Mem. S.A.It. Suppl. Vol. 3, 359

Memorie della
Supplementi

A Pipeline-Oriented Data Management System: Design and Implementation with an OODBMS *

N. Lama, C. Vuerli, F. Pasian

INAF/Osservatorio Astronomico di Trieste, Via G.B.Tiepolo 11, I-34131, Trieste, Italy e-mail: `family-name@ts.astro.it`

Abstract. A project is described, aiming at the design and implementation of DMC, a data management system built on top of a commercial Object-Oriented Database Management System (OODBMS) and particularly suited to handle data processing pipelining tasks. Although the design of the system is quite general, specific reference is made to its incarnation developed for the ESA Planck mission. Beating heart of the Planck IDIS information system, DMC is the tool chosen by the two Data Processing Centres (DPCs) as a common language for the data handling applications being developed. Particular reference is made to the design of the model, to the data structures and to the portability to other experiments of the main features of the service suite provided.

Key words. surveys – data processing – pipeline – catalogs – astronomical data bases: miscellaneous – data grids

1. Introduction

The aim of the project is to provide a pipeline-oriented data management system specialized with data products required by data processing modules (see Fig.1) . The underlying principle of DMC is to have a service tool through which a pool of applications can store and retrieve their data products from a number of geographically distributed data repositories. These concepts make the DMC a tool particularly suited to data grid applications Pasian et al. (2003).

Originally required within the framework of the Planck IDIS (Integrated Data and Information System) Working Group, the system has been designed so as to be fully portable to other experiments/missions/projects. Design details are given in Vuerli et al. (2001a), Vuerli et al. (2001b) and Lama et al. (2002).

2. DMC model: THE CORE

The DMC has a multi-tier software architecture which is object-oriented and is organised into independent layers: the DMCI (DMC Interface) and the physical implementation (see Fig.2). The DMCI is the User Interface (or Presentation Layer), a

Send offprint requests to: N. Lama

* Poster at http://sait.oat.ts.astro.it/MSAIS/3/POST/Lama_poster.jpg

Correspondence to: via G.B. Tiepolo 11, I34131 - Trieste - Italy

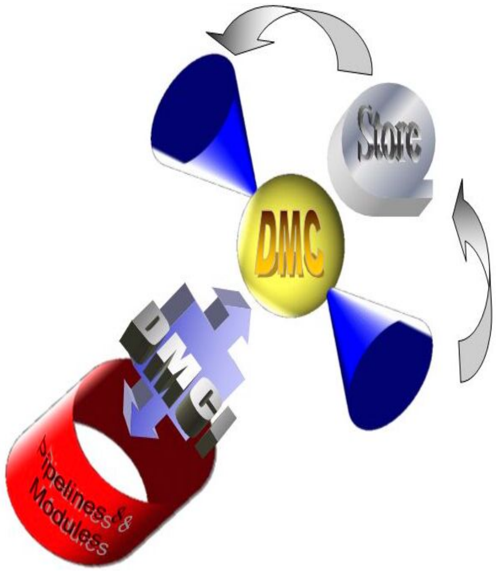

Fig. 1. Pipeline oriented Data Management System design

set of interfaces (API-like) through which scientific applications can exploit the DMC services. These interfaces hide the actual physical implementation from the user or the calling application. The DMC Physical implementation is the Data Services Layer which communicates directly with the Database.

Crucial objective was to hierarchically develop the DMC; the result is that the DMC is implementated by a Business Services Layer, related to application oriented objects, plus a DMC Core implementation. The latter is the Basic Services Layer, which implements the foundation for the data handling. It provides a set of basic services portable to all those experiments that are pipeline/module oriented. The core organizes data within internal objects aiming at speeding up data exchange between modules that belong to the same pipeline.

The DMC offers a sort of virtual directory service called Mission; each Mission maps all the products related to a particular phase/status of the project. In the specific Planck incarnation, the DMC services are integrated with the other tools that establish the IDIS information system, in particular, the Java wrapper (an interface to C/Fortran modules), the Process Coordinator (pipeline/module scheduler) and the Federation Layer (permissions handling, and user login services)

3. Data structures: OBJECTS

The DMC data model is composed of an inventory of objects suitable for the variety of data representations that might arise along the pipeline processing path. Here follows just a little taste of the implemented objects, making specific reference to Planck-specific ones, defined in Vuerli et al. (2001a) and Vuerli et al. (2003).

Store – Basic "service" object, store implements the capability to access different data repositories in parallel. It manages a list of Mission phase instances, each acting as a virtual directory and root for DMC objects browsing.

Pipeline/module configuration objects – Pipeline and modules signatures information (name, version,..) define the clients that can produce data. It is also the interface for any external process coordinating mechanism.

Pipeline, Module runtime description info – Client references to particular run of module/pipeline. They carry all the user parameters and track all the created/modified products on a data history log.

Telemetry – Users can store raw or processed telemetry and telecommands. The related services concern backup and basic lookup for Quick Look Analysis, data interpretation and creation of time series.

Time Series – The time series data structure of a signal is an array of data samples referring to a particular time interval. The Planck Time-Ordered Data (TOD) specialization are housekeeping parameters and scientific data measured by radiometer, each one characterized with different

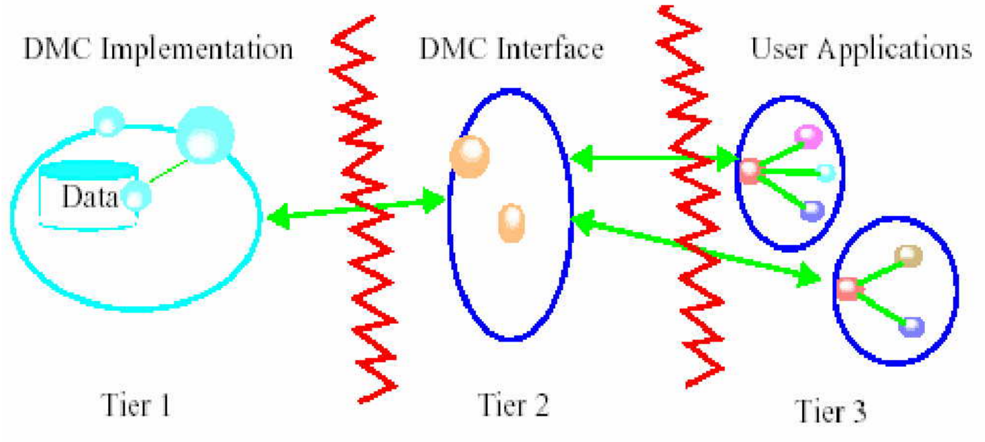

Fig. 2. DMC multi-tier layout

accuracy and ancillary information (data source, physical unit, etc.).

Pointing Data – Pointing Data are time series of observation pointing information. The data can refer to a particular sky pixelization scheme – HEALPix Gorski et al. (2002) is the standard for Planck – but DMC deals also with triplets of angles, directly.

Flags – Flag objects carry the information inferred from the data analysis procedures at the various steps of the data processing pipelines. For instance: values out of range, missing data, interesting data, interpolated data, etc.

Additional objects – Other objects may be relevant to the specific pipeline DMC is designed to be attached to. For Planck, objects currently under construction are: SpinAxis tables, detailed pointing information, beams characterization, Maps of observed sky, Spectrum and source Catalogues.

4. Data retrieval and lookup facilities: QUERIES

The key feature of the DMC tool is the possibility to browse data products following their processing path. Other data retrieval features are the following ones.

Alias lookups – Users can store aliases, i.e. mnemonic references to the products by which they can straightforwardly retrieve the related objects.

Version lookup – Objects can be versioned (tagged with a version info) and hence retrieved accordingly to version information.

Lookup by User, Module, Pipeline – The DMC can retrieve products of a particular module run, possibly within a certain pipeline run, and if the case, owned by a specified user.

Parameters – Often users wish to query data produced with specific values of certain parameters, or with values within a specified interval.

Flags – It is possible to query data tagged with particular flags information.

Time series queries – Time series can be queried by data source (radiometer, frequency channel, etc.) and time interval.

Additional services – Advanced lookup services are under construction: lookup time series by sky position, sub-map extraction, etc.

5. Implementation issues: TECHNOLOGY

5.1. COTS adopted

The programming language is JAVA (to ensure high portability), and JNI for ad hoc integration with non-java client modules. The OODBMS chosen for the project is the Versant database integrated with the J/VERSANT Interface (JVI). This choice has been supoorted Planck-wide.

5.2. Core implementation

The Data Model has been designed to reflect data usage and so as to be OODBMS-oriented. Data are organized within a graph structure modeling pipeline path. This has been done aiming at exploiting fast data browsing by link and preventing time expensive internal queries traversing the databases to find and evaluate starting point objects.

5.3. Transaction mechanism implementation

The DMC provides multiple database connection within sophisticated locking models (optimistic locking, session transaction shared among different data repositories).

5.4. Time series implementation

Time series are internally managed as segmented arrays. Data are buffered within data chunks forming a segmented array structure that allows the DMC to manage huge-sized data. This architecture can also be tuned for the particular parallel algorithm requirements. Data can also be stored in compressed form.

5.5. Object tagging/logging

The history of the processing path of data products is logged as a list of the client runs that contributed to it.

5.6. MAP design

For the map objects, sophisticated design issues for sky map data structures are being addressed, inspired by Geographic Information System (GIS) data system spatial datasets. Advanced data retrieval features are being designed.

6. Conclusions

In the future, a FITS file implementation of the DMCI will be developed. Modules that rely on DMCI will be able to store data within database structures or FITS files transparently.

A first version of the DMC has been released on February 2003; it is being currently upgraded while undergoing beta tests at the LFI DPC Bread Board Model pipeline integration site. The completion date for the full system is planned to be the end of 2003.

Acknowledgements. We wish to thank the Research and Science Support Department of ESA ESTEC for their testing activity, the Planck IDIS community and the LFI DPC Consortium institutes for comments and suggestions.

References

Vuerli C. et al., (2001a) PLANCK Int. Doc.: IDIS DMC Users Requirements Document, PL-COM-OAT-UR-004
Vuerli C. et al., (2001b) PLANCK Int. Doc.: IDIS DMC Data Model Specification, PL-COM-OAT-SP-001
Gorski K. et al., (2002) in: Astronomical Data Analysis Software and Systems XI, ASP Conference Proceedings, Vol. 281, Astronomical Society of the Pacific, p. 107.
Lama N., Mercier C. (2002) PLANCK Int. Doc.: IDIS DMC Architectural Design Document, PL-COM-OAT-AD-002
Pasian F. et al., (2003) these proceedings
Vuerli C. et al., (2003) in preparation