UDK: A European Environmental Data Catalogue

Oliver Günther
Institut für Wirtschaftsinformatik
Humboldt-Universität zu Berlin
Spandauer Str. 1
10178 Berlin, Germany
guenther@wiwi.hu-berlin.de

Helmut Lessing and Walter Swoboda
Niedersächsisches Umweltministerium
Archivstr. 2
30169 Hannover, Germany

Abstract

In this paper we give an overview of the UDK project, an international software engineering effort to facilitate access to environmental data. The UDK (Environmental Data Catalogue) is a metainformation system and navigation tool that documents collections of environmental data from the government and other sources. Potential users of the system include government agencies, industry, as well as the general public. Technically, the UDK is based on a three-way object model that distinguishes between environmental objects, environmental data objects, and UDK (meta) objects. Each (real-world) environmental object is described by a collection of environmental data objects. Each environmental data object is in turn associated with exactly one UDK object that specifies its format and contents. UDK objects are classified in accordance with an object-oriented inheritance hierarchy. They may be connected in a hypertext fashion to represent semantic associations, such as maintenance responsibilities and aggregation relationships. In January 1995, a first version of the UDK was made available in Austria; most German states will follow shortly. We also expect a prototypical version of the UDK to be available on the World Wide Web by early 1996.

Introduction

Efficient techniques for the management of environmental data are useful not only for decision makers in government or private industry. In environmental protection, the public demand for openness and data transparency is unusually high. Recent legislation reflects this trend, especially at the European level. According to a recent EU recommendation on environmental information (modeled after the American Freedom of Information Act), almost all data stored at public agencies should be made available to any citizen on demand (Council of the European Communities1990). The few exceptions refer to privacy issues, such as the protection of industrial secrets, which need to be shielded from public view by appropriate authorization mechanisms. In general, however, public records on private companies are not exempted from this ruling.

As a result of these political developments, there is a major demand for appropriate tools to manage and aggregate environmental information distributed in a heterogenous computer network. A particular need exists for convenient navigation aids that help users to take advantage of such a distributed information system, regardless of their computer literacy. Starting from some environmental query or problem formulation, such a navigation aid should help users to localize the relevant data sets and to retrieve them quickly and in a user-friendly manner. An essential prerequisite for both navigation and data transfer is a *metainformation system* that contains data about the format and the contents of the available data.

The *UDK (Environmental Data Catalogue)* is such a metainformation system and navigation tool. It documents collections of environmental data from the government and other sources. These data sets may be available either online or by request to the responsible data administrator. Potential users of the system include government agencies, industry, as well as the general public. The UDK helps them to get answers to the following questions:

- Which relevant information is principally available for a given problem?
- Where is this information stored?
- How can this information be retrieved, and how long does it take?

The UDK design presented in this paper is the result of several years of research and development (Lessing and Schmalz 1994, Lessing and Schütz1994, Lessing1989). In 1990, the Environmental Ministry of the State of Lower Saxony launched a research project with funding from the German Federal Environmental Protection Agency. Two years later, an international working group was formed to oversee the UDK design and its further development into a practical software tool. In 1994, Austria passed an Environmental Information Law that introduced the UDK as the official navigation tool for all environmental information on record. In January 1995, a first version of the UDK was made available in Austria; most German states will follow later this year. The UDK is currently also under evaluation by several other European countries, including Switzerland, Italy, Sweden, and Norway.

The UDK Object Model

The UDK is based on a *three-way object model* that distinguishes between environmental objects, environmental data objects, and UDK (meta) objects (Fig. 1). The term *environmental object* is used to describe the real-world objects making up the environment. This includes natural entities, such as lakes and biotopes, as well as man-made objects, such as factories or highways. Each environmental object is described by a collection of *environmental data objects*. There are a variety of ways to obtain such a description, including measurements, observations, but also value judgements. A typical environmental data object is a series of measurements that captures the concentration of a certain substance in a river (the corresponding environmental object).

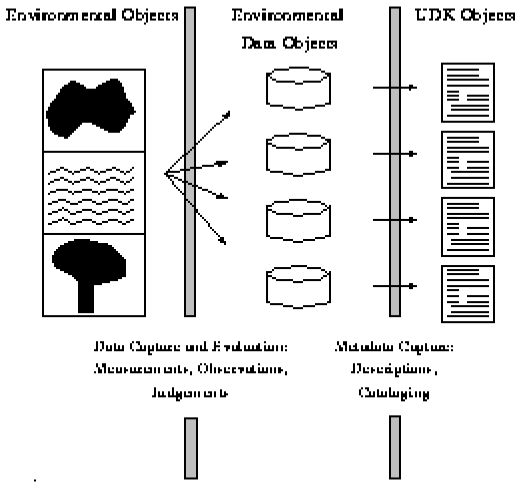

Figure 1: Three-way object model of the UDK

According to a recent definition (Swoboda et al. 1995), there are two criteria an environmental data object should fulfill:

1. *Environmental Context:* Environmental data objects are information that refers to real, material components of the environment (environmental objects), or to processes or projects that are related to such environmental objects;
2. *Data Capture:* Environmental data objects are information that is available in the form of a written text, sound, or images (analog or digital). It has been obtained by one of the following ways of data capture:
    - descriptive;
    - empirical;
    - by collection and aggregation of other data;
    - by a simulation or some other modeling process;
    - by value judgements.

Each environmental data object is in turn associated with exactly one metadata object that specifies its format and contents. On the screen, each such *UDK object* is represented by one or more masks; see Figure 2 for an example.

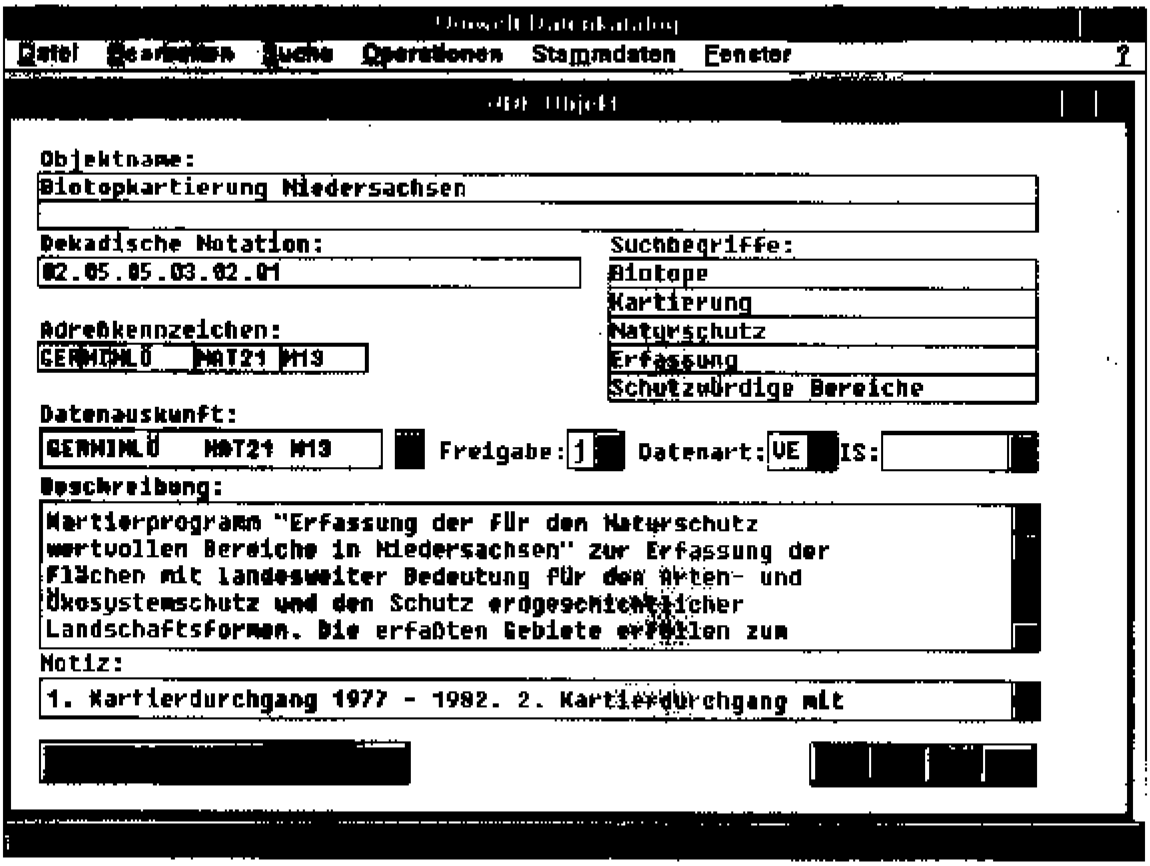

Figure 2: A mask representing a UDK object

UDK objects may exist for environmental data objects at various aggregation levels simultaneously. Consider, for example, a national groundwater database that contains a large number of measurements from all over the country. There is one UDK object representing this database as a whole. In addition, however, there may be one UDK object each for the measurements from a certain county, there may be UDK objects representing the measurements from a particular station, and there may even be UDK objects that represent single measurements. There may also be UDK objects for groupings that are orthogonal to this aggregation hierarchy, such as UDK objects representing the measurements that were taken in a given month. The decision to create an object is based on a cost/benefit analysis, depending on the particular applications a user has in mind. An important issue in this context is the maintenance of the semantic associations between those UDK objects and the corresponding environmental data objects; see Section 4 for further details.

Up to now, UDK objects have been identified by their position in the *primary tree*, a directed graph whose nodes correspond to the UDK objects and whose edges represent responsibilities of agencies and departments for particular sets of UDK objects, as well as part-of-relationships between large data collections (e.g. a groundwater database) and their components (e.g. the data sets corresponding to particular measuring stations). This approach to identify objects is unsatisfactory for a variety of reasons. Most importantly, UDK objects may lose their identity when they are relocated in the primary tree due to some reorganization (such as the transfer of a department from one ministry to another). In this case, the objects that were relocated have to be recreated under a new ID at the new location. As an alternative, we are currently investigating the possibility to use *object identifiers (OIDs)*, a concept well-known from the domain of object-oriented databases. OIDs are created by the system; they are usually not visible by the user. To guarantee universal uniqueness, the generation of the OID is usually based on the CPU number, as well as the current date and time-of-day.

UDK Object Classes and Inheritance

In order to structure the wide variety of UDK objects, and to facilitate both their capture and their administration, we recently presented a first proposal for a *UDK class concept* (Lessing, Günther, and Swoboda 1995). Here we distinguish between seven classes of environmental data objects:

3. project data (construction projects, environmental impact studies, etc.)
4. empirical data (measuring series, laboratory data, etc.)
5. data about facilities (factories, buildings, etc.)

6. maps
7. expertises and reports
8. product data
9. model data (simulations etc.)

For each of these seven classes there is a UDK class that contains the describing UDK objects. Each UDK class corresponds to a mask that is used for the capture and administration of the corresponding UDK objects. The basis for this pragmatic proposal were the user requirements that came up during the first months of UDK data capture. Obviously, this classification needs to be reviewed and possibly extended from time to time in order to reflect the change of user requirements. We feel it is important, however, that this top-level classification reflects a consensus of all UDK participants.

Another extension that is currently planned concerns the *vertical* structure of this classification. In particular, we intend to turn this flat class structure into an object-oriented class hierarchy that allows the inheritance of object attributes. The hierarchy should be structured as follows.

- The root of the hierarchy (level 0) consists of the generic class *UDK_Object* with four obligatory attributes: the unique object identifier (OID), the object name, the date when the object was last modified, and the agency (or the person) that is responsible for the object. Optional attributes, such as a textual description, may be included as well. Note that this generic class is not an abstract class, i.e., it may contain objects that are not included in any of its subclasses.
- Level 1 contains a relatively small number of classes that represent a consensus between all UDK participants. Currently, this level would correspond to the seven classes described above. Changes at this level are subject to negotiation between the UDK member countries.
- On the subsequent levels of the hierarchy, participating countries or agencies are free to introduce additional subclasses depending on their particular requirements. This kind of flexibility is important not only for efficiency reasons. It is also crucial in order to secure acceptance for the UDK throughout its intended user community, especially in government agencies at the national and the local level.

Class attributes are inherited along this class hierarchy in an object-oriented manner. This includes the possibility to upgrade selected attributes from being optional to being required. It also means that attributes that are specific to a certain subclass, but not to its superclass(es), can be masked out when looking only at the superclass. For example, consider a particular topographic map m and its UDK object . m is an element of the class *topographic_map*, which is a subclass of the class *map*. If one now looks at the UDK object through the mask corresponding to the class *map*, one only sees the attributes of *map*. The additional attributes that may have been introduced to describe *topographic* maps, as opposed to general maps, are not visible in this case.

This feature, which is typical for object-oriented environments, is a crucial element of standardization in the presence of application-specific extensions on the class hierarchy levels 2 and below. Any tool that is supposed to work at the national (or international) level across particular agencies or user communities can rely on the availability of the attributes defined at level 1. Maintenance and version management are other issues that need to rely on a stable class and attribute structure at the higher levels of the object hierarchy. It is therefore important to take organizational and technical precautions to make sure that users observe this principle in the presence of user-specific extensions and the resulting complexity in the class structure. The technical details of the implementation of these lower hierarchy levels are still subject to discussion.

Semantic Associations Between UDK Objects

Orthogonal to the class hierarchy described in the previous section, the UDK offers users the possibility to connect *concrete UDK objects* with each other in a hypertext fashion. The resulting structures are directed graphs whose nodes correspond to UDK objects and whose edges represent semantic associations between them or between their respective environmental data objects. The semantics of those edges may vary; we will later propose a type system for edges to make this aspect more explicit. Note that those semantic nets are completely independent of the class hierarchy described in the previous section. While the nodes of the class hierarchy are *UDK object classes*, the nodes of the structures described in the sequel represent *concrete UDK objects*.

The most important such graph structure is the *primary tree* or *primary catalogue*. Each UDK object corresponds to exactly one node of this tree structure, i.e., there is a 1:1 relationship between primary tree nodes and UDK objects. The links in the upper part of the tree serve to represent responsibilities of agencies and departments for particular sets of UDK objects. The agency that is in charge of a UDK object has to make sure that its information is correct and up-to-date. It is also responsible for the creation and deletion of UDK objects in the associated subtree(s). In the lower part of the tree, the links are used to represent part-of-relationships between large data collections (e.g. a groundwater database) and their components (e.g. the data sets corresponding to particular measuring stations). An example is given in Figure 3 that depicts the UDK objects related to a groundwater database. Here the solid arrows make up the primary tree; their semantics varies between ``is-responsible-for'' (in the upper part of the tree) and ``is-an-aggregation-of'' (in the lower part).

Depending on particular user requirements, there may also be *secondary catalogues* to represent other semantic associations. Like the primary tree, a secondary catalogue is a directed graph whose nodes each correspond to exactly one UDK object. Other than in the case of the primary tree, however, the resulting structure does not have to be a tree anymore. Note also that a UDK object can be referenced by any number of secondary catalogues.

A typical application of secondary catalogues concerns the representation of additional aggregation relationships that are not represented in the primary tree. In Figure 3 these kind of associations are pictured as dotted arrows. These kind of links are often useful in order to refer users to relevant aggregated data sets first before, upon request, giving them access to more detailed data. Another application of secondary catalogues is the construction of personal association structures. The ``debate'' association in Figure 3 (dashed line) is an example for such a structure. For these free structures the system does not require users to restrict themselves to a tree structure. Similar to the freedom one has for linking pages in the World Wide Web, any directed graph structure is permitted, including graphs with cycles. The idea is to give UDK users a maximum amount of flexibility to connect and associate the various information items making up their working environment. With an attractive user interface, this option should be of great interest to a large group of users. What is important is that it has to be reasonably easy to create personal UDK objects and links. Furthermore, it is essential that those ``personal'' structures can be isolated from the public part of the UDK, so users can build confidential structures that are visible just for them or for their team.

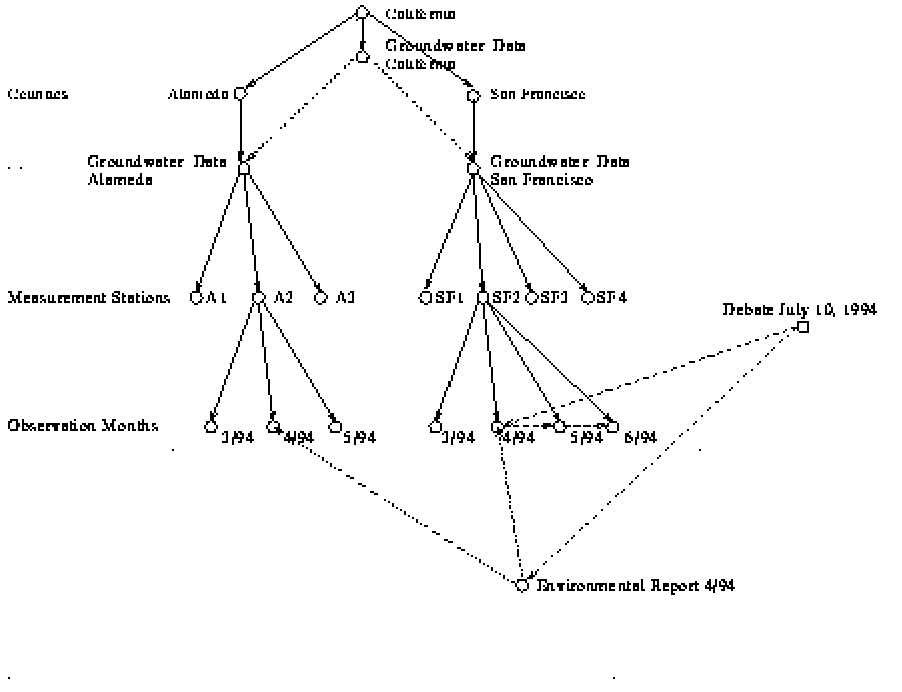

Figure 3: A selection of UDK objects and their associations in the context of a groundwater database

In summary, it is important to note that the links connecting UDK objects may have a great variety of semantics. These different types of links need to be made explicit in the UDK by a labeling scheme. Users should have the option to choose the types of links they want to see at a given time. This would allow them to see a UDK object in a variety of contexts and to switch back and forth between those different representations. On the screen this could be supported, for example, by different colors and drawing modes for different types of links (Fig. 3).

Conclusions and Future Work

The UDK is a metainformation system and navigation tool that documents collections of environmental data from the government and other sources. Given the extreme success of the World Wide Web (WWW), we expect a significant amount of this kind of data to be available via WWW in the very near future. At this point there is no question that the Web is the most promising option to follow the spirit of the EU guideline and to make environmental information *really available* to anybody who is interested.

The UDK could play a major role in helping users to navigate in this overwhelming information pool, to identify which data is relevant for a given query, and to retrieve it fast and in a user-friendly manner.Both Austria and the German state of Baden-Württemberg have recently commissioned WWW implementations of the UDK, and we expect prototypical versions to be available on the Web by early 1996. The technical realization is straightforward. Each UDK object can be turned into a Web page. HTML links can then be used to implement primary and secondary catalogues and to establish connections to the actual environmental data objects. (A slightly different approach was presented by Kramer and Quellenberg (1996) and Koschel et al. (1996)). Given the openness of the Web paradigm, the integration of most relevant data formats (such as postscript documents, sounds, or images) is easily possible.

References

Council of the European Communities (1990, July). Council Directive (90/313/EEC) of 7 June 1990 on the freedom of access to information on the environment. *Official Journal of the European Communities L158*, 56-58.

Koschel, A., R. Kramer, R. Nikolai, W. Hagg, and J. Wiesel (1996). A federation architecture for an environmental information system incorporating GIS, the World Wide Web, and CORBA. In this volume.

Kramer, R. and T.Quellenberg (1996). Global access to environmental information. In R. Denzer, D. Russel, and G. Schimak (Eds.), *Proc. 1995 International Symposium on Environmental Software Systems*, International Federation for Information Processing (IFIP), London, pp.209-218. Chapman and Hall.

Lessing, H. (1989). Umweltinformationssysteme - Anforderungen und Möglichkeiten am Beispiel Niedersachsens. In A. Jaeschke, W. Geiger, and B. Page (Eds.), *Informatik im Umweltschutz*, Berlin/Heidelberg/New York. Springer-Verlag.

Lessing, H., O. Günther, and W. Swoboda (1995). Ein objektorientiertes Klassenkonzept für den Umwelt-Datenkatalog (UDK). In *Space and Time in Environmental Information Systems*, Marburg. Metropolis-Verlag.

Lessing, H. and R. Schmalz (1994). Der Umwelt-Datenkatalog Niedersachsens. In A. Engel (Ed.), *Umweltinformationssysteme in der öffentlichen Verwaltung* . R. v. Decker's Verlag.

Lessing, H. and T. Schütz (1994). Der Umwelt-Datenkatalog als Instrument zur Steuerung von Informationsflüssen. In L. Hilty, A. Jaeschke, B. Page, and A. Schwabl (Eds.), *Informatik für den Umweltschutz*, Marburg. Metropolis-Verlag.

Swoboda, W., H. Lessing, P. Grolimund, and O. Günther (1995). Metadatenklassen im Umwelt-Datenkatalog (UDK). In F. Huber-Wäschle, H. Schauer, and P. Widmayer (Eds.), *Proc. GISI 95*, Berlin/Heidelberg/New York. Springer-Verlag.