

Don Quixote

Records

Selected to illustrate the following:

- Works that have parts
 - What we now refer to as the work Don Quixote was originally two different works published separate in time
- Different expressions of the same work
 - translations or other “minor” derivatives
 - where the original author is considered to be the main responsibility of the resource, not considered as new works in cataloguing
- Other derivations
 - main entry person is different from the original author, considered to be new works
- Publications with augmentations/supplements
 - In addition to the main text there are illustrations, forewords and other additions that are of interest to find

The records in the example are mainly from National Library of Spain and Bodleian Library (UK) and can be found in Europeana. All records are in MARC and retrieved from the original institutions using Z39.50. Most records include a 856-field with a link to a digital resource with the exception of a few (which are alternative MARC records for comparable resources in Europeana).

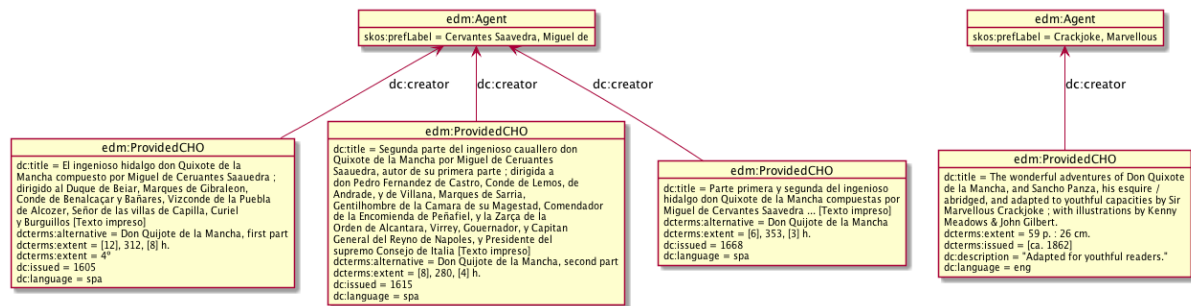
Questions/requirements

Using end-user terminology

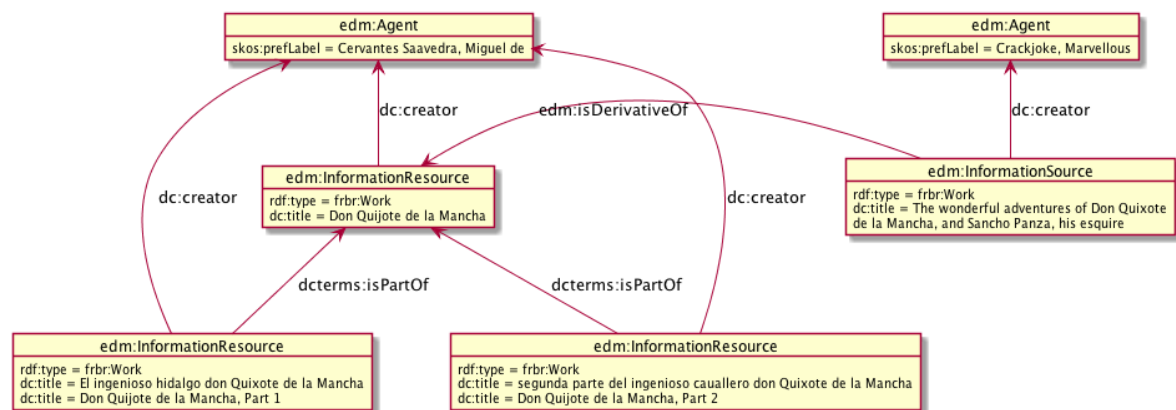
- Find:
 - All works by an author
 - Versions of a work, filtered by language, form, time
 - Editions with specific augmentations
 - Works about the author/about the work
 - Related works (based on)
- Support exploration
- The search results need to be presented in a meaningful way to enable identification, i. e. to give the user a clear idea what the resources are.

Current representation and potential use of FRBR entities:

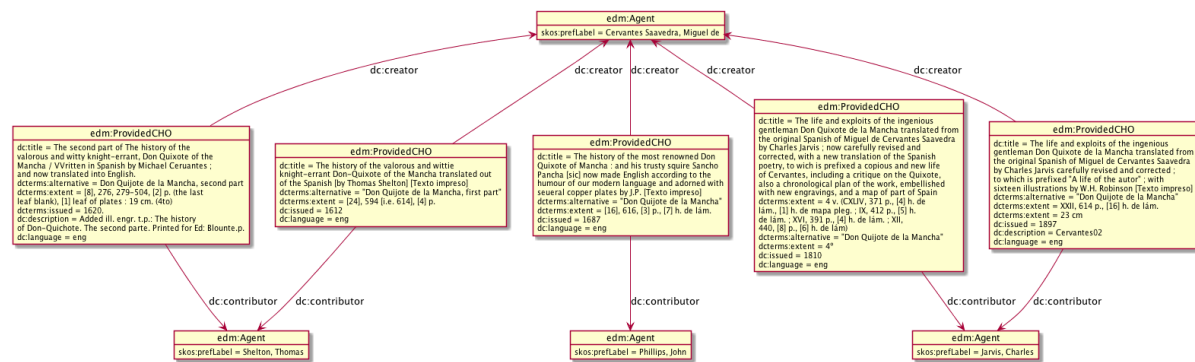
From:



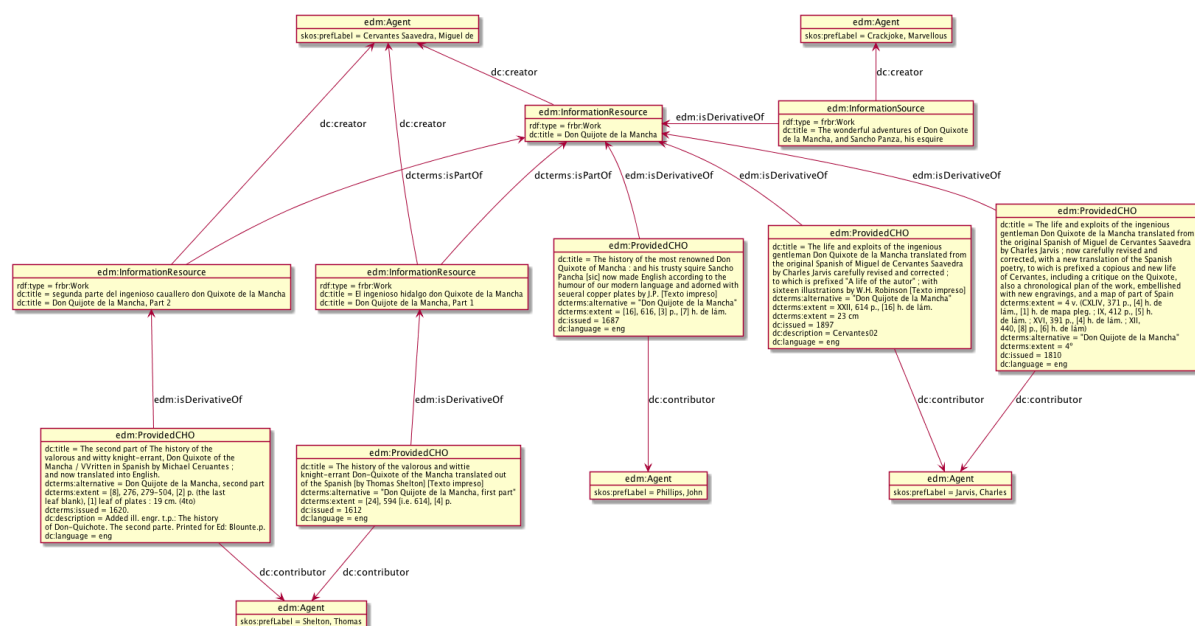
To:



From:



To:



FRBRoo modelling

FRBRoo is an extensive framework of classes and properties and the actual data models that we implement (based on FRBR) will always have to be adapted to the information we implement a model for, end user requirements and many other design criteria.

A first round of modelling was done manually to explore different use of classes and properties. Afterwards, we have used a rule-based interpretation of the MARC records to produce rdf and diagrams according to two different FRBRoo modelling “paradigms” that we have considered relevant for our collection of test-records:

- **Derivation-based**

Every distinct intellectual contribution is modelled using individual Work and Self-contained Expression with derivative relationships between works that are adaptations, translations etc. This is a rather detailed way of modelling.

- **Realisation-based**

Translations are modelled without the use of separate works for each translation. All translation-expressions are realisations of the same work. Derivations are still used for abridgements/adaptations.

Our motivation for inspecting different modelling paradigms was to analyse consequences such as different results and problems when mapping to EDM.

Some modelling decisions

- **Using F1 Work (used in both modelling approaches)**

We did not come across any issue that required the use of specialisations of F1 Work.

- **Using P148 to relate the part of the work to the whole (used in both)**

We consider this to be the preferred solution for modelling works with parts. The alternative is to use Complex Work and *has member*, but this is potentially ambiguous because this property can be used for alternatives to other members of the work as well as components of the overall concept.

- **F28 Expression Creation (used in both)**

Persons are related to works and expressions by the use of F28 Expression Creation Event. This is consistently used in both models for all works.

- **F27 Work Conception (only used in realisation-based models)**

In the models we needed a way to distinguish between the “original” creator of the “first/representative” expression and creators of subsequent derivations. The first is what we typically consider to be the author and the latter are persons with secondary responsibility (translators etc). This requirement can be solved by using “in the role of” for P14 which means that we need to enhance use of FRBRoo with an additional type system¹ for P14. As an alternative to this solution we have explored the use of F27 because it potentially is a much easier (and FRBRer-like) way of distinguishing the person we consider as the primary creator from persons that otherwise only are considered to be contributors (in DC-terminology). The current definition, however, of this class implies that it is not intended for this purpose.

- **“Original expressions” (only used in derivation-based models)**

In the Derivation-based models we have created entities for the “original expressions” in all records. This means that even if the record describes a translation (with an expression created by the translator) we have created an entity for an expression that was created by Cervantes.

Original expressions are not included in the realisation-based models.

The main reason for this difference was to explore the effect of this instance when mapping to EDM.

- **Manifestation Product Type and Publication Expression (used in both)**

All records are modelled using Manifestation Product Type and a Publication Expression that incorporates one or more Self-Contained Expression. We have attempted to identify all additional content from the use of relator codes (but this of course easily creates errors).

¹ imho this should be avoided for two good reasons. Properties of properties are difficult to implement in rdf and other systems (although it can be solved with subtyping). Secondly, I think it is bad of the FRBRoo model requires that we have to implement and use additional type-systems to express very common information.

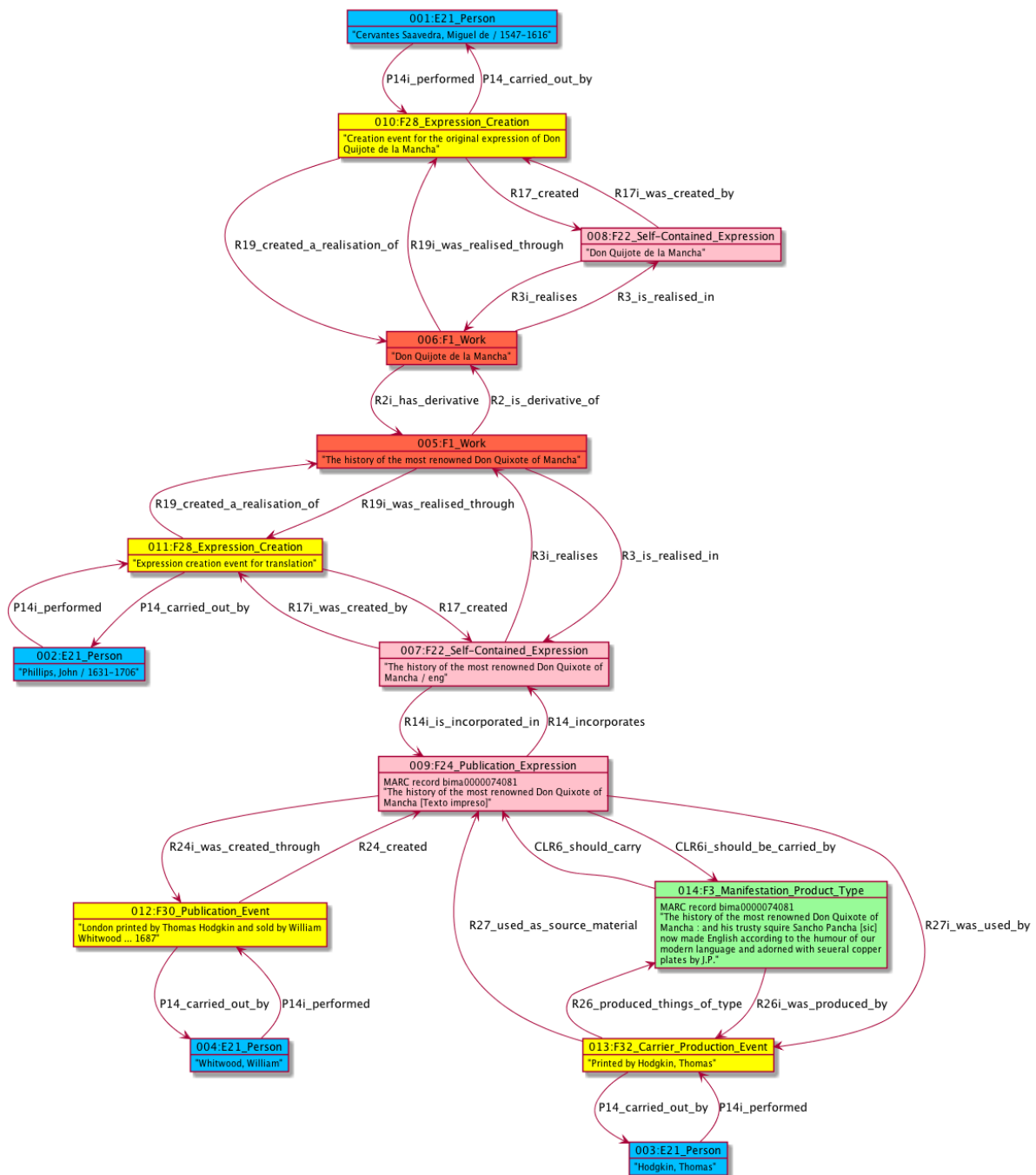
- **Always a Self-Contained Expression (both models)**

To have a consistent use of entities across all records we are always using a Publication Expression that incorporates at least one Self-Contained Expression.

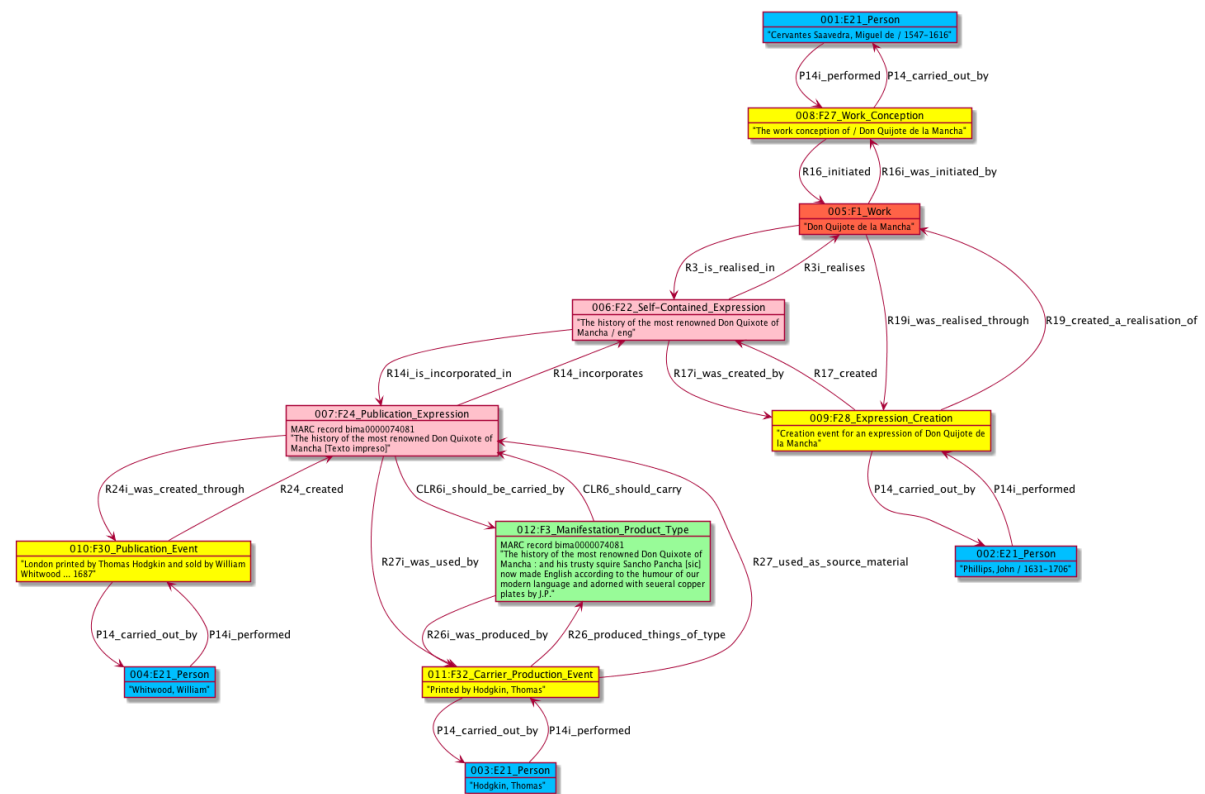
Example FRBROO models

The following two figures show some of the design choices in our alternative models (see <http://november.idi.ntnu.no/frbrized/rest/db/edm/index.html> for complete set of records).

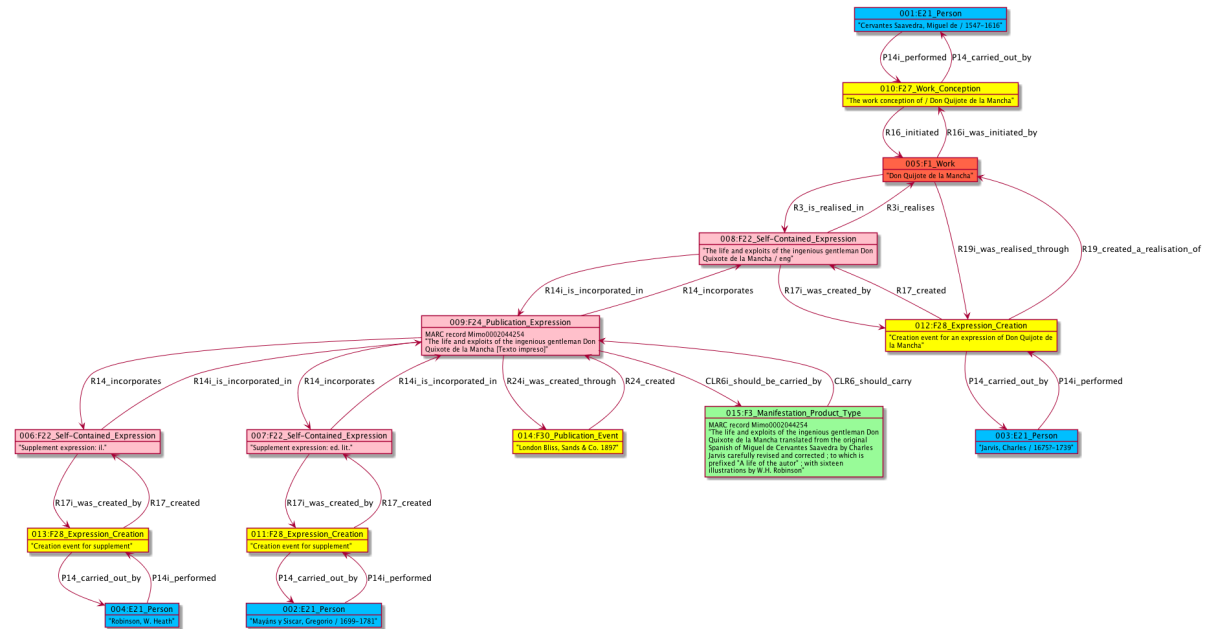
Derivation-based



Realisation-based

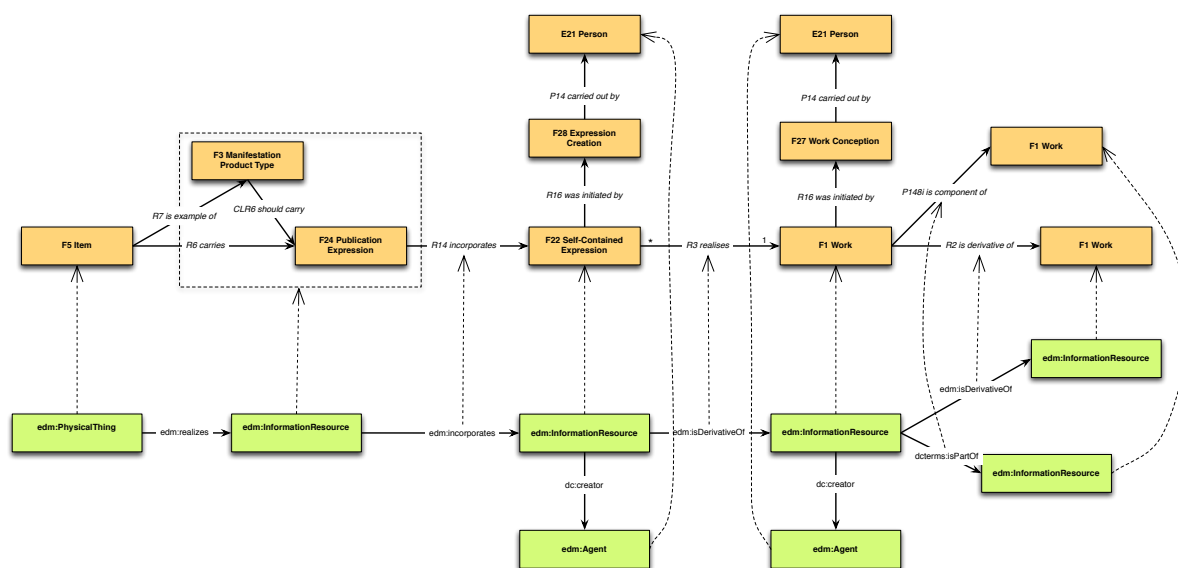


Realisation-based with multiple incorporated expressions



Mapping rules

Mapping the realization-based FRBRoo-model to EDM



The diagram illustrates the relationships between various entities and their realizations in a model. The entities are represented by orange boxes, and the relationships are represented by arrows with labels.

Entities:

- edm:PhysicalThing
- edm:InformationResource
- edm:Agent
- F3 Item
- F3 Manifestation Product Type
- F24 Publication Expression
- F22 Self-Contained Expression
- F1 Work
- E21 Person
- F28 Expression Creation

Relationships:

- edm:PhysicalThing *edm:realizes* edm:InformationResource
- edm:InformationResource *edm:incorporates* edm:InformationResource
- edm:InformationResource *dc:creator* edm:Agent
- edm:InformationResource *dc:isPartOf* edm:InformationResource
- edm:InformationResource *edm:DerivativeOf* edm:InformationResource
- edm:InformationResource *dc:creator* edm:Agent
- edm:InformationResource *dc:isPartOf* edm:InformationResource
- F3 Item *R7 is example of* F3 Manifestation Product Type
- F3 Manifestation Product Type *CLR8 should carry* F24 Publication Expression
- F24 Publication Expression *R14 incorporates* F22 Self-Contained Expression
- F22 Self-Contained Expression *R3 realizes* F1 Work
- F1 Work *P148 is component of* F22 Self-Contained Expression
- F1 Work *R2 is derivative of* F1 Work
- F1 Work *R19 was realized through* F22 Self-Contained Expression
- F22 Self-Contained Expression *P16 was initiated by* F1 Work
- F1 Work *P148 is component of* F1 Work
- F28 Expression Creation *P14 carried out by* E21 Person
- F28 Expression Creation *R16 was initiated by* F1 Work
- F28 Expression Creation *R19 was realized through* F1 Work
- F28 Expression Creation *P148 is component of* F1 Work

Notes:

- E.g. the original expression and work, or it can be a derived expression/work e.g. in the case of a translation
- This part will only be included in the model if the incorporated F22 is NOT the original expression

Mapping single records is potentially a deceiving process. A solution that applies to one record may cause errors if applied to other records. Mapping from FRBRoo to EDM has to be verified by:

- Applying the mapping to a representative set of records that includes relevant variations
- Additionally, the mapping must be verified by evaluating the result from merging/combining records. Solutions that work well for each record in a set, does not necessarily give the desired result when merging entities from multiple records.

Mapping rules explored

We have explored mapping using FRBRoo records in rdf as input for an XSLT-based mapping procedure. The result of the process is EDM in rdf which then has been transformed to graphs using plantUML.

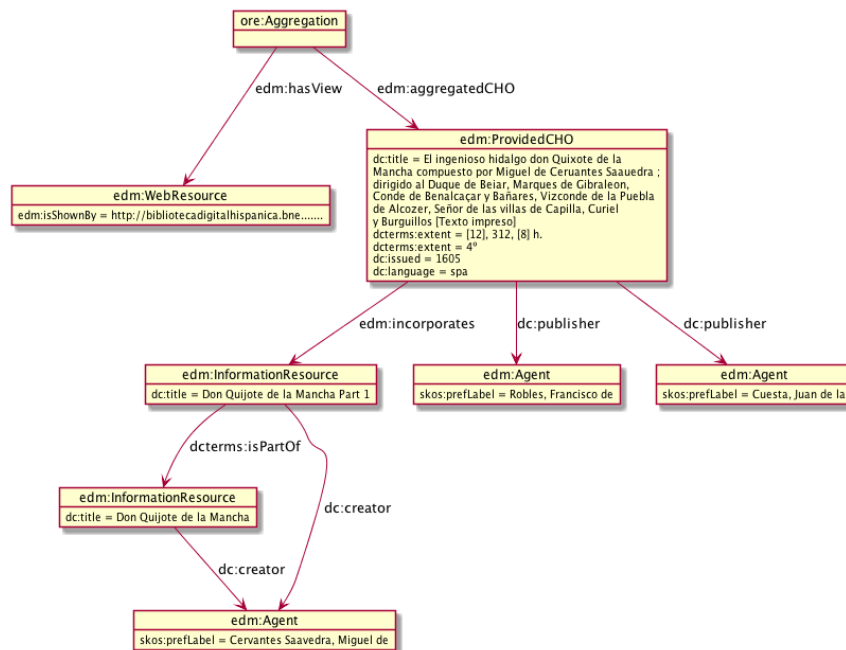
This is a preliminary description of ideas and challenges:

- **edm:InformationResource for F3 and F24**
Because of the 1:1 relationship between F3 Manifestation Product Type and F24 Publication Expression, both entities can be mapped to a single edm:ProvidedCHO.
- **edm:incorporates and edm:InformationResource**
Publication Expression that incorporates multiple Self-Contained Expression are mapped using an InformationResource for the Self-Contained Expression.
PS! Ideally we wanted to avoid using an edm:InformationResource when there is only a single Self-Contained Expression, but it turns out that this creates inconsistencies between records where this expression is the only content and records where this expression is one of several.
- **edm:Agent**
All persons are treated as first class objects and mapped to edm:Agent
- **dc:creator and dc:contributor**
Persons with primary and secondary responsibility for the resource are expressed as dc:creator. This is maybe too generic, because it does not make it possible to use this property to identify works.
- **mapping F1 Work and F22 Self-Contained Expression to edm:InformationResource instances**
The use of edm:InformationResource for both F1 Work and F22 Self-Contained Expressions seems to be unproblematic. The graphs should be able to answer queries related to persons names, titles etc regardless of the actual type of instances. The most crucial is that the graph collocates InformationResource that represent the same kind of entity.
- **Reducing the number of InformationResource objects**
A straight-forward transformation of FRBRoo models to a corresponding structure based on InformationResource will in some records appear to create “redundant” objects. This is typically found in the realisation-based models (e.g. see realisation-based edm-model for bima0000007171).
The reduction of entities is rather logical and straight forward to implement for the derivation-based models. In this case we could consistently merge all works and expressions into a single InformationResource. In the derivation-based

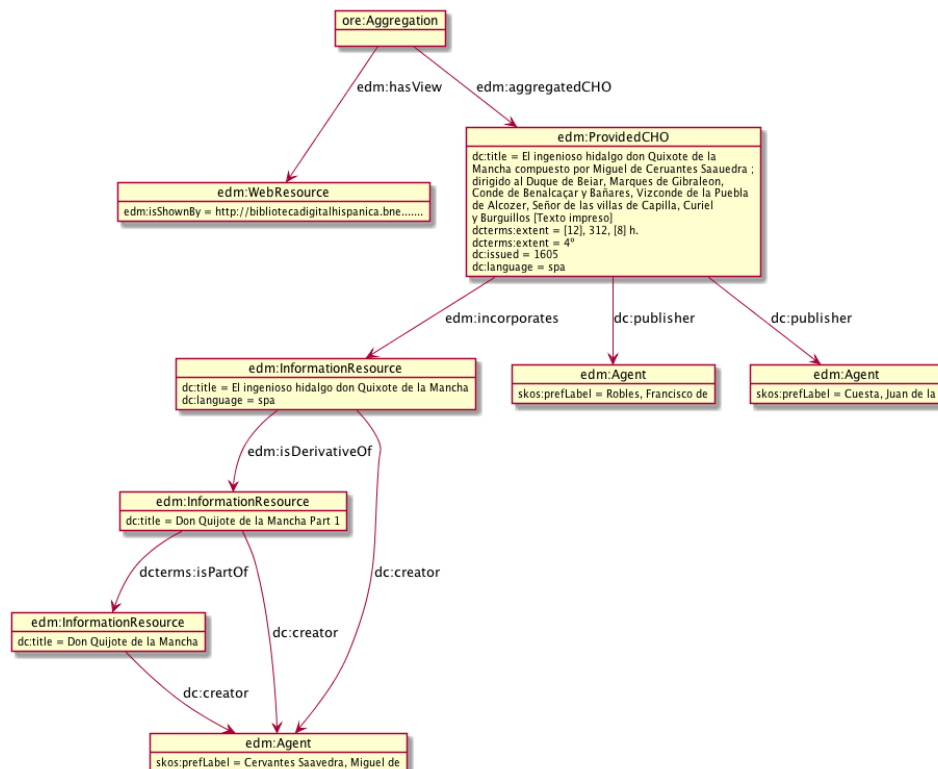
model we implicitly include “original expressions” which also means that we kind of create relationships between expressions (although through derived works). This kind of merging is more difficult to implement consistently for realisation-based models. In some cases we want to have different InformationResources for eg. translation and the work it realises, but in other cases (mainly for the expressions created by Cervantes that realises his own work) it is sufficient to have a single InformationResource.

Example result 1 (different results):

Mapped from derivation-based:

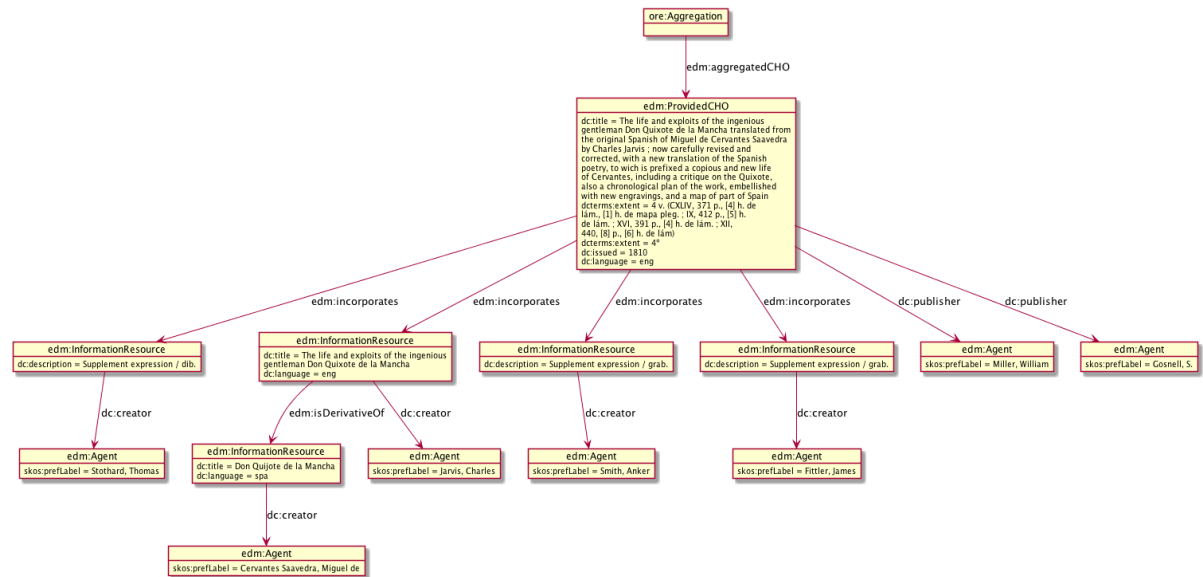


Mapped from realization-based:

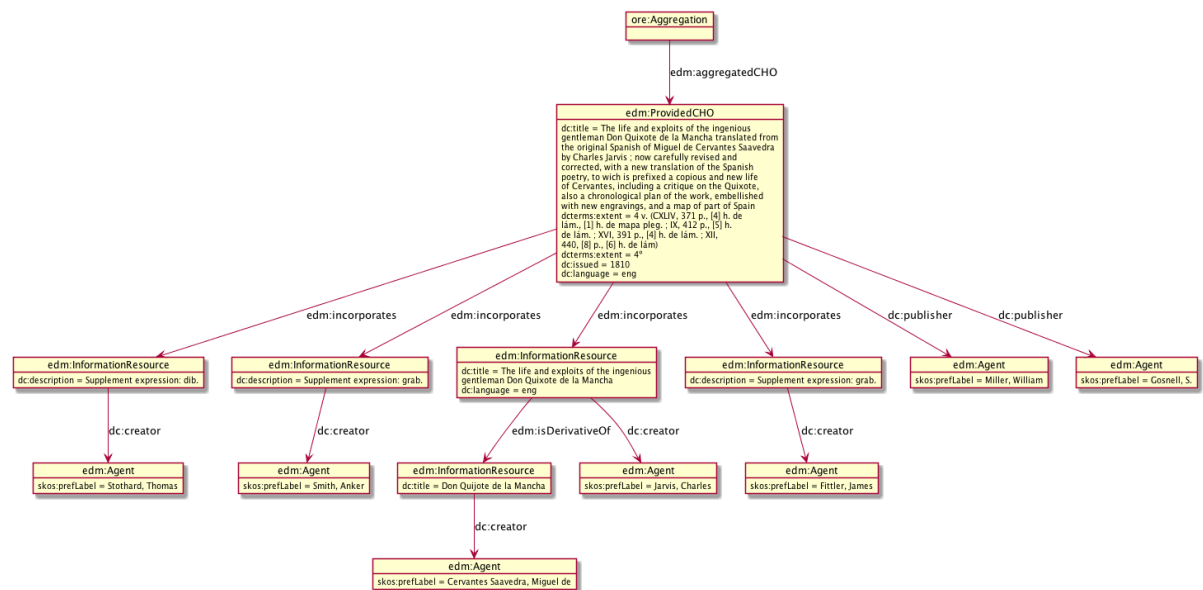


Example 2 (results are the same)

Derivation-based:

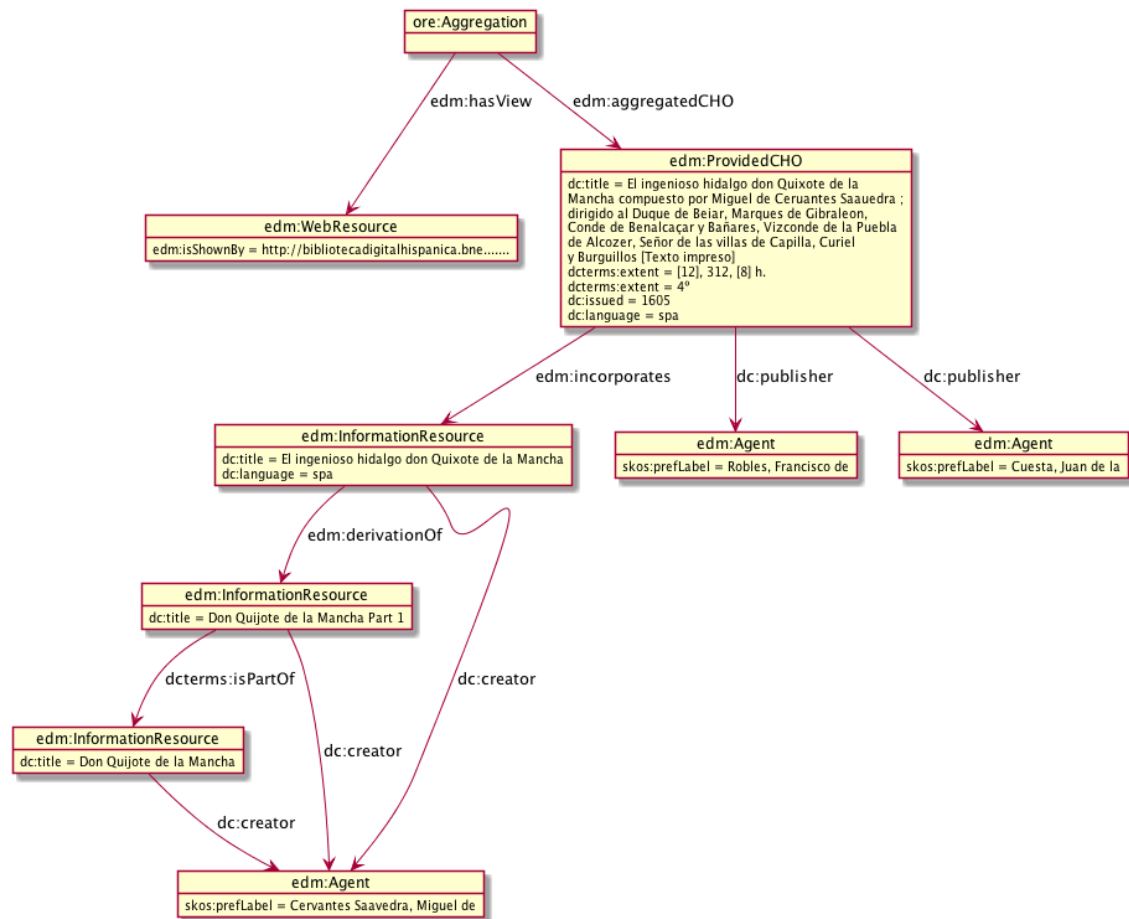


Realisation-based:



The use of dc:creator for works and expressions

Is sometimes problematic when we include both the work and expression as distinct edm:InformationResources. The following example is not able to answer our question of what works Cervantes have created because he is also the creator of an expression.



The problem is that we often need to include the creator of the expression such as in the case of translations:

