COMPTEL PROCESSING AND ANALYSIS SOFTWARE SYSTEM: COMPASS (REQUIREMENTS AND OVERVIEW)

J.W. den Herder [1], H. Aarts [1], K. Bennett [2], R. Diehl [3], W. Hermsen [1], M. Johnson [3], J. Lockwood [4], M. McConnell [4], J. Ryan [4], V. Schönfelder [3], G. Simpson [4], H. Steinle [3], A.W. Strong [3], B.N. Swanenburg [1], C. de Vries [1], C. Winkler [2], I. Wood [2]

[1] Laboratory for Space Research Leiden, P.O. Box 9504, 2300 RA Leiden, The Netherlands
[2] Space Science Department of ESA/ESTEC, Noordwijk, The Netherlands
[3] Max-Planck-Institut für Extraterrestrische Physik, Garching, FRG
[4] University of New Hampshire, Durham, USA

ABSTRACT

In this paper the COMPTEL Processing and Data Analysis System COMPASS is described. A general overview of the system software is given with a focus on the software engineering aspects.

INTRODUCTION

COMPASS is the software system which will support the analysis of the data from COMPTEL, one of the four astronomical experiments aboard NASA's Gamma Ray Observatory. COMPTEL, a Compton telescope, is the joint product of the Max Planck Institut für Extraterrestrische Physik in Garching-bei-München (BRD), the University of New Hampshire (USA), the Space Science Department of the European Space Agency and the Laboratory for Space Research Leiden (The Netherlands) (Schönfelder et al., 1984). During its first one and a half years in orbit COMPTEL map the sky in the energy range from 1 to 30 MeV. Thereafter in-depth studies of selected objects or regions of the sky are planned. In addition, COMPTEL is able to collect spectra of gamma-ray bursts (after receiving a trigger from BATSE, one of the other GRO instruments) and to collect neutrons from solar flares. In figure 1 the main scientific output is shown schematically. The expected lifetime of GRO is at least 5 years.

A special software environment to process and analyse the COMPTEL data was required because of the following reasons:

* Analysis of the data from a Compton telescope is rather complex and CPU intensive. Duplication of work between the participating institutes should be avoided and it should be possible to exchange the data between the sites;

* The size of the programs needed to perform the scientific analysis was estimated to be 100.000 lines of executable Fortran code (it turns out to be closer to 200.000). As all institutes are involved in the science, it was agreed that the development of the software should be shared between them;

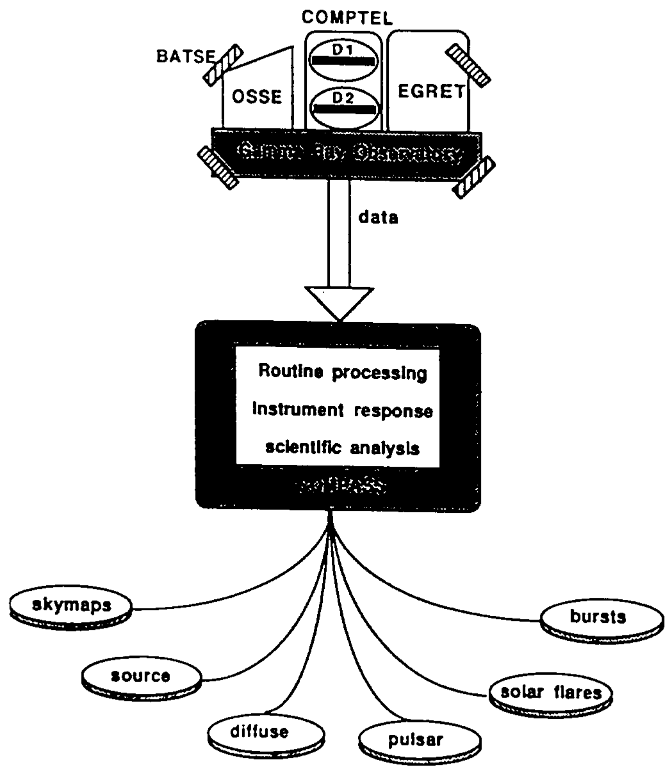

Fig.1. Schematic overview of the COMPTEL processing and analysis software system COMPASS

* All the institutes have different computers and operating systems. In order to facilitate exchange of software and to protect it against changes in the operating environment the scientific software should be independent of the actual host computer.

* The volume of data to be processed and analysed is large (25 Gbyte of raw data per year). This requires special software to handle and to control such volume;

* It is expected that the methods will evolve during the mission and that re-processing of the data can be important. At least it should be known under which conditions (software versions and control parameters) certain data were produced.

In this paper some aspects of COMPASS are described with a focus on software engineering and data handling. After describing the development approach, the system requirements and the system design, a global overview of the scientific components is given including the routine processing, the software to determine the instrument characteristics and the scientific analysis.

DEVELOPMENT APPROACH

In a project expected to last over more than 15 years (the development time and the expected life time of GRO) and with an involvement of over 30 people situated at different locations, it is crucial to have a common development approach. A top-down development approach was selected because this would allow the collaboration to define in a early stage the different components (subsystems) to be developed by each of the sites.

This top-down approach started with the definition of the development approach itself and the definition of the software requirements for the system. Subsequently the requirements for all its components (subsystems) were specified. Next an architectural design was made (for the full system and for each subsystem) and finally the detailed design together with the code. Each part of COMPASS was completed by an appropriate test plan and report as well as a user manual. Configuration control was strictly exercised for any of the generated documents as well as for the program units. Also the interfaces between the various subsystems were controlled.

The top-down approach made it possible to assign special parts of the software to each institute. However, modifications to some of the application subsystems required at a later stage in the project, could have been avoided if work on the central supporting software had been completed at an earlier stage. Also the fact that not all scientific methods were fully established in the early days (e.g. a lack of detailed scientific prototyping), resulted in some cases in software which did not meet its requirements (e.g. the throughput of some software was not sufficient) and which made significant modifications in a late stage necessary.

In addition to the development approach some other aspects of the project had to be fixed. Fortran 77 was selected as coding standard and none of the computer specific extensions was allowed for. Nevertheless, certain computer specific limitations were still encountered during the porting of the software from one computer type to another. Also a number of external software packages were selected, mainly on the basis of the availability on a large number of platforms. This included the CERN graphical display packages HBOOK and HPLOT, the relational database ORACLE, a maximum entropy package MEMSYS and the NAG mathematical libraries. Furthermore, where applicable, the data are stored in FITS format in order to allow other packages such as AIPS and IRAF to display COMPTEL data.

SYSTEM REQUIREMENTS

As part of the top-down development approach the requirements for COMPASS and for each of its subsystems were specified. The toplevel requirements include the following items:

* The system should be functionally equivalent at all sites;

* There should be a uniform interface to the host computer allowing for an easy exchange of the software between the sites;

* There should be a uniform user interface to all COMPASS software which also takes care of the differences between the Job Control Language at each of the host computers;

* The should be proper software configuration management. The tools should allow for evolutionary changes to the software and the status of the software should be traceable to any point in time;

* The data generated by COMPASS should be properly ordered and catalogued using the same identification code at all sites;

* The heritage of generated data within COMPASS should be traceable including the original data, the applied programs and the relevant control parameters;

* The exchange of data between the sites should be feasible allowing for different local file systems and data representations at any of the sites;

* There should be test environments which provide the same functionality as the production environment, and which permit the development and testing of new scientific applications, before they are released to the production environment.

As will be shown belolw, these requirements are implemented in a number of supporting system components. In addition, the primary purpose of COMPASS is, of course, to process and assist in the analysis of the scientific data. For this the following top level requirements were specified:

* Provide tools to process the raw data including corrections to remove time variable effects on the instrument and to express the data in meaningful physical units;

* Provide tools to determine the instrument response (module response, telescope response, response to instrumental and atmospheric background);

* Provide tools to perform specific scientific analysis in order to extract meaningful scientific results (e.g. timing analysis, sky mapping, source recognition, burst analysis, diffuse gamma-ray emission etc.);

* Provide tools to display scientific results achieved in COMPASS.

These requirements will be met by various scientific subsystems which will be discussed in some detail once the design of the supporting system components has been presented.

SYSTEM DESIGN

The design of the COMPASS system will be presented using three different views: the conceptual design gives a global view of the system, the data model provides an overview of the entities managed and controlled by the software and the activity model gives an overview of the various scientific functions. Together these provide an overview of the system.

Conceptual Design

In figure 2 the conceptual design of COMPASS is given. COMPASS is represented as a number of shells around the host computer. All independent functions are handled by separate layers. This makes each part of the software relatively immune to changes in another part of the software which is a big asset during maintenance of the system.

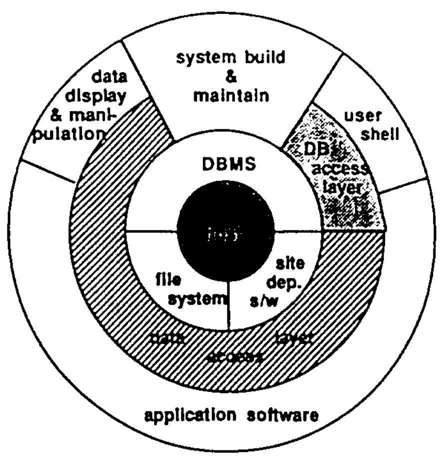

Fig.2. Conceptual design of COMPASS

The inner shell isolates COMPASS from the specific host computer and the data storage system. This shell contains three parts: program units which take care of site dependent system calls (time, bit shifting, terminal and printer access, etc.), program units which take care of the site dependent part of the local file system (such as filename convention, differences between the Fortran OPEN calls) and the database management system (DBMS). The DBMS contains, from the software perspective, a full image of the COMPASS system. It contains descriptions of all the available software at any of the four sites, of all data used and/or generated within COMPASS and all control parameters for the software.

The second shell, which is independent of the specific host computer, provides a number of services to the application developer and to the users. The data access layer separates the application programs from the data storage and allows modifications in the physical data structure without the need to modify the application software. This is a major benefit for a project of this size and duration. It also allows for the combination of data measured during different periods without modifying the application software (different physical datafiles are combined in the data access layer into a single logical datafile). The database access layer defines a number of operations to retrieve and store control parameters in the database.

The third shell is that which is seen by the users. The application layer contains the programs which perform the various steps in the data processing and analysis. There are 18 subsystems in this layer and their functions are described in more detail below. The "user shell" is a menu driven system which allows the user to select a job to run, to provide its control parameters, to select data from the database and to submit the job to the batch queue or to execute the job interactively. The user shell will check on the availability of the data and, when needed, retrieve the data first from archive. In addition, the user shell provides some additional functionality to the user: one may select control parameters from a previous run of the same task (program) or one may combine a number of tasks into one job in which the output of one program is used as input for the next program.

The data display and manipulation subsystem consists of three parts: a number of special display tasks for a given data-type (such as events), a task to download any of the data generated within COMPASS to a PC or workstation and commercial software to manipulate the data. There has been no selection of a single package for this purpose as this depends heavily on the available hardware at the four sites.

There is one zone in the diagram which is referred to as "system buond and maintain". Its function is twofold: it allows a privileged user to add components (application software or external data) to the system by inserting the appropriate records in the DBMS and it allows all users to query the database. In addition it handles all the communication between the different sites for database entities, software and data.

Data Model

A simplified data model is given in figure 3 using Entity Relationship Diagrams.

Each of the entities in the shaded boxes is stored in the DBMS which contains a full image of COMPASS. A USER may start one or more JOBs where each JOB may consist of one or more (connected) TASKS (programs). For each run of a task the CONTROL PARAMETERS are stored as well as the used and/or created data. This data is described by a DATA DESCRIPTOR in the DBMS. Each TASK specified by a single SOFTWARE DESCRIPTOR consists of a number of SOFTWARE LIBRARIES linked in a specified order. An important asset of this image in the database is that at any point in time the heritage of data can be traced and that previous versions of software can be re-installed and previous control parameters can be queried (making use of the functionality of the DBMS).

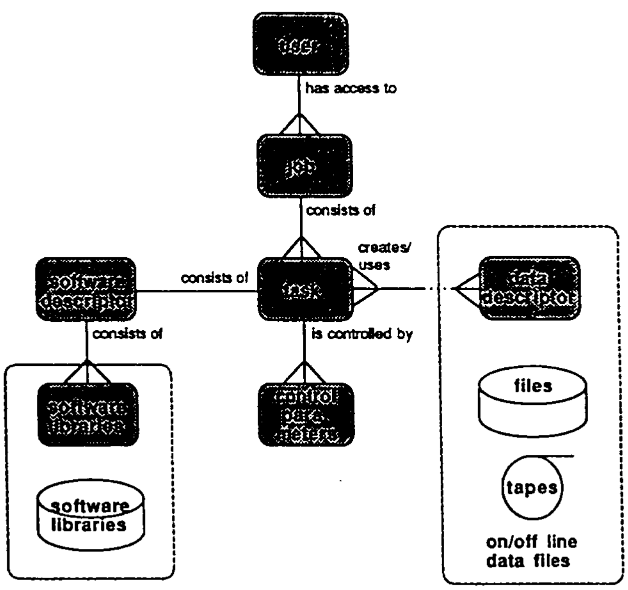

Fig.3. Simplified data model (entities stored in the DBMS are specified in the shaded boxes). Dotted lines indicate optional relations, solid lines mandatory relations between two entities, a single line denotes a one-to-one relation and a fork a one-to-many relations or many-to-one relation.

DATA DESCRIPTORS are the key entity used to facilitate data access and to protect COMPASS from site specific conventions for file names. Each DATA DESCRIPTOR has a number of attibutes which may facilitate its usage:

* a unique identifier consisting of the generating site, its type and a sequential number. This allows unique references to the data throughout COMPASS at all four sites;

* a time validity interval specifies the time for which the data are valid;

* the creating task, time and user help to separate valuable data from data produced by a too old version of the software and to trace back its heritage (creating task and related control parameters);

* a user supplied title may contain any scientific relevant information;

* a quality flag which can be used to specify the quality of the generated data;

* an access flag which can be used to provide certain users with a limited access to the data (e.g. guest investigators will only be interested in higher level data).

For the other entities stored in the database, such as the software descriptors, the task descriptors and the related control parameters, similar attributes are stored.

In order to facilitate the development of new scientific methods and to allow users to test new software without interfering with the production environment test environments can be created. For each test environment a set of new database tables is created. A user working in a test environment has read only access to the production environment (and thus data) and read and write access to his test environment (see figure 4). These test environments were implemented by exploiting the functionality of the DBMS (synonyms and views) (Johnson, 1989).

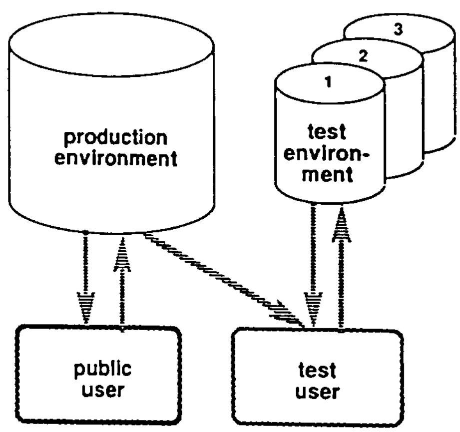

Fig.4. Implementation of test environments with the same functionality as the production environment

Activity Model

In figure 5 a simplified activity model for COMPASS is presented and three main components can be distinguished: the processing of data in order to correct the measured data for time dependent variations in the instrument and convert the raw data in physical units; the determination of the instrument response including the experimental background and finally the scientific analysis. Indicated in this activity model is that some of the activities are exercised mainly at one of the institutes (processing) or distributed over all institutes (scientific analysis). In addition it is shown that the COMPASS system is used for the analysis of the calibration data and the generation of simulated data.

A global overview of the three groups of activities is presented below.

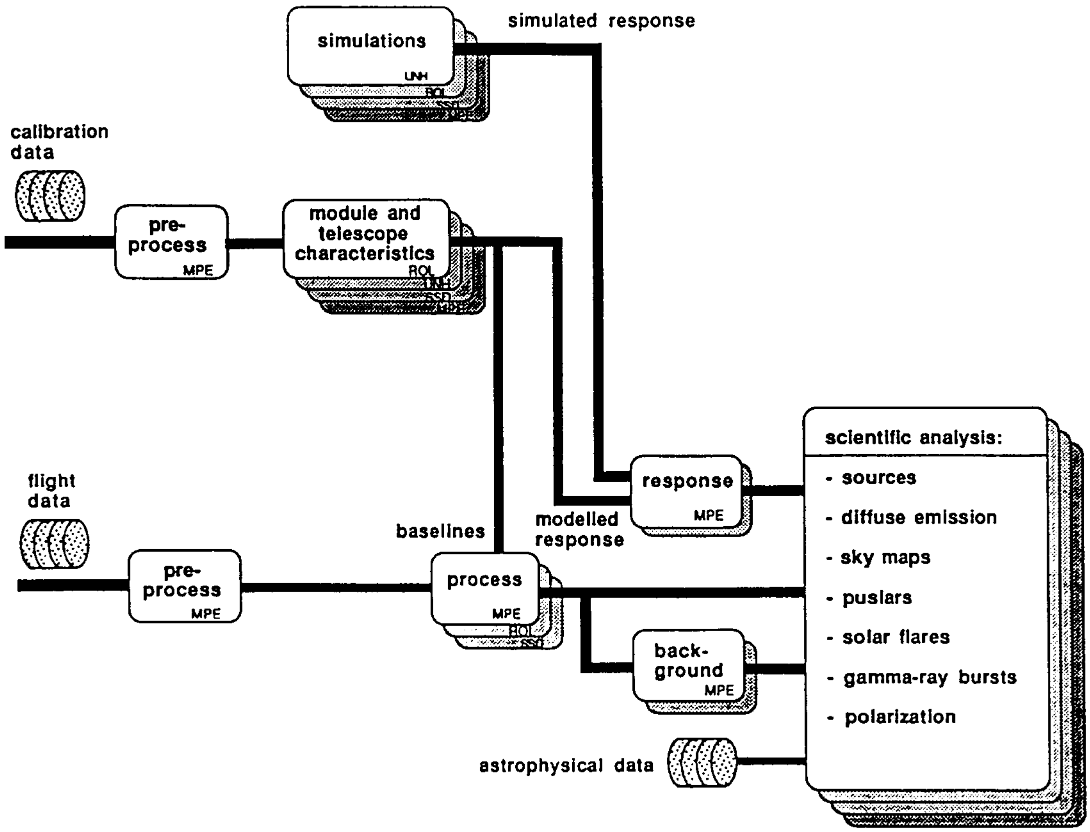

Fig.5. Simplified activity model for COMPASS indicating the main activities (boxes), the datastream (solid lines). The dataflow is from left (top) to right (bottom).

DATA PROCESSING

COMPTEL generates various types of raw data including in-flight calibration data, insttrument state of health data, burst data and event data (see also Schönfelder, these proceedings). The raw events are characterized by various pulse height outputs, a time-of-flight measurement, a pulse shape discriminator measurement and a time tag (with a accuracy of 1/8 msec). In addition COMPTEL may, when triggered by BATSE, generate burst spectra.

As part of the routine processing the state of health of the instrument is checked and the various operational modes of COMPTEL are separated (e.g. during the some parts of the orbit the instrument is switched off). Next the raw data are converted into physically meaningful units. The establishment of correction factors for gain fluctuations in the 154 photo multiplier tubes is a major goal in this step ("in-flight calibration"). Finally the energy deposit and location of each event in the detectors and the angle between the incoming gamma and the scattered gamma in the upper detector is determined ("event & burst processing"). The raw burst spectra are also converted into a proper energy scale. After this step, the resultant data products can be interpreted without particular knowledge about the operational conditions of COMPTEL and data measured during different periods can be combined.This is visualized in figure 6.

As is indicated in figure 6 the data processing which is planned for the flight data, was also applied to the prelaunch telescope calibration data.The main benefit of this is twofold: significant experience with routine data processing could be obtained before the actual launch and there is a consistent handling of telescope calibration data and the flight data. In addition COMPASS is used for the determination of the characteristics of COMPTEL.

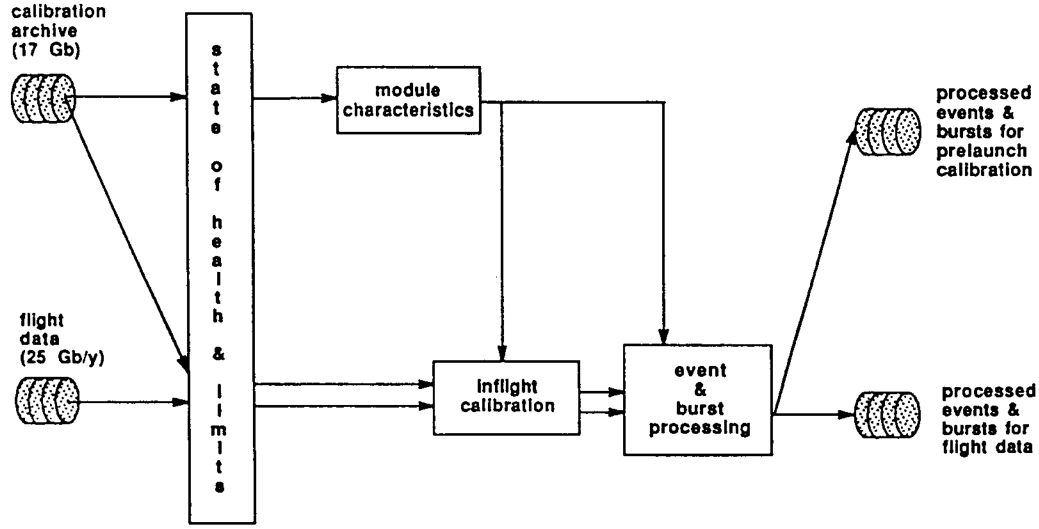

Fig.6. Routine data processing. The same approach is followed for prelaunch calibration data as for flight data.

INSTRUMENT RESPONSE

The second step is to determine the instrument response. This includes the determination of the point spread function, the energy response of the telescope and the energy response of the burst system using prelaunch calibration data (see also Diehl, these proceedings). As is indicated in figure 7 the telescope response can be extracted from different sources, e.g. based on the single detector response the telescope response can be calculated ("modelled telescope response"). This can be compared to the measured response during the prelaunch calibration period for selected energies and positions. Finally this information is supplemented by simulation data to interpolate and extrapolate to energies and angles not covered during the prelaunch calibration ("simulated response"). This Monte Carlo code has been based upon the CERN package GEANT. Special software was required for the energy response of the burst system as there is a clear angular dependence in this response (due to intervening material between the burst location and the detectors).

In figure 7 the point spread function is represented in a 3 dimensional "data space" given by the direction of the scattered photon between the upper and lower detector and the Compton scatter angle in the upper detector (Diehl, 1989). Since only the scatter angle in the upper detector is know, a source generates a cone-like pattern in this dataspace. In this space an important fraction of the scientific analysis will be performed.

Apart from this instrument response the instrumental and atmospheric background requires special attention. Various sources of information will be combined to make the best estimates for the instrumental background including:

* internal symmetries of the instrument assuming that the instrumental background is symmetrical;
* smoothing of measured data and the use of special observations (high latitude with a virtually empty sky);
* application of special data selections to enhance background features;
* study of features in the data as a function of the position in orbit including special modes of operation;
* simulations of radiation induced in the spacecraft.

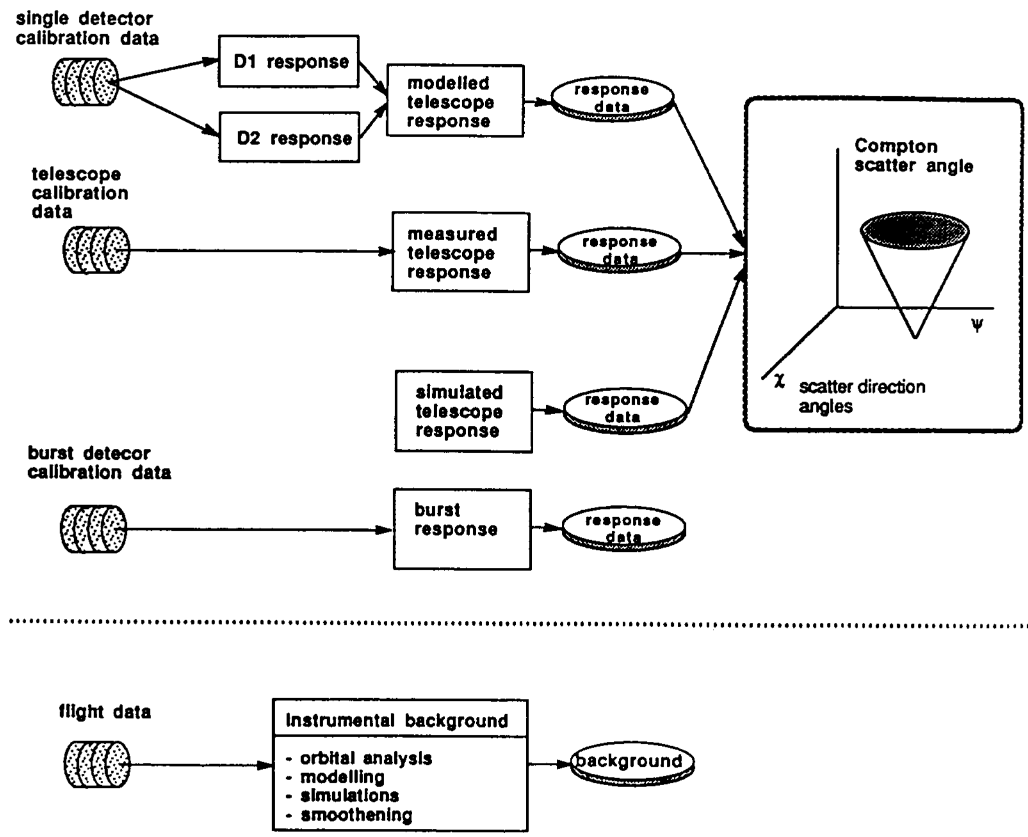

Fig.7. Instrument response determination

SCIENTIFIC ANALYSIS

For the various scientific objectives software has been developed and a global overview of these subsystems is presented in figure 8.

Sky imaging and source detection

Using the imaging capacity of the Compton telescope an image of the sky can be extracted from the event data using the maximum entropy method (Strong et al., 1990 and these proceedings). In addition sources can be searched for using the likelihood method. Because the point-spread function is sensitive to polarization of the gamma-ray radiation the degree of polarization of strong sources might be determined also. Once the strongest sources are recognized the remaining gamma-ray radiation will be studied in terms of the diffuse galactic gamma-ray emission (and of course an instrumental and atmospheric background component) using the maximum likelihood approach. Various model components will be included in the latter analysis including standard CO and HI maps and models forthe inverse Compton radiation.

Pulsar analysis

In addition to this analysis the event data can be searched for pulse emission (1/8 msec resolution). Various standard techniques are employed including the Z^2 test (Buccheri and de Jager, 1989). In order to enable the detection of pulsed gamma-ray emission several hundred pulsars will be regularly monitored by a number of radio observatories to provide timing parameters used in the COMPTEL analysis (Busetta, these proceedings).

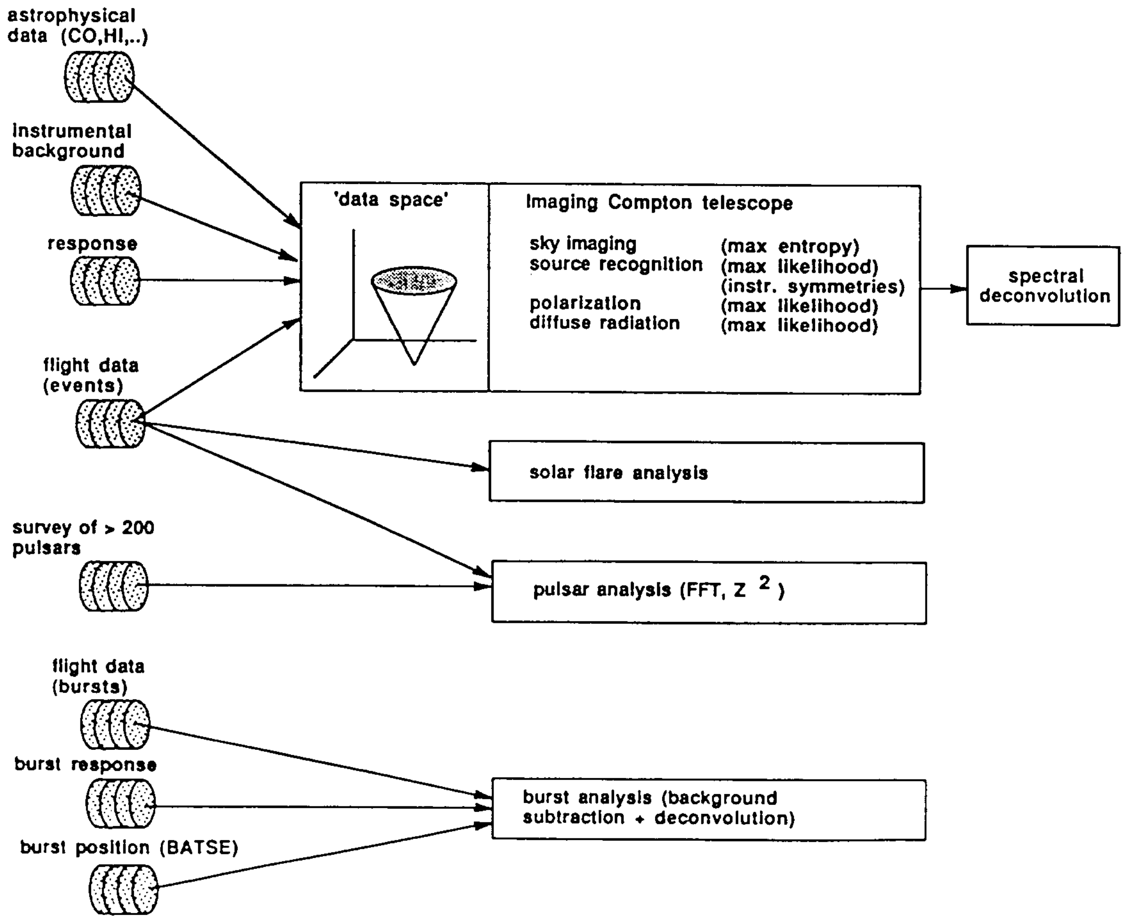

Fig.8. Global overview of the COMPTEL scientific analysis

Transient sources

When BATSE detects a gamma-ray burst a trigger will be sent to COMPTEL and COMPTEL will record (using two of its detectors) the energy spectra in the range from 0.1 to 12 MeV (Winkler et al., 1989). These spectra will be background subtracted and deconvolved using the maximum entropy method. For a proper deconvolution the position of the transient source is required.

Solar flares

When such a burst is coming from the direction of the sun it is assumed to be a solar flare and COMPTEL will, in addition to the recording of the gamma-ray burst, change its mode to the solar neutron mode (Lockwood et al., 1989). In this mode the time-of-flight measurement will be used to determine the neutron energy and results (energy spectra and flux time profiles) will be extracted.

REFERENCES

Buccheri, R. and de Jager, O.C., 1989, Detection and description of periodicities in sparse data, in "Timing Neutron Starts", H.Ogelman and E.P.J.van den Heuvel ed., Kluwer, Dordrecht

Busetta, M., et al., Pulsar Analysis within COMPASS, in these proceedings

Diehl, R., 1989., Data Analysis of the COMPTEL instrument on the NASA Gamma Ray Observatory, in "GRO science workshop", Greenbelt, USA

Diehl, R., et al., Response Determination of COMPTEL using Calibration Measurements, Models and Simulations, in these proceedings

Johnson, M., 1989. Die Verwendung von ORACLE als Systemkomponenten für Astronomie und Raumfahrt, in "Deutsche ORACLE Anwender Gruppe""

Lockwood, J.A., et al., 1989. COMPTEL as a Solar Gamma-Ray and Neutron detector, in "GRO science workshop"", Greenbelt, USA

Schönfelder, V., et al., 1984. IEEE Trans. Nucl. Sci. NS-31(1)

Schöonfelder, V., et al. The GRO-COMPTEL Mission: Instrument Description and Scientific Objectives, in these proceedings

Strong, A.W., et al., 1990. Maximum Entropy Imaging with COMPTEL data, in Proceedings of the XXI Cosmic Ray Conference, Adelaide, Australia

Strong,A.W., et al.., Maximum Entropy Method for Sky Imaging and Spectral Deconvolution, in these proceedings

Winkler, C., et al., 1989, Observing Cosmic Gamma-Ray Bursts with COMPTEL, in "GRO science workshop", Greenbelt, USA