Network-Based Fire Engineering Supported by Agents

Udo F. Meissner, Uwe Rueppel, Mirko Theiss
Darmstadt University of Technology, 64287 Darmstadt, Germany (sekretariat@iib.tu-darmstadt.de)

Summary

Building design in Civil Engineering is characterized by the cooperation of experts in multiple disciplines. Close cooperation of engineers in different fields is the basis of high product quality, short development periods and a minimum of investment costs. For each building the engineers have to create a new fire engineering model. The consistent realization of the fire engineering model in all details has high demands on communication, collaboration and building models. Thereby, to preserve the related design models consistent to each other and compatible with the rules of fire engineering is a complex task. In addition, regulations and guidelines vary according to the building location, so the knowledge base must be integrated dynamically into the planning process.

This contribution covers the integration of engineers and design models into a cooperation network on the basis of mobile agents. The distributed models of architectural design, structural planning and fire engineering are supported. These models are implemented as XML-based models which can be accessed by mobile agents for information retrieval and for processing tasks. Agents are provided to all planners, they are enabled to check up the distributed design models with the knowledge base of the fire protection regulations,. With the use of such an agent each planner is supported to check up his planning for accordance with the fire protection requirements. The fire-engineering-agent analyzes the design and detects inconsistencies by processing fire protection requirements and design model facts in a rule-based expert system. The possibility to check the planning information at an early state in the sense of compatibility to the fire protection regulations enables a comprehensive diagnosis of the design and the reduction of planning errors.

1 Introduction

1.1 Research Context

The contribution presented herein is based on the research within the program "Network-based Cooperative Planning Processes in Structural Engineering" [1] supported by the Deutsche Forschungsgemeinschaft (DFG), the German National Research Foundation. This research program supports various projects within the context of the network-based cooperation and bundles different activities in Informatics in Civil Engineering in Germany (http://www.dfg-spp1103.de).

This contribution provides a new approach to manage the distributed planning processes in fire protection engineering based on agent technologies. Hereby, an agent can be seen as the virtual representative of the fire protection engineer in the computer network. During the building design phase the software agent moves through the World Wide Web monitoring related planning activities, searching for distributed planning information, checking up the building design with fire protection rules or communicating with other agents. Shortly, the agent is a mobile and autonomous software robot supporting the fire protection engineer in his engineering tasks.

1.2 Organization of the Paper

The paper is organized as follows. Section 2 introduces the idea of the multi-agent planning system for distributed fire protection engineering. Section 3 describes aspects of fire protection engineering. The fire protection model and the rule model are presented in section 4 and 5. Software agents and agent-based information processing are introduced in section 6. Section 7 und 8 outlines the modelling of information transport agents and the fire engineering agent. Finally, section 9 summarizes the contribution.

2 Multi-agent Planning System for Distributed Fire Protection Engineering

Fire protection engineering is a task that includes planners of many disciplines. After defining a fire protection concept for a building, every planner has to pay attention to the fire protection concept. As shown in the use case of Figure 1 every planner has to check up his work for fire protection deficiencies. In order to support the interdisciplinary processes of the fire protection model evolution in a distributed environment, the integration of all participants of the entire planning process is necessary. Therefore data, methods and knowledge of the partial models have to be coupled for collaboration, while taking into consideration that the planners are spatially distributed.

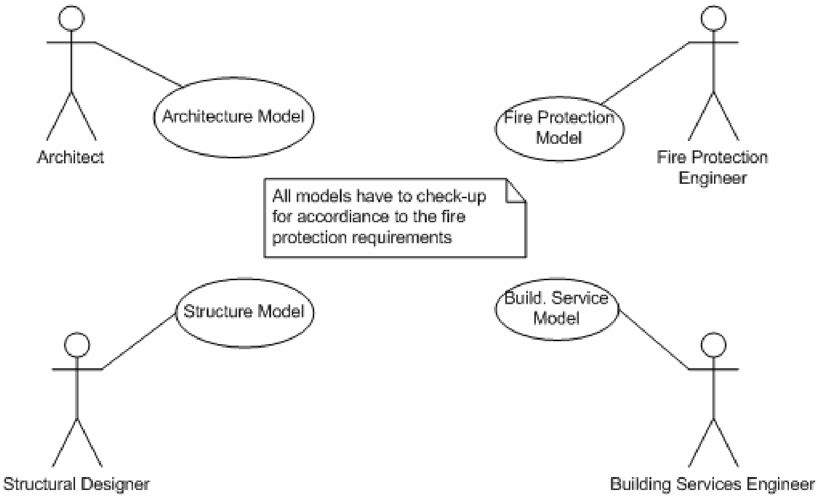

Figure 1: Product models in building design

The concept of collaborative work of distributed planners must support sequential, parallel and iterative planning processes within an integrated information platform. Figure 2 shows the use cases of our approach to support fire protection engineering with a multi-agent planning system. Every planner has a proxy in the form of a stationary agent on his platform. This platform is permanently available. The proxy tries to answer questions on behalf of its planner; in addition it answers queries to model information needed by other planners in the project. The transportation of this model information is supported by mobile agents. The check-up of fire protection requirements is done by a service agent – the fire protection agent. This agent is available for all planners to check up their work for accordance to the defined fire protection model.

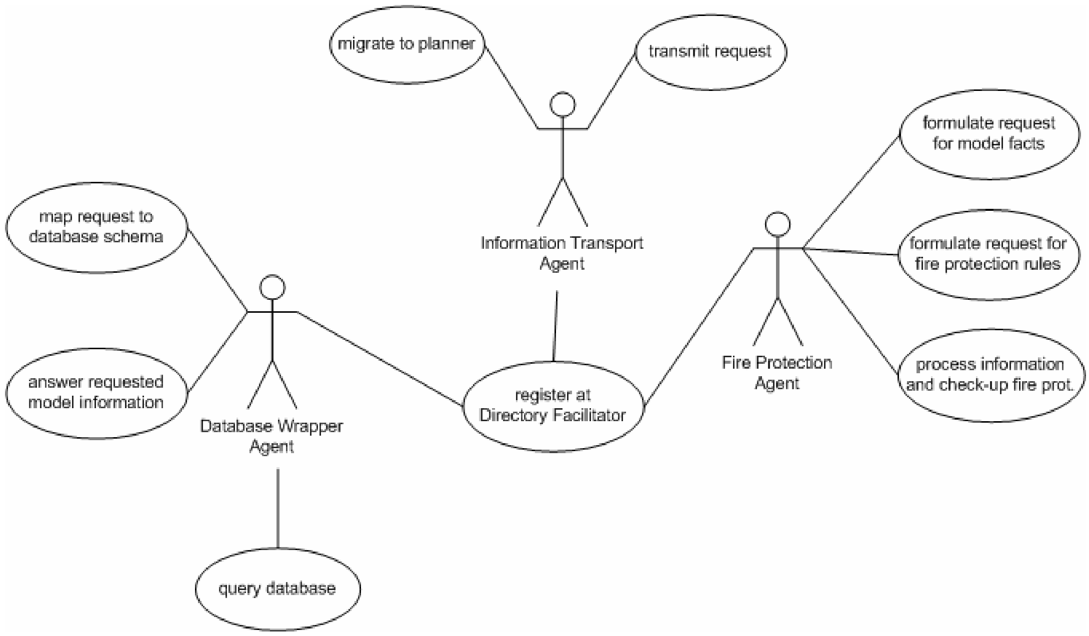

Figure 2: Use case for supporting fire protection engineering by a multi-agent system

3 Fire Protection Engineering

As a result of huge fire disasters fire protection has become one of the central aspects for administrative authorities in the process of licensing the building design. The fire protection planning in building design is the dominant aspect for the prevention of fire and for the protection of life and property in the case of fire. In Germany fire protection is divided into two domains, the preventive and the defensive fire protection. The preventive fire protection contains all structural, technical and organizational fire protection aspects. The fire fighting and rescue are aspects of the defensive fire protection (Figure 3). Within the preliminary planning the elementary requirements for an effective personal rescue and an optimal fire-fighting are created by the preventive fire protection [2]. For that purpose the building geometry and the adjustment of escape routes are important. Furthermore, requirements for the building components are determined with regard to the fire resistance.

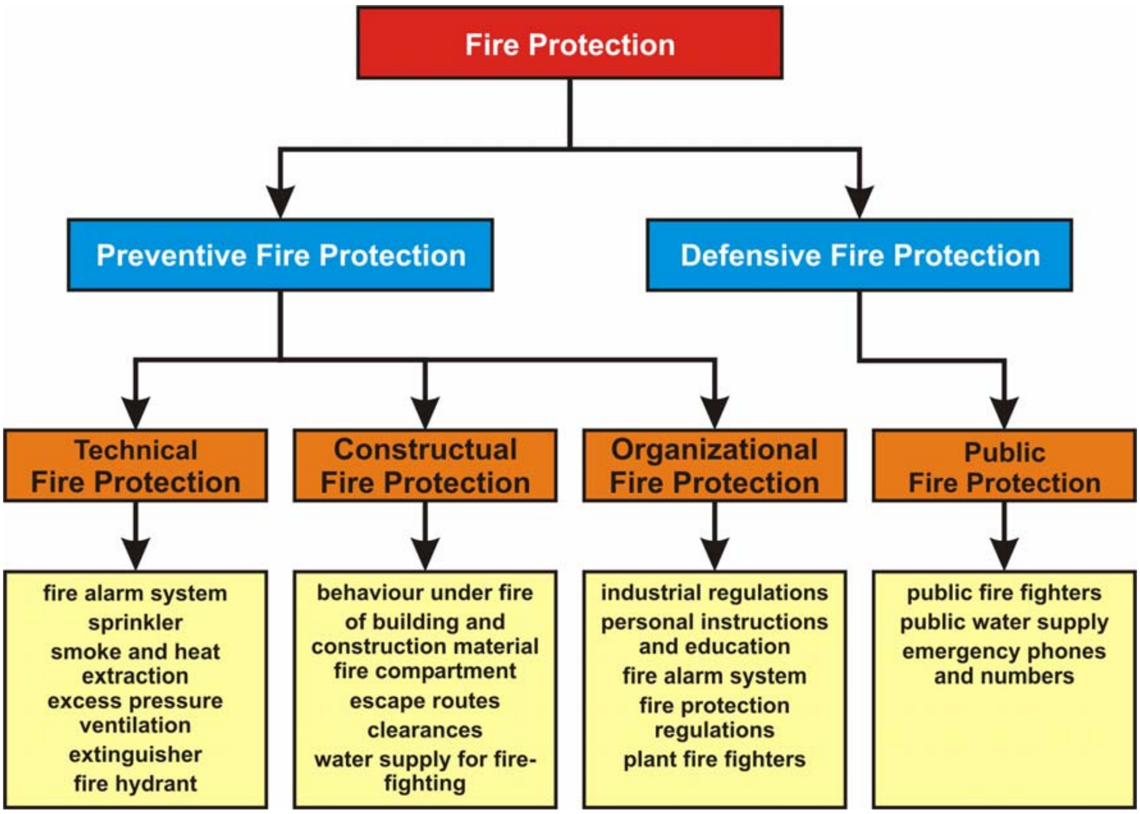

Figure 3: Two dimensional structure of fire protection in Germany

All measures to guarantee the fire protection in a building contribute to the for planning objectives, which improve the safety level of a building [3]. Following to the definition of the protection objectives and the specification of fire protection elements its realization in the detailed planning is an integral component of a holistic fire protection concept [4]. According to the type and size of a project, planners from different fields (e.g. statics, construction, heating, ventilation, electric, and geotechnics) are involved. In order to avoid inconsistent planning states and failures in the flow of information between these planners it is necessary to establish a network for communication. This network must include the activities of planners as well as the state of models and the flow of information. Furthermore, methods to validate the planning results with regard to completeness and effectiveness must be provided. In addition to the described problems in collaboration, a large number of codes and regulations for fire protection models of buildings have to be evaluated.

The focus of the presented approach is the preventive fire protection engineering. This part has to be recognized during the whole planning and construction phase and is the basis for the defensive fire protection.

4 The Fire Protection Model

For every building a fire protection concept has to be created to describe all methods to prevent fire. The interpretation of a text-based fire protection concept is the task of an experienced planner. For the processing of a fire protection concept in a distributed network of planners a transparent model is necessary. Thus, a new fire protection model was developed and implemented in the CAD-system Autodesk Architectural Desktop (Figure 4).

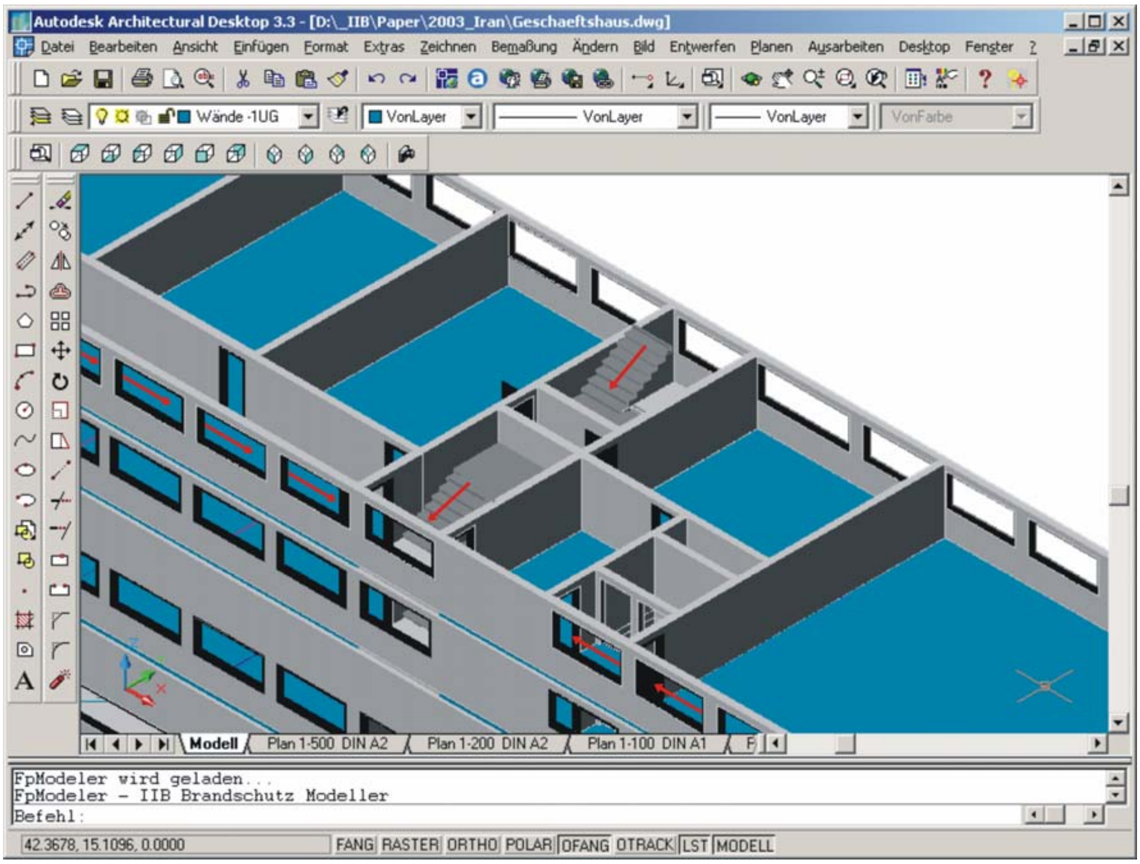

Figure 4: CAD-model with marked escape routes

During creation of the fire protection model the new information is directly associated with the building model. A floor in the building model, for example, can be identified explicitly as an escape route. The definition of an escape route defines special requirements on the linked building components. These requirements must be permanently checked up during the planning process. The floor and the walls are already linked together in the building model, so that this connection can be used to check up these requirements [5]. That means that, for example, a structural engineer finds restrictions due to fire resistance demands in the choice and design of the structural elements. The technical services for facilities have to be designed according to the fire protection model, so that planned rescue routes are free of smoke from flammable components [2].

After instantiating the elements of the fire protection model the planning information has to be made available to all planners. Especially, the civil engineering planning process is characterized by the use of many applications with proprietary data formats. For an effective collaboration the data exchange has to be defined in a neutral data format. According to that the fire protection model is parsed to XML, once based on the schema specifications of the IAI [6] and secondly based on an own schema.

5 Modeling Fire Protection Rules and Requirements

In Germany fire protection as well as building design supervision is the sovereign right of the federal states. Their codes differ in details in every state. In result there is a great number of design codes and special building regulations (for multi-storey buildings, hospitals, etc.) in Germany as shown in Figure 5 To consider these complex standards completely and without errors during the design process is a challenge in collaborative fire protection engineering.

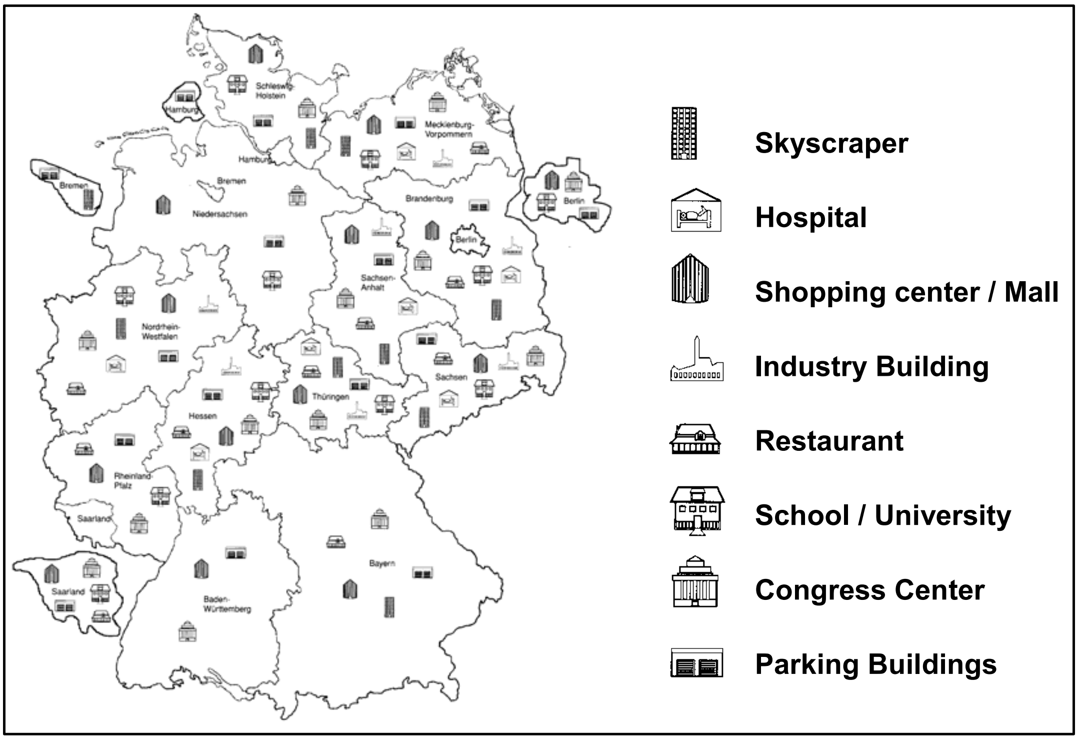

Figure 5: Fire protection codes for building with special purposes in Germany

The definition of a pure data model is not sufficient to check up the functionality of planning information for fire protection. The information defined in a fire protection concept is associated with the rules from codes and guidelines which must also be modeled. The processing of this knowledge can be realized by the efficient combination of model data and rules. Therefore, a three-tier knowledge model for the preventive fire protection was developed (Figure 6). On the one hand this model consists of the data models for the building, on the other hand of the rule model to represent the guidelines and regulations of fire protection.

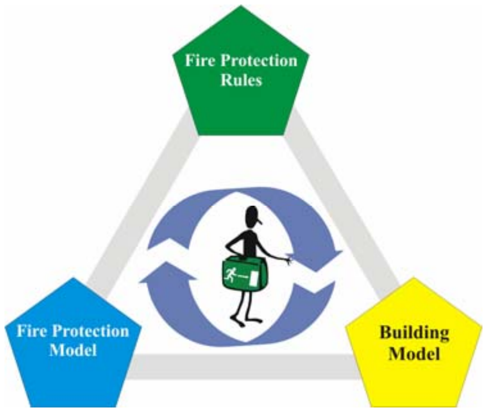

Figure 6: Three-tier architecture to model and process fire protection information

For the application of the fire protection rules to the data models, the rules must be defined and structured in a processable form. Therefore the rules are structured in a first step by the type of the regulation in a second step by the related building element. This approach enables an object-oriented processing of necessary rules. The rule structure optimizes the validation process as well as the communication process for acquiring necessary rules.

By means of a problem-specific user interface (Figure 7) the CLIPS-based rules [7] can be defined in a graphic editor, to be processed and to be integrated in the above described structure [8]. Also complex rules can be set up conveniently by the combination of several rules.

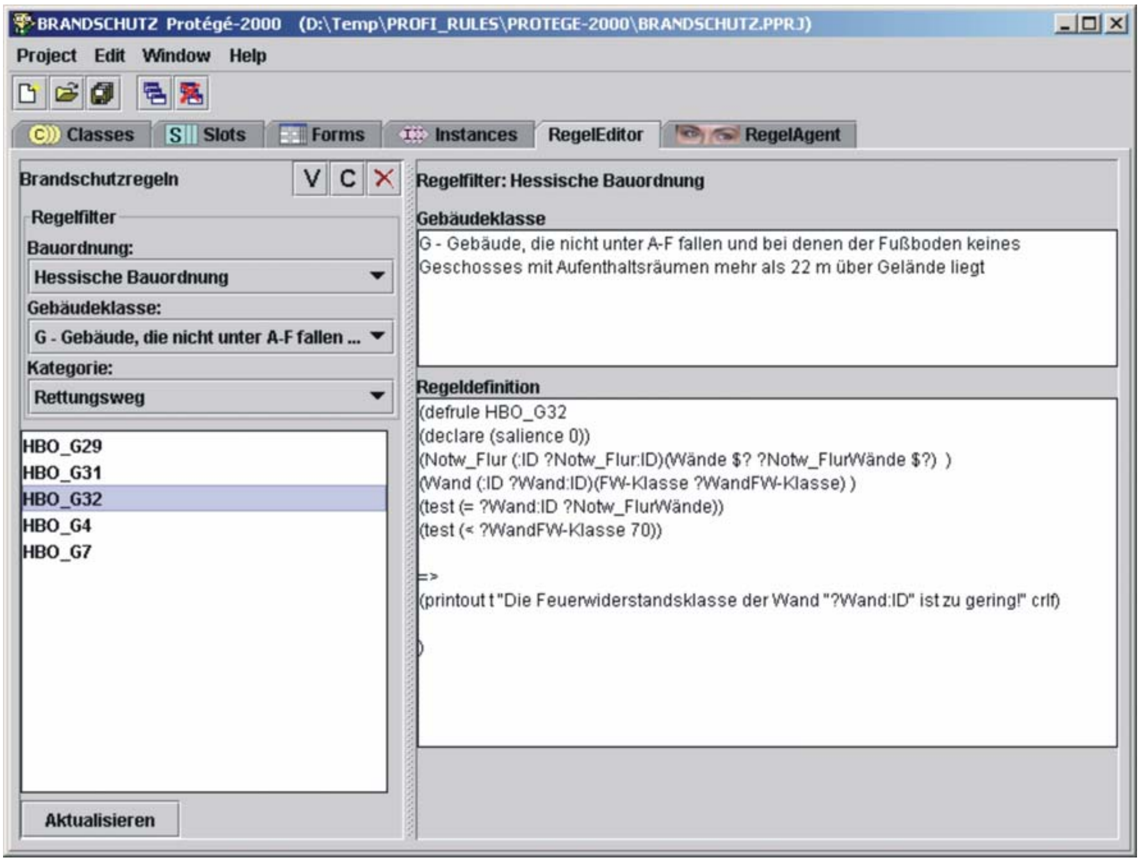

Figure 7: Editor for defining rules of fire protection element validation

6 Agent-based Information Processing

As described in section 2 the distributed planning system for fire protection engineering is developed by the use of software agents. Software agents have the characteristics of delegacy, competency and proactive behavior. Delicacy means the discretionary authority to autonomously act on behalf of the client. Actions include making decisions, committing resources, and performing tasks. Competency is the capability to manipulate the problem domain environment effectively to accomplish the prerequisite tasks, this includes specialized communication proficiency. Proactive behavior is the ability to adapt behavior to reach a goal.

A software agent is a software module which operates in a special software environment, the agent platform. The environment sets up on the operating systems and establishes connections to other software applications or databases for agents. The essential difference between objects and agents is the achievement of objectives. Objects are defined by their attributes and methods. The object encapsulates the state of the attributes and contains a variety of methods which are used to change the state. An object communicates with another through interfaces, to accept commands from the public. An object does not act like an agent from its own motives. Agents decide due to their inner state and due to their environment, whether actions are executed. An agent can also refuse to fulfill orders due to its knowledge [9]. Therefore, an agent encloses the knowledge and objectives to fulfill its intentions.

Most agents do not need to change their place to fulfill their intentions. In some cases it is very useful when an agent can migrate to another platform. For migration the agent stops its tasks and packs its data, both are transmitted to the new platform. The agent continues its work on the new platform. The main benefit of the migration is the possibility to process information right after the retrieval.

Collaborative agents interact with each other to share or exchange information for specialized services to affect a synergism. While each agent may uniquely speak the protocol of a particular operating environment, they generally share a common interface language which enables them to request specialized services from other agents as required [10]. For the representation of the dynamics and the cooperation the Unified Modeling Language (UML) [11] is expanded to the Agent-UML (AUML) [12].

7 The Information Transport Agent

The qualities of agent systems described in section 6 are utilized for the development of a distributed fire protection engineering. As shown in section 2 the different product and rule models have to be retrieved by an agent for processing purposes. The information transport agent outlined in this section offers the service of information transport from the distributed databases to the fire protection agent.

The information transport is divided into two steps. The first step is the transport of the data query from the fire protection agent to the database wrapper agent; the second step is the transport of the query from the database to the fire protection agent (Figure 8). In result of this the information transport agent has two main characteristics: mobility and communication.

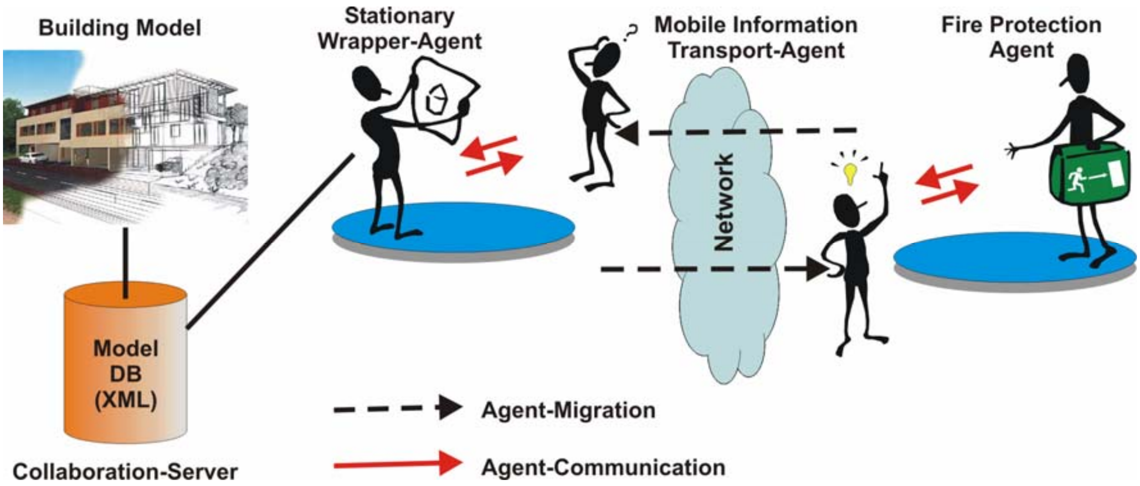

Figure 8: Data retrieving scenario supported by stationary and mobile agents

In [13] this communication is shown in detail. The information transport agent uses the same ontology and interaction protocols as the database wrapper agent described in this article. In this contribution the mobility will be shown in detail. The multi-agent system Jade [14] used for this planning system offers no inter-platform mobility from itself. The implemented mobility supports only intra-platform mobility, which means that an agent can migrate between different containers on the same platform but not between different platforms.

Jade and its source code is distributed under the Lesser General Public License (LGPL). We use it to add inter-platform mobility to the Jade platform by adding some components to the Jade environment. Jade includes an interface that allows external applications to use Jade as a library and to launch the Jade runtime from the application itself. This wrapper agent defines a proxy object whose methods are permitted to access the agent and the platform. A new method now is enabled to restart a migrated agent.

The control interface for agent migration is the developed agent migration service. Every platform with inter-platform agent migration support has to instantiate this service. The service offers the sending and receiving of agents. The process of agent migration themselves contains several steps: After starting the migrations process, the agent class and all inner classes are packed. To restart the agent all starting parameters also have to be saved and packed. Before sending the agent all data, collected on the platform, have to be packed. The whole resulting part is transported as the content of an ACL-message. The content is hashed by a MD5 hash key for security reasons. The message finally will be send to the agent migration service of the destination platform. The process of reactivating the agent after migrating is right the reverse way as the sending process.

By the use of the communication interface, described in [13] and the shown migration service, an information transport agent is enabled to query and to receive model data. The agent gets his objective by the fire protection agent to query for diverse model information. The transport agent analyses the address of the building planner and migrates to his platform. On this platform the agent starts a communication with the local database wrapper agent and hands out the query for the model data. This query will be mapped to the local database schema by the database wrapper agent to process the query. The result will be mapped to the public building model ontology and passed to the waiting information transport agent. This agent migrates back to the fire protection agent with the result. The processing of this data is described in the next section.

8 The Fire Protection Agent

To enable the validation of fire protection requirements for all cooperating planners, the fire protection agent has been integrated into the multi-agent planning system. This agent has to support the planners in building design with all tasks shown in the use case of Figure 2 for the fire protection purposes.

To do this, the agent has to be enabled to process the modeling facts in accordance with the requirements from the design codes und the fire protection model of the building. As described in section 5 the rules in fire protection are declaratively styled. Rule-based expert systems are well known in processing declarative rules. The Java-based Expert System Jess (Java Expert System Shell) [15] is one of the most used rule-based expert systems and the reference implementation for the Java Rules Engine API [16]. Jess processes rules formatted in the CLIPS-schema [7], so the rules modeled in section 5 can be processed by the Jess engine. Another advantage is the Java-based architecture of Jess. The Jess API can be integrated directly in an agent of the Jade system. Rule-based systems act with rules and facts. Facts describe the state of a domain, the rules the context of the domain and the facts. A sample for a fact is the edge length of a square; the corresponding rule is that every edge of a square has the same length.

In order to process fire protection knowledge the fire protection agent has to appraise information from all involved models. These are the facts from the planning models and the relevant rules from the rule model. At first, the relevant fire protection element has to be identified in the fire protection model. Thereafter, all relevant fire protection rules have to be retrieved from the fire protection rule model by the information transport agent and the corresponding database wrapper agent. In addition the corresponding planning model elements have to be retrieved from the building model databases in the same way. After retrieving all information the fire protection agent checks whether it has all facts to process the rules. This is an iterative process, so step by step all rules and facts needed for the fire protection requirement check-up are transported to the fire protection agent. As a result the fire protection elements have been checked for accordance with the valid fire protection regulations. This approach supplies the planner with the possibility to check his design for consistency with the required fire protection model.

9 Conclusion

The contribution shows an approach for supporting planning activities in building design by multi-agent systems. The consistent definition and development of fire protection models in the building planning process is essential for the protection of life and property in the case of fire. The developed prototype makes fire protection planning information available in a network. Furthermore, it enables the processing of this information for all planning partners. This is afforded by agents supplying processing methods within the multi-agent planning system. The planner can check his design at every planning state for accordance with the fire protection requirements by use of a fire-protection-agent. The fire-protection-agent analyzes the design and detects inconsistencies. The possibility to check the planning information at an early state against the knowledge of the fire protection regulations enables a comprehensive diagnosis of the design and the elimination of planning errors.

10 References

[1] Deutsche Forschungsgemeinschaft (2000): *Schwerpunktprogramm 1103 - Vernetzt-kooperative Planungsprozesse im Konstruktiven Ingenieurbau*, http://www.dfg-spp1103.de (status 1.4.2004)

[2] Schneider, U. and Lebeda, C. (2000) *Baulicher Brandschutz*, Verlag W. Kohlhammer, Stuttgart, Germany.

[3] Kingsohr, K. and Messerer, J. (2002) *Vorbeugender baulicher Brandschutz*, Verlag W. Kohlhammer, Stuttgart, Germany.

[4] Loebbert, A. and Pohl, K.D and Thomas, K.-W. (2000) *Brandschutzplanung für Architekten und Ingenieure*, Verlag Rudolf Müller, Köln, Germany.

[5] Rueppel, U. and Meissner, U.F. and Theiss, M. (2002) *An Agent-based Platform for Collaborative Building Engineering*. In: Proceedings of the 9th International Conference on Computing in Civil & Building Engineering, Taipei, Taiwan.

[6] International Alliance for Interopability (IAI): *ifcXML XML-Schema Definition - Version 1.02.* http://www.iai-international.org – ifxXML, status: 06/2002.

[7] CLIPS (2002) *CLIPS - A Tool for building Expert Systems*, Gary Riley, http://www.ghg.net/ clips/CLIPS.html, status: 12/2002

[8] Theiss, M. (2002) *Dynamische Integration von technischen Wissen in den Bauplanungsprozess.* In: Bilek: 14. Forum Bauinformatik, Fortschritt-Bericht VDI, pages 117-123, Bonn, Germany.

[9] Bigus, J.; Bigus, J. (2001): *Intelligente Agenten mit Java programmieren*, Addison-Wesley, München.

[10] Brenner, W.; Zarnekow, R.; Wittig, H. (1998): *Intelligente Software-Agenten*, Springer Verlag, Berlin.

[11] Object Management Group (OMG): *Spezification of the Unified Modeling Language (UML),* Version 1.4, 2001.

[12] Odell, J.; Parunak, H.V.D.; Bauer, B. (2000): *Extending UML for Agents*, AOIS Workshop at AAAI.

[13] Hartmann, D.; Meissner, U.F.; Rueppel, U.; Bilek, J.; Theiss, M. (2004): *Integration of Product Model Databases into Multi-Agent Systems*. In: Proceedings of the 10th International Conference on Computing in Civil & Building Engineering, Weimar, Germany.

[14] Bellifemine, F. (2004): *Jade - Java Agent Development Framework*, Tilab Italia, http://jade.cselt.it (as of 1.4.2004), Turin, Italien

[15] Friedman-Hill, E.J. (2003): *Jess in Action – Java Rule-based Systems*, Manning Publucations, 2003.

[16] Community Development of Java Technology Specifications (2002): *JSP 94 - Java Rules Engine API,* http://java.sun.com/jcp/, Bea Systems.