
IVOA

@


IRSA

ipac

NASA/IPAC Infrared Science Archive

IRSA | Data sets | Search | Tools | Help

Login

Search for Source

Name or Coordinates

Search

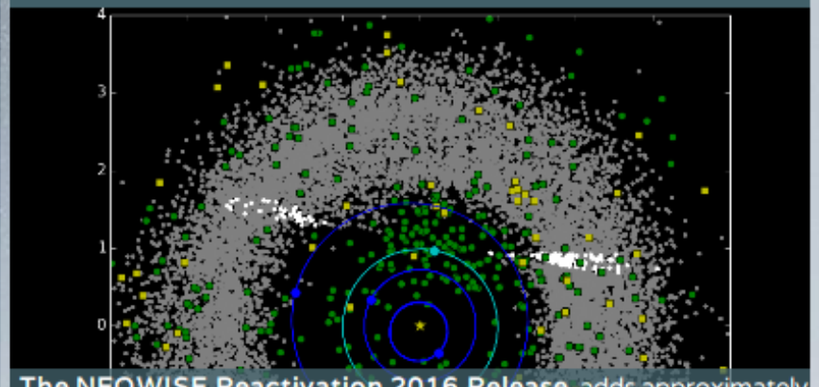
Radius 10 arcsec

Search Catalog: WISE

Search

Catalogs | Images | Finder Chart | {http://} VO/API

NEOWISE Reactivation 2016 Data Release



The NEOWISE Reactivation 2016 Release adds approximately 2.56 million image sets and 19.7 billion source detections to the public archive. These data were acquired during the second year of the NEOWISE Reactivation Mission.

Past News | Featured Images

Spitzer | WISE | Herschel | Planck | 2MASS | IRAS | COSMOS | Less ▲

AKARI | BLAST | BOLOCAM | IRTS | ISO | MSX | PTF | SWAS

VO services at IRSA

- Simple Image Access Protocol v1 (SIA)
- Simple Cone Search (SCS)
- Table Access Protocol (TAP)
- Publishing Registry

Data Formats

Users can upload

- IPAC Table
- VOTable
- HDF5
- FITS
- JSON*
- JSON5*

Users can fetch

- FITS
- IPAC Table
- HDF5
- VOTable
- JSON*
- JSON5*
- CSV
- TSV
- Text
- HTML

- A canonical mapping from VOTable to JSON would be nice.

Simple VO services

- Simple Image Access Protocol v1 (SIA)
  - 368 Tables, 53 Million Rows, 93 Columns

Simple VO services

- Simple Image Access Protocol v1 (SIA)
  - 368 Tables, 53 Million Rows, 93 Columns
- Simple Cone Search (SCS)
  - 411 Tables, 118 Billion Rows, 123 Columns
  - Biggest table is 42 Billion Rows, 48 Columns

Table Access Protocol

- sync and async
- 451 Tables, 119 Billion Rows, 123 Columns

Table Access Protocol

- sync and async
- 451 Tables, 119 Billion Rows, 123 Columns
- +72 tables, 613 Billion Rows, 7 Columns in a non-RDBMS backend

Table Access Protocol

- sync and async
- 451 Tables, 119 Billion Rows, 123 Columns
- +72 tables, 613 Billion Rows, 7 Columns in a non-RDBMS backend
- Largest VO provider in the world by rows or rows*columns?

TAP Enables Users

- TAP covers pretty much all of the custom, off the beaten track queries that users want to do.

TAP Enables Users

- TAP covers pretty much all of the custom, off the beaten track queries that users want to do.
- Our ADQL implementation covers all of the ObsCore uses cases, but does not support the more esoteric geometry.

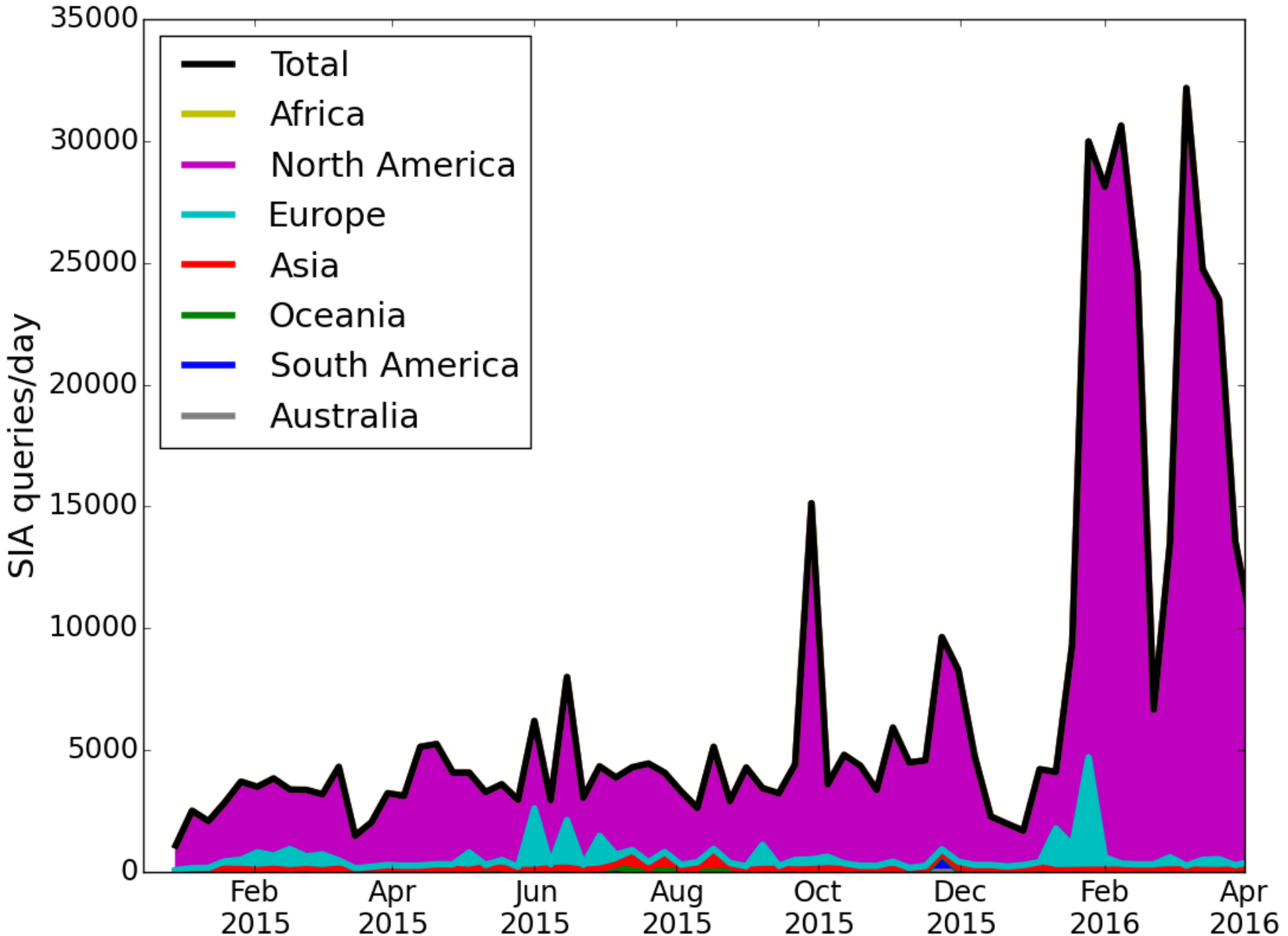
TAP Enables Users

- TAP covers pretty much all of the custom, off the beaten track queries that users want to do.
- Our ADQL implementation covers all of the ObsCore uses cases, but does not support the more esoteric geometry.
- Users seem fine with that

TAP/async

- Our implementation only supports running jobs immediately.
- The spec does not require services to handle running jobs immediately.
- TOPCAT can not rely on being able to run jobs immediately.
- So TOPCAT can not submit jobs to IRSA, but it can monitor them.
  - Users do care when they can not use TOPCAT

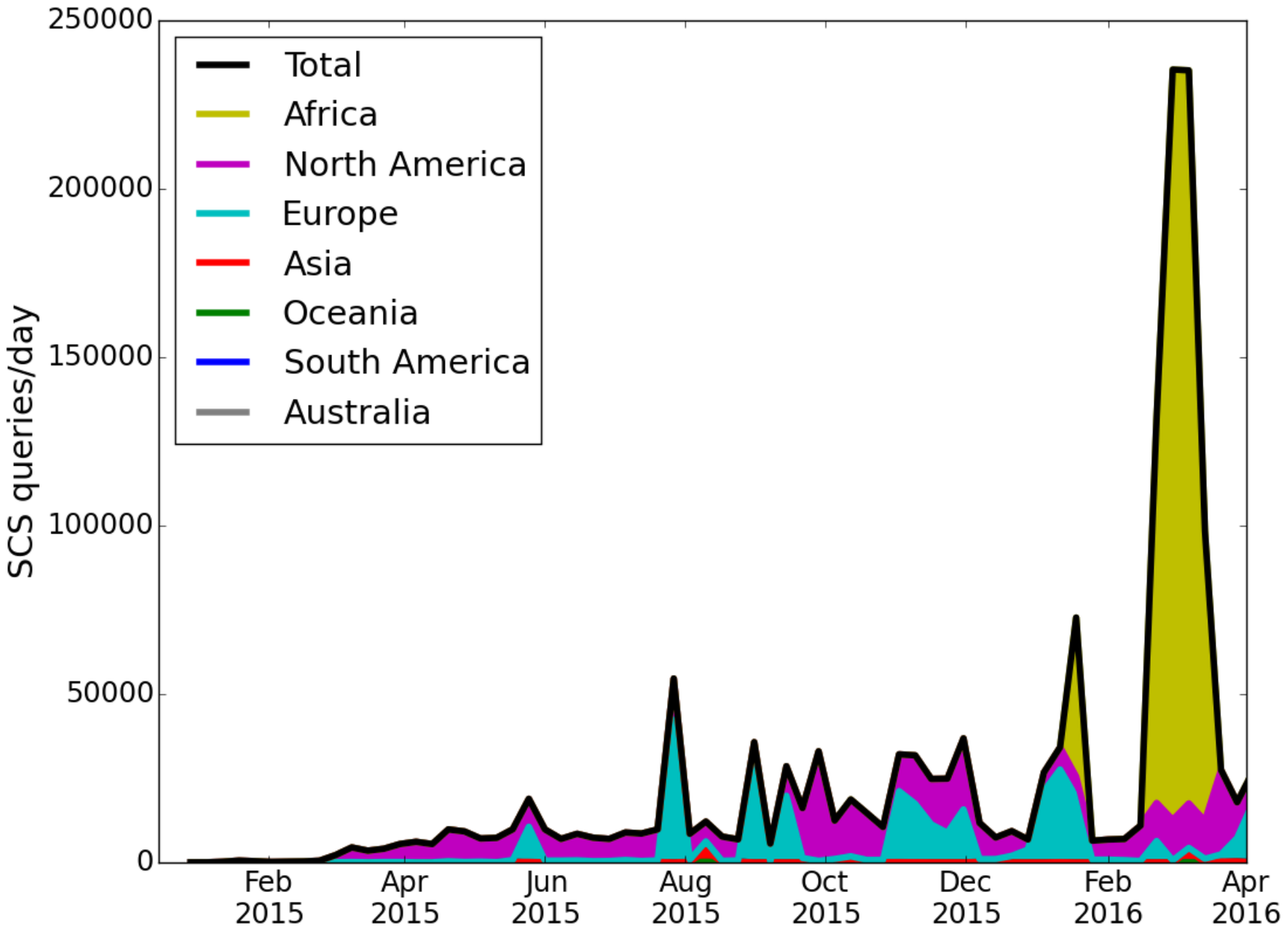
Simple Image Access Traffic



SIA Traffic in 2015

- We got really popular in Montreal, CA in 2016.
- A few new releases, some significant
  (e.g. NEOWISER Year 2, 2MASS mosaics, PTF)
- Per quarter
  - 300-400 unique IP's
  - ~200 unique subnets
    (e.g. 134.4.xxx.xxx ~ IPAC)

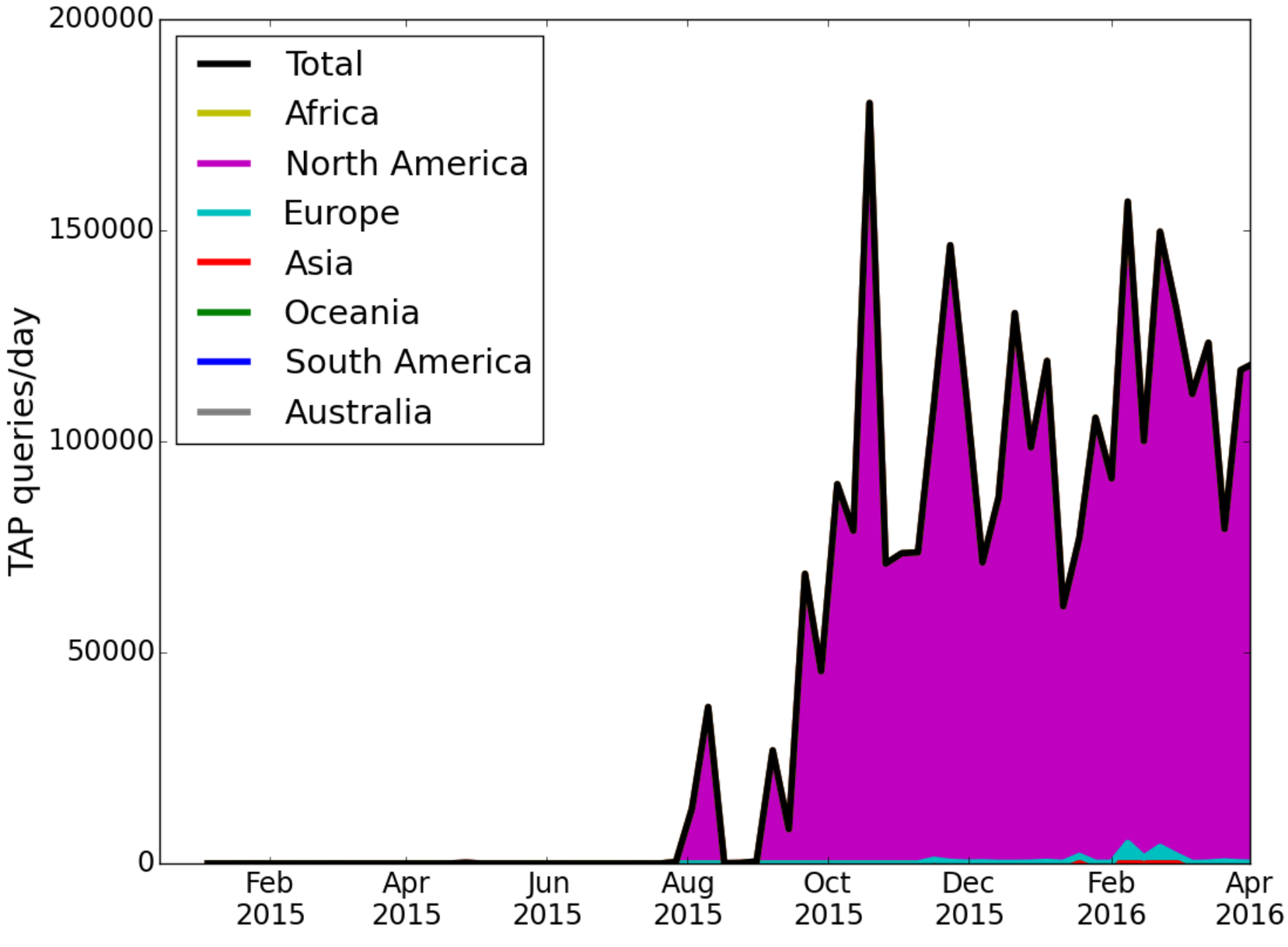
Simple Cone Search Traffic



SCS Traffic in 2015

- Very enthusiastic TOPCAT user from South Africa
- Per quarter
  - 150-300 unique IP's
  - 100-150 unique subnets

TAP/sync Traffic



TAP/sync Traffic in 2015

- Numbers dominated by a single NEOWISE-R scientist doing some special research

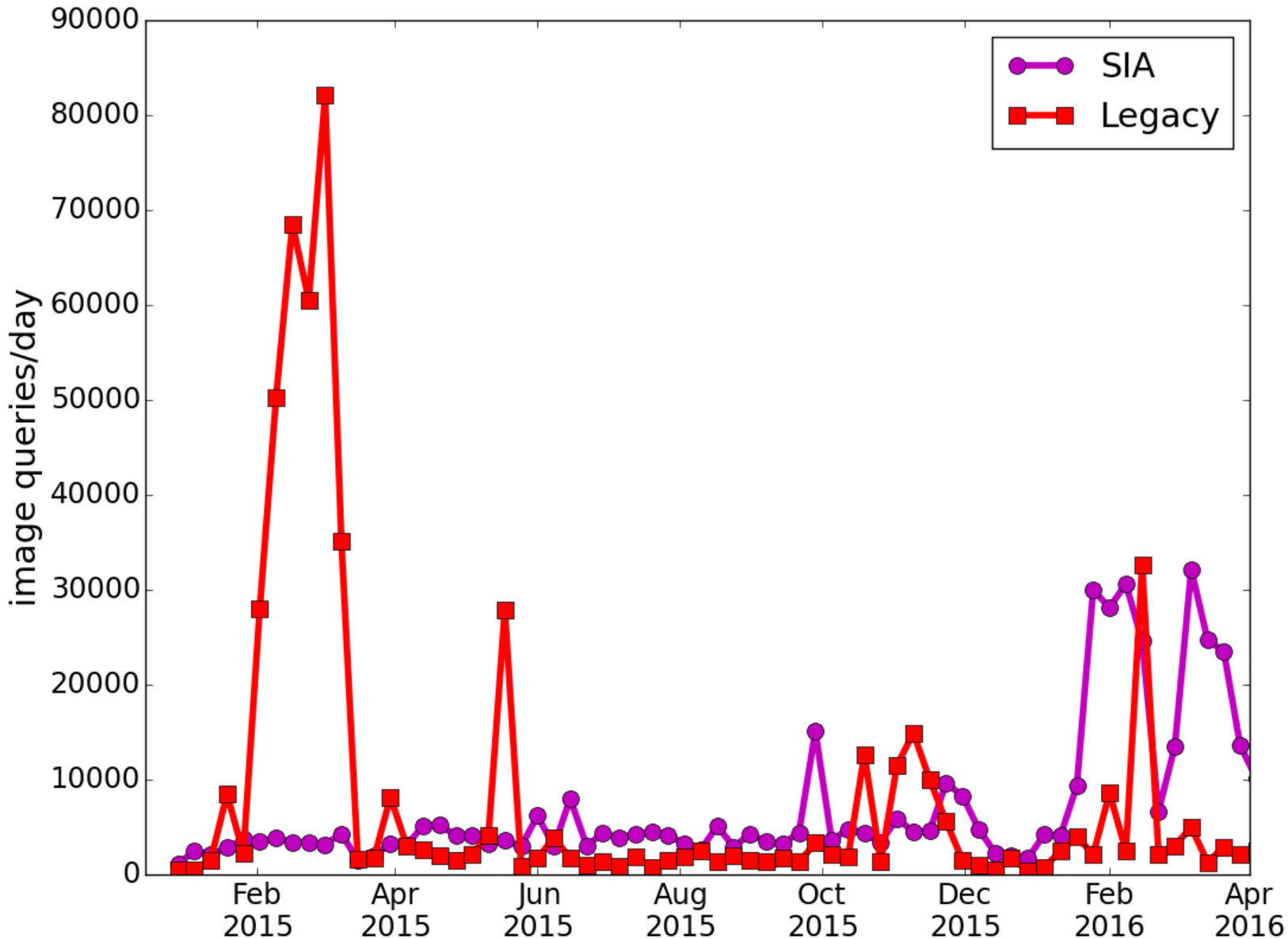
```
select mjd, w1mpro_ep w1sigmpro_ep
from mini_mep
where (cntr_mf = 2264175701351057759)
```

- Otherwise numbers vary from 300-10,000
- 20-170 different IP's
- 10-70 different subnets

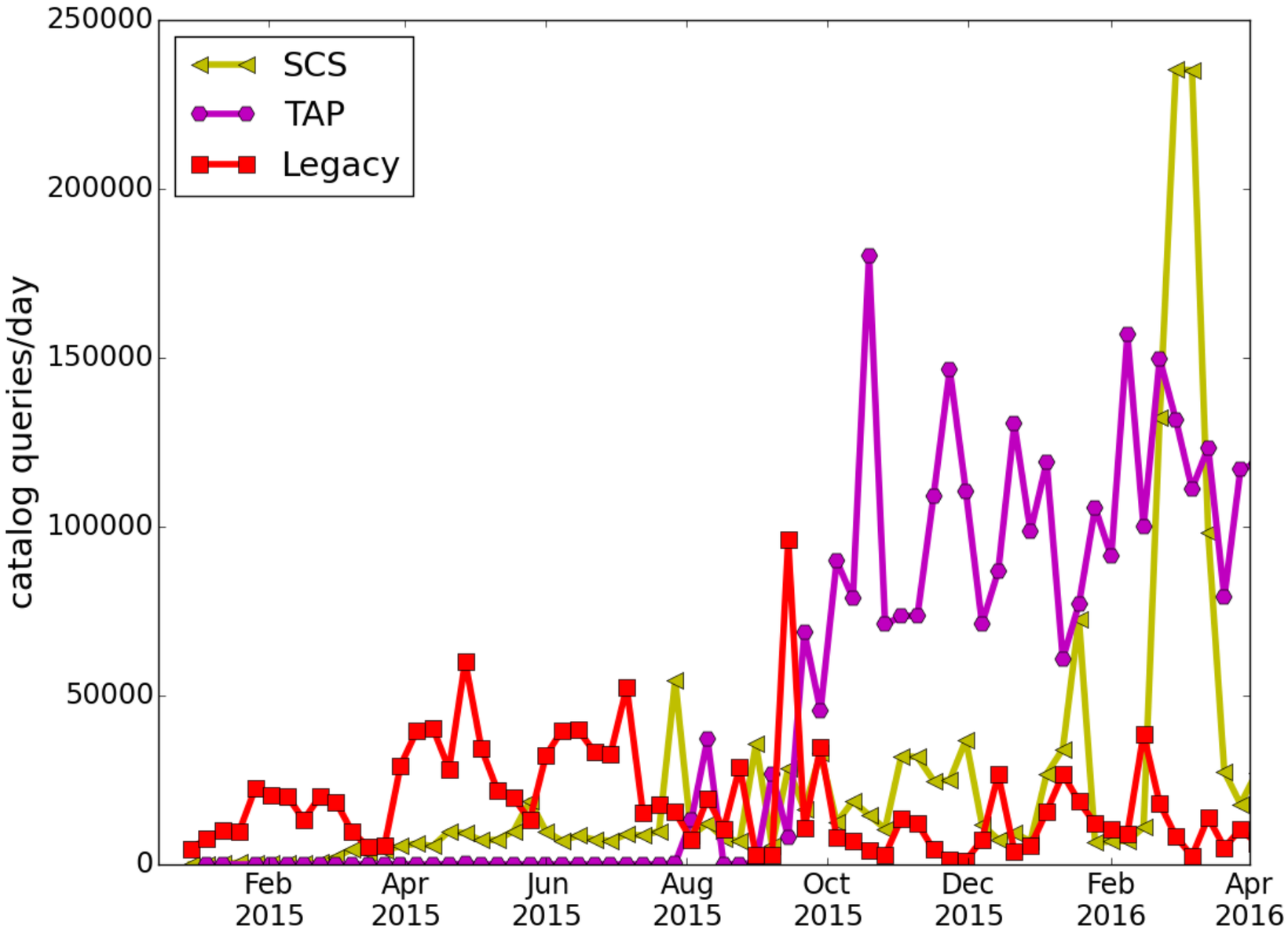
TAP/async Traffic in 2015

- Very little usage
  - 115 completed queries in a year
  - But some of those were HUGE
    (Germans tiling the sky to search WISE for Planet Nine)
- 21 unique IP's
- 12 unique subnets

VO vs Legacy Image API's



VO vs Legacy Catalog API's



Enthusiastic Users

- With all of this expressive power available, we do run into problems with users who try to write some expensive queries.
- This is not a new problem. We have had enthusiastic users for a long, long time.

Service Limits

- We are still experimenting with how to implement limits.
- In general, we do not try to curb 'abusive' behavior unless it affects other users.
    - One person's abuse is another persons science.
- All of the limits are there because someone caused trouble by going beyond them.
- We still have to manually kill jobs sometimes.

Synchronous Service Time Limits

- SCS jobs with a search radius < 0.01 degrees are run in-process. These queries finish fast enough such that we do not need to put in limits.
- Other synchronous jobs (SIA, SCS, TAP/async) are put into a 5 minute Slurm queue.
- That is about as long as a TCP/IP connection can remain open over the internet without us pinging the user.

Synchronous Service Connection Limits

- A single IP can only have 1000 TCP/IP connections at a time.
- TCP/IP connections hang around for a minute, so if someone is bombarding us with millions of fast cone searches, this will effectively rate limit them to 1000 requests/minute.

No Limits on Query Parameters

- We do not impose any limits on query parameters (e.g. cone search radius)
- There are some catalogs where 180 degree cone searches are reasonable (e.g. IRAS).
- Whether a query finishes quickly depends on some parameters in a non-linear way (e.g. HDF5 is fast, VOTable is slow)

Asynchronous Service Limits

- Three day limit to complete the query.
- Then the user has three more days to fetch the result before it is deleted from our system.
- Results are converted in memory to the format requested by the user. This puts a limit of about 100 GB on the final size of the result.

Registry

- Because we have hundreds of tables, we decided to run our own publishing registry.
- We used an existing implementation (perl script and xml files)
- It has been fairly painful, and we are still working out some details.

Future Plans

- Submit our variant of ADQL with restricted geometry for standardization.
- Submit an update to the UWS spec for running jobs immediately.

Future Plans

- Create a master CAOM table from all of our existing image and spectra metadata tables.
  - This is a first step in implementing ObsCore.
  - It will also make it easier to implement SIA v2 and Simple Spectral Access
- Consolidate SIA backends and put in more imagesets

http://irsa.ipac.caltech.edu