8

Modelling Structure and Shape in the Construction Industry

Peter Willems
TNO Building and Construction Research,
Delft University of Technology, dept. of Civil Engineering.
P.O. box 49, 2600 AA Delft, The Netherlands
Phone: +31 15 2608485, Fax: +31 15 2122182,
E-mail: pw@bouw.tno.nl

Abstract

Which steps can be taken to enhance the efficiency of the information infra-structure in the building industry? This paper will point out the importance for a shape independent structure for product data. Besides, a more flexible 'plug-in' geometry for certain civil engineering objects will be discussed and demonstrated in a case for highway design data.

Keywords

Product data structure, shape representation, highway design.

1 INTRODUCTION

The last decades many branches of industrial product development show a rapid transition from a paper-based information infra-structure to a computer based information infra-structure. The information infra-structure in an industrial production process is basically a chain (lattice) of activities. At each activity node information is added, transformed or deleted. The activity nodes are linked through information channels guiding the information flows.

In the old days this system was processed manually and carried by (mostly) paper media, such as various reports, figures, drawings, sketches, memo's, and so on. With the advent of computer systems successively many activity nodes were automated or supported by so-called CAx-applications. However, for a considerable period of time the information flows between these 'islands of automation' are essentially not affected. The information output of an activity node is sent to a printer or plotter to change from a computer-based medium into paper-based medium. At the next activity node this transition is reversed using scanners, digitizers but mostly keyboard input. The paper-based information flows are principally meant for human interpretation and therefore humans are crucial to perform an information exchange between two activity nodes. Gradually the information infra-structure started to show close resemblance with a highway network with insufficient capacity. Besides its inefficiency such a system exhibits also other inconveniences:

M. J. Pratt et al. (eds.), *Product Modeling for Computer Integrated Design and Manufacture*

- error-prone
  because of the multiple (re)interpretation and manual copying of data (data conversion is often needed).
- incomplete
  missing data because it was never entered in the system or because it was lost or damaged during the exchange.
- inconsistent
  inconsistency may arise easily because of the substantial amount of redundant data, especially in (annotated) technical drawings.
- laborious
  the minimisation of the occurrences of the above mentioned communication failures makes a manually driven paper-based information system very laborious.

This unsatisfactory situation can only be improved by a complete transition of the information infra-structure from a human-interpretable paper-based level to a machine-interpretable electronic level. Of course, humans will always keep control, but they should not play a role anymore as an intermediary element in the primary chain of product data flow. This is the ultimate goal which maybe will never be achieved completely, but it gives at least the direction which steps can be taken to improve the efficiency of the information infra-structure. These steps ranges from network technology (LAN, WAN, Intra-net) via distributed computing (CORBA, HTML, VRML, Java) to work-flow management and last but not least product data technology (STEP).

While in most branches of industrial production (aircraft, automotive, chemical) this transition is rapidly taking place successfully, the building industry seems to fall behind. For example 3D-CAD and product modelling are hardly used in the design process of building products. Somehow the necessary effort to implement these technologies does not balance the advantages. Why is that ?

- one-of-a-kind character of building products.
  This means that each physically realised product is, in principle, unique and more or less designed from scratch. Design costs press directly on the total costs of the realised product, which is not a stimulus to analyse various alternative designs thoroughly.
- poor support for preliminary design
  Contemporary CAD-applications offer hardly any support for preliminary design, i.e. the computer equivalent for the rough sketch using a fat pencil.
- inadequate shape representations
  The applied shape representations often encountered in general purpose CAD-applications (e.g. solid modelling) are optimised for mechanical engineering (part design). Solid models are in particular rather awkward to represent building products. Many disciplines need connectivity data which is better represented by n-manifold or non-manifold representations.
- multiple views
  various disciplines have different views on one and the same building product, which often leads to discipline dependent shape representations. This multi-representation aspect for shape definition is not addressed in CAD-applications.

Which steps can be taken to enhance the efficiency of the information infra-structure in the building industry? This paper will point out the importance for a shape independent structure for product data. Besides, a more flexible 'plug-in' geometry for certain civil engineering objects will be discussed and demonstrated in a case for highway design data.

2 STRUCTURING OF PRODUCT DATA

The multi-shape representation issue is also a major obstacle to exploit the data structure of the shape representation to organise the product data. This leads automatically to the question: which shape? There is no such thing as *the* shape of a product. Many shapes may exist for longer or shorter periods, especially during the design stage. The intended application determines which shape representation is appropriate, which may vary widely between, e.g.

- n.c.-milling (solid model with features),
- rendering (often a set of polygons plus material surface properties),
- finite element analysis (meshed model, possibly with reduced dimensionality and idealisations),
- energy transmission in buildings (surface model plus adjacency topology),
- mechanical engineering (centre lines, rotational axes, features),
- bill of quantities (lengths, areas, volumes).

No shape representation can claim to be the 'mother of all shape representations', i.e. that it contains all the necessary information to derive all other shape representations. And finally, how do I organise my product data in the very early stages before there is any shape at all?

Eventually, this must lead to a structure to store product data, not as an augmented shape model, but attached to a shape neutral product model kernel. This structure will differ for each particular product, however, at a certain level of abstraction they must show a close similarity on a common high description level. Over the years a variety of so called reference models have been developed for this purpose. A well-known early example is, for instance, the General AEC Reference Model (Gielingh, 1988). Another more recent developed model is the PISA Product and Process model (PISA, 1996). It is one of the results of the PISA project, which was funded under the European ESPRIT information technology project. It may well serve as an example reference model to demonstrate some fundamental required modelling principles.

The PISA Product and Process model (P&P model) uses five classification mechanisms which are referred to as dimensions. The dimensions are considered to be orthogonal, hence, entities can be formed by combining up to five independent entities from each of the five dimensions. The next sections will discuss the various dimensions and the different modelling principles that played a role for selecting this particular set of axiomatic entities.

2.1 Product, process or peripheral

The first dimension discriminates between information that describes a certain state (product) and information that describes the transition from one state to another (process). In the PISA P&P model the concept of product modelling as a solution for integrating applications has been extended and fused with the modelling of processes. Product and process modelling are strongly interrelated: a product is always the result of a process, while a process needs input products to transform, to aid the production or simply to consume.

2.2 Specification

The second dimension is called the specification dimension. Specification information can be supplied at different levels. The level determines the scope or the number of information objects that are affected by that specification level.

- Specifications on the occurrence level typically affect only one information object. An information attribute that has a different value for every object (and the fact that two objects have the same value is regarded as coincidental) should be specified at this level. A classic example are the location and orientation attributes for objects in a geometric model.

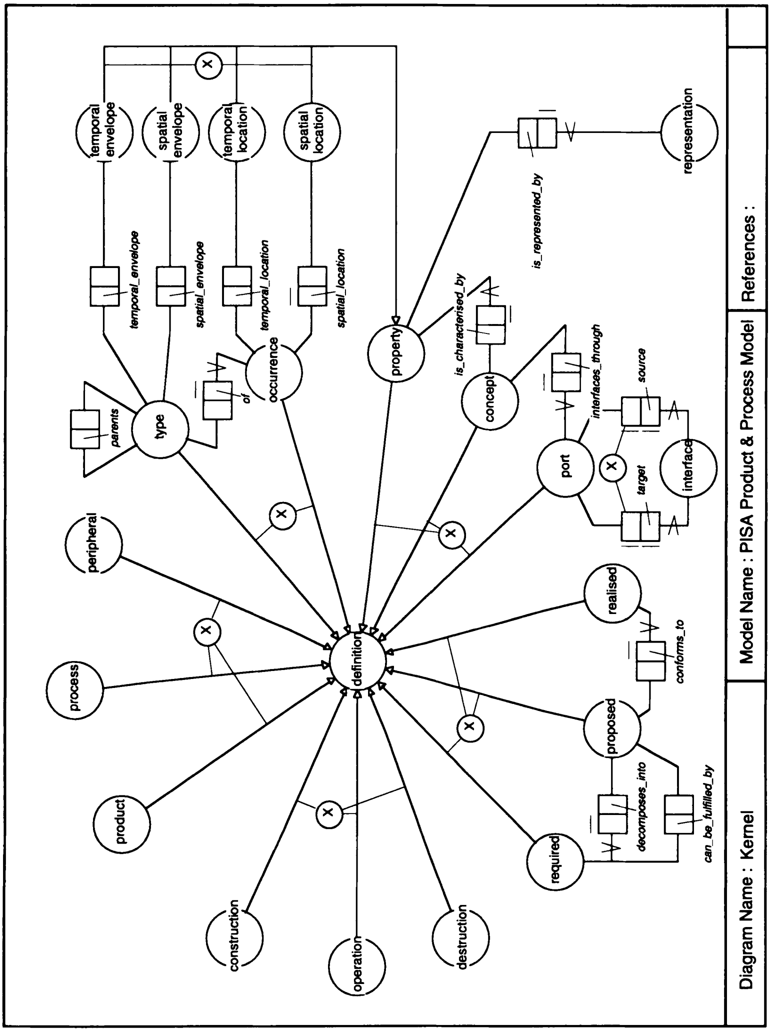

Figure 1 NIAM diagram of the PISA Product and Process model.

- Specifications on the type level affect all the occurrence objects that depend directly or indirectly on a particular type object. An information attribute that has the same value for groups of objects and that fact is considered intentional, should be specified at the type level. A corresponding classic example is the shape specification that all occurrences of the same type must share.

2.3 Decomposition

The prime objective of decomposition is to organise the information objects with respect to the level of detail they represent. The classical approach is to distribute the information objects in a tree structured graph, where the root node represent the total scope of the model and the leaf nodes represent the finest granularity of detail. Each intermediate node, i.e. a node that has both child nodes and a parent node, represent the assembly of all its siblings and is, at the same time, a sibling of a node that represent a more global level of detail. An assembly/part structure is another way to contemplate decomposition. In an assembly/part structure atomic parts are the building stones to constitute sub-assemblies, which in their turn are parts in a more complex sub-assembly until a final top-assembly is reached. This metaphor also demonstrates that sub-assemblies never overlap, i.e. each part has only one parent and can be uniquely addressed by specifying the path of sub-assemblies to arrive at that part.
The PISA P&P model has an explicit dimension to realise modular decomposition structures. Two entities fulfil this desired modularity: *required* and *proposed.* The proposed object plays the assembly role while the required object plays the part role, i.e. a proposed object may be decomposed into a set of required objects and a required object can be fulfilled by a proposed object. This last relationship can be utilised as the interface between (dis-)connected modules or layers.

2.4 Connectivity

Decomposition organises the information objects over several layers or levels of detail, hence a vertical structuring mechanism. Connectivity determines how the information objects of the same layer are connected, hence a horizontal structuring mechanism. Connectivity is the main structuring mechanism of product and process modelling. If the decomposition and specification dimensions in a model are ignored (by collapsing all assembly-like objects until only atomic component parts persist and by expanding all occurrence objects with a copy of the entire inherited value set) the relations that remain will be the connectivity relations. The connectivity dimension constitutes a connected network graph of information objects. A connectivity network is a unique structure, i.e. there is no such thing as *the* decomposition structure of a product and process model but there is only one connectivity structure.

2.5 Life cycle

The life cycle dimension offers in essence a classification mechanism (there are no branching relationships defined) between the three main life cycle phases of a product or a process:

- *operation*
  the life cycle phase during which a product or process is fully operational.
- *construction*
  the life cycle phase during which a product or process is under construction, i.e. the structuring process of integrating sub-products or sub-processes (with private life cycles) into a more complex product or process.
- *destruction*
  the life cycle phase during which a product or process is under destruction, i.e. the destructuring process[1] of disintegrating a complex product or process into the constituting sub-products or sub-processes.

---

[1]) Using the term destruction with respect to a process is probably not obvious, yet a complex process consists of sub-processes that were possibly already operational before the actual start of the main process and which may continue after its end. Compare for instance all the sub-processes that are needed to bootstrap and shutdown an operating system.

2.6 Applications of a P&P reference model

Although the PISA P&P model can be applied for direct instancing the semantic content is too general to be of much significance for this purpose. The justification for the existence of a reference model is to ease the integration and the inter-operability of product and process models that are developed for a particular branch of industry. The idea is to develop a layered hierarchy of models ranging from broad scoped high level models to narrow scoped low level models all conforming to a common P&P model. In such a framework each slot should contain a model of a well defined and limited scope which could act as a parent model for more specific models or as a server model for other client models. The conformance of the applied structuring mechanisms in each model help to achieve a kind of plug-in architecture encouraging the reusability of the well developed models and the replacement of less achieving models.

3 SHAPE MODELLING STRUCTURES

Product description attached to a conceptual modelling structure is one step, but eventually the various shape models must be plugged into this structure, how can this be done?
One approach will be discussed in this paper and can be demonstrated at the hand of a product domain in civil engineering that can be generally labelled as infra-structural objects, i.e. highways, railroads, subways, bridges, tunnels, and so on. On a certain level of abstraction all these objects can be described by 1-dimensional shape structures.
This paper will focus on highway design. In the current situation it appears that most road design projects seems to start from scratch. E.g., a wide spread road design package uses basically 3D points aggregated in chain-like structures called strings representing boundaries between lanes or carriageway edges. Such a low-level representation can not offer much assistance for reusing previous designed components. Still a user of these (realised) infra-structural objects (e.g. driving a car), experiences a lot of repetition, similarities, sometimes even monotony. In other words: because of this restricted variation reusing components should be very common.

3.1 Space Deformation Tree representation

In 1989 the Dutch ministry of transport and public works initiated a project to define an information model to exchange road design data on a more semantic level, hoping to find a level that better fits the definition of reusable components. The development of the Road Model Kernel (Willems, 1990) was the driving force that led to a new type of shape description: the Space Deformation Tree representation (SDT-rep). The basic principle of this representation is as follows.
In most shape modelling data structures, e.g. the ISO-STEP integrated resource model for topology and geometry (ISO, 1994), geometry is represented as a set of geometric carriers that can be referenced from specific nodes in the topology graph, i.e. vertices reference points, edges reference curves and faces reference surfaces. The SDT-representation places these geometric carriers in a structure of its own, i.e. a graph of modelling spaces of various dimensionalities connected through relations of type *embeds/is_embedded_by*. Such a modelling space graph may exist independently from its referencing topology graph(s). The embedding relations can be established by specifying translation and rotation components, but also by stating spatial deformation components. In other words: the embedding space interprets the embedded space as a *parametric space*.

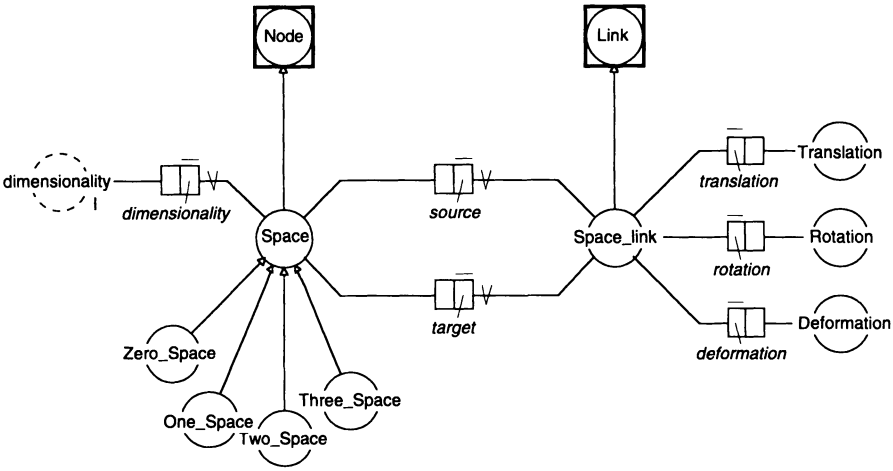

Figure 2 Data model for modelling spaces. The basic structure is inherited from a general graph structure and the *embeds/is_embedded_by* relation is objectified by the Space_link entity.

To demonstrate the relative ease the Space Deformation Tree representation offers to model complex shapes the modelling steps to create a ring of Moebius will be presented.

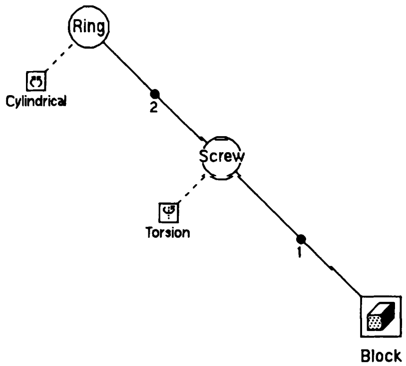

Figure 3 Shape graph of a ring of Moebius in three steps: block, screw, ring.

The shape graph (figure 3) is very simple: one shape primitive and two shape assembly nodes. The three-dimensional shape primitive is a block with length $2 \cdot \pi \cdot R$, with R the required radius of the ring. The root space that hosts this block is embedded in the root space of a shape assembly (Screw) and twisted by a torsion deformation. The block in this space transforms into a screw with a 180° pitch. Finally the root space of the Screw shape assembly is embedded in a root space of another shape assembly (Ring) and bent through a cylindrical deformation. Figure 4 shows the resulting shapes of each modelling step.

Figure 4 Block → screw → ring of Moebius.

3.2 Road modelling case

A more practical example is presented in figure 5. Three shape models (wire model, surface model and solid model) representing the same product, for instance a bridge or viaduct. It is not uncommon that these shape models co-exist concurrently for different applications. By sharing the same geometry structure (here labelled 'horizontal alignment') these models will follow automatically all changes that may be applied to the road geometry.

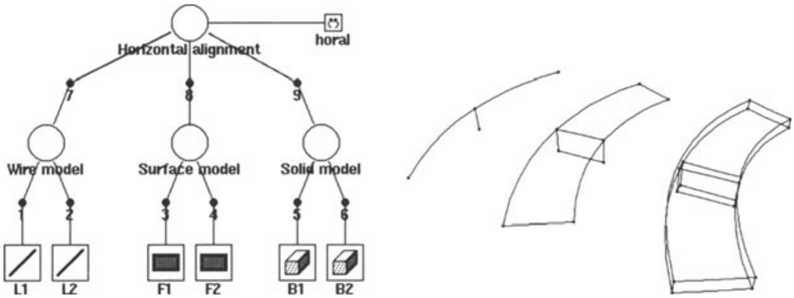

Figure 5 Multiple shape models sharing the same geometry.

This approach supports the definition of a highway according a conceptual decomposition structure and, afterwards, plug-in the actual geometry for its final location. Besides, it offers the possibility to build up a kind of standard part library for frequently encountered road 'features' like access-roads, various types of crossings, intersections, and so on.
The Road Model Kernel project (later: Road Shape Model Kernel) is attempting to raise the product description to a level that supports reusability as much as possible. To realise this a clear distinction is made between those parts of the road that are static or quasi-static (mostly cross-sections) and the dynamic parts that tend to vary for each occurrence (alignments). The static parts can be modelled according the product modelling concepts discussed in chapter 2 (decomposition, connectivity). These structures are fit for parameterisation to be stored as re-

usable objects. The dynamic parts are plugged in as instance properties in the actual data models (see figure 6).

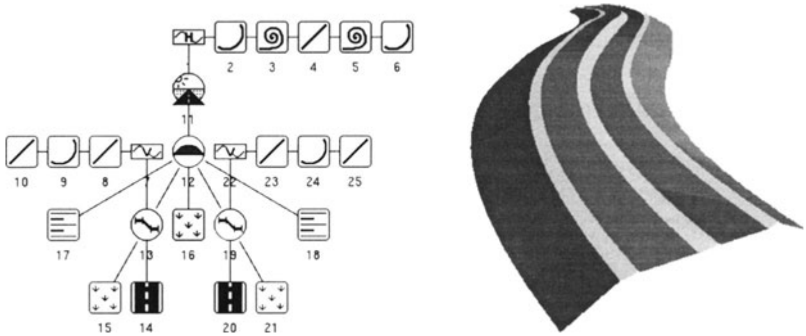

Figure 6 Road model consisting of a static part: road entity (11), longitudinal decomposed into road sections (here only one: 12), transversally decomposed into slopes (17, 18), carriageways (14, 20), verges (15, 21) and central reserve (16); and a dynamic parts: one horizontal alignment (1-6) and two vertical alignments (7-10, 22-25).

To be able to build the various shape models (cross sections, plan views, longitudinal sections, and so on) from this road model a Space Deformation Tree is generated as intermediate representation. Figure 7 shows a part of this generated representation near the leaf nodes: undeformed basic shape primitives (rectangles, blocks) are successively moulded according a width alignment (width variations as function of the longitudinal axis) and a super-elevation alignment (torsion-like variations as unction of the longitudinal axis).

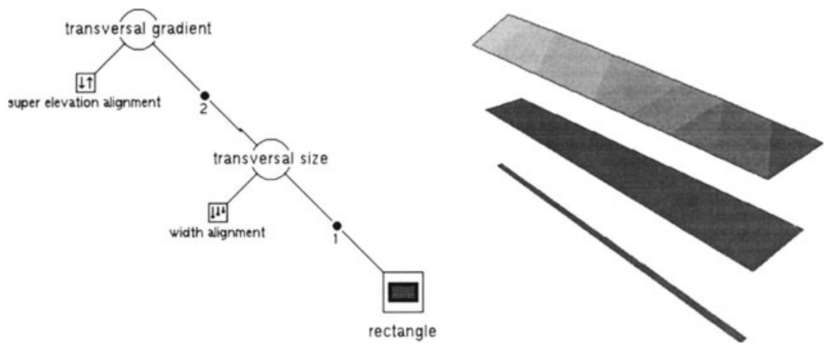

Figure 7 First three nodes in the shape tree starting from a leaf node: rectangle -> application of the width deformation -> application of the super-elevation deformation.

3.3 SDT and the PISA P&P model

The hierarchical structure of the Space Deformation Tree representation follows very nicely the general decomposition structure of a product model. A complex shape can be disassembled in various modules and stored separately. These modules can be referenced from the central backbone of the product model structure. Multiple shape models are able to share common geometrical resources that are defined on a higher level in the product model hierarchy.
The PISA P&P model offers a rich set to classify a shape as required, proposed or realised; as a separate occurrence shape or a shape type that can be shared; relate it with temporal aspects; identify it with a life cycle view; and so on.

4 CONCLUSIONS

- The highly multi-disciplinary character of the building industry demands, probably more than for instance in the mechanical industry, multiple shape models to represent/analyse the same (building) product.
- The recognition of this fact makes it less obvious to use the data structure of a shape model and augment it with product definition data. In particular because the various shape models differ too much (dimensionality, idealisation) to integrate them in one general purpose shape model.
- Product definition data need a structure of its own, satisfying general product modelling principles as defined for instance in the PISA Product and Process Model.
- The Space Deformation Tree representation has several features that makes it particular fit to interface with a product modelling kernel. This is definitely true for a certain product domain in civil engineering, i.e. infra-structural objects like highways, railroads, bridges and tunnels.

ACKNOWLEDGEMENT

This research is sponsored by the Technology Foundation (STW) as part of the Computer Integrated Construction project (DCT99.1891).

REFERENCES

Gielingh, W.F. (1988), General AEC Reference Model (GARM), TNO report BI-88-150, Delft.
The ESPRIT-PISA Project Consortium (1996), The PISA Project: A Survey on STEP, Shaker Verlag, Aachen, ISBN 3-8265-1118-2.
ISO 10303-42 (1994), Industrial automation systems and integration - Product data representation and exchange - Part 42 : Integrated generic resources : Geometric and topological representation, Geneva.
Willems, P.H. (1990), Road Model Kernel, TNO report B-89-831, Delft.