User Manual 4.10 Data management system

De Wiki
Aller à : navigation, rechercher
User Manual 4.10 Data management system

Sommaire

- 1 Introduction
  - 1.1 Scope
  - 1.2 Javadoc
  - 1.3 Links
  - 1.4 Useful Documents
  - 1.5 Package Overview
- 2 Features Description
  - 2.1 Data providers
    - 2.1.1 Default provider
    - 2.1.2 Adding a provider
    - 2.1.3 Using the data management system
    - 2.1.4 Adding new data
  - 2.2 File formats
    - 2.2.1 Static Potential
    - 2.2.2 Variable Potential
    - 2.2.3 Geomagnetic coefficients
    - 2.2.4 Ionospheric coefficients
    - 2.2.5 Solar activity
    - 2.2.6 Pole data
    - 2.2.7 TAI-UTC shift
    - 2.2.8 Orbital data
    - 2.2.9 Third body ephemeris
- 3 Getting Started
- 4 Contents
  - 4.1 Interfaces
  - 4.2 Classes
- 5 Tips & Tricks
  - 5.1 Strengths
  - 5.2 Weaknesses

Introduction

Scope

The scope of this chapter is to present data management. This section presents the three modules of the data management system originally provided by Orekit :

- how does the data management system work?
- how to set it up?
- how to use it?

- how to add data to what already exists?
- what are the pros and cons of this system?

Javadoc

The data objects are available in the package `fr.cnes.sirius.patrius.data`.

Library	Javadoc
Orekit	Package fr.cnes.sirius.patrius.data

Links

Modèle:SpecialInclusion prefix=$theme sub section="Links"/

Useful Documents

Modèle:SpecialInclusion prefix=$theme sub section="UsefulDocs"/

Package Overview

The data loading process is organized through three main objects.

The `DataProvider` classes handle data sources. Each one of them has a particular type of source it can browse. The `DirectoryCrawler` performs a bottom-first search in a directory tree. The `ZipJarCrawler` works alike, but inside a compressed file. The `ClassPathCrawler` handles a list of data files and/or compressed files that are in the classpath (it can not search recursively like the `DirectoryCrawler` though). Finally, the `NetworkCrawler` works like the `ClassPathCrawler`, although in its case, it has a list of URLs instead of files. There is no limit to the number of DataProviders a program can use at once.

The Providers are listed and put to work through the `DataProvidersManager` singleton. This is the single point of access to the data management system. It contains a list of Providers that are queried every time a user needs data.

The various crawlers provide streams to the `DataLoader`. From these streams, the DataLoaders can reconstruct data that was stored in files (either compressed or not), even if some files come from different sources. These streams effectively separate the machine world from the program world, because they hide the former to the latter. Therefore, parsing data from a new format only means creating a loader, and being able to read another kind of file means creating a `DataProvider`. Note that the DataLoaders usually serve as a facade for the higher layers of the program.

Features Description

Data providers

Default provider

The data management system can use a system-wide property, `orekit.data.path`, as an entry point for default data. This default data must be file-based (either a file system entry point or a java resource) and either a directory or a zip/jar file. Setting a default provider is not mandatory, and must be done explicitly by :

- setting a value to `orekit.data.path`,
- calling `addDefaultProviders` on the data provider manager.

Adding a provider

Modèle:SpecialInclusion prefix=$theme sub section="Provider"/

Using the data management system

The data management system main operation is through the `feed` method. This method takes a `DataLoader`, and a regexp string matching the name of files the `DataLoader` is able to process. In this method call :

- the `DataProviders` list is traversed in the priority order.
- the first `DataProvider` providing a file matching the regexp is the one (and only) used to feed the `DataLoader`.

Adding new data

Modèle:SpecialInclusion prefix=$theme sub section="Adding"/

File formats

Patrius can read a variety of files:

- **Static potential files**

These files contains static potential coefficients up to a certain order and degree and are used to compute Earth (or any other body) static potential perturbation.

- **Variable potential files**

Theses files contains variable potential coefficients up to a certain order and degree and are used to compute Earth (or any other body) variable potential perturbation.

- **Geomagnetic coefficients files**

These files contains geomagnetic coefficients and are used to compute Earth (or any other body) geomagnetic field.

- **Ionospheric coefficients files**

These files contains ionospheric data and are used to compute Earth ionospheric correction.

- **Solar activity files**

These files contains solar flux and geomagnetic coefficients are used to get solar activity in order to compute drag perturbation.

- **Earth Orientation Parameters files**

These files contains earth orientation parameters (polar motion, LOD, etc.) used in frames transformation.

- **TAI-UTC shift files**

These files contains TAI-TUC shift.

- **Orbital data files**

These files are used to store orbital ephemeris

- **Third body ephemeris files**

These files are used to store third-body orbital ephemeris (Sun, Moon, etc.)

The following sections describe all file readable in PATRIUS. When the file format is not described anywhere, a short description is detailed below the tab.

Static Potential

Static potential coefficients files contains potential coefficients up to a certain order and degree. Warning: at very high order and degree (> 100), some numerical quality issues can appear and results may be degraded.

Data type	Data provider	Format	Reader
GRGS	Not direct link. This link provides GRGC potential coefficients [1]	[2]	GRGSFormatReader
EGM	[3]	See below	EGMFormatReader
ICGEM	[4]	[5]	ICGEMFormatReader
SHM	Not provided (not used any more, replaced by ICGEM format)	See below	SHMFormatReader

EGM file format (*Column text file*)

Column index:
1:Degree of coefficients
2:Order of coefficients
3:Tesseral-sectorial cosinus coefficient
4:Tesseral-sectorial sinus coefficient
5:Sigma applied to the cosinus
6:Sigma applied to the sinus

SHM file format (*Column text file*)

Column index:
2:Degree of coefficients
3:Order of coefficients
4:Tesseral-sectorial cosinus coefficient
5:Tesseral-sectorial sinus coefficient

Variable Potential

Variable potential coefficients files contains potential coefficients up to a certain order and degree. Warning: at very high order and degree (> 100), some numerical quality issues can appear and results may be degraded.

FES2004 is used to model oceanic tides.

Data type	Data provider	Format	Reader
GRGSRL02	[6]	[7]	GRGSRL02FormatReader
FES2004	[8]	See below	FES2004FormatReader

FES2004 file format (*Column text file*)

Column index:
1:Doodson number
2:Darwin number
3:Degree of coefficients
4:Order of coefficients
5:Sinus coefficient positif
6:Cosinus coefficient positif
7:Sinus coefficient negatif
8:Cosinus coefficient negatif
9:Positif coefficient
10:Epsilon positif
11:Negatif coefficient
12:Negatif epsilon

Geomagnetic coefficients

Geomagnetic coefficients fiels contains geomagnetic coefficients up to a certain order and degree.

Data type	Data provider	Format	Reader
COF	IGCRF data: [9] WMM data: [10]	See below	COFFileFormatReader

COF file format (*Column text file*)

Column index:
1:Degree of coefficients
2:Order of coefficients
3:g coefficient at position n,m
4:h coefficient at position n,m
5:dg coefficient at position n,m
6:dh coefficient at position n,m
g and h are the Gauss coefficients of main geomagnetic model (nT)
dg and dh are the Gauss coefficients of secular geomagnetic model (nT/years)

Ionospheric coefficients

Ionospheric coefficients files contains ionospheric coefficient to model the state of the ionosphere.

Data type	Data provider	Format	Reader
CCIR12	N/A	See below	R12Loader
USK(NEWUSK)	N/A	See below	USKLoader

CCIR12 file format (*Column text file*)

Column index:
1:Year
2:Month
3:Unused
4:Midval
5:Unused
6:Midday

USK file format (*Line text file*)

Solar activity

Solar activity contains solar flux and geomagnetic coefficients for a given timespan. ACSOL and NOAA files usually provide data for several years.

Data type	Data provider	Format	Reader
ACSOL	N/A	See below	ACSOLFormatReader
NOAA	N/A	See below	NOAAFormatReader

ACSOL file format (*Column text file*)

Column index:
1:Julian day since 1950
2:Seconds of the day (UTC)
3:Flux at JD + seconds
4-11 are the three-hours AP of the intervals

NOAA file format (*Column text file*)

Column index:
1:Day
2:Flux at the day
3:The background flux
4-11:the three-hours AP of the intervals
12:Year
13:Unused character flag

Pole data

These files contains earth orientation parameters (polar motion, LOD, etc.) used in frames transformation.
These data are usually valid of a timespan of several days/months, although there is no theoretical limit.

Data type	Data provider	Format	Reader
Bulletin B	[11]	[12]	BulletinBFilesLoader
EOP 05 C04	[13]	See below	EOP05C04FilesLoader
EOP 08 C04	[14]	See below	EOP08C04FilesLoader
Datas "Rapid and Prediction" (TXT)	[15]	[16]	RapidDataAndPredictionColumnsLoader
Datas "Rapid and Prediction" (XML)	[17]	Same datas than the column file but write in the xml format	RapidDataAndPredictionXMLLoader

EOP 05 C04 and EOP 08 C04 files format (*Column text file*)

Column index:
1:Date at 0h(UTC)
2:Modified Julian Day
3:x(")
4:y(")
5:UT1-UTC(s)
6:LOD(s)
7:DPsi(")
8:DEps(")
9:x error (")
10:y error (")
11:UT1-UTC error (s)
12:LOD error(s)
13:Dpsi error(")
14:DEpsilon error(")
x and y are the coordinales of the Celestial Ephemeris Pole.
UT1: Universal time, the time of the earth clock
LOD: Length Of Day is the excess of revolution time

TAI-UTC shift

These files contains TAI-TUC shift. Official data contains shift since the beginning of space era (1961).

Data type	Data provider	Format	Reader
Gap TAI-UTC	[18]	See below	UTCTAIHistoryFilesLoader

TAI-UTC file format (*Column text file*)

Column index:
1:Begining date-Ending date
2:Difference TAI-UTC between the begining date and the ending date

Orbital data

These files are used to store orbital ephemeris.
TLE provides orbit at one date. Ephemeris is then retrieved thanks to SGP4/SDP4 propagation model.
SP3 files provide orbit over an extended time span (ex: 1 day).

Data type	Data provider	Format	Reader
TLE	[19]	[20] TLESeries	
SP3	GPS and GLONASS only [21]	[22]	SP3File

Note: * SP3 files can be written by anyone. They are however mainly used ton provide GPS and GLONASS ephemeris.

Third body ephemeris

Third body ephemeris files contains ephemeris for third bodies (Moon, Sun, Jupiter) over an extended time span (ex: 1 or several days).

Data type	Data provider	Format	Reader
JPL	[23]	[24]	JPLCelestialBodyLoader

Getting Started

Modèle:SpecialInclusion prefix=$theme sub section="GettingStarted"/

Contents

Interfaces

The data package includes the following interfaces :

Data

Class	Summary	Javadoc
DataLoader	Interface for loading data files from DataProvider data providers.	...
DataProvider	Interface for providing data files to DataLoader file loaders.	...

Classes

The data package includes the following classes :

Data

Class	Summary	Javadoc
DataProvidersManager	This class is the single point of access for all data loading features.	...
DirectoryCrawler	This class handles data files recursively starting from a root directories.	...
NetworkCrawler	This class handles a list of URLs pointing to data files or zip/jar on the net.	...
ZipJarCrawler	This class browses all entries in a zip/jar archive in filesystem or in classpath.	...

Tips & Tricks

Strengths

- Lightweight implementation. The providers never load data, they merely provide streams on demand to the loaders.
- Scalable for using data from several heterogeneous sources.
- Scalable for new data types : the user only needs to create a new `DataLoader` implementation to use a new data type in this system.

Weaknesses

- The user must be aware the data loading overhead happens any time a `DataLoader` is fed, so the user must manage its loaders so that they are fed only once.
- Several sources for the same type of data cannot be used, since only the last provider added is used to feed data to a loader - unless the user manages the providers list accordingly, knowing one can only add elements or reset the whole list.
- The regexp is the only way to match a data file and a `DataLoader`.
- As of today, the data management system is a thread-hostile singleton : a multithreaded application shares the same providers list for all threads, and it may deadlock on a concurrent access!

Menu de navigation

Outils personnels

Espaces de noms

Variantes

Affichages

- Lire
- Voir le texte source
- Historique
- Exporter en PDF

Plus

Rechercher

Rechercher Lire

PATRIUS

- Welcome

Evolutions

- Main differences between V4.15 and V4.14
- Main differences between V4.14 and V4.13
- Main differences between V4.13 and V4.12
- Main differences between V4.12 and V4.11
- Main differences between V4.11 and V4.10
- Main differences between V4.10 and V4.9
- Main differences between V4.9 and V4.8
- Main differences between V4.8 and V4.7
- Main differences between V4.7 and V4.6.1
- Main differences between V4.6.1 and V4.5.1
- Main differences between V4.5.1 and V4.4
- Main differences between V4.4 and V4.3
- Main differences between V4.3 and V4.2
- Main differences between V4.2 and V4.1.1
- Main differences between V4.1.1 and V4.1
- Main differences between V4.1 and V4.0
- Main differences between V4.0 and V3.4.1

User Manual

- User Manual 4.15

- User Manual 4.14
- User Manual 4.13
- User Manual 4.12
- User Manual 4.11
- User Manual 4.10
- User Manual 4.9
- User Manual 4.8
- User Manual 4.7
- User Manual 4.6.1
- User Manual 4.5.1
- User Manual 4.4
- User Manual 4.3
- User Manual 4.2
- User Manual 4.1
- User Manual 4.0
- User Manual 3.4.1
- User Manual 3.3

Tutorials

- Tutorials 4.15
- Tutorials 4.14
- Tutorials 4.13.5
- Tutorials 4.12.1
- Tutorials 4.8.1
- Tutorials 4.5.1
- Tutorials 4.4
- Tutorials 4.1
- Tutorials 4.0

Links

- CNES freeware server

Navigation

Outils

- Information sur la page
- Citer cette page

- Dernière modification de cette page le 10 novembre 2022 à 10:51.

- Politique de confidentialité
- À propos de Wiki
- Avertissements

-