OBJECT-ORIENTED DATA-INTEGRATION BETWEEN DIGITAL ARCHITECTURAL PHOTOGRAMMETRY AND CAAD

Urs Hirschberg, Dipl. Arch. ETH
Architecture and CAAD,
Swiss Federal Institute of Technology, ETH Zürich,
Switzerland

KEYWORDS: Architecture, Automation, Knowledge-Base, CAD, Modeling

ABSTRACT

The integration of digital architectural photogrammetry with computer aided architectural design (CAAD) has great potential for a variety of architectural applications. While both, digital photogrammetry systems (DIPS) as well as CAAD programs essentially rely on point-coordinates in three Dimensions, their datastructures still differ significantly. So the data-integration becomes non-trivial, as soon as more than the most basic geometric information needs to be shared between the systems. This is necessary in semi-automated photogrammetry systems, where an object-oriented structure, similar (but not identical!) to the one found in todays CAAD systems, can provide qualitative guidance in the computer measurement.

In this paper an approach to an object-oriented data-integration between a standard CAAD program and a DIPS is described. It was developed in the framework of an ongoing research project for which this integration is essential. The data-integration relies on the concept of a knowledge base, which enables an object-oriented structuring of geometric data suitable for photogrammetric evaluation. The knowledge-base can be shared between the CAAD system and the DIPS, thus enabling a very efficient run-time transfer of data. This run-time transfer between the systems, which is necessary for the modelling of approximations, is done by means of a specially developed transfer protocol. It could theoretically be used by any 3D-CAD package, not just the one used in the prototype implementation.

1. THE PROJECT DIPAD

1.1 Introduction

The Integration of Computer Aided Architectural Design (CAAD) and digital photogrammetry has long been recognized as a powerful technology. Its possible applications range from the documentation of historical monuments to the generation of digital town models and virtual reality environments.

The commercial programs for architectural photogrammetry available on the market today are restricted to manual measurement [1][2]. Therefore the accuracy of the delivered data largely depends on the dexterity of the operator. On top of that the application of these programs is usually quite time-intensive and requires special training. These shortcomings may account for the fact that these programs are still rather rarely made use of in the architectural practice.

The system for digital photogrammetry and architectural design (DIPAD) currently under development at the Swiss Federal Institute of Technology aims to overcome these shortcomings. The system not only transfers traditional photogrammetric techniques which usually involved very expensive mechanical equipment to a computer-terminal (which is, essentially, what the mentioned commercial packages do). Most importantly it takes advantage of the digital format of the data to increasingly automatize the time-consuming measurement process.

In this paper the main focus will be on the integration of the datastructures of the employed digital photogrammetry system (DIPS) and a standard CAAD-system. For the semi-automatic measurement techniques employed in our DIPS, the point and line-based data-structures traditionally found in digital photogrammetry systems are inadequate.

We will argue in this paper, that the data-integration with CAAD, or in fact the introduction of any object-structure with semantic implications in the DIPS, is the prerequisit for a qualitative control and for an increasing automation of the photogrammetric processing.

1.2 Principles of DIPAD

The research environment DIPAD aims to develop an easy to use tool for the photogrammetric generation of accurate, reliable and well structured 3D CAAD models of architectural objects.

The system essentially consists of a DIPS and a standard CAAD program with a true programming interface (AutoCAD). With the combination of these two systems, a model-based approach is pursued. The human operator interprets the scene by modelling a geometric approximation of it in the CAAD program (figure 3). The measurement is then handled automatically by the DIPS, based on this approximation. This procedure is referred to as 3D feature extraction.

1.3 CAD-based 3D feature extraction

In order to establish feature extraction that provides for high precision as well as for reliability, a top-down strategy is chosen. The semantic object model is used to detect the features described by this model. Thus only relevant features are extracted and redundant information and data complexity are reduced to a minimum.

The three-dimensional position of the object is derived by simultaneous multi-frame feature extraction, whereby the object model is reconstructed and used to triangulate the object points from corresponding image points.

It is evident that in most cases linear boundaries (edges) of an architectural feature contain more information than the vertices (corners) of this feature. Although edges are only a small percentage of the whole image content, they have major importance for the description of object discontinuities. The CAD-based 3D feature extraction routine takes advantage of this knowledge. It first locates the edges of the features to be measured and then derives the vertices as intersections of appropriate lines.

The position of the edge is determined with subpixel precision by fitting a second-order polynomial in the direction of the gradient. The maximum point of the fitting curve corresponds to the subpixel position of the edge. The covariance matrix of the estimated polynomial parameters represents the accuracy of the edge point.

The 3D feature-extraction is described in more detail in [4][5].

1.4 Automation

We described the 3D feature extraction as a semi-automated top-down procedure. A CAD-generated feature is matched with corresponding images. In fact, any computer-vision strategy has ultimately a very strong top-down component. Theorists have pointed out that this is true for human perception as well [7][8]. Seeing is largely recognizing, is so to speak a top-down much more than a bottom-up process.

Human perception happens simultaneously at many different levels. If we want to, we can perceive the world around us as being composed of lines or of colours. But it is most natural for us to see it as being composed of objects. It´s at the level of objects that we can understand the world. To lift the degree of automation to a higher level, we argue that it is necessary that also in architectural photogrammetry, the evaluation should be based on the notion of objects.

This is true not only for the providing of qualitative guidance in the computer measurement process, which we will discuss later on. The notion of objects is also the prerequisit for a possible interpretation of the scene by a computer-program. It should be mentioned here, however, that an automatic interpretation of an architectural object faces many difficulties, not the least of them being that there's just simply no one correct way to model any architectural object. This is obvious not only from well-known texts about architectural theory [9][10][11]. It also quite simply has to do with the fact that no two CAAD-operators will model the same building the same way.

This doesn't mean that automation has to stop here. Increasing automation of the whole modelling and measuring process could instead be achieved by a computer-learning mechanism, that allows the user to teach his modelling preferences to the system. An object-oriented data-integration is, as we will show, the essential prerequisit for these functionalities.

2. DATASTRUCTURES IN DIPS AND IN CAAD

2.1 General Considerations

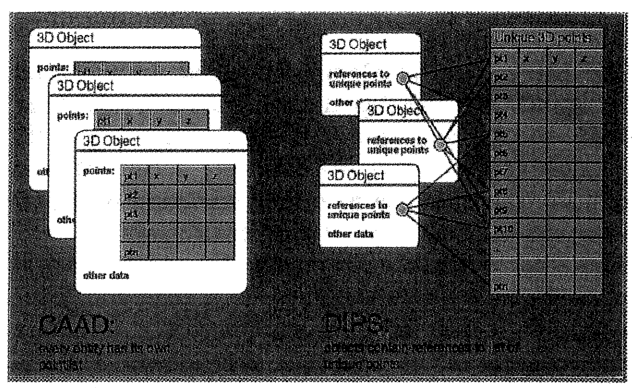

Figure 1: Schematic comparison of Datastructure in DIPS and CAAD: uniqueness of points is not adhered to in CAAD

Comparing the datastructures that are commonly used in digital photogrammetry and the ones found in CAAD systems, one essential difference can be stated as universal. It is the use of unique points in photogrammetry that is not adhered to in CAAD. CAAD datastructures are geared towards modelling capabilities, for which discrete elements have proven to be useful. (see figure 1).

Furthermore it can be said that structuring means that go beyond points, lines and layers are to this day rather rare in photogrammetry systems. Means

of higher level groupings, which are indispensable in CAAD, are missing in point- and lineoriented photogrammetry systems.
Another difference is that in CAAD, there is a need for the exchange of data between different parties involved in a project. In photogrammetry this is usually not a necessity. As a consequence, there is no standard format for digital photogrammetry, whereas for CAAD, there is a de facto standard that is used in much of the architecture and engineering-world: DXF.
While there are many research projects going on to improve or refine the status-quo of this data-exchange and the limitations of DXF have been pointed out many times, it is impossible to ignore it in discussions about data-structures in CAAD. The development of a data-integration between DIPS and CAAD must acknowledge this fact. The DXF-structure actually can serve as a model or guideline, how a data-structure that is compatibel with CAAD and provides semantic and object-oriented information, must be set up.

2.2 Data-Structure in the Digital Photogrammetry System

The Datastructure in the DIPS developed for this project can schematically be described as follows: There is one 3D representation, consisting of objects that reference points in a list of unique 3D points (unique meaning that there are no two points with identical coordinates). Additionally there are 2D representations of the same objects allocated for every image used for the evaluation. The objects and the 2D points in every image are stored separately but they always contain a reference to which 3D point or object they correspond. This split in 2D and 3D representations is essential for the feature-extraction procedure described above.
For the data-integration with CAAD this basic structure must be treated as a given.

2.3 Data-Structures in CAAD: DXF, the De-Facto Standard

DXF is one of the most wide-spread data exchange formats in the computer world. Originally dating back to 1982, Autodesk, makers of the CAD program AutoCAD, designed it to provide an exchange-format between different AutoCAD packages on different operating systems. The format has been extended and changed practically with every new version of AutoCAD. Because of the wide spread of this software, the support of a DXF-interface became almost a must for any other CAD package in the market. So, almost by accident, DXF became the de-facto standard for drawing data it represents today.
We will not discuss the value or the problematic aspects of this standard here and rather treat it as a given that serves well to demonstrate certain aspects of CAAD datastructures that are in some way or another part of any CAD program.
As stated above, the main difference to photogrammetric systems is, that in DXF, uniqueness of points is not an issue. Rather it is a normal situation that many elements have points in common but each stores them individually (figure 2). In DXF there are different basic geometric entities that have each their own syntax. Examples are point, line, polyline, trace, circle, arc,etc. There are so-called 2.5D and 3D objects among them, but all can be placed in any orientation in 3D. All entities can be placed on different layers or have additional information (extended entity data) attached. The number of basic entities that can be used in a drawing can be extended at will by creating new complex entities, so-called blocks, out of the existing basic ones.

2.4 The Concept of Blocks in CAAD

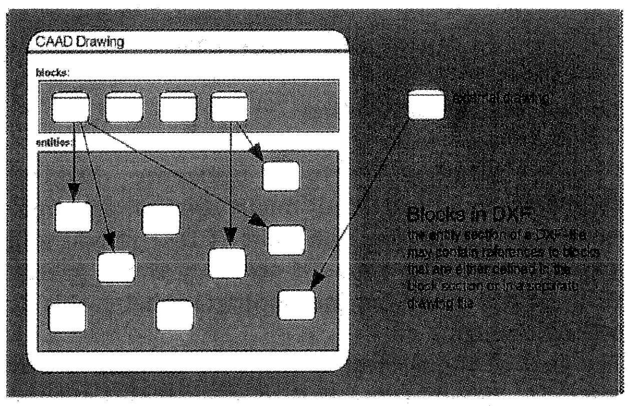

Figure 2: Blocks in DXF: the entity section of a DXF-file may contain (multiple) references to blocks, which are defined in the block section or in a separate file.

Blocks are maybe the most important feature of the DXF format. Blocks are so to speak indipendent mini-drawings that are defined in the block-section of a DXF-file or in a separate file. In the entity-section, those blocks can be referenced like normal entities. This is useful for many things.
It enables hierarchical structures, as blocks themselves can be part of blocks again.
It enables the formation of libraries of complex objects that can be used many times.
It enables a compact file-storage, as reoccurring elements must be defined in one place only.
It enables object-oriented concepts such as different levels of detail for one object, different modes of representation of an object, depending on the context it is in.
The way blocks can be referenced is very powerful, the way they can be manipulated, however, isn't. A block can be moved, rotated and scaled in resp.

around x, y and z-axis uniformly only. There is no way to define the behaviour of its elements in these operations. As soon as a block is taken apart (exploded) it looses this limitation, but also all the many positive powerful aspects.

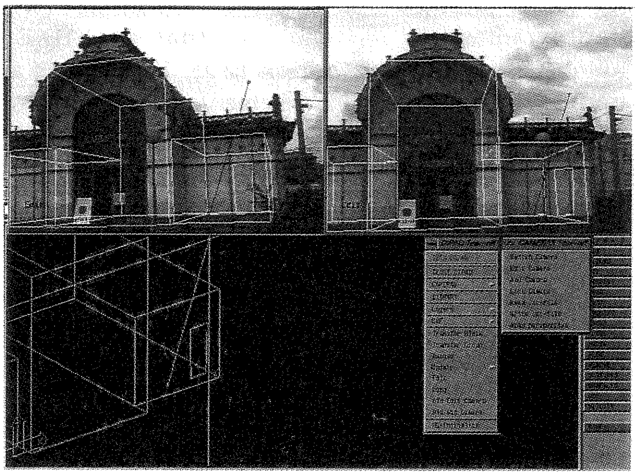

Figure 3: Using the transfer protocol the modelling in the CAAD program can be monitored in the DIPS. So the modelling of approximations is very convenient.

3. DATA-INTEGRATION

3.1 Goals of the Integration

The data-integration between Digital Photogrammetry and CAAD might seem an academic exercise. Why not use some sort of DXF-import and -export, one might argue. If one only wants to deal with points and lines in photogrammetry this is indeed enough.
The goal of the project is however to pave a way towards automation and qualitatively guided measurement. This cannot be done without an object-structure that is capable of including object-oriented constraints. Modelling in our system is understood to be equivalent with interpreting. Interpreting architecture at the level of points and lines is simply inadequate for any purpose. The data-integration indeed follows two essential goals: to allow a smooth and complete transfer of data back and forth between the two systems and to provide a means of including object-oriented constraints in the models that can qualitatively guide the automatic computer-measurement.

3.2 Transfer of Data

The data-integration happens with the aid of a specially developed protocol which allows the photogrammetry and the CAAD system to exchange data at runtime (figure 3). This creates a rather complex environment, where two programs run concurrently and might seem like overkill. Why not introduce some modelling capabilities in the DIPS and make the transfer of data, when the model is completely modelled and measured?
Two premises may need to be repeated here. One is that for modelling, CAAD datastructures offer superior flexibility and functionality. 3D-modelling programs have not for no reason become an industry standard the way they exist today. To include a full fetched 3D modeller in the DIPS would mean to reinvent the wheel in some ways and probably wouldn't make the data-integration much easier.
Secondly, even if a modeller were included, an interface to some CAAD system would be needed still, for the production of plan-drawings etc. The alternative is to create a "do-it-all"-program that is unnecessarily complex and most certainly won´t be able to do some things after all. The great benefit of a complete data-transfer at any stage of the modelling and drawing process is that modelling and measuring become fully integrated. Additional information can be added to the existing drawing at any time. The more features are measured, the more reliable the results become. Therefore objects that are modelled after others have been already measured should not be evaluated without reference to the previous measurements, which they again might improve, too. So the protocol is actually designed to prevent a splitting of the reconstruction-process into a modelling, a measuring and an evaluating or refining stage. Instead, with the help of the protocol, all of these actions can become smoothly integrated procedures that can take place repetitively along one continuous workflow (figure 4).

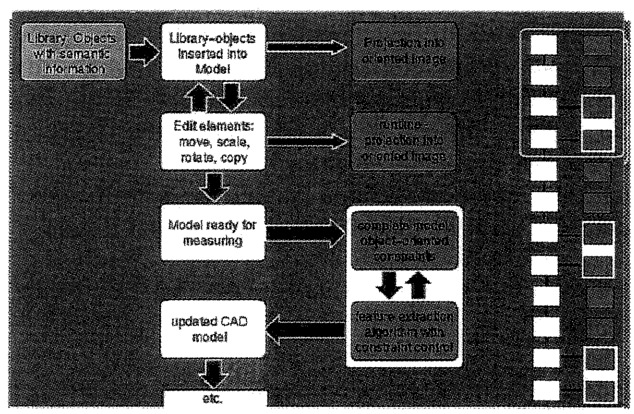

Figure 4: DIPS and CAAD exchange data at runtime. So modelling and measuring become integrated actions along one continuous work-flow.

3.3 Qualitative Guidance

The second goal of the data-integration has not so much to do with convenience or with the philosophies of modular software-design. It has to do with the automatic measurement.
The bundle-adjustment allows reliability-control of results, but only with regard to the observations it processes. Often results are quantitatively within

the calculated tolerance and therefore right, but still wrong qualitatively, as no qualitative constraints were part of the solved equations. Particularly for architectural purposes this is a serious concern. When points that "obviously" lie on one plane or elements that "obviously" touch are just slightly "off" after the measurement, the models generated thus become very cumbersome to use for any further architectural processing. Usually they are hand-edited, which is very time-consuming and also decreases the precision of the model. The qualitative constraints needed to prevent this could impossibly be stored at the level of edges or even of single points only. Higher level groupings, complex objects, possibilities to link elements and to store additional information with them, is the prerequisit for qualitatively controlled automatic measurement.
Being able to indicate equal parts (eg. windows with the same dimensions) in the model is another example for qualitative guidance. This can not only help to keep the resulting model free from unnecessary irregularities. In cases where only parts of those elements are visible, the observations made on many different partly hidden instances of them can add up to a reliable result for all of them.

4. KNOWLEDGE BASE

4.1 Finding an Equivalent to Blocks in DIPS

We have shown the value of blocks in CAAD and stated the need for an equivalent structuring means in the DIPS, in order to achieve a complete data-transfer, qualitative guidance in the measurement and also to pave the way for AI methods that build up on the notion of objects. We have also shown that blocks, the way they are defined in CAAD are inadequate for most automatic measurement. They only allow uniform scaling, which, except for extremely simple objects, can usually not be enough to accomodate the flexibility that an object needs to come to conform with the computer measurement.
So what is needed is a structure equivalent to the blocks-section in DXF, but with the possibility to define them in terms of parameters and constraints. For this we have introduced a knowledge base. Just like the block-section in DXF, it defines the geometry of an object in terms of points, edges and faces. At the innermost level lies a list of unique points, expressed as parametric equations. Theoretically the number of parameters could be anywhere in the range of 0 (as rigid as a block) to 3*the number of points (which - if the equations are written accordingly - means that no constraints for any of the coordinates of the points exist), but a number of 5 or 6 parameters in addition to the standard transformation matrix (which contains three parameters each for scale, rotation and displacement) has proven to be a reasonable maximum beyond which it is preferable to start a sub-entity with an individual set of parametric equations. Of course the parameters of the transformation-matrix are usually constrained in some way as well, especially in the case of nested entities thatare located at a specific location within another block.
This knowledge-base is not a standard CAAD-feature. It had to be invented to provide object-structures with the flexibility to adjust to measurements. We have referred to this as "modelling weak forms" elsewhere [3]. They were implemented in CAAD, making use of extended entity data, which is the DXF-standard for non-standard data, so to speak.

4.2 DIPAD Transfer Protocol

The knowledge-base acts like the block-section of a dxf-file in that it defines entities that are referenced in the entities-section, the actually visible drawing area. For the transfer this means that for complex entities, only the parameters, not the whole definition must be transferred. This reduces the overhead of the transfer considerably and thus improves the performance. The protocol, which is based on the DTM-protocol defined for interprocess-communication by NCSA, does not provide for the transfer of element-definitions. It is limited to the transfer of actual parameters of objects for which the definition is known. It consists of a header-section and a data-section. The header-section contains a label and names (type, blockname, layername, entity-handle) of the following object. The following data-section contains, marked by a tag, the different parameters of the object.
The knowledge-base is the common denominator of the two systems, the transfer protocol the way in which the systems can easily exchange information based on this common ground.
While so far only data of geometric entities needed for measuring have been mentioned, the same mechanism can be used to transfer data about camera-orientations between the CAAD and the photogrammetry system. Of course the cameraorientation is unnecessary information for the CAAD model itself. But to see the camerapositions as parts of the model and to be able to manipulate them in model-space to get to a first approximation is very convenient indeed.
As the structure of the protocol is so simple and so similar to DXF, it would be well possible to use the same protocol to exchange data with our DIPS from a different CAAD system. Although the development of the protocol is not at a point where we want to rpoclaim it as some standard, we feel that it was a good decision to keep the system open in this way and not tie it to one particular software.

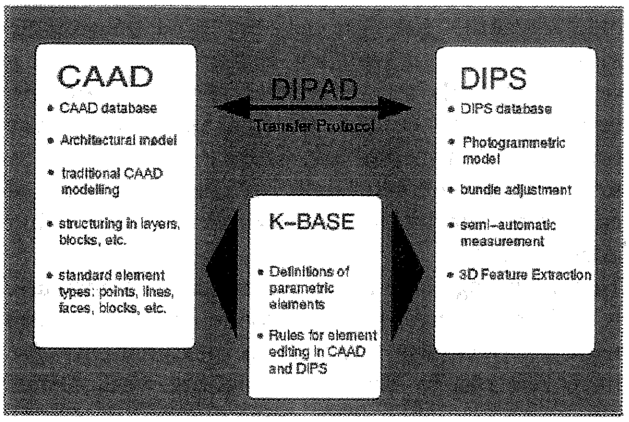

Figure 5: The knowledge-base is a shared resource of both systems. With the DIPAD protocol.only of variable data is transfered.

4.3 Knowledge-Base as a Shared Resource

In order to provide the common ground for the transfer protocol, the knowledge-base must be a shared resource of both systems. It is an ascii-file that is read in once at the beginning of a session and can be reread when it has been changed or extended during a session. The idea is, of course, that elements that have been defined that way once are not lost, but can be reused in later projects again. This allows for the building up of a library of parametrically defined parts that can be defined to best suit the purpose of the user. Whether they are a collection of different roof-types or of different column-shafts or of different window-sills doesn´t make a difference. The blocks can be used and combined in any desirable scale.

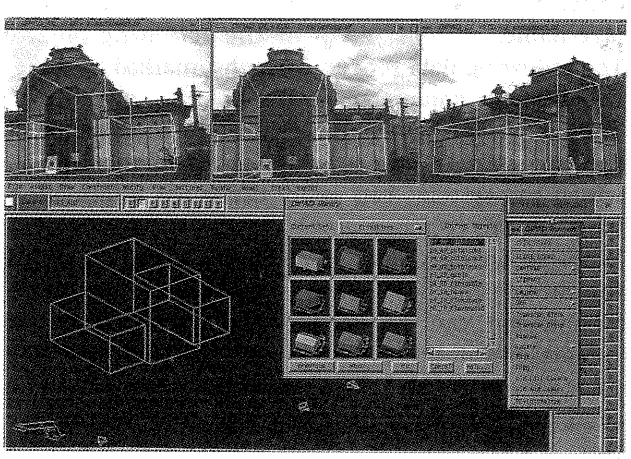

Figure 6: The scene can be modelled using objects with parametric constraints defined in the knowledge-base. These objects are available in a library that can be extended and customized by the user. The camera-icons also visible in the CAAD-window correspond to the exterior orientation of the images used for measurement.

5. CONCLUSION

We have argued for the necessity and described the implementation concept of an object-oriented data-integration between a particular Digital Photogrammetry System and a CAAD program.

One benefit of the approach is the possibility to use both programs in parallel rather than in sequence. But the data-integration also provides the means to guide the automatic measurement with object-oriented constraints. For the object-oriented structure, the DXF-format served as a model. But the relation with DXF should not be overemphasized. Rather, the proposed object-structure translates well into most object-oriented standard 3D formats. Among others also into the VRML format [12], the 3D-standard for the World Wide Web, that has become very popular lately. In contrast to DXF, VRML also offers a means to include image-data as texturemaps in a standard fashion. This can easily be supplied by our set-up and could become very interesting for digital architectural photogrammetry in general, soon.

References

[1] Fellbaum, M., 1992. Low cost systems in architectural photogrammetry, Int. Arch. Photogramm. Remote Sensing, Washington DC, Vol 29, Part B5, pp. 771 - 777.

[2] Schenk, T., 1995. Digital photogrammetric stations, in: Proceedings of 2nd Course in Digital Photogrammetry University of Bonn.

[3] Hirschberg, U., Streilein, A., 1995. CAAD Meets Digital Photogrammetry: Modelling "Weak Forms" for Computer Measurement. Presented paper at the ACADIA'95 Conference, Seattle.

[4] Streilein, A., Hirschberg, U., 1995. Integration of Digital Photogrammetry and CAAD: constraint-based modelling and semi-automatic measurement. Presented paper at the CAAD Futures'95 International Conference, Singapore.

[5] Streilein, A., 1994. Towards Automation in Architectural Photogrammetry: CAD-based 3D Feature Extraction, ISPRS Journal of Photogrammetry and Remote Sensing. 49 (5), pp 4-15.

[6] Madrazo, L., Design Education With Computers, Proceedings of the CAAD futures '91 conference, Zürich, 1991.

[7] Uttal, W. R., On Seeing Forms, Hillsdale, NJ:LEA Publishers, 1988.

[8] Gombrich, E. H., Bild und Auge. Neue Studien zur Psychologie der bildlichen Darstellung. Klett-Cotta, Stuttgart, 1984.

[9] Venturi, R., Complexity and Contradiction in Architecture, Museum of Modern Art, 1966.

[10] Slutzky, R., Rowe, C., Hösli, B., Hg.: Transparenz, Birkhaeuser Verlag 1974.

[11] Mitchell, W. J., The Logic of Architecture. Design, Computation, and Cognition, Cambridge, MA: MIT Press, 1990

[12] Bell, G., Parisi, A., Pesce, M., 1995, The Virtual Reality Modelling Language, Version 1.0 Specification. available at http://webspace.sgi.com/Archive/Spec1.0/index.html