Computer modelling and Case Studies

Mario Lorenz and Stanislav Chavelski

Dept. Of Computer Science At The Technical University Of Chemnitz, Germany

1 Abstract

This paper presents the main concepts of computer modelling and two case studies describing their use in applications for GIS (geographic-informational systems) and VR (virtual reality). It explains the general terms and gives some definitions. The several steps of modelling are investigated from the object-oriented (OO) point of view. The shown approach is illustrated with samples from the fields of GIS and VR.

2 Introduction

The development of software applications in the various realms of computer graphics is based on the specification and design of a computer model which describes the original concerned. The process of model construction (the illustration of the original) is a complex and not trivial succession of some steps of abstraction. The goal of our studies was to find out a common way for the construction of a computer model and to describe it with the help of object-oriented techniques and tools. The results of our studies are shown in the next sections of this article.

3 Definitions

This chapter introduces and describes the fundamental terms.

The term *model* stands in general for an artificial or synthetic (objective or abstract) object. The model allows based on the contained analogies to the original to draw conclutions about the original. A study of the original can be done easy by using the model. Upon closer examination a *model* is the summary of some selected distinguishing features of the real world as well as the arrangement of the relations between the features.

Definition: A *model* is the simplificated replica of a realistic or a virtual (means within the intellectual world) piece of reality that contains all characteristics of the original relevant for a specific problem or a specific object of efforts.

The model is created by making abstractions of the original. That means the model usually not contains all features of the original. A *feature* is a individual characteristic of the original. A defined *state* denotes one of all possible forms of this feature. The change of the forms of the features is the so called *functionality* of the model.

The table below compares the terms of abstraction :

original	abstraction
real world	model
characteristic	feature
form	value
functionality	methods

The goals of modelling are for example the „control of the original“ in any mind and the possibility to make studies over the original in an easy way. That means :

- a fast gain of information about the original
- the modification of the original
- the gain of discovery about the relationship between the original and his environment as well as the relationship between the characteristics of the original
- the creation of a representative of the original

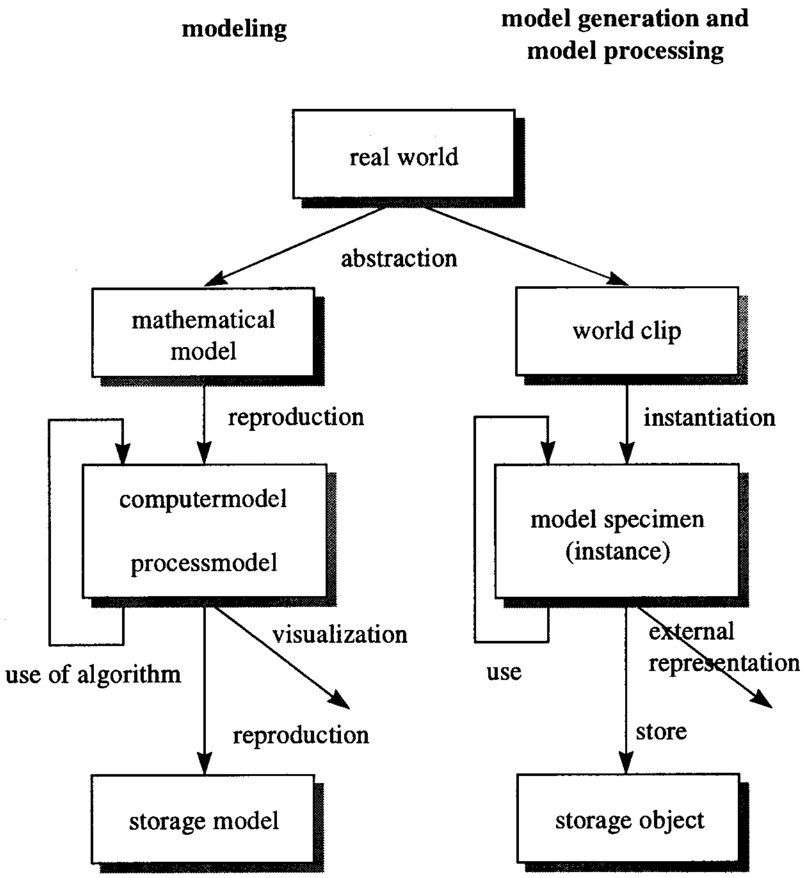


Figure 1: Modelling and model creation

4 Modelling

A generally applicable approach for computer modelling based on the terms defined in section 3 is clarified in this chapter. The process of computer modelling is composed from several complex steps of abstraction and projection. It is shown in Figure 1.

Our studies are based on the assumption that the real world is the original to be modelled. The physical model of any object of the real world is normally a scaled replica of a real object (in the usual meaning), e.g. a model of a terrain, a building or a single room.

The description of a object of this real world with the means of a suitable mathematical structure is known as his *mathematical model.* A mathematical model has a semiotic character, that means it contains no informations about the semantics of the original. A mathematical model can also bee seen as a reflection of a field that is a source for a whole class of objects. The creation of a mathematical model is based on the way of abstractions and the following specification and design of a mathematical structure.

The mathematical model can be seen as a specific form of a scheme of human thoughts. It is an important step in the creation process of a computer model.

A *computer model* of an object is a computer internal representation in the means of stored data which are brought together in a predefined data structure. This representation is also known as a *representation scheme*. Such a scheme is valid for all objects of a class. From this point of view a specific object can be seen as a specific filling of the scheme with defined modelling data. The computer model is in analogy to the mathematical model a semiotic model.

Usually a visualization algorithm produces a visual image of the computer model using an output device for the presentation. The generated image finally shows a replica of the original as a result of a rendering process.

The use of computer modelling and the employment of computer internal methods brings a wide flexibility in controlling real objects. Realizing this approach the main problems are to find out useable algorithms for the various representation schemes and to define correct transformations between the models. For that reason many research activities are oriented towards the goal to find efficient tools for the realization of modelling processes (object-oriented databases, hybrid modelers, new computer internal representation schemes). There are also some special approaches for single parts of the modelling process and for specific fields.

5 Object oriented approach

This chapter presents a different behaviour based on the generally applicable scheme of computer modelling from the object oriented point of view using object oriented terms and techniques. The real world is seen as a set of abstract classes. The first step in modelling is a suitable reduction of complexity by selecting relevant classes and ignore the other ones. The result of this step is the set of relevant classes for the specific means that is called a *world clip.* The world clip can be seen as a set of triple. Each triple has the following notation :

class ::= [name, attributes, methods]

with the following means :

- *name* is the unambiguous ident of a class of the world clip
- *attributes* are the set of features and
- *methods* are the set of functionality that describes the change of features and the interaction between features and classes.

The derivation of a specific model specimen from the world clip is based on the instantiation of the abstract classes. At this time the attributes are obtained with specific and suitable initial values. The processing of the model specimen can be interpreted as the computation of the methods defined with in the world clip and causes the change of the attribute values.

The model specimen passes through some stages in the computation process. The external visualization makes it possible to illustrate the world clip at any time and to draw conclusions about quality and exactness of the model specimen. A model can be called *correct* if it reproduces the behaviour of a real world clip relating to the selected features with a given exactness. A specific or all states of the model specimen can be stored using a storage object.

The ideas above lead us to the following conclusions:

- A abstraction of the real world can be easy done by the way of definition of classes. The classes are specified from the point of view of the specific goal and can be done and documented separately per class.
- The derivation of different model specimen is done by the process of instantiation and gives a wide flexibility. It is possible to take full advantage of the object oriented paradigms.

6 Case Studies

6.1 Modelling a scene of virtual reality (VR)

This chapter presents the use of the approach described above at the example of a simple application of virtual reality. The term *virtual reality* means the audio-visual and tactile arrangement of an apparent world as well as the generation of this world under the usage of the basic (usually physical) laws and rules of behaviour and the integration of the user (the visitor). Therefore it is possible to point out three contents which a VR-system must contain:

- the definition of a computer internal model, that means the modelling in space and time as soon as the definition of model specific conditions
- the suitable computation of the internal computer model with the goal to provide the perception with the human senses (the generation of audio-visual and tactile presentation forms of the model)
- the processing of the model own internal simulation models for the specified laws of behaviour under the direct influence of the user.

The acceptance of virtual reality techniques depends essentially on the naturalness provided to the user. This causes on the one hand a hard closeness to the original means a high quality of the model. On the other hand the feedback of the system must be as good as possible. This means a short answer time to interactions and a realistic simulation of behaviour. When using basic engines in time there are limitations of the model indispensable to ensure a sufficient frame rate because this is one of the most important criterion for acceptance of simulation. The model used as a basis of each VR-application includes essentially the following contents:

- geometry and graphical features of the scene
- description of (natural) laws of the original
- description of the behaviour of the scene regarding

  ⇒ progress of time (animation)

  ⇒ reaction to user interaction.

The first step creating a VR-application is the suitable definition of the world clip to be simulated. In accordance to the approach described above means this the definition of a set of classes which reproduce the original (or a vision of it) in a suitable way. This single step is the main stress of modelling. To master his undoubtedly high complexity the object-oriented method is chosen. From the seven object-oriented paradigms the data abstraction , the inheritance and the polymorphism are very useful for reproduction of the world clip.

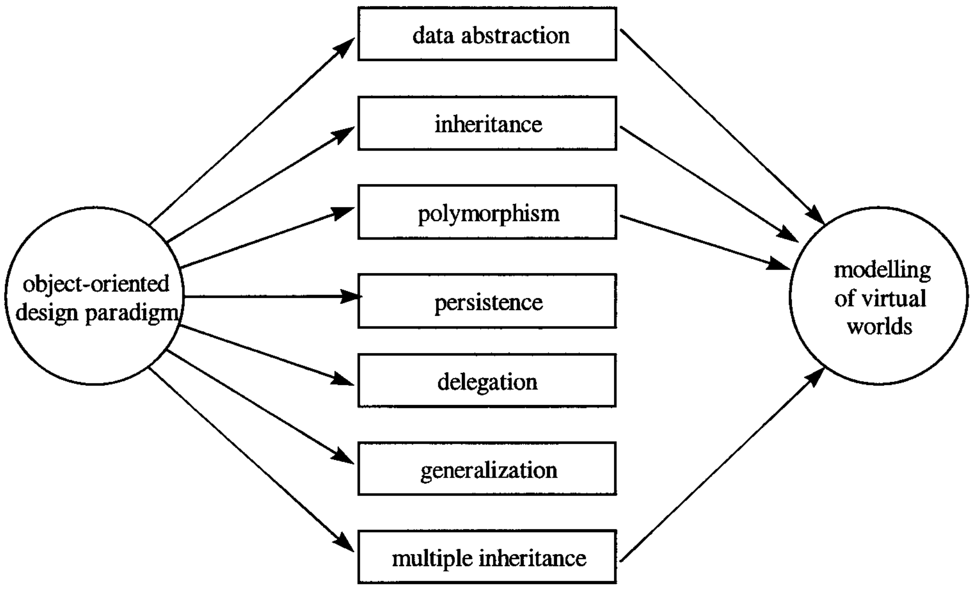


Figure 2: Object-oriented paradigm and modelling of virtual worlds

The class hierarchy employed to describe the world clip is designed in the TOP-DOWN principle. This hierarchy represents the world clip to be modelled with regards to his three-dimensional geometric visual components, their features, their relationships between as well as to the generally applicable laws or rules of behaviour. Whereas the description of the geometric components is performable within a separately part of the hierarchical class tree the integration of the rules and laws can be done by several ways:

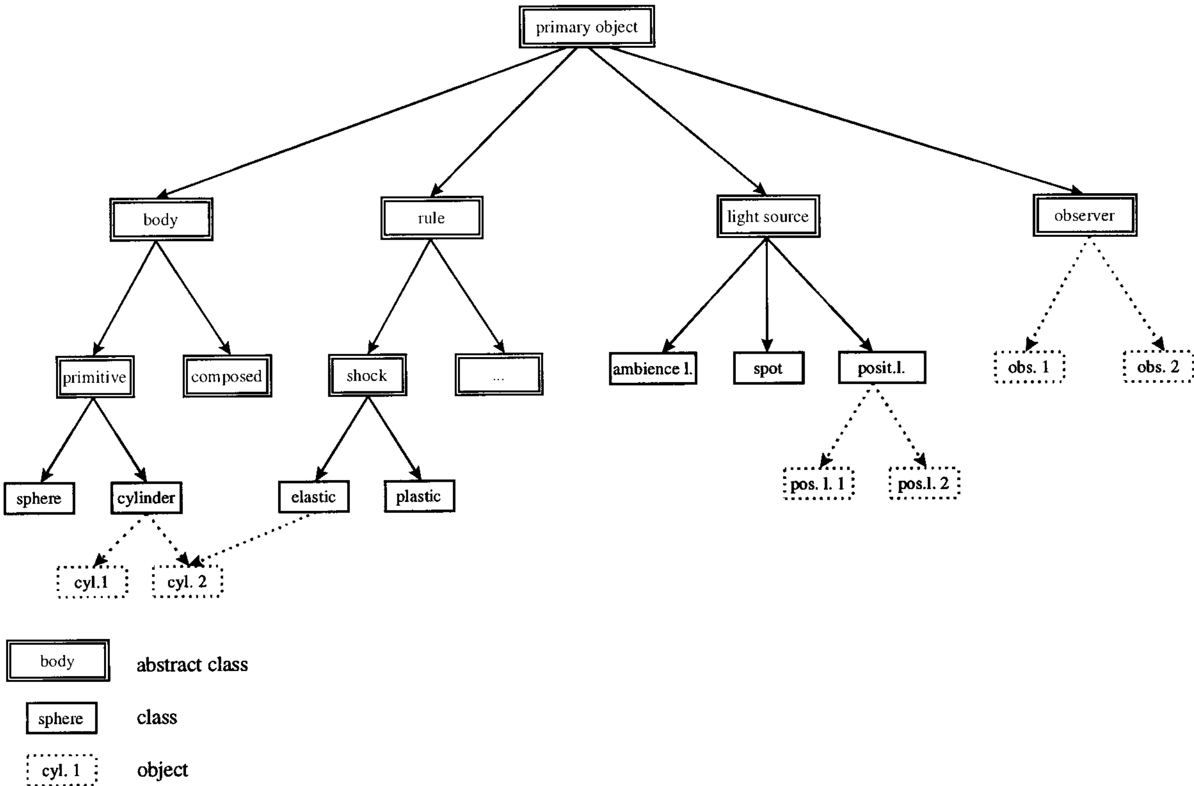
- The model of the object behaviour is a real part of the object model itself, that means the laws and rules are implemented in the form of methods as well as control features within the object classes. It exists only one hierarchy which contains the geometric modelling as well as the functional.

  Example: A class *body* contains a method *shock* implementing the behaviour of the body when it is shocked. A feature *unchangeable* controls whether the body is permanently stationed or not.

- Whereas the model of geometry is analogue to the way above the modelling of the behaviour takes a second separate hierarchy in form of a behaviour tree. This one contains the reproduction of the generally applicable laws or rules, e.g. the formal description of the elastic shock. The class *body* contains only features and methods as a reference. The advantage of such a way is mainly the centralisation of code.

  Example: A class *moveable body* is derived from the class *body*. It obtains the method *reaction to shock* by multiple inheritance from the class *elastic shock.*

It does not make any difference which way of modelling was chosen, the starting point of any modelling is always the definition of the outstanding root class which contains generally applicable laws or rules. Later the design of the kind classes follows. A carefully and far sighted creation of the class hierarchy guaranties a highly quality of the entire model. The following figure show an example of a fictitious world clip as the result of modelling :



The formal description of the several classes occurs in the following notation :

name

base class redef (*redefinition of the method*)
base class redef (*redefinition of the method*)
base class redef (*redefinition of the method*)

attribute: type of attribute

description of attribute
... further attributes

method

description of the method
... further methods

name

To preserve a maximum clarity all class definitions contain a list of features and methods which are obtained from parent classes (base class). In addition there is a list of all features and methods which are redefined by the class (umdef).

In the first draft of the class hierarchy only the absolutely necessary features and methods are considered. In later drafts being the basis of the implementation the detailed working out is made on the relevant points. For better understanding or for reduction of expense as well as for a high system performance suitable abstractions from the original may be necessary. There are two starting points for this:

- a simplified geometric modelling, e.g. the approximation of bended surfaces with facets

- the simplified reproduction of the (natural) rules and laws, e.g. the implementation of ambient light sources as a base brightness.

Both simplifications are permissible until they appear disturbely to the user.

The second step is the instantiation of the model specimen from the results of step one modelling the world clip. The essential difference to the first step is the instantiation of specific objects which represent parts of the original. Whereas the hierarchy (-ies) being created in the first step describe (s) a scene with all possible constellations the result of the second steps is a specific class based on the given class hierarchy (-ies). The mentioned representatives are instantiated from the class hierarchy and the virtual scene is sybolically filled with objects.

Corresponding to the goals and the computing power of the base engine the instantiation can be done in several ways. The possible spectrum reaches from the fix programmed instantiation to the interactive way using special editors, the so called world editors. Whereas the user has to play a secondary role in the process of class definition he can be integrated in the process of instantiation. The user can design the virtual world based on the given class hierarchy listen to his inner voice, means he allocates initial values to the world clip model. Therefore a specific model specimen was reproduced from the universal one that is filled with defined initial values.

This model specimen can be computed in different ways according with the goals of simulation. It is necessary to point out that the processing steps are a part of the model of the world clip and being made useable to the specific model specimen by employment of the object oriented paradigms. Typical steps of computation in the VR-fields are for instance the interactive walk trough a scene, a simulation of a sequence of movements as well as the interactive manipulation of the scene by the user. It is necessary to point out that in the process of model computation the values of some object features may change based on the specified laws and rules but the basic structure of the scene remain unchanged.

There is no need to create new objects in the computation process of common VR-applications but there is in principle no problem to do this. So it is necessary to create a new object of the same class *body* when a body is divided by the influence of tensile forces. This is implementable by instantiation of at least one new object of the class *body*.

Derived from the model computation are some storage processes, e.g. the continuous storage of selected object data and the archiving of a defined state of the model specimen.

The external visualization of the model specimen is the very own request of a VR-application : the generation of a three-dimensional representation of the scene as well as the generation of audio-visual and tactile informations for presentation to the user. The algorithms to do this are also part of the model.

6.2 Example 2: Computermodelling and generalization algorithms for (GIS)

The second example presents the application of the model theory for carrying out of computer aided cartographic generalization for GIS. The goal of the research activity was the study and the application of generalization algorithms for the automatically derivation of geographical maps from one start scale in each other.

Starting point of our research was the development of workable draft for the realisation of one source model (geographical model). The following properties are requisite for the successful design of this geographical model:

- redundant-free storage of even the tiniest spatial data structure for point-object, line-object and surface-object;
- Store of all geometry dates from the earth surface of the urban area using the relations between the data sets and their absolute location. The scale does not figure in this process. The standard principle of image of the data must be met.
- Take heed of the relations and the reference to definite DBMS.

The complex of generalization rules will be used for all object class from the source model in dependence from the destination map scale or landscape area. The sequence of the use of this rules and the priority of the image objects are important. The result of the use of the generalization rules is the production of derived digital model.

The combination of the visualization feature and the signature with the model objects create one new model. This new model can be used for the generation of cartographic maps in different map scale or as data source for GIS.

The problem of the automatic cartographic generalization is a solution only for certain objects or object classes. Our idea is the generation of computer models as image from landscape and independent from the map scale. The realisation of the generalization algorithm creates one model for the destination map scale. This model is the source for the production of the map or for the visualization of a piece of the landscape on the screen with GIS.

Our research activity focuses on the generation of several map models:

- Development and design from workable draft for scale independent source computer model as image of one clip from the real world;
- Development of requirements catalogue for the generation of the model step after the use of the particular generalization rules for all classes;
- Study of the several generalization algorithms, their teamwork and their consequence for the use as generalization rules for all object classes;
- Development of software toolkits for the realisation of the automatic cartographic generalization in geographical information systems (GIS) and used the idea of the geographical computer modelling;
- Prototype development for the verification of the theoretical results and embedding in commercial GIS.

The landscape with its topographic objects and thematic topic builds the original for the derivation in a computer model. The topographic record from the real world is an abstraction of the original. A primary model will be derived from the original by using of generalization rules and restructuring of the space. This primary model is a structured image of the original. The first step of the generation of this primary model of the landscape is the structuring from the 3D-appearance of the earth-surface in topographic objects and object parts as well as their relations. The object structure describes the form, the position and topographic relation between the objects. The object itself is stored with its attributes about geometry, topology and visualization values for the specific map scale.

The next step in the model generation is the derivation of the secondary landscape model from the primary model. At this time the topographic object attributes obtain signature values for the computer maps representation. The new generated model is known as secondary model or cartographic landscape model. According to the signatures and the genaralization rules is the derivation of several secondary models in various map-scales possible. The generation of the tertiary model occurs after the combination with thematic user data. The relations between the generated landscape models are shown in the Figure 3:

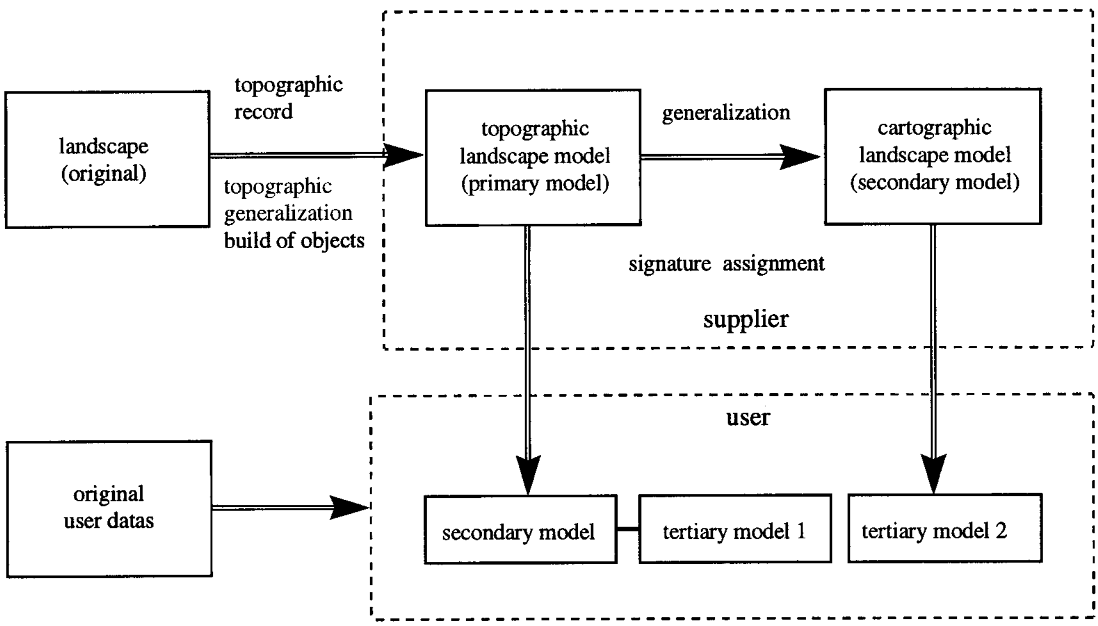


Figure 3 : Relation between model and landscape

From the computer modelling point of view is the primary model generally one space-related computer object model. The computer landscape model is abstracted from the original (real world) and it consists of the computer situation model and of the computer terrain model. The primary model is stored graphic independently. The visualization of the object is result of the assignment with the signature and with the sign key.

The advantage of the cartographic computer modelling is designed in Germany for the first time with the official cartographic information system (ATKIS). The original is the landscape. It can be replaced by the German ground map (DGK 5, scale 1:500) or by air-measurement images. The primary model is the digital or computer landscape model (DLM). The next step is the generation of the digital or computer cartographic model (DKM) from the primary model as a source.

The computer landscape model contains topographical data about the earth surface. The landscape clip is object structured by several semantic criteria. The attributes describe information about the objects and about their geometry, topology, location in the landscape, visualization features etc. Objects from the same type form the object group (class). More object groups mould object area. The object semantic is described by the object types and their attributes.

The digital cartographic model (DKM) contains objects from DLM drawn by 2D-system of coordinates. The object semantic is described by the map object types. By the mean of the signature catalogue is the conversion of the landscape objects in map or cartographic objects possible.

The figure below describes the system design and the data flow:

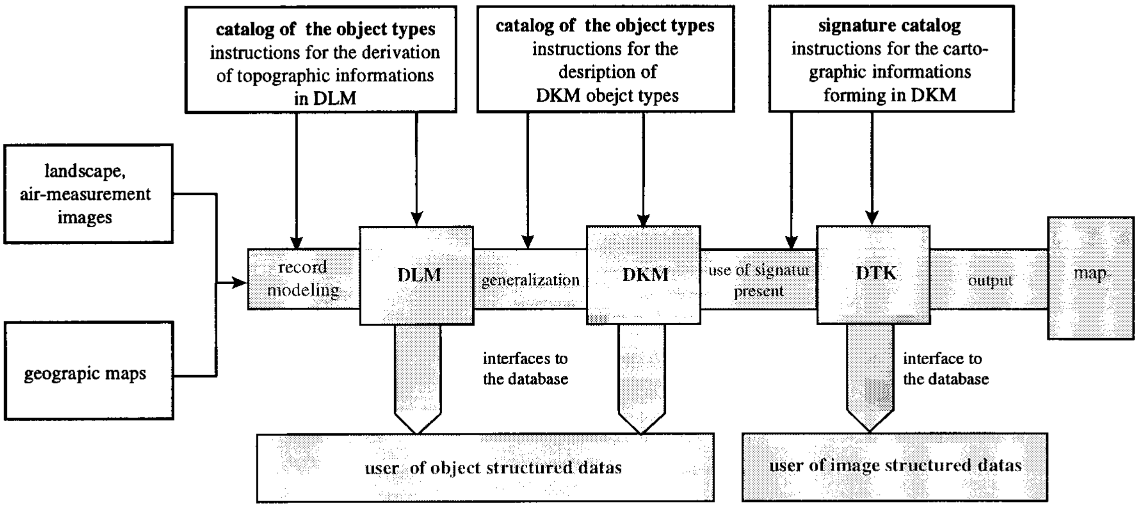


Figure 4 ATKIS-system design

The computer cartographic modelling contains the design and the generation of the computer landscape model (DLM), computer map model (DKM), computer topography maps and analogue topography maps. The following processes are necessary for the derivation of models from the DLM-model: transformation in DKM-objects, cartographic generalization of this objects, signature assignment and visualization of the generalizated objects.

The contents of the ATKIS signature catalogue is structured in two catalogue parts. The DKM-object type catalogue describes the DKM-object types. The DTK signature catalogue describes the graphic design and visualization features of this objects. The DKM-models are generalizated attributed computer models as abstract of the real world. The advantage is in the use of computer generalization rules during the transformation of the DKM-model from the DLM-model. The DKM-model is source for the production of topographic land maps or screen output.

Our study for the realisation of the automatic cartographic generalization generates one draft for the derivation of cartographic computer models. The first goal is the specification and the design of the primary source model. This model is map-scale independent for all objects abstracted from the specific landscape clip It is the base for the derivation of other secondary models. The map-scale for the generation of this secondary model can be chosen by the user. The transition from source model to the secondary model can take place by means of the generalization toolkit. The generalization rules describe the generalization algorithms for the generation of new cartographic models in a new map-scale. The results of the use of the generalization toolkit is the creation of generalizated secondary model. The signature assignment and the instantiation of the specific vizualisation values generate the new model, which is known as a cartographic computer model. It is a source for the map production or source for the use in geographic-information systems. The figure below shown a summary of the sequences and the model derivation:

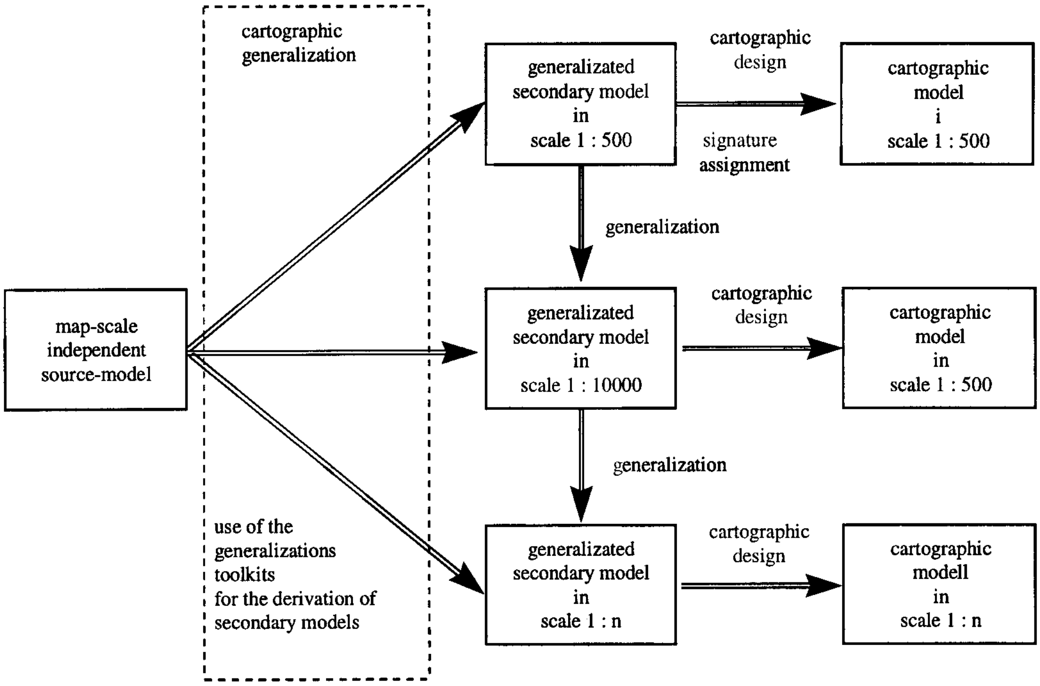


Figure 5 Summary of the sequences and model derivation

7 Summary

The paper describes our research results in various computer graphic fields regarding computer modelling. The efficient specification and creation of the specific computer model is the solution for efficient use and visualization. The use of object-oriented techniques is one way to solve the problems of abstraction of the real world.

References

H. Schreiter, Modellierung technischer Objekte und Anwendungen, Chemnitzer Informatik Berichte, TU Chemnitz-Zwickau, Fak. für Informatik, 1995

W. Kowalk, System-Modell-Programm, Spektrum Verlag, 1996

G. Vickus, Digitale topographische und kartographische Modelle, Schriftenreihe des Instituts für kartographie und Topographie der Universität Bonn, Heft 20, 1994

Arbeitsgemeinschaft der Vermessungsverwaltungen der Länder der BRD (AdV): Amtliches Topographisch-Kartographisches Informationssystem (ATKIS) /Landesvermessungsamt NRW, Bonn, 1988

D. Baumgärtel, Spezifikation eines objekt-orientierten VR-Systems, Diplomarbeit, TU Chemnitz-Zwickau, Fak. für Informatik, 1996

J. Strauß, Interaktions- und Animationstechniken in virtuellen Welten, Diplomarbeit, TU Chemnitz-Zwickau, Fak. für Informatik, 1996