BREAKING THE BOUNDARIES OF CURRENT METADATA

a perspective from VZG

Jakob Voß

2022-09-06

OUTLINE

1. Current trends in metadata
2. Boundaries of metadata in CBS
3. Breaking the boundaries

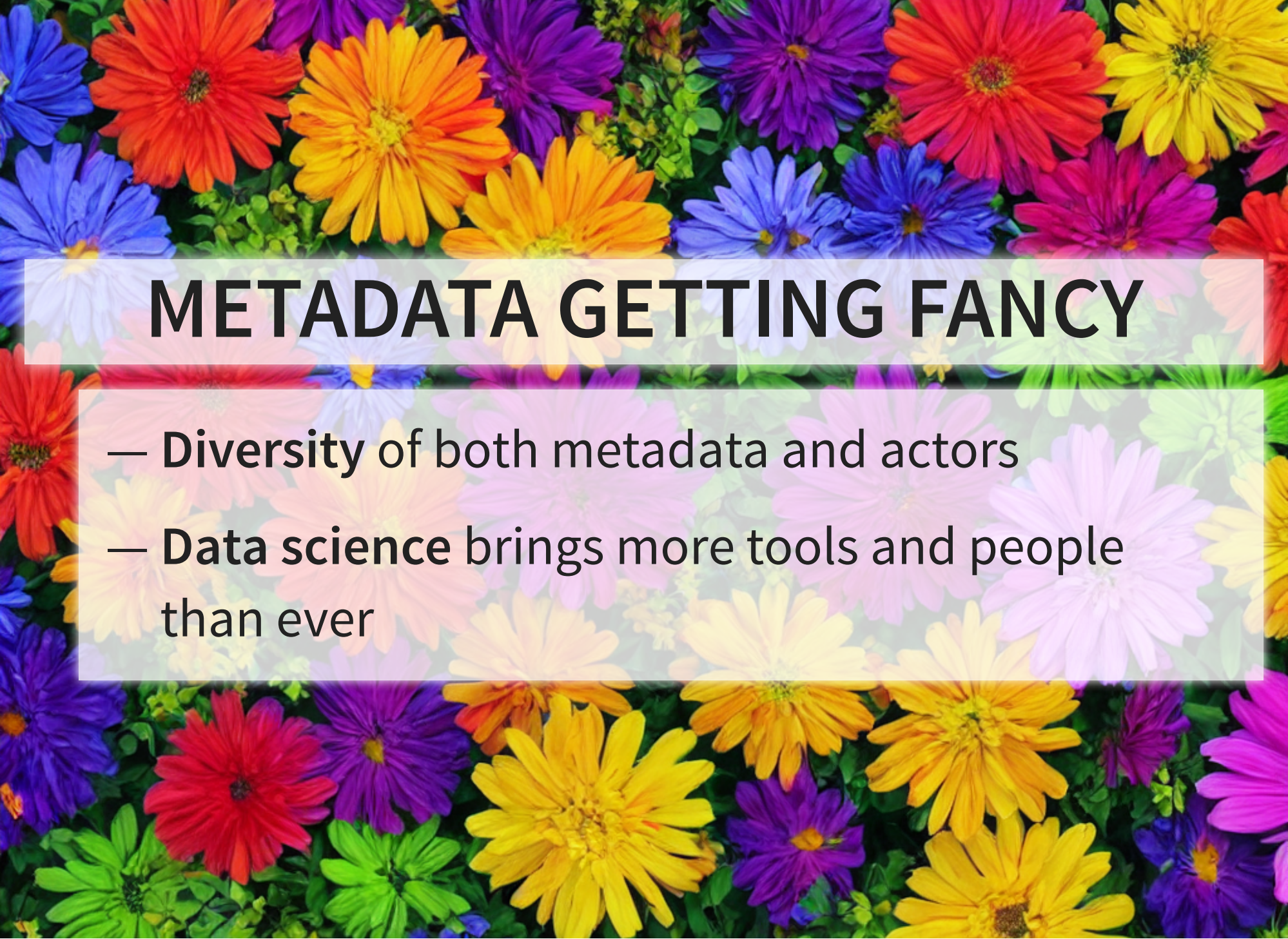
CURRENT TRENDS IN METADATA

OBVIOUS TRENDS

— **Cloud infrastructure**: distributed servers

— **Big data**: continuing growth

— **Machine learning & AI**: not today


METADATA GETTING FANCY
— **Diversity** of both metadata and actors
— **Data science** brings more tools and people than ever

DECENTRALIZATION

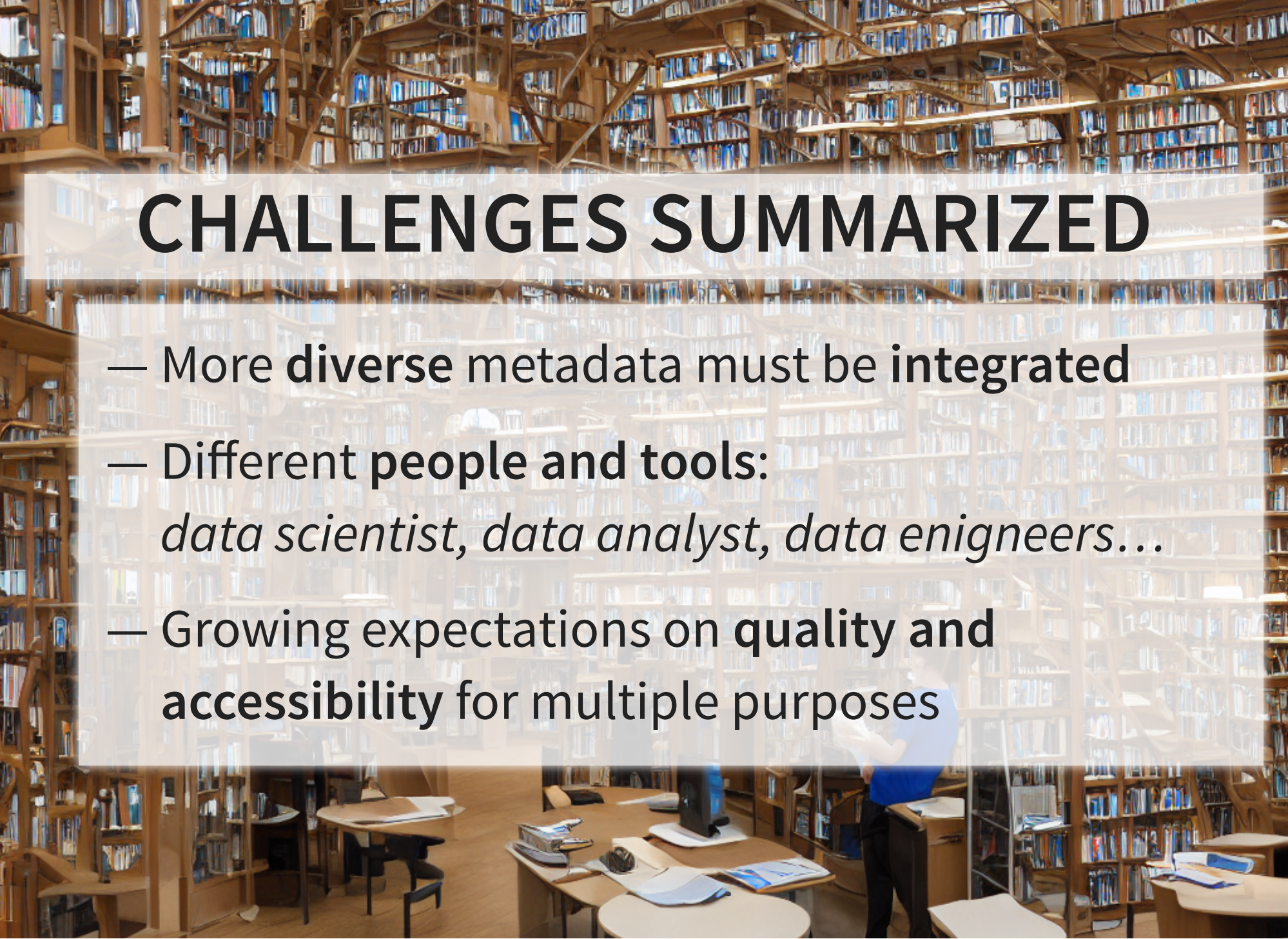
— **Integration** across multiple systems

— **Aggregation** in data warehouses, data lakes…
for analysis

METADATA MANAGEMENT

— **Governance** by quality and observation

— **Active metadata** e.g. logging, alerts…



CHALLENGES SUMMARIZED

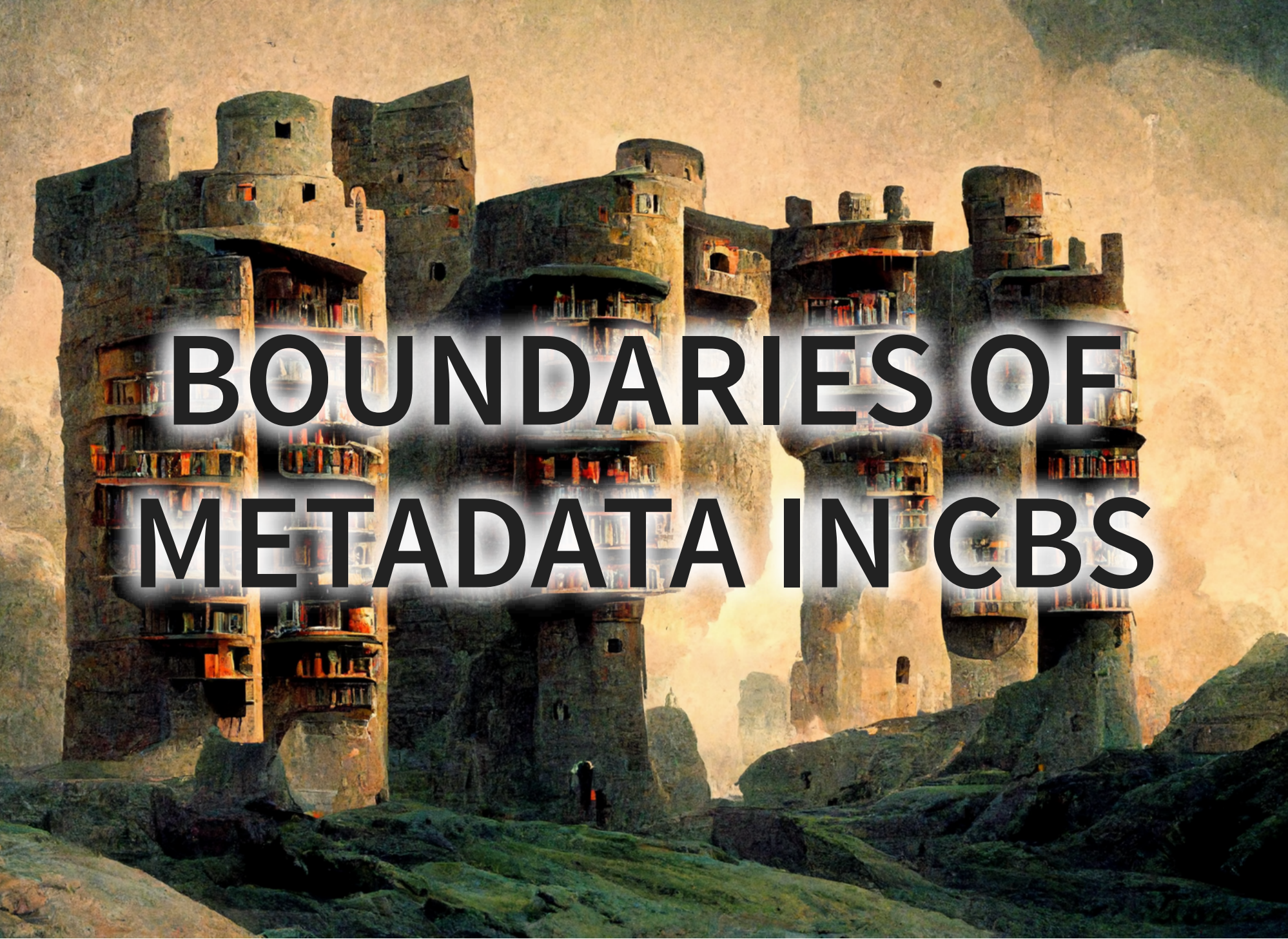
— More **diverse** metadata must be **integrated**

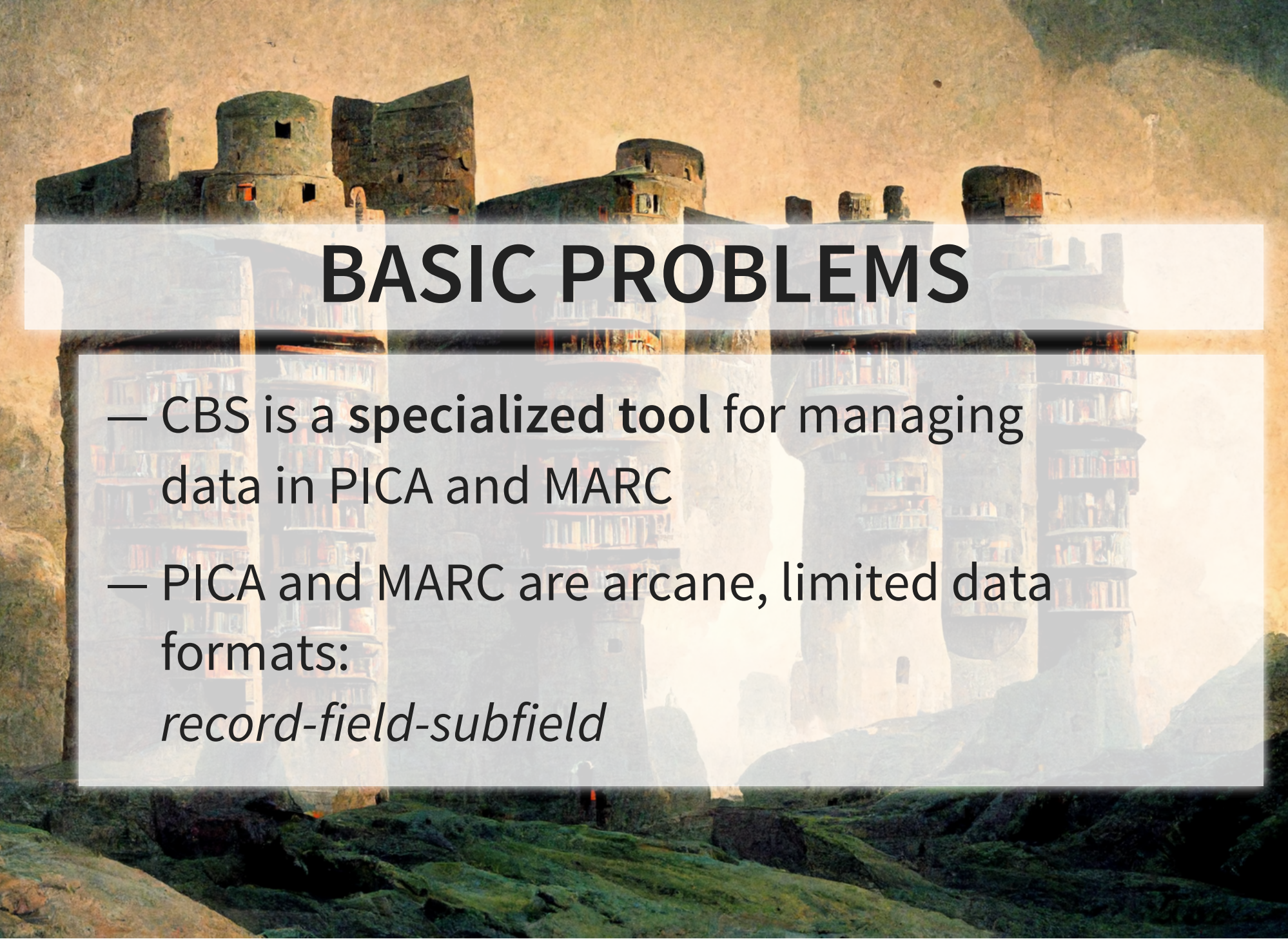
— Different **people and tools**:
data scientist, data analyst, data enigneers…

— Growing expectations on **quality and accessibility** for multiple purposes

EXAMPLE

Number of books
by publisher *X*
in subject area *Y*
held by each library


BOUNDARIES OF
METADATA IN CBS



BASIC PROBLEMS

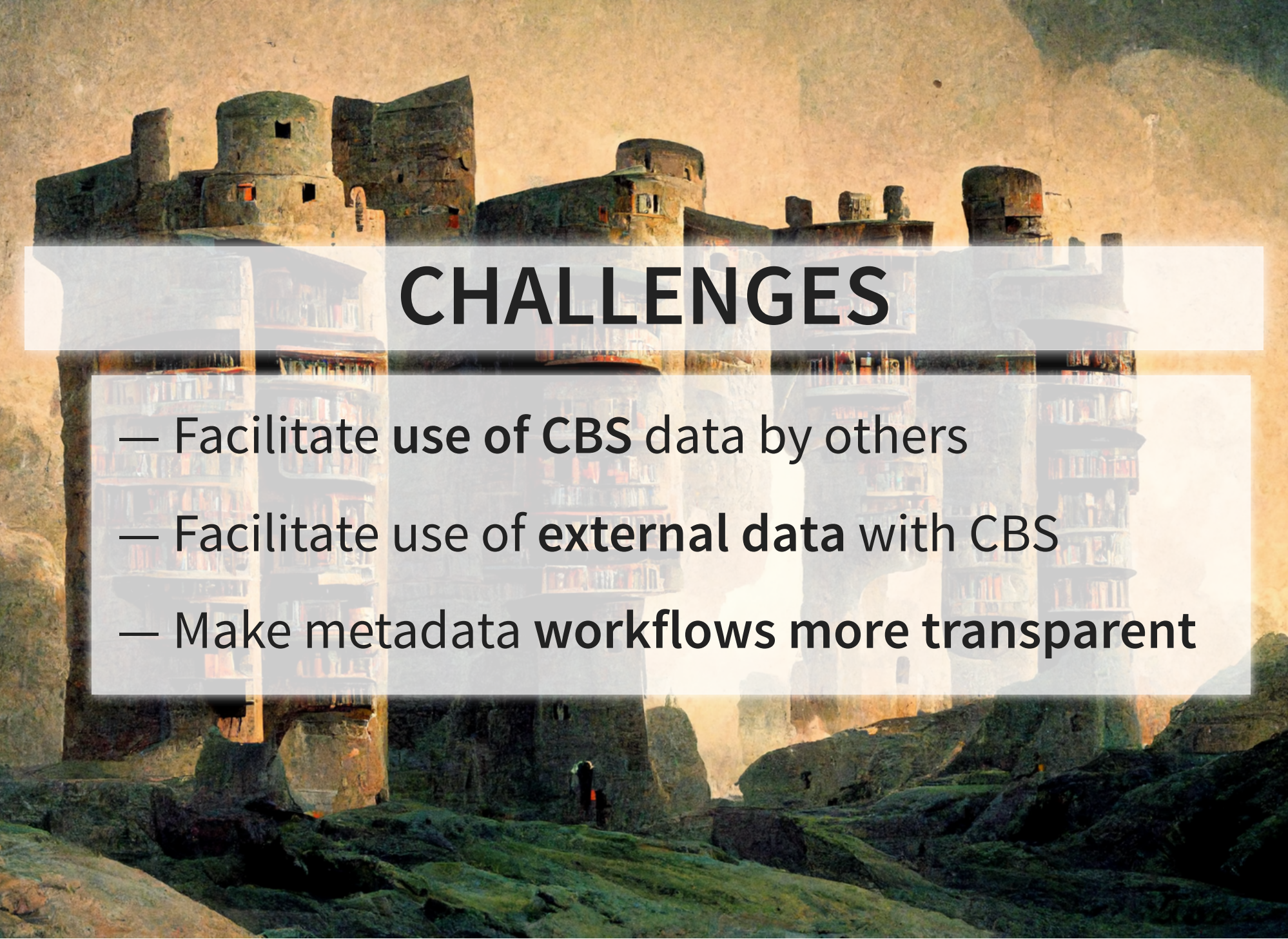
— CBS is a **specialized tool** for managing data in PICA and MARC

— PICA and MARC are arcane, limited data formats:
record-field-subfield

LIMITATIONS

— **Number of people** doing data processing in CBS/PICA/MARC…

— Little accessible **standards and tools**

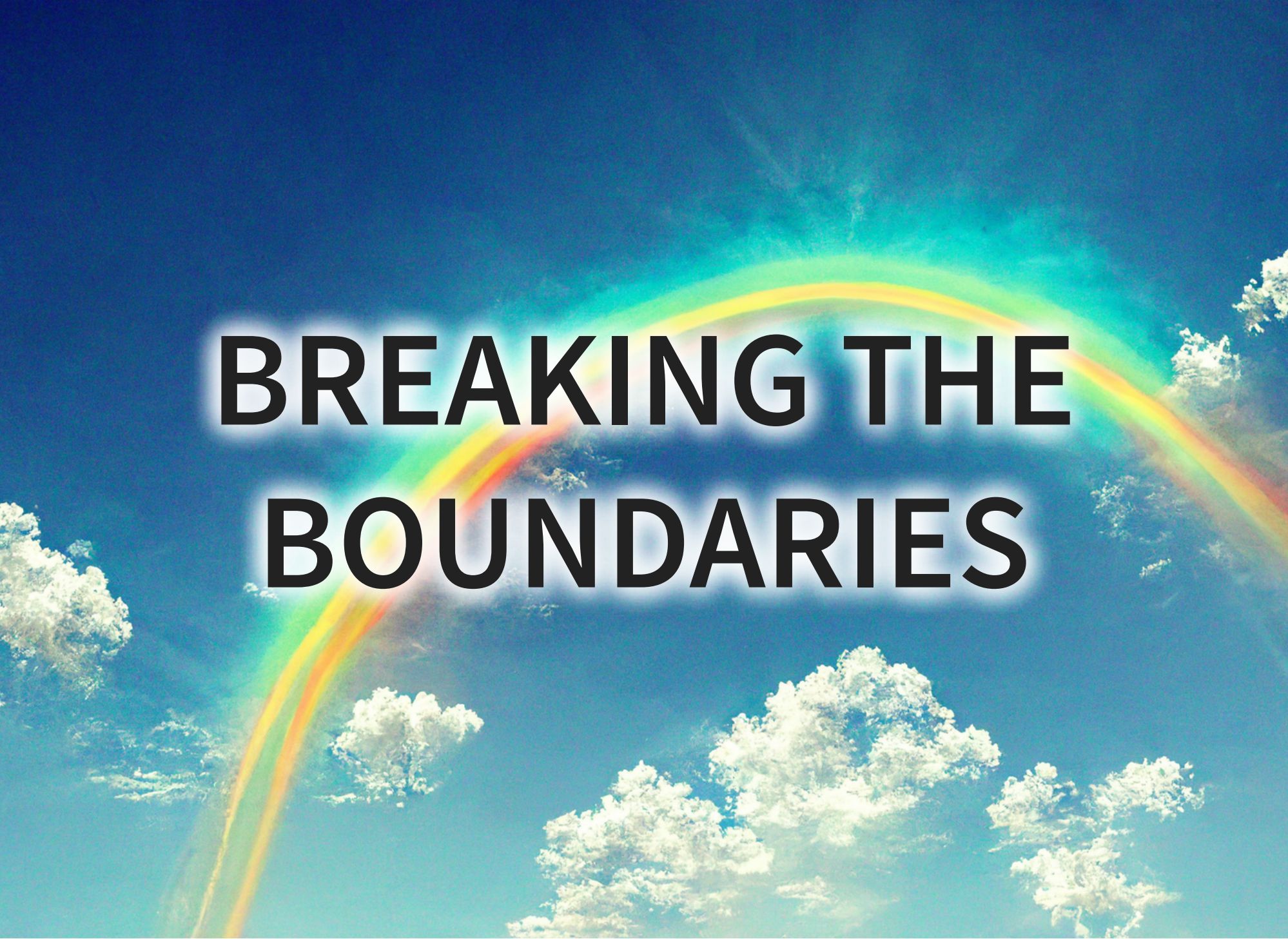


CHALLENGES

— Facilitate **use of CBS** data by others

— Facilitate use of **external data** with CBS

— Make metadata **workflows more transparent**


BREAKING THE
BOUNDARIES

CONFESSION

— It’s complicated

— Two independent strategies

1. STANDARDIZATION

— **Avram Schemas** for MARC- and PICA-formats

— **PICA Patch format** formalizes changes records

— **APIs and tools** that can be used by anyone

ADVANTAGES

— Based on **common web standards** (JSON…)

— **Accessible** by more people with diverse needs

— Not *how* (*take field X, filter by condition Y…*)
but ***what*** (*records with specific condition…*)

2. KNOWLEDGE GRAPHS

- Records in CBS
  - Hierachical Record model (record level)
  - Links between records via $9 and PPN
  - Links via identifiers (DOI, ISSN…)
- External data
  - Author affiliation, addresses, names…

KNOWLEDGE GRAPH

— Create **CBS Knowledge Graph**

— **Expose** via RDF or Graph Database

— **Integrate** with external Linked Data

EXAMPLE

Number of books
by publisher *X*
in subject area *Y*
held by each library

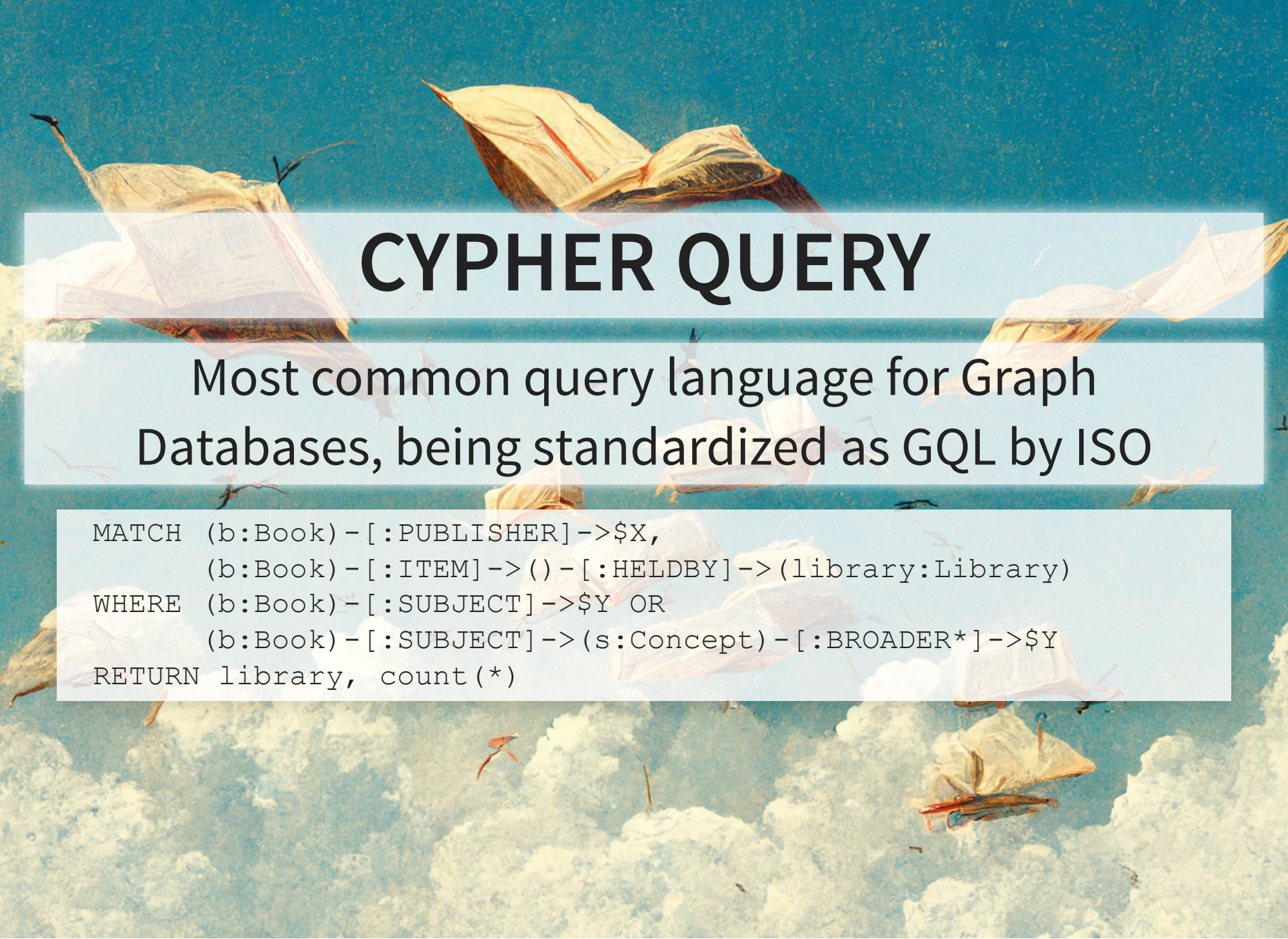
IMPERATIVE SCRIPT

1. Build index of transitive sub-subjects of Y
2. Get books by publisher *X*
   `pica filter "033.n=="$"` (pica-rs)
3. Reduce to books with subject in index
4. Count libraries with holding of the book

SPARQL QUERY

Established query language for RDF

```
SELECT ?library (COUNT(?book) as ?number) WHERE {
  ?book dct:publisher <$X> .
  ?book dct:subject/skos:broader* <$Y> .
  ?bibframe bibframe:hasItem [ bibframe:heldBy ?library ] .
} GROUP BY ?library
```



CYPHER QUERY

Most common query language for Graph Databases, being standardized as GQL by ISO

```
MATCH  (b:Book)-[:PUBLISHER]->$X,
       (b:Book)-[:ITEM]->()-[:HELDBY]->(library:Library)
WHERE  (b:Book)-[:SUBJECT]->$Y OR
       (b:Book)-[:SUBJECT]->(s:Concept)-[:BROADER*]->$Y
RETURN library, count(*)
```

TAKEAWAYS

— Standardization of data languages to process PICA & MARC
  - **Avram Schema** format
  - **PICA Diff** format

— From record-field-subfield to **knowledge graphs**

REFERENCES

- https://format.gbv.de/schema/avram/specificatio
- https://format.gbv.de/pica/patch
- https://deutsche-nationalbibliothek.github.io/pica-rs/
- Background images AI created with stable diffusion and midjourney