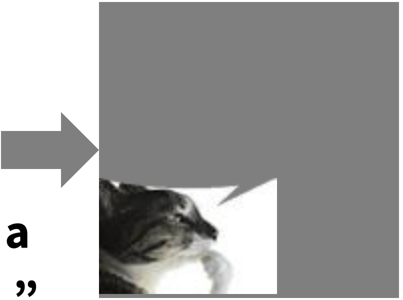
Master thesis

Brainstorm

RML SHACL-form

INPUT

“We want to create a SHACL form for RML”



OUTPUT

Master thesis

We don’t want just an implementation but a RESEARCH QUESTION

-> **Challenge:** Moving from “just coding” to a scientific approach.

What I have done so far...

1. Theoretical Background

- Analyzed RML & SHACL concepts.
- Tried some simple implementation exercices to be familiar with the topic

2. State-of-the-art

- Tested existing SHCL-forms: (Darmstadt, DanieleBeecke, Shaperone)
- Compared them

3. Methodology /Analyse

- Find a way to decide which one to choose
- -> Decided to compare them base on the sh: features they handle or.
- -> Listed the most useful features for an RML form
  - sh:node, sh:or, existing instances…
- -> Choose the best one accordingly
  - Darmstadt

4. Implementation

- Started coding and implementing…

⬆

where I am now
(8. dec. 2025)

Add here excel comparaison

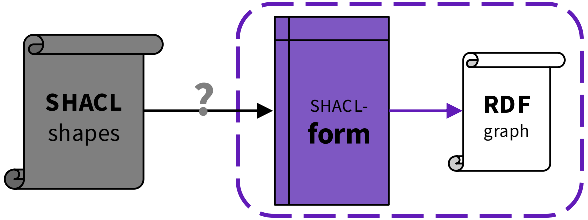
What about the research question ?

Introduction

(see Theoretical Background)

RDF, SHACL, RML

Problem



Related Work

(see State-of-the-art)

Existing SHACL forms (Darmstadt, Shaperone, DanielBeeck) -> handles a subset of SHACL

Research question

~~Is it possible to generate a form that handles any SHACL shape?~~

More concretely

➔ Are there "form-specific" shapes?

➔ Does the SHACL form has to be specific to the (type of) shape we plan to encode?

Use-case: RML. Why?

➔ Complex and useful RDF graph

➔ ∃ existing official shapes to

- test limitations of a form, and
- see how to make it "form-friendly"

Objective 1:

Design a shape for RML, *suitable* for a form. (Define what that means)

Proposal:

Simplify the RML-core shapes given by kg-construct

Objective 2:

Conceptualize the form for RML.

Proposal:

From the state-of-the-art, compare which one has best potential for my use-case by listing (necessary) features

Objective 3:

Pour aller plus loin.. Implement reusable nodes in the form

Proposal:

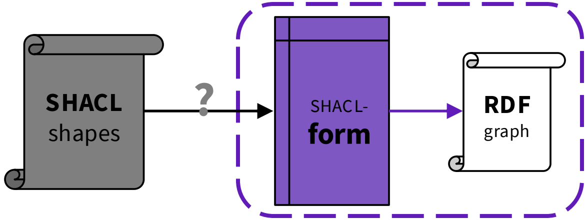
Make use of existing feature of the selected state-of-the-art.

Introduction

(see Theoretical Background)

RDF, SHACL, RML

Problem



Related Work

(see State-of-the-art)

Existing SHACL forms (Darmstadt, Shaperone, DanielBeeck) -> handles a subset of SHACL

Research question

How can we conceptualize ~~form-friendly~~ SHACL shapes for complex domains such as RML?

More concretely

➔ Which features of SHACL (core W3C) are compatible with automated form generation, and which one require adaptations or are fundamentally incompatible?

➔ What design strategy enable us to reconcile validation (SHACL) and user usability (for form).

Use-case: RML. Why?

➔ Complex and useful RDF graph

➔ ∃ existing official shapes to

- test limitations of a form, and
- see how to make it “~~form-friendly~~”

<u>Objective 1:</u>

Design a shape for RML, ~~*suitable*~~ for a form. (Define what that means)

Proposal:

Simplify the RML-core shapes given by kg-construct

<u>Objective 2:</u>

Conceptualize the form for RML.

Proposal:

From the state-of-the-art, compare which one has best potential for my use-case by listing (necessary) features

<u>Objective 3:</u>

Pour aller plus loin.. Implement reusable nodes in the form

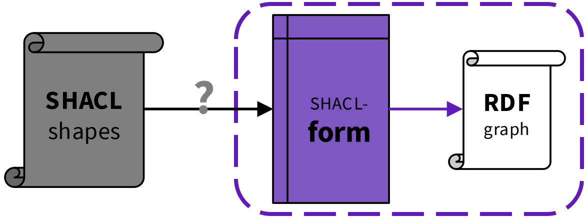
Proposal:

Make use of existing feature of the selected state-of-the-art.

What about the research question ? (again)

idea: from the state-of-the-art,
→ see what's missing
→ see what are the problems

Main Core Problems



We have a disconnect between the tool and the data structure:

1. **The "Perfect Form" doesn't exist:**

- Current tools handle subsets of SHACL, (not every sh: features handled)
- We cannot just "plug and play" any shape. (too complex nested shapes leads to impossible display for the form)

2. **Tree vs. Graph:**

- A form will generate a **Tree** (due to HTML hierarchy/constraint).
- RDF is fundamentally a **Graph**, not just a Tree.
- -> How to deal with it? Artificially modifiy the shape by adding nodes outside of the namespace?

Evolution of the research question

✕ **Attempt 1 (Too broad):**

"Is it possible to generate a form that handles *any* SHACL shape? »

Critique: Likely unachievable or not specific enough.

✕ **Attempt 2 (Vague):**

"How can we conceptualize *_form-friendly_* SHACL shapes for complex domains such as RML? »

Critique: What does "form-friendly" actually mean?

Attempt 3 (Currently exploring):

Is it possible to create a SHACL form that facilitates graph creation (not just tree), specifically via the reuse of nodes created within the form?

Use-case: RML. Why?

- ➔ Fundamentally a Graph, not a tree.
- ➔ ∃ existing official shapes to test limitations of a form
- ➔ Complex and useful RDF graph

1

Create Shapes & Form:

Simplify RML-core shapes (from kg-construct) to be suitable for a form

2

Conceptualize Tree-> Graph:

Enable the possibility to
- Create multiple nodes
- Reuse nodes created elsewhere in the form (bridging the tree-graph gap)

3

Evaluation:

Test the design strategy to reconcile validation (SHACL) and usability

Futur work & Questions

- **Perfect Form:** Does it even exist? Or is the goal just to bridge the specific gap for graph-based editing?
- **User Experience:** Should we include a UX evaluation? (Would require Test Protocol)
- **Modularity:** Can the form itself become modular, mirroring RML's modularity?

Implementation Status (Darmstadt Fork)

TODOs current focus:

i. Improving sh:or behavior to look more like sh:in.
ii. Adding default values for dropdowns (shortcut rml:subjectMap – rml:subject)
iii. Handling prefixes (~OK, but check if needs adaptation).
iv. **The Multi Node challenge:**
    i. Now handled dumbly
    ii. Would like to not touch the shape. Or at least be a bit more intelligent..
v. **The Node Reuse challenge:**
    i. Try to make use of the setClassInstance provider

Technical Challenges:

- Most forms require a « Root Node »
- setClassInstance provider works for data given beforehand, not yet for nodes created inside the form

References (à completer)

Links:

- https://github.com/ULB-Darmstadt/shacl-form
- https://kg-construct.github.io/rml-core/spec/docs/#mapping-graphs-and-the-rml-vocabulary
- https://www.w3.org/TR/shacl/
- https://medium.com/@amivanoff/the-new-rml-cheat-sheet-map-everything-to-the-rdf-with-ease-ba6a767edfaf

Papers:

- https://link.springer.com/chapter/10.1007/978-3-031-39141-5_23
- https://essay.utwente.nl/fileshare/file/96336/viste_BA_EEMCS.pdf