4: Initialize observation vector, σ
5: **for all** Field $f_k \in F$ **do**
6: $\sigma_k \leftarrow Sim(c_i.f_k, c_j.f_k)$
7: Add σ to Σ
8: **end for**
9: **end for**
10: **for all** Field $f_i \in F$ **do**
11: Build a regression model, M, from Σ to model column i using other columns as features.
12: **for all** Field $f_j \in F$ **do**
13: $Corr(j, i) \leftarrow$ Coeff. of feature j in M
14: **end for**
15: **end for**
16: **end function**

Fig. 4. Correspondence Analysis

weigh the matches of the different query clauses to maximize relevance of recommendations. To address this issue, we develop *Correspondence Analysis*, a technique that helps us ascertain the weight of each query clause. We discuss it in detail in Section 2.1. Further, the relevance score for any result is boosted based on different factors such as users' rating, document age, the number of previous hits on the result, *Quality Score* for the result field and others domain specific rules (*e.g.,* boost if the deal was won).

Visualize: Figure 2 illustrates how recommendations are visualized in the SIM tool. On the left-hand pane, one finds a list of key topics that are obtained by clustering the recommendations generated for a particular case field. In the middle pane, we present the recommended case models as well as relevant reference materials. For each result, one can view the *summaries* of different case fields. Also, one may open up a *document preview* for a case field, which shows snapshots of the document regions where the field is documented. SIM also helps users visually discover interesting associations within selected cases through interactive graph visualizations.

2.1 Correspondence Analysis

As we derive content recommendations for a certain field of a case (henceforth referred to as the *target* field), it is important to understand what other fields in the case can serve as *context*. For instance, as we seek recommendations on the case field, *Risks*,

does it make sense to search for *Risks* in cases from the same *Industry* or cases with the same *Solution Offering*? If both *Industry* and *Solution Offering* appear to lend *context* to *Risks*, then how does one weigh the influence of each of these fields? Turns out that even domain experts are unable to conclusively answer such questions. Again, assigning equal weights to the affect of each field in the context does not appear to be ideal (See Section 3). Moreover, since there could be hundreds of fields in case repositories, manually defining weighting schemes may not be feasible. Here, we describe *Correspondence Analysis*[2], an automated approach that analyzes the case corpus to infer how similarities in different case fields correlate with each other. The correspondence output can be used for defining preferential weighting for fields in SIM queries.

For each pair of case fields, say α and β, we define *Correspondence*, $Corr(\alpha, \beta)$, as the degree to which similarity in α corresponds to similarity in β across pairs of cases. A high value for $Corr(\alpha, \beta)$ suggests that α is a good candidate to serve as *context* for β because if we are able to retrieve cases with similar α, then it is likely that the contents of β in those cases may recur in the current case. For the above example, if our analysis of the case corpus shows that cases with the same *Offering* often exhibit similar *Risks*, then a case worker would be interested to find *Risks* in past cases that have the current *Offering*. Thus, it may be worthwhile to assign a high weight for the query clause with *Offering*. Now, when deriving recommendations for a target field, it is important to have the "right" relative weighting for all other fields in the context. We use multiple linear regression as a tool to determine the relative impact that fields in the context can have upon the target field.

Figure 4 discusses the algorithm for correspondence analysis. We sample pairs of cases from the case corpus to observe how the different fields are similar across the case pairs. For each pair of cases, we create an observation vector where each observation measures the similarity of a particular field across the two cases. The similarity function, Sim, depends on the type of field. We use boolean similarity for categorical fields, cosine similarity[3] for text fields and inverse of euclidean distance[4] for numeric fields. Next, in order to assess how similarity of a target field may be influenced by similarities in other fields, we regress the observations from all other fields against the corresponding observations for the target field. The coefficients obtained from a linear regression model can be indicative of how similarities in different fields influence similarity in the target field.

3 Experiments on Contextual Search

In this section, we report experiments conducted to assess the efficacy of our proposed approach in finding the "right" context in order to maximize relevance of recommendations. As discussed in Section 2, when seeking recommendations for a *target* case field, we weigh the query clauses created from the contents of other fields by their respective

[2] Not to be confused with the multi-variate statistical technique with the same name that summarizes categorical data in a two dimensional graphical format

[3] `http://en.wikipedia.org/wiki/Cosine_similarity`

[4] `http://en.wikipedia.org/wiki/Euclidean_distance`

Correspondence scores with the target case field. Our experiments measure the usefulness of such a weighting in improving relevance of recommendations over a baseline approach where equal weights are assigned to each clause in the contextual query.

3.1 Experimental Set-Up

In our experiments, we use a case corpus of 715 case models cataloged from information created during sales engagements at a large IT services company. Each case model was created by aggregating information about a single deal from three different databases within the company. The schema for our integrated case models consisted of 314 fields of different types (*e.g.,* categorical, text, numeric, dates). However, not all case models had all 314 fields; in fact most of them were sparsely populated. For the purposes of our experiments, we choose *Risks* and *Assumptions* as the two target fields for which we generate recommendations. Both of these fields are free text fields; often their contents are organized as a bulleted list of items, sometimes even over a hundred items in a single case. Such lists of *Risks* and *Assumptions* are particularly useful to conduct quality assurance reviews and are a necessary input for crafting clauses in the legal contract when closing a deal.

Competing Approaches: We investigate the efficacy of two approaches of leveraging context in deriving relevant recommendations for a target field. First, we evaluate a *baseline approach* where we construct query clauses out of the contents all non-empty fields in a case model except the target field. In this approach, the matches for all the query clauses are weighed equally while generating recommendations. Second, we apply the *weighted approach*, where we create query clauses from a select set of fields that have a high correspondence score with the target fields. Further, the query clauses are weighed in proportion to the correspondence scores of the respective fields.

Generating Recommendations: We randomly select 8 case models from the case corpus where the fields, *Risks* and *Assumptions* are non-empty. For each case, we construct two queries following the two approaches described above. We execute the queries with Apache Lucene to obtain a ranked list of case models in the corpus with similar contexts. Finally, we retrieve the contents of the target fields in the result case models and present them to an expert who assigns relevance judgments as described below.

Judging Relevance: Judging relevance of a recommendation is hard for anyone who is not actually involved in the case and getting time from case workers to run controlled experiments is always a challenge. However, we manage to work around this issue in the following manner. Note that the case models from which queries were created already have the target fields filled up, so we can use their contents as *ground truth*. Now, the task of comparing two items is much easier than deciding the relevance of a recommendation to a given context. Thus, we ask an expert user who understands the vocabulary of the case domain to compare the recommendation results obtained for a query with its ground truth. The expert chooses one of the 3 labels for each recommendation—*0 : "Not Related"; 1 : "Somewhat Related", 2 : "Related"*. If there is an exact match of any *Risk* or *Assumption* item listed in the recommendation to any item in the ground truth, then it is labeled as "2". If there is some topical match, then the recommendation

Field	*Baseline*		*Weighted*	
	nDCG	*P@20*	*nDCG*	*P@20*
Risks	0.504	0.312	0.668	0.456
Assumption	0.57	0.325	0.688	0.49
Overall	0.535	0.318	0.678	0.473

Fig. 5. Summary of Results

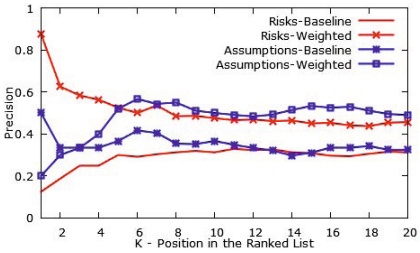

Fig. 6. Precision@K

is marked as "1". Such a labeling strategy helped us collect relevance judgments for each of the top 20 recommendation results derived for the two competing approaches across eight queries for the two chosen target fields; totaling to 640 judgments.

Collecting Metrics: The relevance judgments are used to compute two metrics: *Precision@K* and *Normalized Discounted Cumulative Gain (nDCG)*. For computing precision, a relevance label of 0 is considered irrelevant, labels of 1 and 2 are considered relevant. Now, *Precision@K* is defined as the fraction of relevant results for the top-*K* ranked recommendations. *nDCG* [8] is often used as a measure for evaluating a ranked list with multiple relevance levels. The premise behind Discounted Cumulative Gain (DCG) is that highly relevant documents appearing lower in a search result list should be penalized as the graded relevance value is reduced logarithmically proportional to the position of the result. For a judgment vector of length p, we compute DCG_p as follows:

$$DCG_p = \sum_{i=1}^{p} \frac{2^{rel_i} - 1}{log_2(i+1)}$$

Next, we re-compute DCG_p after sorting the judgment vector and call it Ideal DCG ($IDCG_p$). Finally, $nDCG$ for the judgment vector is defined as DCG_p expressed as a fraction of $IDCG_p$.

3.2 Experimental Results

Figure 5 summarizes the metrics *nDCG* and *Precision@20* as observed for the two competing approaches on our dataset. For *Risks*, the *weighted* approach records an improvement of 16.4% in *nDCG* and 14.4% in Precision@20 over the *baseline* approach. For *Assumptions*, we find increases of 11.8% and 16.5% in *nDCG* and Precision@20 respectively. Thus, on average, across the two case fields, both metrics show an improvement of $\approx$ 15%. Figure 6 plots the values of *Precision@K* for K = 1 through 20 for the two approaches, *baseline* and *weighted*. For the recommendations on *Risks*, the values of *Precision@K* for the *weighted* approach are are consistently higher than those recorded by the *baseline* approach. For *Assumptions*, the curve for the weighted approach is seen to be lagging at K=1,2 but then it surges ahead and thereafter leads the baseline's curve. These results clearly indicate that an intelligent weighting of the different fields in context can improve the relevance of recommendations and that *Correspondence Analysis* can be a viable approach for choosing the weights.

4 Related Work

Recently, there has been a lot of research on Adaptive Case Management and Social BPM technologies for handling ad hoc business processes [23,13,22,21,10]. However, these efforts have largely focused on developing better case modeling techniques to enhance the level of collaboration between case workers in order to increase the throughput of case processing. We believe that research in case management should also attend to the problem of serving information needs of case workers since cases often get stalled because case workers do not have adequate information. Our work is a first step in this direction.

In the past, reuse of business process information, including formal models and implementation artifacts; has found interest in the BPM community [24]. RepoX [19] and MIT Process Handbook [11] allow storage of business process models with free text search and structured search capabilities. Our past work [6] introduced the notion of contextual search and demonstrated its benefit for requirement gathering activities in SAP engagements. This paper improves upon [6] in the following ways. Unlike the work in [6] that only dealt with textual artifacts, the approach presented in this paper can infer an appropriate weighting of context with different types of fields—text, numeric and categorical. Further, the computation of the strength of an associative relationship between two fields in [6] ignored the influence of other case fields; we address this limitation through the multi-variate modeling in *Correspondence Analysis*. Moreover in [6], the results were not evaluated with the help of relevance judgments from experts. Related work in other academic communities include the literature on Knowledge Management [14,12,3] and research on *Case Based Reasoning* [2,17].

5 Conclusions and Future Work

The SIM tool is being actively used in the *Opportunity-To-Order* process at a large IT Services company and has received positive feedback from its users. They believe that this domain-specific knowledge management system delivers much more precise and contextual results than the enterprise-wide search system they used before. The users suggest that the tool reduces dependencies on personal networks and and yields significant productivity improvements as it jump-starts the case work with relevant information. In this paper, we present a controlled experiment to evaluate the efficacy of a key aspect of our system–generating preferentially weighted contextual queries. Future work can look at design of experiments to study the holistic effect of SIM on knowledge worker productivity. Also, we are currently extending our solution along a number of dimensions to deliver more precise recommendations and enrich the user experience for case workers. Our on-going efforts include: construction of large graphs that link information from different sources, development of stocastic graph inference techniques to answer queries on the graphs and a cognitive system to parse natural language queries.

References

1. Case Management - Combining Knowledge with Process, `http://bit.ly/cErahE` (accessed: May 29, 2013)
2. Aamodt, A., Plaza, E.: Case-based reasoning: Foundational issues, methodological variations, and system approaches. AI Communications 7(1), 39–59 (1994)

3. Alavi, M., Leidner, D.E.: Knowledge management and knowledge management systems: Conceptual foundations and research issues. MIS Quarterly 25(1), 107–136 (2001)
4. Carbonell, J., Goldstein, J.: The use of mmr, diversity-based reranking for reordering documents and producing summaries. In: Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 335–336. ACM (1998)
5. Feldman, S., Sherman, C.: The high cost of not finding information. Information Today Inc. (2004)
6. Gupta, M., Mukherjee, D., Mani, S., Sinha, V.S., Sinha, S.: Serving information needs in business process consulting. In: Rinderle-Ma, S., Toumani, F., Wolf, K. (eds.) BPM 2011. LNCS, vol. 6896, pp. 231–247. Springer, Heidelberg (2011)
7. IDC. Quantifying Enterprise Search (2002)
8. Järvelin, K., Kekäläinen, J.: Cumulated gain-based evaluation of ir techniques. ACM Trans. Inf. Syst. 20(4), 422–446 (2002)
9. Kim, J., Xue, X., Croft, W.B.: A probabilistic retrieval model for semistructured data. In: Boughanem, M., Berrut, C., Mothe, J., Soule-Dupuy, C. (eds.) ECIR 2009. LNCS, vol. 5478, pp. 228–239. Springer, Heidelberg (2009)
10. Liptchinsky, V., Khazankin, R., Truong, H.-L., Dustdar, S.: A novel approach to modeling context-aware and social collaboration processes. In: Ralyté, J., Franch, X., Brinkkemper, S., Wrycza, S. (eds.) CAiSE 2012. LNCS, vol. 7328, pp. 565–580. Springer, Heidelberg (2012)
11. Malone, T.W., Crowston, K., Herman, G.A.: Organizing business knowledge. MIT Press (2003)
12. McDermott, R.: Why information technology inspired but cannot deliver knowledge management. California Management Review 41(4), 103–117 (1999)
13. Motahari-Nezhad, H.R., Bartolini, C., Graupner, S., Spence, S.: Adaptive case management in the social enterprise. In: Liu, C., Ludwig, H., Toumani, F., Yu, Q. (eds.) ICSOC 2012. LNCS, vol. 7636, pp. 550–557. Springer, Heidelberg (2012)
14. Nonaka, I.: A dynamic theory of organizational knowledge creation. Organization Science
15. Osiriski, S., Stefanowski, J., Weiss, D.: Lingo: Search results clustering algorithm based on singular value decomposition. In: Proceedings of the Intelligent Information Processing and Web Mining, IIPWM, vol. 4, pp. 359–368 (2004)
16. Reinhardt, W., Schmidt, B., Sloep, P., Drachsler, H.: Knowledge Worker Roles and Actions: Results of Two Empirical Studies. Knowledge and Process Management 18(3), 150–174 (2011)
17. Rissland, E.L., Daniels, J.J.: A hybrid cbr-ir approach to legal information retrieval. In: Proceedings of the Fifth International Conference on Artificial Intelligence and Law (ICAIL)
18. Robertson, S., Zaragoza, H., Taylor, M.: Simple bm25 extension to multiple weighted fields. In: Proceedings of the Thirteenth ACM International Conference on Information and Knowledge Management, pp. 42–49. ACM (2004)
19. Song, M., Miller, J.A., Arpinar, I.B.: Repox: An xml repository for workflow designs and specifications. PhD thesis, Citeseer (2001)
20. Swenson, K.D.: Mastering the Unpredictable. Meghan-Kiffer Press (2010)
21. Swenson, K.D., Palmer, N., Kemsley, S., et al.: Social BPM. Future Strategies Inc. (2011)
22. Swenson, K.D., Palmer, N., et al.: How Knowledge Workers Get Things Done. Future Strategies Inc. (2012)
23. Van der Aalst, W.M., Weske, M., Grünbauer, D.: Case handling: a new paradigm for business process support. Data & Knowledge Engineering 53(2), 129–162 (2005)
24. Yan, Z., Dijkman, R., Grefen, P.: Business process model repositories–framework and survey. Information and Software Technology 54(4), 380–395 (2012)
25. Mukherjee, D., et al.: A Case Based Approach to Serve Information Need. Knowledge Intensive Processes. IBM Technical Report (2013)