A Proposal for Handling Spherical Coordinates in the Next Generation Astronomical Archives

Alex Szalay[1], Robert Brunner[1], Andy Connolly[1], Jeff Pier[2], Don York[3], Carol Lonsdale[4], John Goode[4], Barry Lasker[5], Gretchen Greene[5], Brian McLean[5]

1. The Johns Hopkins University, Baltimore
2. US Naval Observatory, Flagstaff
3. The University of Chicago
4. NASA Infrared Processing Center, Pasadena
5. The Space Telescope Science Institute

Introduction

There are several astronomical surveys which will generate Terabyte sized catalogs of measured parameters of stars and galaxies, and several Terabytes of additional imaging and spectral information (SDSS, GSCII, 2MASS). These surveys share a large set of commonalities, with more expected as they progress in their development. The existence of these surveys will revolutionize the way we do astronomy, these surveys will serve as anchors in the emerging "Digital Sky". As a result,we need to anticipate potential bottlenecks in their usability, while we ensure that they remain as interoperable as possible.

The largest interoperability issue has to do with standards and organization. As the volume of catalogued data in these archives will approach a Terabyte, to ensure the integrity of the data it is prudent to subdivide it into considerably smaller partitions, storing the subdivisions as separate files. This partitioning should utilize spatial information to a certain extent, something which has been done by several existing surveys, either by simulating the POSS plates, e.g. ROSAT, or utilizing their own partitons. Largely, these subdivision are hidden from the user, thus one could argue that it does not matter how a catalogue is organized internally, as long as it provides a common functionality. On the other hand this requires a lot of parallel effort, and there are other bottlenecks on the horizon, where a common subdivision scheme may offer a lot of benefits.

Once there are catalogs with over 100 million objects, performing cross-identifications, even if they are not dynamic, is a formidable challenge. Furthermore, all other subsequent surveys in other wavelengths will try to cross identify their objects against at least one, but possibly many of these archives. Another important issue is that science queries will likely use constraints in specifying boundaries not only in the J2000 equatorial system, but also in other more survey specific ones, not to mention Galactic l, b, supergalactic, different equinox, etc. We also need to address what metadata (data describing the contents of the archive for automated data interchange) needs to be stored, what metadata queries should be supported.

Here, following a short meeting during the IAU 179 in Baltimore, in August 1996, we would like to outline the first in a series of proposals to address some of the issues of common interest. The SDSS, GSCII and 2MASS groups feel a responsibility, as the ones doing active database development in conjunction with the largest astronomy archives, to do so in a fashion acceptable and even adaptable/adoptable by other surveys. There are obvious benefits to setting up some kind of standards or standardized systems (much like FITS standards which, while it may have shortcomings, does have the beauty that we can readily exchange data with our colleagues and with many different image and data processing software packages).

The proposed scheme outlined here can make the indexing and interoperability of the next generation archives easier. In this writeup we deal with three issues, which have far reaching consequences:

- How to subdivide the sky in an archive independent fashion
- How to represent the sky coordinates with most flexibility
- How to share knowledge/algorithms that relate to the sky partitioning.

Subdividing the Sky

The first point of common interest is the definition of common areas over the sky (like Kapteyn's Selected Areas) which can be universally used by astronomical databases. This would substantially speed up cross referencing between future catalogs, and would also enable a more consistent organization between different catalogs. The entire sphere would be uniquely covered by these common regions. The boundaries of each area and the area's nomenclature could (preferably would) be fixed for the foreseeable future. A scheme for such subdivisions has been advocated earlier (Bartlett, ADASS 1994). Here we would like to adopt this idea, and carry it one step further.

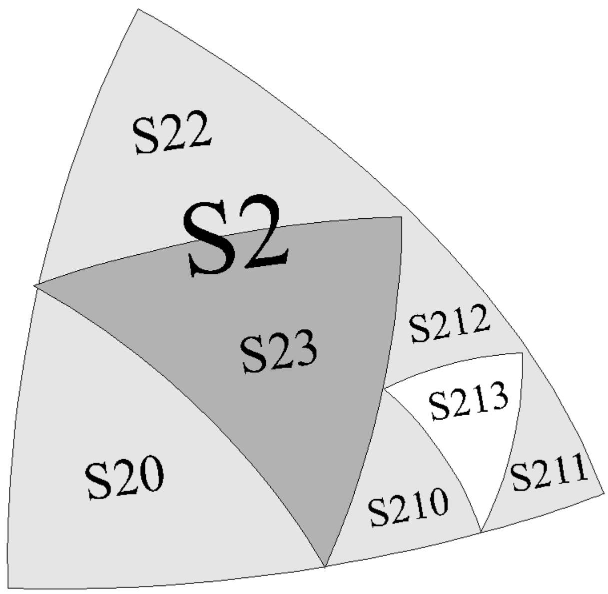

Fig. 1. An illustration of how a spherical triangle S2 is subdivided into four smaller ones. We also indicate how one of these, S21, is subdivided further into S210 ...S213. Note, how the name keeps track of the parents of a given region.

There is an obvious complication with this (or, for that matter, with any) approach:

the density of objects is not uniform over the sky. Furthermore, the densities of different types of objects (and from different types of surveys) follow different distributions. Generally, an efficient partitioning of an archive will require an approximately balanced distribution of the number of objects per partition, a strategy already used by GSC-I.

Our approach to dealing with this complication is the following: instead of taking a given subdivision, we specify an increasingly finer hierarchy of subdivisions, where each level is fully contained within the previous one. There would be a "base set" of selected areas. Each selected area can, if necessary, be divided into a number (proposed to be 4) of approximately equal sub-areas. Each sub-area can be divided further into an additional 4 sub-areas, etc., ad infinitum.

Each base selected area would be assigned a name, perhaps something simple like S2 (S for South, 2 to id one of the four). If some archive has too many objects contained within S2 to handle comfortably, it would divide S2 into sub-areas: S20, S21, S22, and S23, for example. If S21 contained too many objects, it could be further subdivided into four more equal sub-areas designated S210,S211, S212, and S213, etc, etc. This number will be a function of the storage size of the entry per object in a given database.

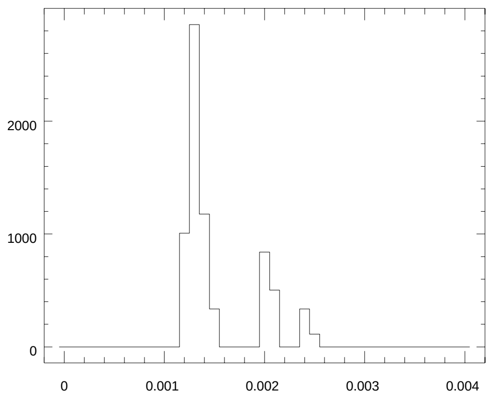

Fig. 2. This figure shows the distribution of the triangle areas of the level 5 subdivision, where the sky is divided into approximately 8000 spherical triangles, each with an area of about 5 square degrees, or 0.00157 steradians. One can see, that not all triangles are exactly equal, but also that the differences are less than a factor of 2. The area is denoted in steradians.

This way, any of these selected areas are uniquely specified by their name, even within the hierarchy – the name will tell which area is the parent, also how deep we are in the hierarchy. In such a way areas in different catalogs map either directly onto one another, or one is fully contained by another. We propose using approximately equilateral spherical triangles, where all three sides arcs of great circles. For simplicity we start from an octahedron, see Fig.3, but any other platonic solid is just as appropriate. Such subdivision schemes are well known in computer graphics (e.g. Ahn, Graphics Gems V, p. 179).

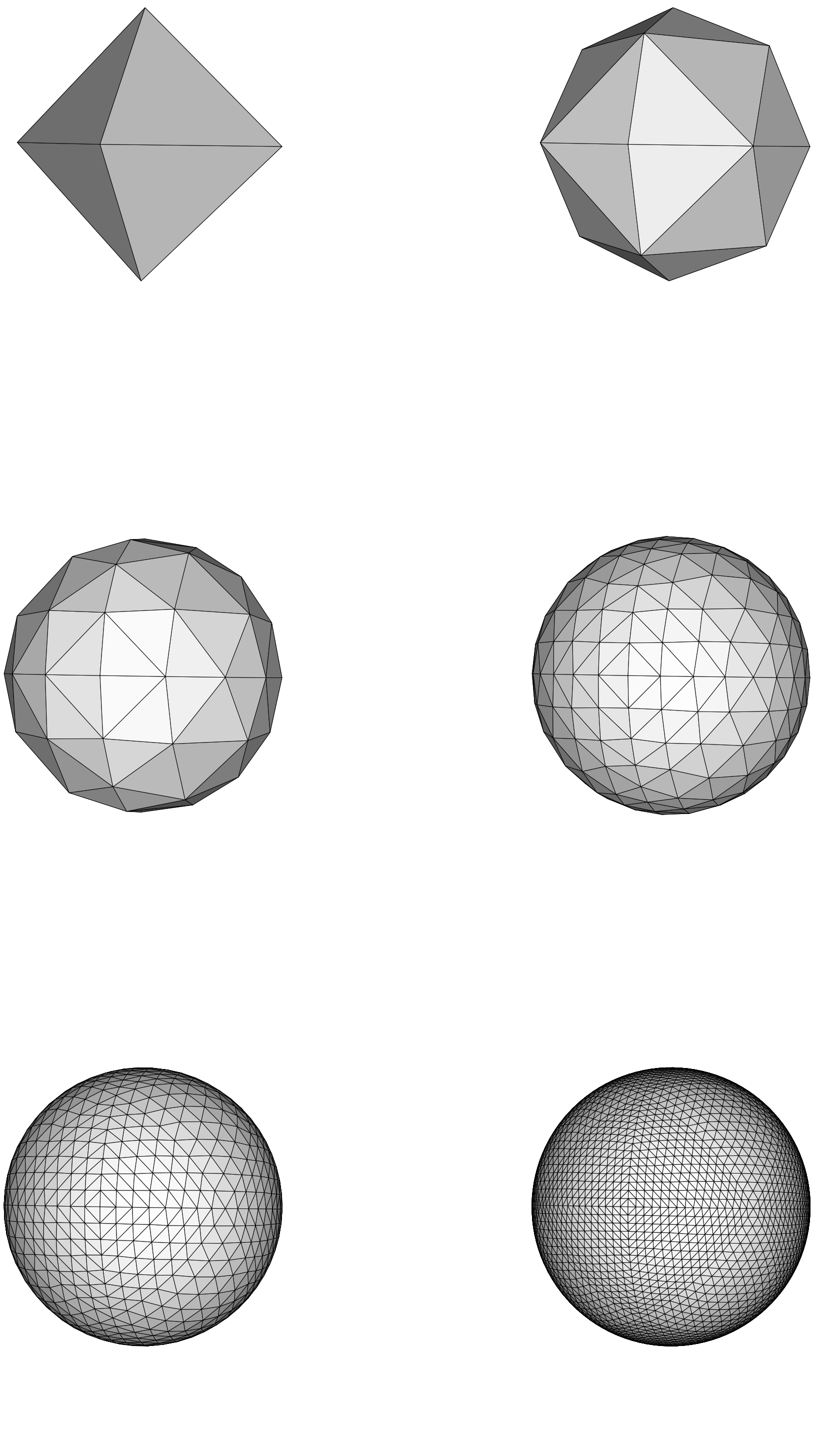

Fig. 3. A sequence of subdivisions, starting from the octahedron, down to level 5, corresponding to 8192 spherical triangles. The triangles have been plotted as planar ones for simplicity.

A rapid lookup method has been constructed to provide access to the objects in the particular areas given a location in a user's favorite coordinate system. The subroutine returns the smallest triangle which contains the given point. The typical lookup time is about 6 microseconds on a 150 MHz DEC Alpha. Another lookup is by the symbolic name of the particular triangle, this can be converted easily into a hash function, providing an even faster access. The spherical partitioning and querying concepts will be encapsulated in a public domain class library that is database and survey archive independent.

Storing and Querying Cartesian Coordinates

We would like to promote the idea of storing the coordinates in a Cartesian form, i.e. as a triplet of x, y, z values per object. The x, y, z numbers represent the position of objects on the unit sphere. While at first this may seem to increase the required storage (three numbers per object vs two), it makes querying the database for objects within certain area(s) of the celestial sphere, or involving different coordinate systems MUCH more efficient. Again, this is by no means new, this was used highly successfully by GSC-I.

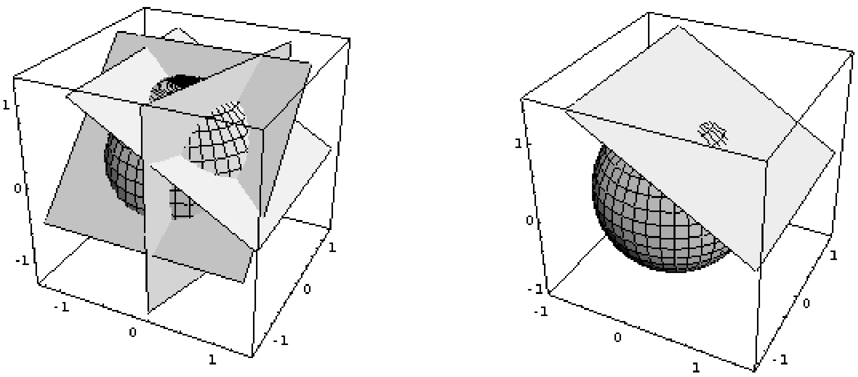

Fig. 4. Schematic representation of how the 3-dimensional Cartesian representation of coordinates make queries like objects within a certain spherical distance to a given point, or combination of constraints in multiple coordinate systems become particularly simple, since they correspond to testing linear combinations of the Cartesian coordinates instead of complicated trigonometric expressions.

To simplify the coordinate class conversion and storage, a coordinate classs library has been developed. This approach treats individual cartesian (*i.e. x, y, or z*) and angular coordinates (α or δ) as separate objects. The different coordinate systems (full cartesian coordinate, equatorial coordinate, galactic coordinate, etc ...) are then constructed from these primitive objects. Conversions are performed by functions which convert between the angular and cartesian coordinate objects. This approach allows the conversion to be

specified at run time, allowing recalibrations to be seamlessly integrated into the framework. This entire class library makes absolutley no reliance on a commercial database framework and is written in draft ANSI/ISO C++.
Most common types of queries will be like:

a. all objects within a certain spherical distance of a given point can be recast into intersections on the unit sphere with a plane
b. objects above a given latitude in any spherical coordinate system are above a three dimensional plane, intersecting the sphere at that latitude circle.
c. any great circle, thus any longitude is represented by the intersection of the sphere with a plane going through the origin.
d. Boolean combinations (AND, OR, NOT) of the above

All these queries are so-called half-space queries, where each of the constraints excludes half of the three dimensional space (but not half the sky!). Evaluating these queries is very inexpensive in terms of CPU cycles, since they all involve a computation of a linear combination rather than a spherical distance for example. Also, even though the query may contain constraints specified in several different coordinate systems, these coordinates need not be stored or even evaluated for each object, only linear combinations of the Cartesian are evaluated for every constraint. In this way, every spherical coordinate system is handled in the same fashion.

Although the internal coordinate system is really internal, we need to define what the center and orientation of the unit sphere should be – i.e. which coordinate system should it transform to. We propose the J2000 system tied ab initio to the ICRF (International Celestial Reference Frame), since all the astrometric catalogs like Hipparcos and Tycho will do things exclusively in that system.

The Next Step

The SDSS project is well along in implementing the above sky partitioning scheme in a C++ class library, which can be run outside the database, to make life for the different projects concievably using different databases easier. The class library will automatically handle all the different queries above, and will be able to return spherical coordinates in an arbitrary system. The class library will be made available to interested groups.

The partitioning scheme will be extended in the near future to handle linear combinations of magnitudes, size, surface brigthness, etc., where each project can go its own way, but there will be a general class library available, which can easily be customized for the particular cases.

Points of common interest that will be addressed in the future include this color partitioning, query algorithm sharing, magnitude definitions, and survey metadata standardization. We hope that this document is only the first in a series, and will start a useful discussion about how to achieve a true interoperability in the next generation archives.

References

Potential Applications of GSC-II for GAIA Operations, B.M.Lasker, B.J.McLean, H.Jenkner, M.G.Lattanzi and A.Spagna 1995, Proceedings of the workshop - "Future Possibilites for Astrometry in Space", p137. ESA Sp-379