Empowering the Knowledge Worker: End-User Software Engineering in Knowledge Management

Witold Staniszkis(✉)

Rodan Development, Wyczółki 89, 02-820 Warsaw, Poland
Witold.Staniszkis@rodan.pl

Abstract. We present a novel architecture of a knowledge management system meeting the end-user software engineering requirements, thus empowering the knowledge worker to eliminate such intermediaries as system analysts and application programmers. Advantages of direct representation of user requirements in executable knowledge management application specifications, as well as the resulting system agility and ease of maintenance are highlighted. The state-of-the-art in the end-user software engineering area pertaining to the knowledge management systems realm comprises information about the on-going research and development efforts. The principal features of a knowledge management system toolbox are described, comprising among others, such functional areas as semantic modelling of knowledge object repositories, and adaptive management of knowledge management processes. Finally we succinctly discuss the end-user oriented methodology guiding specification of the knowledge management application solutions.

Keywords: Knowledge management · End-user software engineering · Dynamic workflow · Semantic content modelling · Knowledge maps · Adaptive case management

1 Introduction

Rapid growth of the international trade and cooperation on the one hand and the global Information and Communication Technology (ICT)-driven communication powered by the Internet have fuelled unprecedented expansion of global collaboration in practically all walks of human activity. Virtual organisations spanning not only diverse countries but also entire regions become an ubiquitous and dynamic phenomenon. A good example are the European research programmes based on international project consortia, i.e. virtual organisations, characterised by well-defined goals to be attained within a specific time frame.

Also the nature of human activities has undergone a dramatic change resulting in more than 50 % of workers being classified as "knowledge workers", a termed coined by Peter Drucker over half of century ago, whose productivity underlies the competitive advantage of all developed economies. Indeed, again according to Peter Drucker [13], productivity of the knowledge workers represents the major management challenge of the 21st century.

Notwithstanding the ubiquity of such ICT environments as networking, email, social media and content management enhancing the capability of goal-oriented collaborating

S. Hammoudi et al. (Eds.): ICEIS 2015, LNBIP 241, pp. 1–17, 2015.
DOI: 10.1007/978-3-319-29133-8_1

teams, jointly known as organization 2.0 platforms, much needs to be done to leverage investment in the intellectual capital represented and produced by the knowledge workers.

A survey of knowledge worker activities reported by Nathaniel Palmer [41] reveals that over 60 % of the working day is spent in unstructured and often unpredictable work patterns. This telling result explains, at least partially, the common fallacies of the business process management (BPM) projects aiming at supporting human collaboration within the knowledge-intensive work activities. Clearly a novel approach is needed to support the non-production (in the Fredric Taylor sense) work processes of the knowledge worker.

The major advantage of the end-user-driven design and development of the knowledge management application solutions is the elimination of intermediaries, such as system analysts and application programmers, thus enabling the direct representation of the user requirements in executable application specifications. Direct involvement of the end-users in the development process leads to increased system agility and ease of maintenance. The ubiquitous cloud environments provide flexibility, and relative low cost, of computing and storage resources, that can be readily obtained and easily adjusted to the current application workload. All of the above characteristics are a perfect match for the requirements of the transient and goal-oriented knowledge management application solutions.

The non-IT users of the knowledge management development tools should be able to design and implement fully functional knowledge management solutions comprising a repository of information objects organized according to a semantic model, providing the principal view of the repository information to the system users, as well as the process management functionality supporting execution of the knowledge workers' procedures and tasks.

The substantial impact of the end-user development is exemplified by data published by the US Bureau of Labour and Statistics in 2012, quoted in [24], showing that there have been in the United States fewer than 3 million professional programmers but more than 55 million people have been using spreadsheets and databases at work, many of whom write formulae and queries to support their job.

A significant challenge in involving non-IT professional developers creating complex application solutions, notwithstanding the scope of automated development tools support (e.g. application generating wizards), is the notorious lack of sound software engineering practices, such as quality assurance of implemented solutions, which often precludes sufficient reliability and robustness of the resulting applications.

Our research and development work in the area of the knowledge management software tools initiated within the ICONS FP5 research project [20] and further expanded within the eGovBus FP6 research project [14], as well as the ensuing engineering of the research results resulting in development of the OfficeObjects® knowledge management platform [37], provided us with the solid basis for design, construction, and implementation of agile end-user-oriented knowledge management application solutions.

OfficeObjects® is a proprietary JEE (Java Enterprise Edition) framework integrated with several specialized open source components supporting such functionality as the full text search, business intelligence and reporting, as well as the portal environment.

In the following sections we discuss the principle user requirements, defining the functional scope of the knowledge management software tools, and the underlying application development methodology, which had provided the guidelines for design and development of the OfficeObjects® knowledge management software tools, as well as the pertinent state-of-the-art research and development results.

Further we succinctly present the end-user-oriented development features of the OfficeObjects® architecture highlighting the strengths and challenges of the knowledge management software tools, and finally we present the end-user oriented development methodology.

2 The Knowledge Management Application Requirements

The challenges facing knowledge workers, particularly those having direct negative effect on their productivity, have been identified in the already mentioned study performed by Nathaniel Palmer [41] repeatedly in 2011 and 2013. Table 1 summarizes the results obtained in the 2013 survey, where column "%" provides the proportion of respondents giving the positive answer, and the remaining columns refer to the KMS feature areas, shown in Fig. 1, relevant to the corresponding challenge.

Table 1. Knowledge worker challenges vs. the KMS features.

Knowledge workers' challenge[a]	%	1	2	3	4	5	6
Lack of visibility into the current state or status of others' work supporting your own	71			**X**	**X**	**X**	**X**
Difficulty tracking "to do" items or task lists	45			**X**	**X**		
Difficulty organizing and assembling the right team	51	**X**	**X**	**X**			
Difficulty managing documentation and information needed for a given project	57		**X**	**X**			
Difficulty finding co-workers/collaborators with the right experience	53		**X**	**X**		**X**	**X**
Difficulty determining the next step or course of action	36			**X**	**X**		**X**

1. Enterprise 2.0 Ontology
2. Knowledge Representation
3. Content Repository
4. Workflow Process Management
5. Enterprise 2.0
6. Knowledge Integration

[a] [41]

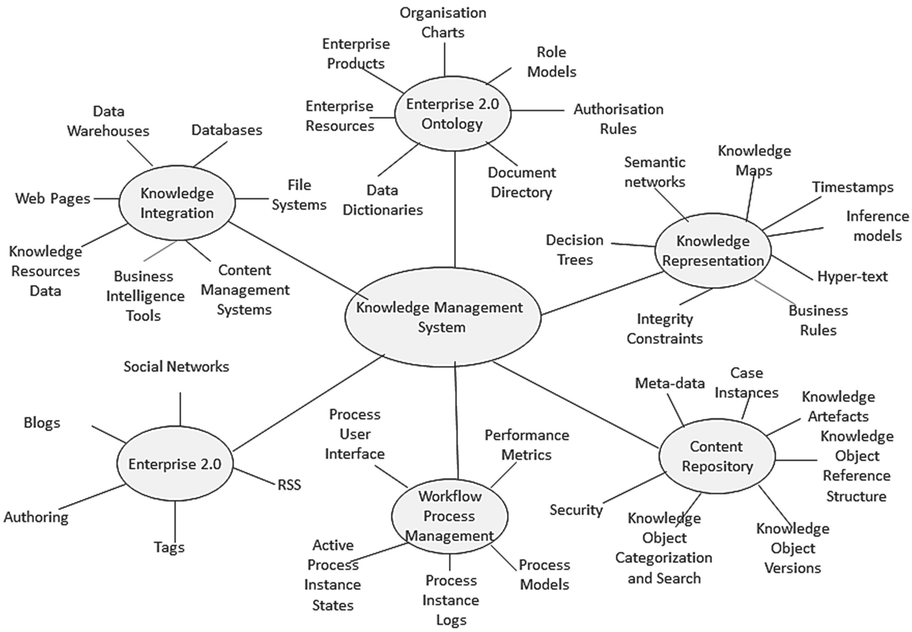

Fig. 1. Feature requirements of the knowledge management system.

The analysis results clearly indicate the importance of the “Content Repository” features providing means to alleviate obstructions impeding the knowledge worker productivity, immediately followed by such feature areas as “Workflow Process Management” and “Knowledge Representation”.

The KMS feature model has been introduced in [20], serving subsequently as the road map of the OfficeObjects® development project, undergoing revisions motivated by experience derived from a number of large scale knowledge management applications. Another important lesson learnt in the course of these application projects was the utmost importance of empowering the KMS end-users to ensure their active participation, not only in the user requirements analysis, but first of all in the KM solution development and maintenance processes.

The rapidly growing end-user software engineering (EUSE) field has also influenced the focus of the OfficeObjects® software architecture design to embrace the EUSE techniques and methodologies. The user-oriented assessment of the eGovernment service bus system [14] developed with the use of the OfficeObjects® platform, in particular of its service design and development tools, has shown that non-programming IT technicians were able to develop complex services published in the Web.

The ensuing development of the subsequent versions of the OfficeObjects® platform has been concentrated on the ergonomic aspects of end-user interfaces, both in the area of application solution development tools, and the functional system areas, such as the content repository, workflow process graphic interfaces, and the HCI features.

The existent and emerging software standards pertaining to the OfficeObjects® platform have been incorporated in the software design in order to facilitate high acceptance level of the end-users and IT professionals, as well as to support interoperability with information systems and data sources that may be integrated within the knowledge management application solutions.

3 The KMS Research Activities

The **architecture of knowledge management systems** is a field of intensive research and development effort. Notwithstanding the research and innovation currently under way, the comprehensive integrated end-user development tools supporting agile development of advanced KM application solutions are rarely meeting the advanced knowledge management system requirements. Apart from the OfficeObjects® platform [37], the closest example is a prototype of the knowledge management platform presented in [27]. Analogously to our approach, the above authors propose a distributed platform replicating functional components to achieve system scalability through the use of load balancing under the varying workload conditions. Also the virtual organizations, possibly involving several independent partners, are envisaged as the prime users of the proposed system. The system is supporting advanced content management solutions, but it does not provide application development tools oriented towards the end-user software engineering community. System security is a significant concern in knowledge management as well as in the generic collaborative systems, these issues are discussed at length in [47, 58] respectively.

The **End-User Software Engineering (EUSE)** field has been growing significantly over the last several years, evolving from the spreadsheet financial models, through the graphic user interface implementations, to the end-user developed mashup applications. The Service-Oriented Architecture (SOA), providing an integration platform for accessing domain-specific application environments, has enabled development of complex and robust applications by non-programmers.

It is the common believe that the knowledge management application design and specification tools are to provide an abstraction level concealing the underlying technological complexity of a KMS platform, thus enabling the end-user developer to concentrate on the application requirements of the KM solution. A comprehensive overview of current end-user development tools has been presented in [24]. The field has been growing considerably over the last several years and a number of important research initiatives have been published. A composition model facilitating the programming-illiterate knowledge workers to develop rich internet applications, integrating preexisting software components to be published in a graphic web interface (a mashup), has been presented in [30]. Other mashup frameworks bridging the perspective of the service based software development and the end-user development have also been presented in [35, 36].

Development of Web 2.0 tools and techniques has enabled end-users to move from content and personalization to functionality supported by the user-developed web services. A number of such projects, spanning from ambient intelligence, through to wizard-based process development, have been presented at the AVI Workshop held in Rome on May 25–29 2010 [9]. The use of design patterns in the end-user development projects has also been growing as presented in [61]. A good example of a design pattern repository is the MIT process library described in [31].

Semantic Knowledge Content Modelling, similar to the OfficeObjects® knowledge map approach, has been proposed in [12]. The platform, serving the cultural heritage applications, is a closed software system providing no development tools for the system users. The corporate knowledge management domain is represented by an advanced prototype of a knowledge management system SKMS (Smart Knowledge Management System) presented in [32]. The platform provides a powerful document structuring mechanism in the form of dynamic categorization trees, but similarly to the above solutions, it neither provides tools for specification of the knowledge management or scientific workflow processes, nor it allows for semantic modelling of the knowledge repository content.

Several KM systems currently under development are equipped with formal ontology models in the form of semantic nets, as represented by the Topic Maps ISO standard (ISO 13250), mostly supporting semantic browsing features referencing the repository and external information objects. An example of Topic Maps-based semantic net implementation is the DREAM platform presented in [4] utilized for semantic indexing and search of visual objects. Topic Maps are also used for categorization of documents on the basis of their meta-data attribute values. Examples of such architectures may be found in [6] as well as in [10, 42, 60].

The role of an ontology model in the knowledge management system has been extensively discussed in [11, 59]. It is generally agreed that an ontology specification language can be seen as a knowledge representation language, which should guarantee that every concrete ontology enjoys the following properties: (i) it is a surrogate for the things in the real world; (ii) it is a set of ontological commitments; and (iii) it is a medium for human expression. In other words, an ontology may be specified without any particular reasoning paradigm in mind, and it does not necessarily have to be a theory of representational constructs plus inferences it recommends, or a medium for efficient computation.

Many tailor-made ontology specification languages have been defined so far. In the context of the DARPA Knowledge Sharing Effort, for example, Gruber defined *Ontolingua* [17]. The language was developed as an ontology layer on top of KIF [16], which allowed frame style definition of knowledge representation models (such as classes, slots, and subclasses). Other languages, such as *Conceptual Graphs* [52, 59], have also been popular for specifying ontologies.

Recently, the XML-based W3C Web Ontology Language (OWL) [38, 59] has gained wide popularity. The language is characterized by very high expressiveness, but to get some guarantees with respect to computability, a user has to limit herself to a well-understood fragment of OWL, called OWL DL, based on Description Logics (DL) [2, 3, 7, 59].

The **Human Computer Interaction** field, enriched by ubiquity and growing computing power of mobile devices, such as smartphones and tablets, as well as the new mobile context-aware software standards exemplified by HTML5, offers significant opportunities for new intelligent applications based on knowledge management systems, such as the OfficeObjects® platform. Development of the graphic user interface, as well as configuring of the mobile device apps serving as clients, represents important challenges for the end-user KM application development. The field is rich with research projects concentrating on issues of automatic generation of mobile device graphic interfaces on the server side, as described in [8, 26], as well as the component-based end-user development of complex graphic interfaces integrating heterogeneous data sources and application functions, such as mashups described in [30, 35].

The Ambient Intelligence field is a growing application area to be supported by the end-user software development tools, like those available in the OfficeObjects® platform, either as a new solution development by parameterization of the existing design patterns, or as an application of the off-the-shelf components. Examples of such application solutions have been presented in [1, 28].

Workflow Management Platforms available as the cloud computing services are subject of many research efforts, and consequently quite widely published, in particular in the eScience area. Many projects concentrate on workflow tools and run-time platforms supporting scientific workflows moving vast amounts of data resulting from scientific experiments. Automation of data interchange is a subject of many publications in particular related to the field of HPC (High Performance Computing), among others interesting results are presented in [21, 49, 62, 63].

All of the presented system prototypes use the workflow management platforms as a middleware layer responsible for coordination of scientific computation tasks, providing facilities for parallel scheduling of complex computations, and passing intermediate result data among such computations. Ubiquity of these solutions in the scientific computation community bodes well for other application areas, such as among others the knowledge management field.

New workflow paradigms are being proposed in response to the growing need to support and measure efficiency of the knowledge work. Working methodologies, such as SCRUM for example, are becoming ubiquitous not only in the software development work. One of the significant proposals of the new workflow paradigm is the Role Model developed by Keith Harrison-Broninski [18, 19].

A set of lightweight methods called "agile" are being developed in recent years [37] to better fit the dynamic nature of projects and organizations. Agile methods adopt a dynamic process control model, which is meant for processes that are not always well defined and are sometimes unpredictable and unrepeatable.

A comprehensive discussion of the scientific workflow models is provided in [57] highlighting a number of issues that are still open. Among others according to D. Talia, the outstanding problems include (a) adaptive/dynamic workflow models. (b) service-oriented workflows in cloud computing infrastructures, and (c) workflow provenance and annotation mechanisms and systems.

Adaptive Case Management (ACM) is a fast growing area of management innovation, rather than the computer science research, fuelled by the widely believed constatation that the classic graph-oriented workflow models are incompatible with the nature of the knowledge work. A convincing proof is provided by the already presented results of the survey conducted by Nathaniel Palmer [41], as well as by explicit calls for the BPM paradigm shift in [5, 50, 56]. Additional argumentation, calling for a major overhaul of the presently available workflow process and content management architectures, may be found in [33, 34, 39, 40, 45, 46, 53–55]. Another important line of thought, discussed in [22, 23, 25], is the data orientation of the ACM platforms considering the rich knowledge object repository structures and the semantic modelling to be the principal support vehicles of the knowledge work. Indeed for a growing engineering field anchored in purely practical issues, the intensity of general interest, exemplified by the number of publications, is astonishing. In fact, this vouches for the real practical impact of the knowledge worker efficiency issues, as stated by Peter Drucker at the turn of the 20th century [13].

The ACM field, notwithstanding its practical flavor, attracted the attention of the computer science research community approaching the existing issues from a theoretical vantage point. One of such projects, initiated at the Sorbonne University in Paris has been presented in [48].

4 The OfficeObjects® KM Architecture

The OfficeObjects® software architecture, presented in Fig. 2, has been evolving over the last 4 years to provide the comprehensive set of features required for the knowledge management application development. As we stressed in the preceding discussion, the end-user orientation has been the major focus of our design and development effort. The presented software architecture meets the application requirements included in the knowledge management feature model shown in Fig. 1.

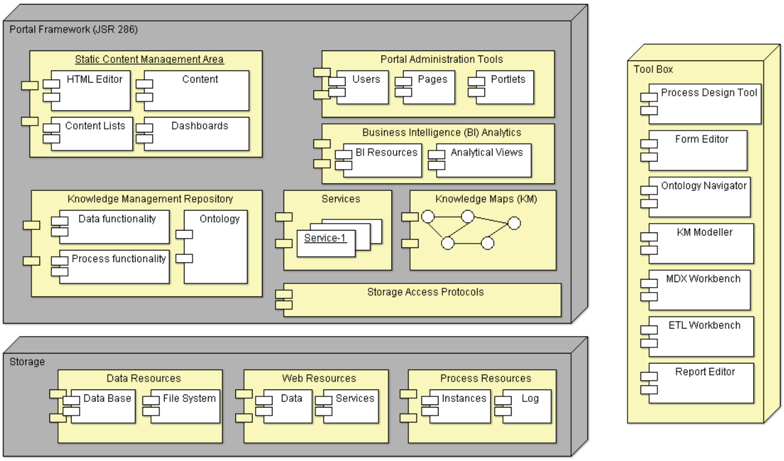

Fig. 2. OfficeObjects® platform architecture.

The OfficeObjects® functional modules are deployed within three principal packages installed in the virtualized processing environment. The user-visible functionality, representing the application solutions, is deployed within the **JSR 286 Portal Framework** [29] providing a rich and mature environment for the end-user-oriented mush up application development.

A rich and extensible library of portlets supports the state-of-the-art Enterpise 2.0 solutions packaged within the **Static Content Management Area**. The portal administration tools are available within the **Portal Administration Tools** pages. Both functional areas render themselves readily for the end-user software development, which is usually based on the use of assorted web applications.

The knowledge management functionalities, comprising the OfficeObjects® components, as well as the integrated open source software components, such as, among others, the community TIBCO Jaspersoft report server incorporating the Mondrane ROLAP engine [43] executing the Multidimensional Expressions (MDX) analytical language [51]. The above functionalities may be deployed as portlets, depending on the knowledge management solution requirements, respectively within the **Knowledge Management Repository** and the **Business Intelligence (BI) Analytics** areas.

The Knowledge Management Repository publishes all OfficeObjects® services dedicated to content, process and ontology management. An important knowledge management tool the **Knowledge Maps,** based on the Topic Maps ISO 12350 standard, supports creation and delivery of semantic models, superimposed on the knowledge repository content, providing semantically enriched knowledge artefact navigation and selection functionality. A knowledge map may comprise references to the repository information objects as well as to the external information objects, such as web pages, Wikipedia entries, database queries etc. The knowledge maps and the dynamic object categorization trees used in advanced knowledge management systems prove to be intuitive and user-friendly.

The KMS features concerned with the integration of the external knowledge resources, data, and services are supported by the OfficeObjects® Service Broker module facilitating deployment of complex services within the Portal Framework developed with the use of OfficeObjects® tools and deployed in the OfficeObjects® Work-Flow platform.

The **Ontology** model, supported by the Topic Maps Ontology Navigator, comprises all information concerning the KMS user environment, such as the organization structure, user accounts and access rights, role models, etc., as well as the semantic model features comprising controlled vocabularies, data dictionaries, information object class specifications, and the knowledge map definitions.

All of the above components of the run-time OfficeObjects® architecture are supported by the **OfficeObjects® Tool Box** providing design and development functions for the users specifying a knowledge management application solution. The **Process Design Tool** coupled with the **Form Editor** provide tools to specify the workflow process BPMN model and the corresponding process GUI. The **Knowledge Maps (KM) Modeller** may be based on any available UML Class Diagram tool exporting the XMI notation to be subsequently processed by the OfficeObjects® Ontology Manager module and mapped onto the ontology structure to form a Knowledge Maps definition.

The scope of design specifications supported by the Tool Box components becomes apparent in the context of the design decision trees, discussed in Sect. 5.

The **MDX Workbench,** the **Extract-Transform-Load (ETL) Workbench**, and the **Report Editor**, are used to develop data marts, and the associated ROLAP models, within the integrated **Business Intelligence** solution. Although, all of these tools require data analysis skills, they may be used by no-IT personnel, hence they fall into the broad class of the EUSE tools.

The underlying data **Storage** package represents systems and facilities, such as data base management systems, file systems, web services, and web pages, that may be referenced to select and retrieve information objects accessible via the Knowledge Management Repository reference structures.

The **workflow process instances** managed by the OfficeObjects® WorkFlow platform are stored in a WfMC run-time meta-model format. Event data resulting from execution of workflow process instances are recorded in the form of process logs, which subsequently may be used to generate process execution reports and ROLAP models. The workflow process definitions are available via the OfficeObjects® Process Design Tool and may be exported/imported with the use of the WfMC XPDL notation.

The OfficeObjects® Repository data model is presented in Fig. 3 as the UML class diagram of information resources coupled with a set of interfaces representing the repository referential structure. The repository contains instances of information object classes, where an object may belong to only one object class determined by the meta-data model. The physical structure of an information object instance, i.e. the number, size and type of binary artefacts (files), stored in an object, is completely arbitrary, thus independent of the corresponding information object class.

Semantics of the KM repository are dependent on its referential structure, i.e. on information object classification and assignment to respective object collections. The classification and assignment actions are subdivided into three principal modes, namely the **Automatic** mode, the **Manual** mode, and the **Knowledge Map** mode. The last variant may be considered a variation of the Automatic mode.

The automatic collection represents the following object collection semantics; (a) Full Text Retrieval pertain to the entire population of all information classes automatically indexed and made eligible for retrieval on the basis of their textual content, (b) the remaining three automatic collections, i.e. the **Categorization Tree**, the **Meta-data Search**, and the **Register**, pertain to the population of a single object class only. The categorization trees support a hierarchical access path to information objects selected on the basis of the meta-data attribute values, and the registers are a chronological ordering of objects within the corresponding class or a subordinated sub-class defined by a selection predicate referencing the meta-data attributes.

The manual collections, such as the case files or repository folders, represent a manual, information-bearing classification process, since most often the allocation activities may not be reproduced on the basis of the meta-data values. In fact, the allocation decisions are implemented by the direct user actions. However, in some applications it may be possible to perform such allocations automatically, if appropriate information, such as for example the case file identifier, are present in the meta-data of the information object to be categorized.

The knowledge map is constructed and maintained automatically, controlled by the construction rules, defined on the meta-data attributes, and by the appropriate mapping rules. The mapping rules decide, which meta-data attributes are to be represented in the corresponding knowledge map topics (nodes), and the construction rules determine the

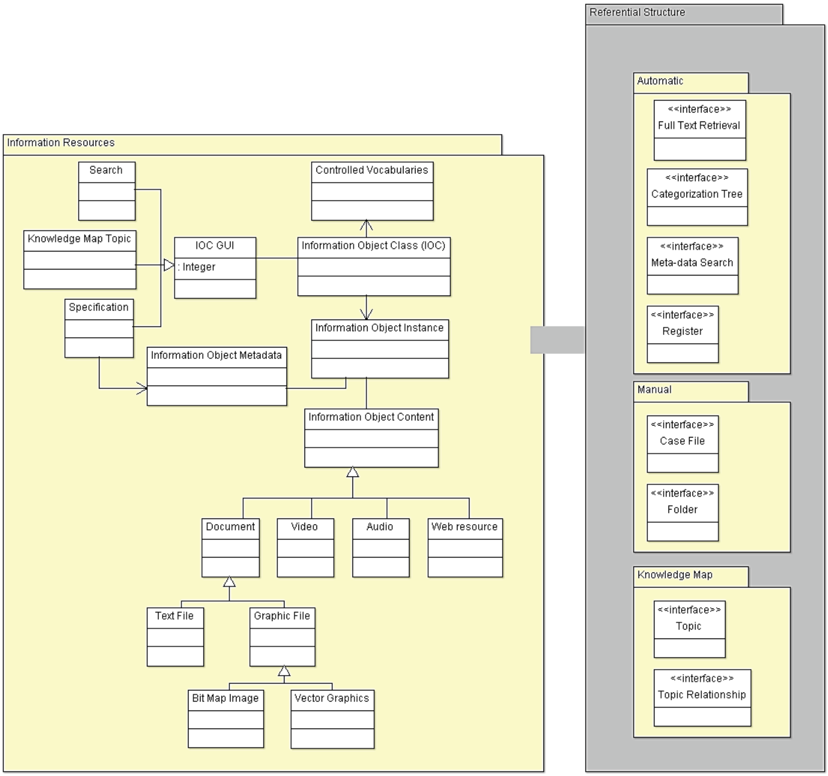

Fig. 3. OfficeObjects® repository data model.

relationships maintained among the knowledge map topics, thus establishing the required traversal path within the map.

5 The KMS Solution End-User Specification Methodology

We have selected two knowledge management application design and specification areas to illustrate the merits and limitations of the OfficeObjects® application development tools, in particular their eligibility for the end-user. We need to make a reservation, that we expect the computer literacy of the end-user system developer, often such a role being called the power-user, at least on the level of an expert spreadsheet user or a personal database user. As we mentioned before, such qualifications are ubiquitous among the professionals using computers for their work.

We concentrate on two principal design areas of the knowledge management system functional spectrum, namely on the knowledge repository and on the workflow management platform, shown in Figs. 4 and 5 respectively. A convention used in both mind maps is the **X** symbol meaning that the decision branch and all descending children are ineligible for the end-user, due to their complexity calling for the professional IT skills.

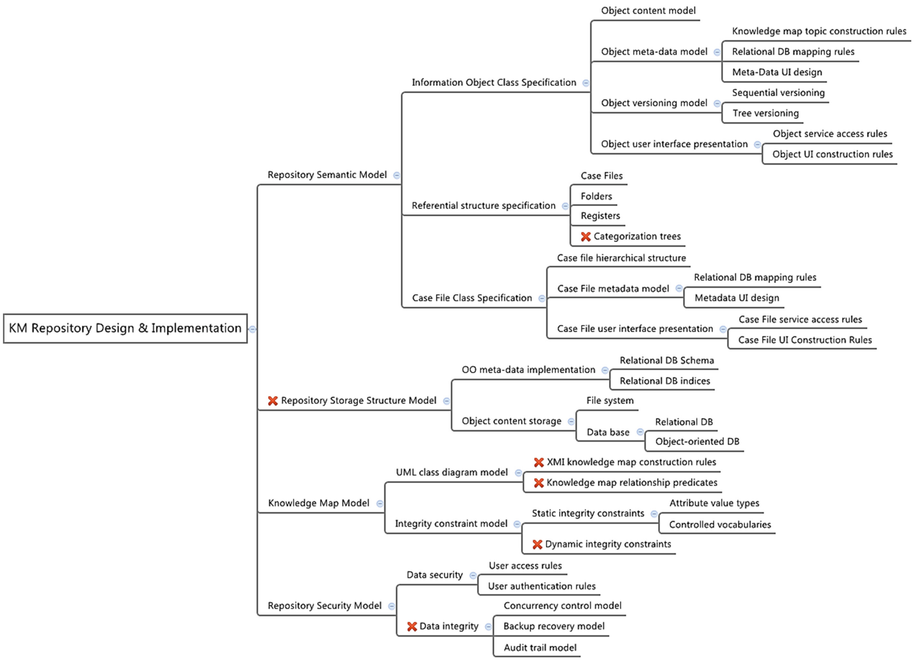

Fig. 4. OfficeObjects® repository specification decision tree.

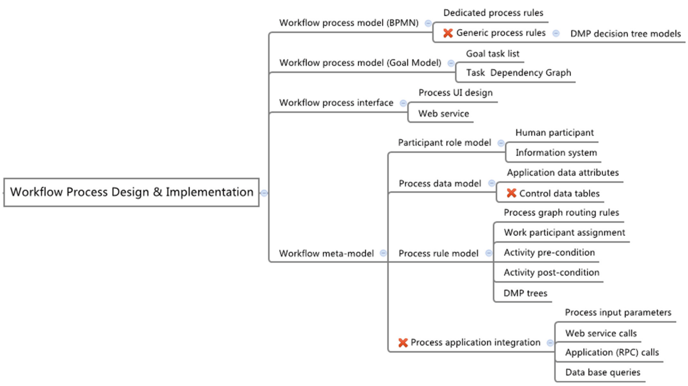

Fig. 5. OfficeObjects® Workflow specification decision tree.

The **Repository Semantic Level** includes all design decisions, either pertaining to the conceptual model of the repository knowledge resources, or to the underlying data structure specifications providing the building blocks for the higher level constructs, such as the meta-data specifications of an information object class. Design specification, which we believe might be too complex for the non-programming user, are the categorization tree materialization queries, since they require advanced SQL operations such as the JOIN and GROUP BY queries.

All of the other design specifications pertaining to the semantic modelling of the knowledge resources, such as the automatic assignment predicates aligning information objects within the target referential objects, such as **Registers** and **Case Files**, are well within the grasp of the power-user. All in all, it is quite possible, that the power-users define a complex repository data model, albeit some OfficeObjects® methodology and tools training is advisable.

On the other hand, definition of the **Repository Storage Structure Model** requires decisions calling for the specialized data management skills, hence it usually rests beyond capabilities of even advanced power-users. The solution here is to apply default physical data structure configurations, pre-configured in the software distribution version, offering good performance support for typical repository usage patterns.

The **Knowledge Maps Model** is a critical feature of the most of the knowledge management applications supporting semantic views over the information objects stored in the knowledge repository. Superimposing an UML class diagram model over the Topic Maps ontology, and maintaining references between topics and information objects, allows the repository user to select and manipulate the knowledge resources, i.e. the information objects, according to a domain-oriented semantic data model. Navigation in the network of binary topic relationships, linking internal and external knowledge artefacts, constitutes a powerful search platform guiding navigation along the associative selection paths.

The knowledge map design may proceed in a "top down" manner, starting from the UML class diagram referencing the information object classes and linking them within appropriate relationships, or using a "bottom up" method, defining the topic relationships and the associated relationship predicates directly using the Topic Maps formalism. The latter method may not be advisable for the power-users.

The recommended design methodology is to define the UML class diagram of a knowledge map, tag the relationships with the selected association predicates defined over meta-data attributes of the associated classes, and to automatically generate the Topic Maps specifications via the XMI interface.

We also assume that both **Dynamic integrity constraints** as well as **Data integrity** rules and procedures may be too complex for a non-IT professional and they will usually require help from the system administration staff. Notwithstanding the above limitations, we may safely claim that a working knowledge management repository may be designed, specified and maintained by non-IT professionals possibly supported by the system familiarization rudimentary training.

The second important design realm of the knowledge management application solution implementation is the **Workflow Process Design & Implementation** area. The scope of design decisions facing the system designer is depicted in Fig. 5. Most of the application specification tools, such as the process graph specification, the graphic user interface form editor, the functional rule specification language, and the process participant role model, have proven to be sufficiently user friendly to be productively employed by the power-users.

We find that specifying generic workflow models, employing the dynamic process modification features [14], may exceed the capabilities of the power-user. On the other hand, parameterizing such processes, available in the process pattern library, is quite straightforward and may readily be performed by the users.

In order to address the requirements identified in Sect. 2, rather than utilizing the BPMN graphical process model, one may specify the **Goal Model** workflow process [37] much more suitable for planning and executing the project-oriented activities. The Goal Model processes are specified as the check list of all process tasks, the participant assignment rules for each task, and the dependency graph representing the precedence relationships among tasks. Task execution is scheduled only for tasks that are not bound by any precedence relationship.

The process goal is met when all tasks have been executed. Such process specification and maintenance tasks as interpreting the process **control data tables** comprising the workflow process run-time meta-models, for diagnostic and performance-oriented process design purposes, may require assistance from the process administration staff. Also the **process application integration** specifications, may either require the power-users to undergo substantial training, or collaboration with the process administration staff.

6 Conclusions

The end-user oriented methodology underlying development of the knowledge management application solutions has been verified in the course of a number of application projects. Among others, a large-scale knowledge management application system had been implemented in the period of 2010–2012 serving a community of 2000 scientists working for 20 research organizations.

The knowledge management system is currently used as a networking tool to support co-operation of industrial organizations and research institutes according the recommendations of the Open Innovation model.

The platform, which serves as a tool supporting communication and cooperation, as well as providing information pertaining to the resources and skills possessed by the participating organizations, facilitates their co-operation and the dissemination of best practices in the area of the research work and management.

The lessons learnt during design and development of the above system confirm, that all major application functions were indeed developed without the recourse to classic application programming languages, such as Java or C ++. The only hurdle to overcome by the non-programming developers were the Java Script validation codes. Although the power-users were successfully involved in the system development effort, provision of sufficiently thorough training materials, as well as of the technical help available online could significantly improve the implementation process.

References

1. Aggarwal, J.K., Ryoo, M.S.: Human activity analysis: a review. ACM Comput. Surv. **43**(3) (2011)
2. Baader, F., Sattler, U.: An overview of tableau algorithms for description logics. Stud. Logica **69**(1), 5–40 (2001)
3. Baader, F., Calvanese, D., McGuinness, D.L., Nardi, D., Patel-Schneider, P.F. (eds.): The Description Logic Handbook: Theory, Implementation, and Applications. Cambridge University Press, Cambridge (2003)

4. Badii, A., Chattun, L., Meng, Z., Crouch, M.: The DREAM framework: using a network of scalable ontologies for intelligent indexing and retrieval of visual content. In: Proceedings of the International Conference on Web Intelligence and Intelligent Agent Technology. IEEE (2009)
5. Bider, I., Johannesson, P., Perjons, E.: Justifying ACM: why we need a paradigm shift in BPM. In: Fischer, L. (ed.) Empowering Knowledge Workers – New Ways to Leverage Case Management. Future Strategies Inc., New York (2014)
6. Cahier, J-P., Ma, X., Zaher, L.: Document and item-based modeling; a hybrid method for a socio-semantic web. In: Proceedings of the ACM DocEng Conference, Manchester, UK, 21–24 September 2010
7. Calvanese, D., De Giacomo, G., Lenzerini, M., Nardi, D.: Reasoning in expressive description logics. In: Robinson, A., Voronkov, A. (eds.) Handbook of Automated Reasoning. Elsevier Science Publishers, Amsterdam, Chap. 23 (2001)
8. Wiza, W., Walczak, K., Chmielewski, J.: Mobile interfaces for building control surveyors. In: Cellary, W., Estevez, E. (eds.) Software Services for e-World. IFIP AICT, vol. 341, pp. 29–39. Springer, Heidelberg (2010)
9. Costabile, M.F., De Ruyter, B., Mehandjiev, N., Mussio, P.: End-user development of software services and applications. In: AVI 2010, Rome, 25–29 May 2010
10. Damen, D., Pinchuk, R., Fontaine, B.: Creating topic maps ontologies for space experiments. In: Proceedings of the International Conference on Topic Maps Research and Applications (TMRA) (2009)
11. Davis, R., Shrobe, H., Szolovits, P.: What is a knowledge representation? AI Mag. **14**(1), 17–33 (1993)
12. Doerr, M., Iorizzo, D.: The Dream of a global knowledge network – a new approach. ACM J. Comput. Cult. Herit. **1**(1), 1–23 (2008)
13. Drucker, P.F.: Management Challenges for the 21st Century. Elsevier Ltd., Oxford (1999)
14. The Administrative Process Generator Design report, FP6-IST-2004-26727, Advanced eGovernment Information Service Bus (eGovBus), March 2006
15. User-oriented assessment of the eGovBus prototype, FP6-IST-2004-26727, Advanced eGovernment Information Service Bus (eGovBus), December 2008
16. Ginsberg, M.L.: Knowledge interchange format: the KIF of death. AI Mag. **12**(33), 57–63 (1991)
17. Gruber, T.R.: A translation approach to portable ontology specifications. Knowl. Acquis. **5**, 99–220 (1993)
18. Harrison-Broninski, K.D.: Human Interactions – The Heart and Soul of Business Process Management. Meghan-Kiffer Press, Tampa (2005)
19. Harrison-Broninski, K.: Types of business process. In: Fischer, L. (ed.) How Knowledge Workers get Things Done-Real World Adaptive Case Management. Future Strategies Inc., New York (2012)
20. Intelligent Content Management System (ICONS) Project Presentation, FP5 IST-2001-32429 (2002)
21. Juve, G., Deelman, E.: Scientific workflows and clouds. Crossroads Plugging Cloud **16**(3), 14–18 (2010). http://www.acm.org/crossroads
22. Khoyi, D.: Data orientation. In: Swenson, K.D. (ed.) Mastering the Unpredictable. Future Strategies Inc., New York (2010)
23. Khoyi, D., Swenson, K.D.: Templates, not programs. In: Swenson, K.D. (ed.) Mastering the Unpredictable. Future Strategies Inc., New York (2010)
24. Ko, A.J., et al.: The state of the art in end-user software engineering. ACM Comput. Surv. **43**(3), 44 (2011)

25. Kraft, F.M.: Improving knowledge work. In: Swenson, K.D. (ed.) Mastering the Unpredictable. Future Strategies Inc., New York (2010)
26. Lakshman, T.K., Thuijs, X.: Enhancing enterprise field productivity via cross platform mobile cloud apps. In: Proceedings of the MCS 2011, Bethesda, Maryland, USA, 28 June 2011
27. Langenberg, D., Kind, C., Darnes, M.: The KnowledgeCloud Project, BMWi, Germany (2011)
28. Lee, Y., et al.: MobiCon: a mobile context-monitoring platform. Comm. ACM **55**(3), 54–65 (2012)
29. Sezov, R.: Liferay, Portal Administrator's Guide. Liferay Inc., Los Angeles (2009)
30. Lizcano, D., Alonso, F., Soriano, J., Lopez, G.: A new end-user composition model to empower knowledge workers to develop rich internet applications. J. Web Eng. **10**(3), 197–233 (2011)
31. Malone, T.W., Crowston, K., et al.: Organizing Business Knowledge: The MIT Process Handbook. MIT Press, Cambridge (2003)
32. Mancilla-Amaya, L., Sanin, C., Szczerbicki, E.: The e-decisional community: an integrated knowledge sharing platform. In: Proceedings of the 7th Asia-Pacific Conference on Conceptual Modelling (APCCM 2010), Brisbane, Australia (2010)
33. Matthias, J.T.: User requirements for a new generation of case management systems. In: Fischer, L. (ed.) Taming the Unpredictable – Real World Adaptive Case Management: Case Studies and Practical Guidance. Future Strategies Inc., New York (2011)
34. McCauley, D.: Achieving agility. In: Swenson, K.D. (ed.) Mastering the Unpredictable. Future Strategies Inc., New York (2010)
35. Mehandjiev, N., De Angeli, A.: End-user mashups – analytical framework. In: WAS4FI 2012, Bertinoro, Italy, 19 September 2012
36. Nestler, T., Namoun, A., Schill, A.: End-user development of service-based interactive web applications at the presentation layer. In: EICS 2011, Pisa, 13–16 June 2011
37. OfficeObjects® Software Product, Rodan Systems S.A., Warsaw, Poland (2010). www.rodan.pl
38. W3C, The XML-based Web Ontology Language. http://www.w3.org/TR/owl-features/
39. Palmer, N.: BPM and ACM. In: Fischer, L. (ed.) Taming the Unpredictable – Real World Adaptive Case Management: Case Studies and Practical Guidance. Future Strategies Inc., New York (2011)
40. Palmer, N.: Case management megatrends. In: Fischer, L. (ed.) How Knowledge Workers Get Things Done – Real World Adaptive Case Management. Future Strategies Inc., New York (2012)
41. Palmer, N.: Where is ACM today? realities and opportunities. In: Fischer, L. (ed.) Empowering Knowledge Workers – New Ways to Leverage Case Management. Future Strategies Inc., New York (2014)
42. Park, J.: Topic maps, dashboards and sensemaking. In: Proceedings of the International Conference on Topic Maps Research and Applications (TMRA) (2008)
43. Mondrian 3.0.4 Technical Guide, Pentaho Org., USA (2009)
44. Pepper, S.: The TAO of topic maps: finding the way in the age of infoglut (2009). http://www.ontopia.net/topicmaps/materials/tao.html
45. Pucher, M.J.: The elements of adaptive case management. In: Swenson, K.D. (ed.) Mastering the Unpredictable. Future Strategies Inc., New York (2010)
46. Pucher, M.J.: The strategic business benefits of adaptive case management. In: Fischer, L. (ed.) How Knowledge Workers Get Things Done – Real World Adaptive Case Management. Future Strategies Inc., New York (2012)

47. Ruiz, C., Alvaro, G., Gomez-Perez, J.: A framework and implementation for secure knowledge management in large communities. The ACTIVE project website (2011). http://www.active-project.eu/
48. Rychkova, I., Kirsch-Pinheiro, M., Le Grand, B.: Automated guidance for case management: science or fiction? In: Fischer, L. (ed.) Empowering Knowledge Workers – New Ways to Leverage Case Management. Future Strategies Inc., New York (2014)
49. Shams, K.S., Powell, M.W., Crockett, T.M., Norris, J.S., Rossi, R., Soderstrom, T.: Polyphony: a workflow orchestration framework for cloud computing. In: Proceedings of the 2010 10th IEEE/ACM International Conference on Cluster, Cloud and Grid Computing. IEEE, New York (2010)
50. Silver, B.: Case management: addressing unique BPM requirements. In: Fischer, L. (ed.) Taming the Unpredictable – Real World Adaptive Case Management: Case Studies and Practical Guidance. Future Strategies Inc., New York (2011)
51. Spofford, G.: MDX Solutions. Wiley, New York (2001)
52. Sowa, J.F.: Conceptual graphs for a database interface. IBM J. Res. Dev. **20**(4), 336–357 (1976)
53. Swenson, K.D.: The nature of knowledge work. In: Swenson, K.D. (ed.) Mastering the Unpredictable. Future Strategies Inc., New York (2010)
54. Swenson, K.D.: Advantages of agile BPM. In: Fischer, L. (ed.) Taming the Unpredictable – Real World Adaptive Case Management: Case Studies and Practical Guidance. Future Strategies Inc., New York (2011)
55. Swenson, K.D.: Case management: contrasting production vs. adaptive. In: Fischer, L. (ed.) How Knowledge Workers Get Things Done – Real World Adaptive Case Management. Future Strategies Inc., New York (2012)
56. Swenson, K.D.: Innovative organizations act like systems, not machines. In: Fischer, L. (ed.) Empowering Knowledge Workers – New Ways to Leverage Case Management. Future Strategies Inc., New York (2014)
57. Talia, D.: Workflow systems for science: concepts and tools. ISRN Softw. Eng. (2013). doi: 10.1155/2013/404525
58. Tolone, W., Ahn, G.J., Tanusree, P.: Access control in colaborative systems. ACM Comput. Surv. **37**(1), 29–41 (2005)
59. van Harmelen, F., van Harmelen, F., Lifschitz, V., Porter, B.: Handbook of Knowledge Representation. Elsevier Science, San Diego (2007)
60. Vatant, B.: Managing complex environments with topic maps. In: Proceedings of the Knowledge Technologies Conference 2001, Austin, USA (2001)
61. Verginadis, Y., Papageorgiou, N., Apostolou, D., Mentzas, G.: A review of patterns in collaborative work. In: Proceedings of the ACM GROUP 2010 Conference, Sanibel Island, Florida, USA, 7–10 November 2010
62. Vockler, J-S., Juve, G., Deelman, E., Rynge, M., Berriman, G.B.: Experiences using cloud for a scientific workflow application. In: Proceedings of the ACM ScienceCloud 2011 Conference, San Jose, California, USA, 8 June 2011
63. Zinn, D., McPhillips, T., Ludascher, B., Simmhan, Y., Giakkoupis, M., Prasanna, V.K.: Towards reliable, performant workflows for streaming-applications on cloud platforms. In: Proceedings of IEEE/ACM International Symposium on Cluster, Cloud and Grid Computing. IEEE, New York (2011)