Interoperability for Heliophysics and Planetary Time Series Data via HAPI

IN017-0006

AGU FALL MEETING
Online Everywhere | 1–17 December 2020

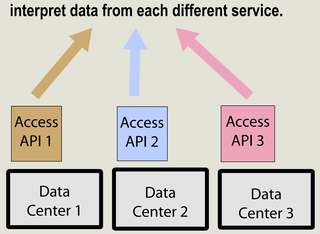
JOHNS HOPKINS APPLIED PHYSICS LABORATORY

Jon Vandegriff (JHU/APL), Robert S Weigel (GMU), Jeremy Faden, D Aaron Roberts (GSFC), Todd A King (UCLA/ESS), Bernard T Harris (GSFC), E. W. Grimes (UCLA), Scott A Boardsen (GSFC), Robert M Candey (GSFC), Nand Lal (GSFC), Douglas Lindholm ((LASP), Chris Lindholm (LASP), In Sook Moon (ICLA), Lawrence E Brown (JHU/APL)

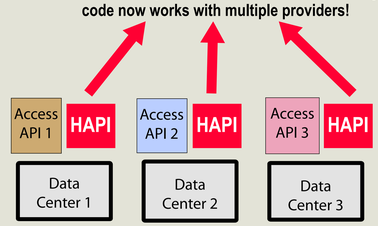
What is HAPI?
Heliophysics Application Programmer's Interface

a standardized server access interface for time series data;
a lowest common denominator that any data center can add;
a single interface clients can use to access data from many places

Lots of data wrangling needed to interpret data from each different service.



A standardized access interface simplifies the analysis of data from different places since the same data wrangling code now works with multiple providers!



Each data center can keep their custom access mechanism.

HAPI Interface Details

A HAPI server has four endpoints:

http://server/hapi/capabilities
http://server/hapi/catalog →
http://server/hapi/info
http://server/hapi/data

http://server/**hapi/catalog**

Returns a JSON list of dataset identifiers from this server.

Takes no request parameters.

A human readable title for each dataset is optional.

Dataset identifiers need to be unique for any given server.

```
{
   "HAPI" : "2.0",
   "status": {"code": 1200, "message": "OK"},
   "catalog" :
   [
      {"id": "ACE_MAG",
        title:"ACE Magnetometer data"
      },
      {"id": "data/IBEX/ENA/AVG5MIN"},
      {"id": "data/CRUISE/PLS"},
      {"id": "any_identifier_here"}
   ]
}
```

A data request looks like this:

http://datashop.elasticbeanstalk.com/hapi/data?
id=CASSINI_LEMMS_PHA_CHANNEL_1_MIN_AVG&
time.min=2004-07-01T06:00:00.000Z&
time.max=2004-07-01T07:00:00.000Z&
include=header

Try it now!

Full details are in the specification document.

`https://github.com/hapi-server/data-specification`

Server and Client Implementations

Generic Server - a HAPI front end that is ready to use with only back-end plug-ins needed for your data.

`https://github.com/hapi-server/server-nodejs`

Server Verifier `http://hapi-server.org/verify`

- a mechanism for testing the validity of a server
- can be run locally or an online version is available
- has been very useful for new server developers!

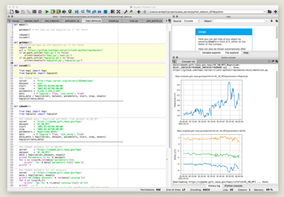
Live Data Servers `http://hapi-server.org/servers`

See list below for institutions serving data via HAPI

Client Options -- reading data from HAPI servers

Python Client - create your own read routines

`https://github.com/hapi-server/client-python`

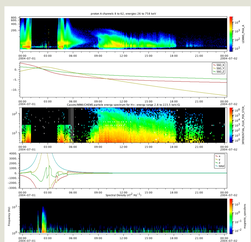


Demonstration code that reads from HAPI servers. The plotting is just for demo purposes.

A few lines of code can read an arbitrary time range into a Python data structure that will be the same for any data source.

Autoplot - Java visualization tool

`http://autoplot.org`



SPEDAS - IDL suite that reads many data sources, including HAPI

`http://spedas.org`

Load data using GUI

Simple code to get HAPI data from any server.

```
hapi_load_data, /catalog, server='http://datashop.elasticbeanstalk.com/hapi'

hapi_load_data, /info, dataset='LEMMS_PHA_CHANNEL1_FULL_TIME_RES', $
          server='http://datashop.elasticbeanstalk.com/hapi'

hapi_load_data, trange=['04-07-01', '04-07-02'], $
          dataset='LEMMS_PHA_CHANNEL1_FULL_TIME_RES', $
          server='http://datashop.elasticbeanstalk.com/hapi'

tplot, 'proton_spectrum'
```

Server Status

These institutions have implemented HAPI servers.

Server	Number of datasets	Implementation	Months to create	Willing to update?	Current usage
CDAWeb at GSFC	2528	new code, impl. spec from scratch		yes	>60,000 in 2020
SSCWeb pass-thru	236	node.js front end; live SSCWeb access	0.2	yes	some
LISIRD at LASP	34	mods to LaTiS code base (Scala)	~1	yes	light
Univ. of Iowa	19	mods to Das2 server (Java)	0.375	yes	light
DataShop (APL)	9	Java custom implementation	2	yes	light
FunTech	20	C# custom implementation	2-3	probably	light
AMDA	640	node.js front end; custom back end	0.17	probably	just started
CCMC / ISWA	*147*	*works now; recent compliance in progress*	*<1*	*yes*	*some*
PDS / PPI Node		*in progress*			
ESAC / CSA		*planned for early 2021; use node.js at first*			

HAPI 3.0 – expected changes

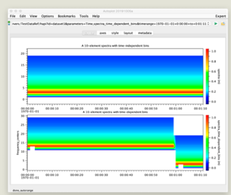
Issue numbers (shown in parentheses) can be looked up on Githuib:
https://github.com/hapi-server/data-specification/issues

- (77) the request interface changes from this:
  http://server/hapi/data?**id**=ACE_MAG&**time.min**=2004-001Z&**time.max**=2005-001Z&parameters=mag_GSE
  to this
  http://server/hapi/data?**dataset**=ACE_MAG&**start**=2004-001Z&**stop**=2005-001Z&parameters=mag_GSE
  (and similarly for the `info` request -- change **id** to **dataset**)
- (94) new endpoint: **server.org/hapi/about**
  - JSON metadata about server, including a way for server to advertise ability to handle multiple requests simultaneously
- (81) add optional unitsSchema to dataset info with string indicator telling which units framework is used
  - The three that we know about right now are:
    - UDUNITS2.2.26, AstroPy units (current AstroPy is 3.2.1), the CUNITS conventions used in CDF files
  - The unitsSchema values for each of these would be
    - udunits2 astropy3 cdfunits
- (82) allow references in metadata (JSON info response); reduces duplicate info seen in some bin ranges
- (83) allow time-varying bins – ranges or centers can come from data parameter that change with time
- (97) clarification of some error states (error 1405: time range outside valid range is not actually needed)

Data with time-varying bin ranges will be supported.

For many data sets, the energy ranges of frequency ranges of spectra are fixed over time. But some data sets have spectra whose bins change with time. Plotted here are two toy data sets, one with fixed bins over time and the second with bins that change near the end of the time range.

```
{   "HAPI": "3.0",
    "status": {"code": 1200, "message": "OK"},
    "startDate": "2016-01-01T00:00:00.000Z",
    "stopDate": "2016-01-31T24:00:00.000Z",
    "parameters":
      [ { "name": "Time", "type": "isotime",
          "units": "UTC", "fill": null,"length": 24 },
        { "name": "proton_spectrum","type": "double",
          "size": [16], "units": "particles/(sec ster cm^2 keV)",
          "fill": "-1e31",
          "bins": [
                { "name": "energy",  "units": "keV",
                  "ranges": "energy_ranges" } ] },
        {
          "name": "energy_ranges", "type": "double",
          "size": [16,2],
          "units": "keV", // same as proton_spectrum units
          "fill": "-1e31"
        }
      ]
}
```



For time varying bins, the bin ranges (or centers) refer not to fixed values, but to a variable who value at each time step will be used to obtain the bin ranges (or centers).

To receive HAPI update emails:
send a request to: `hapi-news@hapi-server.org`