# Implementing CIDOC CRM Search Based on Fundamental Relations and OWLIM Rules

**Vladimir Alexiev, PhD, PMP**
**Data and Ontology Management Group**
**Ontotext Corp**

Workshop on Semantic Digital Archives (SDA 2012)
Part of TPDL 2012
Paphos, Cyprus, 27 Sep 2012

# Presentation Outline

- Background and significance of CIDOC CRM
- Fundamental Concepts and Relations
- Example: Thing from Place: definition, graphical (network representation), SPARQL query
- Corrections and rationalization of FRs
- Inverses, Transitive properties, no Reflexive closure
- Parallel-Serial networks, decomposing a FR into sub-FRs, implementing with RDFS and OWL
- OWLIM and OWLIM Rules
- FR Implementation, Performance

# Ontotext Cultural Heritage Projects/Clients

- Clients: UK, KR, SE, NL, BG, US

The National Archives  YALE CENTER FOR BRITISH ART

THE BRITISH MUSEUM  de Bibliotheek  ITCC  GÖTEBORGS STADSMUSEUM

- Research projects executed by Ontotext

PrestoSpace  MOLTO  Research SPACE  europeana think culture

- Projects using OWLIM: EU, PL, JP

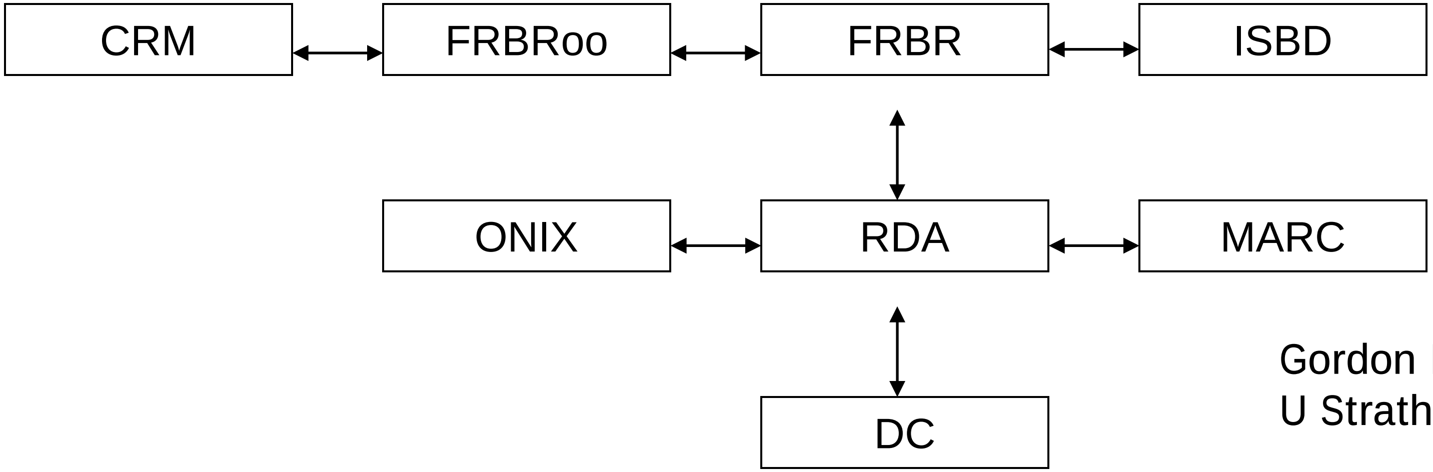
CHARISMA  3D·COFORM Tools & Expertise for 3D Collection Formation  CYFROWE MUZEUM NARODOWE W WARSZAWIE  NII  LODAC

# CIDOC CRM

- Created by International Committee for Documentation (CIDOC) of International Council of Museums (ICOM)
  - More than 10y of development, official standard ISO 21127:2006
  - Available at http://www.cidoc-crm.org/
  - Maintained by CRM SIG, crm-sig@ics.forth.gr
- Provides a common semantic framework to which any CH data can be mapped
  - Intended to promote shared understanding of CH data and a "semantic glue" to mediate between different CH sources
  - Few classes (82) and properties (142); quite expressive because it is abstract
  - Original focus: history, archaeology, cultural heritage (CH)
  - Used in various projects, including libraries, archives, museums

# Importance of CRM

- CIDOC CRM can map and subsume various domain specific standards, thus allowing to compare, unify and inter-map them
  - E.g. influenced LIDO (events), EDM (subjects, events), mapped EAD, mapped UNIMARC, created FRBR as ontology (FRBRoo), etc
- Everything is connected… at the community (human) and technical (Semantic Web) levels



Gordon Dunsire,
U Strathclyde

# Ontotext CRM Experience

- FP7 MOLTO: museum data is based on CRM
  - Multilingual Online Translation. Knowledge infrastructure, interoperability between natural language and structured queries,
  - Museum object descriptions in 15 languages. Gotehnburg Museum case
- ResearchSpace project of the British Museum is based on CRM
  - Advising British Museum and Yale Center for British Art on representing their collections in CRM
- Providing feedback and contributing to RDF definition of CRM
- Implementing CRM search based on Fundamental Relations

# CIDOC CRM SEARCH

# Fundamental Concepts and Relations (FC, FR)

- CRM data is usually represented in semantic web format (RDF) and comprises complex graphs of nodes and properties.
  - How can a user can search through such complex graphs? The number of possible combinations is staggering.
- New Framework for Querying Semantic Networks (FORTH TR419, 2011)
  - "Compresses" the semantic network by mapping many CRM entity classes to a few "Fundamental Concepts" (FC) : **Thing, Place, Actor, Event/Time, Concept/Type**
  - Maps whole networks of CRM properties to fewer "Fundamental Relations" (FR)
  - FC and FRs serve as a "search index" over the CRM semantic web and allow the user to use a simpler query vocabulary.
  - FR categories include: **type, part, from/generator, similar/same, met, refers/about, borders/overlaps, by** and some of their inverses
  - Matrix declares 114 FRs (18 of them very similar) and 18 "specialization FRs" (e.g. Thing **acquired at** Place is specialization/part of Thing **from** Place)
- Fundamental Categories and Relationships for intuitive querying CIDOC-CRM based repositories (FORTH TR-429, Apr 2012, 153 pages)
  - Defines FRs over all combinations of FCs

# FR by FC Matrix

| Domain (select) | Range(query parameter) | | | | |
|---|---|---|---|---|---|
| | Thing | Actor | Place | Event | Time |
| Thing | 8.has met<br>9.refers to or is about<br>10.is referred to by<br>3.has part<br>7.is similar or same with<br>5. from<br>4.is part of<br>was made from | 8.has met<br>5.from<br>9.refers to or is about<br>10.is referred to by<br>12.by<br>Used by<br>Created by<br>Modified by<br>Found or<br>acquired by | 9.refers to<br>10.is referred to at<br>5.from<br>Used at<br>Created at<br>Found or<br>acquired at<br>Was created/produced by<br>person from<br>Is/was located at | 9.refers to<br>10.is referred to by<br>5.from<br>Destroyed in<br>Created in<br>Modified in<br>Used in | 5.from<br>Destroyed on<br>Created on<br>Modified on<br>Used on |
| Actor | 8.has met<br>6.is owner or creator of<br>9. refers to<br>10.is referred by | 4.is member of<br>3.has member<br>8. has met<br>5.has generator<br>6.is generator of<br>9.refers to<br>10.is referred by | 8.has met<br>5.from<br>9.refers to<br>10.is referred to at | 9.refers to<br>10.is referred to by<br>5.from<br>8.has met<br>Brought into existence at<br>Taken out of existence at<br>Performed action at<br>Influenced | 9.refers to<br>5.from<br>8.has met<br>Brought into existence at<br>Taken out of existence at<br>Performed action at<br>Influenced |
| Place | 8.has met<br>6.Is origin of<br>9.refers to or is about<br>10.is referred by | 8.has met<br>6.Is origin of<br>9.refers to or is about<br>10.is referred by<br>8.has met | 4.is part of<br>3.has part<br>11.borders or overlaps with | 9.refers to<br>10.is referred by<br>8.has met | 5.from<br>10.refers to<br>8.has met |
| Event | 6.is origin of<br>10.is referred by<br>9.refers to or is about<br>8.has met<br>created<br>destroyed<br>modified<br>used | 12.by<br>10.is referred by<br>9.refers to or is about<br>8.has met<br>brought into existence<br>took out of existence | 9.refers to or is about<br>10. is referred to at<br>5.from | 9.refers to or is about<br>10.is referred by<br>3.has part<br>5.from | 9.refers to or is about<br>5.from<br>starts<br>ends<br>has duration |

# Thing from Place: A Sample FR

All alternatives through which a Thing's **origin** can be related to Place

a Thing (part of another Thing)* is considered to be "from" Place if it:

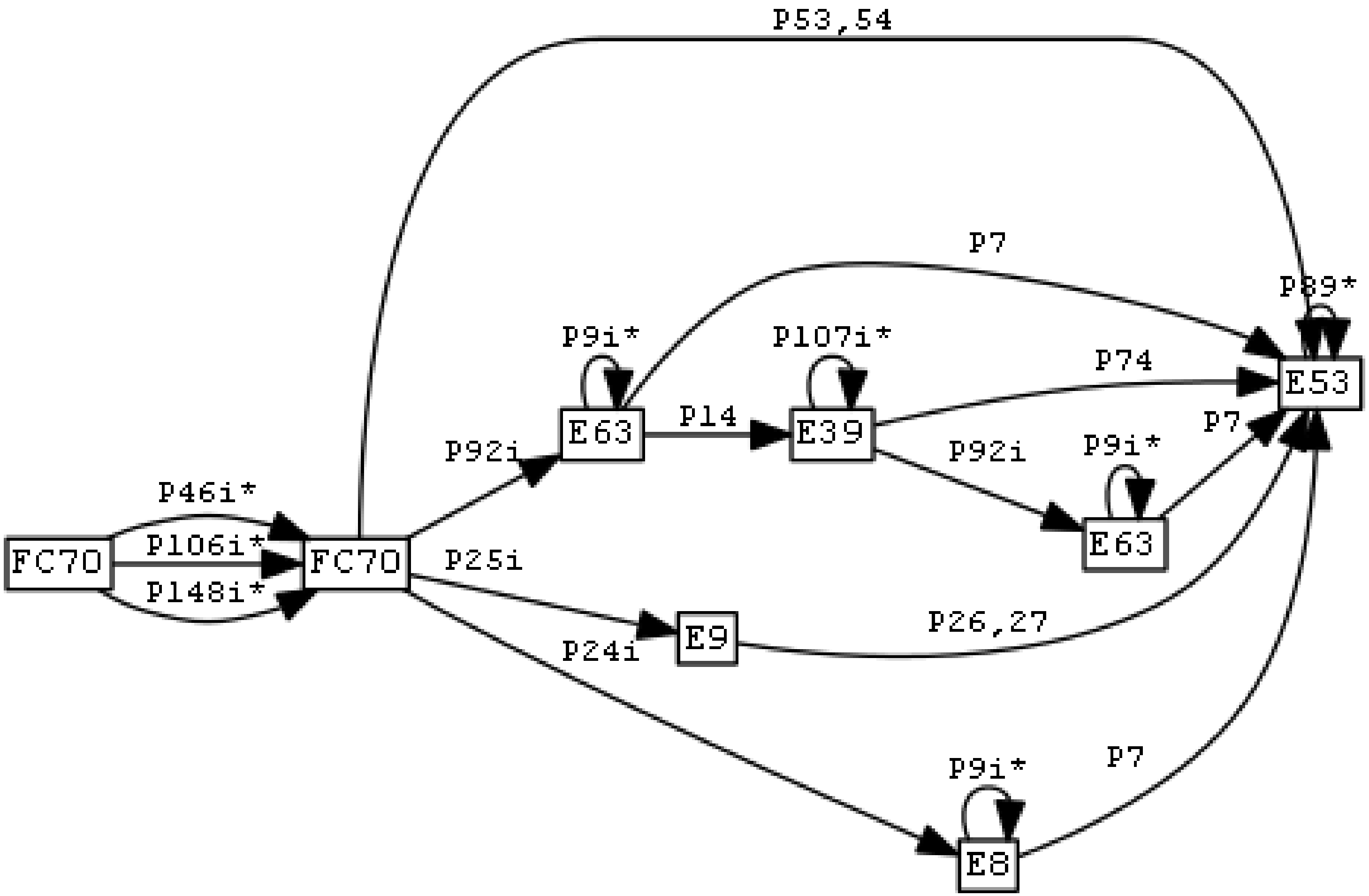
- is formerly or currently located at Place (that falls within another)*
- or was brought into existence (produced/created) by an Event (part of another)*
  - that happened at Place (that falls within another)*
  - or was carried out by an Actor (who is member of a Group)*
    - who formerly or currently has residence at Place (that falls within another)*
    - or was brought into existence (born/formed) by an Event (part of another)* that happened at Place (that falls within another)*
- or was Moved to/from a Place (that falls within another)*
- or changed ownership through an Acquisition (part of another)*
  - that happened at Place (that falls within another)*

# Thing from Place: Definition (CRM Classes & Properties)

```
FC70_Thing --(P46i_forms_part_of* | P106i_forms_part_of* | P148i_is_component_of*)-> FC70_Thing:
 {FC70_Thing --(P53_has_former_or_current_location | P54_has_current_permanent_location)-> E53_Place:
  {E53_Place --P89_falls_within*-> E53_Place}
 OR FC70_Thing --P92i_was_brought_into_existence_by-> E63_Beginning_of_Existence:
  {E63_Beginning_of_Existence --P9i_forms_part_of*-> E5_Event:
   {E5_Event --P7_took_place_at-> E53_Place:
    {E53_Place --P89_falls_within*-> E53_Place}
   OR E7_Activity --P14_carried_out_by-> E39_Actor:
    {E39_Actor --P107i_is_current_or_former_member_of* -> E39_Actor:
     {E39_Actor --P74_has_current_or_former_residence  -> E53_Place:
      {E53_Place --P89_falls_within*-> E53_Place}
     OR E39_Actor --P92i_was_brought_into_existence_by-> E63_Beginning_of_Existence:
      {E63_Beginning_of_Existence --P9i_forms_part_of*-> E5_Event:
       {E5_Event --P7_took_place_at-> E53_Place:
        {E53_Place --P89_falls_within* -> E53_Place}}}}}}}
 OR E19_Physical_Thing  --P25i_moved_by-> E9_Move:
  {E9_Move --(P26_moved_to | P27_moved_from)-> E53_Place:
   {E53_Place  --P89_falls_within*-> E53_Place}}
 OR E19_Physical_Object --P24i_changed_ownership_through-> E8_Acquisition:
  {E8_Acquisition --P9i_forms_part_of*-> E5_Event:
   {E5_Event --P7_took_place_at-> E53_Place:
    {E53_Place --P89_falls_within*-> E53_Place}}}}
```

# Thing from Place: Graphical Representation

- Although defined as a **tree** of property paths, the FR is better depicted as a **network** through a simple merge of leaf-level nodes



# Thing from Place: SPARQL Query

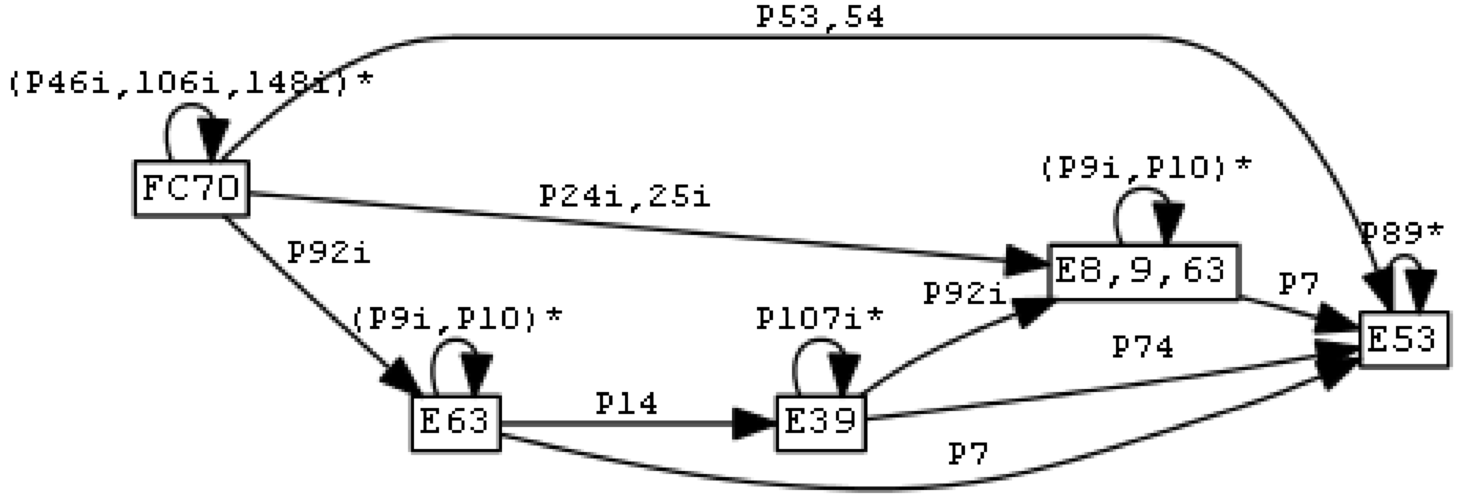
```
select ?t ?p2 {
?t a FC70_Thing. ?t (P46i_forms_part_of* | P106i_forms_part_of* | P148i_is_component_of*) ?t1.
  {?t1 (P53_has_former_or_current_location | P54_has_current_permanent_location) ?p1}
  UNION
  {?t1 P92i_was_brought_into_existence_by ?e1. ?e1 P9i_forms_part_of* ?e2.
      {?e2 P7_took_place_at ?p1}
      UNION
      {?e2 P14_carried_out_by ?a1.
        ?a1 P107i_is_current_or_former_member_of* ?a2.
          {?a2 P74_has_current_or_former_residence ?p1}
          UNION
          {?a2 P92i_was_brought_into_existence_by ?e3. ?e3 P9i_forms_part_of* ?e4.
           ?e4 P7_took_place_at ?p1}}}
  UNION
  {?t2 P25i_moved_by ?e5. ?e5 (P26_moved_to | P27_moved_from) ?p1}
  UNION
  {?t2 P24i_changed_ownership_through ?e6.
    ?e6 P9i_forms_part_of ?e7. ?e7 P7_took_place_at ?p1}.
?p1 P89_falls_within* ?p2}
```

- This query is very complex and expensive, especially when you need to combine with other FRs into composite queries

# Thing from Place: Corrections and Rationalization

- Allowed paths of mixed properties (e.g. P46i,P106i) at the beginning
- Allowed a loop P9i* at E9 (Move forms part of a bigger event) by merging the nodes E8, E9, and the second E63
- Allowed P10_falls_within in addition to P9i_forms_part_of (after consultation with the original authors)
- Skipped P26,P27: they are subproperties of  P7, so it's enough to check for P7

Ø Simpler than the original,  but still quite complex



# Inverses, Transitive properties

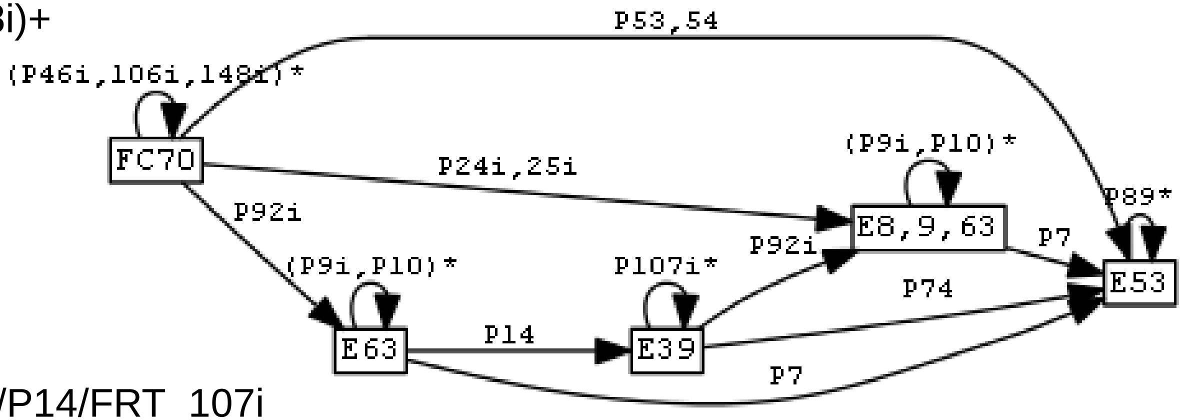
- Most CRM properties have inverse (symmetric properties are their own inverse)
  - FRs use CRM properties in both directions: forward (e.g. P53_has_former_or_current_location) and inverse (P24i_changed_ownership_through)
  - It's useful to rely on owl:inverseOf inferencing
- FRs use transitive closure to traverse "part" hierarchies
  - CRM has physical object parts, conceptual object parts, sub-places, sub-events
  - CRM scope notes suggest 14 properties (and inverses) should be transitive: P9 P10 P46 P86 P88 P89 P106 P114 P115 P116 P117 P120 P127 P148.
  - In addition to these "atomic" properties, disjunctions of properties often also need to be declared as transitive.
  - It's useful to rely on owl:TransitiveProperty inferencing.

# No Reflexive Closure; Parallel-Serial Networks

- FRs often use reflexive-transitive closure (0 repetitions)
  - E.g. Thing from Place: can relate directly to a place, or to any of its super-places
  - We have opted **not** to use reflexive closure in the implementation, since it would generate a lot of trivial facts (self-loops).
  - We use disjunction instead: the iterated property is applied 0 times in the first disjunct, and *n* times in the second
- FRs are defined *mostly* as parallel-serial networks of properties
  - Can be seen from the SPARQL Property Path constructs and is explained below

# Decomposing Thing from Place into sub-FRs

```
  # self-loops and simple disjunctions
FRT_46i_106i_148i := (P46i|P106i|P148i)+
FRT_9i_10 := (P9|P10)+
FRT_107i := P107i+
FRT_89 := P89+
FRX_53_54 := (P53|P54)
FRX_24i_25i := (P24i|P25i)
   # growing fragments
FRX_92i := P92i | P92i/FRT_9i_10
FRX_92i_14 := FRX_92i/P14 | FRX_92i/P14/FRT_107i
FRX_FC70_E8_9_63 := FRX_92i_14/P92i | FRX_24i_25i
FRX_FC70_E8_9_63_P7 := FRX_FC70_E8_9_63/P7 | FRX_FC70_E8_9_63/FRT_9i_10/P7
FRX7 := FRX_53_54 | FRX_FC70_E8_9_63_P7 | FRX_92i_14/P74 | FRX_92i/P7
FRX7_P89 := FRX7 | FRX7/FRT_89
FR7 := FRX7_P89 | FRT_46i_106i_148i/FRX7_P89
```



- "Sub-FRs" are auxiliary relations used to build up the final FR
- The numbering comes from CRM property and entity names
- Prefixes: FR: final result, FRT: transitive, FRX: non-transitive, FC70 or E: from/to that class

# Implementing Parallel-Serial with RDFS and OWL

| Pattern | Construct | Implementation |
|---|---|---|
| inverse | prop := ^prop1 | prop1 owl:inverseOf prop2. |
| parallel | prop := prop1\|prop2 | prop1 rdfs:subPropertyOf prop.<br>prop2 rdfs:subPropertyOf prop. |
| serial | prop := prop1/prop2 | prop owl:PropertyChainAxiom (prop1 prop2). |
| transitive | prop := prop1+ | prop1 rdfs:subPropertyOf prop.<br>prop owl:TransitiveProperty |
| reflexive-transitive | prop := prop1 prop2* | Converted to the following:<br>prop := prop1 \| (prop1/prop2+) |

- 3 RDFS and OWL constructs are sufficient to implement parallel-serial networks: subPropertyOf, TransitiveProperty, PropertyChainAxiom
  - In OWLIM, they are implemented using Rules
- So can't we stick to these constructs and not use OWLIM Rules at the application level?

# Type Checking and Conjunctive Properties

- The original FR definition supposes type checks for every node (FC70, E63…), e.g.:
  ?x FR7_from_place ?y := ?x a FC70_Thing;  ?x FR7 ?y; ?y a E53_Place.
- In many cases type checks can be skipped since they are implied by property ranges (e.g. P53 P54 P7 P47 P89 imply E53)
- In other cases type checks are required in the middle of a network. E.g. "Thing about X" is a family of FRs, where X is Thing, Place, Actor, Event
- For this we'd need conjunctive properties, which are not part of OWL2
  - OWL RL can be extended with role conjunctions without restrictions or increase in complexity
  - There is a proposal to include conjunctive properties in OWL 3

# OWLIM

- A commercial semantic repository by Ontotext
  - Incremental assert **and** retract
  - High-performance: fully-materializing, replication cluster, strong benchmark results, good concurrent query response, cloud deployment
- Used in some landmark semweb projects
  - Runs BBC Sports, World Cup 2010 and the Olympics 2012
  - linkedlifedata.com semantic warehouse used by top-20 pharmaceuticals
- Quite a following in cultural heritage
  - The National Archives, The British Museum, Yale Center for British Art
  - FP7: 3D COFORM, CHARISMA, MOLTO
  - LOD.AC, Polish Digital National Museum

# OWLIM Rules

- Allow simple unification and in/equality constraints
  - OWLIM implements OWL2 QL and RL using these rules
  - Custom rules are treated just like OWL (system) rules
  - E.g. sub-property, transitive, inverse reasoning:
    x p1 y; p2 <rdfs:subPropertyOf> p2 [Constraint p1!=p2] => x p2 y
    p <rdf:type> <owl:TransitiveProperty>; x p y; y p z => x p z
    p1 <owl:inverseOf> p2; x p1 y => y p2 x
    p1 <owl:inverseOf> p2; x p2 y => y p1 x
- Advantages:
  - Speed: forward-chaining & full materialization (translated to Java bytecode for speed ), so query answering is very fast
  - "Reversible": when a triple is retracted, all consequences with no other support are retracted
- Disadvantages
  - Inflexible: if rules are changed, the repository needs to be reloaded.
    (Better implement generic rules that work on TBox assertions about properties.)
  - Proprietary to OWLIM
    (Ontotext is considering proposed standard rule languages in future versions)
  - Don't support real negation (e.g. instance is **not** of a given class or its super-classes)

# FR Implementation

- Once the FR is decomposed to sub-FRs, implementation is straightforward. E.g. this sub-FR is implemented as:
FRT_46i_106i_148i := (P46i|P106i|P148i)+

```
x <crm:P46i_forms_part_of> y => x <rso:FRT_46i_106i_148i> y
x <crm:P106i_forms_part_of> y => x <rso:FRT_46i_106i_148i> y
x <crm:P148i_is_component_of> y => x <rso:FRT_46i_106i_148i> y
<rso:FRT_46i_106i_148i> <rdf:type> <owl:TransitiveProperty>
```

  - Important to extract common sub-FRs between FRs, to facilitate reuse
- We implemented 11 FRs of Thing:
  - refers to or is about Place; from Place; is/was located in Place
  - has met Actor; by Actor
  - refers to or is about Event; has met Event
  - is made of Material; is/has Type; used technique; identified by Identifier
- Use 44 CRM properties. Took 86 rules, 10 axioms, 26 sub-FRs

# Bug in "Thing has met Event"

- Acquisition
  - Often modeled as E8_Acquisition (changes owner), E10_Transfer_of_Custody (changes keeper), E80_Part_Removal (removes object from old collection), E79_Part_Addition (adds object to new collection)
  - An event at which meet: object, buyer, seller, old collection, new collection
  - Object (E22_Man-Made_Object) is P46i_forms_part_of old collection before acquisition (E78_Collection) and new collection after acquisition (E78_Collection)
- FC70_Thing --FR12_was_present_at-> E5_Event :=

```
FC70_Thing --(P46i_forms_part_of | P106i_forms_part_of | P148i_is_component_of)* ->
   FC70_Thing --P12i_was_present_at-> E5_Event:
      E5_Event --P9i_forms_part_of*-> E5_Event
```

- Causes all objects in a collection to have met (witnessed) the addition of all other objects in the collection!
  - For new objects: logically impossible. For old objects: useless
  - Quadratic growth of data, exponential slowdown of data loading
  - BM has 1.5M objects in its collection, so the slowdown is unbearable

# How did this bug make me feel?

- Took a couple of hours of debugging triples to diagnose

Ø Inference is powerful, but may expose unintended consequences

- *Karakondjul (Greek and Bulgarian): poltergeist, house troll*

# Performance

- A concern was expressed that materializing sub-FR triples may increase the repository size too much and slow it down?
- Small repository of RKD data
  - 11 Rembrandt paintings: 1.5M triples, including 0.5M object triples (complex data about each painting, researches, documents, etc) and 1M thesaurus triples (people, places, etc)
  - FRs added only 25.8k triples, which is 1.7% of the total data or 5.1% of the object data à no perceptible slowdown
- Medium repository of BM data
  - Over 150k BM objects, about 20M triples
  - FR searches show no noticeable slow-down
  - Pending: all 1.5M BM objects
- OWLIM performs well on 10s B triples
  - Examples: linkedlifedata.com (public), The National Archives, BBC
  - So increases in the number of triples up to 50% are trivial
- Compare the raw SPARQL query on slide 13

# Thanks for your attention!

- Questions/Discussion

- Contact: vladimir.alexiev@ontotext.com