Knowledge Management Architecture – Principles and Tendencies

Marian Pompiliu CRISTESCU
"Lucian Blaga" University of Sibiu, România
Corina Ioana CRISTESCU
Faculty of C.S.I.E., Academy of Economic Studies, Bucharest, România
marian.cristescu@ulbsibiu.ro ci_cristescu@yahoo.com

Algorithmic research is an established knowledge engineering process that has allowed researchers to identify new or significant problems, to better understand existing approaches and experimental results, and to obtain new, effective and efficient solutions. While algorithmic researchers regularly contribute to this knowledge base by proposing new problems and novel solutions, the processes currently used to share this knowledge are inefficient, resulting in unproductive overhead. Most of these publication-centered processes lack explicit high-level knowledge structures to support efficient knowledge management. The authors describe a problem-centered collaborative knowledge management architecture associated with Computational Problem Solving (CPS).

Keywords: *Knowledge Management Architecture, algorithmic research, ontology, Knowledge-Based Systems*

1 Introduction

Problem Solving Environments (PSEs) are platforms that provide all of the computational facilities required to solve a target class of problems. PSE features include advanced solution methods, the automatic and semi-automatic selection of solution methods, and easy means for incorporating novel solution methods.

There are some successful single-user mathematical PSEs that have been created, including Matlab, MathCAD, Maple, and Mathematica. Researchers are increasingly focusing on the development of collaborative PSEs [1], an extension that enhances human intelligence (HI) by providing a communication infrastructure that encourages collaborative (even synchronous) problem solving among individuals in geographically distributed locations. Despite having a wide variety of domain-specific features, strengths of PSEs can be measured in terms of the following levels of collaboration: data sharing, software warehousing, application sharing, and workflow.

In contrast, Knowledge-Based Systems (KBS) focus on the capture, formalization and application of strong domain knowledge. Two promising KBS candidates are ontologies (concerned with the representation of static domain knowledge) and problem solving methods (PSM) (aimed at describing KBS reasoning processes in a manner that is both implementation- and domain-independent). PSMs are also associated with the dynamic reasoning of knowledge [3]. If a KBS supports PSM, then the KBS possesses the capabilities of a PSE. Accordingly, it is possible to measure the KBS's strength using the collaboration levels mentioned above. In addition to adding a certain level of automation to the organization of the global knowledge structure, a PSE (with knowledge engineering support) can assist in problem solving efforts by providing a greater amount of advanced Machine Intelligence (MI).

2. CPS Ontology

Ontologies serve as explicit specification of domain concepts and their relations. Standards for describing ontologies include Topic Maps, RDF/RDFS and DAML+OIL. The list of authoring tools for modeling domain knowledge includes Protégé 2000, OntoEdit and OilED. A complete overview of ontological concepts and issues can be found in [2].

We propose using three conceptual spaces—problem, solution, and implementation—to provide a detailed and accurate description of CPS ontology.

Problem space objects consist of uniquely identifiable computational problems, solution space objects consist of algorithmic solutions, and implementation space objects assist in carrying out the solutions. A high-level CPS KB abstraction can be modeled as a collection of these objects, their intra-relations within each space, and their inter-relations across the three spaces.

As shown in Fig. 1, equivalent problems are grouped together, sub-problems, super-problems, and variant problems are also indicated. A computational problem is associated with the following attributes: problem name, description, problem category, equivalent problems, sub-problems, super-problems, variant problems, formal definition, input variables, output variables, output measure, problem status, existing solutions, and related publications. An algorithmic solution is associated with the following attributes: solution name, target problem, description, pseudo code, complexity, problem solving strategy, existing implementations, and related publications.

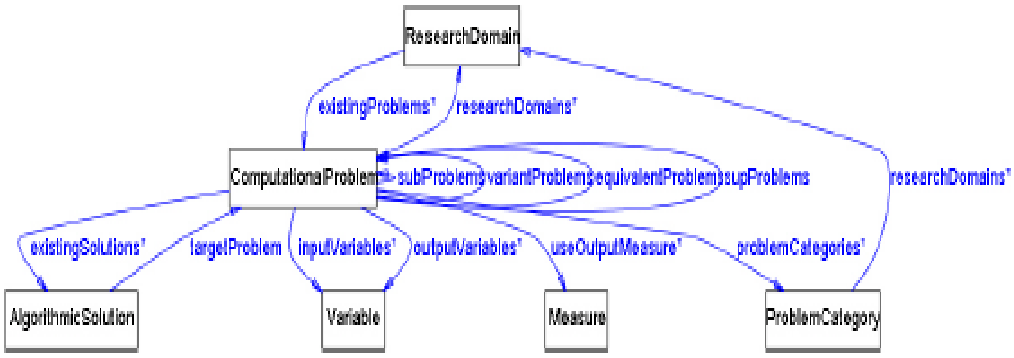

Fig.1. Conceptual Map of CPS

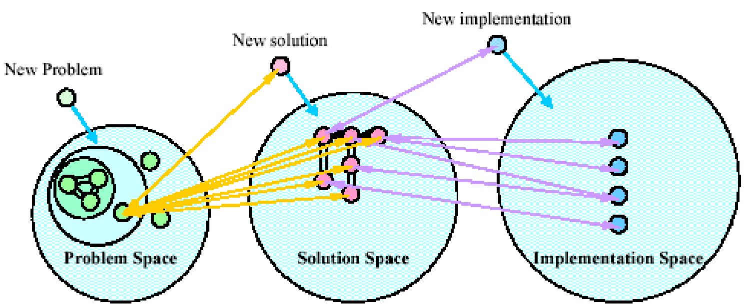

Fig.2. Ontology of CPS

Finally, an implementation is associated with the following attributes: implementation name, target solution, description, environment, offline execution, online execution, programming language, and related publications. We then model these concepts and describe the cross-relationships among OpenCPS knowledge objects using an RDF schema. A partial visualization generated by Protégé 2000 is shown in Fig. 2.

3. Determining KM Architecture

Following guidelines for running a KM project or creating a KM portal as described by [1] and [3], we propose a KM architecture that consists of the following four parts (see Fig. 3):

1. People, specifically those who produce and use knowledge objects. Member roles include administrator, knowledge author, knowledge reviewer, and technology designer. If no active knowledge authors can be found, it should be considered a warning sign that the KM initiative in question is failing.

2. Knowledge objects, meaning sharable information based on extracted knowledge structures. The three knowledge object types in OpenCPS are computational problems, al-

gorithmic solutions, and implementations.
3. Technical infrastructure. Technology enables the capture, storage, and delivery of content at the user's discretion. The costs associated with constructing, maintaining, and improving technological interfaces are a key issue for enablers. A thorough description of what to consider when choosing a CMS can be found in [2].

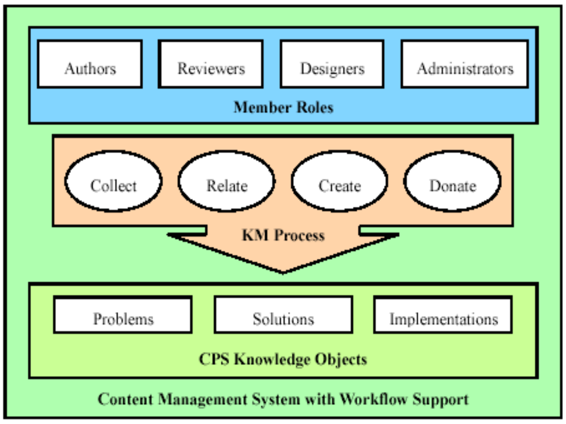

Fig.3. KM Architecture of OpenCPS

4. Knowledge management processes. The KM life cycle models shown in Table 1 [4] can be simplified for our proposed KM architecture for two reasons: we already extracted the well-formed CPS knowledge structure, and content management was delegated to the workflow-enabled CMS. When investigating the essential components of human interaction and creativity, we discovered that [Hneiderman 00] has proposed four core concepts: a) new knowledge is built on previous knowledge, b) powerful tools can support creativity, c) refinement is a social process, and d) creative work is not complete until it is disseminated. The resulting four-phase "generate excellence (Genex)" framework—collect, relate, create, and donate—provides a perspective that can be applied in the form of non-linear knowledge management processes in our work.

Table 1. Knowledge Management Life Cycle Modeless

Model	Phase 1	Phase 2	Phase 3	Phase 4	Phase 5	Phase 6
Despres and Chauvel	Create	Map/bundle	Store	Share/transfer	Reuse	Evolve
Gartner Group	Create	Organize	Capture	Access	Use	
Davenport & Prusak	Generate	Codify	Transfer			
Nissen	Capture	Organize	Formalize	Distribute	Apply	
Amalgamated	Create	Organize	Formalize	Distribute	Apply	Evolve

4. Measuring the Knowledge Flow

To support research of large scale or hard-core problem solving over a long time period, we can highlight important CPS objects to attract the attention of researchers and new participants. By adopting the commonly accepted axiom - the more valuable the information, the greater its access rate, we established a means of measuring knowledge flow—that is, the process of sharing knowledge among people or knowledge processing mechanisms [4], [5].
Knowledge flow measurement can help identify important topics within a collaborative knowledge portal. Some simple measures are provided by associating the actions of knowledge workers according to the four knowledge management processes-collect, relate, create and donate, which are highly related to human creativity.

5. Conclusions

In this paper we have proposed a problem-centered collaborative knowledge management architecture that differs greatly from publication-centered approaches [6], [7]. The OpenCPS knowledge portal uses three conceptual spaces to create formal knowledge objects that define the frontiers of CPS research domains. The problem space, which corresponds to well-defined computational problems, is the heart of the CPS research domain.
We summarize the following four advantages of our approach:
1. Both knowledge objects (formalized) and content objects (free-formatted) are by default publicly accessible to portal users. Us-

ers can employ these objects to create personal research surveys in the form of an online document. The more profuse the content, the more effective these collect and relate processes become.

2. Participants will be encouraged to donate personal knowledge or other portal content objects that will help illuminate a research domain with better materials than any single researchers could ever generate on their own. Donating the most recent results or surveys and other research efforts will make it easier to locate unsolved computational problems or to make use of existing algorithmic solutions.

3. Knowledge objects that represent and visualize solution implementations can be used to mediate interactions between users and conceptual algorithmic solutions. They are useful for both academic and corporate/industrial applications.

4. Access rights and workflows are enforced for content objects, so to allow a personal workspace for each user, which can be used to store content objects and to exchange information with other members.

Finally, the implementation space is a practical view of the CPS research domain that corresponds to existing implementations. CPS researchers can search a problem space to see if a computational problem is well-defined, if algorithmic solutions are available, or if there is a need for a new computational problem object and the potential for a collaborative effort to create it.

References:

[1] - Applehans, W., Globe, A., and Laugero, G.: “Managing Knowledge: A Practical Web-Based Approach (Addison-Wesley Information Technology Series),” ISBN 0-21-43315-X, 1998;

[2] - Chin Jr., G., Leung, L. R., Schuchardt, K., Gracio, D: “New Paradigms in Problem Solving Environments for Scientific Computing,” IUI2002, San Francisco, California, USA, 39-46, 2002;

[3] - Hai, Zhuge: “A Knowledge Flow Model for Peer-to-Peer Team Knowledge Sharing and Management,” Expert Systems with Applications 23, 23-30, 2002;

[4] - Lawrence, S., Giles, C. L., Bollacker, K.: “Digital Libraries and Autonomous Citation Indexing,” IEEE Computer, 32, 6, 67-71, 1999;

[5] - Mack, R., Ravin, Y., Byrd, R. J.: “Knowledge portals and the emerging digital knowledge workplace”, IBM Systems Journal, 40, 4, 925-955, 2001;

[6] - Morris, S., Yen, G., Wu, Z., and Asnake, B.: "Time Line Visualization of Research Fronts," Journal of the American Society for Information Science and Technology, 54, 5, 413-422, 2003;

[7] - Nissen, M. E.: “An Extended Model of Knowledge-Flow Dynamics,” Communications of the Association for Information Systems, 8, 251-266, 2002.