Systemik Solutions 2023

Issues Paper

Heurist Sustanability – RO-Crate Architecture

Australian Research Data Commons

#### Contents

Document Status 4

Document Scope 4

Heurist Data Export 5

Heurist Data to RO-Crate 6

Structural 6

Record Types 7

Base Fields 8

Fields 9

Vocabularies/Terms 10

Records 11

Abstracted 12

Workflow 14

Questions 15

Heurist to RO-Crate Modelling (Draft) 16

Heurist entity check list 16

Custom vocabulary terms 16

Context 16

Record Type 17

Base field and field 18

Vocabularies and terms 20

Static properties 20

Property mapping 21

Example 22

Records 22

Property mapping 22

Field values 23

Date/Temporal 23

Geospatial 23

File or media URL 24

Term 24

Record pointer 25

Relationship marker 25

Root data entity 25

Full example 26

### Document Status

|  |  |
| --- | --- |
| Version | V1.0 |
| Issue Date | June 2023 |
| Stakeholder | CDL Project |
| Author | Yang Li, Systemik Solutions  Ian McCrabb, Systemik Solutions |
| Reviewer |  |
| Distribution | CDL Project |
| Related Documents |  |

### Document Scope

This document canvasses architectural issues related to design of…

# Heurist Data Export

Tha complete archive of the Heurist database can be imported from the Heurist database portal. The archive is a zip package contains the following:

* The HML export of the database structure (Database\_Structure.xml)
* The HML export of the data.
* The SQL database dump of the whole database.

# Heurist Data to RO-Crate

## Structural

Heurist uses record types for different types of entities. In each record type, it uses fields (details) to define the record type structures. Base fields are independent definitions from record types. A base field can be attached to one/multiple record type(s) as a field with some extra information specific to the record type such as the display label. Each record is an instance of a record type. Besides some common properties of the record such as id, date added, date modified, each record stores the values of fields defined on that record type.

The structural way of mapping Heurist data to RO-Crate entities is to map each component in Heurist such as record types and fields to RO-Crate entity types. This is the more generical way of mapping data, which potentially can apply to any Heurist database. However, the RO-Crate generated from this approach may not be intuitive, and may be difficult to understand for people without the Heurist background.

Note that out-of-the box, Heurist contains certain amount of data structures by default such as predefined record types, fields and vocabularies. These may not be suitable for archiving purposes as they are not used in every project.

The following are some HML snippets from OMAA about these generic components in Heurist.

### Record Types

****

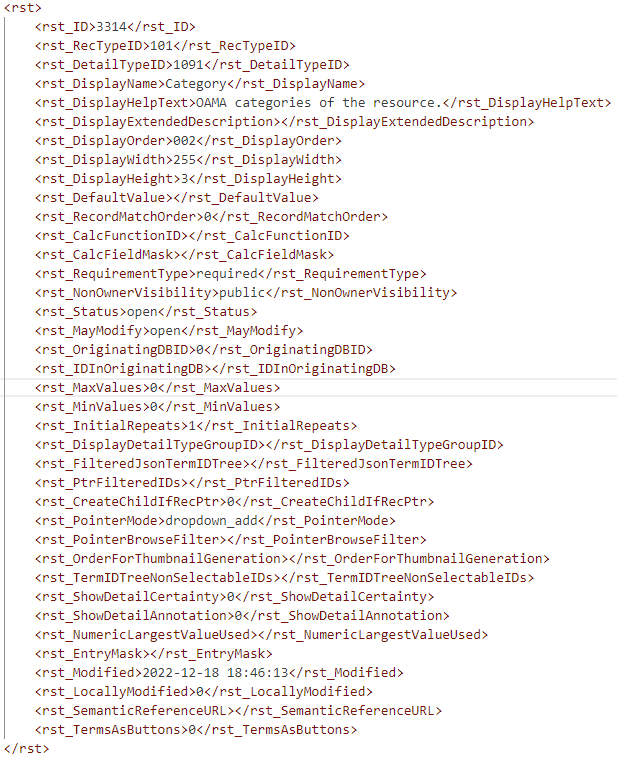
### Base Fields

****

Heurist supports the following field types:

* Terms
* Numeric
* Text (Single line)
* Memo Text (Multi-line or HTML)
* Date/Temporal
* Geospatial
* File or media URL
* Record pointer/Foreign key
* Relationship marker

### Fields

****

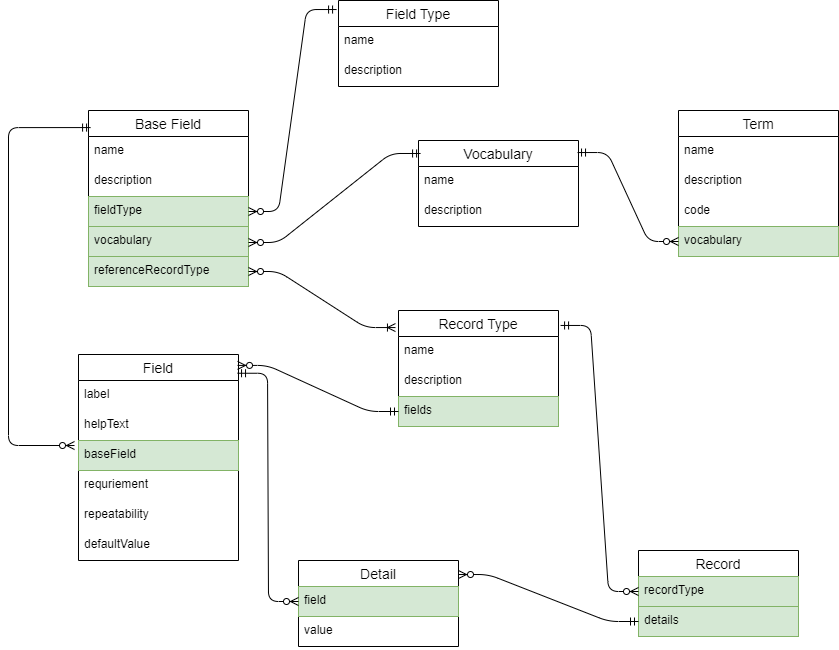
### Vocabularies/Terms

****

### Records



The following ERD is an example of mapping the Heurist structural data into RO-Crate JSON-LD:



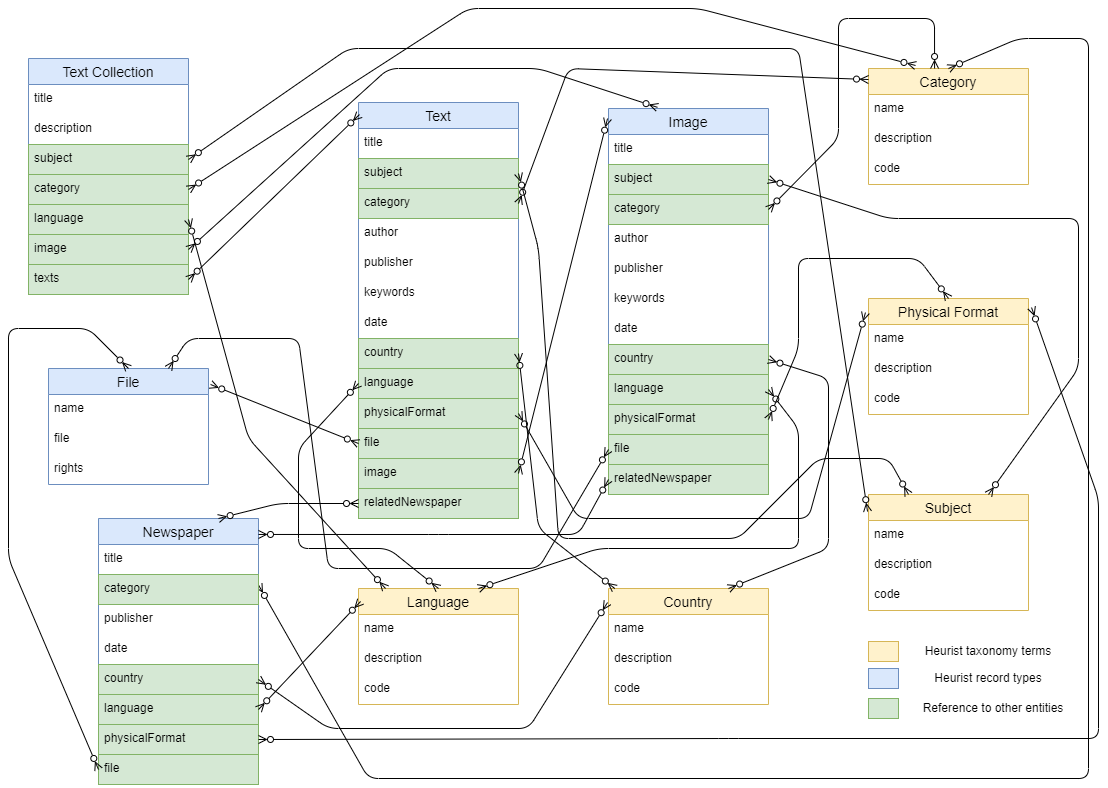
## Abstracted

The abstracted way of mapping Heurist data to RO-Crate entities is to map each record type in Heurit to a RO-Crate entity type. For example, it has ‘Text’, ‘Newspaper’, ‘Image’ record types. In RO-Crate, corresponding entity types can be defined in RO-Crate.

This would be a more intuitive way of presenting data in RO-Crate. However, as each Heurist database will have different structures, the archiving process needs to be designed and implemented on project basis.

The difficulty is to automatically match project entities, fields, taxonomies to standard RO-Crate vocabulary terms, or extend the RO-Crate with custom vocabulary terms. This process will need human design input but could produce a machine-readable mapping output which can be used for the Heurist-to-RO-Crate export tool.

The following is an example ERD of mapping OMAA data to RO-Crate JSON-LD:



# Workflow

Proposed workflow for migration of OMAA (and subsequent Heurist db's) is as follows:

|  |  |  |
| --- | --- | --- |
| Phase | Description | Output |
| Export | Once off export from Heurist | HML structure, HML data and SQL dump |
| Cross Walk | Generate an RO-Crate to ‘Abstracted’ model. All vocabulary references are generated as custom vocabularies.  PHP Library to convert:   * HML structure to RO-Crate entities. * HML data to RO-Crate entities   PHP Library to generate RO-Crate constituents:   * Metadata JSON * Preview HTML * Heurist HML and files (documents and images…) * SQL snapshot | RO-Crate |
| Explore | Open RO-Crate in Describo to explore data structures. |  |
| Mapping | Map abstracted entities (Heurist record types, fields and vocabularies) to schema.org vocabularies | Cross walker configuration |
| Iterate | Rerun:   * Cross walk * Explore * Mapping | Refine cross walker configuration |

Table 3 – Workflow

As we tackle subsequent Heurist db's we can use the latest Heurist cross walker as a departure point. As Heurist design generally works from prebuilt record types as a departure point anticipate that as we move through Heurist db’s the majority of records and fields will be able to be pre-mapped.

Where that mapping might be erroneous, we would expect this to be picked up by the analyst using Describo in the Explore phase.

# Questions

* Is it an acceptable RO-Crate if we embed all class/property names as custom vocabularies except a few common terms such as “name”, “identifier”. Is a web presence of the vocabulary descriptions mandatory in this case?
  + Answer seems to be that it is acceptable, and a web presence is not mandatory.
* How does the DHRA collection handle those custom vocabularies?
  + Answer seems to be that they do structural structural mapping from ohrm to an RO-CRate and generate custom vocabularies.
* Is there an approach for users to map the custom vocabulary to a standard schema.org vocabulary in a later stage, such as Describo?
  + Answer seems to be that Describo doesn’t really support this.
* Where do RO-Crate Profiles fit in?
  + RO-Crate Profile spec is in draft format. Linguistics profile in draft format for LDCA? Profiles seem to be domain specific whilst Heurist is cross domain.
* Profile builder
  + Outside Scope but propose approach of Heurist cross walk configuration map. Profiling might be domain specific task undertaken on RO-Crates
* RO-Crate Validator
  + Outside Scope. Propose approach is to validate RO-Crates using Describo.

# Heurist to RO-Crate Modelling (Draft)

## Heurist entity check list

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Tag** | **Description** | **Status** |
| Record type group | rtg | The category group of Heurist record types. |  |
| Detail type group | dtg | The category group of Heurist base fields. |  |
| Vocabulary group | vcg | The category group of Heurist vocabularies. |  |
| Ontology | ont | Not sure how these are used in Heurist. Vocabulary terms from Heurist do have a property called trm\_OntID which is referencing to an ontology by its ID. |  |
| Term | trm | Heurist terms. Note that the vocabularies defined from the Heursit UI are terms as well. The vocabulary will be the root node of the vocabulary terms hierarchy. | Mapped as DefinedTermSet and DefinedTerm. |
| Record Type | rty | Heurist record types. | Mapped as custom class type. |
| Detail Type | dty | Heurist base fields. | Mapped as custom property type. |
| Record structure | rst | Heurist fields. | Not mapped since fields are mapped through the base fields. |
| Relationship constraint |  | Not used in OMAA. |  |
| File extension to MIME type | fxm | Associate file extensions to MIME types. |  |
| Record | record | Heurist records. | Mapped as entities with custom types. |

## Custom vocabulary terms

### Context

Custom terms should be declared in the context. The URL of a term declared in the context should be reserved under the RO-Crate public namespace (<https://w3id.org/ro/terms/>) with the project namespace "omaa".

For example:

{

"@context": [

"https://w3id.org/ro/crate/1.1/context",

{

"Text": "https://w3id.org/ro/terms/omaa#Text",

"TextCollection": "https://w3id.org/ro/terms/omaa#TextCollection",

"subject": "https://w3id.org/ro/terms/omaa#subject"

}

],

"@graph": []

}

### Record Type

Each record type should be defined as a custom term of class in RO-Crate.

#### Static properties

|  |  |
| --- | --- |
| **Name** | **Value** |
| @id | The URL specified for the term in the context. E.g. https://w3id.org/ro/terms/omaa#Text |
| @type | rdfs:Class |

#### Property Mapping

|  |  |
| --- | --- |
| **Name** | **Map to** |
| rty\_ID |  |
| rty\_Name | rdfs:label: The name of the record type needs some transformation to PascalCase. e.g. "OMAA newspaper" to "OMAANewspaper". |
| rty\_OrderInGroup |  |
| rty\_Description | rdfs:comment |
| rty\_TitleMask |  |
| rty\_CanonicalTitleMask |  |
| rty\_Plural |  |
| rty\_Status |  |
| rty\_OriginatingDBID |  |
| rty\_NameInOriginatingDB |  |
| rty\_IDInOriginatingDB |  |
| rty\_NonOwnerVisibility |  |
| rty\_ShowInLists |  |
| rty\_RecTypeGroupID |  |
| rty\_RecTypeModelIDs |  |
| rty\_FlagAsFieldset |  |
| rty\_ReferenceURL |  |
| rty\_AlternativeRecEditor |  |
| rty\_Type |  |
| rty\_ShowURLOnEditForm |  |
| rty\_ShowDescriptionOnEditForm |  |
| rty\_Modified |  |
| rty\_LocallyModified |  |

#### Reserved types

The following class names should be reserved for RO-Crate:

* CreativeWork
* Dataset
* File
* DefinedTerm
* DefinedTermSet
* GeoCoordinates
* GeoShape

When a record type from Heurist have a name conflicting with the reserved class names, the value of rdfs:label should prepend the Heurist database name. For example, for the "File" record type in OMAA database, the label of custom term for the "File" record type should be "OMAAFile".

#### Example

{

"@id": "https://w3id.org/ro/terms/omaa#Text",

"@type": "rdfs:Class",

"rdfs:label": "Text",

"rdfs:comment": "OAMA texts"

}

### Base field and field

Each field on a record type could be defined as a custom term in RO-Crate. However, this will result in duplicate terms since some fields are used across multiple record types. In this case, using base field for the custom terms may be a better option.

#### Static properties

|  |  |
| --- | --- |
| **Name** | **Value** |
| @id | The URL specified for the term in the context. E.g. https://w3id.org/ro/terms/omaa#nameOrTitle |
| @type | rdf:Property |
| domainIncludes | This specifies the relationship between the RO-Crate property and the RO-Crate class, which should be an reference to a record type term. |
| rangeIncludes | This specifies the types of RO-Crate classes can be used as the values of this property.  The following lists the class types should be used for different Heurist field types:   * Term type field: {"@id": "https://schema.org/DefinedTerm"} * File type field: {"@id": "http://schema.org/MediaObject"} * Spatial type field: [{"@id": "http://schema.org/GeoCoordinates"},{"@id": "http://schema.org/GeoShape"}] * Record pointer type field: the URL of the term for the target record type. For example: {"@id": "https://w3id.org/ro/terms/omaa#OMAAFile"} |

#### Property mapping

|  |  |
| --- | --- |
| **Name** | **Map to** |
| dty\_ID |  |
| dty\_Name | rdfs:label. The name should be converted to camelCase. Texts inside parentheses should be ignored. For example: "Name or Title (please rename appropriately)" to "nameOrTitle".  It should also check whether the label is taken by another record type. In this case, prepend the record type name to the label. For example: "File" field in "File" record type, the label should be "fileFile" given there is also a field called "File" (different type or purpose) in the "Text" record type. |
| dty\_Documentation |  |
| dty\_Type |  |
| dty\_HelpText | rdfs:comment |
| dty\_ExtendedDescription |  |
| dty\_EntryMask |  |
| dty\_Status |  |
| dty\_OriginatingDBID |  |
| dty\_NameInOriginatingDB |  |
| dty\_IDInOriginatingDB |  |
| dty\_DetailTypeGroupID |  |
| dty\_OrderInGroup |  |
| dty\_JsonTermIDTree |  |
| dty\_TermIDTreeNonSelectableIDs |  |
| dty\_PtrTargetRectypeIDs |  |
| dty\_FieldSetRectypeID |  |
| dty\_ShowInLists |  |
| dty\_NonOwnerVisibility |  |
| dty\_Modified |  |
| dty\_LocallyModified |  |
| dty\_SemanticReferenceURL |  |

#### Example

{

"@id": "https://w3id.org/ro/terms/omaa#place",

"@type": "rdf:Property",

"rdfs:label": "place",

"rdfs:comment": "A pointer to a place or location record",

"domainIncludes": {"@id": "https://w3id.org/ro/terms/omaa#Text"},

"rangeIncludes": {"@id": "https://w3id.org/ro/terms/omaa#Place"}

}

## Vocabularies and terms

Each vocabulary from Heurist could be created as entity of class DefinedTermSet. Terms within the vocabulary can be list in property hasDefinedTerm. Each term from Heurist could be created as entity of class DefinedTerm.

### Static properties

|  |  |
| --- | --- |
| **Name** | **Value** |
| @type | DefinedTermSet for vocabularies. DefinedTerm for terms. |
| hasDefinedTerm | Only applies to vocabularies. Contains a list of references to terms. |

### Property mapping

|  |  |
| --- | --- |
| **Name** | **Map to** |
| trm\_ID | @id. The Heurist term ID prefixed by #trm\_. E.g. #trm\_167 |
| trm\_Label | name |
| trm\_InverseTermId |  |
| trm\_Description | description |
| trm\_Status |  |
| trm\_OriginatingDBID |  |
| trm\_NameInOriginatingDB |  |
| trm\_IDInOriginatingDB |  |
| trm\_AddedByImport |  |
| trm\_IsLocalExtension |  |
| trm\_Domain |  |
| trm\_OntID |  |
| trm\_ChildCount |  |
| trm\_ParentTermID |  |
| trm\_Depth |  |
| trm\_Modified |  |
| trm\_LocallyModified |  |
| trm\_Code | termCode. Only applies to terms. |
| trm\_SemanticReferenceURL |  |
| trm\_IllustrationURL |  |
| trm\_VocabularyGroupID |  |
| trm\_OrderInBranch |  |

### Example

#### Vocabulary

{

"@id": "#trm\_9470",

"@type": "DefinedTermSet",

"name": "Category",

"description": "",

"hasDefinedTerm": [

{"@id": "#trm\_10048"},

{"@id": "#trm\_10047"}

]

}

#### Term

{

"@id": "#trm\_10048",

"@type": "DefinedTerm",

"name": "1. Settlement",

"description": "",

"termCode": ""

}

## Records

Each record from Heurist could be created as an entity with the custom type (defined by record types) in RO-Crate.

### Property mapping

|  |  |
| --- | --- |
| **Name** | **Map to** |
| visibility |  |
| visnote |  |
| id | @id. The Heurist record ID prefixed by #rec\_. E.g. #rec\_1660 |
| type | @type. The custom term defined by the record type. |
| citeAs |  |
| title |  |
| added |  |
| modified |  |
| workgroup |  |
| detail | See [Field values](#_Field_values) section. |

### Field values

Each field value from a Heurist record can be created as a property with the name of custom term created from the base field.

For the following field types, the value of the field can be inserted as the property value as is.

* Numeric
* Text (Single line)
* Memo Text (Multi-line or HTML)

### Date/Temporal

For fields in Date/Temporal type, the values need to be converted into [ISO 8601 date format](https://en.wikipedia.org/wiki/ISO_8601).

For example:

Year only:

1764

Date only:

2023-07-07

Date and time (UTC):

2023‐07‐07T06:48:58Z

Date and time (UTC+07:00):

2023‐07‐07T13:48:58+07:00

Date range:

2023-07-07/2014-01-21

### Geospatial

For values in Geospatial type, a separate entity in [GeoCoordinates](https://schema.org/GeoCoordinates) or [GeoShape](https://schema.org/GeoShape) type should be created and referenced by the property value. The @id of the geo entity should be a generated UUID (v4) prefixed by #.

For example:

{

"@id": "#rec\_1660",

"@type": "Place",

"nameOrTitle": "Kraton Kesultanan Yogyakarta",

"mappableLocation": {"@id": "#84553dc6-e50e-4a7d-8b99-22846d84d15d"}

},

{

"@id": "#84553dc6-e50e-4a7d-8b99-22846d84d15d",

"@type": "GeoCoordinates",

"latitude": "-7.807651",

"longitude": "110.363953"

}

Heurist store geospatial data in WKT format. If it's a point, create a GeoCoordinates entity and specify the latitude and longitude.

If it's in other shapes, create a GeoShape entity and specify the corresponding shape attributes (box, circle, line, polygon etc.)

For example:

{

"@id": "#84553dc6-e50e-4a7d-8b99-22846d84d15d",

"@type": "GeoShape",

"box": "-7.807651,110.363953 -5.823423,112.324234",

}

### File or media URL

For values in file or media URL type, a separate entity in [File](https://schema.org/MediaObject) type should be created and referenced by the property value. The @id of the file entity should be the local path relative to the RO-Crate root directory, or the file URL if it's a remote file.

For example:

{

"@id": "#rec\_1688",

"@type": "OMAAFile",

"nameOrTitle": "Pedoman 1959.12.01.p.2",

"fileFile": {"@id": "ulf\_52\_Pedoman 59.12.01.p.2.pdf"}

},

{

"@id": "ulf\_52\_Pedoman 59.12.01.p.2.pdf",

"@type": "File",

"name": "Pedoman 59.12.01.p.2.pdf",

"contentSize": "1062000",

"encodingFormat": "application/pdf",

"uploadDate": "2023-05-31T05:01:14"

}

For local files, it should have the name, contentSize (in byte), encodingFormat, and uploadDate populated.

For remote files, it should also have the contentUrl populated.

### Term

For values in term type, the value of the property can reference the corresponding DefinedTerm entity. For example:

{

"@id": "#rec\_463",

"@type": "Text",

"category": {"@id": "#trm\_10047"},

},

{

"@id": "#trm\_10047",

"@type": "DefinedTerm",

"name": "2. National boundaries",

"description": "",

"termCode": ""

}

### Record pointer

For values in record pointer type, the value of the property can reference the corresponding entity representing the target record. For example:

{

"@id": "#rec\_463",

"@type": "Text",

"place": {"@id": "#rec\_1660"},

},

{

"@id": "#rec\_1660",

"@type": "Place",

"nameOrTitle": "Kraton Kesultanan Yogyakarta",

}

### Relationship marker

TBD

## Root data entity

The root data entity of RO-Crate should contain the name and description of the Heurist database. Then it should list all vocabularies (no term) and records from Heurist under hasPart. For example:

{

"@id": "./",

"@type": "Dataset",

"name": "Opening Australia's Multilingual Archive",

"description": "Opening the Multilingual Archive of Australia brings together historical materials from national and international collections in languages other than English. We seek to rethink and enlarge narratives about Australia that come solely from English-language sources, by showing modern Australia to be a complex multilingual creation. This project is funded by an Australian Research Council Discovery Project grant.",

"hasPart": [

{"@id": "#trm\_9470"},

{"@id": "#rec\_463"},

{"@id": "#rec\_1660"},

{"@id": "#rec\_1688"}

]

}

## Full example

{

"@context": [

"https://w3id.org/ro/crate/1.1/context",

{

"Vocabulary": "https://w3id.org/ro/terms/omaa#Vocabulary",

"Term": "https://w3id.org/ro/terms/omaa#Term",

"Text": "https://w3id.org/ro/terms/omaa#Text",

"nameOrTitle": "https://w3id.org/ro/terms/omaa#nameOrTitle",

"category": "https://w3id.org/ro/terms/omaa#category",

"author": "https://w3id.org/ro/terms/omaa#author",

"date": "https://w3id.org/ro/terms/omaa#date",

"place": "https://w3id.org/ro/terms/omaa#place",

"textFile": "https://w3id.org/ro/terms/omaa#textFile",

"OMAAFile": "https://w3id.org/ro/terms/omaa#OMAAFile",

"fileFile": "https://w3id.org/ro/terms/omaa#fileFile",

"Place": "https://w3id.org/ro/terms/omaa#Place",

"mappableLocation": "https://w3id.org/ro/terms/omaa#mappableLocation"

}

],

"@graph": [

{

"@type": "CreativeWork",

"@id": "ro-crate-metadata.json",

"conformsTo": {"@id": "https://w3id.org/ro/crate/1.1"},

"about": {"@id": "./"}

},

{

"@id": "./",

"@type": "Dataset",

"name": "Opening Australia's Multilingual Archive",

"description": "Opening the Multilingual Archive of Australia brings together historical materials from national and international collections in languages other than English. We seek to rethink and enlarge narratives about Australia that come solely from English-language sources, by showing modern Australia to be a complex multilingual creation. This project is funded by an Australian Research Council Discovery Project grant.",

"hasPart": [

{"@id": "#trm\_9470"},

{"@id": "#rec\_463"},

{"@id": "#rec\_1660"},

{"@id": "#rec\_1688"}

]

},

{

"@id": "https://w3id.org/ro/terms/omaa#Text",

"@type": "rdfs:Class",

"rdfs:label": "Text",

"rdfs:comment": "OAMA texts"

},

{

"@id": "https://w3id.org/ro/terms/omaa#nameOrTitle",

"@type": "rdf:Property",

"rdfs:label": "nameOrTitle",

"rdfs:comment": "The main name or title for the object. Title of a work, family name of person, name of organisation etc.",

"domainIncludes": [

{"@id": "https://w3id.org/ro/terms/omaa#Text"},

{"@id": "https://w3id.org/ro/terms/omaa#OMAAFile"},

{"@id": "https://w3id.org/ro/terms/omaa#Place"}

]

},

{

"@id": "https://w3id.org/ro/terms/omaa#category",

"@type": "rdf:Property",

"rdfs:label": "category",

"rdfs:comment": "OAMA categories of the resource.",

"domainIncludes": {"@id": "https://w3id.org/ro/terms/omaa#Text"},

"rangeIncludes": {"@id": "https://schema.org/DefinedTerm"}

},

{

"@id": "https://w3id.org/ro/terms/omaa#author",

"@type": "rdf:Property",

"rdfs:label": "author",

"rdfs:comment": "Entities responsible for making contributions to the resource.",

"domainIncludes": {"@id": "https://w3id.org/ro/terms/omaa#Text"}

},

{

"@id": "https://w3id.org/ro/terms/omaa#date",

"@type": "rdf:Property",

"rdfs:label": "date",

"rdfs:comment": "The single or principal date of the item (may also include time and/or have fuzzy limits)",

"domainIncludes": {"@id": "https://w3id.org/ro/terms/omaa#Text"}

},

{

"@id": "https://w3id.org/ro/terms/omaa#place",

"@type": "rdf:Property",

"rdfs:label": "place",

"rdfs:comment": "A pointer to a place or location record",

"domainIncludes": {"@id": "https://w3id.org/ro/terms/omaa#Text"},

"rangeIncludes": {"@id": "https://w3id.org/ro/terms/omaa#Place"}

},

{

"@id": "https://w3id.org/ro/terms/omaa#textFile",

"@type": "rdf:Property",

"rdfs:label": "textFile",

"rdfs:comment": "A related resource that is substantially the same as the pre-existing described resource, but in another digitisation format.",

"domainIncludes": {"@id": "https://w3id.org/ro/terms/omaa#Text"},

"rangeIncludes": {"@id": "https://w3id.org/ro/terms/omaa#OMAAFile"}

},

{

"@id": "https://w3id.org/ro/terms/omaa#OMAAFile",

"@type": "rdfs:Class",

"rdfs:label": "OMAAFile",

"rdfs:comment": "OAMA file"

},

{

"@id": "https://w3id.org/ro/terms/omaa#fileFile",

"@type": "rdf:Property",

"rdfs:label": "fileFile",

"rdfs:comment": "A related resource that is substantially the same as the pre-existing described resource, but in another digitisation format.",

"domainIncludes": {"@id": "https://w3id.org/ro/terms/omaa#OMAAFile"},

"rangeIncludes": {"@id": "http://schema.org/MediaObject"}

},

{

"@id": "https://w3id.org/ro/terms/omaa#Place",

"@type": "rdfs:Class",

"rdfs:label": "Place",

"rdfs:comment": "A simplified record for a Place, for routine use eg. for site location, place of birth, place of publication. Use Place (detailed) where more information is needed."

},

{

"@id": "https://w3id.org/ro/terms/omaa#mappableLocation",

"@type": "rdf:Property",

"rdfs:label": "mappableLocation",

"rdfs:comment": "A geospatial object providing a mappable location for the record - can be used for any type of record with spatial location",

"domainIncludes": {"@id": "https://w3id.org/ro/terms/omaa#Place"},

"rangeIncludes": [

{"@id": "http://schema.org/GeoCoordinates"},

{"@id": "http://schema.org/GeoShape"}

]

},

{

"@id": "#trm\_9470",

"@type": "DefinedTermSet",

"name": "Category",

"description": "",

"hasDefinedTerm": [

{"@id": "#trm\_10048"},

{"@id": "#trm\_10047"}

]

},

{

"@id": "#trm\_10048",

"@type": "DefinedTerm",

"name": "1. Settlement",

"description": "",

"termCode": ""

},

{

"@id": "#trm\_10047",

"@type": "DefinedTerm",

"name": "2. National boundaries",

"description": "",

"termCode": ""

},

{

"@id": "#rec\_463",

"@type": "Text",

"nameOrTitle": "Ontdekkingen in de Zuidzee : en berichten aangaande de La Pérouse en zijne tochtgenoten, opgemaakt uit sporen van zijne reis op onderscheidene eilanden en landen der Stille Zee gevonden : alsmede aangaande een groot eiland thans door Fransche vluchtelingen bevolkt",

"category": [

{"@id": "#trm\_10047"},

{"@id": "#trm\_10048"}

],

"author": "La Pérouse, Jean-François de Galaup, comte de",

"date": "1799/1800",

"place": {"@id": "#rec\_1660"},

"textFile": {"@id": "#rec\_1688"}

},

{

"@id": "#rec\_1660",

"@type": "Place",

"nameOrTitle": "Kraton Kesultanan Yogyakarta",

"mappableLocation": {"@id": "#84553dc6-e50e-4a7d-8b99-22846d84d15d"}

},

{

"@id": "#84553dc6-e50e-4a7d-8b99-22846d84d15d",

"@type": "GeoCoordinates",

"latitude": "-7.807651",

"longitude": "110.363953"

},

{

"@id": "#rec\_1688",

"@type": "OMAAFile",

"nameOrTitle": "Pedoman 1959.12.01.p.2",

"fileFile": {"@id": "ulf\_52\_Pedoman 59.12.01.p.2.pdf"}

},

{

"@id": "ulf\_52\_Pedoman 59.12.01.p.2.pdf",

"@type": "File",

"name": "Pedoman 59.12.01.p.2.pdf",

"contentSize": "1062000",

"encodingFormat": "application/pdf",

"uploadDate": "2023-05-31T05:01:14"

}

]

}