ANCHOR™

A Philosophy of Modeling

Lars Rönnbäck

up to change

No two objects can have all the same properties.

Gottfried Wilhelm Leibniz

Everything flows.

Heraclitus

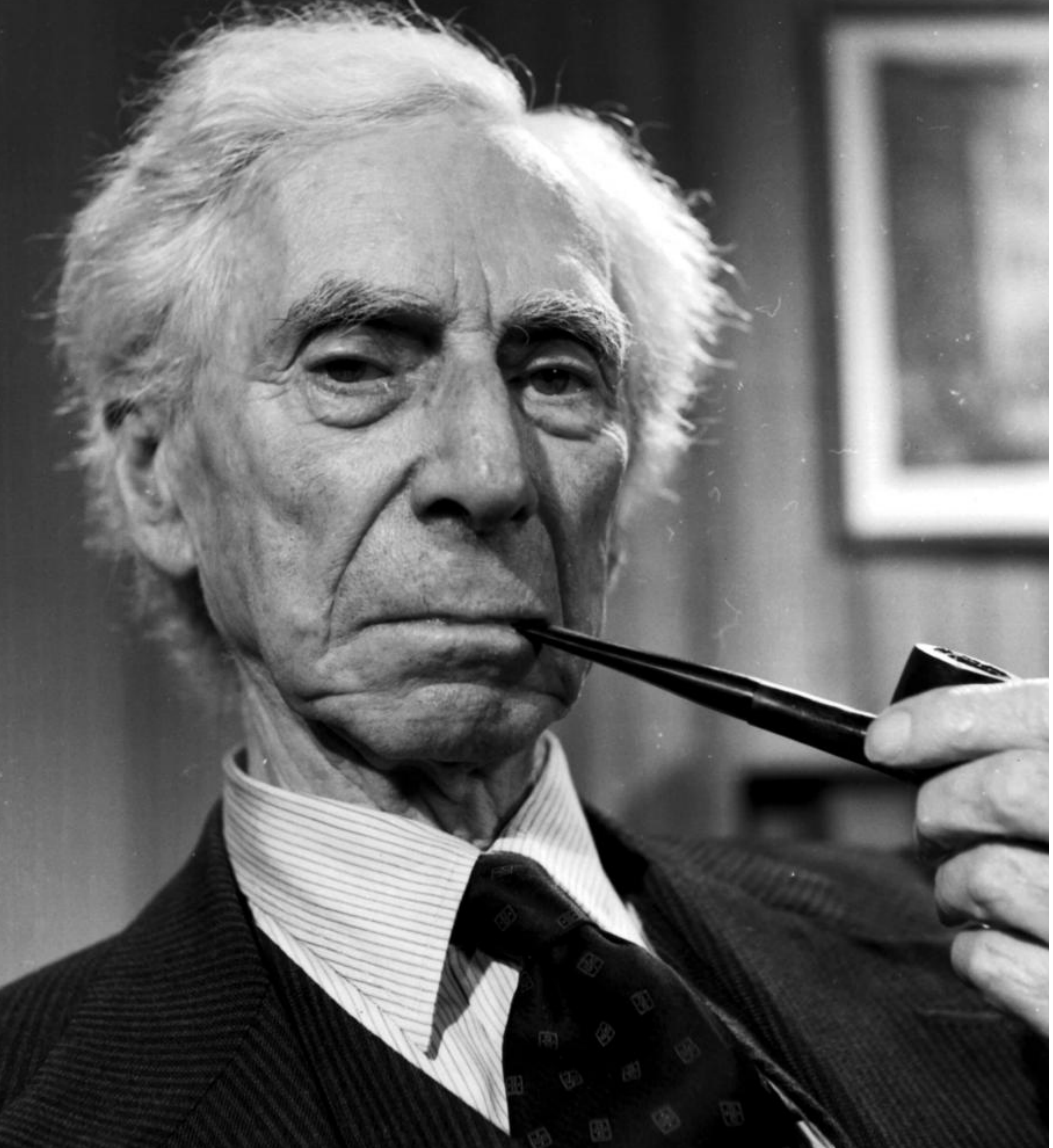
All things are momentary.

Vasubandhu

The Buddhist Doctrine of Momentariness

The reality we can put into words is never reality itself.

Werner Heisenberg



Some things are more nearly certain than others.

Bertrand Russell

Our subjectivity is so completely our own.

Spike Jonze

One should always aim at being interesting, rather than exact.

Voltaire

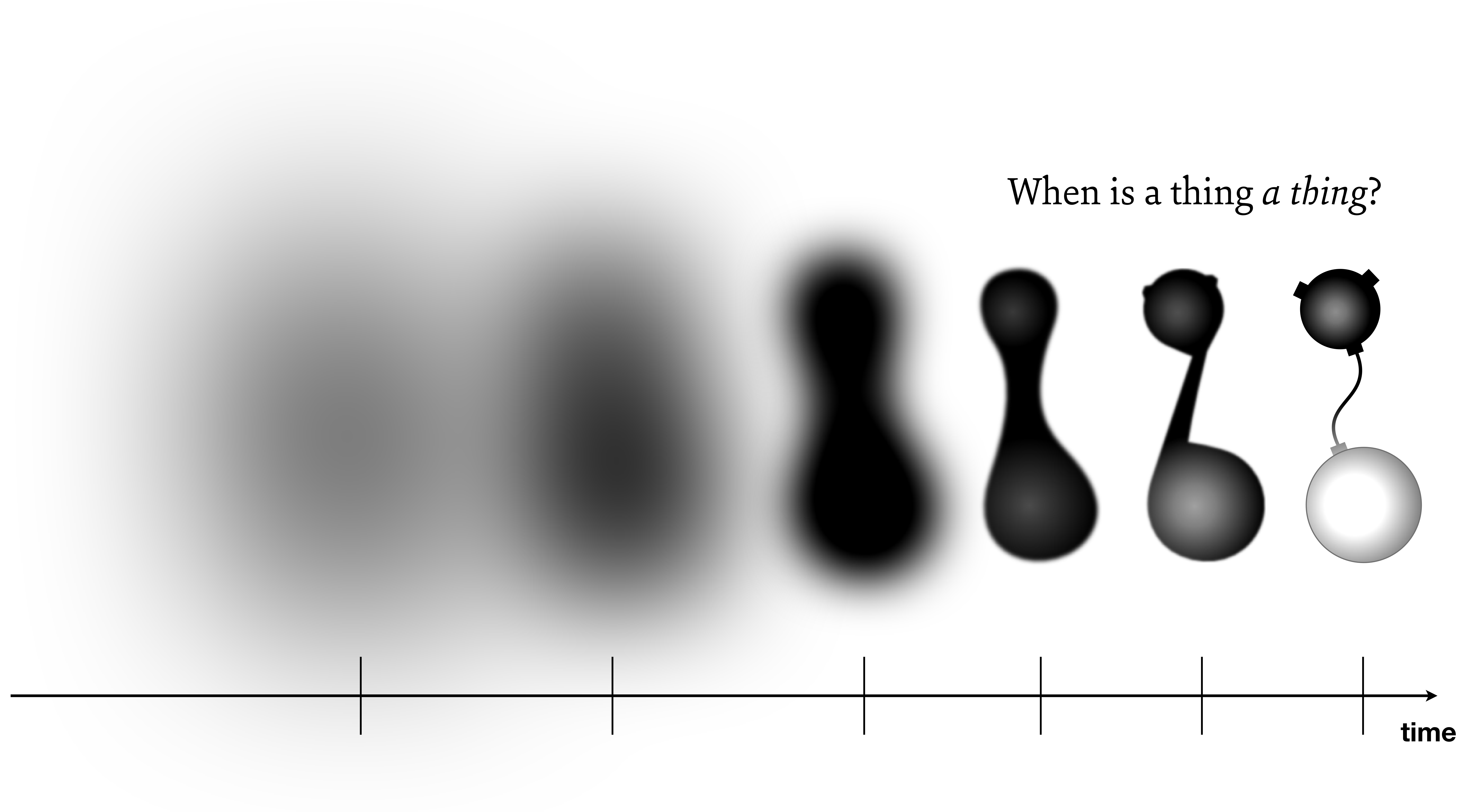
If it quacks like a duck…

James Whitcomb Riley

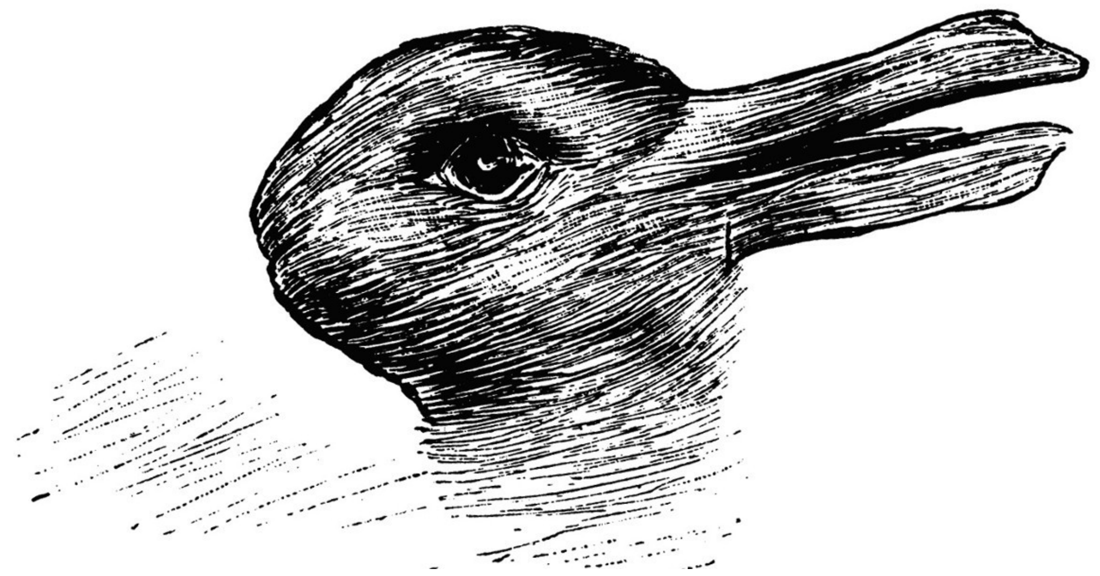
Same same, but different.

Thai salesperson

When is a thing *a thing*?


When is a thing *a thing*?
time

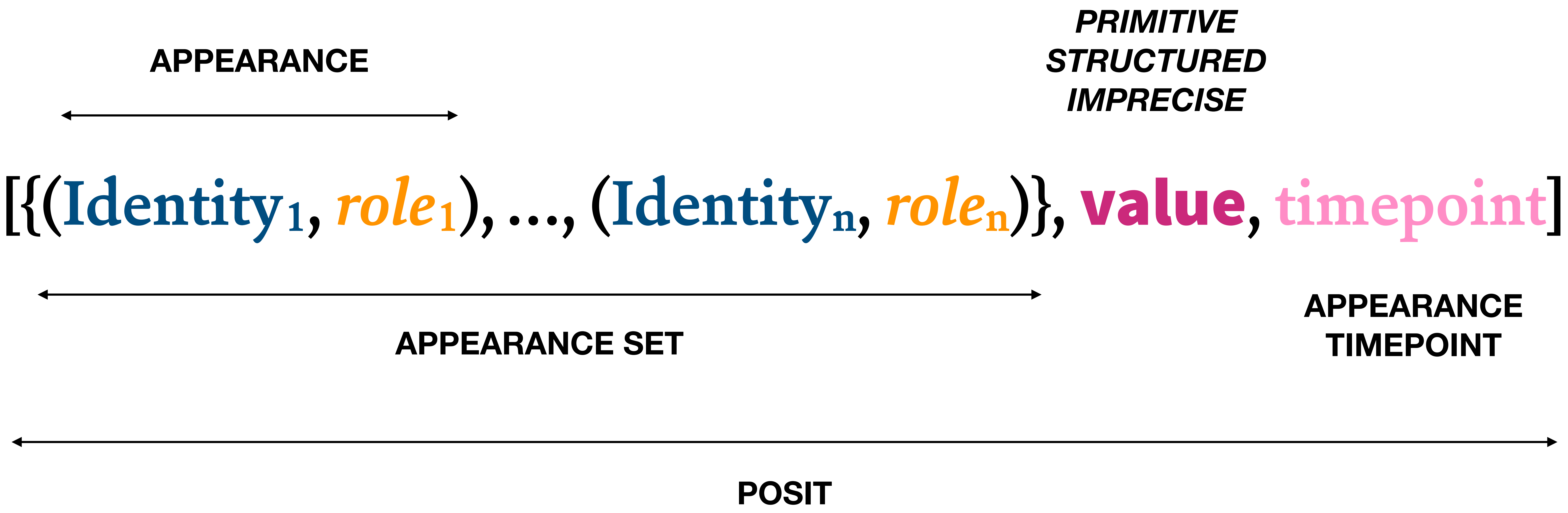
- Things are important
- Their properties are important
- Their classification is ephemeral
- Values are always imprecise
- Changes can be captured through sampling
- Only identities are immutable
- We record statements about reality, and we do not record reality itself
- The certainty of true reality cannot be captured
- Opinions may differ and may be revised
- Keep the number of things that have to be agreed upon small



Transitional Modeling

Lars Rönnbäck

[{(,), ..., (,)}, ,]


APPEARANCE
PRIMITIVE
STRUCTURED
IMPRECISE
[{(Identity1, role1), …, (Identityn, rolen)}, value, timepoint]
APPEARANCE SET
APPEARANCE
TIMEPOINT
POSIT

The least we must agree upon are appearances

APPEARANCE

PRIMITIVE
STRUCTURED
IMPRECISE

$$[\{(\text{Identity}_1, \textit{role}_1), \ldots, (\text{Identity}_n, \textit{role}_n)\}, \textbf{value}, \text{timepoint}]$$

APPEARANCE SET

APPEARANCE
TIMEPOINT

POSIT

…which lets us disagree about everything else

[{(42, *husband*), (43, *wife*)}, **married**, 2004-06-19]

[{(42, *name*)}, **Lars Samuelsson**, 1972-08-20]
[{(42, *hair color*)}, **gray**, 2022-02-22]

[{(42, *name*)}, **Lars Samuelsson**, 1972-08-20]
[{(42, *name*)}, **Lars Rönnbäck**, 2005-03-30]
[{(42, *hair color*)}, **brown**, 1973-02-13]
[{(42, *hair color*)}, **gray**, 2022-02-22]

VALUES MAY CHANGE OVER TIME *(APPEARANCE TIMELINE)*

DATA [{(42, *name*)}, **Lars Rönnbäck**, 2005-03-30]

DATA [{(42, *hair color*)}, **gray**, 2022-02-22]

DATA [{(4711, *name*)}, **Person**, 2019-10-20]

PERIDATA [{(42, *thing*), (4711, *class*)}, **active**, 1972-08-20]

RESERVED ROLE

RESERVED ROLE

POSIT
IDENTITY

555 ⟵ [{(42, *name*)}, **Lars Rönnbäck**, 1972-08-20]

556 ⟵ [{(42, *hair color*)}, **gray**, 2022-02-22]

557 ⟵ [{(42, *husband*), (43, *wife*)}, **married**, 2004-06-19]

POSIT IDENTITY

555 ⟵ [{(42, *name*)}, **Lars Rönnbäck**, 1972-08-20]

556 ⟵ [{(42, *hair color*)}, **gray**, 2022-02-22]

557 ⟵ [{(42, *husband*), (43, *wife*)}, **married**, 2004-06-19]

CERTAINTY **ASSERTION TIMEPOINT**

[{(555, *posit*), (42, *ascertains*)}, **90%**, 2019-10-20]

ASSERTION

555 ⟵ [{(42, *name*)}, **Lars Rönnbäck**, 1972-08-20]

556 ⟵ [{(42, *hair color*)}, **gray**, 2022-02-22]

557 ⟵ [{(42, *husband*), (43, *wife*)}, **married**, 2004-06-19]

[{(555, *posit*), (42, *ascertains*)}, **90%**, 2019-10-20]
[{(556, *posit*), (42, *ascertains*)}, **–80%**, 2019-10-20]
[{(557, *posit*), (42, *ascertains*)}, **0%**, 2019-10-21]

COMPLETE UNCERTAINTY

557 ⟵ [{(42, *husband*), (43, *wife*)}, **married**, 2004-06-19]

558 ⟵ [{(42, *husband*), (43, *wife*)}, **married**, 2005-06-19]

CERTAINTY MAY CHANGE OVER TIME *(ASSERTION TIMELINE)*

[{(557, *posit*), (42, *ascertains*)}, **90%**, 2019-10-20]

[{(557, *posit*), (43, *ascertains*)}, **-100%**, 2019-10-20]

[{(557, *posit*), (42, *ascertains*)}, **0%**, 2019-10-21]

[{(558, *posit*), (42, *ascertains*)}, **100%**, 2019-10-21]

[{(558, *posit*), (43, *ascertains*)}, **100%**, 2019-10-21]

$555 \longleftarrow$ [{(42, *name*)}, **Lars Rönnbäck**, 1972-08-20]

$556 \longleftarrow$ [{(42, *hair color*)}, **gray**, 2022-02-22]

$557 \longleftarrow$ [{(42, *husband*), (43, *wife*)}, **married**, 2004-06-19]

METADATA

[{(555, *posit*), (2001, *job*)}, **success**, 2019-10-20]

[{(2001, *started at*)}, **2019-10-20 10:15**, 2019-10-20]

[{(2001, *user*)}, **SQLAgent**, 2019-10-20]

- From here we could go on and also define:
  - Constraints (and cardinality) as posits
  - Identifiers (natural keys) as posits
  - Structures (such as ensembles) as posits
  - Inheritance, Trustworthiness, Consensus, Contradictions, and so on…

Schemaful Databases

How to model without classes

Lars Rönnbäck

beard
color

name

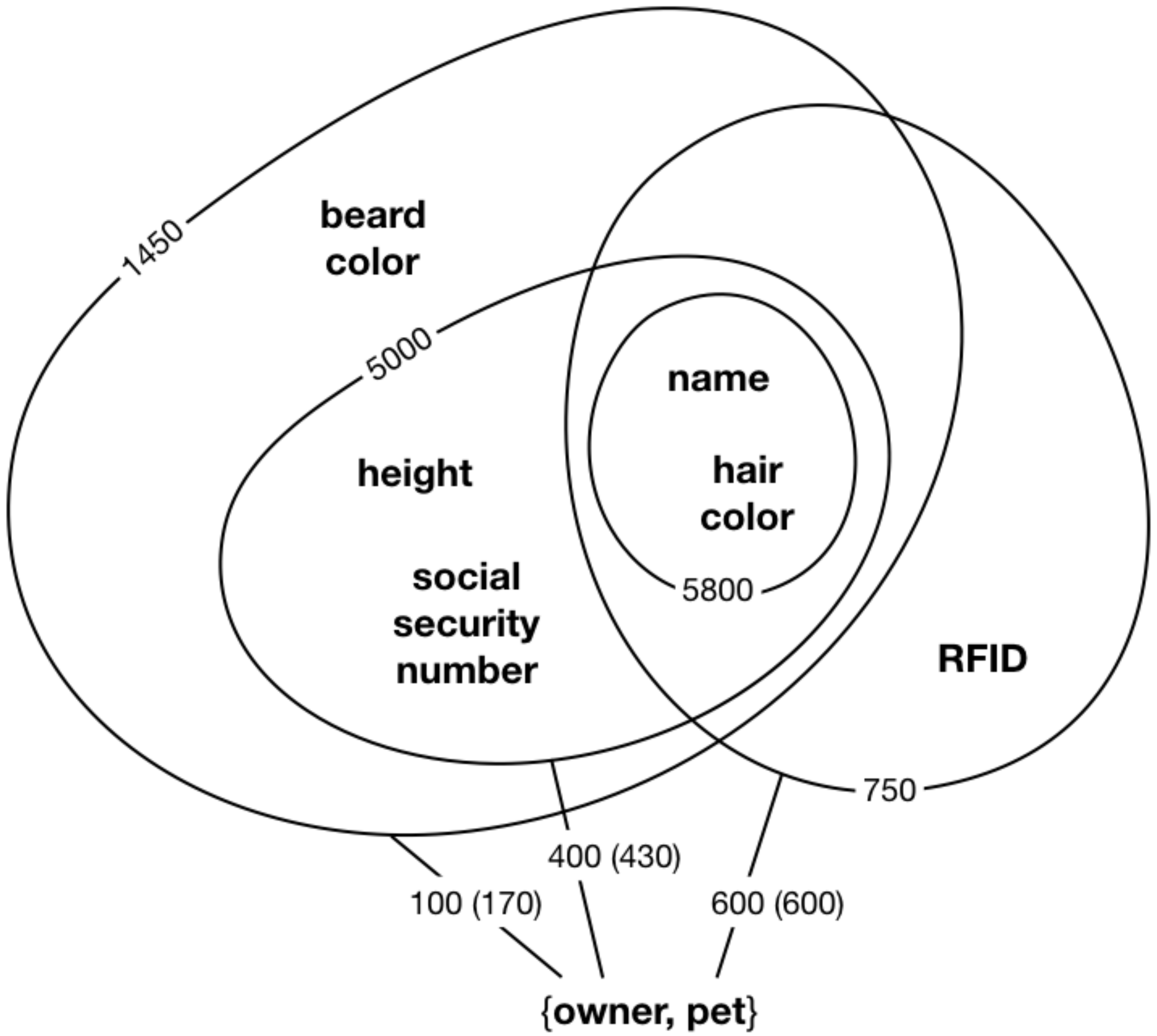
height

hair
color

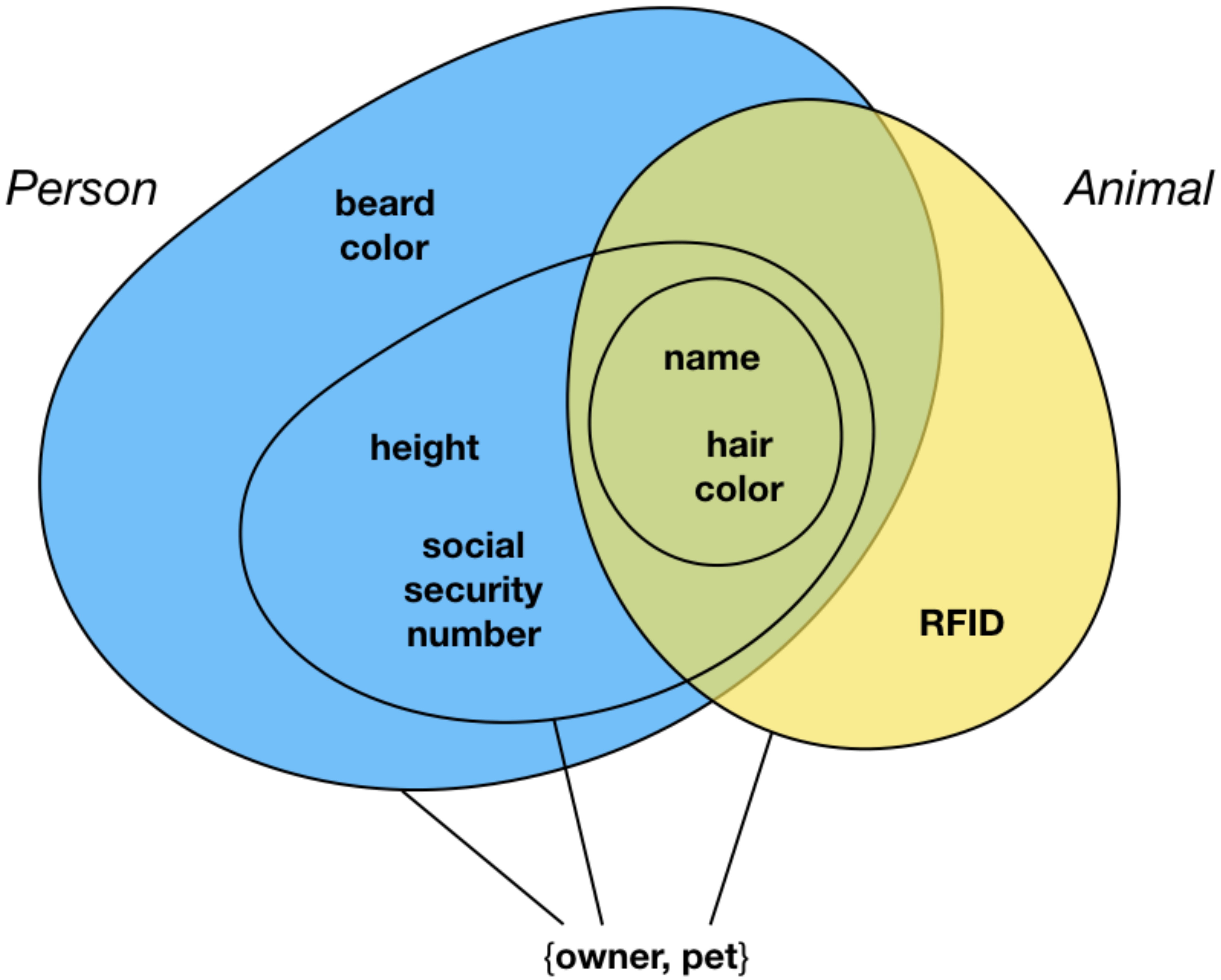
social
security
number

RFID

owner, pet



1450
beard color
5000
height
social security number
name
hair color
5800
RFID
750
400 (430)
100 (170)
600 (600)
{owner, pet}

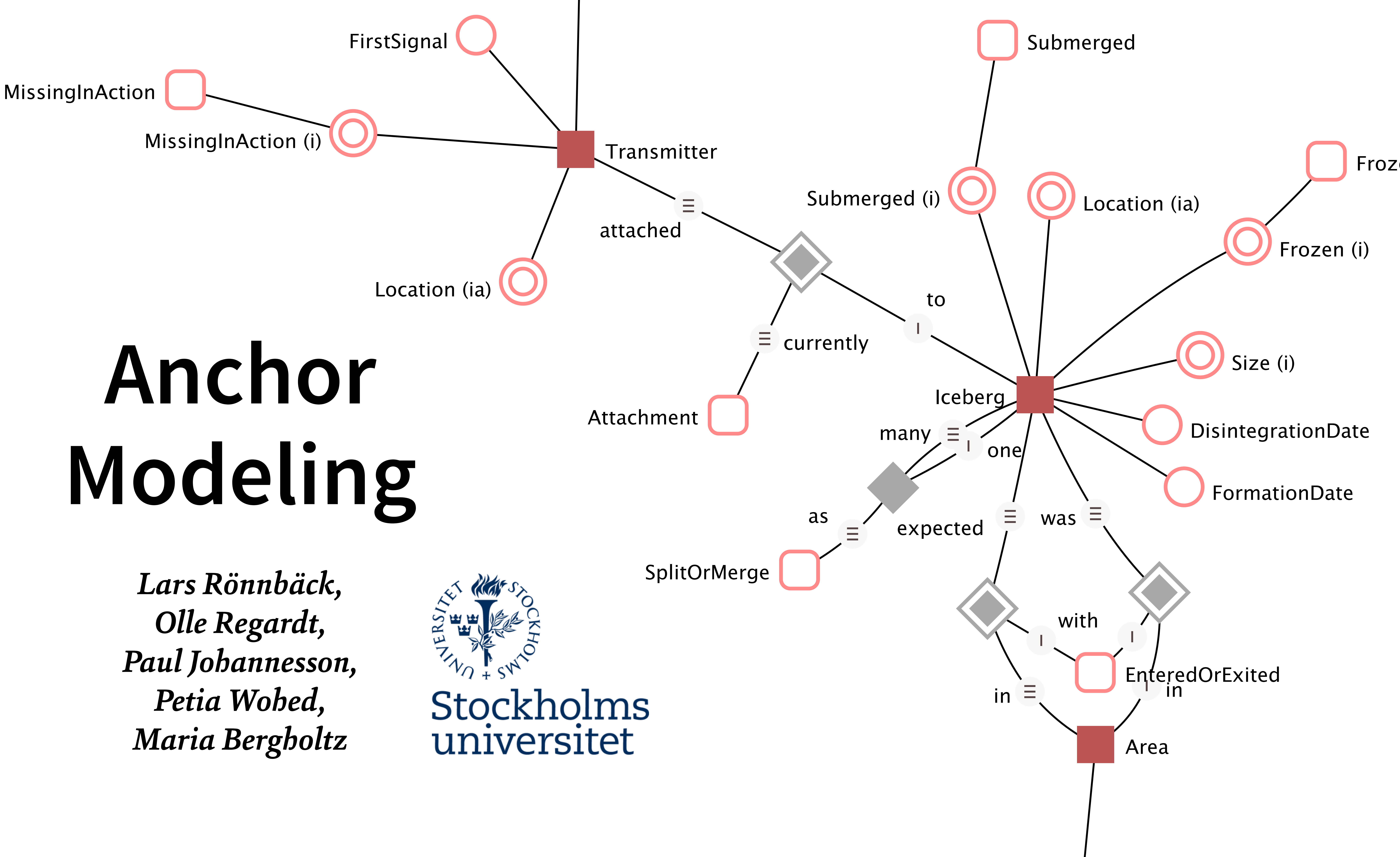


Person
Animal
beard color
height
social security number
name
hair color
RFID
{owner, pet}

Anchor Modeling

Lars Rönnbäck,
Olle Regardt,
Paul Johannesson,
Petia Wohed,
Maria Bergholtz

Stockholms universitet



DATA [{(42, *name*)}, **Lars Rönnbäck**, 1972-08-20]

DATA [{(42, *hair color*)}, **gray**, 2022-02-22]

DATA [{(4711, *name*)}, **Person**, 2019-10-20]

PERIDATA [{(42, *thing*), (4711, *class*)}, **active**, 1972-08-20]

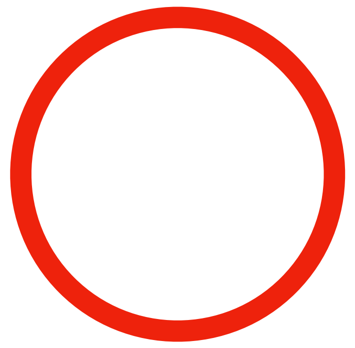
An *anchor* named PE_Person **holds the identities for things of the Person class.**

DATA [{(42, *name*)}, **Lars Rönnbäck**, 1972-08-20]

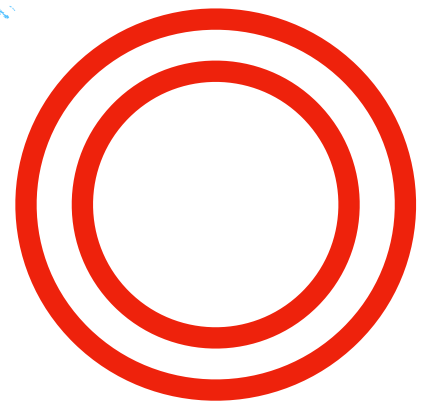
DATA [{(42, *hair color*)}, **gray**, 2022-02-22]

DATA [{(4711, *name*)}, **Person**, 2019-10-20]

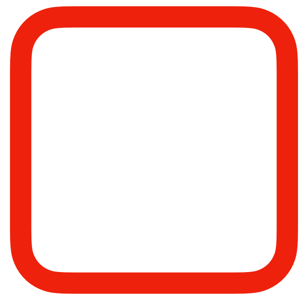
PERIDATA [{(42, *thing*), (4711, *class*)}, **active**, 1972-08-20]



A *static attribute* named PE_NAM_Person_Name holds a reference to an identity and a primitive value.



A *historized attribute* named PE_HAC_Person_HairColor holds a reference to an identity, a primitive value, and a time point since when it came into effect.



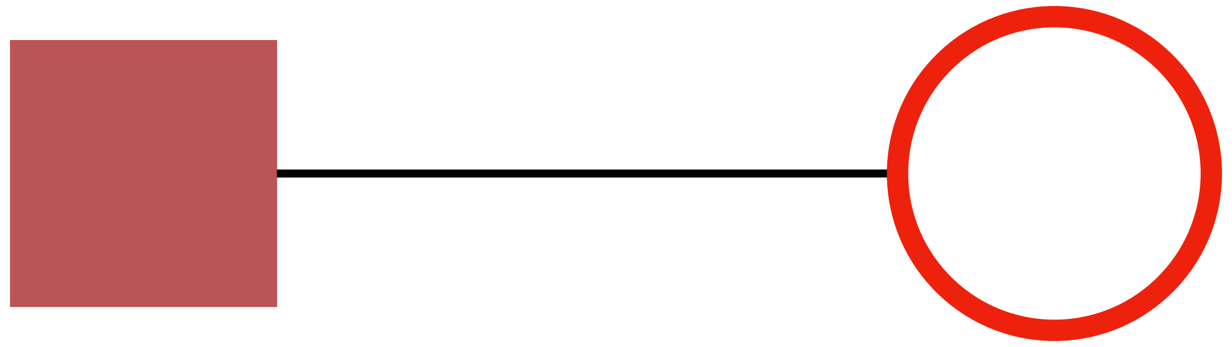
A *knot* named COL_Color holds an enumeration of primitive values along with their own identites.

DATA [{(42, *name*)}, **Lars Rönnbäck**, 1972-08-20]

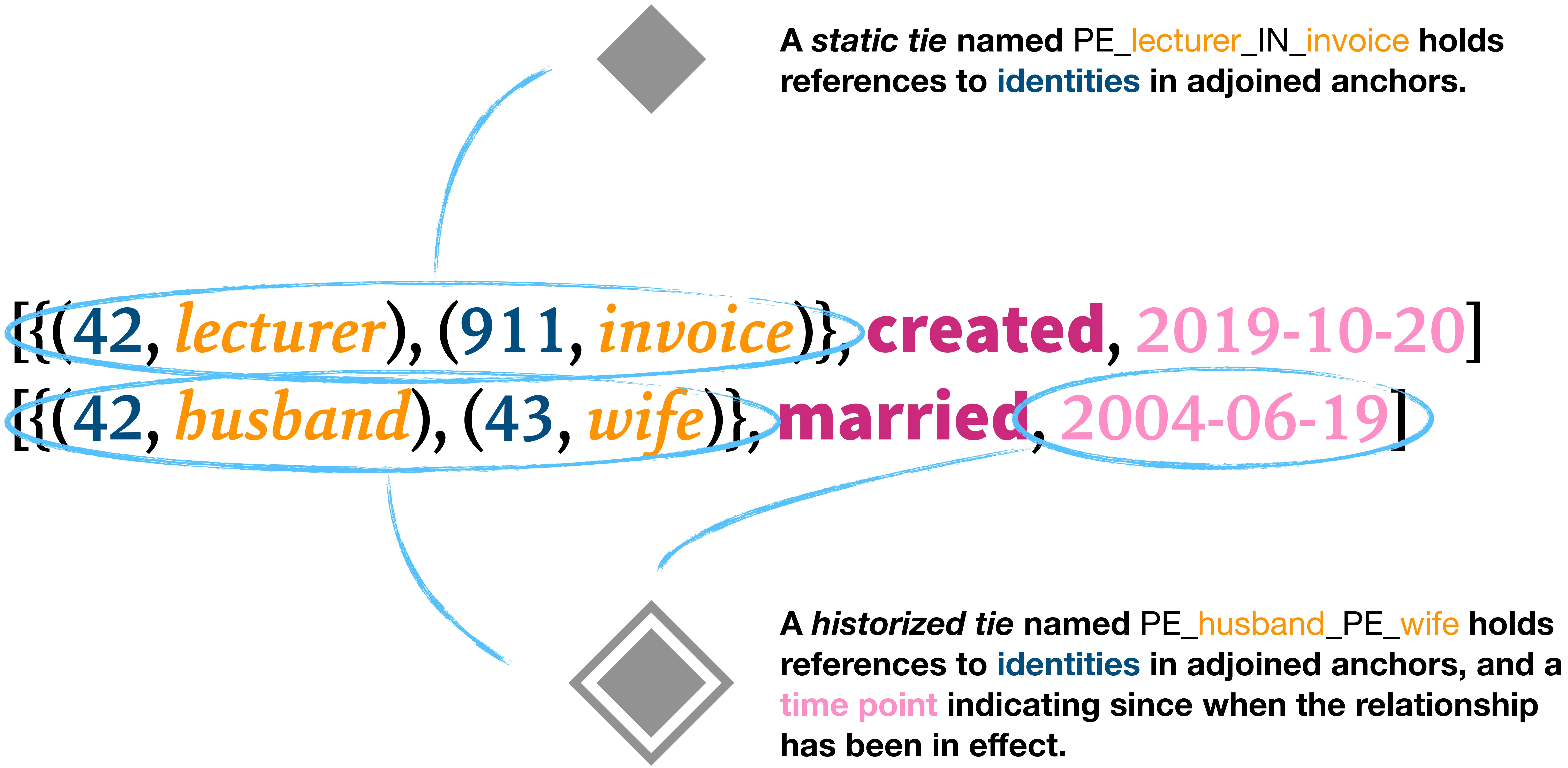
DATA [{(42, *hair color*)}, **gray**, 2022-02-22]

DATA [{(4711, *name*)}, **Person**, 2019-10-20]

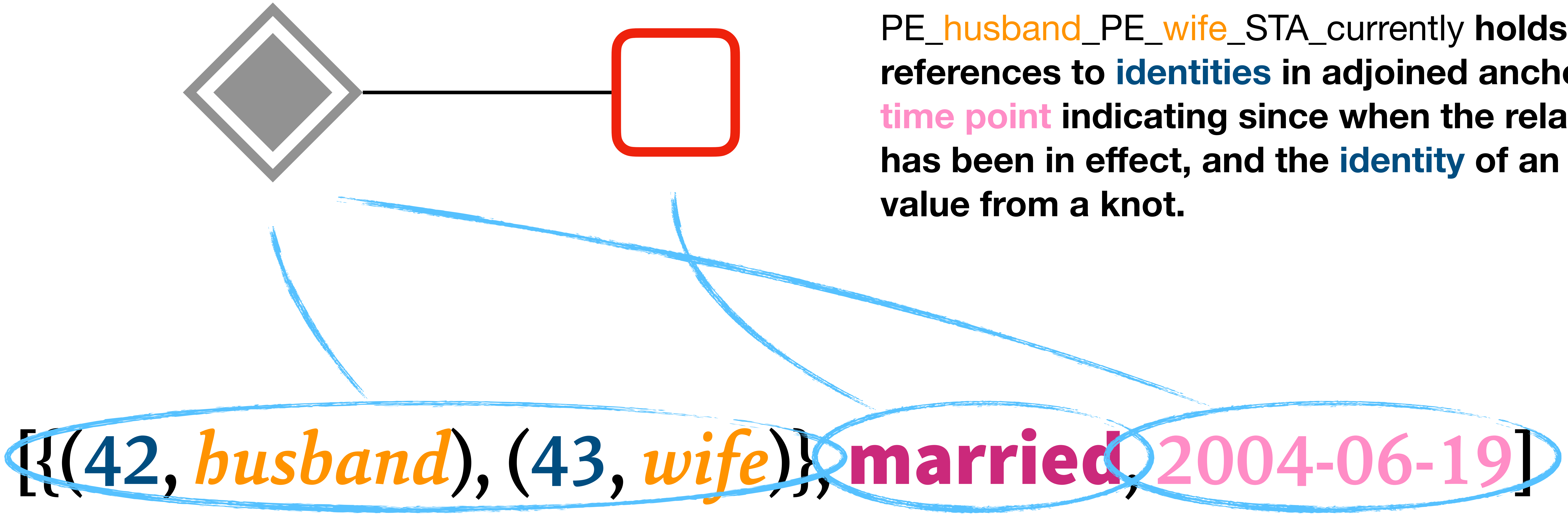
PERIDATA [{(42, *thing*), (4711, *class*)}, **active**, 1972-08-20]



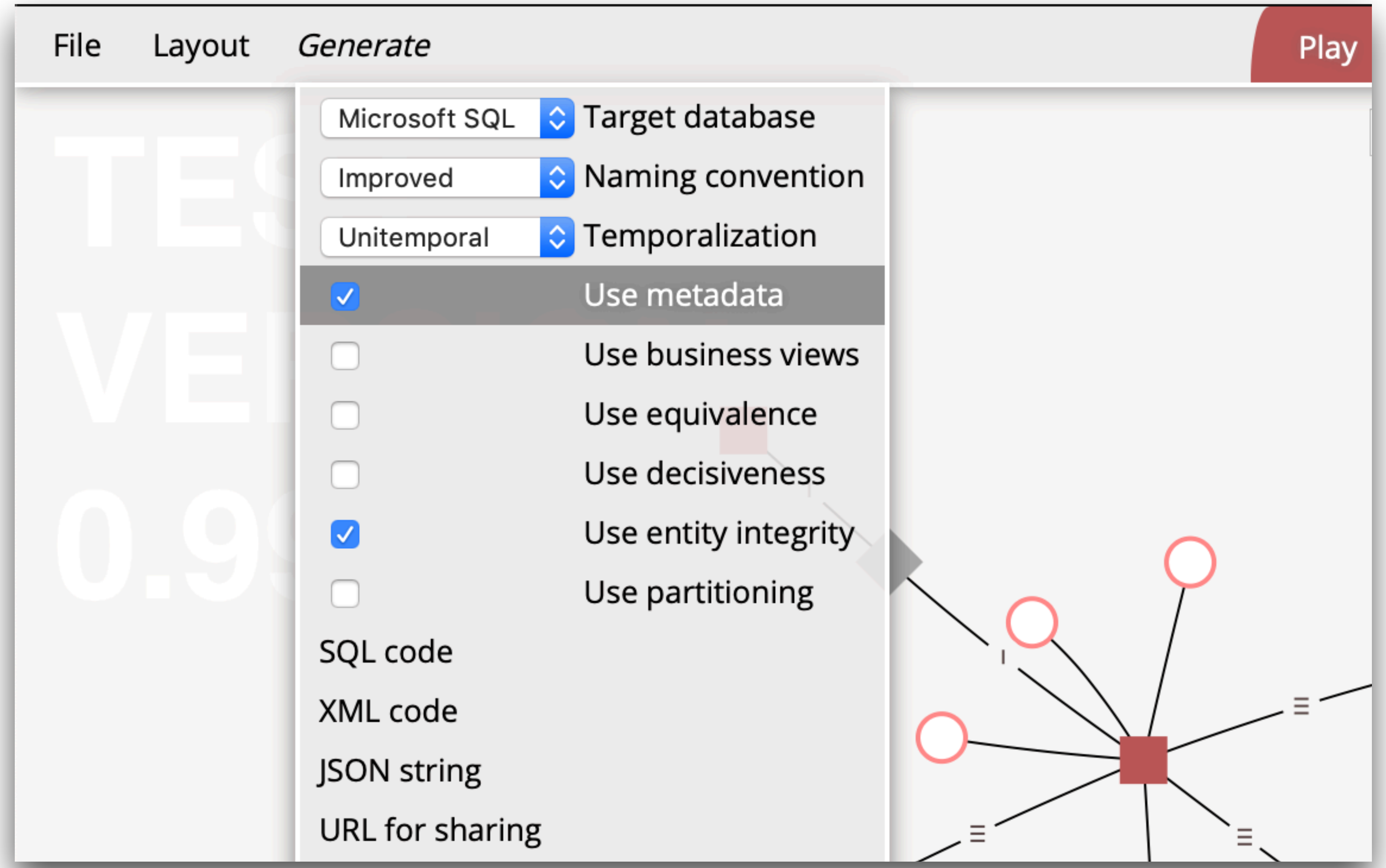
Knots are equivalent to an anchor with a single static attribute.



A *static tie* named PE_lecturer_IN_invoice holds references to identities in adjoined anchors.
[{(42, *lecturer*), (911, *invoice*)}, **created**, 2019-10-20]
[{(42, *husband*), (43, *wife*)}, **married**, 2004-06-19]
A *historized tie* named PE_husband_PE_wife holds references to identities in adjoined anchors, and a time point indicating since when the relationship has been in effect.



A *knotted historized tie* named
PE_husband_PE_wife_STA_currently **holds references to identities in adjoined anchors, a time point indicating since when the relationship has been in effect, and the identity of an enumerated value from a knot.**



A ***knotted historized tie*** **named** PE_husband_PE_wife_STA_currently **holds references to identities in adjoined anchors, a time point indicating since when the relationship has been in effect, and the identity of an enumerated value from a knot.**

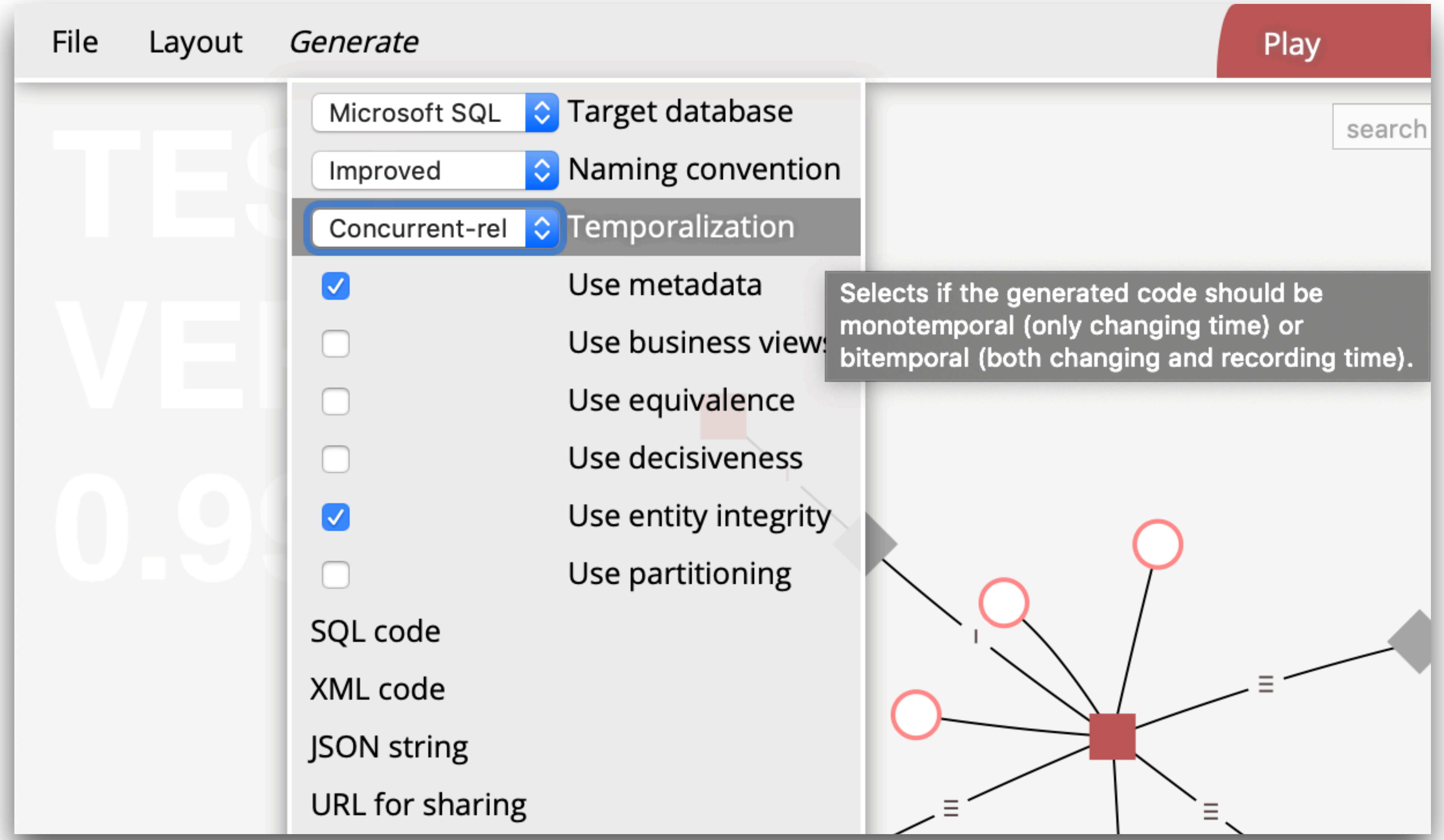
With metadata it also holds a reference to a metadata identity.

555 ⟵ [{(42, *name*)}, **Lars Rönnbäck**, 1972-08-20]

[{(555, *posit*), (2001, *job*)}, **success**, 2019-10-20]

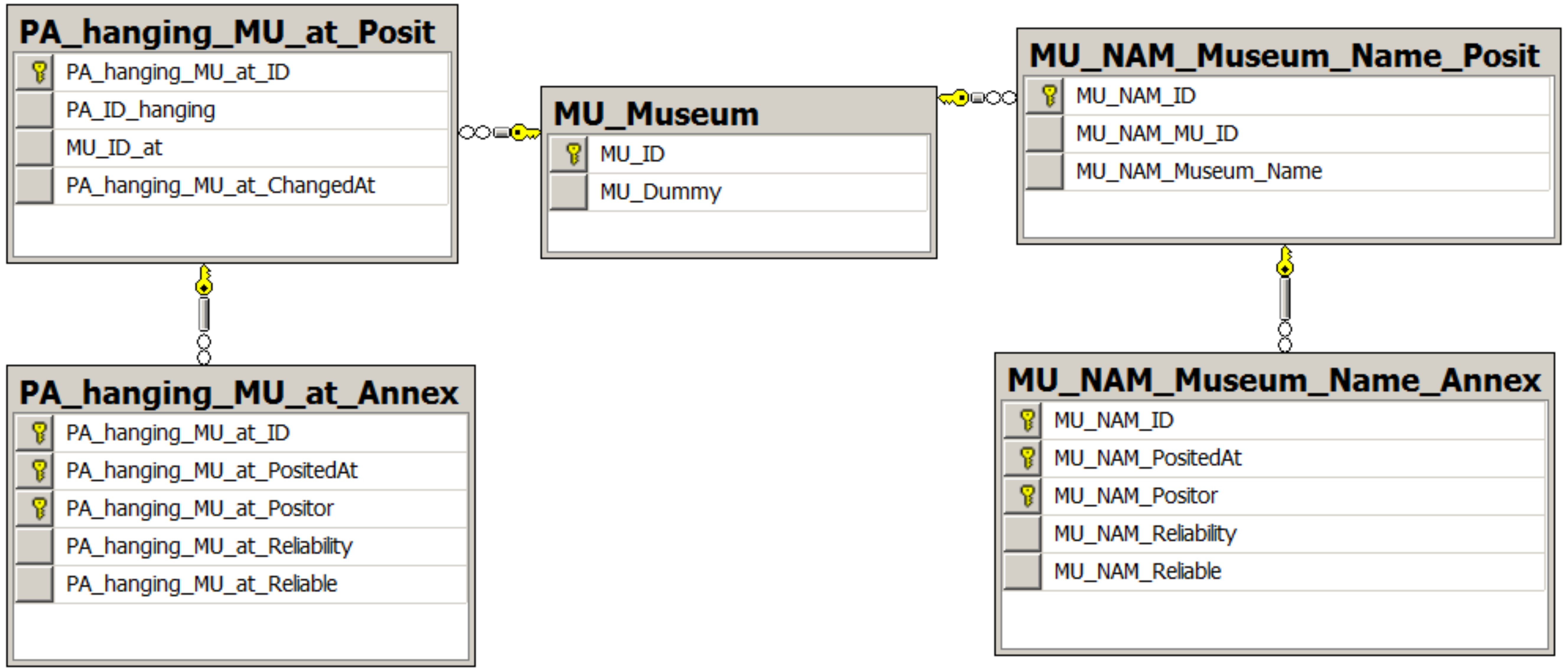
[{(2001, *started at*)}, **2019-10-20 10:15**, 2019-10-20]

[{(2001, *user*)}, **SQLAgent**, 2019-10-20]



In *uni-temporal* Anchor modeling, only posits are stored.

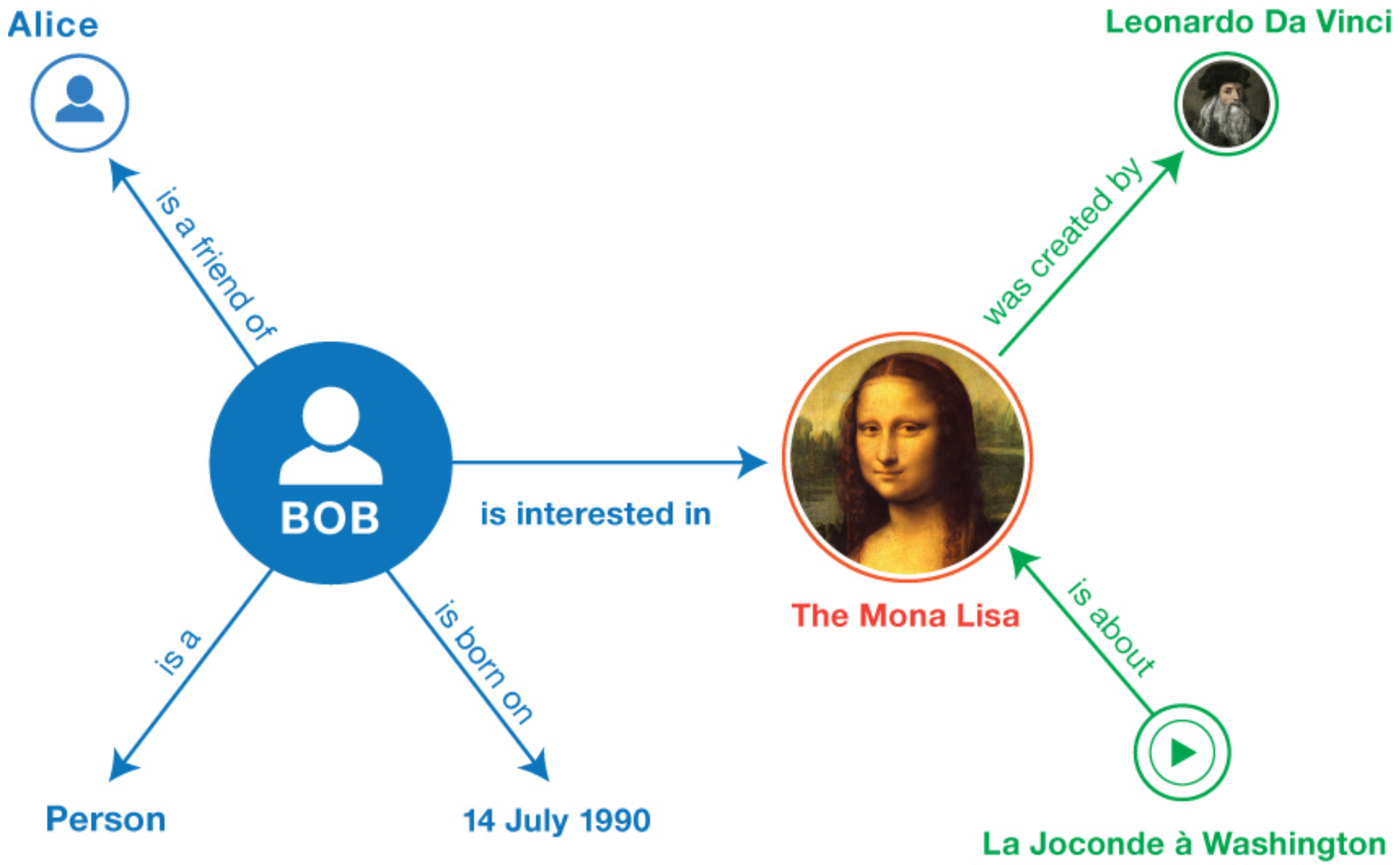
In *concurrent-reliance-temporal* Anchor modeling an Annex table is added, in which assertions are stored.



Semantic Triples

World Wide Web Consortium

W3C®



<Bob> <is a> <Person>
<Bob> <is a friend of> <Alice>
<Bob> <is born on> <the 4th of July 1990>
<http://example.name#BS12> <known as> <Bob>
<http://example.name#AW8> <known as> <Alice>

<Bob> <is a> <Person>
<Bob> <is a friend of> <Alice>
<Bob> <is born on> <the 4th of July 1990>
<http://example.name#BS12> <known as> <Bob>
<http://example.name#AW8> <known as> <Alice>

[{(http://example.name#BS12, *thing*), (http://xmlns.com/foaf/0.1/Person, *class*)}, **active**, 1990-07-04]

[{(http://example.name#BS12, *is a friend*), (http://example.name#AW8, *of*)}, **active**, 2014]

[{(http://example.name#BS12, *is born on*)}, **the 4th of July 1990**, 1990-07-04]

[{(http://example.name#BS12, *known as*)}, **Bob**, 1990-07-04]

[{(http://example.name#AW8, *known as*)}, **Alice**, 1992-10-11]

<Person> <type> <Class>
<is a friend of> <type> <Property>
<is a friend of> <domain> <Person>
<is a friend of> <range> <Person>

<Person> <type> <Class>
<is a friend of> <type> <Property>
<is a friend of> <domain> <Person>
<is a friend of> <range> <Person>

[{(http://xmlns.com/foaf/0.1/Person, *type*)}, **Class**, 1998]
[{(http://xmlns.com/foaf/0.1/Dog, *type*)}, **Class**, 1998]

[{(http://xmlns.com/foaf/0.1/Person, *is a friend*),
(http://xmlns.com/foaf/0.1/Person, *of*)},
property, 1998]

CONSTRAINTS

[{(http://xmlns.com/foaf/0.1/Dog, *is a friend*),
(http://xmlns.com/foaf/0.1/Person, *of*)},
property, 1998]

<Person> <type> <Class>
<is a friend of> <type> <Property>
<is a friend of> <domain> <Person>
<is a friend of> <range> <Person>
<is a good friend of> <subPropertyOf> <is a friend of>

901 ⟵ [{(http://xmlns.com/foaf/0.1/Person, *is a good friend*),
(http://xmlns.com/foaf/0.1/Person, *of*)},
property, 1998]

RELATED CONSTRAINTS

902 ⟵ [{(http://xmlns.com/foaf/0.1/Person, *is a friend*),
(http://xmlns.com/foaf/0.1/Person, *of*)},
property, 1998]

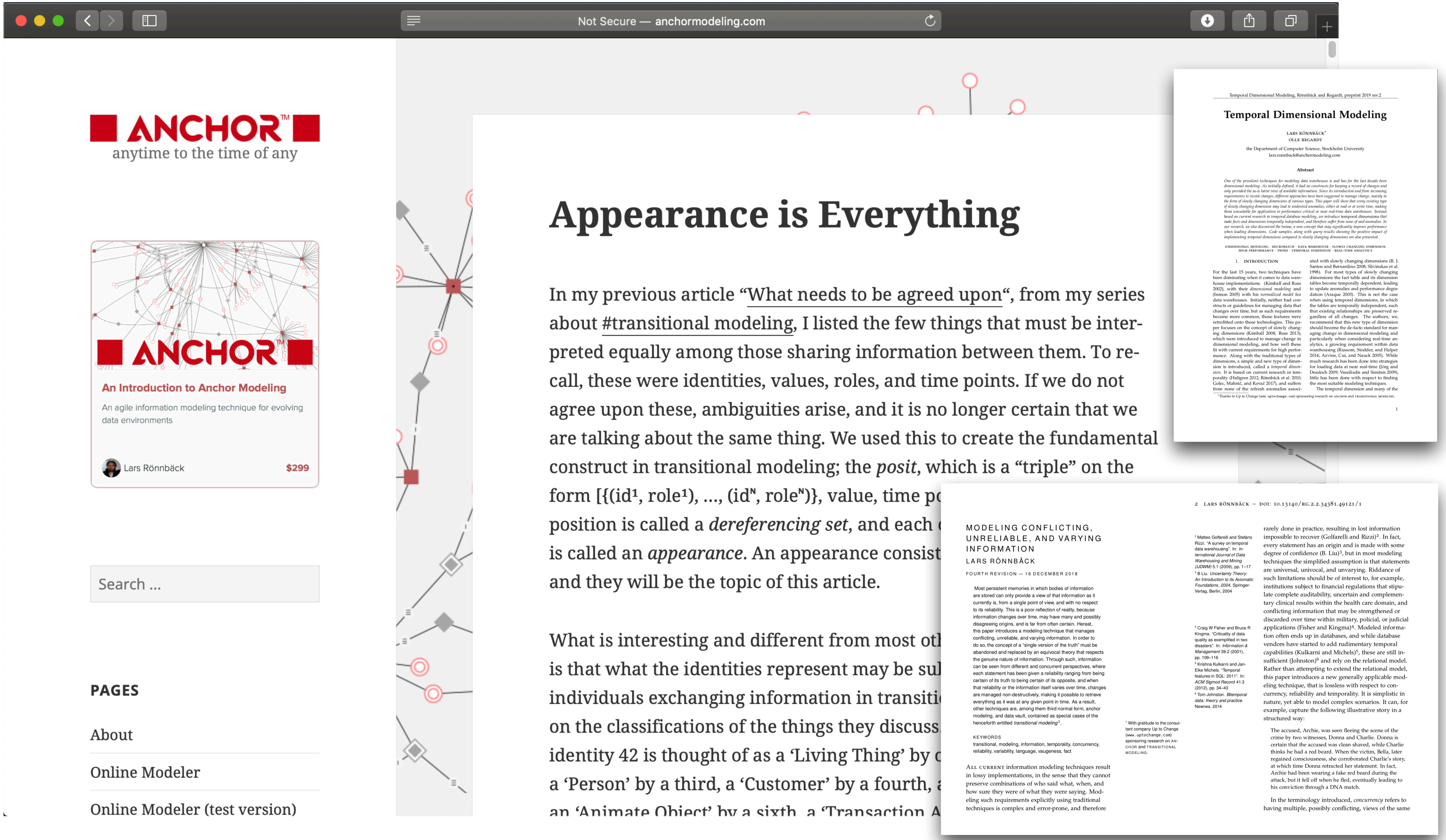
[{(901, *constraint is*), (902, *constraint of*)}, **sub property**, 2001]

Let's Twine Again

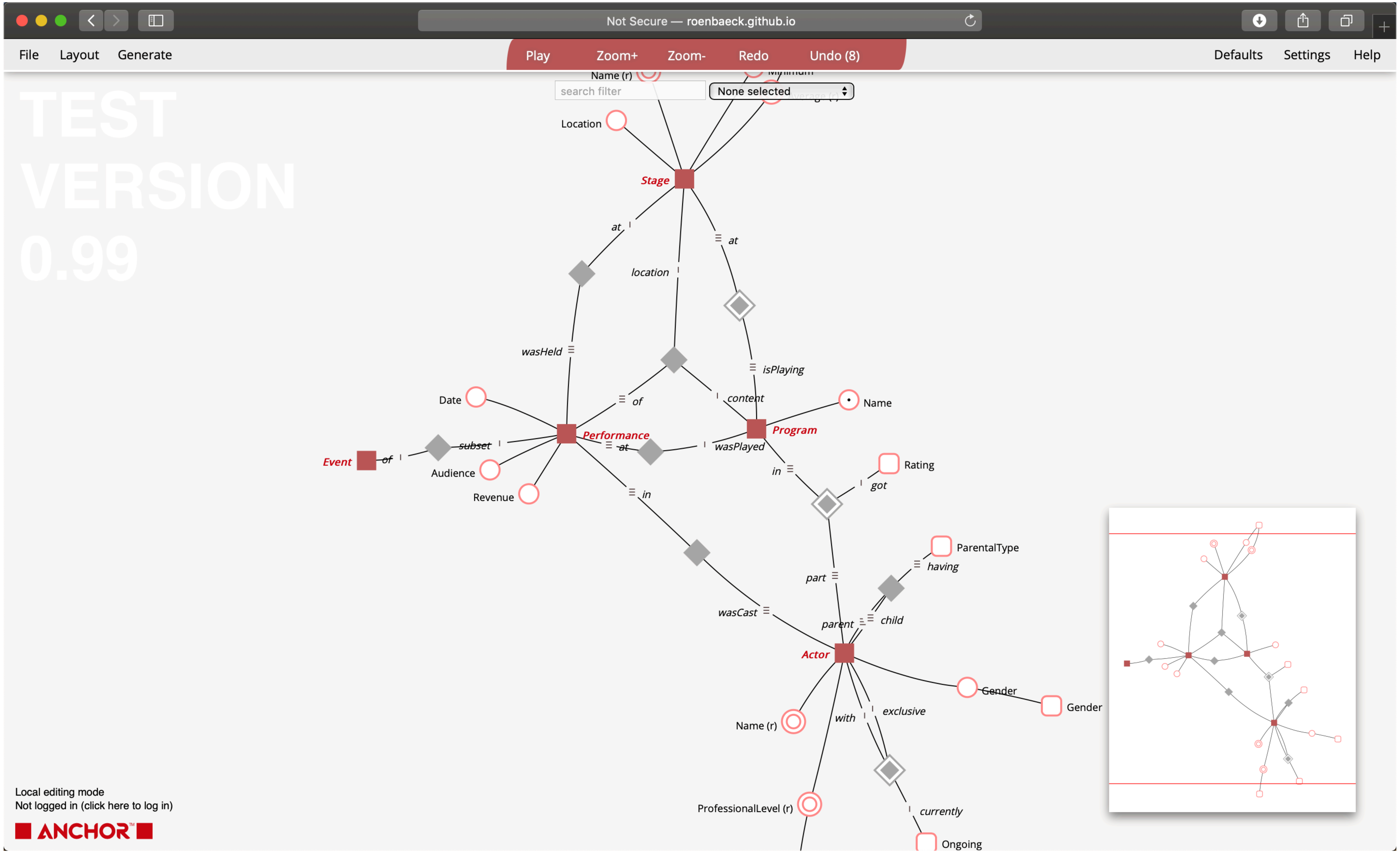
Additional Resources

Lars Rönnbäck

www.anchormodeling.com



www.anchormodeling.com/modeler/latest



github.com/Roenbaeck/sisula

github.com

README.md

sisula

sisula, short for "simple substitution language", is a language for producing text output from XML input.

The current version is built in JavaScript and should run using HTA in any Windows version from the last decade. There are no special requirements or dependencies. A legacy version using JScript in Windows Scripting Host is also available.

ETL

The ETL branch contains an SQL driven ELT framework for data warehouse automation. This framework can be used with SQL Server and is particularly useful for Anchor Modeling. There is a playlist of video tutorials on how to use it available here: https://www.youtube.com/playlist?list=PLG6-3kKEOyYlWEaEFzhcARtjqHU6zn1cH

Sisulator

The sisulator takes an XML file as input and converts this into a JSON-compatible object according to a mapping ruleset. It will then process a number of sisulets as specified in the given directive, which recieve the object as input. The sisulets are parsed and the sisula language substituted to JavaScript/JScript using regular expressions, after which the JavaScript/JScript is evaluated and the output stored.

History

sisula was introduced in Anchor Modeling in order to replace XSLT for producing text output, and a first JavaScript version of the Sisulator is built into its modeling tool. This version is derived from that work.

www.linkedin.com/in/ronnback