23 Syntheses in object oriented analysis

Lene Rold

23.1 INTRODUCTION

At the National Museum, Copenhagen, we have since 1987 developed and used a database system for registration of the museum's collections.

The first phase of the database registration, the transferral of data from manual registers, will be finished in 1993, and the base will by then contain approximately 1 million records. The next phase, now under way, is the development of a system for registering new accessions. This involves a new structure for the base and transferring data from the old base to the new system.

The DBMS we use is Oracle, and as a result our system development is characterised by the relational paradigm. We are, however, more and more aware that the relational world view does not fully satisfy our needs, particularly when we want more research oriented computer technology. We have therefore tried using object oriented theories for the system analysis.

This article explains our analysis and gives insight into the development of a database design and applications based on the analysis.

We do not claim the analysis to be a fully realised object oriented analysis — but rather an analysis inspired by object oriented ideas. There are several reasons for this.

To begin with, our database project is hardly able to exploit all the possibilities of object orientation — our museum objects don't, for example, interact much with one another, neither do they normally activate other processes.

Further, object orientation is still so new that there is as yet no broadly accepted notation for presentation of analysis diagrams and data models — with the consequence that we have had to utilise a home–made, relation–like notation.

Most important of all is that implementation of the analysis has to be done in a relational context. This is due to the fact that there are no object oriented DBMS commercially available. We imagine that we will use Oracle, which is a relational DBMS, for some time. Applications will, however, be made more user–friendly by using a variety of Windows techniques.

23.2 SYSTEM DEVELOPMENT IN GENERAL

System development, generally speaking, consists of an iterative process of

1) Analysis
2) Design
3) Implementation

The system is defined by the analysis, and a data model and system rules are worked out in the design phase. The model is made visible and usable on the computer by implementation.

In recent years development has been fastest in the implementation and design phases. Database management systems have in this short time gone from hierarchical systems through networks to relational, and are now on the way toward object orientation. Implementation, which in the nature of things has always been the most visible, is today miles from the monochrome command line systems that only could be used by computer experts. Today, machines can be used almost without previous introduction and can even handle multimedia information such as pictures, movement and sound.

The analysis phase has until now been comparatively weaker. There is no doubt that many systems have been developed without any real analysis phase, but rather with analytical work spread throughout the design and implementation phases.

There are many reasons for the weakness of the analysis phase. An essential reason is historically determined since earlier there were severe limitations to what advantageously could be computerised. Interest was primarily directed towards accounting and administrative routines, and since one did not imagine real changes but only mechanical simulation of manual routines, there was no need for deep analysis. For the same reason one thought it sufficient for the system designer, who often had a computer science or engineering background, to acquire enough analysis methodology and technique to engage in a dialogue with the professionals for whom the system was to be built.

Since then one has begun to question to what extent administrative routines always reflect the best basis for programming. In fact, it seems that both administrative and functional routines can survive for long periods and thereby both control and hinder new thinking and development. It is thus no longer good practice for a systems analyst to begin analysis by asking the professional to «show me what you do». Today the questions are more subtle, in the direction of «tell me how you think». The best would naturally be for the professional to work with a team of analysts, and there are developments in this direction.

The latest recommendations for systems analysis are thus to look away from the existing routines and to try to analyse *reality*, or rather that part of reality for which one is developing a system. D. Frost calls this «the universe of discourse» in his book on expert systems (Frost 1986). G. Booch states

> «Object–oriented analysis is a method of analysis that examines requirements from the perspective of the classes and objects found in the vocabulary of the problem domain» (Booch 1991:37).

23.3 AN OBJECT ORIENTED ANALYSIS OF MUSEUM ACTIVITIES

As mentioned in the introduction, we already have a database at the National Museum for registering the museum's collections. It is a traditionally built relational database based on an analysis of earlier manual registration practice and is, as such, a model of the museum's administration — not of research in the reality in which the objects once participated. But research is also a part of our museum's activities.

It was clear to us quite early (Rold 1990) that, as our museum activities extend further than administering the collections, it was necessary to re–examine our *universe of discourse*. In other words we needed to try to build a database that was also capable of registering research into the reality in which the museum's objects once partook. In the new analysis we define:

- The universe of discourse is the universe of archaeological/ethnological research and museum management;
- The universe consists of events in which an object or a collection of objects can participate.

The diagram (Figure 23.1) can also be read as a mathematical formula:

1) One event can involve one or more objects;
2) One or more objects can participate in many events;
3) Many objects can be components of a given object;

Both *event* and *object* are quite abstract quantities. One should not at this point make up a catalogue of recognised events and objects. The concept "event" is rather a synthesis of a long series of observations of the universe of discourse, for example that objects seem to be fabricated, used, collected, excavated, etc.

The concept *object* perhaps demands a more precise definition, since an object is not necessarily a physical object. For instance it could also be a dance, an experience or something similar. In our universe of discourse an object is for now "something" that is a subject of museum activity.

The universe of discourse must not be confused with *the real world* to which the object once belonged. Reconstruction models of the real world must, for good reasons, build on our observations and interpretations of events and objects. Thus our universe can be used for model building

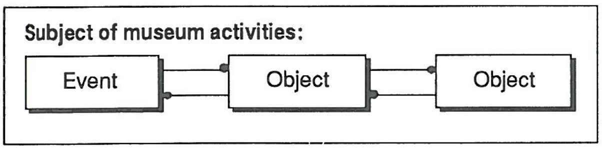

Figure 23.1: An object can, for example be a grave, wherein many other objects such as swords, pottery, etc. occur.

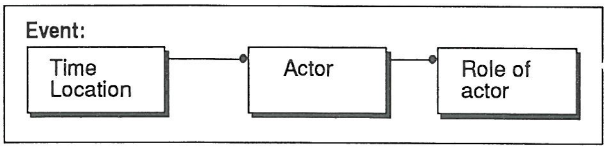

Figure 23.2: An event is characterised via time, place, and actor dimensions, but also by its type, for example a fabrication event, or an administrative event.

but is itself, as said earlier, only a model of museum activity.

23.3.1 Definition of *event*

An event is defined as something that takes place in time and space, perhaps on account of "someone" or "something". At museums of cultural history there will nearly always be human or institutional actors in an event, but the story would likely be different at natural history museums.

The analysis reaches the conclusion that events have common characteristics in dimensions describable by an old refrain: «who, what, where, when and why?» (Figure 23.2).

1) An event happens at a given location at a given time
2) One or several actors can be involved in an event
3) An actor can participate once or more times with one or more roles

Booch defines *event* by:

> «Things that happen, usually to something else at a given date and time, or as steps in an ordered sequence." (Booch 1991:141)

For example one could imagine that an object's fabrication occurs in three phases, where two types of components were made at two different factories, and assembly of the object was done somewhere else.

There is, by the way, an interesting question about the concept of event: «Is it an event when one does research on an object?». The definition of our universe cannot entirely exclude this possibility. But one can perhaps say that the universe in itself is research on subjects for museum study. Thus the question can be reformulated as «Is it an event when one does research on research?».

Pragmatically one can try out the question by recording time, location and the person who does the research in the system, and thereby discover that one registers data about where, when and by whom research in an object or event is done. This is not necessarily uninteresting, even though what one primarily wishes to register is the research result itself.

It may be useful to register the fact that two researchers disagree on the age of a flint axe for example. One can accomplish this by encapsulating the system "in itself" — that is to say that one can also enter new records with information about records entered into the database at earlier times.

This means that the basic structure can, for example, contain two records of the fabrication time for a flint axe, and connect with these records two other records that respectively register who, what, where, when and why the original records were produced.

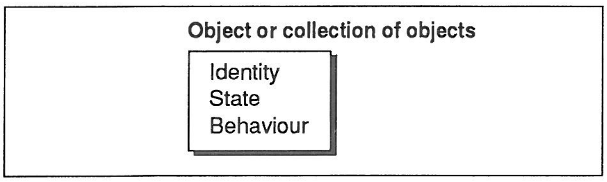

Figure 23.3

23.3.2 Definition of *object*

Objects too are characterised by common abstract concepts such as identity, state and behaviour (Figure 23.3):

1) An object or a collection of objects has an identity, a state and a behaviour in an event

Booch defines identity, state and behaviour in the following way:

> «Identity is that property of an object that distinguishes it from all other objects.» (Booch 1991:84)

> «The state of an object encompasses all of the (usually static) properties of the object plus the current (usually dynamic) values of each of these properties.» (Booch 1991:78)

> «Behaviour is how an object acts and reacts, in terms of its state changes and message passing.» (Booch 1991:80)

In a museum context an object's behaviour is the role it plays in a particular event. A large metal cauldron could, for example, be a washtub when fabricated, but later be used as a firewood container. In a collection or accession event, it should be possible to register why a given large metal cauldron was collected, making precise whether the interest in the cauldron is because it was an example of a washtub or a firewood container. It is interesting to point out though that an object's role in an accession event logically is purely and simply a *museum object*.

The object's role can also be defined from criteria other than function. A role can be conditioned by religious, social, sexual or age factors, just to name a few meaningful factors connected with the concept of role.

Object roles are dealt with through various classification systems, for example "OCM, Outline of World Cultures" (Murdock 1982). As we know, classification is an extraordinarily difficult subject, but problems of classification are not a primary subject in this analysis except that if a given classification system is based on functional or semantic factors, a given classification should be connected with the object's behaviour in an event, while typological classification systems should be connected with the object's state.

Reality is unfortunately not quite that simple since many classification systems are a mixture of function, semantics and typology, but it is not necessarily a weakness that the analysis reveals problems of this sort.

The analysis also shows that when an object partakes in an event, its name should be connected with its function, meaning, or form rather than with its identification, even though names seem to be important factors in identification — "washtub" contra "firewood container".

In the analysis for the existing database one assumed that a name is a part of the object's identification, which has led to a long series of problems. For example in the original database one could register several names for a large cauldron, but could not register whether a cauldron was fabricated as a firewood container and later used as a washtub or vice versa. In practice this may not mean so much since «we know how things hang together», but it is not certain that one will know 500 years from now — just consider medieval testaments in their original languages and in translation, and compare this with archaeological evidence!

Names are also connected with factors such as professional, dialect, or original language designations, not to mention nicknames like the "Sun Wagon" — one of the National Museum's famous finds.

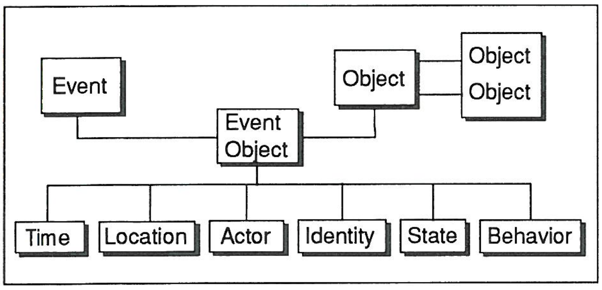

Figure 23.4: The prototype does not account for possible one-to-many relations such as several actors, measurements, materials, etc.

When analysing an object's state one has to take a position with respect to its kind. A physical object can be described as in the physical world via dimensions such as size, material and colour. An abstract object, for example a dance, could perhaps be described by its choreography.

Also here the analysis points out that an object's state depends on the events in which it partakes. It is characteristic of objects that they change form, for example they become worn, but also that they can be changed totally or go into another context than originally thought. For example Danish teachers in public school once had their students transform cutlery like forks or spoons into bracelets as part of the more artistic activities in basic training.

23.3.3 The analysis diagram

Through the definitions we have established until now, an analysis diagram can be made which abstractly explains our universe of discourse (Figure 23.4).

In the diagram the analysis is far from worked out in all details, but the fundamental ideas are established in a programmable outline, from which one could make a prototype for testing. In other words one can test the ideas without dealing with real data at all.

23.4 DEVELOPMENT OF A DATA MODEL

Based on the analysis diagram (Figure 23.4) we can develop a data model in a relational design notation (Figure 23.5). The data model does not account for possible one–to–many relations, for example that there may be several actors in one event. The model's purpose is only to serve as a prototype for testing the fundamental ideas of the analysis. The central tables EVENT, EVENTRELATION and COMMENTS will be discussed below.

23.4.1 The table EVENT

The EVENT table collects all those dimensions of an object which are relevant for our universe of discourse. The primary key, eventno, is a machine generated unique number, allocated mechanically every time a new record is entered. When an object participates in several events, new records are created and the object's identification is transferred to them.

The foreign key, eventtype, is fetched from a list of events which will contain, for example, *usage, fabrication, accession, excavation, moving, research*, etc.

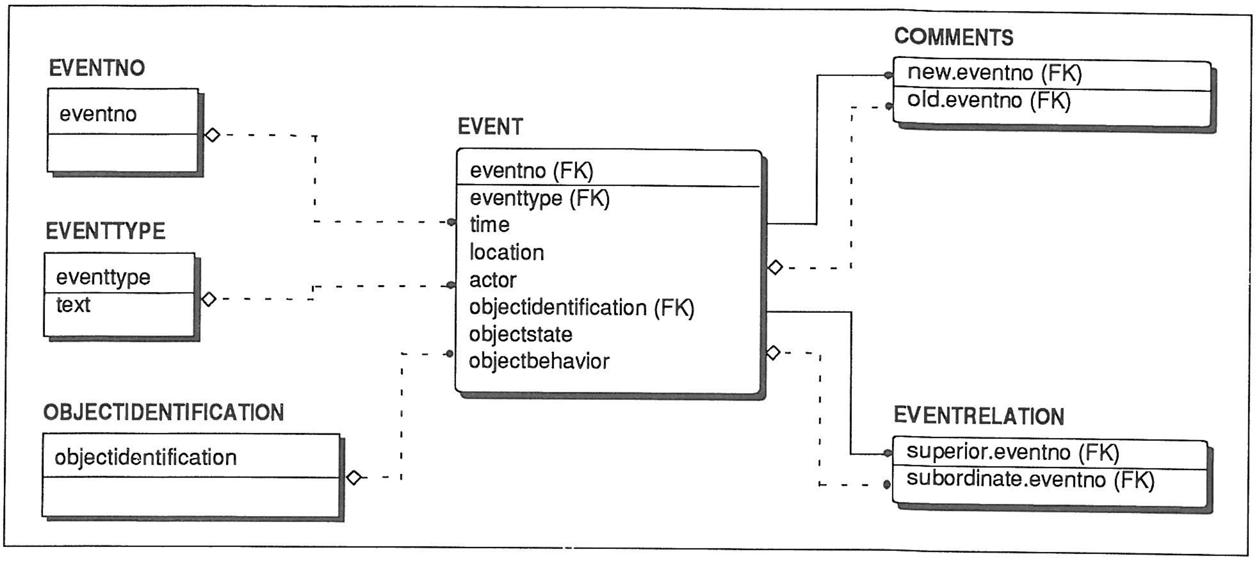

Figure 23.5: The relational datamodel for the prototype consists of only 5 tables.

The foreign key, objectidentification, is a machine generated unique number that identifies the object, which may as well be identified in one or more other ways, for example by an inventory number.

23.4.2 The table EVENTRELATION

When two or more objects are related this is registered in EVENTRELATION

It is quite normal that several objects are accessioned together, for example by donation, even though the objects may have no other relationship than being donated on a certain date by a certain donor.

It is of more interest that one can, for example, register that one object has been used to produce another object. One can, for instance, have a mould for making ornaments, and a number of ornaments cast using that form. Here the form's individual usage events correspond to the fabrication events of each individual ornament.

Too, it is conceivable that fragments of an object may have been accessioned many years ago, and that one has recently received yet another fragment of the same object. Here we will be able to couple the fragments by their fabrication event, but not by their accession events.

A common relation occurs between rather abstract objects like *livingrooms* which may only exists on the basis of the furniture that were placed there. The object *livingroom* may still be registered separately, and its usage event can be related to the usage events of each individual piece of furniture as well as probably the accession events.

23.4.3 The table COMMENTS

With any record in EVENT one can connect information of the same type as EVENT's own information — for example information about when, where, and by whom the original record was inserted. For example this is important in a research context where there can be uncertainty about the dating of an object. One really encapsulates the structure in itself, whereby one can register information about information about information indefinitely.

Encapsulation of this sort, which is a hierarchical data structure, can also be done by inserting eventno as a foreign key attribute in EVENT itself, but the case tool we use is not tolerant of a structure where a table can refer to itself.

23.4.4 Testing the model

While testing the model it may be necessary to give up abstraction in certain areas. For example one may define what an *event* is according to the definition of the analysis: «An event is a situation in which an object can be found». For example:

- Fabrication;
- Usage;
- Find: excavation or collection;
- Accession;
- Administration.

The list of events can be expanded, provided the definition still applies.

Further, the listed events are regarded as superior events that can be supplied with special

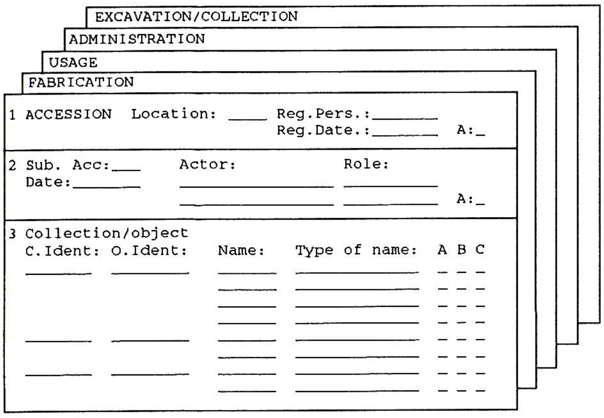

Figure 23.6:In a window environment it is possible to show all events in which a certain object has taken part. More than one object may be shown at one time making it easier for the user to relate the fact that two objects have been used in the same event.

types. For example a collection's event can occur via:

- Excavation;
- Detector;
- Scientific collection.

23.4.5 The attributes *time, place,* etc.

Dimensions time, place, etc. appear as attributes of the EVENT table. It is not required of an event that every such attribute contains data. By testing the model we can establish that, purely as an abstraction, it is able to satisfy the requirements we have made and thereby is an adequate model of our *universe of discourse.*

But an abstraction is one thing and reality is another, and it turns out that some problems arise when developing the model further. The problem concerns the representation of the dimensions. *Time* can for instance be represented by cultural or stylistic historical classification, date, year, interval, scientific dating, etc. This means that the time dimension in the EVENT table instead of containing, as in the abstraction, a single attribute must in the real model be represented by just as many attributes as there are ways to represent time.

Development into a real data model reveals that every dimension in the abstract model can be represented in several ways, so the result is an EVENT table with a number of attributes of the order of magnitude of the number of dimensions multiplied by the number of representation types. This can easily amount to several hundred attributes.

Logically seen this is no problem, but in practice such a construction would be unreasonably difficult to handle with respect to indexing, among other problems, and would thereby decrease the efficiency of every database function.

23.4.6 Metalanguages

At this point in our development, it is worth pointing out a needle's eye through which the camel can possibly be put. The solution could be found in the development of metalanguages. By this is meant a language that can be used to interpret other languages. For example the language *Metatime* could consist of a common time scale into which all ways of representing time can be fit.

Note that the concept of metalanguage is rather more sophisticated than the construction of simple conversion factors, which are not essentially different from establishment of standards. A metalanguage should not, for example, influence the real data, but just interpret them. Thus a metalanguage will be used as a superstructure to interpret both the base's internal data and the content of external bases — rather than as a database structure.

23.4.7 From abstraction to reality

Before developing the abstract data model further, it is a good idea to regard the system from an application viewpoint.

The application sketched in Figure 23.6 would best be implemented in a window– or hypercard environment where the user can easily get access to all routines and database functions.

The first screen image, ACCESSION, deals with an accession event. The other screen images, for example FABRICATION and ADMINISTRATION, are identical to ACCESSION. There is thus a widespread reuse of the attributes *time, place, actor, objectidentification, state and behaviour,* just as implied by the design of the data model. Application labels can, however, be changed to words that clarify the semantic content of a field. For example, *actor* in a fabrication event can be supplied with the label *maker* or *artist.*

The screen image is divided up into three sections:

1) Registration concerning the *place* where the event occurs.
   Further, information about when the registration takes place, and who does it.

   Location: is the name of the museum where accession happens, perhaps the name of a department in the museum.
   Reg.Pers./Reg.Date: the name of the person who does the registration, and the date when it

occurs. This is system information that has not earlier been mentioned in the system development.
A: A "button" for entering information.

2) Registration concerning time and actors.
Sub.Acc.: Subordinate event which contains further explanations about the event, for example subordinate accession situations such as "purchased", "donation" or "exchanged".
Date: The date of the accession, which often differs from the registration date.
Actor/role: the name of the actor and his role. For example "LR"/"Responsible curator", "BMN"/"Donor".
A: "Button" for entering information about the information.

3) Registration concerning object identification, object state, and object behaviour.
C.Ident.: Indicates whether the accession involves a collection of objects. If the collection has its own identification, this can be registered.
O.Ident.: Every object, whether or not it is part of a collection, is given an identification, for instance, an inventory number.
Name/type of name: Collections as well as objects can be registered under several names, although only names are relevant to an accession situation. If an object has had another role in its fabrication situation, it will often have had another name, the fabrication name will be entered when registering fabrication information.
A: "button" for entering information.
B: "button" to register the form the object has in its accession situation. The object's form typically belongs to the accession situation since, for example, physical dimensions can first really be observed when one has the object at hand. The original form, including reconstruction, also of inscriptions and ornamentation, will on the other hand be registered during the fabrication or usage situations.
C: "Button" for behaviour. The role the object plays in an accession situation will typically be, as mentioned earlier, "museum object". This need not be registered, but if the intention with the accession is defined via the object's current or earlier role, this can be registered.

It can be seen from this application sketch that all the dimensions *time*, *place*, identification, object state, etc. are relevant. Their meaning or content is, however, quite different in an accession situation than it would be in for instance an administrative registration concerning an object's travels through storehouse, exhibition and loan placements. However an object's form can be relevant even in such an administrative registration, for example if it is damaged.

After this investigation from an application viewpoint the next step towards the concrete system involves defining attributes, system rules, etc. and falls outside the scope of this paper. I shall however end this report with a few comments on the structure of the actual system.

23.5 VIRTUAL BASE AND CONCRETE VERSIONS

In an institution like the National Museum with many departments and objects from quite assorted times and places, one will discover that every department uses its own classification systems, inventory methods, etc. In our case we have chosen to respect the individual departments' special characteristics. This is partly because our registration project's first phase involved registration from older and sometimes very old sources, to which one must of course be faithful when transferring to EDP since otherwise one could never let the source material go and trust the electronic registration.

Another reason to respect the departments' special characteristics is connected with their individual prerequisites and wishes, and as well with their resources for research and exposition aided by the database. One cannot expect an equally high activity level everywhere, and it would not serve anybody's purpose to restrict high activity or to press a department with low activity.

It is clear that high activity and insight will influence further development of the system, and that this will eventually benefit everybody.

With these goals it is relevant to view the database as a virtual base, under continuous development, and consisting of a series of concretely worked out versions that are also being developed further. The versions can by need be regarded as and function as stand–alone bases, or they can be joined together with aid of the common structure.

23.6 CONCLUSION

One of the most exciting things about working with system development is to discover that things can be done in many different ways.

When one considers that the National Museum's first database with around one million records only functioned for five years before changes were needed, one can reasonably question how long the new system will be able to satisfy us. Now that we have more possibilities to register scientific data, we will doubtlessly soon be confronted with still more demands.

On the basis of today's knowledge about national and international system developments we must admit that there are quite large differences in the systems' analysis background and design, which makes integrating the databases difficult. It is, however, hardly a viable solution to seek standardisation of these areas, if only because standardisation cannot avoid being limitation. This would be all too high a price to pay, even though our visions also concern giving one another access to information in order to be able to work globally with our cultural heritage, both scientifically and in exposition.

Instead we can once again point to metalanguages as a possible solution. If we give access to our systems' structures, we can in co–operation construct metalanguages that interpret the individual bases in relation to one another. This demands, naturally, that our systems are well documented both syntactically and semantically.

It is therefore both important and pleasing that scientific institutions and museums have so far shown great openness about their own systems, and also willingness to discuss, and that throughout the past 10–20 years more computer science professionalism has gradually been developed within our circle of researchers and museum people.

References

Frost, R.A.
1986 *An introduction to Knowledge Base systems.* Collins.

Booch, G.
1991 *Object Oriented Design with Applications.* The Benjamin/Cummings Publishing Company, Inc.

Rold, L.
1990 Applications of computer science in museums and humanistic research. In D. Andrew Roberts (ed.) *Terminology for Museums. The second Conference of The Museum Association.*

Murdock, G. P. (ed.)
1982 *Outline of Cultural Materials.* Human Relations Area Files, Inc. 5th revised edition.

Author's address

Lene Rold
Nationalmuseet, Dokumentationsenheden
Ny Vestergade 11, baghuset 2.sal
DK–1471 København K