

THE GAIA PARAMETER DATABASE

J.H.J. de Bruijne, U. Lammers, M.A.C. Perryman

European Space Agency, Research and Scientific Support Department, ESA-ESTEC, 2200AG Noordwijk, The Netherlands

ABSTRACT

The parallel development of many aspects of a complex mission like Gaia, which includes numerous participants in ESA, industrial companies, and a large and active scientific collaboration throughout Europe, makes keeping track of the many design changes, instrument and operational complexities, and numerical values for the data analysis a very challenging problem. A comprehensive, easily-accessible, up-to-date, and definitive compilation of a large range of numerical quantities is required, and the Gaia parameter database has been established to satisfy these needs.

The database is a centralised repository containing, besides mathematical, physical, and astronomical constants, many satellite and subsystem design parameters. At the end of 2004, more than 1600 parameters had been included. Version control has been implemented, providing, next to a ‘live’ version with the most recent parameters, well-defined reference versions of the full database contents. The database can be queried or browsed using a regular Web browser (<http://www.rssd.esa.int/Gaia/paramdb>).

Query results are formatted by default in HTML. Data can also be retrieved as Fortran-77, Fortran-90, Java, ANSIC, C++, or XML structures for direct inclusion into software codes in these languages. The idea is that all collaborating scientists can use the database parameters and values, once retrieved, directly linked to computational routines. An off-line access mode is also available, enabling users to automatically download the contents of the database.

The database will be maintained actively, and significant extensions of the contents are planned. Consistent use in the future of the database by the Gaia community at large, including all industrial teams, will ensure correct numerical values throughout the complex software systems being built up as details of the Gaia design develop. The database is already being used for the telemetry simulation chain in ESTEC, and in the data simulations for GDAAS2.

Key words: Gaia; Reference Values; Database.

1. INTRODUCTION

The Gaia parameter database is a centralised repository of Gaia-related parameters. It contains, besides mathematical, physical, and astronomical constants, many satellite and subsystem design parameters. Examples are: the size of the astrometric field of view, the read-out noise of the MBP pixels, the adopted mass of Jupiter, the coefficient of thermal expansion of Silicon Carbide, the quantum efficiency (QE) and modulation transfer function (MTF) of the Astro CCDs as a function of wavelength, wave-front error (WFE) maps, etc. At the end of 2004, more than 1600 parameters had been included. This paper describes the main features of the database, including the naming scheme (Section 2), the top-level data categories (Section 3), the various access modes (Section 4), and the available output formats (Section 5). Section 6 concludes with future planned developments.

The database is accessible both directly at <http://www.rssd.esa.int/Gaia/paramdb> and indirectly through the main Gaia web page. Although the database is easy to use, some novel users may benefit from the many practical examples provided in the database user manual (Lammers 2004), available in Livelink and from the database entry page (see Figure 1). This manual also provides technical details on the software implementation and the various components of the database environment, issues only briefly covered in the Appendix of this paper.

2. NAMING SCHEME

The database employs a four-level hierarchical parameter naming scheme, reflecting the design of the satellite. The rationale behind this choice, and the naming scheme in general, is described in de Bruijne et al. (2003). All database entries have self-explanatory names, unambiguous descriptions, and full literature references. They furthermore carry — if available — a conventional (\LaTeX -format) symbol (e.g., ‘ ω ’ for the spin rate), the date of last modification of the parameter, and a status flag which can take the values ‘CONF’ (for confirmed), ‘TBC’ (when a value is present but from an obsolete reference source), and ‘TBD’ (when some value is present but without any

Gaia Parameter Database

maintained by the [Gaia Database Librarian](#)

Versions

live ▾

[ChangeLog](#)

Access

- [Search \(live\)](#)
- [Browse \(live\)](#)

Documentation

- [Basics](#)
- [Concepts](#)
- [User Manual](#)
- [Glossary](#)

Parameter search in live version

?		Level-1 is:	-Any-	More
?	and	Level-2 is:	-Any-	More
?	and	Level-3 is:	-Any-	More
?	and	Level-4 is:	-Any-	More
?	and	Class name matches:	<input style="width: 100%;" type="text"/>	
?	and	Kind is:	-Any-	More
?	and	Direction is:	-Any-	More
?	and	Case is:	-Any-	More
?	and	Status is:	-Any-	More
?	and	Parameter is:	<input checked="" type="radio"/> all <input type="radio"/> basic <input type="radio"/> derived	
?	and	Parameter is:	<input checked="" type="radio"/> all <input type="radio"/> scalar <input type="radio"/> multidimensional	
?	and	Description matches:	<input style="width: 100%;" type="text"/>	
?	and	Source matches:	<input style="width: 100%;" type="text"/>	
?	and	Parameter changed last: <	eternity ▾ ago	

Render output as: HTML ▾

Figure 1. The default Gaia parameter database search form, as it appears on the main parameter database entry page <http://www.rssd.esa.int/Gaia/paramdb>. Pressing the 'Search' button will return a form like the one displayed in Figure 2.

reference). The database has been developed and is being maintained in accordance with the guidelines set forth in the Gaia conventions and notations document (Bastian 2004). As an example, SI units, with appropriate prefixes, are consistently used throughout the database.

3. DATA CATEGORIES

All parameters in the database can be categorised either as **basic** or as **derived**. Basic parameters are entries, the value of which is set explicitly in the database ('hard-coded [numerical] values'). Derived parameters are entries, the value of which is derived within the database from other parameters (which, in turn, can be 'derived' as well). For all derived parameters, the relevant dependencies are provided in the form of formulae in the 'Expression' field (see Figure 2). All dependencies are expanded and processed — with double-precision arithmetic — with each new query: all dependencies are thus truly transparent.

All database parameters can also be categorised either as **scalar** or as **multi-dimensional**. Scalar data comprise integer-, double precision-, and string-type entries.

Multi-dimensional data are currently formatted in Flexible Image Transport System (FITS) format. If needed, other output formats, such as XDF, can be provided in the future.

4. ACCESS MODES

Access to the Gaia parameter database is currently limited to members of the Gaia scientific community at large (i.e., all members of the working groups). A username/password can be obtained from the database librarian, who can be contacted at 'gailib@rssd.esa.int'.

4.1. On-line Access

The database can be **queried** or **browsed** using a regular Web browser such as Internet Explorer, Netscape, or Mozilla.

The **search** form (see Figure 1) represents the four-level hierarchical structure of the database, and contains several selection options (e.g., free-text search in

Parameter search results (DB Version: –live–:2004–09–28T16:47:00)

Name	Value	Unit	Basic	Scalar	Status	Description	Source	Expression	LaTeX label	Last modified
Astro:AF:CCD_Number_AL	11		true	true	CONF	Number of CCDs in AF in the along-scan direction (i.e., number of CCD strips in AF)	Astrium, 15 January 2004, 'Astro focal plane assembly: requirements specification, compliance matrix, and verification matrix', GaiaFPA.SP.00147.T.ASTR, issue 3, revision 1, Requirement FPA120			2004-03-01T09:00:00
Astro:AF:CCD_Number	110		false	true	CONF	Number of CCDs in AF		CCD_Number_AL * Astro:CCD_Number_AC		2004-03-01T09:00:00

Figure 2. Example of a query result (obtained after pressing the ‘Search’ button in Figure 1) containing two parameters (lines). The various fields (columns) are explained briefly in Section 2. The same form, but containing only one parameter, is also the end-result of each browse operation (see Figure 3).

the ‘Description’ and/or ‘Source’ fields, selection of all ‘basic parameters’, selection of all ‘scalar parameters’, etc). Search results are returned as depicted in Figure 2. Multi-dimensional data files can either be downloaded to the user’s local disk or be displayed on the screen by means of a dedicated Java applet provided by Ivo Busko (STScI). By default, queries return up-to-date parameter values from the ‘live’ version of the database, although provision is made for retrieval of ‘obsolete’ reference versions of the database (see Figure 1, left column). A change log tracks major changes, such as the suppression of parameters, parameter name changes, etc (see Figure 1, left column).

The **browse** mode, accessible from the main entry page by clicking on ‘Browse (live)’ (see Figure 1, left column), has no search options and functions by means of clickable, expandable nodes which represent the database hierarchy (see Figure 3). Individual parameters are returned as for the search mode (see Figure 2).

4.2. Off-line Access

An off-line database access mode is also available, enabling users to download (selected parts of) the contents of the database either manually from the command line or automatically, possibly at regular intervals, by means of a script. Further details on this access mode, including practical examples, are available from the user manual (Lammers 2004).

5. OUTPUT FORMATS

Query outputs are formatted by default in HTML, convenient for visual inspection on the screen (see Figure 2). Other ‘human’ output formats, such as L^AT_EX and Portable Document Format (PDF), are also supported. Data can also be retrieved as Fortran-77, Fortran-90, Java, ANSIC, C++, or XML structures for direct inclusion into software codes in these languages; a Comma-Separated Value (CSV) output format caters for Microsoft Excel spreadsheets. Practical examples of how to link ‘local

software’ to the database are provided in the user manual (Lammers 2004).



Figure 3. Example of an intermediate step of a browse operation. The ‘Ground Segment’ hierarchical node has been expanded by the user, exposing all associated sub-levels (none in this case) and all individual parameters available at this hierarchical level. A mouse click on any one of these parameters (e.g., ‘Antenna_Diameter’) will return a result window like the one displayed in Figure 2.

6. DISCUSSION & FUTURE DEVELOPMENTS

The Gaia parameter database is an ESA-coordinated service to the Gaia community at large. The database has been released after an extensive contents review by the GST and various working groups. It is currently actively used in various environments, e.g., in the telemetry software package GTMSIM (Lammers 2005), in the GDAAS-related software GaiaSimu/GIBIS/GASS, and in on-board detection software.

The database is maintained actively, and significant extensions of its contents are planned (for example at payload and service module level). Several desirable functional enhancements have been identified, such as the introduction of: (i) a parameter dependency mechanism (similar to that provided by a ‘Makefile’) to guarantee that all dependencies, including hidden ones in, e.g., string-type parameters, are explicitly revealed; (ii) ‘interval arithmetics’ to automatically solve for ‘representation issues’; (iii) vector-type scalars (arrays); (iv) an automated email change notification system for working group leaders/coordinators; (v) hyperlinks to parameters in expressions for derived parameters; and (vi) pre-defined user-feedback forms.

Consistent use of the database by the Gaia community at large — including all industrial teams from the start of the non-competitive phase B2 — will ensure correct numerical values throughout the complex software systems being built up as details of the Gaia design develop.

ACKNOWLEDGMENTS

It is a pleasure to thank the Scientific and Local Organising Committees, chaired by Catherine Turon, Michael Perryman and Yves Viala, for orchestrating an enjoyable symposium. We thank the many members of the Gaia scientific community at large who provided constructive feedback on the database and encourage those unfamiliar with this tool to visit <http://www.rssd.esa.int/Gaia/paramdb>.

REFERENCES

- Bastian, U., 2004, ‘Reference systems, conventions, and notations for Gaia’, technical note Gaia-ARI-Bas-003 (version 4.0)
- de Bruijne, J.H.J., Lammers, U., Perryman, M.A.C., 2003, Gaia technical report Gaia-JdB-007 (revision 1)
- Lammers, U., 2004, Gaia technical report Gaia-UL-001 (revision 1.3)
- Lammers, U., 2005, ESA SP-576, this volume

APPENDIX: TECHNICAL ASPECTS

This Appendix provides some technical background information for the interested reader who wishes to gain a better understanding of the implementation aspects of the system. Additional background information can be found in the database user manual (Lammers 2004a).

The very core of the Gaia parameter database is a central, relational, SQL-based database system which holds the parameters and all associated attributes in a collection of logically linked tables. The open-source system MySQL¹ has been chosen as baseline. The on-line user is not directly exposed to the SQL-level but interacts with the system through either a dynamically generated query form or a tree-like browse interface (Section 4.1). Both of these interfaces are realised through a central software layer implemented in the PHP² scripting language. The software serves three main purposes:

- (i) dynamically creating the user interfaces (query form and tree-view);
- (ii) generating SQL-queries to the database system from the contents of the query form; and
- (iii) receiving matching result sets from the database and rendering the results into the desired output format.

The result rendering is a two-stage process. First, the query results from the database are converted on the fly into an in-memory XML³ structure by means of PHP’s DOM facilities. From there, the rendering into the final output formats is driven by a collection of XSLT (eXtensible Stylesheet Language Transformation) stylesheets on the server side. There is exactly one stylesheet per supported output format (Section 5), one of which is XML itself.

Using XML as intermediate data format is a very flexible approach since it allows the seamless conversion to almost any other format through the XSLT technique. Even client-side XSLT rendering of XML is possible (users wishing to experiment with this must enable XSLT processing in their browser). All XSLT stylesheets that control the conversion to the supported output formats are publicly accessible in a directory under the main URL. They may be used as templates for special client-side XSLT rendering of the XML output provided by the current system.

Version control is implemented through the widely-used Concurrent Versions Systems⁴ (CVS). The user is not exposed to the raw CVS level but the central management software provides a simple interface for the selection of all available content versions and their transparent retrieval (see the label ‘Versions’ in Figure 1).

¹<http://www.mysql.com>

²<http://www.php.net>

³<http://www.w3.org/XML>

⁴<http://www.cvshome.org>