To Build an e-Sphere

Version 1.3

Changes since version 1.2

Addition of Full Keplerian Eccentricity calculations - Facilities added for a full solution to Kepler's "Equation of the Centre" using an iterative method.

Addition of Planetary Perturbation Calculations - Facilities added to calculate perturbations to the orbits of Jupiter and Saturn.

Addition of extra Lunar Perturbation Calculations - Facilities added for calculations of a fuller set of lunar orbital perturbations.

Change to observing location selection mechanism - Old form based system replaced by clickable map.

Addition of two new Planetary Location Displays - A 360° panoramic display showing positions of all the planets, the sun and the moon. A plan view of the Solar System.

Changes since Version 1.1

Addition of "The Loughton List" - A module that displays the positions of objects from the "Loughton List" of interesting objects.

Updated Moon Module - Changes to the moon module to handle an enhanced list of perturbations to the lunar orbit. Thus improving the accuracy of lunar positions.

Updated Planet Module - Changes to handle "Shifting" elements for planetary orbits.

Improved Elements for Planetary and Lunar Orbits - Improves the accuracy of planetary and lunar positions.

Changes since version 1.0

Addition of Temporary Objects Module - A module that displays the positions of temporary visitors to our skies. These include Comets, NEOs, Supernovae and Meteor Radiants.

Addition of Hi-Definition Downloads - A modification that enables people to download higher resolution images of e-Spheres for later handling with a program such as Photoshop and intended for high quality printing.

Changes to the e-Sphere Control Panel - Changes to the e-Sphere control panel to facilitate the above changes.

Lunar parallax corrections - A simple correction is made to the "Natural Horizon View" charts to allow for the position of the observer on Earth.

Introduction

The LAS e-Spheres provide a range of sky charting options. The system will plot charts of constellation patterns, the moon, the sun, the planets, some of the brighter minor planets, comets, NEOs and other interesting objects. Charts can be provided for any time of the day on any day from a few months in the past to any time in the future. To do this each e-Sphere is calculated from raw data and plotted to match the computing equipment used and the user requirements.

This document describes how the e-Spheres are produced. In order to achieve this for a mixed audience many of the terms used are described in detail. To enable those familiar with these terms to skip the explanations, these descriptions are printed in blue italics.

This document is not a complete "build it yourself" guide. It is designed to give an appreciation of what is involved and methods used, rather than a complete instruction kit.

Plotting Charts

The position of nearly every object plotted on an e-Sphere is, at some point in the plotting process, described in the normal astronomical system of Right Ascension (RA) and Declination (Dec). All e-Spheres work to a Mean Equator and Equinox of J2000.0.

RA is similar to the lines of longitude used to map the Earth. One difference is that RA starts from a fixed point in the sky rather than the position of a rather old but highly venerable telescope at Greenwich, England. This fixed point is known as the "First Point of Aries". The other difference is that RA is normally measured in hours rather than degrees. In this context 1 hour = 15 degrees.

Dec is similar to the lines of latitude used to map the Earth. The celestial equator is the strip of sky above the Earth's equator. Points not on the celestial equator are measured using the number of degrees north or south of the celestial equator. Points north have a positive Dec, with the celestial north pole being +90°. Points south of the celestial equator have a negative Dec, with the celestial south pole being -90°.

Mean Equator and Equinox: The axis of the Earth wobbles a lot but fortunately it does this very slowly. This means that the true celestial poles, and hence the true celestial equator, continually change positions in the sky. It also means the fixed reference point moves. We still call it the First Point of Aries but it is no longer within the constellation of Aries. For convenience we use the positions at the start of the year 2000. The errors in doing this are small but grow a little more each year. Astronomers normally change their reference epoch every 50 years. So it is unlikely that the e-Spheres system will change epoch in the next 10 years. For the orbits of the planets and the moon the epoch is simply a start point from which to start the calculations.

To plot a "Traditional Planisphere View" chart, i.e. one centred on the celestial north pole, it is only necessary to convert the RA and Dec to polar coordinates and then convert the polar coordinates to Cartesian coordinates used by the graphics package within PHP.

Polar coordinates are a system of defining the position of something by measuring how far it is from a fixed point and in which direction. So a polar coordinate consists of a distance and an angle. Beware, both the "Traditional Planisphere View" (Polar Charts) and the "Natural Horizon View" (Alt-Azimuth Charts) we will see later use polar coordinates.

Cartesian coordinates were invented by a chap called René Descartes while lying in bed watching a fly, or so the story goes. As a philosopher he was able to spend a lot of time in bed. Cartesian coordinates are similar to the way a grid works on a map. Any position can be described by how far along (usually left to right) and how far up it is from a fixed point. Computer scientists make life a little more complicated though, because most computer graphics packages describe how far along (left to right) and how far down a position is. This adds a negative sign to the up part of the maths used to calculate the position. Don't we just love computer scientists!

The "Traditional Planisphere View" e-Spheres and the sky discs for the self print planispheres available on the LAS website are produced by the same software. The switch is achieved by use of different parameters when invoking the software.

To calculate the distance from the celestial north pole we simple subtract the Declination from 90°. Thus an object at declination +80° is just 10° from the north celestial pole. An object at declination -30° is 120° from the celestial north pole. To calculate the angle is a little more difficult. For plotting the e-Spheres we define the point in the sky north of the observer as our reference point (0°). The RA is first converted to degrees by multiplying by 15. We then need to allow for the rotation of the Earth. To do this we need to know the local sidereal time. There are 4 steps to calculating the local sidereal time.

$$x = a \sin \theta$$
$$y = a \cos \theta$$

Converting polar coordinates to Cartesian coordinates
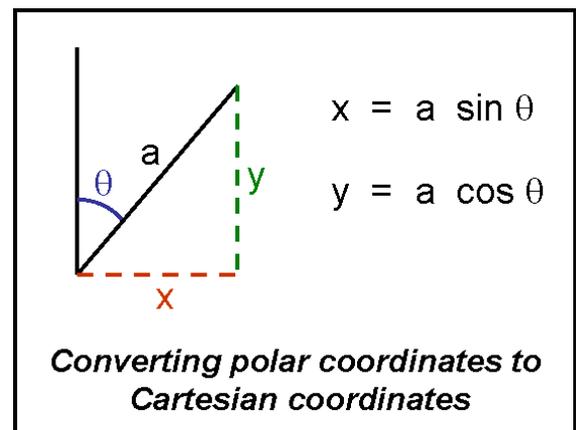
1) Calculate the Julian Date for the start of the current year. PHP provides a function for calculating the Julian Date for any time and date, but the function is half a day out so this has to be corrected. (If they ever fix this bug, lots of web sites will fail.)

2) Use this Julian Date to calculate the Greenwich sidereal time at the start of the year. This involves the use of constants that can be obtained from astronomical data books.

3) Calculate the number of mean solar days (and part days) since the start of the year. Use this to calculate the number of sidereal days since the start of the year.

4) Correct for the longitude of the observer. Dividing the longitude in degrees by 15 gives the number of hours correction. Points west of the Greenwich meridian are earlier and points east are later.

From this we know the RA of the point directly overhead (Zenith point). Adding or subtracting 180° gives us our reference point in the sky due north of the observer.

One further complication is that sky charts are designed to be used held over the observer's head. This means the charts have to run backwards (anti-clockwise). This is achieved by subtracting the angle of each object from 360°.

By following these steps we get a plot of the whole sky down to about declination -30° (the limit of data stored for e-Spheres). To plot the horizon oval we need to identify where the horizon is. This is done by taking points every 4 degrees round the local horizon and then projecting those onto the equivalent equatorial coordinates. The equations for this spherical projection are shown overleaf.

To convert Hour Angle to RA we simply subtract it from our local sidereal time. We can then convert the RA and Dec of each horizon point to polar coordinates and then on to Cartesian coordinates.

The PHP function that joins the dots will then draw a series of very short lines that produce the horizon oval. This is done on a separate image. The space in the centre of this oval is turned transparent and the image is then used to mask out the areas of the sky chart that are below the horizon. Other information (date, time, location and LAS details) is then plotted around the outside of the horizon oval.

$$\sin \delta = \sin \alpha \, \sin \phi + \cos \alpha \, \cos \phi \cos A$$

$$\cos H = \left(\frac{\sin \alpha - \sin \phi \, \sin \delta}{\cos \phi \, \cos \delta} \right)$$

Where α = Altitude above horizon
   A = Azimuth
   ϕ = Observer's latitude
   δ = Declination
   H = Hour angle

Converting Horizon coordinates to Equatorial coordinates

There are two problems with this process.

1) The spherical projection requires us to perform reverse trig functions (arc sin and arc cos). At school we simply used books of tables to look these up. Your calculator will now do this for you. But beware of what they didn't tell you: there is no unique answer to each reverse trig function. The tan of 45° is 1 but the tan of 225° is also 1. In each case there is more work to do to identify which of two possible answers is correct.

2) When selecting horizon points it is important to stay clear of the points directly north and south of the observer, otherwise the PHP maths routines break down and we get infinity values.

Note: The horizon oval is a different shape for each latitude. The further south the observer is the more squashed north to south the oval is. Further north the horizon oval is closer to circular. The position of the centre of the oval also changes with latitude. The software also keeps track of the top, bottom and sides of the oval so that the image can be cropped to match the latitude and get rid of wasted space. The actual size of the e-Sphere image changes slightly with latitude.

The same technique for plotting the horizon oval is used to generate the front cover of the self print planispheres.

To plot the "Natural Horizon View" e-Spheres it is necessary to convert all the points from RA and Dec to the equivalent Horizon coordinates, Altitude and Azimuth. This is done by first converting the RA values to Hour Angle by subtracting them from the Local Sidereal Time. Then we use another spherical transform that is the reverse of the one used to convert horizon coordinates to equatorial coordinates.

$$\sin \alpha = \sin \delta \, \sin \phi + \cos \delta \, \cos \phi \cos H$$

$$\cos A = \left(\frac{\sin \delta - \sin \phi \, \sin \alpha}{\cos \phi \, \cos \alpha} \right)$$

Where α = Altitude above horizon
   A = Azimuth
   ϕ = Observer's latitude
   δ = Declination
   H = Hour angle

Converting Equatorial coordinates to Horizon coordinates

Converting 640 points with this spherical projection imposes a considerable load on the servers, so efficient coding is important. Frequently required calculations are performed once and the values stored.

Once converted to horizon coordinates the values then need to be converted to the equivalent polar coordinates; 60° Altitude becomes 30° below the zenith.

Once again, the points then need to be converted to Cartesian coordinates to plot them with the PHP graphics package.

The horizon circle (and in this case it is a circle) can then be added. As for the "Traditional Planisphere View", the horizon circle for the "Natural Horizon View" is plotted onto a separate image, the centre made transparent and the separate image used as a mask to remove any graphics projecting below the horizon (e.g. lines joining stars in constellation patterns). In this case the horizon circle is simply a circle with radius 90° centred on the zenith point. No transforms are required to produce this circle.

A diagram summarising the processes for plotting fixed objects, such as constellations, can be seen below.

Coordinate conversions for plotting fixed stars
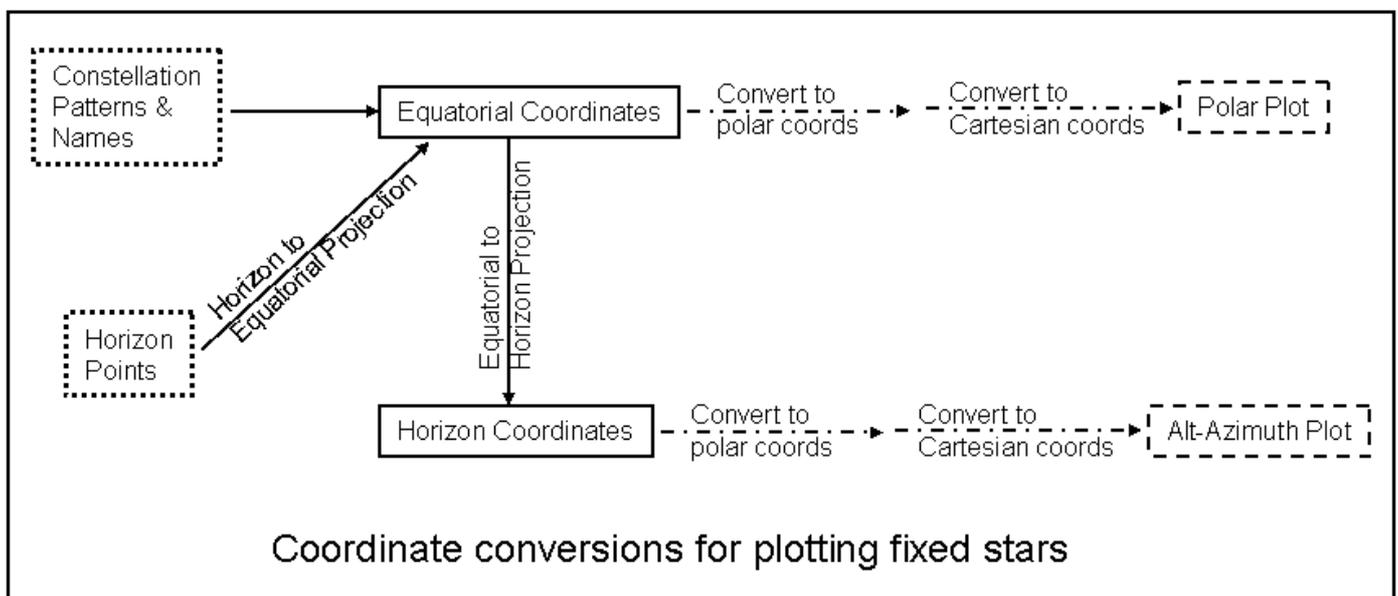
Constellation Patterns

The data for describing the constellations and their names is held in an array of data structures within the e-Spheres code tree.

As a compromise between reducing the amount of mathematics the system needs to perform every time a chart is plotted and the convenience of entering the positions of all the stars, the RA for each star is held in hours and decimal parts of an hour and the Dec for each star is held in degrees and decimal parts of a degree. This negates the need to calculate the decimal parts from minutes and seconds.

The patterns are drawn in an "etch-a-sketch" style, where it is sometimes necessary to retrace the drawing steps when a star is at the junction of two or more lines. Sometimes this is achieved by entering the star, or a string of stars, twice or thrice in the data and sometimes it is achieved by

plotting two or more sub-patterns for the constellation. The method chosen is optimized for each constellation.

The position and plotting angle for each constellation name is also entered into the data for each constellation.

At the time of writing there are 640 individual plot points to describe and name the constellations. Each of these had to be calculated and entered by hand. There were no short cuts to this activity. However this only had to be done once and the data is now backed up!

Plotting the lines is achieved by calculating the position of each star in Cartesian coordinates and then using a PHP function to join each pair of dots.

Plotting the names is more difficult. For the "Traditional Planisphere View", the names are simply rotated by the same amount that the whole sky disc is rotated (c.f. local sidereal time). This means they maintain their orientation compared to the constellation patterns. Plotting the names on the "Natural Horizon View" is more difficult. Objects need to be rotated by a different amount at different points on the chart. This is achieved by calculating the end point of the name label on a simple polar plot and converting this back to RA and Dec. The position of both the start and end points are then calculated for the Horizon plot and the angle then calculated. This angle is then used to plot the name. This method is not 100% accurate because the distortion produced by the Polar plot is not the same as that produced by the Alt-Azimuth plot (see chapter on Distortions). The small errors do not change the positions of the stars but the orientation of the name labels relative to the constellation patterns do not exactly match those on the "Traditional Planisphere View". This does not affect the use of the charts but does mean the labels are sometimes a little less tidy than on the "Traditional Planisphere View".

On the "Traditional Planisphere View" all stars are plotted. This is quicker than doing the spherical transforms necessary to find out if the star is above or below the horizon. Also, the same software is used to produce sky discs for the self print planispheres and these need the whole sky disc. Unwanted stars are masked out at the end of the e-Sphere production process.

On the "Natural Horizon View" stars that are far below the local horizon are not plotted as they would not be visible on the plot. Stars a little way below the local horizon are plotted because although they are not visible (and the dot representing the star would be masked out by the production process), it is useful to have the part of the line that is visible. In this case the information to decide if the star is visible is readily available from the altitude data.

To save processing time, the altitude of each star is calculated first. If the star is not to be plotted then the azimuth calculation is not completed.

Planets and Sun

The planets and sun are moveable objects. Their RA and Dec does not remain constant. The advantage of e-Spheres over the printable planispheres is that e-Spheres can show the positions of these moveable objects at any time. However the calculations to do this add another level of complexity, and hence even more load on the servers. We don't pay for processor time but efficient coding is still required if the e-Spheres are to display within an acceptable time. Fortunately there are a lot fewer planets than there are stars, otherwise the server load would be unacceptable.

Calculating the position of a planet is a multi-step process.

1. Calculate the position of the planet in an imaginary circular orbit within its own coordinate system. For this we require the orbital period, the position of the planet at the epoch for which the planet's data is recorded and longitude of the perihelion.

2. Correct the position to allow for the orbit being elliptical. There is no arithmetic solution for Kepler's "Equation of the Centre". Facilities are provided for three methods, a simple approximation, the approximation with a correction function and a full iterative solution to Kepler's equation. None of the planets have a highly eccentric orbit so sufficient accuracy can be achieved for the maximum size to which the e-Spheres are plotted.

3. Calculate the Heliocentric Longitude, the radius vector (distance from sun) and the Heliocentric Latitude.

4. Project the position of the planet on to the ecliptic plane to find the Projected Heliocentric Longitude and the Projected Radius Vector.

5. Refer the planet's position to that of the Earth. This gives us the projected position of the planet within the ecliptic. To do this we first have to perform the steps above for the Earth. The position of the Earth however stays constant for all the planets. The process used for making this projection is different for inferior plants and superior planets (i.e. those closer to the sun and those further away).

6. Convert from Geocentric Ecliptic Coordinates to Equatorial Coordinates. For this we need to know the Obliquity of the Ecliptic (the angle between the ecliptic plane and the Earth's equatorial plane).

$$M = \frac{360\ D}{365.2422\ T_p} + \varepsilon - \omega$$

$$\nu = M + \frac{360}{\pi}\ e\ \sin M$$

$$\iota = \nu + \omega$$

$$r = \frac{\alpha\ (1 - e^2)}{1 + e \cos \nu}$$

$$\psi = \sin^{-1}(\ \sin(\iota - \Omega) \sin i\ )$$

$$\iota' = \tan^{-1}(\ \tan(\iota - \Omega) \cos i\ )$$

$$r' = r \cos \psi$$

Where
   M = mean anomaly
   D = days since planet epoch
   T_p = orbital period of planet
   ε = longitude at planet epoch
   ω = longitude of Perihelion
   ν = true anomaly
   e = eccentricity of orbit
   ι = heliocentric longitude
   r = length of radius vector
   α = semi-major axis of orbit
   ψ = heliocentric latitude
   Ω = longitude of ascending node
   i = inclination of the orbit
   ι' = projected heliocentric longitude
   r' = projected radius vector

Finding Position of Planet (Ecliptic Coordinates)

Data for each of the planets is held in arrays. Included in this is the epoch for the data, thus enabling different planets to be listed at different epochs.

Any corrections for perturbations to planetary orbits are added after all the orbits have been calculated. These perturbations are dependent on the positions of some of the other planets.

The position of the sun for calculating its RA and Dec is found by calculating an imaginary twin to Earth but with an orbit 180° twisted round from Earth. The distance to the sun is found from the Earth's orbit data though.

for inferior planets

$$\lambda = 180 + L + \tan^{-1}\left(\frac{r'\sin(L - \iota')}{R - r'\cos(L - \iota')}\right)$$

for superior planets

$$\lambda = \tan^{-1}\left(\frac{R\sin(\iota' - L)}{r' - R\cos(\iota' - L)}\right) + \iota'$$

for all planets

$$\beta = \tan^{-1}\left(\frac{r'\tan\psi\sin(\lambda - \iota')}{R\sin(\iota' - L)}\right)$$

Where

λ = geocentric ecliptic longitude
L = heliocentric longitude of Earth
R = length of radius vector for Earth
β = geocentric ecliptic latitude

Converting Ecliptic Coordinates to Geocentric Ecliptic Coordinates

At the time of writing, all the planets (Mercury through to Neptune) are covered. Pluto is not covered, it is very dim and the highly eccentric orbit means the errors involved in using the approximations to Kepler's equation would be unacceptable.

Two minor planets are covered. Of the dwarf planets, Ceres is by far the brightest, (nearly 7 magnitudes brighter than Pluto at opposition). Ceres also has a relatively low eccentricity. Of the other minor planets, Vesta is the brightest and again has a reasonably low eccentricity.

$$\alpha = \tan^{-1}\left(\frac{\sin\lambda\,\cos\varepsilon - \tan\beta\,\sin\varepsilon}{\cos\lambda}\right)$$

$$\delta = \sin^{-1}(\sin\beta\,\cos\varepsilon + \cos\beta\,\sin\varepsilon\,\sin\lambda)$$

Where
ε = obliquity of ecliptic

Converting Geocentric Ecliptic Coordinates to Equatorial Coordinates

For the "Natural Horizon View" charts the RA and Dec of the Equatorial Coordinates are converted to Altitude and Azimuth using the same functions as for the constellations. For both views the onwards conversion to Cartesian Coordinates is also performed by the same functions used for plotting the constellations.

Note: The module for producing an iterative solution to Kepler's "Equation of the Centre" rarely needs more than 3 iterations to reach a solution and normally completes in 2 iterations for the planets. The module is protected against an excessive number of iterations. Calculating the error for each iteration stretches the mathematics package beyond it's inherent precision. This problem is solved by careful scaling of the calculations to maximise precision.

The Moon

The moon is more complex again than the planets. However, this time there is a saving in that we don't have to refer the position of the moon to the Earth because we have calculated the orbit with reference to the Earth. The orbit of the moon around the Earth is calculated in a similar way to that of the planets around the Sun. Again the eccentricity is low enough to enable us to use an approximation to Kepler's equation. A full iterative solution to Kepler's equation can also be achieved by using the same module used for the planets.

However the effect of the sun's gravity on the moon is too large to ignore. Fortunately we don't have to perform iterative gravity calculations. The effects of the sun on the moon have been studied in detail for a long time and the major effects have been summarised into a series of corrections which can be applied arithmetically. In the simple version we make 5 corrections of this type in all. Several options are available to make an enhanced set of corrections.

To calculate the position precisely we would need to make further gravitational corrections. The errors from not doing this are small enough not to bother us in everyday use of the e-Spheres. However, a more accurate plot could be useful for observing solar eclipses and lunar occultations.

$E_V = 1.2739 \sin(2(\iota - \lambda_0) - M_M)$ Where λ_0 = Sun's ecliptic longitude and M_M = Moon's mean anomaly

$A_e = 0.1858 \sin(M_0)$ Where M_0 = Sun's mean anomaly

$A_3 = 0.37 \sin(M_0)$

Then Moon's corrected anomaly $M'_M = M_M + E_V - A_e - A_3$

And $E_C = 6.2886 \sin(M'_M)$

$A_4 = 0.214 \sin(2 M'_M)$

Then Moon's corrected longitude $\iota' = \iota + E_V + E_C - A_e + A_4$

$V = 0.6583 \sin 2(\iota' - \lambda_0)$

Then Moon's true longitude $\iota'' = \iota' + V$

Applying Corrections to the Moon's Orbit

The necessary coordinate conversions for the "Natural Horizon" and "Traditional Planisphere" views are once again performed by the same functions used to plot the constellations.

For the "Natural Horizon View" charts a simple parallax correction is made. The calculations above give the position of the moon as seen against the background stars by an imaginary person standing at the centre of a transparent Earth. The comparative closeness of the moon means that when seen close to the horizon the moon would appear up to one degree lower in the sky. Higher in the sky the parallax would appear less. The small changes in parallax due to the changing distance to the moon are not currently corrected. At the scale used for e-Sphere display these small changes in parallax are not noticeable.

At the moment, no parallax corrections are made for the "Traditional Planisphere View" chart. The calculations for parallax corrections in Equatorial Coordinates are more complex than those for Alt-Azimuth coordinates.

Temporary Objects Module

Temporary objects are those objects that either move relatively quickly against the field of background stars or are only apparent for a short period.

Comets and NEOs

Comets and NEOs are handled exactly the same way. The only differences are cosmetic.

The orbits of comets and NEOs are very eccentric. This makes calculating the orbit much more difficult than that for the major planets. Ensuring a reasonable accuracy for highly eccentric orbits would impose a serious processing load on the server. Instead, a PHP data array is entered with a list of positions in RA, Dec and Date/time. The data array also contains the name, the type (comet or NEO) and cosmetic information to assist in the display.

If the time and date requested is within the list of times and dates in the array the object is plotted. Otherwise the object is ignored.

To plot the object the two data points either side of the requested time and date are found. The software then performs a linear interpolation to find a best estimate for the position at the time and date requested.

For the "Traditional Planisphere View" this position is then converted to Cartesian coordinates using the same functions as for stars and planets. For the "Natural Horizon View" the position is first converted to Horizon Coordinates and then to Cartesian coordinates using the same functions as for stars and planets. The object is then plotted with a suitable label. If the track is also plotted the label is omitted.

For comet and NEO tracks, the data points in the array are plotted in turn and the points joined together "etch-a-sketch" style. "Tick" marks are plotted at each data point. Labels for each point are twisted to be at right angles to the track at that position. To reduce clutter, the constellation patterns and names are printed in grey on the "Natural Horizon View" when tracks are requested.

There are two problems associated with linear interpolation:

Where the track describes a sharp turn on the sky, the interpolation tends to cut the corner. This looks wrong on tracks and is indeed wrong. It also means the object will be plotted in the wrong position.

Where objects appear to accelerate or decelerate against the sky, the linear interpolation will either underestimate or overestimate the motion since the last known point.

To reduce the effects of both these problems and thus produce smoother curves and motion, there is a facility to add intermediate points to the data array. By turning off both the "tick" and the "label" associated with the intermediate point, clutter is avoided.

Where points are very close together it is possible to turn off the label but leave the "tick" in place. This enables people to count the ticks where they are too close for labels.

Supernovae and Other Fixed Objects

Supernovae and other fixed objects are simply entered into the data arrays with a name, type and a position in RA and Dec. They are plotted using the same functions as the constellations and planets to convert coordinates as required. The symbol plotted depends on the type of fixed object.

Meteor Radiants

Details of the radiants for major meteor showers are entered into the data arrays with a start date/time, end date/time, shower name and position in RA and Dec. Meteor radiants are only plotted if the date/time requested is within the recognised limits for that meteor shower. They are plotted using the same functions as the constellations and planets to convert coordinates as required. A "Gun Sight" symbol is used, but beware, the gun sight is pointing at you.

Note: Meteor radiants move slowly east across the sky. At the moment no corrections are made for this daily motion and the position at maximum is used. The radiant is not an object, just a notional point from which the meteors appear to radiate so the errors in doing this are not significant.

Note: Data for meteor showers is entered in the tables in a way that allows them to repeat every year. Small corrections can be applied to get the date/time exact for each year, but if left alone they will repeat in a way that gives a reasonable result.

Satellite Flares

Satellite flares are entered into the data arrays listing the name (e.g. Iridium), the position in Alt-Az Coordinates, actual time of flare, start date/time for display, end date/time for display and the place from which the flare can be observed. The place needs to be entered using the same abbreviations used to list locations for e-Spheres.

Iridium flares are only displayed in the "Natural Horizon View" and only for the place the data is valid for. The Alt-Az data is simply converted to Cartesian coordinates to enable plotting.

Note: It is not intended to use the satellite flares function for general display. The parallax for satellites is large. Also flares are only visible over a very small area. People more than a few miles from the point the data is intended for will not see the flare as displayed and may well not see the flare at all. This function is intended to be used for the production of one off charts for interesting events.

Loughton List Module

"The Loughton List" is a list of interesting objects to observe in the night sky. The list was put together by Malcolm Zack and a team of observers. The list is graded by difficulty into three categories, Bronze, Silver & Gold. It is expected that the list will grow with time. At the time of launching the list contained 75 objects, all of which are visible from the UK.

The list of objects is held in a database. The Name, RA, DEC, grade and type of each object are included in the database. A Pandora utility is run each time the database is updated. This pre-calculates some of the values required for calculating positions, thus reducing the amount of processing required each time the list is displayed.

	Field	Type	Collation	Attributes	Null	Default	Extra
☐	LL	int(4)			No	None	AUTO_INCREMENT
☐	RA	time			No	None	
☐	DEC_Degrees	int(4)			No	None	
☐	DEC_mins	float			No	None	
☐	Name	text	latin1_swedish_ci		No	None	
☐	Constellation	text	latin1_swedish_ci		No	None	
☐	Grade	enum('B','S','G','P','U','O')	latin1_swedish_ci		No	U	
☐	Type	text	latin1_swedish_ci		No	None	
☐	NGC	text	latin1_swedish_ci		Yes	NULL	
☐	IC	text	latin1_swedish_ci		Yes	NULL	
☐	RA_dec	double			Yes	NULL	
☐	RA_dec_deg	double			Yes	NULL	
☐	Sin_DEC	double			Yes	NULL	
☐	Cos_DEC	double			Yes	NULL	
☐	Notes	text	latin1_swedish_ci		Yes	NULL	
☐	alt	char(1)	latin1_swedish_ci		Yes	NULL	
☐	Alt_P	char(2)	latin1_swedish_ci		Yes	NULL	

Loughton List Database Structure

When selected, the list is read from the database and positions of each object calculated from the RA & DEC data held in the database. For the "Natural Horizon View" charts the RA and Dec of the Equatorial Coordinates are converted to Altitude and Azimuth using the same functions as for the constellations. For both views the onwards conversion to Cartesian Coordinates is also performed in a similar way to the constellations.

When "The Loughton List" objects are displayed the constellation patterns and constellation names are displayed in light grey to reduce the clutter on the chart. Similarly the displays of Sun, Planets and Moon are displayed in "faded" colours to reduce clutter.

The e-Sphere Control Panel has also been updated to allow "The Loughton List" objects to be selected or deselected as required.

Solar System Display Module

The new Solar System display module provides two new plots.

One of these new plots shows the positions of all the planets, the Sun and the Moon as seen from the centre of the Earth. Unlike the other plots, this new plot ignores the horizon and plots the planets in a 360° panorama as seen from the centre of a transparent Earth. It is in reality an RA - DEC based plot covering declinations between +30° and -30° and RA 0h to 24h.

To implement this plot the RA and DEC figures are simply scaled and used directly as Cartesian coordinates. The RA axis is plotted backwards to match the view from the northern hemisphere. The ecliptic plane is also plotted onto the chart.

On the large scale version of this display it is also possible to display bars indicating which constellation the ecliptic plane passes through.

Note: These bars show the actual positions of the 13 constellations that the ecliptic plane appears to pass through. This does not match the "Star Signs" used by western astrology for two main reasons, The astrological star signs never did match the constellations properly and western astrology, unlike Hindu astrology, has chosen not to correct for the *Precession of the Equinoxes*. After more than 2000 years the errors are quite large. The thirteenth constellation arises because the borders of the constellations were revised in 1930.

The other new plot is a plan view of the whole solar system. Because the orbital distances of the outer planets are much larger than the inner planets the Sun-Planet distance scale is logarithmic but with a small offset to avoid huge apparent changes in the orbit of Mercury. This is the software equivalent of an orrery.

To implement this plot the Projected Heliocentric Longitude and the Semi-Major Axis derived as part of the Planets Module are used . These are converted directly to Cartesian coordinates and scaled to match the display.

The Solar System display module uses a new top level module to build the graphics displays. It is possible to select either display or a combination of both displays in one image. The positions of the Planets, the Sun and the Moon are calculated by the same Planets Module and Moon Module as is used for the normal e-Sphere production. Additions to the Planets and Moon modules are triggered by the new top level Solar System display module. These additions control the plotting of the positions of each object in the two new Solar System displays.

The Whole Process

Coordinate conversions for plotting e-Spheres
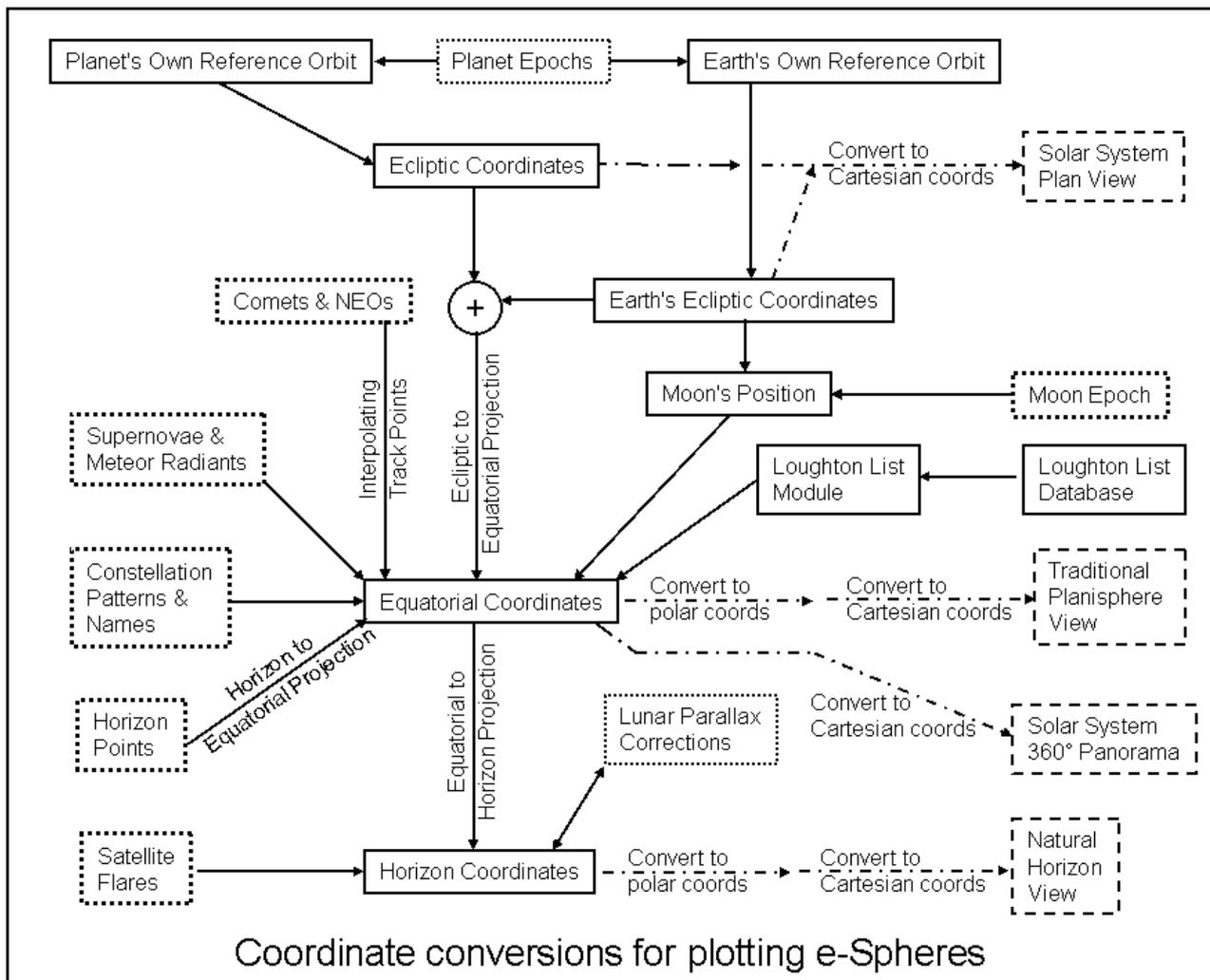
To produce a full e-Sphere with constellations, planets, sun, moon, Loughton List objects, supernovae, meteor radiants, comets, NEOs, satellite flares and Solar System Views all these processes need to take place. There are multiple coordinate conversions, but the common point through which all but the satellite flares process and the Solar System Plan View pass is the equatorial coordinates, the RA and Dec that we use to find our way round the sky. From this point either a "Traditional Planisphere View" or a "Natural Horizon View" can be produced. The whole process can be seen above.

Graphics

The PHP language allows PHP scripts to build virtual images on virtual canvasses. All the e-Sphere charts are built this way. Only once the chart is complete is it sent to the browser. The chart appears to the browser as if it is a static PNG image. The whole process is normally completed in less than 1 second.

Control Panel

Where e-Spheres are embedded into other web pages they are either pre-fixed to display one set of data or come with a small control panel. This small control panel enables the user to select the time to the nearest hour. The e-Sphere then displays the next occurrence of that time. These small control panels are simple to use and take up very little space.

There is however a more complete control panel that is designed to be displayed free standing and normally opens in a reduced size window close to the top left of the screen.

The major part of this control is a simple HTML form. This enables users to select a number of things:

- Which objects to display: 6 simple on/off buttons are provided for Constellation Patterns, Constellation Labels, Loughton List objects, Temporary Objects, Comet and NEO Tracks and Sun, Moon and Planets.

- Location: Viewing location is now selected by means of a clickable map. Two favourite LAS locations (Theydon Bois and Kelling Heath) are provided as is the Isle of Wight Star Party camp site. A number of towns and Cities over northern Europe, Scandinavian coastal regions and into the Arctic Circle at Svalbard are selectable. Most of the Scandinavian population lives around the coastal area

- The type of display: "Natural Horizon View" or "Traditional Planisphere View". A spare button is available for a future 3rd display. This button is protected from curious people selecting it.

- The normal on screen display or an option to download a high definition version designed for printing.

- Time: It is now possible to select any time to the nearest minute using drop down lists. It is also possible to type directly into the "Hours" and "Mins" boxes. This system avoids the chance of people entering invalid times. There is no point in providing a resolution of less than 1 minute.

In the middle of the right-hand side there is a calendar display. This is provided by a separate PHP file and is displayed as an inline floating frame. This allows any date to be selected from approximately 6 months prior to the current date to any date in the future. Buttons at the top allow users to step backwards and forwards through the months and forwards 3 months at a time. When not displaying the current date a reset button is also available. Selecting a day is done by simply clicking on the date required.

This "date picker" communicates with the main form by means of JavaScript. This allows the selections already made on the main form to be held and not reset to default status.

The map of Europe and Scandinavia was drawn, by hand, using a PC based drawing tool. The map is actually a background image and is available in 4 different sizes. The overlaying image containing the observing locations is generated within PHP from a data file. This enables the addition of new locations without having to redraw the map.

A version of this clickable map is also embedded into one of the Loughton List Pages.

Distortions

We perceive the sky as if it is the inside of a large sphere. Our methods of listing positions are based on this concept of a fixed sphere. Attempting to plot a spherical surface on a flat sheet of paper or a flat screen will always produce distortions.

The Polar plot used for the "Traditional Planisphere View" does not look distorted when viewed in its entirety. But when a section off to one side is viewed, as it is after the mask is applied, the view looks very distorted when compared to the actual sky. The "east" "west" directions printed around the horizon oval look odd.

The "Natural Horizon View" looks less distorted but there are still significant distortions. Until someone builds a spherical screen or a holographic 3D display these distortions are unavoidable.

The Future

Moon Phases

Having the positions of the Earth, Sun and Moon means it would be possible to produce a reasonably accurate calculation of the moon phase and even an orientation for the lunar crescent. This could be added to the e-Sphere either as a separate graphic in the margin area or as a varying icon on the sky chart. The extra processor load would be small.

Lunar Parallax

The moon is the closest celestial object to the Earth that we plot on the e-Sphere charts. Being relatively close, the position of the moon on the background stars appears different from different observing locations. In the worst case, when the Moon is closer to the Earth than normal and being observed close to the horizon, the Moon can appear up to about 1° away from the geometric calculation based on the centre of the Earth.

Parallax corrections are currently only made for the "Natural Horizon View" (alt-az) plot. Making the changes to the "Traditional Planisphere View" is a little more involved but could be added.

Making an allowance for the changing distance to the moon is of limited value but may be added at some point in the future. This could be useful when viewing solar eclipses.

Problems

Reverse Trig Functions

The problems with inverse trig functions noted above reoccurs at many places within the e-Spheres code. It is a tricky problem to overcome and makes testing difficult and lengthy. Sometimes a piece of code seems to work over a timescale of several years, but then a few variables come together to push the angles into the wrong quadrant and the inverse trig function produces the wrong value. Typically a small change in time might make a planet suddenly jump across the sky. One of the test harnesses used to test e-Spheres presents the position of the sun, moon and planets in a graphical form. This makes it easier to spot sudden jumps.

Maths Package

Scientific endeavours, such as this, often make use of double-precision maths packages. Although PHP has the concept of a double precision floating point number, it actually handles it exactly the same way it handles a single precision floating point number. This puts a definite cap on the accuracy which can be achieved in the calculations and means that calculations need to be performed in a way that doesn't push the precision to its limits.

Computers hold decimal numbers as something called a floating point number. This is just an efficient way of holding numbers whilst allowing people to perform calculations on very large numbers (such as bank bonuses) and very small numbers (such as savings account interest rates). The precision of the PHP maths package is sufficient for most web and business use, except banking. However it is not really enough for serious science or engineering applications.

On the other hand, web servers are not usually designed to perform lots of high precision arithmetic and using double precision floating point numbers does increase the processing load considerably. It is possible that the processing overhead would mean that we could not really use a double precision package anyway. A full "Natural Horizon View" e-Sphere plot (sun, moon, planets and constellations) performs approximately 6000 trig functions and 8000 arithmetic functions. When the servers are quiet this is all performed in a little under 1 second. When the servers are busy there is a noticeable delay in producing an e-Sphere.

The accuracy of e-Spheres is not just limited by the resolution of the maths package. The technique of calculating planetary positions by Kepler's Laws alone is not precise. Errors of up to 10 min of arc can occur in the positions of Jupiter and Saturn. To calculate the positions accurately we need to take the interactions between the planets into account. This "many body problem" can be solved by iterative processing, calculating the effect of each planet on each other, correcting the positions and then repeating the calculations. This is way beyond the capabilities of the sort of computers normally used for web serving. Fortunately equations are available that approximate the full iterative solution to these interactions. Also, osculating orbital elements have these corrections built in, but are only valid for a limited period. On the scale that e-Spheres are displayed the residual errors are very small and should not exceed the pixel resolution of our charts.

Objects close to the horizon will show an appreciable shift in observed position due to refraction. This varies with atmospheric conditions. Refraction will effect all objects visible within the same region in a similar way. We do not correct for refraction.

Graphics Package Plug-In

The basic graphics package built into PHP has very limited functionality. It is barely sufficient to handle the requirements of a simple website displaying graphical information. On its own it would be insufficient to produce e-Spheres.

Fortunately PHP is shipped with a plug-in extension to the graphics package produced by a private individual. This extension provides a wealth of useful facilities. Unfortunately some of these extra facilities are dogged with bugs.

The Text Rotate function has shown many bugs. Rotating anything more than about 45° produced several problems. The text walked right across the page and some of the characters, especially punctuation characters, walked around within the text. Some characters appeared nearly two characters down the text. Full stops appeared at the top of the text. This function was judged not fit for purpose. So e-Spheres uses its own text rotate function. Unfortunately this increases the load on the servers.

Unfortunately, to produce a text rotate function required the image rotate function, also part of this plug-in extension. This too had several bugs. Fortunately there were work rounds for these bugs but they entailed using a larger than necessary canvas. This increased the workload on the servers further.

Note: Following a recent update release of PHP some of the bugs with rotating text and image rotate no longer manifested themselves. The release notes showed no changes to this extension package and the PHP support website still showed those bugs as awaiting a fix, some from 5 years back. This appears to demonstrate a lack of control over the code management process, so to avoid the risk of these bugs suddenly reappearing e-Spheres is currently using its own in-built text rotate function.

However, without this plug-in extension package e-Spheres would not be possible and the Pandora image upload and display facilities would be very clunky.

Clashing Labels

Sometimes labels for different objects clash. This is very noticeable with the planets because they all orbit the sun close to the plane of the ecliptic. An algorithm for avoiding clashes would improve the readability of the charts when planets are close in the sky. Plotting planetary labels at 45° (as we do currently) reduces the number of clashes.

Acknowledgements

Much of the mathematics and methods used to do the calculations required for e-Spheres is taken from the book "Practical Astronomy With Your Pocket Calculator" by Peter Duffett-Smith, second edition, reprinted 1983.

This is an excellent book which takes the reader through the processes step by step. Although there are still some things left for the reader to work out for themselves, it is the simplest and most complete explanation of how to do these calculations that I have read. My copy is well read and personally annotated. It is a working book that spends very little time filed away.

Appendix 1 - Tools and Languages

PHP: The main tool used to produce e-Spheres is the PHP Script Language. This is a computer programming language produced by the PHP group. PHP is specifically aimed at producing web pages and hence is designed to interwork with the HTML language used by all web browsers. The PHP scripts send a mixture of Plain Text, HTML, JavaScript and Images to your browser. The PHP scripts also use a small quantity of SQL for handling data stored in a Mysql database.

PHP is a full function computer language. PHP files have a suffix that ends in ".php". The PHP code is not sent to your browser, it is interpreted by a module attached to the web server running on the server farm.

HTML: The HTML sent to your browser tells it how to lay out each web page. It also describes links to other pages and forms to collect information from you to tell the system what you want to display.

An HTML file will usually have a suffix of ".htm" or ".html". You will often see this at the end of the addresses used to download the file.

Note: Although all web browsers use the HTML language, they do not all handle the features the same way, some features are not handled by all browsers and some features are not handled by any of the browsers. In some cases these differences are small and simply result in minor cosmetic changes to the display. In other cases it is necessary to identify which browser is being used and change the code accordingly. e.g. The Hi-Def download facility requires different file headers for Internet Explorer to those used for all other browsers.

JavaScript: JavaScript is a programming language designed to be run by your web browser. Small quantities are used to enhance the operation of forms, such as the e-Sphere Control Panel, and links to other web pages.

One example of this is the date picker used in the e-Sphere Control Panel. The date picker is a separate PHP file that displays a calendar in a small window embedded in the e-Sphere Control Panel form. When you change the date, the date picker runs a piece of JavaScript to send that information to the HTML form that comprises the bulk of the control panel.

JavaScript is not the same as the Java language. The e-Spheres system does not use the Java language.

SQL: Standard Query Language is the most common computer language used to handle data in Relational Database Systems. Some of the data used by the e-Spheres system is stored in one of the databases used to hold data for the LAS and Skycamp websites.

MySQL: MySQL is a relational database system. Relational database systems hold data in a way which makes it easy to search and to relate one piece of data to another. Information on the Loughton List objects is held in a MySQL database.

MySQLi: MySQLi is a PHP library that enables a PHP script to access and store data in a MySQL database. The SQLi library is distributed with the PHP language.

Linux: Linux is the operating system that runs on the server farm that supports the LAS and Skycamp websites. Linux provides basic system security, file handling and other essential computer system tools. Linux is very similar to the Unix operating system in the way it works and the way it looks to the user.

Apache: Apache is the web server that runs on the server farm that supports the LAS and Skycamp websites. Apache contains within it the PHP interpreter module that interprets the PHP scripts written for the e-Spheres system. It also provides essential web security systems.

CPANEL: Cpanel is a collection of tools used to organise files, edit files, build databases, maintain databases and backup data. Cpanel is not visible to normal users of the LAS or Skycamp websites.

All the PHP files used to produce e-Spheres are written using the code editor contained within the Cpanel package.

Pandora: Pandora is a secure web site management system produced specifically for the LAS and Skycamp websites. Pandora is written in PHP and uses HTML, JavaScript, SQL, MySQL and MySQLi.

Appendix 2 - Testing

The accuracy of both the Planets and Moon modules is checked against positional data from NASA and the US Naval Observatory using a semi-automatic test harness. Data from the BAA handbook has proved to be of insufficient accuracy to use as a gold standard for testing so this has been abandoned.

At the moment, e-Spheres is tested on several versions of Internet Explorer and up-to-date versions of Firefox, Google Chrome and Safari on Microsoft Windows. Tests are also performed on an up-to-date version of Firefox running on a small Linux net-book.

Many of the other browsers use the same core code used in Firefox or Chrome, so displays should look very similar. It is not feasible to run tests on the dozens of minority browsers.

The lack of an Apple computer means that e-Spheres are not thoroughly tested for Apple the Apple environment. Similarly no testing is performed in the Unix environment.

e-Spheres is not made available on the LAS Mobile web site.

Any problems with e-Spheres should be reported to the email address on the relevant web site.