IMPLEMENTING THE LEXICON FOR PRACTICAL USE

Kees Woestenenk
STABU, The Netherlands

ABSTRACT: The LexiCon can be regarded as an Object Library for the construction industry, similar to STEPlib for the process industry, which is based on STEP part 221, and the POSC/CAESAR library for the off-shore industry. The model behind the LexiCon, however, is much simpler than the models of the other libraries, allowing for construction experts to take an active part in populating the LexiCon with objects which are of interest for the construction industry.

The LexiCon has been introduced at several occasions: the ECPPM '98 conference in Watford, UK, the ICIS Delegates Assembly 1999, Prague, the CEC99 conference in Espoo, Finland, as well as the IAI, ISO TC184/SC4 (STEP) and ISO TC59/SC13. SC13 earlier formulated the work item 12006-3 "Building construction – Organization of information about construction works – part 3: Framework for object oriented information exchange" and installed at its meeting in June 1999 in Vancouver Working Group 6 for this task. SC13 also established a 'Standing Conference' of interested parties on the subject, such as IAI, ICIS, ISO TC184/SC4/WG3/T22, ISO TC10/SC8, CIB and aecXML. The Standing Conference had its first meeting in June 1999, in Vancouver and a second one in October 1999 in München, Germany. WG6 had its first meeting in October 1999 and a second one in January 2000, both in Ede, The Netherlands. The LexiCon structure is accepted as a starting point for the proposed framework, together with the Epistle Version 3 model and the Swedish BAS-CAAD model.

The idea behind the LexiCon is to provide a 'common language' for storing and exchanging data between applications, between participants in construction processes and between owners and users of products resulting from construction activities. This paper will discuss how applications can use the structure of the LexiCon as well as the contents. It will show that data about a 'Built Object' can be stored within the LexiCon structure troughout the whole lifetime of that Built Object, from brief, design, assembly and use till dismantling or demolishment. Doing so, broadens the meaning of the terms 'brief' to 'demolishment' from project-related to object-related, or in other words, it will show that these terms apply to the life stages of every single (Built) Object. On the other hand, applications will continue targeting distinct processes. Thus an application targeting the client's brief for a building will differ from an application specifying the requirements of a curtain wall (which also could be regarded as a kind of client's brief, but this time with the curtain wall as the subject), while still using the same data structure. Applications based on the LexiCon will instantiate the Built Object classes in the LexiCon or may instantiate their own classes derived from the LexiCon classes, using the LexiCon's Functions and Quantity classes.

Another usage of the LexiCon will be the provison of general information, such as building regulations, product information, cost data and quality assessments. Using the LexiCon will assure that for a single Built Object class all relevant data will be easy to find, as long as the application takes care of associating these data with that class.

KEYWORDS: classification framework, object orientation, product models

1. THE LEXICON RELATED TO STANDARDIZATION ACTIVITIES

Information technology related standardization activities for the construction industry can be divided into three major streams: product modelling, EDI and classification. STEP and IFC are based on product models. EDI uses its own message structure. Specification systems, cost information systems and product information systems are generally structured according to a classification system.

A product model provides an image of real world things. For the construction industry 'products' are the things that result from, or are subject to construction activities. A construction product varies in size and complexity from raw resources, like sand, to large and complex structures, such as an airport. There is a tendency to take the whole lifecycle of the product, and henceforth its behavior in its use phase, into account, which increases the complexity of the model rather dramatically. Main aspects of the product are structure, behavior and changes due to activities and processes.

For EDI the grammar and the tokens of a message are of major importance. Most messages currently deal with logistics in construction processes, varying from ordering and transport of products (from manufacturers) to tendering.

Classification systems provide views on things, which are means to find (or group) things according to certain characteristics. The construction industry has developed quite sophisticated classification systems in the past; apparently these systems fitted well in the way the industry handled information.

Merging the three streams together leads to a picture in which the product model takes care of the structure of things, whereas EDI provides messages related to these things and a classification system groups these things according to certain characteristics. The streams are – at least partly – complementary to each other. The truth is, however, that these three streams did not merge yet, and are more or less separately maintained by different bodies.

On June 4th 1999 a meeting was held in Vancouver, Canada, sponsored by ISO, and organized by ISO TC59/SC13 (Building Construction - Organization of information about construction works). Invited to this meeting were representatives of ISO TC184/SC4/WG3/T22 (STEP/AEC), IAI, CIB W78, ISO TC10/SC8, ICIS and EDIBUILD. The intention of this meeting was to start merging the streams mentioned above. The result of the meeting was the establishment of a Standing Conference of interested parties, to meet on a regular basis. ISO TC59/SC13 was urged to actually make a start with their work item WI 12006-3 *Building construction – Organization of information about construction works – Part 3: Framework for object oriented information exchange.* At the meeting of ISO TC59/SC13 on June 6th 1999, also in Vancouver, it was decided that a new working group, WG6, has to proceed with work item WI 12006-3, as urged by the coordination meeting.

The initiative for an 'object oriented framework' came originally from STABU. At the ICIS-conference in 1995 in Berlin, Germany, STABU proposed to develop a 'List of Parts' (Woestenenk, 1995). The idea of the 'List of Parts' was further developed in the Netherlands by an organization called BAS (*Bouw Afsprakenstelsel)*, a cooperation between groups dealing with all kinds of construction information (geometry, quality, costs). STABU is one of the participants. The work in this group resulted in what is now called the LexiCon. This LexiCon is one of the starting documents for WG6.

Part 2 of the Standard developed by WG2 of ISO TC59/SC13 (ISO/CD 12006-2:1997) deals with 'traditional' classification. The approach taken by WG6 differs from the view oriented approach of traditional classification, and is much more akin to the product model approach as we know from STEP and, more precisely, the model behind the class libraries of EPISTLE Version 3. With respect to the needs of the construction industry, WG6 strives the framework

to be a subset of EPISTLE V3. On the other hand, this framework should also be compatible with other AEC specific developments, such as the APs already developed within STEP (e.g. AP 225) and the Industry Foundation Classes (IFC) developed by the International Alliance of Interoperability (IAI). Also, the EDI message structure should fit into this framework. The Standing Conference is meant to provide a discussion platform for this to be achieved.
The LexiCon will be made compliant with the results coming from WG6 and the discussions in the Standing Conference.

2. CURRENT STATUS OF THE LEXICON

The LexiCon has been presented at several international conferences (lately at the ECPPM '98 conference in Watford, UK, CEC 99, Espoo, Finland, August 1999, and INCITE 2000, Hong Kong, January 2000). It has also been discussed within the International Construction Information Society (ICIS), within the International Alliance of Interoperability (IAI) and, as mentioned above, within ISO TC59/SC13 and ISO TC184/SC4. A description of its structure can be found in (Woestenenk, 1999). Following is a brief description of its structure and contents.
The LexiCon describes so called Built Objects, using standardized attributes. Built Objects are concepts relevant to the construction industry. These concepts are known to the industry by their names; the problem is, however, that these names are interpreted differently by different people and in different circumstances. What the LexiCon tries to do is to provide an unambiguous description of a named concept through a set of attributes. Attributes should be computer interpretable. Defining the concepts this way makes these concepts independent of their names, thus allowing for synonyms and homonyms, as well as language independency. In earlier publications the following EXPRESS-G model (see Figure 1.) was shown describing the model on which the LexiCon is based.

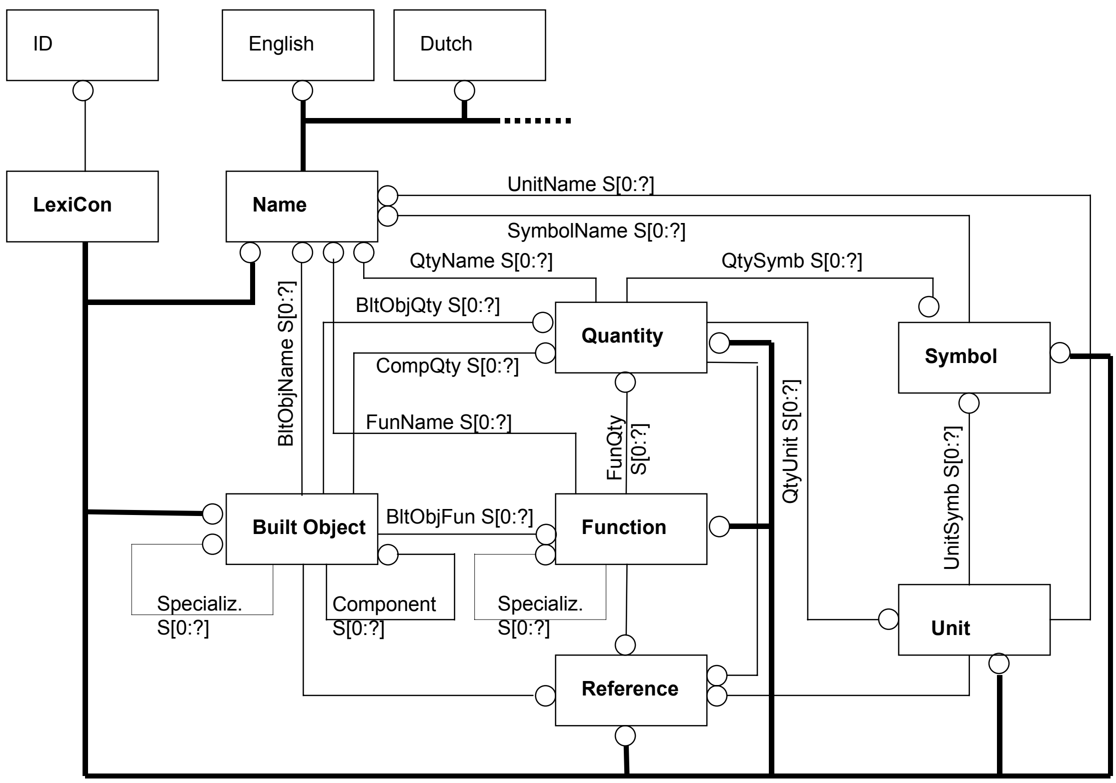

Figure 1. Early LexiCon model

A new version of the model will follow the recommendations of ISO TC59/SC13/WG6, the working group responsible for the development of the structure of an object oriented classification framework.

With a not yet fixed structure a tool has been developed by STABU to populate the LexiCon. This tool is a Windows desktop application, showing the concepts as classes in a specialization hierarchy and the description of each class in an adjacent pane (see Figure 2.).

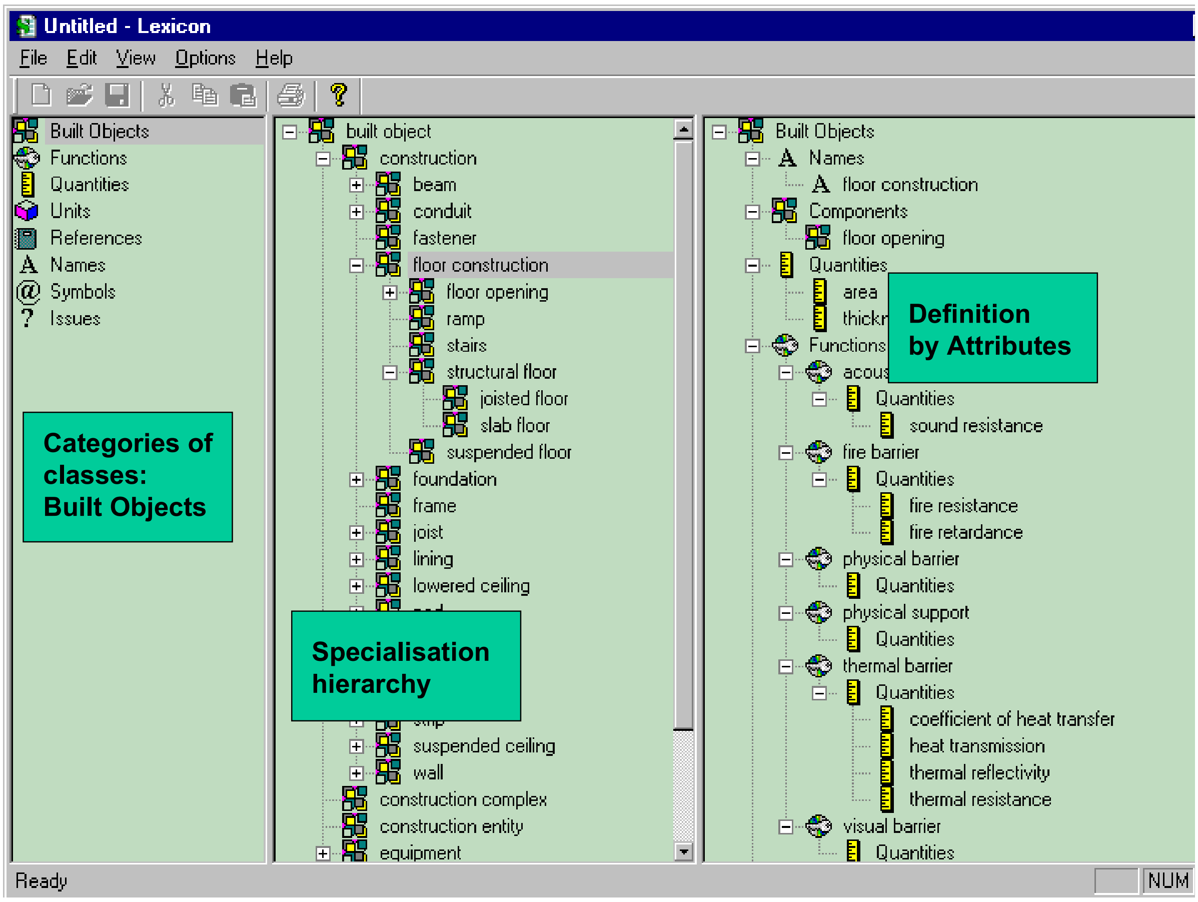

Figure 2. The LexiCon tool

Ultimately, a Web-based tool will be developed to allow for publishing the LexiCon on the internet. A preliminary version of such a tool has been developed by TNO Building and Construction Research (BOUW), The Netherlands.

At the time of this writing the current content of the LexiCon consists of examples. The idea is to populate the LexiCon after discussion and approval of the contents on an international platform, and with national bodies implementing national versions in their native languages.

3. A POSSIBLE USE OF THE LEXICON: SPECEXPLORER

Based on the LexiCon an application prototype, called SpecExplorer, has been developed by STABU. With SpecExplorer it is possible to write any non-geometrical specification in any stage of the life cycle of a facility, including the client's brief, space descriptions, performance specification, descriptive specification and facility management information. It is capable of maintaining different states of information next to each other. Specifications may be linked to the type information from the LexiCon, these links are optional, however. Whether or not a specification is linked to the LexiCon, the structure of SpecExplorer specification is similar to that of the LexiCon, although SpecExplorer extends the LexiCon structure wherever this is applicable.
As a prototype the functionality of SpecExplorer is limited and rather basic. The idea is to split this prototype into a general specification engine, with plug-ins dedicated to special tasks.

3.1 The structure of SpecExplorer

The structure of SpecExplorer is shown if Figure 3, and is similar to the structure of the LexiCon.

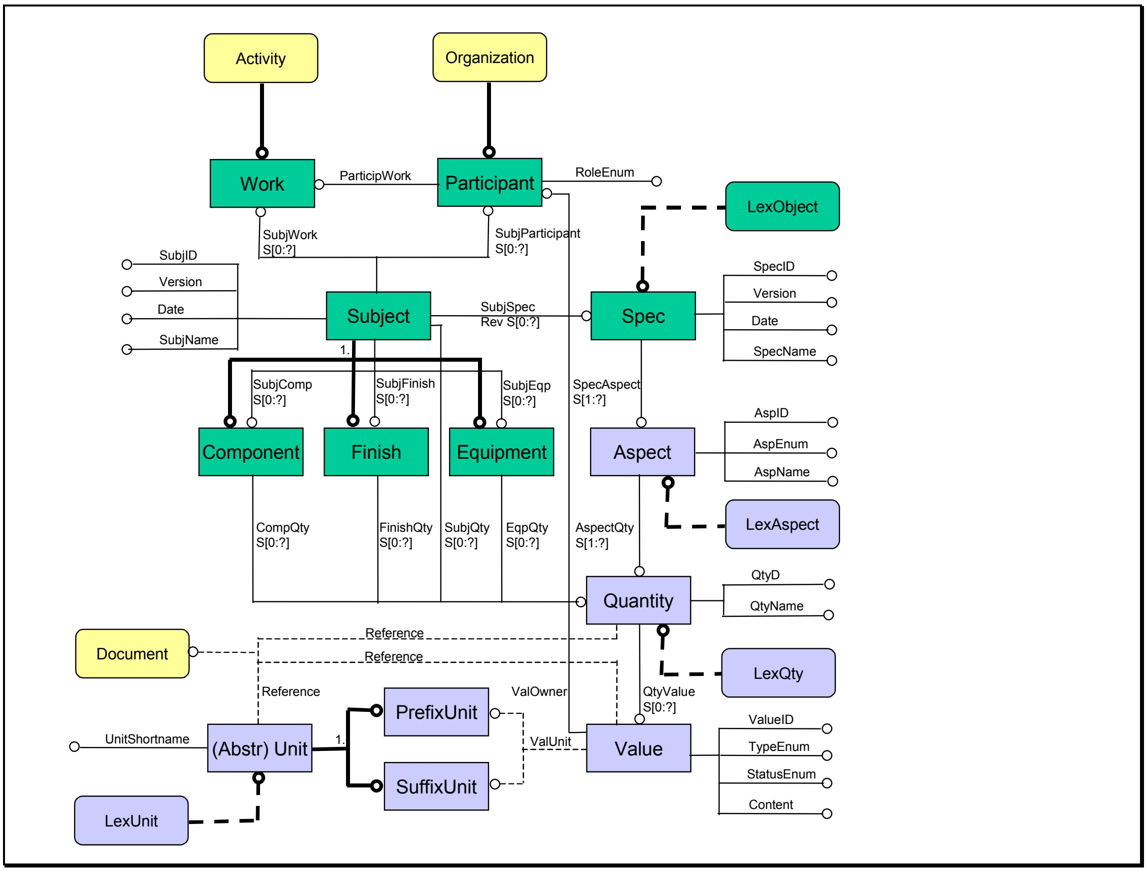

Fig. 3. EXPRESS-G Diagram of SpecExplorer

Subjects are the instances of entities being specified. A Subject can either be a Space, a Physical Part (permanent or temporary) or Equipment or Furniture. *Components*, *Finishes* and *Equipment* are defined subtypes of Subjects. These subtypes may also be associated with Subjects. Through the Subject-Component association a composition/aggregation hierarchy can be modelled. A Component has a *part-o*f relationship with a Subject. A Component can be part-of more than one Subject. Finishes differ from Components in that they cannot exist on their own, but are always dependent of a Subject that is the base for the Finish. Equipment on the other hand has a much more loose relationship with the Subject: the Subject is not functionally dependent of Equipment nor is Equipment functionally dependent of a Subject. Subjects are specified in two ways: either through associated *Specs*, or directly through associated *Quantities*. A Spec is a specification describing a number of Aspects of a Subject. A Subject can be described by a Spec, a Spec might be shared by a number of Subjects. For example, a number of *door* instances in a project might share the same Spec, when they only differ from each other by location. A Spec can optionally be derived from the specification of a Built Object in the LexiCon. A Spec *consists-of* a number of Aspects. An Aspect is the context for a number of Quantities. For example, a door might play a role as a fire barrier in a building, in this context the fire-resistance performance of the door will be a relevant Quantity. Aspects can optionally be derived from Aspects in the LexiCon (where they were originally called Quantity Sets). A Subject can also be directly described by Quantities. These descriptions can be used in addition to Specs. In the example of the *doors* above the location could be an additional Quantity associated with each instance. A Spec also might have Quantities with different Values per instance. In this case the Value will be associated through the direct association between Subject and Quantity. For example, the earlier mentioned location of the doors could be part of the Spec, allowing to be different for each instance.

A *Quantity* is a descriptor type *consisting-of* one or more Values. Quantities can optionally be derived from Quantities in the LexiCon. As in the LexiCon, a distinction can be made in Quantity types, such as Properties (e.g. shape), Performances (e.g. fire resistance) and Constraints (e.g. contract conditions).

A *Value* provides the actual data of a Quantity. A Value can be expressed in a Unit, which is either attached as a prefix, a suffix or both. Some Values have no Units at all (e.g. constraints). Units can optionally be derived from Unit types in the LexiCon. A Value has an associated State: *required*, *proposed* and *actual*. The State describes how the Value should be interpreted. There can be only one Value per State. Another distinction between Values is their format. For example, some Values are numeric and can be used in calculations, others, such as text, are only human interpretable, others, such as classes, are interpretable through reference and some could be expressions.

Quantities, Values and Units might refer to external documents, such as Standards. For example, a standard might define a number of strength classes, and a Value could refer to one of these classes through the symbol provided for that class in the standard (class B25, NEN 5950). In this case, class is the Unit in which the Value is expressed.

Subjects might be further associated with *Participants* and *Work*. In addition, Participants can also be associated with Values. The association with Participants can either describe the involvement of a Participant with a Subject, or the rights of a Participant with regards to a Subject or a Value (e.g. ownership or right to modify). The association with Work can describe tasks associated with the Subject, e.g. the task to design the Subject, or to assemble the Subject from its Components, or to clean the Subject as part of the maintenance work. Work also implies the involvement of Participants. Participants are derived from Organizations, and Works are derived from Activities, both of which are concepts external to SpecExplorer.

3.2 The functionality of SpecExplorer

A major difference between SpecExplorer and the LexiCon is the way data are presented to the user in both applications: the LexiCon presents types within a specialization hierarchy, whereas SpecExplorer presents its subjects in a composition hierarchy.

SpecExplorer allows to set up a composition hierarchy and to provide specifications of each node in that composition hierarchy. The nodes represent the Subjects in the earlier described structure. The root of the composition hierarchy represents the Subject as a whole, which is called the Project. The terms Subject and Project are treated as more or less synonyms here: a Project is a Subject with a set of associated tasks. Although SpecExplorer is developed to describe Projects, it is in no way restricted to that use. It could be used for any specification of any subject. In most cases, however, the SpecExplorer will be used for Project specification.

In the composition hierarchy, the root node represents the upper boundary of the Subject/Project. Similarly, the ultimate leaf nodes represent the lower boundaries. However, separate Subjects/Projects might be linked through their nodes, hence a leaf node could point to another SpecExplorer database for a more detailed specification, and the root node could point to a larger Project of which it is a part. The result would be a distributed database, linked through Subjects, which would be an alternative for distributed databases organized by discipline.

Each node in the composition hierarchy will have its own specification. This combination is complete in itself, it should not depend on other external data (except referenced data outside the model). As mentioned above, the specification of a Subject may be a combination of a shared Spec and some instance specific Quantities. The use of a shared Spec prevents redundant specifications whereas many Subjects normally are of certain types with a number of instances.

SpecExplorer provides links with the LexiCon, providing the possibility of using the LexiCon types as templates for specification. Links are dedicated, which means that subjects are derived from Built Objects in the LexiCon (through Specs), while Aspects, Quantities and Units can be derived from their corresponding types in the LexiCon.

The current implementation of SpecExplorer is a prototype. In Figure 4. a screen dump is shown.

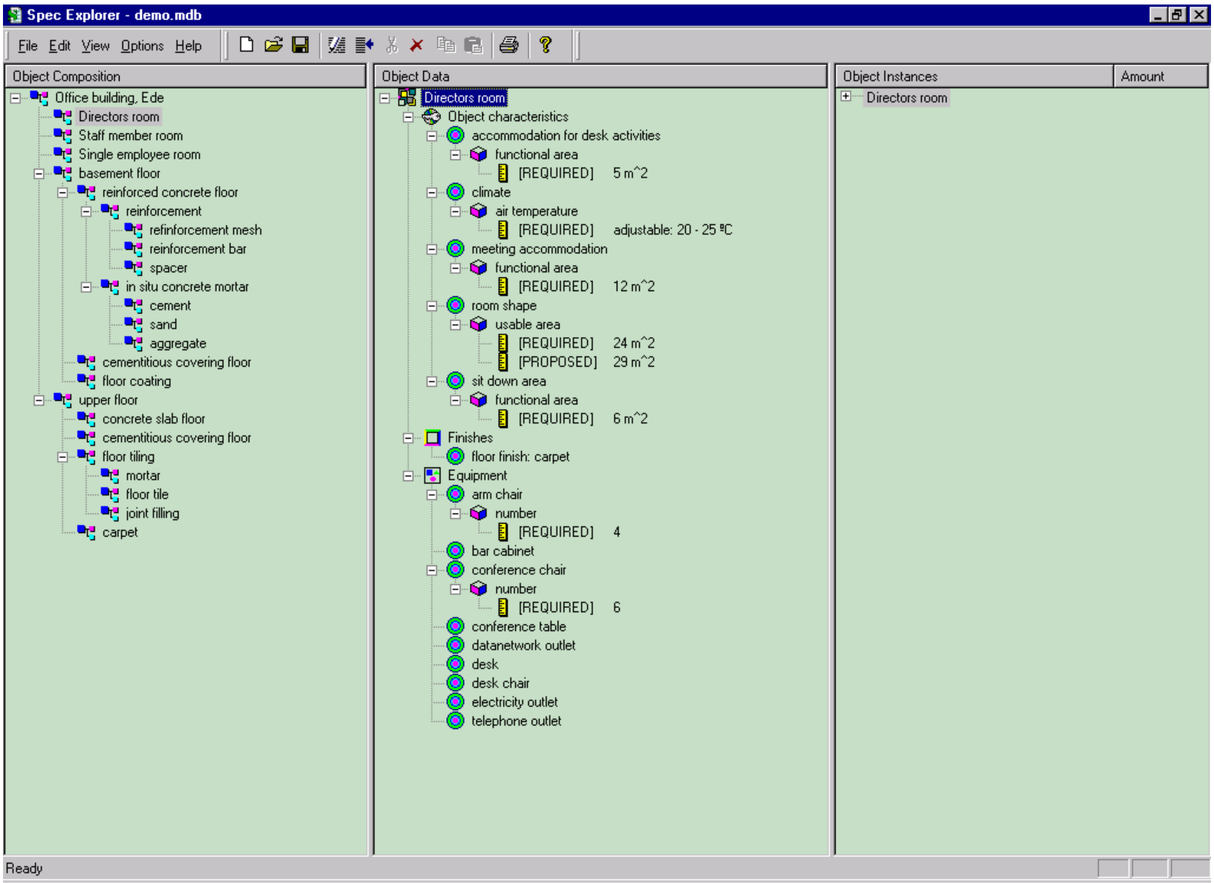

Fig. 4. Screen dump of SpecExplorer

SpecExporer combines a variety of what are separate descriptions and specifications in current practice. For example, in current practice a description of Spaces is contained in a separate document, next to other documents, such as a tendering document containing the specification of work, and yet another document containing the client's brief. Figure 4. shows a combination of these aspect views in SpecExplorer. In the middle pane the specification of a space (Directors room) is shown. In this specification there are some Values having the Status *required*, and other Values having the Status *proposed*. Quantities with Values of the Status required are actually part of the Client's brief, whereas Values with Status proposed follow from the design. Figure 4. shows also physical parts (basement floor, upper floor) as well as their composition. In current practice the composition of a physical part is normally provided in a descriptive specification, the required performance of a part, on the other hand, would be given in a performance specification. In SpecExplorer these aspects are still visible yet combined in a node, with the composition in the left pane, and the (performance) specification in the middle pane (not opened, therefore not visible here).
As a prototype, SpecExlorer combines all kinds of specification types in a single application. For a real application this would result in a very complex application. Thus, for the further development of SpecExplorer a component based application is anticipated, consisting of an engine that could be used to store and retrieve data in/from data storage and plug-in applications for dedicated uses. The engine will represent the structure, as described above, and the plug-in applications would provide interfaces similar to the current dedicated applications, such as briefing tools, specification systems, scheduling applications and the like.

3. Conclusion

SpecExplorer shows one possible implementation of the LexiCon for use in every day practice. Although still in its infancy, it shows the power of an object oriented approach, capable of the integration of construction related information for the whole life cycle of a facility.

Another usage of the LexiCon will be the provison of general information, such as building regulations, product information, cost data and quality assessments. Using the LexiCon will assure that for a single Built Object class all relevant data will be easy to find, as long as the application takes care of associating these data with that class.

REFERENCES

Woestenenk, K.: 1995, *Proposal for international conversion tables for parts and functions*, STABU, The Netherlands.

Woestenenk, K.: 1999, *The LexiCon,* STABU, The Netherlands.