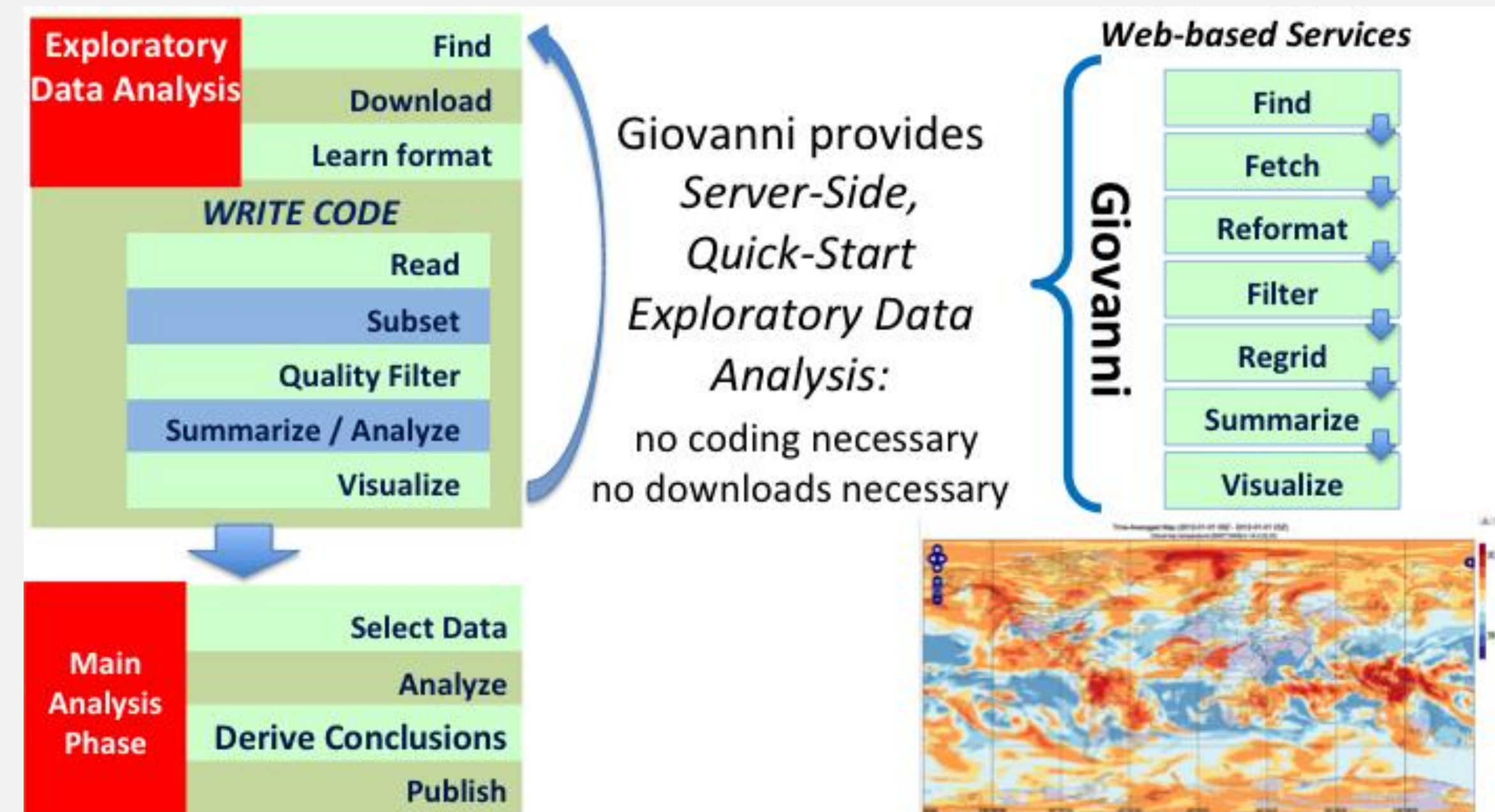


Analyzing a 35-Year Hourly Data Record: Why So Difficult?

If Big Data is hard for data center systems, what about the average science user?

Big Data at the GES DISC

- The Goddard Earth Sciences Data and Information Services Center offers Earth science data to the science community
- GES DISC provides the Giovanni tool for exploratory data analysis



<http://giovanni.gsfc.nasa.gov/giovanni/>

- How much data can we process for a Giovanni visualization?
- We will then incorporate limits into the Giovanni User Interface to prevent workflow failures

North American Land Data Assimilation System

- Land Hydrology model output
 - Soil moisture and temperature, runoff, evapotranspiration,...
- Available in netCDF
 - community standard
 - random-access format
- Archived as 1 file per timestep
- ~5000 users in 2013

Spatial Resolution	0.125 x 0.125 deg
Grid Dimensions	464 longitude x 224 latitude
Grid Cells per timestep	103,936
Files / Day	24
Days / Yr	365
Years	>35
Total Timesteps	> 310,000
Total Values per Variable	~ 32,000,000,000 (32 billion)

A Simple, but Popular, Task: Get the time series for a point

NCO (netCDF Operators): Fast tool for processing netCDF data files: <http://nco.sourceforge.net/>

Try #1: Wildcard on command line

```
go> nccat -d lat,40.02,40.02 -d lon,-105.29,-105.29 NLDAS_NOAH0125_H_002_soilm0_100cm/*/*.nc out.nc
```

-bash: /tools/share/COTS/nco/bin/nccat: Argument list too long

311,135 files
1 per timestep

Try #2: Read file list from standard input (NCO feature for working with lots of files)

```
go> cat in.nccat | nccat -d lat,40.02,40.02 -d lon,-105.29,-105.29 -o out.nc
```

nccat: ERROR Total length of fl_lst_in from stdin exceeds 1000000 characters. Possible misuse of feature. If your input file list is really this long, post request to developer's forum (<http://sf.net/projects/nco/forums/forum/9831>) to expand FL_LST_MAX_CHARS.

*Later expanded after post to developer's forum.

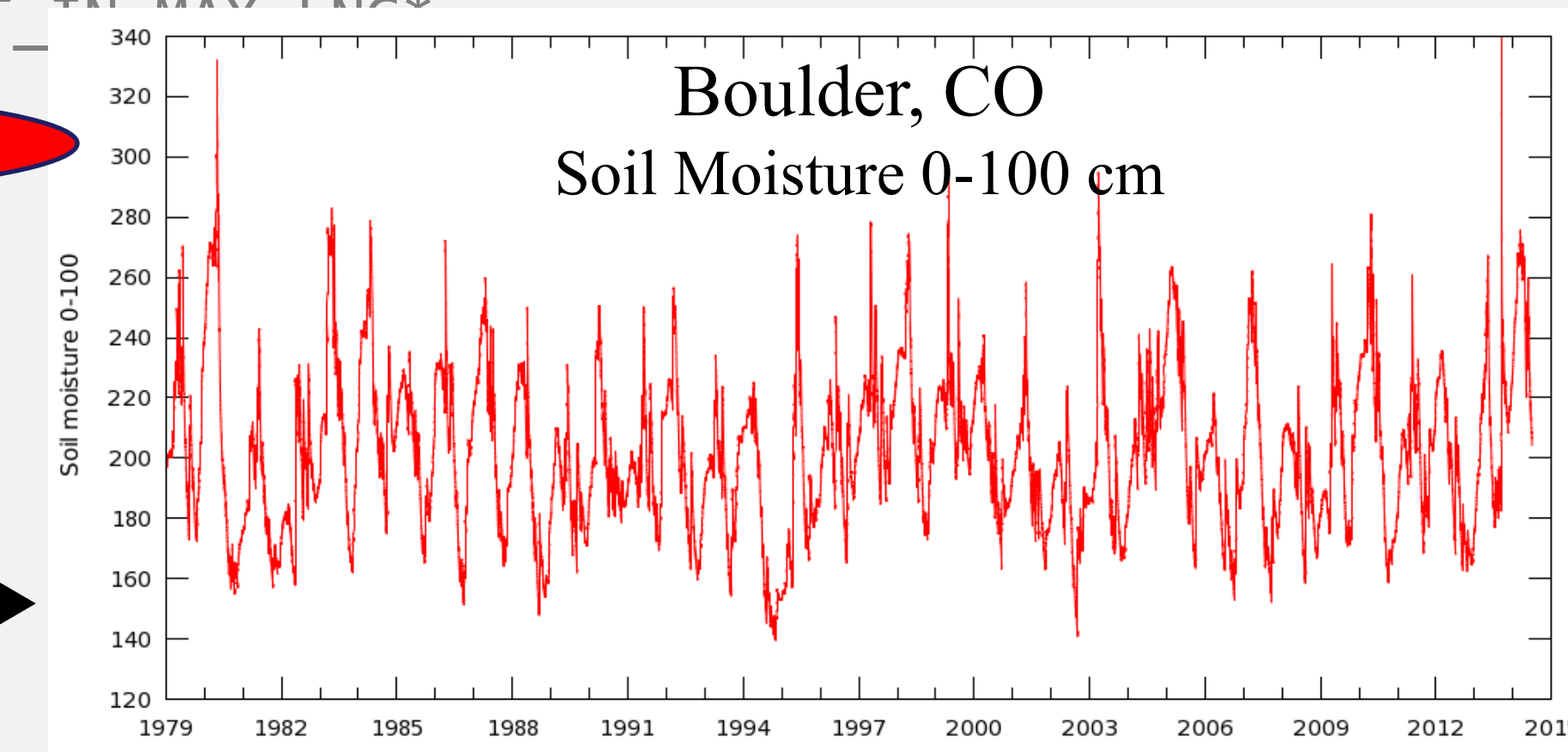
311,135 files ⇒ 21,796,250 chars

Try #3: Create local, short symlinks, then read from stdin

```
ln/1979/n1979010201.nc -> /var/scratch/clynnes/agu2014/data/1979/scrubbed.NLDAS_NOAH0125_H_002_soilm0_100cm.19790102T0100.nc
```

311,135 files ⇒ 7,165,105 chars (whew!)

SUCCESS!



Computing an Area Average Time Series (over Colorado)

Pre-concatenation of data can be helpful*

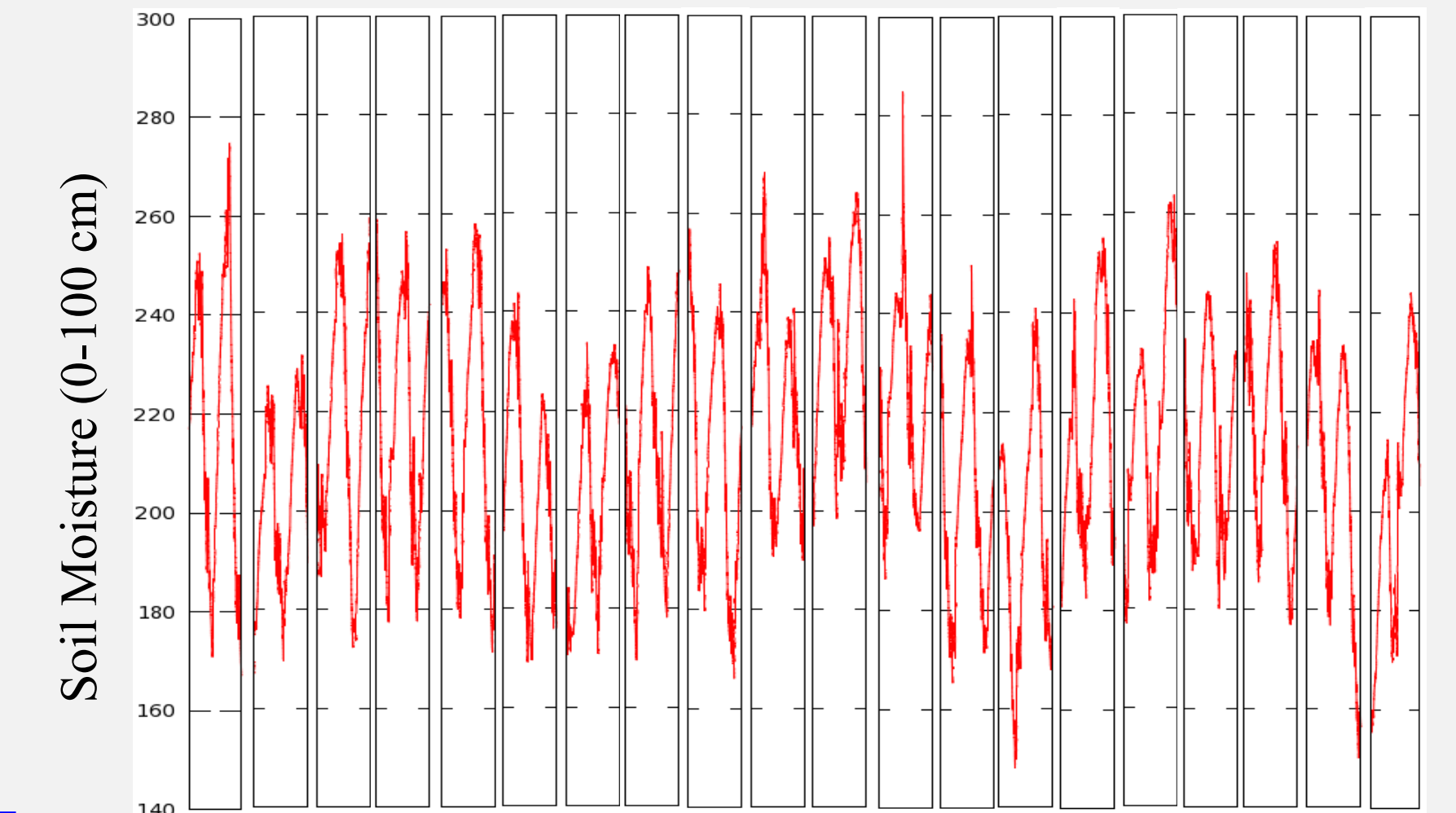
- Use `nccat` to concatenate hourly into daily or monthly files

Time Length of File	Avg File Size	Elapsed Time (min)	
		Processing	Aggregation
1 hour	0.4 MB	805	0
1 day	9.1 MB	61	75
1 month	285 MB	42	74

- But it makes some analysis (e.g. diurnal) unwieldy
- *Thanks to R. Strub and W. Teng for the idea

Also, Divide and Conquer

- Split input file list into sections, process, stitch together to reduce memory and temp. disk



Parallel Processing Made Easy Slightly Less Difficult

- Split input file list into sections, process *in parallel*, then stitch together
- **GNU make**: parallel job execution and control engine (-j option) for the common folk

```
# Makefile example: extract a time series* over Boulder, CO from hourly files
SPLITFILES = $(shell split -d -l 15556 in.symlink in. && ls -c1 in.??)
OUTSPLIT = $(SPLITFILES:in.=%=chunk.%.nc)
BBOX_ARGS = -d lat,40.02,40.02 -d lon,-105.29,-105.29

all: split boulder.nc

split:
    @echo $(SPLITFILES)

chunk.%.nc: in.%
    sort $^ | nccat -O -o $@ $(BBOX_ARGS)

boulder.nc: $(OUTSPLIT)
    nccat -O -o $@ $^
```

1. Split list of input files (in.symlink) into chunks: in.00, in.01...

2. Extract and concatenate each chunk

3. When all sections are done, stitch them together

*The actual Makefile for Area Average Time Series is more complicated than this point time series example.

- To Run 2 jobs in parallel: `make split && make -j 2 -l 9. all`

"-j 2" for 2 jobs at a time

"-l 9" defers new jobs until load drops below 9

Colorado Area Average Time Series (Monthly Files)

Parallel Jobs	Elapsed Time (min.)	
	netCDF-3	netCDF-4
1	42	26
2	42	21
3	36	20

Parallelization is not quite a silver bullet

Recommendations

- **To Data Product Designers / Producers**
 - Include the Time Dimension, even for single-time-step files.
 - It makes reorganizing time segments easy (with NCO's `nccat`)
- **To Data Centers**
 - Provide more remote analysis capability for scientists.
 - GrADS Data Server: <http://www.iges.org/grads/gds/>
 - Live Access Server: <http://ferret.pmml.noaa.gov/LAS>
 - Giovanni: Federated Giovanni in progress (*Talk IN52A-05, MW-2020 Friday, 11:20*)
 - <insert yours here>
 - Provide users with tips and tricks for Big Data processing.
 - No reason users need to re-learn our lessons the hard way.
 - Provide or point to tools and recipes for splitting and stitching data.

