

Provenance

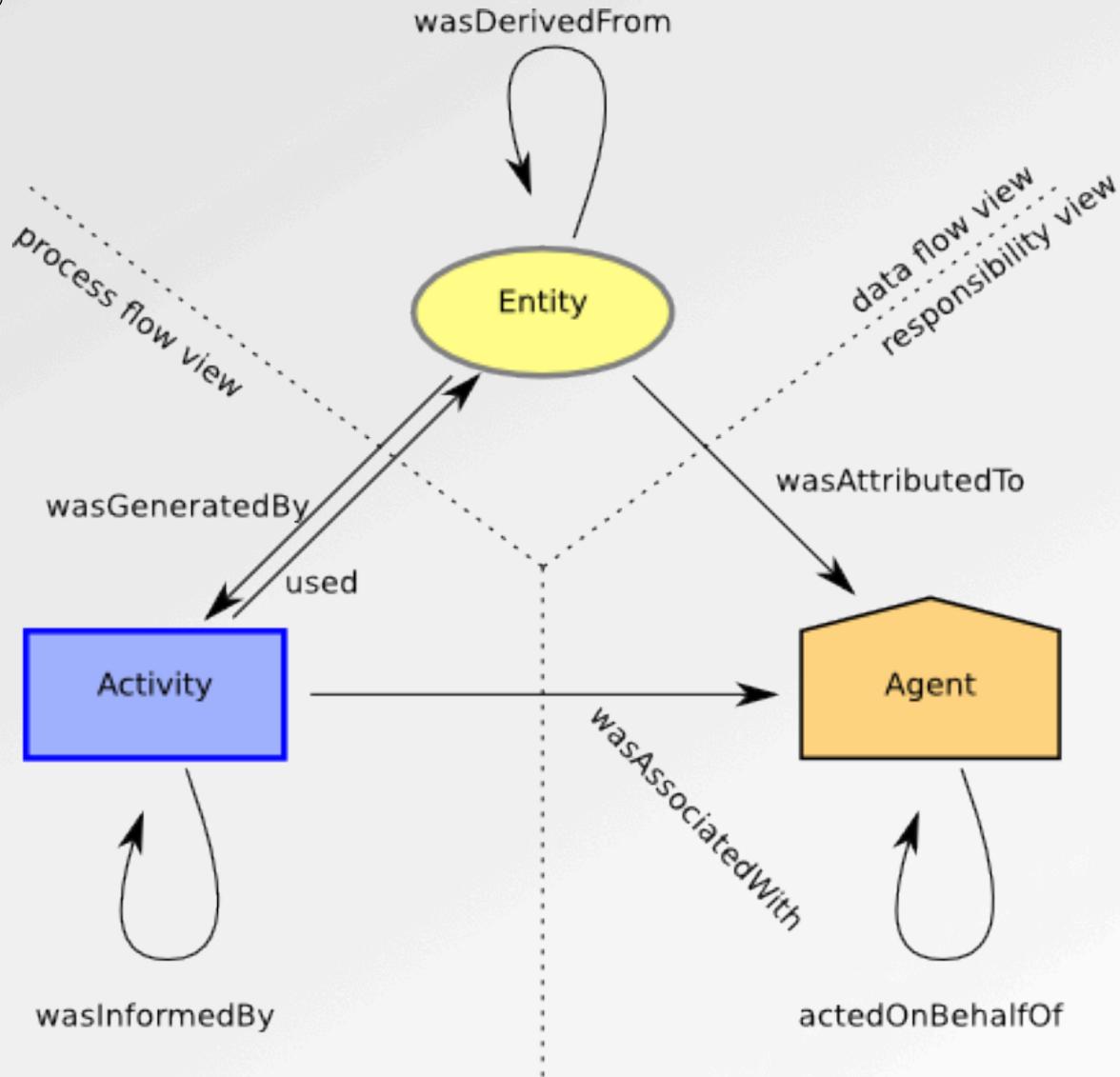


Provenance Views

- Data
 - data quality and history inspection
 - mostly recorded in FITS HISTORY headers
 - exposed in a text file
- Experiment
 - **definition** provenance
 - structure / flowchart view of the **whole**
 - **deployment** provenance
 - execution environment
 - **execution** provenance
 - profiling and exec. logs
 - functions calls and variables
 - input / intermediate / output values
- **Evolution**
 - → → versioning

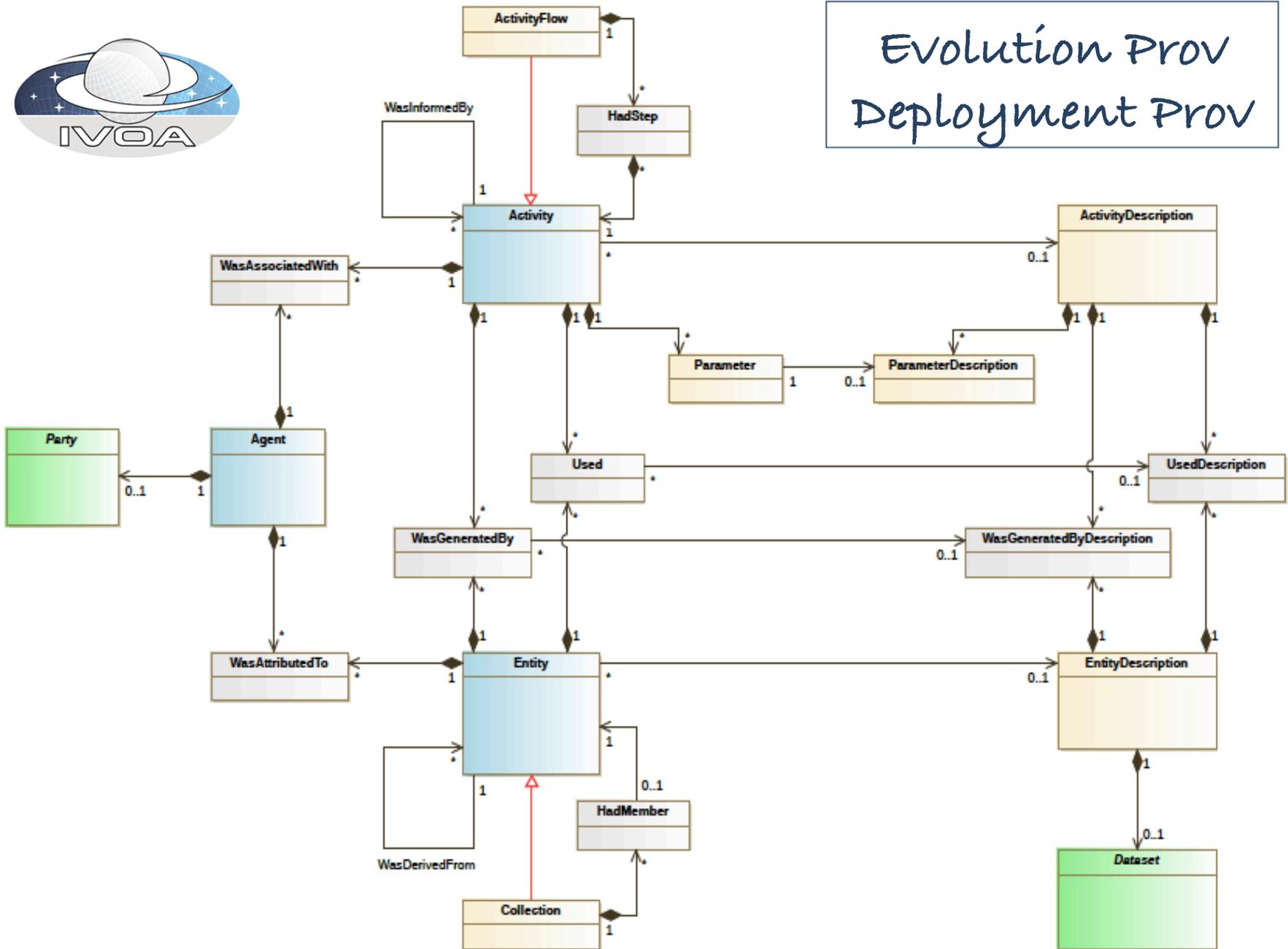


Provenance Views





Evolution Prov Deployment Prov



Application/User Levels in CTA

- Simulations
 - execution/evolution provenance
- Pipelines on raw data
 - execution provenance (exec. logs and profiling)
- Public DL3 → DL5 Archive
 - data provenance
- User Desktop (non-controlled environment)
 - definition provenance
 - structure / flowchart view of the experiment
 - deployment provenance
 - environment
 - evolution provenance
 - versioning tools



Provenance Capture in the Local Desktop

INSTITUTO DE ASTROFÍSICA DE ANDALUCÍA, IAA-CSIC

The collage illustrates the integration of various astronomical data processing and analysis tools. Red arrows trace the flow of data and provenance information across different applications:

- IRAF (Image Reduction and Analysis Facility)**: The central tool for image processing, shown with its logo and interface elements.
- VizieR**: A web-based interface for searching and downloading data from various astronomical catalogs.
- NASA/IPAC Extragalactic Database (NED)**: A database for extragalactic objects, showing search results and object details.
- Pfam**: A protein family database used for sequence alignment and analysis.
- IDL**: A programming language used for scientific data analysis and visualization.
- RepeatMasker Web Server**: A web-based tool for identifying and masking repetitive DNA sequences.
- Fortran**: A high-performance programming language used for scientific computing.
- Python**: A versatile programming language used for data analysis and automation.
- Other Tools**: Includes VizieR search criteria, a table of astronomical data, and various visualization plots.

```
# CIG Vhel e_Vhel r_Vhel Dist MType e_MType OptAssym r_MType Bmag e_Bmag
1 7299.0 3.0 1 96.9 5.0 1.5 1 14.167 0.271 0.173 0.571 0.040 13.383
2 6993.0 6.0 2 94.7 6.0 1.5 0 1 15.722 0.324 0.255 0.278 0.031 15.157
3 4.0 1.5 0 1 16.057 0.507 0.246 0.354 15.457
4 2310.0 1.0 3 31.9 3.0 1.5 0 1 12.918 0.424 0.252 0.863 0.017 11.685
5 7865.0 10.0 3 105.9 0.0 1.5 0 1 15.602 0.364 0.225 0.131 0.118 15.128
72 5164.0 9.0 2 68.5 5.0 1.5 1 1 14.445 0.325 0.315 0.367 0.028 12.735
```

Search Criteria

Find catalog

Clear

Preferences

max: 50

HTML Table

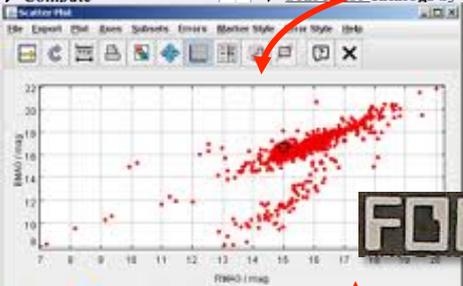
All columns

Compute

Search for catalogs by column description (UCD)

cross 11708 tables

Sesam) or Position:



FORTRAN

IRAF
Image Reduction and Analysis Facility

RepeatMasker Web Server

RepeatMasker screens DNA sequences in fasta format against a library of repetitive elements and returns a mask ready for database searches as well as a table annotating the masked regions.

Basic Options

Large sequences will be queued, and may take a while to process.

Enter the file to process:

Or paste the sequence(s) in FASTA format:

Select output format: html tar file table

Select email address:

Submit Sequence

Pfam: Search DNA vs. Pfam

Protein families database of alignments and HMMs

Pfam: Search DNA vs. Pfam

This form allows you to compare your DNA sequence against the whole of Pfam using the [Uzarc](#) software package

Cut and Paste your DNA sequence here, fasta format

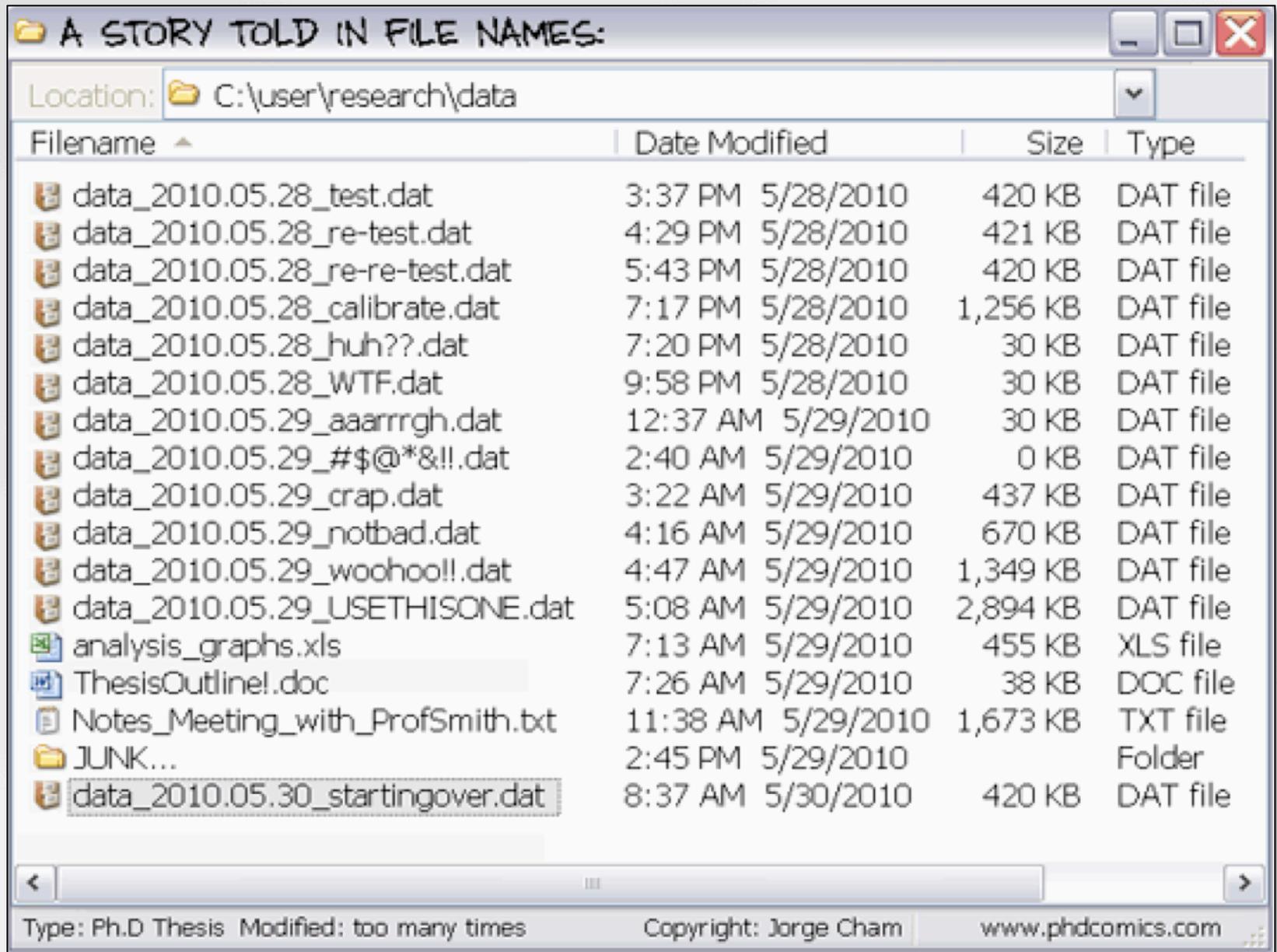
Searcher now use the Sanger Blast queues. Presently the email option is available.

It takes around 2 minutes for a 1,000 bp sequence, and around 2 hours for a 500k sequence, depending on how many matches you get in them and the load on the sanger centre system.

Output options for alignments

Output for gene predictions

Provenance Capture in the Local Desktop



The screenshot shows a Windows Explorer window titled "A STORY TOLD IN FILE NAMES:" with the address bar set to "C:\user\research\data". The window displays a list of files and folders with columns for filename, date modified, size, and type. The files are named with dates and humorous or descriptive text, such as "data_2010.05.28_test.dat" and "data_2010.05.29_#*\$@*&!!.dat". The file "data_2010.05.30_startingover.dat" is highlighted.

Filename	Date Modified	Size	Type
data_2010.05.28_test.dat	3:37 PM 5/28/2010	420 KB	DAT file
data_2010.05.28_re-test.dat	4:29 PM 5/28/2010	421 KB	DAT file
data_2010.05.28_re-re-test.dat	5:43 PM 5/28/2010	420 KB	DAT file
data_2010.05.28_calibrate.dat	7:17 PM 5/28/2010	1,256 KB	DAT file
data_2010.05.28_huh??.dat	7:20 PM 5/28/2010	30 KB	DAT file
data_2010.05.28_WTF.dat	9:58 PM 5/28/2010	30 KB	DAT file
data_2010.05.29_aaarrgh.dat	12:37 AM 5/29/2010	30 KB	DAT file
data_2010.05.29_#*\$@*&!!.dat	2:40 AM 5/29/2010	0 KB	DAT file
data_2010.05.29_crap.dat	3:22 AM 5/29/2010	437 KB	DAT file
data_2010.05.29_notbad.dat	4:16 AM 5/29/2010	670 KB	DAT file
data_2010.05.29_woohoo!!.dat	4:47 AM 5/29/2010	1,349 KB	DAT file
data_2010.05.29_USETHISONE.dat	5:08 AM 5/29/2010	2,894 KB	DAT file
analysis_graphs.xls	7:13 AM 5/29/2010	455 KB	XLS file
ThesisOutline!.doc	7:26 AM 5/29/2010	38 KB	DOC file
Notes_Meeting_with_ProfSmith.txt	11:38 AM 5/29/2010	1,673 KB	TXT file
JUNK...	2:45 PM 5/29/2010		Folder
data_2010.05.30_startingover.dat	8:37 AM 5/30/2010	420 KB	DAT file

Type: Ph.D Thesis Modified: too many times Copyright: Jorge Cham www.phdcomics.com



Provenance in script-based methodology

Scripts orchestrate analysis and **connect** data and tools

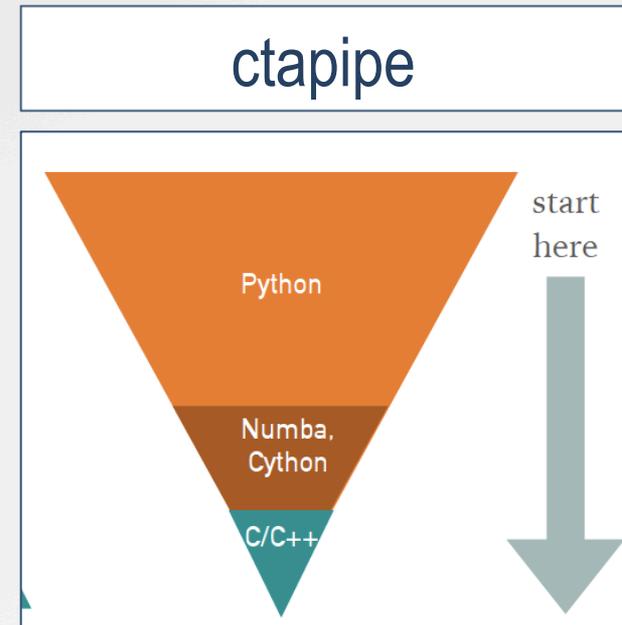
Python scripts as a glue

Challenges

- encode control/loops
- level of granularity
- non-controlled environment

Lesson learned

- prov. capture /inspection /analysis **MUST** be:
 - non-intrusive
 - user-friendly



noWorkflow Tool

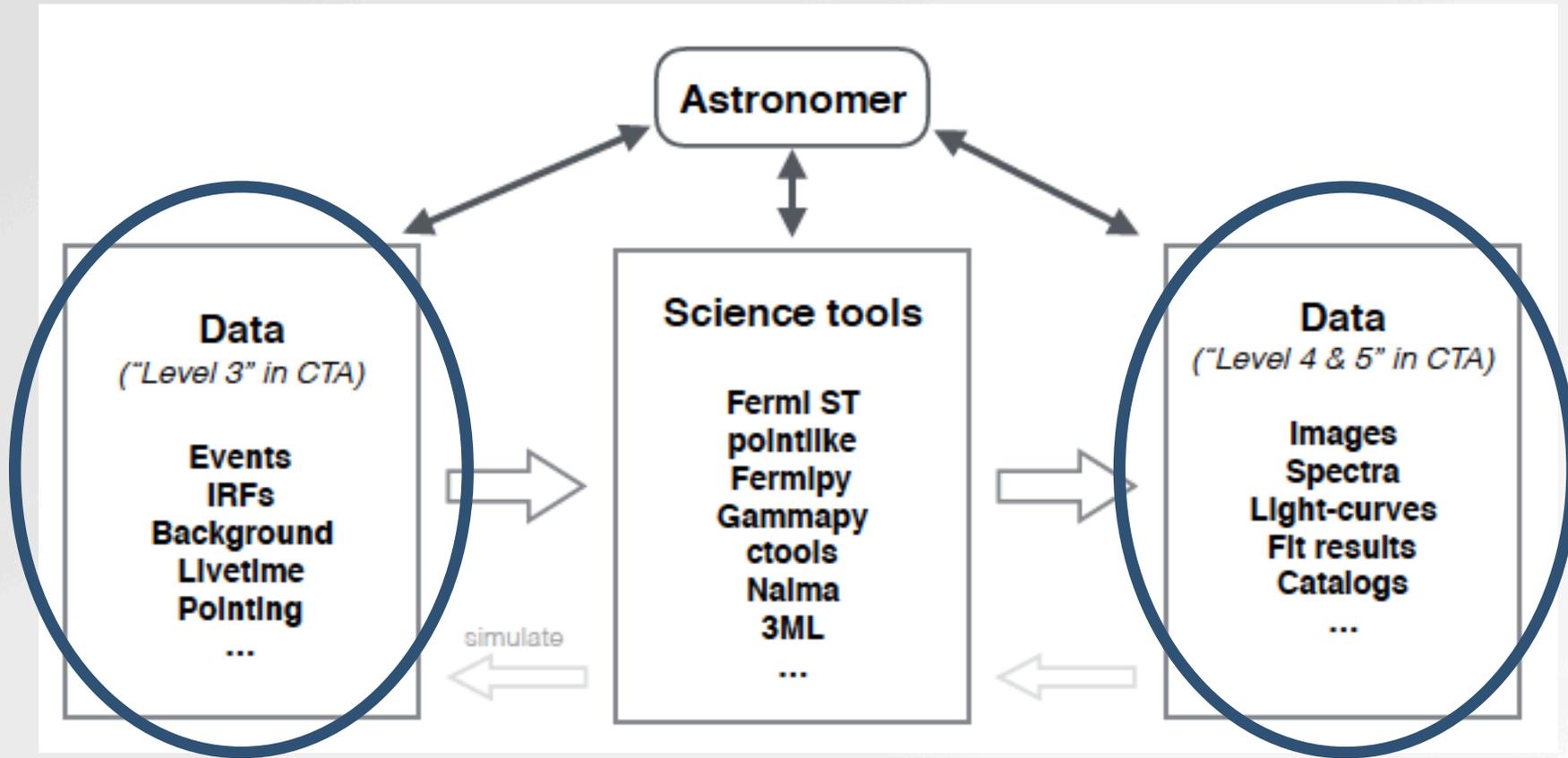
Captures process provenance for a **data analysis** working **methodology** based on **python scripts** and trial/error **exploration runs**.

- Provenance storage: SQLite DB + File System
- Provenance sharing of a local working session: `.noworkflow` folder
- Jupyter Notebooks support

Provenance capture of what happens inside a Python script

- Definition Provenance
 - Abstract Syntax Tree Analysis (code parsing / heuristics)
- Deployment Provenance
 - Python modules: `os`, `socket`, `modulefinder`, ...
- Execution Provenance
 - Profiling and reflection (reimplementation of I/O functions)

Provenance in CTA Public Archive



- Pack of data files in a working session
- Connection with Local Desktop Provenance?

Querying with Provenance

Input Dimensions

- Identifier (entity/activity/agent)
- Focus
 - Data progenitors
 - Processes involved
 - Agents responsibility
 - Versioning
- Representation
 - Graph
 - Prov. File/VOTable
 - Whole Pack with Data Products
- Time direction (back/forward)
- Granularity
- Parameter vs. Language based queries

Exposing Provenance

Response Dimensions

- Representation
 - Graph
 - Prov. File/VOTable
 - Whole Pack with Data Products
- Analysis and Inspection
 - Graph
 - Diff-based
 - Browseable and granularity (Links to prov. services in Datalink)
- Pack structure → Reproducibility
 - Prov. file as the descriptor of a pack of interlinked files
 - Data, documentation, scripts,...

Storing Provenance in Archive

- Provenance is about **relationships**
- Despite their name RDBs are not well suited for relationships
- Fixed schema of RDBs do not adapt well to changes
- Relationships are first priority in noSQL/**Graph** databases

