Elias Tzortzakakis (ICS-FORTH): Draw.io to Triples

Presentation at the 57th CIDOC CRM

Objective

Use a draw.io diagram as the visual representation of a CIDOC-CRM example in order to

- transform the diagram to triples
- Validate the diagram

Example input

https://app.diagrams.net/
Example output
Requirements

- **Instances:** Each set of boxes must define 1 instance. It must consist of
  - 1 Class definition box
  - 0..1 rdfs:label box
  - 0..1 image (image existence is actually translated to P138i has representation(x,y) where x the current Class instance and y a E36 Visual item with rdfs:label the base64 encoding of the image

- **Properties:** Each arrow must have been appropriately connected to the instances and include an integrated (not a separate object) text value that identifies it

Error Handling

- No triples file creation in case there exist error messages. Error messages refer to detected inconsistency cases such as incorrect domain/range specification of properties. May result in correction of:
  - arrow direction
  - arrow identifier label
  - property domain or range instance

- Output of Warnings and partial output with additional information of what could not be transformed into triples (e.g. for manual handling)
  - Interpretation of instances as Primitive classes
  - Properties with domain a primitive class
  - Unknown classes/properties usage

[https://isl.ics.forth.gr/cidoc_services/](https://isl.ics.forth.gr/cidoc_services/)

Username/password crm_user/cidoc_services
Then navigate to option “Draw.io diagrams to triples”

[https://demos.isl.ics.forth.gr/RDFV/](https://demos.isl.ics.forth.gr/RDFV/)

TODOS

- MD proposed to add this as a didactic tool
- Reporting of wrong labels
- Multiple inheritance
- PC classes
- Configuration input
  - Namespace
- Models to load (different CRM version, FMs, PC Classes etc)