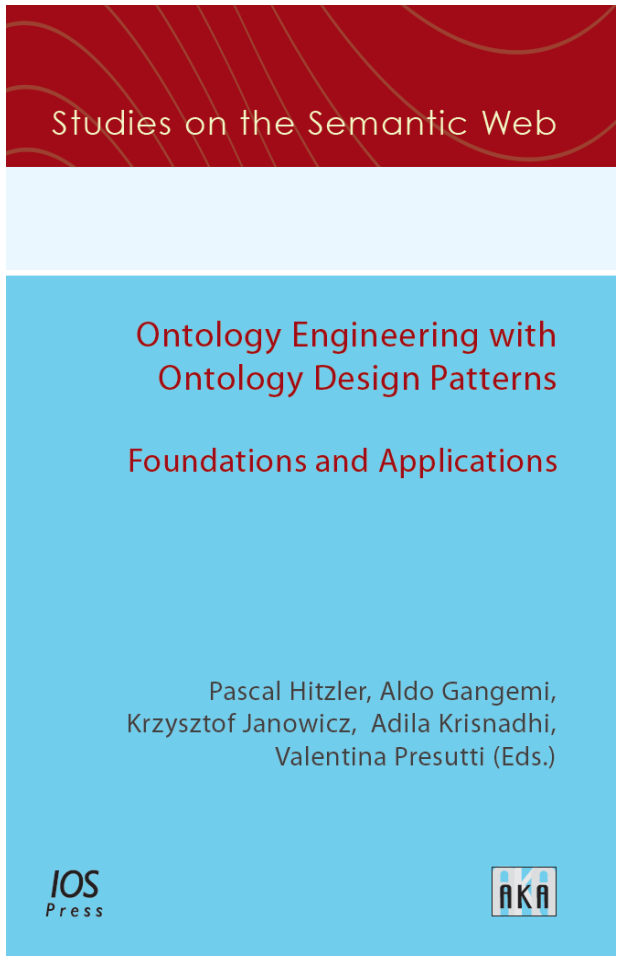
Introduction to ODPs and first pattern examples

Pascal Hitzler
Data Semantics Laboratory (DaSe Lab)
Data Science and Security Cluster (DSSC)
Wright State University
http://www.pascal-hitzler.de

See IOS Press booth



Pascal Hitzler, Aldo Gangemi, Krzysztof Janowicz, Adila Krisnathi, Valentina Presutti (eds.),
Ontology Engineering with Ontology Design Patterns: Foundations and Applications.
Studies on the Semantic Web.
IOS Press/AKA Verlag, 2016.

25% off flyer at
http://ontologydesignpatterns.org/wiki/Odp:News/17

Supplementary material for the chess example at
http://dase.cs.wright.edu/content/pattern-driven-linked-data-publishing-primer

This Tutorial (all parts)

- **Pascal Hitzler (60 mins):**
  Introduction and first examples
- **Monika Solanki (30 mins):**
  Example "modeling vaccine traceability"

coffee

- **Pascal Hitzler (60 mins):**
  Example "GeoLink Modular Ontology"
- **Agnieszka Lawrynowicz (30 mins):**
  Example "Reporting Event ODP"

lunch

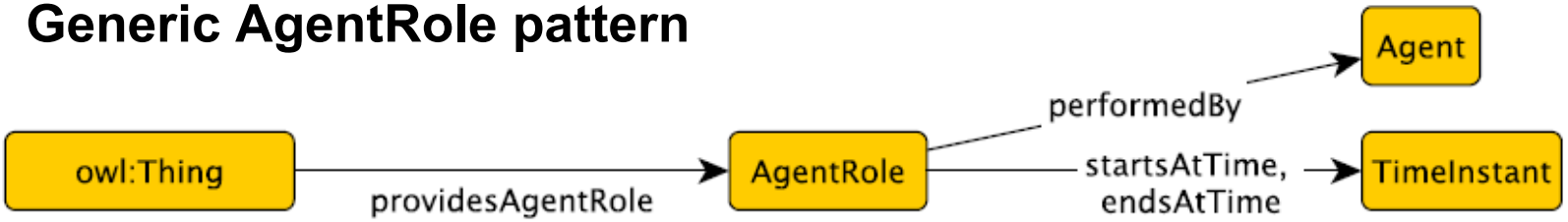
- **Karl Hammar with all others (3h):**
  Hands-on, the WebProtege XDP plug-in

What this is about

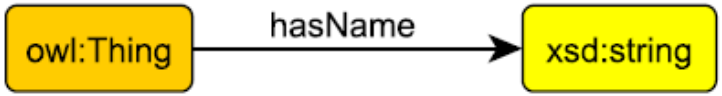
- A tutorial about ontology modeling best practices.
- Coming from the “Ontology Design Patterns” community.
- Recommended by us for all types of ontology modeling, including as graph schema for linked data and knowledge graphs.
- We are approaching a point where our experiences can consolidate into crisp recommendations, but we’re not quite there yet. I.e. there’s still work (and research) to be done.
- Join us if you’re interested:
  Google Group called “Ontology Design Patterns”

Very first example

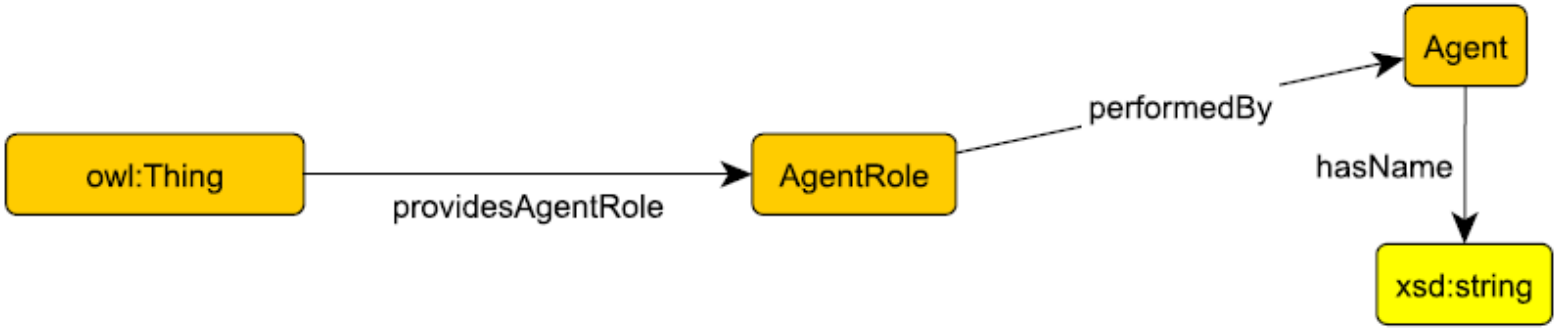
Generic AgentRole pattern



Generic NameStub pattern

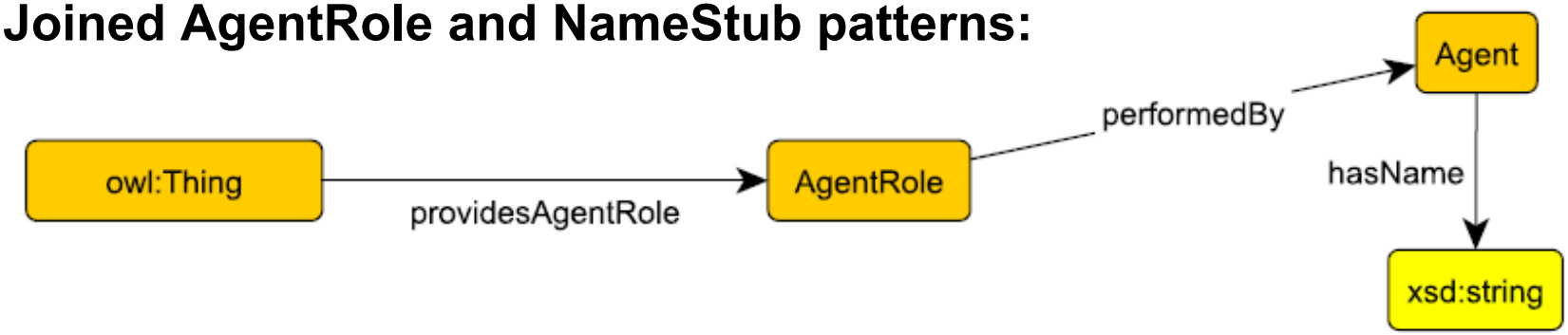


Joined:

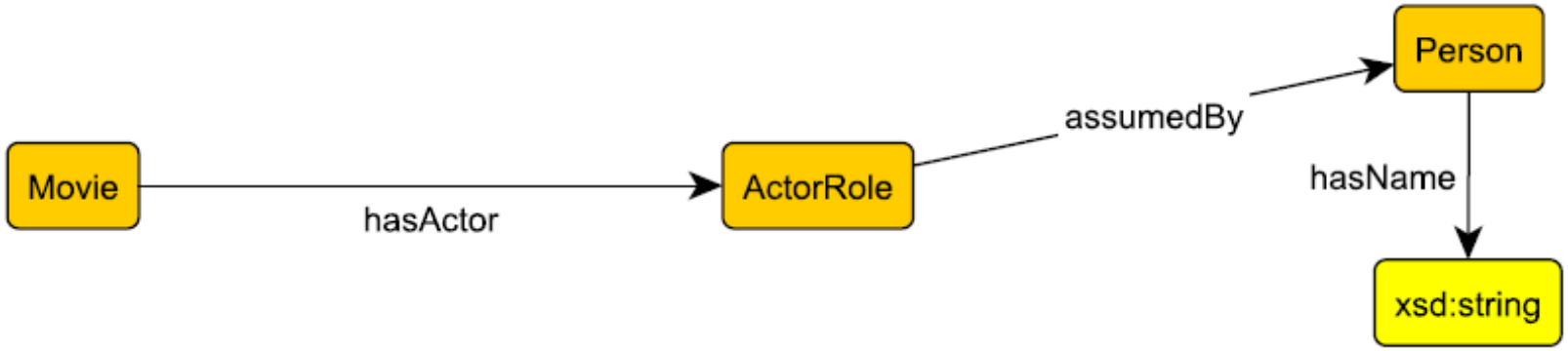


Very first example

Joined AgentRole and NameStub patterns:

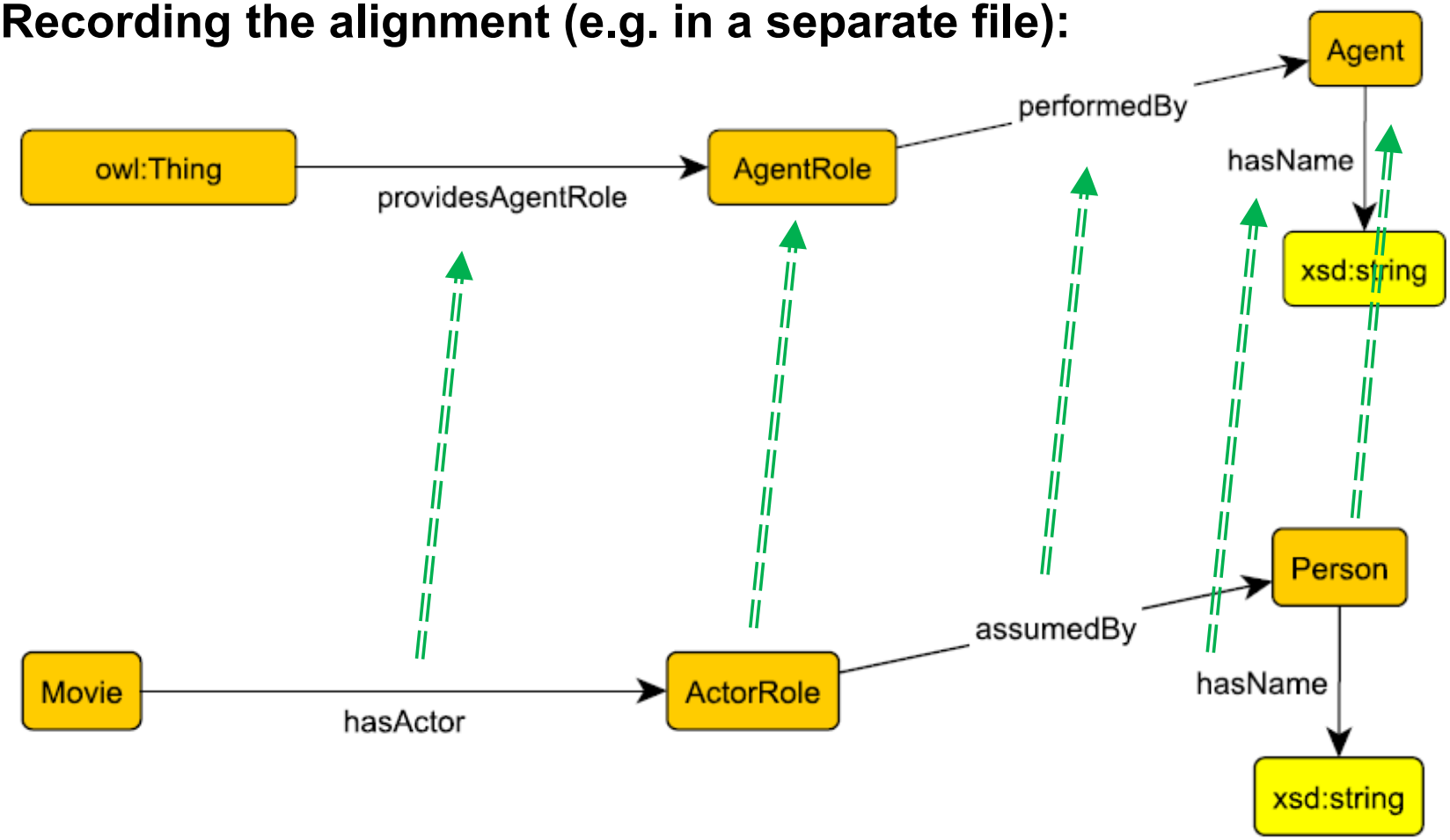


Used as a template for a concrete modeling problem:

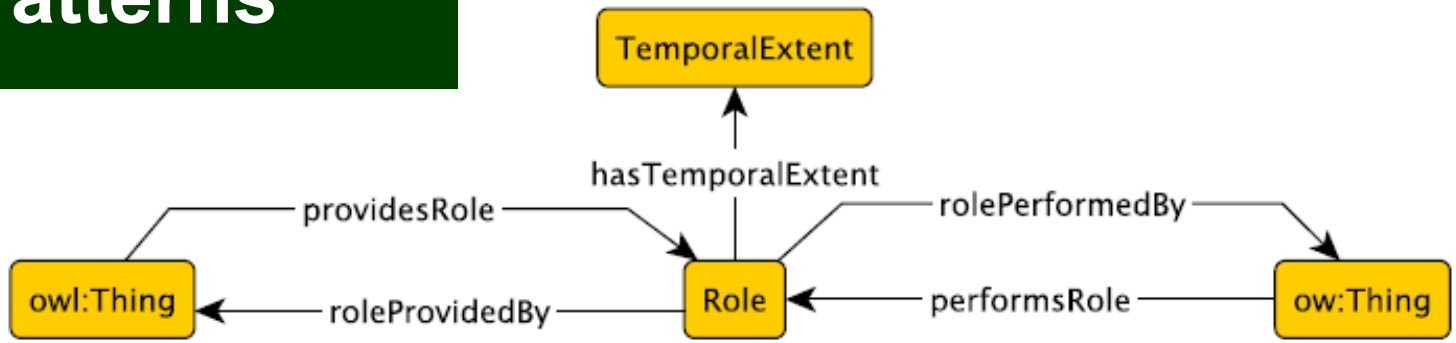


Very first example

Recording the alignment (e.g. in a separate file):

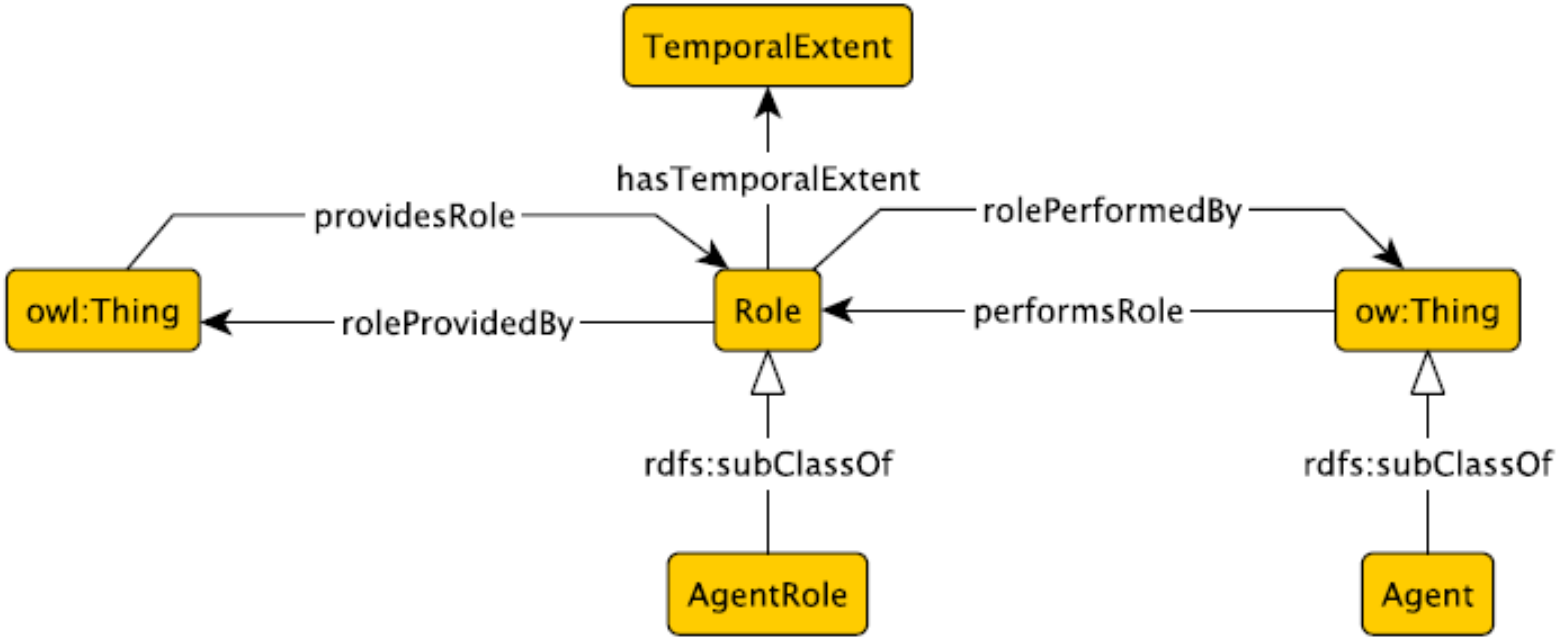


The Role Patterns



$\top \sqsubseteq \forall \text{providesRole}.\text{Role}$	range
$\exists \text{roleProvidedBy}.\top \sqsubseteq \text{Role}$	domain
$\text{providesRole} \equiv \text{roleProvidedBy}^-$	inverse
$\top \sqsubseteq \forall \text{performsRole}.\text{Role}$	range
$\exists \text{rolePerformedBy}.\top \sqsubseteq \text{Role}$	domain
$\text{rolePerformedBy} \equiv \text{performsRole}^-$	inverse
$\text{Role} \sqsubseteq \exists \text{hasTemporalExtent}.\text{TemporalExtent}$	existential
$\sqcap \forall \text{hasTemporalExtent}.\text{TemporalExtent}$	scoped range
$\sqcap (\leq 1\ \text{roleProvidedBy}.\top)$	range cardinality
$\sqcap (\leq 1\ \text{rolePerformedBy}.\top)$	range cardinality
$\text{Role} \sqsubseteq \exists \text{roleProvidedBy}.\top \sqcap \exists \text{rolePerformedBy}.\top$	existentials
$\texttt{DisjointClasses}(\text{Role}, \text{TemporalExtent})$	disjointness

The Agent Role Pattern



Axioms: all previous plus the following.

$$\text{AgentRole} \sqsubseteq \text{Role}$$

$$\exists \text{rolePerformedBy}.\text{Agent} \sqsubseteq \text{AgentRole}$$

$$\text{AgentRole} \sqsubseteq \forall \text{rolePerformedBy}.\text{Agent}$$

OWLAx

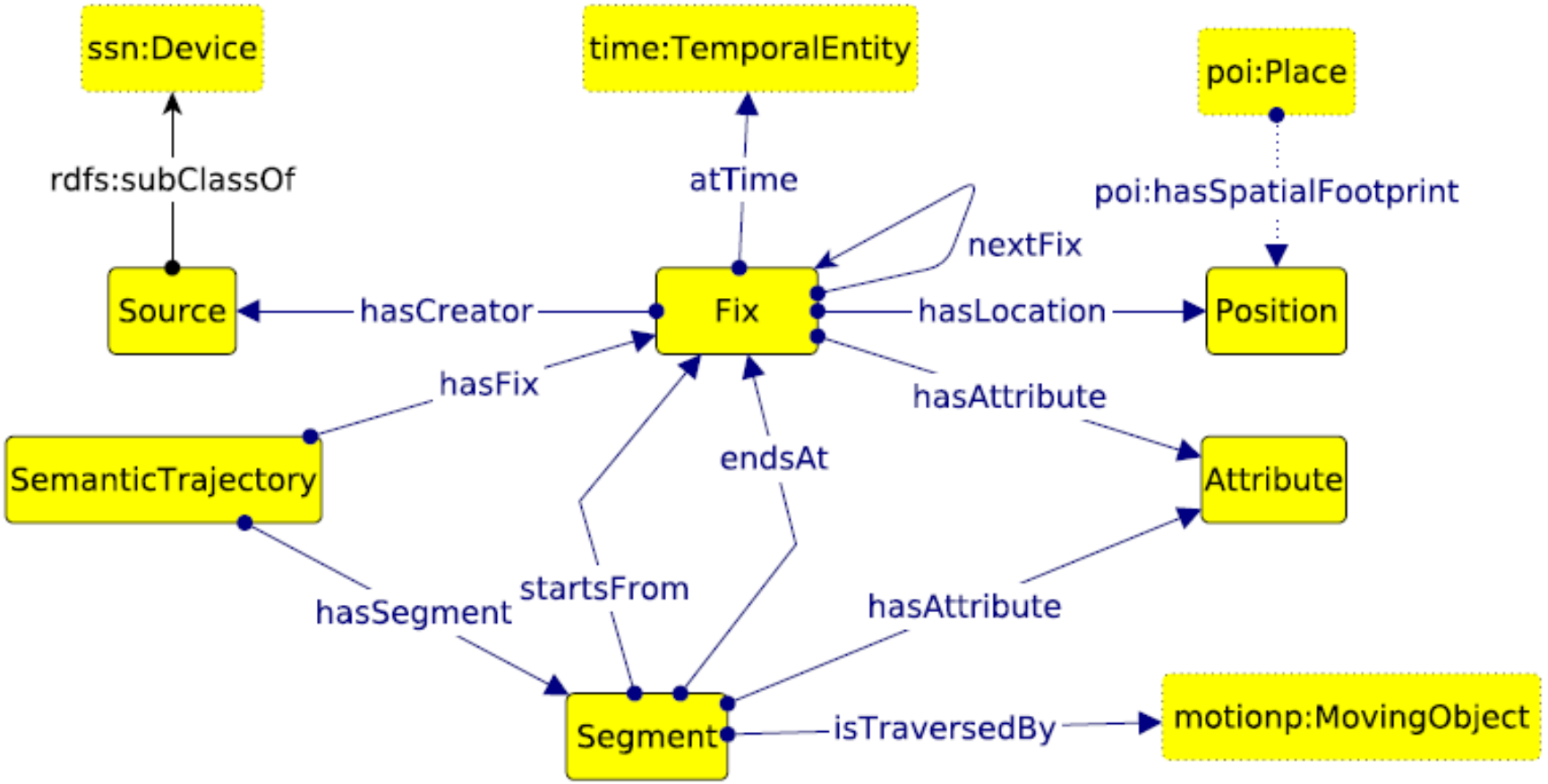
Ontology Axiomatization Support (OWLAx)

- **Protégé Plug-In**
- **Md. Kamruzzaman Sarker, Adila A. Krisnadhi, Pascal Hitzler, OWLAx: A Protege Plugin to Support Ontology Axiomatization through Diagramming. Proceedings Posters and Demos Track at ISWC 2016.**

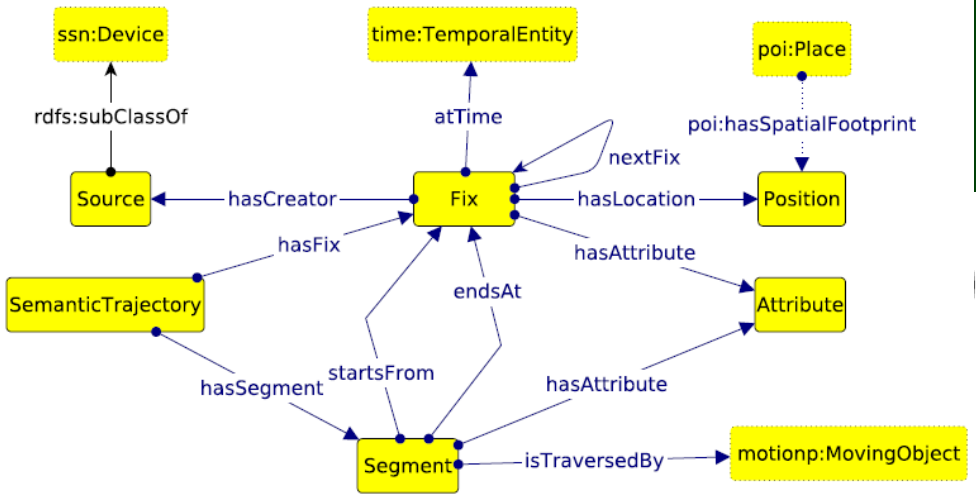
- **Insert class diagram using graphical UI**
- **System asks you whether to include corresponding axioms (taken from a pool of most common axioms for the diagram)**
- **You can of course also manually add further axioms.**

http://dase.cs.wright.edu/content/ontology-axiomatization-support

Trajectory pattern



Trajectory pattern



$$Fix \sqsubseteq \exists atTime.time{:}TemporalEntity \sqcap \exists hasLocation.Position \sqcap \exists hasFix^{-}.SemanticTrajectory$$

$$Segment \sqsubseteq \exists startsFrom.Fix \sqcap \exists endsAt.Fix$$

$$\top \sqsubseteq \leq 1 startsFrom.\top$$

$$\top \sqsubseteq \leq 1 endsAt.\top$$

$$Segment \sqsubseteq \exists hasSegment^{-}.SemanticTrajectory$$

$$startsFrom^{-} \circ endsAt \sqsubseteq hasNext$$

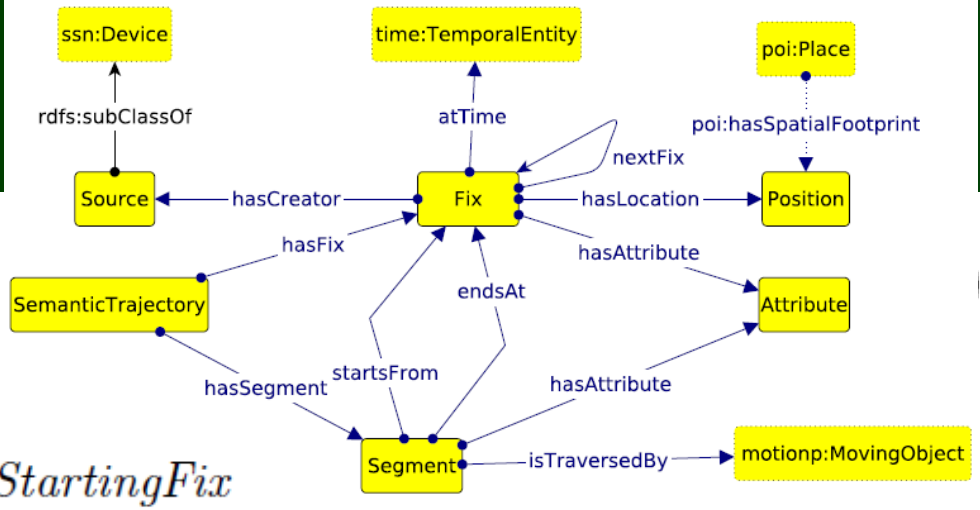
$$hasNext \sqsubseteq hasSuccessor$$

$$hasSuccessor \circ hasSuccessor \sqsubseteq hasSuccessor$$

$$hasNext^{-} \sqsubseteq hasPrevious$$

$$hasSuccessor^{-} \sqsubseteq hasPredecesor$$

Trajectory pattern



$$Fix \sqcap \neg \exists endsAt^{-}.Segment \sqsubseteq StartingFix$$

$$Fix \sqcap \neg \exists startsFrom^{-}.Segment \sqsubseteq EndingFix$$

$$Segment \sqcap \exists startsFrom.StartingFix \sqsubseteq StartingSegment$$

$$Segment \sqcap \exists endsAt.EndingFix \sqsubseteq EndingSegment$$

$$SemanticTrajectory \sqsubseteq \exists hasSegment.Segment$$

$$hasSegment \circ startsFrom \sqsubseteq hasFix$$

$$hasSegment \circ endsAt \sqsubseteq hasFix$$

$$\exists hasSegment.Segment \sqsubseteq SemanticTrajectory$$

$$\exists hasSegment^{-}.SemanticTrajectory \sqsubseteq Segment$$

$$\exists hasFix.Fix \sqsubseteq SemanticTrajectory$$

$$\exists hasFix^{-}.SemanticTrajectory \sqsubseteq Fix$$

Spatiotemporal Extent

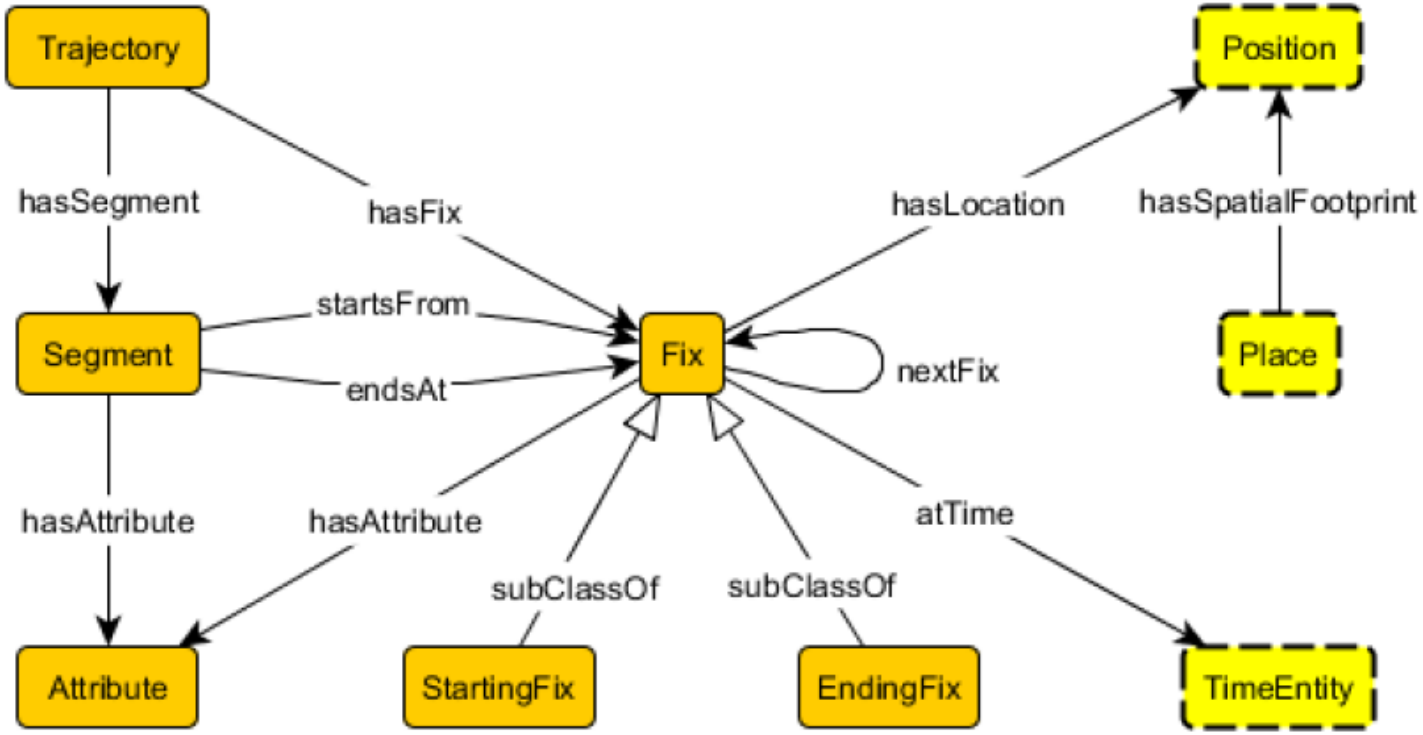


Fig. 1. Partial class diagram of the Trajectory Pattern from [2]. The dashed boxes indicate classes which are themselves (external) patterns, i.e., they need to be specified using a concrete module, or partial ontology.

Yingjie Hu, Krzysztof Janowicz, David Carral, Simon Scheider, Werner Kuhn, Gary Berg-Cross, Pascal Hitzler, Mike Dean, Dave Kolas, A Geo-Ontology Design Pattern for Semantic Trajectories. In: Thora Tenbrink, John G. Stell, Antony Galton, Zena Wood (Eds.): Spatial Information Theory - 11th International Conference, COSIT 2013, Scarborough, UK, September 2-6, 2013. Proceedings. Lecture Notes in Computer Science Vol. 8116, Springer, 2013, pp. 438-456.

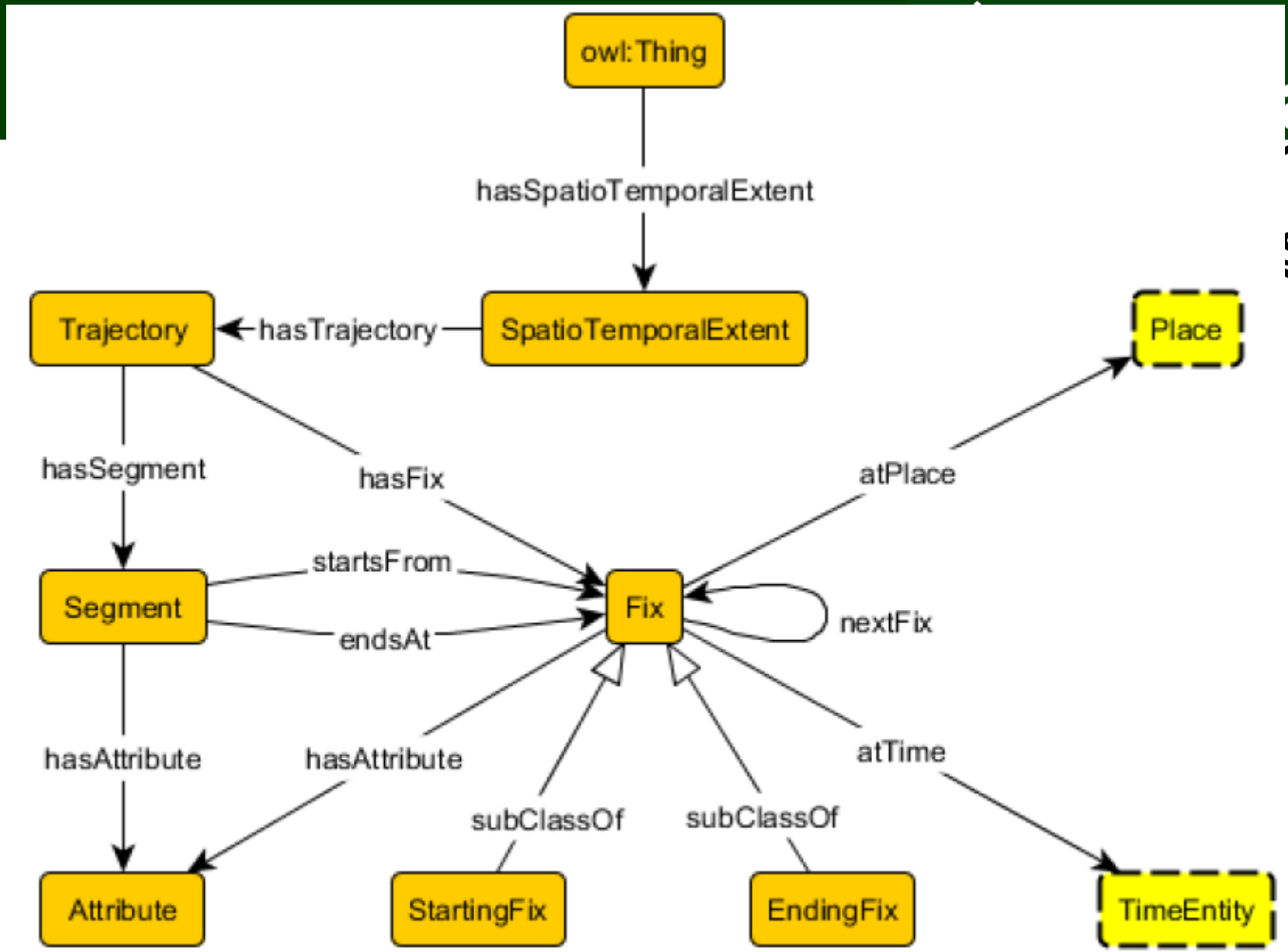


Fig. 2. Class diagram for the Spatiotemporal Extent pattern.

Axioms

Those inherited from the trajectory pattern, plus

$$\mathtt{SpatioTemporalExtent} \sqsubseteq \exists \mathtt{hasTrajectory}.\mathtt{Trajectory}$$

$$\mathtt{SpatioTemporalExtent} \sqsubseteq \forall \mathtt{hasTrajectory}.\mathtt{Trajectory}$$

$$\top \sqsubseteq \forall \mathtt{hasSpatioTemporalExtent}.\mathtt{SpatioTemporalExtent}$$

Spatiotemporal Extent

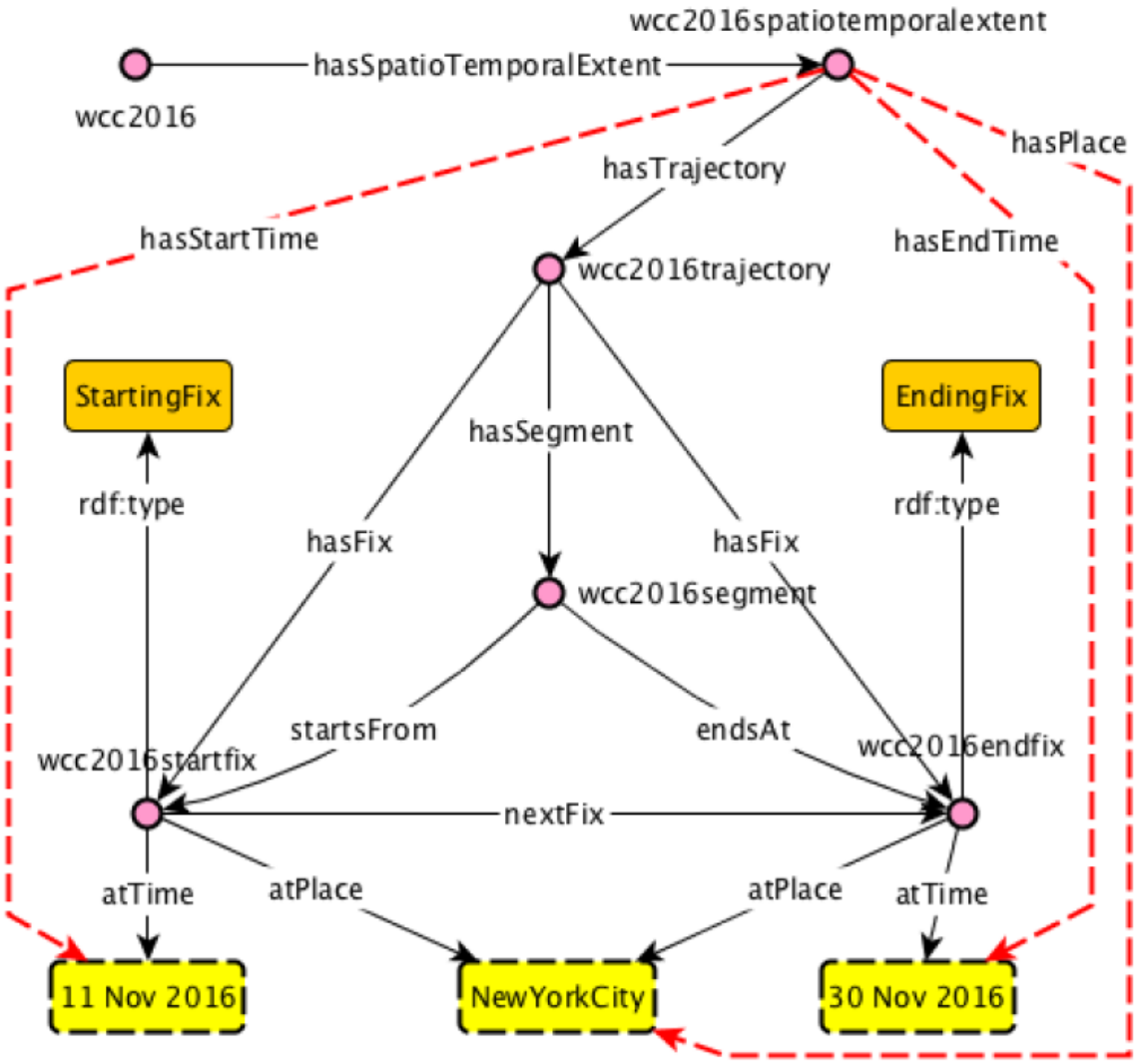


Fig. 3. Example for stationary trajectory: World Chess Championship 2016. The dashed red arrows indicate so-called shortcuts, which are discussed in the text.

Spatiotemp. Extent

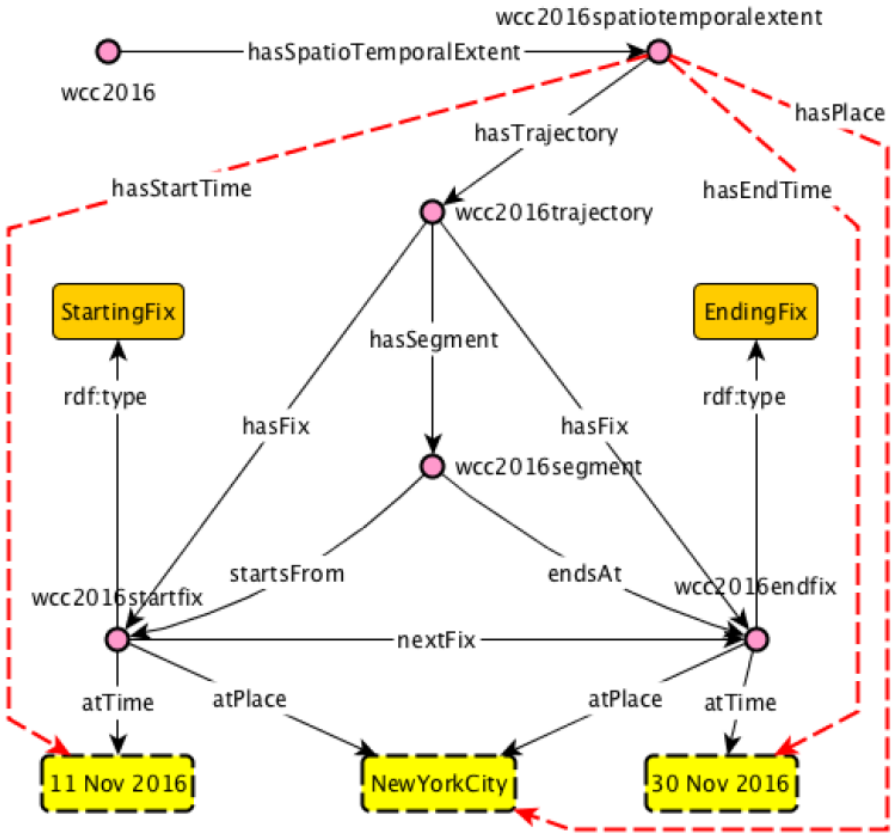


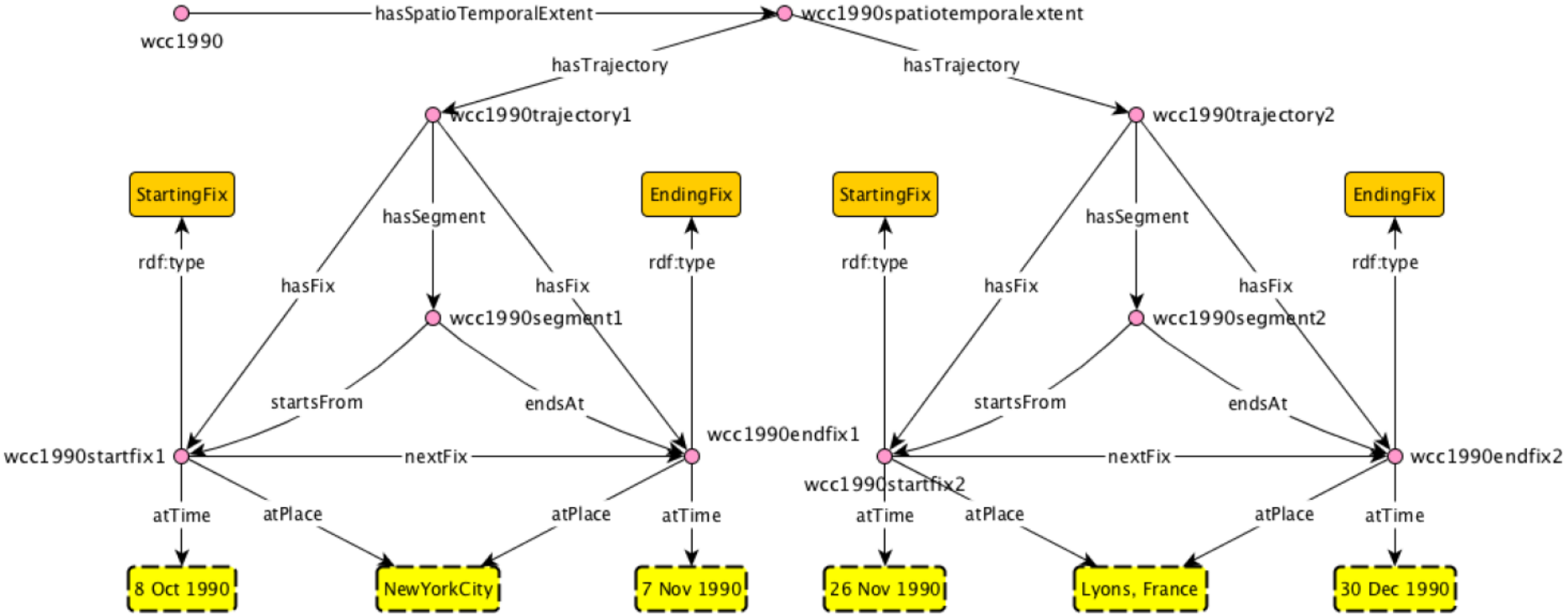
Fig. 3. Example for stationary trajectory: World Chess Championship 2016. The dashed red arrows indicate so-called shortcuts, which are discussed in the text.

$$\texttt{SpatioTemporalExtent}(x) \wedge \texttt{hasTrajectory}(x, y) \wedge \texttt{hasFix}(y, z) \wedge \texttt{StartingFix}(z) \wedge \texttt{atTime}(z, w) \rightarrow \texttt{hasStartTime}(x, w)$$

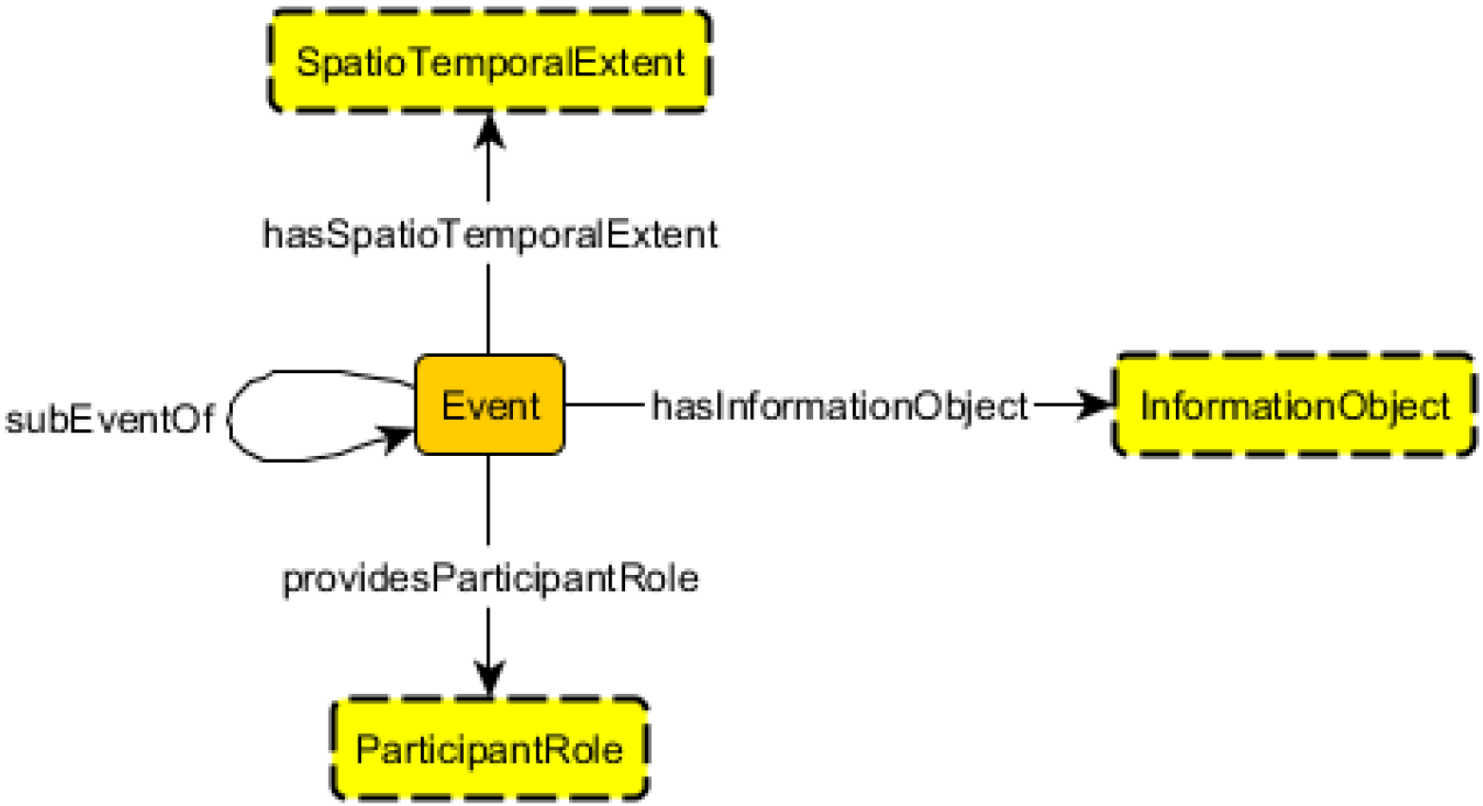
$$\texttt{SpatioTemporalExtent}(x) \wedge \texttt{hasTrajectory}(x, y) \wedge \texttt{hasFix}(y, z) \wedge \texttt{EndingFix}(z) \wedge \texttt{atTime}(z, w) \rightarrow \texttt{hasEndTime}(x, w)$$

$$\texttt{SpatioTemporalExtent}(x) \wedge \texttt{hasTrajectory}(x, y) \wedge \texttt{hasFix}(y, z) \wedge \texttt{atPlace}(z, w) \rightarrow \texttt{hasPlace}(x, w)$$

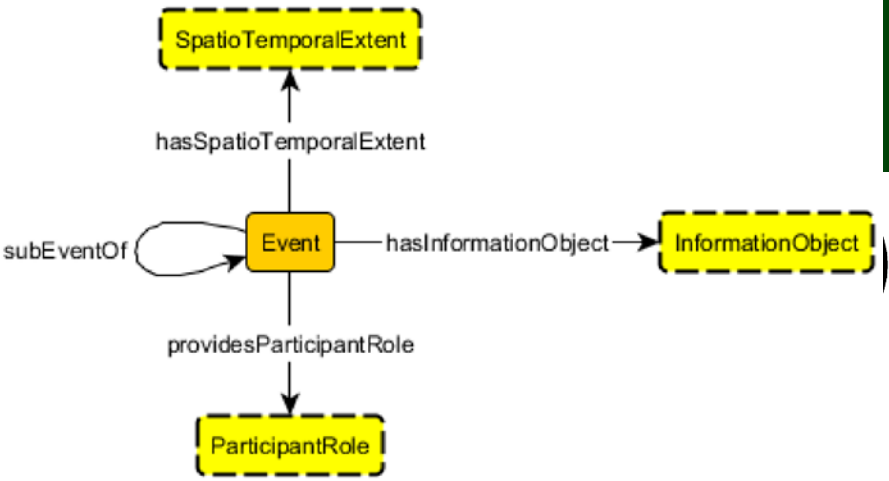
Spatiotemporal Extent



Spatiotemporal event



Spatiotemporal event



$$\mathsf{subEventOf} \circ \mathsf{subEventOf} \sqsubseteq \mathsf{subEventOf}$$

$$\exists \mathsf{subEventOf}.\top \sqsubseteq \mathsf{Event}$$

$$\top \sqsubseteq \forall \mathsf{subEventOf}.\mathsf{Event}$$

$$\mathsf{Event} \sqsubseteq \exists \mathsf{hasSpatioTemporalExtent}.\mathsf{SpatioTemporalExtent}$$

$$\top \sqsubseteq \forall \mathsf{hasSpatioTemporalExtent}.\mathsf{SpatioTemporalExtent}$$

$$\mathsf{Event} \sqsubseteq \exists \mathsf{providesParticipantRole}.\mathsf{ParticipantRole}$$

$$\top \sqsubseteq \forall \mathsf{providesParticipantRole}.\mathsf{ParticipantRole}$$

$$\mathsf{Event}(x) \wedge \mathsf{providesParticipantRole}(x, p) \wedge \mathsf{subEventOf}(x, y) \rightarrow \mathsf{providesParticipantRole}(y, p).$$

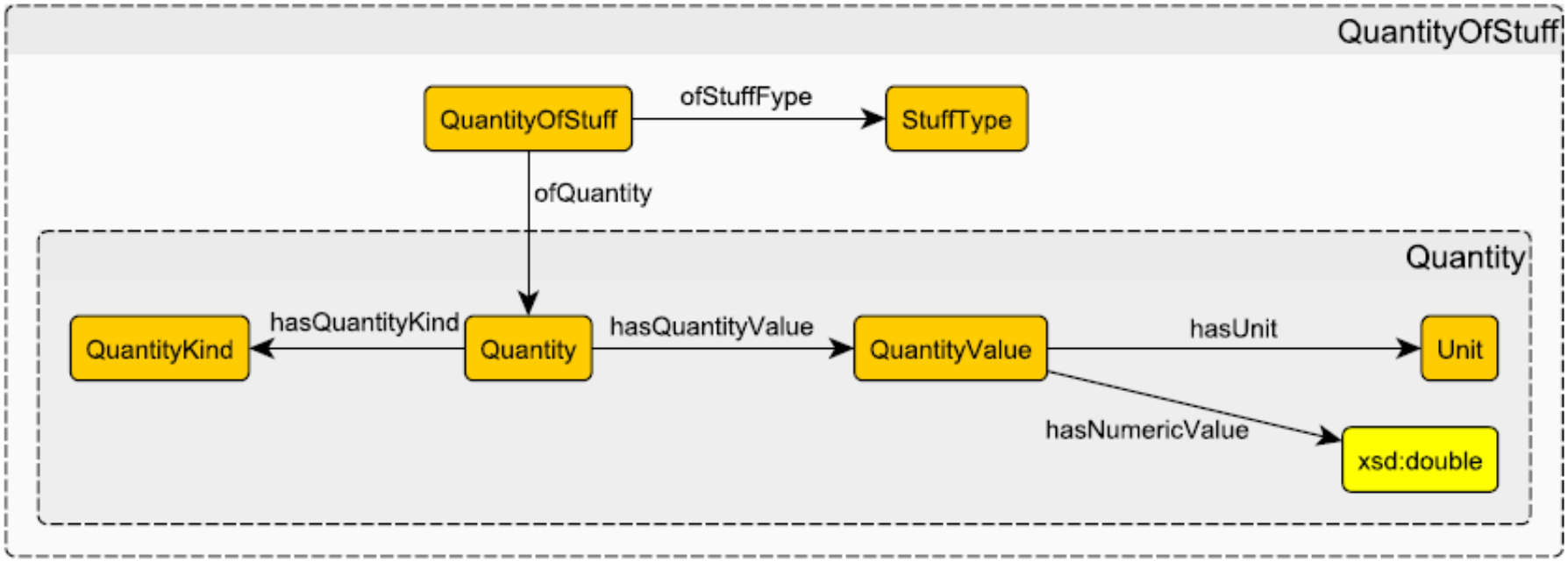
$$\mathsf{Event}(x) \wedge \mathsf{hasSpatioTemporalExtent}(x, w) \wedge \mathsf{subEventOf}(x, y) \wedge \mathsf{Event}(y) \wedge \mathsf{hasSpatioTemporalExtent}(y, z) \rightarrow \mathsf{subSTEOf}(w, z)$$

$$\top \sqsubseteq \forall \mathsf{hasInformationObject}.\mathsf{InformationObject}$$

$$\mathtt{AllDisjointClasses}(\mathsf{Event}, \mathsf{SpatioTemporalExtent}, \mathsf{ParticipantRole}, \mathsf{InformationObject})$$

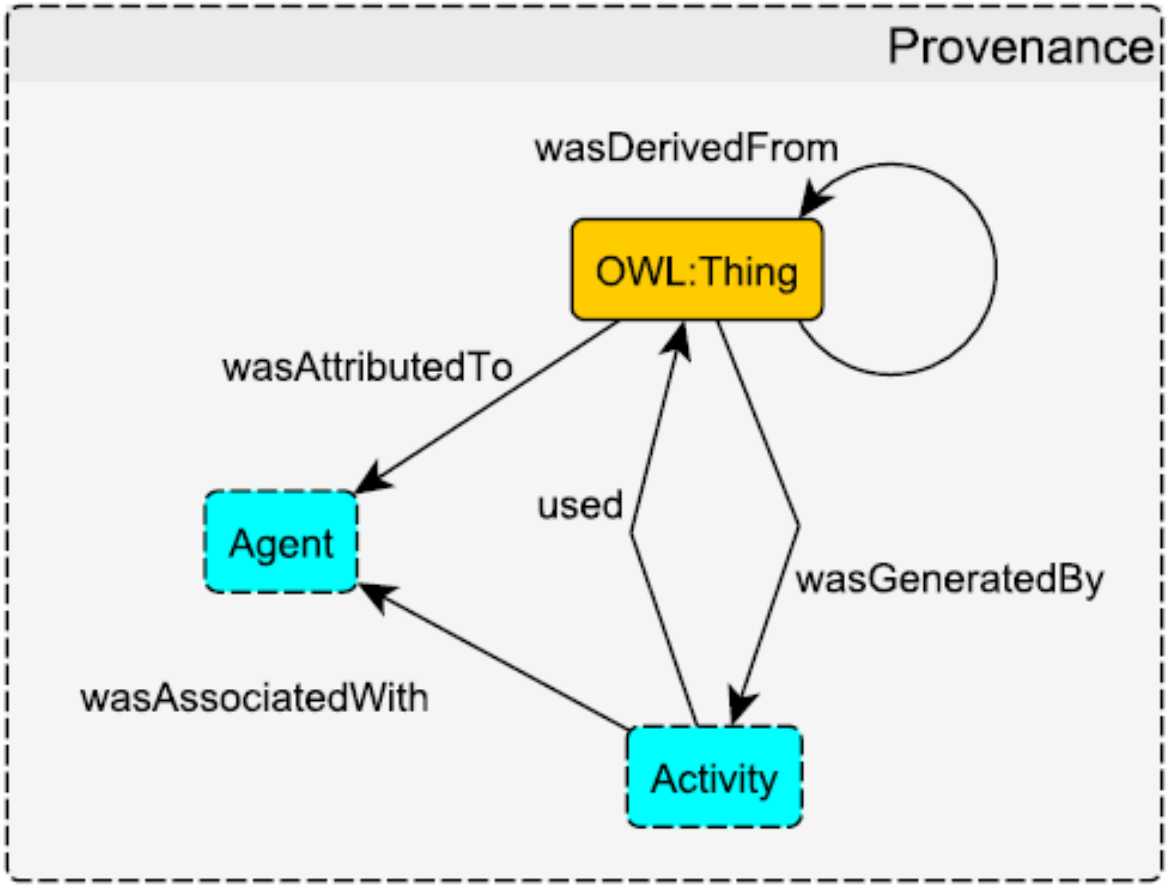
Quantities and Units

Borrowed from the QUDT ontology



Provenance

Borrowed from PROV-O



Thanks!

References

Pascal Hitzler, Aldo Gangemi, Krzysztof Janowicz, Adila Krisnathi, Valentina Presutti (eds.), Ontology Engineering with Ontology Design Patterns: Foundations and Applications. Studies on the Semantic Web. IOS Press/AKA Verlag, 2016/2017.

Hitzler, Krötzsch, Rudolph, Foundations of Semantic Web Technologies, CRC/Chapman & Hall, 2010

Adila Krisnadhi, Ontology Pattern-Based Data Integration. Dissertation, Department of Computer Science and Engineering, Wright State University, 2015.

References

Adila A. Krisnadhi, Pascal Hitzler, Krzysztof Janowicz, On capabilities and limitations of OWL regarding typecasting and ontology design pattern views. In: V. Tamma, M. Dragoni, R. Goncalves, A. Lawrynowicz (eds.), Ontology Engineering. 12th International Experiences and Directions Workshop on OWL, OWLED 2015, co-located with ISWC 2015, Bethlehem, PA, USA, October 9-10, 2015, Revised Selected Papers. Lecture Notes in Computer Science 9557, Springer, Heidelberg, 2016, pp. 105-116.

Pascal Hitzler, Markus Krötzsch, Bijan Parsia, Peter F. Patel-Schneider, Sebastian Rudolph, OWL 2 Web Ontology Language: Primer (Second Edition). W3C Recommendation, 11 December 2012.

References

Yingjie Hu, Krzysztof Janowicz, David Carral, Simon Scheider, Werner Kuhn, Gary Berg-Cross, Pascal Hitzler, Mike Dean, Dave Kolas, A Geo-Ontology Design Pattern for Semantic Trajectories. In: Thora Tenbrink, John G. Stell, Antony Galton, Zena Wood (Eds.): Spatial Information Theory - 11th International Conference, COSIT 2013, Scarborough, UK, September 2-6, 2013. Proceedings. Lecture Notes in Computer Science Vol. 8116, Springer, 2013, pp. 438-456.

Adila A. Krisnadhi, Pascal Hitzler, A Core Pattern for Events. In: Proceedings WOP 2016 at ISWC 2016.

Adila A. Krisnadhi, Pascal Hitzler, The Stub Metapattern. In: Proceedings WOP 2016 at ISWC 2016.

Adila A. Krisnadhi, Pascal Hitzler, Krzysztof Janowicz, A Spatiotemporal Extent Pattern based on Semantic Trajectories. In: Proceedins WOP 2016 at ISWC 2016.

References

Md Kamruzzaman Sarker, Adila A. Krisnadhi, David Carral, Pascal Hitzler, Rule-based OWL Modeling with ROWLTab Protege Plugin. In: Proceedings ESWC 2017. To appear.

Md. Kamruzzaman Sarker, David Carral, Adila A. Krisnadhi, Pascal Hitzler, Modeling OWL with Rules: The ROWL Protege Plugin In: Takahiro Kawamura, Heiko Paulheim (eds.), Proceedings of the ISWC 2016 Posters & Demonstrations Track co-located with 15th International Semantic Web Conference (ISWC 2016), Kobe, Japan, October 19, 2016. CEUR Workshop Proceedings 1690, CEUR-WS.org 2016.

Md. Kamruzzaman Sarker, Adila A. Krisnadhi, Pascal Hitzler, OWLAx: A Protege Plugin to Support Ontology Axiomatization through Diagramming, In: Takahiro Kawamura, Heiko Paulheim (eds.), Proceedings of the ISWC 2016 Posters & Demonstrations Track co-located with 15th International Semantic Web Conference (ISWC 2016), Kobe, Japan, October 19, 2016. CEUR Workshop Proceedings 1690, CEUR-WS.org 2016.

Some patterns and their use in the chess ontology

Pascal Hitzler
Data Semantics Laboratory (DaSe Lab)
Data Science and Security Cluster (DSSC)
Wright State University
http://www.pascal-hitzler.de

Worked Example: Chess

- Establish a searchable repository for chess data.

- Starting point are PGN files.
- Should be extendable with other information from
  - Chess websites
  - Wikipedia
  - Geographic data
  - News
  - Etc.

- Use an ontology for information integration.

GeoVoCamps modeling approach

DaSe Lab

SSC

- Collaborative modeling, group ideally has
  - More than one domain experts.
  - People familiar with the base data.
  - People understanding possible target use cases.
  - An ontology engineer familiar with the modeling approach.
  - Somebody who understands formal semantics of OWL.

- Domain experts are queried as to the main notions for the application domain.
  - E.g. for chess, these would include
    - Chess game; move; opening; tournament; players; commentary

GeoVoCamps modeling approach

DaSe Lab

SSC

- From available data and from application use cases, devise competency questions, i.e. questions which should be convertible into queries, which in turn should be answerable using the data.

1. Who played against Kasparov in the round 1994 Lineares tournament? Did (s)he play as a white or black player?
2. What is the first move taken by the black player in the Sicilian Defense opening?
3. Find all games in which Bobby Fischer, playing black, lost in the poisoned pawn variation of the Sicilian Defence opening.
4. Are there any recorded games using the Grünfeld Defence from before the 20th century?
5. What did Kasparov say about his opponent's first two moves in his commentary about his game against Topalov in the 1999 Tournament in Wijk ann Zee?
6. Who was the first non-Russian world champion after Fischer?
7. Did Bobby Fischer ever play against a grandmaster in Germany?
8. List all world championship games won by forfeit.

GeoVoCamps modeling approach

- **Then prioritize which notions to model first. In the chess case, e.g.**

  chess game
  move/half-move
  players
  opening
  tournaments
  commentary

GeoVoCamps modeling approach

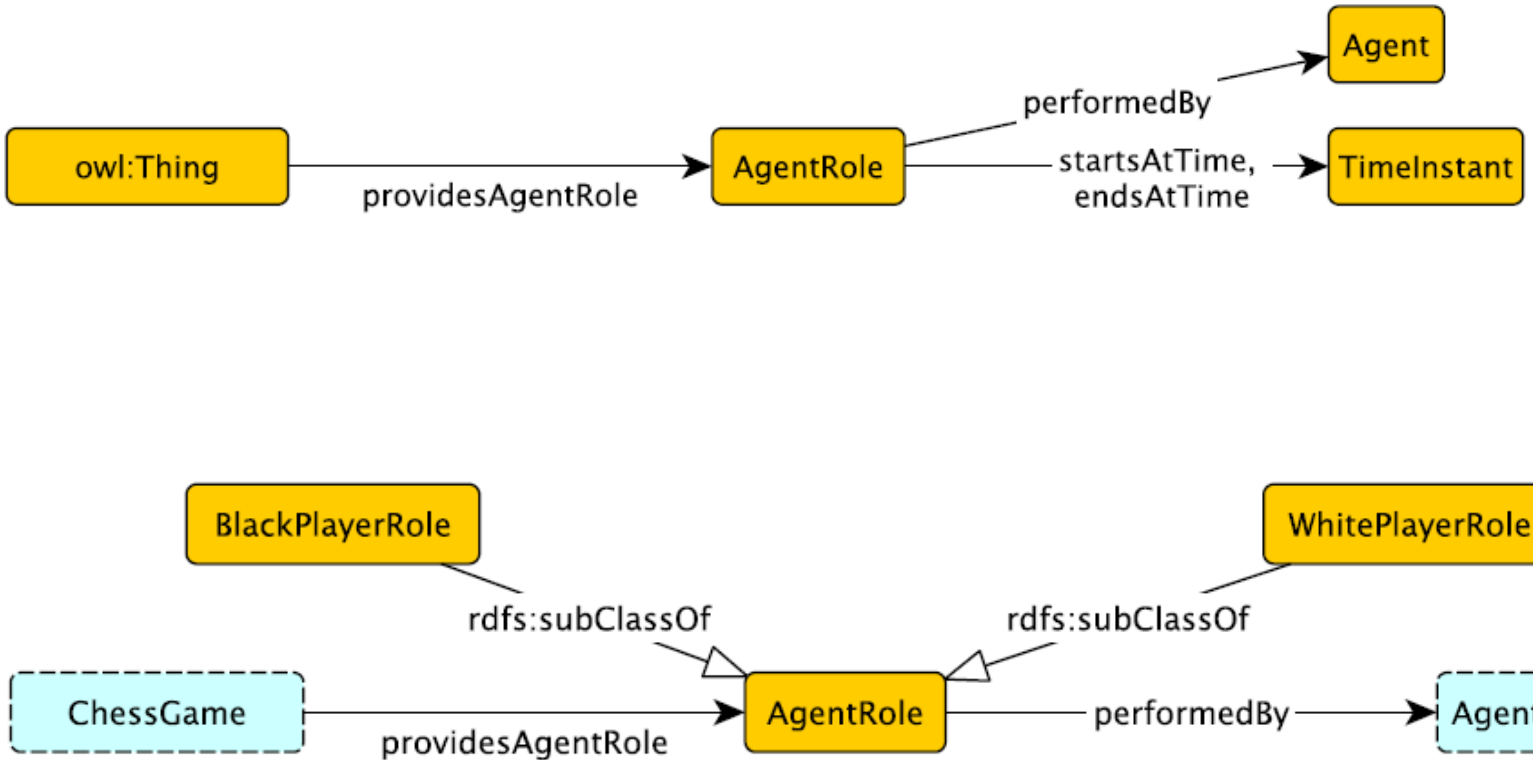
DaSe Lab

SSC

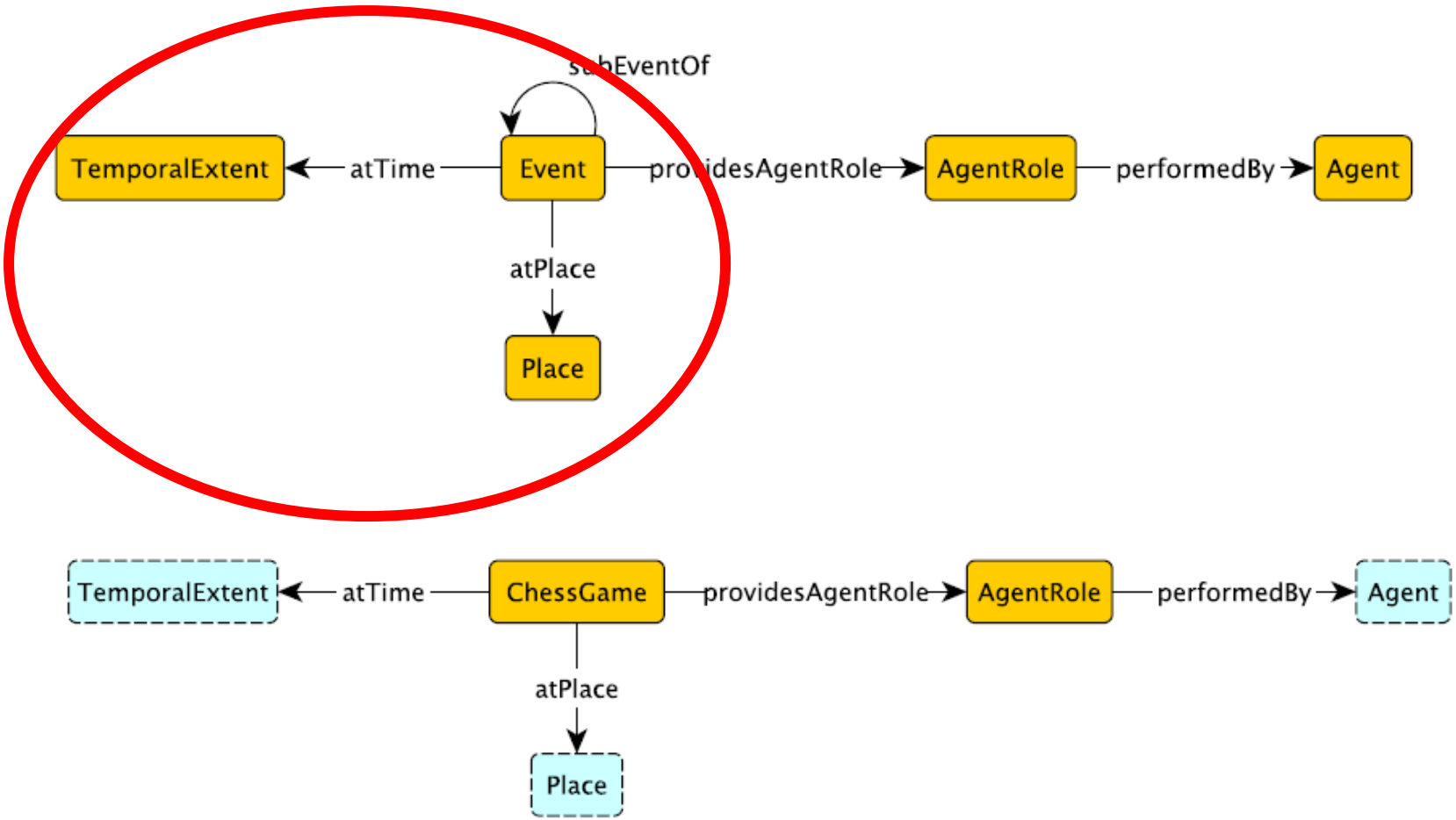
- **Understand the nature of the things you are modeling.**

Chess game	...	An Event
Half-move	…	A Subevent of a chess game
Player	…	The Role of an Agent
Opening	…	this is probably complex
tournaments	…	Events
commentary	…	this is again more complex

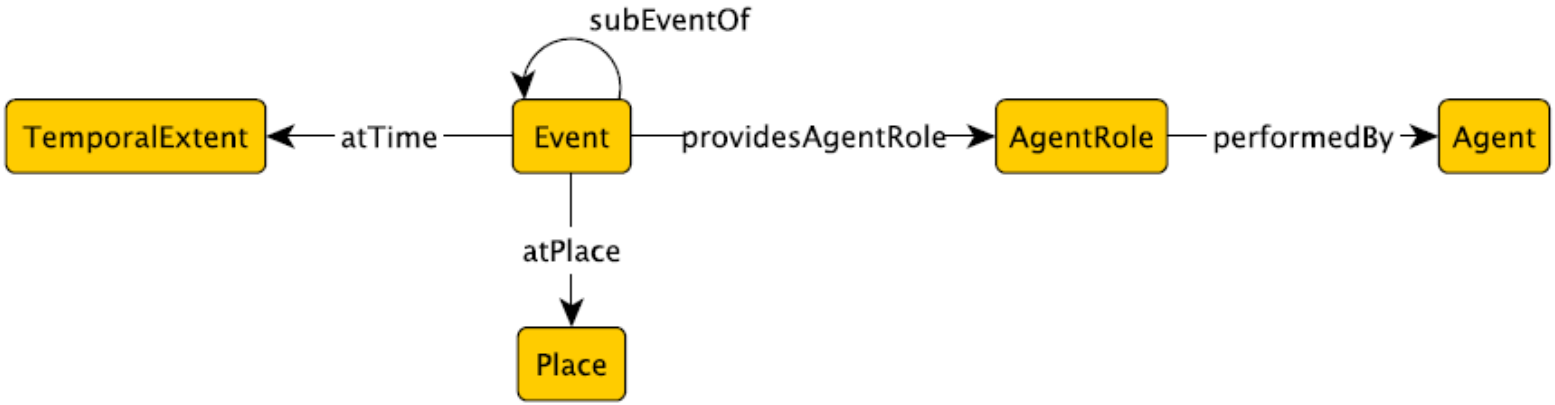
Player as AgentRole

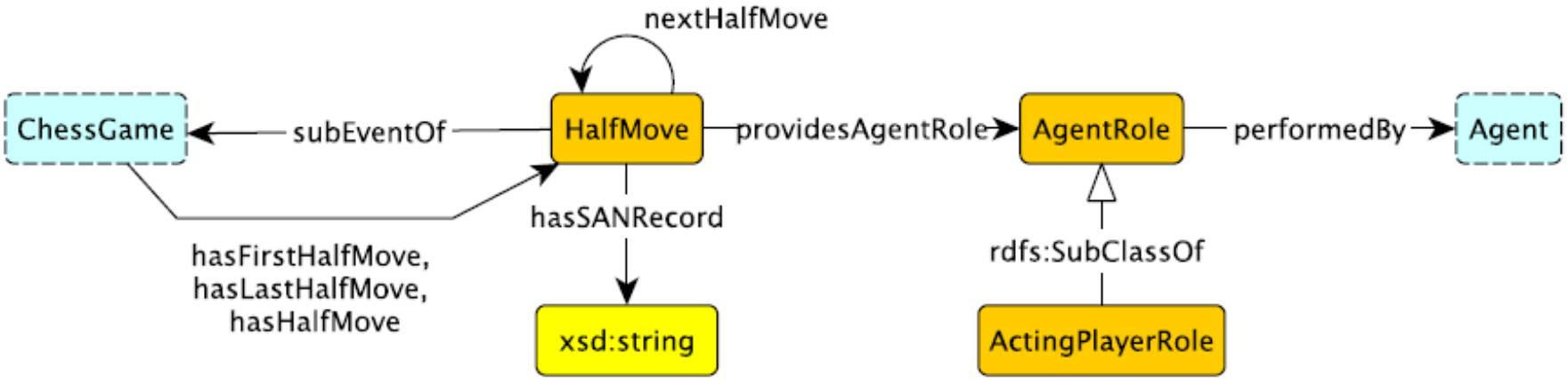


ChessGame as Event



Half-moves


subEventOf
TemporalExtent
atTime
Event
providesAgentRole
AgentRole
performedBy
Agent
atPlace
Place

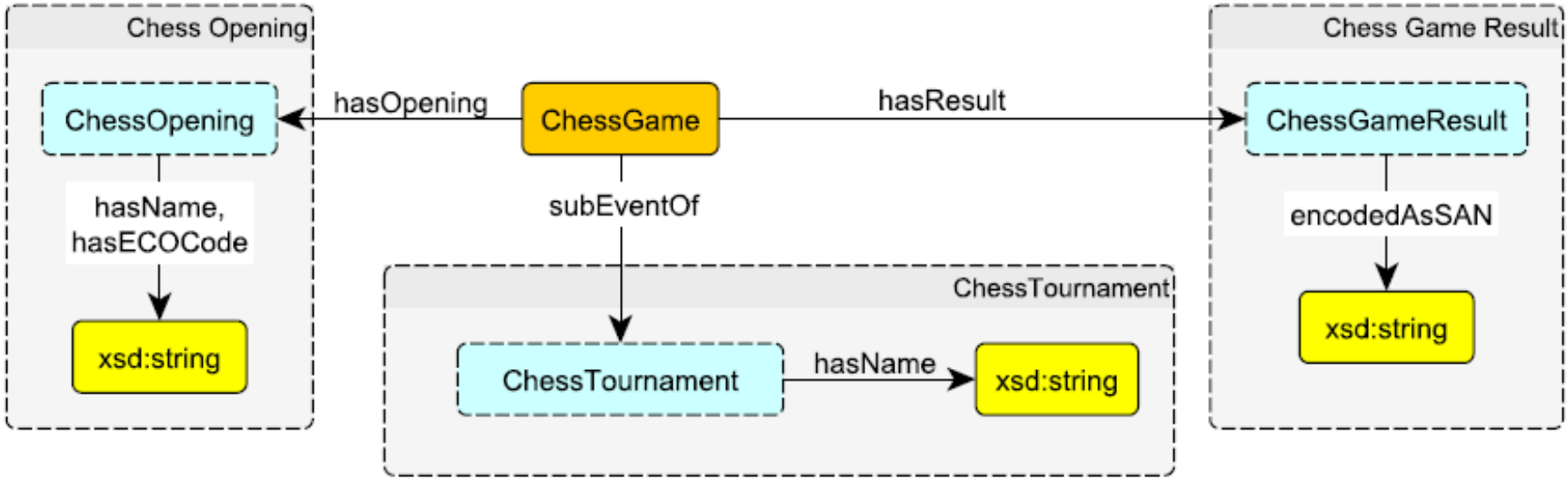

nextHalfMove
ChessGame
subEventOf
HalfMove
providesAgentRole
AgentRole
performedBy
Agent
hasSANRecord
hasFirstHalfMove,
hasLastHalfMove,
hasHalfMove
xsd:string
rdfs:SubClassOf
ActingPlayerRole

Opening, game result, etc.

We call these “stubs”.

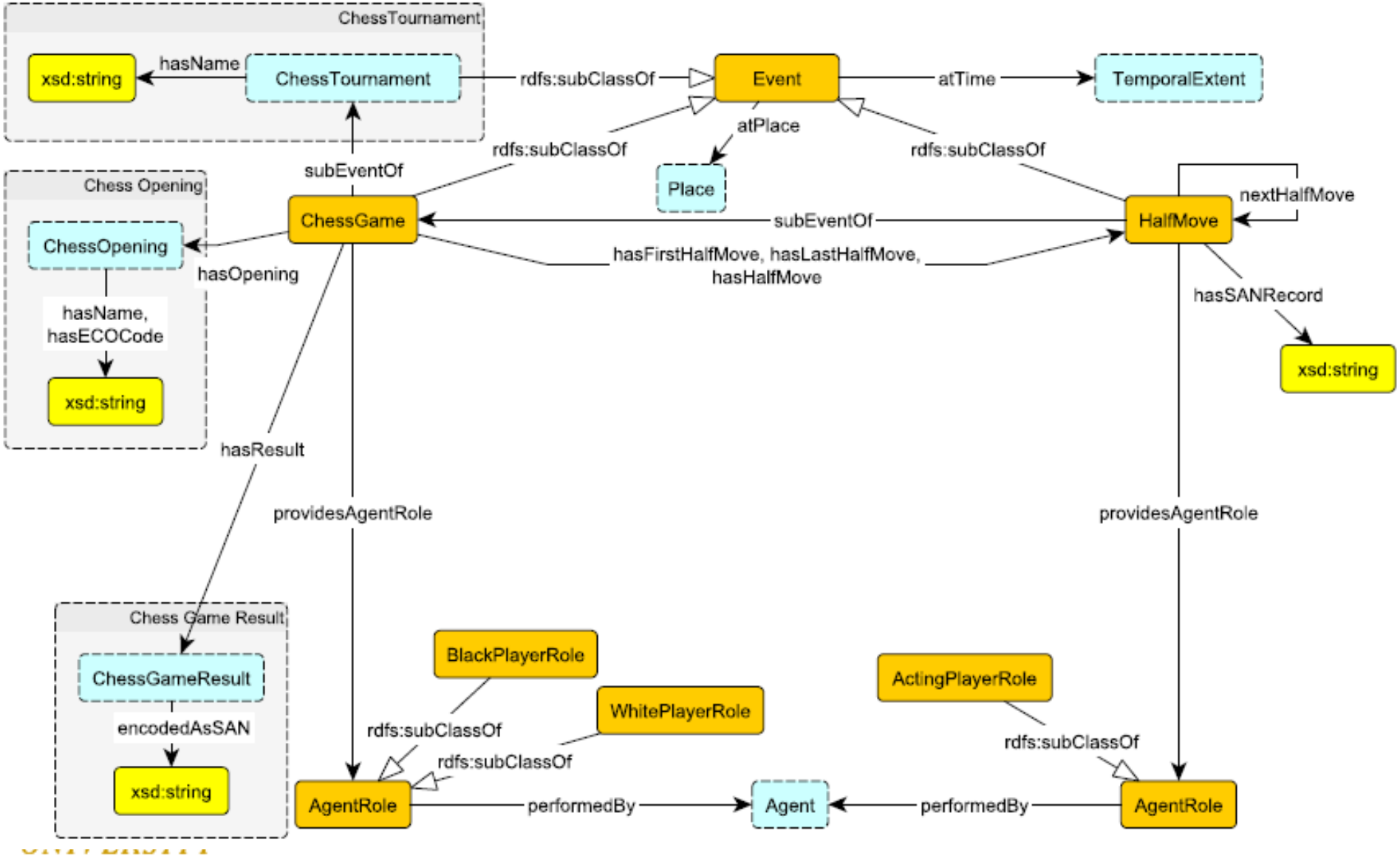
I.e. we’re aware that more fine-grained modeling will be needed for some use cases.

But currently there’s no reason to do it (not in use case, no data), so we only provide “hooks” for future development of the ontology.



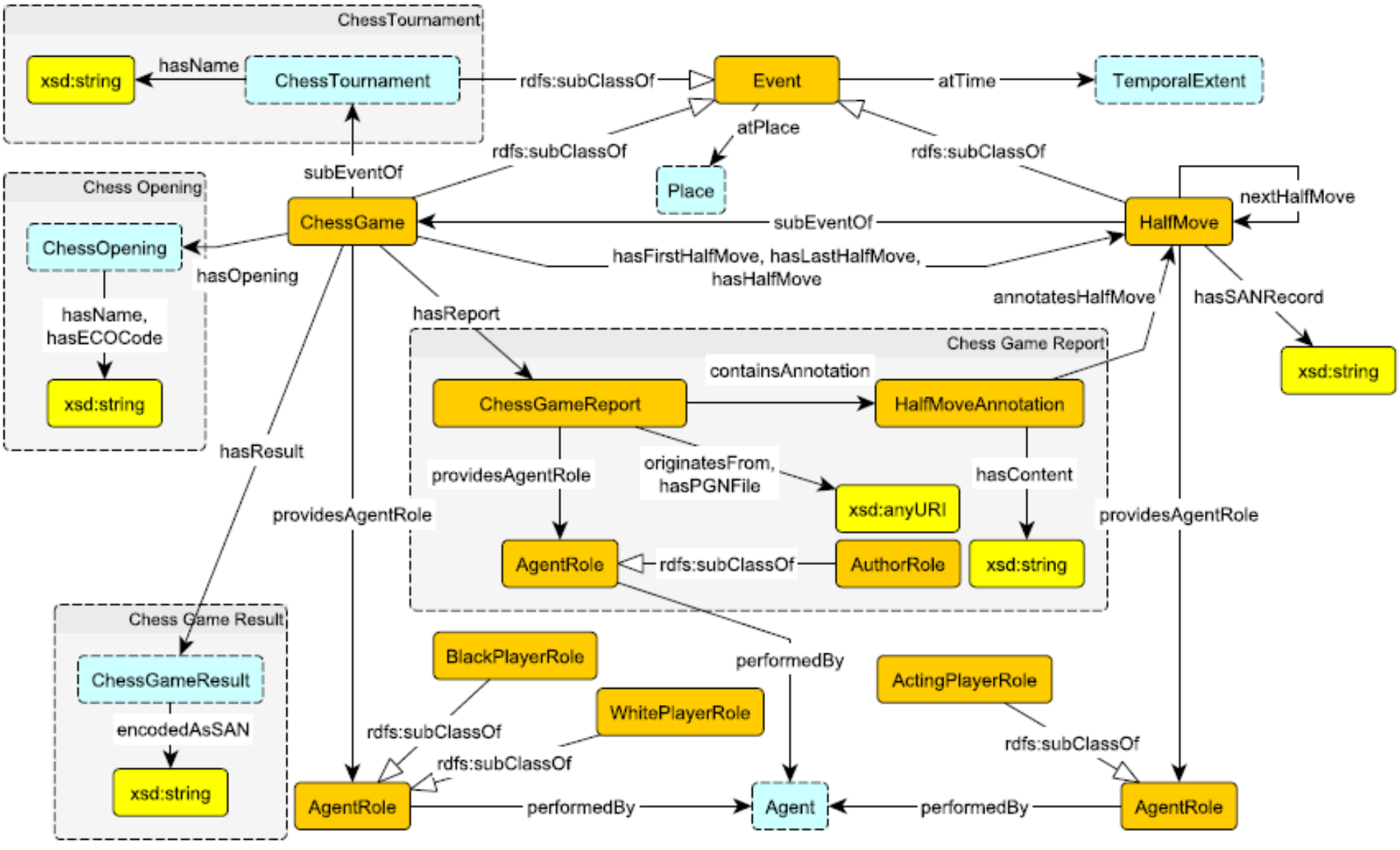
Putting things together

DaSe Lab



Adding commentaries

DaSe Lab



Adequacy check

- Triplify sample data using the ontology.
  Does it work?
- Check if competency questions can be answered.
- Add axioms as appropriate (the graph is only for intuition, the OWL axioms are the actual ontology).
- (there are more post-hoc details to be taken care of, but let's leave it at that)

Axioms

Axioms in this case are mostly straightforward:

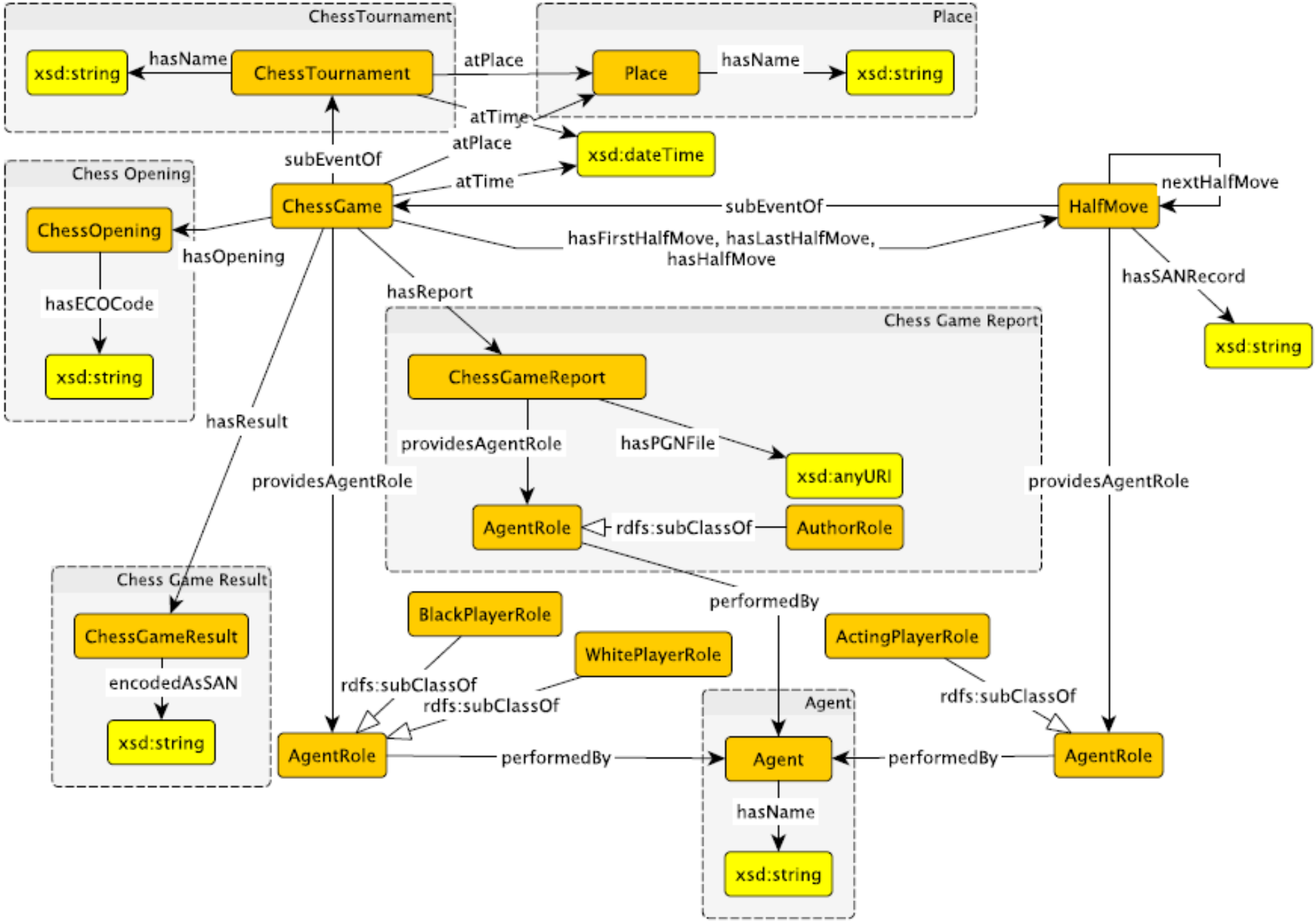
- **Inherited from Event or AgentRole**
- **Scoped domain/range restrictions, possibly with some cardinalities**
- **Basic existentials**
- **Non-cyclicity of half-move sequence**

What about adding, e.g., the following?

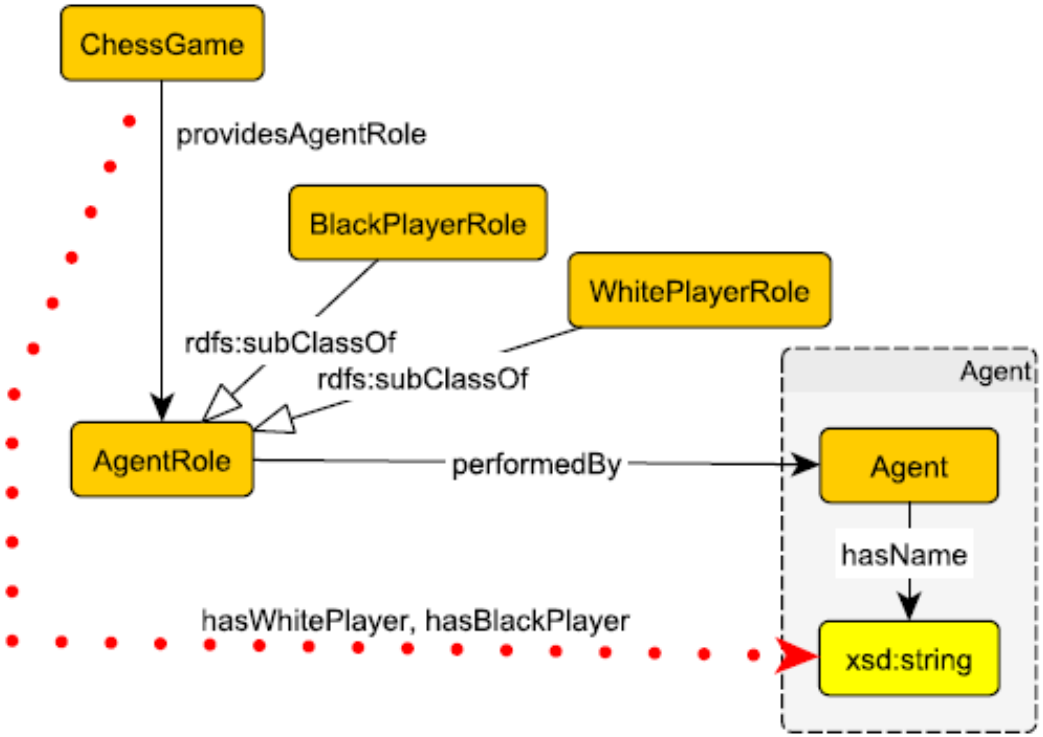
$$\mathsf{ChessGame} \sqsubseteq \geq 0 \mathsf{subEventOf}.\mathsf{ChessTournament}$$

If one of the roles of axiomatization is to improve human understanding of the ontology, then such axioms are helpful!

Shortcuts and Views


ChessTournament
xsd:string
hasName
ChessTournament
atPlace
Place
Place
hasName
xsd:string
atTime
xsd:dateTime
subEventOf
atPlace
atTime
Chess Opening
ChessOpening
hasOpening
ChessGame
subEventOf
HalfMove
nextHalfMove
hasFirstHalfMove, hasLastHalfMove,
hasHalfMove
hasECOCode
xsd:string
hasReport
hasSANRecord
xsd:string
Chess Game Report
ChessGameReport
providesAgentRole
hasPGNFile
xsd:anyURI
hasResult
providesAgentRole
providesAgentRole
AgentRole
rdfs:subClassOf
AuthorRole
Chess Game Result
performedBy
BlackPlayerRole
WhitePlayerRole
ActingPlayerRole
ChessGameResult
encodedAsSAN
xsd:string
rdfs:subClassOf
rdfs:subClassOf
rdfs:subClassOf
Agent
AgentRole
performedBy
Agent
performedBy
AgentRole
hasName
xsd:string

Shortcuts



$$\text{ChessGame}(x) \wedge \text{pAR}(x, y) \wedge \text{WhitePlayerRole}(y) \wedge \text{performedBy}(y, z) \wedge \text{Agent}(z) \wedge \text{hasName}(z, s) \rightarrow \text{hasWhitePlayer}(x, s)$$

$$\text{ChessGame}(x) \wedge \text{pAR}(x, y) \wedge \text{BlackPlayerRole}(y) \wedge \text{performedBy}(y, z) \wedge \text{Agent}(z) \wedge \text{hasName}(z, s) \rightarrow \text{hasBlackPlayer}(x, s)$$

Translating the rules

$$\mathsf{ChessGame}(x) \wedge \mathsf{pAR}(x,y) \wedge \mathsf{WhitePlayerRole}(y) \wedge \mathsf{performedBy}(y,z) \wedge \mathsf{Agent}(z) \wedge \mathsf{hasName}(z,s) \rightarrow \mathsf{hasWhitePlayer}(x,s)$$

$$\mathsf{ChessGame}(x) \wedge \mathsf{pAR}(x,y) \wedge \mathsf{BlackPlayerRole}(y) \wedge \mathsf{performedBy}(y,z) \wedge \mathsf{Agent}(z) \wedge \mathsf{hasName}(z,s) \rightarrow \mathsf{hasBlackPlayer}(x,s)$$

$$\mathsf{ChessGame} \sqsubseteq \exists R_1.\mathrm{Self}$$

$$\mathsf{WhitePlayerRole} \sqsubseteq \exists R_2.\mathrm{Self}$$

$$\mathsf{Agent} \sqsubseteq \exists R_3.\mathrm{Self}$$

$$R_1 \circ \mathsf{pAR} \circ R_2 \circ \mathsf{performedBy} \circ R_3 \circ \mathsf{hasName} \sqsubseteq \mathsf{hasWhitePlayer}$$

However note that the introduction of additional role chains may cause violations of regularity restrictions.

ROWLTab

DaSe Lab

SSC

Modeling OWL with Rules (ROWLTab)

- **Protégé Plug-In**
- Md. Kamruzzaman Sarker, David Carral, Adila A. Krisnadhi, Pascal Hitzler, Modeling OWL with Rules: The ROWL Protege Plugin. Proceedings Posters and Demos Track at ISWC 2016.
- Md Kamruzzaman Sarker, Adila A. Krisnadhi, David Carral, Pascal Hitzler, Rule-based OWL Modeling with ROWLTab Protege Plugin. In: Proceedings ESWC 2017.
- **Enter rules using interface very similar to SWRLTab.**
- **But rules are converted into OWL axioms (whenever possible) instead of DL-safe rules.**

E.g., $\mathrm{Pig}(x) \rightarrow \mathrm{Mammal}(x)$ **becomes** $\mathrm{Pig} \sqsubseteq \mathrm{Mammal}$

and thus carries the correct semantics.

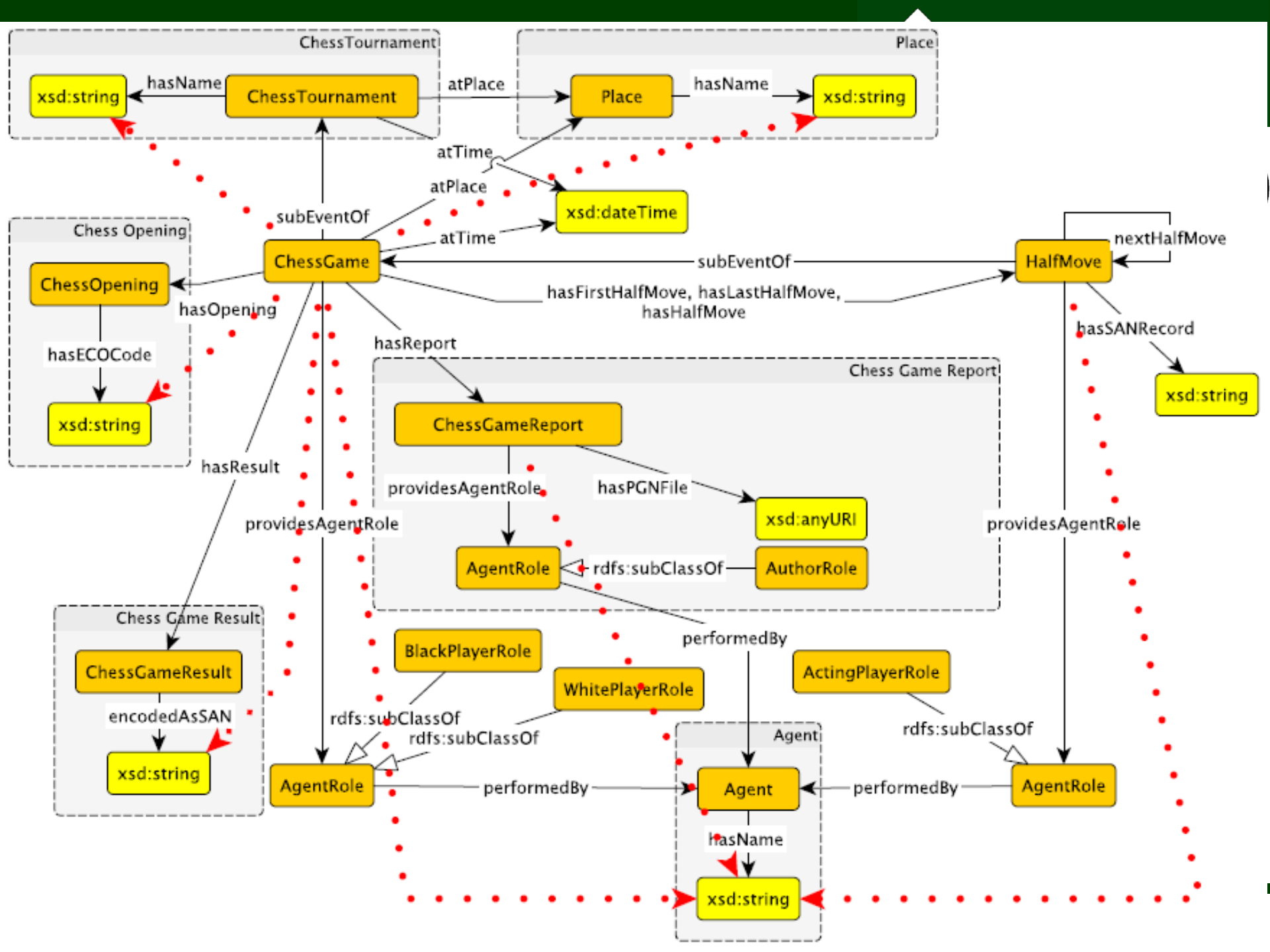
http://dase.cs.wright.edu/content/modeling-owl-rules

We evaluated that ROWL leads to quicker modeling with fewer errors.

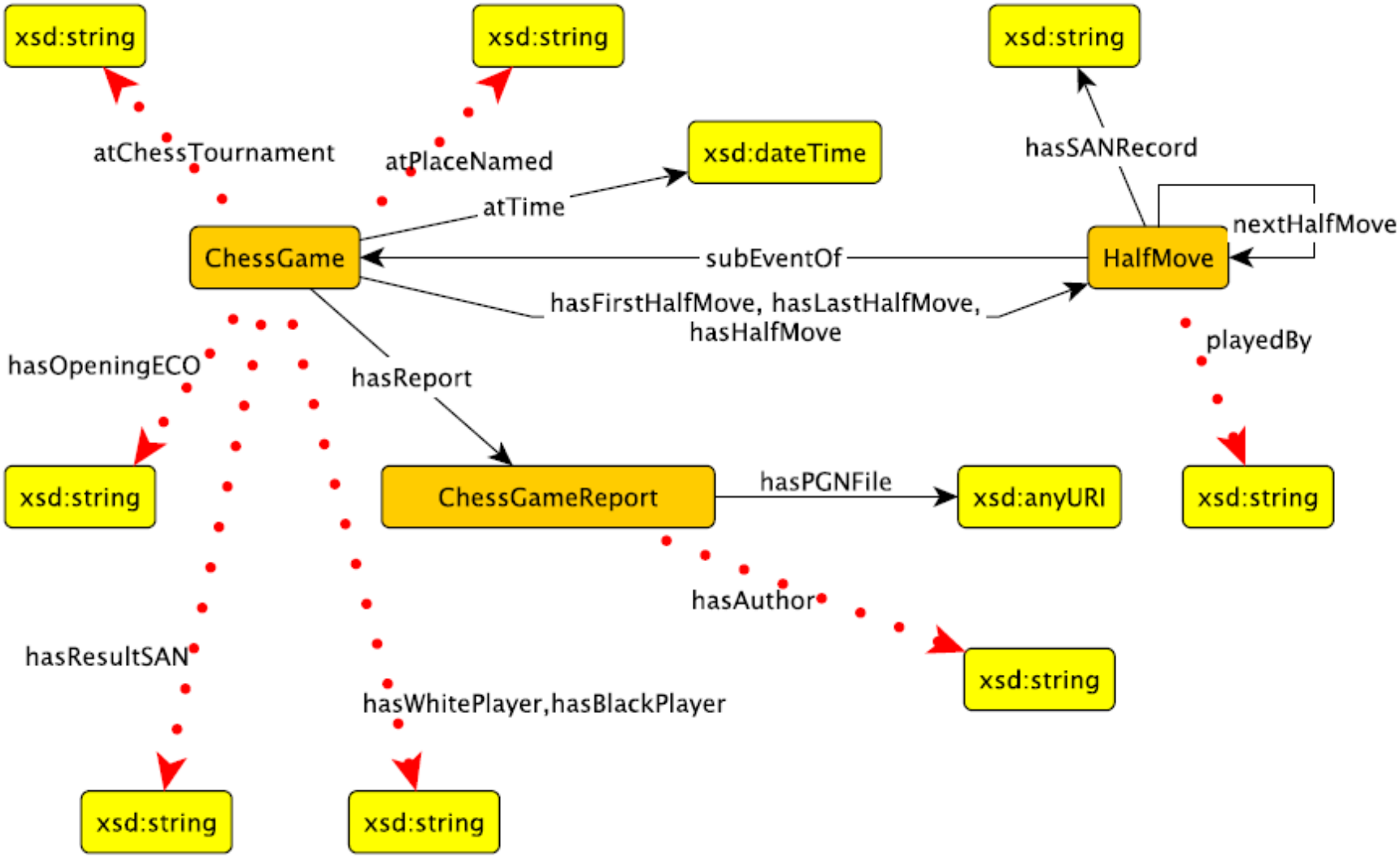
WRIGHT STATE UNIVERSITY

http://dase.cs.wright.edu/content/rowl

And see full paper here at ESWC2017


ChessTournament
xsd:string
hasName
ChessTournament
atPlace
Place
Place
hasName
xsd:string
atTime
atPlace
xsd:dateTime
subEventOf
atTime
Chess Opening
ChessOpening
hasOpening
hasECOCode
xsd:string
ChessGame
subEventOf
HalfMove
nextHalfMove
hasFirstHalfMove, hasLastHalfMove,
hasHalfMove
hasSANRecord
xsd:string
hasReport
Chess Game Report
ChessGameReport
providesAgentRole
hasPGNFile
xsd:anyURI
providesAgentRole
providesAgentRole
AgentRole
rdfs:subClassOf
AuthorRole
hasResult
Chess Game Result
ChessGameResult
encodedAsSAN
xsd:string
performedBy
BlackPlayerRole
WhitePlayerRole
ActingPlayerRole
rdfs:subClassOf
rdfs:subClassOf
rdfs:subClassOf
Agent
AgentRole
performedBy
Agent
performedBy
AgentRole
hasName
xsd:string

Simplified View

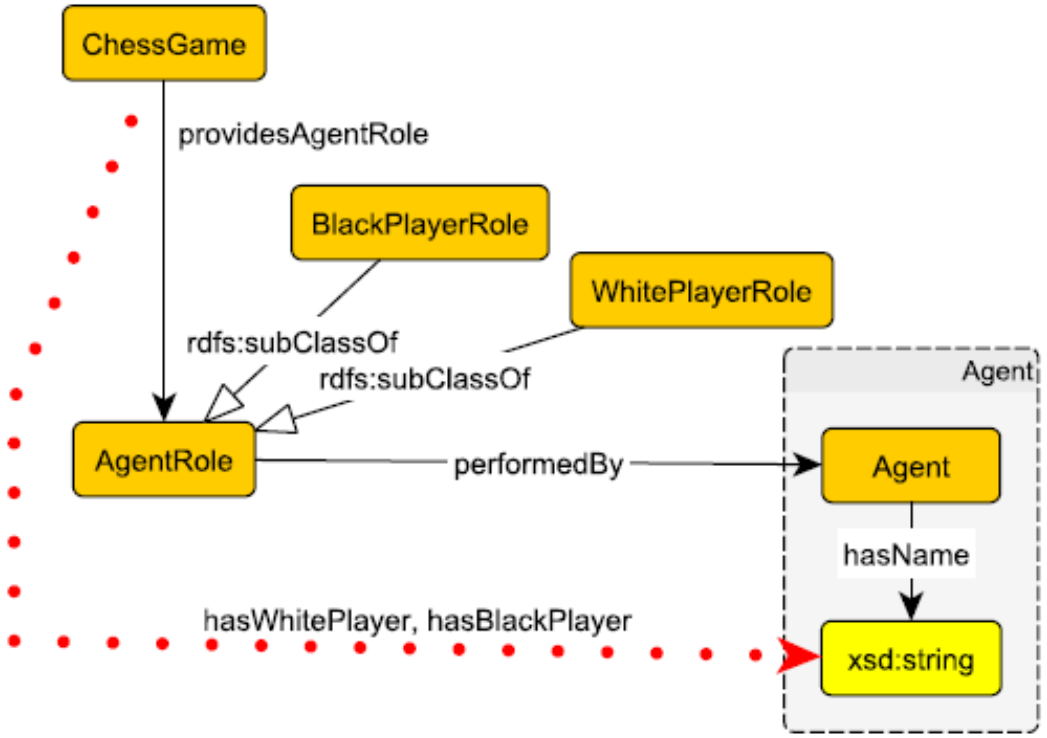


Mapping from Views

DaSe Lab

We used rules (axioms) to express the mapping from the ontology to the view.

The reverse direction is much more tricky.



$$\text{ClassA}(x) \wedge \text{ClassB}(y) \wedge C_1(x_1) \wedge \cdots \wedge C_n(x_n) \wedge R_1(y_1, y_2) \wedge \cdots \wedge R_k(y_k, y_{k+1}) \rightarrow \text{shortcut}(x, y).$$

$$\text{shortcut}(x, y) \rightarrow \text{ClassA}(x) \wedge \text{ClassB}(y) \wedge \exists x_1 \ldots \exists x_n \exists y_1 \ldots \exists y_n (C_1(x_1) \wedge \ldots \cdots \wedge C_n(x_n) \wedge R_1(y_1, y_2) \wedge \cdots \wedge R_k(y_k, y_{k+1}))$$

Mapping from views

Existential rules may be suitable in principle.

$$\text{shortcut}(x, y) \rightarrow \text{ClassA}(x) \wedge \text{ClassB}(y) \wedge \exists x_1 \ldots \exists x_n \exists y_1 \ldots \exists y_n (C_1(x_1) \wedge \ldots \\ \cdots \wedge C_n(x_n) \wedge R_1(y_1, y_2) \wedge \cdots \wedge R_k(y_k, y_{k+1}))$$

However automated reasoning with the potentially rather complex rule heads requires investigations, in particular if it is to be integrated with ontology reasoning.

A specific case are right-hand-side role chains:

$$R \sqsubseteq R_1 \circ \cdots \circ R_n,$$

Thanks!

References

Pascal Hitzler, Aldo Gangemi, Krzysztof Janowicz, Adila Krisnathi, Valentina Presutti (eds.), Ontology Engineering with Ontology Design Patterns: Foundations and Applications. Studies on the Semantic Web. IOS Press/AKA Verlag, 2016/2017.

Hitzler, Krötzsch, Rudolph, Foundations of Semantic Web Technologies, CRC/Chapman & Hall, 2010

Adila Krisnadhi, Ontology Pattern-Based Data Integration. Dissertation, Department of Computer Science and Engineering, Wright State University, 2015.

References

Víctor Rodríguez-Doncel, Adila A. Krisnadhi, Pascal Hitzler, Michelle Cheatham, Nazifa Karima, Reihaneh Amini, Pattern-Based Linked Data Publication: The Linked Chess Dataset Case. In: Olaf Hartig, Juan Sequeda, Aidan Hogan (eds.), Proceedings of the 6th International Workshop on Consuming Linked Data co-located with 14th International Semantic Web Conference (ISWC 2105), Bethlehem, Pennsylvania, US, October 12th, 2015. CEUR Workshop Proceedings 1426, CEUR-WS.org, 2015.

Adila A. Krisnadhi, Pascal Hitzler, A Core Pattern for Events. In: Proceedings WOP 2016 at ISWC 2016.

Adila A. Krisnadhi, Pascal Hitzler, The Stub Metapattern. In: Proceedings WOP 2016 at ISWC 2016.

References

Adila Krisnadhi, Pascal Hitzler, Modeling With Ontology Design Patterns: Chess Games As a Worked Example. In: Pascal Hitzler, Aldo Gangemi, Krzysztof Janowicz, Adila Krisnathi, Valentina Presutti (eds.), Ontology Engineering with Ontology Design Patterns: Foundations and Applications. Studies on the Semantic Web. IOS Press/AKA Verlag, 2016/2017.

Adila Krisnadhi, Nazifa Karima, Pascal Hitzler, Reihaneh Amini, Victor Rodriguez-Doncel, Krzysztof Janowicz, Ontology Design Patterns for Linked Data Publishing. In: Pascal Hitzler, Aldo Gangemi, Krzysztof Janowicz, Adila Krisnathi, Valentina Presutti (eds.), Ontology Engineering with Ontology Design Patterns: Foundations and Applications. Studies on the Semantic Web. IOS Press/AKA Verlag, 2016/2017.

OntoPedigree: Design patterns for event-based traceability in provenance-aware supply chains

Monika Solanki
https://w3id.org/people/msolanki
@nimonika
University of Oxford

Motivation

Visibility* in supply chains

Visibility is the ability to know exactly where things are at any point in time or where they have been and why.

---

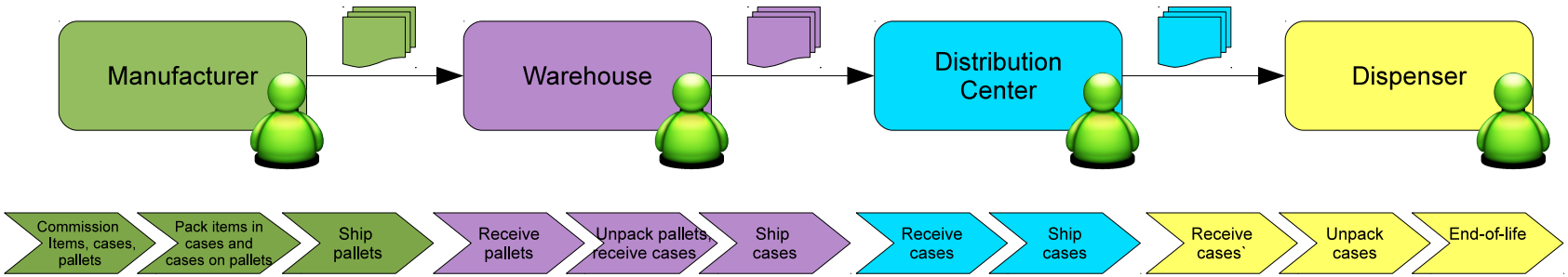
*http://www.gs1.org/docs/GS1_SupplyChainVisibility_WhitePaper.pdf

Enabiling Visibility

Data/Knowledge Sharing

Information and knowledge need to be interlinked, shared and made available consistently along the supply chain not least for regulatory reasons but also due to increasing consumer demands of being able to track and trace commodities.

Pharmaceutical supply chains



Flow of goods and flow of information (Abstraction)

Visibility in Pharmaceutical supply chains

Crucially Important!

Counterfeiting has increasingly become one of the major problems prevalent in these chains. The WHO estimates that between five and eight percent of the worldwide trade in pharmaceuticals is counterfeit.

Pharmaceutical supply chains

GS1 standards* for Visibility

- GS1: a neutral, not-for-profit organization dedicated to the design and implementation of global standards and solutions to improve the efficiency and visibility in supply chains.
- Core GS1 standards: EPCIS 1.1 & CBV 1.1
- GS1 US Secure Supply Chain Task Force: preliminary implementation guidelines* for applying GS1 Standards to U.S. Pharmaceutical supply chains for track and trace.

---

*http://www.gs1.org/healthcare/standards

*www.gs1us.org/RxGuideline

EPC, EPCIS and CBV

- The Electronic Product Code (EPC)*: provides products with unique, serialised identities.
- Electronic Product Code Information Services (EPCIS v1.1)*: provides a set of specifications for the syntactic capture and informal semantic interpretation of EPC based product information.
- CBV* supplements EPCIS by defining the structure of vocabularies and specific values for the vocabulary elements.
- Events as abstractions for traceability.

---

*http://en.wikipedia.org/wiki/Electronic_Product_Code

*http://www.gs1.org/epcis

SW & LD for Visibility in Supply chains

Research and Application

- Ontological representation of EPCIS events
- OBDA approach towards the transformation of EPCIS RDBMS
- Automated generation of provenance-based traceability artifacts from streaming EPCIS events.
- Identifying and classifying exceptions in events
- Validation of externally acquired traceability artifacts.

EPCIS(1.1) Events: An informal Intuition

One generic and four specific physical event types

For this talk,

- *EPCISEvent*: the generic EPCIS event.
- *ObjectEvent*: an event that occurred as a result of some action on one or more entities denoted by EPCs.
- *AggregationEvent*: an event that happened to one or more EPC-denoted entities that are physically aggregated.
- *TransactionEvent*: an event in which one or more entities denoted by EPCs become associated or disassociated with one or more identified business transactions.

Data model components

What(product(s)), Where(location), When(time), and Why(business step and disposition) of events (product movement) occurring in any supply chain.

- EPCs
- Time
- Read Points
- Business Location
- Business steps (commissioning, packing, shipping...)
- Disposition (in_transit, retail_sold, returned...)
- Action (ADD, OBSERVE, DELETE)

EEM*: The EPCIS Event Model

- Focuses on a tight conformance with the EPCIS 1.1 standard and Simplicity.
- Explicitly defines relationships with CBV entities through CBVVocab*.
- EEM has been mapped* to PROV-O*.

---

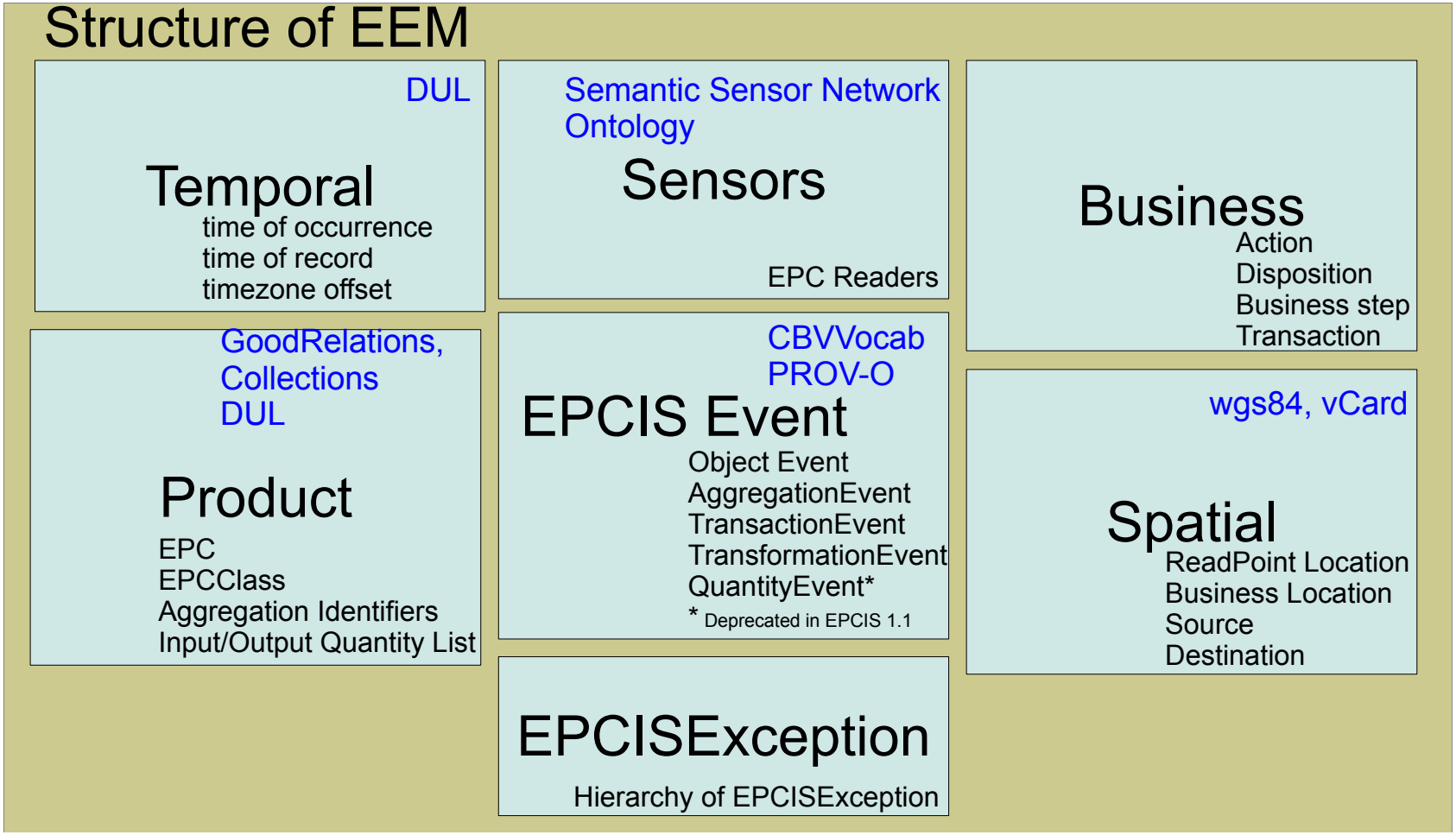
*http://purl.org/eem#

*www.w3.org/ns/prov-o

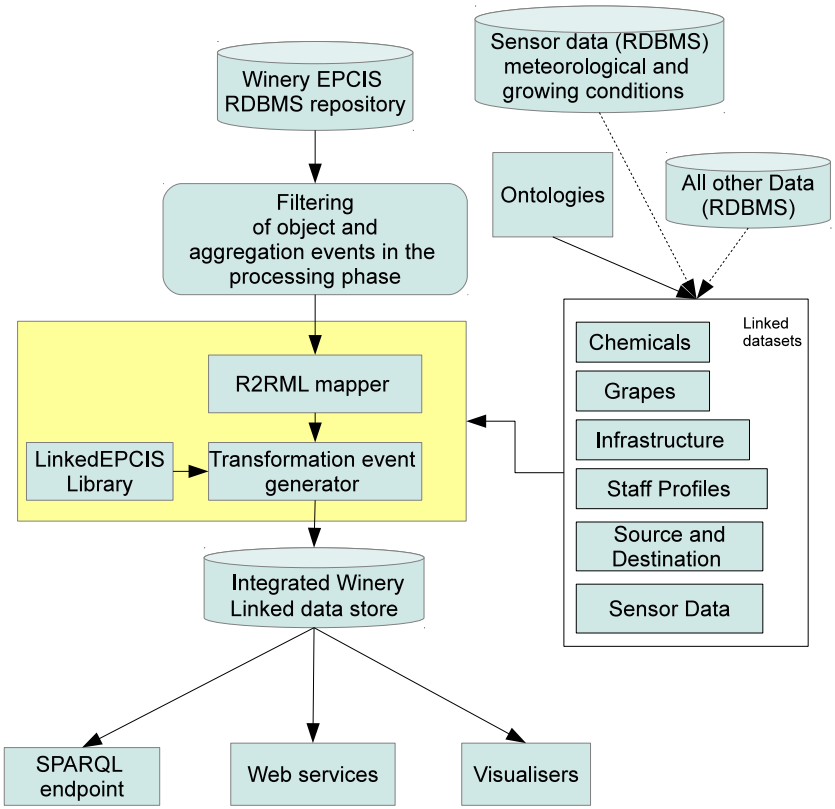
*http://purl.org/cbv#

*http://fispace.aston.ac.uk/ontologies/eem_prov.html

EEM Modules


Structure of EEM
DUL
Temporal
time of occurrence
time of record
timezone offset
Semantic Sensor Network Ontology
Sensors
EPC Readers
Business
Action
Disposition
Business step
Transaction
GoodRelations, Collections DUL
Product
EPC
EPCClass
Aggregation Identifiers
Input/Output Quantity List
CBVVocab
PROV-O
EPCIS Event
Object Event
AggregationEvent
TransactionEvent
TransformationEvent
QuantityEvent*
* Deprecated in EPCIS 1.1
wgs84, vCard
Spatial
ReadPoint Location
Business Location
Source
Destination
EPCISException
Hierarchy of EPCISException

The OBDA approach



M. Solanki et al. Modelling and Linking transformations in EPCIS governing supply chain business processes. EC-Web 2014.

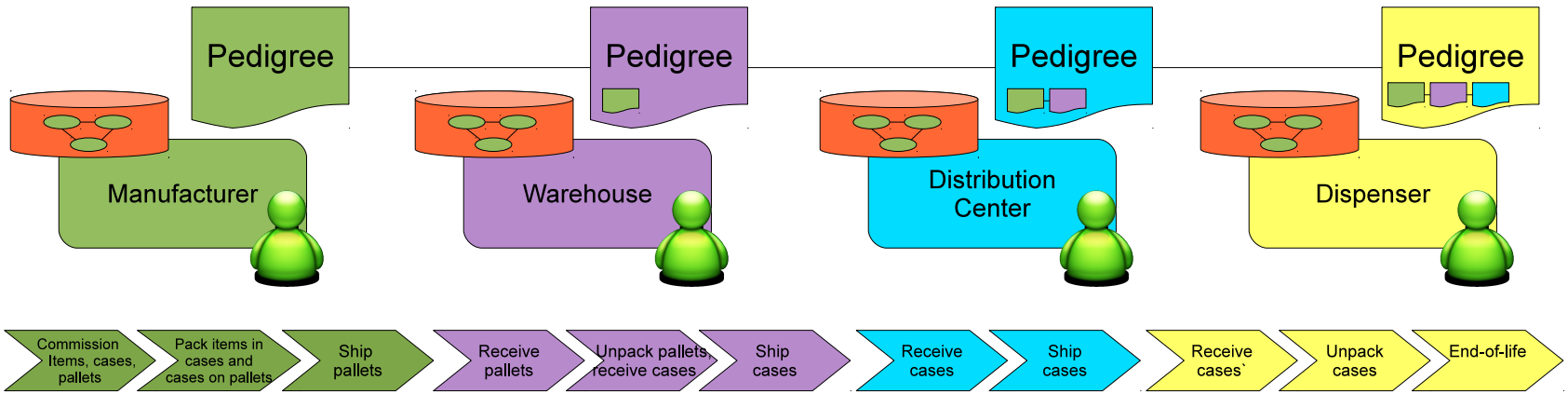
Pedigrees

- Most widely prevalent in the pharmaceutical industry.
- Pedigree (e-pedigree) is an audit trail that records the path and ownership of a drug as it moves through the supply chain.
- Each stakeholder involved in the manufacture or distribution of the drug adds information to the pedigree.
- "Event-based Linked Pedigrees": pedigrees based on a relevant subset of the captured EPCIS events.

---

cf. COLD, DeRiVE @ ISWC 2013

Pharmaceutical supply chains



Flow of linked pedigrees (Abstraction)

M. Solanki et al. EPCIS event-based traceability in pharmaceutical supply chains via automated generation of linked pedigrees. ISWC 2014.

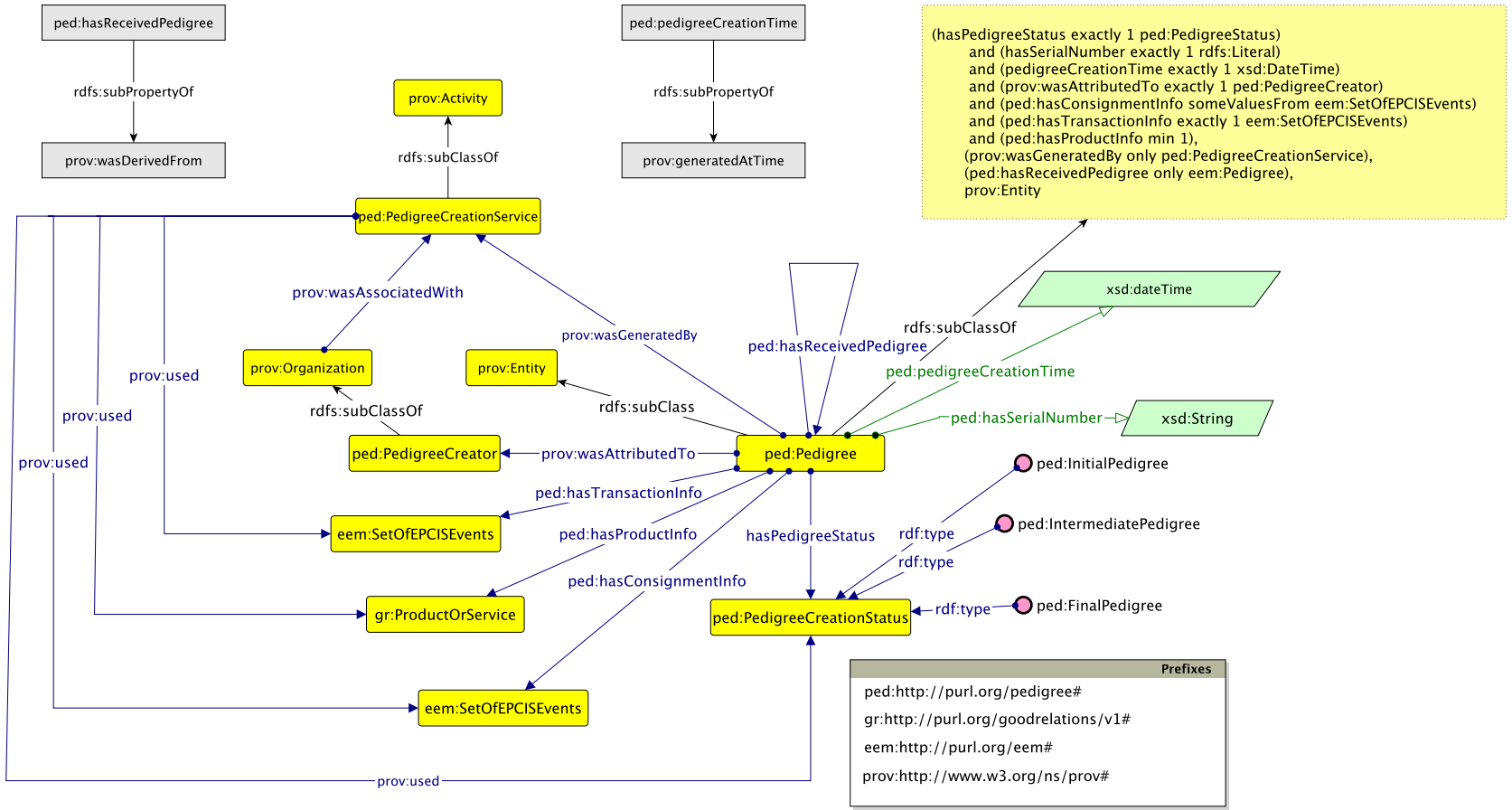
Requirements

- Certification and digital signatures
- Product information
- Location information
- Consignment information
- Transaction information
- Partner pedigree information

Competency questions

- Who is the creator of the pedigree ?
- What is the supply chain creation status of a given pedigree?
- Which are the business transactions recorded against a particular consignment?
- What are the events associated with pedigrees created between dates X and Y?
- Which products have been shipped together?
- Which other pedigrees are included in the received pedigree?

OntoPedigree: A CO design pattern



Linked Pedigree: Axiomatisation

```
Class: ped:Pedigree
       SubClassOf:
        (hasPedigreeStatus exactly 1 ped:PedigreeStatus)
         and (hasSerialNumber exactly 1 rdfs:Literal)
         and (pedigreeCreationTime exactly 1 xsd:DateTime)
         and (prov:wasAttributedTo exactly 1 ped:PedigreeCreator)
         and (ped:hasConsignmentInfo some eem:SetOfEPCISEvents)
         and (ped:hasTransactionInfo exactly 1 eem:SetOfEPCISEvents)
         and (ped:hasProductInfo min 1),
        (prov:wasGeneratedBy only ped:PedigreeCreationService),
        (ped:hasReceivedPedigree only eem:Pedigree),
        prov:Entity
```

Automated generation of Linked Pedigrees

- Streams of EPCIS events, where each EPCIS event is a named graph
- Algorithm to extract EPCIS events from streams using INSTANS, an incremental SPARQL query engine
- Counterfeit EPC detection as a side-effect of generating linked pedigrees

---

M. Solanki et al. EPCIS event-based traceability in pharmaceutical supply chains via automated generation of linked pedigrees. ISWC 2014.

Evaluation

EPCIS Event volumes

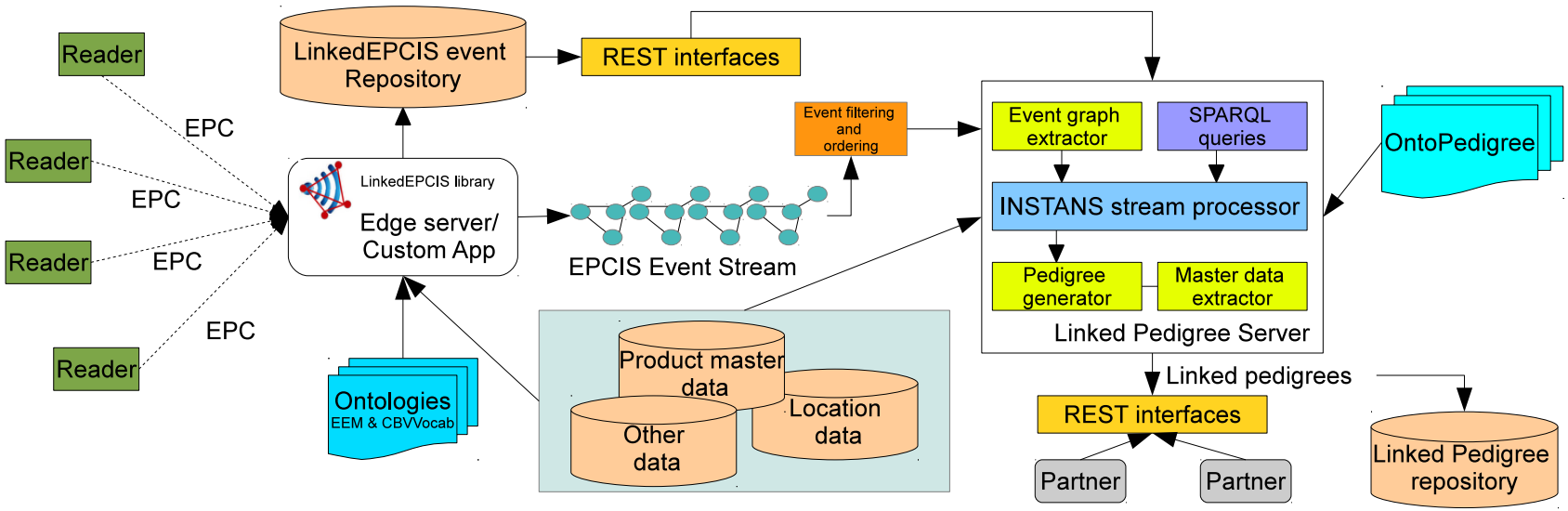
- Data Sources: Sample EPCIS relational data, Grey literature, interviews, surveys, EPCIS experts
- Assumption: an average rate of production as 6 days per week and 10 hours per day,
- Commissioning events generated based on the number of items ranging from 24,000 to 102,000 per day or approximately 40 to 170 per minute.
- Aggregation and shipping events generated considering aggregated items ranging from 100 to 500 (increments of 100) per case and number of cases per pallet ranging from 20 to 100 (increments of 20).
- Tumbling window sizes of 3, 5, 7 and 10 hours respectively.

Evaluation

			100-500 per case	20-100 per pallet
Window size (hrs)	Items/min. event stream velocity	Commissioned events	Aggregation events (increments of 100)	Shipping events for each of the aggregates (increments of 20)
10	120	72000	720/360/240/180/144	36/18/12/9/7
				18/9/6/5/4
				12/6/4/3/3
				18/9/6/5/4
				7/4/3/2/2
	170	102000	1020/510/340/255/204	51/26/17/13/11
				26/13/9/7/5
				17/9/7/5/4
				13/7/5/4/3
				10/5/4/3/2

Table 1. Number of commissioning, aggregation and shipping events for a window size of 10 hours and item commissioning rate of 120 and 170 per minute

Evaluation: Architecture and Implementation



EPCIS Exceptions

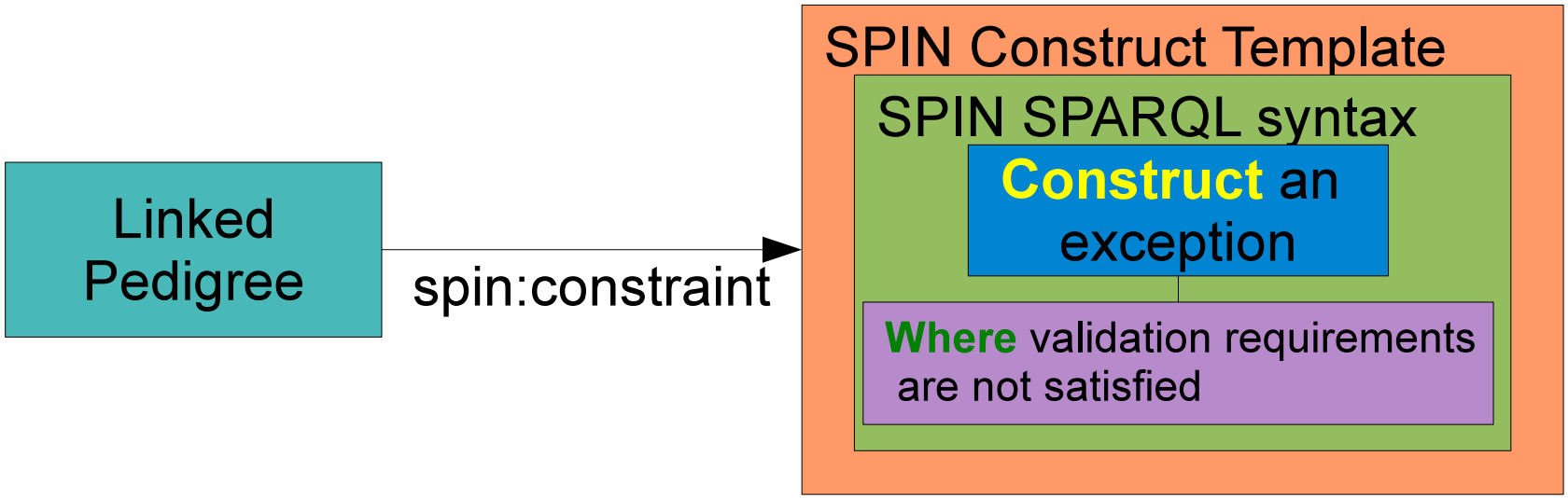
Typical examples

- (e1) Pedigree serial number discrepancy
- (e2) product inference problem - the inability to infer about products contained in an outer container without disaggregation using pedigree information
- (e3) quantity inference problem - the inability to derive the total quantity of items packed in an outer container without disaggregation using pedigree information
- (e4) missing or incorrect containment hierarchy between items and their containers - source of counterfeits.
- (e5) incomplete pedigree data
- (e6) pedigree data with broken chains, i.e., missing intermediate stakeholder pedigree information.

Validation requirements

- **Incomplete pedigree**: Mandatory information missing
- **Pedigree data has broken chain**: Pedigrees from other partners are missing
- **Pedigree based, receiving and shipping event correlation**: EPCs in receiving events do not match the EPCs in the shipping events.
- Temporal validity of shipping and receiving events
- Missing parent-child aggregation

Specifying validation rules



http://spinrdf.org/sp.html

http://www.topquadrant.com/spin/tutorial/

Incomplete pedigree

```
CONSTRUCT
{
 _:b0 a eem:PedigreeIncompleteException;
        spin:violationRoot ?this;
        eem:eventOccurredAt "timeLiteral"xsd:datetime;
        eem:associatedBusinessStep cbv:receiving;
        ....other triples about the exception
        rdfs:label ``Incomplete pedigree exception''.
}
```

M. Solanki et al. Detecting EPCIS Exceptions in linked traceability streams across supply chain business processes. SEMANTiCS 2014.

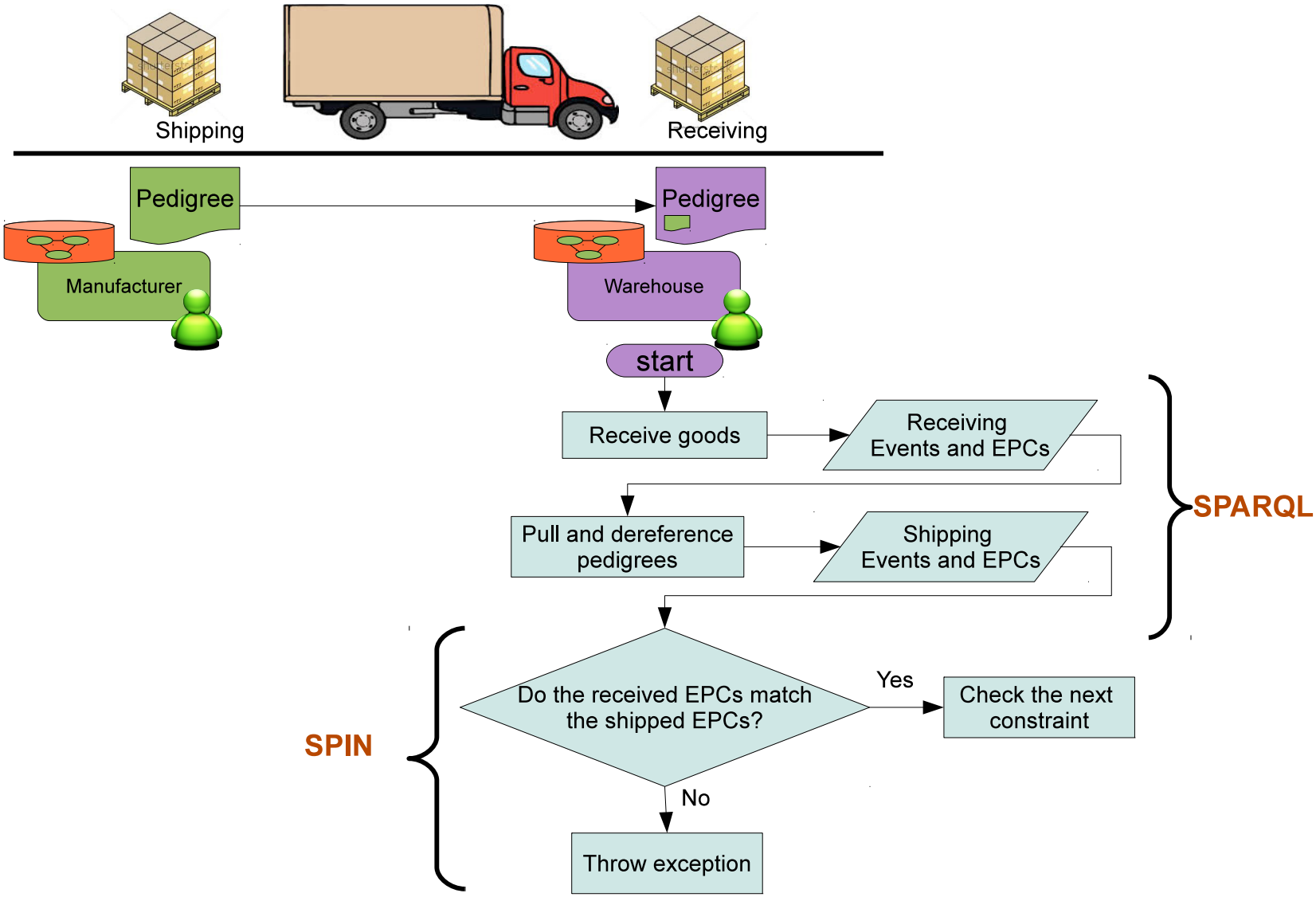
Incomplete pedigree

```
WHERE
{
...
}
FILTER NOT EXISTS{ ped:hasPedigreeStatus ?PedigreeStatus;
                   ped:hasSerialNumber ?serialNumber;
                   ped:pedigreeCreationTime ?pedTime;
                   prov:wasAttributedTo  ?pedigreeCreator;
                   ped:hasConsignmentInfo ?setOfConsEvents;
                   ped:hasTransactionInfo SetOfShipEvents;
                   ped:hasProductInfo productInfo.}
```

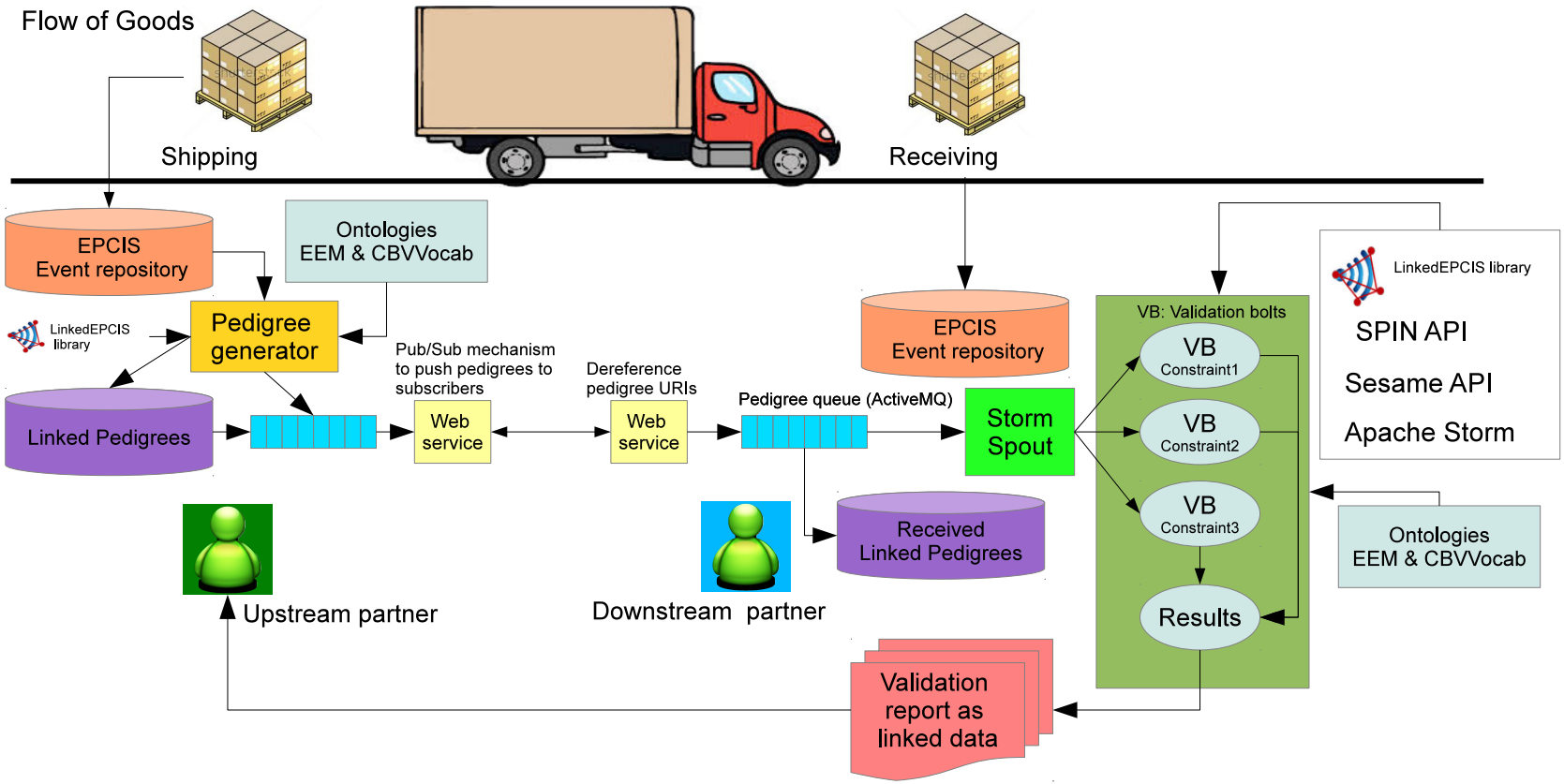
Pedigree data has broken chain

```
CONSTRUCT
{
 _:b0 a eem:BrokenPedigreeChainException;
    ..same as the CONSTRUCT above..
    rdfs:label ``Broken pedigree chain exception''
}
WHERE
{   ?this a ped:Pedigree;
    ped:hasPedigreeStatus ped:IntermediatePedigree;
    ped:hasReceivedPedigree+ ?pedigree.
 FILTER NOT EXISTS {
         ped:hasPedigreeStatus ped:IntermediatePedigree;
         ped:hasReceivedPedigree+ ?pedigree.}
}
```

Receiving and shipping event correlation



Implementation



Summary

- Semantic Web standards, ontologies and linked data can be utilised to record and represent real time supply chain knowledge
- Complex Event Processing over continuous streams of semantically interlinked EPCIS event datasets enable automated generation of linked pedigrees, detection of exceptions and validation of integrity constraints.
- Rule based frameworks can be integrated with distributed realtime computation systems such as Apache Storm to process real time streams of supply chain data.
- The proposed approach is domain independent and can be widely applied to most scenarios of traceability as long as there is conformance to EPCIS 1.1 in the supply chain.

Further information

- M. Solanki and C. Brewster. OntoPedigree: A content ontology design pattern for traceability knowledge representation in supply chains. Semantic Web Journal, 2015
- M. Solanki and C. Brewster. Enhancing visibility in EPCIS governing Agri-food Supply Chains via Linked Pedigrees. To appear, International Journal on Semantic Web and Information Systems Volume 10, Issue 3, 2015
- M. Solanki and C. Brewster. EPCIS event-based traceability in pharmaceutical supply chains via automated generation of linked pedigrees. ISWC 2014. Springer-Verlag.
- M. Solanki and C. Brewster. A Knowledge Driven Approach towards the Validation of Externally Acquired Traceability Datasets in Supply Chain Business Processes. EKAW 2014. Springer-Verlag.
- M. Solanki and C. Brewster. Modelling and Linking transformations in EPCIS governing supply chain business processes. EC-Web 2014. Springer-LNBIP.
- M. Solanki and C. Brewster. Detecting EPCIS Exceptions in linked traceability streams across supply chain business processes. SEMANTiCS 2014. ACM-ICPS.
- M. Solanki and C. Brewster. Consuming Linked data in Supply Chains: Enabling data visibility via Linked Pedigrees. COLD2013 at ISWC, volume Vol-1034. CEUR-WS.org proceedings, 2013.
- M. Solanki and C. Brewster. Representing Supply Chain Events on the Web of Data. DeRiVE at ISWC. CEUR-WS.org proceedings, 2013.
- `http://windermere.aston.ac.uk/~monika/ontologies.html`
- `http://windermere.aston.ac.uk/~monika/publication.html`

Modular Ontology Design and Use Case: The GeoLink Example

Pascal Hitzler
Data Semantics Laboratory (DaSe Lab)
Data Science and Security Cluster (DSSC)
Wright State University
http://www.pascal-hitzler.de

This Tutorial (all parts)

- **Pascal Hitzler (60 mins):**
  Introduction and first examples
- **Monika Solanki (30 mins):**
  Example "modeling vaccine traceability"

coffee

- **Pascal Hitzler (60 mins):**
  Example "GeoLink Modular Ontology"
- **Agnieszka Lawrynowicz (30 mins):**
  Example "Reporting Event ODP"

lunch

- **Karl Hammar with all others (3h):**
  Hands-on, the WebProtege XDP plug-in

EarthCube GeoLink Scenario

The NSF EarthCube Program:

Developing a Community-Driven Data and Knowledge Environment for the Geosciences

“concepts and approaches to create integrated data management infrastructures across the Geosciences.”

“EarthCube aims to create a well-connected and facile environment to share data and knowledge in an open, transparent, and inclusive manner, thus accelerating our ability to understand and predict the Earth system.”

EarthCube GeoLink Scenario

DaSe Lab

SSC

GeoLink: An EarthCube "Building Block" project (2014-2017)

How to realize data search across many large-scale geoscience data repositories, such that

- **The approach is extendable to new repositories.**
- **The scope can extend across all of the Geosciences.**
- **The search capabilities can be made more fine-grained in the future if desired.**

Central idea: Use a modular, extendable ontology for the integration of metadata.

GeoLink

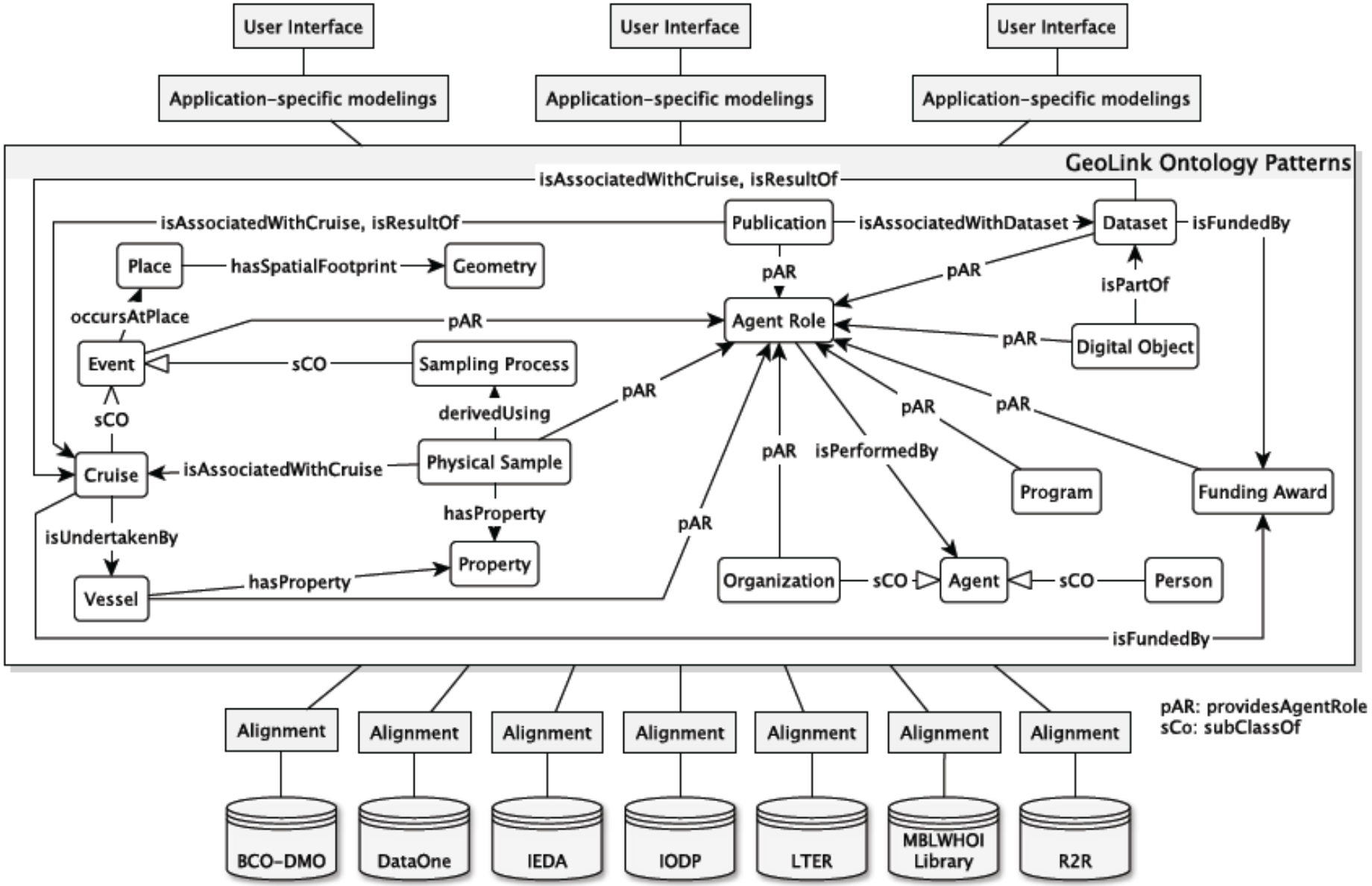
An interactive demonstration of the integrated GeoLink data is available at

http://demo.geolink.org

At http://www.geolink.org/ there are links to the complete schema, a SPARQL Endpoint, publications, etc.

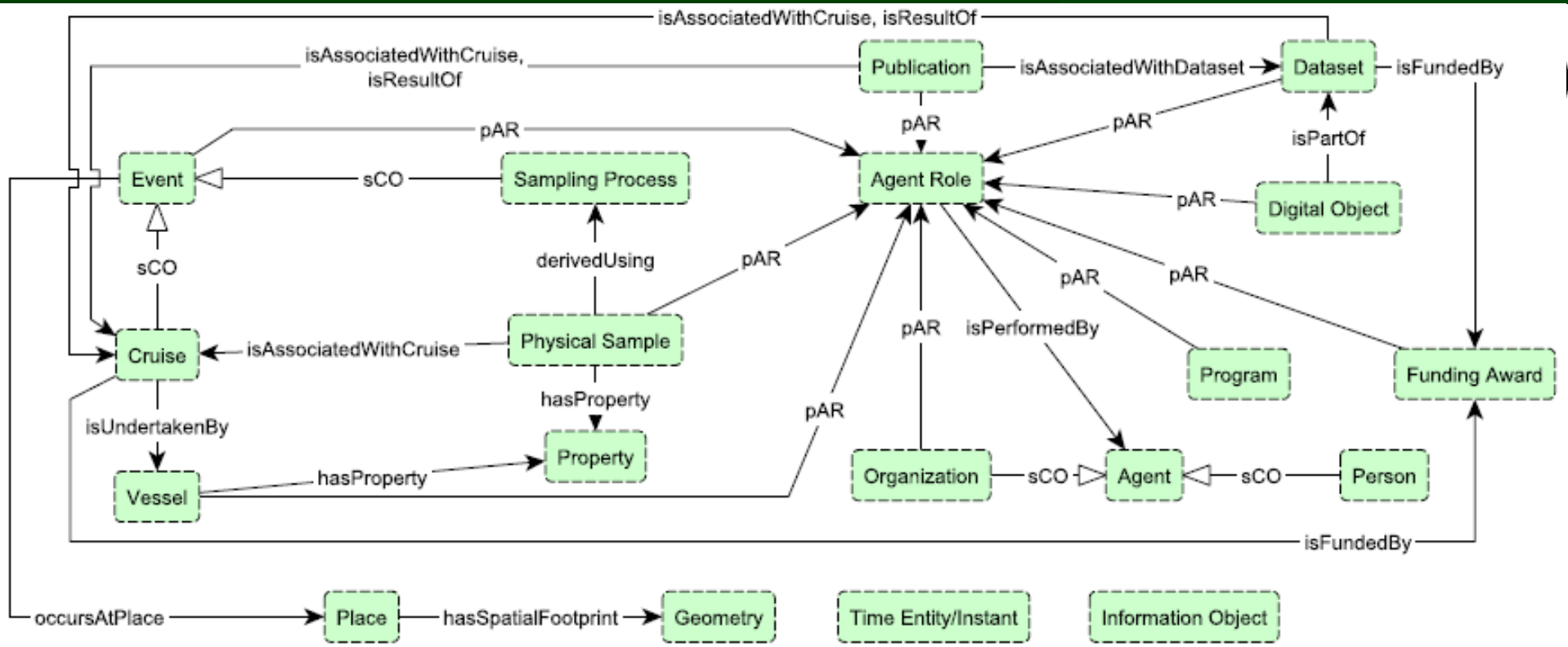
The GeoLink Framework

DaSe Lab



The GeoLink Modular Ontology

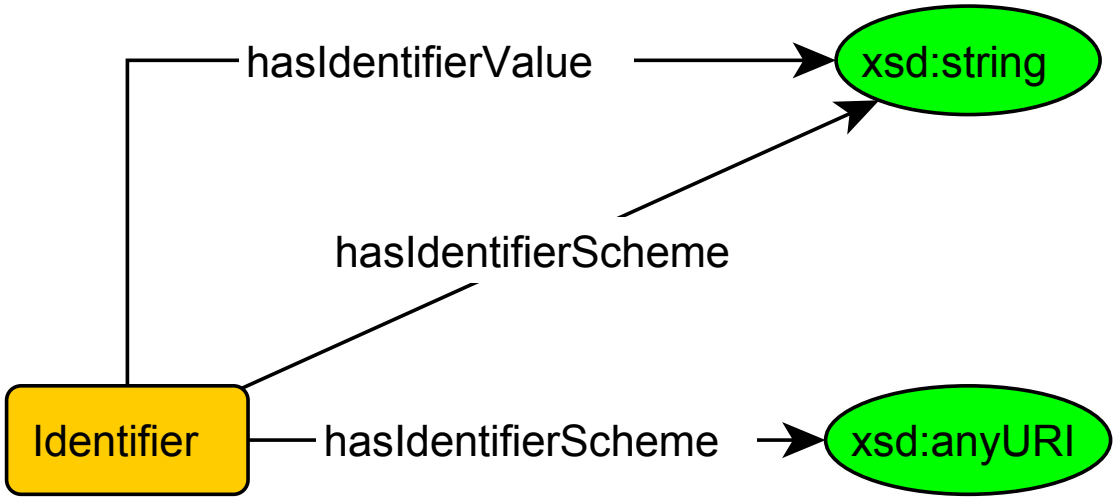
DaSe Lab



High-level overview of the GeoLink Modular Ontology (GMO).

Each box stands for a module, which has been modeled in its own right.

Identifier pattern



$$Identifier \sqsubseteq (\leqslant 1\ hasIdentifierScheme.(xsd:anyURI \sqcup xsd:string))$$

$$Identifier \sqsubseteq (=1\ hasIdentifierValue.xsd:string)$$

$$\exists hasIdentifierScheme.(xsd:anyURI \sqcup xsd:string) \sqsubseteq Identifier$$

$$\exists hasIdentifierValue.xsd:string \sqsubseteq Identifier$$

$$Identifier \sqsubseteq \forall hasIdentifierScheme.(xsd:anyURI \sqcup xsd:string)$$

$$Identifier \sqsubseteq \forall hasIdentifierValue.xsd:string$$

OWLAx

DaSe Lab

SSC

Ontology Axiomatization Support (OWLAx)

- **Protégé Plug-In**
- **Md. Kamruzzaman Sarker, Adila A. Krisnadhi, Pascal Hitzler, OWLAx: A Protege Plugin to Support Ontology Axiomatization through Diagramming. Proceedings Posters and Demos Track at ISWC 2016.**

- **Insert class diagram using graphical UI**
- **System asks you whether to include corresponding axioms (taken from a pool of most common axioms for the diagram)**
- **You can of course also manually add further axioms.**

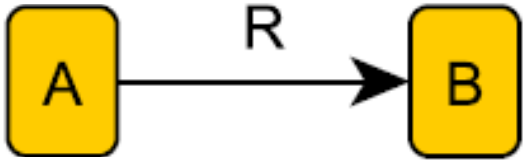
http://dase.cs.wright.edu/content/ontology-axiomatization-support

Axioms – Systematically

DaSe Lab

SSC

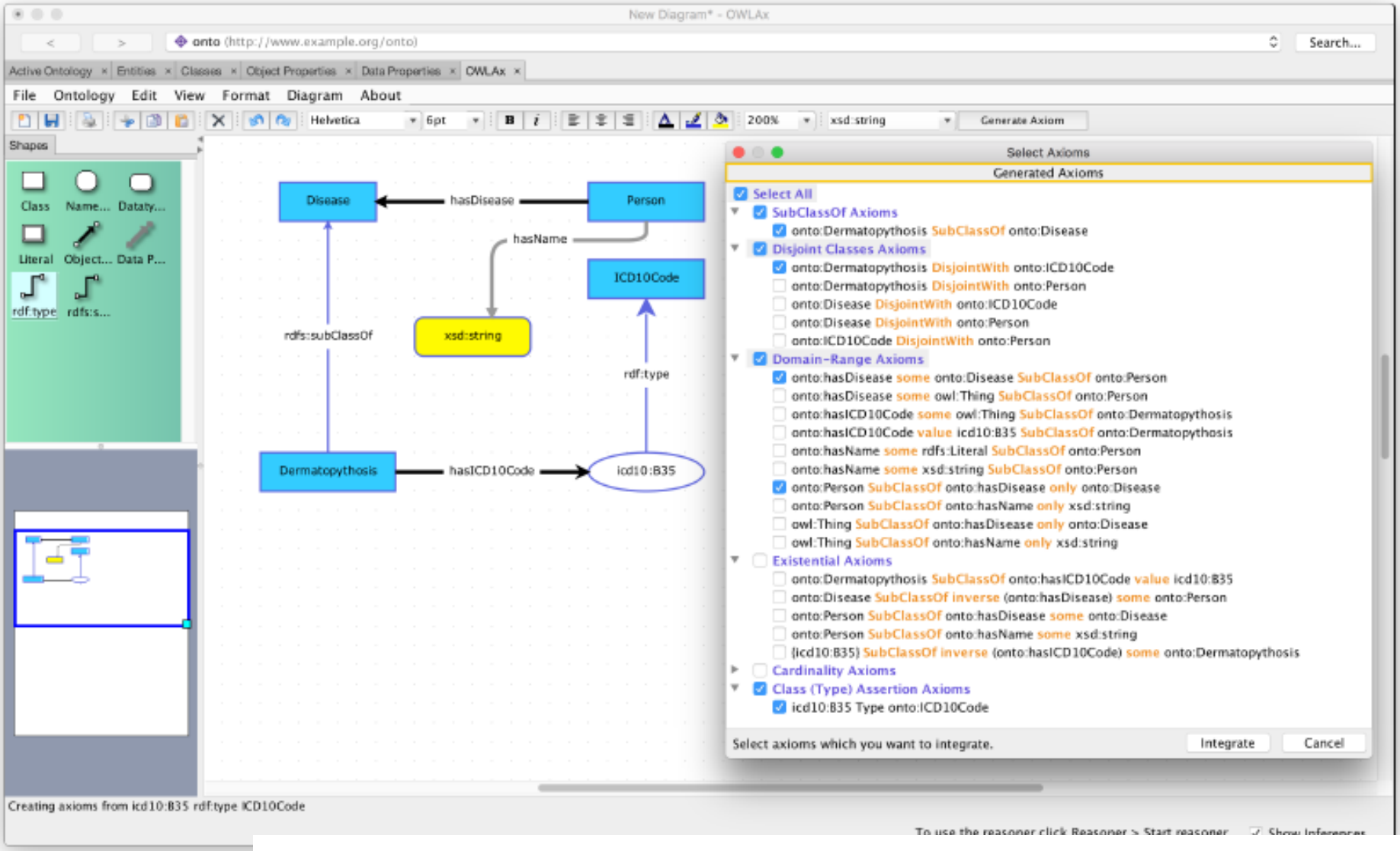
1. $A \sqcap B \sqsubseteq \bot$
2. $\exists R.\top \sqsubseteq A$
3. $\exists R.B \sqsubseteq A$
4. $\top \sqsubseteq \forall R.B$
5. $A \sqsubseteq \forall R.B$
6. $A \sqsubseteq R.B$
7. $B \sqsubseteq R^{-}.A$
8. $\top \sqsubseteq \leq 1 R.\top$
9. $\top \sqsubseteq \leq 1 R.B$
10. $A \sqsubseteq \leq 1 R.\top$
11. $A \sqsubseteq \leq 1 R.B$
12. $\top \sqsubseteq \leq 1 R^{-}.\top$
13. $\top \sqsubseteq \leq 1 R^{-}.A$
14. $B \sqsubseteq \leq 1 R^{-}.\top$
15. $B \sqsubseteq \leq 1 R^{-}.A$



1. A DisjointWith B (disjointness)
2. R some `owl:Thing` SubClassOf A (domain)
3. R some B SubClassOf A (scoped domain)
4. `owl:Thing` SubClassOf R only B (range)
5. A SubClassOf R only B (scoped range)
6. A SubClassOf R some B (existential)
7. B SubClassOf inverse R some A (inverse existential)
8. `owl:Thing` SubClassOf R max 1 `owl:Thing` (functionality)
9. `owl:Thing` SubClassOf R max 1 B (qualified functionality)
10. A SubClassOf R max 1 `owl:Thing` (scoped functionality)
11. A SubClassOf R max 1 B (qualified scoped functionality)
12. `owl:Thing` SubClassOf inverse R max 1 `owl:Thing` (inverse functionality)
13. `owl:Thing` SubClassOf inverse R max 1 A (inverse qualified functionality)
14. B SubClassOf inverse R max 1 `owl:Thing` (inverse scoped functionality)
15. B SubClassOf inverse R max 1 A (inverse qualified scoped functionality)

OWLAx Protégé plug-in

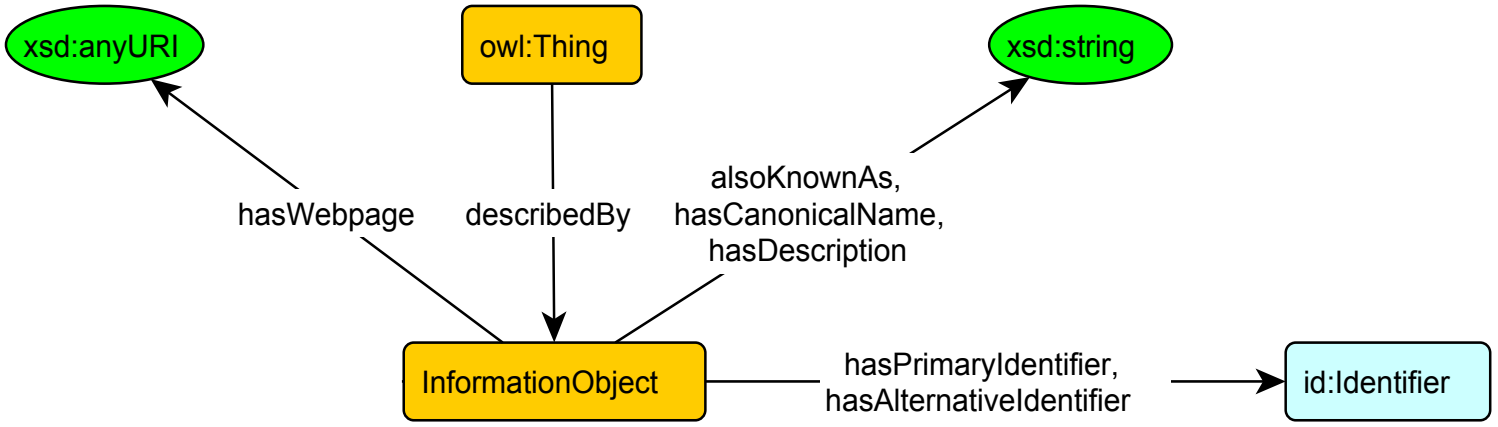
DaSe Lab



WRIGHT STATE UNIVERSITY

In: Proc. ISWC 2016 poster & demos
http://dase.cs.wright.edu/content/ontology-axiomatization-support

Information Object pattern



$$\top \sqsubseteq (\leqslant 1\ describedBy.InformationObject)$$

$$InformationObject \sqsubseteq (=1\ describedBy^{-}.\top)$$

$$InformationObject \sqsubseteq \neg\exists describedBy.InformationObject$$

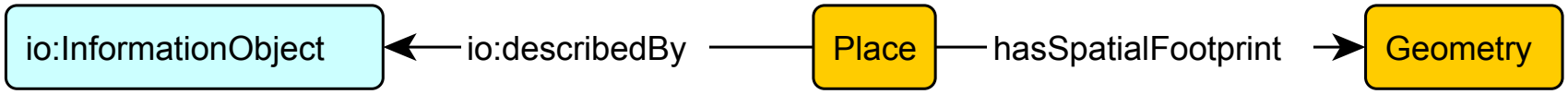
$$\exists hasWebpage.xsd{:}anyURI \sqsubseteq InformationObject$$

$$\exists alsoKnownAs.xsd{:}string \sqsubseteq InformationObject$$

$$\exists hasCanonicalName.xsd{:}string \sqsubseteq InformationObject$$

$$\exists hasDescription.xsd{:}string \sqsubseteq InformationObject$$

Place



Alignment to external ontologies or vocabularies,
rather than direct reuse:

GeoSPARQL, `http://www.opengis.net/ont/geosparql`

```
PREFIX geo:  <http://www.opengis.net/ont/geosparql#>
```

Good axioms

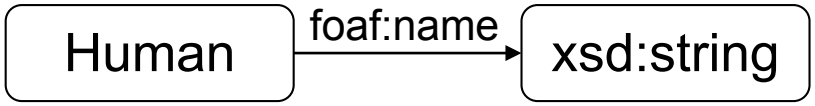
Specificity matters: Problems with domain/range.

Recommendations often heard (but are problematic):

- **Indicate domain and range for your properties.**
- **Reuse as many existing vocabularies as you can.**

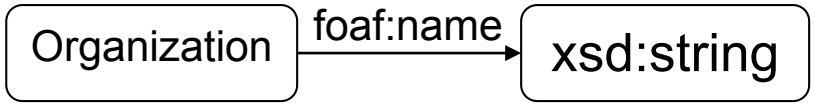
But there are problems with this:

Ontology 1:



domain(foaf:name) = Human

Ontology 2:



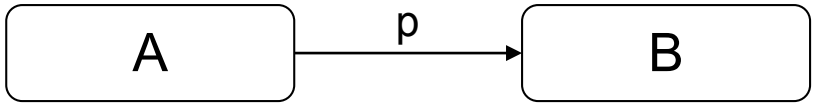
domain(foaf:name) = Organization

Logical consequence after merge: Human ≡ Organization

Recommendations

- Make rich axiomatizations
- Avoid re-use of external vocabularies
  (rather provide an additional file with mappings for those who want to use it)
- Avoid naïve domain and range axioms.

Alternative to naïve domain/range: scoped domain and range.

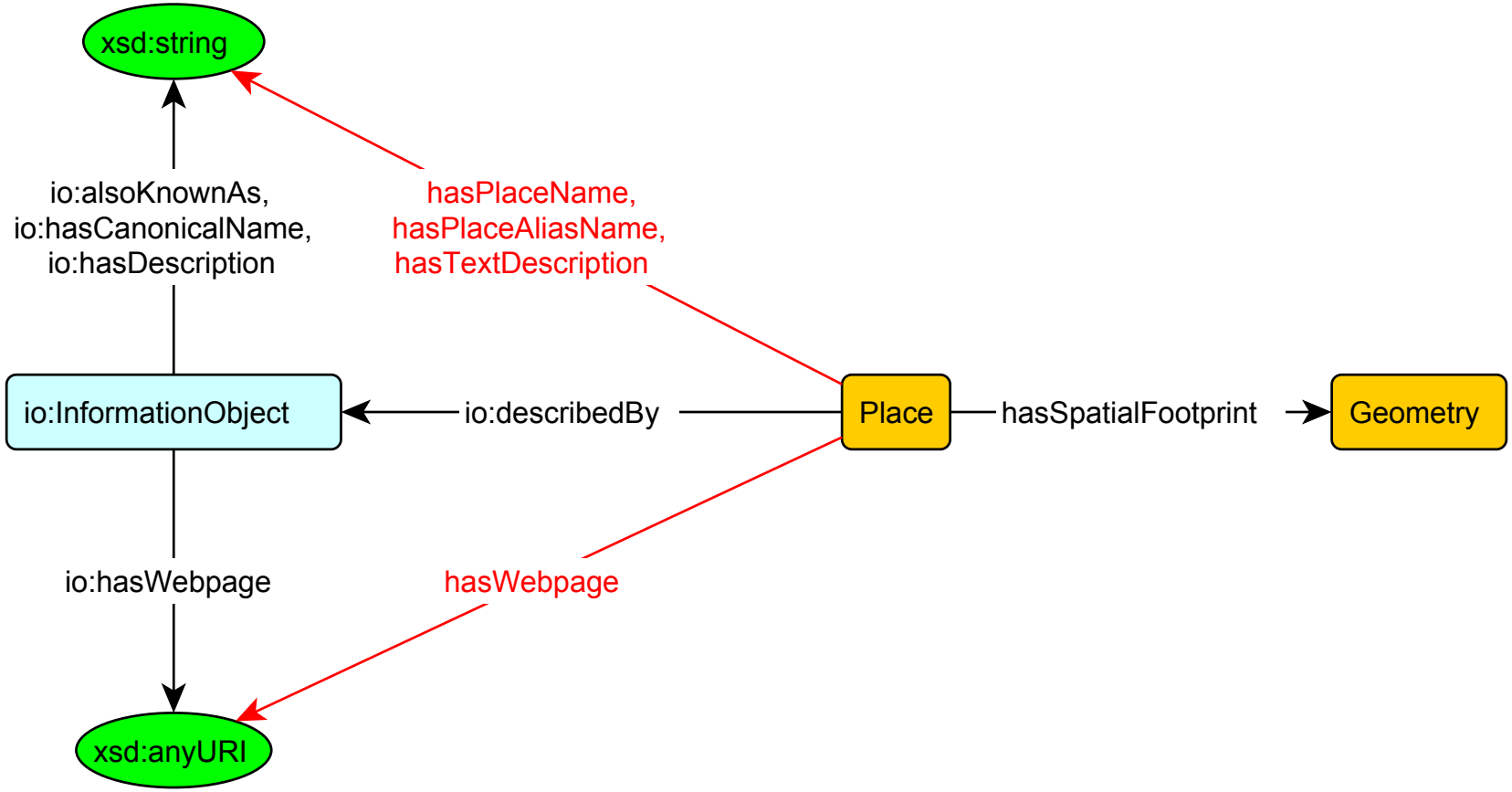


$$A(x) \wedge p(x,y) \rightarrow B(y)$$ **scoped range**

$$B(y) \wedge p(y,x) \rightarrow A(x)$$ **scoped domain**

both rules can be expressed in OWL.

Place shortcuts



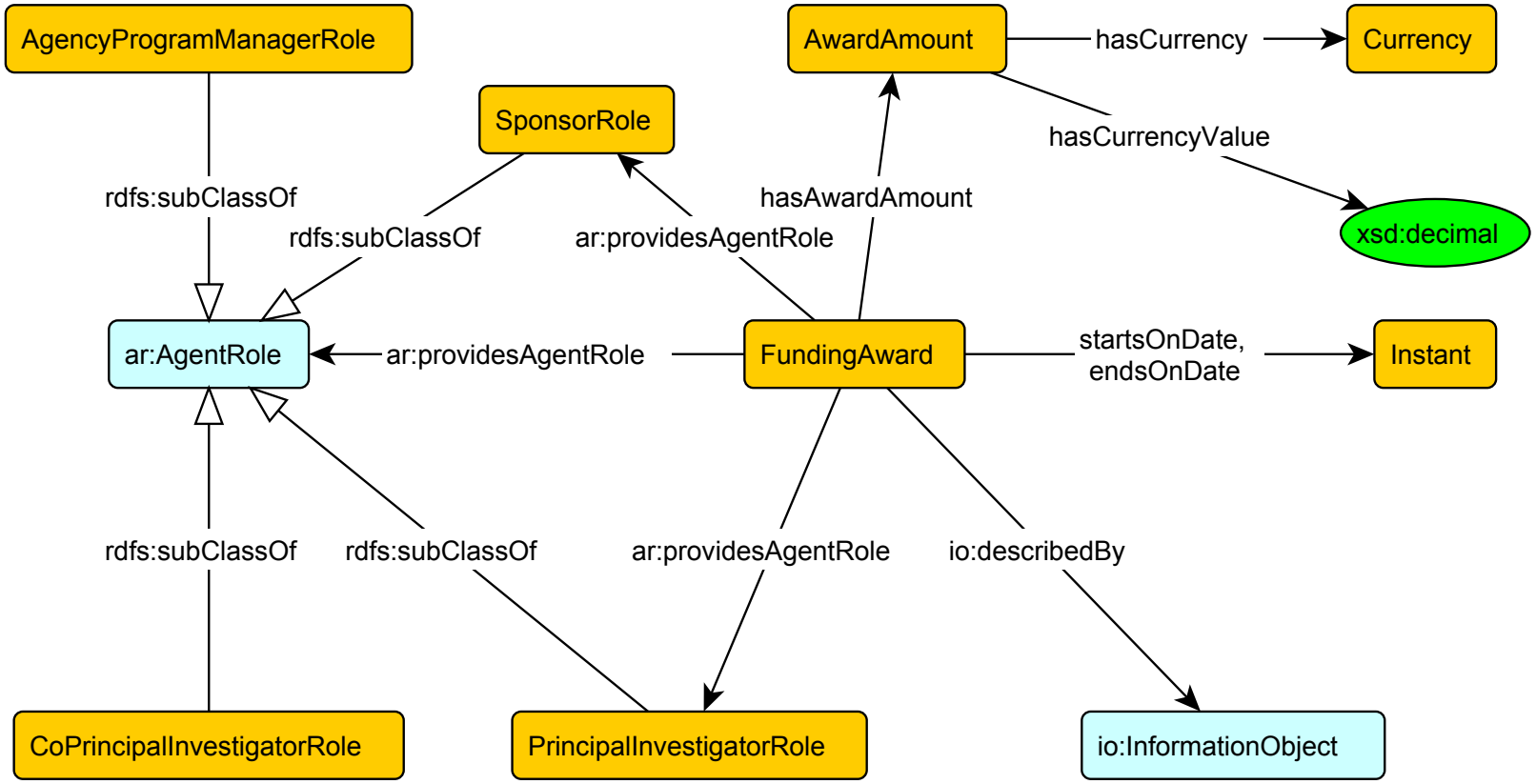
$$Place(x) \wedge io{:}describedBy(x, y) \wedge io{:}hasCanonicalName(y, z) \rightarrow hasPlaceName(x, z)$$

$$Place(x) \wedge io{:}describedBy(x, y) \wedge io{:}alsoKnownAs(y, z) \rightarrow hasPlaceAliasName(x, z)$$

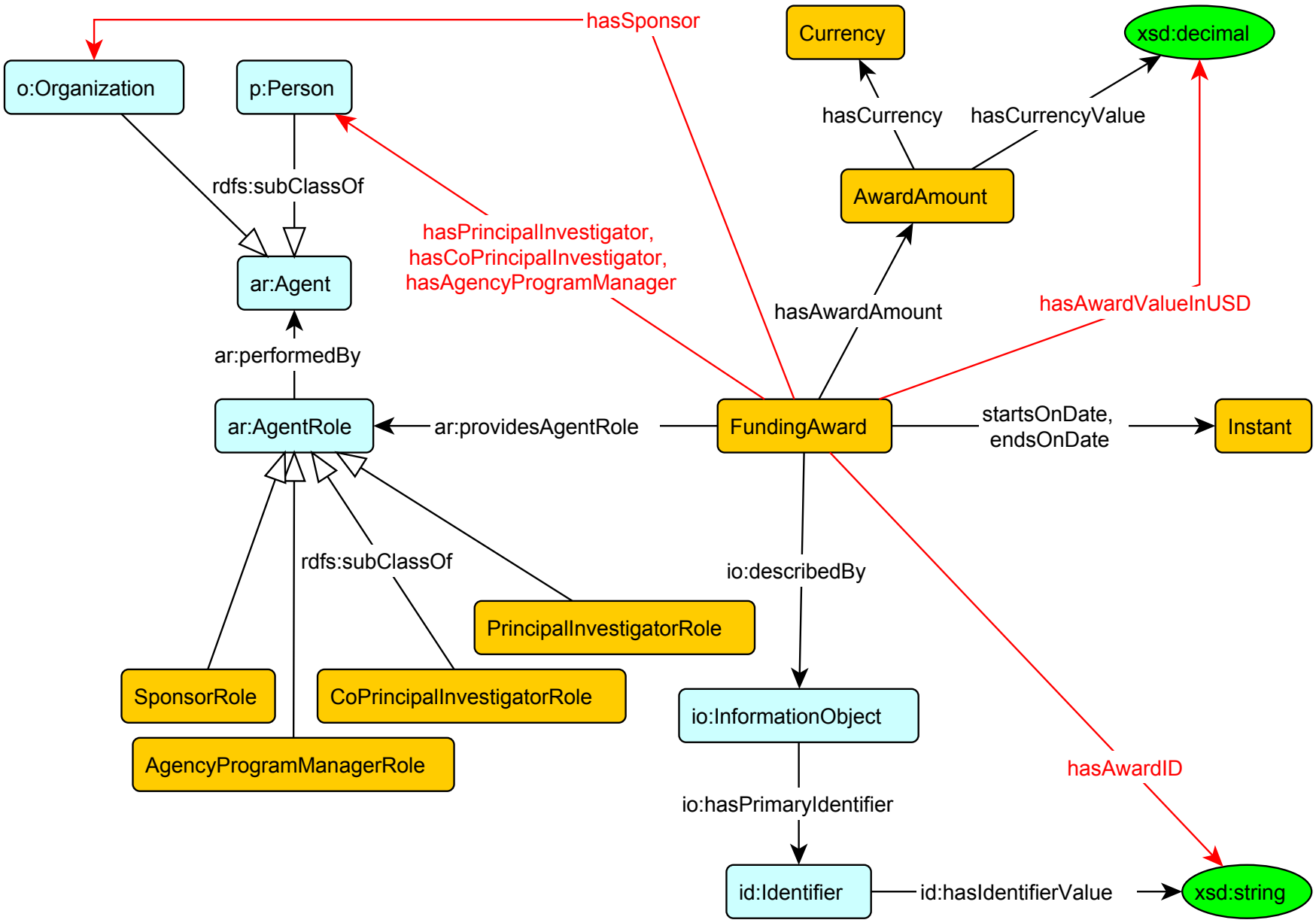
$$Place(x) \wedge io{:}describedBy(x, y) \wedge io{:}hasDescription(y, z) \rightarrow hasTextDescription(x, z)$$

$$Place(x) \wedge io{:}describedBy(x, y) \wedge io{:}hasWebpage(y, z) \rightarrow hasWebpage(x, z)$$

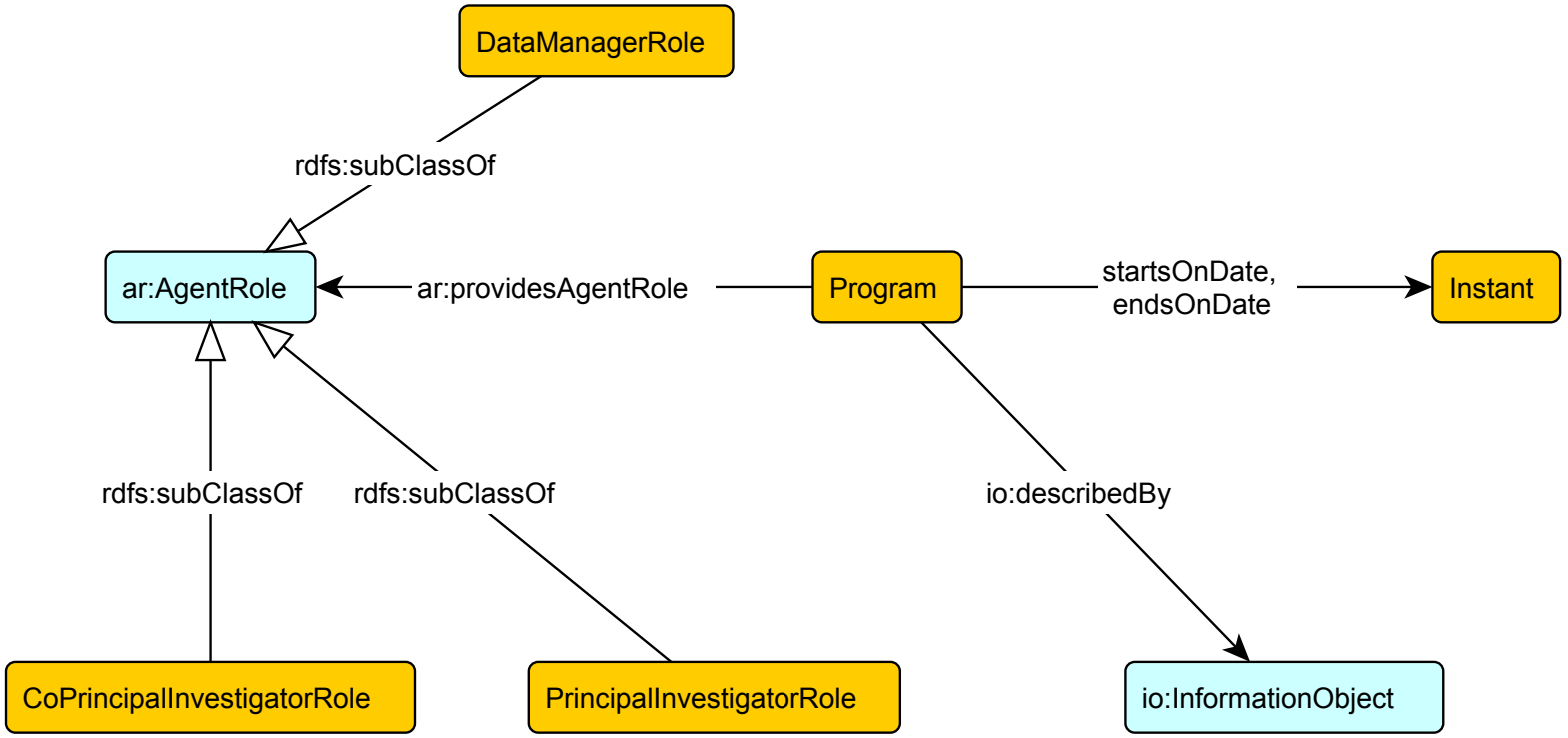
Funding Award pattern



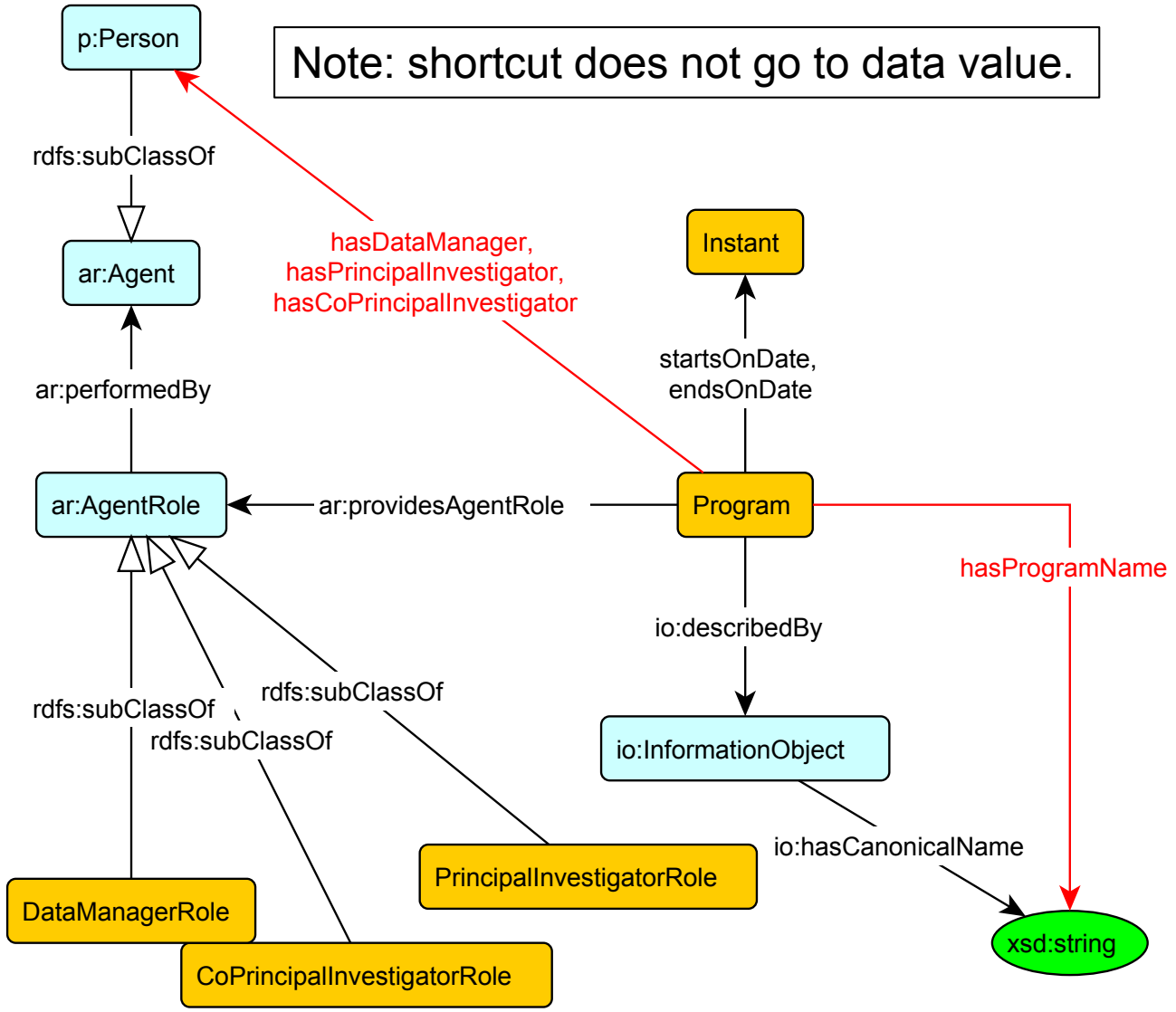
Funding Award shortcuts



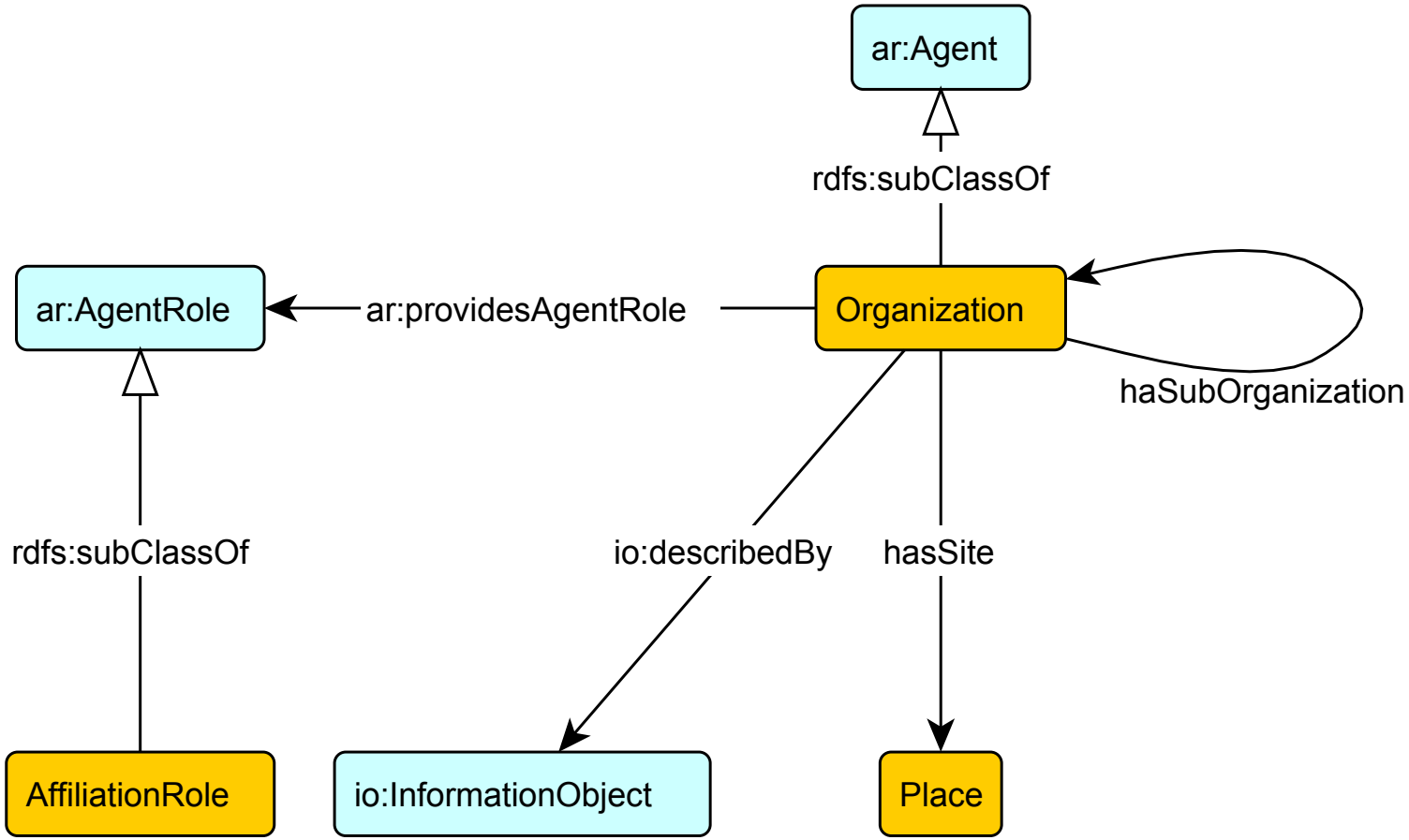
Program


DataManagerRole
rdfs:subClassOf
ar:AgentRole
ar:providesAgentRole
Program
startsOnDate,
endsOnDate
Instant
rdfs:subClassOf
rdfs:subClassOf
io:describedBy
CoPrincipalInvestigatorRole
PrincipalInvestigatorRole
io:InformationObject

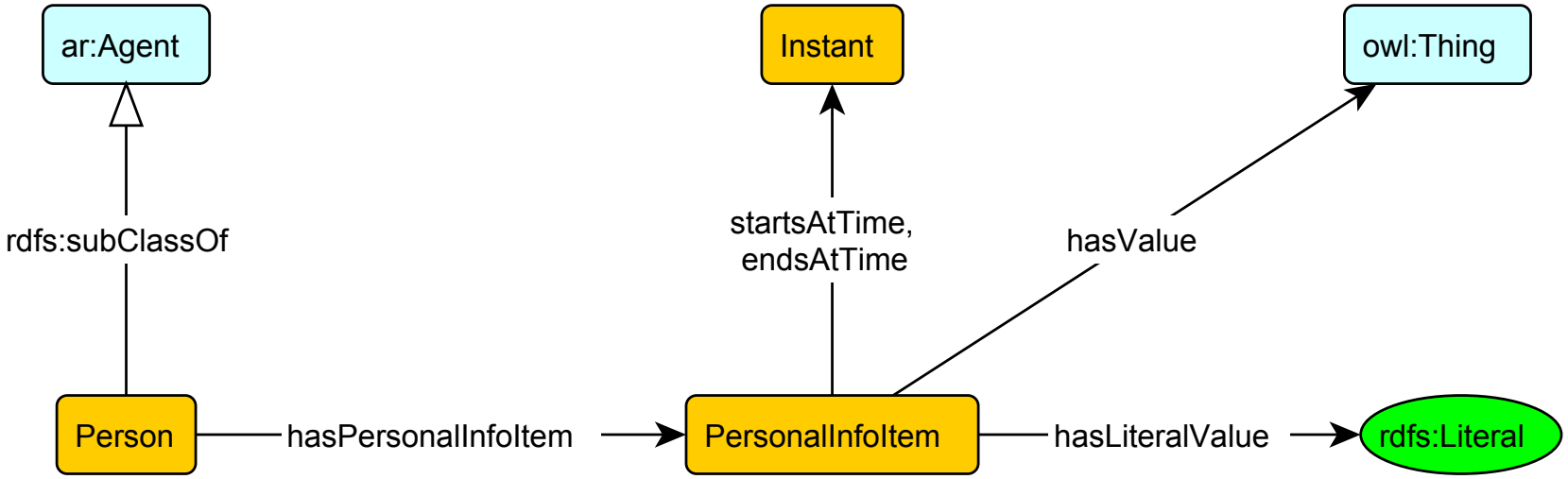
Program shortcuts



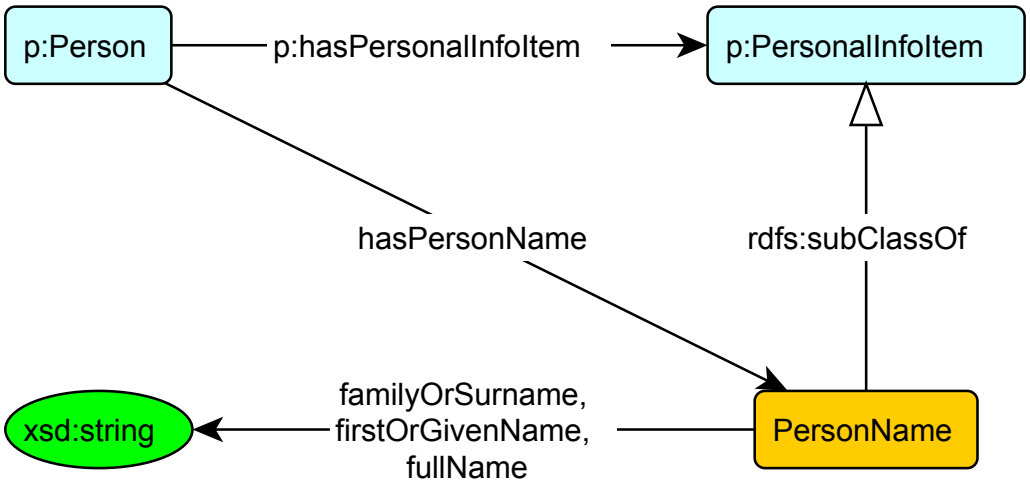
Organization pattern

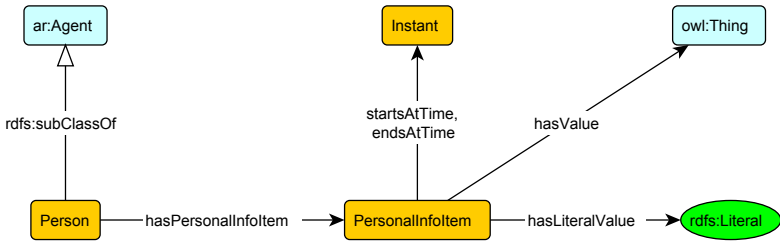


Person


ar:Agent
Instant
owl:Thing
rdfs:subClassOf
startsAtTime,
endsAtTime
hasValue
Person
hasPersonalInfoItem
PersonalInfoItem
hasLiteralValue
rdfs:Literal

Person Name





$$Person \sqsubseteq ar{:}Agent$$

$$PersonalInfoItem \sqsubseteq (=1\ hasPersonalInfoItem^{-}.Person)$$

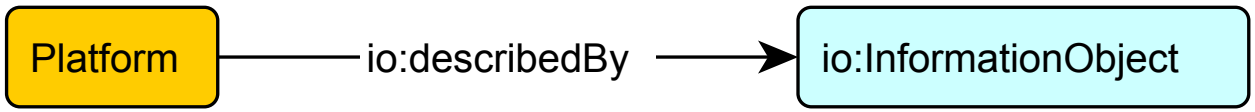
$$PersonalInfoItem \sqsubseteq (=1\ startsAtTime.Instant)$$

$$PersonalInfoItem \sqsubseteq (\leqslant 1\ endsAtTime.Instant)$$

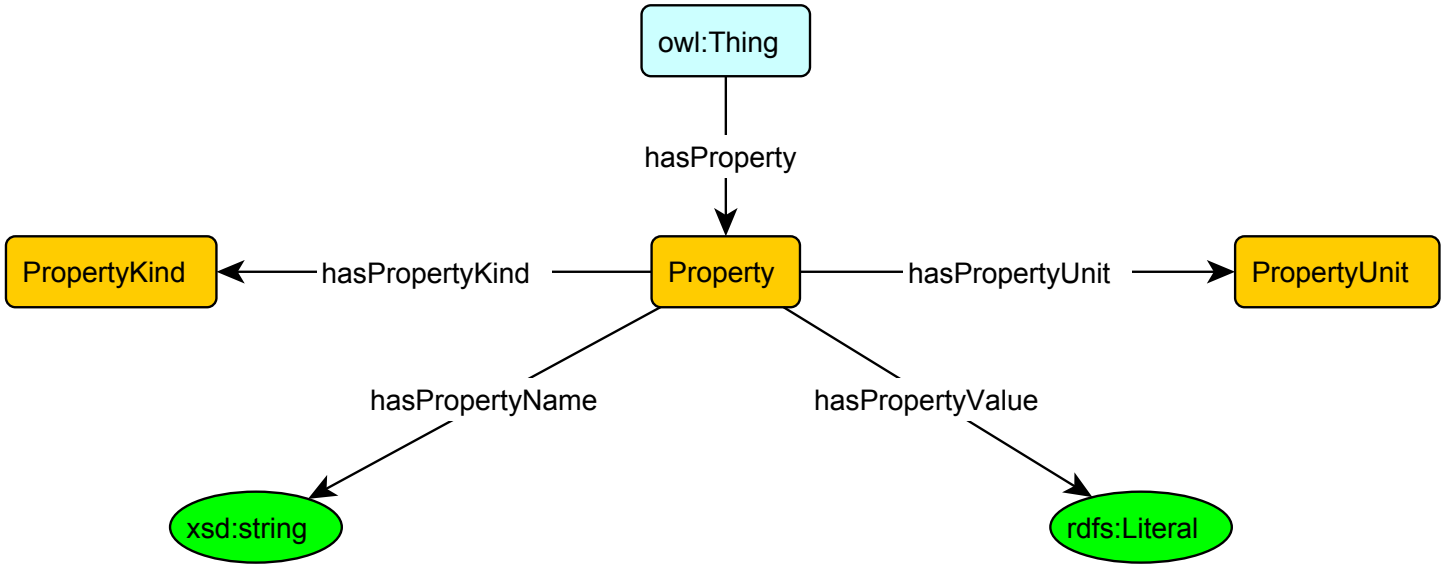
$$\exists hasPersonalInfoItem.PersonalInfoItem \sqsubseteq Person$$

$$\exists startsAtTime.Instant \sqsubseteq PersonalInfoItem$$

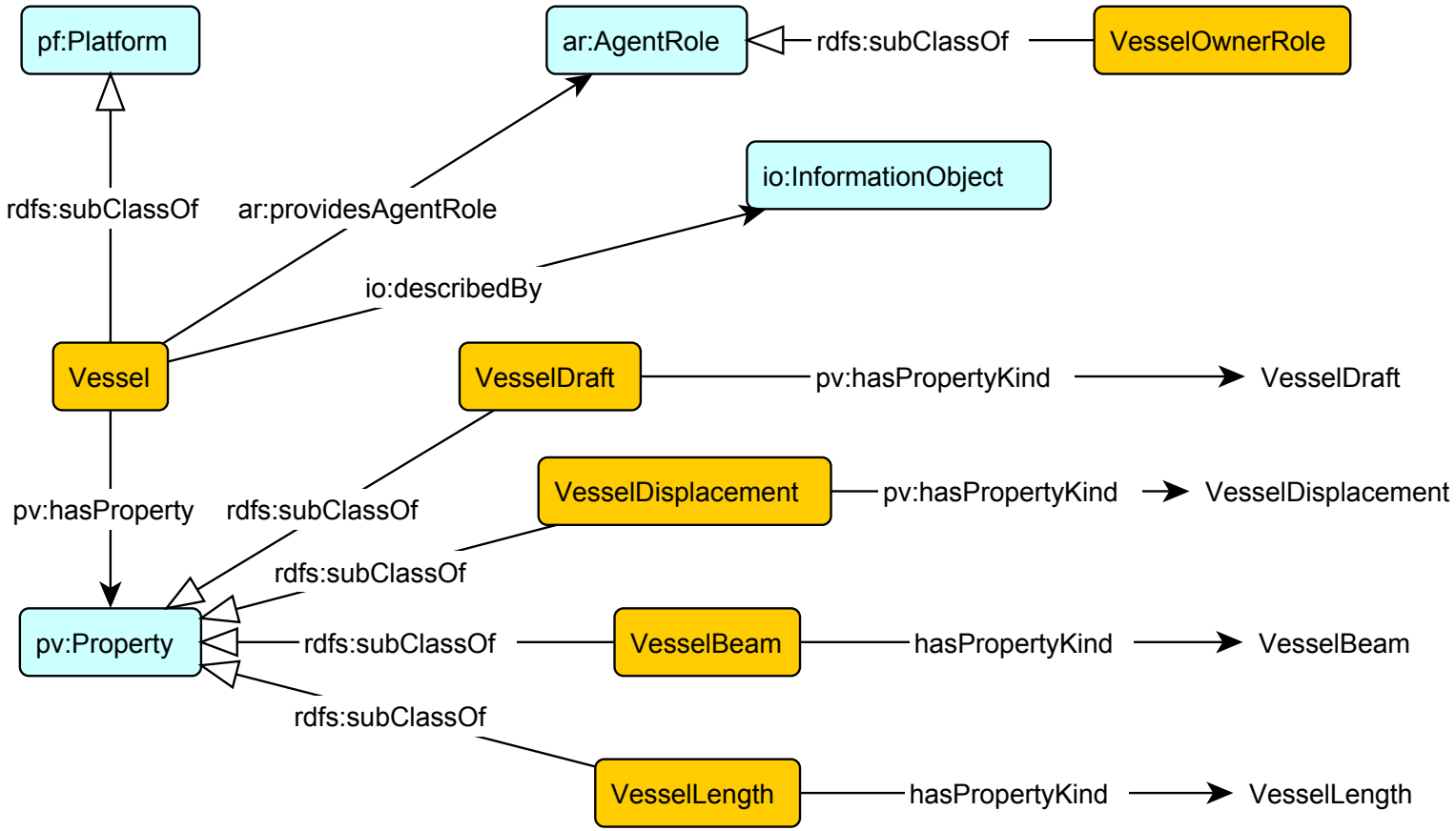
Platform pattern (stub)



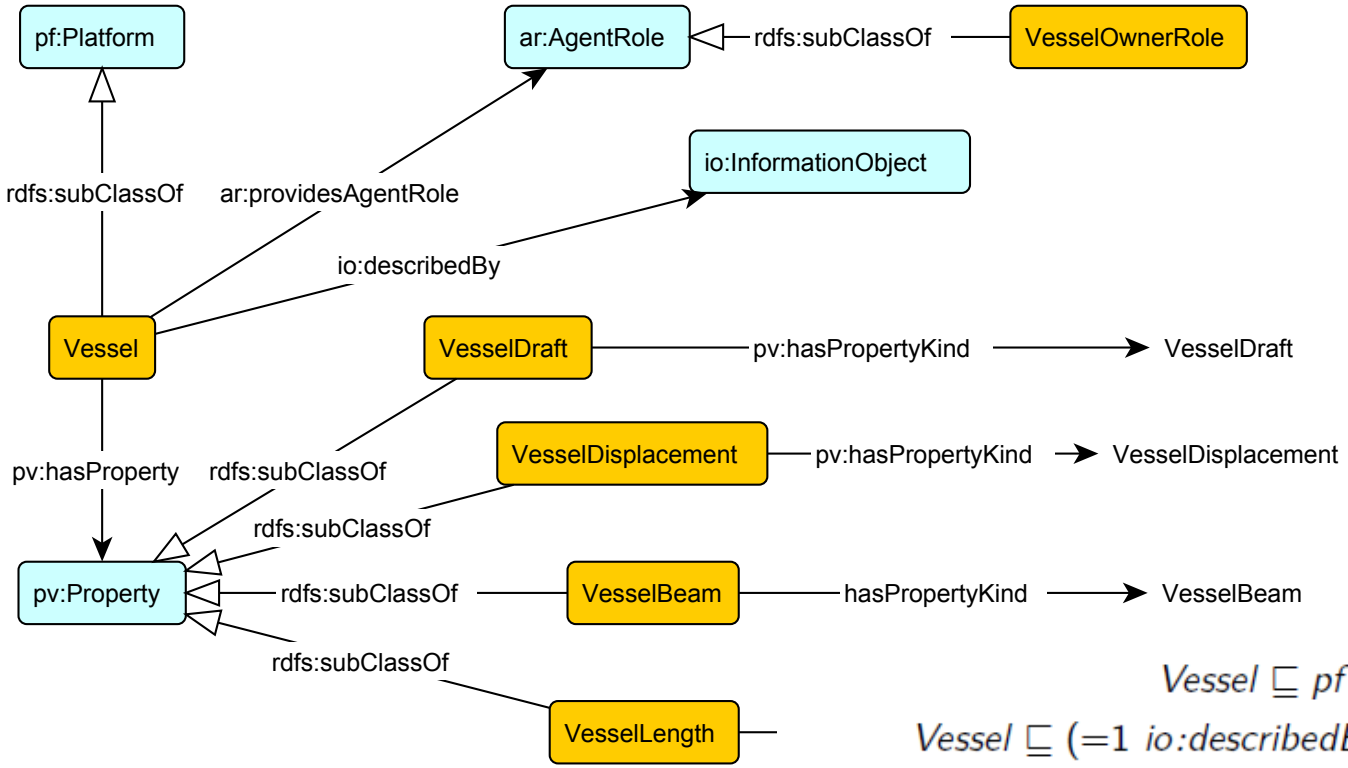
Property Value pattern



Vessel module



Vessel module



$$Vessel \sqsubseteq pf:Platform$$

$$Vessel \sqsubseteq (=1\ io:describedBy.io:InformationObject)$$

$$Vessel \sqsubseteq \exists ar:providesAgentRole.VesselOwnerRole$$

$$Vessel \sqsubseteq \exists pv:hasProperty.VesselBeam$$

$$Vessel \sqsubseteq \exists pv:hasProperty.VesselDisplacement$$

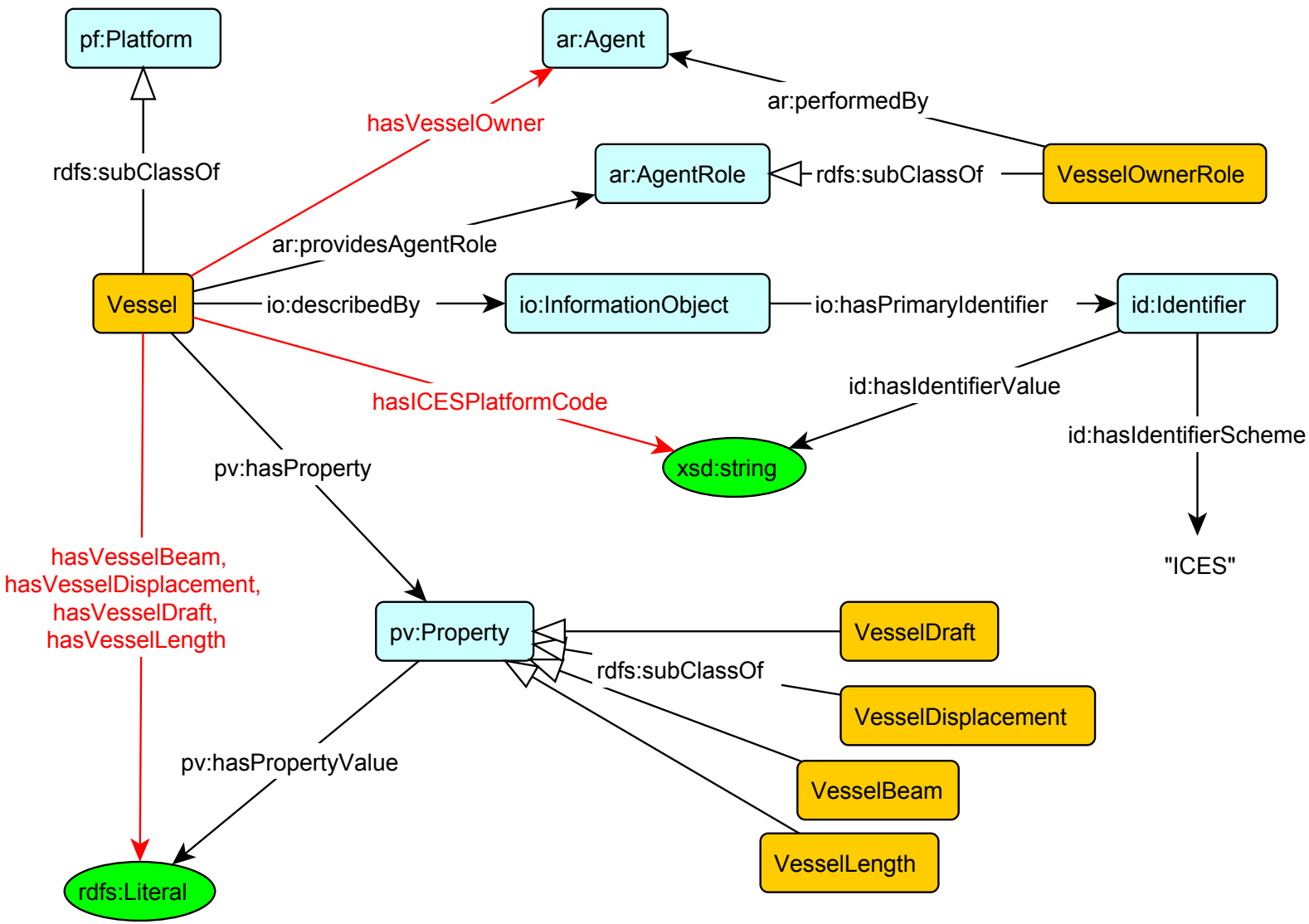
$$Vessel \sqsubseteq \exists pv:hasProperty.VesselDraft$$

$$Vessel \sqsubseteq \exists pv:hasProperty.VesselLength$$

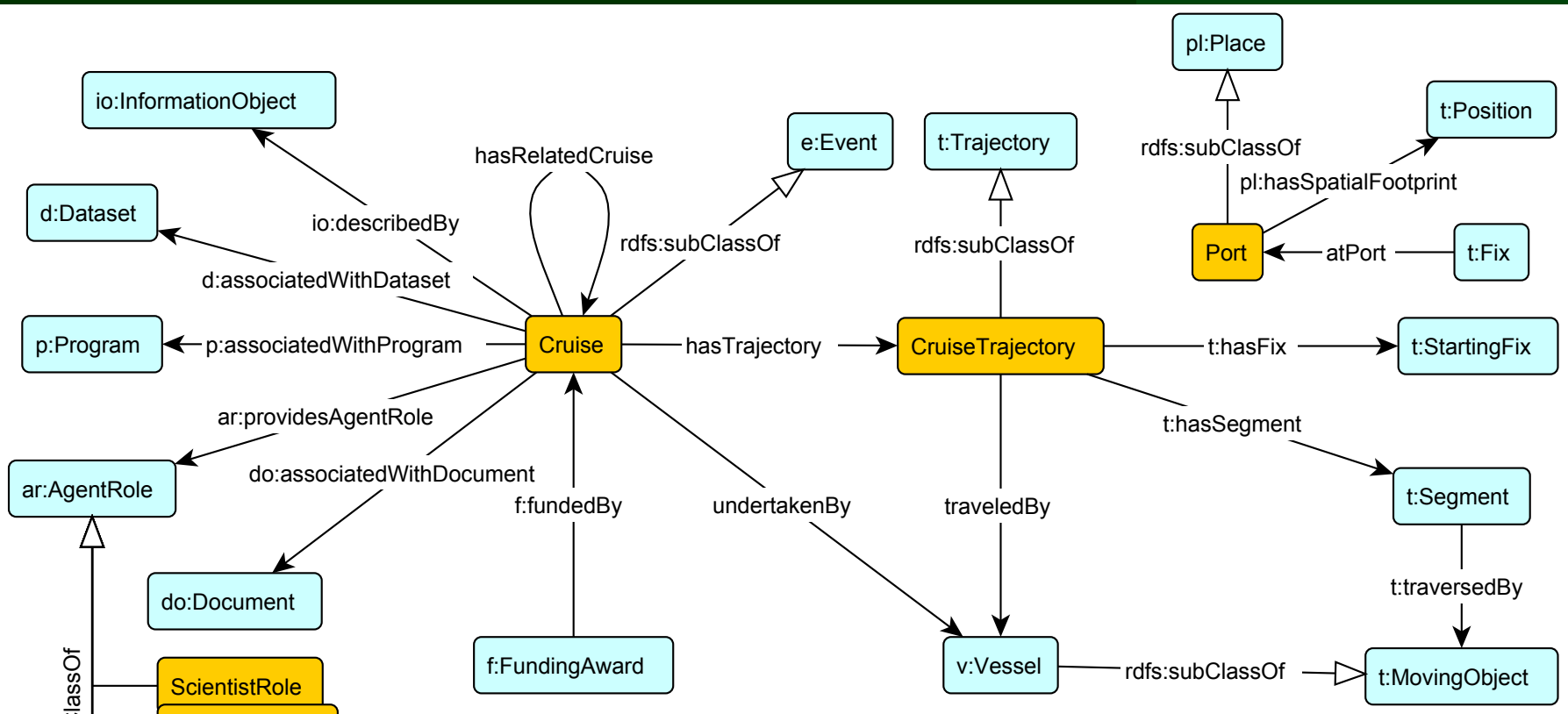
$$VesselBeam \sqsubseteq \exists pv:hasPropertyKind.\{VesselBeam\}$$

$$VesselDisplacement \sqsubseteq \exists pv:hasPropertyKind.\{VesselDisplacement\}$$

Vessel shortcuts



Cruise module



$$Cruise \sqsubseteq e{:}Event$$

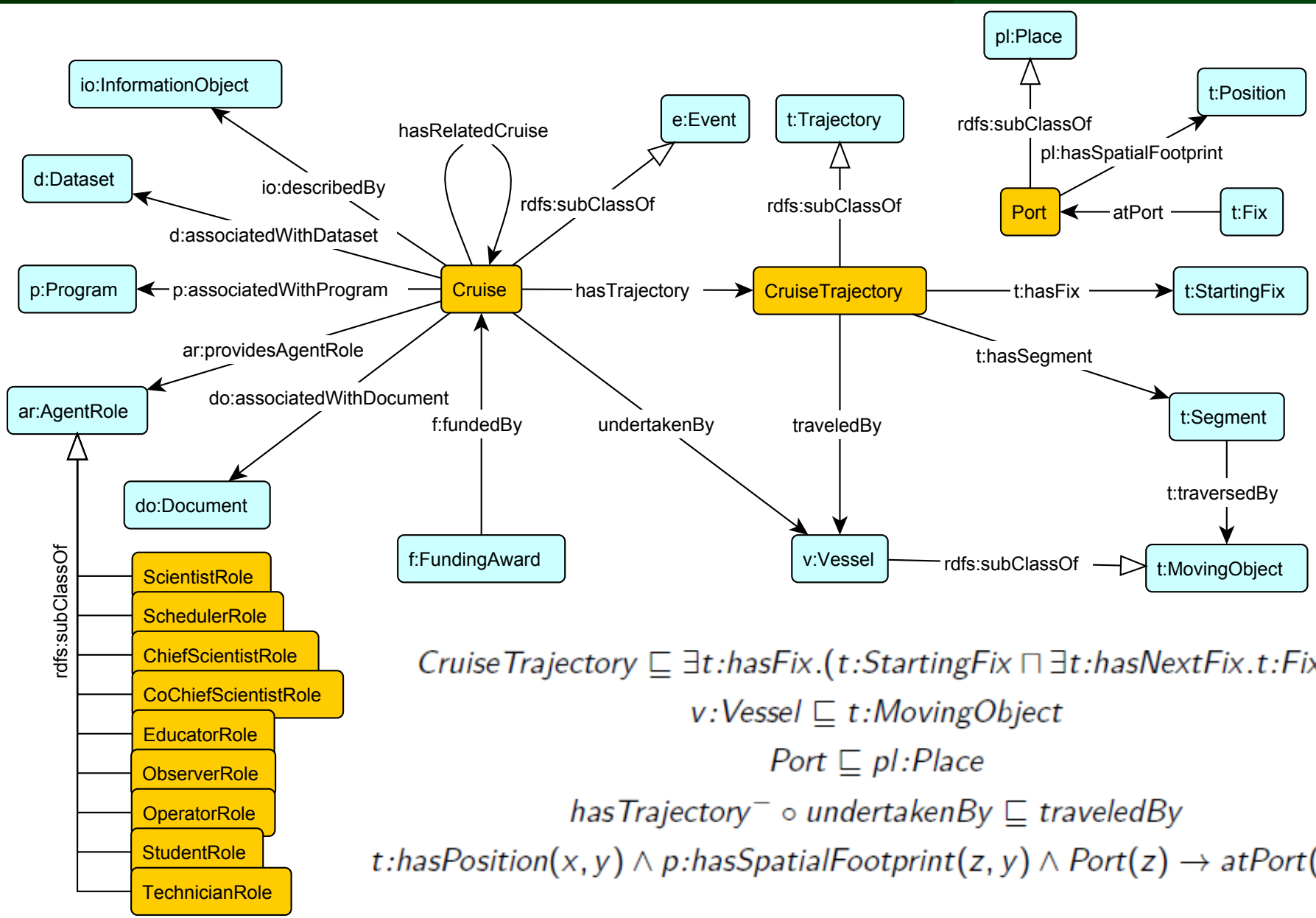
$$Cruise \sqsubseteq (=1\ hasTrajectory.CruiseTrajectory)$$

$$Cruise \sqsubseteq (=1\ undertakenBy.v{:}Vessel)$$

$$Cruise \sqsubseteq (=1\ io{:}describedBy.io{:}InformationObject)$$

$$CruiseTrajectory \sqsubseteq t{:}Trajectory$$

Cruise module



$$CruiseTrajectory \sqsubseteq \exists t{:}hasFix.(t{:}StartingFix \sqcap \exists t{:}hasNextFix.t{:}Fix)$$

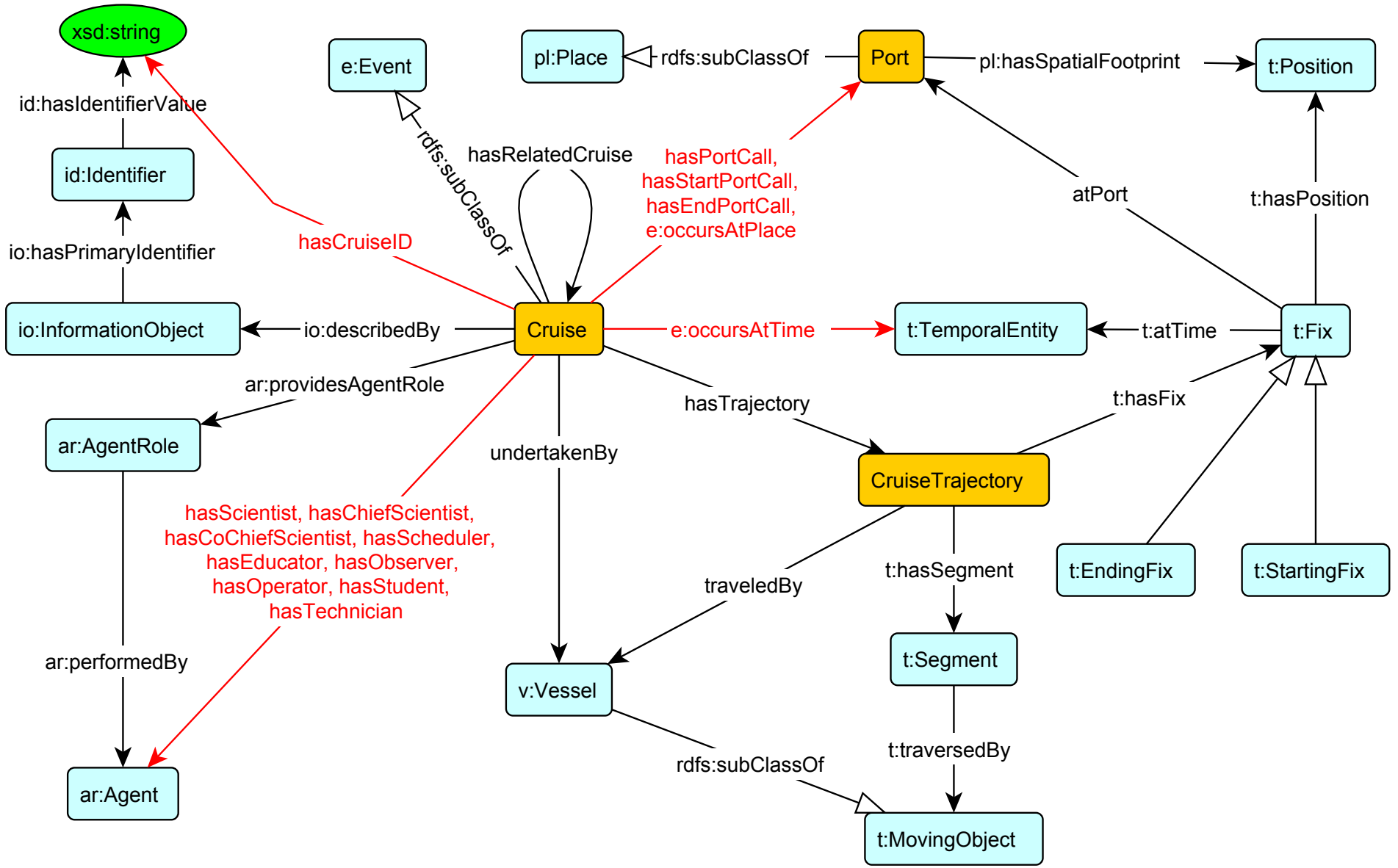
$$v{:}Vessel \sqsubseteq t{:}MovingObject$$

$$Port \sqsubseteq pl{:}Place$$

$$hasTrajectory^{-} \circ undertakenBy \sqsubseteq traveledBy$$

$$t{:}hasPosition(x,y) \wedge p{:}hasSpatialFootprint(z,y) \wedge Port(z) \rightarrow atPort(x,z)$$

Cruise shortcuts



What we need

- A critical amount of simple, general-purpose patterns
  - Well-documented
  - Not too generic, not too specialized
  - Interrelated (e.g., different versions with different granularity of the same notion)
- Languages for describing patterns.
- Languages for describing modular ontologies based on patterns.
- Tools for working directly with patterns in ontology engineering (see afternoon session – Karl Hammar's work)

Thanks!

References

Pascal Hitzler, Aldo Gangemi, Krzysztof Janowicz, Adila Krisnathi, Valentina Presutti (eds.), Ontology Engineering with Ontology Design Patterns: Foundations and Applications. Studies on the Semantic Web. IOS Press/AKA Verlag, 2016/2017.

Hitzler, Krötzsch, Rudolph, Foundations of Semantic Web Technologies, CRC/Chapman & Hall, 2010

Adila A. Krisnadhi, Yingjie Hu, Krzysztof Janowicz, Pascal Hitzler, Robert Arko, Suzanne Carbotte, Cynthia Chandler, Michelle Cheatham, Douglas Fils, Tim Finin, Peng Ji, Matthew Jones, Nazifa Karima, Kerstin Lehnert, Audrey Mickle, Tom Narock, Margaret O'Brien, Lisa Raymond, Adam Shepherd, Mark Schildhauer, Peter Wiebe, The GeoLink Framework for Pattern-based Linked Data Integration. In: Proceedings of the Posters and Demos session at the 14th International Semantic Web Conference (ISWC 2015), Bethlehem, Pensylvania, USA, October 2015.

References

Adila A. Krisnadhi, Yingjie Hu, Krzysztof Janowicz, Pascal Hitzler, Robert Arko, Suzanne Carbotte, Cynthia Chandler, Michelle Cheatham, Douglas Fils, Tim Finin, Peng Ji, Matthew Jones, Nazifa Karima, Audrey Mickle, Tom Narock, Margaret O'Brien, Lisa Raymond, Adam Shepherd, Mark Schildhauer, Peter Wiebe, The GeoLink Modular Oceanography Ontology. In: Marcelo Arenas, Óscar Corcho, Elena Simperl, Markus Strohmaier, Mathieu d'Aquin, Kavitha Srinivas, Paul T. Groth, Michel Dumontier, Jeff Heflin, Krishnaprasad Thirunarayan, Steffen Staab (eds.), The Semantic Web - ISWC 2015 - 14th International Semantic Web Conference, Bethlehem, PA, USA, October 11-15, 2015, Proceedings, Part II. Lecture Notes in Computer Science 9367, Springer, Heidelberg, 2015, 301-309.

Adila Krisnadhi, Ontology Pattern-Based Data Integration. Dissertation, Department of Computer Science and Engineering, Wright State University, 2015.

References

Yingjie Hu, Krzysztof Janowicz, David Carral, Simon Scheider, Werner Kuhn, Gary Berg-Cross, Pascal Hitzler, Mike Dean, Dave Kolas, A Geo-Ontology Design Pattern for Semantic Trajectories. In: Thora Tenbrink, John G. Stell, Antony Galton, Zena Wood (Eds.): Spatial Information Theory - 11th International Conference, COSIT 2013, Scarborough, UK, September 2-6, 2013. Proceedings. Lecture Notes in Computer Science Vol. 8116, Springer, 2013, pp. 438-456.

Eva Blomqvist, Pascal Hitzler, Krzysztof Janowicz, Adila Krisnadhi, Thomas Narock, Monika Solanki, Considerations regarding Ontology Design Patterns. Semantic Web 7 (1), 2016, 1-7.

The Reporting Event ODP

Agnieszka Ławrynowicz[1], Ewa Kowalczuk[1]

[1]Faculty of Computing, Poznan University of Technology, Poznan, Poland

May 28, 2017
Tutorial: Modular Ontology Modeling with Ontology Design Patterns at ESWC2017

Motivation

- emerging need for storing and investigating not only information about a particular event, but also the provenance of the information and circumstances of its provision
- properties attributed to an event are not stored as facts, but as a narrative of a particular agent, which could differ from the narratives of other agents

Do existing event ontologies address the subjectivity of event properties? 1/2

- CIDOC Conceptual Reference Model (CRM): does not allow to mark the level of property value probability or attribute a property assignment to a particular agent
- Linking Open Descriptions of Events ontology (LODE): allows for linking events to media objects presenting them and thus denoting sources
- IPTC NewsML-G2 controlled vocabulary: allows to define whether an event and its time interval are confirmed or not; IPTC rNews model introduces a property rnews:accountablePerson for linking to a person responsible for a particular news item (as a whole).

Do existing event ontologies address the subjectivity of event properties? 2/2

- Simple Event Model (SEM): introduces a subclass of sem:Constraint, called sem:View, allowing to mark some attributed property as a belief (point of view) of a particular sem:Authority. The property assignment is constant: there is no means of representing the fact that a view changed over time
- BBC Storyline Ontology introduces a notion of nsl:Storyline, to denote the editorial perspective of an event or a group of events. It can be attributed to a specific owner. Storylines have a larger span than a single event. They can include nsl:StorylineSlots: real world events or inner storylines.

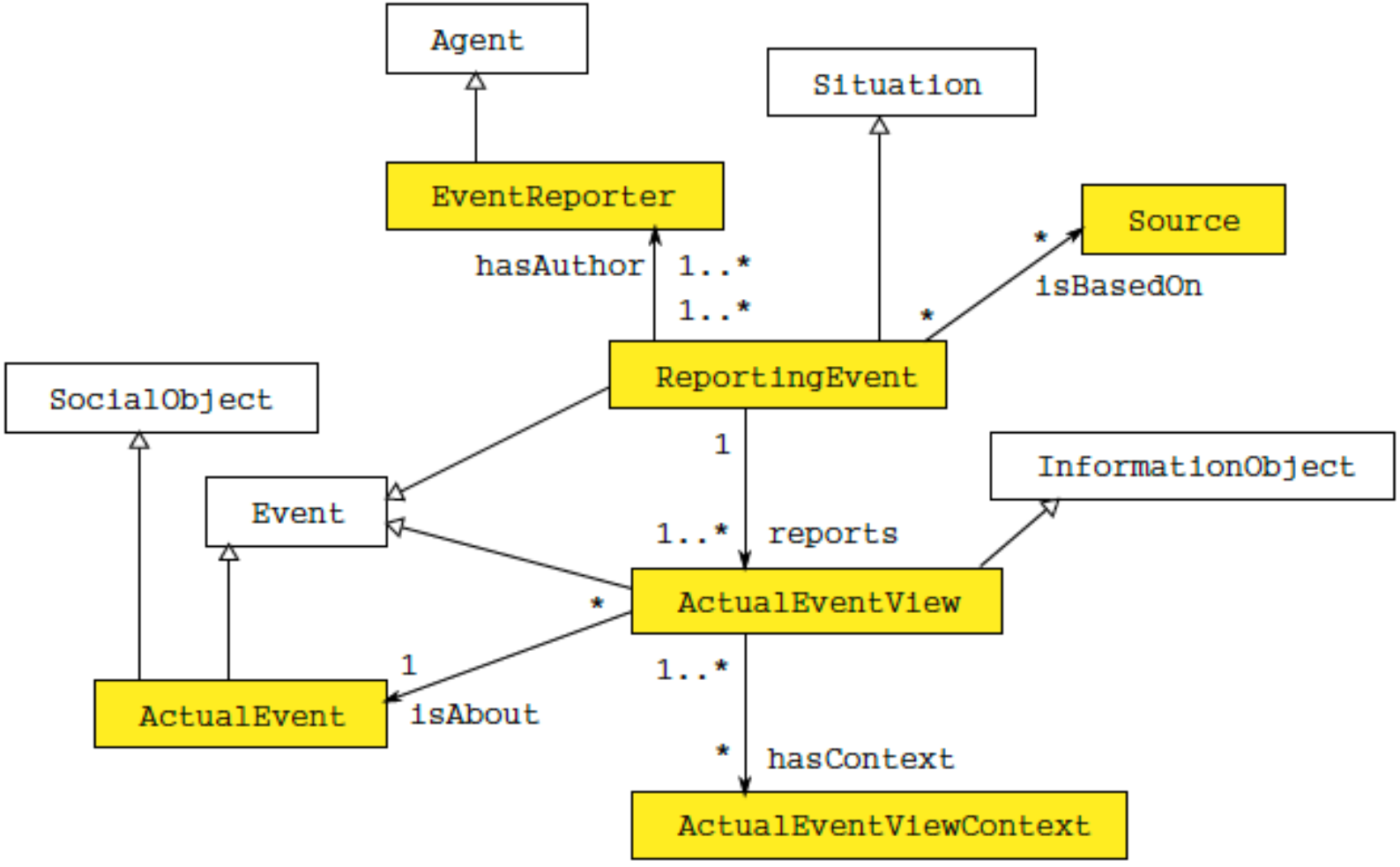
Reporting Event ODP: Intent

To allow for modelling situations in which the knowledge about an event cannot be treated as certain. It is particularly useful for cases in which two or more agents provide different, contradictory information about the same event. Also for modelling situation in which a single agent provided contradictory information about the same event at different points in time. The pattern allows for stating different circumstances of an act of the information provision.

Competency questions

- What characteristics (e.g. date, participants, cause) is an actual event said to have?
- Which agent made a statement about an actual event?
- On which sources these statements were based?
- What were the circumstances of providing information about an actual event?

Reporting Event ODP



http://ontologydesignpatterns.org/wiki/Submissions:ReportingEvent

Pattern formalization

$\mathsf{ActualEvent} \sqsubseteq \mathsf{Event}$
$\mathsf{ActualEvent} \sqsubseteq \mathsf{SocialObject} \sqsubseteq \mathsf{Object}$
$\mathsf{ActualEventView} \sqsubseteq \mathsf{Event}$
$\mathsf{ActualEventView} \sqsubseteq \mathsf{InformationObject} \sqsubseteq \mathsf{Object}$
$\mathsf{ActualEventView} \sqsubseteq\ = 1\ \mathsf{isAbout}.\mathsf{ActualEvent}$
$\mathsf{ActualEventView} \sqsubseteq \forall \mathsf{isAbout}.\mathsf{ActualEvent}$
$\mathsf{ActualEventView} \sqsubseteq\ = 1\ \mathsf{reports}^{-1}.\mathsf{ReportingEvent}$
$\mathsf{ActualEventView} \sqsubseteq \forall \mathsf{hasContext}.\mathsf{ActualEventViewContext}$
$\mathsf{ActualEventViewContext} \sqsubseteq \exists \mathsf{hasContext}^{-1}.\mathsf{ActualEventView}$
$\mathsf{ReportingEvent} \sqsubseteq \mathsf{Event}$
$\mathsf{ReportingEvent} \sqsubseteq \mathsf{Situation}$
$\mathsf{ReportingEvent} \sqsubseteq \exists \mathsf{reports}.\mathsf{ActualEventView}$
$\mathsf{ReportingEvent} \sqsubseteq \forall \mathsf{reports}.\mathsf{ActualEventView}$
$\mathsf{ReportingEvent} \sqsubseteq \exists \mathsf{hasAuthor}.\mathsf{EventReporter}$
$\mathsf{ReportingEvent} \sqsubseteq \forall \mathsf{isBasedOn}.\mathsf{Source}$
$\mathsf{EventReporter} \sqsubseteq \mathsf{Agent}$
$\mathsf{EventReporter} \sqsubseteq \exists \mathsf{hasAuthor}^{-1}.\mathsf{ReportingEvent}$
$\mathsf{Source} \sqsubseteq (\mathsf{Event} \sqcup \mathsf{Object})$
$\mathsf{reports} \sqsubseteq \mathsf{isSettingFor}$

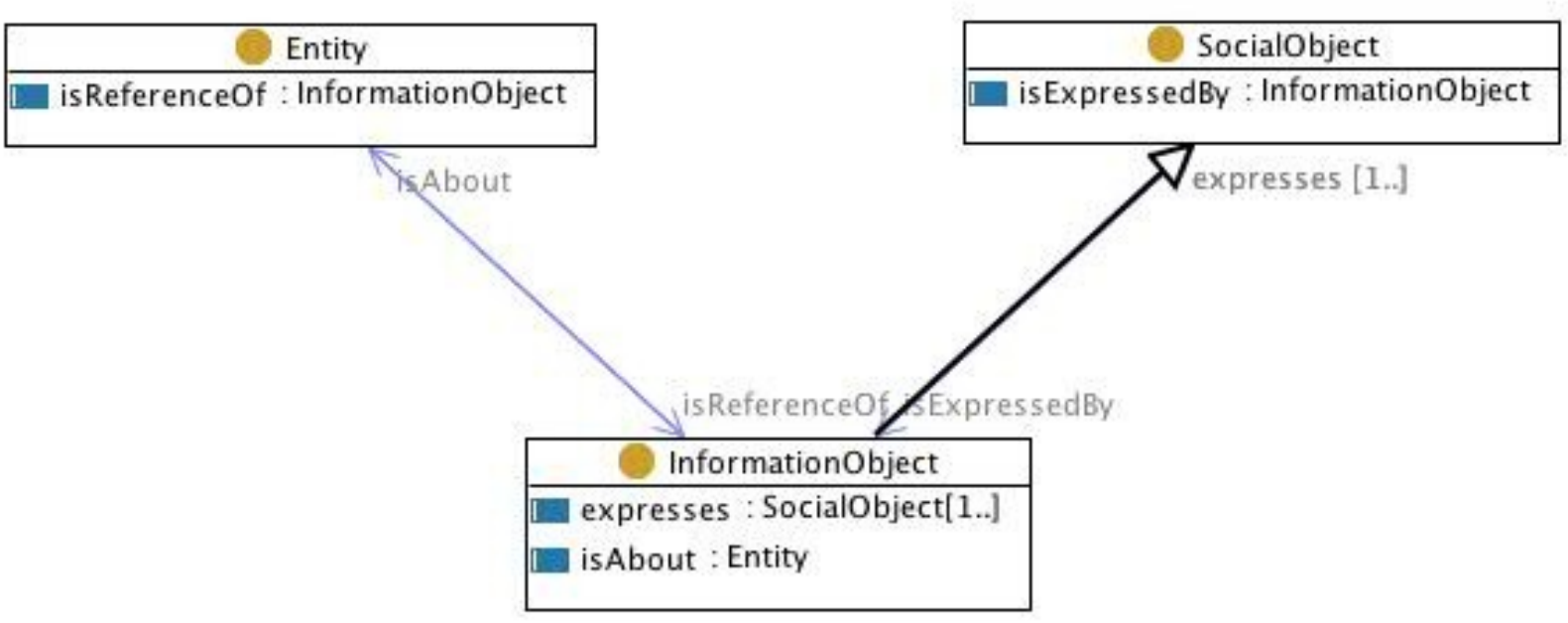
Intension Extension ODP 1/2

Intent: To represent the meaning of an information object: the concepts it expresses, the things it is about.

Competency Questions:

- What is the meaning of an information object?
- What information objects express this meaning?
- What is this about?
- How can I call this?

Intension Extension ODP 2/2



http://ontologydesignpatterns.org/wiki/Submissions:IntensionExtension

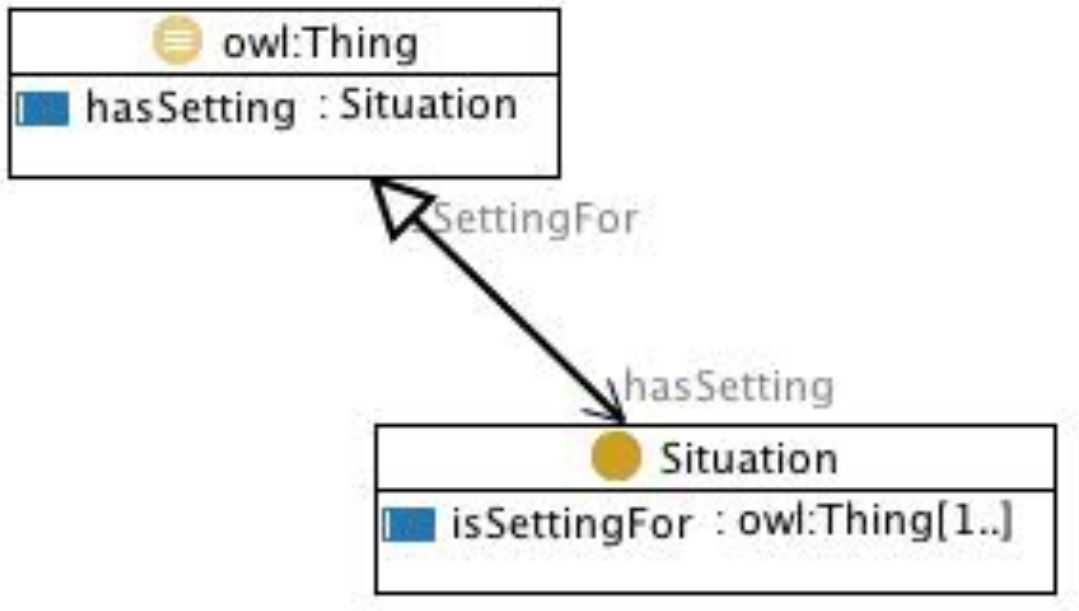
Situation ODP 1/2

Intent: To represent contexts or situations, and the things that are contextualized.

Competency Questions:

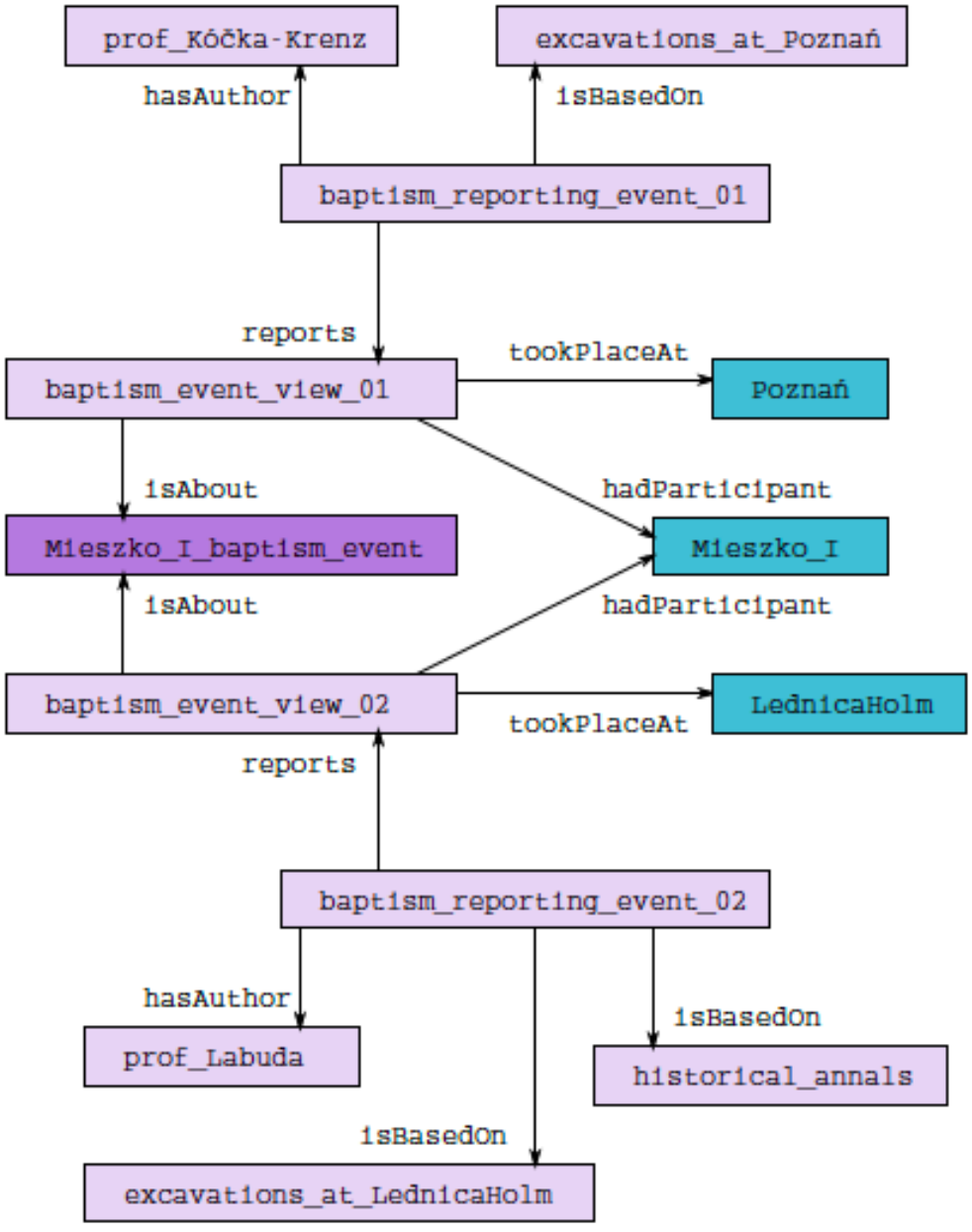
- What is the context or situation of something?
- What are the things present in this context or situation?

Situation ODP 2/2



http://ontologydesignpatterns.org/wiki/Submissions:Situation

Sample use of Reporting Event ODP: Historical Debate



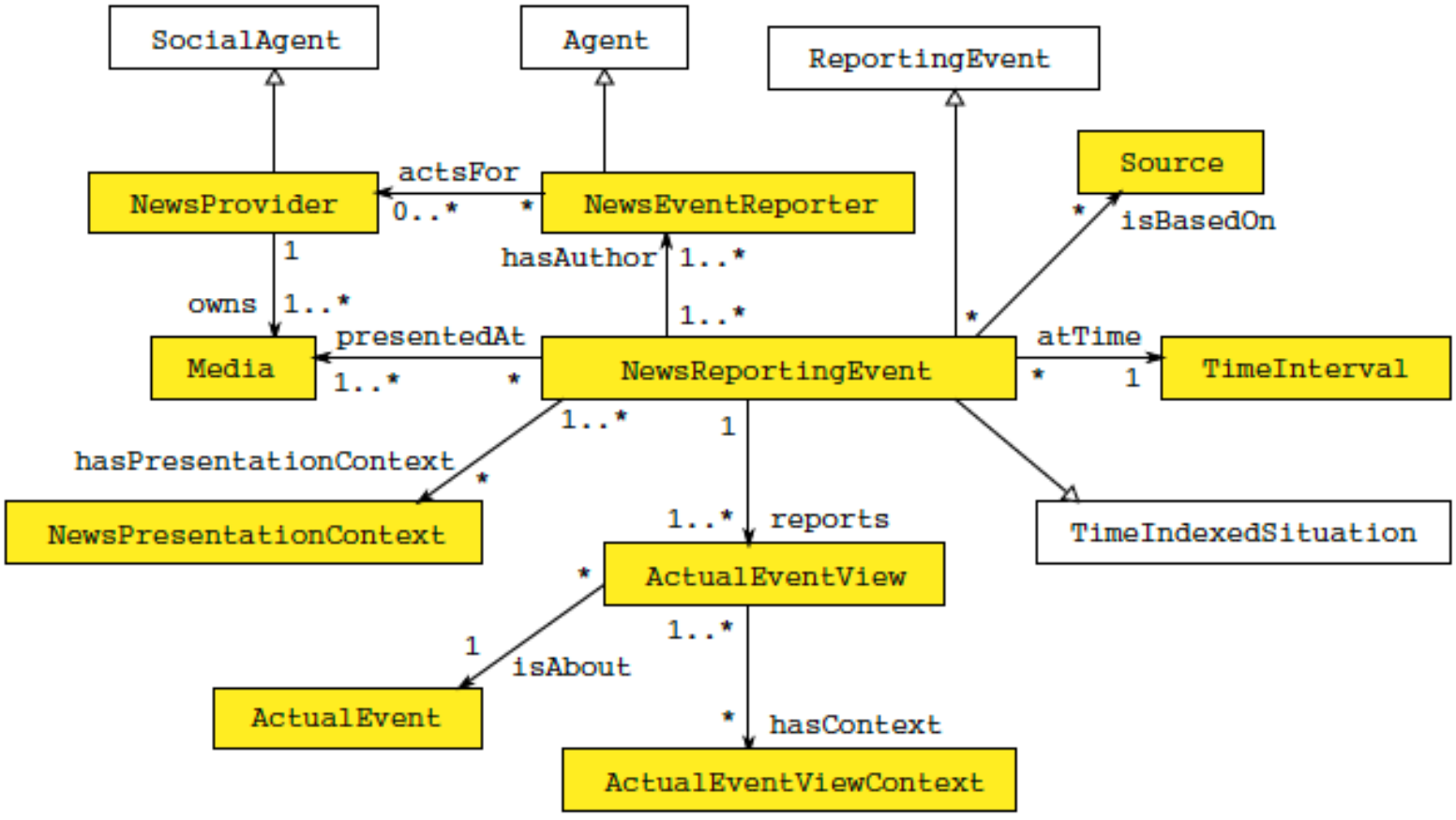
News Reporting Event ODP: Intent

Can be used for modelling situations in which we are not certain that a particular actual event has the properties which were described in a news message. We want to define the properties of an actual event which were reported (time, place, actors, subevents, cause, effect etc.), but not to treat them as universal, verified knowledge. The pattern also allows to define who is responsible for a particular description of an event and how this description is dealt with.

Competency questions

- What aspects of an actual event were presented in the news message?
- Who reported an actual event? Which news provider they represented?
- When was a certain actual event reported for the first time?
- What actual events are presented in a certain medium/by media of a certain news provider?
- How was an actual event presented?

News Reporting Event ODP



http://ontologydesignpatterns.org/wiki/Submissions:NewsReportingEvent

Pattern formalization

$\mathsf{NewsReportingEvent} \sqsubseteq \mathsf{ReportingEvent}$
$\mathsf{NewsReportingEvent} \sqsubseteq \mathsf{TimeIndexedSituation}$
$\mathsf{NewsReportingEvent} \sqsubseteq \exists \mathsf{hasAuthor}.\mathsf{NewsEventReporter}$
$\mathsf{NewsReportingEvent} \sqsubseteq \exists \mathsf{presentedAt}.\mathsf{Media}$
$\mathsf{NewsEventReporter} \sqsubseteq \mathsf{EventReporter}$
$\mathsf{NewsEventReporter} \sqsubseteq \exists \mathsf{hasAuthor}^{-1}.\mathsf{NewsReportingEvent}$
$\mathsf{NewsProvider} \sqsubseteq \mathsf{SocialAgent}$
$\mathsf{NewsProvider} \sqsubseteq \exists \mathsf{owns}.\mathsf{Media}$
$\mathsf{Media} \sqsubseteq \ = 1\ \mathsf{owns}^{-1}.\mathsf{NewsProvider}$
$\mathsf{NewsPresentationContext} \sqsubseteq \exists \mathsf{hasPresentationContext}^{-1}.\mathsf{NewsReportingEvent}$
$\mathsf{hasPresentationContext} \sqsubseteq \mathsf{isSettingFor}$

Acting For ODP

Intent: To represent that some agent is acting in order to forward the action of a social (non-physical) agent.

Competency Questions:

- Who is working for which organization?
- Who is representing the company?

`http://ontologydesignpatterns.org/wiki/Submissions:ActingFor`

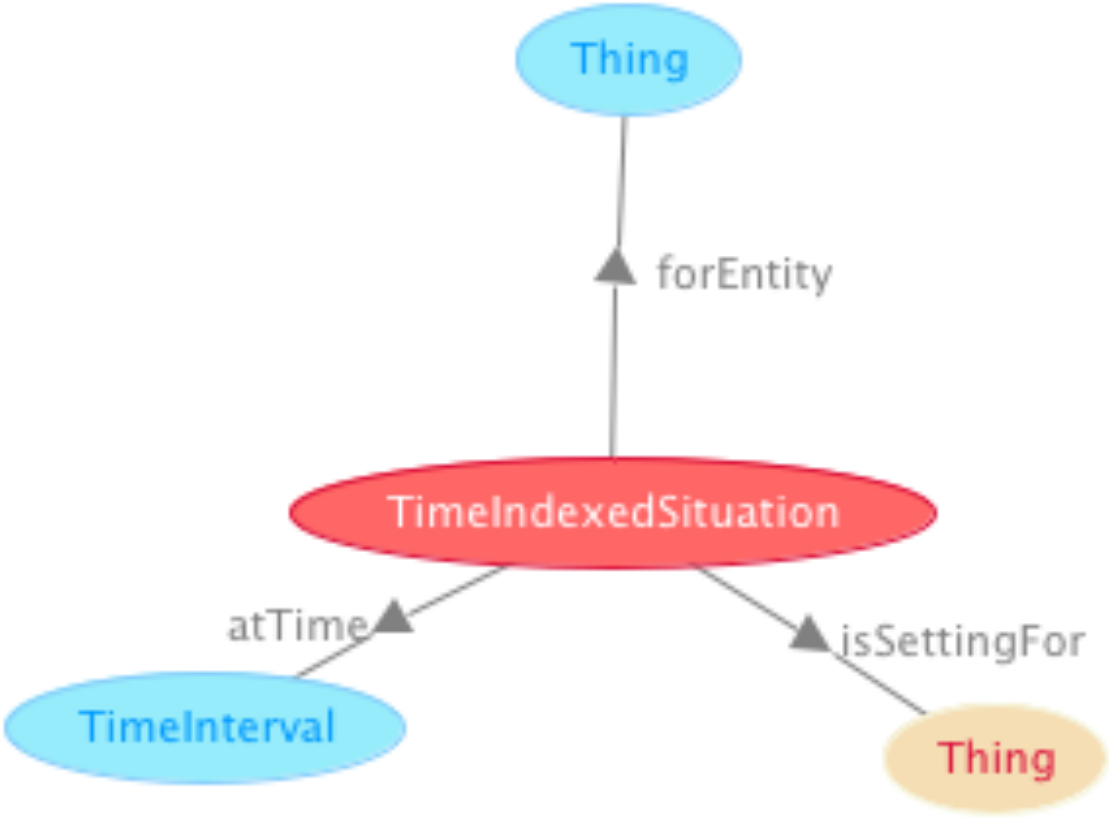
Time Indexed Situation ODP 1/2

Intent: To represent time indexed situations.

Competency Questions:

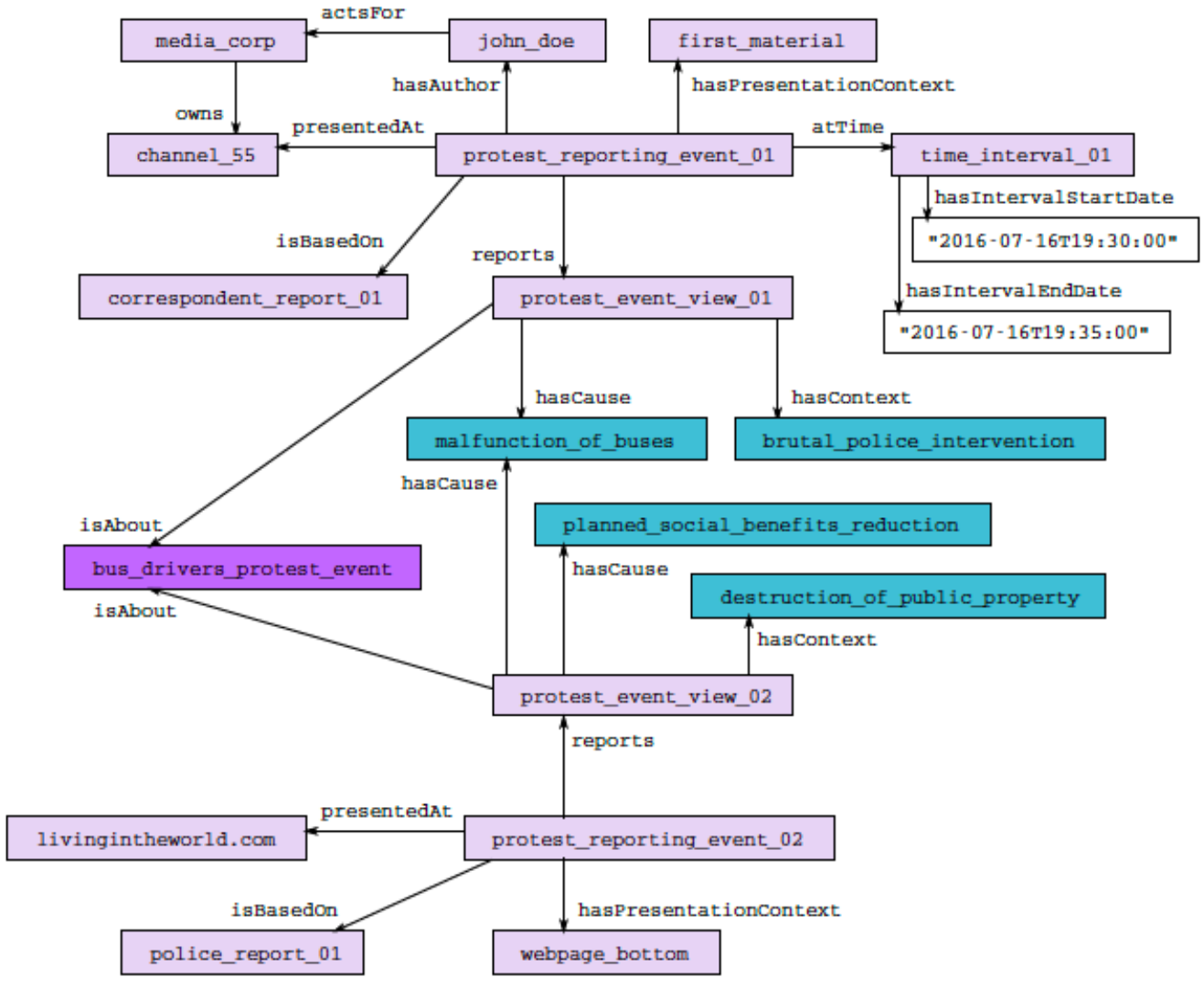
- At what time did a certain situation occur?
- What situations occurred at a certain time?

Time Indexed Situation ODP 2/2



http://ontologydesignpatterns.org/wiki/Submissions:TimeIndexedSituation

Sample use of News Reporting Event ODP: Presentation of Social Unrests



Thank you!

JÖNKÖPING UNIVERSITY
School of Engineering

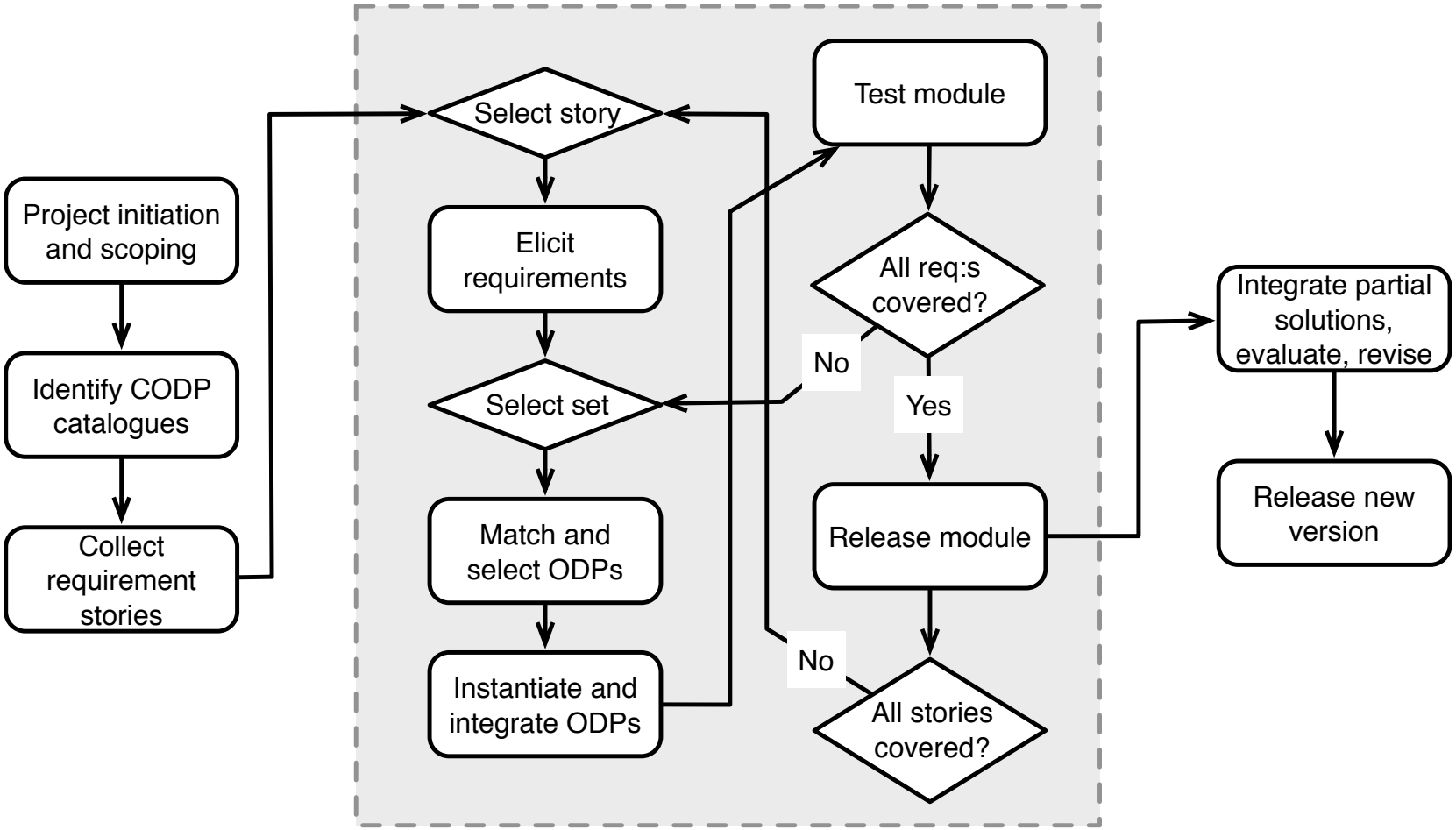
Introduction to XD and XDP

Karl Hammar

2017-05-28

eXtreme Design

- *”a family of methods and associated tools, based on the application, exploitation, and definition of Ontology Design Patterns (ODPs) for solving ontology development issues”* – Presutti et al.
- Agile, iterative, pair development, testing emphasis
- Requirements written as user storys formalised as Competency Questions, Contextual Statements, Reasoning Requirements
- Tight customer integration
- Key steps: find ODP, instantiate ODP, integrate solution


Project initiation and scoping
Identify CODP catalogues
Collect requirement stories
Select story
Elicit requirements
Select set
Match and select ODPs
Instantiate and integrate ODPs
Test module
All req:s covered?
No
Yes
Release module
No
All stories covered?
Integrate partial solutions, evaluate, revise
Release new version

XD for WebProtégé (XDP)

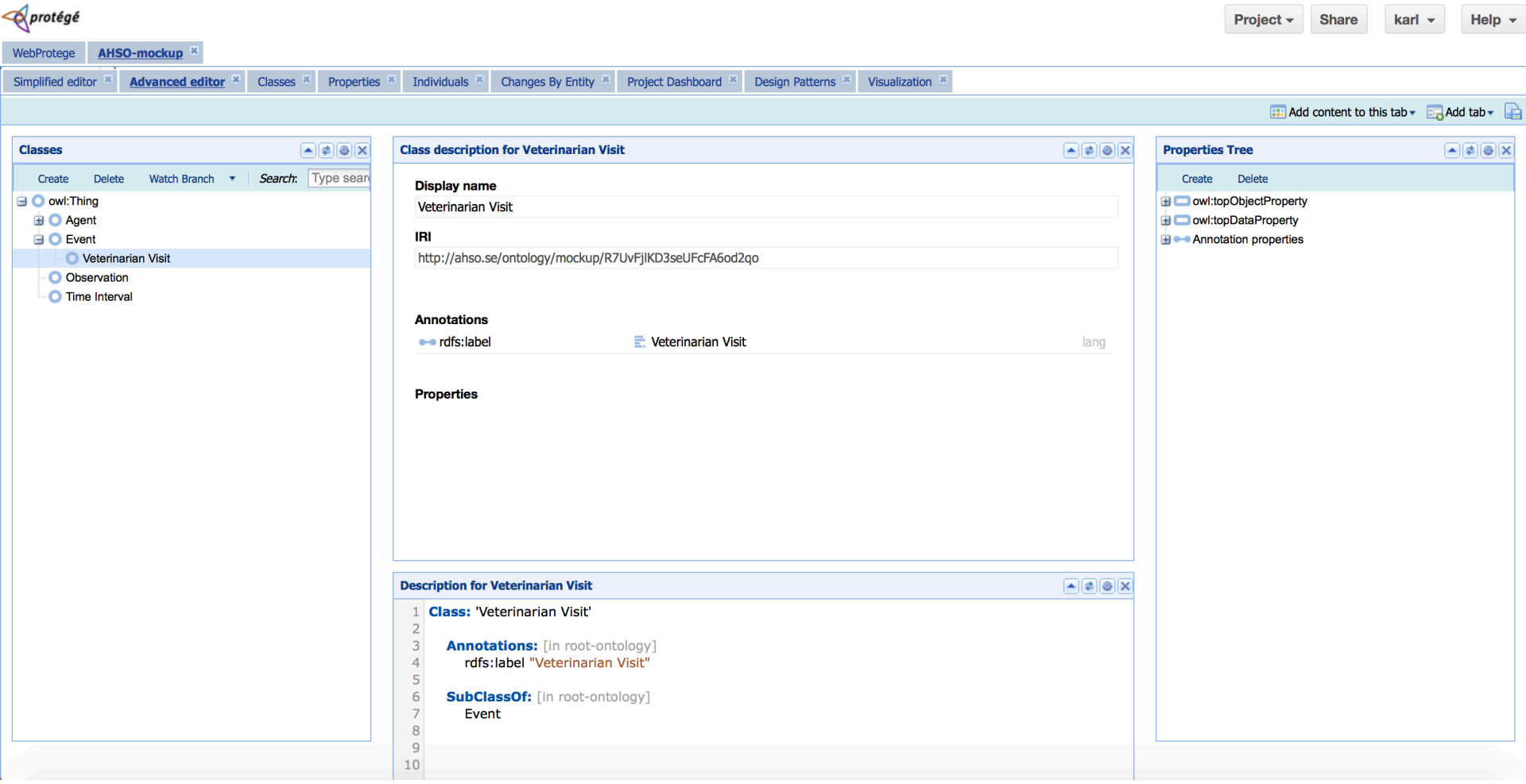
Fork of WebProtégé including tooling to support some XD steps:

- Find ODPs
- Instantiate ODPs (template-based or specialisation-based)
- Integrate ODPs into solution (basic alignment)

Also includes visualization, courtesy of code from the VisualDataWeb project and new UI tabs for advanced editing

Some restrictions of WebProtégé:

- No reasoning
- ODP namespaces cloned, not imported


protégé
Project
Share
karl
Help
WebProtege
AHSO-mockup
Simplified editor
Advanced editor
Classes
Properties
Individuals
Changes By Entity
Project Dashboard
Design Patterns
Visualization
Add content to this tab
Add tab
Classes
Create
Delete
Watch Branch
Search:
Type sear
owl:Thing
Agent
Event
Veterinarian Visit
Observation
Time Interval
Class description for Veterinarian Visit
Display name
Veterinarian Visit
IRI
http://ahso.se/ontology/mockup/R7UvFjlKD3seUFcFA6od2qo
Annotations
rdfs:label
Veterinarian Visit
lang
Properties
Description for Veterinarian Visit
1 Class: 'Veterinarian Visit'
2
3 Annotations: [in root-ontology]
4 rdfs:label "Veterinarian Visit"
5
6 SubClassOf: [in root-ontology]
7 Event
8
9
10
Properties Tree
Create
Delete
owl:topObjectProperty
owl:topDataProperty
Annotation properties

protégé

Project | Share | karl | Help

WebProtege | AHSO-mockup

Simplified editor | Advanced editor | Classes | Properties | Individuals | Changes By Entity | Project Dashboard | Design Patterns | Visualization

Add content to this tab | Add tab

ODP Selector

ODP Category Selector

Select Category

ODP Search

Query:

Search | Reset

Results list

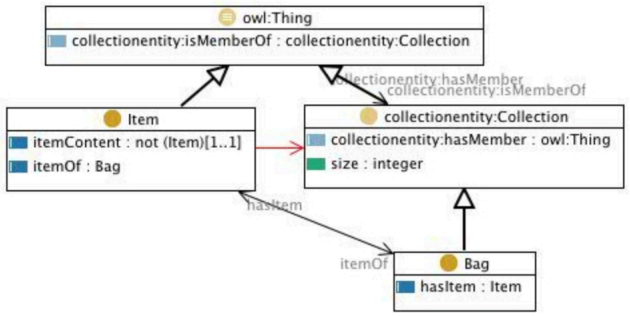
Name
Affordance
agent role
Airline.owl
Aquatic resource observation ontology
AquaticResources
Bag
BasicPlan
BasicPlanExecution
CatchRecord
Classification
ClimaticZone
Co-participation
collection entity
CommunicationEvent
Communities

ODP Details

Use this Pattern

Pattern Description | WebVOWL Visualisation

Graphical representation



General description

Name	Bag
Intent	To model bags of items (elements). The Bag is characterized by a collection that can have multiple copies of each object.
Solution description	The Bag is characterized by a collection that can have multiple copies of each object. This is performed through the Item entity. The Item is linking exactly one resource through the relationship itemContent.
Consequences	

CODP Instantiation Wizard

X

Instantiation Method Selection

CODP Instantiation | CODP Visualisation

Select the appropriate Content Ontology Design Pattern instantiation method from the choices below. For a discussion on their respective attributes and effects, see http://goo.gl/dv8pA3

(•) **Template-Based Instantiation**

In this method the CODP building block is treated as a template that is instantiated into the target ontology module by way of copying and renaming its constituent classes and properties. Advantages of this method include that CODP-level generic concepts that may be off-putting to less experienced modellers are not included in the final ontology, but only the CODP structure is kept. Disadvantages include that future alignment to other ontologies using the same CODPs may be complicated, as the IRIs of COPD-level concepts are not kept.

( ) **Import-Based Instantiation**

In this method the original CODP is imported into the target ontology module, and instantiation is performed via specialization of CODP classes and properties using subsumption axioms. Advantages of this method include increased traceability and ease of alignment with other CODPs, as IRIs of CODP-level concepts are maintained.

Back | Finish | Next

CODP Instantiation Wizard

X

CODP Instantiation | CODP Visualisation

Please provide labels for the ODP entities below that make sense when adapting the ODP to your domain.

Classes

|	item	==>	My item class
|	(collections) Bag	==>	My bag class

Object Properties

|	item content	==>	my item has some content
|	item of	==>	is item in my bag
|	has item	==>	my bag has my item

Back | Finish | Next

CODP Instantiation Wizard

X

CODP Instantiation | CODP Visualisation

Generate preview

Axiom Preview | VOWL Preview

```
Prefix: owl: <http://www.w3.org/2002/07/owl#>
Prefix: rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
Prefix: xml: <http://www.w3.org/XML/1998/namespace>
Prefix: xsd: <http://www.w3.org/2001/XMLSchema#>
Prefix: rdfs: <http://www.w3.org/2000/01/rdf-schema#>



Ontology: <wptmp:entity>


ObjectProperty: <wptmp:entity#is item in my bag>

   Domain:
      <wptmp:entity#My item class>

   Range:
      <wptmp:entity#My bag class>


ObjectProperty: <wptmp:entity#my item has some content>

   Domain:
      <wptmp:entity#My item class>


ObjectProperty: <wptmp:entity#my bag has my item>

   Domain:
      <wptmp:entity#My bag class>
```

Back | Finish | Next

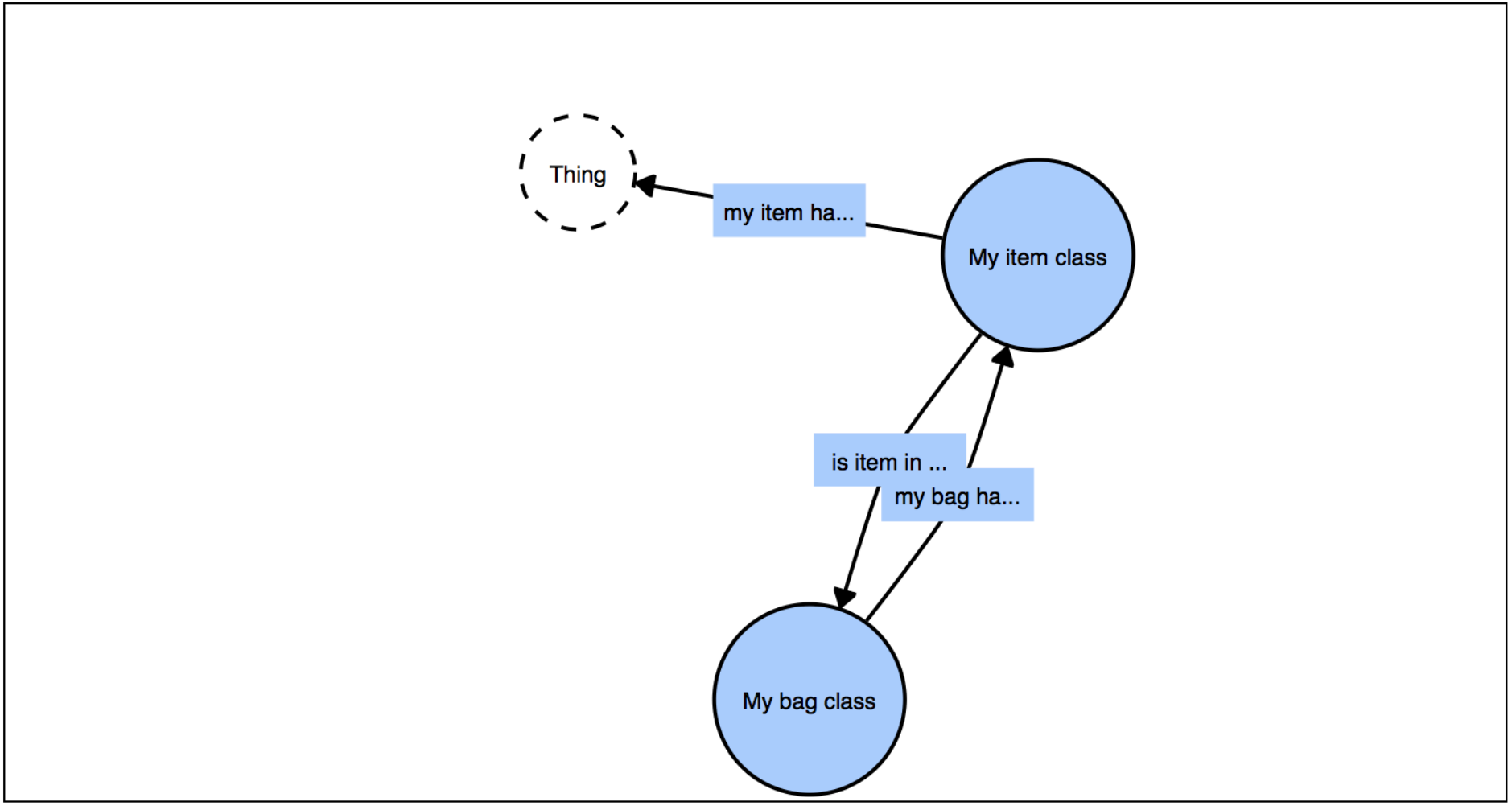
CODP Instantiation Wizard

X

CODP Instantiation | CODP Visualisation

Generate preview

Axiom Preview | VOWL Preview



Back | Finish | Next

Hands-On Session

- Using XDP and ODPs, construct an ontology covering a set of requirements and structuring a set of provided data, in the policing domain.
- Inspired by a real-world project and real-world data.
- Goal: try out and learn about the method, the tooling, and look at some ODPs.

CAVEATS

- ODP Portal, tooling, etc. are mirrored from the Internet, to account for the lack of Internet connectivity. Some IRIs you see here do not exist in the real world, or lead to content that is not the same as what you see here.
- Further: ODP quality varies greatly in the real world: stale IRIs, bad documentation, bad illustrations, dependency on remote references, etc.
- Thus: four specifically suggested patterns have been tampered with in order to simplify modelling. E.g., merged import closure, added missing documentation, added some common-sense assumptions.
- Finally: this is beta-quality research software. Expect some bugs.

Get started

Poll: Who wants some Google Refine/OpenRefine introduction as well?

- WiFi SSID “ODP Tutorial”, password: “eswc2017”
- Instructions: http://ontologydesignpatterns.org/instructions.txt
- Data: http://ontologydesignpatterns.org/data.zip
- XDP Instance: http://wp.xd-protege.com
- ODP Portal: http://ontologydesignpatterns.org
- WebVOWL Instance: http://vis.xd-protege.com

JÖNKÖPING UNIVERSITY

School of Engineering