TITANBROWSE: USING A NEW PARADIGM FOR ACCESS TO HYPERSPECTRAL DATA. P. Penteado[1] and D. Trilling, [1]Department of Physics and Astronomy, Northern Arizona University, NAU Box 6010, Flagstaff AZ 86011-6010, pp.penteado@gmail.com.

Introduction: There are many tools and standards that allow immediate retrieval of a variety of astronomical observations, mostly derived from the VO paradigm, but these are not applicable to remote sensing observations of Solar System bodies. While astronomical observations are confined to a 2D spatial domain (coordinates in the sky), with all objects always observed from essentially the same point of view, remote sensing observations span different perspectives. More elaborate tools are needed, both to evaluate the geometrical conditions of the observations, and to archive them in an accessible way. Here we present titanbrowse, a database, exploration and visualization system which solves these difficulties, to enable full use of all Cassini VIMS observations of Titan.

Cassini VIMS observations of Titan: VIMS is an imaging spectrometer that records in each observation (commonly called a data cube) 352 bands at 64×64 pixels each. Since the arrival of Cassini at the Saturn system, in 2004, until July 2010, approximately 2×10^4 cubes comprising 15×10^6 spectra were recorded. This makes it impossible for a user to directly examine all the spectra. A simple cube database, such as provided by PDS, is not enough, due to several limitations: 1) In each cube there is typically a large variation in observation geometry over the spatial pixels. Thus, often the most useful unit for selecting observations of interest is not a whole cube but rather a single spectrum (one spatial pixel). 2) The pixel-specific geometric data included in the standard pipelines has too few variables calculated (such as latitude, longitude, and illumination angles), and all the geometry is calculated at only one point per pixel. Particularly for observations near the limb, or at high relative velocities, it is necessary to know the actual extent of each pixel. 3) It is not possible to identify all the spectral features of interest by direct inspection. Thus, it becomes necessary to make database queries not only by metadata, but also by the spectral data. For instance, one query might look for atypical values of some band, or atypical relations between bands, denoting spectral features (such as ratios or differences between bands). 4) There is the need to evaluate arbitrary, dynamically-defined, complex functions of the data (beyond just simple arithmetic operations), both for selection in the queries, and for visualization, to interactively tune the queries to the observations of interest. 5) The process of making the most useful query for some analysis is typically interactive, with the user needing to explore how different functions of the data vary over the observations. This requires an efficient database (so that the queries are fast), and integration with a visualization system, so that queries can be quickly interactively changed. Having to export data to files, then import them into a visualization system, would usually make this process too slow and inefficient

Titanbrowse: These problems were the motivation for the development of a new database system for hyperspectral observations of planetary bodies, called titanbrowse, since we created it for Cassini VIMS observations of Titan [1]. The same solutions can be readily adapted for observations of other bodies by other instruments. The framework is particularly well-suited for other imaging spectrometers, currently a standard instrument on every Solar System exploration mission.

Implementation. We found that standard relational database software lacks key functionality needed to fulfill the requirements above. Titanbrowse was implemented in Interactive Data Language (IDL), since it provides these necessary features: 1) Efficient array processing and advanced array semantics: Archived hyperspectral data from cube files fit well into a read-only two-table database (one table for cubes, another for spatial pixels), where queries and processing can be well handled through vector operations on these tables, for which IDL is extremely well-suited. 2) The possibility of dynamically evaluating arbitrary functions of the variables (table columns). These functions are used both as the search criteria in the queries, and to interactively inspect the results and adjust the queries, and go beyond simple arithmetic and logical expressions (as in typical database systems). Changes can be freely experimented with, while inspecting the results interactively, since the functions are dynamically compiled at runtime. 3) Titanbrowse includes visualization tools, integrated to the database interface, so that results can be immediately inspected. These tools were built using IDL's standard library functions for the graphical interface and the cartographic projections. Figure 1 shows a screenshot of the current interface. 4) The creation of the functions of interest to make queries and visualize data can make use of the extensive built-in IDL library, with many functions common to scientific processing, but not usually found in database systems. For instance, the query or the visualization might use a function that calculates the area of an absorption band in the spectrum, or spherical geometry functions, or some arbitrary user-defined spectral indicator (such as a derived from Principal Component Analysis (PCA), integral transforms, or functions derived from theoretical models).

Results: Among other uses, this system has allowed us to discover the first tropical lakes on Titan [2]: Cassini observations had long before shown that Titan has large methane lakes on its polar regions [3]. Despite considerable community interest in searching for them in the tropical regions — where the Huygens probe landed on a dry lake bed — none had been found, until we used titanbrowse to search for them. Experimenting with functions to query the whole database, we were able to select those where the spectra indicated the presence of a lake. This revealed that, out of the millions of spectra in the database, a few dozen, all of a small region in the tropics, were consistent with the surface being covered in liquid. The data used in this study had been public for years, but since inspecting all the spectra was not feasible without a tool like titanbrowse, the lakes had remained undetected.

Current development: We are working on an online accessible version of titanbrowse. This will remove the users' need to download and install software and data. Users will be able to perform complex queries with integrated visualization using only a Java client, accessible through a web browser. The server will use IDL to access the data and process the queries, communicating the results to the Java server through a Java bridge. This demonstration server will be deployed on an Amazon Web Services server, so that it can be easily reconfigured, redeployed or scaled as needed, with a very low cost for the development and demonstration stages.

References:

[1] Penteado, P.F. (2009) *PhD Thesis, The University of Arizona,* [2] Griffith, C.. et al. (2012) *Nature, 486,* 237. [3] Stofan, E. et al. (2007) *Nature,* 445, 61..

Additional Information: More information about titanbrowse, including, when available, its online implementation, can be found at

http://ppenteado.net/titanbrowse

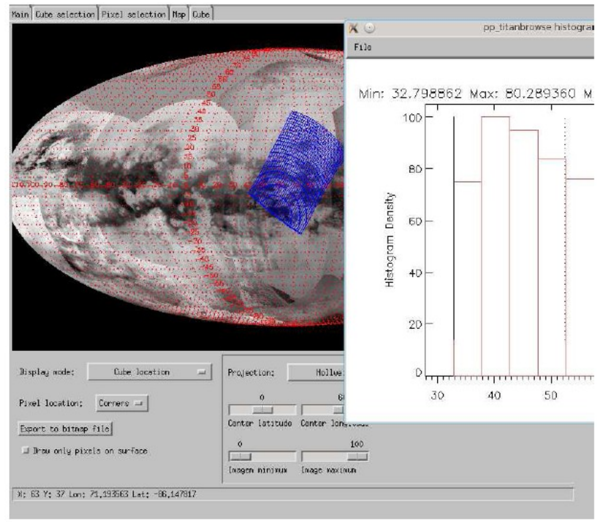

Figure 1: Example of the current titanbrowse graphical interface. The visualization on the left is an interactive map widget, in this case showing the location of the spectra currently selected by a query (pixel edges drawn as blue lines over the grayscale map). The visualization partly shown on the right side is a histogram widget, showing the result from evaluating a user-defined function, for each selected spectrum. This assists the user when defining queries, to determine adequate ranges of values, and when exploring data, looking for clusters or outliers. The histogram and the map are drawn automatically, inside the user interface, without requiring downloads. Use of these two visualizations of query results and function evaluations is essentially the method we used to discover the first tropical lakes on Titan [2].