Development of Databases for the Systems for Acquisition, Processing And Distribution of Satellite Data

E.A.Loupian
Russian Space Science Institute,
Moscow, Russia
evgeny@d902.iki.rssi.ru

A.A.Mazurov
Russian Space Science Institute,
Moscow, Russia
alexei@d902.iki.rssi.ru

R.R.Nazirov
Moscow, Russia
rnazirov@iki.rssi.ru

A.A.Proshin
Moscow, Russia
andry@d902.iki.rssi.ru

E.V.Flitman
Moscow, Russia
flitman@d902.iki.rssi.ru

Abstract

The paper presents the technology for creation of satellite data archives (databases) with remote access over the Internet. The special attention is devoted to storage of data files in the database. For this purpose the FDB (File Data Base) technology has been developed to allow operating with files as database objects.

1. Introduction

Satellite remote sensing data have been widely used during last decades for various environmental monitoring tasks and natural resources observation. One of key elements of any remote sensing system is the Information System for Acquisition, Processing and Distribution of Data (IS APDD), which should accomplish the following main tasks:

- acquisition (registration) of data downlinked from satellites
- data preprocessing
- data archives management
- performing specialized (thematic) data processing
- providing delivery of data to end-users.

Development of global computer networks and the Internet provides new possibilities for the development of IS APDD systems. Global networking allows for interconnection of different elements of information systems and organization of data delivery to the end-users based on new principles. This has led to development of new generation of IS APDD, that use Internet technology for all structural elements [1-4].

2. Satellite Data Archiving System (SDAS)

One of most important component of IS APDD undoubtedly is the Satellite Data Archiving System (SDAS). The present paper covers the problems of building satellite data archives, organization of their automatic feeding, and providing operational access to data over the Internet.

Any particular satellite system has certain specifics in different data types, data transmission rate, data volumes, and requirements in time of delivery to end-users. Naturally, it is impossible to build a universal SDAS capable of handling any possible type of satellite data. The known attempts to build a universal system [5] has resulted in extremely complicated and incomprehensive systems, hardly adaptable to implementation of new applications. The task for the present work was different: rather than developing a universal SDAS, the main goal was to develop a universal technology, that could provide a unified approach for solution of typical tasks in building SDAS for any particular satellite system. These typical tasks can be categorized as follows:

- building data archive and organizing data storage
- data archive administration
- providing access to data archive for the end-users.

The main specifics of these tasks are briefly described below.

Building data archive and organizing data storage

In practical development of IS APDD the data from different satellites are commonly used. Accordingly, the may differ in

Proceedings of the Workshop on Computer Science and Information Technologies CSIT'99
Moscow, Russia, 1999

format, volume, attributes, as well as their application. However, they also have much in common, and the tasks of building different data archives and providing user access appear quite similar. Naturally, this requires elaboration of unified approach to design of such systems. Use of standardized technology for the architecture of SDAS for different satellite data essentially simplifies the support of new data types and administration of existing data archives.

Archive administration.

One of the most important requirements for the modern SDAS is its full accordance with both satellite information acquisition/processing systems and systems for data dissemination and application. For the effective operations of SDAS it should implement maximum automation in both feeding new data to archives and transferring data to particular consumers. The term “consumer” denotes not only the end-user of satellite data, but also entire information system that utilizes satellite data and their processing results. Also it is important to add, that for many tasks most elements of SDAS must either operate on-line or be capable of switching to on-line mode. Hence, the following tasks are the most important for archive administration:

1. Automatic feeding of data to archive.
2. Archive elements access control.
3. Creating subsets of archive.
4. Transferring data to off-line storage, for long-term permanent archiving.
5. Restoration of data from off-line storage.
6. Logging of database changes.
7. Archive integrity control, error diagnostics.

Providing end-user access to data archive

The users may use various procedures to access data in archive, depending on particular types of data. However, in most cases a range of standard basic routines must be implemented to provide access to data. To provide cross-platform access and

independence, these routines should be implemented on base of WWW technologies. The following is list of most common request types to be implemented:

1. Search archive by one or several conditions.
2. Archive directory listing (by years, months, etc.).
3. View quicklook images for the selected data.
4. Receive data files over the network.
5. Order data files for later delivery (when network delivery in not applicable).

3. Satellite Data Center of Space Research Institute (IKI RAN)

Thus, the requirements listed must be taken into account during the design of SDAS. This paper presents the satellite data system, developed according to these requirements in Satellite Data Center of Space Research Institute (IKI RAN) [6-7]. The following is the brief description of the developed system. The principal structure of the system is shown in the chart (fig.1). The special attention is devoted to File Data Base (FDB) technology

3.1 Chart description

The scheme uses the following convention:

1. Solid lines denote data transfers (data files or other information streams). This does not include database requests.
2. Dashed lines denote control flow. This may be inter-process calls between components or database requests.
3. Parallelograms denote the general-purpose software products developed by third parties.
4. Squares denote software products developed for the information system (data independent).
5. Ellipses denote software products developed for the information system (data specific).
6. Circles denote data itself.

3.2 Brief description of system components

- **DB server**

Database server implementing (at least) subset of SQL data query language.

- **Data files (I-III).**

1. Data files I - source data files produced by archive feeding utilities (see below). Usually these are data files after (pre)processing accompanied by information files with data attributes. The files may be result of local processing or received over the network
2. Data files II - data files kept in local storage in a database system. They may be identical to files I or results of their processing. Typical example is creating a quicklook image corresponding to source data file. The source information files are not stored to the database, their contents (data files attributes) is filled into the database.
3. Data files III - data files moved from local storage to autonomous data media (magnetic tapes, compact disks).

- **Database**

The database built on FDB (File Data Base) technology, which has been developed specially to provide storage of data files in a database. This technology is described below in more detail.

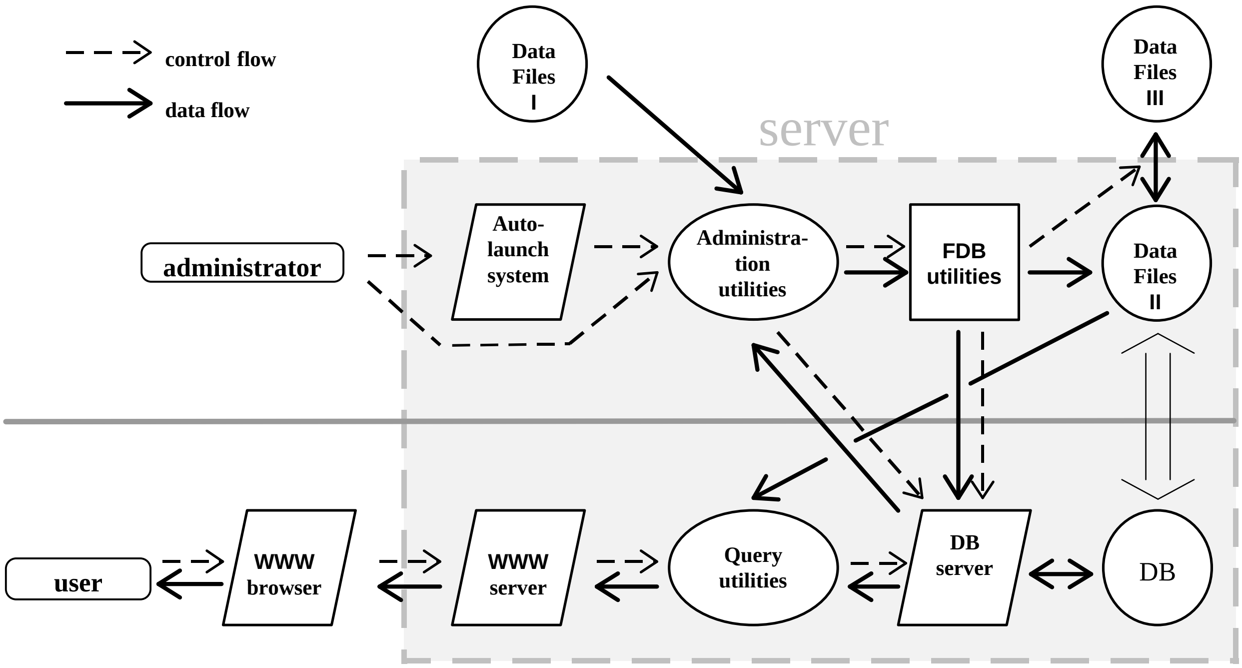

Figure 1 The principal structure of the system (SDAS)

- **Administration utilities**

A range of utilities designed for batch mode operation. They perform calls to standard FDB utilities with specific entry parameters. Their operation is illustrated in the upper half of the scheme. All utilities support both autonomous operation and manual start by archive administrator. Archive feeding routines (below) will illustrate the operations and interception with other elements of the system.

- **Query utilities**

Utilities in CGI (Common Gateway Interface) standard. Provide the possibility for the end-user to query archive for specific data. The utilities make the corresponding query to the database and return the user a selection from the database information with references to relevant data files in local storage.

- **Auto-launch system**

A software package for automatic launching of utilities and control for the results of their operation.

- **WWW browser**

Any software for viewing the documents by HTTP protocol, supporting HTML 2.0 or higher, and CGI protocol for launching applications on WWW server.

- **WWW server**

The requirement to WWW server is supporting CGI protocol.

3.4. FDB (File Data Base) technology.

The technology is a range of utilities corresponding to the developed structure of tables, designed to build and administer databases oriented to storage of data files. Usage of specialized utilities for basic operations (e.g. adding files to archive, removing data files, etc.) ensures strict correspondence between file operations and changes in database tables. This approach provides much better data integrity, than with independent file and database operations. It is important, that this also allows for the unified approach to storage of files, independently of their specifics. The utilities are designed for autonomous batch operations and support the standard logging for operation control and error diagnostics. Actually, the package is a super-shell over the relational database server, providing possibility to store new object “data file” in a database and perform basic operations with this object (adding file, replacing file, removing file, moving to remote media, restoring from remote media to local storage, etc.).

Physically the files are stored in a directory structure corresponding to the structure of database. Data files, referenced in a database table, are kept in a disk directory corresponding to this database, in a subdirectory, corresponding to this particular table. This structure is kept for both local file storage and files on remote media (CD-R disks, etc.), to provide easy transfers to and from autonomous media.

In a database table structure the “data file” object is realized by 4 fields, that correspond to file attributes: initial filename, file size, archive volume, internal storage filename. Each field name is composed from file object name and special postfix denoting file attribute type, that is used by FDB utilities to recognize the field. For instance, the object named “picture” will be represented in the database table by the following fields: picture_NAME, picture_SIZE, picture_VOLUME, picture_STORAGEFILE.

To provide functionality the following utilities have been implemented:

- CREATE - creating new FDB table
- ADD - adding new records

- REMOVE - removing records
- DETACH - moving files to off-line storage media
- ATTACH - restoring files from off-line storage

The utilities has been implemented on top of specialized library, developed in C language. Hence, their range can be increased for specific tasks in case of need. Interoperation with the database server is implemented in a separate module basing on programming interface provided by particular DB server. For migration to another SQL server this module should be re-written. At present, the FDB package supports 2 different DB servers: "miniSQL" (by Hughes Technology),

3.5. The functionality of archive administration subsystem.

The interoperation of system components will be described by operation of feeding new data to archive as an example. This is one of basic operations for the archive maintenance, it is implemented through a dedicated administration utility. It can add new and/or replace existing database records and corresponding data files in archive. Upon start the utility is supplied with source files to be added to archive. Usually, the files are received from special directory, where they are previously stored either after local processing or from the local-area network. On first step, the files are analyzed, and bad files, if any, are discarded. Next step is data processing, if required. Typical processing example is preparing quicklook images. After that, the utility composes the entry data for FDB utility ADD, in the form of a text file containing strings with data to be stored to database and references to corresponding data files. Certain information from the database may be required to prepare data for feeding to archive. In this case relevant database queries are performed. Next, the FDB utility ADD is called to issue DB server commands to add new records and, synchronously, store data files to local storage. The ADD utility returns relevant exit code to indicate result of operation and saves operation information to corresponding logfile. The feeding utility itself returns exit code to enable auto-launch system to control operations.

Other administrative operation (transfer files to off-line storage, removing files) are implemented similarly. For batch operation, they must be supported by specially developed utilities. It should be noted, that with the help of FDB utility DETACH data files can be moved to off-line storage media, and restored back to local storage with utility ATTACH. It is essential, that FDB utilities perform file operations, like removing or replacing files, only with files stored locally.

3.6. Query processing procedure.

User access to the system has been implemented through WWW interface. The user can perform a wide range of queries, such as search for data by date/time, search for satellite images by coordinates or regions, etc. The query parameters are specified using HTML forms; alternatively, the user may select from pre-defined queries.

The query is translated to WWW server using CGI method GET. The query itself is a call to CGI script with corresponding parameters. The role of scripts is actually performed by query request utilities. Each type of query is supported by a specific utility, that actually queries a database server for data. Based on received data selection, the utility then creates an HTML document. It usually contains tabular data, each table row corresponding to a single data file. Along with information fields, it may contain references to data files and/or their quicklooks. The references may point only to local-storage files; for off-line-stored files the table displays file location.

The composed HTML document is then returned to user's WWW browser. Following file references, the user can view quicklooks and download the data files. Using the query results, more queries may be issued possibly of different types. An example of sequential archive querying is browsing the archive directory (see the catalogue of RESURS series satellite data on the SPUTNIK Web server *(http://sputnik.infospace.ru/resurs/engl/resurs.htm).*

First, the user selects month of interest, then using information about data availability by days of the month, selects particular day to view quicklook images.

The procedures of operations with data can be easily changed and adapted for optimal access to particular data types.

3.7. General information on implementation.

The first variant of the described technology has been implemented at the Weather Satellite Data Acquisition and Processing Center at IKI RAN. The implementation was based on the following facilities:

1. Computer/Operating System: SUN SPARCStation 5, OS Solaris 2.5.1
2. WWW server: Apache
3. Administration/query utilities programming language: Perl
4. DBMS: MiniSQL 2, mySQL
5. Auto-launch system: crontab and specialized software.

4. Conclusion

The presented technology has been proved to be a universal and efficient mean for the development and maintenance of satellite data archives. Using this approach, the following systems has been sussessfully implemented:

- geostationary weather satellite GOMS data archive *(http://sputnik.infospace.ru/goms/engl/goms_e.htm);*
- operational catalog of data from Russian Earth-resource observation satellite RESURS-01 No.3 *(http://sputnik.infospace.ru/resurs/engl/resurs.htm);*

- archive of fragmented data of AVHRR instrument installed on NOAA series weather satellites *(http://smis.iki.rssi.ru/data/regions/reg_s_e.shtml)*.

The same technology is planned to be utilized for development of archives and catalogues of data from other satellites: RESURS-01 No.4, OKEAN, OKEAN-O, METEOR, METEOR-3M. In authors opinion, this will allow to decrease significantly the expenses for development and maintenance of these systems.

References

1. Zakharov M.Yu., Krasheninnikova Yu.S. Loupian E.A., Mazurov A.A., Nazirov R.R., Flitman E.V. Khokhlova N.L. Open Information System for Satellite Data Users Support // Preprint of The Space Research Institute Pr-1929. 1995.

2. Zakharov M.Yu., Krasheninnikova Yu.S. Loupian E.A., Mazurov A.A., Nazirov R.R., Flitman E.V. Khokhlova N.L. Possibilities of organization of operational user access to weather satellite data.// In “New information technologies and remote methods of industruial observations of fishing regions of the World ocean”. M.: VNIRO. 1996. pp. 91-111.

3. Zakharov M.Yu., Loupian E.A., Mazurov A.A., Nazirov R.R., Flitman E.V. Development of Fully Automated Systems for Satellite Data Acquisition with Remote Control over the Internet // 2nd International Symposium on "Reducing the Cost of Spacecraft Ground Systems and Operations". Proc. 1997. RAL Keble College. Oxford. P 76.1-76.10.

4. N.A. Abushenko, S.A. Bartalev, A.I. Belyaev, V.V. Ershov, M.Yu.Zakharov, E.A. Loupian, G.N. Korovin, V.V. Koshelev, Yu.S. Krasheninnikova, A.A. Mazurov, N.P. Minko, R.R. Nazirov, S.M. Semenov, S.A. Tashchilin, E.V. Flitman. V.E. Schetinsky Experience and Perspectives of Near-Realtime Satellite Monitoring of Russia’s Territory for the Needs of Forest Fires Services // Earth Research from Space 1998. N 3. P.89-95

5. Dopplick T. A Scince User’s Guide to the EOSDIS Core System (ECS) Development process // EOSDIS Science Office Technical paper 60-TP-003-01 1995.

6. Zakharov M.Yu., Loupian E.A., Mazurov A.A., Nazirov R.R., Flitman E.V. Present State and Prospects for Further Development of the SMIS IKI RAN Information System for Support of Satellite Data Users // 2nd International Symposium on "Reducing the Cost of Spacecraft Ground Systems and Operations". Proc. 1997. RAL Keble College. Oxford. P 36.1-7

7. Zakharov M.Yu., Loupian E.A., Nazirov R.R., Mazurov A.A., Flitman E.V. IKI RAN Information System for Support of Users of Meteorological Satellites Data // Proc.1st International Symposium on “Reducing the Cost of Spacecraft Ground Systems and Operations”. 1995. RAL Chilton, Oxfordshire. P 60.1-60.6.

8. Operational Space Systems. // Ì .: Gidrometizdat 1993. p.16.

9. Selivanov A.S., Tuchin Yu.M. Operational Earth-Observashin System Resurs-01 // Earth Research from Space 1988. N 3. P.101-106