

Proceedings of the Third Workshop on Electro-Communication and Information

WECI-III 1999

March 7th, 1999

Tokyo, Japan



Supported by :

Indonesian Society on Electrical, Electronics, Communication and Information (IECI)

The Institute of Electronics, Information and Communication Engineers (IEICE)

The Graduate School of Information Systems (IS)

University of Electro-Communications (UEC)

Organized by :

Indonesian Society on Electrical, Electronics, Communication and Information (IECI)

Chapter Japan

Distributed Expert System Architecture for Automatic Object-Oriented Software Design

Romi Satria Wahono and Behrouz H. Far

Department of Information and Computer Sciences, Saitama University

Email: romi@cit.ics.saitama-u.ac.jp

Abstract :

The goal of this project is developing an expert system that aims to help software designers while designing object-oriented software by automating object-oriented design process. Some novel points are : using an interactive user interface, to enable software designer and expert system can communicate with each other in order to determine the best design, without disturbing the creativity of the software designer; making it possible for distributed expert system, to overcome the data sharing and the interoperability problems in groupware. In this paper we present the research project outline and the research approaches.

Keywords : Distributed expert system, Automatic software design

Abstrak :

Sasaran dari proyek penelitian ini adalah membangun expert sistem yang bertujuan untuk membantu para desainer software ketika mendesain software berorientasi obyek, dengan meng-otomatisasi proses pendesainan software. Beberapa pendekatan baru yang digunakan adalah dengan; menggunakan user interface yang interaktif, sehingga desainer software dan expert sistem dapat berkomunikasi satu dengan yang lain untuk menentukan desain terbaik, tanpa mengurangi kreatifitas dari desainer software, membuat expert sistem terdistribusi, sehingga diharapkan masalah data sharing dan interoperability pada groupware dapat diatasi. Pada paper ini kami menguraikan proyek penelitian secara menyeluruh dan juga pendekatan-pendekatan yang kami gunakan.

Kata Kunci : Expert sistem terdistribusi, Pendesainan software secara otomatis

1. Introduction

In this paper we report on distributed expert system for supporting object-oriented software design.

The four challenges of object-oriented design are to identify the objects and classes needed to implement the software, describe the relationships between the identified objects and classes, define the behavior of the objects by describing the function implementations of each class, and refine and organize classes by using inheritance to share common structure [6][4]. These are very complicated challenges because of there dependence on heuristic [1].

On the other hand, changes of social and technology's field result in the change of software development framework, from command /order system into cooperation system in groupware using network technology as

media [2]. It makes an urgent need of a new CASE tool, which can handle data sharing, interoperability, compatibility and portability problems.

In this research, we try to develop distributed expert system therefore it can overcome certain difficulties in the object-oriented design, difficulties in the very large knowledge base system, and it can fulfill the need of a new CASE tool for groupware.

2. Research Background

2.1. Known problems in object-oriented design process

Object-oriented development has now become a major approach in the design of software systems. Similar to the growth of structured methods we saw a number of

object-oriented design methods appear, such as Booch [3], OMT [4], Coad and Yourdon, Shlaer and Mellor, etc. Thereafter two of the major methods gurus, Iva Jacobson and Jim Rumbaugh, joined Grady Booch, defined a common Unified Modeling Language (UML) [5]. The above methods are fairly similar in concepts but with significant differences in approach and notation.

Researchers and software designers come to a conclusion that identifying objects including associations, attributes is very complicated and can not be solved yet, by the above mentioned methods, regarding of the difficulties of heuristic [1][6].

This is mainly due to lack of formalism for object-oriented software design. In this research, we suggest a new approach to software design by formulating design patterns and rules, and storing them to knowledge bases on expert system.

2.2. Problems with very large scale knowledge based systems

Applied Artificial Intelligence systems have long been suffering from the lack of the large-scale knowledge bases. Data search and retrieval in such knowledge bases is a bottleneck. We think that distributed knowledge base and efficient data communication is a possible solution.

This research project is a first step towards realizing this idea. We think that this will become a milestone in developing expert system by blending the network and expert system technology.

3. System Architecture and Design

Figure 1 shows an overview of our viewpoint on distributed expert system. This system is composed of User Interface, Communication Engine, Reasoning Engine, Distributed Knowledge Base, and Documentation Engine that run their own task. Each engine with a running autonomous task is called an *expert unit*.

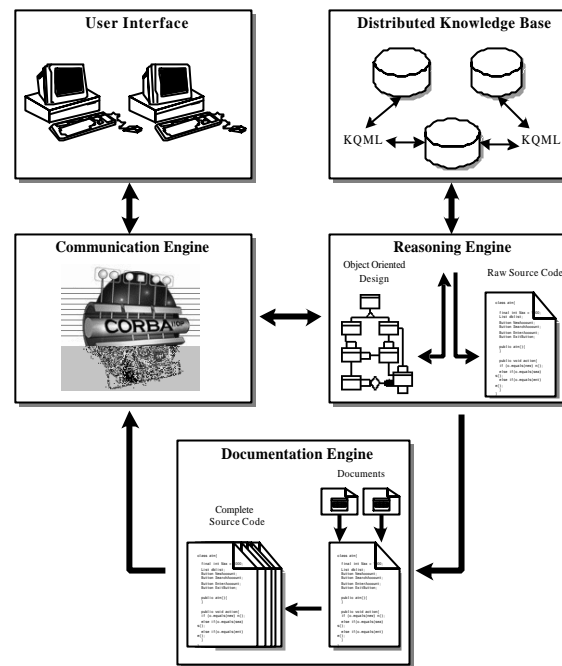


Figure 1 : System Architecture

3.1. User Interface

There are four functions that are managed by the user interface unit, they are object-oriented design editor, reasoning engine process (result) viewer, Java source code viewer and documentation (help) function. Using this user interface unit, software designer and expert system can communicate with each other in order to determine the best design. User interface unit occupies client position in this system.

3.2. Communication Engine

The communication engine is mainly responsible for maintaining connection between user interface unit and another expert units besides of message management. We are implementing the communication engine by using CORBA (Common Object Request Broker Architecture) distributed object, because it can handle data sharing, interoperability, compatibility and portability problems on complex application.

3.3. Reasoning Engine

Reasoning engine is the kernel of the system. This expert unit plays an important role in

reasoning about the design requests coming from the user and the design rules or patterns that are stored in the knowledge base. Actually, reasoning engine is composed of the object-oriented reasoning module and the source code reasoning (generation) module. These two, address two kinds of design rules in distributed knowledge base.

3.4. Distributed Knowledge Base

Distributed knowledge base is composed of some knowledge bases, using Knowledge Query and Manipulation Language (KQML) as communication language. In this research we intend to capture and reason with the sort of knowledge that software design experts use in design. Software design knowledge includes *design product knowledge* and *design process knowledge*.

The design product knowledge relies on the perspective that the design system is viewed. It includes domain-specific concepts, constraints and qualitative or quantitative models of the system. And the design process knowledge is viewed as a progression toward a goal by applying detailing patterns. The design process knowledge involves *design rules* acquired from human design, and *tacit knowledge* to make such pattern operational. Actually, design rules have two kinds of rules, the rules for object-oriented design and the rules for source code generation.

3.5. Documentation Engine

Documentation engine plays the role of reasoning engine's assistant. The documentation engine performs three main tasks:

- Documenting all tasks that are produced by reasoning engine,
- Preparing and reformatting the above mentioned data for to be transferred to the user interface units,
- Assisting user or software designer to create good software documents.

How this system works is shown in Figure 2. The dynamic model (Figure 2) shows the time-dependent behavior of this system. Using interactive user interface, software designer

sends design request through communication engine to the expert system, and then retrieves the result. The design requests are:

1. Identifying Object,
2. Identifying Associations,
3. Identifying Attributes,
4. Refining with inheritance.

Communication between the user and the expert system is running until the best design is determined. And finally, user sends the source code generation request to get a complete source code from the expert system

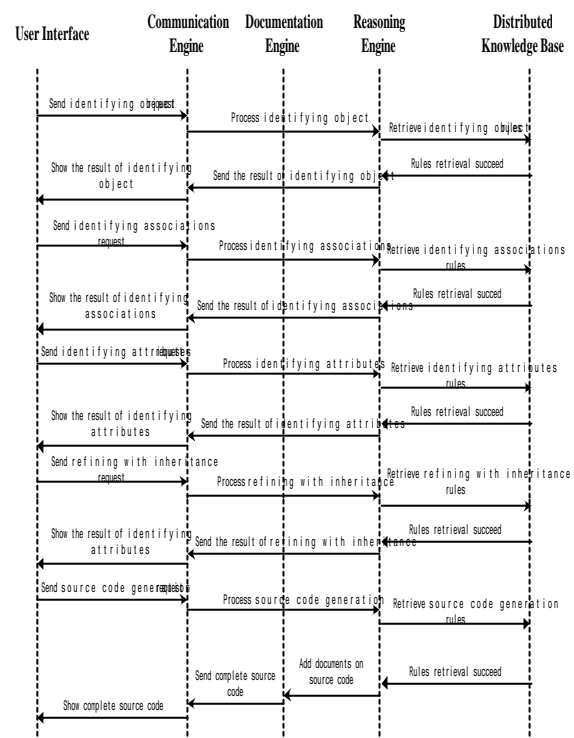


Figure 2 : Dynamic Model

Figure 3 shows a functional model of this system. The functional model shows how values are computed, without regards for sequencing, decisions, or object structure. The functional model shows which values depend on which other values and the functions that related them. We use data flow diagram, because it is useful for showing function dependencies.

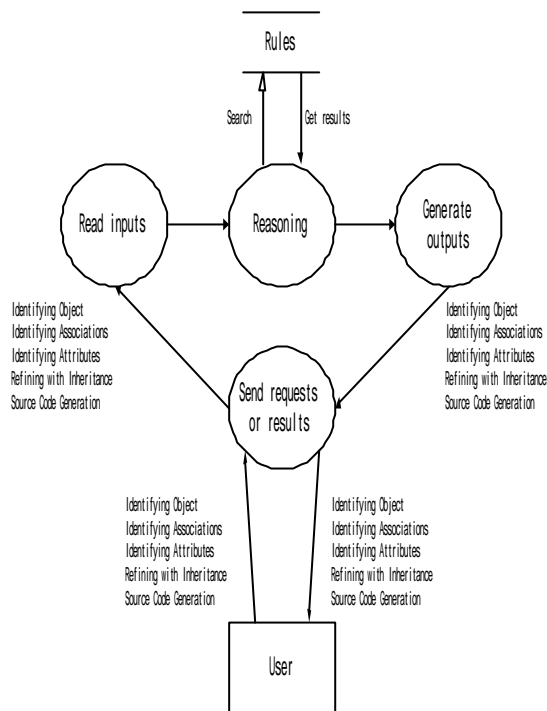


Figure 3 : Functional Model

4. Conclusion

In this paper, we presented a distributed expert system architecture for automatic object-oriented software design. The proposed approach differs from existing methods, we try to solve the difficulties in object-oriented design by developing an expert system which contains rules for automating design process. It has an interactive user interface, therefore software designer and expert system can communicate with each other in order to determine the best design, without disturbing the creativity of the software designer.

5. References

- [1] 加藤貞行, “オブジェクト識別についての一考察とその効果”, *情報処理学会研究報告*, Vol.90, No.100,1998.
- [2] Drucker P., “The Coming of the New Organization”, *Harvard Business Review*, Jan.-Feb. 1998, pp. 45-53, 1988.
- [3] Booch G., “Object-Oriented Design with Applications”, *Benjamin/Cummings*, 1991.
- [4] Rumbaugh J., Blaha M.,Premerlani W., Eddy F. and Lorenson W., “Object Oriented Modeling and Design”, *Prentice Hall*, 1991.
- [5] Booch G. and Rumbaugh J., “Unified Method for Object-Oriented Development Tech. Rep.”, *Rational Software Corporation*, 1995.
- [6] Ian M. Holland and Karl J. Lieberherr, “Object-Oriented Design.”, *ACM Computing Surveys*, Vol. 28, No. 1, March 1996.
- [7] Behrouz H. Far, Takeshi Takizawa, and Zenya Koono, “An Expert System for Reproducing Human Cognitive Processes in Automatic Software Design”, *World Congress of Expert System*, Portugal, 1994.
- [8] B. H. Far, H. Mukai, Zenya Koono, “Intelligent Agents for Electronic Commerce”, *Proceedings of The 11st Annual Conference of JSAI*, 1997.
- [9] Z. Koono, B.H. Far, T. Sugimoto, M. Ohmori and Hui Chen, “A Systematic Approach for Design Knowledge Acquisition from Documents”, *Proceedings of The Third Japanese Knowledge Acquisition for Knowledge-Based Systems Workshop*, 1994.
- [10] Yonglei Tao, “Using Expert Systems To Understand Object-Oriented Behavior”, *The 26th SISCSE technical symposium on Computer science education*, 1995.
- [11] 河野 善彌, 陳 慧, B.H. ファー, “人の設計知識の構造とソフトウェア工学”, *情報処理学会ソフトウェア工学研究会 97-SE-114*, pp. 33-40, May 1997.
- [12] B.H. Far and Zenya Koono, “Ex-W-Pert System: A Web-Based Distributed Expert System for Groupware Design”, *World Congress on Expert Systems' 96*, pp. 545-552, Seoul, Korea, February 4-9, 1996.