Distributed Knowledge Management based on Software Agents and Ontology*

Michal Laclavik[1], Zoltan Balogh[1], Ladislav Hluchy[1], Renata Slota[2], Krzysztof Krawczyk[3] and Mariusz Dziewierz[3]

[1] Institute of Informatics, SAS, Dubravska cesta 9, Bratislava 84237, Slovakia
laclavik.ui@savba.sk
[2] Institute of Computer Science, AGH-UST, al. Mickiewicza 30, Cracow, Poland
rena@uci.agh.edu.pl
[3] ACC CYFRONET AGH, Nawojki 11, 30-950 Cracow, Poland
krafcoo@icsr.agh.edu.pl

Abstract. In this paper we present the use of ontology for knowledge representation and handling in Software Agent Systems. Motivation has come from Pellucid IST project where we need to capture and capitalize employee's knowledge in organization. This knowledge is then presented to other employees as they work on particular tasks. The Protg ontology editor and JADE multi-agent system is used for implementation. Ontology is usually used in intra-agent communication for agents to understand each other; we used ontology also as knowledge data model to store knowledge as instances of ontological terms into object database, thus agents can access and manipulate knowledge data directly and still stay lightweight.

1 Introduction

Motivation for this article has come from Pellucid project. Pellucid (Platform for Organizationally Mobile Public Employees) is European Project IST-2001-34519. The Pellucid System is particularly aimed to capture, record and capitalize the knowledge of current employees about their work in an organization [1].

Pellucid uses the socalled Intelligent Software Agents based on FIPA standards [2],[3] for different activities as capitalizing and data mining of knowledge, personalizing of knowledge, indexing and analyzing organizational document repositories or for integration of existing systems in organization. The ability to use ontology makes Pellucid easy to customize for different problem domains by simply changing domain ontology.

Human knowledge is based not only on facts which are true or false but also on uncertain knowledge which is true or false partially. Several methods can be used to represent such knowledge, e.g. probability measures, fuzzy logic

* This work was supported by EC Project Pellucid 5FP RTD IST-2001-34519 and Slovak Scientific Grant VEGA 2/3132/23

or computing with words [4]. Some methods are known to represent uncertain knowledge even in agent systems by e.g. extended FIPA-SL language; however, uncertain knowledge is still quite complicated and not understandable especially for the agents themselves. When using uncertain knowledge or knowledge where true and false facts are not strongly defined, computer systems cannot discover new facts in existing knowledge base using logical operators. This is known as a fundamental problem of contradictory knowl-edge in computer systems [5].

This is why knowledge base in Pellucid consists only of strongly true facts. Such facts are structured and defined by ontologies. By evaluation of administration application and pilot sites of the project we discovered that for administration application where business processes are well defined it is reasonable and useful to use knowledge based on facts rather than on uncertain knowledge.

Recent knowledge systems built on ontological or other models are mostly centralized solutions [6]. Pellucid uses hybrid approach, where agents can access certain knowledge directly from centralized organizational memory or they can ask specialized agents to provide them with knowledge handled only by this particular agent; e.g., in Pellucid monitoring agent has knowledge about workflow actors, activities or workflow processes since it represents interface to workflow management system; information and search agent has certain knowledge about documents and can do full text searching or content and context base searching of knowledge in documents or capitalization agents providing other agents and users with knowledge of similarity measures among documents [7],[8]. Agent based approach as created in Pellucid thus combines distributed and centralized approach which seems to be extremely useful for the area of administration application.

In this article we will devote attention mostly to usage of ontology for knowledge management in Pellucid architecture. Therefore, only two types of agents will be partially presented: Part of Personal Assistant Agent, called Query Agent, and part of Role Agent, called Answering Agent.

2 Knowledge Management Architecture

When a user wants to extract some knowledge from the information system, he/she has to create a query. Our goal is also to automatically prepare all possible questions one can ask. Such questions are created from ontology. In addition, the goal is to create such an algorithm which is able to answer questions whether such knowl-edge is presented in information system. Pellucid system has the ability to capture and record knowledge from the user. Simple actions as contacting someone in a particular workflow activity, reading or updating documents are captured. Pellucid Agents have also some data-mining capabilities for capitalization of knowledge. Data-mining algo-rithms and detail description of information capture is out of scope of this article; we believe however, that together with gathering information from information system based on pre-prepared questions they are the basis of the future type of information systems.

Information System for querying of knowledge [11], [12] consists of three main ele-ments:
- Query Agent (QA) - Agent used by user to ask any possible questions based on used ontology. This agent includes user interface as well. There are several such agents in the system, each representing different user.
- Answering Agent (AA) - Agent which is able to answer all possible QUERY-REF ACL messages where FIPA-SL is used as content language.
- Organizational Memory (OM) - where the knowledge is stored.

2.1 Pellucid Modules

In this section we briefly describe generic modules of Pellucid. This description of functionality is needed for better understanding of described modules. Generic version of Pellucid system has three modules: Intelligent Contact Management Module, Intel-ligent Document Management Module and Critical Time Management Module.

List of contacts is presented in each organization in a form of contact database, searchable by keywords with sorting ability, etc. Intelligent Contact Management Module will provide users with intelligent contact list related to the activity, which is performed by a user in contact relation to a particular activity. Each module uses its ontology to define relations between knowledge entities such as activity, document, contact, etc., and uses common techniques to capture, capitalize and return knowledge to a user. We will explain knowledge management functionality on simplified Contact Module [12].

3 Knowledge Manipulation

Agents are manipulating with knowledge by message passing among them. FIPA defined standard called Agent Communication Language (ACL) which is used for agent communication also in our architecture. The main part of ACL message is the content of message. FIPA Semantic Language (FIPA-SL) is used as content language in Pellucid architecture [2], [3], [13]. FIPA-SL queries are also past to querying engine to return knowledge from Organizational Memory. Each agent must understand terms used in content of the message in the same way, thus these terms are defined by on-tologies.

Ontology Ontology in Pellucid defines the meaning of the terms in used content language and the relation among these terms. The bases of the Pellucid ontology are "Event" and "InformationResource". Each event is connected with an activity (task) in workflow, with an employee, and can be connected with "InformationResource" (Document, Contact) as well. (See Fig. 1)

Ontology will later include more generic elements, which will help extend it with domain specific ontologies, different for each application area. Pellucid does not sup-port domain specific ontology such as "Ontology of installation of traffic lights" which is one pilot site of Pellucid, but we believe that by extending described extract-ing algorithm Pellucid will be able to query knowledge for different domains, simply by changes in domain ontology. We developed a system which is able to work with knowledge represented by any domain specific ontology. However, ontology used in Pellucid is well suited for repre-senting knowledge in organizations, where administration processes or workflow are presented, because it relates user activities with information resources and particular type of work.

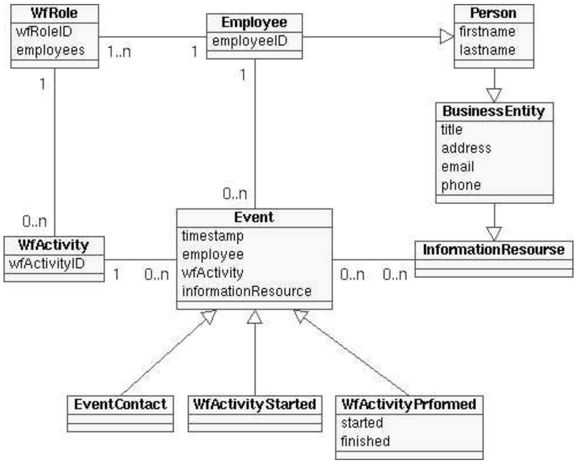


Fig. 1. UML scheme - Bases of Pellucid ontology

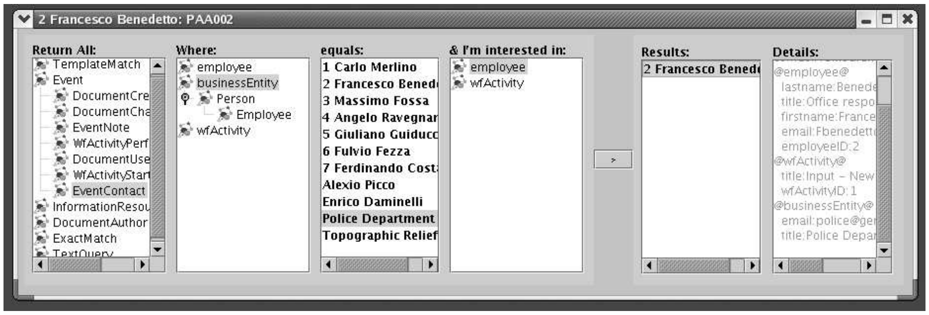
3.1 Capturing and Extracting Knowledge

Capturing knowledge is done by user input or by simple storing of user actions such as workflow activity performed, someone contacted by email or reading of a document. Using of Workflow Management System helps Pellucid to interconnect user actions with work activities of an employee. This enables Pellucid to serve knowledge to different employee in the future when employee works on the same task (workflow activity). However, capturing of knowledge is out of the scope of this article, so we mentioned it only for better understanding of complete architecture.

Ontology as represented in JADE [14] needs to define predicates, concepts and terms. Concepts are e.g. contacts, documents, workflow activities or employees. Predicates connect concepts, e.g. EventContact connects Employee, Contact and WfActivity and it means that contact was contacted by employee in particular activity. Thus predicates are such concepts, which we can ask if it is true or false. Concept is object in ontology and term is property of the concept. Event and its child classes are treated as predicates in Pellucid ontology. We can understand each predicate as a sentence representation in human language, e.g. for WfActivityPerformed sentence is ”Employees which performed current workflow activity” or for EventContact sentence is ”Contacts which were contacted in current workflow activity by particular em-ployee”. As we can see, sentences are based on Event Name and its interconnection with Employee, InformationResource and WfActivity. Our algorithm for creating questions is non-deterministic and decisions which way to go in creating questions is up to the user.

```
pQuestion = SelectPredicate();
cKnown = SelectConcept(GetAllConcepts(pQuestion));
iKnown = SelectInstance(GetAllClasses(cKnown));
cNeeded = SelectConcept(
  GetAllConcepts(pQuestion) - cKnown
);
return createQuestion(
  pQuestion, cKnown, iKnown, cNeeded
);
```

This simplified algorithm describes how we create FIPA-SL query by using ontology and GUI for user selection of elements. The result of this algorithm is FIPA-SL query as in the example bellow and also as on the Fig. 2. Answering agent transforms this query to objects and compares object instances in OM to this query using Java reflection methods. Automatic querying of InformationResources based on Event type and its connection to a workflow activity is the basis of Pellucid ontology and algorithms. By simple extending InformationResource with Document and creating several Events as docu-mentUsed or DocumentCreated extra functionality is added.


2 Francesco Benedetto: PAA002

Return All:
TemplateMatch
Event
DocumentCre
DocumentCha
EventNote
WfActivityPerf
DocumentUse
WfActivityStart
EventContact
InformationResou
DocumentAuthor
ExactMatch
TextQuery

Where:
employee
businessEntity
Person
Employee
wfActivity

equals:
1 Carlo Merlino
2 Francesco Bened
3 Massimo Fossa
4 Angelo Ravegnar
5 Giuliano Guiducc
6 Fulvio Fezza
7 Ferdinando Cost
Alexio Picco
Enrico Daminelli
Police Department
Topographic Relief

& I'm interested in:
employee
wfActivity

Results:
2 Francesco Bened

Details:
@employee@
lastname:Benede
title:Office respo
firstname:France
email:Fbenedett
employeeID:2
@wfActivity@
title:Input – New
wfActivityID:1
@businessEntity@
email:police@ger
title:Police Depar

Fig. 2. Query creation GUI. Query on the picture can be read as follows: Return All EventContacts where BusinessEntity (contact contacted) is Police Department and I am interested in the employee, which made the contacts. Note that ontological elements adopt tree structure from inheritance of ontology elements.

4 Conclusion

In this paper we described how agent based architecture can be used in distributed knowledge management systems. We focused on the knowledge based on true facts only, because this way we do not have to deal with contradictory knowledge and logical operations can be used for evaluating and discovery of new knowledge. It is clear that using of such knowledge is not usable for all possible knowledge management applications, but it is extremely useful for the area of applications dealing with administration processes where workflow is reasonably well defined. We believe similar approach with some modification can be and will be used in next generation of knowledge management systems in the commercial area as well.

References

1. Pellucid Consortium: Pellucid Project Technical Annex. (2001)
2. FIPA: FIPA Specification ACL Message Structure. http://www.fipa.org/. (2000)
3. FIPA: FIPA Communicative Act Repository Specification. http://www.fipa.org/. (2000)
4. Paul P. Wang (Editor): Computing with Words. ISBN: 0-471-35374-4, (2001)
5. Michael Wooldridge: Introduction to MultiAgent Systems, ISBN: 047149691X, (2002)
6. Ontoweb Consortium: Ontoweb Portal, http://www.ontoweb.org/ (2002)
7. R. Slota, K. Krawczyk, M. Dziewierz, J. Kitowski, S. Lambert: Agent paradigm for accessing document repositories in Pellucid platform, Proceedings of EuroWeb 2002 Conference, Oxford, UK, 17-18 (December 2002), pp.192-194.
8. R. Slota, M. Majewska, M. Dziewierz, K Krawczyk, M. Laclavik, Z. Balogh, L. Hluchy, J. Kitowski, S. Lambert: Ontology Assisted Access to Document Repositories for Public Sector Organizations. PPAM Conference (2003)
9. DARPA: DAML Website, http://www.daml.org/ (2002)
10. Telecom Italia Lab: JADE website, http://sharon.cselt.it/projects/jade/ (2002)
11. Pellucid Consortium: Technical Report of Pellucid Architecture, version 2003-04-15, (2003)
12. M. Laclavik, Z. Balogh, L. Hluchy, G. T. Nguyen, I. Budinska, T. T. Dang: Pellucid Agent Architecture for Administration Based Processes, IAWTIC 2003, Vienna (2003)
13. FIPA: FIPA Ontology Service Specification, http://www.fipa.org/ (2000)
14. Giovani Caire: JADE Tutorial Application-defined Content Languages and Ontology, http://jade.cselt.it/ (2002)
15. University of Amsterdam: Ontology Bean Generator for JADE, http://www.swi.psy.uva.nl/usr/aart/beangenerator/ (2002)