KfK 5277
Januar 1994

Design Specification of the Scientific Database System for MIPAS Satellite Experiment

E. Kapetanios
Institut für Angewandte Informatik
Energie- und Stoffumsetzungen in der Umwelt

Kernforschungszentrum Karlsruhe

Kernforschungszentrum Karlsruhe

Institut für Angewandte Informatik
Energie- und Stoffumsetzungen in der Umwelt

KfK 5277

Design Specification of the Scientific Database System for MIPAS Satellite Experiment

Epaminondas Kapetanios

Kernforschungszentrum Karlsruhe GmbH, Karlsruhe

Als Manuskript gedruckt
Für diesen Bericht behalten wir uns alle Rechte vor

Kernforschungszentrum Karlsruhe GmbH
Postfach 3640, 76021 Karlsruhe

ISSN 0303-4003

Entwurfsspezifikation eines Wissenschaftsdatenbanksystems für das MIPAS Satellitenexperiment

Bei der Betrachtung eines wissenschaftlichen Experiments sollte auf zwei Hauptpunkte hingewiesen werden. Als erstes muß die Geschichte der Generierung wissenschaftlicher Datenprodukte aufgezeichnet werden, damit ein Mechanismus verfügbar wird für die Verfolgung der abgeleiteten Datenprodukte. Als nächstes muß ein Datenbanksystem mit den Merkmalen der Verwaltung von wissenschaftlichen Daten entworfen werden, beispielsweise Zugriff auf große Datenmengen, Verwaltung unterschiedlicher Datentypen (numerische Daten, Text, Graphiken, Bilder, usw.), die Erklärung und Integrationsaspekte durch Metadaten, die Überwachung der Datenbankgenerierung, usw.

Ein Wissenschaftsdatenbanksystem wurde entworfen, das den Bedürfnissen der Fernerkundung aus dem Weltraum entspricht und Atmosphärenforschung unterstützt, in unserem Fall das MIPAS-Satellitenexperiment innerhalb der von der ESA (European Space Agency) geplanten ENVISAT-1 Satellitenmission. Grundlage des Entwurfs ist ein erweitertes föderatives Datenbankschema, wobei autonome und heterogene Komponenten integriert sind. Dazu gehören die operationale Datenbank für die Generierung von MIPAS Datenprodukten, das massive Speichersystem mit dem dazugehörigen Dateiverwaltungssystem, das Visualisierungsverwaltungssystem und das Dokumentationssystem für wissenschaftliche Ergebnisse.

Ein objektorientiertes Datenbanksystem (Forschungs- und Entwicklungsdatenbank) wird einen Objektraum liefern, worüber das globale Schema als gemeinsames Datenmodell für alle Komponenten modelliert werden soll. Es wird Metadaten in Form von Navigations- und Annotationsdaten verwalten. Es muß die Beziehungen innerhalb und zwischen den einzelnen Objekten erfassen - beides Messwertdaten und Metadaten als Objekte modelliert - damit die Integration und Adressierung von Objekten aus den entsprechenden Objekten ermöglicht wird, d. h. die Adressierung der relevanten Meßwertdaten im massiven Speichersystem mit Hilfe von Metadaten, die Komposition von unterschiedlichen Objekttypen (Annotation durch Text oder Graphiken), usw. Darüberhinaus wird eine Schichtenarchitektur für den Zugriff auf TeraBytes in Betracht genommen, wobei die Anforderungen wissenschaftlicher Anwendungen für sowohl eine direkte Handhabung von Dateien, als auch durch ein Datenbanksystem berücksichtigt werden.

Der in der objektorientierten Datenbank modellierte Objektraum erfaßt auch die Wissensbasis, worauf die Ableitungshistorie beruht. Dieses wird durch die Kopplung des Daten- und des Prozessmodells erreicht. Auf dieser Basis sollte ein Inferenzmechanismus aufgebaut werden, damit die Konsequenzen der Wiederverarbeitungsanforderungen auf das Datenmodell ermittelt werden können.

Schlagworte: Wissenschaftsdatenbanken, föderative Datenbanken, Multimedien, aktive Datenbanken, dynamisches Verhalten, Ableitungshistorie, Objekt-Orientiertheit.

Abstract

Dealing with a scientific experiment, two major issues should be pointed out. At first, the scientific data generation process history must be captured, in order to provide a tracing mechanism for any kind of derived data. Secondly, a database system has to be designed, coping with the characteristics of scientific data management, like access of vast amounts of data, managing various types of objects (numerical data, text, graphics, images, etc.), their explanation and integration aspects in terms of metadata, monitoring of the database generation, etc.

A design approach of a scientific database system (SDBS) has been taken, for the needs of remote sensing from space concerning atmospheric research, in our case of MIPAS satellite experiment. The design of SDBS has been based upon an extended federated schema, providing an integration facility of heterogeneous and autonomous components. These are the operational database dealing with the generation of scientific data products, the mass storage system and its associated file management system, dealing with the management of scientific datasets, the visualization management system, and an authoring system for documentation of scientific results.

An object-oriented database system (Research and Development Database) will provide an object space over which the global schema should be modelled, as the common data model for all heterogeneous components. It will manage metadata in terms of both navigation and explanation data. It must capture the intra- and interrelationships among the various objects–both datasets and metadata considered as objects–enabling the integration and addressing of objects from their counterparts, i.e. the addressing of the relevant datasets at the mass storage system with help of metadata, and the composition of different kinds of objects, for explanation and annotation purposes, addressed at the various heterogeneous components. Furthermore, a layered architecture for accessing terabytes of scientific data will also be considered, taking in account the requirements of scientific applications for both handling with files directly, and/or through a database system.

The object space modelled over the object-oriented database system, will also express the knowledge base of the derivation history coupling the data and generation process models. An inference mechanism should be built upon this knowledge base in order to extract the consequences of reprocessing requirements on the scientific data.

Keywords: Scientific databases, federated databases, multimedia, active databases, information systems behavior modeling, derivation history, object-orientedness.

Contents

List of Figures

Introduction

Dealing with a scientific experiment, two major issues should be pointed out. At first, the scientific data generation process history which must be captured, in order to provide a tracing mechanism for any kind of derived data. Secondly, a database system has to be designed, coping with the characteristics of scientific data management, like access of vast amounts of data, managing various types of data (numerical data, text, graphics, images, etc.), dealing with explanation and integration aspects in terms of metadata, monitoring of the database generation, etc. These are considered to be the main characteristics of scientific experiments concerning atmospheric research too.

In the last years, an increasing demand is emerging on developing and improving database systems for dealing with scientific data, on the purpose of monitoring global climate change. Efforts in this area have been untertaken by universities and other research institutes [SFD93, HGW93], aiming at the contribution of database technology in understanding climate dynamics. Specifically, building up a scientific database system for doing atmospheric research has common characteristics with related systems which should support other scientific disciplines.

Conventional database techniques are not adequate for handling of large amounts of scientific data, gathered during a scientific mission, like the satellite mission[1] considered in this paper. The remote sensing instrument (MIPAS), delivering raw data of atmospheric parameters, must be complemented with a scientific information system located in the ground segment (KfK Research Centre - MIPAS PAC[2]) of the satellite mission. It deals with the further processing of raw data, generating and managing data products of various abstract levels. The core segment of this information system will be the scientific database for operational and research data. According to the deficiency of conventional database systems to cope with non-business-oriented applications, a scientific database system design approach is presented in this paper, which is based on an extended federated schema, enhanced by a knowledge base system through which the data derivation history is going to be managed.

At first, some scenarios will illustrate the specific requirements coming out of atmospheric

[1] ENVISAT-1

[2] Processing and Archiving Centre

scientists' work with the system to be developed (chapter 1). In chapter 2, the most characteristic design issues, in association with the deficiencies of conventional database systems, have been stated out, in order to provide a system capable of supporting atmospheric research. The system requirements and design issues described in the first two chapters drive the design approach of the scientific database system presented in chapter 3. The scientific database schema is based on an extended federated schema, including various types of components, which are not regarded to be only conventional DBMSs. It has also been allocated to a federated server consisting of autonomous system components providing a distributed client-server environment over a common local area network.

Chapter 1

On scenarios of scientific data manipulation requirements

We will try to illustrate the design issues of the' proposed architecture by making use of some characteristic scenarios of scientific data handling requirements, as they have been specified by the prime investigators of MIPAS satellite experiment, concerning atmospheric research from space. In parallel, the deficiences of conventional database systems will be made clear, with respect to the whole spectrum of functionalities that must be provided. At this point, an overview of the main system requirements is given, as they have been specified in [Kap93], aiming at making the reader familiar with the system functionality.

At first, raw data received from the satellite platform must be calibrated and further processed, leading to more abstract data levels up to trace gas 2D/3D maps of the earth's atmosphere. This will happen on an operationally basis (24 hours a day and for 4 years), requiring a data archivation facility of all intermediate scientific data products, like interferograms, calibrated spectra, trace gas profiles, 3D gridded atmospheric data and maps, as well as some special data products which are going to be derived in off-line mode. A mass storage system for archiving and handling the derived scientific data (ca. 10-12 TeraBytes) must be provided, in terms of managing not only secondary, but also tertiary memory.

Furthermore, monitoring facilities of data products generation and displaying of low levels data (e.g., calibrated spectra) in order to undertake radiometric corrections and calibration process improvement, are considered to be essential functionalities during the generation process. The generation process should be automated, except of the derivation processes of special data products, which will be activated interactively. A powerful visualization system should complement the 3D spatial data model of the third data level, for purposes of looking inside the atmosphere.

The following scenarios will demonstrate the increasing demand of providing a scientific database system enhanced by properties which cannot be supported efficiently by conventional database systems.

1.1 The scenario of derivation history

Acting in a processing environment like the operational data products generation, the need of capturing and modeling the generation process must be addressed. Developing and improving the calibration process physics, or the physics concerning the subsequent process of trace gas profiles generation, would lead to an algorithm modification and, consequently, to a retroactive processing–modification of the archived data products. This would trigger a set of reprocessing actions on the data products that have been extracted by subsequent processes, which are timely-sequentially related to the modified ones.

For example, prime investigators would like to activate a new calibration method on infrared spectra (interferometry), in order to increase the possibility of a right identification of chemical compounds in atmospheric volumes, resulting at validated data products of trace gas profiles. This, in turn, should inform the scientists of which scientific data products, which have been generated based upon the precedent ones, should be affected by an intended algorithm modification, and of which processes must be reactivated for the regeneration of the affected data products.

Furthermore, improving calibration process physics and providing radiometric corrections, presupposes the capture of knowledge expressing the conditions under which derived data products have been processed. Thus the description of derived data should be enhanced by the related algorithms and their parameters, which can also be considered as metadata. Typical queries addressing the derivation history concern with a certain algorithm version and the related data product version. Of course, the submitted queries would be related to the temporal aspects of data capture, from the satellite mission point of view (orbital data related access), and of data generation, from the scientific database system point of view (generation time related access).

1.2 The scenario of observing atmospheric phenomena

The observation of atmospheric phenomena - like creation and expansion of ozone hole, chemical reactions of trace gases contributing to the depletion of ozone, tracing of certain gases among different atmospheric layers, etc. - through a suitable visualization system, should allow references back to the original data, from which observed phenomena have been derived. Thus the verification and validation of observed phenomena goes through the whole processing chain, starting from very low data levels (interferograms). The announcement of the occurence of an atmospheric event will follow, after making sure that, what has been observed by the prime investigators, is a real fact and not an artifact. The need of keeping track of the generation process along all data levels arises at this point.

But doing research means that the prime investigator is not able to define, in advance, its whole mini-world, because there are some issues and relationships that should be extracted and provided by the system itself. Consequently, a partially unknown universe of discourse (UoD) has to be modelled. For example, in our case, prime investigators wish the system to instantiate the relationships among various observed phenomena, and between observed phenomena and trace gases. With similar modeling issues are also faced scientists from other scientific disciplines, like molecular biology [Fre91].

It's probably known that the discovery of atmospheric phenomena should not only be based on an interactive communication way with the scientific database system, but also on an alerting mechanism through which prime investigators are going to be informed. The extension of upper (in case of carbon dioxide) or lower (in case of ozone) limits on trace gas concentrations, should trigger an action of signalling attention that must be paid on the phenomenon. Passive behavior of the system (interactively reacted) cannot always guarantee that atmospheric phenomena will be really observed and, consequently, analysed.

1.3 The scenario of explaining scientific results and data

One of the most important issues of dealing with a scientific experiment, is the notion of metadata and its management for accessing and explanation purposes of scientific data. Metadata will cover a wide spectrum of information affecting scientific data products and scientific results. During the generation process, source data (orbital and instrument data) will be used, which has been captured on the satellite platform and delivered together with the associated measurements data. Orbital data will be used for the efficient access to the required data products, according to a multidimensional model of space and time. Instrument data will annotate derived data products. This source or housekeeping data are, e.g., orientation and position of instrument in space, instrument status, relevant temperatures, status of transmission quality, resolution, time of acquisition, etc. [Inc92].

We have seen in the previous section, that it's not sufficient to annotate scientific data with only source data. They must also be annotated with information concerning the algorithms with which are derived. Otherwise, scientific analysis of the derived data would be done under extremely difficult conditions, leading often to false assumptions. This kind of metadata concerning the derivation history must also be modelled and managed efficiently. Access to data products, according to specific processes and transaction-generation time, will constitute the first step towards the modification, for example, of the calibration process.

Moreover, annotating graphical representation of observed atmospheric phenomena, would enrich the semantics of observations, in terms of both source information and phenomenon explanation. An explanation of a scientific result or assumption should also be documented, creating a text database. This must be brought in conjuction with the facts (observed phenomena, derived data products, source data, etc.), from which a scientific conclusion or assumption has been implied.

Chapter 2

Design issues of the scientific database system

Designing a database system for manipulating scientific data, is not a trivial effort. There are many substantial differences from conventional database systems (relational, network, hierarchical) which have been designed for business-oriented applications. Based upon the system requirements and the scenarios of the scientific data handling given in the previous chapter, we will try to illustrate the major design issues, in order to take an approach of the scientific database schema architecture.

- **Scientific data types and structures**

  Starting with the data and their representation, two main features must be addressed: their complexity and the diversity of data types. It has been pointed out that multidimensionality is a dominant feature in scientific database systems [SW85]. Scientific data are not only measurements data, but also the related housekeeping data concerning the orbital data, instrument status, acquisition time, etc. It is the housekeeping data that invoke the complex-multidimensional structure of our scientific data (figure 2.1).

  Accessing the measurements data, will be done in terms of space and time. This assumes modeling facilities of complex data structures for multidimensional data.

  The most conventional systems support only simple data types for the attributes of the entities. Emerging technologies can provide a better modeling approach through the definition of more abstract and complex data types [FJP90]. This can be achieved only at the logical level, like extended relational models (Postgres, Straburst, Genesis, etc.), or also at the physical one, like object-oriented database systems [Kim90b, Kim90a, Heu92]. Other approaches are based on extensible database systems [CDRS86], providing a database toolkit supporting the definition of new data types and their operators, as well as of new storage structures.

  Focusing on the diversity of data types, we are faced with formatted (i.e., measurements and source data) and unformatted data (i.e., graphics, text) which must also

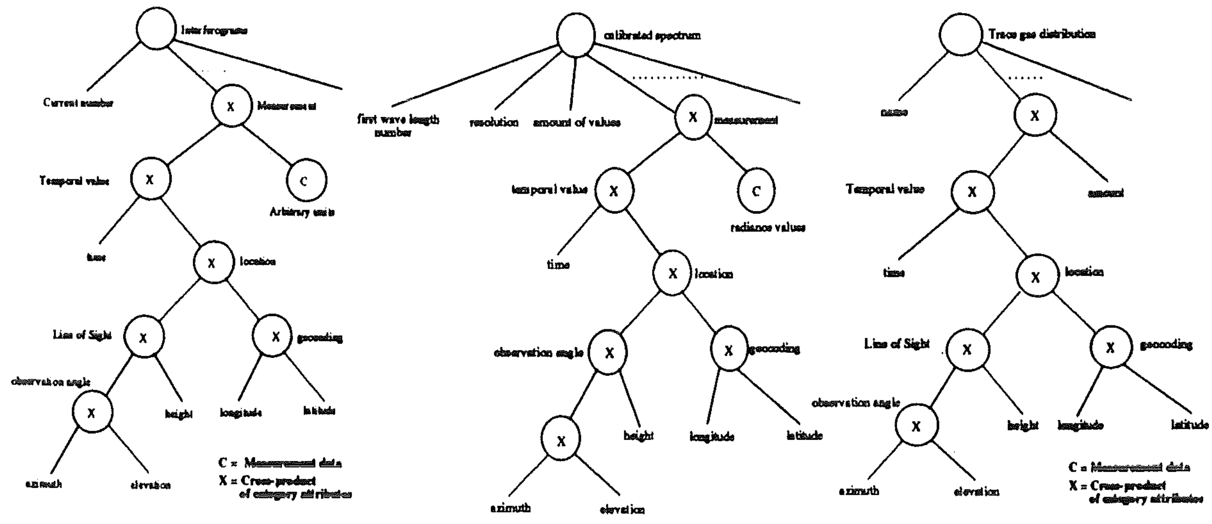


Figure 2.1: Complex-multidimensional structure of MIPAS data

be managed and accessed, in order to observe and explain atmospheric phenomena. They are all associated to each other. Graphics data will undergo frequent changes according to reprocessing requirements of lower level data. On the other side, text fragments will be generated for explanation and documentation purposes of scientific results and assumptions. Retrieval techniques could be added in order to improve retrieval quality.

Graphics, text and measurements data have a common feature; they are all of large or unbounded size and constitute the long-field data upon which their related structures have to be built [Loc88]. Text management can be dealt with established technologies. Text fragments can be considered as objects, and there are already object-oriented database systems, like O2 [BDK92], that can support both text and image data types. But accessing graphics data in terms of its contents, requires the bridging of gap between database and visualization systems [HS90, KASS93]. Prime investigators should not be forced to write their own programs for visualization of predefined files. They want to observe spatially and timely specified atmospheric volumes and/or phenomena. The system has to provide visualization facilities through locating and accessing the related datasets to be visualized, and subsequently, to use them as input to the suitable visualization package.

Additionally, no database system has been evolved with a clear technical identity of "graphics Database Management System" [Loc88]. The definition and modeling of a spatial database providing a 4-dimensional gridded dataset–a 3-dimensional space (longitude, latitude, altitude) and dimension of time–complemented with a cartographical model for the projections of the observed phenomena on earth, seems to counterbalance the deficiency resulting from the lack of supporting graphics data management. Established geographical information systems provide modeling facilities up to 2.5 dimensions [BF91]. Some research efforts are addressing the development of 4-dimensional geographical information systems [HLW91], but are concentrated on topological structures and operations, and not on dealing with scientific phenomena such as in atmospheric chemistry, or other scientific disciplines, like medicine and biology.

The spatial database model must be integrated within the scientific database system. It will be instantiated as a result of trace gas profiles generation (third level of generation process). Visualization should be executed by rendering specified projections of timely specified volume objects on the 3-dimensional gridded data space.

- **Active capabilities required**

  The generation process in operational mode consists of several steps that must be correctly sequenced. These processing steps constitute long-run activities, in most cases longer than a single transaction, which must be coordinated by workflow controls in their asynchronous activation and execution. As a basis for the workflow control, an event- or data-triggered invocation of actions must be used, a mechanism provided by an Active Database Management System (ADBMS) [MD89, Ber92]. The same mechanism must also be used to alert scientists when certain patterns of events or data are detected.

  The most conventional database systems don't provide such a mechanism, and are considered to be passive. They react only on submitted queries. The development of ADBMS was motivated by the need to have timely and customizable response to critical events and situations. Generally speaking, triggers are event-condition-action triples. They are operations that are automatically executed, whenever a specific event occurs and a condition over a database state or state change holds, due to the definition in [vdVK93].

  The required usage of active capabilities of the scientific database system, will be the user notification, the creation of an abstraction level for organizing related actions on the occurence of an external or internal event, and the integrity enforcement. With the first one, scientists will be alerted on the occurence of events which need attention from a scientific point of view, such as the decreasing ozone concentration in a certain atmospheric volume. With the second one, the workflow control will be specified and, furthermore, the activation of a series of actions for the needs of reprocessing requirements. With the third one, constraint evaluation can be triggered when specified events or situations are detected, such as the quick-look facilities of low levels data. In contrast, conventional database systems evaluate integrity constraints immediately on the event of updating the database, during or at the end of a transaction.

- **Managing TeraBytes of data**

  The execution of a scientific experiment, collecting and generating vast amounts of data, must be supported by a mass storage system providing not only secondary, but also tertiary storage level. The generated data products have to migrate to the mass storage system, after a certain time period, in which they are needed by the sequenced generation processing steps, and therefore, will be temporarily stored and managed, locally, by the operational database system. A file storage and management system of the mass storage system manages the stored files, with a parameterized migration policy between the secondary and tertiary storage levels [Mil88, omsst90].

  The mass storage system cannot be considered only as a supplementary archive medium of the system. Accessing of selected stored files, will be caused by the needs of reprocessing specified data products. Furthermore, keeping track of the whole generation process (going back from observed phenomena to the lower data levels), presupposes the capture of relationships among the various data products, which cannot be expressed using only a file system. This kind of semantics should be captured and expressed one level above the file system in usage, which can be generally called data system level [Sho93].

  This semantically enriched access on TeraBytes cannot be provided by conventional database management systems, because they cannot access directly tertiary storage [Sho93]. In addition, there are some software packages for accessing files from mass storage with specialized software and operations [BFGR93] on data files, but they don't deal with semantics, they only improve the file system design and performance (we remain at the physical level of the scientific database system). Scientific efforts are underway, addressing the integration of database management systems with file systems for mass storage [SD91, SFD93].

  On the other side, from a scientific applications point of view, direct access to files from mass storage should also be supported by the underlying system. Using the database system as file searching and location mechanism, will enhance the flexibility and performance of the scientific database system. Thus the scientific applications will not exclusively access files through the database system, but will also access, directly, the layer of the file system (figure 2.2).

- **Operational and scientific data management**

  Thinking of a scientific database system, in terms of its operational and scientific aspects, two main perspectives should be kept in mind. The perspective of monitoring the scientific experiment and data generation process (operational point of view), and the perspective of getting out the information the prime investigators need (scientific point of view). Each perspective has some characteristic features concerning the data types to be handled, the creation and access policy, their functionality, etc.

  Operational data types are, at most, time series data, raw data blocks, event data, which are self keying. The most recently used data are going to be stored locally

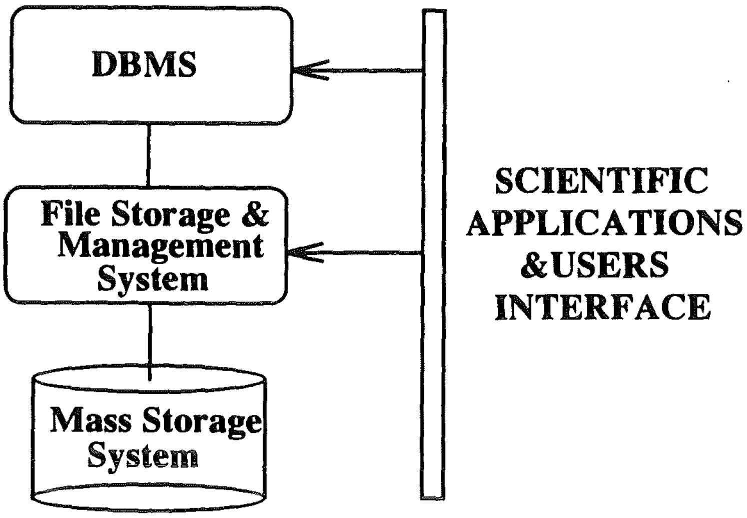

DBMS
File Storage & Management System
Mass Storage System
SCIENTIFIC APPLICATIONS &USERS INTERFACE

Figure 2.2: Layered architecture of a scientific database

(operational database). Old data will be automatically discarded as new data is written. They will migrate to the mass storage system. Operational data will be annotated by metadata (source data) which are handled only at the physical level, which will be interfaced with the algorithms of the generation processes.

As stated above, the various processing steps must be sequenced by an event-triggering mechanism of the operational database. This capability should be extended with a remote procedure call mechanism (message routing facilities), for the execution of the generation processing steps in a distributed environment, consisting of clustered powerful workstations and/or supercomputer environment. Moreover, the operational database can be thought of as multiple database. A reference spectral database will be used by the processing step of the trace gas profiles generation [Wet93, Fia84]. Other databases containing radiosonde or meteorological data must also be used. Each of these datasets can be considered as a separate database needed when data products are being generated or analysed.

Arrangement of the generated data products into files, must be achieved not only historically according to when they have been added to a file (only sequencial access provided), but also using indexing techniques for a more efficient retrieval technique required–case of inversion algorithm for the identification of chemical compounds (generation of trace gas profiles) [Wet93]. File structures must be easily changeable and extensible, so as to accomodate evolving data structures.

From the scientific point of view, information should be extracted out of the system, with respect to the operational data being generated and to the archived data on the mass storage system. Accessing the stored data, in order to get out the information needed, has to be done on a semantically enriched level, dealing with multidimensional structures (search in space and time), as well as different data types (graphics, text), and their interrelationships with derived scientific data. To this extent, the extraction of information goes through the creation and management of metadata.

In addition, a dynamical schema modeling facility should enable the modeling of a partially unknown Universe of Discourse (UoD), a fact playing an important role in scientific experiments, whereby the scientists' mini-world cannot be totally defined a priori. In case of atmospheric research, the relation of certain trace gases to a particular phenomenon, or among different phenomena, cannot be a priori defined.

Chapter 3

The scientific database schema architecture

The design issues can be summarized in a 3-dimensional space (figure 3.1), for the representation of what kind of scientific database system is going to be designed, according to the three dimensions of interpretation, intended analysis and source.

The dimension *level of interpretation* indicates the various levels of metadata needed as an ancillary information for the processing, analysis and explanation of scientific data. In our case, metadata ranges from calibration data, derivation and validation data, up to interpretation data, as explanation or annotation data, and scientific reports.

The dimension of *intended analysis* refers to the access policy to a shared pool of data for scientific analysis purposes, under the assumption that all scientific data are subject to further analysis. In our case, analysis concerns with time series (calibration physics development), as well as with multidimensional objects, represented as discrete sampling of 4-dimensional functions, e.g., (x=longitude, y=latitude, z=altitude, t=time).

The third dimension *source* refers to the integration aspects of the different components of data, which will provide a certain functionality to the whole system. In our case, integration of various heterogeneous components, such as the operational database, the Research and Development Database, the mass storage system, graphics and text data management systems, will be provided at a higher level, enabling the handling of all components from a common interface to the outside world. It's necessitated by the fact that a high modularity must be achieved, and the fact of efficient data handling according to different functionalities which cannot be provided by a single DBMS.

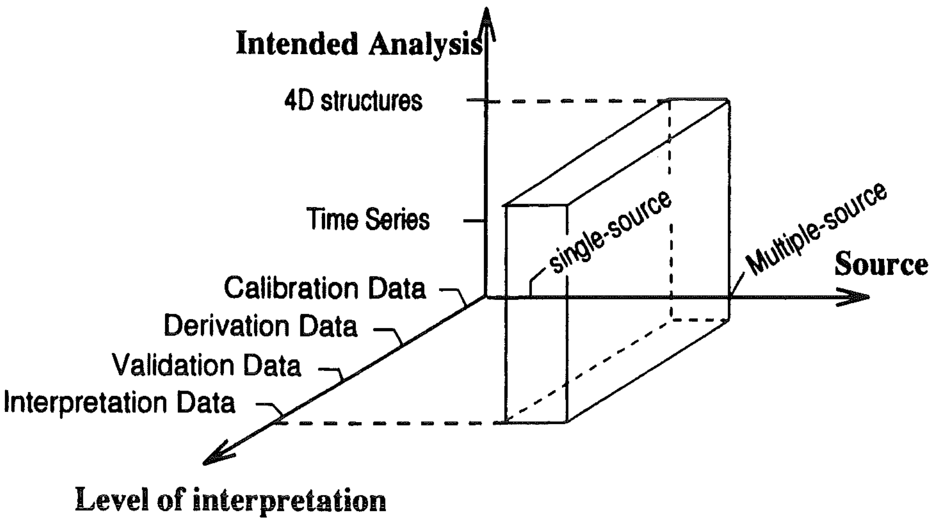


Figure 3.1: The scientific database system volume

3.1 The federated database reference schema architecture

The best-case scenarios for the management of scientific data have rarely been anything more than the archiving of a computer-compatible magnetic tape, with an analog catalog which briefly describes where data are going to be found due to generation source, time and location of acquisition [CSea89]. If a prime investigator wished to locate and browse the data, he had to know where the data set has been archived, the specific data of interest, and who to contact to order the data. This kind of file oriented accessing degrades the functionality of a scientific data management system, and don't support a knowledge or information access to the data, employing artificial intelligence technologies in a distributed heterogeneous environment. Within this scope, an extended federated database schema design approach will be presented for the scientific database system.

Definition. A *multidatabase* is a distributed system that includes a global component to access globally shared information, and multiple local autonomous components that manage only information at their sites [BH90, BHP92]. The distinctions are in the structure of the global component, and how it interacts with the local components.

A taxonomy of global information-sharing solutions according to the definition given above, has been specified with respect to how tightly the global system integrates the local databases. Following this specification given in [BH90, BHP92], one can distinguish between *distributed databases*, *global schema multidatabases*, *federated databases*, *multidatabase language systems*, *homogeneous multidatabase language systems*, and *interoperable systems.*

Definition. A *federated database system* is a collection of cooperating but autonomous component database systems [SL90]. The components are integrated to various degrees. A component can continue its local operations and at the same time participate in a federation. There is no single global schema, but a federated one composed of the components export schemas, which provide the description of the information to be shared with the global system.

A key characteristic of a federation is the cooperation among independent systems. These components may be characterized among three orthogonal dimensions: distribution, heterogeneity, and autonomy. The types of heterogeneity can be divided due to the differences in database management systems, and due to differences in the semantics of data. Thus heterogeneity also occurs when there is a disagreement about the meaning, interpretation, or intended use of the same or related data [SL90].

Definition. An *extended federated architecture* allows the access to data from systems other than database management systems [SL90]. The components may be of different types of data management, such as file server, a database machine, a distributed DBMS, etc.

Building up a federated database system, a five-level schema architecture (figure 3.2) has been specified [SL90, She88], in order to provide an adequate architecture for supporting the three dimensions of a federated database system–distribution, heterogeneity, and autonomy. It will be used as a reference architecture. The ANSI/SPARC three-level schema architecture is only adequate for describing the architecture of a centralized DBMS.

According to the five-level reference schema architecture, the external, federated, export, component, and local schemas are included. The *local schema* is the conceptual schema of a component DBMS, expressed in the native data model of the component DBMS. The *component schema* is the data model, called the *canonical* or *common data model* of the federated database system. It is derived by translating the local schemas, and describes the local schemas using a single representation, enhanced with semantics that are missing in a local schema. The component schema facilitates negotiation and integration tasks performed by the system.

The *export schema* represents a subset of a component schema that is available to the federation. The *federated schema* is the integration of multiple export schemas and includes the information on data distribution. There may be multiple federated schemas, one for each class of federation users. The *external schema* defines a schema for a user and/or application or a class of users/applications.

Transforming, filtering, and constructing processors are the mechanisms which underlie the mappings among the various schemas. The *transforming processors* are used in order

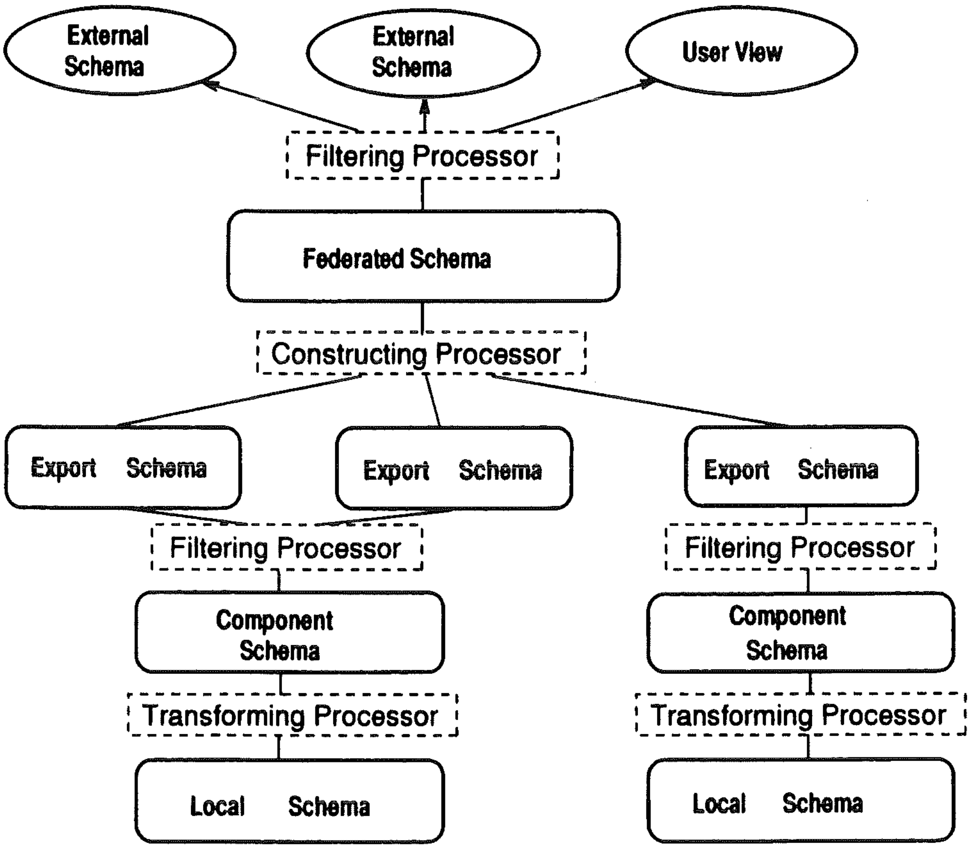


Figure 3.2: Five-level Reference Schema Architecture of a Federated Database System

to transform commands on a component schema into commands on the corresponding local schema. The *filtering processor* can be used to provide the access control by limiting the set of allowable operations on the component and federated schema, as they can be defined by the export and external schemas respectively. The *constructing processor* transforms commands on the federated schema into the commands on one or more export schemas, supporting the distribution feature of the federation. There are also other kind of processors (e.g., accessing processors) supporting the participation of components without local schema (database management component).

Processors and schemas are the basic elements, which can be combined in order to create various federated database system architectures, with missing or additional elements, according to the characteristics of the database system to be designed. Considering the design issues of the scientific database system, as stated above (chapter 2), an *extended federated schema* architecture will be presented, underlying the scientific database schema architecture.

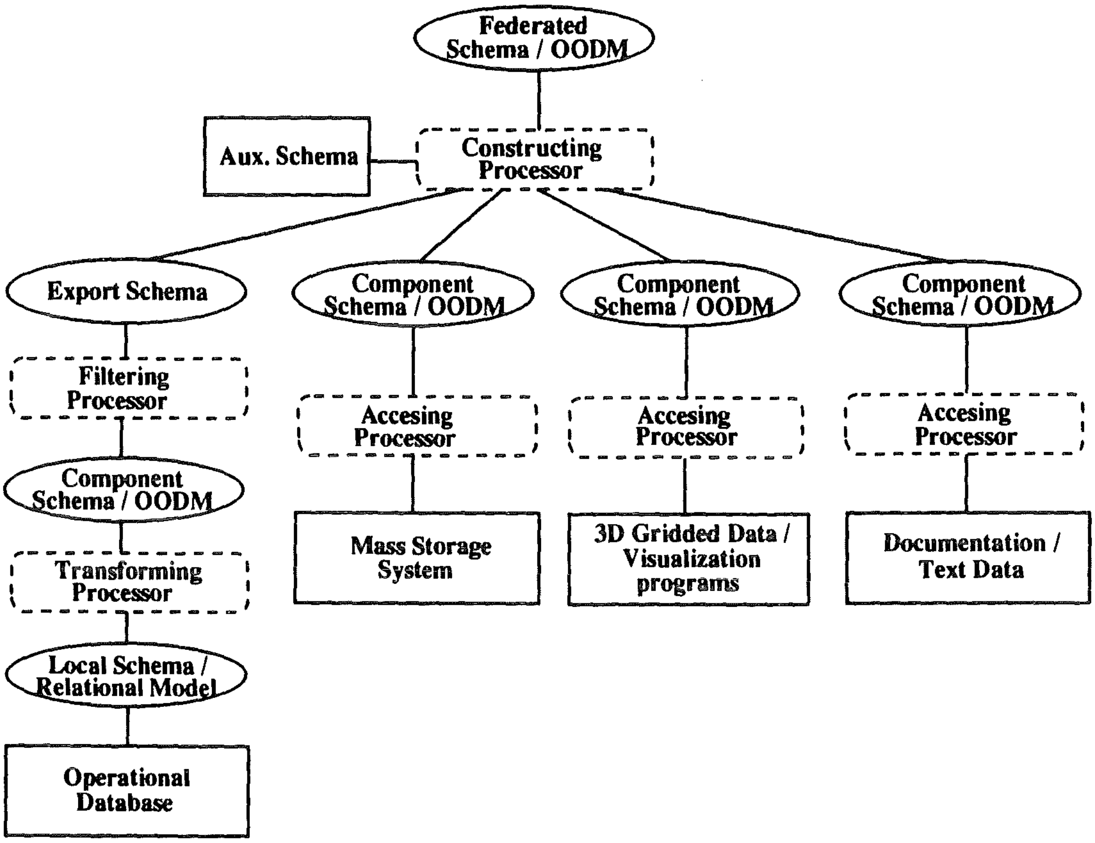


Figure 3.3: The extended federated architecture of the scientific database schema

3.2 The extended federated schema architecture of the scientific database system

As illustrated in figure 3.3, the scientific database schema architecture has been specified, providing the integration facilities required by the system. The participating components are the atypical components of the mass storage system with its file and storage management system, an authoring and documentation system, the visualization management system, as well as a component DBMS for the operational database.

A *local schema* can only be considered in the case of the component DBMS for the operational database. The local schema of the operational database will be based upon the relational model, as regarded to be more suitable for handling with scientific data during the generation phase (section 2). The atypical components don't provide any local schema, since they are not typical DBMS, but different types of data management systems.

The *component schemas* are considered as the common data model of the federation.

Taking in account the divergency among the various components, and the need to model also data from the atypical components, the common data model to be chosen must provide more enriched semantics than the underlying components. It refers to semantic models [PM88, HK87]-they complement work on knowledge representation (in artificial intelligence)-with its primary concepts of explicit representation of objects, attributes of and relationships among objects, type constructors for building complex types (aggregation, grouping, association), and IS-A relationships. Some well-known semantic models have been described in the literature, like the Extended Entity-Relationship model (EER), the Functional Data Model (FDM), the Semantic Association Model (SAM*), the Binary Model, etc.

A semantic model also complements the object-oriented paradigm of programming languages. An object-oriented data model (OODM) could provide the common data model over which the component schemas are going to be defined. The reason is that it is based on the most semantic model primary concepts [Nie89, Kim89, McL91, LAC$^+$93], and could support a unique platform of modeling also data from atypical components (no local schema) by considering them being generally objects [CHT86, MRT91, ISea93]. However, there are still some differences between a semantic model and an object-oriented one, but it is out of the scope of this paper.

The *export schema* has the purpose to facilitate access control to the operational database by limiting the set of allowable operations which can be submitted. It is associated with a *filtering processor* responsible for the implementation of the access control aspects. Therefore, the autonomy feature of the operational database component is increased.

The *federated schema* will also be based on the same object-oriented data model, like the component schemas. It will take the form of schema integration, and therefore, the federated scientific database is considered to be *tightly coupled* (as opposed to *loosely coupled* federations). A uniform object space is going to be provided (everything is regarded to be an object). The mass storage system files, the visualized 2d/3d datasets and/or atmospheric phenomena, the scientific reports, the scientific data itself, explanation or annotation data, are all objects. At this level, not only intra-object relationships are going to be defined, but also **inter-object relationships**, following the concepts for multimedia databases as stated in [Mas87, Loc88].

A *constructing processor* from the federated schema to the component schemas, will partition and/or replicate an operation submitted on objects, modelled in the federated schema, into operations that are accepted by the corresponding components. It will also merge objects returned by several components, taking in account synchronisation aspects for the purpose of a contemporary display of various data types, e.g., explanation of a visualized phenomenon through a scientific report, or description of derived data by annotation data. An *auxiliary schema* will provide information not available from the participating components, like data residing in other external information systems, and are related to those managed by the scientific database system under consideration.

The *accessing processor* to the mass storage system will execute access procedures against a stored file, probably, through an access library system (HDF or netCDF data access libraries). This will cause the retrieval of a certain file or a subset of the data included. Some issues to be addressed by accessing processors include local concurrency control, commitment, backup, and recovery. They have to be provided by the file and storage management system (UniTree, EpochServ, etc.) of the mass storage system. Similarly, the *accessing processor* to the documentation system will provide access to several scientific reports, related to observed phenomena and/or other scientific assumptions based on the scientific data generation.

The *accessing processor* to the 3-dimensional gridded spatial data model, facilitates the access to the requested, in order to be visualized, 2-dimensional (e.g., user-defined slices perpendicular to one of the three axes) or 3-dimensional data subsets (e.g., user-defined volume areas) of the volume dataset. The information on how to visualize a dataset will also be stored in the database with the dataset itself, without bringing the scientists in a situation of writing visualization programs. It will be stored in form of programs or scripts instantiated within a certain visualization package. The specified dataset is going to be ingested and converted to the internal format used by the visualization package [KASS93].

The *transforming processor* will translate the commands, from the source language–data manipulation language of the component schema (object-oriented data model as common data model)–to SQL commands of the operational database (relational model) component. It provides a data model transparency, hiding the differences in query languages and data formats. A mapping schema is also used by the transforming processor, in order to couple with both data schemata, object-oriented and relational ones [Kim93].

3.3 Allocating processors and schemas to the scientific database system components

A client-server architecture is considered to be the backbone of the scientific database system design, consisting of real-time clients–for the operational needs of the system–and regular clients–for scientific data evaluation and information access purposes. The server can be viewed as a federation of heterogeneous systems–*federated server*–providing a uniform interface to the outside world, but also autonomous access facilities. As it has been depicted in figure 3.4, the federated server consists of five main components: The *Operational Database System*, an *Object-Oriented Database System*, referred also as *Research and Development* (R&D) database system, the *File Management System* of the mass storage system, the *Visualization Management System*, and an *Authoring System.*

Local autonomy and stand-alone access are extremely important in case of adding a new

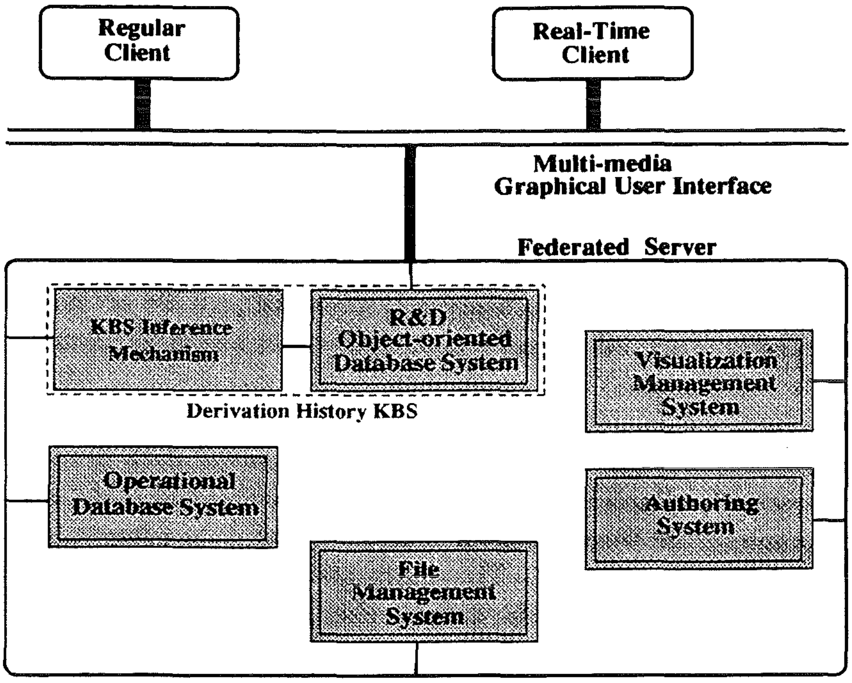
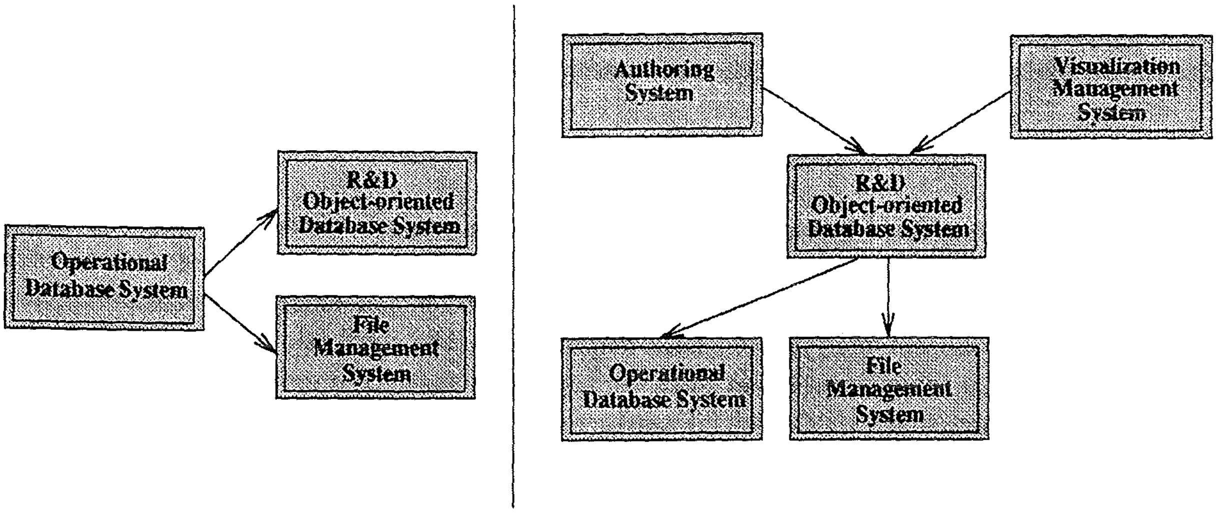

Regular Client
Real-Time Client
Multi-media Graphical User Interface
Federated Server
KBS Inference Mechanism
R&D Object-oriented Database System
Derivation History KBS
Visualization Management System
Operational Database System
Authoring System
File Management System

Figure 3.4: A federated client-server architecture for the scientific database system

component–a knowledge base system or an expert system–to the cluster. Each component will undertake a certain functionality according to the main requirements and design issues considered so far. The *Operational Database System* deals with the generation process, monitoring the creation of the scientific data products. It should provide the features as stated in section 2, as well as the ability to access other databases, such as reference spectra or radiosonde databases, needed during the generation phase in operational mode. *It is the site where the local schema (relational model) with its transforming processor are going to be allocated.*

All schemas and processors, starting with the component schema and going towards the federated schema, are going to be allocated at the *Object-Oriented Database System* (R&D). It will provide an object space for all objects affecting the scientific experiment (multidimensional structures–space and time, graphics objects, text objects, etc., and their interrelations), acting as manager of metadata. It will use as a long-field data server the mass storage system through its *file management system* for the data that has been generated, and/or the *Operational Database System* for the data being generated. It's the site where an integrated view of the scientific database system is going to be provided by the common object space. A closely related approach has been taken by [CKTL93, Loc88, Mas87, CHT86], considering the design of a Multi-Media Database System, and to some extent by [SFD93], considering the design of a scientific database for the needs of global climate research.


Operational Database System
R&D Object-oriented Database System
File Management System
Authoring System
Visualization Management System
R&D Object-oriented Database System
Operational Database System
File Management System

The *Visualization Management System* complements the scientific database system, trying to bridge the gap between data management and visualization. It's the site where the 3-dimensional gridded spatial data is going to be modelled and managed. From this volume data, datasets of two- or three-dimensions will be accessed–the site where the associated accessing processor is allocated too–and, consequently, transformed into display files rendered by the appropriate visualization programs. Thus the possibility of addressing observed phenomena by time and location coordinates[1] is increased and enables the capture of the relationship between visualized and derived data.

The accessing processor of the documentation–text data is going to be allocated to the *Authoring System*, a hypertext based system providing an information web over the text fragments to be managed and interrelated [FCF91].

The accessing processor of the *Mass Storage System* will deliver files or subsets of files, stored and managed by the corresponding file management system (e.g., UniTree, EpochServ), for on-line and near-line storage levels. They will be addressed by both the *Research and Development Database*–information accessing–and the *Operational Database System*–generation and reprocessing facilities. Each file is regarded to be an object, and the *File Management System* component–the site where this accessing processor is allocated–is considered to be the physical storage extension of the *Object-Oriented Database System.*

Allocation of processors to different sites implies the existence of *communication processors*, which must also be placed on each component. The federated server can also be viewed in two perspectives according to its interoperability issues, the perspective of gen-

[1] and not by specifying files to be visualized

erating scientific data products (figure 6a), and the perspective of accessing them (figure 6b). A Graphical User Interface (GUI) is going to be built upon the Object-Oriented Database System, providing a multi-media user interface.

The derivation history Knowledge Base System

The federated server has been extended by the Knowledge Base System (figure 3.4), which consists of the knowledge about the derivation history of scientific data, and an inference mechanism providing the consequences on the data model that changes in the derivation processing environment may have. This implies that the derivation history knowledge is based upon the coupling of the data and process models, incorporating them in a common knowledge/data model which captures both data and process model semantics. The derivation process model is considered to be the behavior model of the scientific information system, in terms of its scientific data creation perspective. Various behavior modeling approaches have been stated in the past, aiming at the explicit specification of information system dynamics [Lau88, MBJK90, CKO92] with the data model expressing the information system statics.

A unifying approach to modeling data and process model, requires a semantic model as already described in section 3.2. The concepts of generalization, classification, aggregation, etc., are regarded to be essential for modeling of processes and their interrelationships, as well as the relationships with the derived data [PK88, Mar90, Hsi93]. Therefore, processes and derived data are viewed as being objects, and subsequently, the object-oriented database system component, with its associated object-oriented data model, seems to be a suitable platform of modeling both models and its unification. However, types of relationships, like temporal relations where specific object-types are related by synchronous or asynchronous characteristics constructing a higher level object, or other artificial intelligence concepts, like heuristics, uncertainty, constraints, must also be incorporated in the semantic model, together with the object-oriented concepts found therein.

Conclusion

The scientific database schema has been presented in this paper, following the concept of an extended federated schema design. The schema design approach has been driven by the integration aspects of autonomous and heterogeneous system components allocated to a federated server providing the operational and research/development database of the scientific information system.

The autonomous operation of the existing components enables a high modularity of the system design and an efficiency of dealing with great amounts of data, due to performance degradation expected, if only a single DBMS is considered. On the other hand, providing a unique interface to the whole system requires integration facilities which can be achieved by the creation of an object space representing datasets and metadata as objects, capturing also their interrelationships. The latter is the core area of bringing datasets and metadata together, in various types of representation (images, text, graphics, etc.).

Furthermore, we argue that there is no optimal solution to the problem of choosing a certain database management system. The choise is strongly related to the data management policy which must be examined, in order to develop a system which is optimized either towards read-actions or towards write-actions. According to the differences in data handling policies between the operational (write-actions optimization) and the research/development database (read-actions optimization), two separate database systems must be integrated and communicate through a suitable gateway. One based on the relational model approach for the operational needs, and the other one based on the object-oriented model approach for information accessing needs. The object-oriented model can be viewed as an extension of the relational one, but providing also encapsulation and inheritance properties considered to be essential for modeling the object space, and providing the modeling mechanism of the derivation history knowledge base.

An increased effort due to design and maintanance of two database schemas can be minimized by the fact that the relational model can be subsumed in the object-oriented model, and that no updates of the operational database schema are allowed. Thus the object-oriented model can be mapped onto the relational one by a well-founded mapping mechanism. Autonomism has also been strenghtened by the fact of independent accessing on the mass storage system, without intervention of the operational database, improving the performance values of the system.

Bibliography

[BDK92] Francois Bancilhon, Claude Delobel, and Paris Kanellakis. *Building an Object Oriented Database System: The Story of O2*, chapter 5, pages 41–52. Morgan Kaufmann Publishers, 1992.

[Ber92] Mikael Berndtsson. Active Databases: The next generation databases. Technical Report TR-92-02-001, 1992. University of SKOVDE - Department of Computer Science.

[BF91] R. Bill and D. Fritsch. *Grundlagen der Geo-Informationssysteme*, volume 1. Wichmann, 1991. Hardware, Software und Daten.

[BFGR93] St. Brown, M. Folk, Gr. Goucher, and Russ Rew. Software for Portable Scientific Data Management. *Computers in Physics*, 7(3):304–308, May 1993.

[BH90] M. W. Bright and A.R. Hurson. Multidatabase Systems: An Advanced Concept in Handling Distributed Data. In Marshall C. Yovits, editor, *Advances in Computers*, volume 32. Academic Press, 1990.

[BHP92] M. W. Bright, A. R. Hurson, and S. H. Pakzad. A Taxonomy and Current Issues in Multidatabase Systems. *IEEE Computer*, pages 50–59, March 1992.

[CDRS86] Michael J. Carey, David J. DeWitt, Joel E. Richardson, and Eugene J. Shekita. Object and File Management in the EXODUS Extensible Database System. In *Proc. of the 12th Inter. Conf. on Very Large Data Bases*, pages 91–100, Kyoto, Japan, August 1986.

[CHT86] S. Christodoulakis, F. Ho, and M. Theodoridou. The Multimedia Object Presentation Manager of MINOS: A Symmetric Approach. In *Inter. Conference on Management of Data*, pages 295–310, 1986.

[CKO92] Bill Curtis, Marc Kellner, and Jim Over. Process Modeling. *Comm. of ACM*, 35(9), September 1992.

[CKTL93] S. Chakravarthy, V. Krishnaprasad, Z. Tamizuddin, and F. Lambay. A Federated Multi-media DBMS for Medical Research: Architecture and Functionality. Technical Report UF-CIS-TR-93-006, University of Florida, Department of Computer and Information Sciences, January 1993.

[CSea89] W. Campbell, N. Short, and et al. Adding Intelligence to Scientific Data Management. *Computers in Physics*, 3(3):26–32, May/June 1989.

[FCF91] Edward A. Fox, Qi Fan Chen, and Robert France. Integrating Search and Retrieval with Hypertext. In Emily Berk and Joseph Devlin, editors, *Hypertext/Hypermedia Handbook*, Software Engineering Series, chapter 21, pages 329–355. McGraw-Hill, 1991.

[Fia84] Jaroslav Fiala. Spectral Databases for Chemical Compound Identification. *Computer Physics Communications*, 33:85–92, 1984. North-Holland Amsterdam.

[FJP90] J. C. French, A. K. Jones, and J. L. Pfaltz. Scientific Database Management. Technical Report 90–21, Univ. of Virginia, August 1990. Report of the International NSF Workshop on Scientific Database Management.

[Fre91] Karen S. Frenkel. The Human Genome Project and Informatics. *Comm. of ACM*, 34(11):41–51, November 1991.

[Heu92] Adreas Heuer. *Objektorientierte Datenbanken, Konzepte, Modelle, Systeme.* Addison Wesley, 1992.

[HGW93] N. I. Hachem, M. A. Gennert, and M. O. Ward. An Overview of the Gaea Project. *Bulletin of the Technical Committee on Data Engineering*, 6(1):29–32, 1993.

[HK87] Richard Hull and Roger King. Semantic Database Modeling: Survey, Applications, and Research Issues. *ACM Computing Surveys*, 19(3):201–260, September 1987.

[HLW91] N. W. J. Hazelton, F. J. Leahy, and I. P. Williamson. On the Design of Temporally–Referenced, 3–D Geographical Information Systems: Development of Four–Dimensional GIS. In *GIS/LIS*, 1991.

[HS90] W. Hibbard and D. Santek. Visualizing Large Data Sets in The Earth Sciences. In Gr. M. Nielson and Br. Shriver, editors, *Visualization in scientific computing.* IEEE Computer Society Press Tutorial, 1990.

[Hsi93] Donovan Hsieh. A Logic to Unify Semantic-network Knowledge Systems with Object-oriented Database Models. *Journal of Object-Oriented programming*, 6(2):55–67, May 1993.

[Inc92] BOMEM Inc. On Ground Signal Processing Report. Final report, phase b, MBB Deutsche Aerospace, 1992.

[ISea93] Hiroshi Ishikawa, Fumio Suzuki, and Fumihiko Kozakura et al. The Model, Language, and Implementation of an Object-Oriented Multimedia Knowledge Base Management System. *ACM Transactions on Database Systems*, 18(1):1–50, March 1993.

[Kap93] Epaminondas Kapetanios. PROMISE: A Preliminary Study for MIPAS Scientific Experiment. Technical Report KfK 5209, Nuclear Research Centre, Karlsruhe/Germany, August 1993.

[KASS93] Peter Kochevar, Zahid Ahmed, J. Shade, and C. Sharp. A Simple Visualization Management System: Bridging the Gap Between Visualization and Data Management. Technical report, University of California, Berkeley, April 1993. Sequoia 2000.

[Kim89] Won Kim. Oject Oriented Approach to managing Statistical and Scientific Databases. In Z. Michalewicz, editor, *Fifth International Conference, V SS-DBMS, Proceedings*, Lec. Notes in Computer Science, pages 1–13. Springer Verlag, 1989.

[Kim90a] Won Kim. Architectural Issues in Object-Oriented Databases. *JOOP*, pages 29–38, march–april 1990.

[Kim90b] Won Kim. Object-Oriented Databases: Definition and Research Directions. *IEEE Transactions on Knowledge and Data Engineering*, 2(3):327–341, September 1990.

[Kim93] Won Kim. Object-Oriented Database Systems: Promises, Reality, and Future. In *Proc. of the 19th VLDB Conference*, pages 676–687, Dublin, Ireland, August 1993.

[LAC$^+$93] M. Loomis, T. Atwood, R. Catell, J. Duhl, G. Ferran, and D. Wade. The ODMG Object Model. *JOOP*, 6(3):64–69, June 1993.

[Lau88] Georg Lausen. Modeling and Analysis of the Behavior of Information Systems. *IEEE Transactions on Software Engineering*, 14(11):1610–1620, November 1988.

[Loc88] Peter C. Lockemann. Multimedia Databases: Paradigm, Architecture, Survey and Issues. Interner bericht 1588, Univ. of Karlsruhe, September 1988.

[Mar90] Victor M. Markowitz. Representing Processes in the Extended Entity-Relationship Model. In *Inter. Conf. on Data Engineering*, IEEE, pages 103–110, Los Angeles, California, 1990.

[Mas87] Yoshifumi Masunaga. Multimedia Databases: A Formal Framework. In *Proc. IEEE Comp. Soc. Office Automation Symposium*, pages 36–45. IEEE Comp. Soc. Press, 1987.

[MBJK90] J. Mylopoulos, A. Borgida, M. Jarke, and M. Koubarakis. Telos: Representing Knowledge about Information Systems. *ACM Transactions on Information Systems*, 8(4):325–362, October 1990.

[McL91] Dennis McLeod. A Perspective on Object-Oriented and Semantic Database Models and Systems. In Rajiv Gupta and Ellis Horowitz, editors, *Object-Oriented Databases with Applications to CASE, Networks, and VLSI CAD*, Prentice Hall Series in Data and Knowledge Base Systems, chapter 2, pages 12–25. Prentice Hall, Englewood Cliffs, 1991.

[MD89] D. R. McCarthy and U. Dayal. The Architecture of an active DBMS. In *Proc. of ACM SIGMOD Inter. Conf. on Management of Data*, pages 215–224, Portland, Oregon, 1989.

[Mil88] Stephen W. Miller. A Reference Model for Mass Storage Systems. In Marshall C. Yovits, editor, *Advances in Computers*, volume 27. Academic Press, 1988.

[MRT91] Carlo Meghini, Fausto Rabitti, and Costantino Thanos. Conceptual Modeling of Multimedia Documents. *IEEE Computer*, pages 23–29, October 1991.

[Nie89] Oscar Nierstrasz. A Survey of Object-Oriented Concepts. In Won Kim and Frederick H. Lochovsky, editors, *Object-Oriented Concepts, Databases, and Applications*, chapter 1, pages 3–21. Addison-Wesley Publishing Company, 1989.

[omsst90] IEEE Technical Committee on mass storage systems and technology. Mass Storage System Reference Model. Technical report, IEEE, May 1990. Version 4.

[PK88] Walter D. Potter and Larry Kerschberg. A Unified Approach to Modeling Knowledge and Data. In R. A. Meersman and A. C. Sernadas, editors, *Data and Knowledge*, pages 265–291. Elsevier Science Publishers B. V. (North Holland), 1988.

[PM88] Joan Peckham and Fred Maryanski. Semantic Data Models. *ACM Computing Surveys*, 20(3):153–189, September 1988.

[SD91] M. Stonebraker and J. Dozier. Large Capacity Object Servers to Support Global Change Research. Technical report, University of California, Berkeley, September 1991.

[SFD93] M. Stonebraker, James Frew, and Jeff Dozier. The Sequoia 2000 Architecture and Implementation Strategy. Technical Report CA 94720, University of California, Berkeley, 1993.

[She88] Amit P. Sheth. Building Federated Database Systems. In *Distr. Processing Techn. Committee, Newsletter*, pages 50–58, Chicago, 1988.

[Sho93] Arie Shoshani. A Layered Approach to Scientific Data Management at Lawrence berkeley Laboratory. *Bulletin of the Technical Committee on Data Engineering*, 16(1):4–8, 1993.

[SL90] Amit P. Sheth and James A. Larson. Federated Database Systems for Managing Distributed, Heterogeneous, and Autonomous Databases. *ACM Computing Surveys*, 22(3):183-236, September 1990.

[SW85] A. Shoshani and Harry K. T. Wong. Statistical and Scientific Database Issues. *ieeese*, SE–11(10):1040–1047, 1985.

[vdVK93] M. H. van der Voort and M. L. Kersten. Facets of Database Triggers. Technical Report P.O. Box 4079, CWI, 1009 AB Amsterdam, The Netherlands, April 1993.

[Wet93] Gerald Wetzel. Eignung der Iinfrarotspektroskopie zur Fernerkundung troposphaerischer Spurengase. Technical Report KfK 5183, Institute of meteorology and climate research, Kernforschungszentrum Karlsruhe, June 1993.