

Development of a web ephemeris server

T. Alonso Albi

Informe Técnico IT-OAN 2010-13

Change Record

Version	Date	Author	Remarks
1.0	5-Ago-2010	T. Alonso Albi	First version
1.1	2-Oct-2010	T. Alonso Albi	Revision

Contents

1	Introduction	3
1.1	The JPARSEC library	3
2	Implementation	7
2.1	Design and structure	7
2.2	The HTML and Javascript code	9
2.2.1	The sky and solar system simulator applet	9
2.2.2	Developments in JPARSEC and dependency adaptation	10
2.2.3	Executing Java programs in a server	11
2.2.4	Returning web pages with charts	12
2.2.5	Geolocalization of the user and date selection	13
3	Testing and installing	14
3.1	Memory requirements in the applet	14
3.2	Comparing results with the Anuario	14
3.3	Installing at OAN and IGN	14
3.4	Updating data	14
3.5	Notes about precision, time span validity, and other limitations	15

Abstract

In this document the implementation of a web ephemeris server is described shortly. This server provides accurate astronomical ephemeris for solar system bodies, clear charts of the sky, and allows an interactive simulation of astronomical phenomena like eclipses. The server is installed at the website of the Observatorio Astronómico Nacional, and a link to it is provided at Instituto Geográfico Nacional (www.ign.es).

1 Introduction

I started developing astronomical programs at twelve using different programming languages like Basic, PowerBasic, or Fortran, among others. Without the great effort and time spent in these previous experiences it would have been impossible to consider the development of a web ephemeris server. Another point is the decision I took five years ago of developing all my work using the Java programming language. The idea was to reimplement everything I did in the previous 12 years in Java, and to connect all this astronomy with the developments in astrophysics during my thesis. The result is a consistent and well-documented library of astronomical tools for research and general astronomical tasks.

1.1 The JPARSEC library

The JPARSEC library is currently structured in 33 packages, with 282 class files, and around 217 000 lines of code. There are around 6000 lines of comments and 30 000 lines of documentation (JavaDoc). An additional 60 000 lines of code comes from support classes required to obtain ephemerides by means of ELP2000 and S. L. Moshier algorithms¹. Most of the theories developed so far in celestial dynamics are implemented:

- **Planetary ephemerides:** JPL ephemerides (DE200, DE403, DE405, DE406, DE413, DE414, with the required files for the time span between 1900 and 2100), the Steve L. Moshier fit to DE404, VSOP87 (complete and truncated versions), ELP2000, and Series96. The positions for other bodies (comets, asteroids, trans-Neptunian objects, space probes) can be obtained by solving their orbital elements, for elliptic, parabolic, or hyperbolic orbits.
- **Natural Satellites ephemerides:** the 2007 Martian satellites theory (Mars), E2x3, E5, and L1 (Jupiter), Dourneau and TASS1.7 (Saturn), GUST86 (Uranus). The position of Triton follows the Explanatory Supplement of the Astronomical Almanac. The position of every moon (including dwarf satellites) can be obtained by solving the orbital elements provided by JPL.
- **Other ephemerides:** stars (FK4, FK5, ICRS), artificial satellites, double stars.

¹Another 20 000 lines of code (without considering in this case the HTML documentation) corresponds to end-user programs with GUIs, mainly astrophysical models.

The JPARSEC library includes a large number of other features that are summarized next.

- Planetary ephemerides includes physical ephemeris, orientation of the disk following IAU2000 resolutions, subsolar points, any rise/set/transit times (civil, or nautical and astronomical twilights), constellation, and charts numbers for several books. For natural satellites information is given about the current status (transits, shadows, eclipses), whatever the theory being used. Mutual satellites phenomena are also calculated, with an estimate of the fraction of the disk occulted/eclipsed, and the effect in the apparent magnitude.
- IAU resolutions are supported, and results are available for any output equinox, any kind of coordinates (geometric, astrometric, apparent), any analytical theory, different frames (J2000 or ICRS, FK4 supported on stars). Reduction from rectangular coordinates can be performed in multiple ways, using IAU1980 or IAU2000 methods for precession, nutation, and Greenwich mean sidereal time, as well as other methods used during the years by DExxx ephemeris or IAU resolutions. IAU2006 and 2009 resolutions will be supported soon.
- Almost fully updated, using the last recommendations for precession (Capitaine et al. 2003) mentioned by Hilton et al. 2006, and IAU resolutions about Pluto. Numerical integration theories updated, with the inclusion of JPL DE414 integration or the Martian ephemeris theory by Lainey et al. 2007. Accuracy up to the milliarcsecond level in the reduced coordinates, limited only by the precision of the analytical theories.
- Support for orbital elements from the Minor Planet Center, Celestrak, and also from the commercial program SkyMap. Orbital elements for planets available from the VSOP solution or calculable for any theory solving the orbit from a position and velocity vector. The user can easily port the format of orbital elements from any other program.
- Any call to an ephemeris method has the same input/output parameters for an easy use or input change. Common input parameters are an EphemerisElement object (ephemeris properties), and ObserverElement object (observer position and atmospheric conditions, from a built-in database of more than 4000 cities or 1200 observatories), and a TimeElement object (calculation time, which can be set either in local time, universal time UT1 or UTC, terrestrial time, and barycentric dynamical time). Calculation time can also be given in any calendar used or being in use.
- Compatible with Virtual Observatory protocols. VizieR catalogs, Simbad resolving, CDS, UCD, ADS abstracts, SDSS queries, among others, are supported. Plastic interface can be used to send or receive data (Virtual Observatory tables) from other applications.
- Very powerful graphical capabilities, including the best planetary rendering method freely available. Simultaneous sky and planetary renderings are supported, with a large number of options and the possibility of showing stars up to 16th magnitude by means of internet queries to UCAC3 catalogue (limiting magnitude of 10 without Internet connection). Complex charts are supported through a very simple interface to JFreeChart, that

allows to create sofisticate charts in a couple of lines of code. Charts can be exported to JPG, PNG, BMP, GIF, SVG, PDF, and EPS. At the same time, x-y dispersion charts can be exported to GILDAS package, by means of an automatic script generation algorithm. GILDAS can also be controlled in background using pyGILDAS, allowing any user to execute their own scripts from models. Contour charts in 2d can be created using SGT library. 3d charts are also supported with a similar interface to JMathPlot and VISAD.

- A lot of methods for coordinates transformations, for general astronomical calculations, and for input/output tasks with the data contained in catalogs. Especially remarkable is the HTML and LATEX support, that allows to export query results or tables in these formats, or even to create full pdf documents with easy. Available math operations include numerical integration, derivatives, root search, ponderation, interpolation, function fitting, and dynamical evaluation of expressions. A unit conversion utility is also available. Complex numbers and matrix operations (solving systems, eigenvalues) are supported.
- Astrophysical models available that can be used for scientific research. Among them an LTE/LVG gas radiative transfer model with 3d visualization of cubes, Kurucz 1992 LTE models for stellar atmospheres, a complete model for dust opacity based on Draine (using a Mie scattering code), or rotational diagrams. JPL and COLOGNE databases of molecular spectroscopy are integrated in the package, with no need of Internet connection to use them. The circumstellar disks model of Dullemond is also available, and is integrated with Virtual Observatory tools to provide an almost automatic SED fitting tool, taking the photometry automatically from Vizier catalogs and the stellar properties from Siess evolutionary tracks for pre-main sequence stars. A Java version of RADEX is also provided, including all the molecules/atoms currently available.
- Many more advanced operations very useful for developing applications, like a complete data model system that allows to perform error calculations by applying maths to physical measurements or unit transformations, technical calculations for single dishes (30m) and interferometers (PdBI), calculations with astronomical instruments (telescopes, oculars, CCDs), support to read and write spectra and cubes directly in the .30m and .lmv formats and .fits (as an output from a model for example), advanced net functions (servlets, mails, ftp access), or development of dialogs with integrated functions in buttons or lists.
- Automatic updates supported for any information used in the package that is subject to change: orbital elements of minor bodies, the list of observatories, Earth orientation parameters, solar spots database, and JPL and COLOGNE databases.
- It is written in Java 1.6 (most features compatible with Java 1.4), so it can be used in any java-capable device.

The library is the core of all the work I have done in the last few years, and it cannot be described deeply in this document. Source code is freely distributed under the terms of the GPL license, and is available at <http://conga.oan.es/%7Ealonso/doku.php?id=jparsec>. First years were mainly dedicated to development, but progresively most of the features have come to production phase, so the development of an ephemeris server (as other projects like astrophysical models) have become feasible.

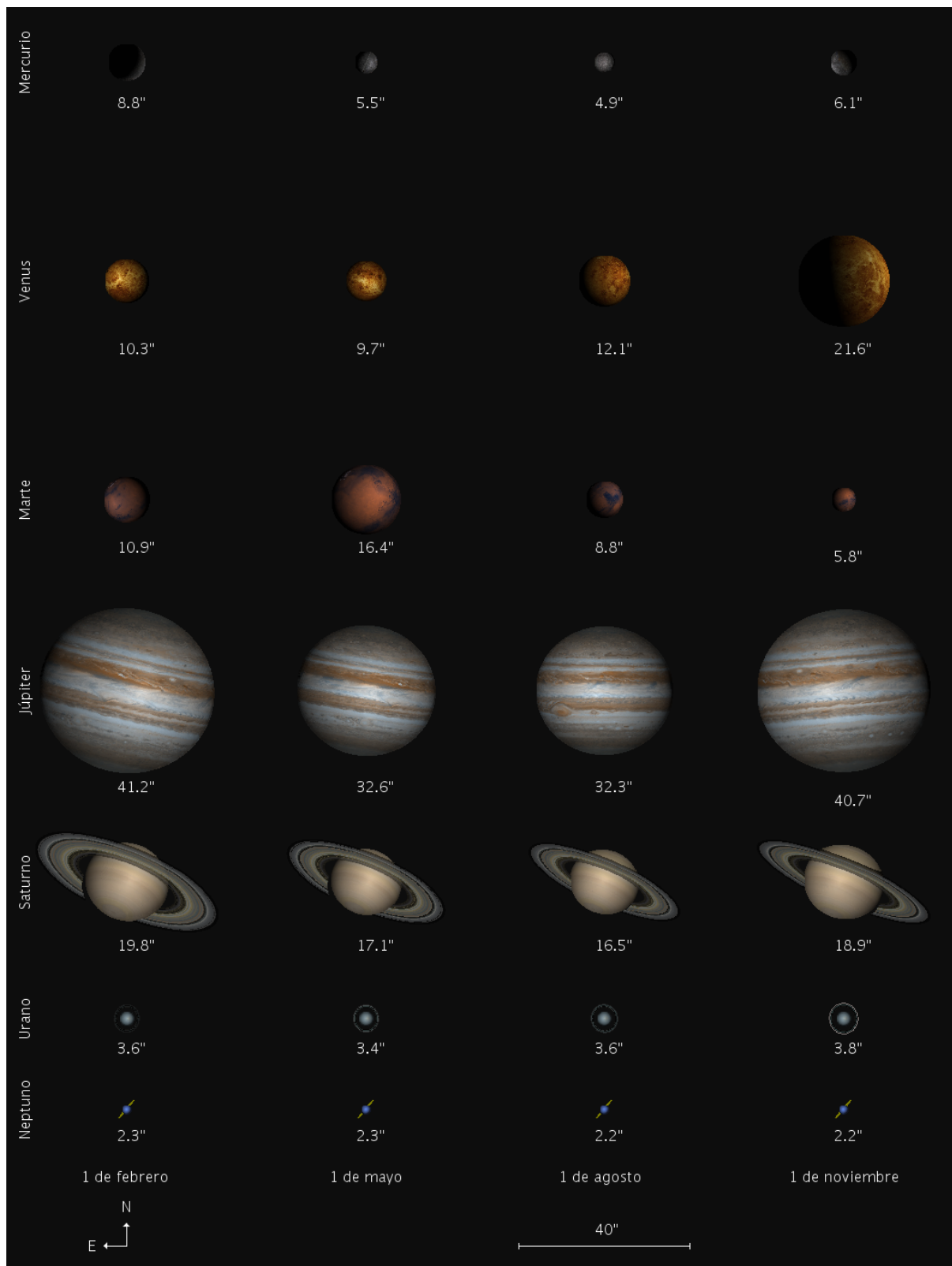


Figure 1: An example of the planetary rendering capabilities of the JPARSEC library, with the apparent vision of the planets during year 1000 B.C. This chart has been generated directly from the web ephemeris server, and a similar one will be included in the next editions of the Anuario.

2 Implementation

The web ephemeris server was developed in June, 2010. Ephemeris calculations and chart generation comes from previous works, whereas the new work is mainly the web interface and some Java development to return the results as a web page. In this section I will analyze how the server was implemented and the different decisions taken during its development.

2.1 Design and structure

The web ephemeris server was initially designed to be installed at the website of the Instituto Geográfico Nacional (IGN), so the structure of this website was inevitably taken as rule by copying some of the files of the server before starting writing HTML code.

The design of the ephemeris server itself was done after listing all services and different kind of calculations that it should implement. As with other services implemented at IGN, a short top menu composed by three main options was included. When clicked, these options were spread out (or spread in if they were clicked again) to show the different sections. The design of the server gave preference to interactive visualization of the sky and ability to retrieve in one click a large amount of information and astronomical events. The typical option to retrieve ephemeris by means of a form, which is usually the only option available in other ephemeris servers, was also present, but it was a secondary possibility.

- **Planetarium:** the first option gave access to an interactive program to simulate the sky. By taking advantage of latest Java technology an advanced simulation of the sky and the planets was shown, with great freedom and performance. This program was also used to simulate eclipses and other events. A link was given to download the applet to use it without Internet connection.
- **Astronomical events:** this second option showed a page with a chart of the sky towards south, the date in different calendars and time scales, the value of some astronomical parameters, the eclipses during this year and the next, moon phases, equinoxes and solstices, and planetary phenomena (oppositions or conjunctions, charts with apparent sight and trajectories, and orbital elements, among others). For some of the previous phenomena some links were provided to simulate them using the applet, and also some other links were available at the bottom of the page to show detailed ephemerides using the third option. The page was generated for Madrid and for the current date, but a little form below it allowed to generate it for any other instant, and for Madrid or Tenerife.
- **Astronomical ephemeris:** this option allowed to query an specific calculation by filling a simple form with the observer position, the calculation time (a date or interval of dates), and the object name. Some special features like the geolocalization of the user to set its geographical position, or the calendar component to introduce dates with ease were included. Ephemeris were available for planets, the Sun or Moon, natural satellites, comets, asteroids, trans-Neptunian objects, dwarf planets, artificial satellites, or space probes.

After some testing me and other astronomers realized rather quickly that the second option (sky events) should be the only one. It already contained links to simulate the sky, so the first

option was covered, and it seemed natural to add at the end another link to a separate form to retrieve ephemerides. The main page is now called `servidorEfem.php` and contains the calls from PHP to Java to directly retrieve the main frame. The separate page for the form itself was improved by automatically retrieving the name of the closest city and its elevation when clicking on a Google map. One limitation of having the form in another page is the impossibility of having the url of the server completely hidden, but now the idea is to migrate the ephemeris server from conga server to the server where the main OAN pages are located. IGN could load this page inside an `iframe` object so that everything would be quite hidden.

Besides a lot of minor esthetic improvements and simplifications (for example, the FreeHEP library was discarded), two major computations were added: the transits of Mercury and Venus through the disk of the Sun, and the events for the satellites of Jupiter and Saturn. In the last case the implementation includes both the events caused by the mother planet (transits, shadow transits, eclipses, or occultations), and mutual events between two satellites (occultation/eclipse of one satellite due to another). Both cases can be simulated with the applet straightforward.

Another service was developed to automatically update the dependencies by downloading and packing certain files, so the ephemeris server can be updated anytime. Since this service didn't work online due to a security manager issue (and to do this is indeed not secure), the service was replaced by a script called `update.sh`. However, there are some limitations in this process (the longitude of the Great Red Spot in Jupiter, and the leap seconds and TT minus UT1 difference), so each year or two years three files in `sky.jar` and `time.jar` will require an update by hand. Time data can be updated using the Anuario, whereas the Great Red Spot longitude can be updated from <http://jupos.privat.t-online.de/rGrs.htm>.

I developed two versions of the server with different levels of precision and performance. The one to be implemented will depend on the speed of the server and other considerations. The fastest version uses Moshier algorithms always, both in the ephemeris server itself and the applet. The startup time for the applet is between 2-3 seconds in Firefox and 1 second or less in Konqueror (since it doesn't use a security manager), and the time required to load the main page for the ephemeris server is around 12 seconds. In the version with better precision the ephemerides are calculated using JPL DE406 between 1900 and 2100, with a precision in the milliarsecond level², after that interval the Moshier fit to DE404 is used. This gives a maximum error of about 2 arcseconds for planets in year 1000 B.C. For the Moon the ELP2000 theory is always used outside that interval. When a set of ephemerides is to be retrieved in certain time span (resulting in a table with more than one row), the algorithm is automatically changed to Moshier to obtain a decent level of performance. However, the performance in this version is relatively poor, main page loads after 24 seconds and the applet needs 12 seconds to start. In addition, in this version the applet needs to download 30 MB of data instead of 20.

These speed results are an important limitation, so the current version to be implemented is the fastest one.

²JPL DE406 is used since it is the algorithm currently adopted by the IAU, although the differences with later versions like DE414/18/21 exceeds the 0.01" level. In addition, this level of precision is difficult to test through this server and against others, without using the JPARSEC library. For example, the IMCCE uses an old method (IAU 1980 set of algorithms) to reduce coordinates, which are supported by the JPARSEC library so they can be reproduced to the milliarsecond, but in this work the latest algorithms recently adopted by the IAU are used instead. JPL Horizons uses slightly more updated algorithms, but gives less digits in the results.

2.2 The HTML and Javascript code

The main page is called `servidorEfem.php`. It contains the PHP code to call Java programs from the server, and a little form that allows to generate the main page for different dates and locations. The main page is generated by a Java service that calculates, among other data, the date in different calendars, some astronomical parameters, lunar phases, eclipses, and includes some links to retrieve additional tables and charts using other services.

This main page also includes links to simulate the astronomical events using an applet. The applet is loaded using the file `Firmamento.html`, that contains the necessary Javascript code to read parameters for the applet in case they are provided.

At the end of the main page a link is provided to another page to fill in a form to query ephemerides, called `formulario.php`. It has three sections: the input of the geographical position to refer the ephemeris to, the input of the date and time or interval of dates, and the object name to calculate ephemeris. This file uses PHP code to obtain the current user IP, and then a Java program from JPARSEC library is executed to obtain the translation of this IP to its geographical position, so that the first section of the form is filled automatically. The second section is filled with the local time of client's computer.

2.2.1 The sky and solar system simulator applet

To show the sky and provide the user with an adequate tool to simulate astronomical phenomena dynamically and interactively an advanced applet was developed. This applet is based on a planetarium software currently in development, and implements latest improvements in sky and planetary rendering achieved in work done as previous contributions to the *Anuario*.

The program allows to select a country and a city in two lists, and the date and time in local time. The projection can be selected between stereographical, cylindrical, polar, and spherical. The coordinate system between equatorial, ecliptic, galactic, and horizontal. The object to focus can be selected between the Sun, the Moon, the planets, and the constellations. The field of view is selected using a given telescope and ocular from two lists. There are some visualization options to show planets with textures, with north/south and rotation axes, and with all satellites. In the last case the specific algorithm to show satellites accurately for each planet is substituted by a general algorithm that uses orbital elements from JPL, with less precision but more number of satellites. There are also some buttons to accept the changes introduced and to calculate the new chart, to search for a given object, or to export the chart to a file. Allowed formats are the classical ones like PNG or JPG (vector graphic formats like EPS or SVG was supported using FreeHEP library, but later discarded to reduce the amount of data to be downloaded). The date is selectable between years -1350 (1350 B.C.) and year 3000. Year zero is not accepted. The algorithms selected are the best possible combination in terms of precision and performance, and generally the most up-to-date. This and other information is provided in the ephemeris server, below the applet itself, and can be seen by dragging downwards the bar on the right.

There are more selectable options in the input parameters of the applet, that can be modified by editing the HTML file `Firmamento.html` that launches it. This can be done locally by downloading the applet, or remotely by including in the browser some parameters like `"?loc=Madrid&coordSystem=1"`. The date and time, country and location, telescope and ocular, central object, projection, and coordinate system can be modified in this way (using

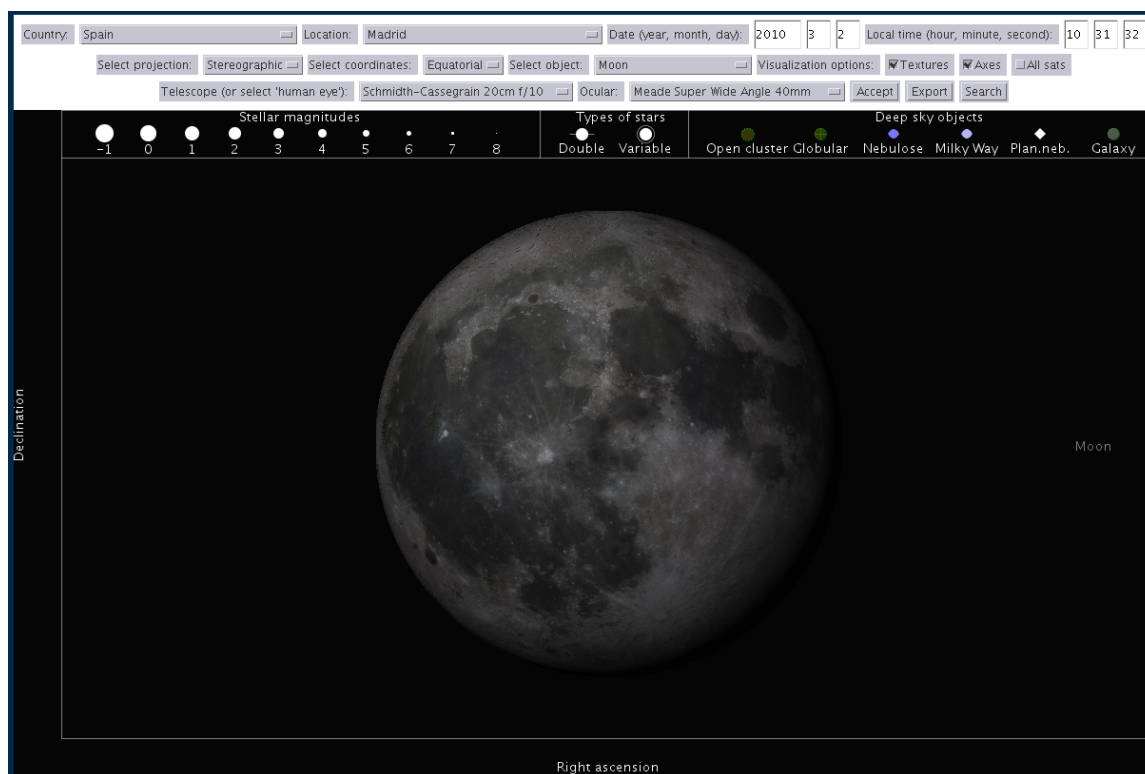


Figure 2: Screenshot of the English version of the applet. The server is implemented in Spanish.

Javascript) to properly show different kind of astronomical events. Another option is to change the color mode to show the applet with black or white background, or even in anaglyph mode to show depth levels (distances to the stars). This feature requires 3d glasses (either red-green or cyan-red ones). In the first option of the astronomical events section, just below the chart showing the current sky towards south, some links are provided to simulate the sky in any color mode.

These parameters provide the necessary flexibility to simulate different kind of astronomical events.

2.2.2 Developments in JPARSEC and dependency adaptation

Most of the development was done in the last four years for other purposes, but some was required to develop the server. This was the case of planetary events (Lunar phases, oppositions and conjunctions, perihelion and aphelion, equinoxes and solstices, and eclipses), which use the *Astronomical Algorithms* book by Jean Meeus. The opposition, conjunction, perigee, apogee, perihelion, and aphelion events were corrected using the Moshier ephemerides (a fit to JPL DE404), and the eclipses implementation by Jean Meeus was used instead of the one previously present in the library due to its better performance.

The files that implements the server include the main dependencies of the JPARSEC library, and are the same files as those for the applet. These dependencies includes .jar files to read orbital elements, earth orientation parameters and time data, telescopes and oculars, Moshier


```
graphicsio-ps-2.1.1.jar:/home/alonso/public_html/servidorEfem/lib/freehep-graphicsio-svg-2.1.1.jar:/home/alonso/public_html/servidorEfem/lib/freehep-graphicsio-2.0.2.jar:/home/alonso/public_html/servidorEfem/lib/freehep-swing-2.0.3.jar:/home/alonso/public_html/servidorEfem/lib/freehep-util-2.0.2.jar:/home/alonso/public_html/servidorEfem/lib/freehep-xml-2.1.1.jar:/home/alonso/public_html/servidorEfem/lib/freehep-xml-2.1.1.jar:/home/alonso/public_html/servidorEfem/lib/freehep-xml-2.1.1.jar:/home/alonso/public_html/servidorEfem/lib/freehep-xml-2.1.1.jar:$command";
    $output = array();
    exec($command, $output);
    foreach ($output as $key => $value)
        echo "$value \n";
    ?>
```

So the PHP program uses Java 1.6 to execute a program from the library, including in the classpath everything that the program could require, and returns the output of the program to the user. Since the Java program is executed directly from command-line, there is no graphic environment in this case, i.e. the program is executed in headless mode (*-Djava.awt.headless=true*). This means that the program is not capable of showing frames or windows on the screen (in the server), which is an additional reasonable condition. This limitation also imposes some minor conditions in the code when generating charts in headless mode.

2.2.4 Returning web pages with charts

The HTML output is created using the class `jparsec.io.HTMLReport`. A report generated with this class has some advantages. For example the class hides the complexity of creating HTML output with links and tables, saving time. Another advantage is that the constructor of the `HTMLReport` object can be directly changed by a `LATEXReport` object, resulting in a LATEX output with no additional modifications in the code most of the times.

However, to return charts in an efficient way is slightly more complicated. The usual approach would be to write the image in some path of the server, and to return the image including that path in the SRC property of the HTML `` tag. This approach is very inefficient since it requires to manage in some way the images created and to eliminate them from the server at some time. The approach considered in the `ephemeris` server is to include the image itself inside the SRC property, using a technique called base 64 encoding. This technique, which is something standard in HTML, encodes the image and transforms it into a set of characters that represents it. As with most standards, Microsoft Internet Explorer does not support it, at least before its version 7. With other browsers like Mozilla Firefox or Opera there is no problem to show the images. To write the image in this format first it has to be encoded in base 64 and written inside the SRC property, including as a prefix `"data:image/png;base64,"` for a PNG image. This technique increments the size of the HTML output by about 15% (usually more for images in PNG format as those used in the server, and less for JPG ones), but has important advantages. The base 64 encoding/decoding functions have been taken from Christian d'Heureuse³, and are found inside the `JPARSEC` library (`jparsec.jar`) at `jparsec.io.image.Base64Coder`.

³See <http://www.source-code.biz/base64coder/java/>.

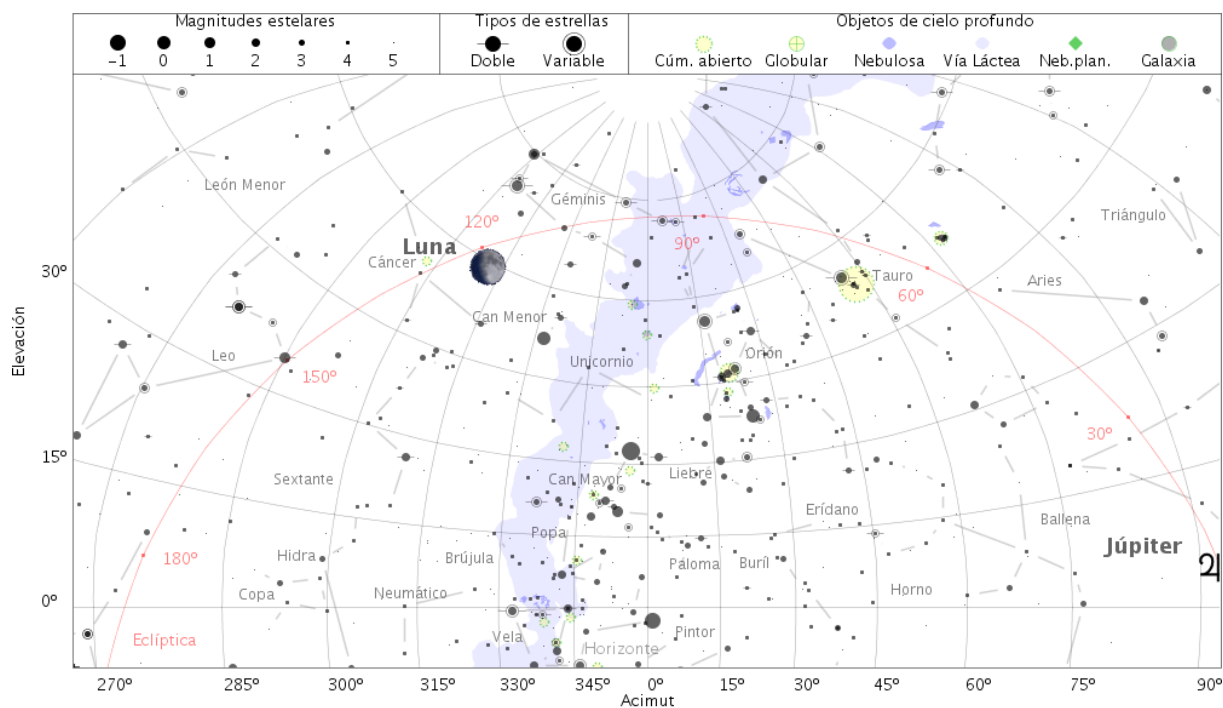


Figure 3: An example chart with the sky in Madrid on March 15, 2011, 19:15 UTC, looking towards south.

2.2.5 Geolocalization of the user and date selection

The geolocalization is done by the PHP script `geolocalization.php`, which is called within the form from `formulario.php`. It gets the client IP address (or the IP of the Internet provider if client's IP is unknown) and then calls `jparssec.observer.ObserverElement` giving as a parameter the IP. The program in that Java class calls the free IP geolocalization service called `IPInfoDB`⁴, giving as a result several lines of text with the location, the latitude and longitude, and the time zone. Later these data is used by means of Javascript to set the fields of the form accordingly.

The `formulario.php` file includes a call to Google Maps API to show the current geographical position of the user in a map. The map can be dragged and zoomed to search for a given location. A double clic on any point of the map will set the coordinates of that position in the corresponding fields. The location name and height above sea level will be set to the closest city available in a database with more than 4000 cities around the world (and another 8000 locations only for Spain).

To introduce dates in the form a free Javascript calendar component called `Tigra Calendar`⁵ is used. The component is very simple to use and is directly integrated in `formulario.php`.

⁴See <http://ipinfodb.com/>.

⁵See http://www.softcomplex.com/products/tigra_calendar/.

3 Testing and installing

Testing phase took place in July, 2010. During two weeks some corrections were done in certain areas, and after some major improvements were added as a results of my own testing and some comments given by other astronomers. Installation came in October, 2010.

3.1 Memory requirements in the applet

The applet requires 300 MB of RAM to work. In recent installations of the Mozilla browser and the Java plugin this requirement is not a problem, since the newer Java plugin included in JRE 1.6u10+ allows to increase the memory of the applets by including some parameters in the web page that launches it. But in other cases the Java plugin should be manually configured to allow applets to use a large amount of memory. In Linux an attempt to solve this problem was done by updating both the browser and the Java installation under Ubuntu Intrepid, without success, whereas an updated Debian Lenny worked fine without any manual configuration.

This configuration process is explained in the section of the Firmamento.html file that describes the applet. Basically it is necessary to launch the *Sun Java 6 Plugin Control Panel*, and under the Java tag, click on the first upper button and add "-Xms300m -Xmx400m" to the (usually empty) field called *parameters of the runtime environment*. After accepting and applying the browser should be reinitialized.

3.2 Comparing results with the Anuario

No significative differences were found when checking results with those from the Anuario. The maximum difference in the eclipses was found to be 1 minute, which is the maximum expected difference since both results are rounded to the nearest minute.

3.3 Installing at OAN and IGN

The installation of the ephemeris server was done in October, 2010. The files were installed at one of the OAN servers at Yebes, and the machine mapped to be accesible from the url <http://www.oan.es/publico/servidorEfem/>. The main page `servidorEfem.php` was renamed to `index.php`. At IGN a page was included with a link to this page.

3.4 Updating data

The `update.sh` script located at `/lib` subdirectory should be executed at regular intervals to maintain `sky.jar`, `eop.jar`, `sunspot.jar`, and `orbital_elements.jar` files updated. Before doing that it is recommended to backup those files. After this process the ephemeris server should be tested (specially the applet that simulates the sky) to check that everything is correct. The script will also inform if the files inside `time.jar` or the longitude of the Great Red Spot in Jupiter (`orbital_elements.jar`) should be updated. To update `time.jar` the Anuario can be used, whereas the GRS longitude can be updated from <http://jupos.privat.t-online.de/rGrs.htm>. The `.jar` files are

standard .zip compressed files, that can be uncompressed using Ark for example (you may need to change the extension to .zip previously).

A complete update requires a few minutes and it is enough to do it once a year.

3.5 Notes about precision, time span validity, and other limitations

There are a few limitations in the precision of the results.

- The calculation of the dates in different calendars follows *Calendrical Calculations*, second edition, by N. Dershowitz and E.M. Reingold. The second edition is known to still contain some errors, whereas in the third edition (2008) they are mostly corrected, but the library is not updated. Before October 15, 1582 the program assumes that the input date is in the Julian calendar, although days between 5 and 14 of that month and year will give error since these dates were inexistent.
- The date given in UT1 time scale will be equal to that in UTC for dates after the last update of the Earth orientation parameters (file eop.jar) in the server. This date is currently August, 1, 2010.
- The main page can be generated for any location and country in the database (4000 cities around the world and 8000 locations for Spain). A little form at the bottom allows to generate the main page for Madrid, Tenerife, and any other location (providing its name). When there are more than one location with the same name the server gives preference to the Spanish one, if there's any. The applet uses a reduced list of locations for Spain, so only the main capitals will be recognized, although the effects of a wrong location in the sky simulation is negligible and another location closer to the user can be selected from the applet with easy.
- The instants of Lunar phases and eclipses follows the *Astronomical Algorithms* of Jean Meeus, which uses a fit to ELP2000 Lunar theory, which is itself a fit to JPL DE200 ephemeris. The implementation uses all terms, so this could result in negligible errors for lunar phases and errors of about 1 minute for eclipses, when comparing results with the entire ELP2000 theory. This is not a big concern since the results of the server are rounded to the nearest minute. In addition, the original ELP2000 theory is known to be inaccurate due to a wrong value of the secular acceleration of the Moon used in JPL DE200 ephemeris, but this problem can be fixed easily. Once corrected, ELP2000 ephemeris can match JPL DE405 up to the arcsecond level in year 1000 B.C.⁶.
- When a set of ephemerides is to be retrieved in certain time span (resulting in a table with more than one row), the algorithm is automatically changed to Moshier to obtain a decent level of performance.

⁶In fact, the *Five Millennium Canon of Solar/Lunar Eclipses* by Fred Spenak shows a difference of only 30 seconds for the eclipses around year 1000 B.C. Anyway, the projection of the Moon position to ancient times is difficult to derive due to uncertainties in the Moon secular acceleration, and its value is not necessarily constant.

- The version to be implemented will be the one with better performance. Not only in this case the precision is far enough, but also the JPL DE406 ephemeris is becoming outdated compared to latest JPL integration theories like DE414 (included in JPARSEC), or DE418 or DE421. The differences in the positions between these latest integrations and DE406 is in the 0.01 arcsecond level, which is quite above the level of precision of the Moshier fit to JPL DE404 in current dates.
- The acceptable range of dates to create the astronomical events page is restricted to years -1350 (1350 B.C.) to year 2998 (2998 A.C.). This is strictly the time span of validity of the Moshier fit to JPL DE404 ephemeris. This time span could be expanded to the past (up to year -3000), with some loss of precision in the inner planets Mercury, Venus, and Mars.
- The ephemeris for artificial satellites follows the simple but accurate algorithm of James Miller (<http://www.amsat.org/amsat/articles/g3ruh/111.html>), whereas the ephemeris for probes are approximate, since it uses orbital elements obtained from JPL Horizons.

References

The development of the JPARSEC library have required more than a hundred of references (A&A/AJ/ApJ papers, books, web pages, catalogs ...), and the list is continuously growing. All that references can be checked online at <http://conga.oan.es/%7Ealonso/doku.php?id=jparsec>. The development of the web ephemeris server was a simple task compared to the complexity of that project.