

Question-Answer system for object-oriented analysis and design

P.I. Sosnin & E.P. Sosnina

Uljanovsk State Technical University, Uljanovsk, Russia

ABSTRACT: This paper describes the results of studies and implementations which can add basic Rational Unified Process (RUP) flows by a special workflow to support "Interaction with experience". Such workflow is developed with orientation on RUP on the basis of a special NetWIQA system that supports question-answer reasoning of a design team. NetWIQA allows designers to create the drafts of the basic RUP samples for their subsequent coding and use in RUP environment or without reference to RUP; set of samples based on the protocol of QA reasoning, its analysis and necessary transformations. NetWIQA is the client-server QA processor with the special command system.

1 INTRODUCTION

The Rational Unified Process (Kroll & Kruchten, 2003) and relative systems of object-oriented analysis and design (OOAD) characterize the modern approach to Computer-Aided Design. Such means give designers an interactive access to useful patterns and sample actions, and it is used to manage the design activities and to borrow the necessary samples with the purpose of their rational implementation in the process of designing and its results.

The RUP gives us the access to normative models of design experience and knowledge prepared to be used in a corporate network. The models of experience in RUP are the richest source of questions, e.g. questions checking conformity of the project to the appropriate standards; questions, which provide personal and collaborative understanding, rational decision-making, and etc. Such questions are invariant to the contents of the project, but are very important in designing.

Unfortunately, RUP has no effective means for current work with questions and answers presented as human reasoning, i.e. reasoning of designers. At the same time reasoning is the basic way of interaction of people with the design experience and its models.

Those OOAD systems, in which interaction with the experience is realized obviously, e.g. in BORE system (Henninger, 2003), use methods and means of access only to typical decisions and are based on Case-Based Reasoning (CBR) approach.

Below we submit the results of our studies and implementations which can add basic RUP flows by

a special flow to support "interaction with experience". Such workflow is developed with orientation on RUP (but irrespectively of RUP) on the basis of a special NetWIQA system that supports question-answer reasoning (QAR) of a design team in a corporate network.

2 RELATED WORK

The domain of our study includes the similar works with experience, knowledge and its models, e.g. Rational Unified Process and similar techniques of automated design; Question-Answer techniques and systems; presentation and processing of experience in Case-Based Reasoning approach to computer-aided design.

RUP and other OOAD toolkits effectively solve the task of processing of patterns and templates (Galic et al., 2000), i.e. every pattern being the certain model of experience presents one of the typical project decisions to be applied in other projects. At the same time RUP structure has no means to support the work with personal knowledge and experience of the every member of a team and collaborative knowledge. The concept and term "Experience" is not practically used in RUP documents and papers.

Last years we have observed a growing interest to Question-Answer (QA) techniques and systems. Many important scientific and practical results were obtained in several domains, e.g. in retrieval of relevant information in distributed, multimedia, multi-lingual and multi-agency data sets; in automated



learning in traditional and distance education; in automated design and decision-making in the different areas. But despite more than 40 year studies “we are just beginning to explore question-answering as a research area” (Hirschman & Gaizauskas, 2001). The main directions of QA research are named in the Roadmap Paper (Burger et al., 2001) and focused on: question taxonomies, its understanding, ambiguities, and reformulations; context and data sources of QA; real time and interactive QA; advanced reasoning and user profiling for QA.

At the same time the directory of IBM papers (www.redbooks.ibm.com) mentions the concept QA, or Question-Answering, only 4 times and doesn't mention the QA work with human experience at all.

Question-Answering systems are also implemented in the work with knowledge and experience in Case-Based Reasoning techniques (Yang at al., 2003; Han at al., 2003; Branting at al., 2004). We consider these approach to be one of the main solutions, which provide the effective modeling and processing of knowledge and experience in Artificial Intelligence domain.

At last, we would like to mention the system BORE (Building an Organizational Repository of Experience) as one of the most related toolkits (Henninger, 2003) to our study, but BORE application of QA is limited to the set of pre-conditions for working with the necessary rules of technique.

3 INITIAL STATEMENTS

Our purpose to develop automated means of interaction with experience is caused by a number of reasons, e.g.:

1. We are sure that modeling of experience is much more important than modeling of knowledge for future development of Artificial Intelligence.

To prove it we should realize that human intelligence appeared while evolution, and was an answer to a question "What can add the genetic forms of experience?" While its evolution, Intelligence got the functions of self-organization, self-development, transfer and use of out-genetic components of experience.

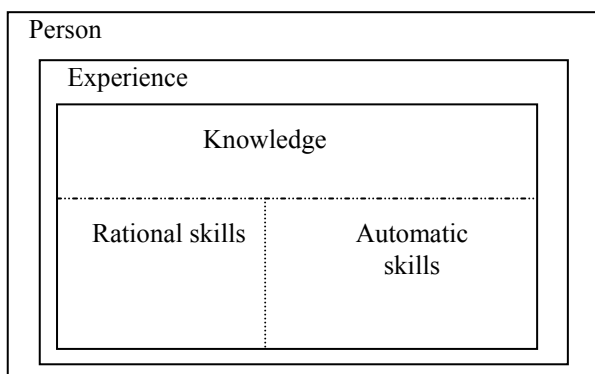


Figure 1 Experience and knowledge

The basic function of Knowledge, being the main part in the structure of Experience (Fig. 1), is its ordering, which increases efficiency in accessing to the components of experience, e.g. rational skills and automatic skills. We consider that the studies in AI domain should be based on Intelligence and Experience as the key concepts.

2. We claim that question-answering, being a natural function in intelligence, is an effective way to access experience for its application and development, e.g. for automated design.

The question is understood by the author as a specific natural/artificial phenomenon active when it is necessary to apply experience. The concept of “question” is understood and defined as a mismatch between the potential experience, necessary to implement a project, and the real experience of a design team.

The question controls the access to experience and its development through the answer and the process of answering, and such management is the essence of a question. We consider that if Case-Based Reasoning is used in forms of Question-Answer Reasoning while designing, it will provide naturalness in interaction of designers with experience and its models. Positive functional potential of QAR is much higher, than functional potential of CBR.

3. We believe that studies in Computer-Aided Design and the solutions are the great source of tasks and ideas to develop Artificial Intelligence. During design, experience and intelligence are used naturally.

Here presented studies and techniques are based on the following hypothesis:

QA interaction with experience consists of natural and artificial components being a complex unit. Such interaction may be much more effective if:

- 1 Question-Answer Reasoning of the designers is step-by-step registered in a special database of the project.
- 2 The results of QAR registering (QA protocol) are investigated, transformed to useful models, and used to support the next steps of interaction with experience in designing.

It has a number of useful interpretations such as:

- 1 QA protocol registers any design process as a research experiment, so representing “the primary measuring information” about design process and about the used control facilities.
- 2 Contents of the protocol reflect a real reasoning, which can be investigated to increase the knowledge about a “phenomenon of reasoning and questions”.
- 3 Each of the registered questions and answers admits its interpretation as an event essential for reasoning and design process, that allows to con-



sider the protocol as “a network of events” ordered in time.

- 4 Protocol is a data structure (QA structure) with its practically useful set of operations. Such interpretation of QA structure corresponds to the experience of Computer Science in the area of data structuring for adequate presentation of a QA protocol.

4 COMPARISON WITH RUP

To indicate the place and role of our NetWIQA system in OOAD framework, we present its possible association with RUP.

Scheme on Figure 2 generally shows that NetWIQA allows designers to create:

- drafts of the basic RUP samples (for example, all kinds of UML diagrams: Use-Case, Classes, Collaboration, Activity and so on) for their subsequent coding and use in RUP environment or other related OOAD means;
- drafts of the basic RUP samples for their application in the current project without reference to RUP (using only the possibilities of NetWIQA);
- a set of samples based on the protocol of QA reasoning, its analysis and necessary transformations (such samples are presented below).

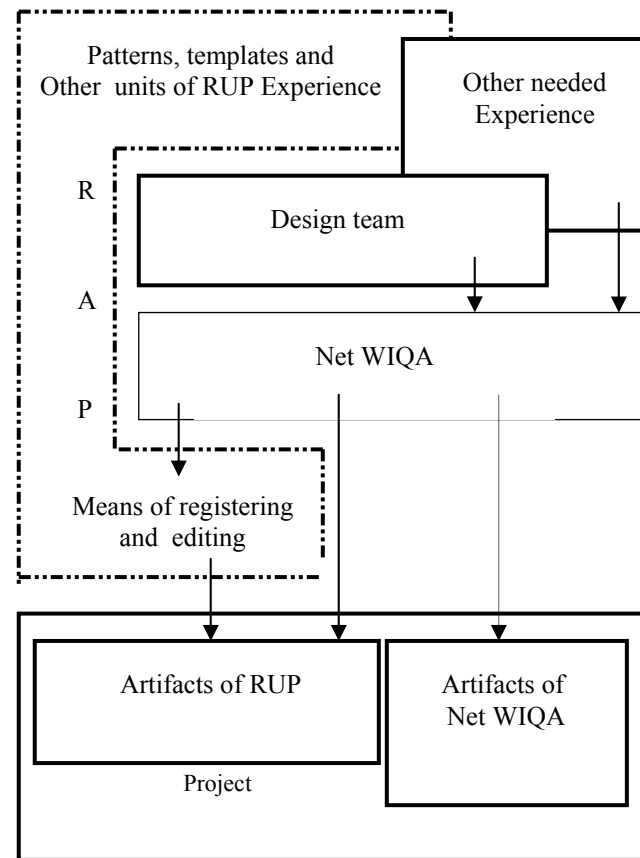


Figure 2 RUP and NetWIQA

The system is based both on RUP models of experience and personal experience of the members of

design team. NetWIQA has a rich set of operations to interact with all models of experience.

5 QA PROCESSOR

First of all processor NetWIQA is a system for supporting of the collaborative design process. This system gives the members of design team an opportunity of QA recording of the step-by-step designing to the uniform QA protocol.

NetWIQA is the client-server processor with the special command system with the following functions:

- to make QA protocols of QAR;
- to represent the protocols or their parts as the applied models (dynamically visualized and controlled graphs; models of prototypes for sample actions; the arguments, motivation and purposes);
- to transform the protocols to network structures (QA event nets, QA Pert nets and QA Petri nets);
- to analyze texts of questions and answers (the analysis is aimed at decoding and representing of semantics of reasoning with first-order predicate logic).

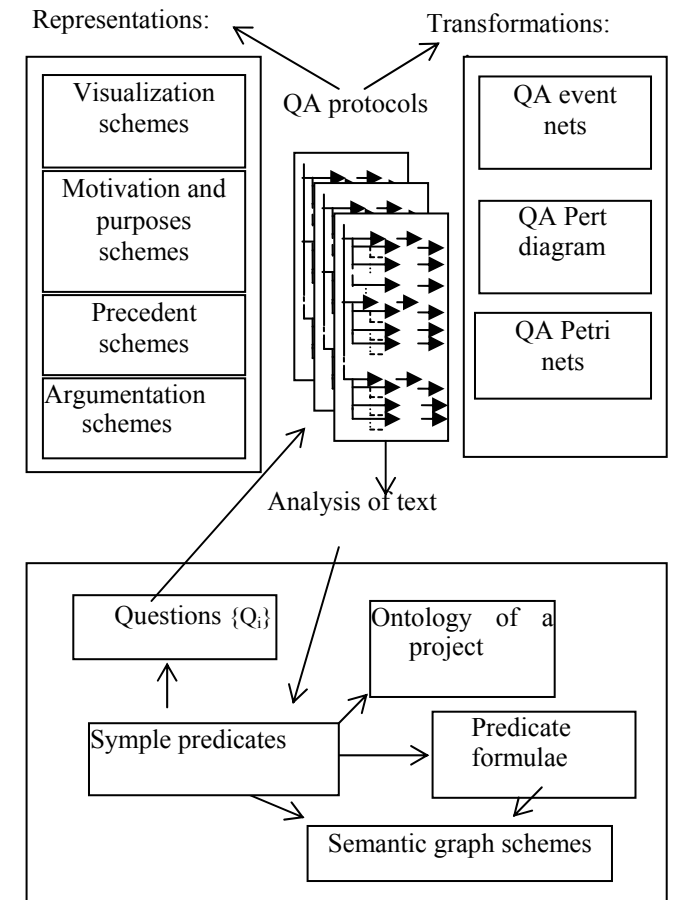


Figure 3 A set of QAR artifacts

QA processor has two set of the commands, e.g. instruction dialogue through the visual list of alternative commands; and plug-in mechanism that pro-



vides an access to default commands and new accessible functions.

These operations of QA processor are enough for development both the basic UML and RUP samples, and QAR samples (Fig. 3). In more detail, our processor is developed according to the component approach as the 3-layer architecture on the basis of Borland MIDAS technology.

It consists of layers:

- 1 QA database and Experience database (Libraries of QA templates and other samples of experience),
- 2 business-logic kernel (server),
- 3 presentation layer ("thin client").

The business-logic layer provides direct data access for the clients, and performs all data processing. Server component of QA processor consists of an application server, system magazine, synchronizer and etc. connected through TCP/IP sockets with Borland Socket Manager utility.

The third presentation layer of the system architecture involves a client library, a library of synchronizing flows, a library of configuration, a client workstation, and the configuration utility to adjust users, working groups and projects. The client workstation uses plug-ins to connect necessary functional modules to expand the set of QA applications, and in such a way additional applications to work with design experience and knowledge are built-in QA environment.

QA processor has the distributed and shared structure. The separate components can function on different computers in the local business network.

QA processor should ensure automated performance of the following main tasks:

- effective information and control support in Computer-Aided Conceptual Design of the automated systems;
- current unified documenting of processes and products, open for intellectual interaction;
- development of data and knowledge bases of complex projects and their interactive use while the conceptual design;
- controlled forming and accumulation of employees experience as QA databases of a design organization;
- control of automated designing, e.g. due to the rational choice of the direction of processes while design decision-making;
- controlled interaction between designers in collaborative work in the structure of a corporate design network;
- communicative document circulation in the corporate network;
- controlled training and learning during designing and operation of complex systems.

6 QA TECHNOLOGY

QA command system supports the workflow "Interactions with Experience" as the special set of applications, each of which perform the certain task of QA technique. At present we have developed about 40 utilities that are accessible to designers through WEB-environment of a client workstation. The structure and the access to the service utilities are realized according to the interactive RUP-model.

Each service task is submitted as:

- its QA pattern that are copied to QA database of the project and applied at the next case of the service task;
- interactive form to support its convenient application.

To demonstrate the functions of these applications let us name a few of them, e.g. "To solve a task", "To answer a question", "To construct the answer", "To prove the answer", "To discuss the answer", "To fill-in the pattern", "To set the requirement", "To generate the specification", "To determine the front of work", "To plan", "To control visualization of QAR fragment", "To generate the educational unit", "To control knowledge". The given utilities show that some applications of QA processor support also operative training for the members of a design team, e.g. to learn the certain design process.

QA technology is applied in any project as coordinated performance of the set of utilities and tasks appointed to the members of a design team. At any time each designer usually works with the dynamic set of the service and project tasks.

The common work begins with the utility "Conceptual design", which calls the following task "To open the project" as the initial condition of QA database of the project. The initial condition of a task-tree is submitted in Figure 4.

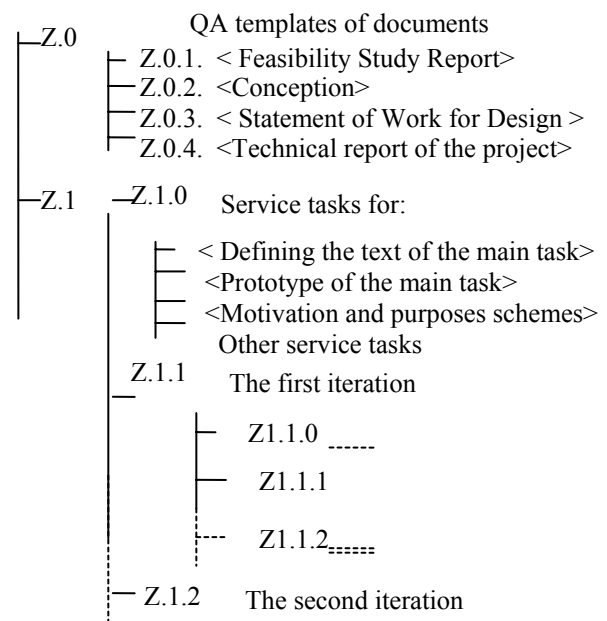


Figure 4 Task structure of the project

The standard QA patterns of further documenting are loaded at the initial state of QA database. So, e.g. the State Russian Standards require that any CAD/CAM/CAE system includes the standard documents, i.e. "Conception" (analogue of "Vision" in RUP), "Statement of Work for Design" and "Feasibility Study Report". Each pattern of documenting represents the system of normative questions which should be copied to QA database and answered during design process.

The system service utilities provide "interaction with experience" for the definite project tasks, which are formulated and solved during design.

At any moment, design and service tasks make a uniform graph, where each task is interpreted as a "question -task".

Each task is coded by a symbol "Z" and index appropriate to a place of this task in the project (index "0" specifies that a task has utility functions). Indexes allow designers to determine not only inner project tasks from utilities, but also the tasks of different iterations of the project.

Each task in a tree can be in one of the following conditions - "decided/solved", "unsolvable", "postponed" and "executed". Any task has the set of attributes with references to its templates in QA database.

Each project work is supported by QAR while decision of the task. NetWIQA step-by-step protocols all questions and answers and their relations (Fig. 5) in QA database of the project. Here we see only QA relations, while the database also includes other relations (for example such as temporary, personification and cause relations).

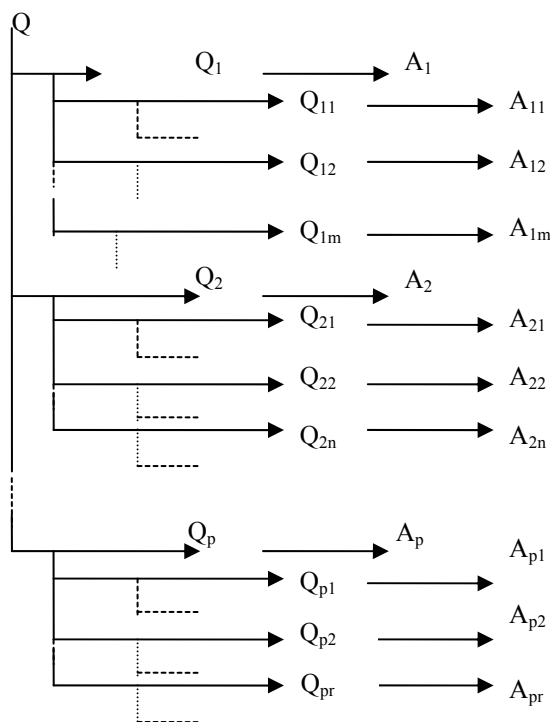


Figure 5 Question-answer structure of the task

Questions and answers get symbolic codes and indexes appropriate to their types and place in the project. Questions have two subtypes, i.e. Z-task and Q-quiry. Q-queries require the answers to be got from experience database. If any Q has not got the answer, it is changed to Z-subtype.

The answer class is wider and includes subtypes "A-answer", "I-idea", "H-hypothesis", "S-specification", "R-requirement", "M-motive", "O-purpose", "E-decision", "P-procedure" and "F-formula".

The QA structure of any task is a dynamic data structure composed during the decision-making and reasoning. There are 3 types of questioning activity:

- automatic reference to experience "automatic skill"
- rational reference to experience, e.g. with a question of Q-subtype;
- rational reference to experience, e.g. with a question of Z-subtype;

The model of QA process is extended to the event QA networks and other network models of coherent sets of questions and answers. It requires to solve the tasks of detection of questions, their identification, adequate coding and modeling, and effective application. For these purposes QA processor has the means of text analysis with automated translation to PROLOG-like and semantic net languages.

7 QA PROCESS

At definite time t_i the reasoning QAR(t) goes to QAR(t_i)-state, which has its causal potential that gives the possibility to move the reasoning forward to the next state QAR(t_{i+1}). In this aspect the "history" of previously made work, represented in QAR(t)-codes, influences the next rational step of reasoning. Next steps both for reasoning and for design can be defined by means of question-answer analysis of QAR(t)-codes.

General statement of each project task should be defined before Question-Answer working with this task. Special definition of the task (as its general statement) uses a special pattern to present a task as 3 structured text blocks.

The first reflects the main purpose of a system under design, which is specified by its potential users. Here we begin the work with the basic Use-Case diagram for the task in UML language.

The second defines the main techniques to perform Use-Case diagram for the task. It provides construction of the basic diagram of business -objects of UML.

The third defines technology of implementation of a system under design. Information of this block is applied in conceptual design as context information.



Analysis of a text T_0 of the general statement of a task and its translation to PROLOG-like language are used for extraction of questions to begin and continue QAR.

More detail it is based on step by step registering of questions and answers in accordance with following technique:

- 1 The set of questions $\{Q_i\}$ is taken from the text \dot{O}_0 and coded by adequate texts $T(Q_i)$.
- 2 Actions of item 1 are executed for each text $T(Q_i)$, therefore the set of questions $\{Q_{ij}\}$ and their codes $\{T(Q_{ij})\}$ is formed. Actions of item 2 are used to control the correctness of question codes and for the choice of those questions $\{Q_k\}$ which will be used for the next step of detailization from the set $Q = \{Q_i\} \cup \{Q_{ij}\}$. Other questions are recorded for their application in the subsequent steps.
- 3 Set of answers $\{A_k\}$ and their codes $\{T(A_k)\}$ is formed and registered in QA database.
- 4 Each text $T(A_k)$ is processed as the text \dot{O}_0 . The cycle 1-4 is repeated until the project comes to the end.

The work with questions, answers, QAR and RUP artifacts are controlled with the help of the appropriate system utilities.

The number of especially important actions (operations, command, plug-in mechanism, application of QA processor includes:

- 1 For questions: detection of obvious questions (on their indicators, predication (through translation on Prolog-like language, identification (on patterns, concrete definition (for types, assignment of meanings to attributes (as to the phenomenon of event type, argumentation.
- 2 For answers: creation, assignment of a type, change of a type, registration of a condition, editing of the contents, assignment of meanings to attributes (as to the phenomenon of event type, argumentation.
- 3 For QA groups: transformation to the node, expansion into QA structure, transformation to the event net, visualization of a network, analysis of a condition, choice of a direction of development, scrolling of dynamics (on inquiries.
- 4 For text: creation, transformation, grammar analysis, semantic analysis, transformation to the semantic graph, visual analysis, supporting of a phenomenon of attention.

8 CONCLUSION

In this paper we present the additional step in development of the application toolkit and technique for OOAD. This step includes the special workflow "Interaction with Experience" applicable to the RUP scheme of OOAD. Such workflow is based on specially organized QAR, its protocol as primary design information and processing for effective conceptual design in collaborative environments.

Methods and means offered, investigated and realized by the authors open some new possibilities to control CAD processes such as: effective monitoring of a QA process; analysis of opportunities of parallel coordination of work in a design team (with the purpose of distribution of work between the executives in typical design decisions); demonstration (at a suitable speed) of the development of events at the certain time interval of automated design; demonstration of the current condition of automated design at the certain moment; training to the typical decisions (CAD samples) and development of design skills; personification of events for subsequent definition of authorship and contribution of the members of a design team.

REFERENCES

- Branting L., Lester J., Mott M. 2004. Dialogue Management for Conversational Case-Based Reasoning. *Proc. of the Seventh European Conference on Case-Based Reasoning (ECCBR 2004), Madrid, Spain: 65-71.*
- Burger J. et al 2001. Issues, Tasks and Program Structures to Roadmap Research in Question & Answering (QA). *NIST.*
- Galic M., Macisaac B., Popescu D. Using a Single Business Pattern with the Rational Unified Process. <http://www.redbooks.ibm.com/>, REDP-3877-00.
- Han P., Shen R., Yang F. 2002. The application of Case Based Reasoning on Q&A System. *Proc of the 15th Australian Joint Conference on Artificial Intelligence (AI2002), Canberra, Australia: 31-37.*
- Henninger S. 2003. Tool Support for Experience-Based Software Development Methodologies. *Advances in Computers Vol. 59: 29-82.*
- Hirschman L. & Gaizauskas R. 2001. Natural Language Question Answering: The View from Here. *Natural Language Engineering Vol. 7: 67-87.*
- Kroll P. & Kruchten Ph. 2003. The Rational Unified Process Made Easy: A Practitioners Guide to the RUP. *Addison-Wesley.*
- Yang F., Shen R., Han P. 2003. Adaptive Question and Answering Engine Base on Case Based and Reasoning Technology, *Journal of Computer Engineering Vol.29 No.1: 27-28.*

