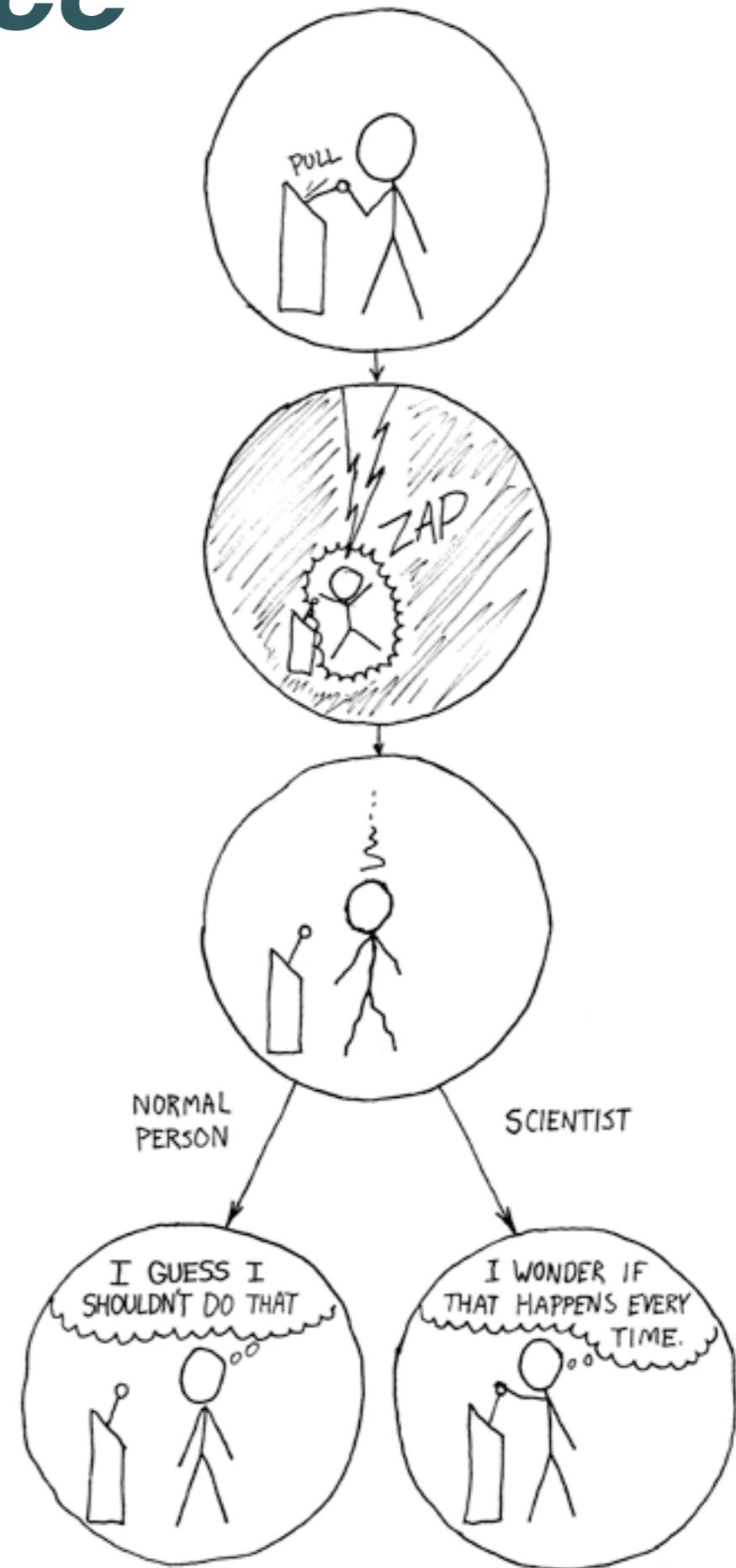
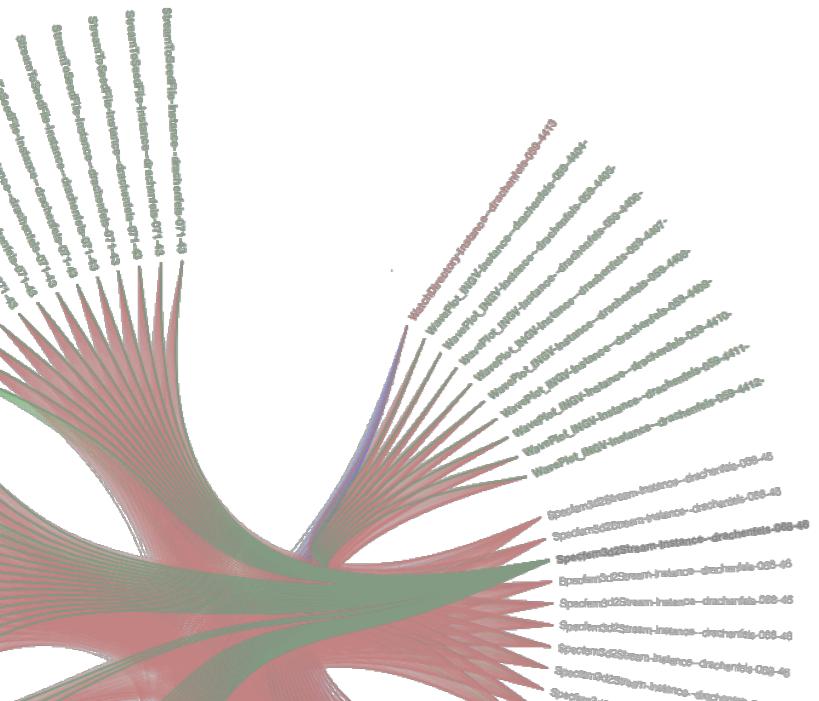


Reproducible Science

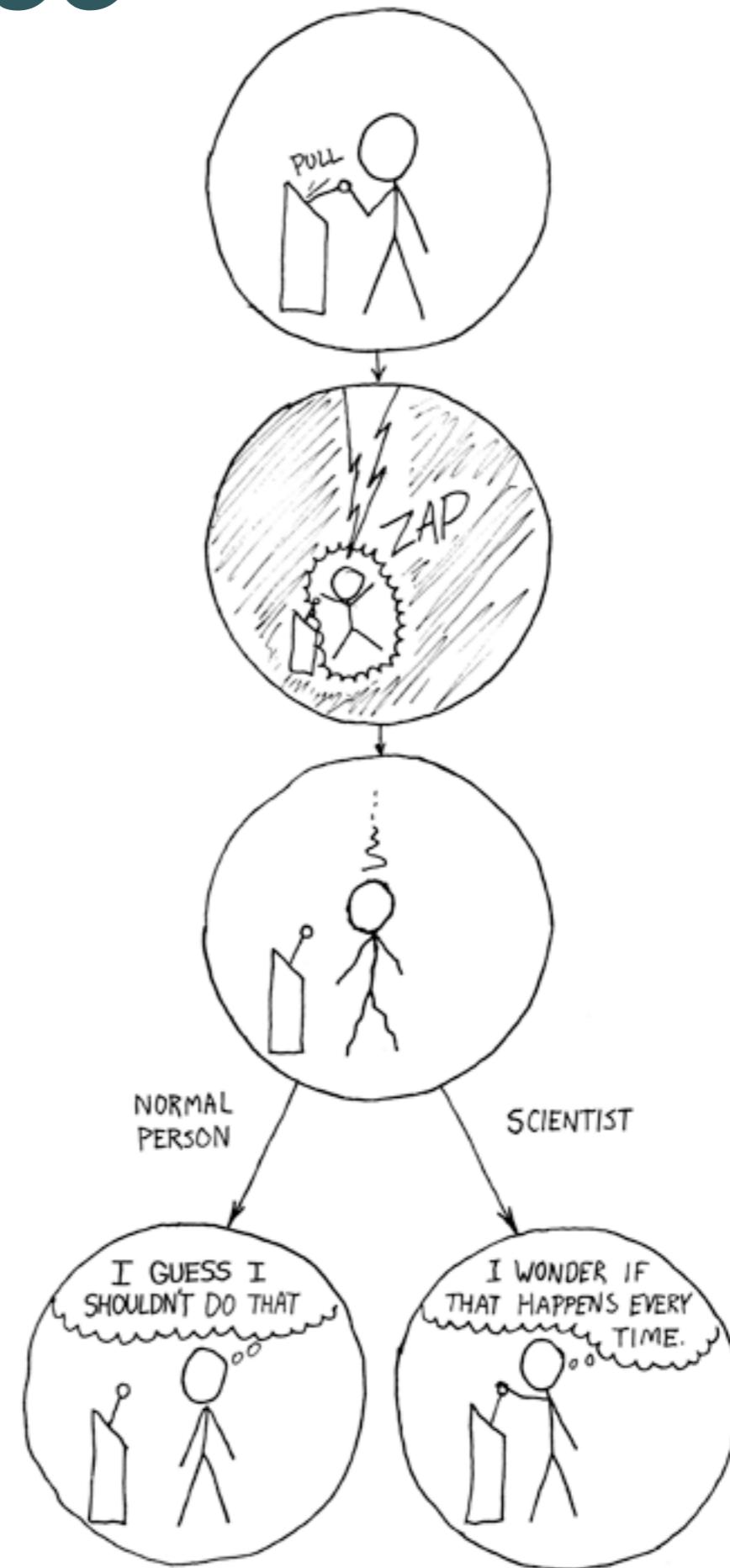
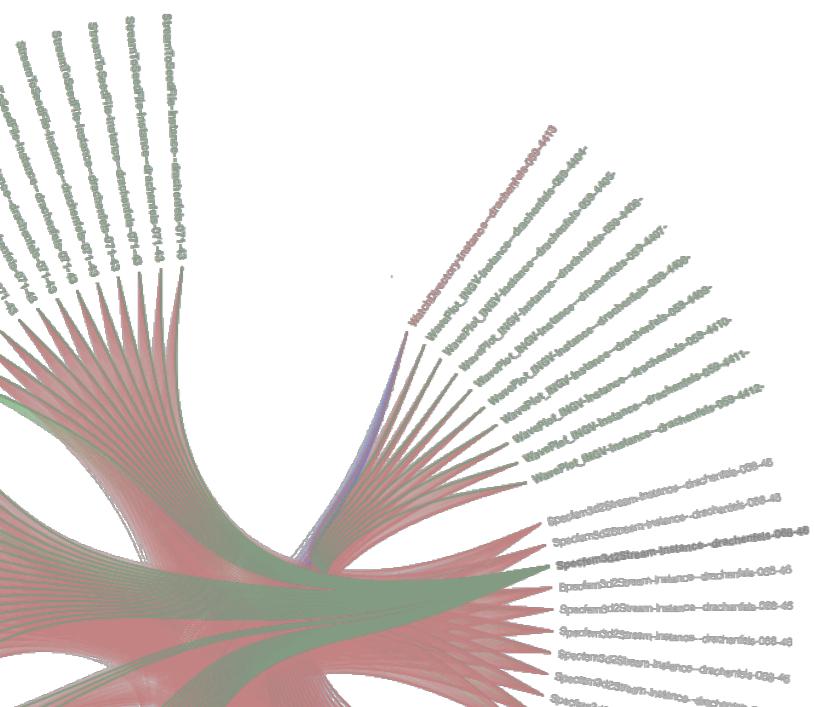


Reproducible Science

What does this suggest?

Scientists aren't normal..

Reproducibility is masochism..



Reproducible Science

What does this suggest?

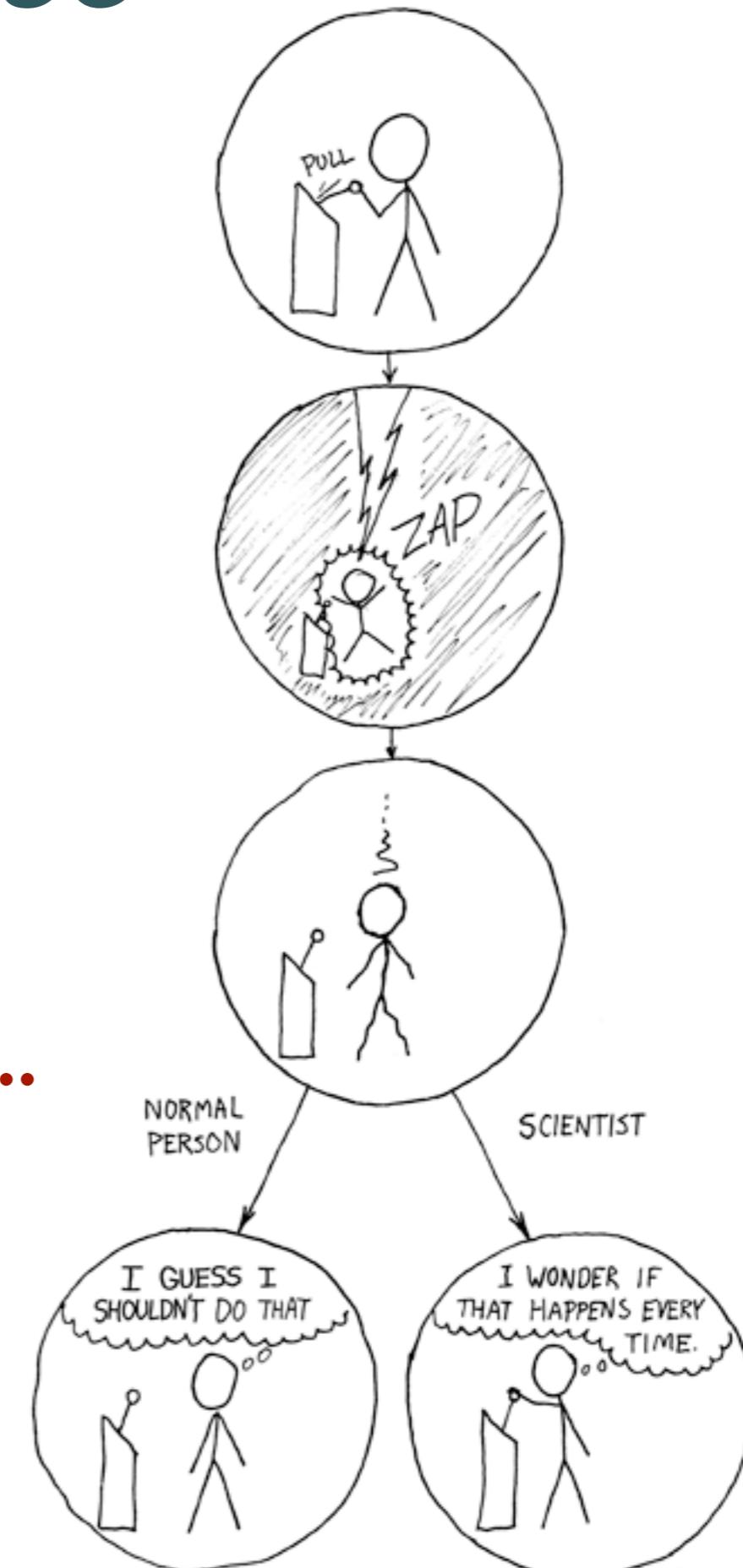
Scientists aren't normal..

Reproducibility is masochism..

**Reproducibility is fundamental
but difficult to achieve.**

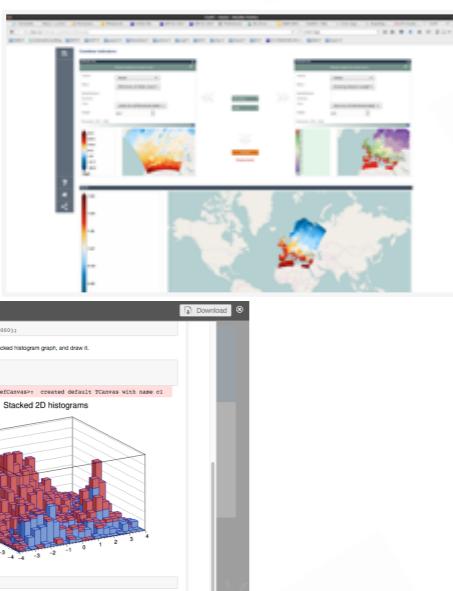
Not always convenient/possible..

We need more than just rerun

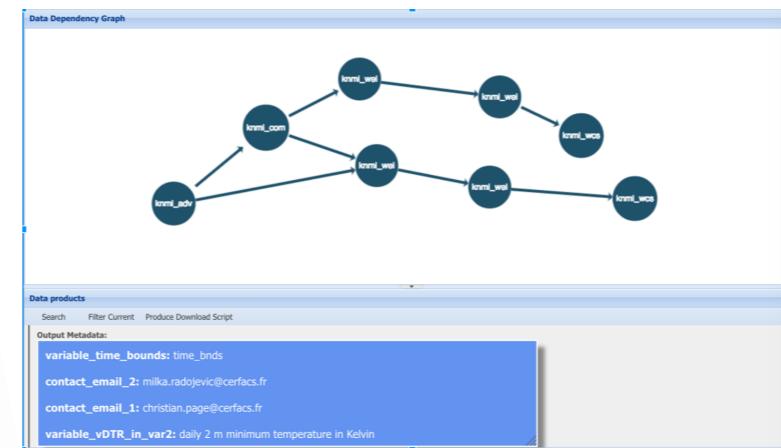


Reproducibility Cycle(s)

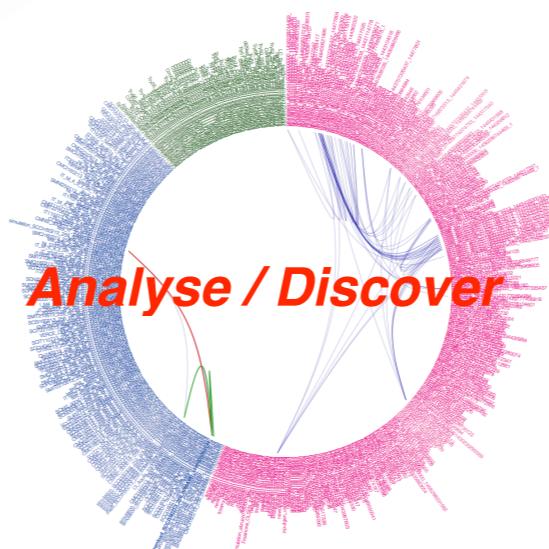
```
root@cerfacs-OptiPlex-5090:~# ./checkdata.sh
...
Total 72
root@cerfacs-OptiPlex-5090:~# ./checkdata.sh
```



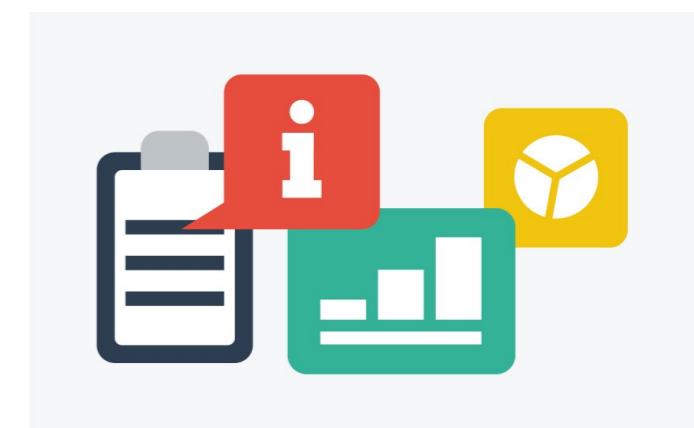
Procedures / Tools / Development



Validate



Repeat / Verify

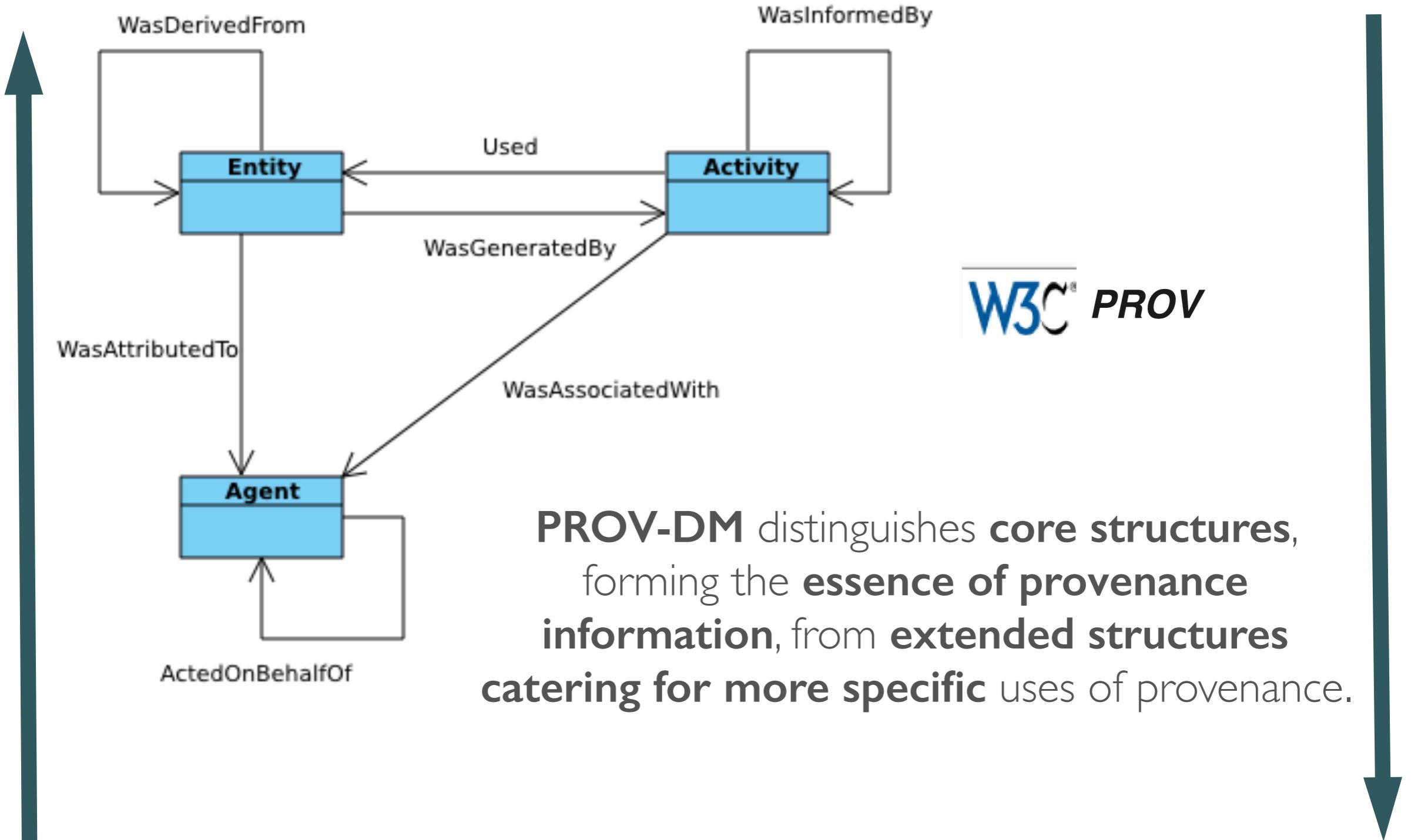


Report / Outreach / Preserve



Provenance Model

Data-Lineage —————→ *Knowledge*



W3C PROV

PROV-DM distinguishes **core structures**, forming the **essence of provenance information**, from **extended structures** catering for more specific uses of provenance.

Enactment ←————→ *Machinery & Data*

Provenance Challenges

“**Automated** system should provide support for a **consistent and effective acquisition** of provenance metadata.” [A. Misra] [I. Foster.]

Expert users are part of the process - configuration and contextualisation:

“How much domain metadata should be contained?”

Scale of the provenance records:

“What level of granularity is needed to describe provenance of complex objects?
“Manage the scale of the provenance records to be recorded and processed”

Multiple levels of understanding:

“Provenance at different levels of abstraction, extract high-level summaries of provenance from detailed records.”

Roles in Computational Research

research-developers (data-scientists):

Empirical implementation and evaluation of new methods and advanced tool

Direct observations at different scales and across execution environments (PaaS).

end-users (domain-scientists):

Use tools via virtual environments

Confidence and trust in the tool requires feedback and contextual information (SaaS). Share - Reuse

data-architects:

The improved understanding of the execution's details lead to better systems.

data-curators and administrators:

Long-term preservation of scientific results and exploitation of tools.

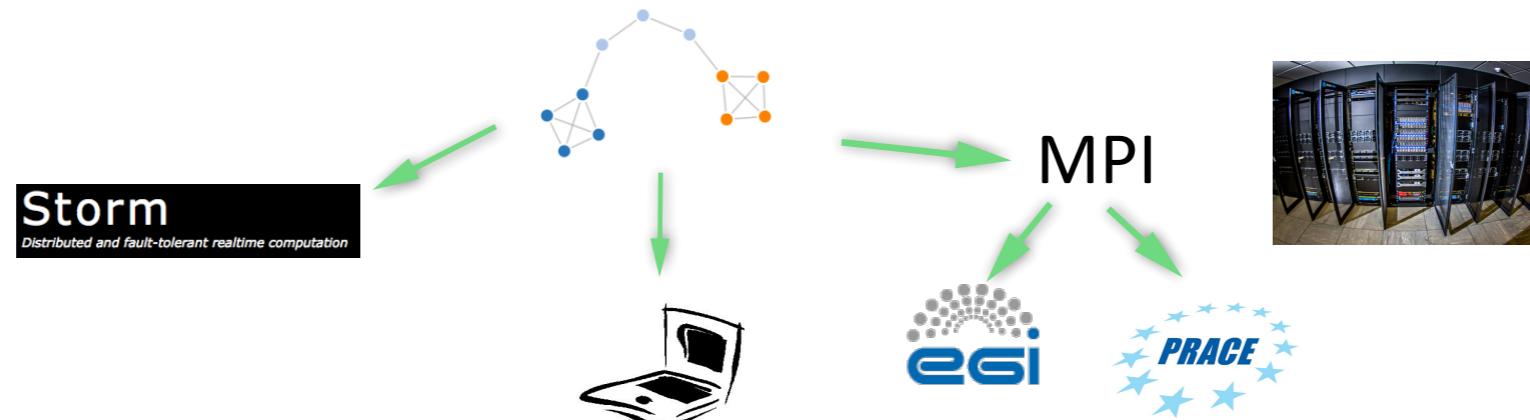
S-PROV

Provenance and Reproducibility in S-PROV:

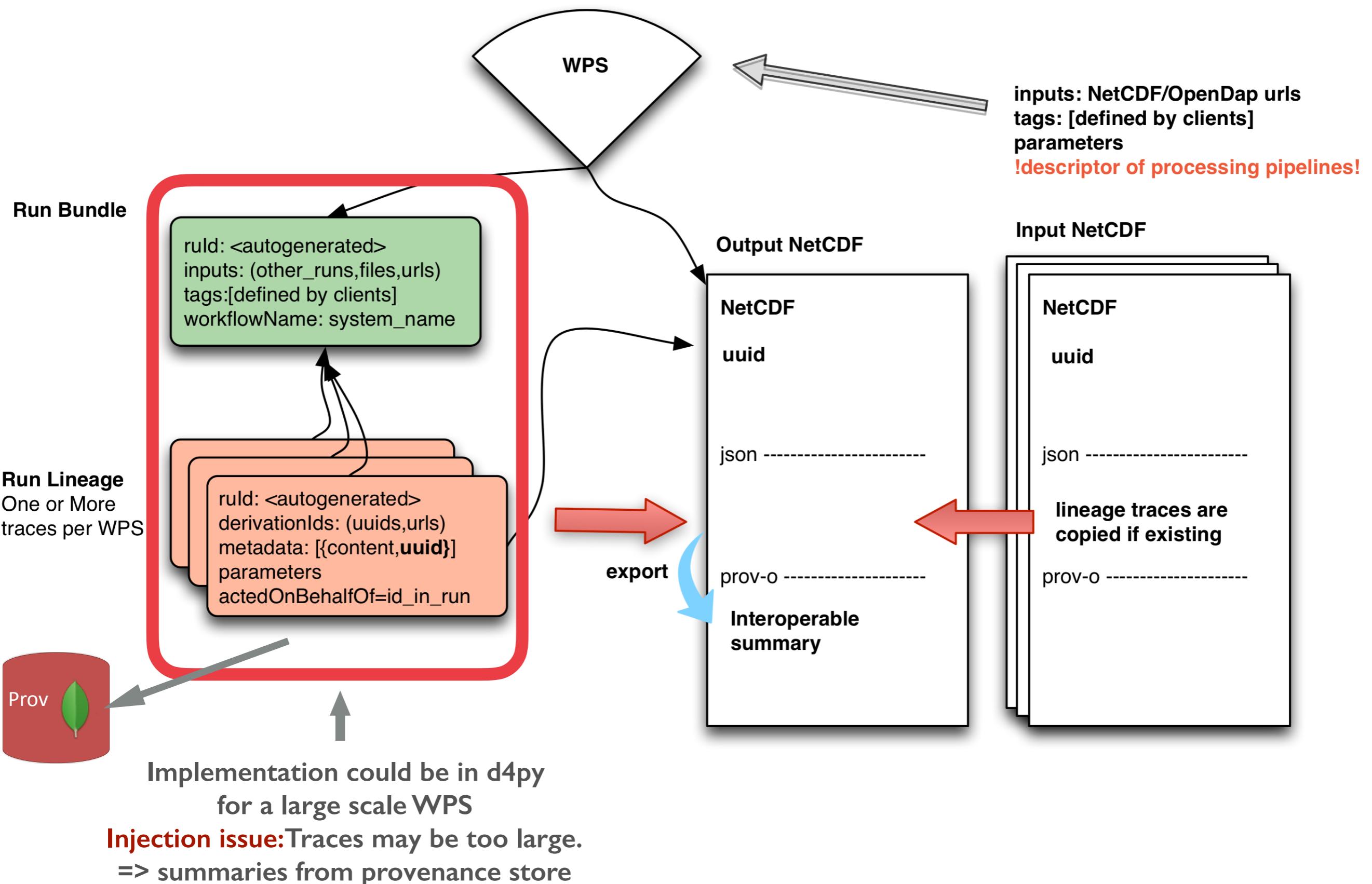
- Represents the **relationships** occurring between the players of a **data-intensive computation** in a **scientific domain**.
- Captures information at **multiple level of detail** within an **holistic data-model** (W3C PROV)
- Allows for **contextual switch** among **expertise** and **understandings**
- **Active Provenance Use Cases:** Monitor, validate, steer, compare, repeat, document.

Target System

- Systems where the data-intensive applications are specified with a **high-level language or API** (eg. ToolBox, dispel4py)
- Users define **abstract, machine-agnostic, workflows**.



Enabling S-PROV in WPS and NetCDF Injection



Enabling S-PROV in WPS and NetCDF Injection

Sensible issues:

Generation of *UUID*:

- The WPS framework should automatically produce and assign UUID
- UUID are used also into the lineage trace to match the output data with its provenance

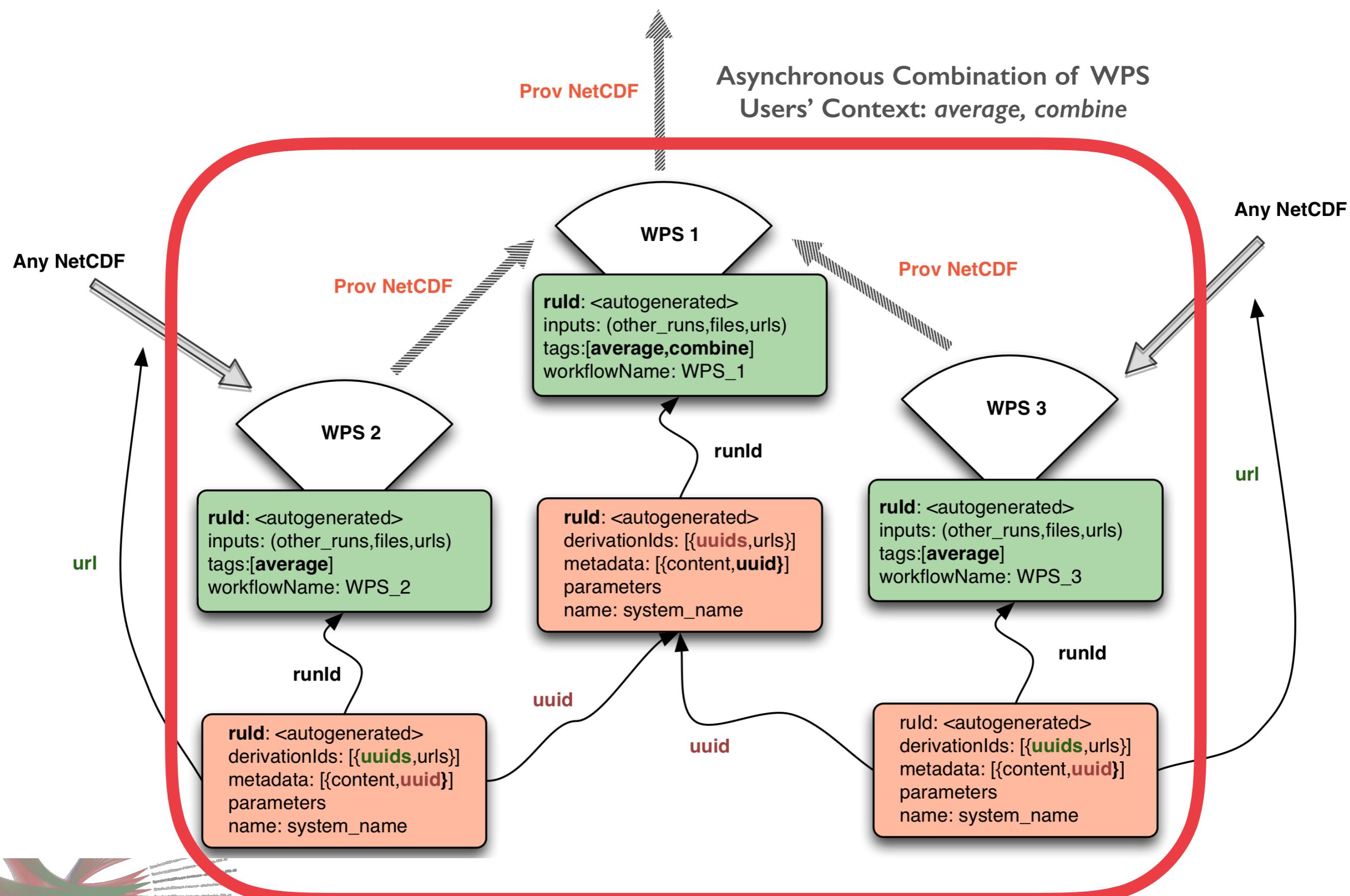
Derivation ID *UUID* (Chaining of WPS and associated outputs):

- IF input files have *prov* attributes we re-use the uuid ad *DatasetDerivationID*
- ELSE we may directly use the links (URLS) to the file.
- Links are used anyway in the Run Bundle to describe the WPS inputs

Tagging (Chaining of WPS and associated outputs)

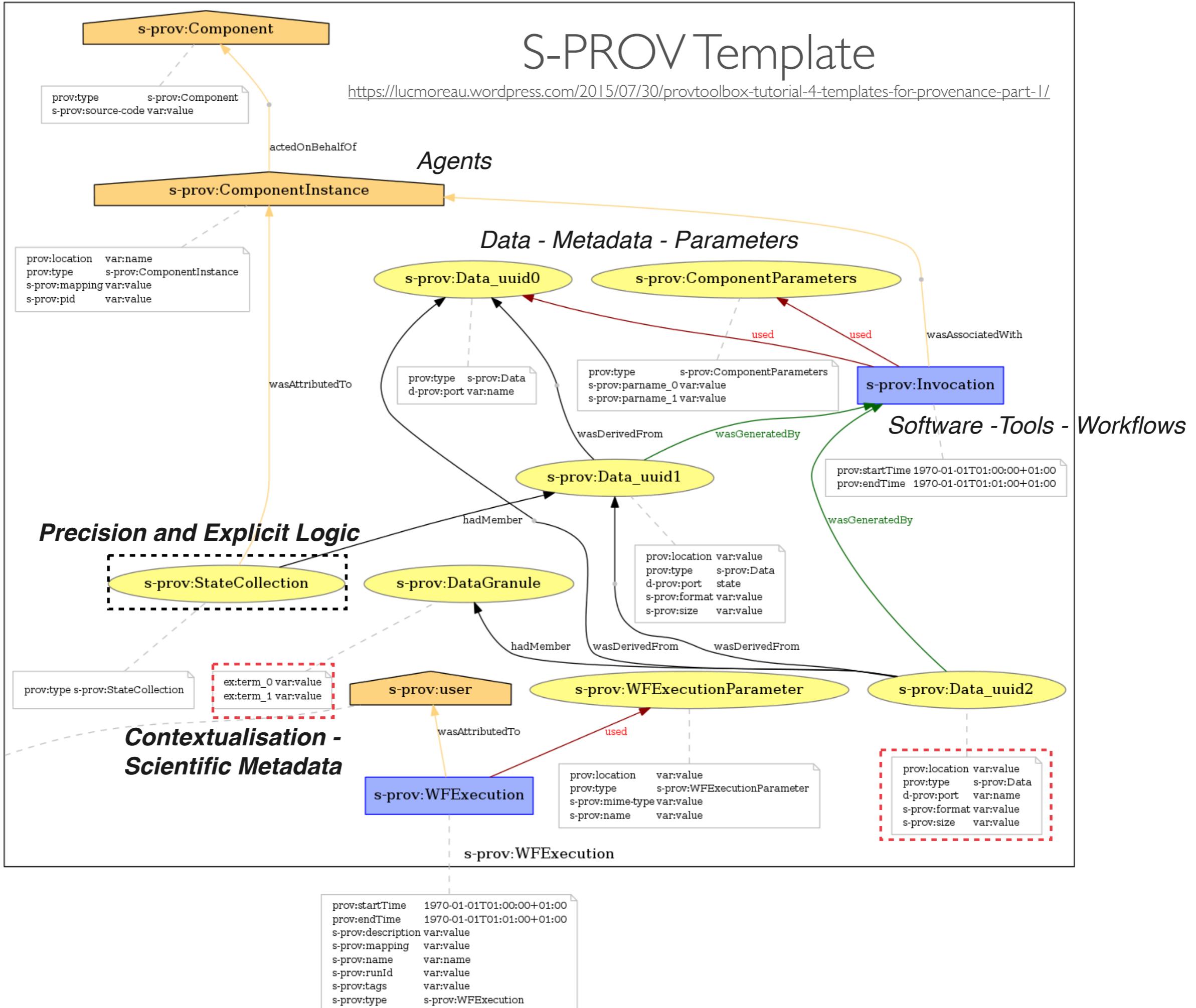
- The WPS should allow for a *tags* parameter
- This will allow to link Run Bundles together, describing workflows according to the users' contexts.

Example, average and combine



S-PROV Template

<https://lucmoreau.wordpress.com/2015/07/30/provtoolbox-tutorial-4-templates-for-provenance-part-1/>



Tools: S-ProvFlow - Reproducibility as a Service

The screenshot displays the S-ProvFlow interface with several panels:

- Monitoring:** System and user Messages panel (left) showing a list of log entries.
- Discovery:** Search on contextual metadata (within and across runs) panel (center-left) with a search interface for terms like "sampling_rate, station".
- Data Dependency Graph:** A central graph visualization titled "Data Resource" and "Stateful invocation". It shows nodes for "MergeImage", "MatchCom", "PyflexPE", and "streamPr", with arrows indicating data dependencies between them. A legend at the top right defines colors for trace-bw, trace-fw, stateful, cross-run, and file types.
- Data products:** Panel showing output metadata for a specific run, including `delta_syn: 0.05`, `id_raw: 043c00e4-0197-11e6-ab20-0025907b26`, `station_raw: CESX`, and `prov:type_raw: waveform`.
- Data File:** Panel showing preview plots for seismic data, including "Phase Arrivals Seismograms" and "STA/LTA" plots for components E and N over time.

Red arrows point from the Monitoring and Discovery panels towards the central Data Resource graph, indicating their integration.

Monitoring:
System and user
Messages

Discovery:
Search on contextual metadata
(within and across runs)

Preview and Download



Visual analytics techniques on provenance

Perspectives on:

- data-intensive processes
- users and applications interactions
- data-reuse
- exploitation of resources..