Quaestor history

What is Quaestor

Quaestor is a Knowledge-based Computational Model Assembling (KCMA) system, in fact an intelligent generator of executable numerical models. KCMA is able to support and facilitate the reasoning process in design and analysis. Being able to freely associate numerical model fragments, practically any problem statement in the domain of the knowledge base can be dealt with. So, on the basis of user provided data and a knowledge base in Quaestor containing computational model fragments, program interfaces and design data (Knowledge-based), executable computational schemes (Computational Model) can be constructed (Assembling) in Quaestor for a wide range of design and analysis applications.

In Quaestor you have to create your own applications comparable to the creating of your spreadsheets in spreadsheet programs or databases in database programs. Quaestor provides the platform for fast and easy development of powerful applications based on existing (and new) relations, programs, databases, spreadsheet, etc.

The creation of applications in Quaestor starts with the implementation of individual knowledge (relations, programs, databases, spreadsheet, including their constrains) into the program. When the individual knowledge is implemented Quaestor can be used to generate executable numerical models. By asking calculation goals to Quaestor, the user starts the intelligent reasoning process in which the implemented knowledge is combined together with required user decisions requested by the system. The resulting numerical models can be repeated using the same knowledge with different data. In this way, Quaestor creates a co-operation between user and computer in which optimum use is made of the specific capabilities of both. The user is typically able to make decisions on the basis of his experience whereas the computer can access and use vast amounts of well-structured relations, programs and data.

The history of Quaestor

The initiation

Ever since the introduction of computers, application modules have been developed which allow certain calculations. At first these applications were used as separate modules in the design or analysing process where the designer was the bridge between these applications and disciplines. The designer went through an iterative process before he could make a satisfying (conceptual) design.

As a rule the objectives to be obtained were and still are defined at the level of the executing party. When observing for instance ship design, the Ship-owner translated the operational and financial aims in number, type and size of ships. These are then translated into installation, construction and necessary supplies by the shipyard. Traditionally, the exchange of thoughts on the interaction of this objective is only brief. Consequently, the concept is not always optimally attuned to the operational objectives of the Ship-owner.

During the following developments, integrated design systems were built which, together with the application programs earlier mentioned, partly automated the interaction between the various applications and between the design process and the designer. As a rule these design systems are ready-made for the shipbuilding industry, have specifications such as loading capacity and speed for a starting point, and usually yield quite well detailed draft and engineering information. Mostly they contain no, or only restricted, mechanisms making use of experience and situations specifically relevant to the business.

In order to reach a quick estimation of optimal choices, Concept Exploration Models were introduced. These models generate a great number of alternative concepts and enable the user to select the most promising from these as a starting point for the more detailed design phase.

Four significant shortcomings of this method are apparent:

- It is common practice that the design concepts and analyses are not usually based on the end-user's ultimate (mainly financial and operational) demands but demands derived from these as regards sizes, speeds and technical preconditions. This discourages the search of the ideal compromise between cost, results, risks and technical possibilities.
- The programs available comprise a somewhat closed process and are not flexible enough to allow a quick and efficient application of new views, preconditions, experiences, applications and problem defining.
- The programs available focus on a certain problem. Problem definition of another kind (e.g. economics, fishing or offshore) requires the development or purchase of a new program, which in turn is often provided with other procedures and applications.
- The programs available are 'hard-coded'; i.e. the user is not able to adjust the programs as they please to their own objectives and requirements. Improvements on the programs can only be made by the suppliers and, therefore, take a long time to be put into effect and seldom lead to the flexibility required by users.

These shortcomings are a problem, especially during the conceptual phase when the creativity and the experience of the user are of vital importance and when designers are to accomplish the task of finding the one and only best solution in an abundance of possibilities within a short period of time.

The start, Quaestor for MS-DOS

In order to overcome these shortcomings, in the late '80s a start was made at MARIN with the development of a system that could control and apply (empirical) knowledge, mainly in numerical form. This development has led to the knowledge-based system Quaestor, a semi-automatic method for the assembling and execution of computational models running on MS-Dos platforms.

Although initially meant for private use and restricted application, the basic principles and the developed prototype turned out to be very suitable for a more general use, especially in conceptual design applications and in feasibility studies. As early as 1993 the Royal Netherlands Navy introduced the application of the Quaestor prototype in her projects. By and by the program was used in various research and development projects. Among other things these projects comprise joint industry projects, a NATO project which resulted in a conceptual naval ship design system, a number of graduation studies from Technical Universities and Colleges, some PhD theses and an industrial propeller design and analysis system. These applications have demonstrated Quaestor as an outstanding environment for industrial and scientific computational knowledge management without the shortcomings described above.

Quaestor for Windows

In 1997 the decision to develop Quaestor for the Windows platform was made. The main reason was the conclusion that the Windows platform would be the major platform for Personal Computers. Moreover, better facilitation by the platform of basic operations such as file handling, printing and editing of data is provided.

In 1998 the initial development was finalised resulting into Quaestor for Windows, with:

- More users friendly, highly improved user interface;
- Improved functionality.

After 1998 the basic Graphical User Interface (GUI) has not changed. However, due to the type of program constant development has been made on the functionality of editing, connecting and processing of data. Furthermore, a lot of effort has been made on automatic report generation and database functionality.

Recent developments

Besides continuous development on the functionality and stability of Quaestor there are three important recent developments:

1. Qnowledge has developed a Qnowledge Licence Manager (QLM) for local area networks, enabling Network administrators the easy setup and administration of user licences for Quaestor.
2. Together with MARIN, Qnowledge has combined a J2EE application with Quaestor to make knowledge bases "web-enabled". In this way, every knowledge base can be activated and used by means of a simple Internet browser over the Inter-, or Intranet.
3. A Quaestor Database Server has been developed to store project data (used in combination with knowledge bases) in a central database. This database server enables Quaestor users to use data of previous projects in Vault as input in a new project or knowledge base. This database server also takes care of version management of project files and knowledge bases.

For the QLM, the web-enabling and the Quaestor Database Server separate agreements with Qnowledge can be made. More detailed information can be sent on request. Also visit our website for the latest information: www.qnowledge.nl.

Future developments

Qnowledge is working on the development for or with Quaestor to operate as part of a national or global knowledge network.

Details of this development cannot be discussed in much detail at this moment. However, please visit our website www.qnowledge.nl for any additional information.