Database Solutions for Research Data Management

Rheinland-Pfälzische Technische Universität Kaiserslautern Landau

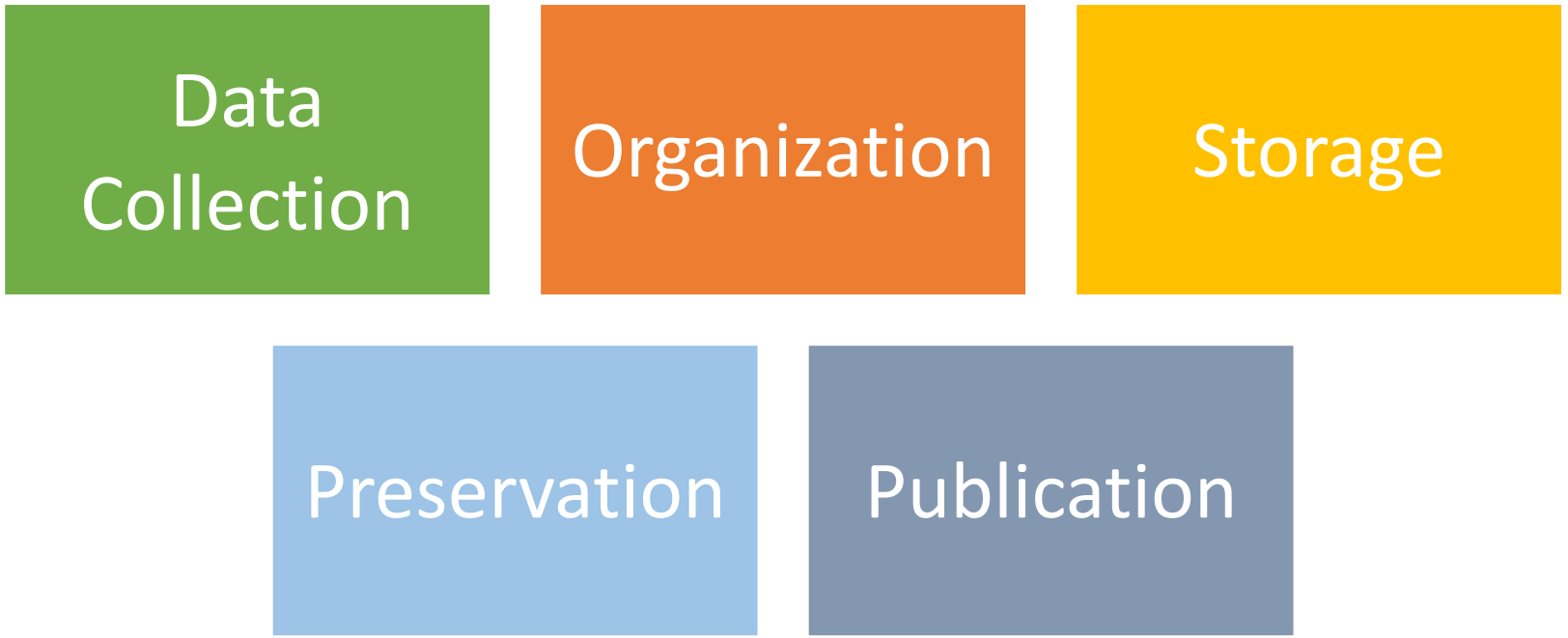
RDM Summer School

Gajendra Doniparthi

19 July 2023

Data PLANT

Research Data Management



Types of research data

- Images
- Text
- Annotated Texts
- Documents
- Audio/Video tapes
- Presentations
- PDFs
- Data Files …

Fair Data vs. Open Data

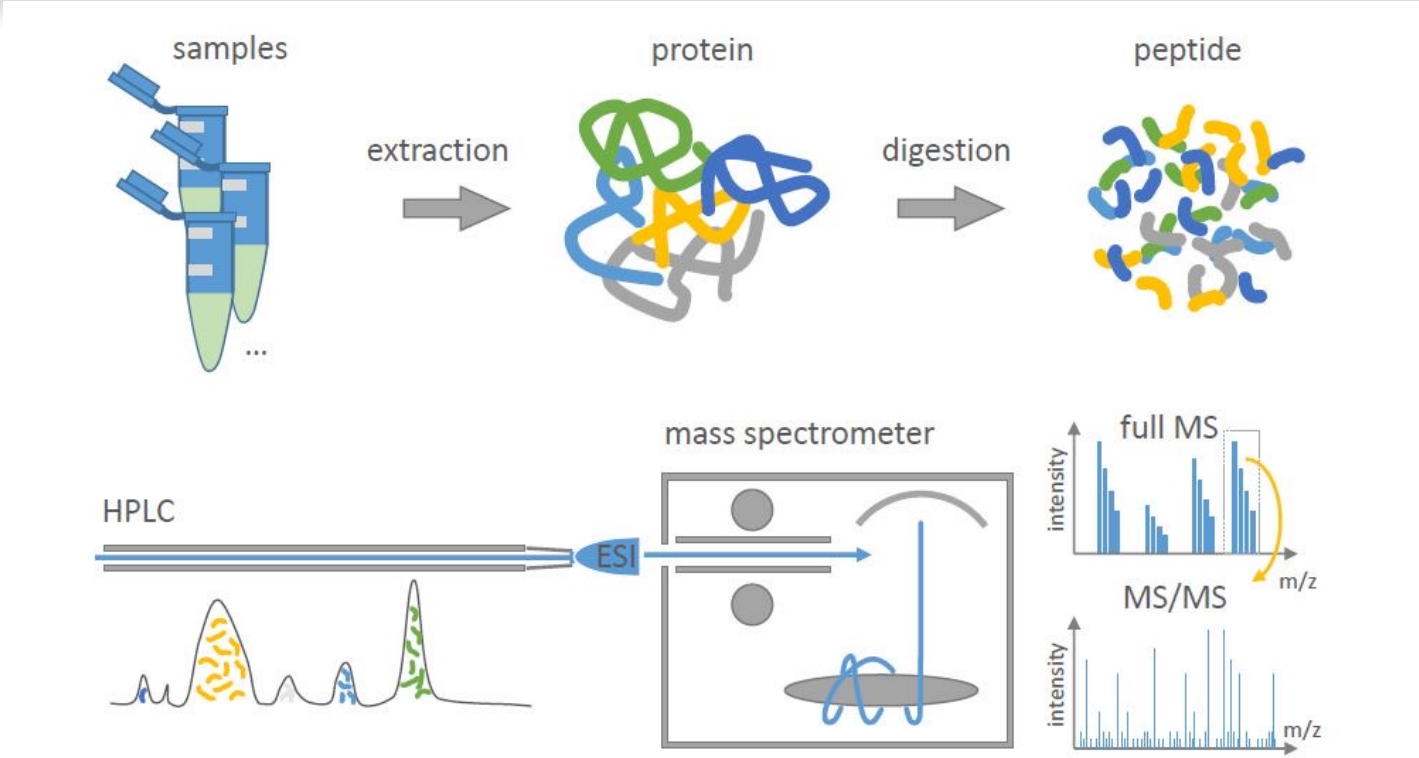
- Findable
- Accessible
- Interoperable
- Reusable

vs.

- Openly accessible
- Exploitable
- Editable
- Sharable

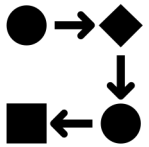
Research workflows are often complex!

Proteomics workflow



Complex workflows mean complex sets of data to manage.

Metadata



Computational workflows

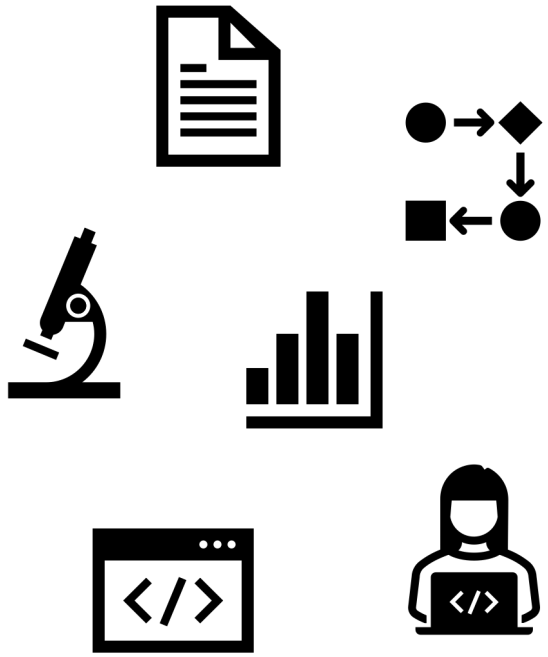
Additional Software

Measurement data

Scripts

Data Analysis

DataPLANT project solve the complexity through ARCs.

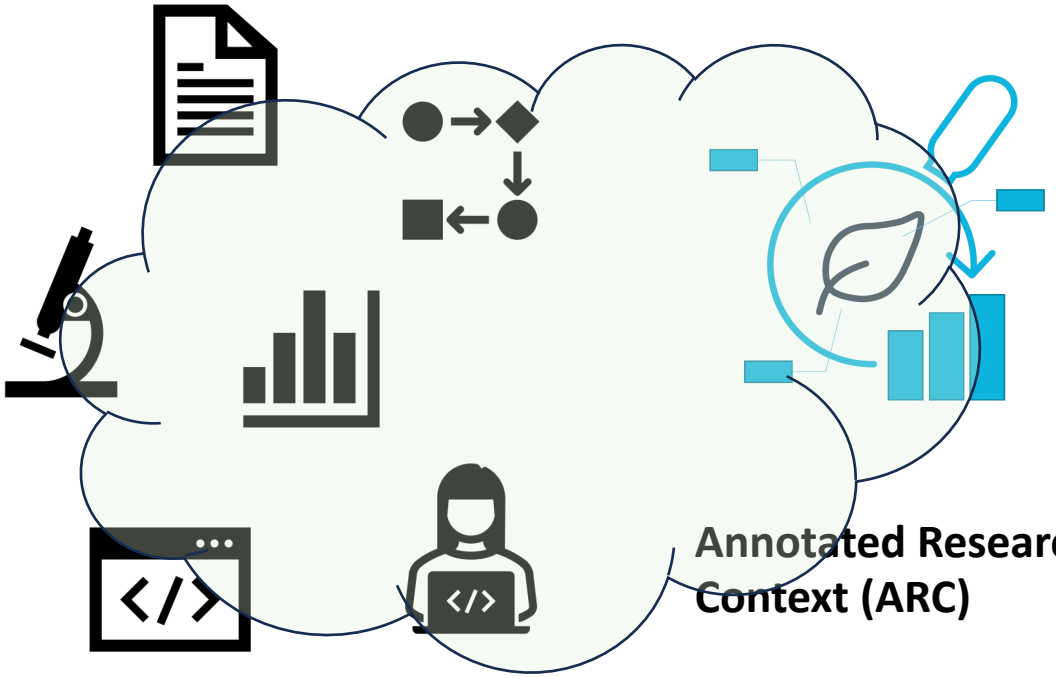


Data PLANT

Annotated Research Context (ARC)

- Reproducibility
- Interoperability
- Forward data to Public Repositories
- FAIR Principles

From the outside, everything is nothing but data!



How to search and explore the Research Data?

Identification

- File types.
- Metadata and raw data files (runs, workflows)
- X-references.

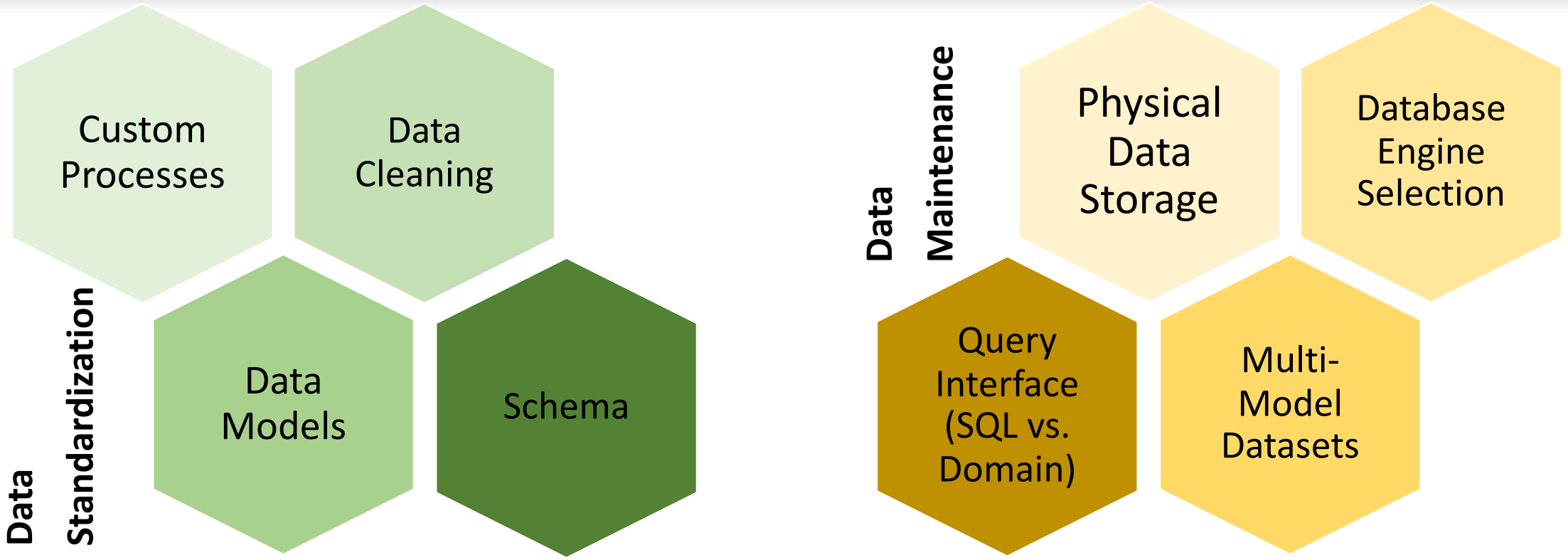
Data Standardization

- Standard formats and data model selection.
- Data cleaning & Transformation.

Maintenance

- Physical vs. Virtual
- Databases -> Data Lakes.
- Data Indexing.
- Query processing.

What are the challenges?



Data PLANT

There are quite a few data models in practice.

Relational 

Key-value 

Graph 

Document 

Text Search 

Query Interface plays an important role.

- Useful to hide the heterogeneity of the data sources.
- Domain query model vs. standard query model.
- SQL vs. No-SQL
- Client-interfacing for query execution.

Polystore Systems can be one of the database solutions.

- Provides an integrated view on multiple data sources.
- Each storage engine may have different data model.
- Query interface to access multiple sources.
- Splits input query into multiple sub-queries.

Data PLANT

Distributed Query Processing Frameworks are different.

- Open-source SQL query engine for Big Data Exploration
- Can access multiple data sources at once
- SQL query interface
- Schema-free JSON model

Can Data Lakes be also one of the solutions?

- Ability to ingest and store data in the form of structured, semi-structured or unstructured.
- Three logical phases -> Ingestion, Maintenance and Exploration
- Data standardization and modeling techniques.
- Advanced query languages and Indexing capabilities.

Moving physical data from ARCs to a Database solution

Identification

- Identify the files to be processed.
- Capturing metadata.
- Automatic schema generation.

Data Transformation

- Standard data model / multi-model
- Data cleaning.
- Data update captures.

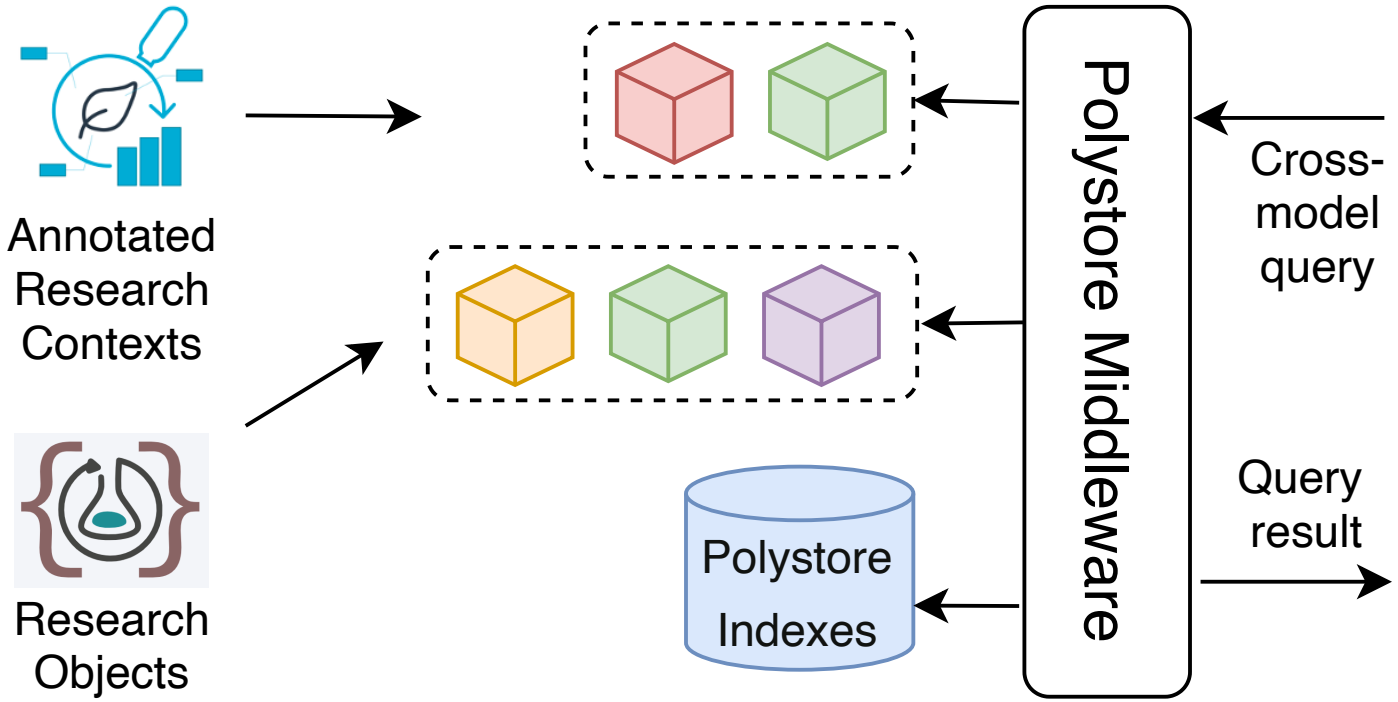
Ingestion & Exploration

- Bulk loading of single/multi-model data.
- Data Indexing.
- Query processing / exploration.

The role of Docker containers in in-situ query processing

- Docker helps query the data directly from individual ARCs.
- Helps develop an on-demand data exploration platform.
- Easy to integrate ARCs with Distributed Query Processing frameworks.
- No need to maintain huge volumes of Research Data in physical database systems.

Should we re-design the wheel?



1. SQL Query Interface
2. Custom Optimiser
3. Custom Adaptive Indexing
4. Designed for Cross-omics

Questions

Data PLANT

Distributed Query Processing Framework – Hands-on

Movies → Relational

Cast & Crew → Document

Apache Drill System Docker Setup

- Clone the repo **gdoniparthi/summerschool** from GitHUB
- Docker pre-requisites (Docker desktop, Docker compose)
- Bootstrap the docker network (*docker compose up –d*).
- Setup PostgreSQL and MongoDB with Apache Drill.
- Execute test queries.

TMDB Dataset Description

Table "public.mo

Column	Type
budget	double precision
genres	json
homepage	character varying
id	integer
keywords	json
original_language	character varying
original_title	character varying
overview	character varying
popularity	character varying
production_companies	json
production_countries	json
release_date	character varying
revenue	double precision
runtime	double precision
spoken_languages	json
status	character varying
tagline	character varying
title	character varying
vote_average	double precision
vote_count	integer

Table "publ

Column	Type
genre	character varying
genre_id	integer
movie_id	integer

Table "publ

Column	Type
keyword	character varying
keyword_id	integer
movie_id	integer

```
movies> db.casting.findOne()
{
  _id: ObjectId("64b6e584c887f773faf3eddb"),
  cast_id: 4,
  character_name: 'Col. Quaritch',
  gender: 2,
  name: 'Stephen Lang',
  order_id: 3,
  movie_id: 19995
}
```

```
movies> db.crew.findOne()
{
  _id: ObjectId("64b6e58c22f1b9739fe169aa"),
  department: 'Editing',
  crew_id: 1721,
  name: 'Stephen E. Rivkin',
  job: 'Editor',
  movie_id: 19995
}
```