Re-engineering the German Integrated System for Measuring and Assessing Environmental Radioactivity

Ernst-Erich-Doberkat
Chair for Software Technology, University of Dortmund and Software Technology Group, Informatik Centrum Dortmund, 44221 Dortmund, Germany

Fritz Schmidt
Institut für Kernenergetik und Energiesysteme, University of Stuttgart, Pfaffenwaldring 31, 70550 Stuttgart, Germany

Christof Veltmann
Software Technology Group, Informatik Centrum Dortmund, Joseph-von-Fraunhofer-Str. 20, 44227 Dortmund, Germany

Abstract

IMIS is an integrated system for measuring and assessing environmental radioactivity, it is run by the German Federal Agency for Radiation Protection (BfS) in cooperation with the Federal Department of Environmental Protection (BMU). It was established in the aftermath of Czernobyl and is aimed at informing federal authorities on the current level of atmospheric radiation, taking appropriate actions and informing the public when this level exceeds critical limits.

The system was conceived in the late eighties and put to operation in the early nineties. From a computer science point of view it is an integrated system comprised of software, hardware and net components. These areas have advanced tremendously since the establishment of the system, consequently it is necessary to reengineer the system taking te progress in these areas into account.

This paper* will first outline the tasks assigned to IMIS with an eye towards their realization, then we will discuss the technical aspects of reengineering the system. The requirements for the new version of IMIS are indicated, and we discuss how they should be implemented. It will turn out that before putting

*A more detailed version may be obtained from the first author; see also `http://ls10-www.informatik.uni-dortmund.de/LS10/Pages/Berichte.shtml`

Environmental Software Systems Vol.2 R. Denzer, D.A. Swayne & G. Schimak (Eds.)

the new system into operation it is advantageous to introduce a prototyping phase. We will discuss the requirements for this prototype.

1 RESPONSIBILITIES OF THE INTEGRATED MEASUREMENT AND INFORMATION SYSTEM

The responsibility of IMIS is described roughly as obtaining and assessing environmental radioactivity. This includes obtaining radiological data pertaining to air, precipitation and waters (German Sea, Baltic Sea, federal waterways). The radiation pertaining to food, fodder, and drinking water is also measured, and collected. The former group of data is gathered by federal agencies, the latter one by the States. Measurement of radiological data is done through some 2600 measurements points distributed evenly over Germany. The collected data are kept in part by the States, in part by federal agencies. In any case the data for environmental radiation are made available to the Federal Department for Environmental Protection. The task of collecting and partially interpreting these data is performed, for enabling the Federal Department of Environmental Protection to issue an advanced warning if artifical radiation is perceptible to an significant level in the environment, to assess the situation in the case of a radiological emergency, to issue recommendations for the protection of the citizens and to initiate appropriate preventive measures.

Collecting the data is a distributed task, federal and state agencies operate some seventy systems for obtaining the data. In addition to collecting data some prognostic computations (e.g. forecasting the propagation of radiation) have to be performed.

IMIS's tasks can be grouped from an information processing point of view *presentation*, obtaining the measurements, representing the corresponding data and making them available, *application*, assessing the data and interpreting them, *storage*, administering and archiving the data, and finally, collecting data streams and distributing data through a local or wide area network.

Currently this system is realized using DEC-stations of the 5000 series, operating under Ultrix with TCP/IP, X-windows, NFS; Ethernet is used as local area network and the Datex-P service provided by the German Telekom is used for wide area networks. The software includes the database system Oracle, UNIRAS for graphics, the geographical information system TERRA, the statistical package SAS, Q-Office for textprocessing and office communication, the PARK-system for prognostical computations. UNIRAS, TERRA and PARK were developed specifically for IMIS by outside vendors which still maintain these systems.

System evolution is very difficult, if not impossible due to heterogeneous software components with rather unusual interfaces and puts using custom developed software to a serious test. The user interface was conceived and developed by the end of the eighties and is still textually oriented. This interface is common to all application functions.

2 REENGINEERING IMIS: REQUIREMENTS

Distributing the data properly is at the very heart of IMIS. The current version distributes and stores in the data local databases at each IMIS node. This implies that maintaining the data in a consistent state becomes of utmost importance. The idea of *not* distributing the data geographically but rather maintaining them in one central location is furthered by progress in database technology and by the availability of sufficiently fast wide area networks. It is, however, not sufficient to have one central data store because there are different, geographically distributed locations in which the data together with certain secondary data have to be made available for political reasons.

This suggests a client/server architecture consisting of several clusters, in which the roots of the corresponding clusters are connected via a fast network. These roots store data which should be locally accessible. From a user's point of view work proper is done on the clients, hence clients should be configured so that they have all necessary tools at their disposal.

The distribution of the aforementioned four tasks among clients and servers can be done in different ways, depending on such parameters as net load, safety and security. It is obviously risky binding IMIS too early to a particular system architecture, and rather advisable doing some exploration first. In the context of system construction this means that prototyping comes into play.

Software Prototyping Prototyping is technique in software technology applied when the requirements for a software product are fuzzy or unstable. Usually a software prototype is built under these circumstances which then constitutes an executable partial model for the software to be constructed. This enables the customer to assess the functionality of the products, the systems engineer gets valuable hints concerning the construction of the product. Experience shows that prototyping is an iterative process because usually the requirements stabilize after going through some prototyping steps. This technique is well known in software engineering, but is not confined to software systems: the *spiral model* is one of the more elaborate process models in systems engineering, it includes prototyping as one of the central tools for risk assessment.

The risk in the sense of the spiral model associated with this reengineering project flow from the following observations:

Although the database product (Oracle) is going to be used in the new version of IMIS, we will use a new release which has some powerful new functionalities. The configuration we are going to work with will be rather heterogeneous, thus it has to be established experimentially to what extent this database product is suitable. Further, networking services are currently in a flux, and studying the specifications given by the network providers did not give us sufficient certainty which setup for a wide area network is suitable. Finally, the *old* IMIS used some products which had been developed specifically for this purpose. These products are replaced by standard software products as far as possible. They have to cooperate in a way apparently not explored yet.

Each of these points is not necessarily so complex that it warrants a specific prototyping phase. Their combination, however, occurs to us as rather risky leading us to the proposal of prototyping the system and possible alternatives before we suggest realising it. This makes it imperative identifying the services which should be available to IMIS, and to have a look at the consequences of distribution for different network topologies.

Services vs. Client/Server Architectures The architectural considerations aim at identifying subtasks which can be performed as autonomous services. Depending on this partition a hardware- and network architecture is developed which is composed from computers of different types, which in turn may come from different manufacturers. A rough identification introduces these three classes of machines: clients as user front-ends, database servers maintaining the system's data, and finally compute servers to be used for modelling complex analysis. Being highly dedicated and working on demand, the latter class is not considered further in this paper. In order to build up a modular and heterogeneous system flexibly, the software should be portable, dependencies on a particular hardware architecture being encapsulated and clearly described. In addition good software engineering practice demands encapsulation of services.

The data flow and hence the performance of a distributed system is determined by the partition of tasks between clients and servers. One of the results we expect from prototyping is finding a suitable partition under the constraints given in IMIS.

Classifying Services According to Their Functionality The classification of software has been indicated above. We have to cater for the following services: operating system, database system, geographic information system (GIS), statistics package, office communication, basis functions (including operating the system, controlling the process of collecting and sending data, computational services, managing dialogs, backup services), application services (including maintenance of data and samples, archive functions, graphics, computing forecasts), and finally documents specific for IMIS (including reports and other specific documents).

The geographical information system and the statistic package are systems which are used by application functions. Hence there are not to be considered on their own but rather in the applications' context.

Given this classification of services a first assignment of services to machine classes can be provided. The idea is keeping the system lean and assigning services only to those nodes which actually need them. We assume that each cluster contains at least one client which does not have any database functionality. This client will be used for graphics and for office communication. It seems to be sensible that this client is realized as a PC system. The graphics and the office communication should communicate using an interface which permits the integration of these functionalities. This enables the users to process all those IMIS documents using spreadsheets and text processing on a client.

Each client should have access to a geographical information system, so that a desktop GIS front end should be selected. The selection implies that the communication between the database and the geographical information system may become critical. The feasible strategies include distributed application, in which the servers contain the database, local databases for each client, so that the data are maintained through a fast LAN, and finally the remote maintenance of data requiring a fast WAN between clients and servers.

Topologies for the Net The current net topology is mainly characterized by the fact that it is realized using the Telekom service Datex-P (X-25 based). Currently we have 59 net nodes, capacity ranging between 4.8 and 9.6 KBit/s. The first idea is to replace this rather slow connection by ISDN, so that each node now has a capacity of multiples of 65 KBit/s.

The second alternative would replace Datex-P by the Datex-M (DQDB based) service with higher transmission rates and more fault tolerant service. This alternative has some implications for the net topology: there are some critical connections leading to BMU which might be advisable to equip with up to 2 Mbit/s. Thus only one database server is needed. It may be accessed from all nodes in the system with sufficient performance. The main drawback for the solution, however, is a political one: it implies that the databases in the States would be merged with the federal database, so that the States would have to access one database which is operated and maintained by the federal government. This seems to be incompatible with the federal structure of Germany.

A technical blend of these possibilities is provided by the third alternative: a backbone net is identified and realized using Datex-M, all other connections are realized through ISDN. This yields a fast backbone net, the traffic between state and federal authorities would not rely on a central database. It would enable the States to maintain their own environmental data. The quality parameters (interconnections, security) are a blend between those of Datex-M and ISDN as well, the same applies for the cost structure.

3 DATABASE ISSUES

Compared to the current IMIS, the database component obtains an even greater weight. We observe that the DBMS used, viz., Oracle, performed commendably and should be used also in the new version. Pragmatic issues recommend this even further: a switch to another DBMS would imply a careful and very early review of the database design. Since the migration is sufficiently complex, we decided not to redesign the database scheme. There are, making data available to applications and maintaining the consistency between different database servers will have to be addressed. Improvements in these areas would be greatly appreciated, and we will discuss three views of tackling this problem.

Hardware It is advisable using two classes of database servers. The current physical location for a database server should remain the location for the central server tasks of this server include administration and archiving. The second class of servers could be installed physically at other places within IMIS. They should make those data available which have to be accessible for nodes located there or nearby. The decision how to handle this situation specifically and how to configure these machines is up to prototyping.

New DBMS Functionalities Oracle offers with its new releases distributed databases and replication as approaches for a distributed information management. In a distributed environment the local DBMSs are organized so that the internal structure is transparent for the application. Accessing data is done in two phases (building up a connection to a DB-server, access proper), each node in a system has to be aware of the storage location of data. Thus each node has to maintain these informations concerning the distribution.

Replication keeps part of the data redundantly in different locations obriating network access. The consistency of the redundant data has to be guaranteed by the DB instances. Consistency of data may require heavy access to the network. The prototype is going to investigate ways of combining techniques of replication and distributed databases.

Mapping Services and the Database to Topologies Summarizing this discussion it is clear that we have to map the services and the DBMS to a particular net topology. Given the new functionalities offered by the DBMS, we decided to suggest realizing a prototype on the base of a net with a fast backbone, and net access provided by the users' needs. This net topology has to be refined, and one of the tasks for prototyping will be to arrive at requirements for the modelling and the hard-/software components of the new IMIS experimentally.

4 REQUIREMENTS FOR THE PROTOTYPE

We are now in a position to specify the requirements for the prototype in greater detail. The following problems will have to be addressed using the prototype: *requirements for the databases* and *distribution of the database* depending on the netstructure, *performance requirements* for the database server, *distributing services* in a client/server architecture, taking particularly standard software products into account, and finally *realization* of graphical user interfaces.

Judging from previous experiences with IMIS the prototype should also address the following questions: *Hardware*: how is the hardware for the central and the decentral servers to be sized and configured? *Software*: what is the interplay between the database and application functions, in particular with respect to visualizing data? *Database*: how are the new functionalities of Oracle to be utilized? How does the database perform when many competing clients impose a high load? Is the optimization performed sufficient or will after all a redesign of the database structure be necessary? What backup concepts are to be recommended? *Organization*: should we use Oracle tools for supporting workflow management systems to optimize the IMIS operation? Exactly what background data have to be kept at clients? *Security*: what redundant assess paths have to be investigated if the net becomes partly inoperational? Can mobile communication help here? What can be done to prevent hardware or software crashes?

Some of these aspects would have to be discussed in greater detail (e.g. security or process modelling). Time and space limitations however prevent us from doing so.*

Structuring the Prototype The prototype should consist of four components which are interconnected through different wide area networks: one central and several local database servers, pools of clients an a mobile component. The clients are used for investigating questions not requiring a DBMS but nevertheless access to a WAN. The mobile component helps investigating data transport independent of physical nets.

The hardware platform will include two sufficiently powerful database servers and Pentium PCs running under Unix and Windows NT, respectively. The software platform is composed of standard software as much as possible, including ArcView as a geographical information system. From a performance point view, implemented in the old IMIS by dedicated software, we have to investigate how the functionalities carry over.

Evaluating the Prototype Prototyping is not a means in itself but part of the systems development cycle, stabilizing requirements or indicating alternative avenues to go. Hence, a prototyping phase is sensible only when the prototype is being evaluated later on. As a general rule, the guidelines for evaluating a prototype should be fixed before the prototype is constructed. Being experimental in nature, performance evaluations should be done whenever quantitative data can be made available. This is not always possible: for example the user friendliness of graphical user interfaces cannot be evaluated numerically, similarly for the user interface, for user friendliness of the overall system, for supporting the organization, and error handling. Details can be fixed here only when the experimental program has been developed.

Let us run briefly through the area in which the prototype will be investigated: *Hardware*: the questions to be pursued here are of an experimental

*The interested reader is referred to the manuscript *Doberkat, E.-E., Schmidt, F., Veltmann, C. Specifying Requirements for a Prototype for Reengineering IMIS (in German), Bundesamt für Strahlenschutz, Freiburg, 1995*, which is available upon request from the first author.

nature, based on the measurements. We expect being in a position making qualified suggestions about dimensioning the computing system. *Software*: combining the database and applications functions will also be investigated experimentally, based on existing possibilities for interfaces. These experiments will be evaluated. *Database*: the new properties of Oracle have to be investigated for combining old and new properties of this DBMS. Testing will have to be performed using load simulation yielding quantitative data. *Organisation*: we did not discuss this point in greater detail, but would like to hint at some ongoing development. Modelling the workflow may be done either through the tools provided by Oracle. As an alternative, we turn to business process modelling with a vintage tool. We will see.

In summary we have reduced the IMIS structure to the structure of a prototype and will anticipate the behavior of the new IMIS from the behavior of the prototype.

5 CURRENT STATE

For practical purposes it was decided that prototyping is being done using two phases. The first phase will be a local evaluation: A local configuration will be built up with just one server and very few clients. This should implement the basic functionalities and just get the system going. Phase 1 will start in early 1997 and will take six months. The second phase will add more servers to the system. They are geographically distributed enabling us to assess the networking facilities. Each server will be accompanied by several clients for implementing the client/server architecture prototypically. These servers will not have the same hardware architecture. We rather prefer to include hardware from different manufactures for not binding IMIS to one particular hardware vendor.

6 CONCLUSION

Reengineering IMIS means not just porting this complex system to a new machine architecture, but rather engineering a new design reflecting the current state of client/server architectures, data base systems and net services.This new design is conservative: those parts worth saving will be ported and the considerable experience that comes with operating IMIS serves as a knowledge base for design decisions. Support for reengineering comes from prototyping, and here we have shown that parameters influence setting up the prototype.

Acknowledgement The work this paper is based on has been partially funded by BfS, the German Bundesamt für Strahlenschutz.