THE UNIVERSITY OF IOWA

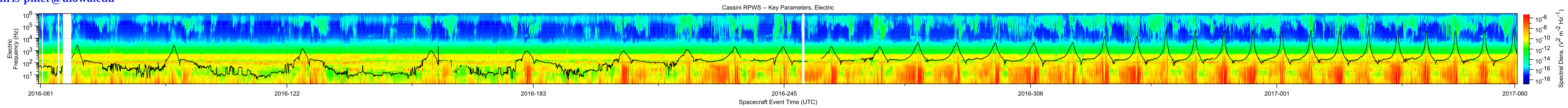
chris-piker@uiowa.edu

IN11C-0047: Efficient Navigation of Long Time-Series Data Sets Using Das2 Tools

C.W. Piker, L.J. Granroth, J.B. Faden, E.E. West,W.S. Kurth   Department of Physics and Astronomy, University of Iowa, Iowa City, Iowa, USA

Cassini-Huygens Mission To Saturn and Titan

das2 http://das2.org



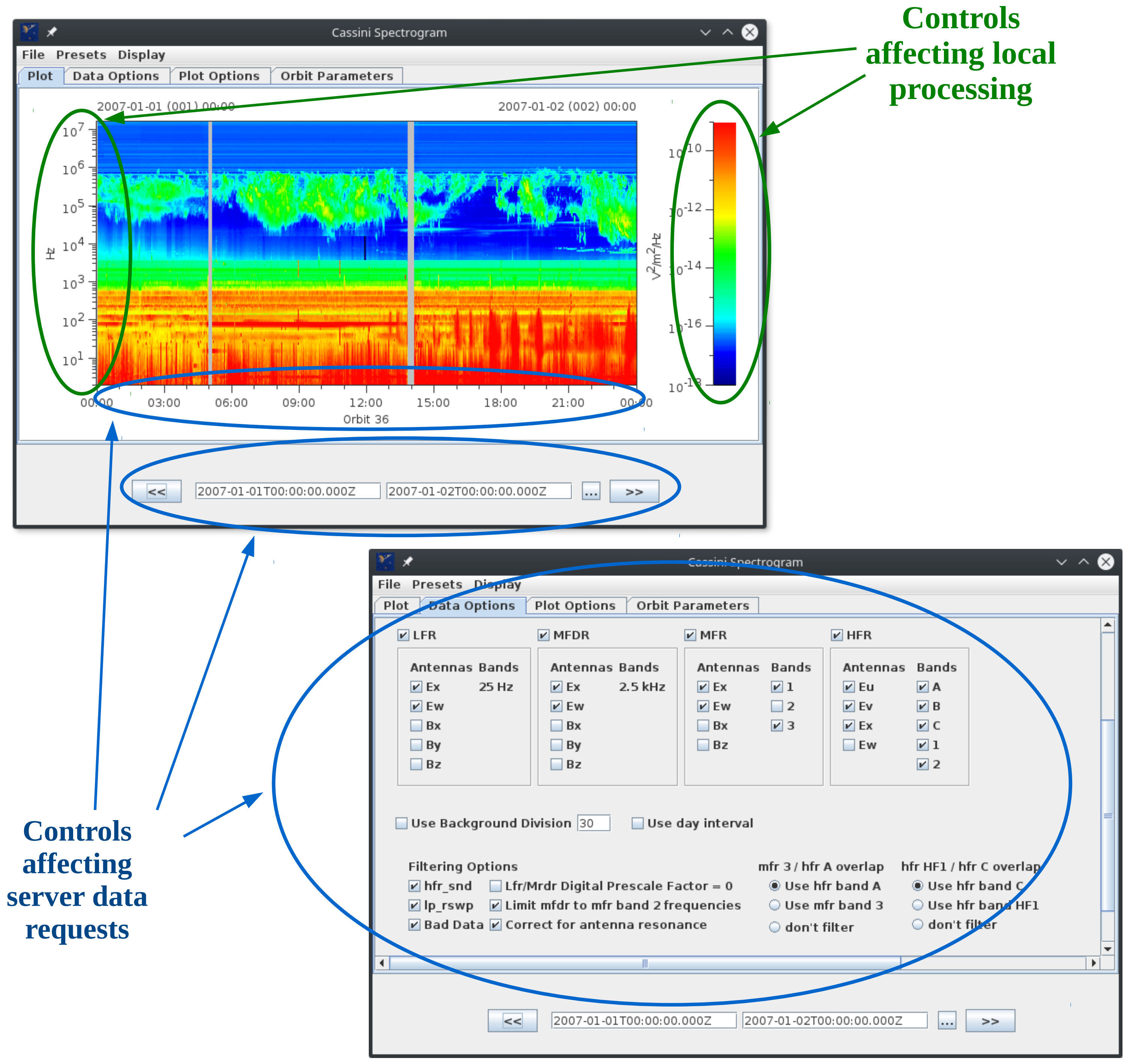
For over 14 years the University of Iowa Radio and Plasma Wave group has utilized a network transparent data streaming and visualization system for most daily data review and collaboration activities. This system, called Das2, was originally designed in support of the **Cassini** RPWS (Radio and Plasma Wave System) investigation, but is now relied upon for daily review and analysis of **Voyager**, **Polar**, **Cluster**, **Mars Express**, **Juno** and other mission results. The system has recently been adopted for low-frequency ground based **radio astronomy** results.

Das2 Design Parameters

- **Scientists care about data and plots, not files.**
  Reader programs hide file format and naming convention details from the rest of the system.
- **Thick clients are required for responsive user interfaces.**
  Custom Java client library and applications render data locally and convert user input into data requests. The server is not required for many operations.
- **Very large datasets can be handled by modest hardware if processed a record at a time.**
  Das2 Stream format transmits a single slice across multiple arrays in each packet. As new array types are encountered in the input files, new array definitions are pushed onto the stream.
- **The number of pixels in a plot sets an upper limit on the amount of data that must be transmitted.**
  Das2 clients can send required time resolution with each data request. If no resolution is specified, the server returns the full resolution output of the reader program.

Das2 Clients

- **dasCore - Java Client Library**
  To provide a responsive user interface, a client library was developed in Java. Most functionality is contained in the core library, client programs were built on this base.
- **Specialized Client Applications**
  Custom applications built on dasCore provide interfaces for **Juno** - Waves, **Cassini** - RPWS, **Voyager** - PWS, **Galileo** - PWS, **Mars Express** - MARSIS, and multiple Earth orbiting missions. http://das2.org/demo-apps.html



2007-01-01T00:00:00.000Z 2007-01-02T00:00:00.000Z

☑ LFR — Antennas: ☑ Ex, ☑ Ew, □ Bx, □ By, □ Bz; Bands: 25 Hz

☑ MFDR — Antennas: ☑ Ex, ☑ Ew, □ Bx, □ By, □ Bz; Bands: 2.5 kHz

☑ MFR — Antennas: ☑ Ex, ☑ Ew, □ Bx, □ Bz; Bands: ☑ 1, □ 2, ☑ 3

☑ HFR — Antennas: ☑ Eu, ☑ Ev, ☑ Ex, □ Ew; Bands: ☑ A, ☑ B, ☑ C, ☑ 1, ☑ 2

□ Use Background Division 30   □ Use day interval

Filtering Options
☑ hfr_snd   □ Lfr/Mrdr Digital Prescale Factor = 0
☑ lp_rswp   ☑ Limit mfdr to mfr band 2 frequencies
☑ Bad Data   ☑ Correct for antenna resonance

mfr 3 / hfr A overlap
● Use hfr band A
○ Use mfr band 3
○ don't filter

hfr HF1 / hfr C overlap
● Use hfr band C
○ Use hfr band HF1
○ don't filter

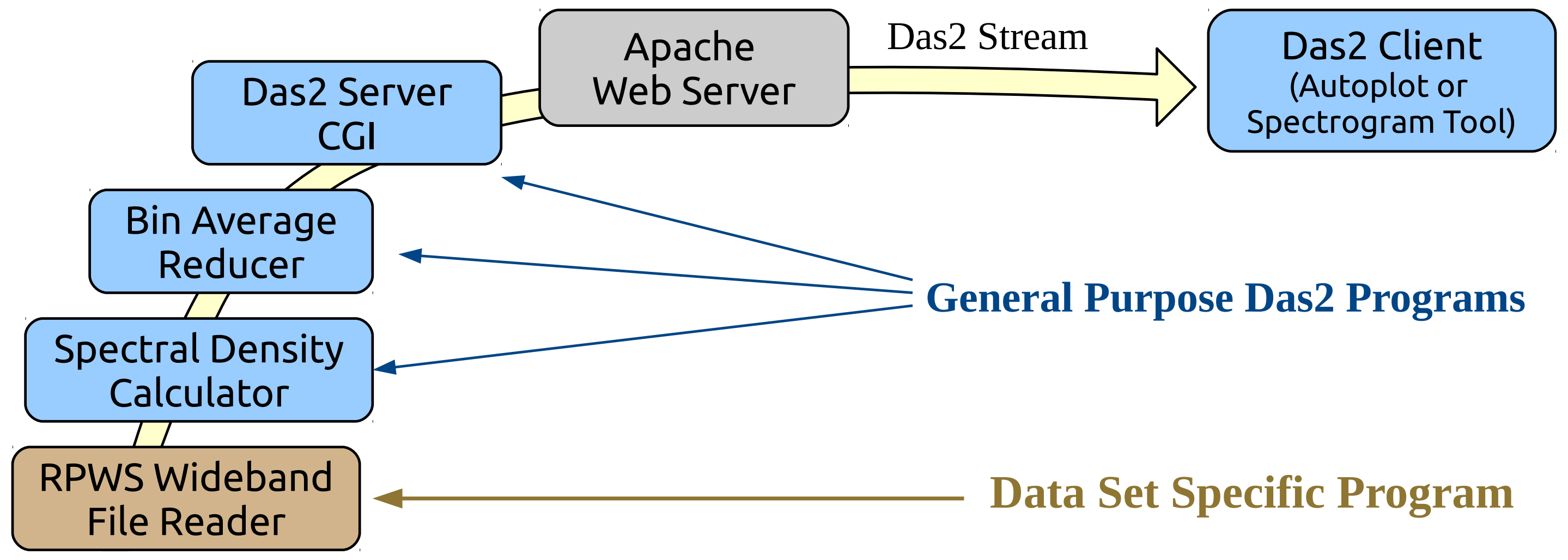
2007-01-01T00:00:00.000Z 2007-01-02T00:00:00.000Z

Controls affecting local processing

Controls affecting server data requests

- **General Client, Autoplot**
  Autoplot, originally built around a modified dasCore, is the most common Das2 client used locally. Integrating the extensive parameter handling needed to completely replace RPWS specific clients is currently underway. http://autoplot.org

Das2 Server Reduces Transmitted Data Volume


Apache Web Server

Das2 Stream

Das2 Client (Autoplot or Spectrogram Tool)

Das2 Server CGI

Bin Average Reducer

Spectral Density Calculator

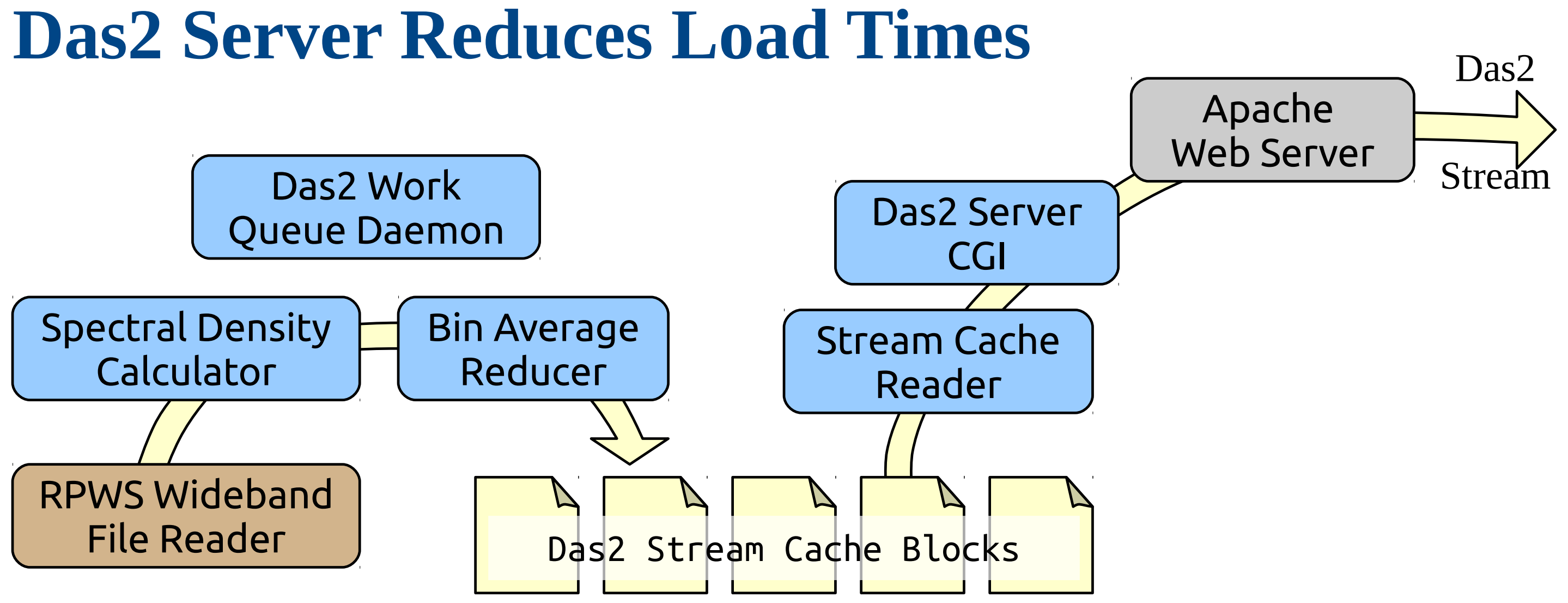
RPWS Wideband File Reader

General Purpose Das2 Programs

Data Set Specific Program

Server averages values into bins no greater than the time duration of a single pixel. Reductions normally preformed by a client are handled before data are transmitted. This reduces network bandwidth and client memory footprint, but reading gigabytes of files on the server is still slow.

Das2 Server Reduces Load Times


Apache Web Server

Das2 Stream

Das2 Work Queue Daemon

Das2 Server CGI

Spectral Density Calculator

Bin Average Reducer

Stream Cache Reader

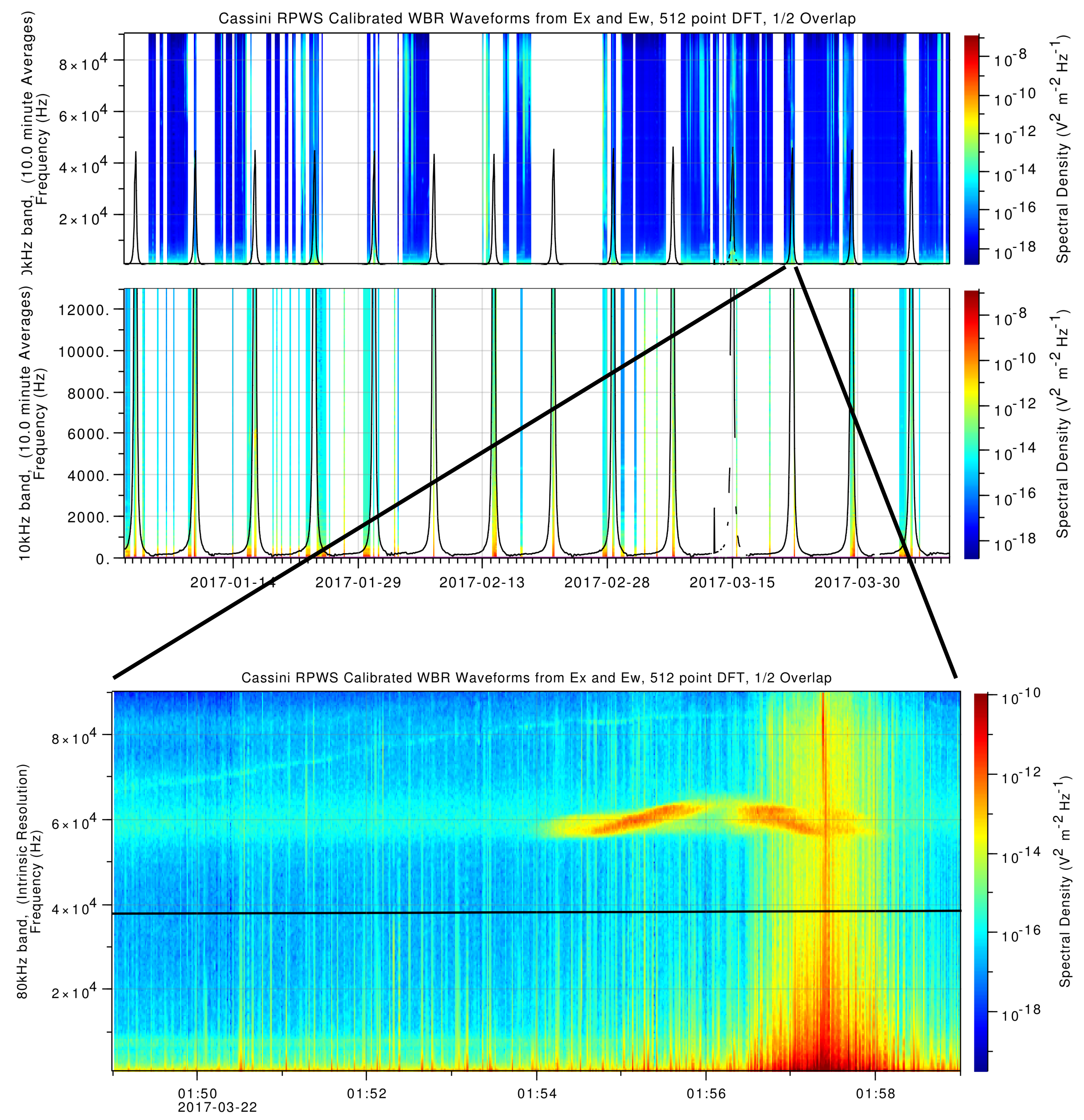
RPWS Wideband File Reader

Das2 Stream Cache Blocks

Work Queue Daemon handles the creation of Das2 Stream files in a handful of useful resolutions in the background. Later when a client requests data, cache blocks that were generated at, **or greater than**, the needed resolution are output. Expensive full resolution file reads are avoided for large time ranges.

As a Result, Latency Becomes Scale Independent

100 day plots backed by 3.4 GB of data transfer and display in 2.1 seconds...



...10 minute high-rate science plot at intrinsic resolution in 3.0 seconds.
Coverage plots are no longer essential since data may be displayed for any time range.

Display Latency Measurements
The 100 Day plots are of RPWS 10kHz and 80kHz spectra with 1 minute resolution f_{ce} and f_{pe} overlays. Original RPWS data occupy over 3.3 GB of disk space in binary 16-bit integer format. Transfer and display time were measured using a 20 Mbps DSL link on 2010 era PC with Autoplot version 20170905a. Intrinsic resolution is typically 2048 captures/sec. in the single spectrogram plot.

Software and Documentation

All General Das2 software is available under the **open source** LGPL license.

Das2 is an interface definition, **not an API**, software may be written in any language. Current interface specification is at: http://das2.org

Libraries

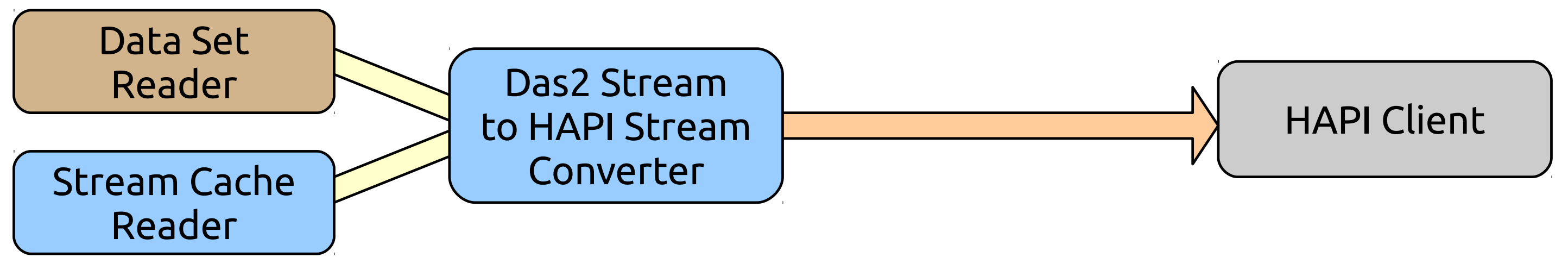
- **dasQCore**, this central Autoplot **Java** library contains components for reading and plotting Das2 Streams. https://saturn.physics.uiowa.edu/svn/das2/core/stable/dasQCore
- **libdas2,** a **C** based library and programs providing parsing, binning, and spectrum calculations. Includes a Python **Numpy** extension. **IDL** bindings in production will allow for **SPEDAS** interoperability. https://saturn.physics.uiowa.edu/svn/das2/core/stable/libdas2

Applications

- **Autoplot** is the recommended GUI tool for reading Das2 streams, may also be used as a library to interface with IDL and Matlab. http://autoplot.org
- **pyServer** is a Python based HTTP request handler and work queue manager https://saturn.physics.uiowa.edu/svn/das2/servers/stable/pyserver

HAPI Support

As of Dec. 3rd 2017 the development branch of the Das2 pyServer software supplies HAPI 1.1 end points using a stream converter.


Data Set Reader

Stream Cache Reader

Das2 Stream to HAPI Stream Converter

HAPI Client

Das2 streams allow multiple array sets to be interleaved in a single stream. Since this is not allowed by the limited HAPI protocol, not all Das2 Streams can be converted to a HAPI stream. The pyServer tries to get around this by splitting one Das2 data source into multiple HAPI nodes each offering a single array set, but this is not always sufficient Especially problematic are the RPWS Low-Rate Complete and RPWS high-rate science waveform data sources.

Lessons Learned, Future Design Parameters

- **Scientists care about data and plots, not servers.**
  Data catalogs should work much like the Internet's Domain Name System (DNS) i.e. a central top level catalog with sub-catalogs devolving responsibility to nodes closest to the data.
- **Time isn't the only navigation parameter.**
  Data source definitions should specify the set of HTTP GET query parameters available and denote which parameters constrain which data coordinate ranges.
- **Text is a useful data type.**
  Stream format is too limited, event lists and scientist's annotations are very useful. Non-numeric data should be supported in the stream definition.
- **Static files on a website can make a useful server.**
  Allowing file aggregations in data source definitions would make ancillary data, such as orbit definitions, discoverable and small data sets easier to incorporate.
- **Download isn't the only flow direction.**
  Provide a service to accept uploaded streams of digitized measurements and annotations.
- **Our phones are computers too.**
  Clients built from the ground up for mobile environments should be considered.

Acknowledgments
Larry Granroth created the precursor Das1 system and organized the creation of Das2
Edward West and **Jeremy Faden** built the Das2 Java client library and the Cassini RPWS Spectrograms tool, which provided an initial foundation for Autoplot
Joseph Groene created many SPICE readers providing spacecraft location data as a Das2 Stream
William Kurth tests our software daily and is understanding when we break things
Prof. Don Gurnett for having the foresight to retain full time staff programmers and for providing funding under Cassini RPWS instrument activities.
The Cassini-Huygens Mission to Saturn and Titan