DaMaLOS 2020

Towards Semantic Representation of Machine-Actionable Data Management Plans

João Cardoso

Leyla J. Garcia

Fajar J. Ekaputra

Marie C. Jacquemot

Tomasz Miksa

José Borbinha

DMP OPIDoR

cnrs Inist Institut de l'information scientifique et technique

ZB MED Information Centre for Life Sciences

SBA Research

TU WIEN

inesc id lisboa

TÉCNICO LISBOA

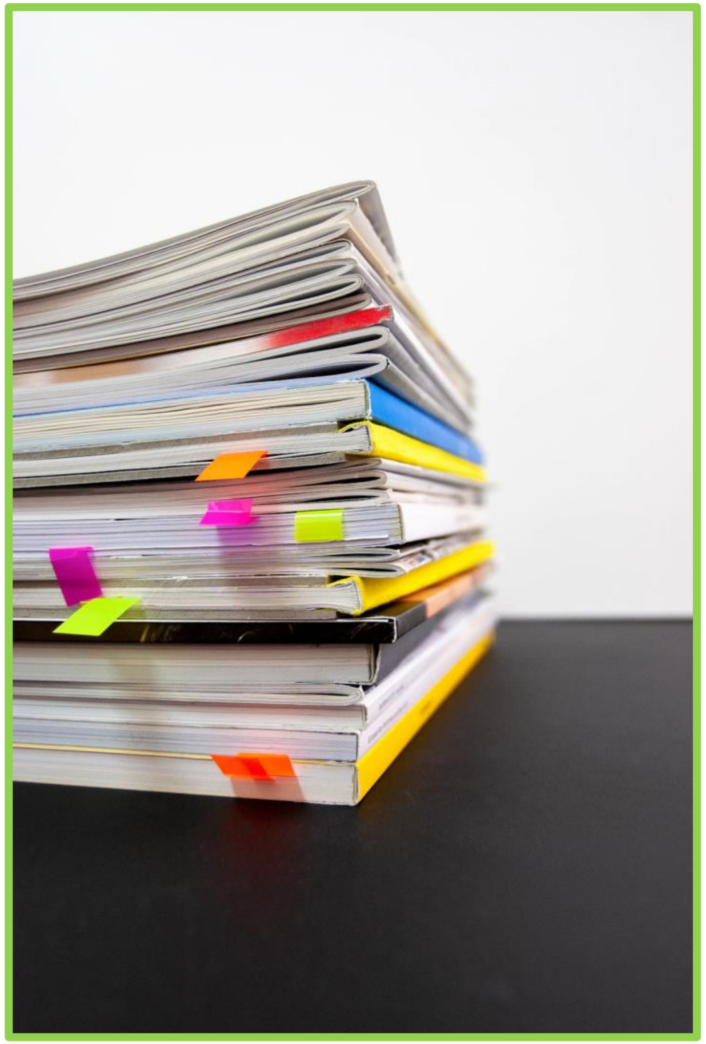
1. Concepts

1.1. What is a Data Management Plan (DMP)?

- A DMP is a **formal document** used to support **Data Management**.

- The DMP describes the **techniques**, **methods** and **policies** on how data should be:
  - Created or Collected
  - Documented
  - Accessed
  - Preserved and Disseminated.

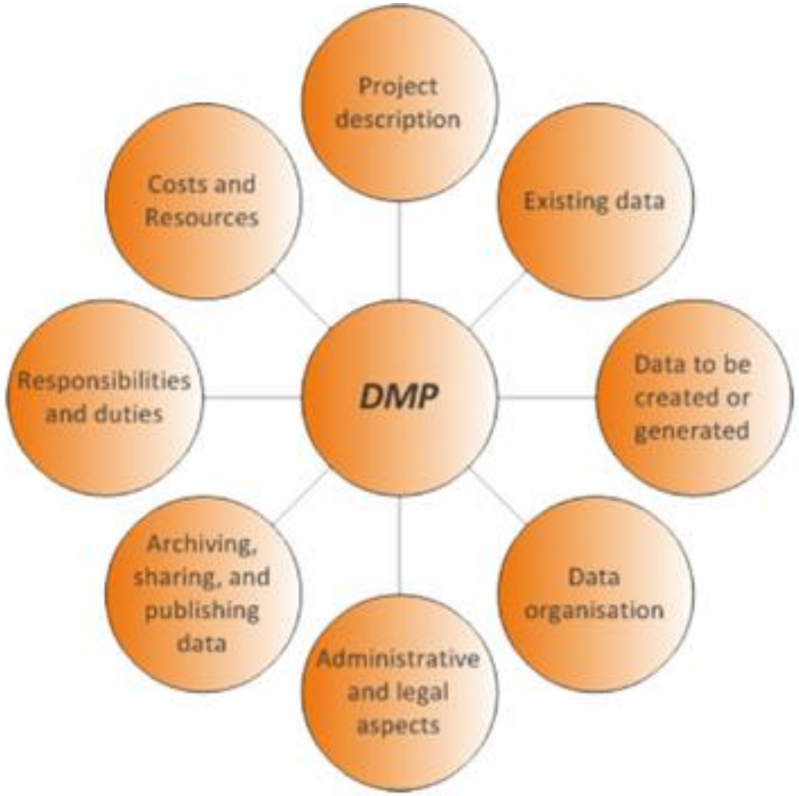
- A DMP should be **created at the start** of a project and **updated** throughout its life-cycle.



1. Concepts

1.2. What should be in a DMP?

- A DMP should contain information on the following topics:
  - **Administrative Data**
    - Staff, responsibilities, funding, etc.
  - **Data**
    - Dataset characterization, formats, metadata standards, technical resources, etc.
  - **Preservation**
    - Dissemination policies, data hosts, licenses, etc.
  - **Costs**
    - Estimates for costs associated with data management.


Project description
Costs and Resources
Existing data
Responsibilities and duties
DMP
Data to be created or generated
Archiving, sharing, and publishing data
Data organisation
Administrative and legal aspects

https://en.uit.no/ub/forskningsstotte/art?p_document_id=473665

1. Concepts

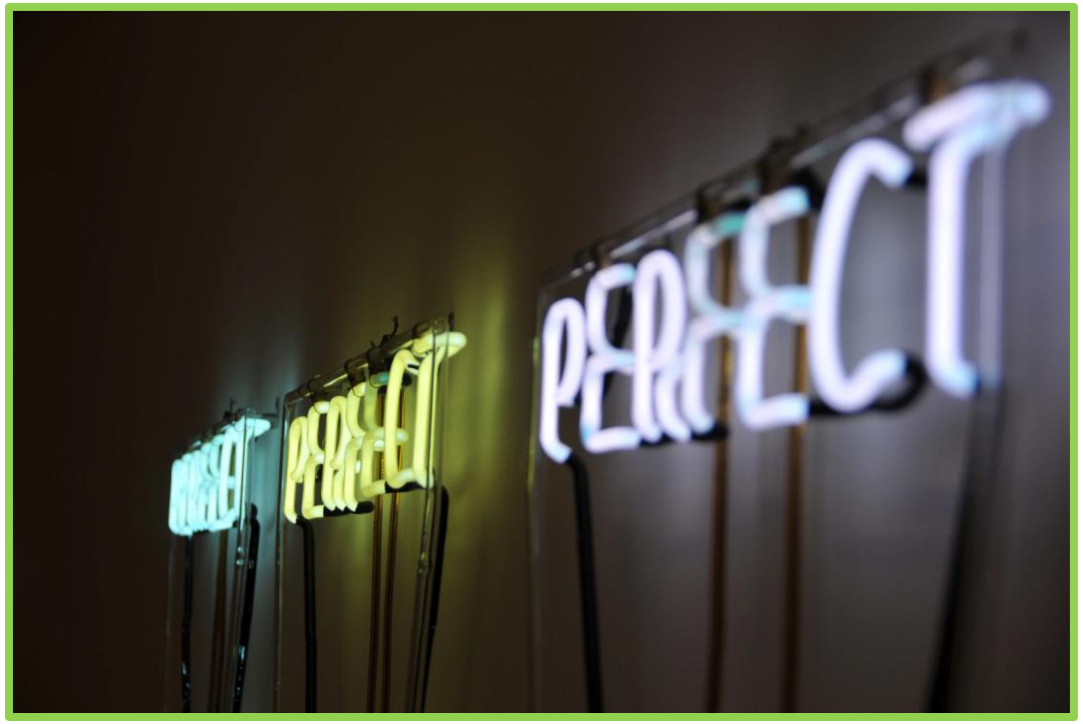
1.2. What should be in a DMP?

- The current DMP is:
  - A mostly **static document.**
  - **Only human** readable.
  - **Based on a template**, provided by the funding agency.
  - **Not published,** or publicly accessible.
  - Rarely **updated**.
  - Considered a **bureaucratic hassle**.

1. Concepts

1.2. What should be in a DMP?

- The ideal DMP should be:
  - Both **machine** and **human readable.**
  - **Shareable**.
  - Compliant with a **standard.**
  - **Interoperable.**
  - A **living document.**
  - An essential part of **data management**.



1. Concepts

1.3. The DMP Common Standard

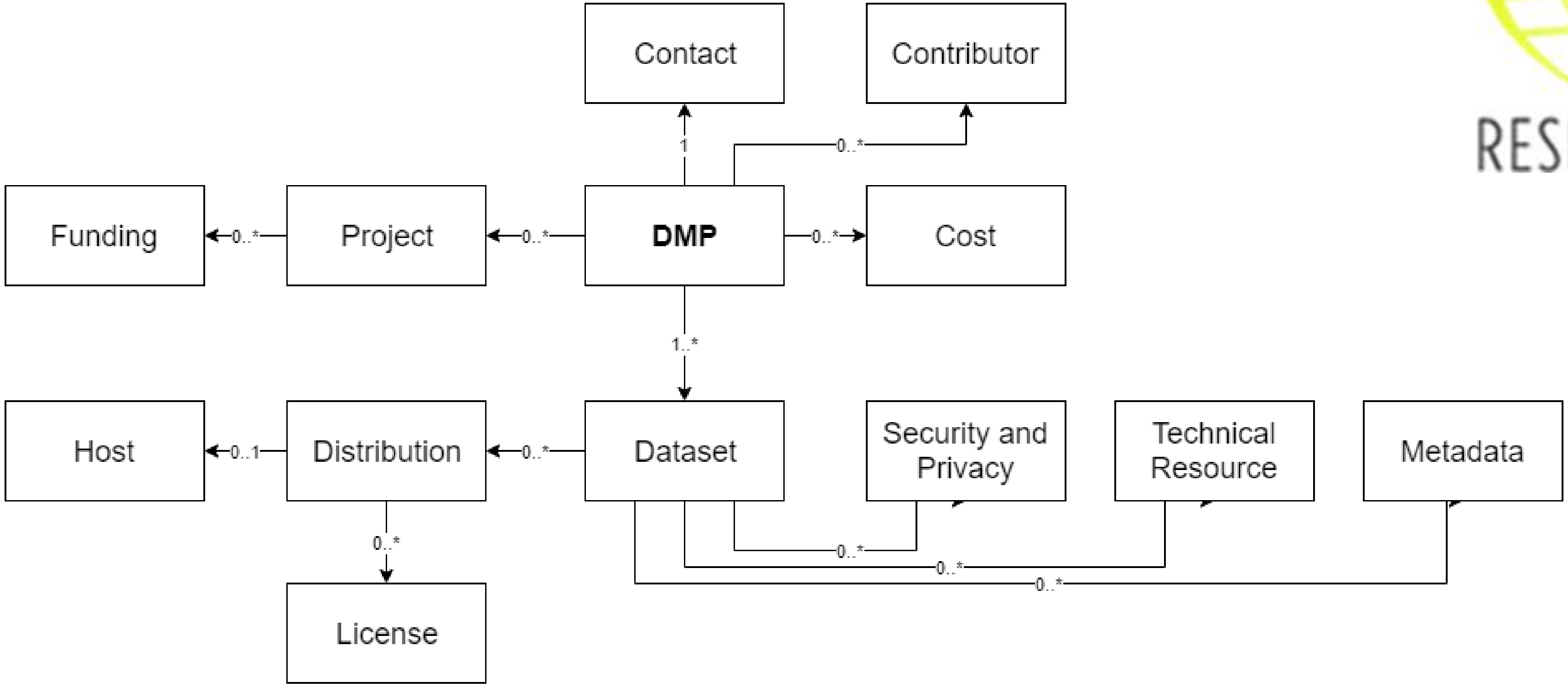
- The RDA **DMP Common Standards (DCS)** Working Group was created to focus on the **standardization of knowledge contained in a DMP.**

- Its objective was to establish an **application profile** that defines a **core set of terms that define a DMP**.

- The application profile is **modular in design** and allows for **extensions**.

1. Concepts

1.3. The DMP Common Standard



Contact

Contributor

Funding

Project

DMP

Cost

Host

Distribution

Dataset

Security and Privacy

Technical Resource

Metadata

License

https://github.com/RDA-DMP-Common/RDA-DMP-Common-Standard

2. Ontology Engineering Process

2.1. Why an ontology?

- The DCS provides **reference serialisations** of the application profile.

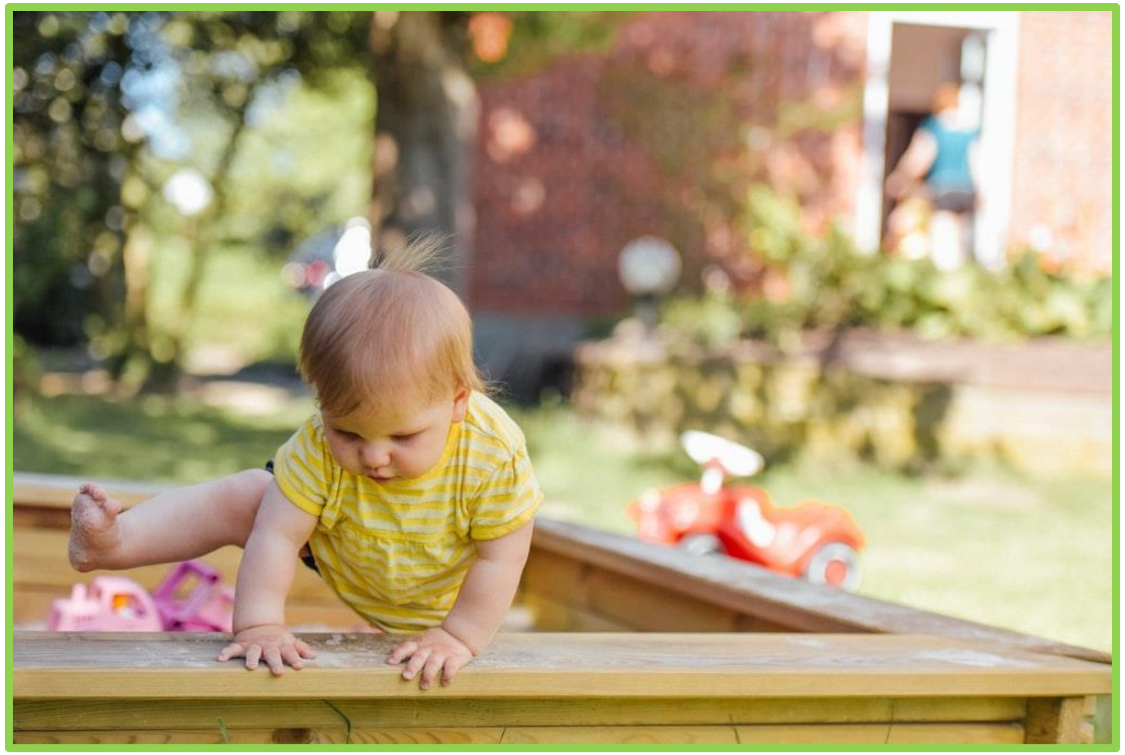
- Our objective was to create a new serialisation with distinct features from the existing serialisations.
  - **Semantic Technologies**
  - **The DMP Common Standard Ontology (DCSO)**

- **Ontologies** allow for the r**epresentation of a shared conceptualisation** of knowledge through the usage of **formal semantics**.
  - Suitable for the creation of **Linked Open Data**
  - Easy to **extend**
  - Enable **reasoning**, and **knowledge inference**

2. Ontology Engineering Process

2.2. The DCSO Baby Steps

- **Initial versions** of the DCSO had several **issues** that prevented it from achieving its full potential.
    - Constraints
    - Controlled vocabularies
    - Ontology reuse
    - Non persistent namespaces



2. Ontology Engineering Process

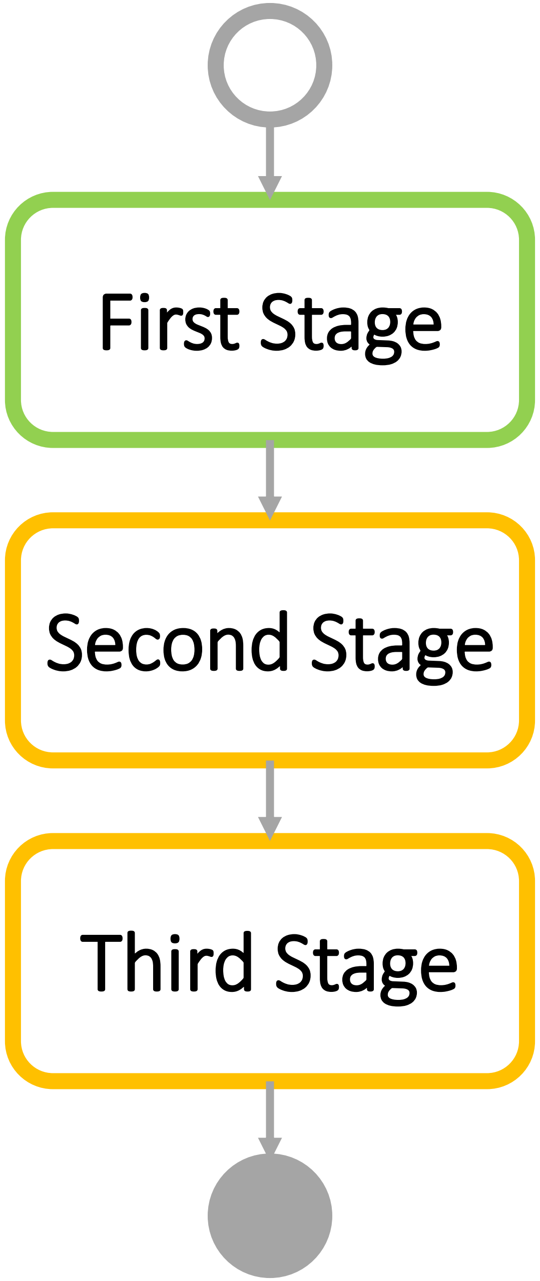
2.3. Creating the DCSO version 3.0.2

- The creation of version 3.0.2 of the DCSO followed a **three iterative stages approach**.

- First Stage
  - Create an **ontology serialisation** of the DCS application profile, and would **integrate terms from selected domain ontologies**

  - Expressed in **Terse RDF Triple Syntax (Turtle)** and **Web Ontology Language (OWL)**

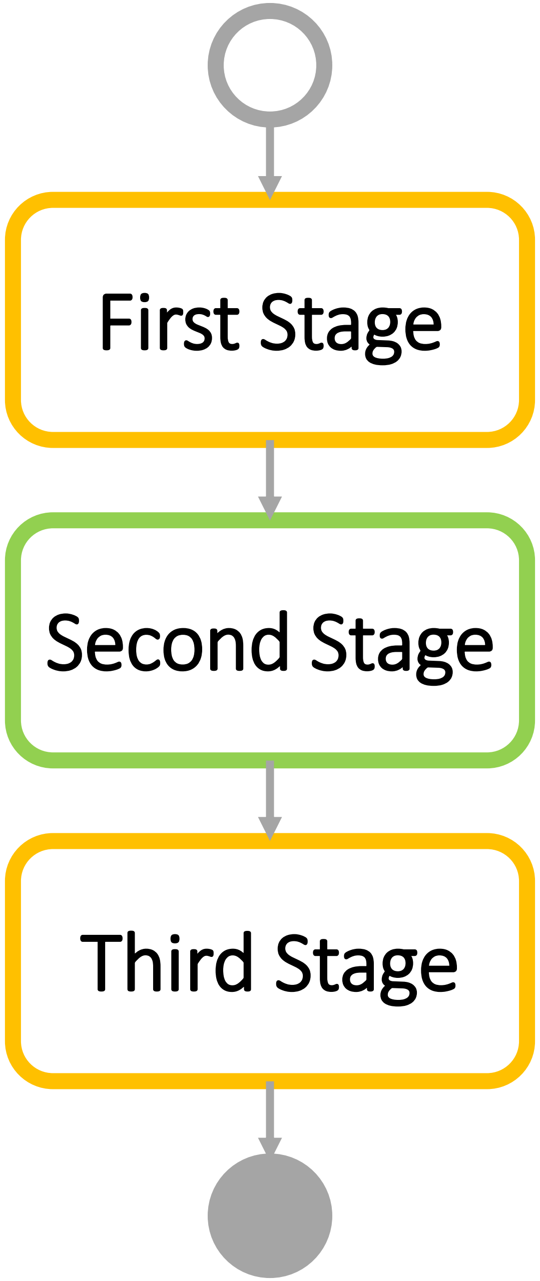
  - Outcome was the creation of the **DCSO Core**



2. Ontology Engineering Process

2.3. Creating the DCSO version 3.0.2

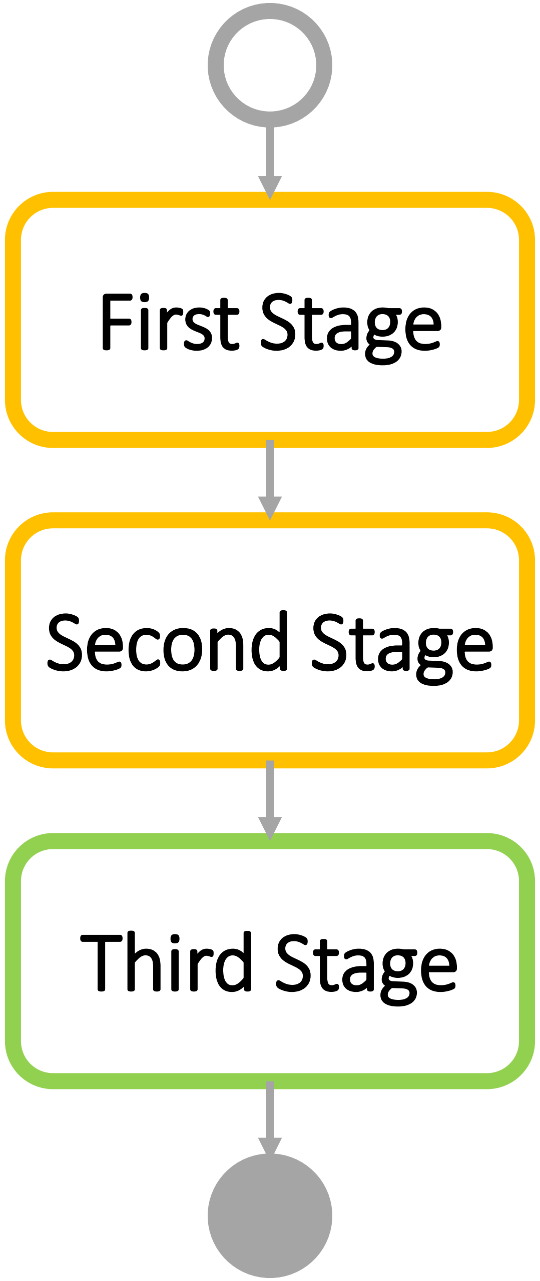
- Second Stage
  - Incorporate the usage of **controlled vocabularies** into the existing ontology
  - Create a **constraint validation layer** using **ShEx**



2. Ontology Engineering Process

2.3. Creating the DCSO version 3.0.2

- Third Stage
  - **Human-readable descriptions** for all resources
  - Default **namespace** was provided by the **W3ID**
  - **Revision of the GitHub repository** where the ontology is published, by adding **documentation** and **reorganising the structure** of the repository

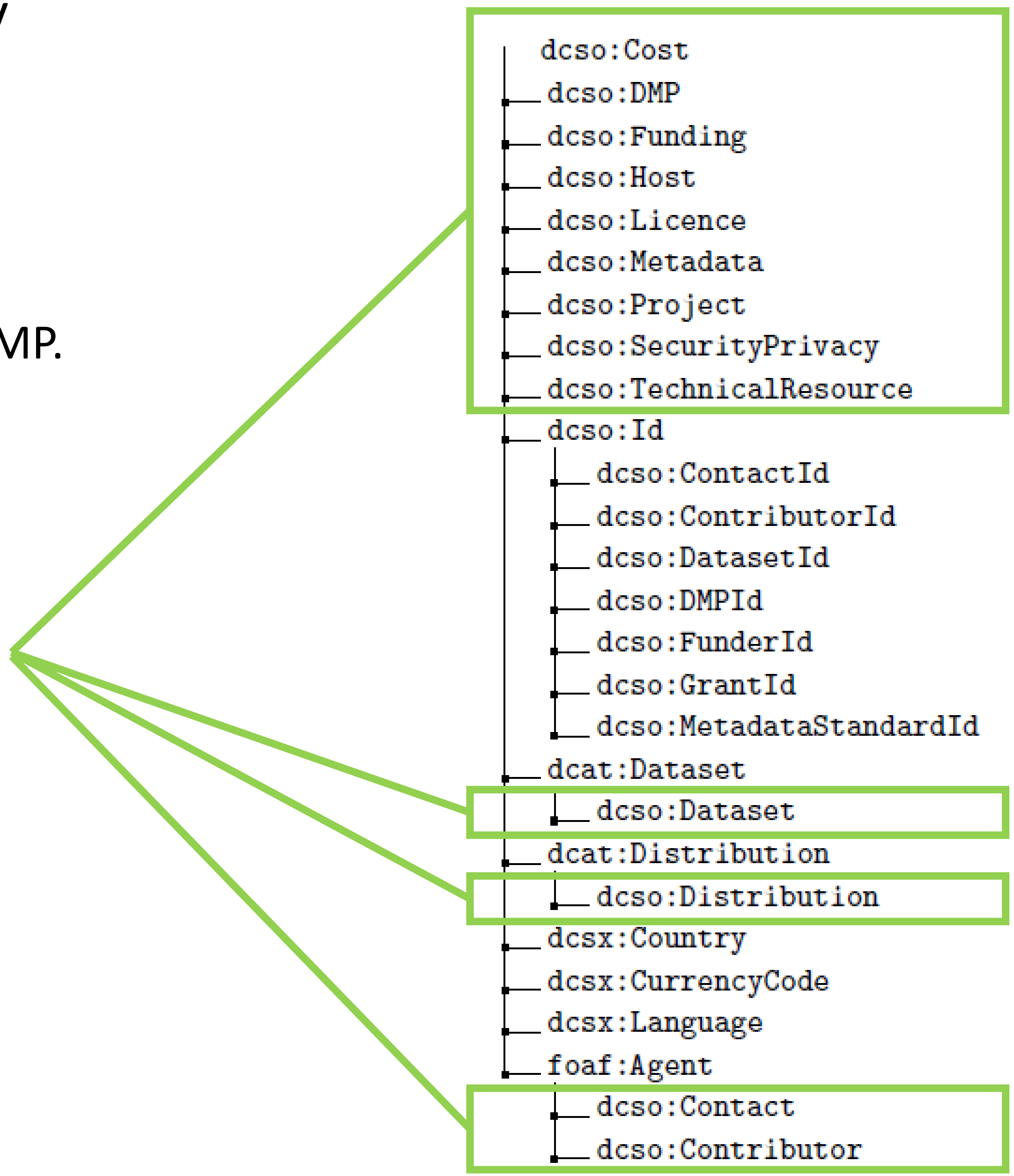


3. DMP Common Standard Ontology

3.1. DCSO Core

- The **DCSO core** represents the **core set of universal elements** defined by the DCS characterisation of a DMP.

- The DCSO Core comprises of **26 classes**
  - 13 of which **match terms** in the DCS application profile
  - 13 are divided into two categories
    - **Identifier** classes
    - **External** classes



```
dcso:Cost
dcso:DMP
dcso:Funding
dcso:Host
dcso:Licence
dcso:Metadata
dcso:Project
dcso:SecurityPrivacy
dcso:TechnicalResource
dcso:Id
    dcso:ContactId
    dcso:ContributorId
    dcso:DatasetId
    dcso:DMPId
    dcso:FunderId
    dcso:GrantId
    dcso:MetadataStandardId
dcat:Dataset
    dcso:Dataset
dcat:Distribution
    dcso:Distribution
dcsx:Country
dcsx:CurrencyCode
dcsx:Language
foaf:Agent
    dcso:Contact
    dcso:Contributor
```

3. DMP Common Standard Ontology

3.1. DCSO Core

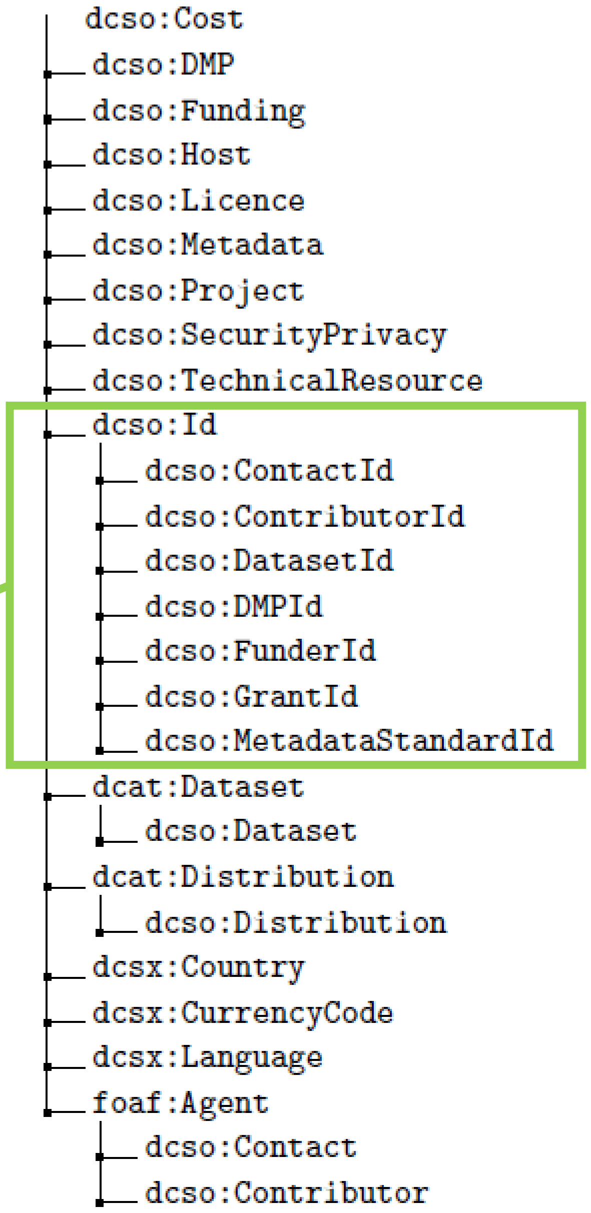
- The **DCSO core** represents the **core set of universal elements** defined by the DCS characterisation of a DMP.

- The DCSO Core comprises of **26 classes**

  - 13 of which **match terms** in the DCS application profile

  - 13 are divided into two categories

    - **Identifier** classes
    - **External** classes



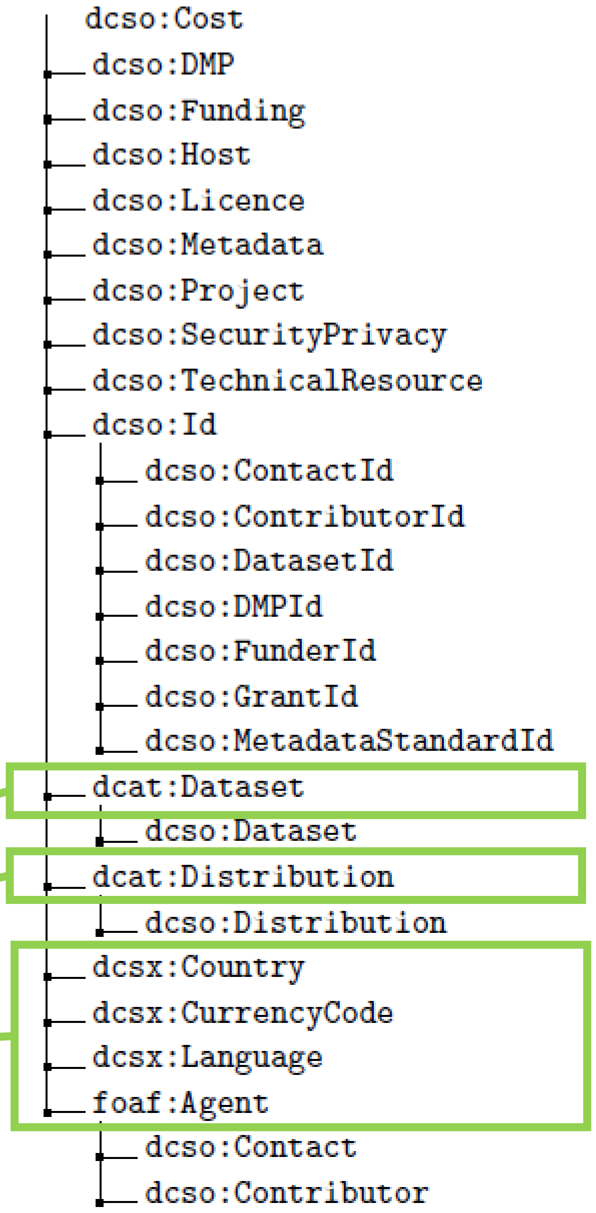
```
dcso:Cost
dcso:DMP
dcso:Funding
dcso:Host
dcso:Licence
dcso:Metadata
dcso:Project
dcso:SecurityPrivacy
dcso:TechnicalResource
dcso:Id
    dcso:ContactId
    dcso:ContributorId
    dcso:DatasetId
    dcso:DMPId
    dcso:FunderId
    dcso:GrantId
    dcso:MetadataStandardId
dcat:Dataset
    dcso:Dataset
dcat:Distribution
    dcso:Distribution
dcsx:Country
dcsx:CurrencyCode
dcsx:Language
foaf:Agent
    dcso:Contact
    dcso:Contributor
```

3. DMP Common Standard Ontology

3.1. DCSO Core

- The **DCSO core** represents the **core set of universal elements** defined by the DCS characterisation of a DMP.

- The DCSO Core comprises of **26 classes**
  - 13 of which **match terms** in the DCS application profile
  - 13 are divided into two categories
    - **Identifier** classes
    - **External** classes



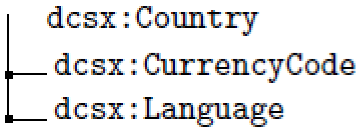
```
dcso:Cost
dcso:DMP
dcso:Funding
dcso:Host
dcso:Licence
dcso:Metadata
dcso:Project
dcso:SecurityPrivacy
dcso:TechnicalResource
dcso:Id
    dcso:ContactId
    dcso:ContributorId
    dcso:DatasetId
    dcso:DMPId
    dcso:FunderId
    dcso:GrantId
    dcso:MetadataStandardId
dcat:Dataset
    dcso:Dataset
dcat:Distribution
    dcso:Distribution
dcsx:Country
dcsx:CurrencyCode
dcsx:Language
foaf:Agent
    dcso:Contact
    dcso:Contributor
```

3. DMP Common Standard Ontology

3.2. DCSX: DCSO Extensions

- The **DCSX ontology** was created to address the DCS core set of terms that require the usage of **standardised controlled vocabularies.**

- Each class represents a **standardised controlled vocabulary**.
  - The **dcsx:Country** class represents the **ISO 3166-1** country codes
  - The **dcxs:CurrencyCode** class represents the **ISO 4217** currency codes
  - The **dcsx:Language** class represents the **ISO 639-3** language codes



3. DMP Common Standard Ontology

3.3. Constraints Validation Layer

- The **DCSO Constraints Validation Layer facilitates compliance validation** with the underlying model.

- **ShEx** was selected as the **representation language**, due to the **expertise** and **familiarity** with the format by the creators.

- The created **ShEx Schemas** follow the **guidelines** established in the DCS application profile.
  - Regarding existence
  - Cardinaility
  - How elements should be combined with each other

4. Going Forward

4.1. Future Work

- Fine tune the **DSW** use of the DCSO as an **export format**
- **Reassess the need** for the definition of **individuals** for the **DCSX**
- **Further integration of terms** from established ontologies
- The DCSO should be **interchangeable with the DCS JSON serialisation**
- **Semantic validation** of DMP documents using the DCSO
- **Continuous update** of the DCSO

THANK
YOU!