Rerum Naturalium Fragmenta

No. 355

Cartographic Features Data Base
Access System
Functional Specifications

A.R. Chaplin, T. Jasko
&
J.R. Whiteway

Glasgow
1983

.

Rerum Naturalium Fragmenta
Tamas Jasko editor
34 Lomondside Ave, Clarkston, Glasgow G76 7UJ, Scotland

1. Introduction

Cartographic features were identified by the Exploration database requirements review as having a very high priority.

It was also emphasized that because of the close relationship between these data and the SEL Graphics System, the logical place for this database was on the SEL system.

It was recommended that a functional specification should be prepared by Software Development describing this database and how it could be set up on the SEL B system.

This functional specification attempts also to describe how other datasets not belonging to the Cartographic Features system can be retrieved and accessed by the Graphics System.

The Cartographic Data Base will contain the data which are required to appear on maps produced on Exploration computers with the exception of

(a) seismic line (shot point) data
(b) CPS1 contour data
(c) map titling information

Data in (a) are handled by the shot point positional data base (SPPDB), data in (b) are handled by CPS1 and associated programs, and data in (c) are handled by general plotting software.

Plot overlays using these three data types and the map base produced from the cartographic data base are merged together by plotting software.

As most cartographic applications require data selection by fairly simple retrieval criteria, using a fully fledged database management system would be perhaps an overkill. In any case, the primary requirement is to have fast data access with relatively low overhead.

At present, the size and complexity of the data necessitates a compromise between physically contiguous storage of logically related data and acceptable standards of data editing (entry, update) speed, making data preparation for base maps a difficult and lengthy task.

On the other hand, a cartographic data access system is required to make data selection, and to some extent, maintenance, an automatic function without the need of human intervention.

A brief outline of the proposed system is given in Section 6. Sections 2—5 discuss the contents, maintenance, access and storage format aspects as a preliminary to technical specification of the program suite required to access the cartographic data.

Note that the use of the database requires parallel development of a map plotting program which can make use of the features included. Discussion of this program is outwith the scope of this document although its requirements are the main reason for cartographic data base development. The limitations of the mapping program will essentially be the limitations of the cartographic data base.

The considerable effort required to develop and implement this system is well justified by the benefits expected, i.e. the ability to produce more maps in a timely fashion. The need to do this has appeared in the Exploration data base requirements as the overriding need of the Exploration groups.

2. Data base contents

The data base consists of data which have geographical location as the main attribute. All locations should be stored as double precision latitude and longitude (1 metre resolution at the Equator requires 9 decimal digit accuracy). Since this is independent of projection, it will enable data to be displayed using straight-forward projection calculations.

Associated with each location of group of locations (see below) are a variable number of attributes. The number of these will vary from a minimum of one, being the name of a coastline for example, to a maximum, which allows for the entire set of posted values required of a well.

The data fall into three categories:

(a) contours

Examples: coastlines, topographic height contours, sea depth contours, field boundaries, licence boundaries

Contour data can have an inside and an outside, thus allowing the mapping program to colour sea depths below 500 feet, for example.

(b) lines

Examples: pipelines, median lines, rivers, projection lines, spider plots

Line data are simply a group of geographical locations which define points on a line.

Storing projection lines (i.e. lines of constant latitude or longitude) can be used to terminate the lines at a special feature, such as the coastline, without having to do a complex intersection calculation between two data types (although this should be possible within the mapping program). The line end-points are digitised and stored.

(c) points

Examples: wells, towns, spot heights/depths

Some data types - such as towns - can exist as more than one category, either as points or contours. Thus the category is subsidiary to the data type and defines the storage/retrieval format.

3. Data base maintenance

The three maintenance facilities that are required are loading the data into the data base, modifying existing data and deleting existing data. Since the data base will be maintained by the Geodata group, the maintenance programs will be operated by specialist personnel and need not be designed to cope with naive users.

A useful facility is automatic amendment logging. Although it may be decided that this is not required immediately, the option of adding it later should be included in the design.

The data to be loaded is of two types:

(i) data that exists solely on the exploration computer system, and
(ii) data that exists primarily on another Britoil computer system (principally the IBM) which will be transferred (with a probable reformatting) to the exploration system.

Type (i) data will require loading from (variously formatted) tapes, in—house digitiser output and manual entry at a keyboard. Type (ii) data will require machine transfer facilities, but the loading program only needs to handle one data format.

Editing is obviously required for type (i) data. Editing of geographical locations will require resorting/indexing of the data, since data will be indexed mainly by location and this is probably best done in a batch-like mode where indexing is deferred

to the conclusion of an editing session. Editing of non-keyed, attribute data is much simpler.
For type (ii) data, there is a need to keep the data on both systems consistent. If this is to be strictly adhered to, then all amendments must be prohibited on the exploration system.

Clearly, deleting data should be allowed, although this could be confined to data about to be reloaded.

This implies that all editing is done on the primary machine and complete, up to date sets of data (e.g. licence boundaries) are periodically reloaded onto the cartographic data base.

Should the requirement for consistency of data between machines be relaxed, then editing would proceed as for type (i) data above.

File management needs

To operate an indexed access method successfully, file management functions presently available (under MPX1.5) have to be extended considerably.

File directories should provide information on the owner of file, the time it was created, the time it was last updated (and by whom), and the time it was last accessed by programs connected to the cartographic system.

Also, it should be recorded whether or not a file was indexed (and if so, when) for cartographic retrieval.

These data along with the parameters of archiving operations should be available directly even if the file itself is relegated to tape storage.
This information can be kept in the form of separate directory files - one for each project (user/owner, volume) and maintained/updated by subroutine calls from the file access system.

Stand-alone programs are needed to perform functions of data administration and directory validation, especially during the transition period.

In this period some programs that use/update the data files will use the new extended directories while others will not yet be connected to directory maintenance routines. After all program modules will be reorganized, the role of stand-alone directory maintenance will be mostly to provide statistics and periodic archiving.

The directory information needed by the cartographic features access system forms a subset of the enhanced file directories proposed for implementation in conjunction of operating system development for the SELs.

For this reason, file access routines will be designed to make full use of the expected or proposed enhancements as soon as they become available, by anticipating, and in some cases assuming these general systems facility extensions.

Among the numerous advantages offered by the enhanced file directories one is particularly useful in reducing disk congestion.

A substantial part of the files, containing less frequently used data can be relegated to tape storage and still remain accessible. By keeping essential information related to these files on disk, data may be restored automatically.

Applications programs, requesting data from these files, will be suspended while the tape restore operations are performed. Apart from a slight delay, apparent only in on-line mode, the file access, whether disk or tape, will be transparent and effective.

On-line users will, of course, have the choice between waiting for the completion of file access or to return to TSM and resume operation of the applications program later (i.e. after the files required become available).

Fully automatic management of data files can be achieved only for data sets permanently resident on the SEL and regularly maintained by Geodata group.

Maps and consequently, indexing, will include various sets other than resident. These include project data of relatively short lifetime, sets retrieved from other machines and user data edited for special purposes or not accessible to other users (even if permanently stored on SEL disk files).

The maintenance of non-resident datasets will be the responsibility of the declared owner.

Data/index integrity

To alleviate problems generated by creating, updating or deleting of data sets outwith the control of the cartographic index system (this can happen most frequently, but not exclusively to non-resident sets) both the file directories and the index files will have housekeeping routines for periodic validation, cross-checking, garbage-collection etc.

Using separate files for indexing, etc. and performing major directory/index updates in batch will reduce the risk of file corruption.

If, during processing of an access request an anomaly is noted the user will be warned of the possibility of an incomplete or inaccurate dataset being retrieved.

This way, the user can initiate actions deemed necessary to correct (or ignore) the anomalies noted.

Inter—machine links

Some of the data used in the system will reside on other machines and copies or subsets of the original files only will be loaded according to the frequency of updates.

Whether permanently used or defined as ad-hoc data sets, these will be presented for indexing prior to use as part of the cartographic database.

Directory entries of non-resident data type can be marked to indicate an approximate lifetime.

The life-time parameter can be used in two ways:

- as an indicator of 'best before' (A warning will be issued if data retrieved for a request have expired lifetime);
- as an indicator that the data set is a candidate for deletion (usually to be kept on tape).

When a fast data link becomes available, the cartographic access system should be extended to forward requests for data transfer. Selective retrieval of temporary datasets thus can be achieved bypassing batch indexing.

4. Data base access requirements

Data base organisation, and retrieval methods are determined by their purpose, to serve as a base for map production.

A map is a representation of a geographic region and the primary attribute of data that appear on the map is that they belong to this region. In the simplest case, all data held on the data base within the map region will have to be plotted.

Thus all retrieval from the data base will, first of all, specify geographical limits and in some cases, this will be the sole retrieval parameter.

The second retrieval parameter is the data type. Each logically different data set will have a designated data type. The examples in Section 2 form an initial list of required data types. The main data type categories,can, of course, be subdivided to indicate format etc. versions.

For some data types (e.g. coastlines) these two parameters (geographical limits and data type) will be sufficient for the most complex access requirements.

Other data may require the storage of additional attributes. The first such attribute, defined for all data types, is the data element name. A set of further 'compulsory' attributes will need to be pre-defined for each data type which will cover all common retrieval parameters.

A number of optional attributes which cannot be specified as retrieval keys but which can be returned with the keyed data, should be allowed.

This will enable, for example, a special well data set which contains reservoir variables to be created, and then loaded the cartographic data base, with the purpose of having the values posted on a map.

The data category (contour/line/point) would not be retrieved upon (as it should be implicit in the request) except as far as this determines the retrieved format etc. so that the mapping program will know how to handle the data.

Resolution (sampling density) is an important feature of digitally stored contour and line category data having considerable importance on graphical output. Because of the quantity of data which can be used to define, for example, a coastline at a scale of 1 : 10 000 it would be impractical to use all this data to draw a map at 1 : 100 000. There is a need to specify a practical resolution for a particular map.

The data retrieval routine should recognize requests to retrieve contour/line points at or near specified resolution levels provided the data are available in a resolution warranting selection.

Note that this only applies to contour and line data that will be more accurate if held at a higher resolution. There are certain line data (such as boundaries of most licences) that would not need variable resolution access.

Program interface

Data retrieval facilities of the system should be accessible to applications programs as a set of subroutines.

Calling programs will have to supply retrieval parameters e.g. geographic window, data type(s) and format(s) and specify a buffer area for the data to be returned.

Access routines will look up indexes for pointers to primary data, allocate the necessary files and place the selectively retrieved data in the buffer provided.

A status variable will be also returned to indicate completion status. This may be OK, or an error/warning indicator or a buffer full indication. In the latter case the calling program can empty the buffer and request more data.

Presentation of data to requesting programs

All data will be returned in the internal storage format (double precision binary floating point numbers, fixed/variable length ASCII character strings, integer words, etc.) as defined for the particular data type requested.

No data transformation will take place at this stage. It is assumed that all data will be standardized to comply with type definitions by editing/updating prior to the indexing run.

While index records will point to data segments out of context, logically related line segments etc. that occupy contiguous

parts of the same primary data file will be returned as a contiguous sequence with the primary ordering preserved.

5. Physical data base design

The aim of any physical data base design is to make the system efficient in terms of its usage of

- disk space
- disk access
- main memory allocation
- CPU processing time

The system outlined here aims to balance these factors with particular consideration to coast line data. These were chosen as a pilot data set for two reasons:

(a) data loaded from the IBM will be appropriately pre-formatted i.e. not typical of the majority of cartographic data

(b) coast line data are a good example of the kind of bulk data to be handled by the system; most other data types will be well served by the techniques applicable to coast line data.

E.g. coastlines are likely to require fine tuning in index entry precision even for large data blocks indexed as units, while other data e.g. well location records can be easily retrieved individually with less precise indexing.

As most retrievals will be based on a latitude/longitude reference, the main index will be constructed with that reference as a primary key. An outline system structure is shown on Fig. 1.

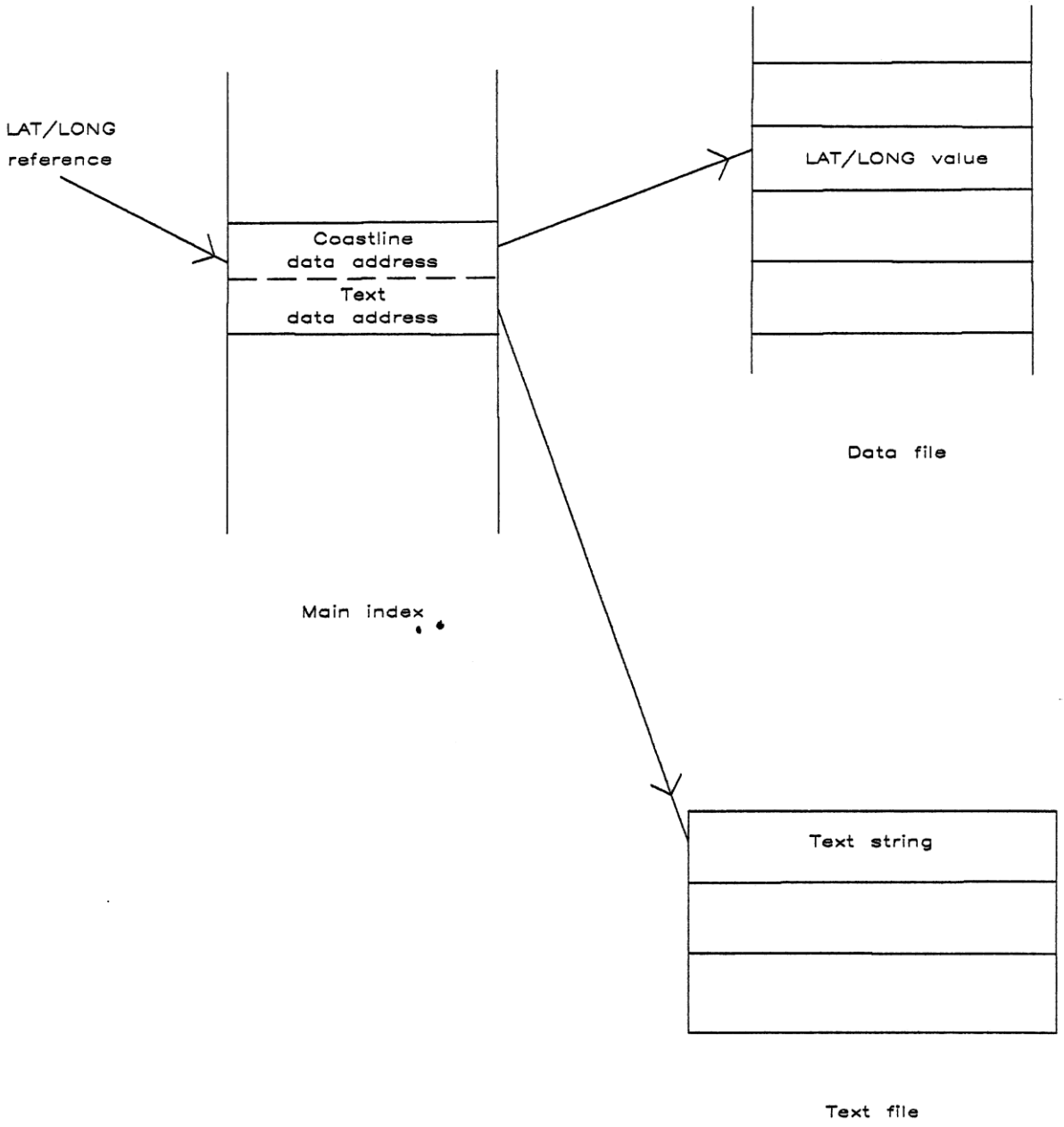

Figure 1

Data structure will allow for file inversion (e.g. on text strings of given length) to extend retrieval capabilities. The main data files would have to be formatted for direct access but the index can be stored either as a sequential file or as a direct access file with the latitude/longitude reference hash coded.

Geographic indexing

Data files become connected for cartographic access through the indexing utilities. Indexing will take place as a background activity even if invoked on-line from a catalogued load module via subroutine call.

The basic indexing entity handled is the file, rather than the records. At the close of a data editing (modification/entry) session the updated file is read through to establish index records of significant geographic locations pointing to data blocks within the file.

These index records, of course, will also contain information on the file pathname, data type, and format. Corresponding directory entries will be marked with the date/time etc. of indexing in addition to information on the last update date. The indexing of files needs to be of sufficient depth to reduce the amount of irrelevant data retrieved even if precision clipping is taking place in the mapping program.

On the other hand, to keep index file sizes at a manageable level indexing will be on an intermediate level both in terms of geographic index entry size (e.g. not exceeding single precision) and also in terms of data pointers, pointing to blocks of data consisting of one or more records.

Naturally, this approach should cope with the characteristics of vastly different data.

Index file structure

The geographic index consists of a number of hierarchically inter-connected files each covering a defined geographic area. The size of individual index files and the area covered by them is a function of feature density. As the data base grows through additions of new data entries/files, more and more index records are generated.

Rather than extend files indefinitely j a multi-tier structure can be realized with each level of files having a given resolution. Where an overflow occurs, direct pointers, i.e. those pointing to actual data, will be replaced by indirect pointers to the next more detailed level of index files.

It is outwith the scope of this document to estimate file sizes actually required, but it is suggested that any future estimates should include an extra 25 per cent to allow for expansion (or should indicate other means of avoiding physical overflow).

Indexing for special retrieval criteria

Some data sets will be searched by other retrieval criteria than geographic window. The feasibility of including some simple forms of data selection by specifying a range (mainly for numeric data) or mask matching (for character data) should be investigated. If warranted by demand, inverted index files can be created for use by this type of search.

It is not expected, however, that the access system will deal with searches on more than a single key field at a time (perhaps in addition to geographic windowing).

The design of the retrieval routines will therefore assume that sufficient pre-selection of data sets has taken place by using other means (automatic or manual data editing techniques, resulting in temporary data sets).

6. Data flow for map production

Figure 2 illustrates the data flow for map production.

(a) Production of map base

The mapping program MAPPER is used to create a map base. Parameter input to this program will specify the geographical limits, projection details, which of the data types are to appear, and other selection details.

Input will also specify which, if any, pieces of plotted information will be accessible for interactive graphic editing. Editing will be restricted to deleting or changing the plotted position of a data attribute (i.e. name on map), not the input data itself.

(b) Production of overlays

Seismic line data and CPS1 contour data will be produced as separate overlays, through the Geo-Graphics and CPS systems, respectively.

(c) Overlay merging, addition of titling

The map base and overlays will be stored as pictures under a project name. This project name will then be used to pick up all pictures to be merged for the final map.

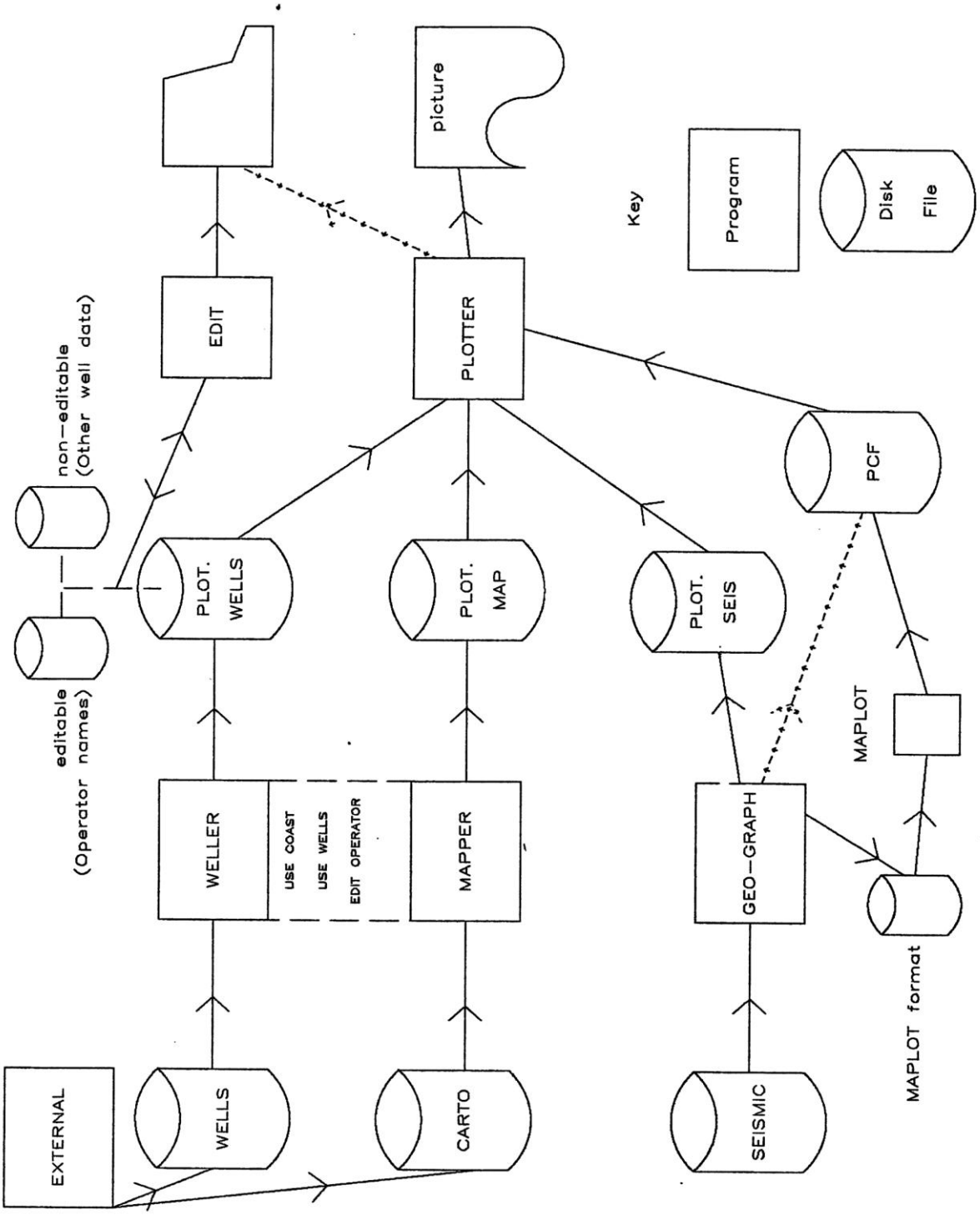

Fig. 2: Data flow for map production

This "compositor" program which will do this will also add titling if required and merge in additional files. These additional files which need not be stored under the same project name.

Display of the final picture will be to any supported graphics device. It will not be necessary to pick up all related files, so if a check plot of a particular overlay is required it can be simply produced.

7. Implementation

Implementation of the cartographic data access system includes the following logical steps:

1) Program design. This includes definition of module groups and individual modules, design of index file formats, decisions on control and data flow, specification of file directory and maintenance utility requirements.

2) Writing of individual modules for indexing. Debugging and testing with a suitable pilot data set e.g. digitised coast line data or other typical line data.

3) Writing of file access interface. Initially using test/dummy calls to file directories, gradually connecting to live directory files. This event will depend on the progress of upgrading the SEL file system in connection with operating system changes.

4) Writing and testing the Fortran driver/interface to connect with applications programs. Utilize inter/machine transfer as available.

5) Extending the system to cover all types/formats of carto-graphic features, gradual loading and indexing all defined data.

Development timescale

Most of the software outlined above will be developed in-house, with the possible exception of the enhanced directory system.

The need for enhanced file management facilities makes all estimates dependent on developments of the operating system. The whole project is expected to take about a year to complete assuming that the details of the operating system/file system changes will be known sufficiently in advance.

Assuming existing trends of system development and man-power availability to continue time requirements can be outlined as follows:

Task	*Elapsed time (weeks)*	*Man days*
Logical & physical design	12	35
Coding of program modules	18	90
Debugging and initial testing	7	45
Additional testing and loading	10	20
Totals	*47*	*190*

The modular design of the programs should make it possible to release parts of the system for use before overall completion. Experience from the testing of released programs will give useful feedback for the tailoring of subsequent modules to user requirements.
(Exploration Data Processing, Glasgow, 1983, 25 p.)