Database preservation toolkit:

a flexible tool to normalize and give access to databases

DLM Forum: “Making the Information Governance Landscape in Europe”

José Carlos Ramalho jcr@keep.pt
KEEP SOLUTIONS // University of Minho

Nov. 12-14, 2014, Lisbon, Portugal

KEEP SOLUTIONS: Projects

- DigitArq, CRAV (2003..[2008-2012])
- RODA (2006..[2008-…[)
- RCAAP (2008-…)
- PPA (2009)
- Open source: RODA, KOHA, DSpace, Moodle, etc.
- Scientific research (european projects)
  - **SCAPE**: Large scale preservation
  - **4C**: the cost for curation
  - **e-arK**: presented by Kuldar

http://www.keep.pt

This work was partially supported by the SCAPE Project.
The SCAPE project is co-funded by the European Union under FP7 ICT-2009.4.1 (Grant Agreement number 270137).

Database Preservation Toolkit

Developed within RODA project

Now stand-alone open-source project

http://keeps.github.io/db-preservation-toolkit/

Imports and exports between DBMS and DB formats

Supports preservation formats: DBML, SIARD

The Past: 2006 - 2009

RODA:
Repositório de Objectos Digitais Autênticos

José Carlos Ramalho
jcr@di.uminho.pt

Miguel Ferreira
mferreira@dsi.uminho.pt

Rui Castro
Rcastro@iantt.pt

Luis Faria
lfaria@iantt.pt

Francisco Barbedo
frbarbedo@iantt.pt

Cecília Henriques
chenriques@iantt.pt

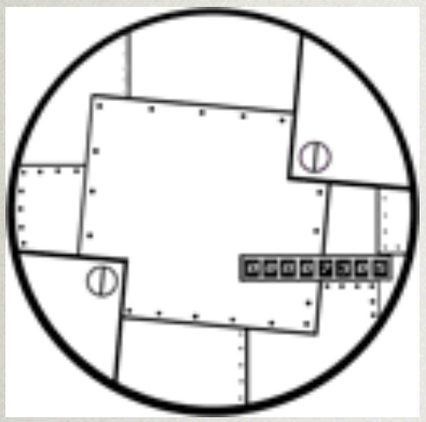
Luis Corujo
lcorujo@iantt.pt

What is RODA?

A digital repository created for archives, with the following features:

- Long term preservation and Autenticity
- Standards based (OAIS, EAD, PREMIS, METS, etc.)
- Following TRAC demands (Trustworthy Repositories Audit & Certification)
- Security policies
- Service Oriented Architecture (SOA)
- Nice and simple design
- Open source
- DGArq and University of Minho

Context



RODA (2006-2009)

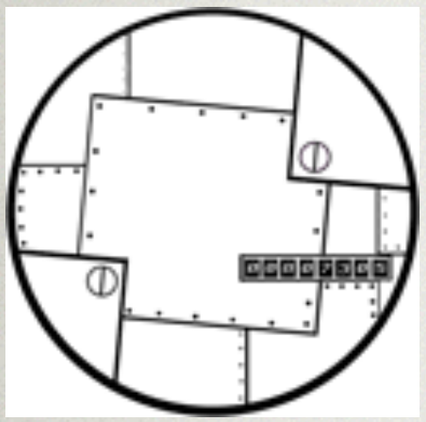
- Metadata management (EAD)
- Digital Object management (...)
- Digital Preservation Policies and protocols
- National Project (Archives National Board)

CRiB
Conversion and Recommendation of Digital Object Formats

CRiB: Preservation Services Digital Repositories (2005-2008)

- Distributed Migration Service
- Migration Service supported by knowledge base
- phd Thesis / U. Minho (Miguel Ferreira)

Context



RODA (2006-2009)

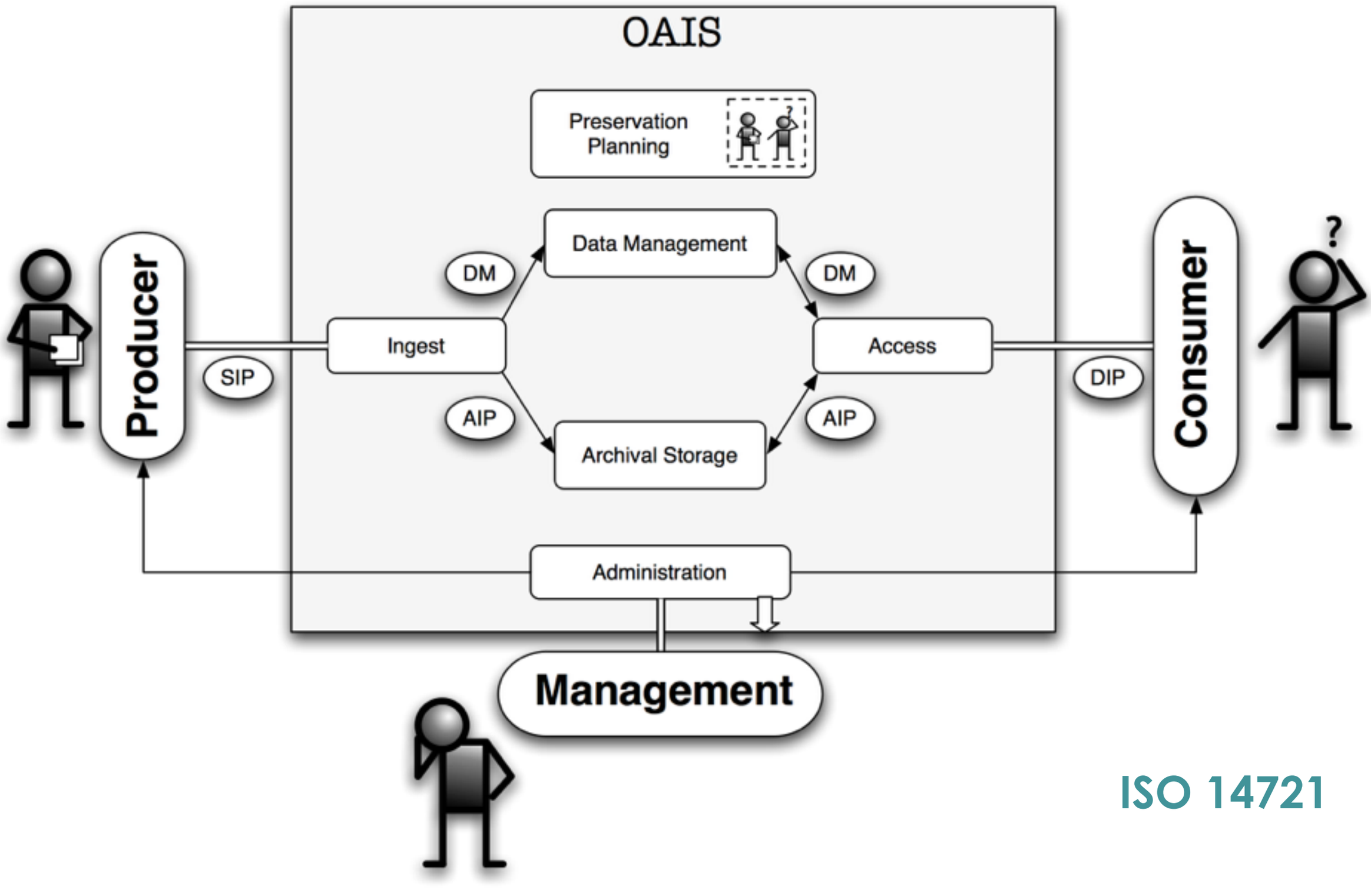
- Metadata management (EAD)
- Digital Object management (...)
- Digital Preservation Policies and protocols
- National Project (Archives National Board)

CRiB
Conversion and Recommendation of Digital Object Formats

CRiB: Preservation Services Digital Repositories (2005-2008)

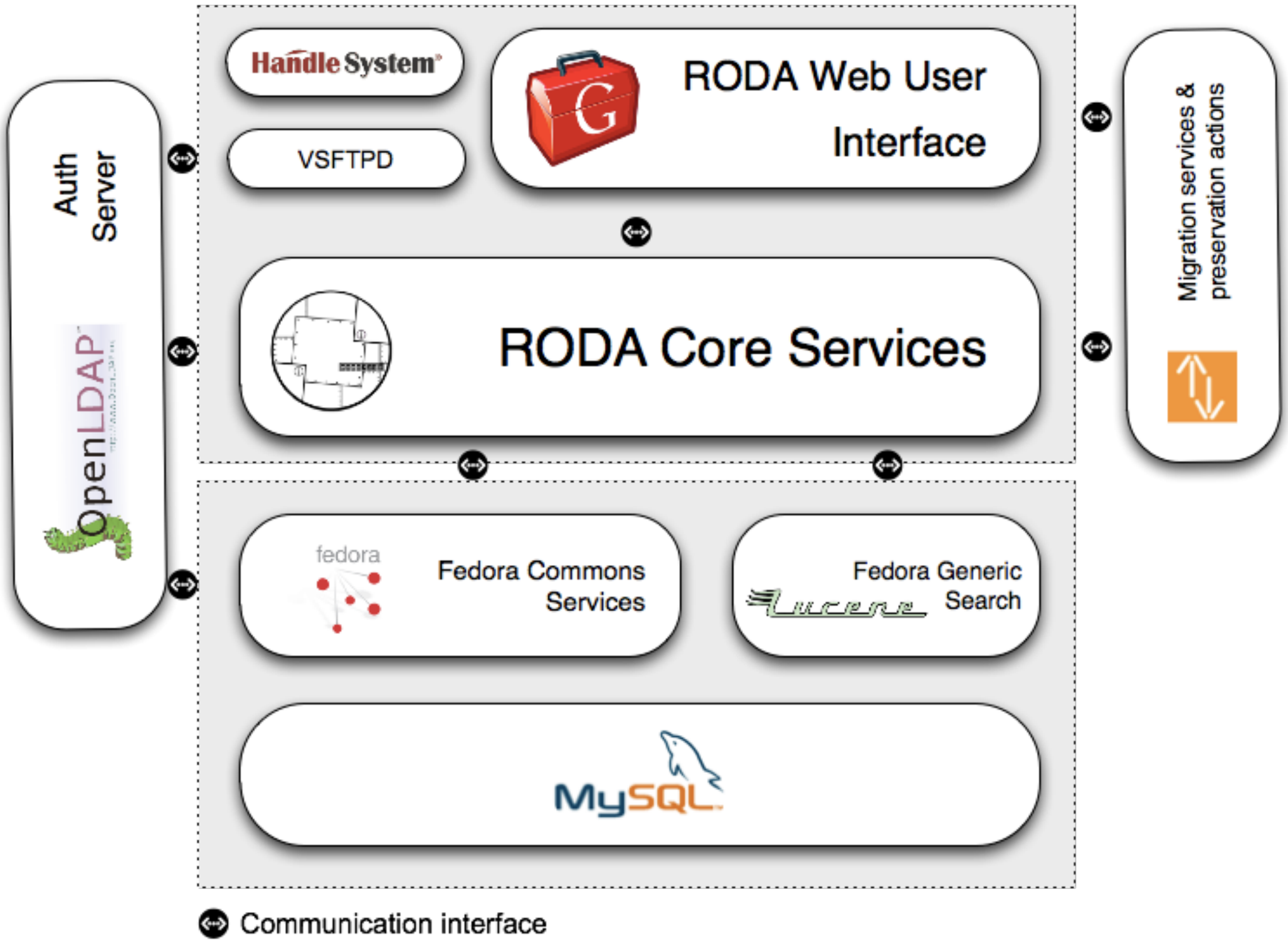
- Distributed Migration Service
- Migration Service supported by knowledge base
- phd Thesis / U. Minho (Miguel Ferreira)

OAIS

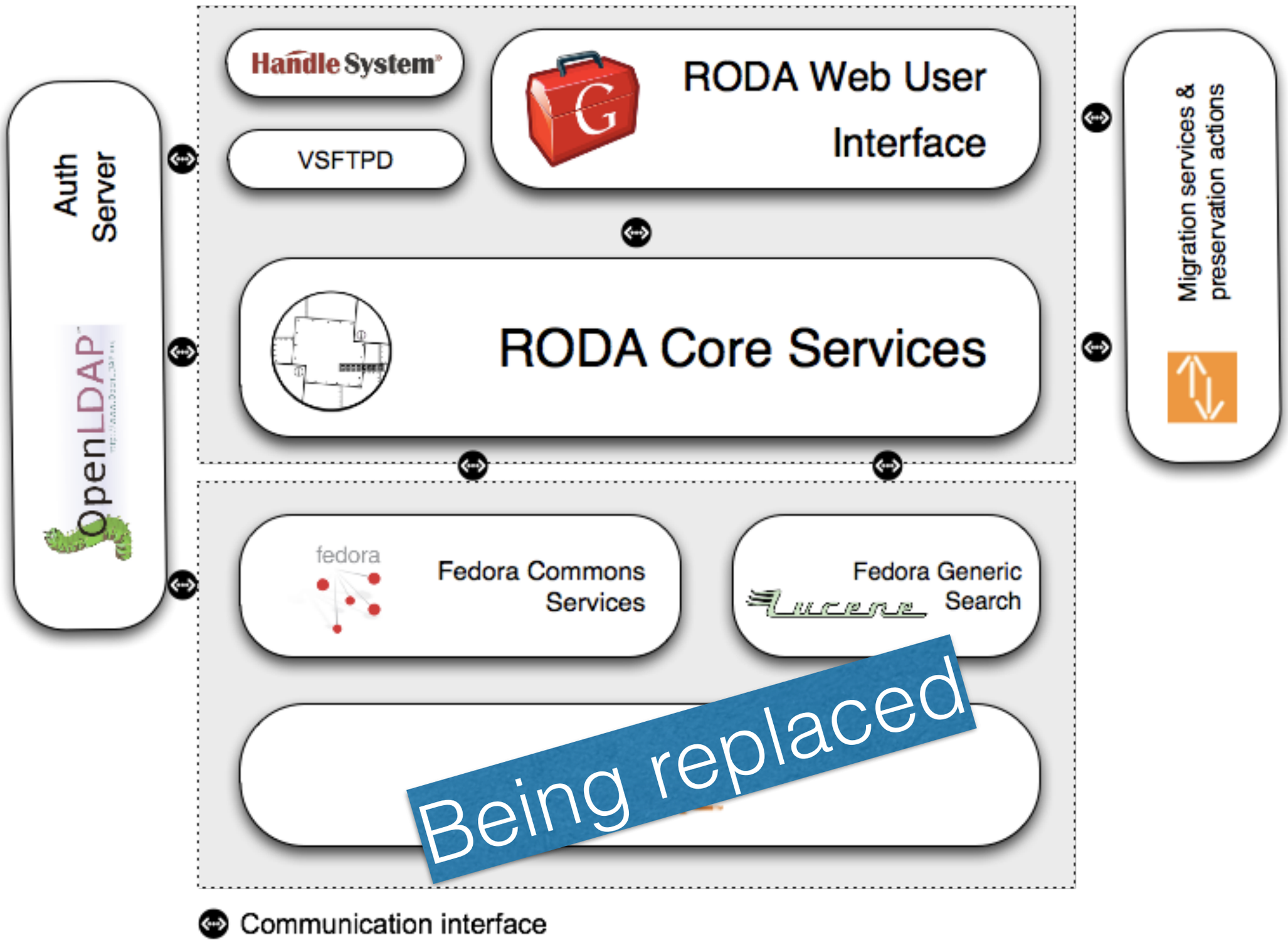


ISO 14721

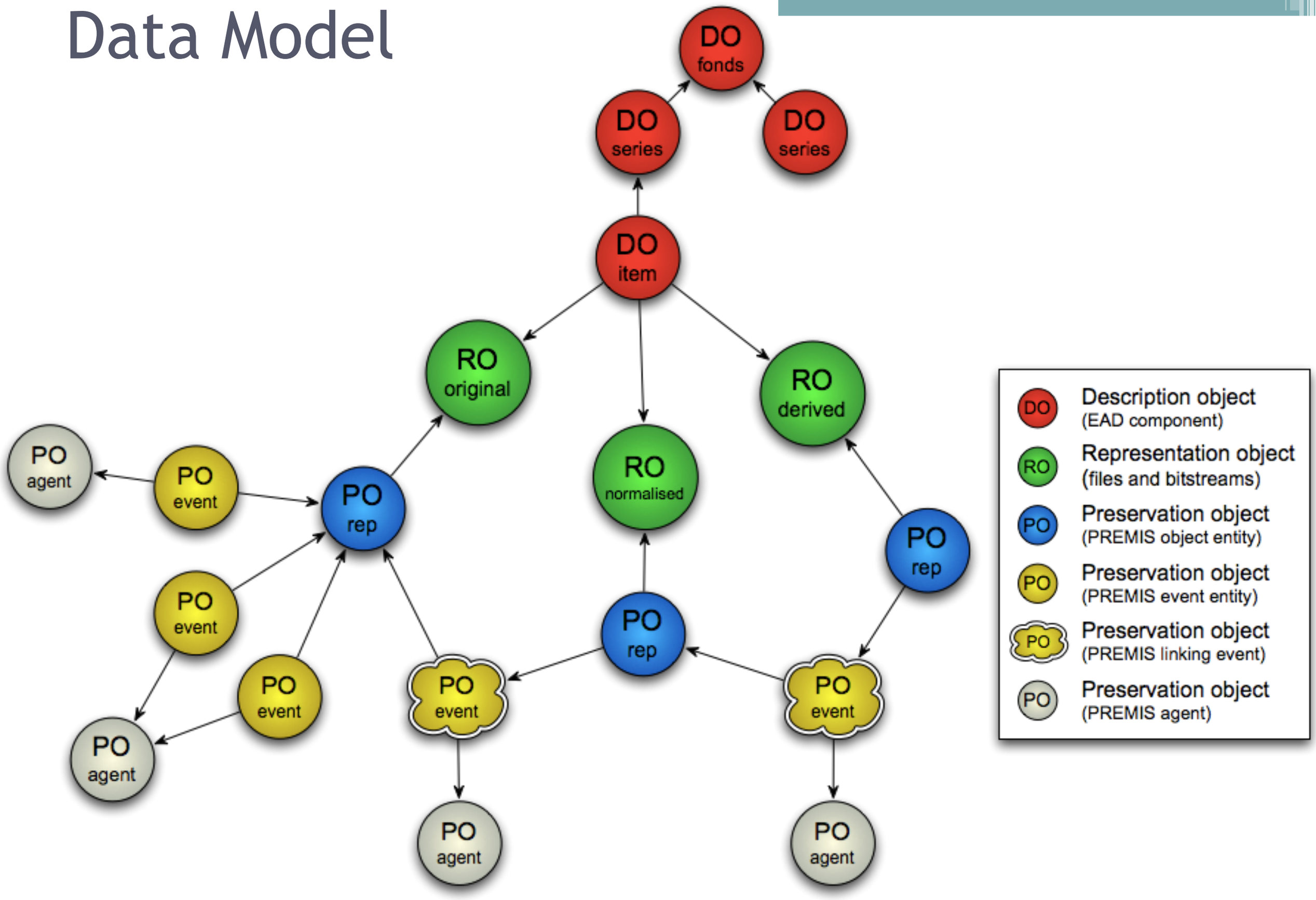
Architecture



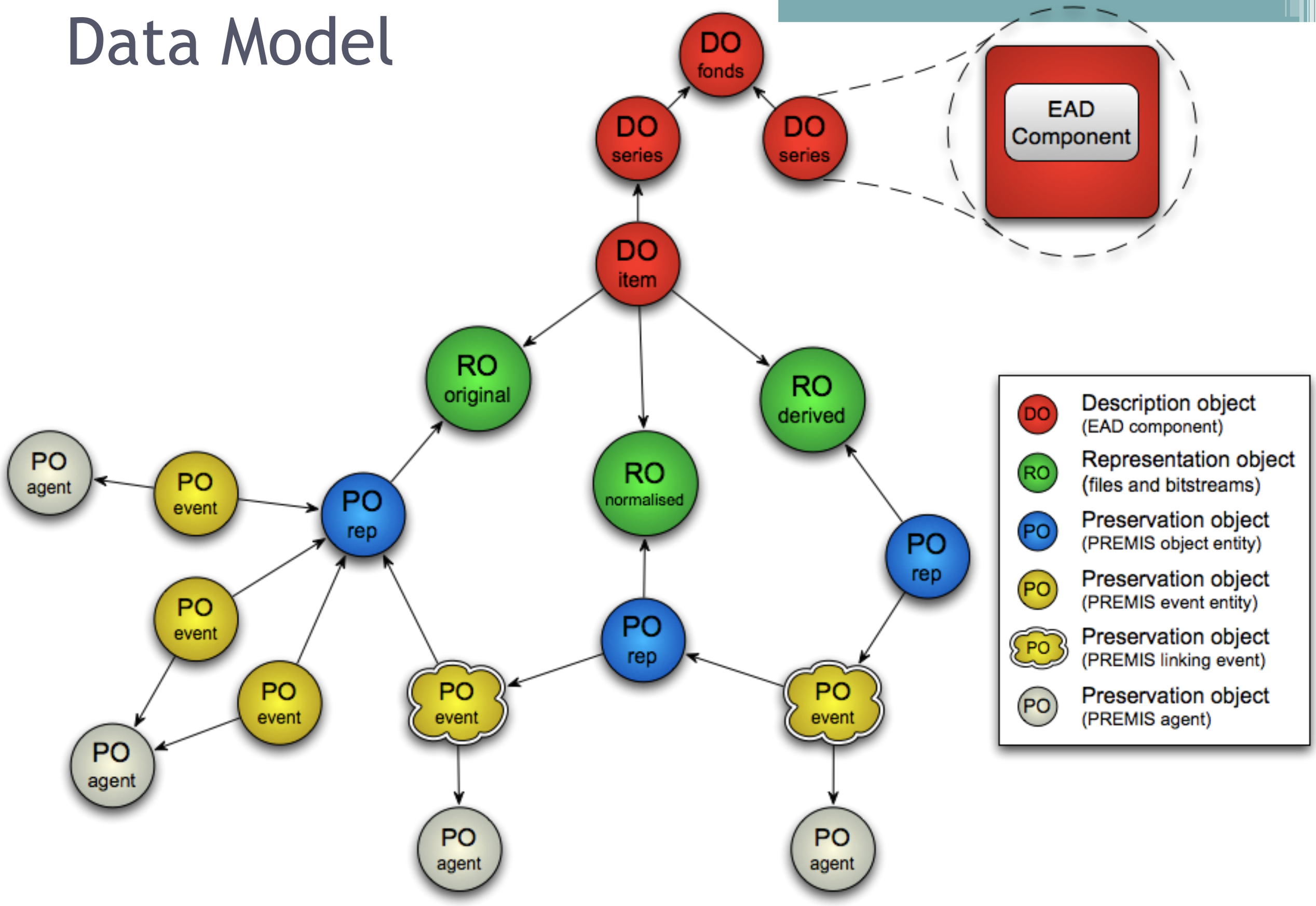
Architecture



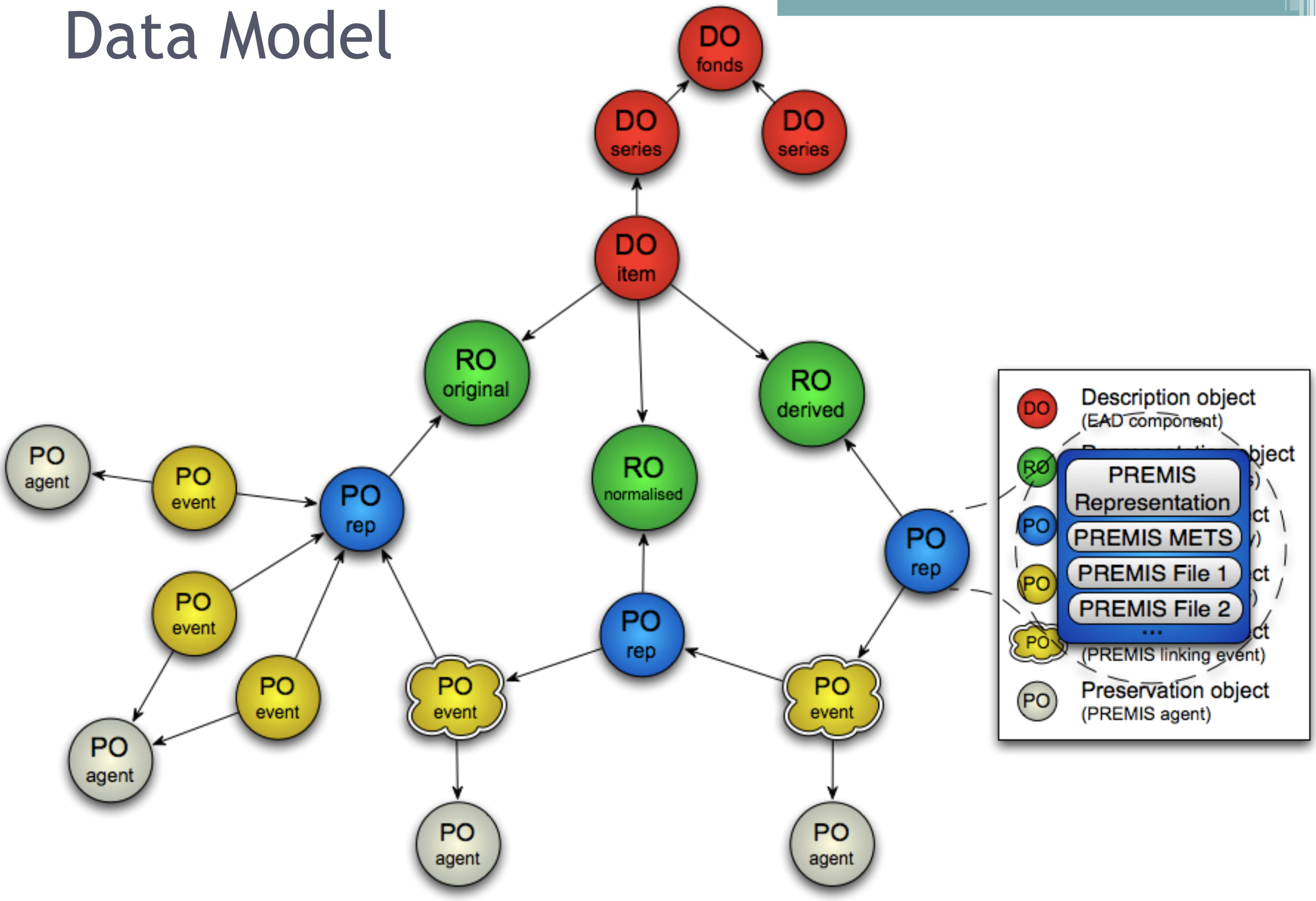
Data Model


DO fonds
DO series
DO series
DO item
RO original
RO derived
RO normalised
PO agent
PO event
PO rep
PO event
PO event
PO rep
PO rep
PO event
PO event
PO agent
PO agent
PO agent
Description object (EAD component)
Representation object (files and bitstreams)
Preservation object (PREMIS object entity)
Preservation object (PREMIS event entity)
Preservation object (PREMIS linking event)
Preservation object (PREMIS agent)

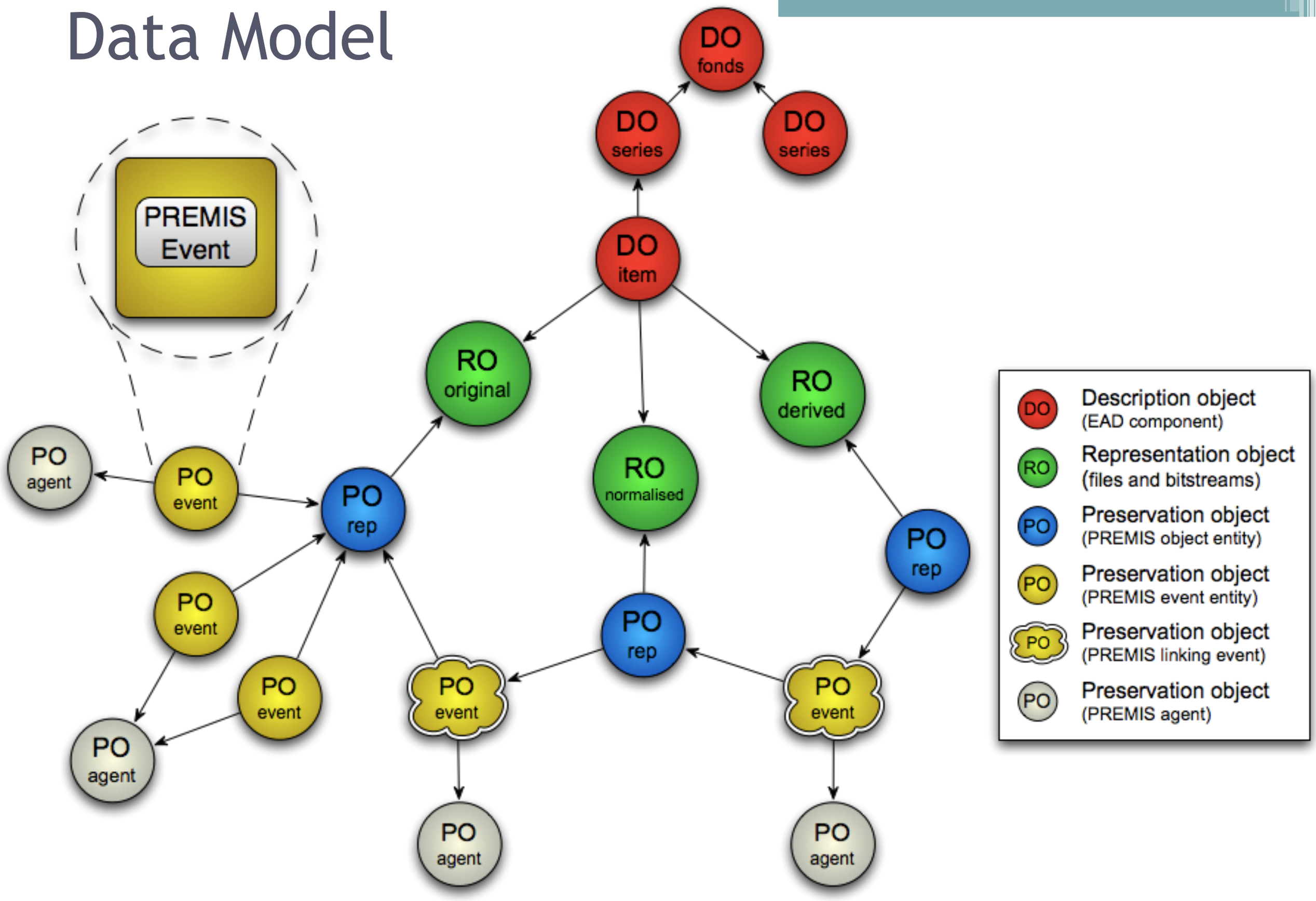
Data Model


DO
fonds
DO
series
DO
series
EAD
Component
DO
item
RO
original
RO
derived
RO
normalised
PO
agent
PO
event
PO
rep
PO
event
PO
rep
PO
event
PO
agent
PO
event
PO
rep
PO
event
PO
agent
PO
agent
Description object
(EAD component)
Representation object
(files and bitstreams)
Preservation object
(PREMIS object entity)
Preservation object
(PREMIS event entity)
Preservation object
(PREMIS linking event)
Preservation object
(PREMIS agent)

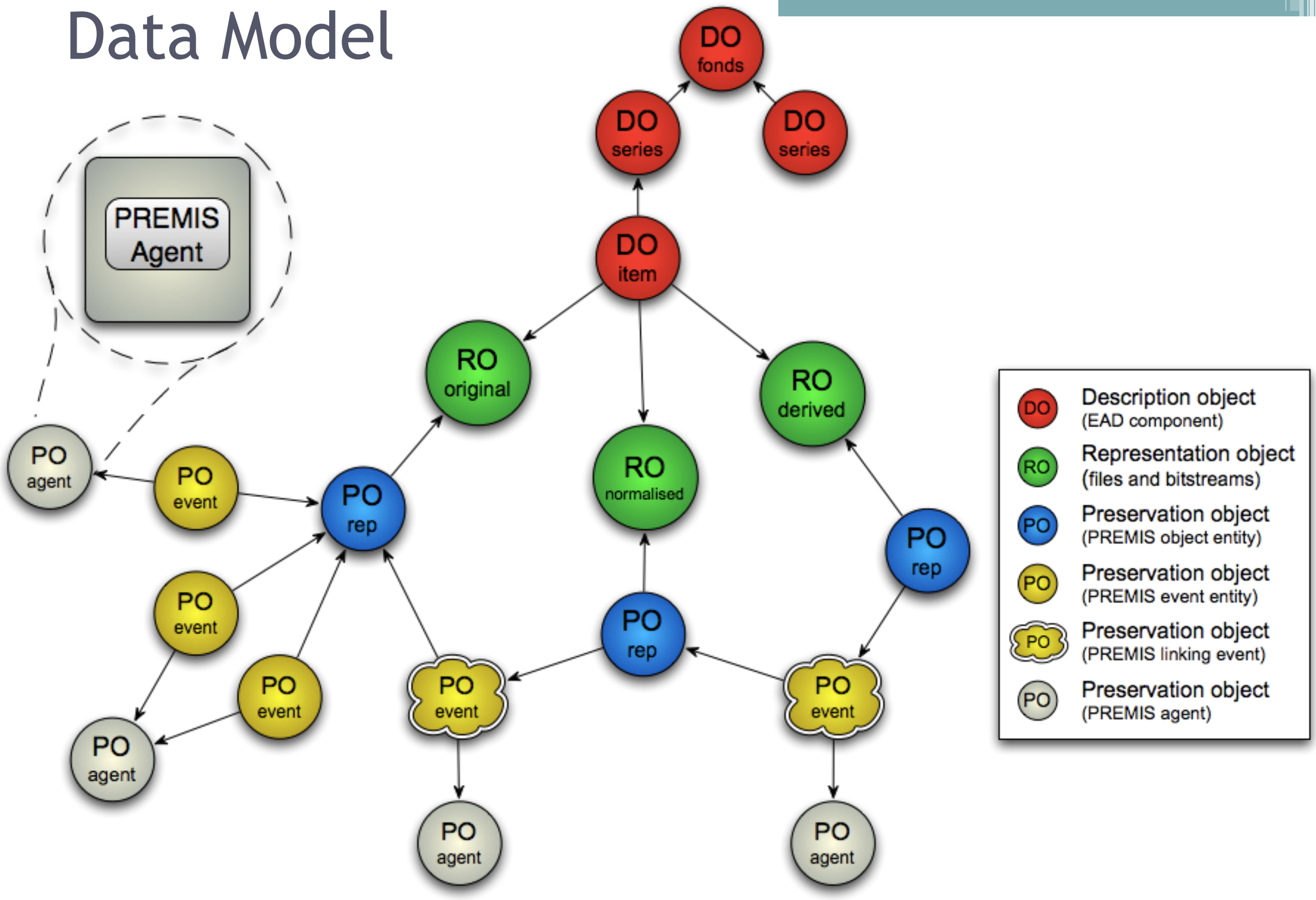
Data Model


DO
fonds
DO
series
DO
series
DO
item
RO
original
RO
normalised
RO
derived
PO
agent
PO
event
PO
rep
PO
event
PO
event
PO
rep
PO
rep
PO
agent
PO
event
PO
event
PO
agent
PO
agent
Description object
(EAD component)
PREMIS
Representation
PREMIS METS
PREMIS File 1
PREMIS File 2
...
(PREMIS linking event)
Preservation object
(PREMIS agent)

Data Model


PREMIS Event
DO fonds
DO series
DO series
DO item
RO original
RO derived
RO normalised
PO agent
PO event
PO rep
PO event
PO event
PO rep
PO rep
PO event
PO event
PO agent
PO agent
PO agent
Description object (EAD component)
Representation object (files and bitstreams)
Preservation object (PREMIS object entity)
Preservation object (PREMIS event entity)
Preservation object (PREMIS linking event)
Preservation object (PREMIS agent)

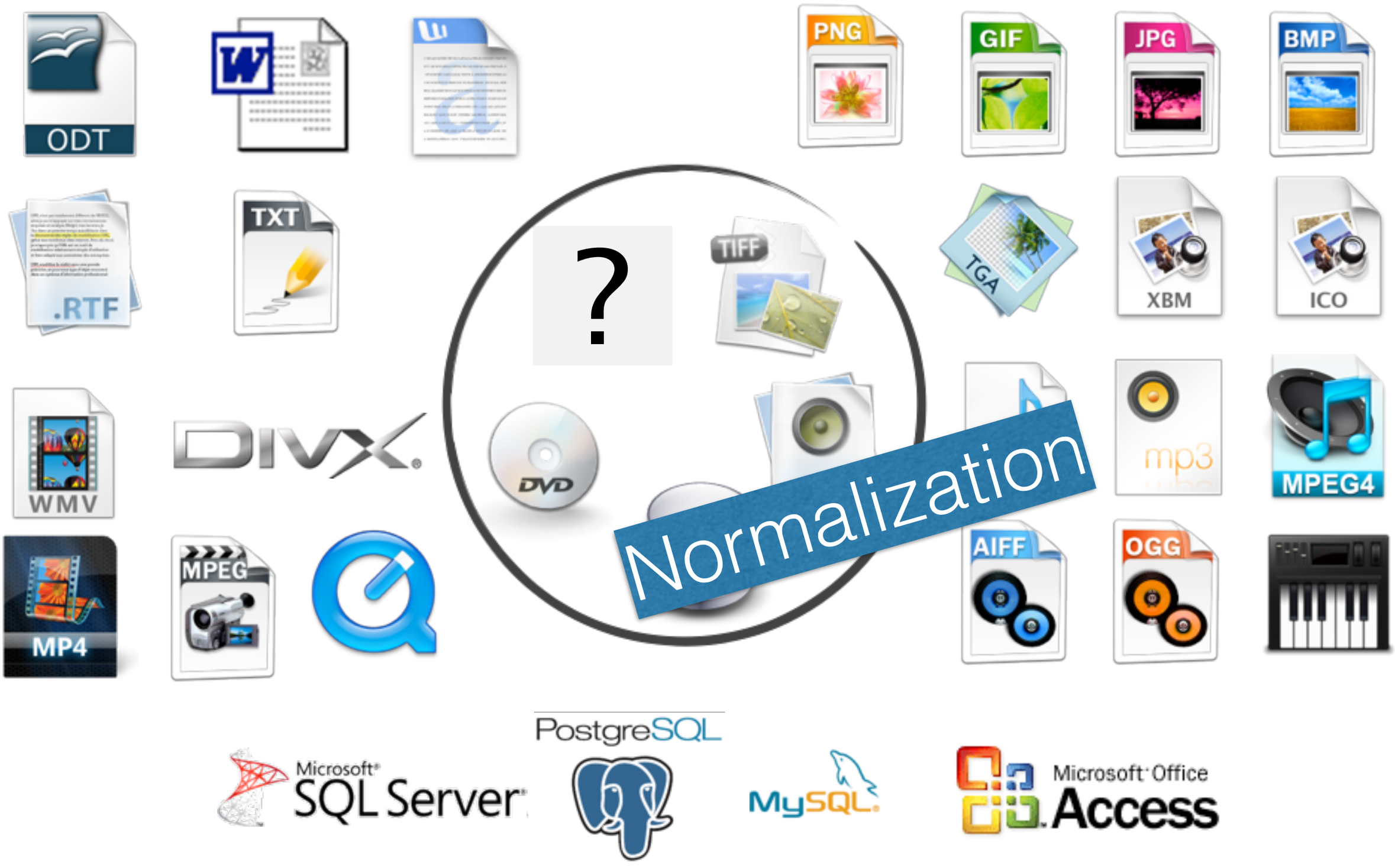
Data Model


PREMIS Agent
DO fonds
DO series
DO series
DO item
RO original
RO derived
RO normalised
PO agent
PO event
PO rep
PO event
PO rep
PO event
PO rep
PO event
PO agent
PO event
PO agent
PO agent
Description object (EAD component)
Representation object (files and bitstreams)
Preservation object (PREMIS object entity)
Preservation object (PREMIS event entity)
Preservation object (PREMIS linking event)
Preservation object (PREMIS agent)

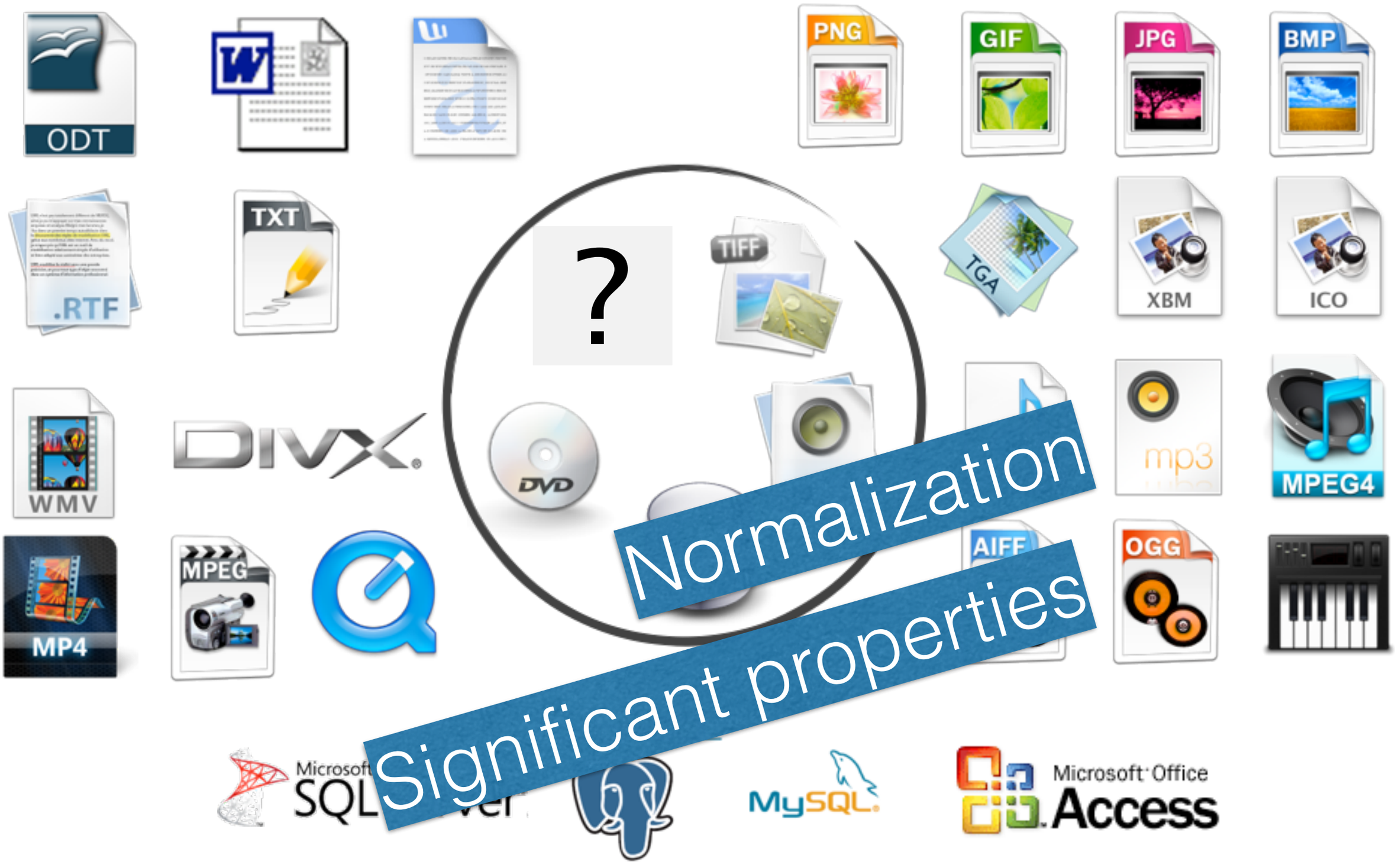
Digital Object Classes

Digital Object Classes

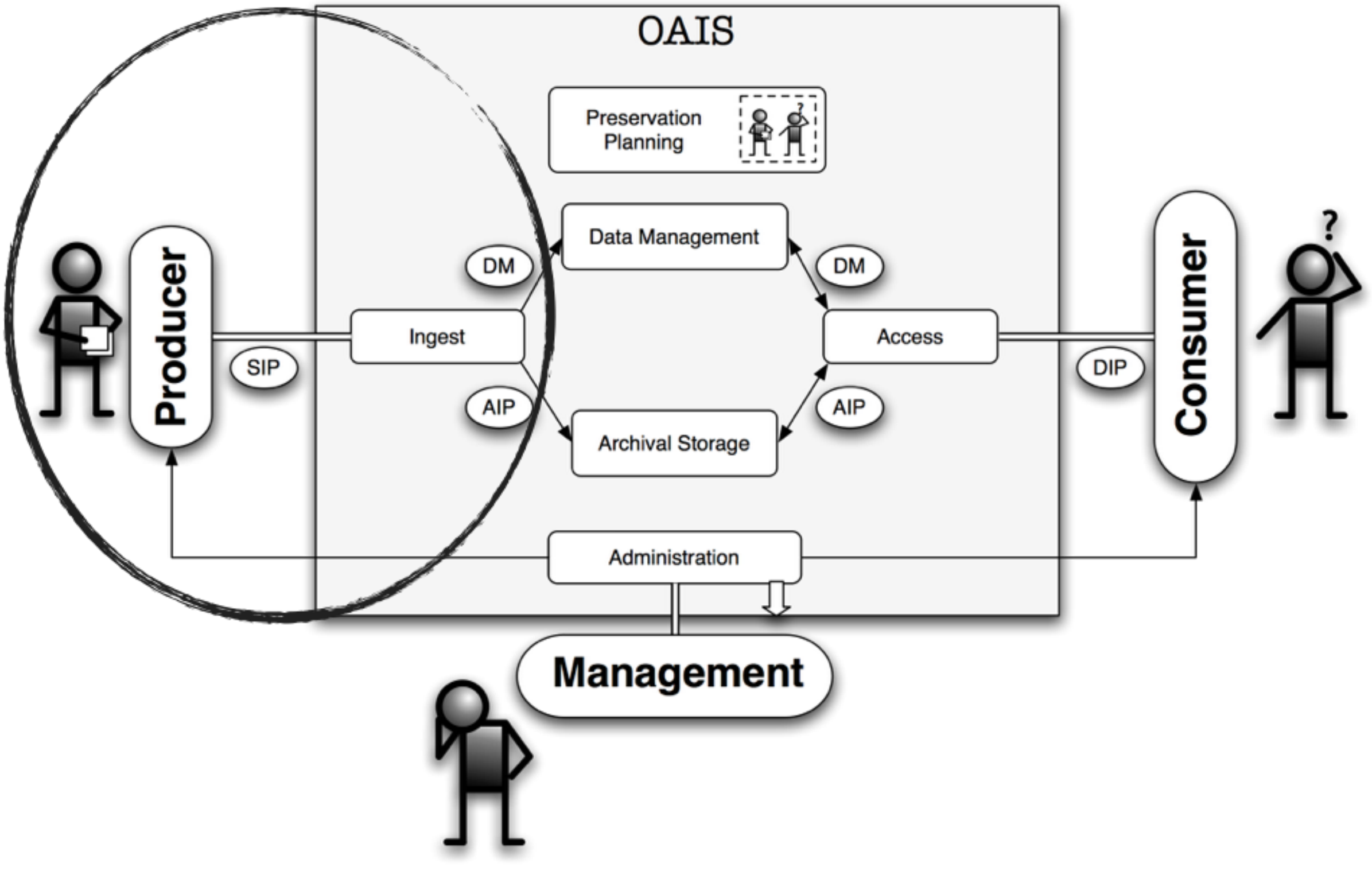
Digital Object Classes



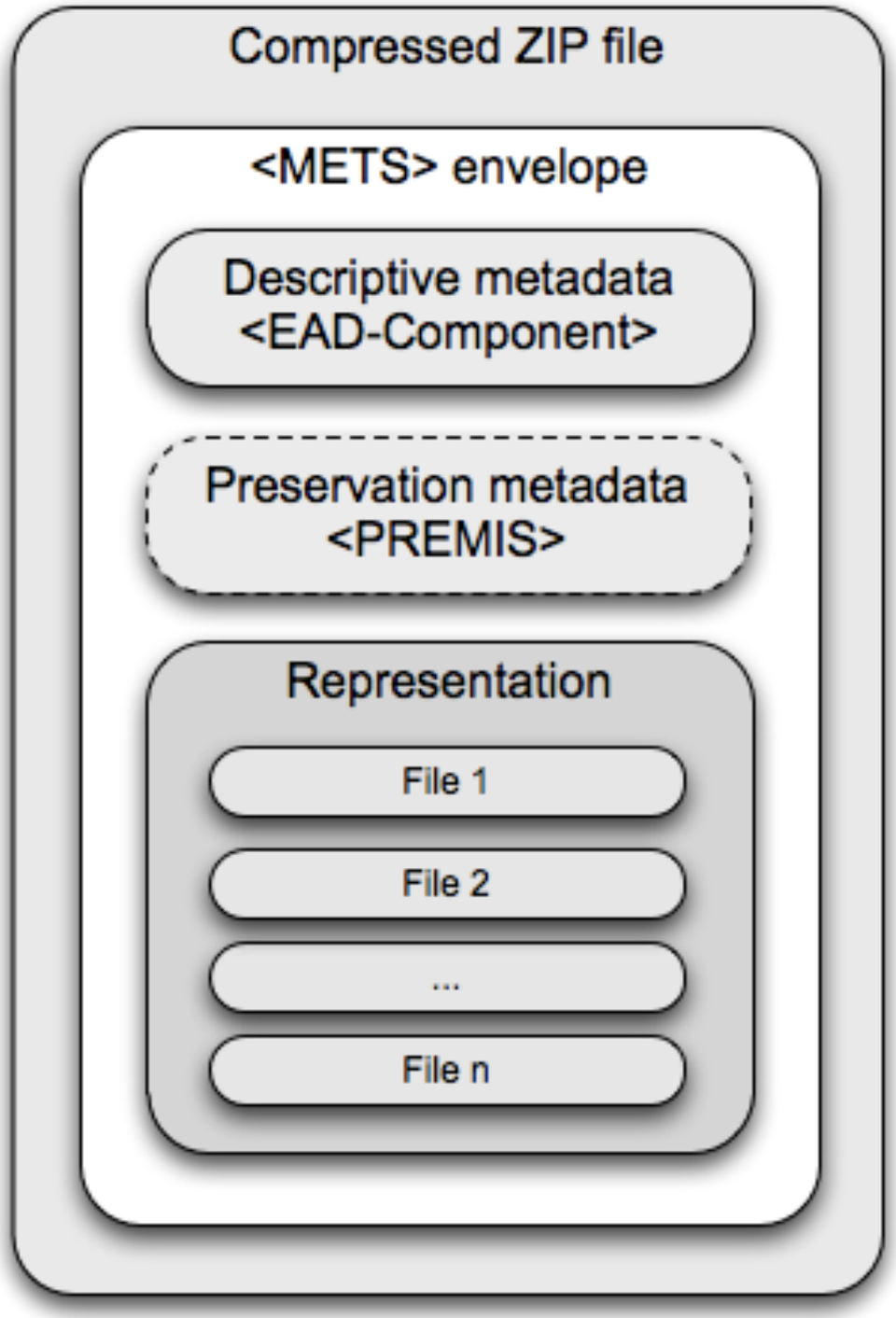
Digital Object Classes



Ingest

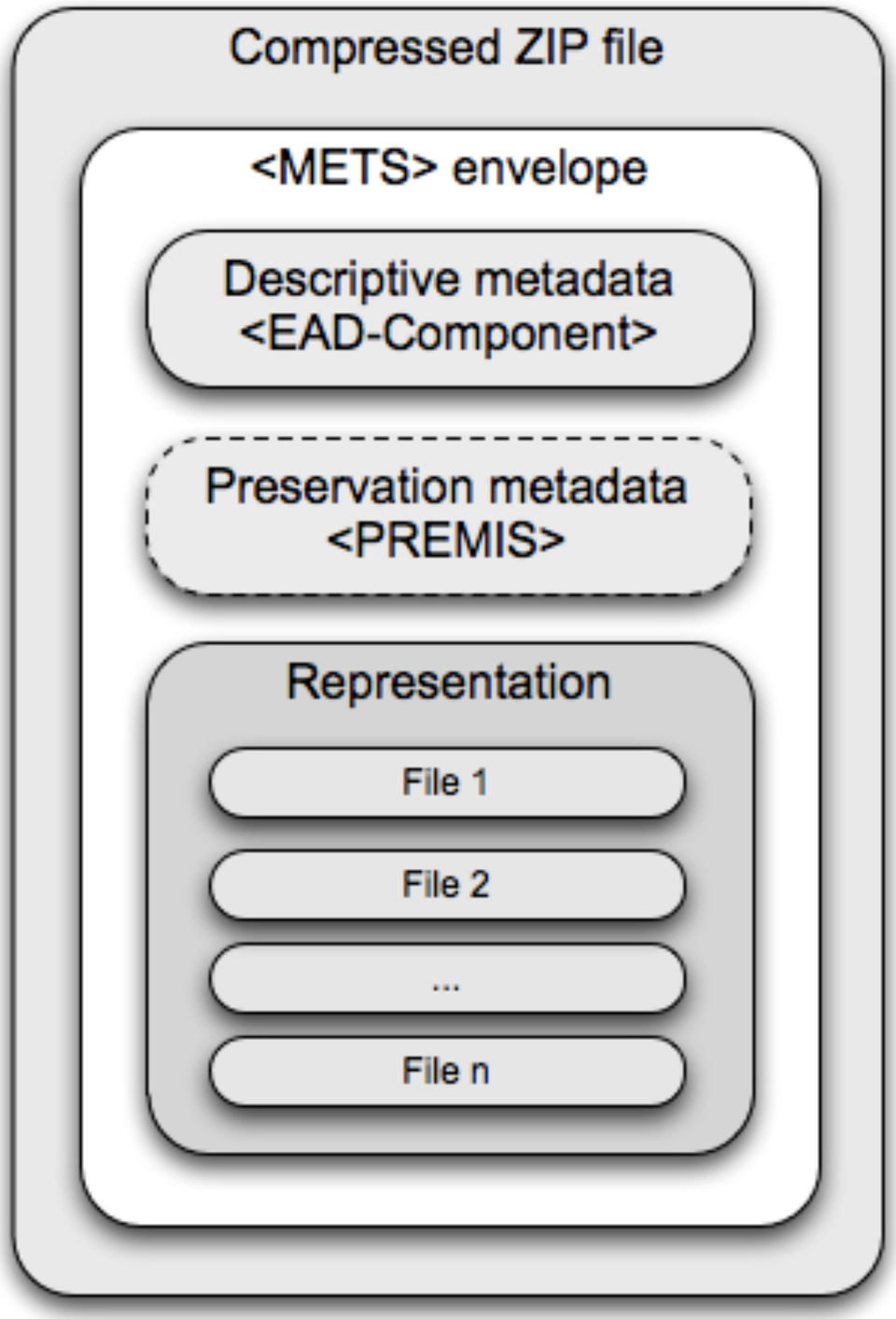

OAIS
Preservation Planning
Data Management
DM
DM
Ingest
Access
SIP
DIP
AIP
AIP
Archival Storage
Administration
Producer
Consumer
Management

SIP structure



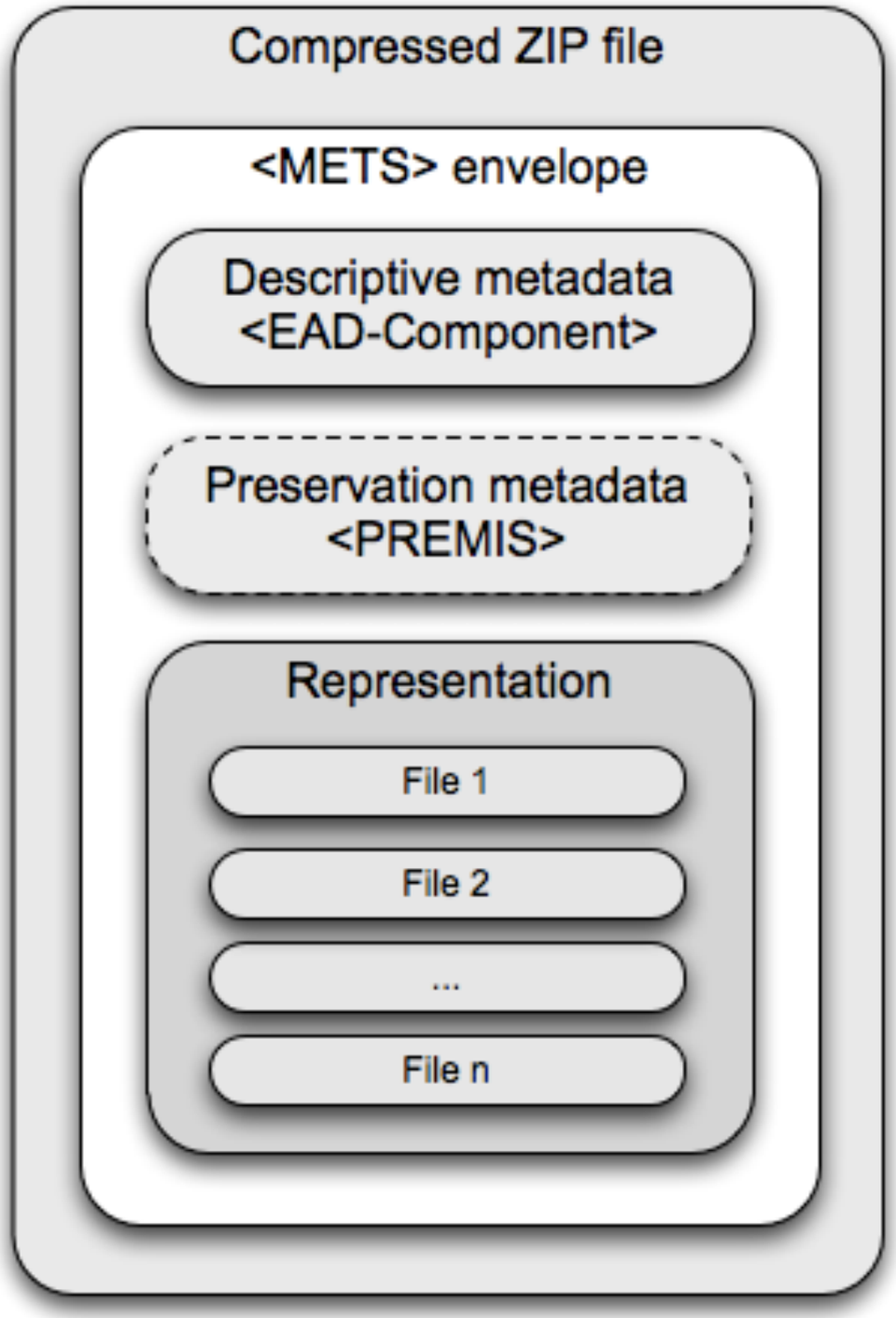
SIP structure

- Object class and format



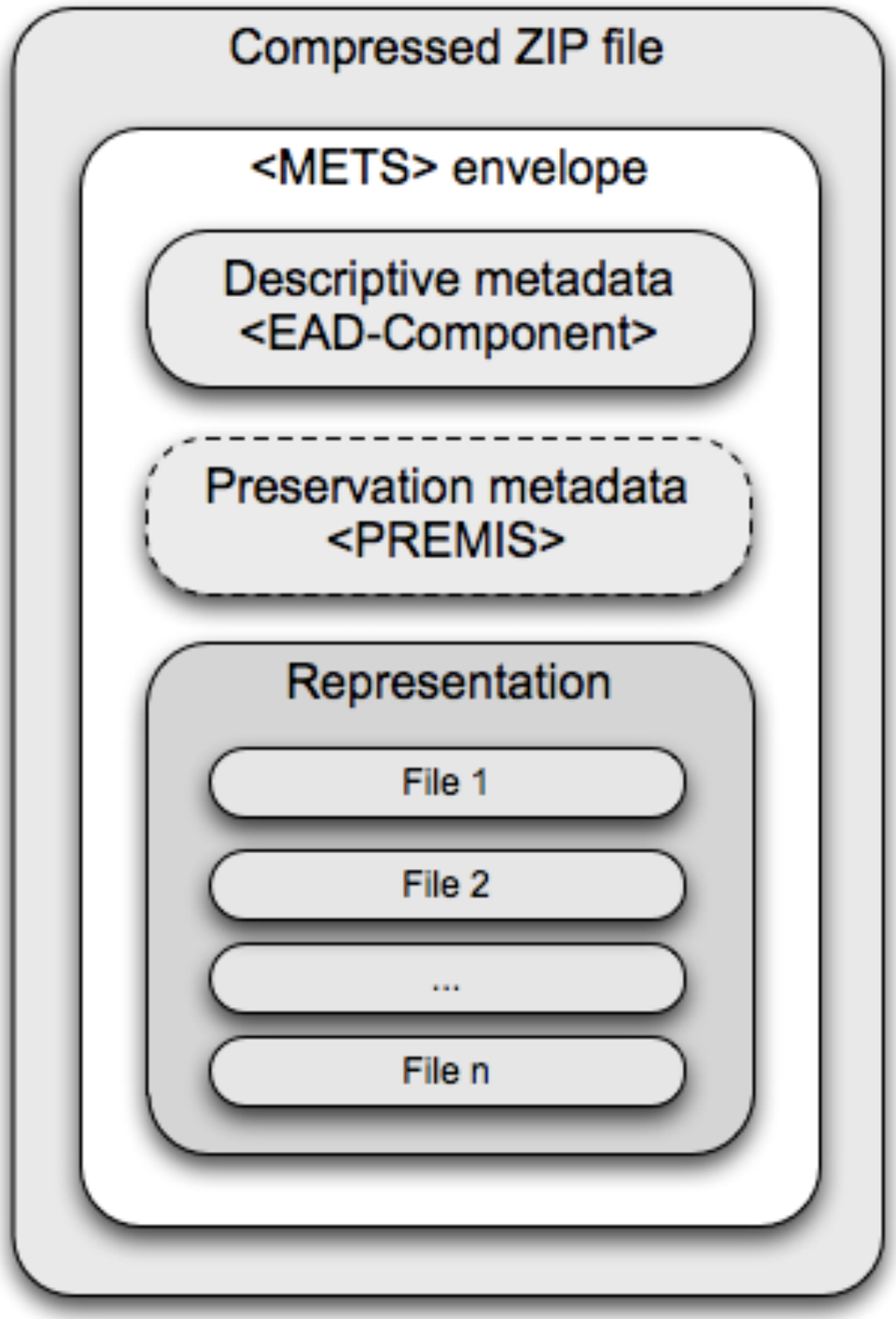
SIP structure

- Object class and format
- Place in classification plan



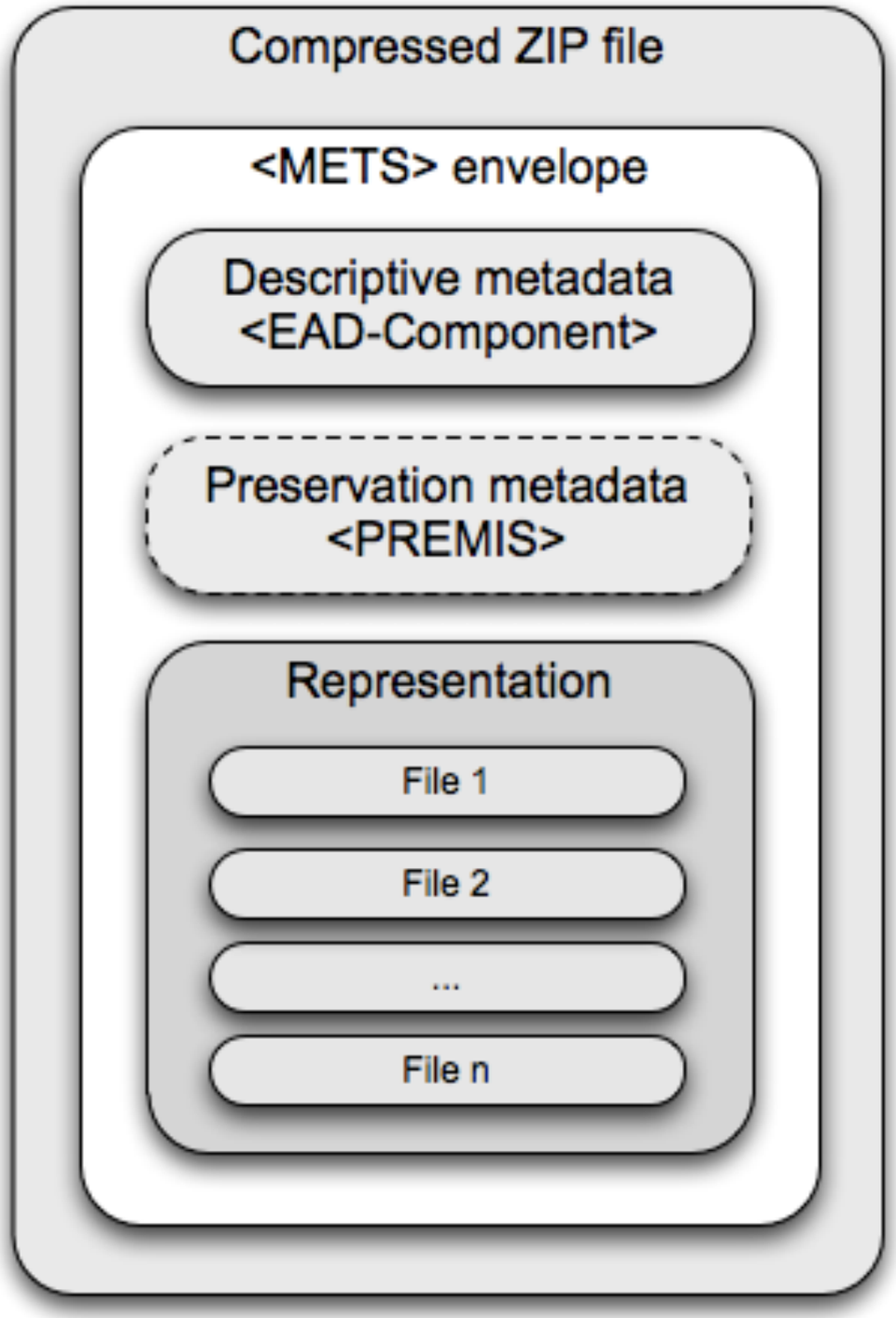
SIP structure

- Object class and format
- Place in classification plan
- Descriptive metadata



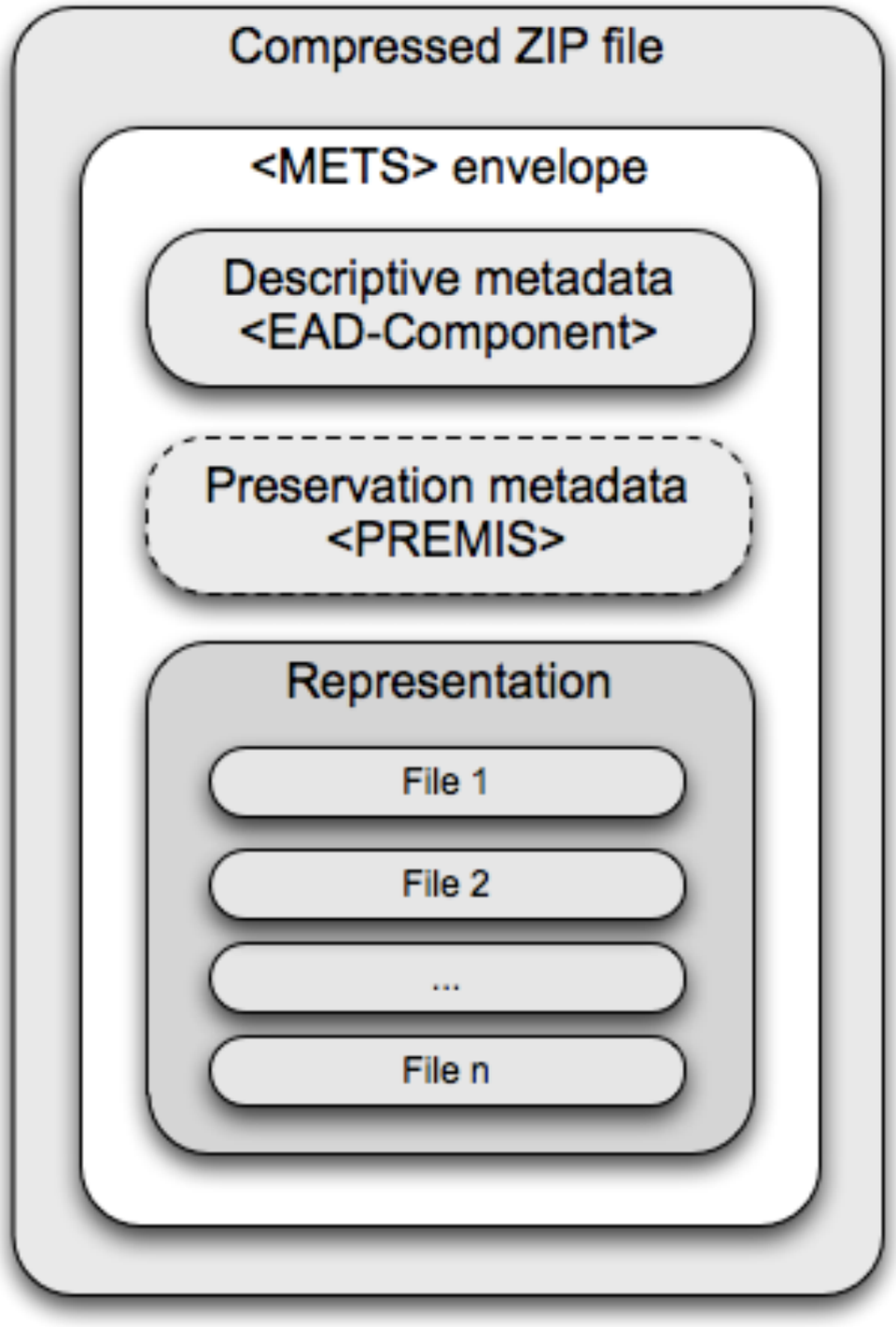
SIP structure

- Object class and format
- Place in classification plan
- Descriptive metadata
- Preservation metadata



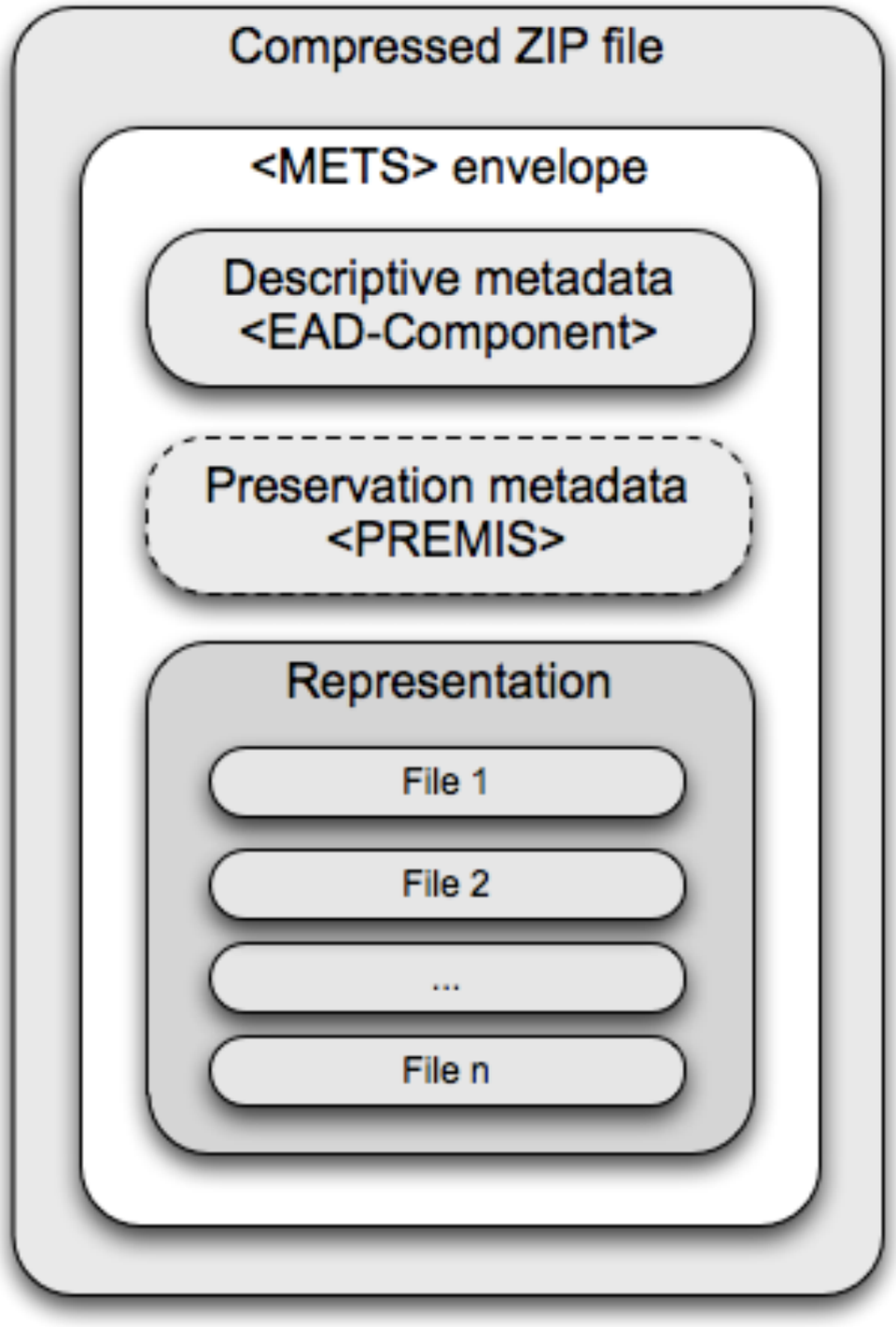
SIP structure

- Object class and format
- Place in classification plan
- Descriptive metadata
- Preservation metadata
- Technical metadata



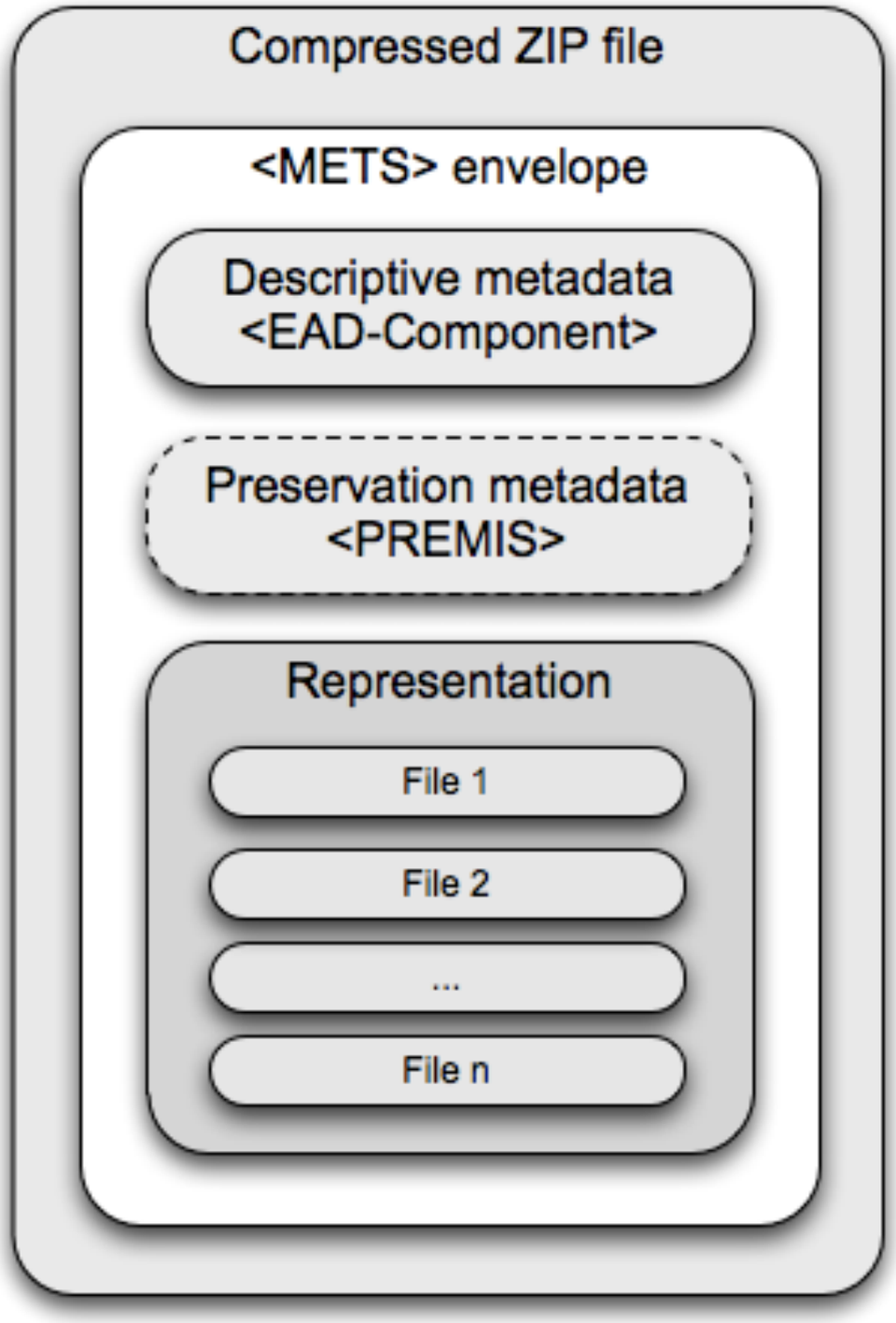
SIP structure

- Object class and format
- Place in classification plan
- Descriptive metadata
- Preservation metadata
- Technical metadata
- Representation files

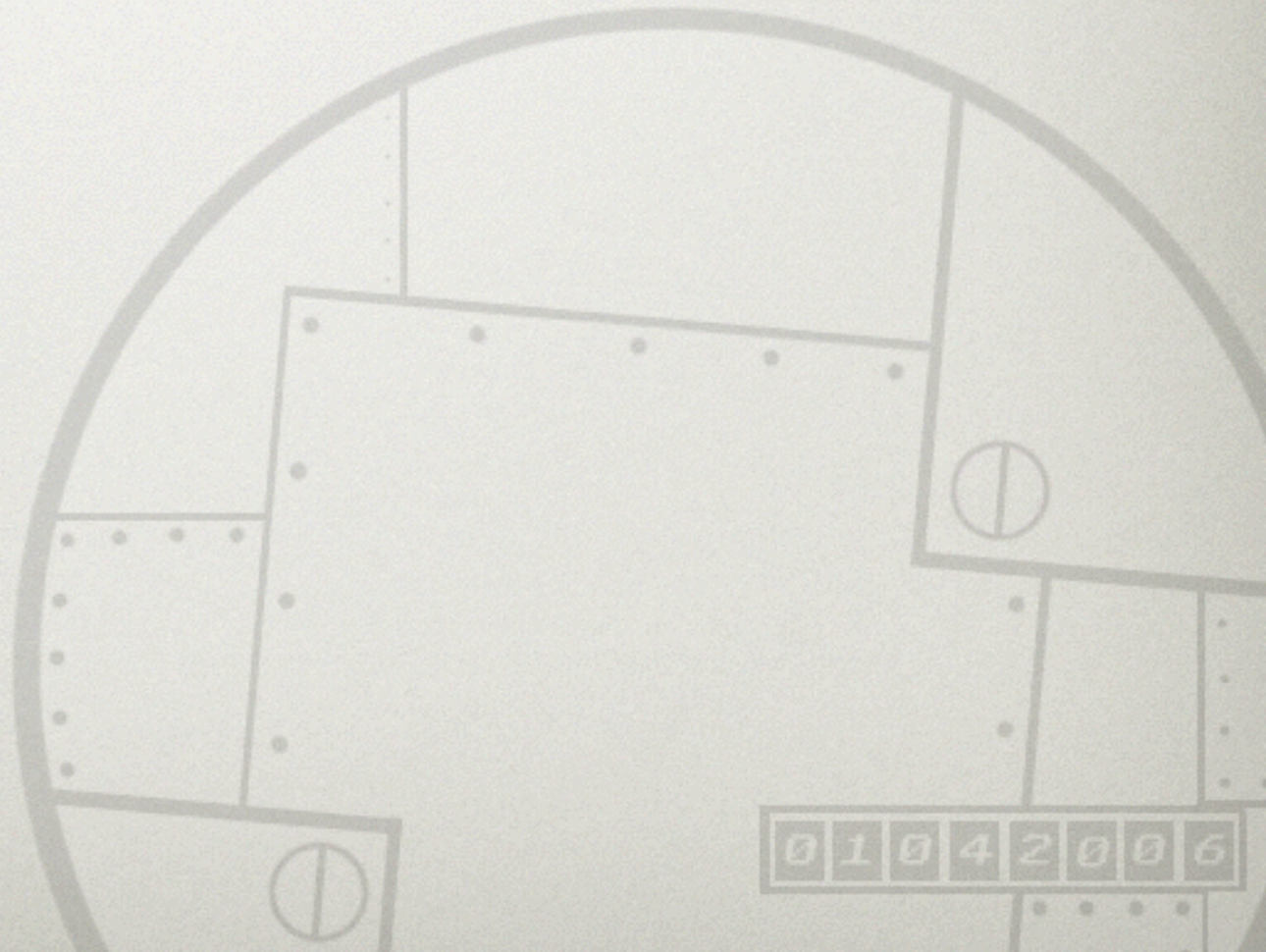


SIP structure

- Object class and format
- Place in classification plan
- Descriptive metadata
- Preservation metadata
- Technical metadata
- Representation files
- Identification of root file

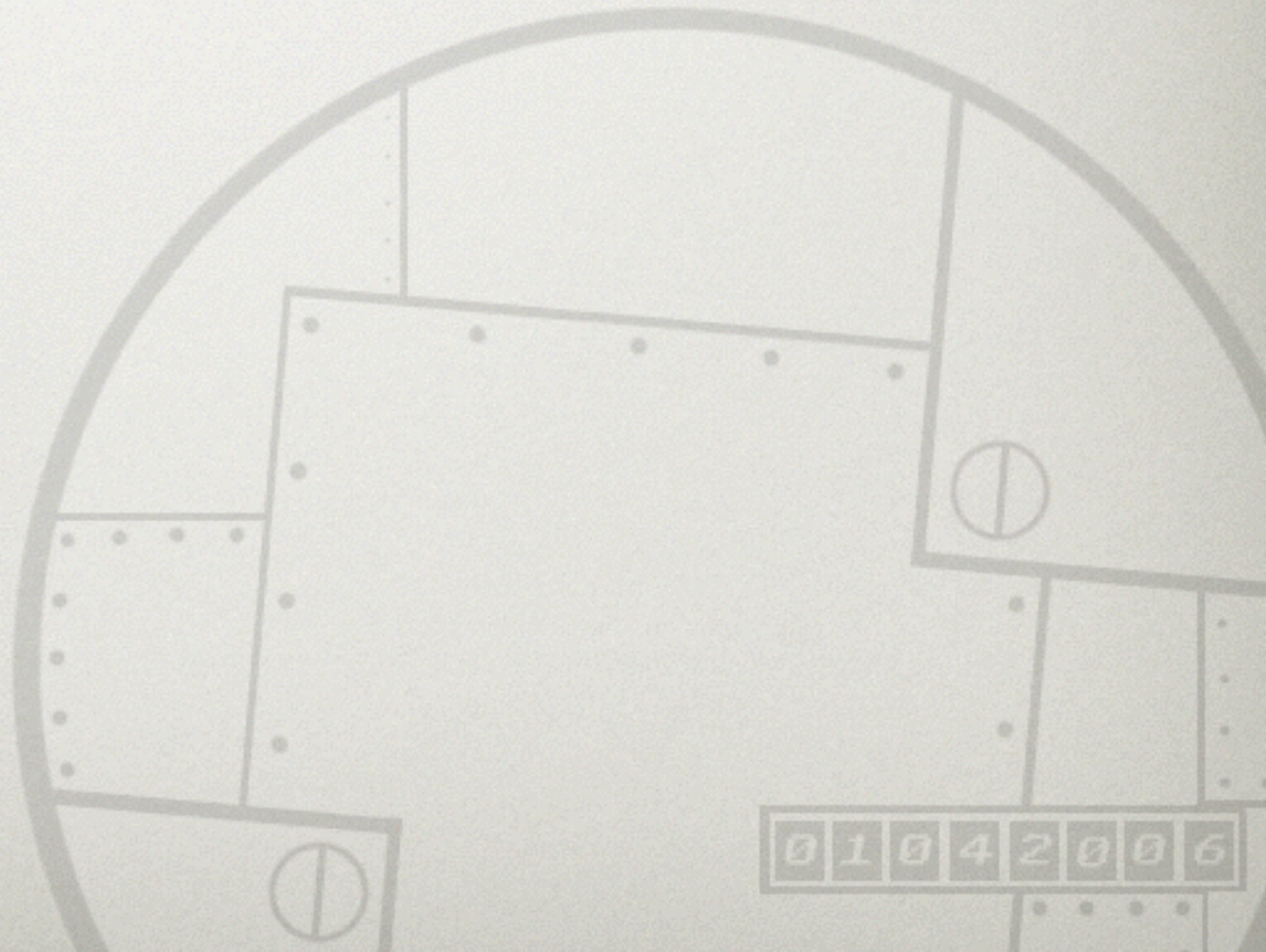


Databases



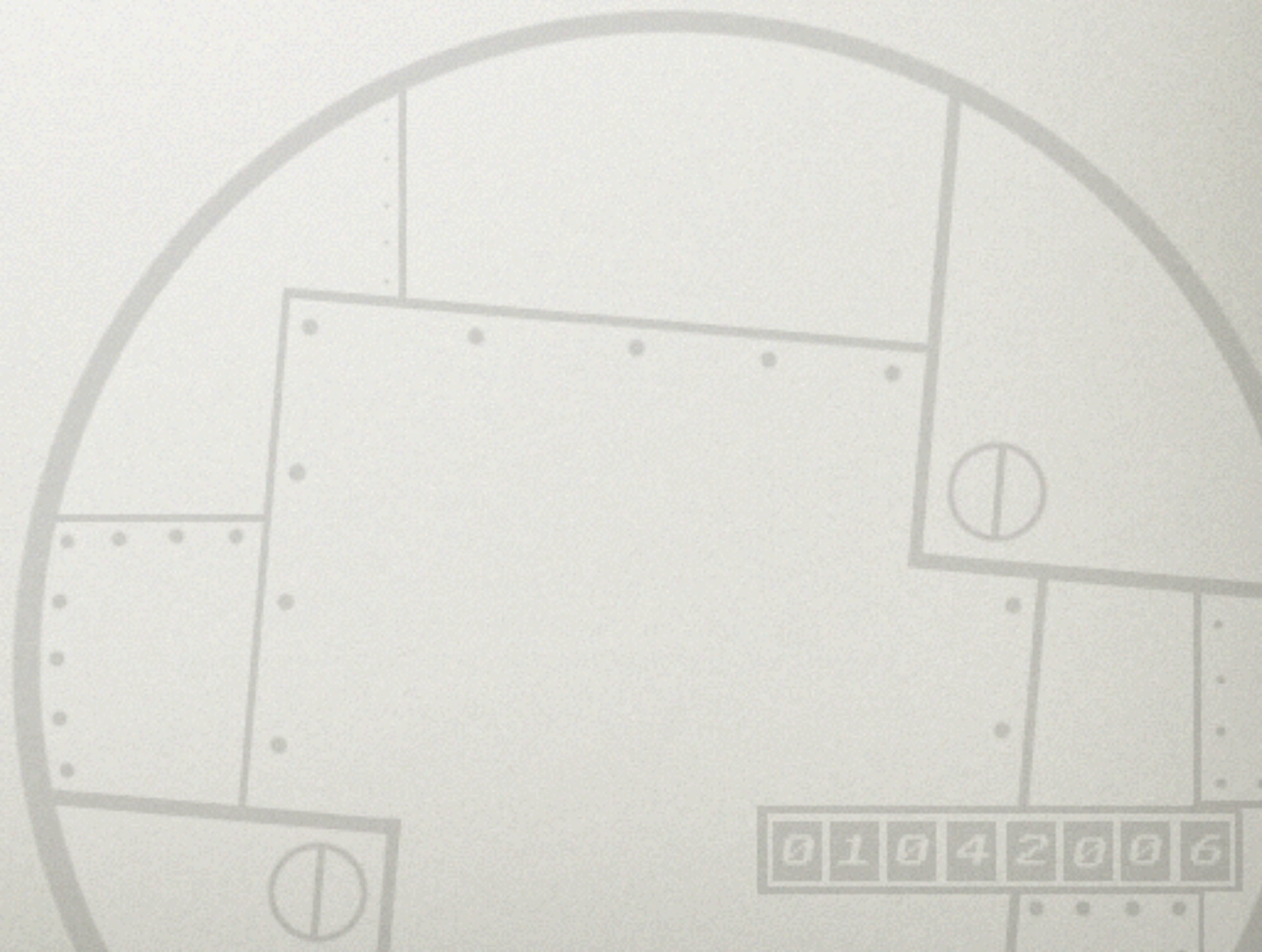
Databases

- **How do we keep archived databases readable and usable in the long term (at acceptable cost)?**



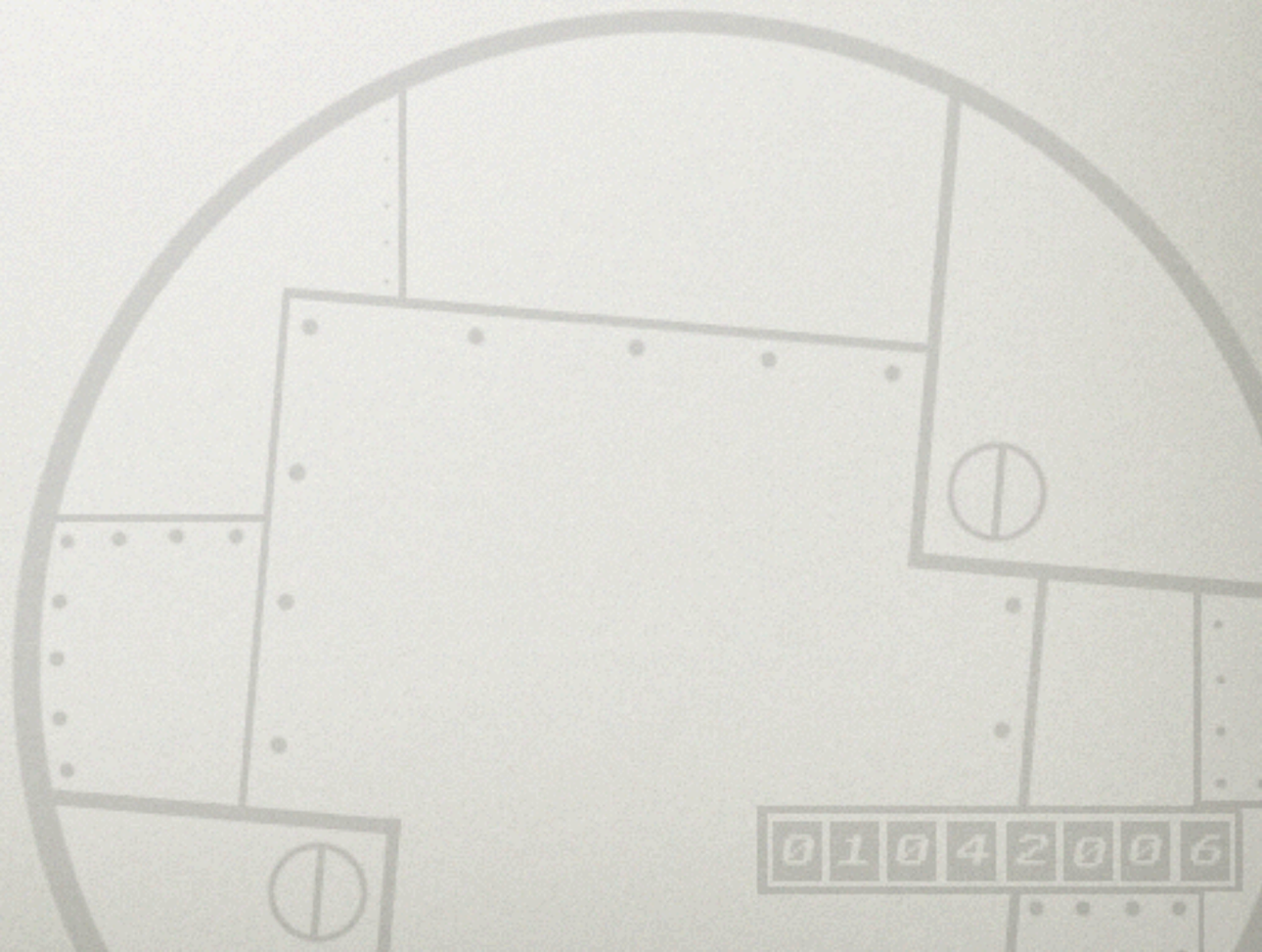
Databases

- **How do we keep archived databases readable and usable in the long term (at acceptable cost)?**
- How do we **separate the data** from a specific database management environment?



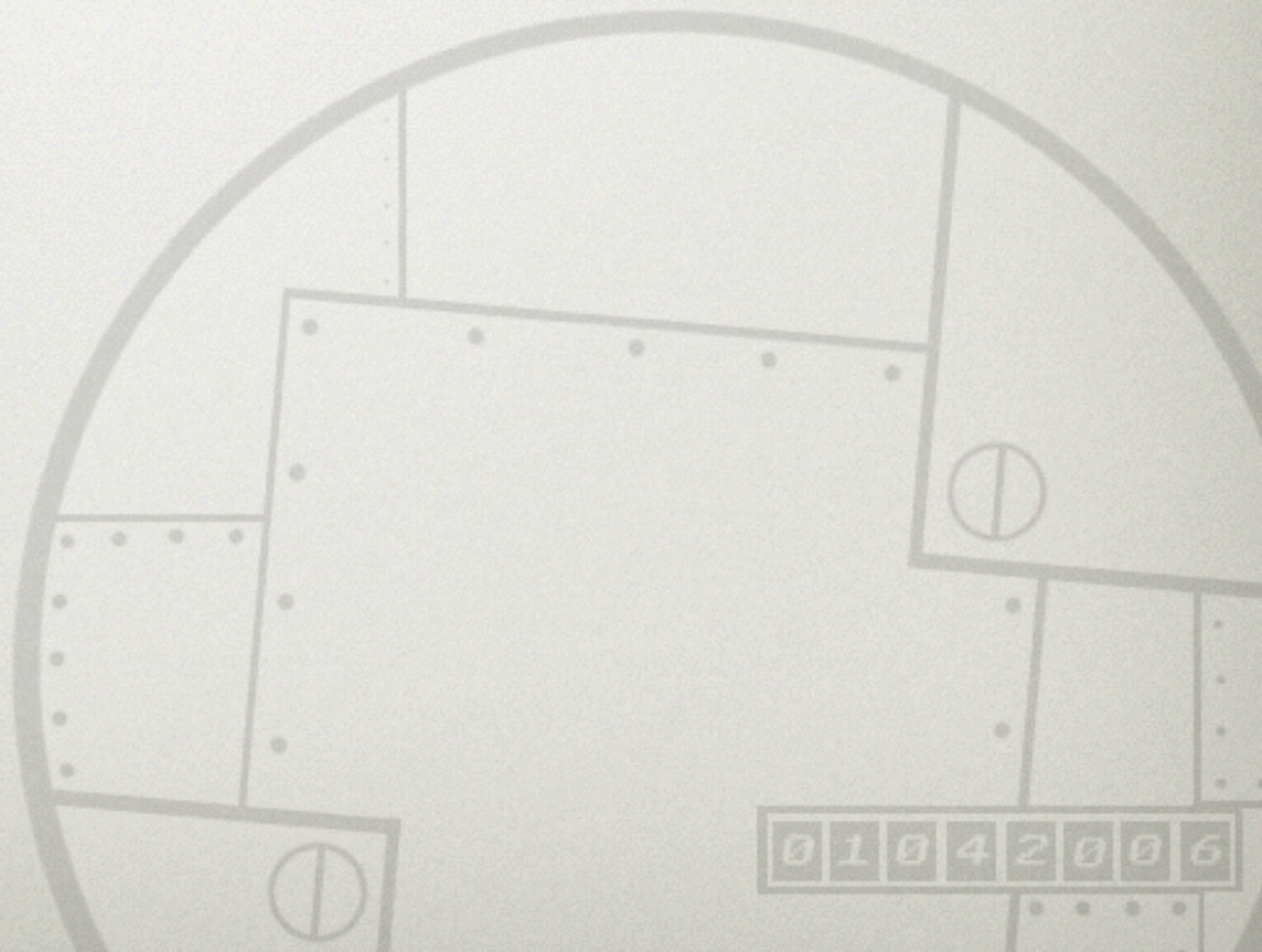
Databases

- **How do we keep archived databases readable and usable in the long term (at acceptable cost)?**
- How do we **separate the data** from a specific database management environment?
- How can we **preserve** the original **data semantics** and **structure**?



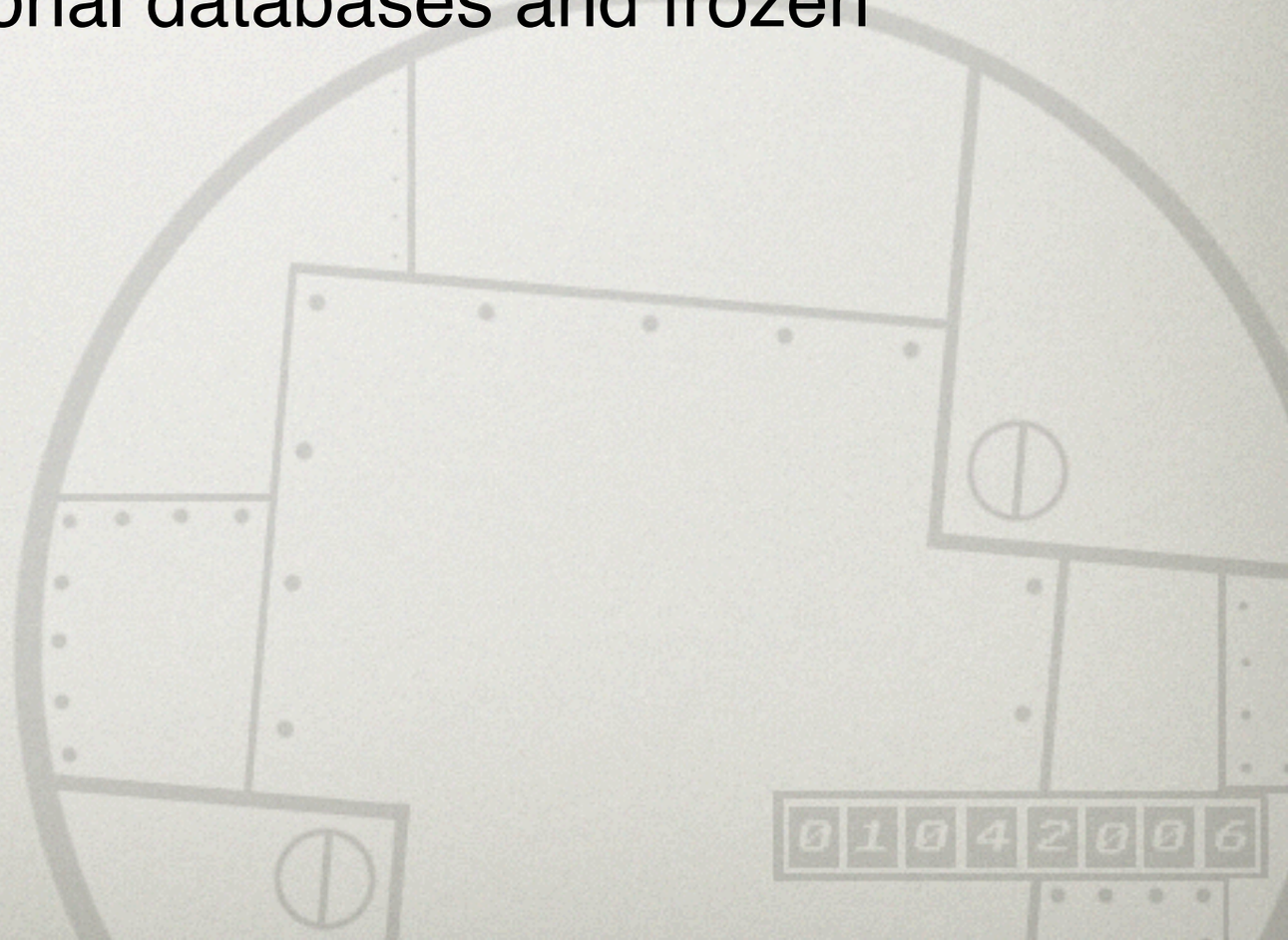
Databases

- **How do we keep archived databases readable and usable in the long term (at acceptable cost)?**
- How do we **separate the data** from a specific database management environment?
- How can we **preserve** the original **data semantics** and **structure**?
- How can we **preserve authenticity** and **provenance** of databases?



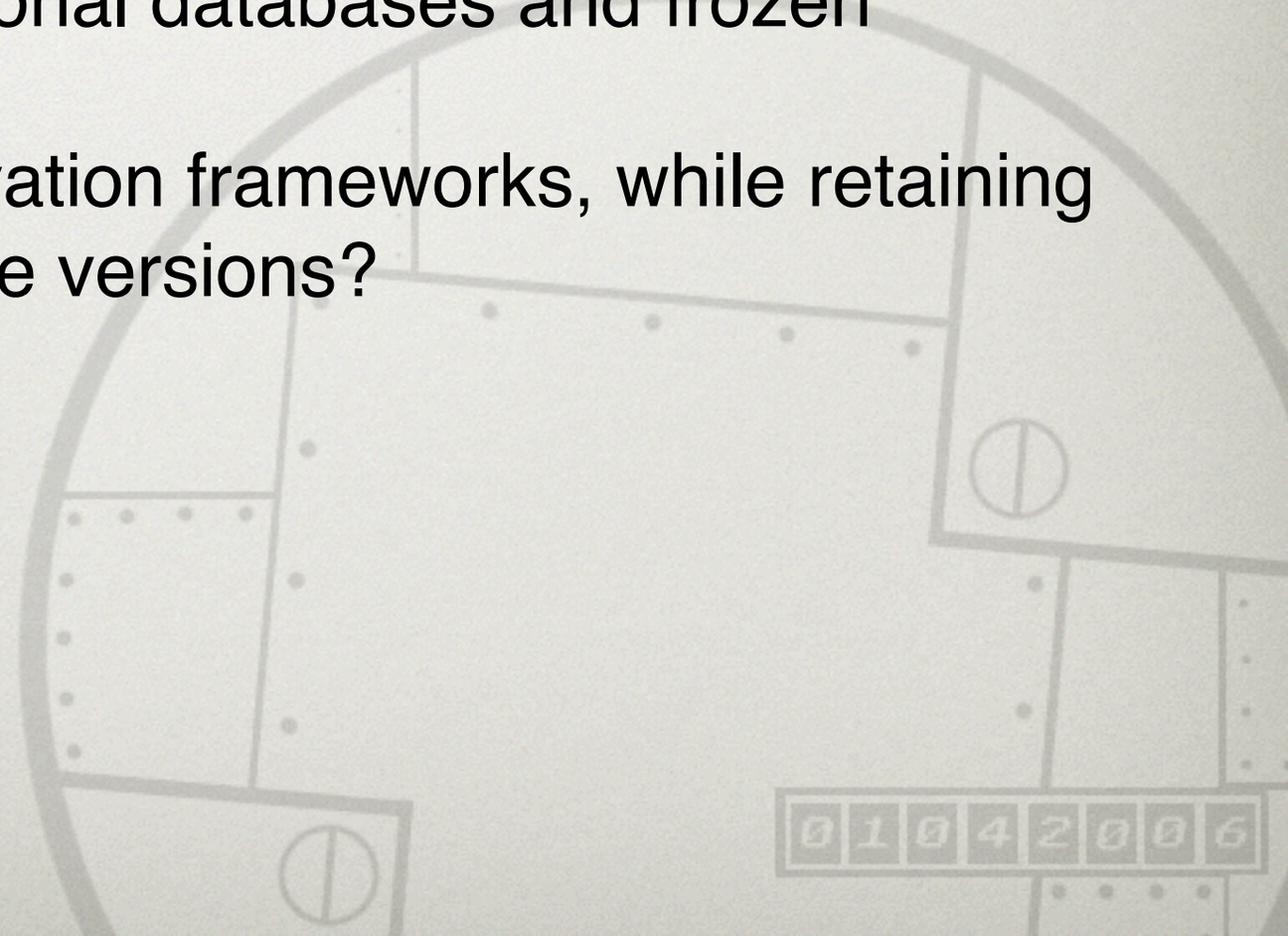
Databases

- **How do we keep archived databases readable and usable in the long term (at acceptable cost)?**
- How do we **separate the data** from a specific database management environment?
- How can we **preserve** the original **data semantics** and **structure**?
- How can we **preserve authenticity** and **provenance** of databases?
- How can we **preserve data** while it continues to **evolve**? (here we can split the problem in two: operational databases and frozen databases)



Databases

- **How do we keep archived databases readable and usable in the long term (at acceptable cost)?**
- How do we **separate the data** from a specific database management environment?
- How can we **preserve** the original **data semantics** and **structure**?
- How can we **preserve authenticity** and **provenance** of databases?
- How can we **preserve data** while it continues to **evolve**? (here we can split the problem in two: operational databases and frozen databases)
- How can we have efficient preservation frameworks, while retaining the ability to query different database versions?



Databases

- **How do we keep archived databases readable and usable in the long term (at acceptable cost)?**
- How do we **separate the data** from a specific database management environment?
- How can we **preserve** the original **data semantics** and **structure**?
- How can we **preserve authenticity** and **provenance** of databases?
- How can we **preserve data** while it continues to **evolve**? (here we can split the problem in two: operational databases and frozen databases)
- How can we have efficient preservation frameworks, while retaining the ability to query different database versions?
- How can multi-user online access be provided to hundreds of archived databases containing terabytes of data?

Databases

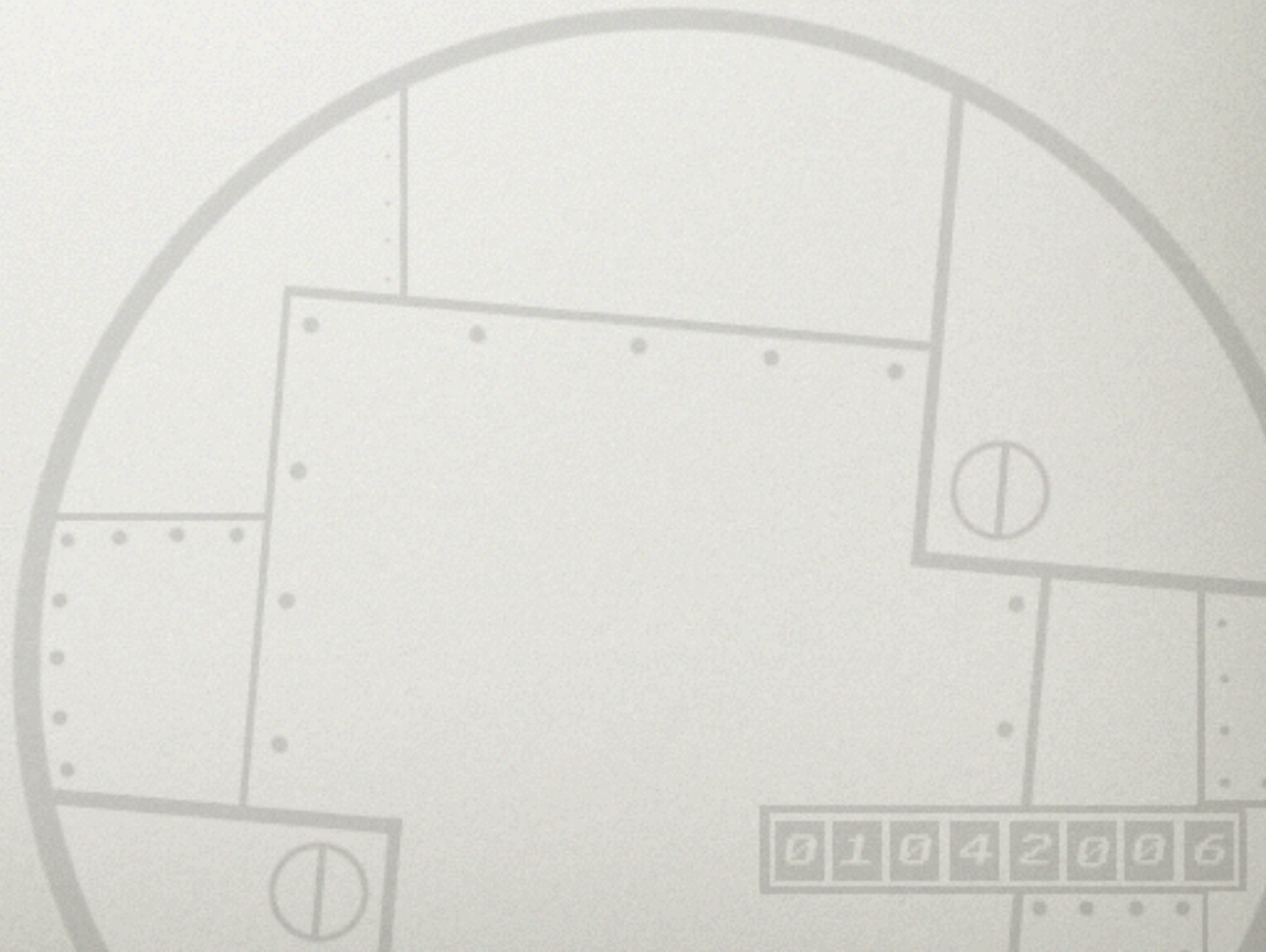
- **How do we keep archived databases readable and usable in the long term (at acceptable cost)?**
- How do we **separate the data** from a specific database management environment?
- How can we **preserve** the original **data semantics** and **structure**?
- How can we **preserve authenticity** and **provenance** of databases?
- How can we **preserve data** while it continues to **evolve**? (here we can split the problem in two: operational databases and frozen databases)
- How can we have efficient preservation frameworks, while retaining the ability to query different database versions?
- How can multi-user online access be provided to hundreds of archived databases containing terabytes of data?
- Can we move from a centralized model to a distributed, redundant model of database preservation?

Databases

- **How do we keep archived databases readable and usable in the long term (at acceptable cost)?**
- How do we **separate the data** from a specific database management environment?
- How can we **preserve** the original **data semantics** and **structure**?
- How can we **preserve authenticity** and **provenance** of databases?
- How can we **preserve data** while it continues to **evolve**? (here we can split the problem in two: operational databases and frozen databases)
- How can we have efficient preservation frameworks, while retaining the ability to query different database versions?
- How can multi-user online access be provided to hundreds of archived databases containing terabytes of data?
- Can we move from a centralized model to a distributed, redundant model of database preservation?

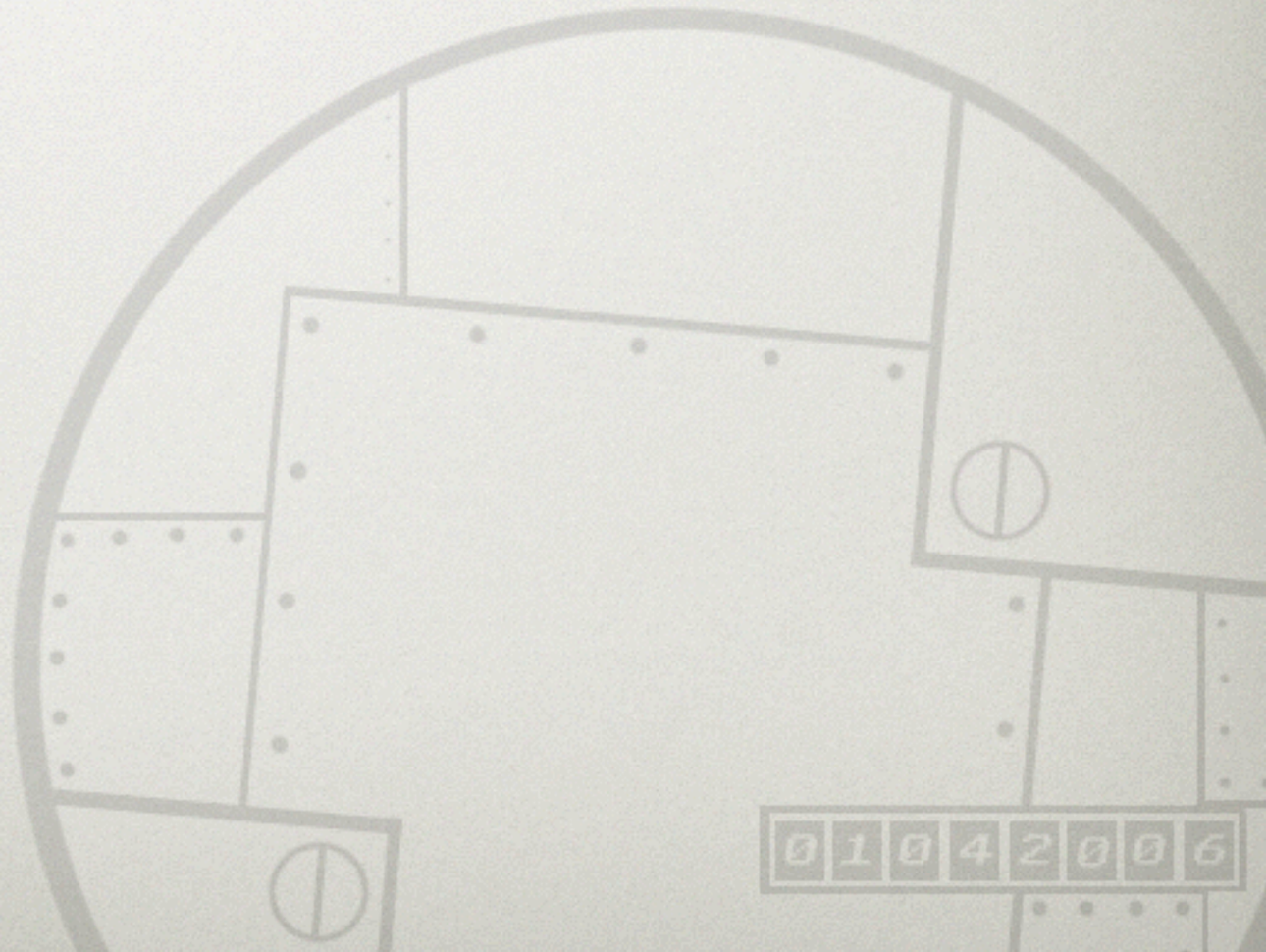
General big questions...

Databases



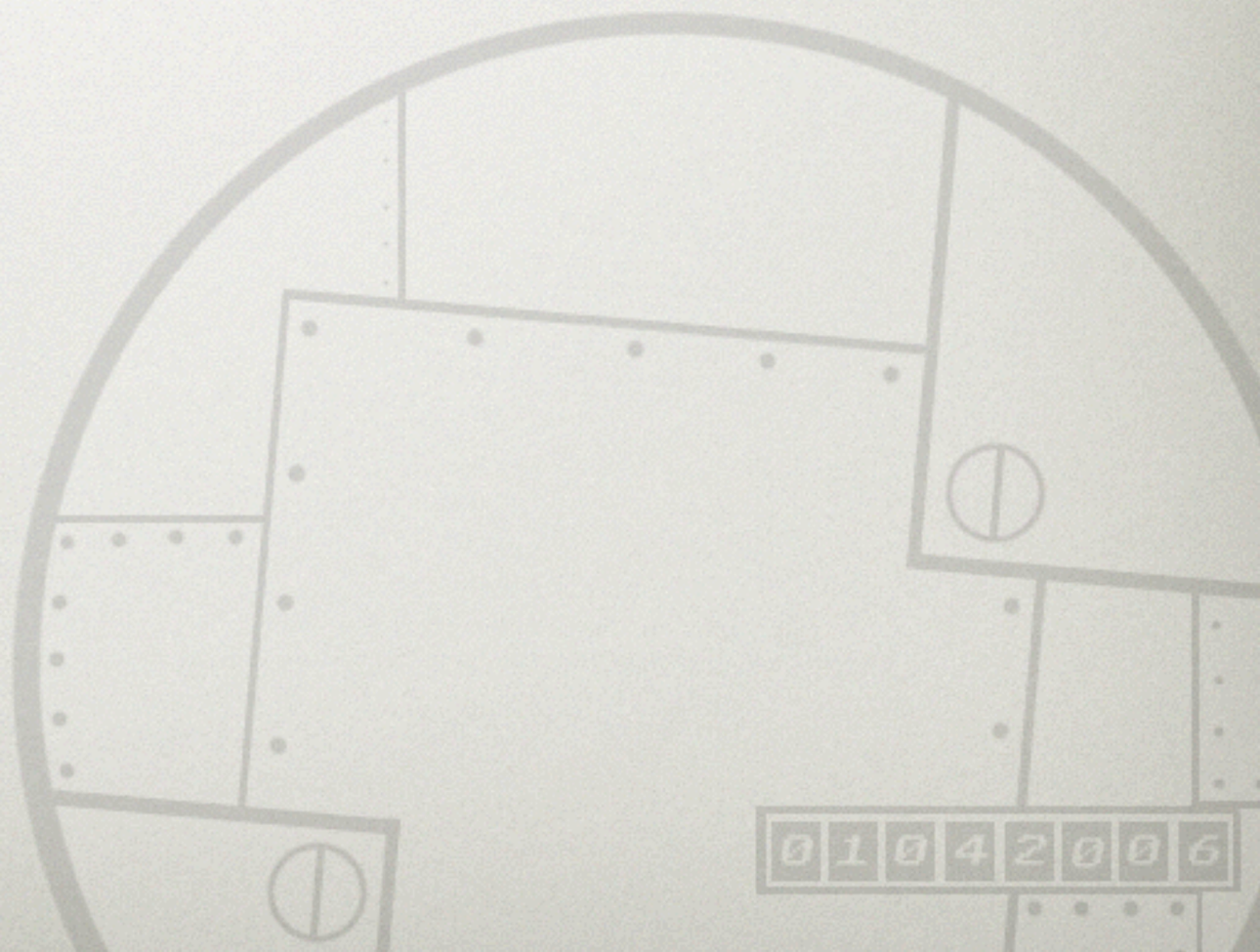
Databases

- **What are the salient features of a database that should be preserved?** ***(significant properties...)***



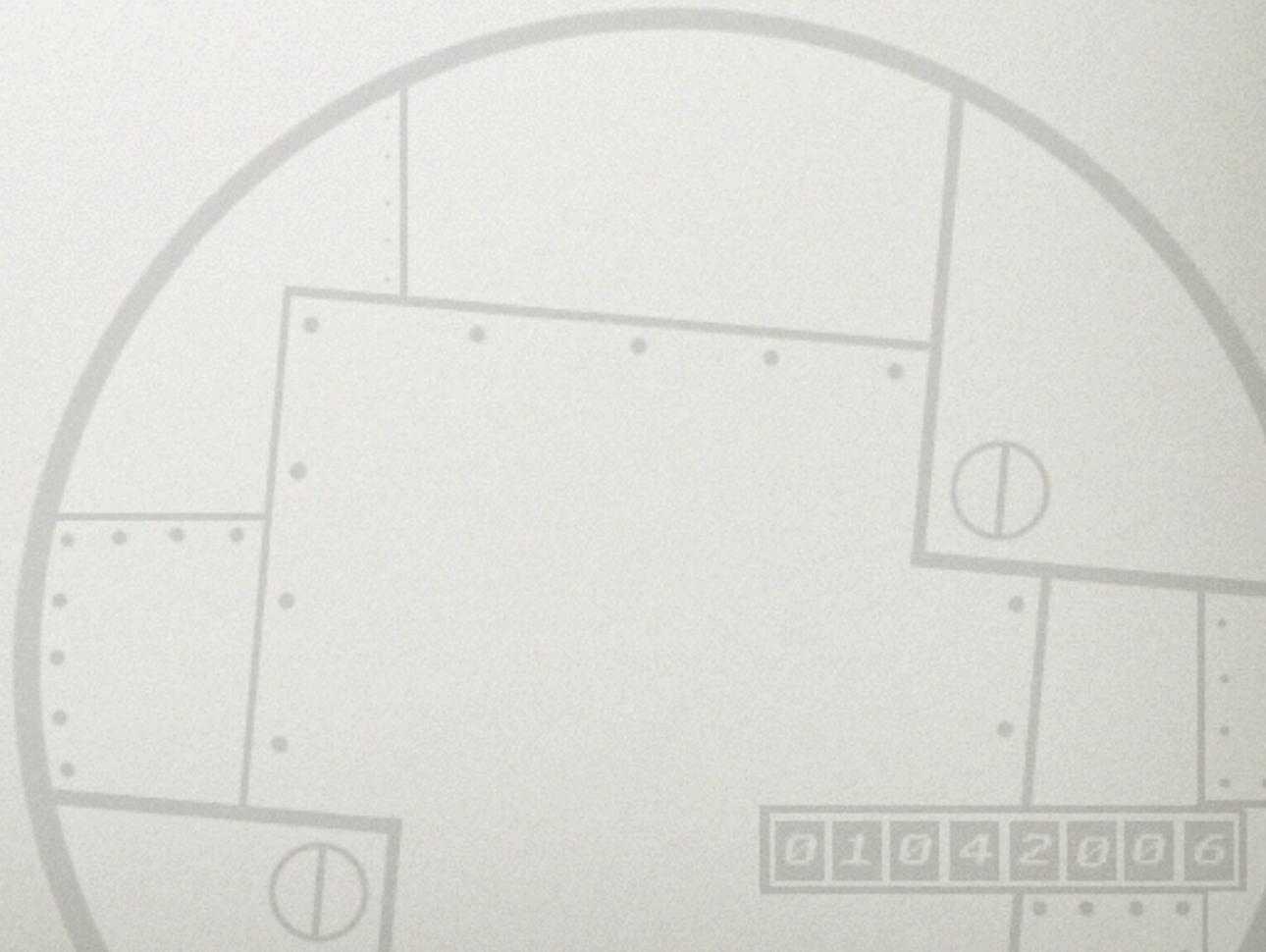
Databases

- **What are the salient features of a database that should be preserved?** ***(significant properties...)***
- What are the different **stages** in the database preservation's life cycle?



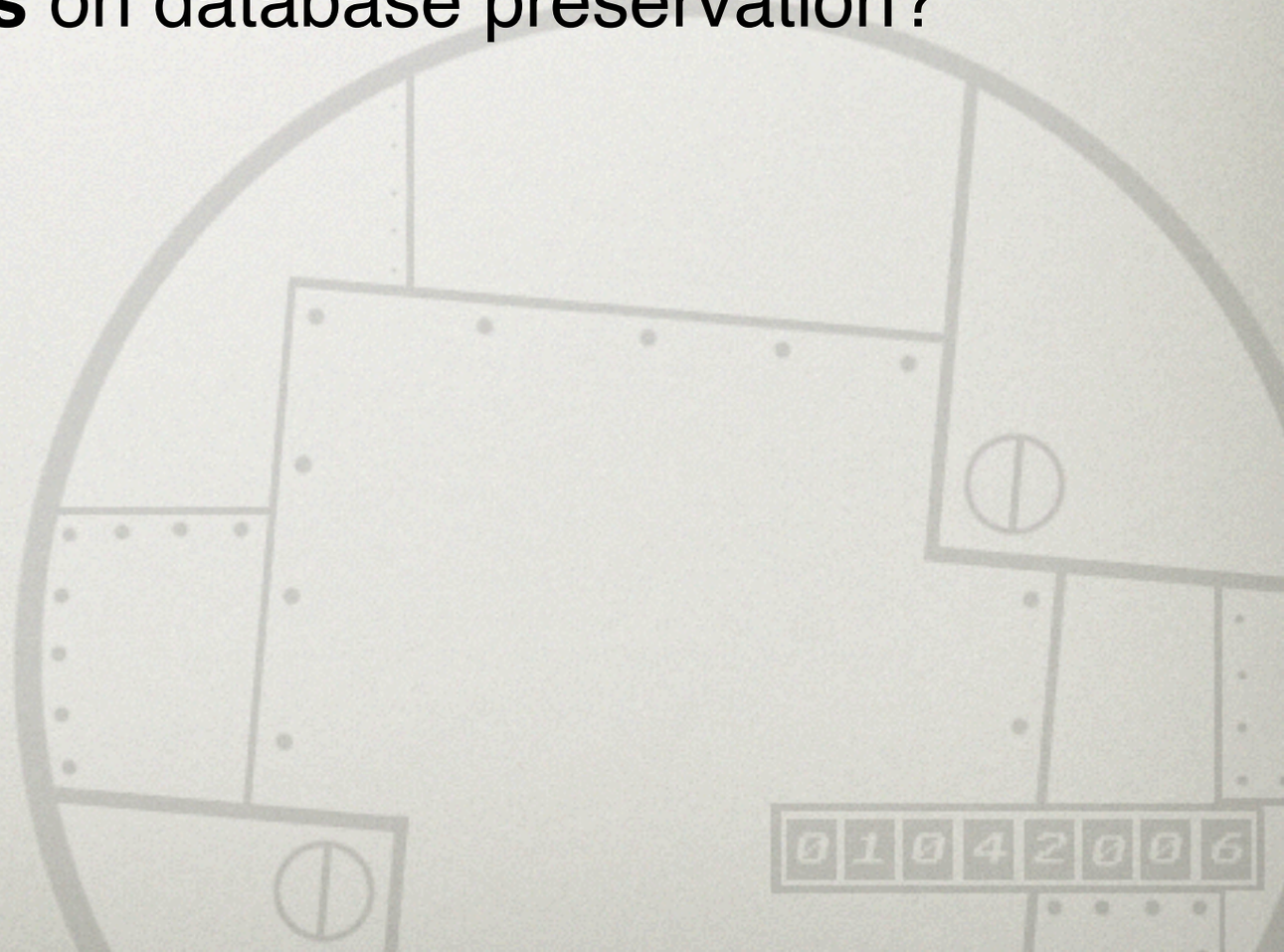
Databases

- **What are the salient features of a database that should be preserved?** ***(significant properties...)***
- What are the different **stages** in the database preservation's life cycle?
- What **documentation** is preserved together with a database, and in what **format**?



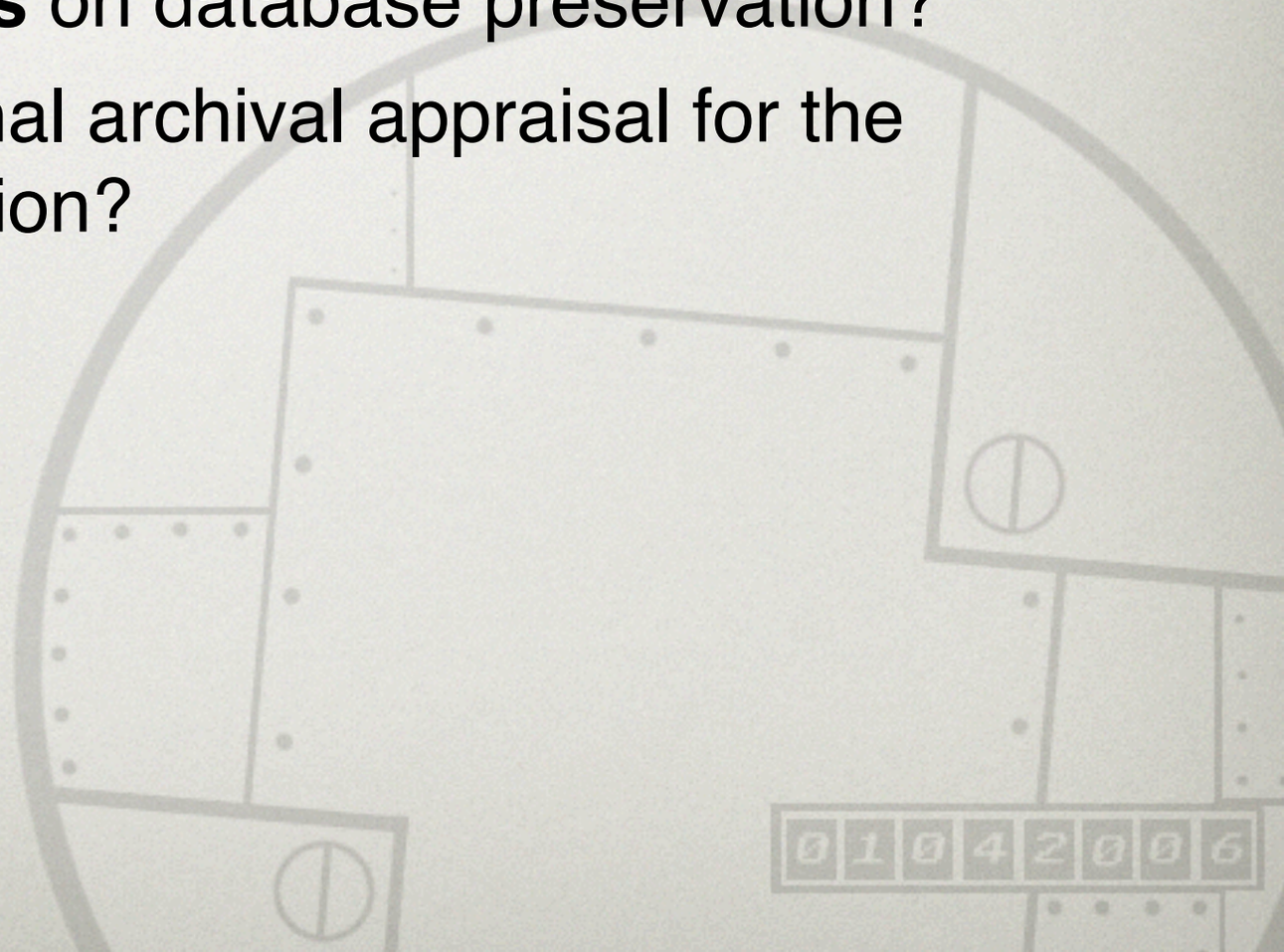
Databases

- **What are the salient features of a database that should be preserved?** ***(significant properties...)***
- What are the different **stages** in the database preservation's life cycle?
- What **documentation** is preserved together with a database, and in what **format**?
- What are the **legal encumbrances** on database preservation?



Databases

- **What are the salient features of a database that should be preserved?** ***(significant properties...)***
- What are the different **stages** in the database preservation's life cycle?
- What **documentation** is preserved together with a database, and in what **format**?
- What are the **legal encumbrances** on database preservation?
- What can be learned from traditional archival appraisal for the selection of databases for preservation?



Databases

- **What are the salient features of a database that should be preserved?** ***(significant properties...)***
- What are the different **stages** in the database preservation's life cycle?
- What **documentation** is preserved together with a database, and in what **format**?
- What are the **legal encumbrances** on database preservation?
- What can be learned from traditional archival appraisal for the selection of databases for preservation?
- To what extent can the preservation strategies, and procedural policies developed by archivists be adapted for databases?

Databases

- **What are the salient features of a database that should be preserved?** ***(significant properties...)***
- What are the different **stages** in the database preservation's life cycle?
- What **documentation** is preserved together with a database, and in what **format**?
- What are the **legal encumbrances** on database preservation?
- What can be learned from traditional archival appraisal for the selection of databases for preservation?
- To what extent can the preservation strategies, and procedural policies developed by archivists be adapted for databases?
- How can we mesure the quality of preservation strategies when they are applied to data- bases? What are **DB significant properties**? *(quality assurance...)*

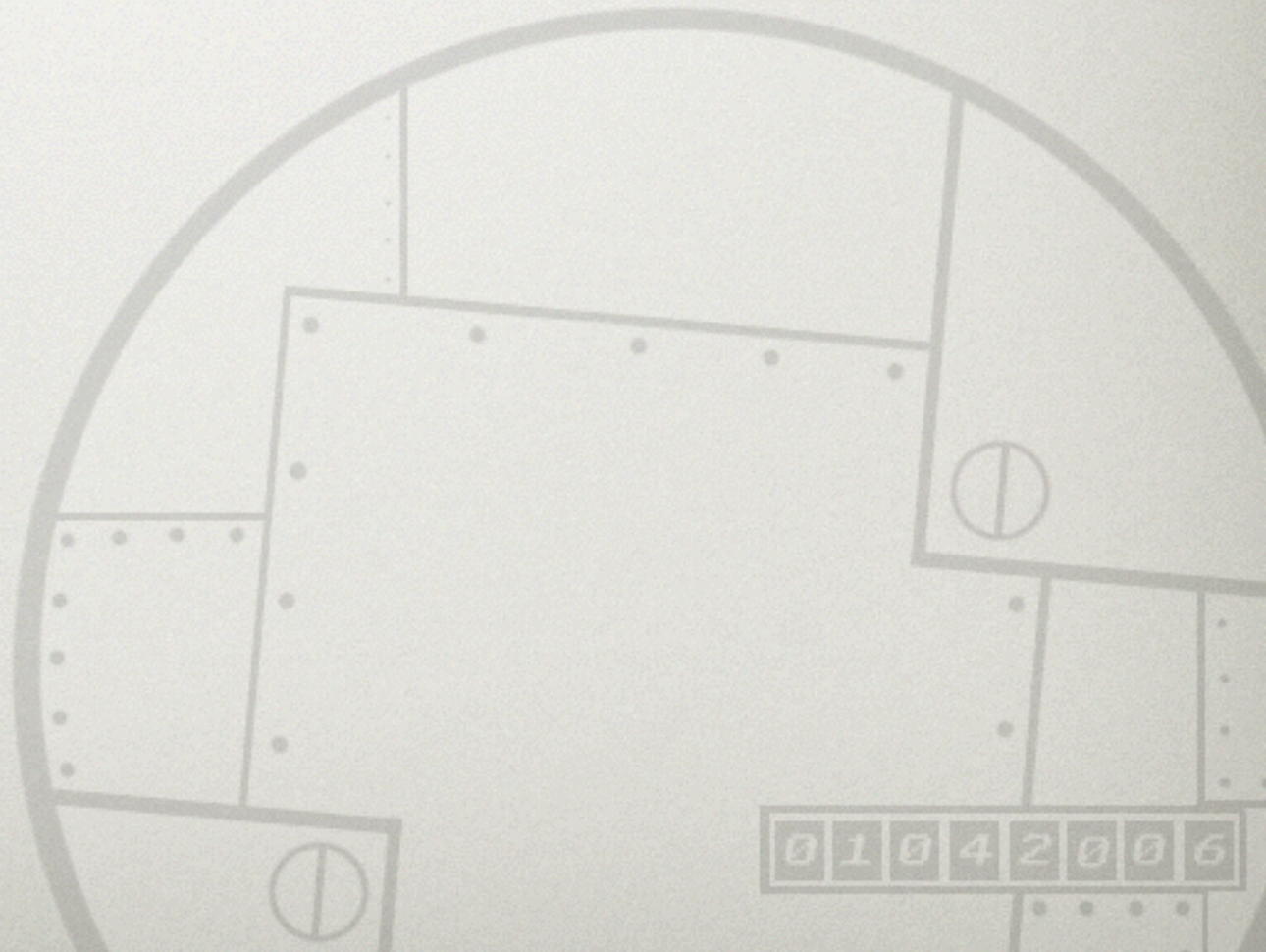
Databases

- **What are the salient features of a database that should be preserved?** ***(significant properties...)***
- What are the different **stages** in the database preservation's life cycle?
- What **documentation** is preserved together with a database, and in what **format**?
- What are the **legal encumbrances** on database preservation?
- What can be learned from traditional archival appraisal for the selection of databases for preservation?
- To what extent can the preservation strategies, and procedural policies developed by archivists be adapted for databases?
- How can we mesure the quality of preservation strategies when they are applied to data- bases? What are **DB significant properties**? *(quality assurance...)*

Technical questions...

Databases: goals

- How do we store them?
- How do we access them?

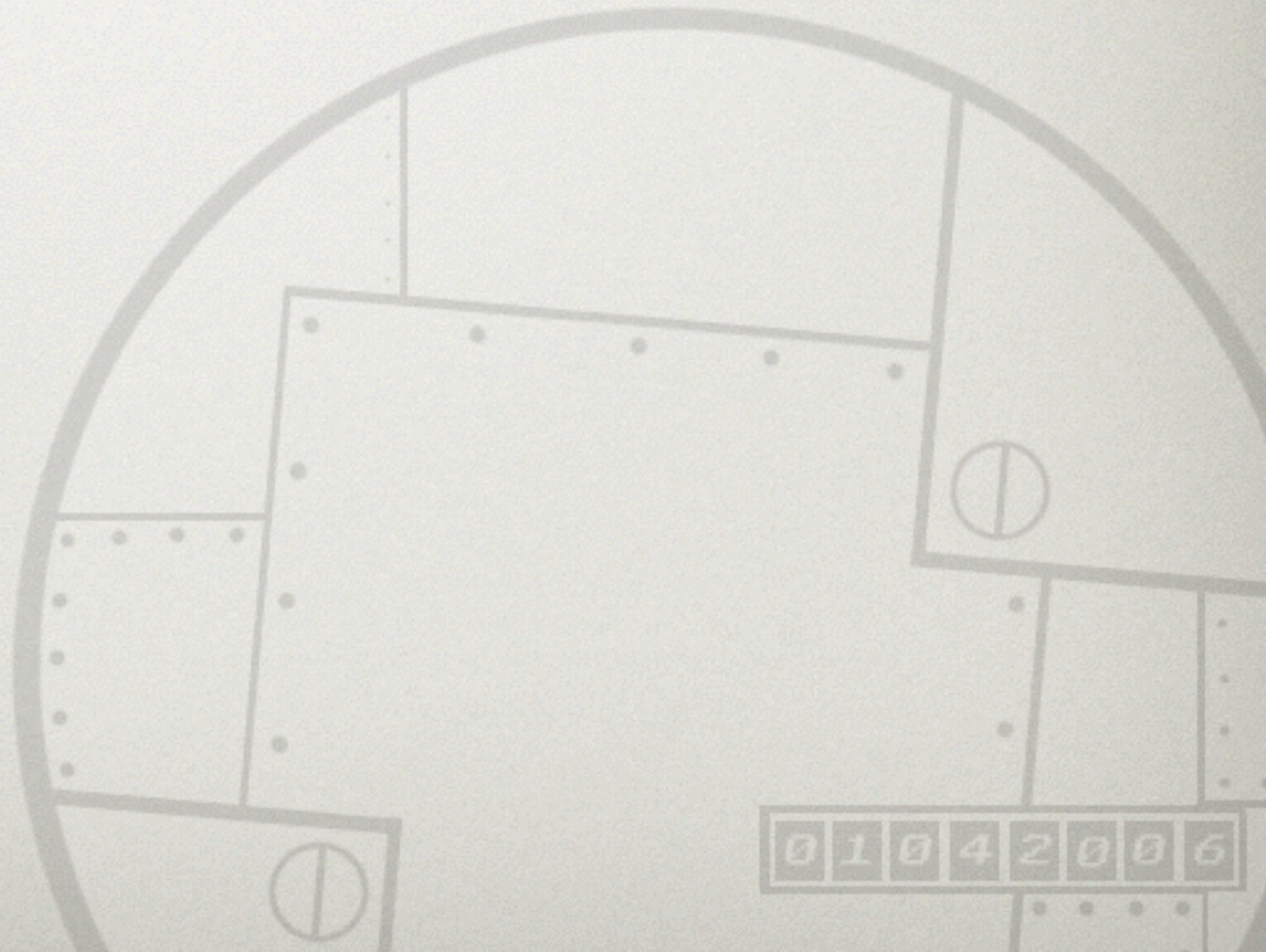


Databases: goals

- How do we store them?
- How do we access them?

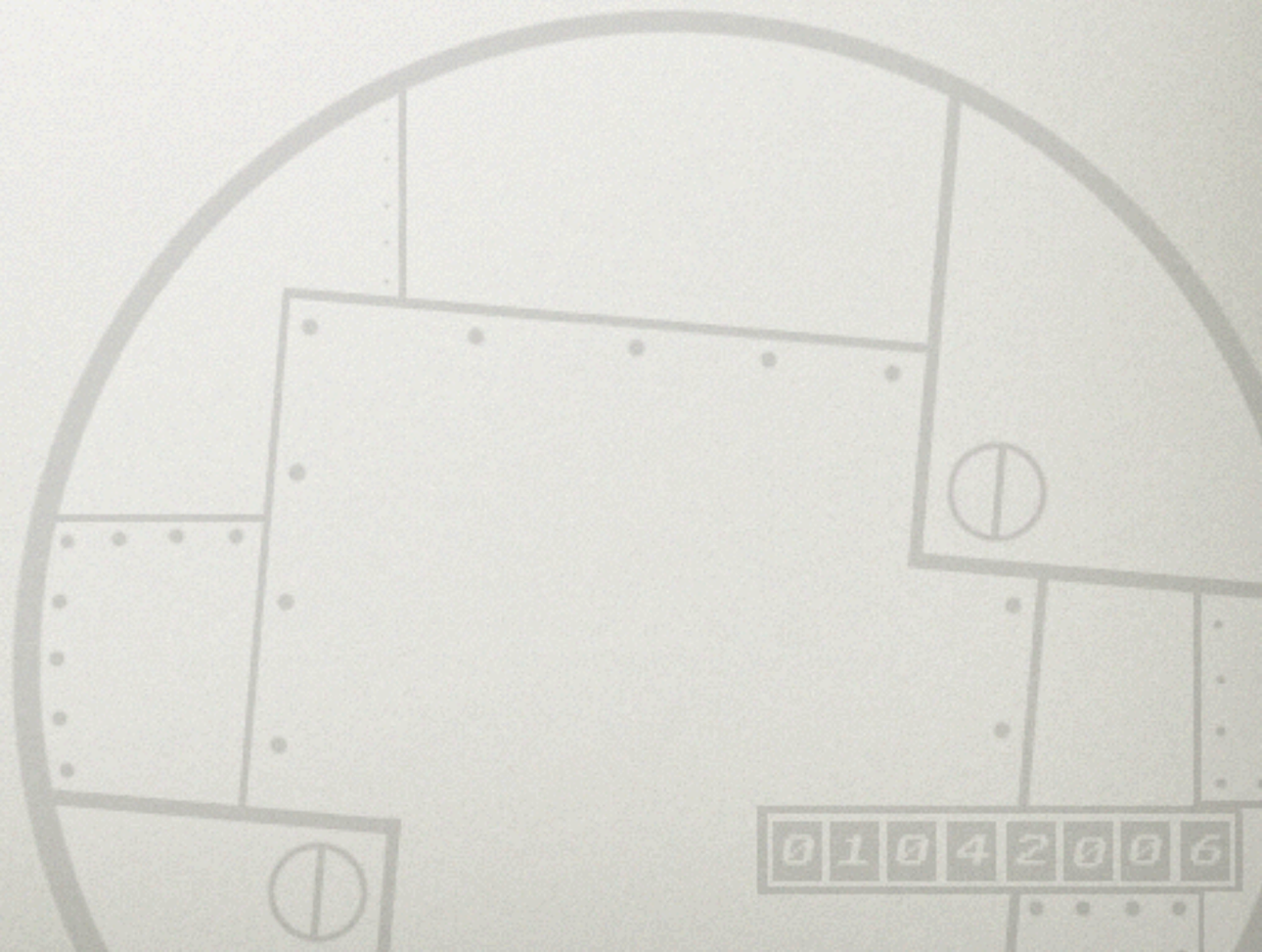
RODA questions...

Databases



Databases

Normal evolution path:
Data => Structure => Semantics



Databases

Normal evolution path:
Data => Structure => Semantics

First prototype:

- Data
- Structure
- Only “frozen” DBs

Databases

- Data?

Normal evolution path:
Data => Structure => Semantics

First prototype:

- Data
- Structure
- Only “frozen” DBs

Databases

- Data?
- Structure?

Normal evolution path:
Data => Structure => Semantics

First prototype:

- Data
- Structure
- Only “frozen” DBs

Databases

- Data?
- Structure?
- Views?

Normal evolution path:
Data => Structure => Semantics

First prototype:

- Data
- Structure
- Only “frozen” DBs

Databases

- Data?
- Structure?
- Views?
- Reports?

Normal evolution path:
Data => Structure => Semantics

First prototype:

- Data
- Structure
- Only “frozen” DBs

Databases

- Data?
- Structure?
- Views?
- Reports?
- Stored Procedure

Normal evolution path:
Data => Structure => Semantics

First prototype:

- Data
- Structure
- Only “frozen” DBs

Databases

- Data?
- Structure?
- Views?
- Reports?
- Stored Procedure
- ...

Normal evolution path:
Data => Structure => Semantics

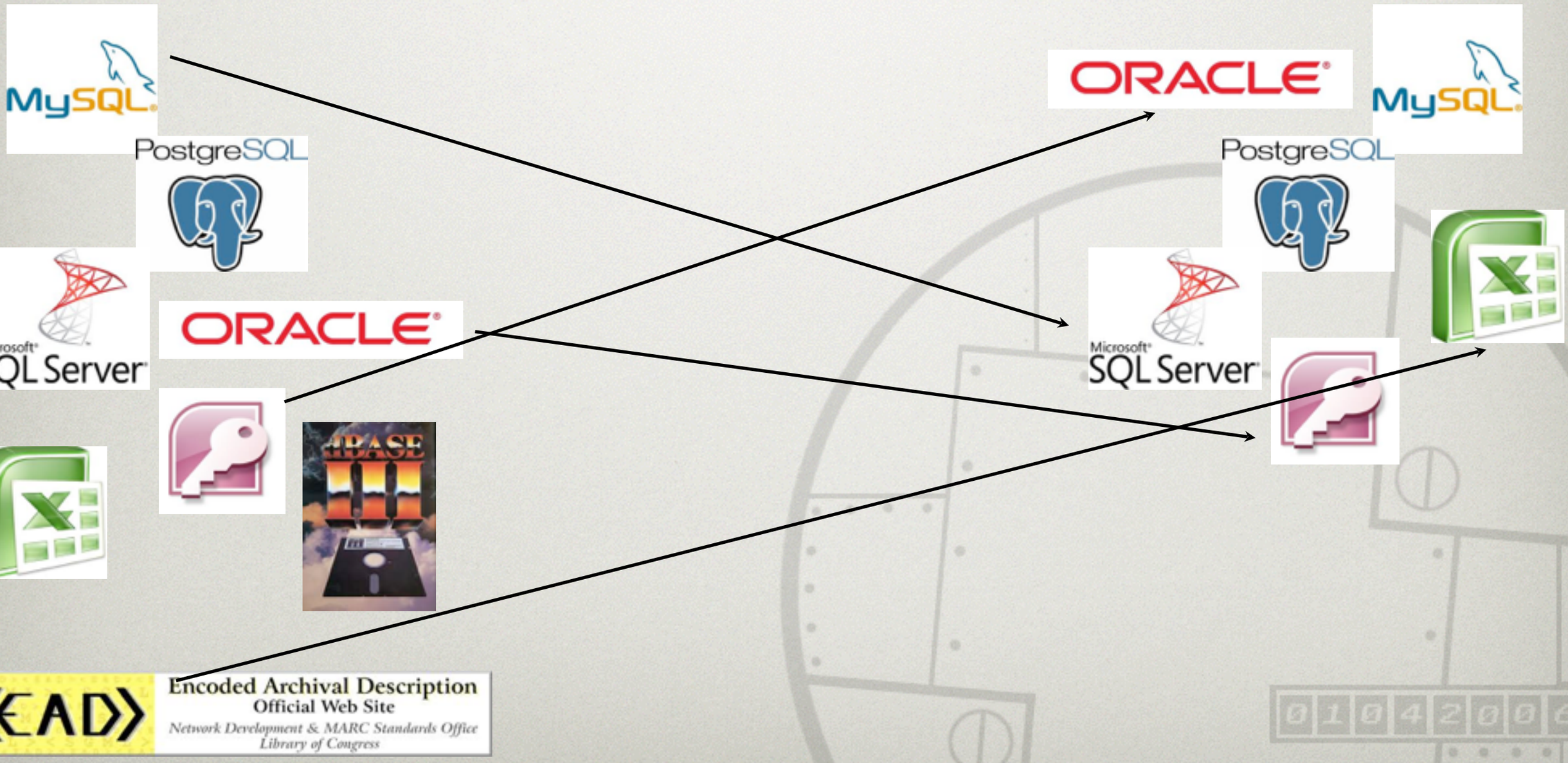
First prototype:

- Data
- Structure
- Only “frozen” DBs

HOW?

The need for an intermediate representation

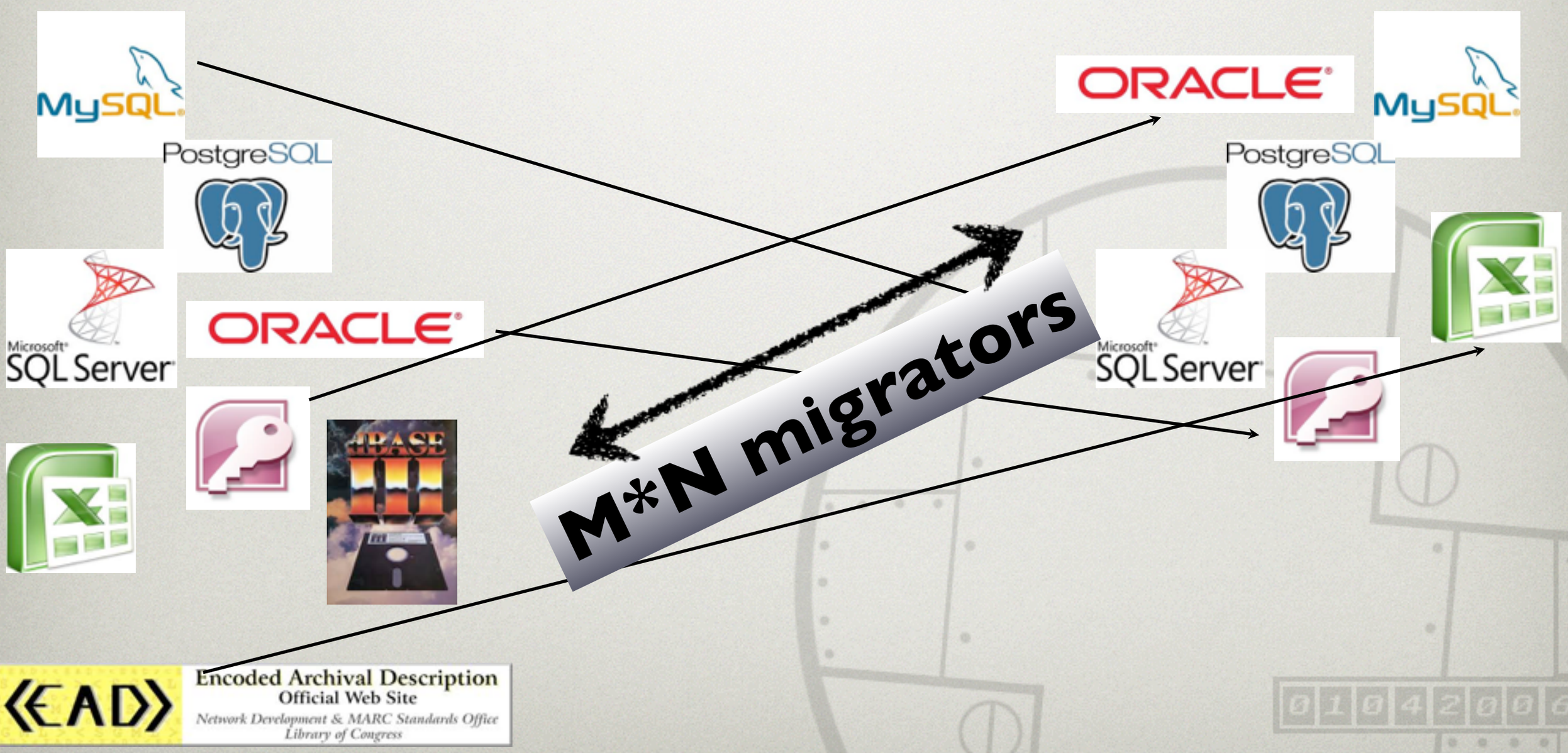
Input Formats (M) **Output Formats (N)**



HOW?

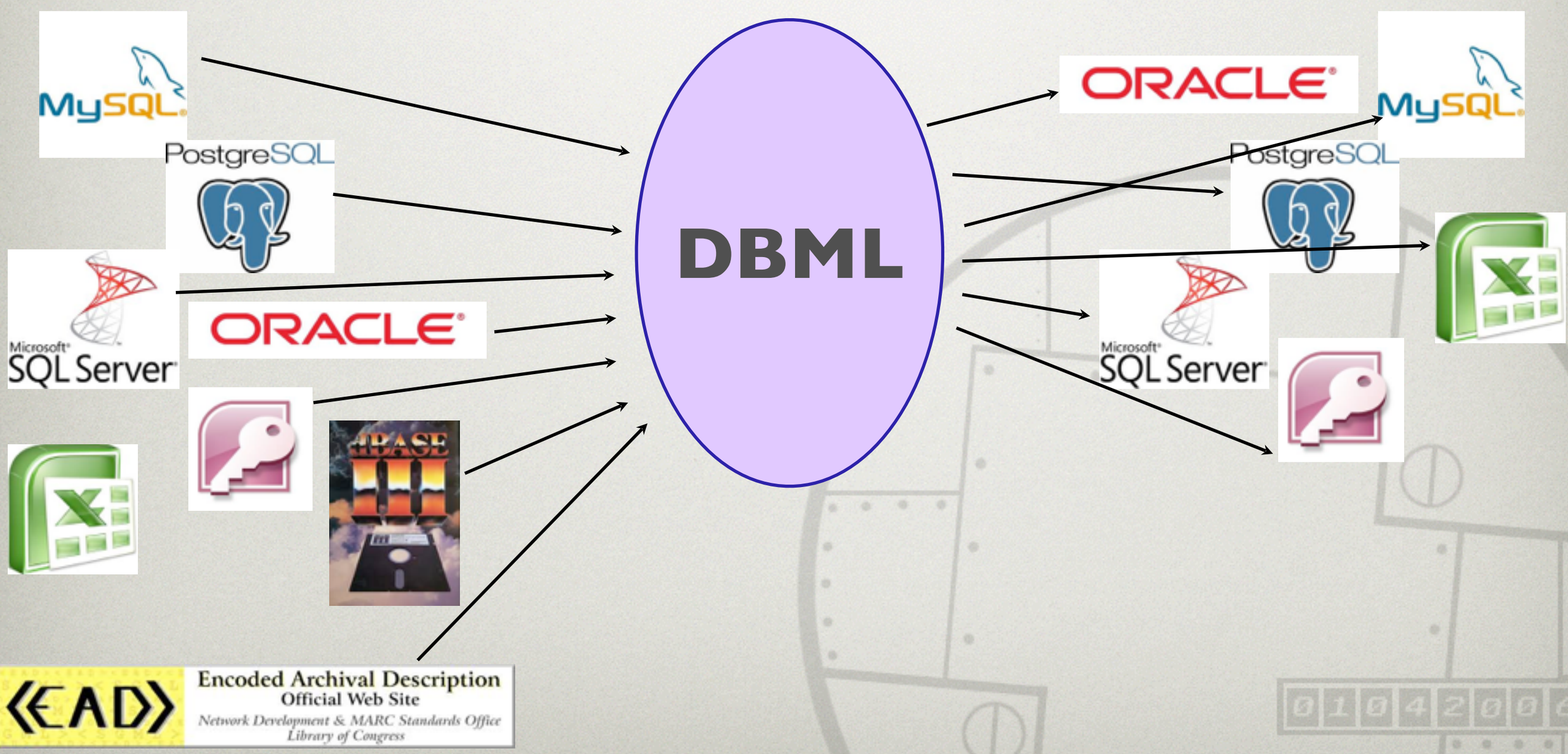
The need for an intermediate representation

Input Formats (M) **Output Formats (N)**



IR: DBML

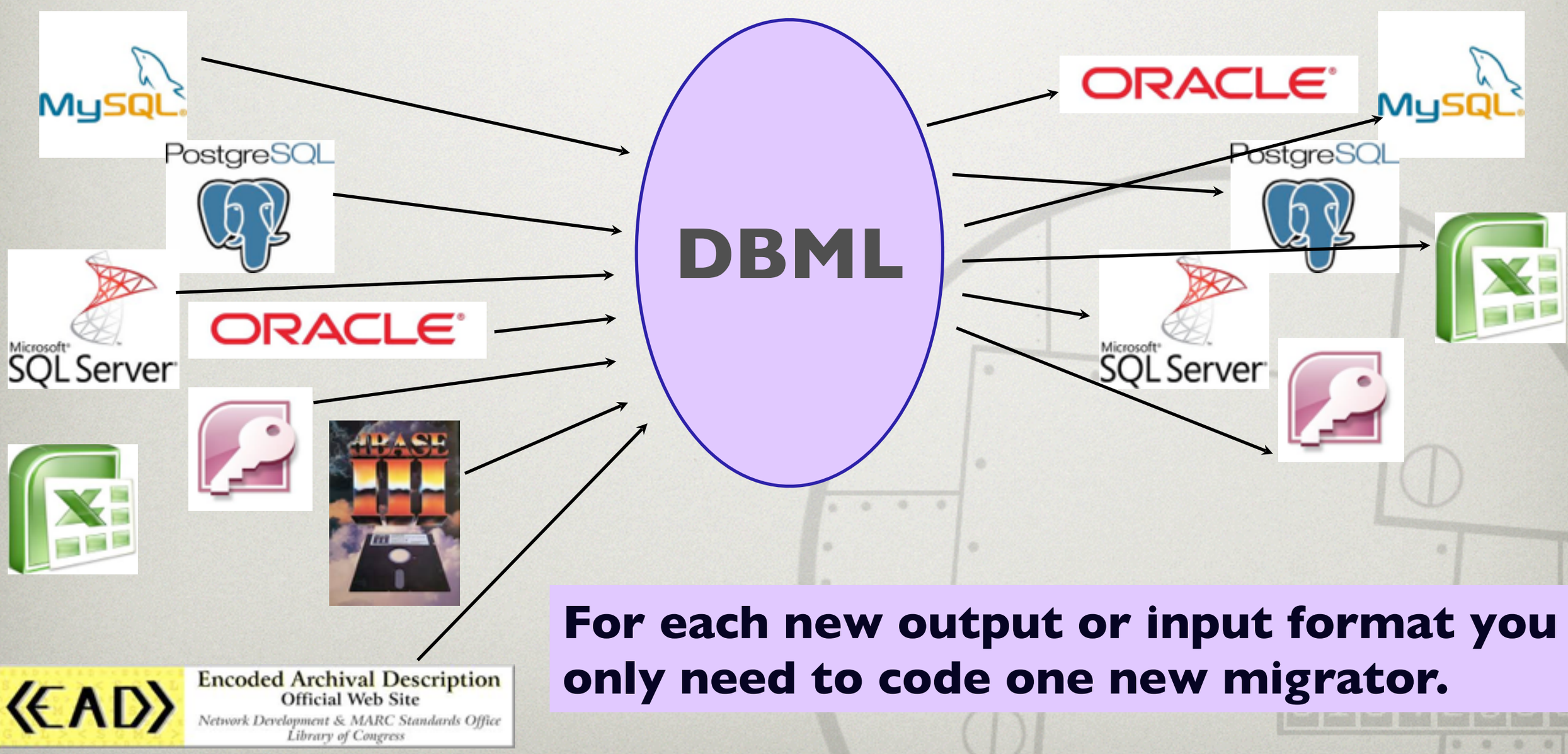
Input Formats (M) Output Formats (N)



«EAD» Encoded Archival Description
Official Web Site
Network Development & MARC Standards Office
Library of Congress

IR: DBML

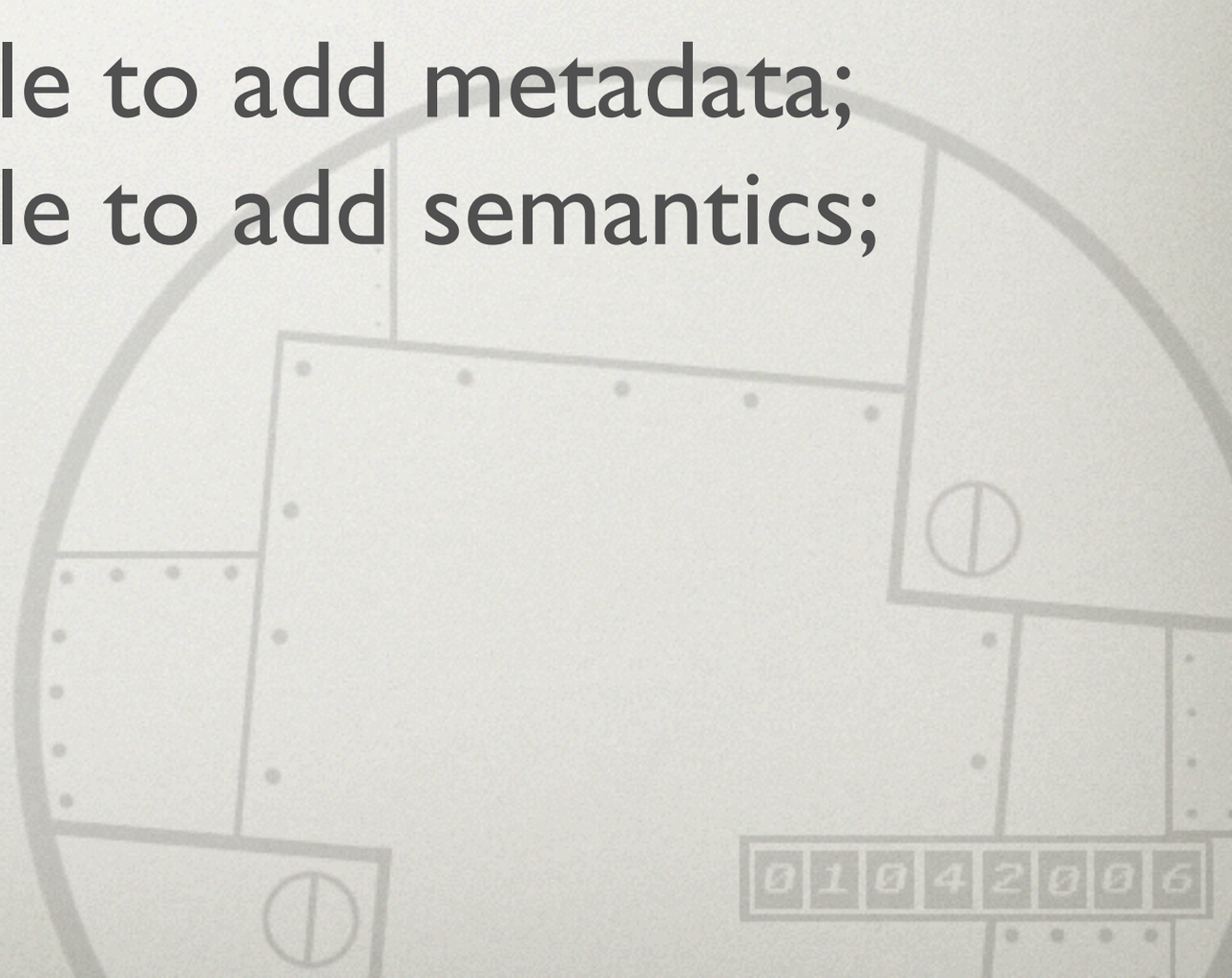
Input Formats (M) Output Formats (N)



For each new output or input format you only need to code one new migrator.

DBML design Principles

- Hardware independent;
- Software independent;
- Easy to process;
- Descriptive;
- It should be possible to add metadata;
- It should be possible to add semantics;

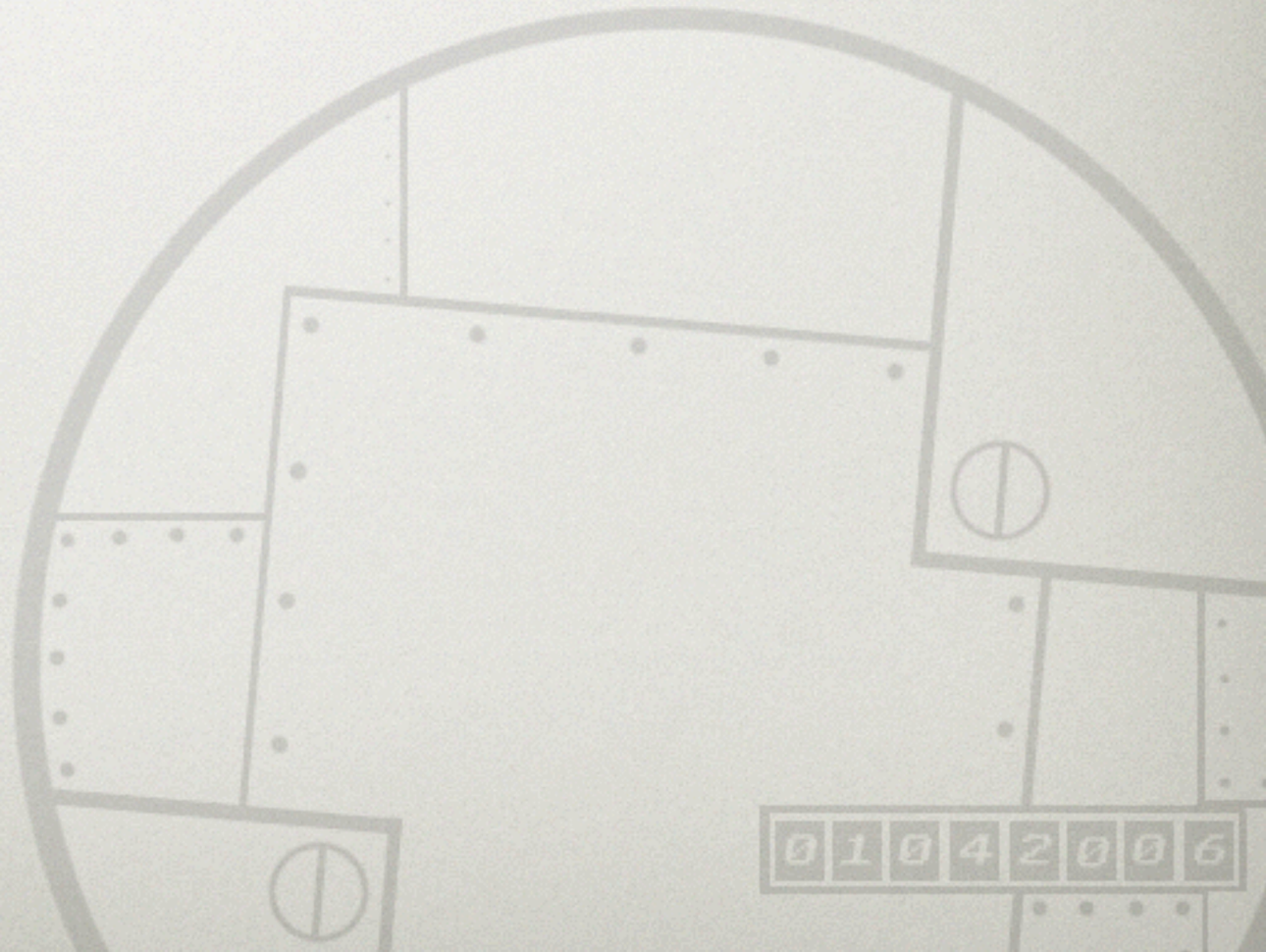


DBML design Principles

- Hardware independent;
- Software independent;
- Easy to process;
- Descriptive;
- It should be possible to add metadata;
- It should be possible to add semantics;

XML was the obvious choice

SIP Structure (DB example)



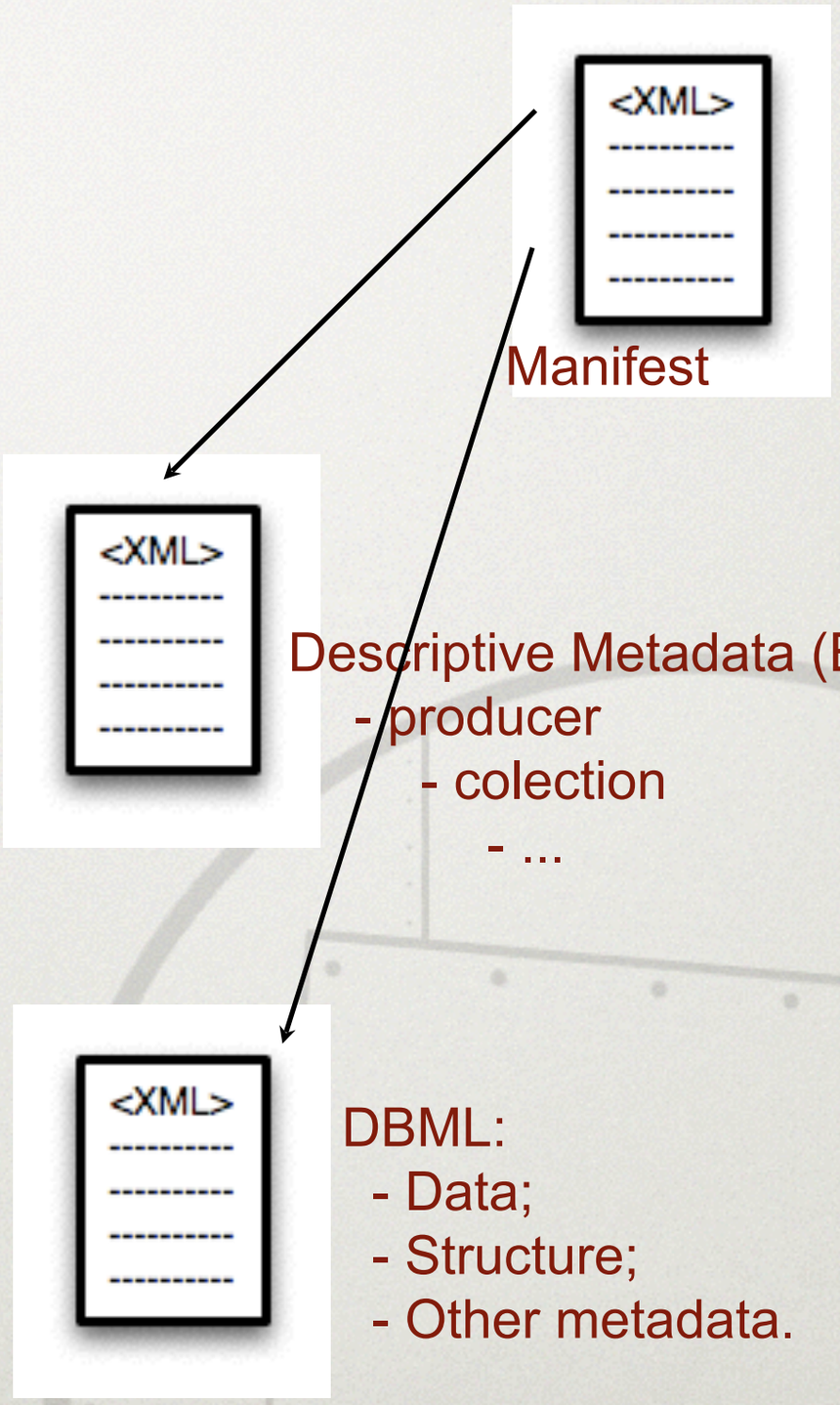
KEEPS Employees

SIP Structure (DB example)

Manifest

KEEPS Employees

SIP Structure (DB example)



SIP Structure (DB example)

Binaries

Descriptive Metadata (EAD):
- producer
  - colection
    - ...

DBML:
- Data;
- Structure;
- Other metadata.

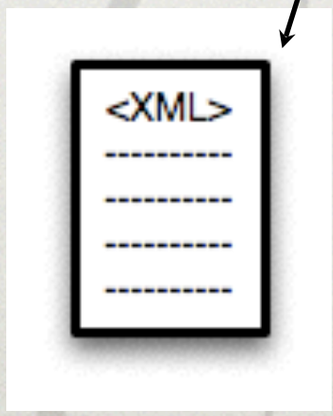
SIP Structure (DB example)

Binaries

Technical Metadata:
- color
- dimensions
- ...

Manifest

Descriptive Metadata (EAD):
- producer
  - colection
    - ...



DBML:
- Data;
- Structure;
- Other metadata.

SIP Structure (DB example)

Binaries

Technical Metadata:
- color
- dimensions
- ...

Manifest

Descriptive Metadata (EAD):
- producer
- colection
- ...

DBML:
- Data;
- Structure;
- Other metadata.

SIP Structure (DB example)

Technical Metadata:
- color
- dimensions
- ...

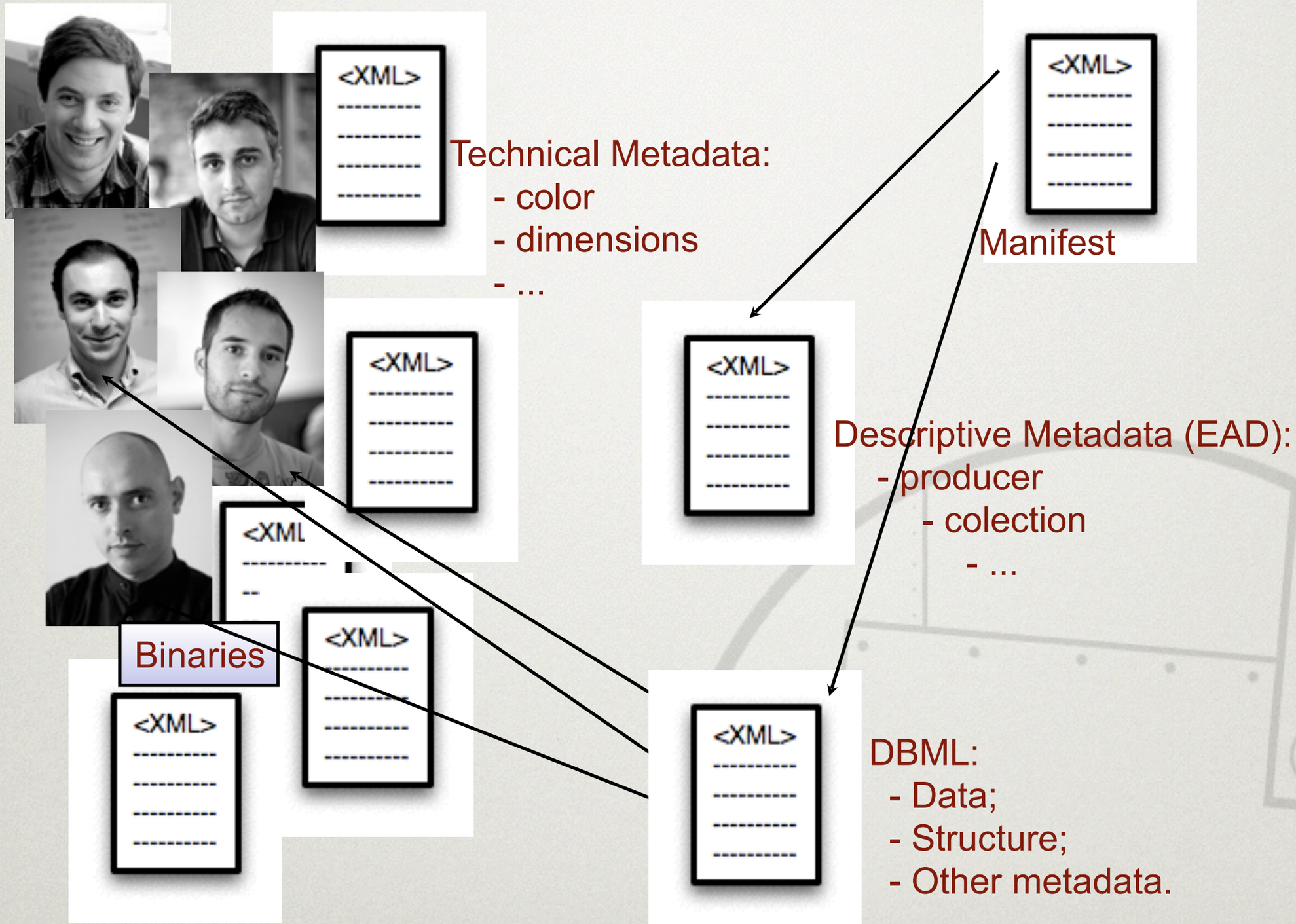
Manifest

Descriptive Metadata (EAD):
- producer
  - colection
    - ...

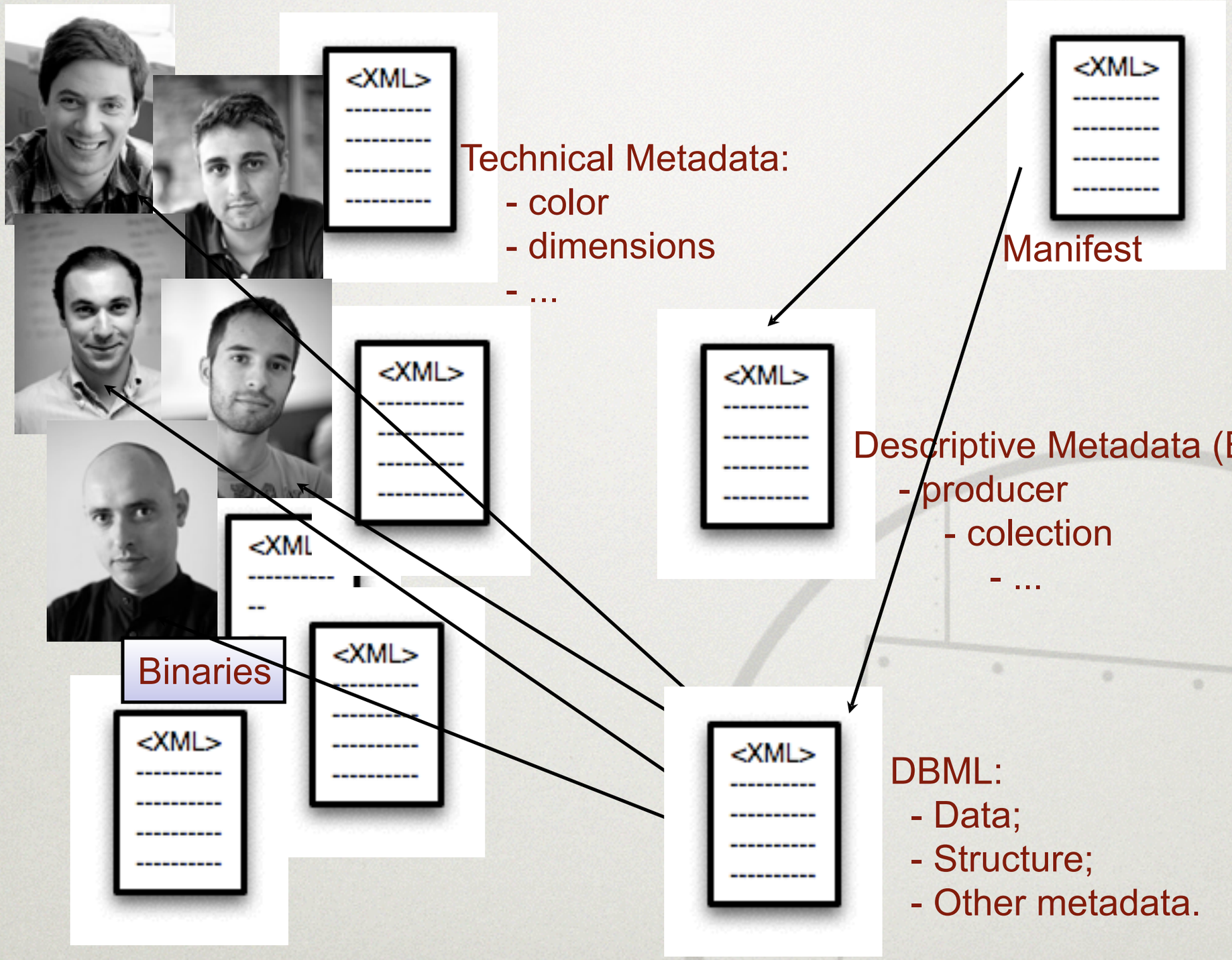
Binaries

DBML:
- Data;
- Structure;
- Other metadata.

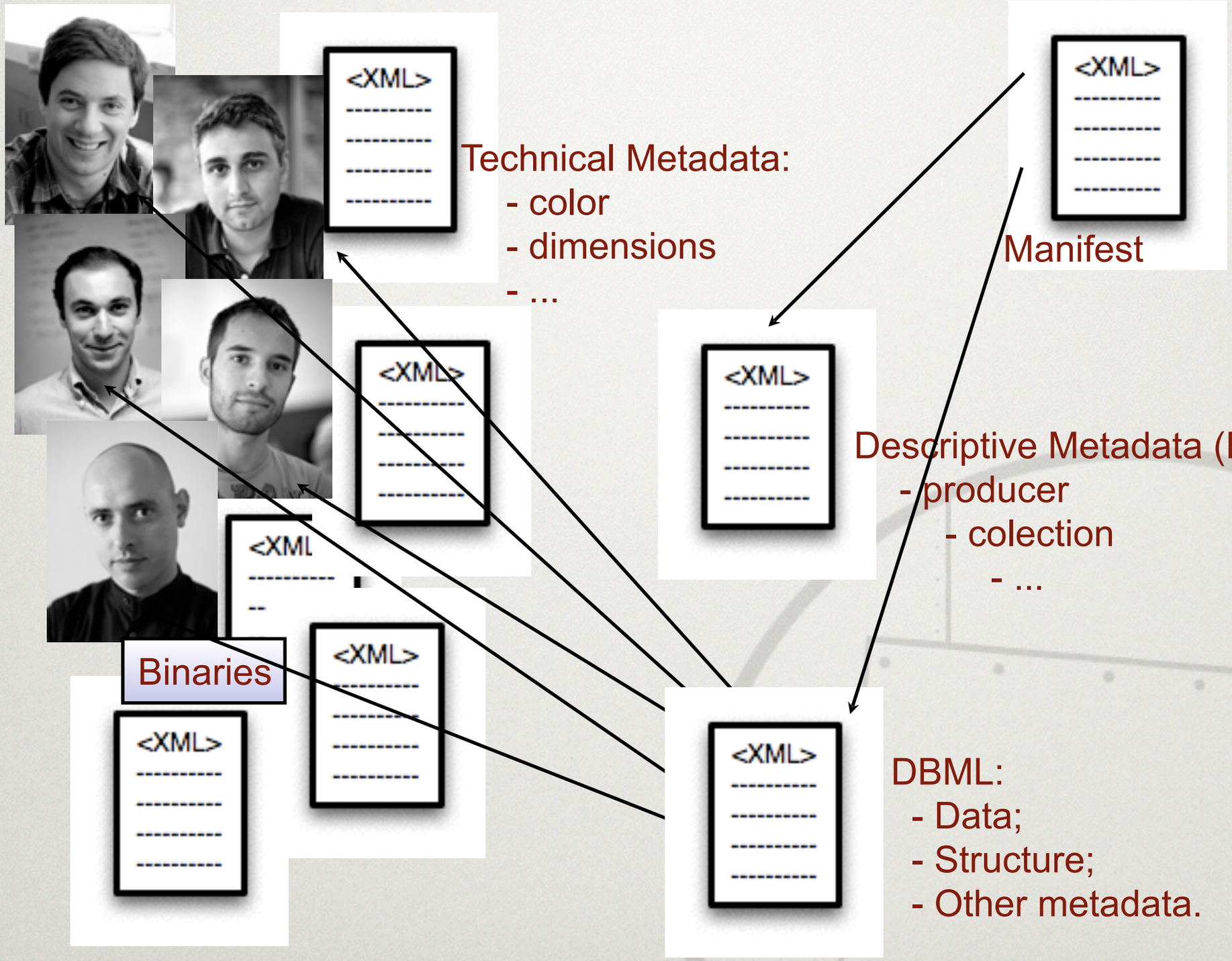
SIP Structure (DB example)



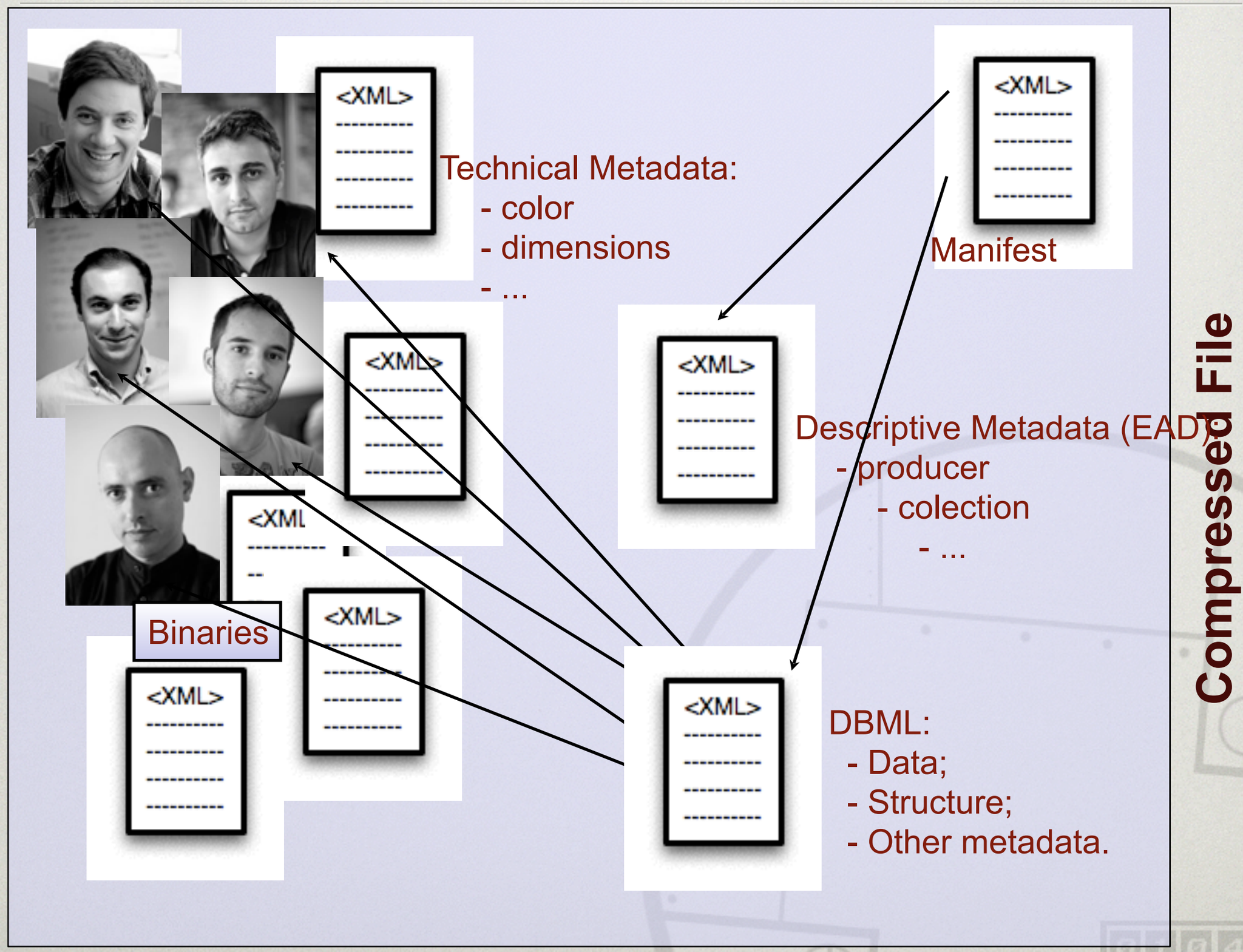
SIP Structure (DB example)



SIP Structure (DB example)

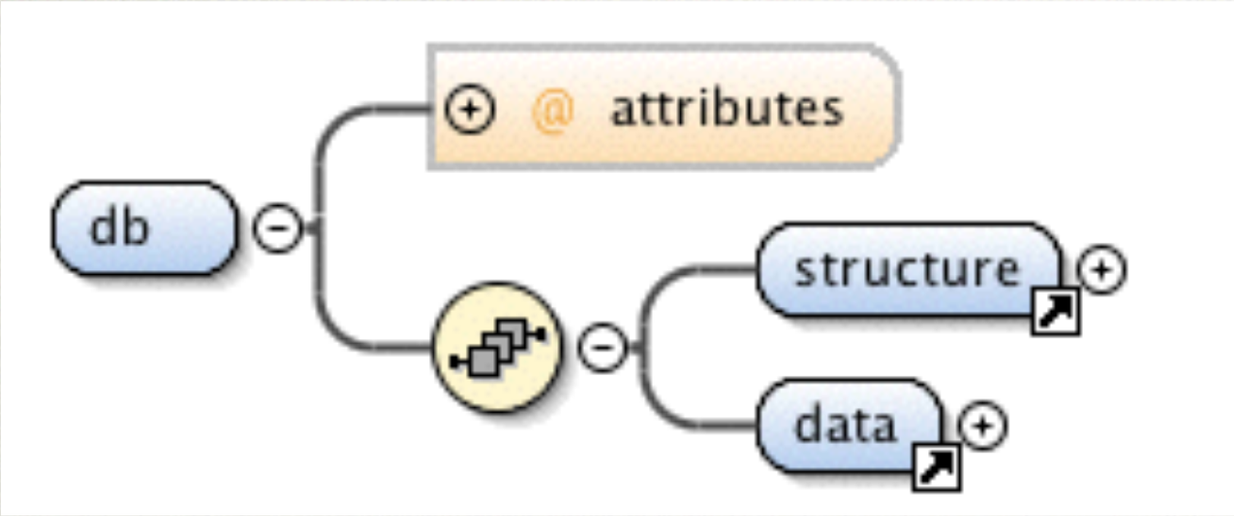


SIP Structure (DB example)

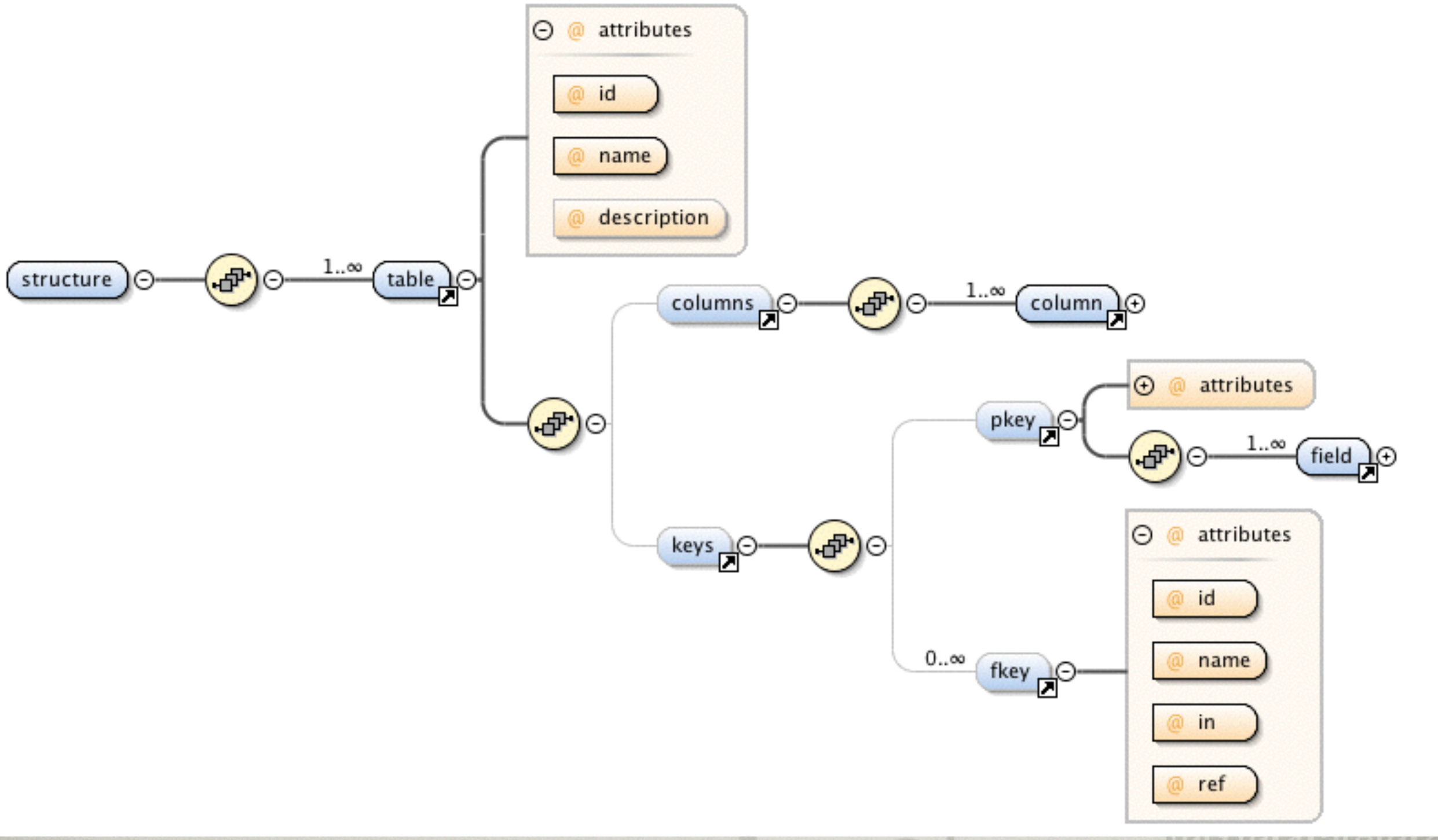


KEEPS Employees

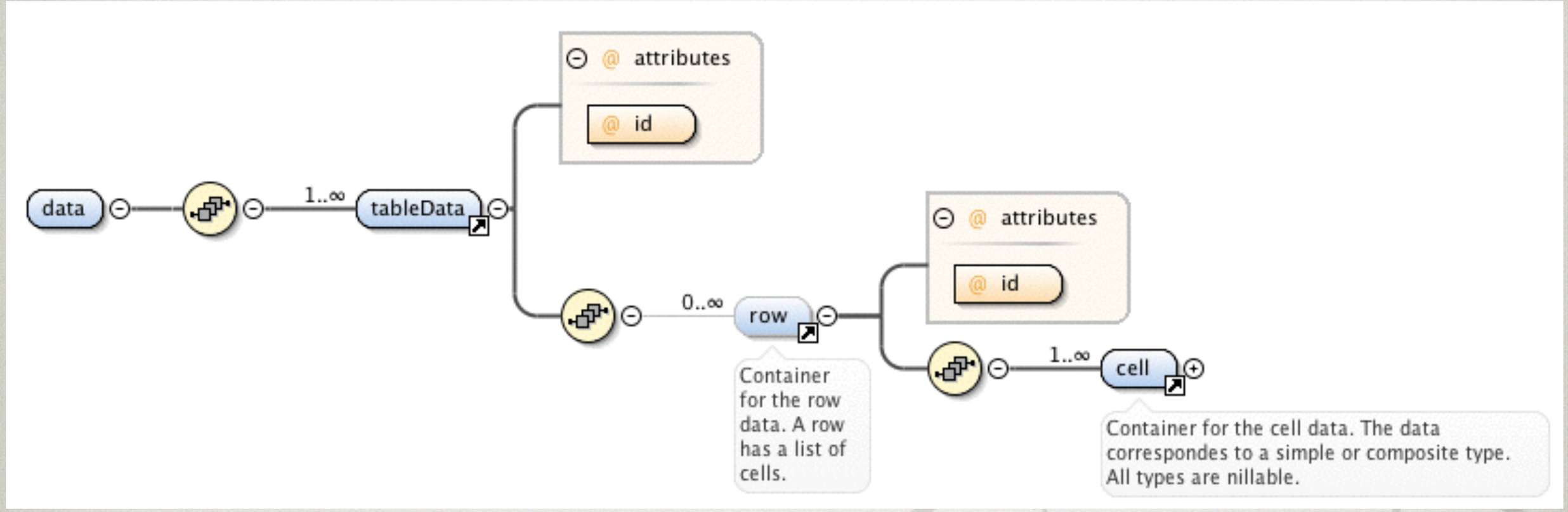
DBML



DBML: Structure



DBML: Data


@ attributes
@ id
data
1..∞
tableData
@ attributes
@ id
0..∞
row
1..∞
cell
Container for the row data. A row has a list of cells.
Container for the cell data. The data correspondes to a simple or composite type. All types are nillable.

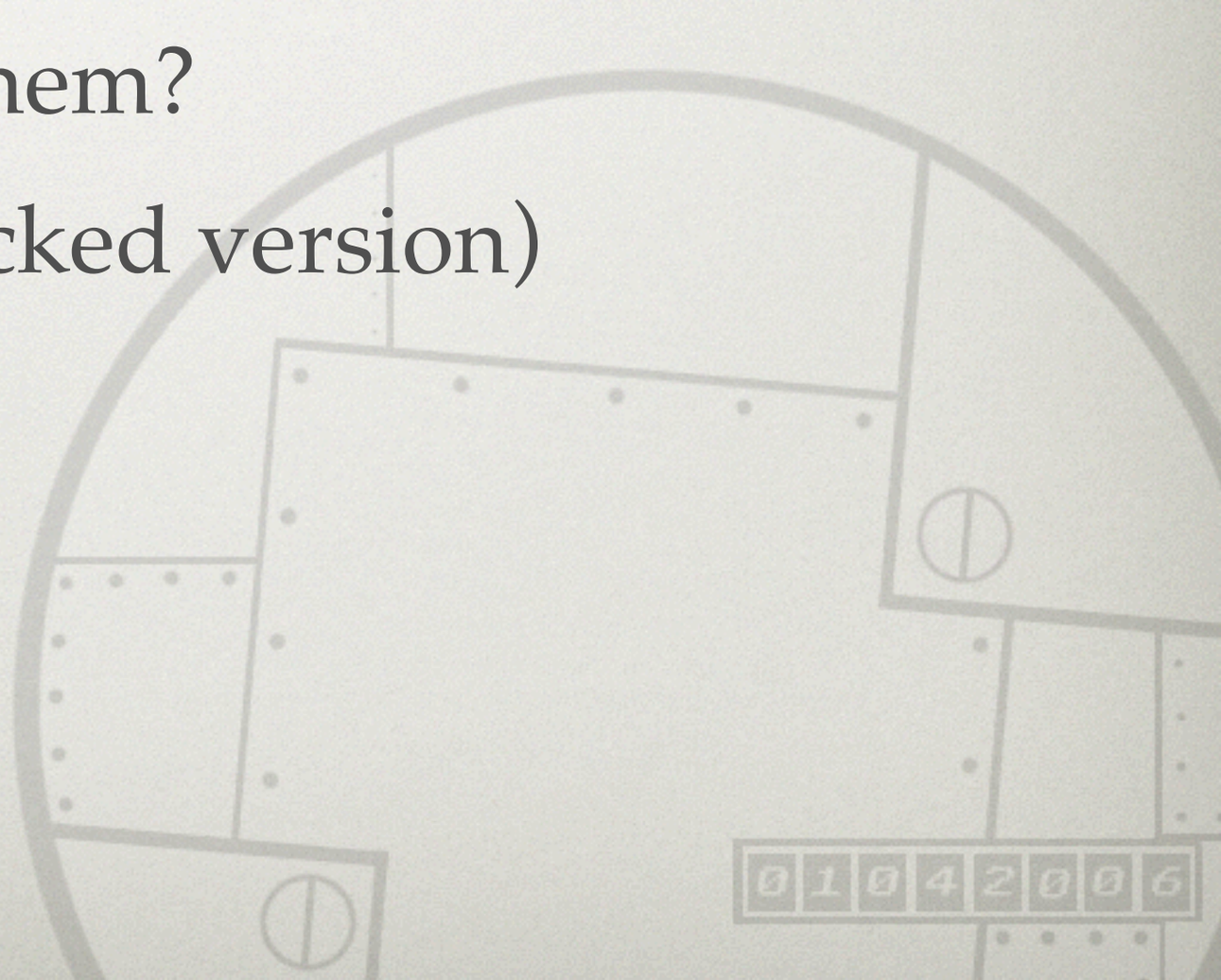
Databases: RODA answers

- How do we store them?

  ★ DBML + binaries + technical metadata

- How do we access them?

  ★ PhpMyAdmin (hacked version)



Databases: RODA answers

- How do we store them?

  ★ DBML + binaries + technical metadata

- How do we access them?

  ★ PhpMyAdmin (hacked version)

RODA answers...

Some problems

- Extracting data is easy:
  - ✦ SELECT * FROM ...
- Extracting the structure is not:
  - ✦ DBMS protect this information;
  - ✦ Each DBMS stores it differently;
  - ✦ Different versions of the same DBMS can also act differently;
  - ✦ We have to “prepare/hack” the DBMS.

Open Planets: “Database Archiving” (Copenhagen 2012)

Two approaches emerged:

- **Preserve the database and the environment allowing authentic access to the information and any accompanying applications**

- **Extract / migrate the raw data and table structure from the original database**

Open Planets: “Database Archiving” (Copenhagen 2012)

Two approaches emerged:

- **Preserve the database and the environment all[illegible] authentic access to the information and any accompanyi[illegible]plications**

Relies on emulation strategy

- **Extract / migrate the raw data and table structure from the original database**

Open Planets: “Database Archiving” (Copenhagen 2012)

Two approaches emerged:

- **Preserve the database and the environment all[illegible] authentic access to the information and any accompanyi[illegible]plications**

Relies on emulation strategy

- **Extract / migrate the raw data and table str[illegible]re from the original database**

Relies on migration strategy

Open Planets: “Database Archiving” (Copenhagen 2012)

Two approaches emerged:

- **Preserve the database and the environment all[illegible] authentic access to the information and any accompanyi[illegible]plications**

Relies on emulation strategy

- **Extract / migrate the raw data and table str[illegible]re from the original database**

RODA+DBML

Relies on migration strategy

Open Planets: “Database Archiving” (Copenhagen 2012)

Two approaches emerged:

- **Preserve the database and the environment all[illegible] authentic access to the information and any accompanyi[illegible]plications**

Relies on emulation strategy

- **Extract / migrate the raw data and table str[illegible]re from the original database**

RODA+DBML

SIARD

Relies on migration strategy

Open Planets: “Database Archiving” (Copenhagen 2012)

Two approaches emerged:

- **Preserve the database and the environment all[illegible] authentic access to the information and any accompanyi[illegible]plications**

Relies on emulation strategy

- **Extract / migrate the raw data and table str[illegible]re from the original database**

RODA+DBML

SIARD

SIARD-DK

Relies on migration strategy

Open Planets: “Database Archiving” (Copenhagen 2012)

Two approaches emerged:

- **Preserve the database and the environment all[illegible] authentic access to the information and any accompanyi[illegible]plications**

Relies on emulation strategy

- **Extract / migrate the raw data and table str[illegible]re from the original database**

RODA+DBML

SIARD

SIARD-DK

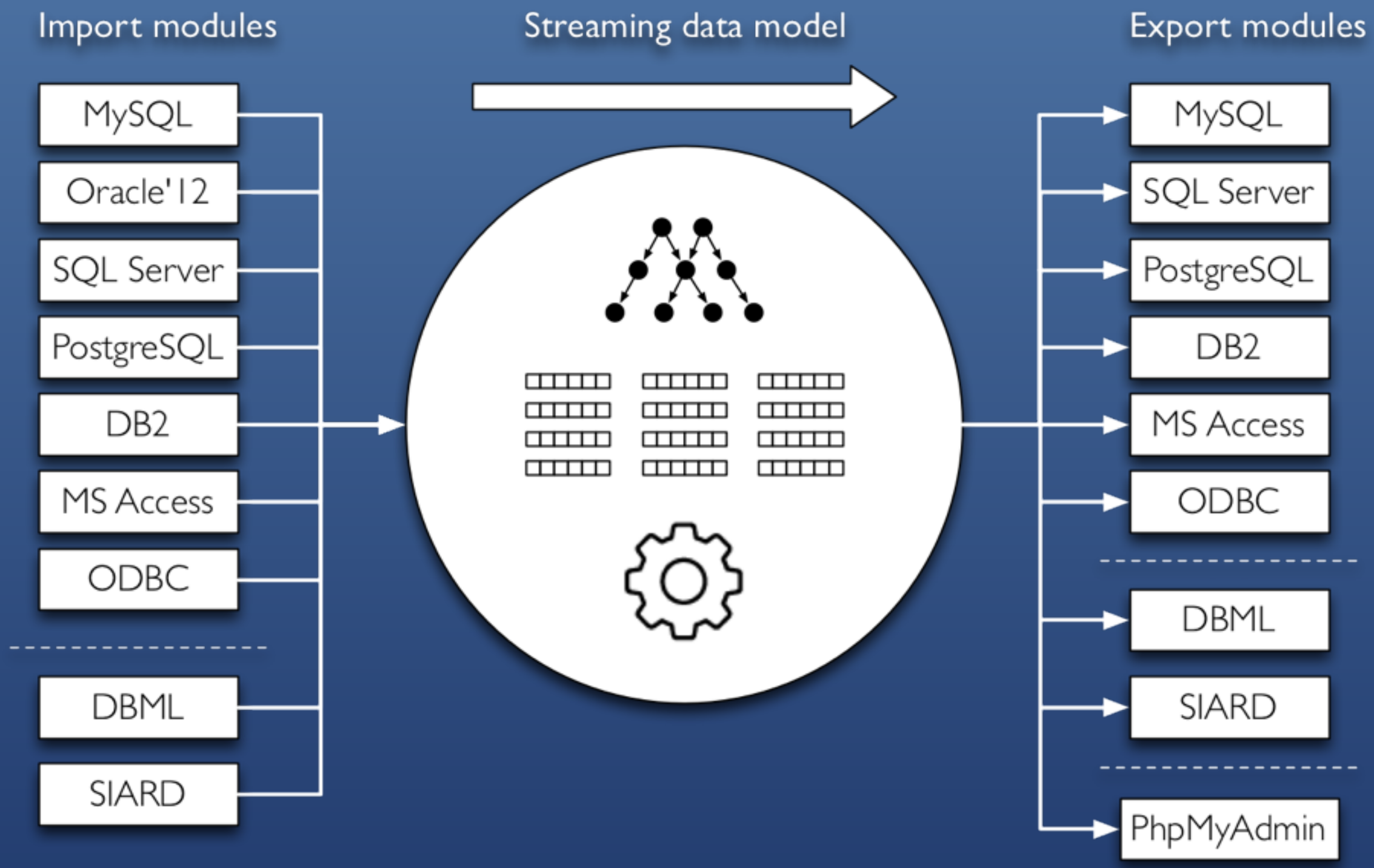
ADDML

Relies on migration strategy

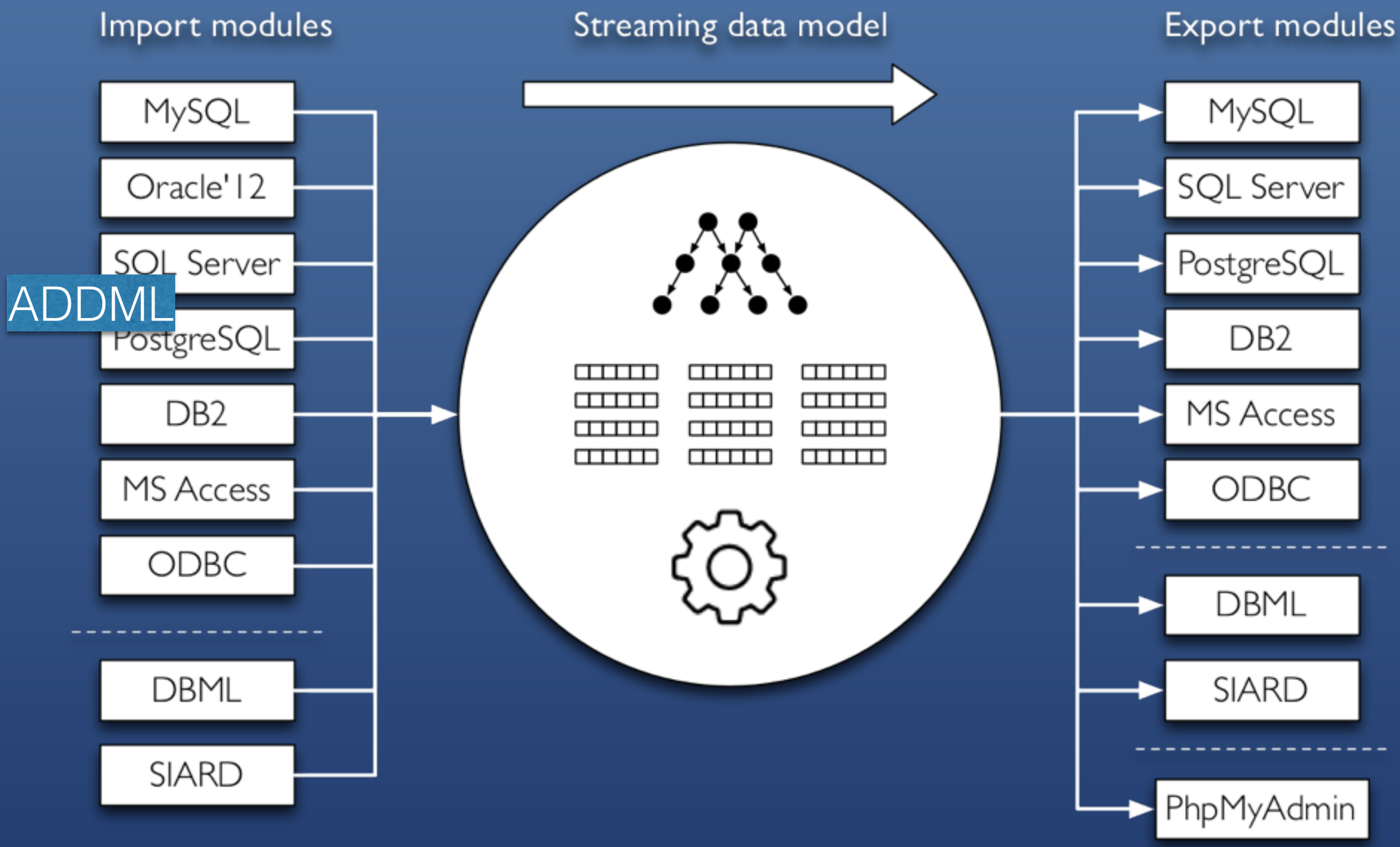
DBML vs SIARD

	DBML	*SIARD*
Data	*Yes (no segmentation)*	*Yes (with segmentation)*
Structure	*Yes (XML)*	*Yes (XSD)*
Stored Procedures	*No*	*Yes (XML)*
Triggers	*No*	*Yes (XML)*
Views	*No*	*Yes (XML)*

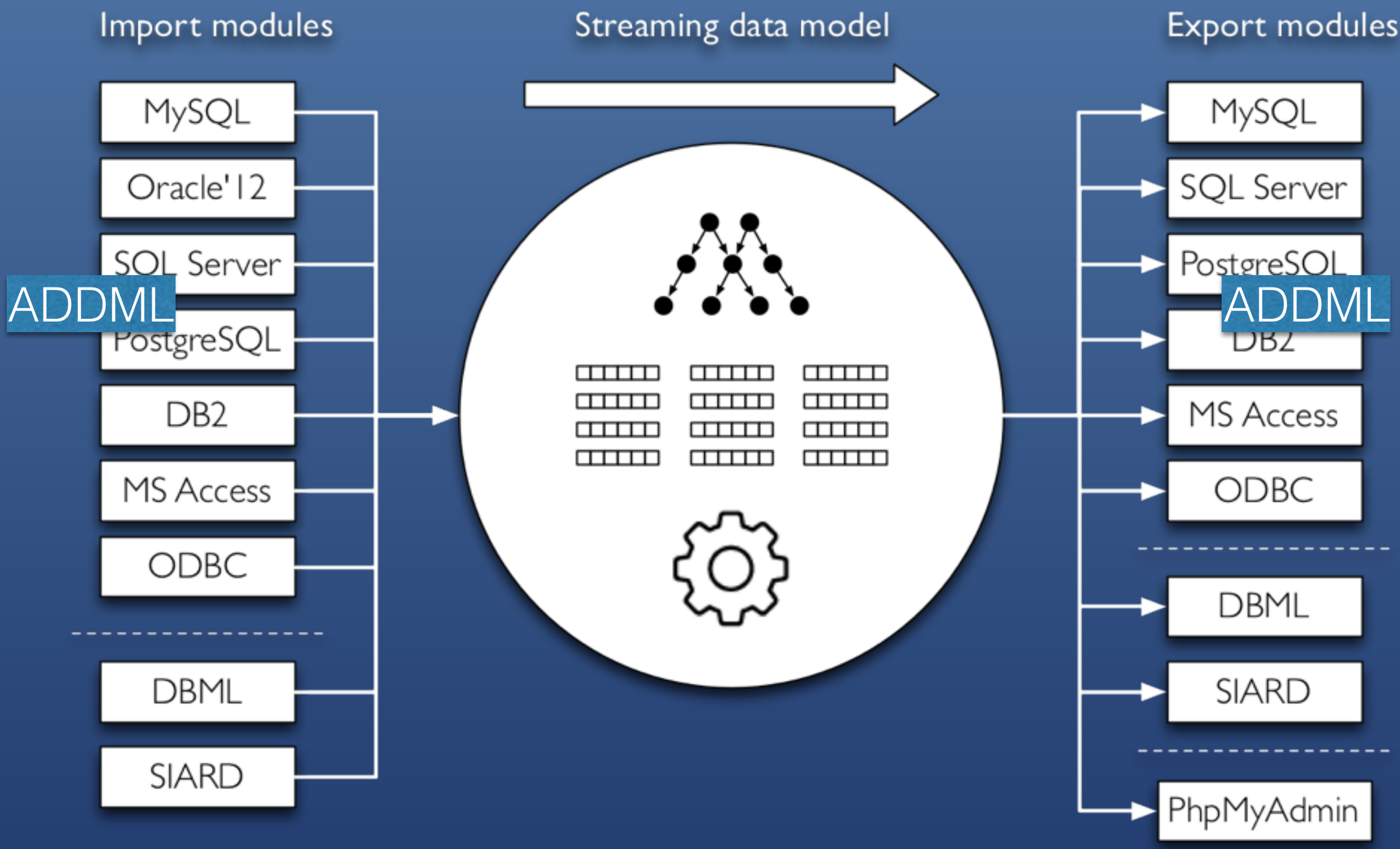
db-preservation-toolkit architecture



db-preservation-toolkit architecture



db-preservation-toolkit architecture

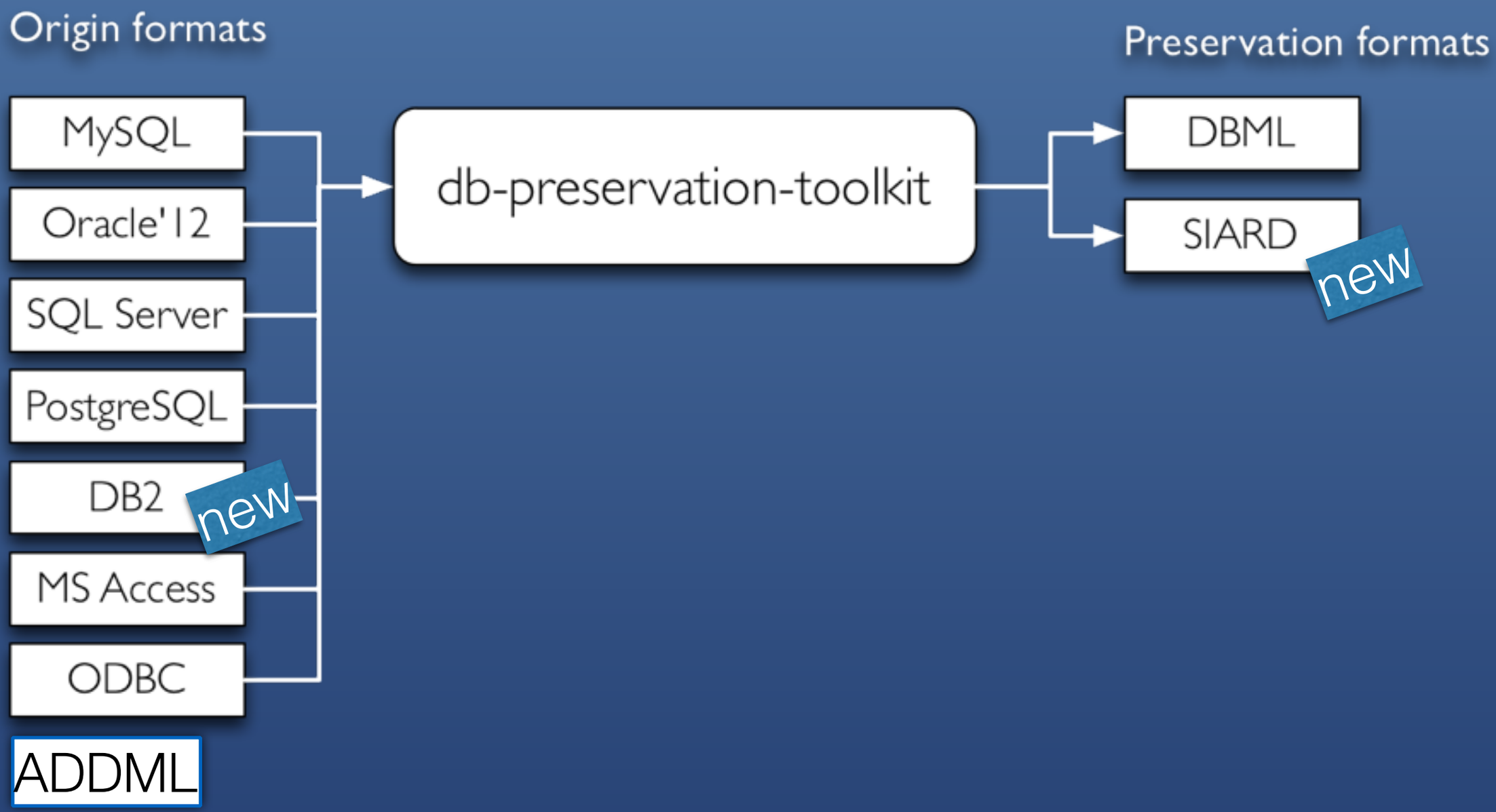


Pre-ingest / Production / SIP creation

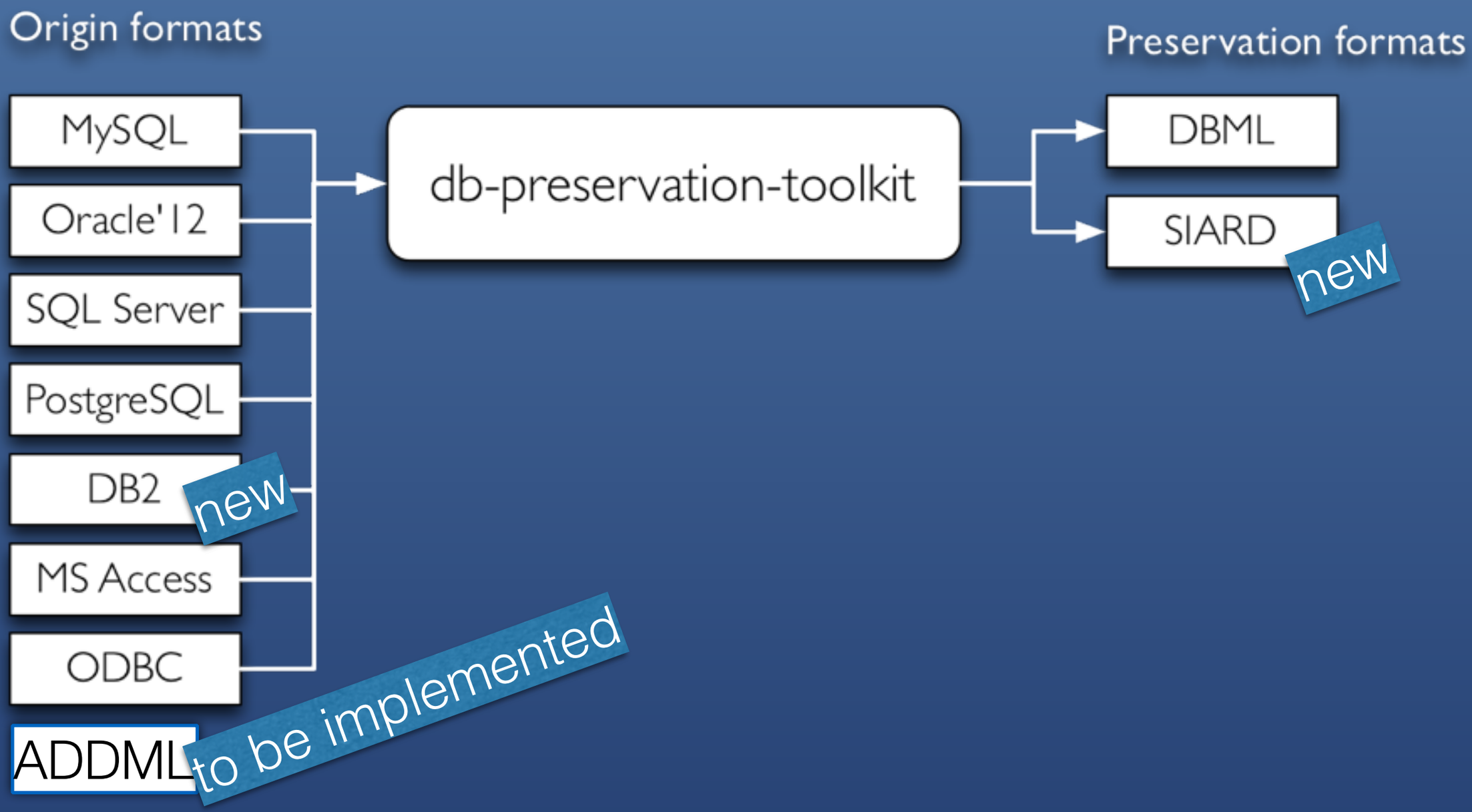
Pre-ingest / Production / SIP creation

Pre-ingest / Production / SIP creation

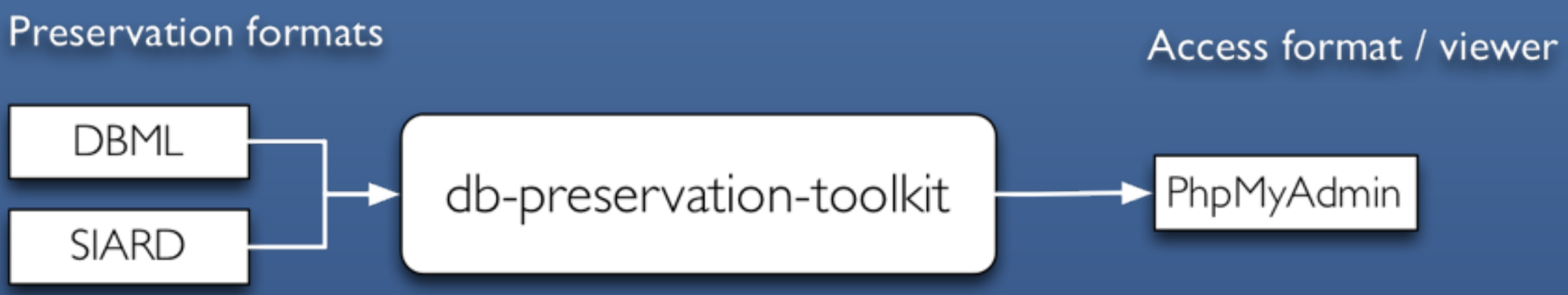
Pre-ingest / Production / SIP creation



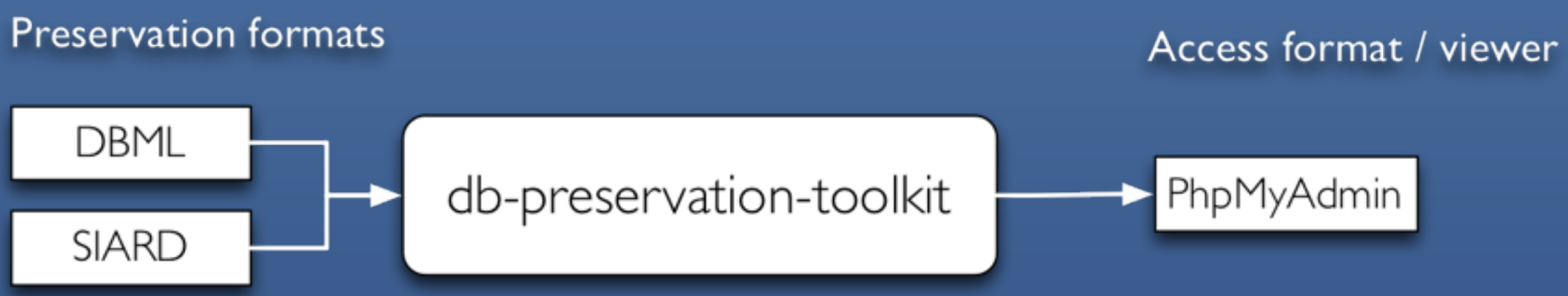
Pre-ingest / Production / SIP creation



Access


Preservation formats
Access format / viewer
DBML
SIARD
db-preservation-toolkit
PhpMyAdmin

Access



Working on RDF viewer, we don’t need to stay stuck in the relational model

New prototype

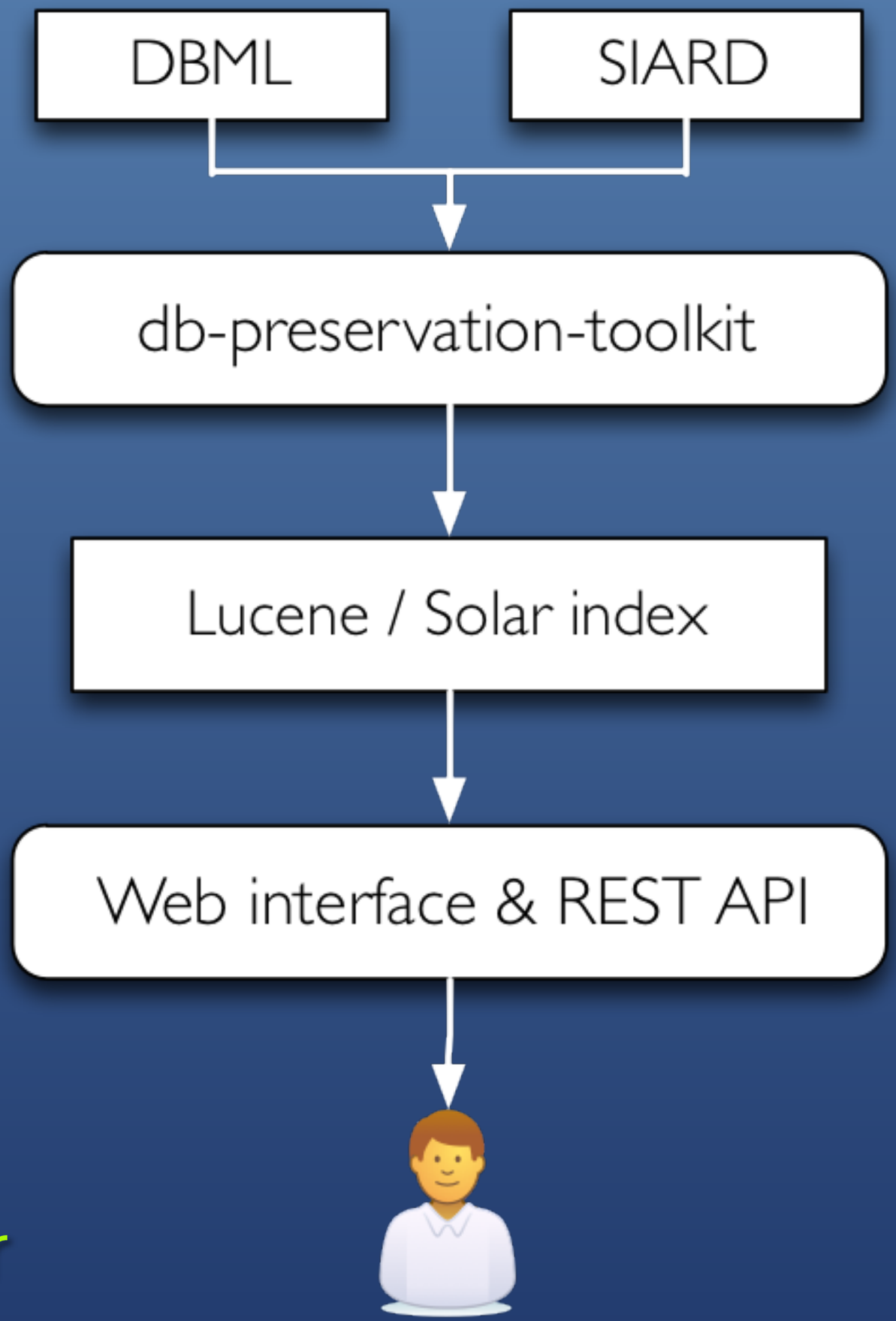
Msc thesis: *"Database Convert Tool: exploring SIARD"*

- *OWL model development*
- *Triple Store Implementation*
- *RDF endpoint implementation*
- *SPARQL based access*

Preservation formats

Fast viewer resources

Fast viewer application



Future plan:
Fast database viewer

Recent work

Phd thesis: Graph view dissemination, relational model reverse engineering, algorithm development (from relational to graph model) [http://repositorium.sdum.uminho.pt/handle/1822/25655]

Msc thesis: SIARD support (DBML replacement)

Msc thesis: RDF representation for databases

Questions?

José Carlos Ramalho

Engineer / Researcher / Teacher

jcr@keep.pt / jcr@di.uminho.pt

KEEPSOLUTIONS

University of Minho SPIN-OFF

ARQUIVOS | BIBLIOTECAS | MUSEUS

www.keep.pt