FOUNDATIONS OF THE ICOS CARBON PORTAL SERVICES

OLEG MIRZOV

ICOS Cities webinar, 9 November 2022

https://doi.org/11676/cpFundamentals

CARBON PORTAL

- "One-stop shop" for ICOS data
- Hosted by Lund University
- Based at the Department of Physical Geography and Ecosystem Science


GEOCENTRUM II

REQUIREMENTS (FORMULATED BY 2015)

- tampering-resistant traceability of data products to raw data
- long-term preservation: data are stored in (or backed up to) external trusted repository
- "online data shop" concept: searching for data, previewing, adding to "data cart", downloading
- the data files must be citeable, with data producers mentioned as authors in the citation
- data downloads and citations should be counted
- the Web services should be useable by machines as well as human users
- CP should provide single sign-on functionality for all the Web services

DESIGN CHOICES (EMERGED BY 2016)

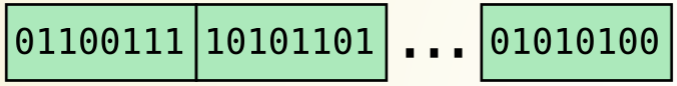
- data objects are immutable, their identifiers are based on content and permit integrity control by design
- RDF technology stack chosen for metadata, due to its purpose of representing machine-interpretable knowledge on the Web
- all the metadata changes are logged as RDF statement assertions/retractions
- the APIs are built on HTTP
- write operations during data upload are idempotent (can be repeated without extra effects)
- operations on data during upload are parallelized by cloning the data stream and sending it to several targets (file storage, parsing/ingestion, backup in a remote repository, integrity control, byte count), and upload succeeds only if all the operations succeed
- authorization is handled individually by each service, authentication is handled centrally by a dedicated service

CARBON PORTAL SERVICES

- Data upload API (technical)
- Hosting of the data in accordance with the FAIR principles
- Search interface for data objects
- Visualizations/previews of data
- Metadata database
- Client app and API for user-specified metadata queries
- Minting of PIDs (including DOIs) and hosting landing pages
- Hosting Jupyter notebooks with direct ICOS data access
- A variety of other, both in-house-built and third-party, services
- The core services are also repurposed for SITES research infrastructure

DATA OBJECT DEFINITION

N bytes ≡ data object data



SHA-256 hashsum

hex	`9f86d081884c7d659a2feaa0c55ad015a3bf4f1b2b0b822cd15d6c15b0f00a08`
base64	`n4bQgYhMfWWaL+qgxVrQFaO/TxsrC4Is0V1sFbDwCgg`
base64url	`n4bQgYhMfWWaL-qgxVrQFaO_TxsrC4Is0V1sFbDwCgg`

(24 characters ⇔ 18 bytes) ⇒ collision probability of 50% for 5.56×10^{21} hashes

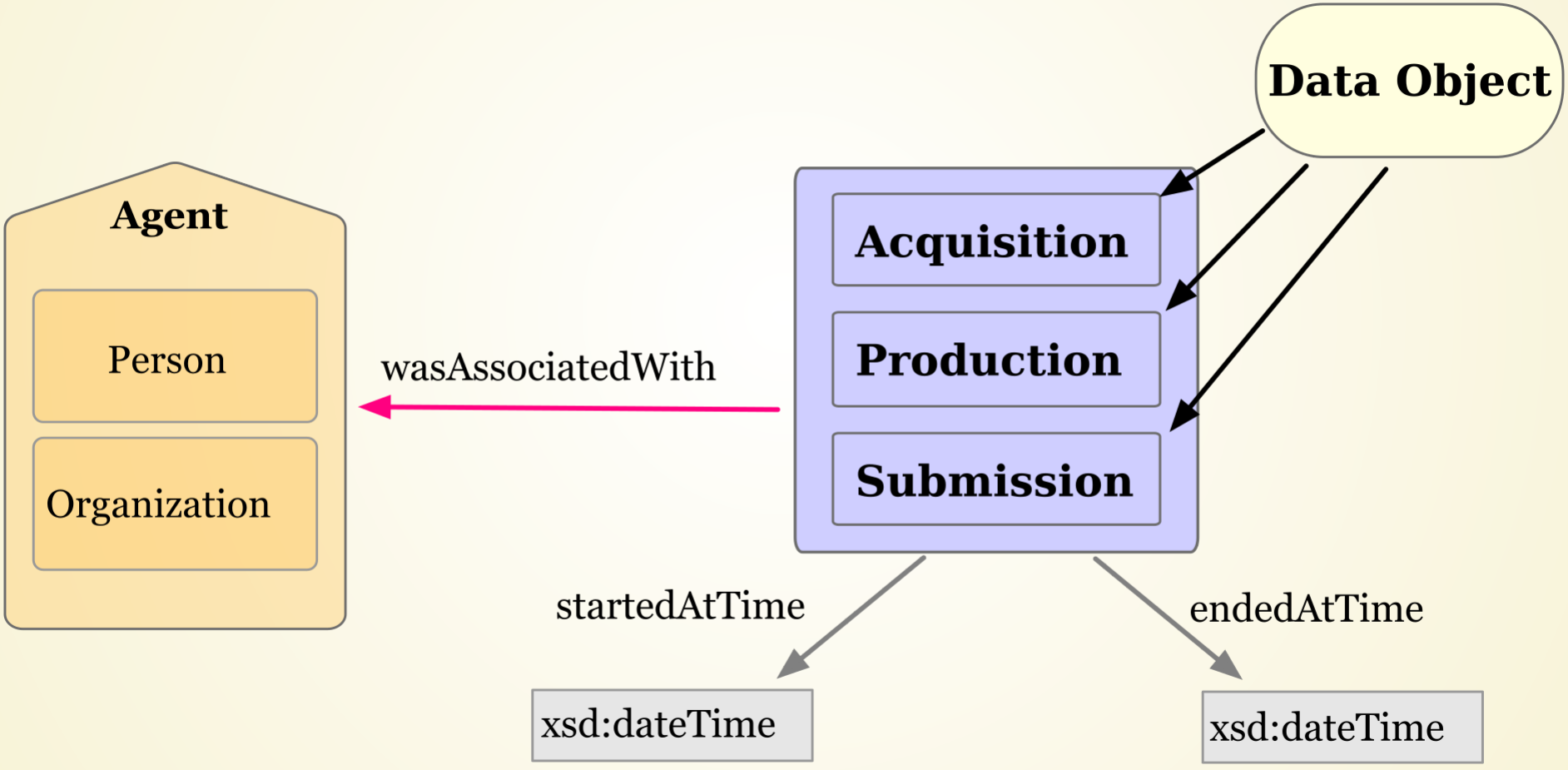
METADATA IS THE KEY

- Describes stations, people, roles, instruments, data objects and their provenance, etc
- Published according to the Linked Data principles
- Expressed and served using standards and technologies from the Semantic Web (RDF) stack

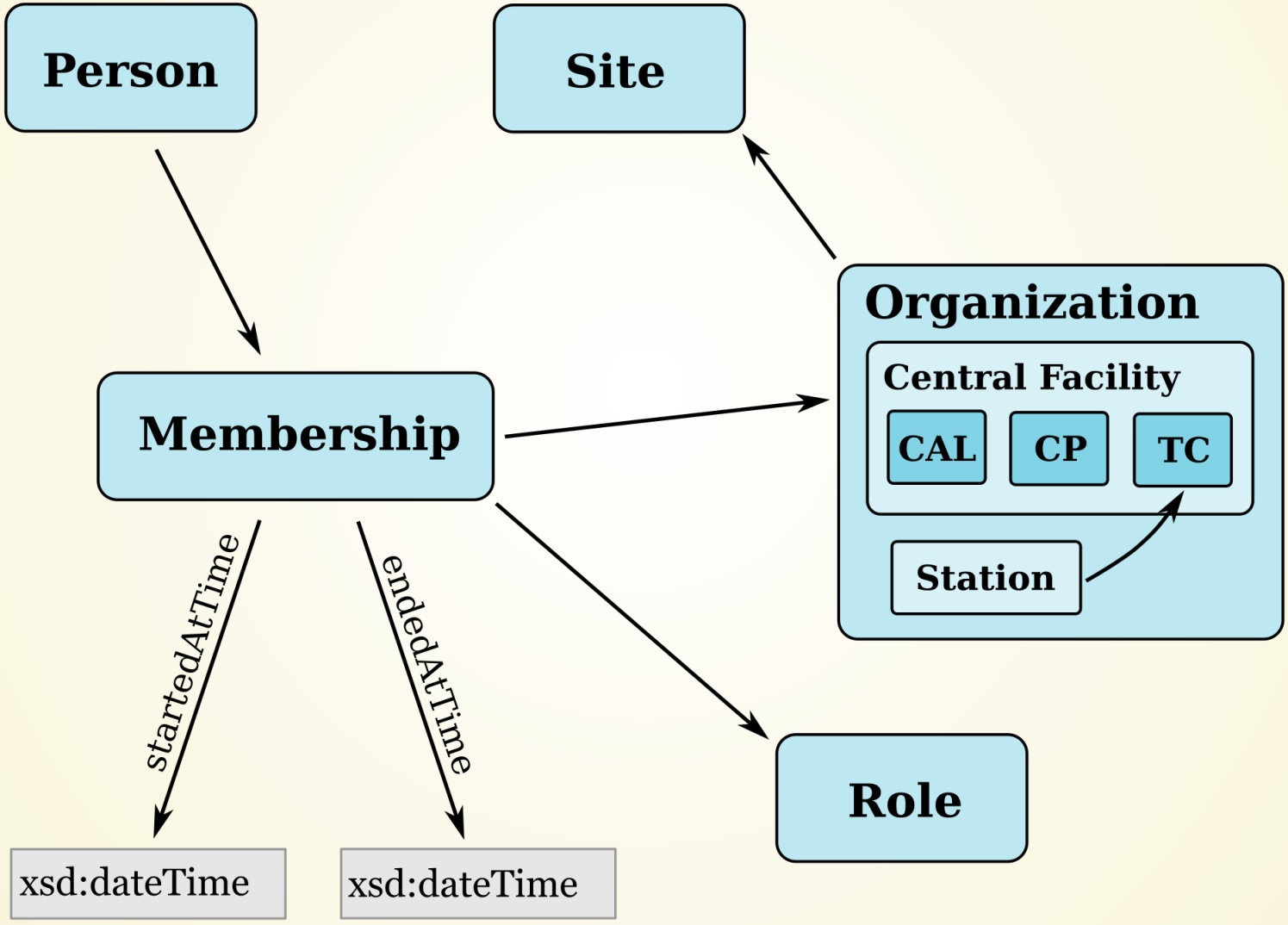
BASIC DATA OBJECT METADATA

Landing page	https://meta.icos-cp.eu/objects/zFf-_y7HHNXh821kwRq4auds
PID	https://hdl.handle.net/11676/zFf-_y7HHNXh821kwRq4auds
File name	ICOS_ATC_L2_L2-2021.1_HPB_131.0_CTS_CO2.zip
Size in bytes	1021496

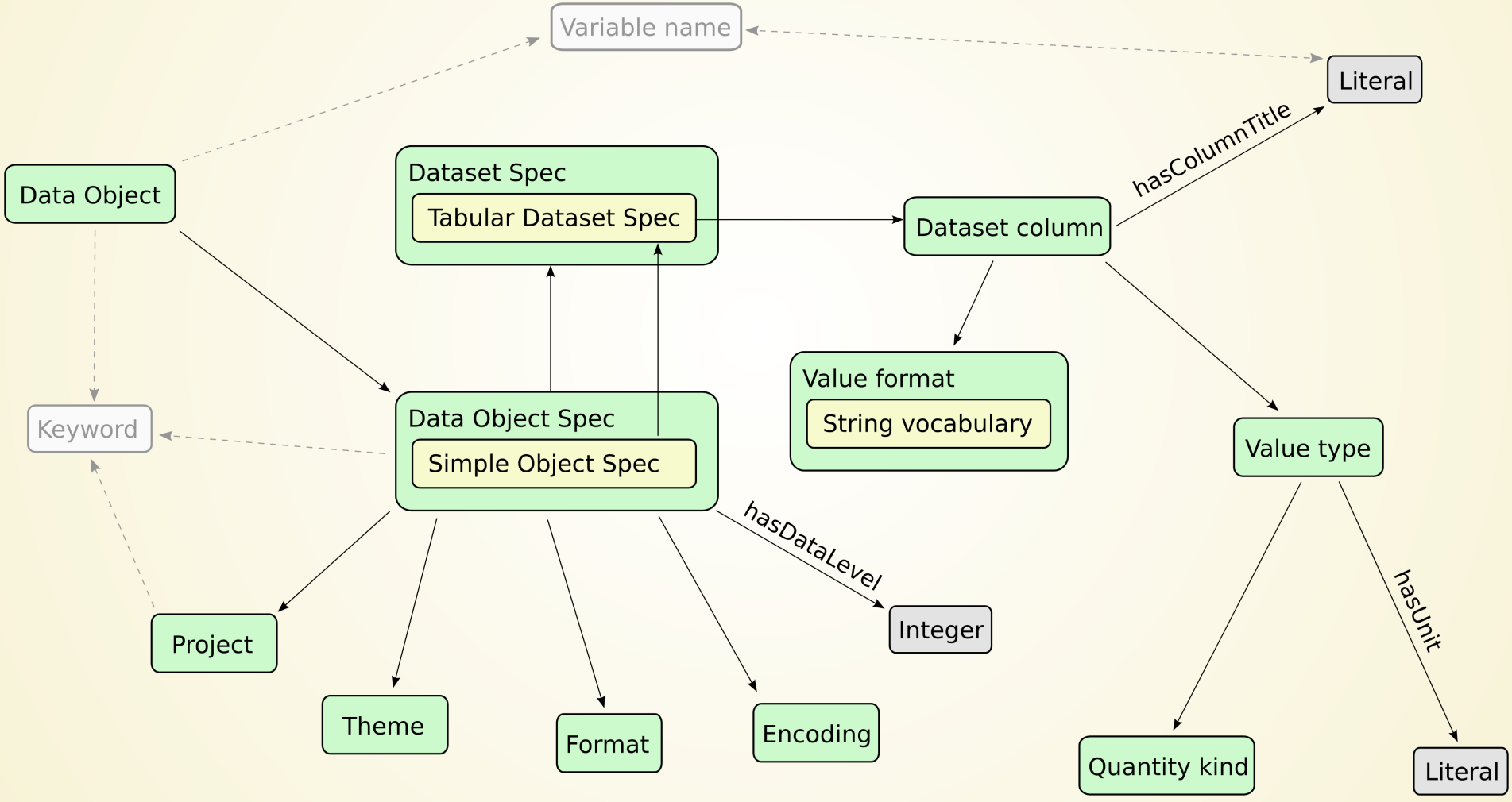
PROVENANCE METADATA



PERSON/ORGANIZATION/ROLES METADATA



DATA OBJECT SPECIFICATION ("DATA TYPE")



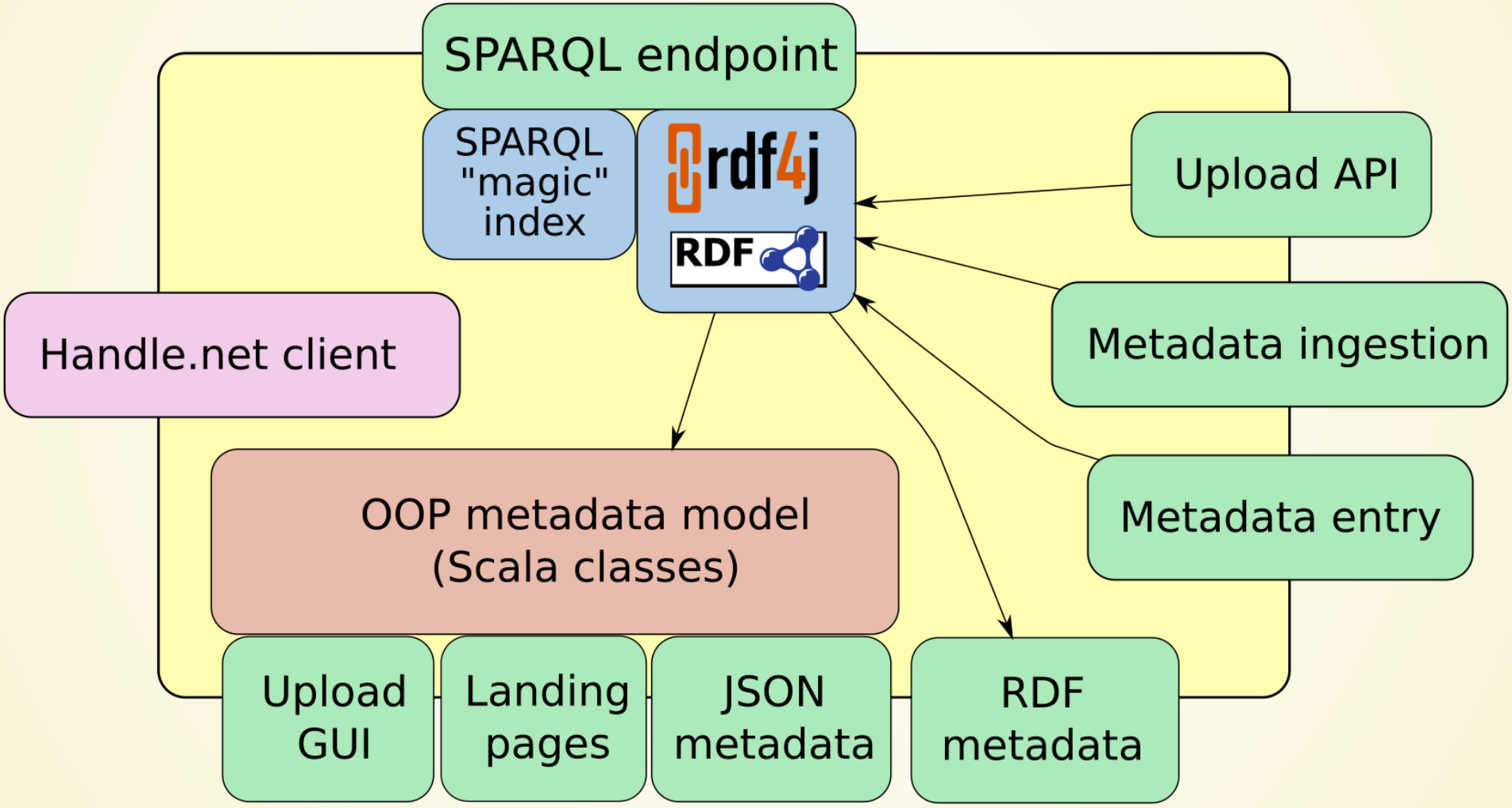
BASIC IDEAS OF RDF

- Every thing gets a URI (URL)—a globally unique id
- Server response to a URI depends on the type of a client
- All the statements are made in the form of triples "**Subject - predicate - object**"
- Objects can be typed literals, the rest is URIs
- Object of one triple can be a subject of another. Thus, the data model is a **directed labeled graph**

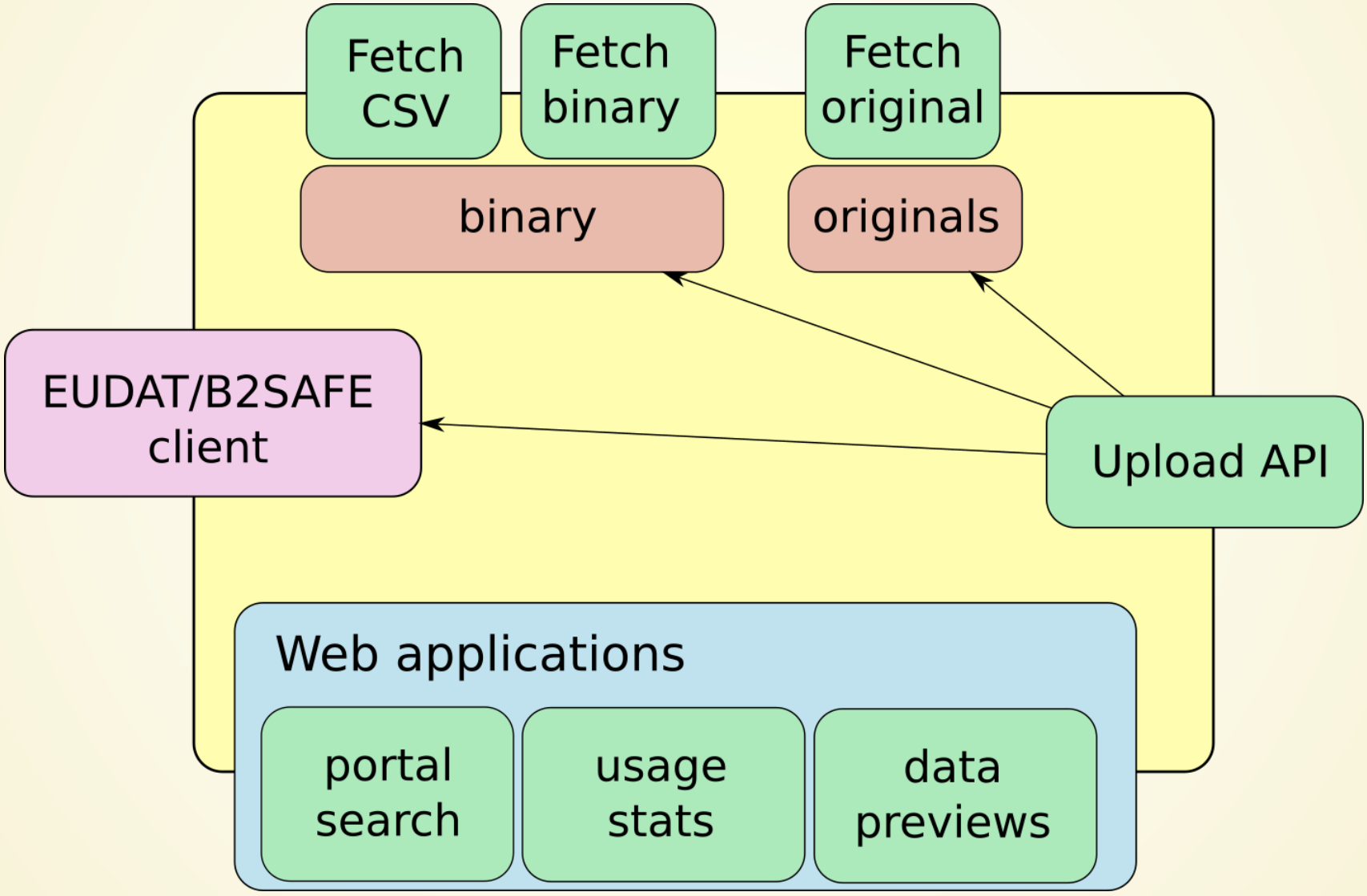
DEMO

- Data object landing page example: https://meta.icos-cp.eu/objects/Jdk1jkyJW15f1pku3dONP96L (ICOS ATC CO2 Release, Hyltemossa, 150.0 m)
- The search app: https://data.icos-cp.eu/

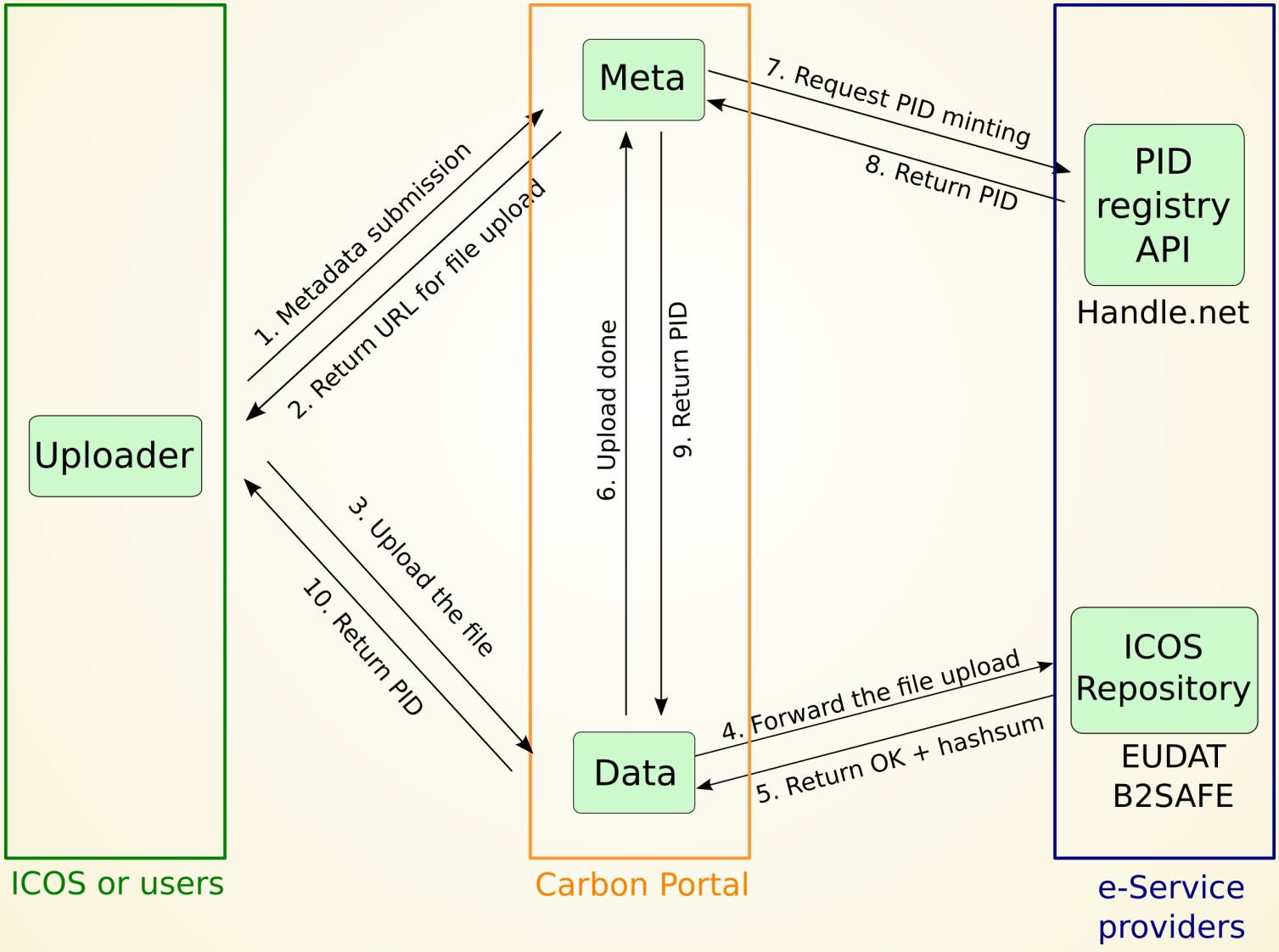
CORE SERVICES: THE METADATA SERVER



CORE SERVICES: THE DATA SERVER



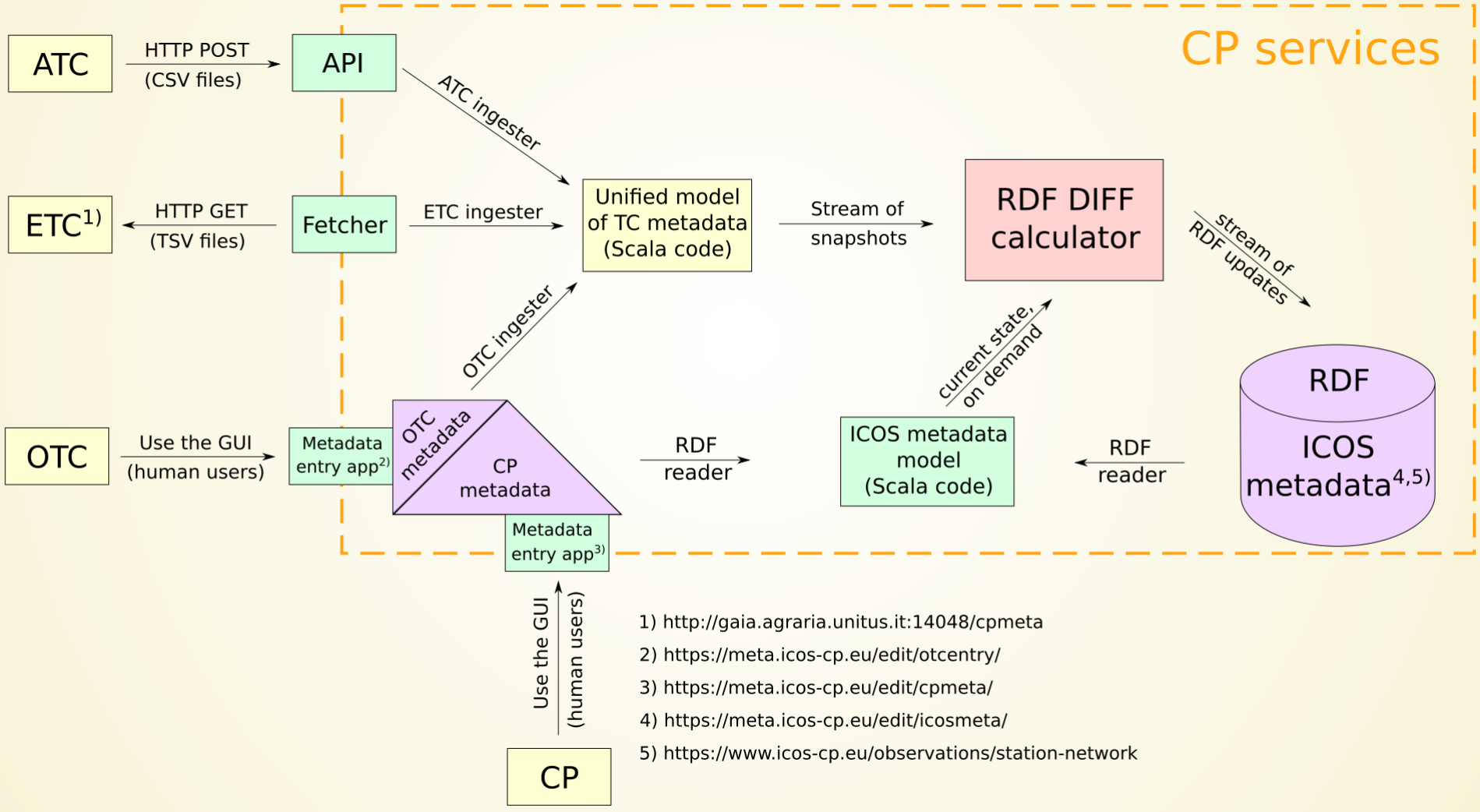
DATA UPLOAD: ALL THE STEPS



DATA UPLOAD: INSTRUCTIONS FOR UPLOADERS

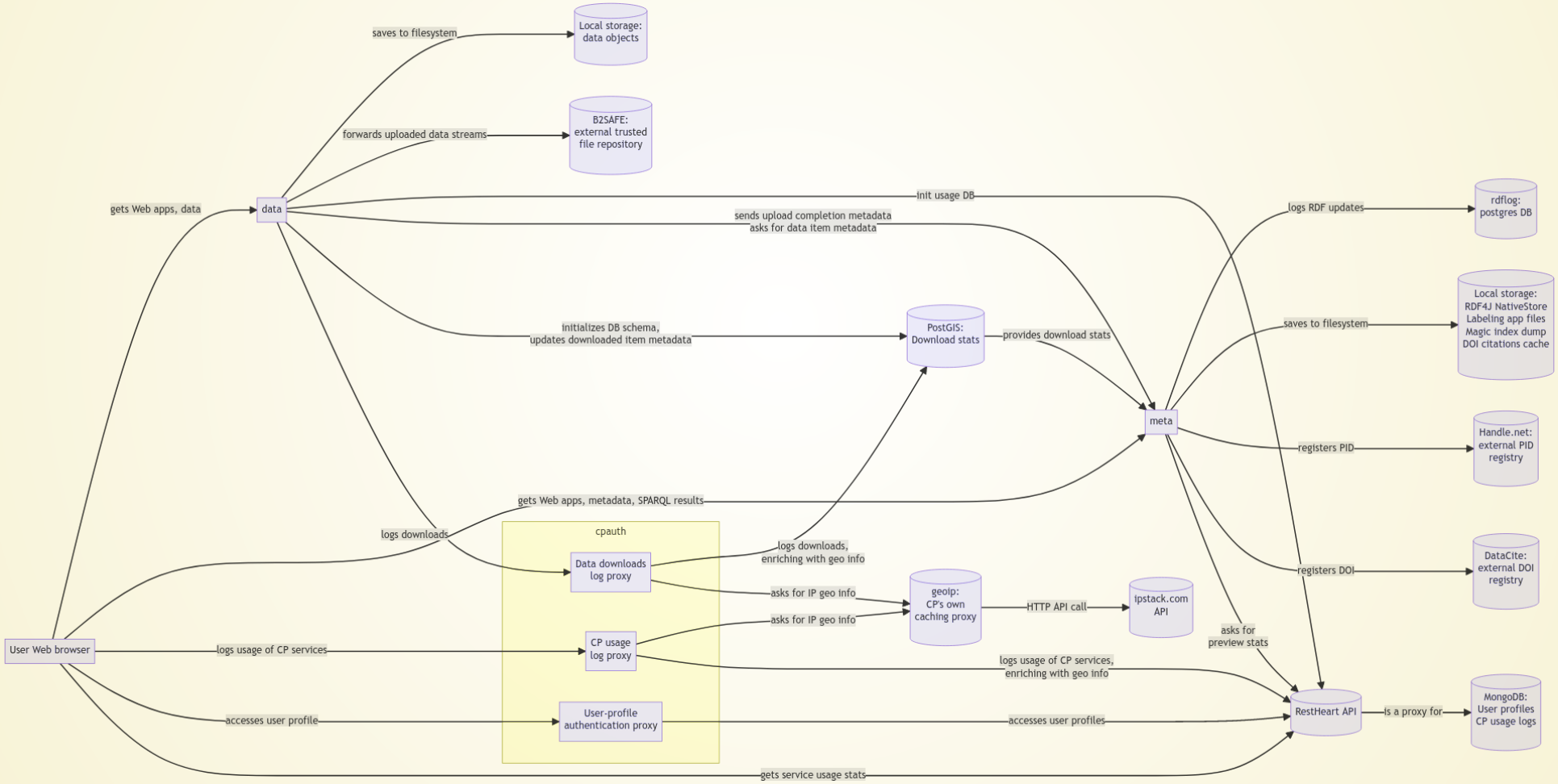
- Step 0: authentication
- Step 1: metadata
- Step 2: data

METADATA FLOW FROM THE THEMATIC CENTERS



1) http://gaia.agraria.unitus.it:14048/cpmeta
2) https://meta.icos-cp.eu/edit/otcentry/
3) https://meta.icos-cp.eu/edit/cpmeta/
4) https://meta.icos-cp.eu/edit/icosmeta/
5) https://www.icos-cp.eu/observations/station-network

MORE DETAILS: COMMUNICATIONS DIAGRAM



MORE DETAILS: TECHNOLOGY CHOICES

- back end: JVM + Scala
- HTTP:
  - reverse-proxy, TLS: NGINX
  - middleware, data streaming: Akka
- RDF/SPARQL: RDF4J + custom-built index to speed up specific SPARQL patterns
- other databases:
  - PostgreSQL for RDF changelogs, data download logs
  - MongoDB/RestHeart for user profiles, Web service usage logs
- front end: Typescript, ScalaJS, Javascript, ReactJS/Redux
- build tools: back end—SBT, front end—Gulp, Webpack
- deployment automation: Ansible, SBT plugins
- server OS: Ubuntu Linux, containerization: Docker, LXD
- Jupyter notebooks, Python utility library for ICOS (meta-)data access

MORE DETAILS: LINKS

- Extra architecture-info slides
- ICOS CP organization GitHub page: https://github.com/ICOS-Carbon-Portal/
- Main ICOS webpage: https://www.icos-cp.eu/