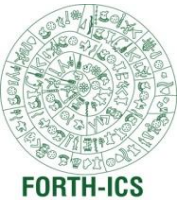
Design and Implementation of a tool for formulating recall-oriented structured queries on semantic networks

Katerina Tzompanaki

Advisors:

Martin Doerr

Yannis Tzitzikas



Institute of Computer Science
Foundation for Research and Technology - Hellas

Computer Science Department
University of Crete

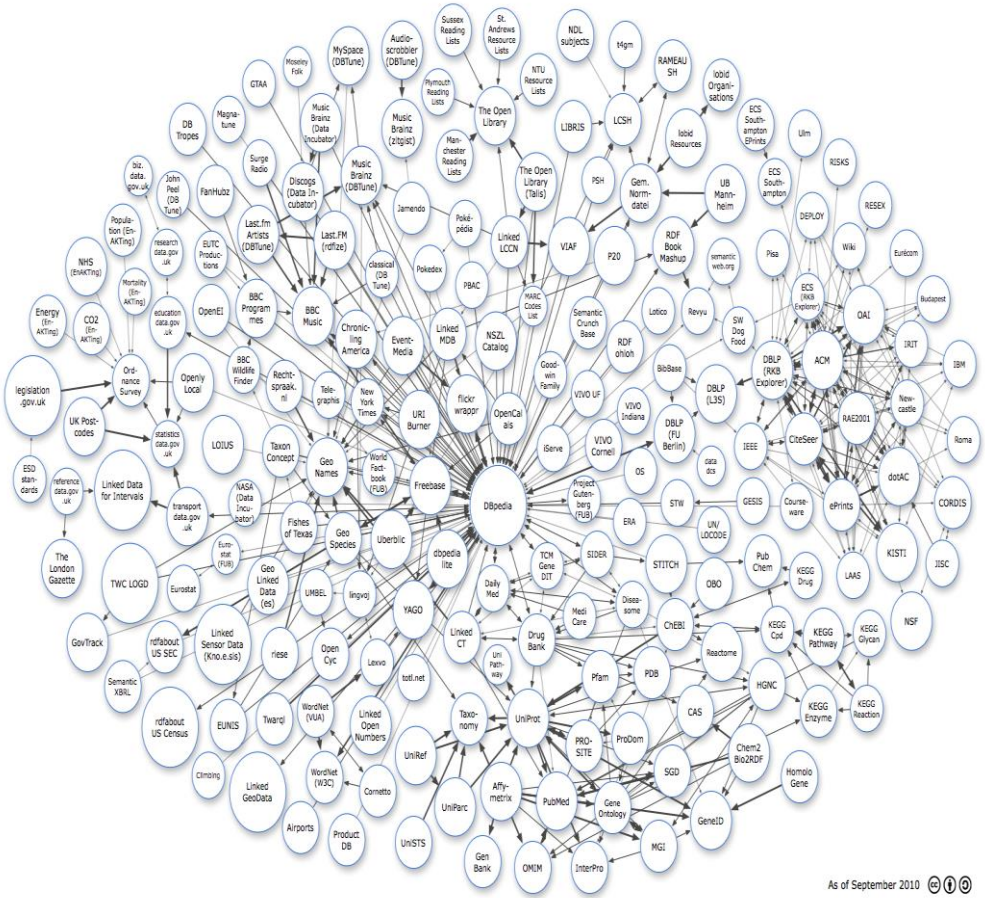
July 24, 2012

Outline

- ***Introduction***
- ***Problem***
- ***Related Work***
- ***Our proposal***
- ***Example***
- ***Implementation***
- ***Validation***
- ***Contributions***
- ***Live Demo***

Introduction

Semantic Web



- *Web of data*
- *RDF/S*
- *RDF Triple Stores*
- *Linked Open Data (LoD):* *Thousands of triple stores to be accessed*

Triple Stores of:

- Heterogeneous data
- social and historical events

- composite structure
- diverse semantics
- multiple kinds of relationships

3 Major Challenges

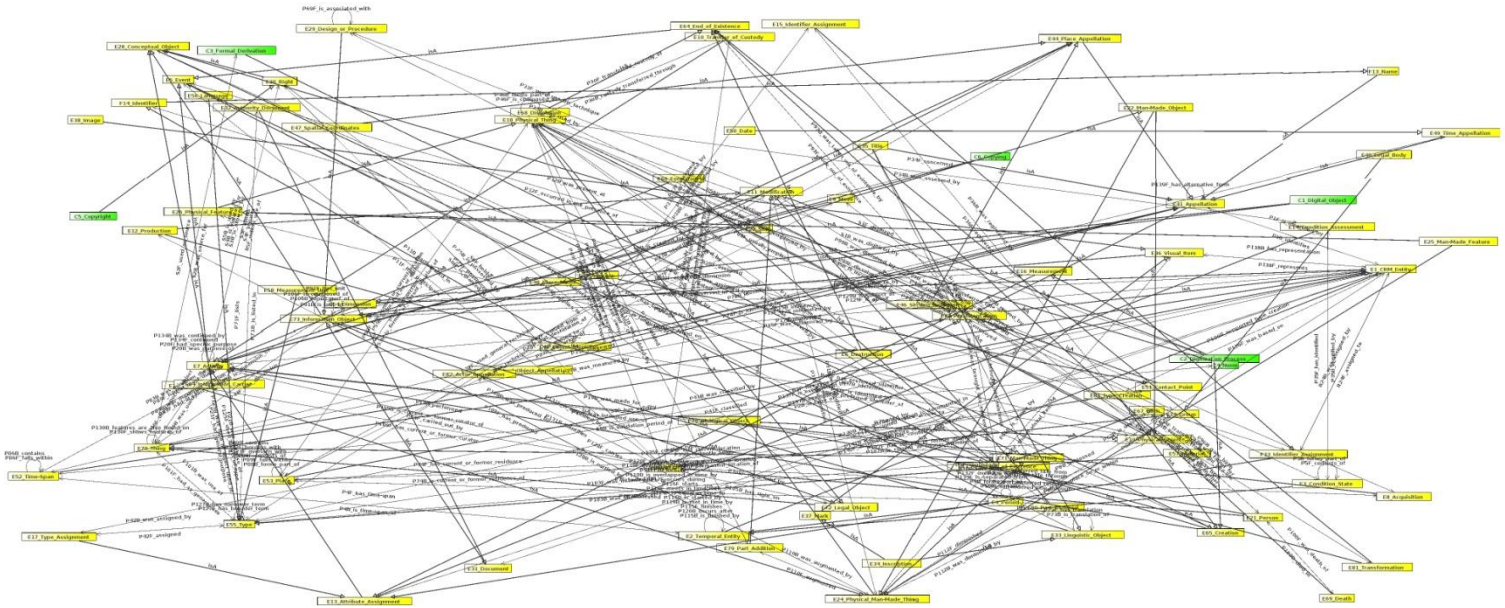
1. **Store:** rich global schema
2. **Query:** effective and easy
3. **Integrate:** reasoning and co-reference resolution

Extensible **core ontology**

- 86 classes
- 137 properties
- museum disciplines, archives and libraries,
- transform, transport and merge information

Problem

How to query rich Semantic Networks?



CIDOC-CRM Visualization by StarLion

How to query rich semantic networks?

Hardships:

- Incomplete information
- Users' ignorance
- querying methods
  - Declarative (keyword search) → *insufficient*
  - Structured (SPARQL) → *complicated*

Related Work

User Interface query formulation techniques:

- **Drawbacks**:
  - SPARQL knowledge dependency
  - Schema structure knowledge dependency
  - Polysemy of natural language

Existing Approach 2

Use of

- rules
- views

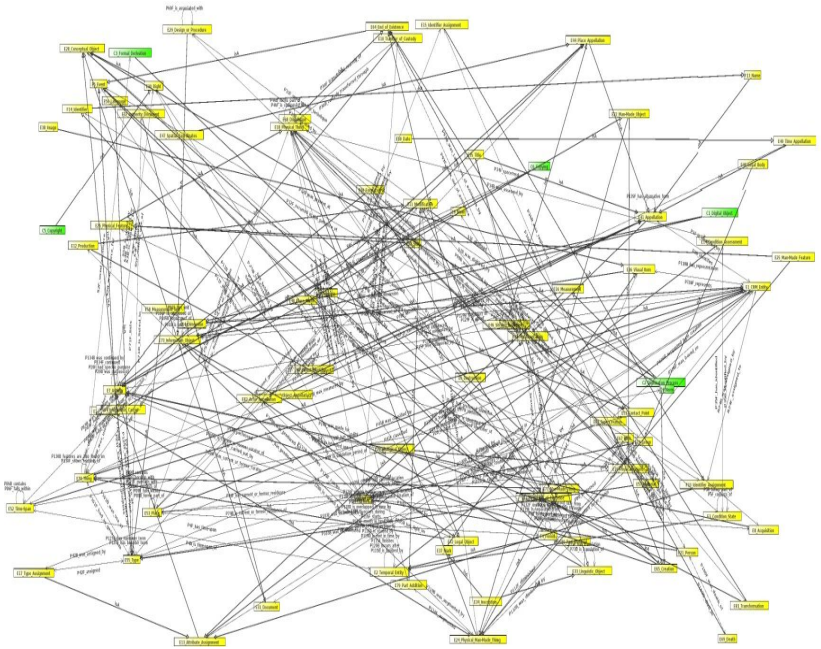
for queries representation (instead of SPARQL)

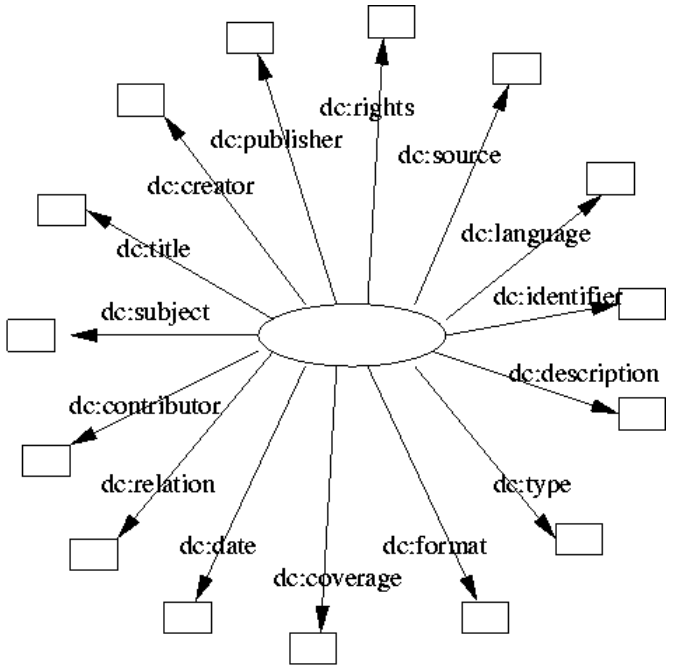
Drawbacks

- requires technical knowledge
- materialization (rules/views) → update/modification difficulties
- performance issues for massive repositories

Existing Approach 3 (1/2)

***Simplify** the network by using "core" elements such as in Dublin Core*





The DC hedgehog model

Existing Approach 3 (2/2)

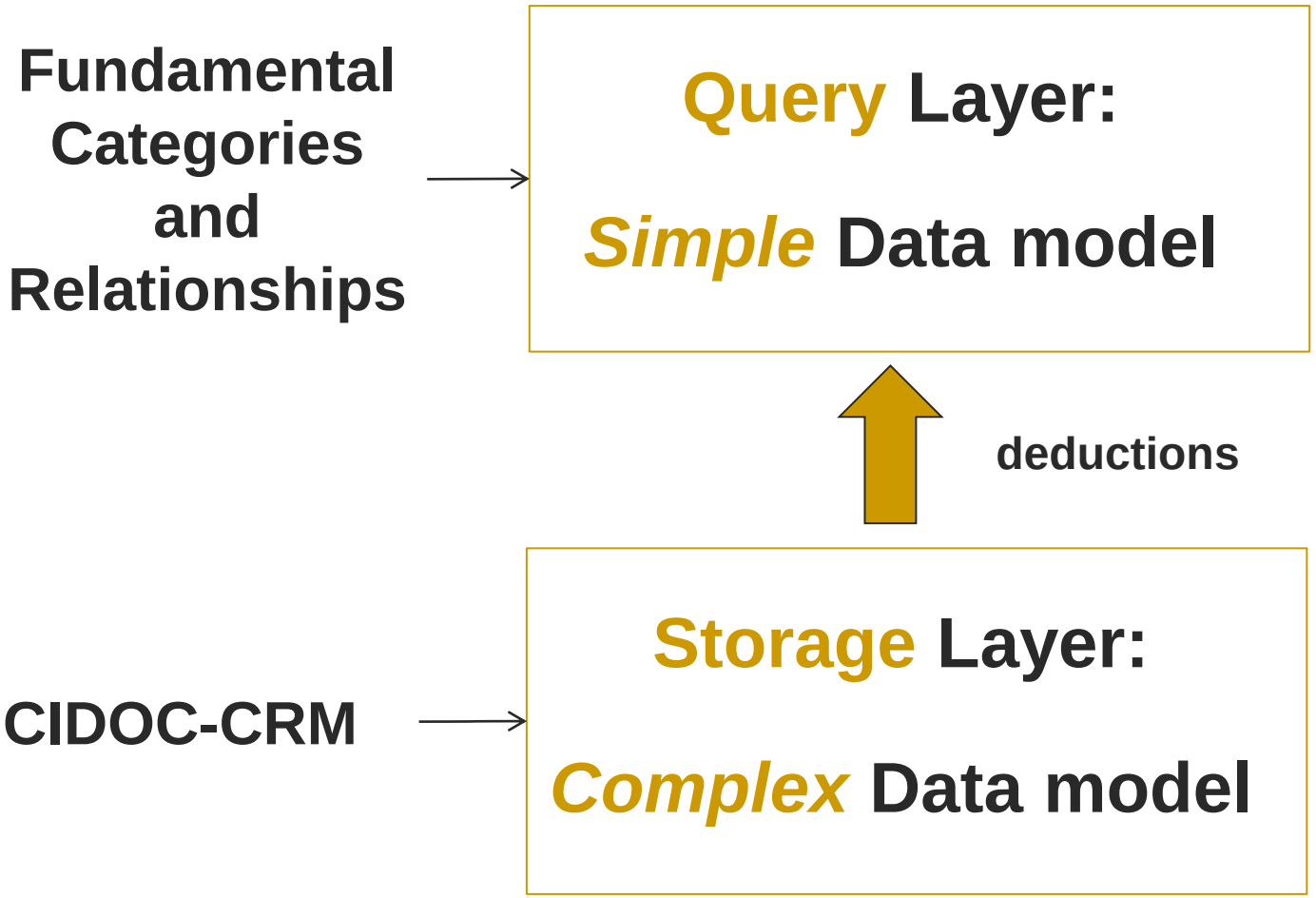
Advantages

- Simplicity

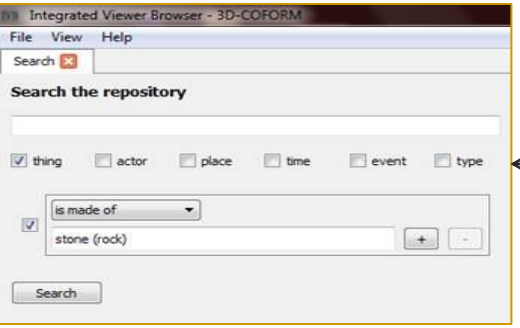
Disadvantages

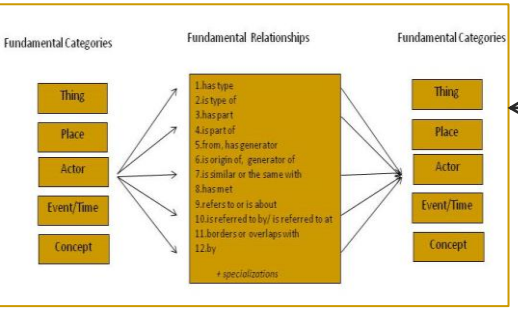
- Poor scientific knowledge mapping
- Low precision
- Lack of expressivity
- Lack of reasoning capability
- Lack of integration capability

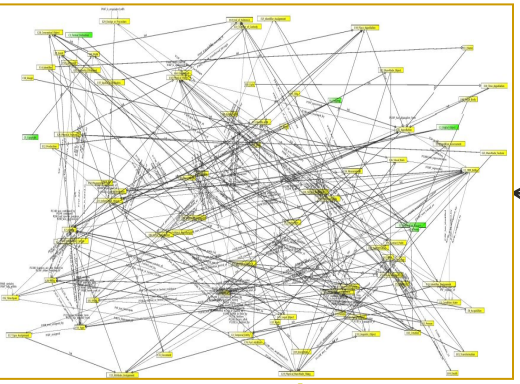
Our Proposal


Fundamental Categories and Relationships
Query Layer:
Simple Data model
deductions
CIDOC-CRM
Storage Layer:
Complex Data model

General System View







Interface

12 Fundamental Relationships!

Fundamental Categories (FCs) and Fundamental Relationships (FRs) Model

One hundred paths

Path's expression language & Fundamental Relationships configuration tool

Complex semantic network

Hundreds of properties

Our Proposal

Fundamental Relationships (FRs)

- **describe:**
  - classification and part-whole structure
  - history
  - subject
- **based on:**
  - intuition
  - experience
  - observation

Fundamental Categories:

- Thing, Actor, Place, Time/Event, Concept

Fundamental Relationships:

- has type /is type of
- is similar to or same with
- is part of (is member of) / has part (has member)
- has met
- from (has founder or has parent) / is origin, founder, parent, creator of
- had (=owns, keeps) / were owned/kept by
- refers to or is about / is referred to by - is referred to at

.. and specializations

Different interpretations of ‘ from’ FR

Thing from Place

- The former or current location of the thing.
- The place where the Thing was created, found or acquired.
- The place where the owner or keeper of the thing resides.
- The place associated (birth, residence) with the creator of the thing

Actor from Event

- The birth event of a person
- The formation event of a Group

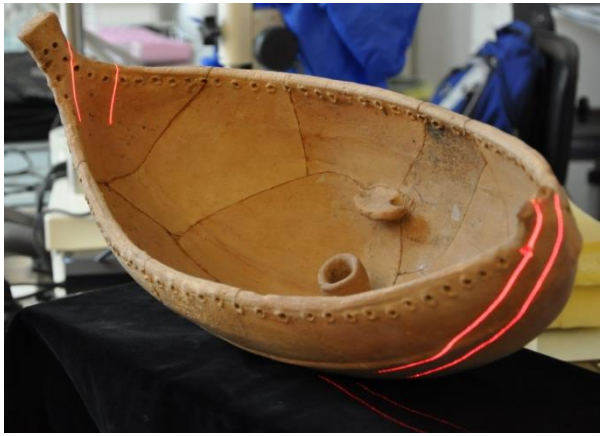
Example

Digitization of the Kazafani Boat

Example:

The "Kazafani Boat"

Found in 1963, during a salvage excavation in the now Turkish occupied part of Cyprus (inaccessible and destroyed site).
Tomb from the 12th century B.C.
Unique object, hand made pottery
40x20.5x23 cm – canoe boat shape
Permanently exhibited at the Nicosia Museum



Workflow

3D scanning – NextEngine

3D model creation – Meshlab

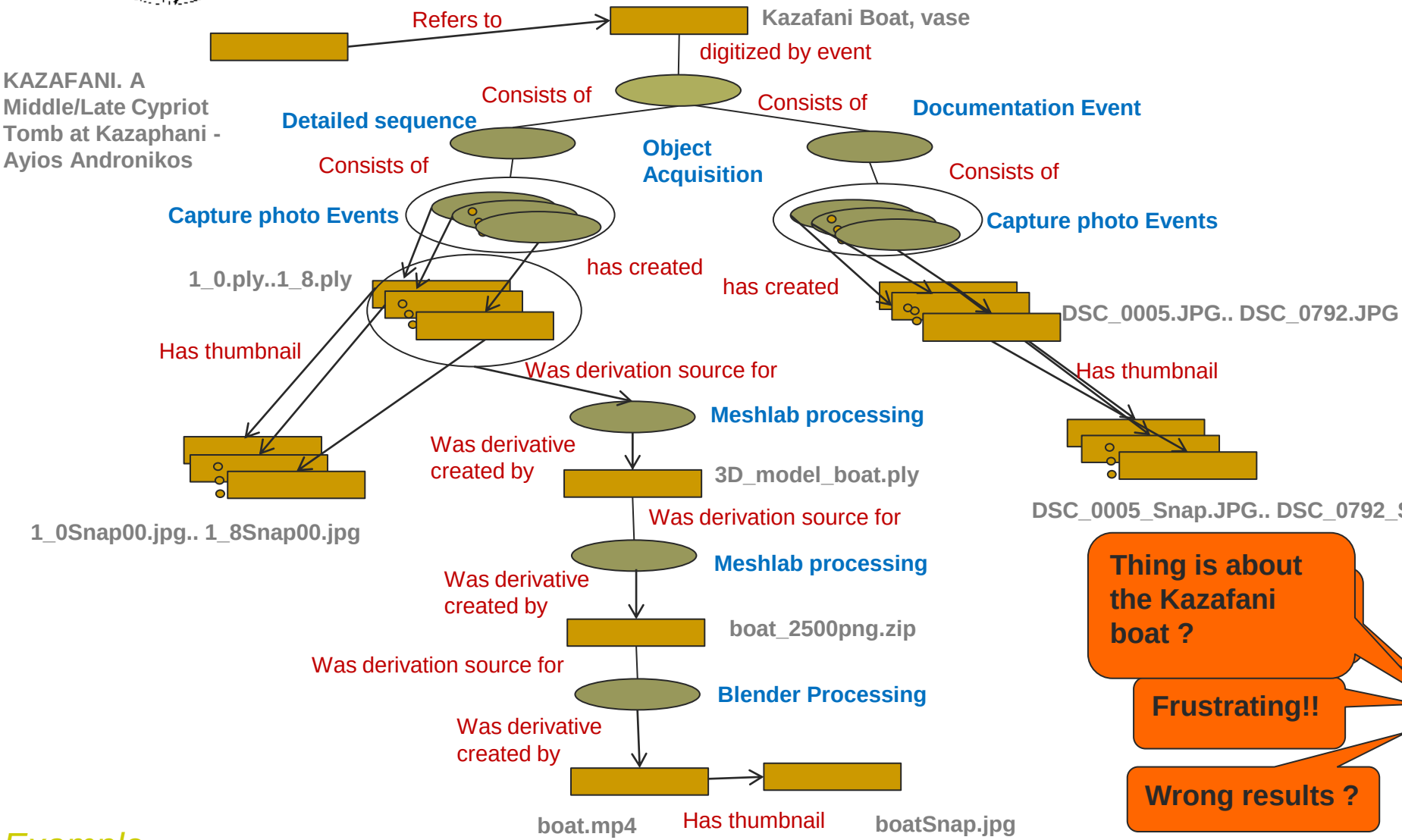
Rapid prototyping

Testing glue, stabilizers, colours

Print final replica

Colour final replica

Metadata tree on the Kazafani boat



Example

Simplified Querying

Traditional way

```
select distinct  $Thing1  $Label
{
$Thing1 rdf:type crm:E70.Thing.
        optional{$Thing1 crmdig:L4F.has_preferred_label $Label.
        }{optional{
        $Thing1 crm:P130F.shows_features_of $Thing2.
        }}UNION
        {optional{
         $Thing1 rdf:type crm:E24.Physical_Human-Made_Thing.
        $Thing1 crm:P62F.depicts $Thing2.
        }}UNION
        { $Thing1 rdf:type crm:E24.Physical_Human-Made_Thing.
         $Thing1 crm:P67F.refers_to $Thing2.
        }}UNION
        { $Thing1 rdf:type crmdig:D1.Digital_Object.
        $Thing1 crmdig:L1F.digitized $Thing2.
        }}UNION
        { $Thing1 rdf:type crmdig:D1.Digital_Object.
        $Thing1 crm:P67F.refers_to $Thing2.
        }}
....
        UNION{
        $Thing1 crm:F1F.is_derivative_of  $tmpThing2.
    $tmpThing2 crmdig:L11B.was_output_of  $DigMachEventX2.
        $DigMachEventX2 crm:P9B.forms_part_of
                $Z1.
        $Z1     crmdig:L1F.digitized $Thing2. }}}
```

Very long SPARQL queries
Very difficult to write and comprehend

Our Framework

E70.Thing -- (F3.is_same_as)[0,n]->E70.Thing
E70.Thing--(F4F.is_compo...Thing
E24.Physical...62F.depicts->
...hing -- P128F.carries
...tion_Object ->
...s_to -> E70.Thing
...Digital_Object ...of)[0,n] ->
D1.Digi...was_output_of -
...e_Event--
...t_of)[0,n] ->
...ation_Process -- L1F.digitized ->
...hing -- (F4F.is_composed_of) [0,n] ->
E70.Thing

Smaller paths
More readable and writable

Schema knowledge

Thing "is about or refers to"

Kazafani Boat

What the end user sees

Example

Implementation

1. ***Paths' language***
2. ***Software***: FR configuration Tool
3. ***Schema***: CIDOC-CRM and CIDOC-CRM digital schema

Paths' Language

- **Domain** :
  - repository schema (i.e. CIDOC-CRM schema)
- **Constants**:
  - classes and predicates from the schema

E70.Thing -- (P46F.is_composed_of)[0,n]-> E70.Thing:
{
E24.Physical Man-Made_Thing--P62F.depicts -> E5.Event
OR
E70.Thing-- P12F.was_present_at -> E5.Event
}

- **Path**
  - sequences of triples (AND conjuctions)
  - OR among paths

Path Expression Example

1. shows features of

2. part-whole

E70.Thing -- (F3.is_same_as)$^{[0,n]}$**-> E70.Thing:**

{E70.Thing--(F4F.is_composed_of) $^{[0,n]}$ -> **E70.Thing:**

{E24.Physical_Man-Made_Thing -- P62F.depicts -> **E70.Thing**

OR

Rules

E24.Physical_Man-Made_Thing -- P128F.carries **-> E73.Information_Object :**

{E73.Information_Object -> P67F.refers_to-> **E70.Thing** }

OR

3. derivatives

D1.Digital_Object --(F1F.is_derivative_of**)$^{[0,n]}$ -> D1.Digital_Object:**

{D1.Digital_Object -- L11B.was_output_of **-> D7.Digital_Machine_Event:**

{D7.Digital_Machine_Event -- (P9B.forms_part_of)$^{[0,n]}$ **->**

D2.Digitization_Process :

2. part-whole

{D2.Digitization_Process --L1F.digitized **-> E70.Thing :**

{E70.Thing -(F4F.is_composed_of) $^{[0,n]}$ -> **E70.Thing**

} } } } } }

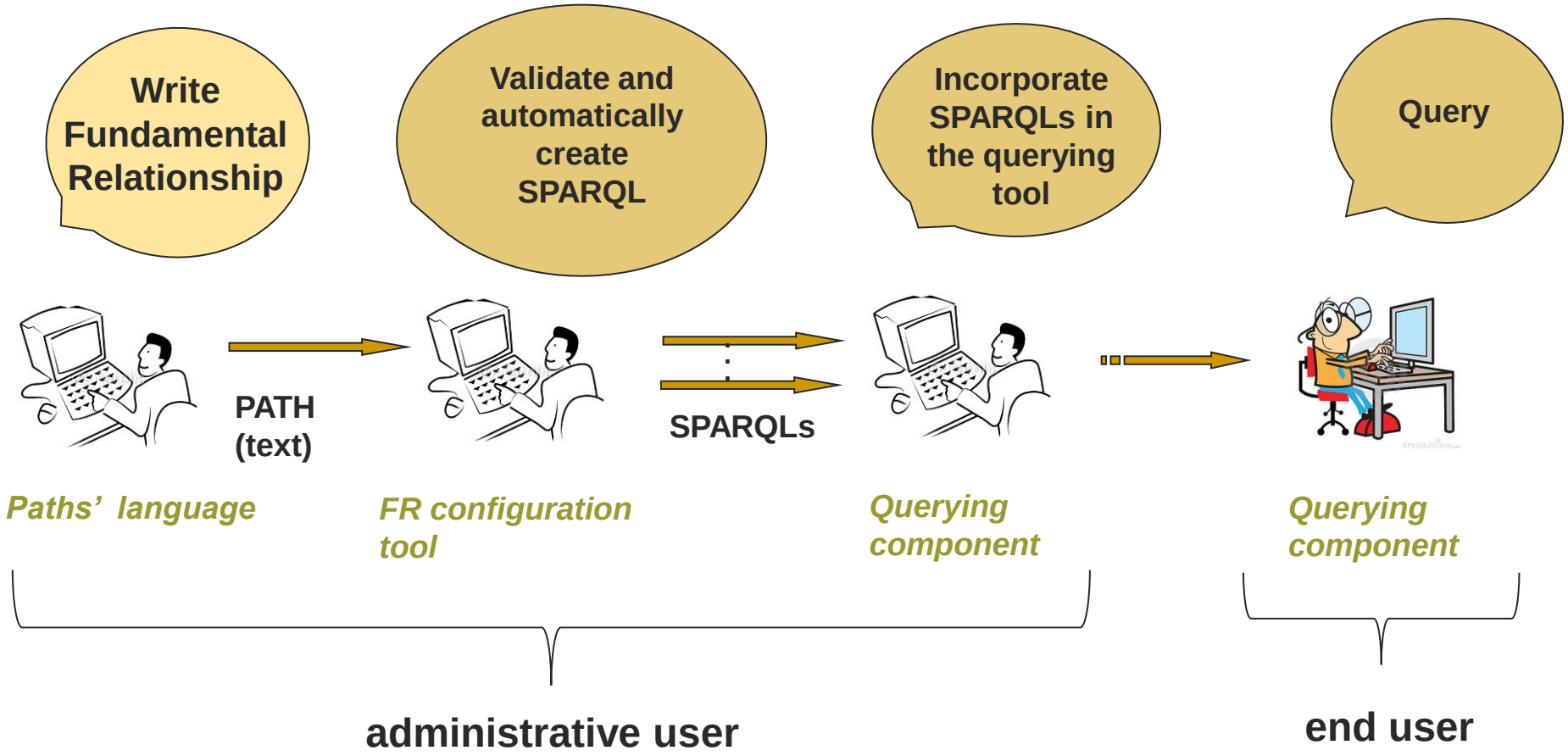
1. ***Path's language***
2. ***Software***: FR configuration Tool
3. ***Schema***: CIDOC-CRM and CIDOC-CRM digital schema

FR configuration tool

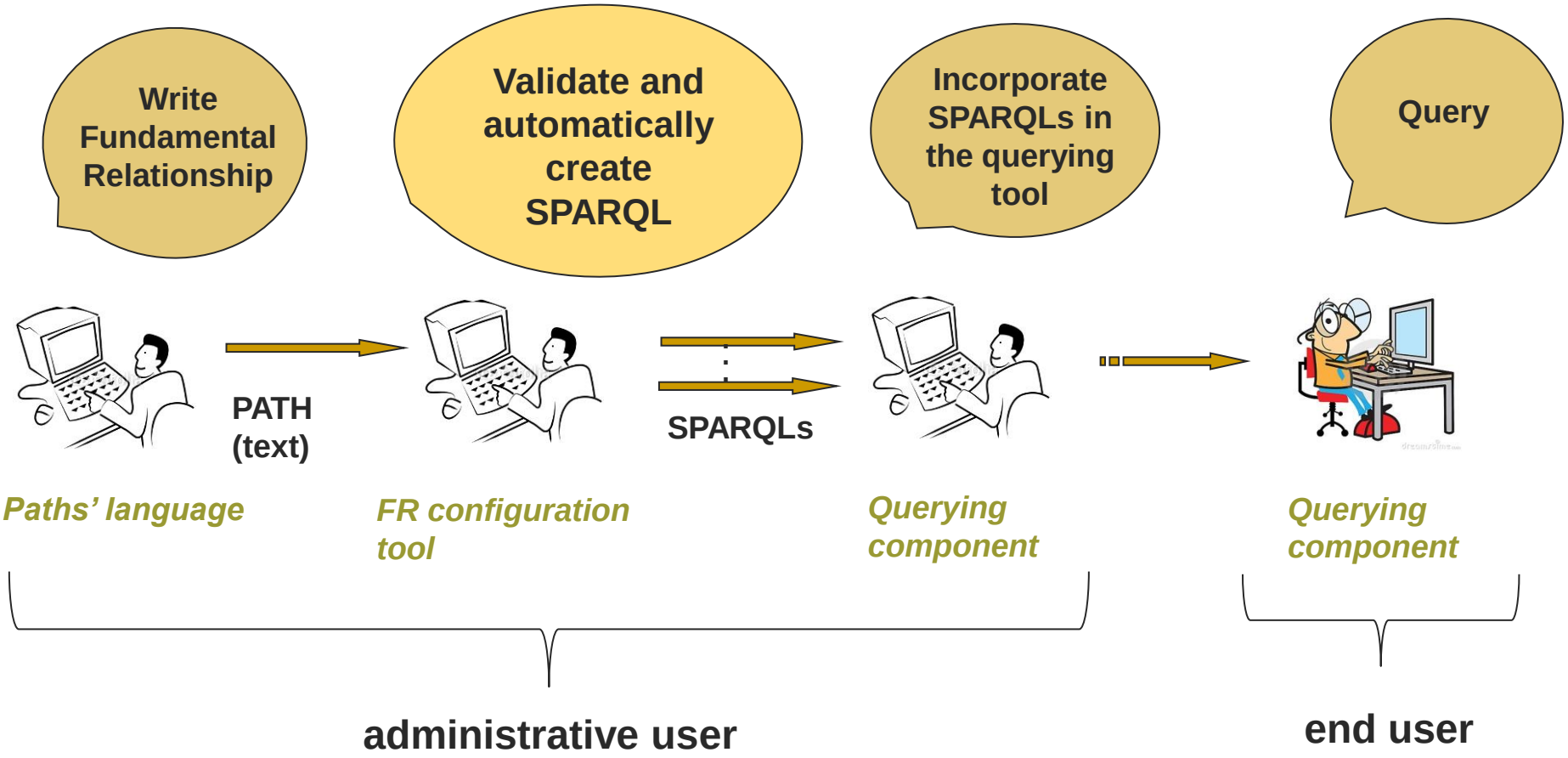
- Validate path
- Check sub-relationship
- Translate Path to SPARQL
- Translate Path to IVB Template *(3D-COFORM feature)*
- Enable-disable multiple instantiation and disjoint cases
- Check schema coverage
- Check for new rules

1. ***Path's language***
2. ***Software***: FR configuration Tool
3. ***Schema***: CIDOC-CRM and CIDOC-CRM digital schema

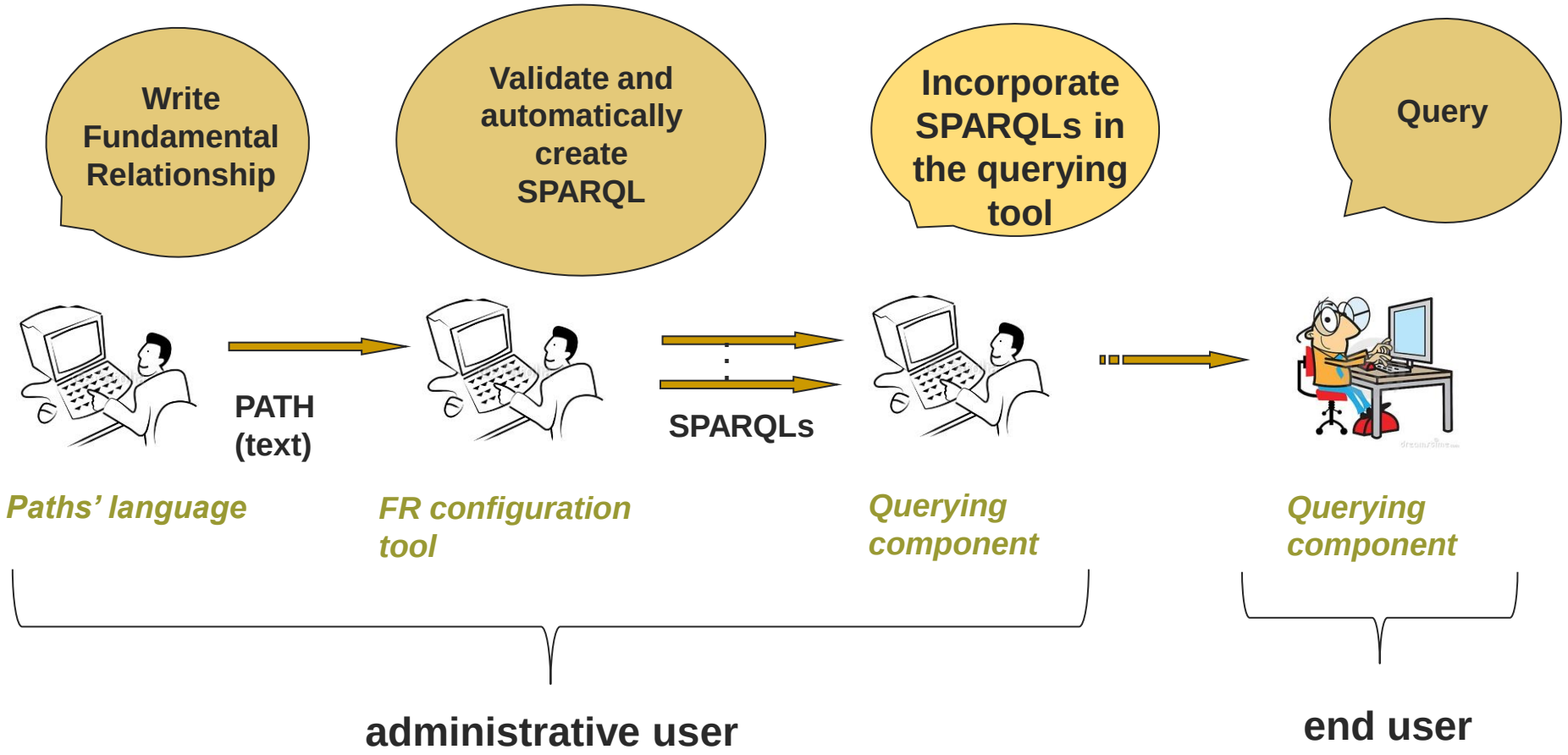
Workflow


Write Fundamental Relationship
Validate and automatically create SPARQL
Incorporate SPARQLs in the querying tool
Query
PATH (text)
SPARQLs
Paths' language
FR configuration tool
Querying component
Querying component
administrative user
end user

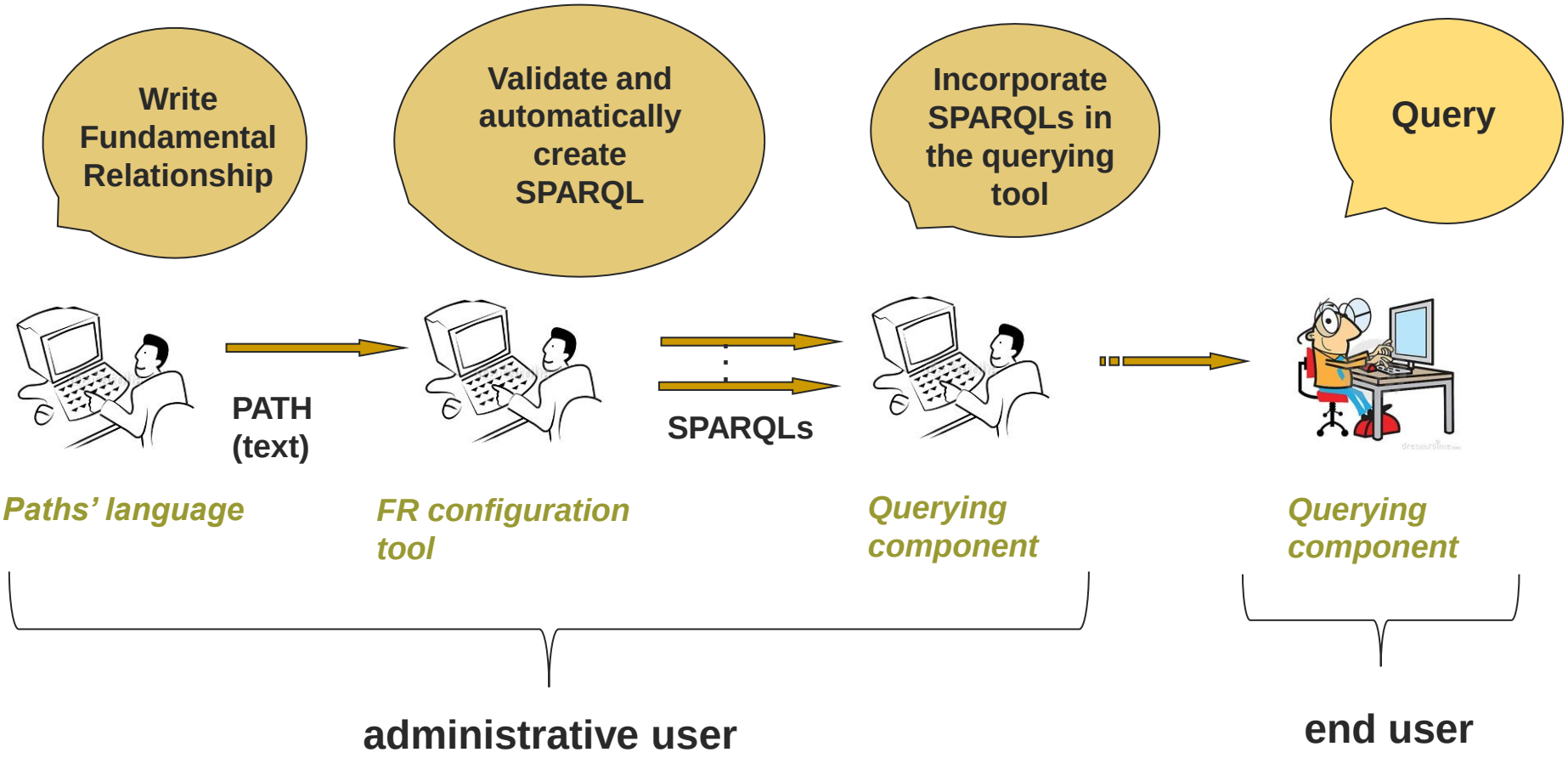
Workflow


Write Fundamental Relationship
Validate and automatically create SPARQL
Incorporate SPARQLs in the querying tool
Query
PATH (text)
SPARQLs
Paths' language
FR configuration tool
Querying component
Querying component
administrative user
end user

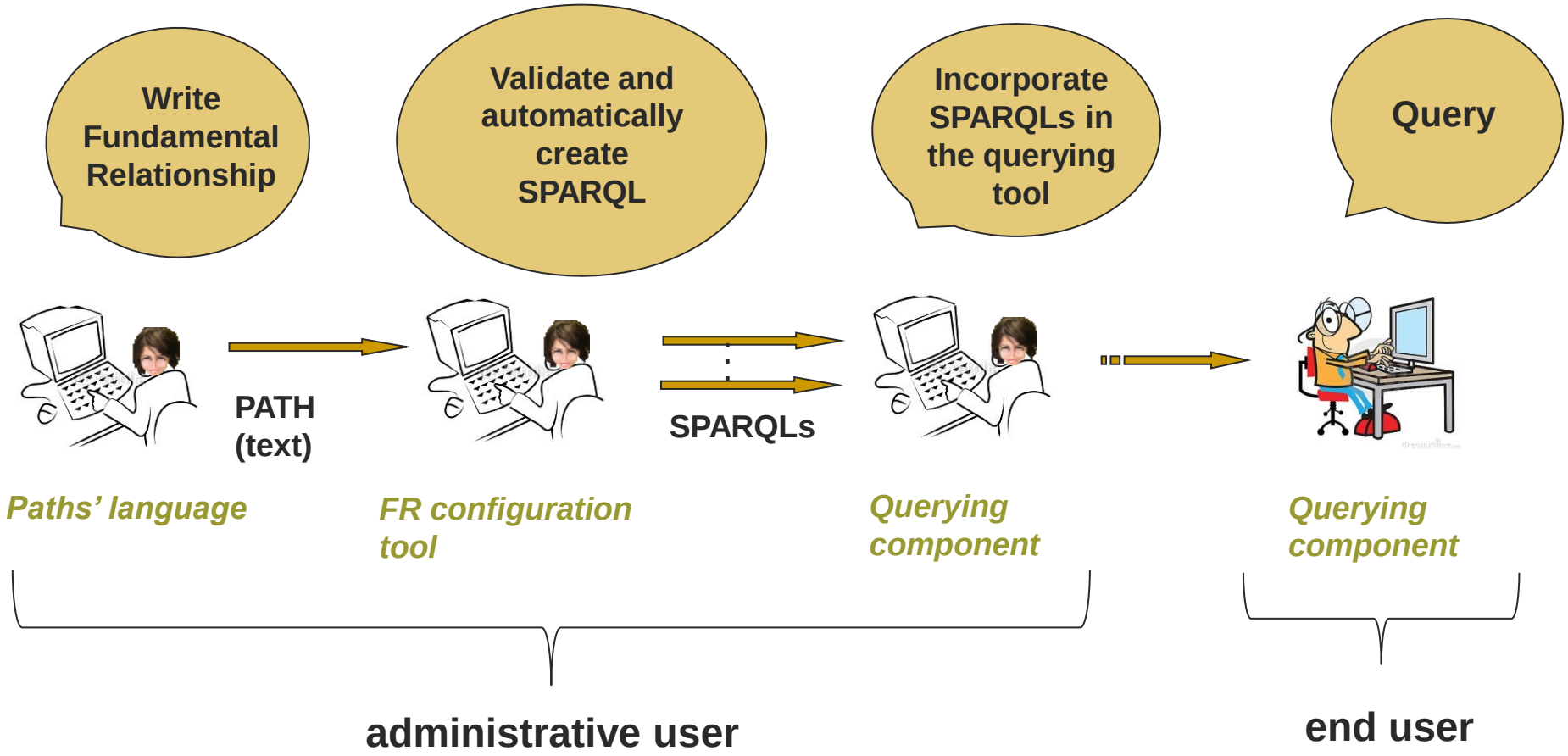
Workflow


Write Fundamental Relationship
Validate and automatically create SPARQL
Incorporate SPARQLs in the querying tool
Query
PATH (text)
SPARQLs
Paths' language
FR configuration tool
Querying component
Querying component
administrative user
end user

Workflow


Write Fundamental Relationship
Validate and automatically create SPARQL
Incorporate SPARQLs in the querying tool
Query
PATH (text)
SPARQLs
Paths' language
FR configuration tool
Querying component
Querying component
administrative user
end user

Workflow

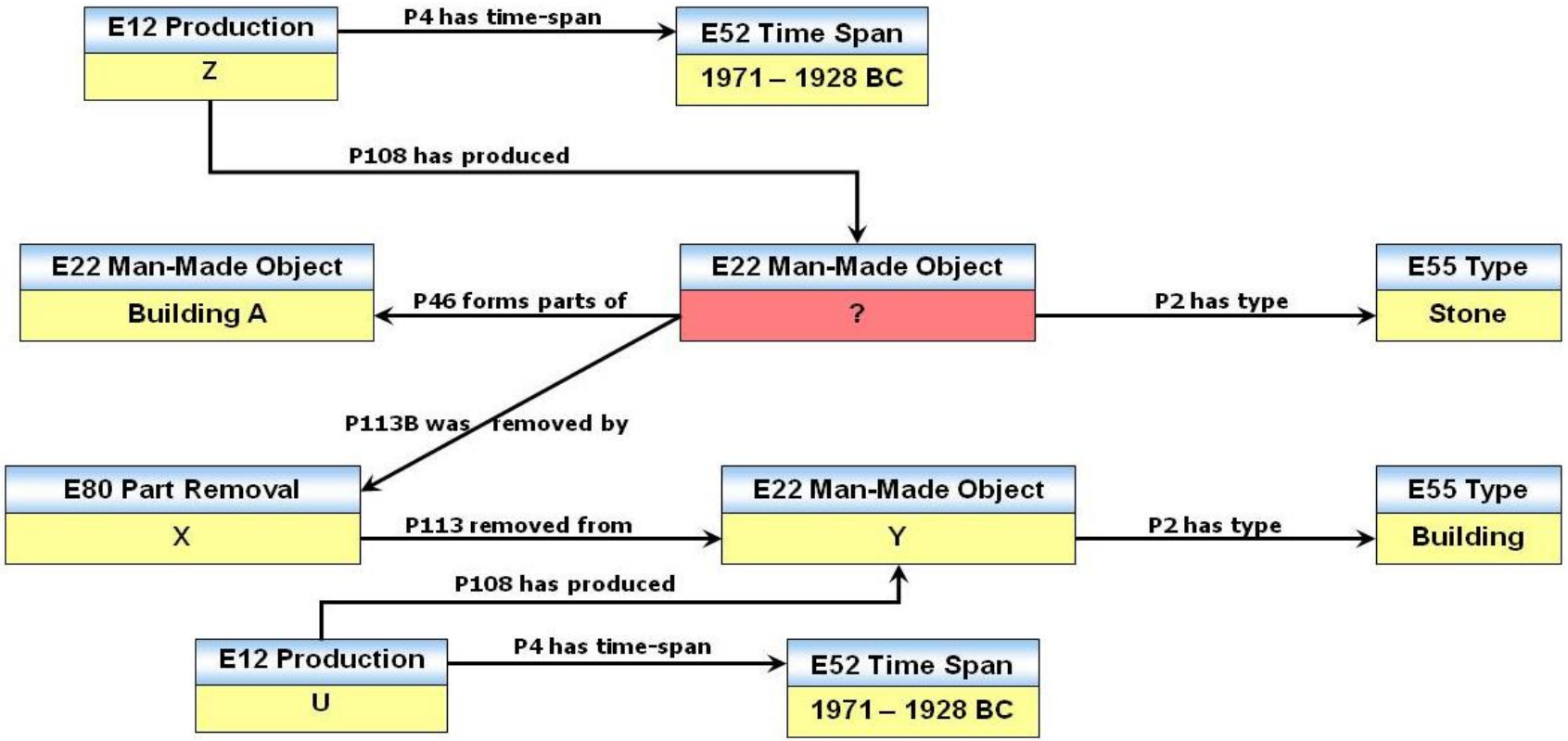

Write Fundamental Relationship
Validate and automatically create SPARQL
Incorporate SPARQLs in the querying tool
Query
PATH (text)
SPARQLs
Paths' language
FR configuration tool
Querying component
Querying component
administrative user
end user

Validation

- ***Used by***
  - 3D-COFORM Project
  - Research Space Project
- ***Real*** archaeologist's ***queries***

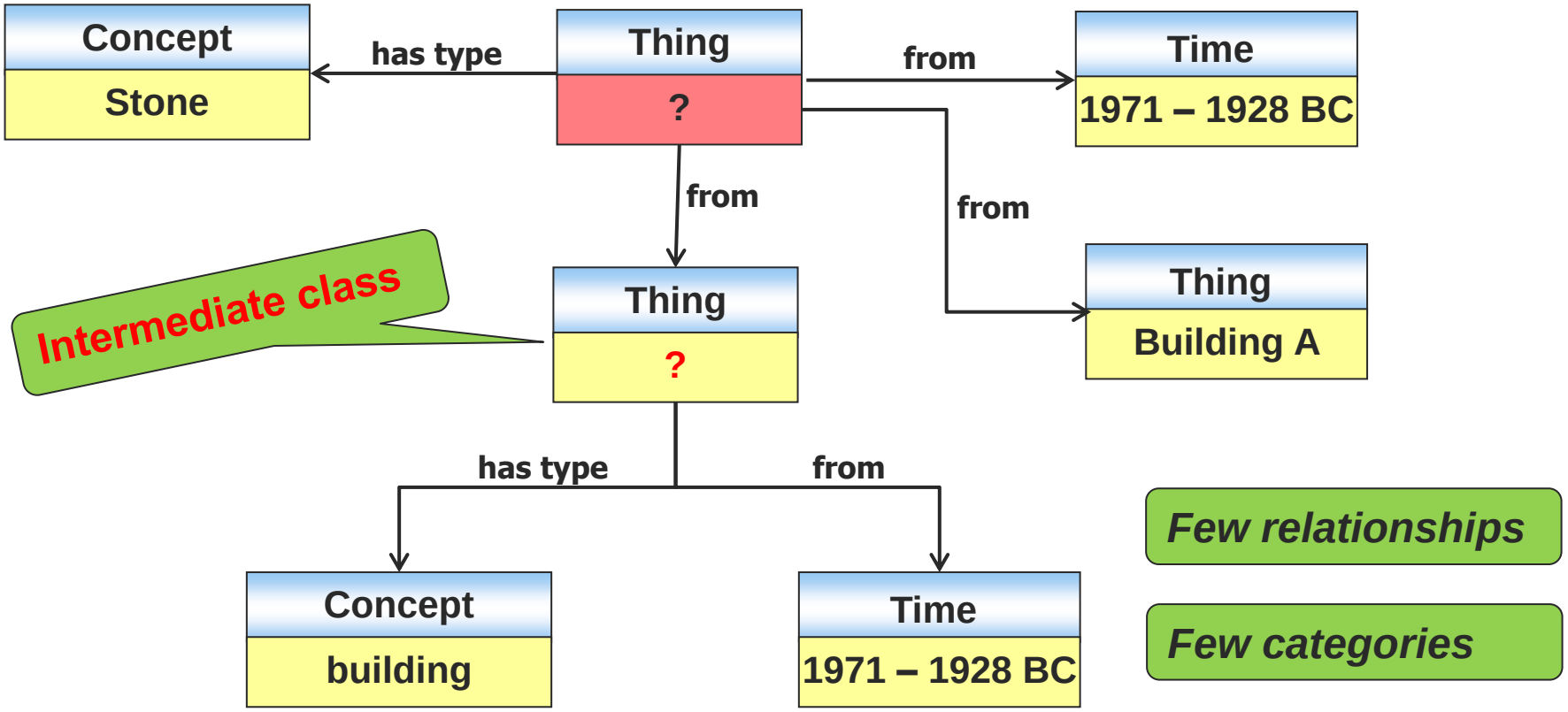
Scientific query example – CIDOC-CRM metadata (1/2)

Find all the stones from Building A that have a previous use and are dated between 1971 - 1928 BC.



Scientific query example – FCs & FRs model (2/2)

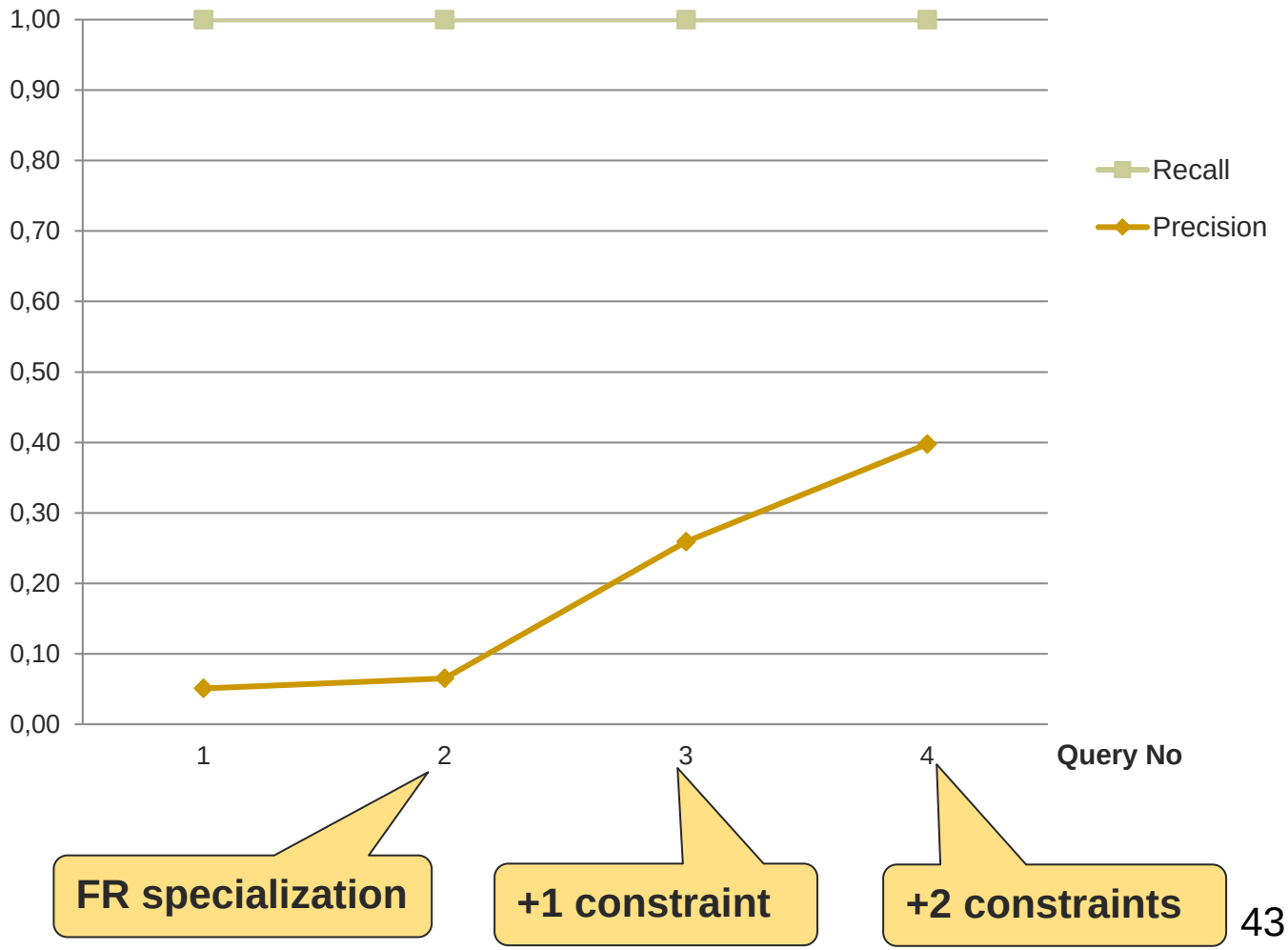
Find all the stones from Building A that have a previous use and are dated between 1971 - 1928 BC.



- ***Used by***:
  - 3D-COFORM Project
  - Research Space Project
- ***Real*** archaeologist's ***queries***
- ***Test queries:*** high recall, tunable precision

Test queries results

Query No	Query
1	Thing **from** Brighton
2	Thing **is located in** Brighton
3	Thing **from** Brighton and **has type** sculpture (visual work)
4	Thing **from** Brighton and **has type** sculpture (visual work) and **is made of** stone (rock)



Contributions

Contributions

• New data model	→	simplified querying
	→	customizable
• Simplified method	→	time saving
	→	easy and efficient
• Paths' language	→	SPARQL avoidance
• Associative querying, rules	→	high recall
• FR specializations	→	precision improvement
• Network's rich schema	→	network integration capability

Live Demo

Thank you for your attention!