Persistent Identifiers

Handles and B2HANDLE

Christine Staiger
SURFsara
Workshop, Utrecht, January 18th 2016

EUDAT receives funding from the European Union's Horizon 2020 programme - DG CONNECT e-Infrastructures. Contract No. 654065

Outline

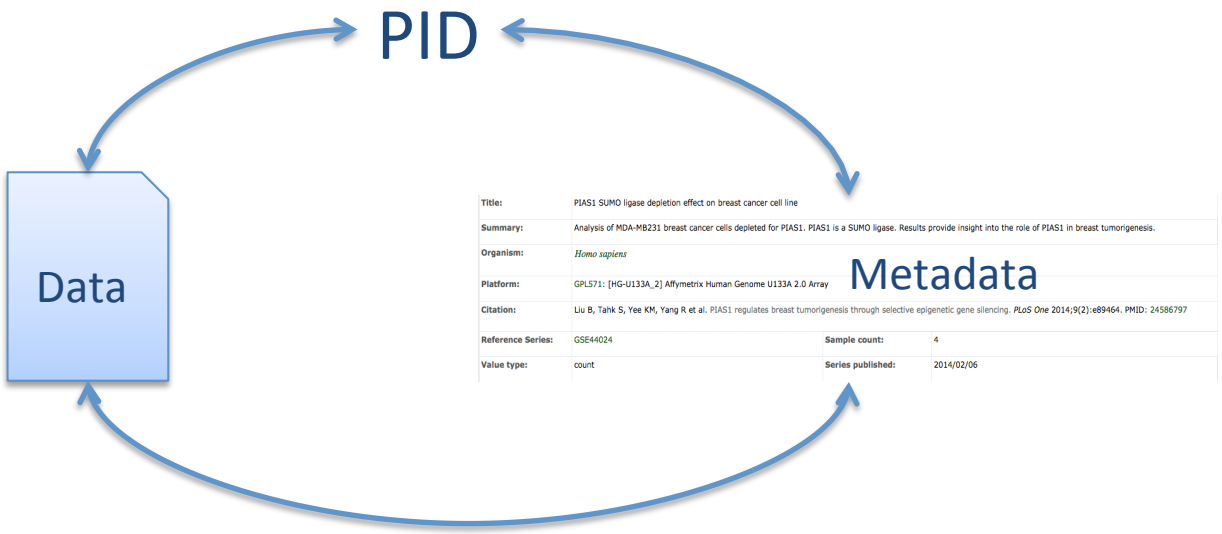
- Why PIDs?
- What are PIDs?
- Use cases
- PID providers and systems

- The Handle system
  - The handle resolution system
  - The relation between Handle and ePIC
  - Hands-on tutorial

What do we want from data?

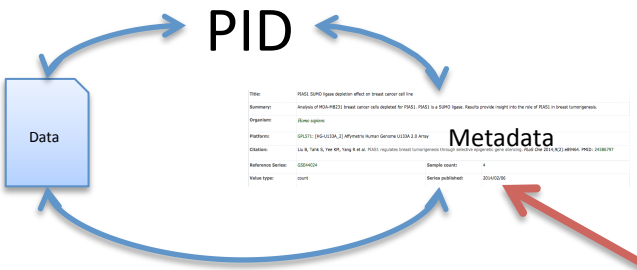
- **Findable** – Easy to find by both humans and computer systems → Metadata
- **Accessible** – Stored for long term, accessed and/or downloaded with well-defined license and access
- **Interoperable** – Ready to be combined with other datasets by humans as well as computer systems;
- **Reusable** – Ready to be used for future research and to be processed further using computational methods.
- http://www.datafairport.org/

What do we need?

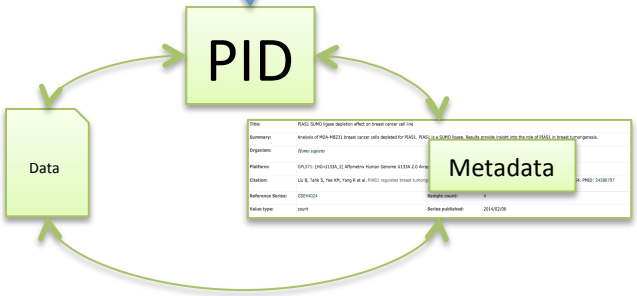


- Persistent Identifier: reference and identify object, either metadata or data object
- Synchronise PID, Data and Metadata during creation, maintenance, update and deletion of a digital object!

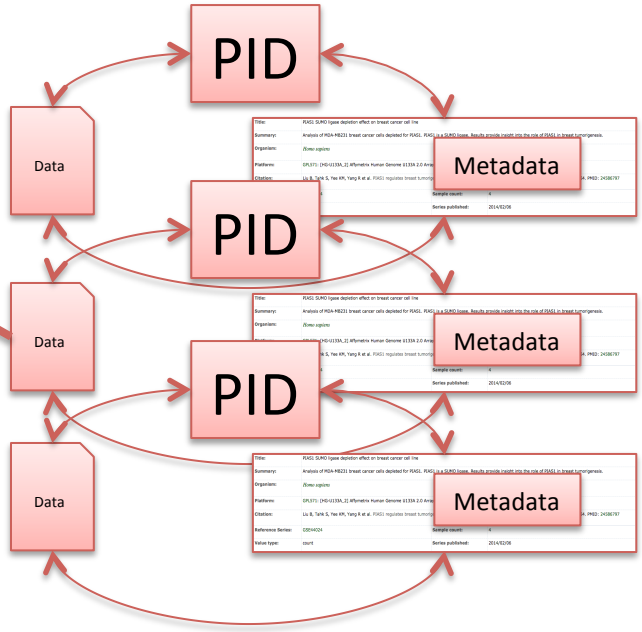
What do we need?



Analysed data



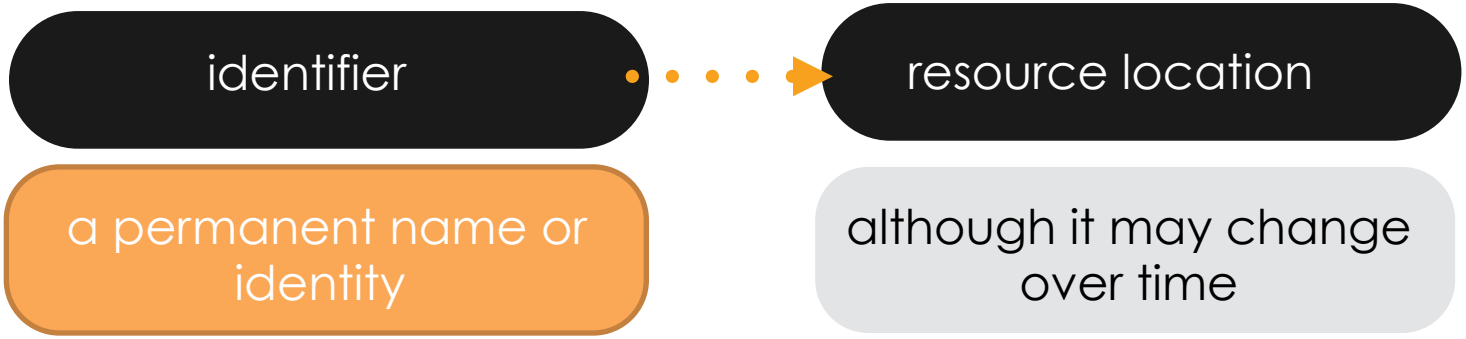
Published data



Raw data

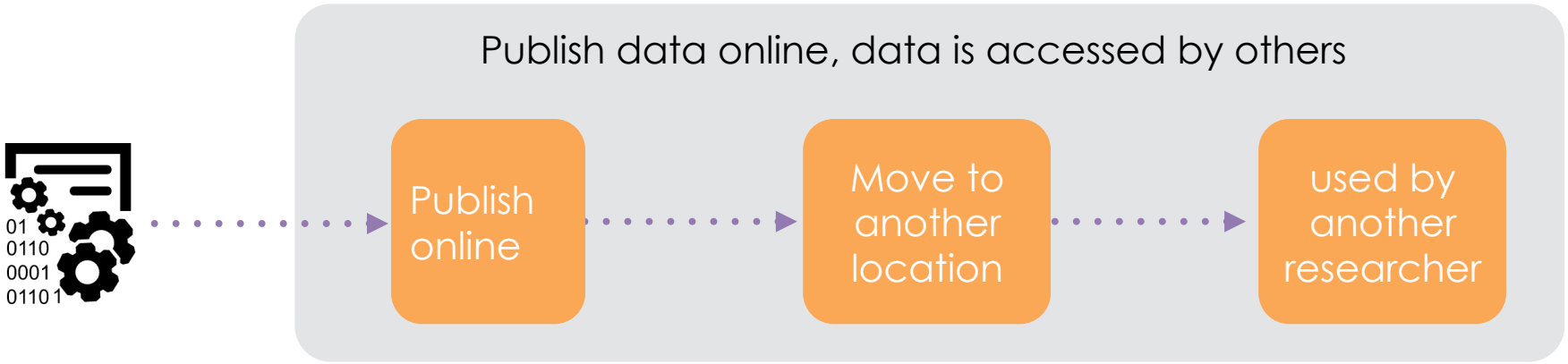
What do we know about Persistent Identifiers?

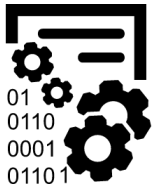
- A Persistent Identifier (PID) is an identifier that is effectively permanently assigned to a resource.



- Pointers to data resources
- Globally unique
- Exist infinitely long (the PID, not necessarily the data)

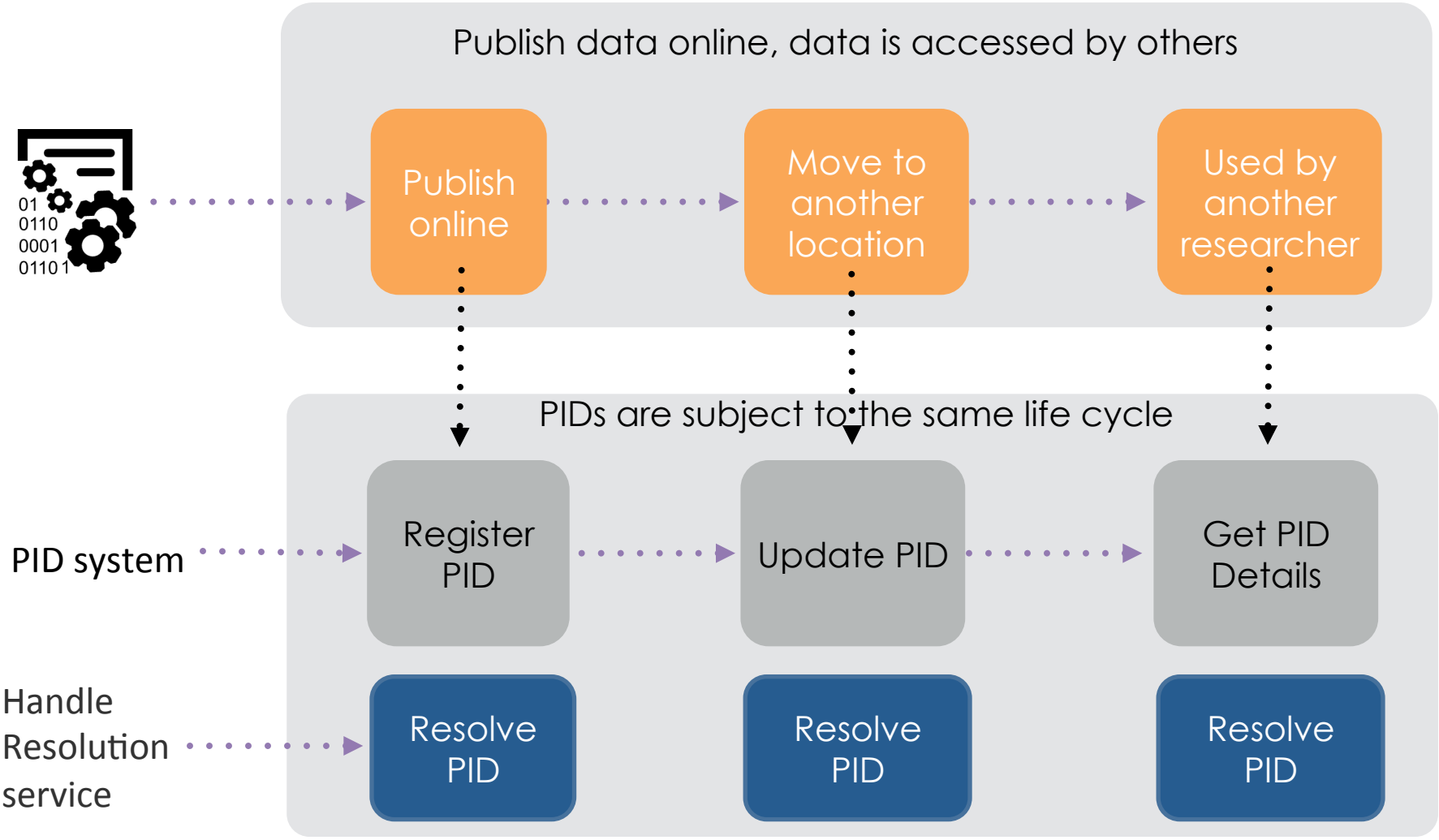
Simple data life cycle, linearised





- Published online: http://www.test.com/test.html
- Other users may cite, access, re-use this url
- Relocate the resource at http://www.example.com/
- Other users are not informed -> 404

Data Life Cycle with PID system



Advantages and Disadvantages

Pro:

- Static reference,
  even if data moves or changes
- Network of persistent links
  Data – metadata relations
  Provenance chains

Con:

- Extra effort
  - What to identify?
  - Coordination across organisations and people
- Organisational discipline to ensure persistence

Use cases

Use Case 1: Digital repositories

- PIDs point to landing page of the digital repository showing metadata
- "Real" data can be downloaded from this page with another link
- E.g. B2SHARE, 4.TU Datacentrum
- PID

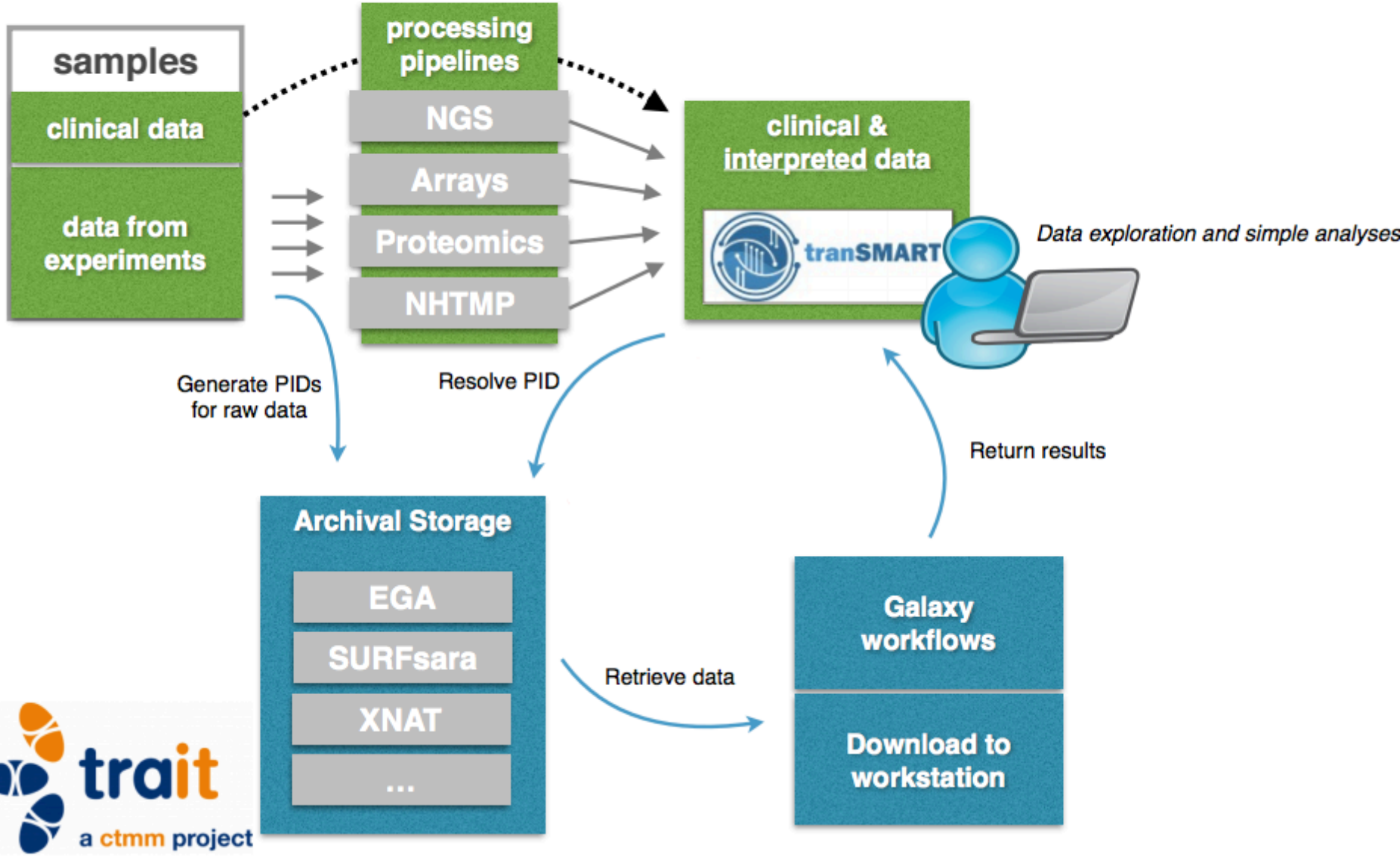
http://hdl.handle.net/11304/3265434c-4b34-11e4-81ac-dcbd1b51435e

resolves to landing page

https://b2share.eudat.eu/records/feafb12e810c489b9e878949c6c35345

Use case 2:
TraIT distritibuted data infrastructure

Molecular profiling dataflow in TraIT



TraIT data ontology

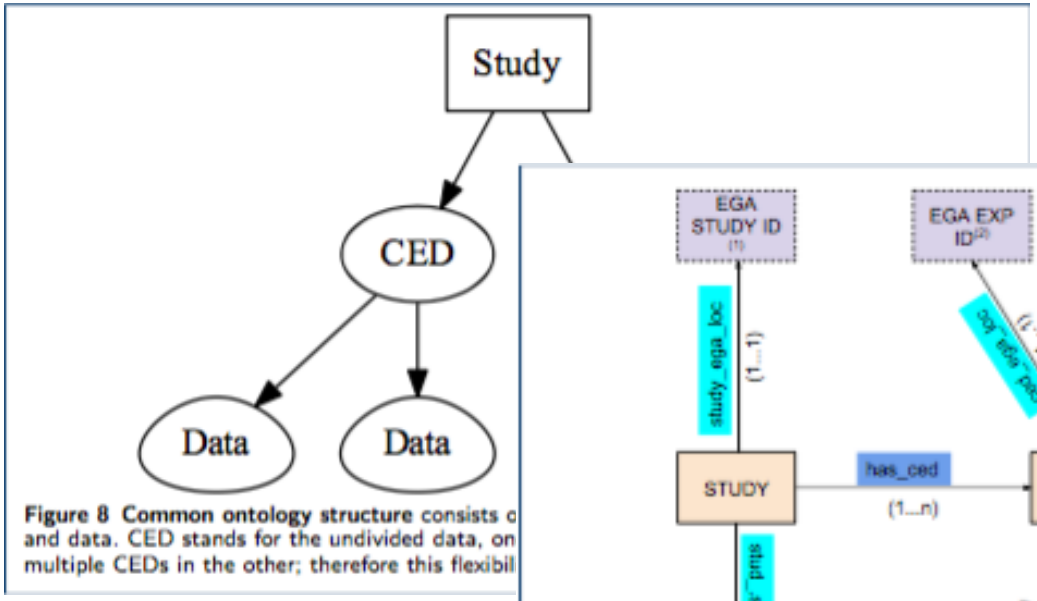


Figure 8 Common ontology structure consists o
and data. CED stands for the undivided data, on
multiple CEDs in the other; therefore this flexibil

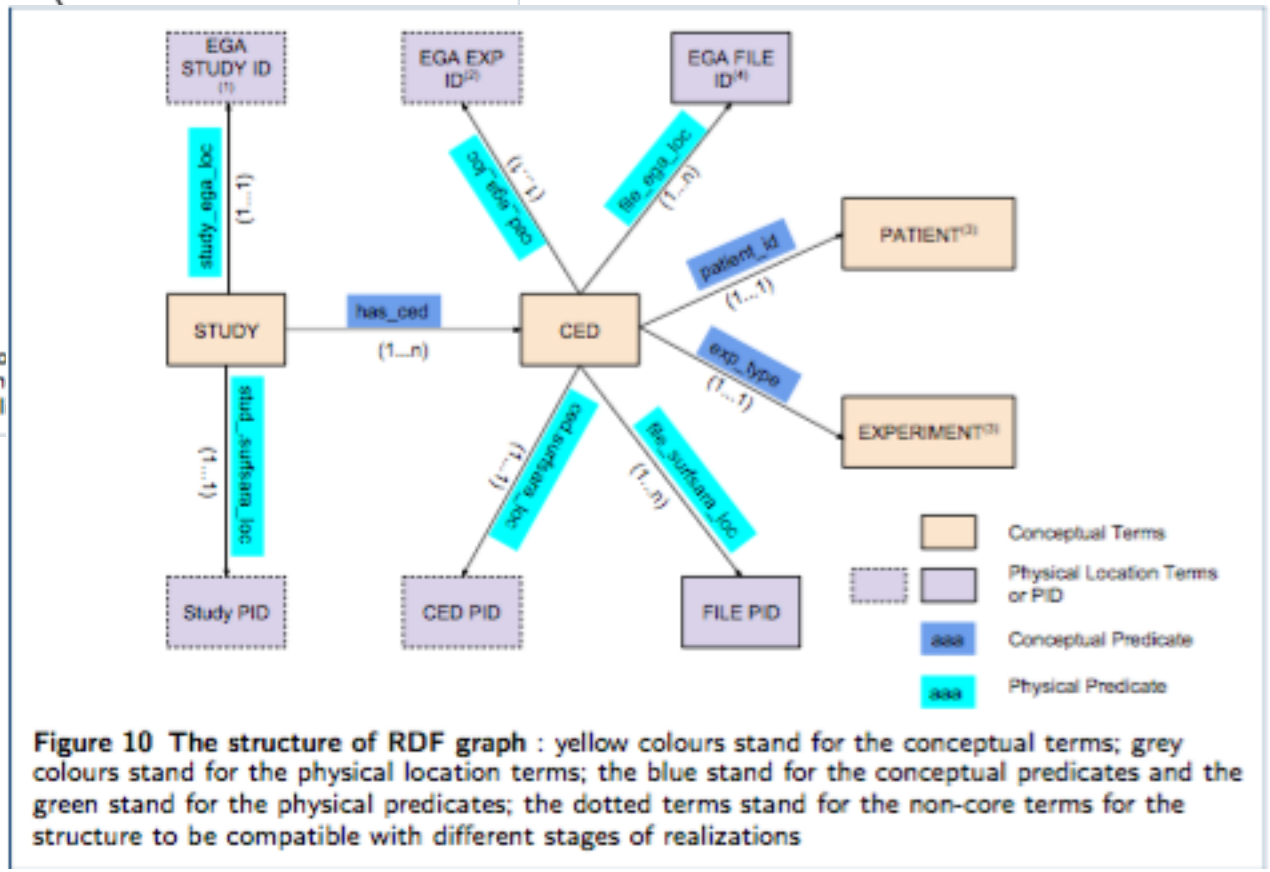
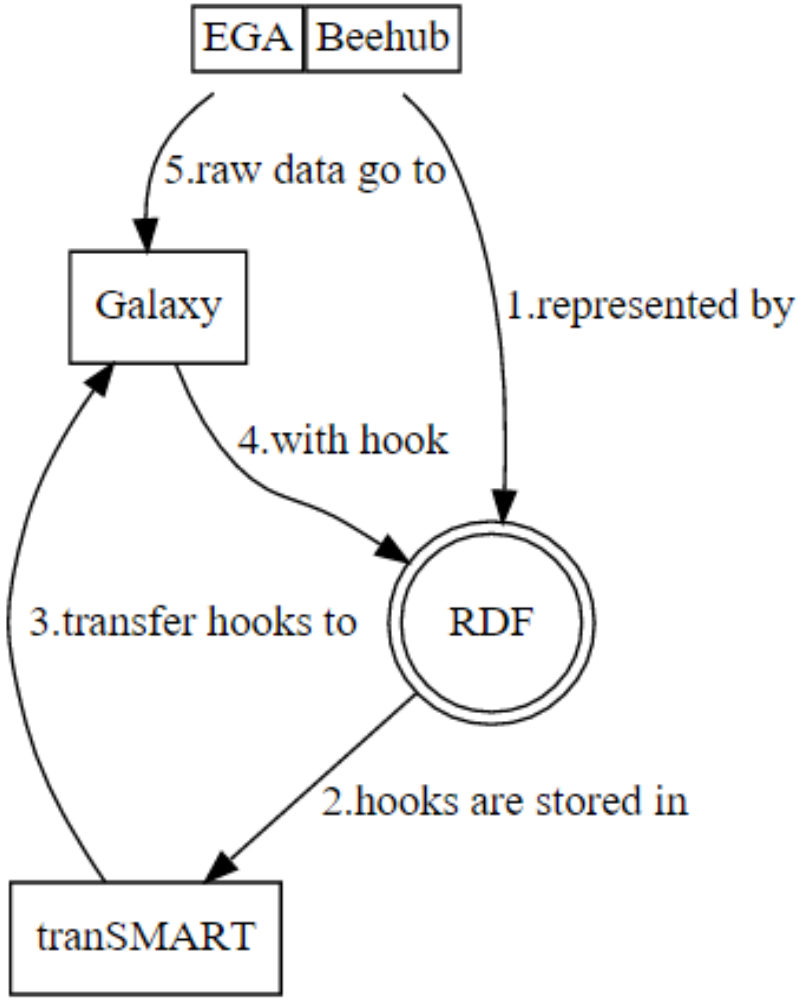


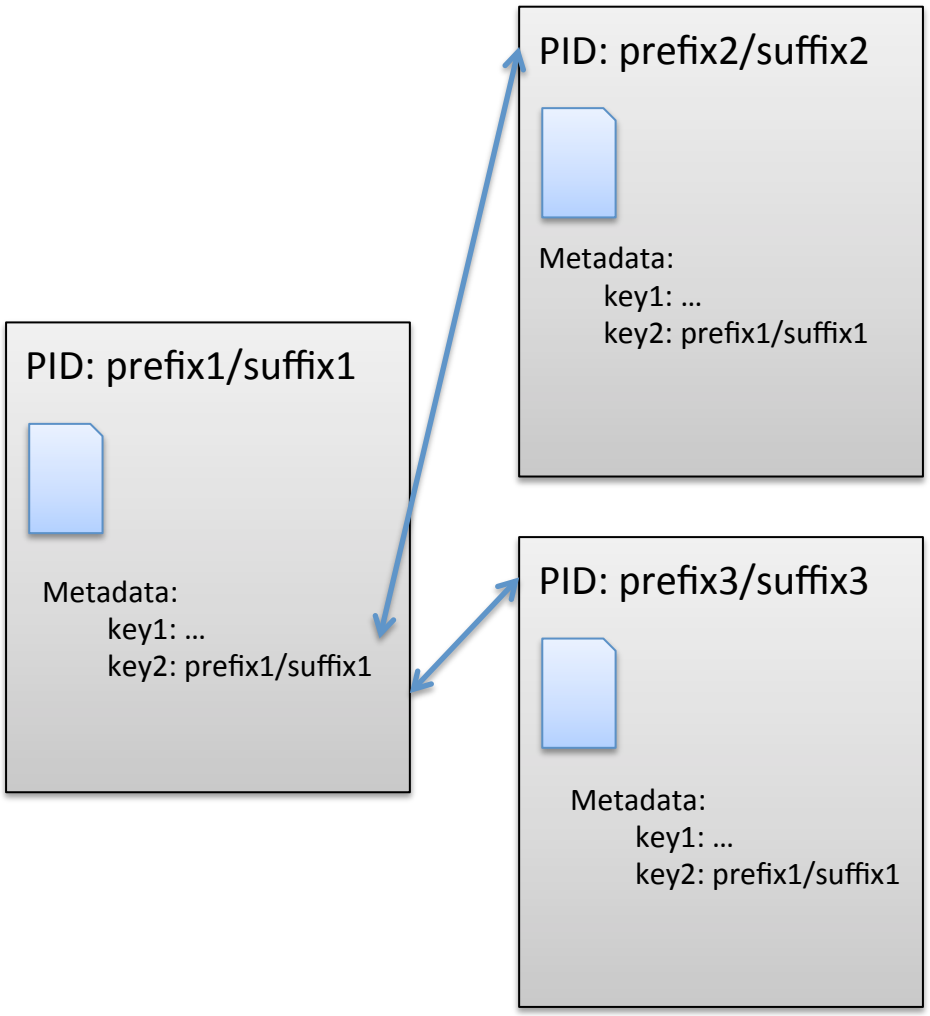
Figure 10 The structure of RDF graph : yellow colours stand for the conceptual terms; grey colours stand for the physical location terms; the blue stand for the conceptual predicates and the green stand for the physical predicates; the dotted terms stand for the non-core terms for the structure to be compatible with different stages of realizations

TraIT data infrastructure



Chao (Cico) Zhang, VU
Sanne Ablen, VU
Jochem Bijlard, VU
Christine Staiger, SURFsara

Use case 3: Modeling Relationships



- Use tightly coupled metadata
- Part of/has part relationships
- Model cohort-patient relationship
- Model patient-samples relationship

Which metadata to store with the PID and which in en extra catalogue ?

Use Case 4: Enabling workflows

- Execute program hidden behind a PID
- Way to refer to workflows → reproducibility

```
In [16]: prefix = "841"

In [17]: suffix = "/5f6fb451-5841-11e4-9665-14109fe83170"

In [18]: ec.getValueFromHandle(prefix, "URL", suffix)
Out[18]: '/Users/christines/PIDs/helloWorld.py'

In [19]: pid = subprocess.Popen([sys.executable, ec.getValueFromHandle(prefix, "URL", suffix)])

In [20]: Hello World!
```

PID systems

Resolultion and the PID pattern

Handle

PID: 21.T12995/B2SAFE-B2STAGE

Resolver: http://hdl.handle.net/

Doi

PID: 10.2189/asqu.2005.50.3.329

Resolver: http://dx.doi.org/

Ark

PID: ark:/13030/tf5p30086k

Resolver: https://nbn-resolving.org/

Exercise

- Resolve the PIDs
- What happens if you resolve a PID with a foreign resolver?

PID systems and issuing authorities

- **URN:NBN (DANS, resolved via archival resource key (ARK))**
  - Policies: PID is persistent and the data it is dereferenced to
  - Wants to be independent from transfer protocols
    → Currently all identifiers start with *http*
    Might change in the future

- **DOI (4TU.datacentrum)**
  - Policies: PID is persistent, data not
  - Well accepted amongst researchers
  - Based on the handle system
  - Datacite, Crossref are prefix issuing authorities

- **Both:**
  - PIDs **point to a landing page**, not the file itself
  - User needs to provide **a minimum set of metadata**
  - Taylored towards the needs of repositories

PID systems and issuing authorities

- **ePIC (European PID consortium)**
  - Policies: PID is persistent, data is not
  - PIDs can point to anything
  - Based on the handle system
  - Handle prefix issuing authority

- **DONA foundation (www.dona.net)**
  - Maintains global handle registry
  - Partners:
    - CNRI (developer of the handle system)
    - GDWG (main partner in ePIC)
    - International DOI foundation (IDF)

The Handle system

- Pure technology!
- Metadata: You can create your **own keyword-value pairs** and store them with the PID

- Policies: **Do it yourself**!

  handles can point to anything

  handles can also be removed, they are not per se persistent

  …

  → Great flexibility for adjusting the system towards your own needs

  → You have to implememnt everything on your own

  → You have to think even more carefully about how you want to facilitate data management

For whom?

- PIDs allow to make a **distinction between data users and data managers**
  - Data users get a PID and can directly access the data, or the metadata stored with the PID
  - Pipelines can programmatically access the metadata and start specific applications

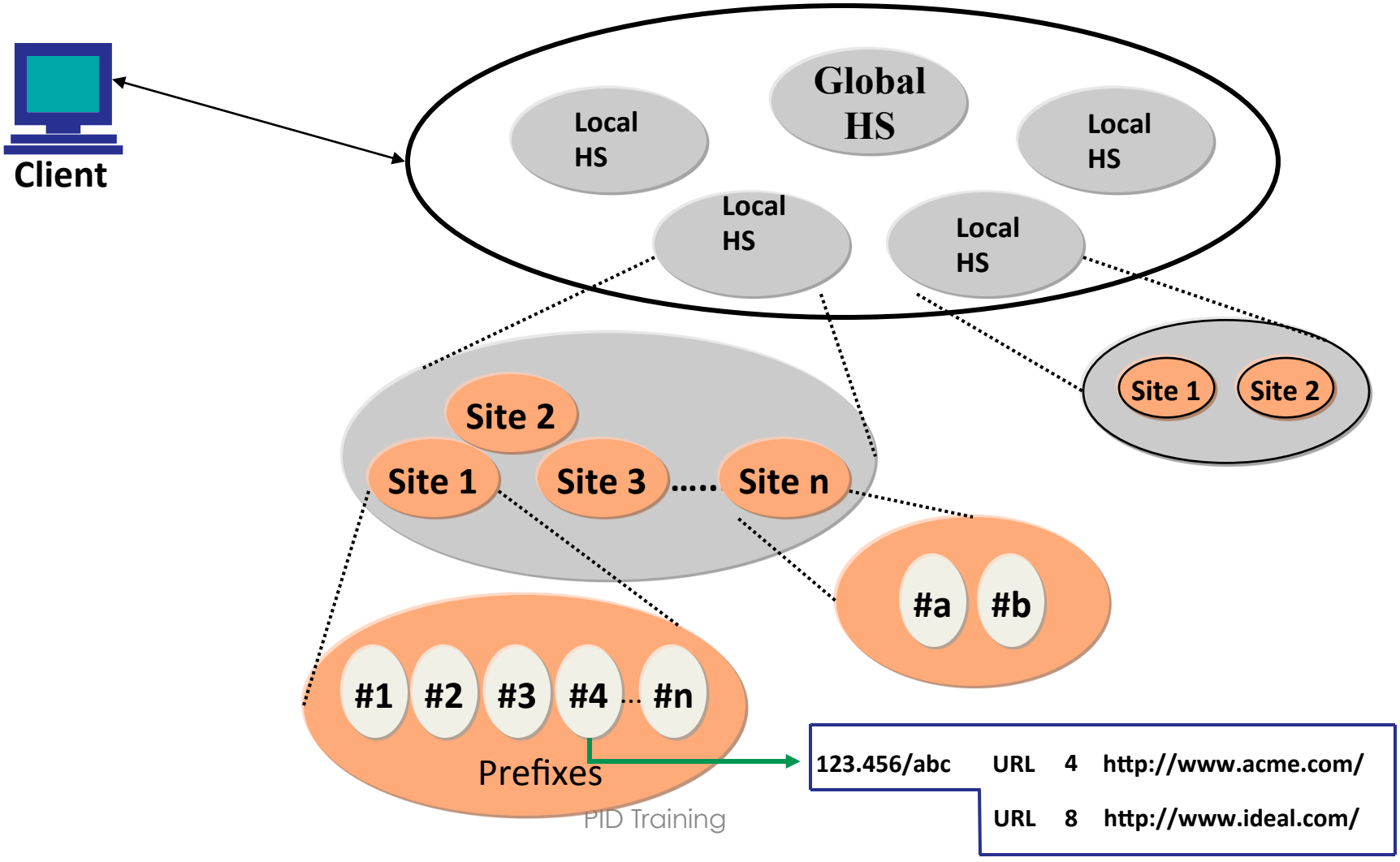
- Requires some serious thoughts about **data organisation** and developing the **code to put data policies into practice**, including code **maintenance**

→ For **bigger research groups or consortia** working in a **distributed data environment**

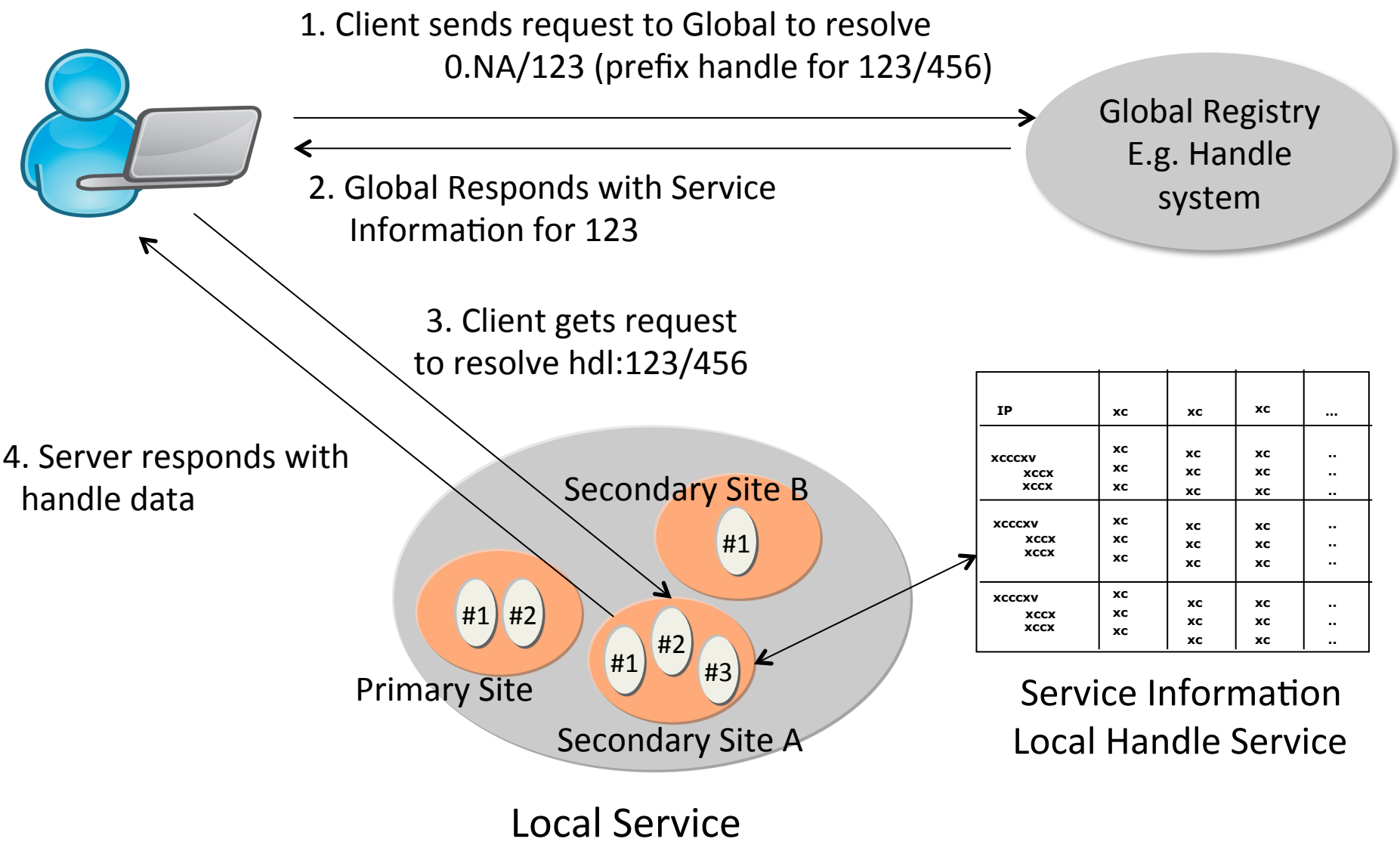
→ For **repositories** who are in need of a host for their PIDs

The Handle system: technical details

Resolution system



Resolving PIDs



Step by Step: Using the HANDLE API

- Register data with a Handle
- GET the details of a Handle
- Modify a Handle record
- Link two files on PID level
- Reverse look-up (not possible via normal Handle API)

Access to training machines and exercises:

http://hdl.handle.net/21.T12995/B2SAFE-B2STAGE

Login/Password: