

Access Predicates Demo

Revision: March 7, 2017; Rendered: May 25, 2021

Illustration of the computation of access predicates for an RDF fact base. Makes use of `scratch_forgetting` and `scratch_definientia`. Formalized with the *PIE* system.

1 Introduction

We assume a RDF knowledge base with facts like

```
rdf_triple(p1, name, 'Sulzer').  
rdf_triple(p1, yob, 1720).
```

2 Specifications of Access Predicates

Specifications of access predicates for `person_name` and `person_yob`. In practice, these would be automatically generated from the schema of the knowledge base.

accessor_spec

Defined as

$$\begin{aligned} \forall pn (bp \rightarrow (\text{person_name}(p, n) \leftrightarrow \text{person_name_bf}(p, n))) & \quad \wedge \\ \forall pn (bn \rightarrow (\text{person_name}(p, n) \leftrightarrow \text{person_name_fb}(p, n))) & \quad \wedge \\ \forall pn (bp \rightarrow (\text{person_yob}(p, n) \leftrightarrow \text{person_yob_bf}(p, n))) & \quad \wedge \\ \forall pn (bn \rightarrow (\text{person_yob}(p, n) \leftrightarrow \text{person_yob_fb}(p, n))) & \quad \wedge \\ \forall pn (\text{person_name}(p, n) \rightarrow bp \wedge bn) & \quad \wedge \\ \forall pn (\text{person_yob}(p, n) \rightarrow bp \wedge bn). & \end{aligned}$$

accessor_spec₁

Defined as

$$\begin{aligned} \forall pn (bp \rightarrow (\text{person_name}(p, n) \leftrightarrow \text{person_name_bf}(p, n))) & \quad \wedge \\ \forall pn (bn \rightarrow (\text{person_name}(p, n) \leftrightarrow \text{person_name_fb}(p, n))) & \quad \wedge \\ \forall pn (\text{person_name}(p, n) \rightarrow bp \wedge bn). & \end{aligned}$$

3 Some Example Queries, Processed by Interpolation

$b(n)$ in the background formula effects that `person_name` is rewritten to `person_name_fb`:

rewrite₁

Defined as

$$\text{definiens}(\text{person_name}(p, n), \\ \text{accessor_spec}_1 \wedge \text{bn}, \\ [\text{person_name_bf}, \text{person_name_fb}]).$$

Input: *rewrite₁*.

Result of interpolation:

$$\text{person_name_fb}(p, n).$$

This is the macro expansion of **rewrite₁**:

$$\begin{array}{l} \exists qr (\forall xy (qx \rightarrow (rxy \leftrightarrow \text{person_name_bf}(x, y))) \wedge \\ \quad \forall xy (qy \rightarrow (rxy \leftrightarrow \text{person_name_fb}(x, y))) \wedge \\ \quad \forall xy (rxy \rightarrow qx \wedge qy) \wedge \\ \quad qn \wedge \\ \quad rpn) \rightarrow \\ \neg \exists qr (\forall xy (qx \rightarrow (rxy \leftrightarrow \text{person_name_bf}(x, y))) \wedge \\ \quad \forall xy (qy \rightarrow (rxy \leftrightarrow \text{person_name_fb}(x, y))) \wedge \\ \quad \forall xy (rxy \rightarrow qx \wedge qy) \wedge \\ \quad qn \wedge \\ \quad \neg rpn). \end{array}$$

For given name compute years in which a person with that name has been born:

rewrite₂

Defined as

$$\text{definiens}(\exists p (\text{person_name}(p, n) \wedge \text{person_yob}(p, y)), \\ \text{accessor_spec} \wedge \text{bn}, \\ [\text{person_name_bf}, \text{person_name_fb}, \text{person_yob_bf}, \text{person_yob_fb}]).$$

Input: *rewrite₂*.

Result of interpolation:

$$\exists x (\text{person_name_fb}(x, n) \wedge \text{person_yob_bf}(x, y)).$$

4 Referential Constraints

The predicate `person` can be accessed to “enumerate” all persons. There is a referential constraint from `person_name` to `person`.

person_spec

Defined as

$$\begin{aligned} & \forall p (\text{person}(p) \rightarrow \text{bp}) \quad \wedge \\ & \forall pn (\text{person_name}(p, n) \rightarrow \text{person}(p)). \end{aligned}$$

rewrite₃

Defined as

$$\begin{aligned} & \text{definiens}(\exists p \text{ person_name}(p, n), \\ & \quad \text{person_spec} \wedge \text{accessor_spec}, \\ & \quad [\text{person}, \text{person_name_bf}, \text{person_name_fb}]). \end{aligned}$$

Input: *rewrite₃*.

Result of interpolation:

$$\exists x (\text{person}(x) \wedge \text{person_name_bf}(x, n)).$$

5 Different Proofs – Different Interpolants

Here only the second returned interpolant is without the redundant occurrence of *person*.

rewrite₄

Defined as

$$\begin{aligned} & \text{definiens}(\exists p (\text{person_name}(p, n) \wedge \text{person_yob}(p, y)), \\ & \quad \text{person_spec} \wedge \text{accessor_spec} \wedge \text{bn}, \\ & \quad [\text{person}, \text{person_name_bf}, \text{person_name_fb}, \text{person_yob_bf}, \text{person_yob_fb}])). \end{aligned}$$

Input: *rewrite₄*.

Result of interpolation:

$$\exists x (\text{person}(x) \wedge \text{person_name_fb}(x, n) \wedge \text{person_yob_bf}(x, y)).$$

Input: *rewrite₄*.

Result of interpolation:

$$\exists x (\text{person_name_fb}(x, n) \wedge \text{person_yob_bf}(x, y)).$$

Index

<i>accessor_spec</i> , 1	<i>rewrite₁</i> , 2
<i>accessor_spec₁</i> , 1	<i>rewrite₂</i> , 2
	<i>rewrite₃</i> , 3
<i>person_spec</i> , 3	<i>rewrite₄</i> , 3