Object-Oriented Composition of a Framework for Integrative Facility Management

Frank Peters, Udo Meißner[1)]

ABSTRACT: This article presents an overview of general requirements for Facility Management under object-oriented modelling aspects. Shortcomings of current IT-environments for Facility Management in Germany are pointed out and a system architecture for a generic framework in Facilities Management is introduced, which is currently under research.

1 INTRODUCTION

The field of activities concerning the management and operation of real estate and buildings as well as the planning, construction and the demolition of buildings is mainly characterized by different and specific views of people involved in those activities (Figure 1).

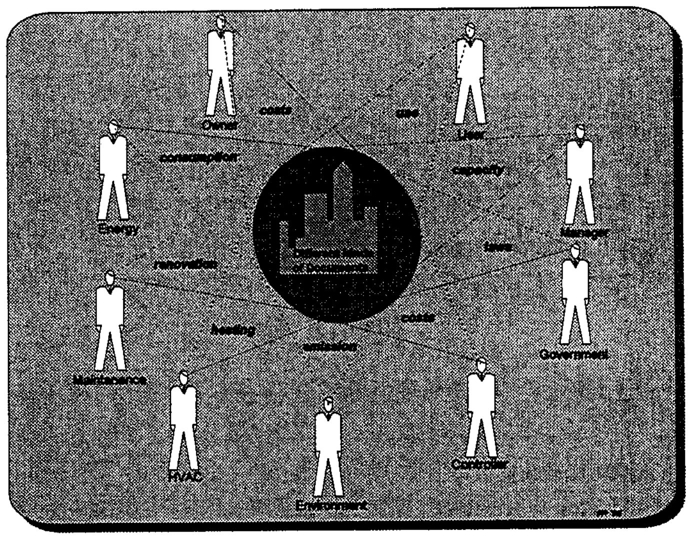

Figure 1: Different views on a building in operation

Therefore, people responsible for facility management need software systems that have the ability to support their specific tasks. This means that the system must deliver answers regarding all relevant information to the user concerning the real estate, the specific building structure and its technical infrastructure. Furthermore, the system must be easy to handle and efficient in use.

1) Prof. Dr.-Ing. Udo Meißner, Dipl.-Ing. Frank Peters
University Darmstadt
Institute for Numerical Methods and Informatics in Civil Engineering
Petersenstr. 13, D-64287 Darmstadt, Germany
email: meissner@iib.bauwesen.th-darmstadt.de
peters@iib.bauwesen.th-darmstadt.de
www: http://www.iib.bauwesen.th-darmstadt.de

Main interests for the management of the real estate are:

- Documentation of plans and legal responsibilities
- Documentation of public developement
- Documentation of the supply and waste disposal facilities
- Overview of environmental properties and influences:
  - sources of pollution
  - current state of pollution
- Information about laws and regulations
- Management of measures for renovation and repair
- Management of the economical ressources
- Management of the ecological ressources

For the management of the buildings, main activities are:

- Management of the technical infrastructure and equipment
- Management of the social infrastructure (cleaning services, janitory-services etc.)
- Management of the energy ressources
- Planning and management of changes in use
- Management of the exploitation

One of the most important issues, highly relevant for nearly all areas of Facility Management, is the commercial and ecological management of the real estate. Owners are interested in the value of the property, their income and the costs involved in the maintenance of the facilities. Moreover, a Facility Management System has to be capable of evaluating and interpreting a wide variety of technical, commercial and legal documents of different types and qualities.

2 SHORTCOMINGS OF COMMON SYSTEMS

In Germany, there are only few codes for building maintenance and nearly no codes for building management. The most important codes for maintenance are [DIN 31051] and [DIN 1076], both covering only small fields of building maintenance. The german regulation for the honorarium of architects and engineers [HOAI] only covers building management until the time immediatly after completion of the building. Computer aided planning methods and data exchange efforts are not yet included in the HOAI. This leads to hesitating changes in the german building industry towards Facility Management, so that the exchange of plans, if at all done electronically, is mainly practiced in the form of line drawings (e.g. DXF). Data exchange according to the forthcoming international standard ISO 10303 [ISO 10303-STEP] is found very seldom. The idea of *product modelling* has not yet been accepted by the german building industry in general.

Because of the heterogeneous nature of the involved processes and the lack of integrating codes, the corresponding IT-environments have also grown to become heterogeneous. Therefore, the market is dominated by a large number of often complex software systems which are purely standalone solutions. These software systems are mainly specialized on parts of building management, and most information exchange is done at the lowest possible semantic level, thus leading to a great loss of information. As a consequence, the acceptance of Facility Management and the entailed comprehensive and integrative view on the lifecycle of buildings is rather underdeveloped.

Almost all available applications are combinations of traditional CAD-systems and traditional **relational** databases. The segregation of alphanumerical and graphical data representation leads to many disadvantages of this approach, as mainly:

- loose coupling of different applications
- unsatisfying support for collaborative group work
- redundancy of data
- lack of central control of information
- no adequate representation of views
- only a few kinds of documents are supported
- record-oriented data-structure
- complex models cannot be managed

3 REQUIREMENTS FOR FUTURE SYSTEMS

To avoid the described disadvantages, a future-oriented Facility Management System has to be an integrative, process-oriented framework, supporting several key features:

- Handling of multimedial data.

  This is due to the wide variety of technical and commercial documents of different types and qualities. For example, the building has been modelled with an CAD program; the enclosed information about the equipment is alphanumerical; the GIS-plans for the real estate are 2D; information about the employees is alphanumerical and sometimes graphical etc.

- Support for collaborative group work.

  The process of planning and use of a building involves many people. To handle the growing complexity and to guarantee a higher quality, the system has to support collaborative group work in heterogenous networks, as they are present in most companies.

- Support of Views, resulting from the different involved FM-core processes

  The extract of information from different processes has to be modelled by Views, based on distributed database systems. For example, the architect is mainly interested in the functionality of the rooms, whereas the HVAC-engineer is more interested in technical information, relevant for planning of installations. Therfore, both need different kinds of information concerning the **same** data.

- Integration of existing IT-tools

  Because of the numerous existing software tools, it is ingenious to ´recycle´ those tools by integrating them into the Facility Management Framework. This kind of migration reduces costs in development and for the end users of the system. The integrated applications must have the same user interface, to provide a consistent IT-environment to the user.

- Support of proprietary and standardized data management

  According to the integrative character of the framework, applications with proprietary data management are included as well as newly developed components using standardized data management such as STEP-SDAI [SDAI].

  The framework has to offer mechanisms for accessing the information within the standardized shared database as well as reliable mechanisms for managing and accessing the application specific data

According to these key-features, a future Facility Management System has to be seen as a framework for integrating software-tools.

It must be pointed out, that a comprehensive and integrative solution for the future can only be achieved by new object-oriented solution techniques. The central problem in system developement is the kind of communication between different applications to guarantee support of proprietary and standardised data management. This means, that not only the syntacs, but furthermore the **semantics** of communication is highly relevant, i. e. the different applications need a common **communication model** which is capable of self adaptation. Therefore, the main issue in developing such a framework is a concept for generic communication components to avoid problems arising from different interface techniques.

4 DESCRIPTION OF FAMOOS

The requirements described are the results of a system analysis of Facility Management for civil engineering. The main goal of an appropriate system architecture for a future FM-Framework is to achieve transparency of the coherences and to determine the priorities of the processes. During the analysis process, a system architecture has been developed as a basis for a prototype implementation of the Facility Management Framework. Worktitle of this project, currently under research is FAMOOS, an acronym for **FA**cility **M**anagement using **O**bject- and user-**O**riented **S**tructures.

To fulfill the determined requirements and to create the basis for convenient and easy modifications and exchangability of different components [Peters, Petersen], the system architecture has been divided into separate layers:

- **User-Interface Layer**

  Due to the wide variety of technical and commercial information in different kinds and qualities, as they exist in most enterprises, the Framework has to support the handling of multimedial information. To fulfill these needs, the user-interface has to be capable of handling these kinds of information in a most efficient way, to support the user with the all the necessary information in a well-structured manner. Furthermore, the user-interface, as well as the other layers need to have a portable architecture to give consistent and ergonomic user-support on the many different GUI's within a distributed and heterogenous network. The user-interface will be realized with the IT-tools described in [Peters, Petersen].

- **Application Layer**

  The integration of external and internal software tools is realized within the application layer. Main integration will be done using pre- and postprocessors, enabling specific applications to access the communication component of the management layer. The advantage of this kind of integration is the extensibility of the application layer with different applications without violation of the already integrated IT-tools.

- **Management layer**

  The Management-layer is the core component of the framwork and mainly responsible for the communication between the shared database and the different applications. To provide multi-user support, the management layer is based on an OODBMS-core with a client-server architecture. The semantic integration, which is also within the responsibility of this layer, will be done on the basis of the generic communication components. These components can be understood as a Object Request Broker [CORBA]. Via the different pre- and postprocessors of the application layer, information requests will be send to the applications in case that the information has not already been transferred into the shared database. The access of this central database will be done using SDAI [SDAI].

A draft of the object-oriented system architecture is illustrated in Figure 2. The consistent object-oriented approach for the Facility Management System was developed from the system analysis, over the software design to C++-applications. The object-oriented database management of product models, the object-oriented multimedial user interface and the generic communication components of different facility management applications are essential features of this system architecture, which is also capable to integrate conventional CAD-applications.

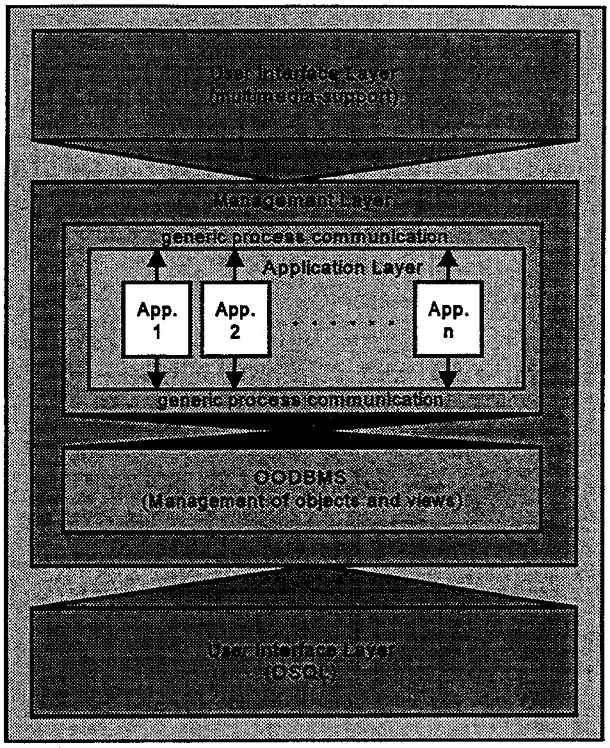

Figure 2: Object-oriented system architecture with generic process communication

5 CURRENT IMPLEMENTATION

To realize the described system architecture by building a prototype system, the Object-Oriented Model Management System (OOMM-System) developed [Rüppel], [Meißner, Rüppel, Peters] will be the central application within the management layer, mainly responsible for the communication with the shared database. The OOMM-System has been developed to ensure continuous software and information support for the structural engineer within the complex process of structural design.

To obtain this high level of integration in structural planning, the degree of semantic information of the data exchange has been improved. The aim of these developments was, that communication between all software-tools involved in the planning process is possible without loss of information.

This has been achieved by regarding the overall structural design process as the design of products, which are described in an object-oriented manner by means of conceptual product models, whereby each technical planner is supported by a subproduct model according to his specific task. Data exchange is then based on the exchange of consistent subproduct models. Therefore, subproduct models form the basis for the integration of different programs within the heterogeneous software environment of structural planning. This approach can be enhanced to the management of a wide variety of complex building information, as described above.

The management of the information flow between all subprocesses has originally been done by using neutral exchange files in STEP-syntax in compliance with the realization of the subproduct models. In the current prototype implementation for the described Framework, the object-oriented database ObjectStore [ObjectStore] is now used for this purpose. The integrated CAD and calculation programs can communicate throughout the OOMM-System using the shared database. Figure 3 shows a screenshot of the recent development. The user interface as well as the core components of the system have been developed to work on either 32bit MS-Windows systems or on several UNIX systems, e.g. SunOS 4.1.

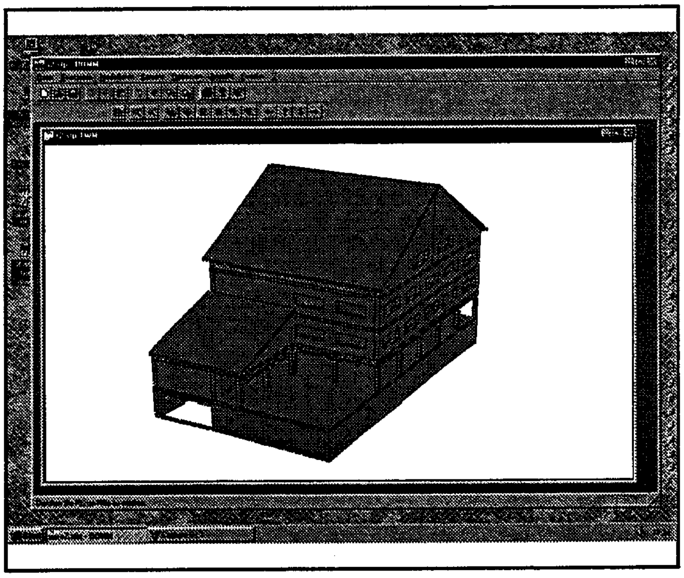

Figure 3: User Interface of the OOMM-System

The next step in development is the implementation of the generic communication components. At present, the research concentrates on feasability studies of different architectures for this purpose.

6 REFERENCES

[CORBA] Object Request Broker Architecture. OMG TC Document 93.7.2

[DIN 1076] Überwachung und Prüfung von Ingenieurbauwerken im Zuge von Straßen und Wegen. German Standard No. 1076

[DIN 31051] Instandhaltung; Begriffe und Maßnahmen. German Standard No. 31051

[HOAI] Honorarordnung für Architekten und Ingenieure. Deutscher Bundesanzeiger, 1991

[ISO 10303-STEP] Industrial automation systems and integration - Product Data Representation and Exchange. ISO-Standard No. 10303

[Meißner, Rüppel, Peters] Integrated Structural Engineering based on Object-Oriented Product Models. Proc. of the ICES '95, to be published

[ObjectStore] Reference Manuals for ObjectStore Release 3. Object Design Inc., Burlington, MA, 1993

[Peters, Petersen] Anforderungen an die Integration von Softwarewerkzeugen. Forschungsberichte des VDI, Reihe 20, Band 131, VDI-Verlag, Düsseldorf, 1994

[Rüppel] Objektorientiertes Management von Produktmodellen der Tragwerksplanung. Technical Report 1/94, Institute for Numerical Methods and Informatics in Civil Engineering, University Darmstadt, 1994

[SDAI] Standard Data Access Interface. ISO/CD 10303-22, 1993