Key Things You Need to Know About RDF

and Why They Are Important

David Booth, Ph.D.
HRG & Rancho BioSciences
david@dbooth.org

Smart Data Conference
18-Aug-2015

Latest version of these slides:
http://dbooth.org/2015/key/

RDF is

fundamentally different

from other data formats – XML, JSON, etc.

This presentation explains why.

But first, some background . . .

Comparing RDF with XML or JSON

WARNING: Improper comparison!

- XML, JSON or any other format could be used in special ways to achieve all of RDF's features
  - But that isn't how they are normally used
- This talk compares RDF with XML and JSON *as they are normally used*

What is RDF?

- "Resource Description Framework"
  - *But think "Reusable Data Framework"*
- Language for representing information
- International standard by W3C
- Mature: 10+ years
- Used in many domains, including biomedical and pharma

RDF Assertions (a/k/a "Triples")

PREFIX ex: <http://.../data/>
PREFIX v: <http://.../vocab/>

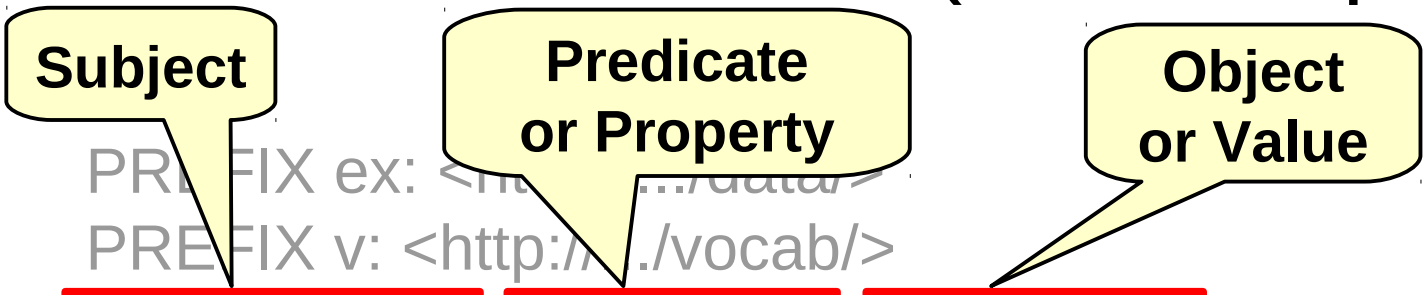
ex:patient319 v:fullName "John Doe" .

ex:patient319 v:systolicBP ex:obs_001 .

ex:obs_001 v:value 120 .

ex:obs_001 v:units v:mmHg .

RDF Assertions (a/k/a "Triples")




```
PREFIX ex: <http:.../data/>
PREFIX v: <http://.../vocab/>
ex:patient319 v:fullName "John Doe".
ex:patient319 v:systolicBP ex:obs_001 .
ex:obs_001 v:value 120 .
ex:obs_001 v:units v:mmHg .
```

RDF Assertions (a/k/a "Triples")

Equivalent English sentence:
Patient319 has full name "John Doe".

PREFIX ex: <http://
PREFIX v: <http://.../voca

ex:patient319 v:fullName "John Doe" .

ex:patient319 v:systolicBP ex:obs_001 .

ex:obs_001 v:value 120 .

ex:obs_001 v:units v:mmHg .

RDF Assertions (a/k/a "Triples")

PREFIX ex: <http://...

PREFIX v: <http://...

ex:patient319 v:fullName "...n Doe" .

ex:patient319 v:systolicBP ex:obs_001 .

ex:obs_001 v:value 120 .

ex:obs_001 v:units v:mmHg .

Equivalent English sentence:
Patient319 has a systolic blood pressure observation obs_001.

RDF Assertions (a/k/a "Triples")

PREFIX ex: <http://...

PREFIX v: <http://...

ex:patient319 v:fullN...

ex:patient319 v:systolicBP ...obs_001 .

ex:obs_001 v:value 120 .

ex:obs_001 v:units v:mmHg .

Equivalent English sentence:
Obs_001 has a value of 120.

RDF Assertions (a/k/a "Triples")

PREFIX ex: <http://.../data/>
PREFIX v: <http://.../vocab/>
ex:patient319 v:fullN
ex:patient319 v:syst
ex:obs_001 v:value 120 .
ex:obs_001 v:units v:mmHg .

Equivalent English sentence:
Obs_001 has units of mmHg.

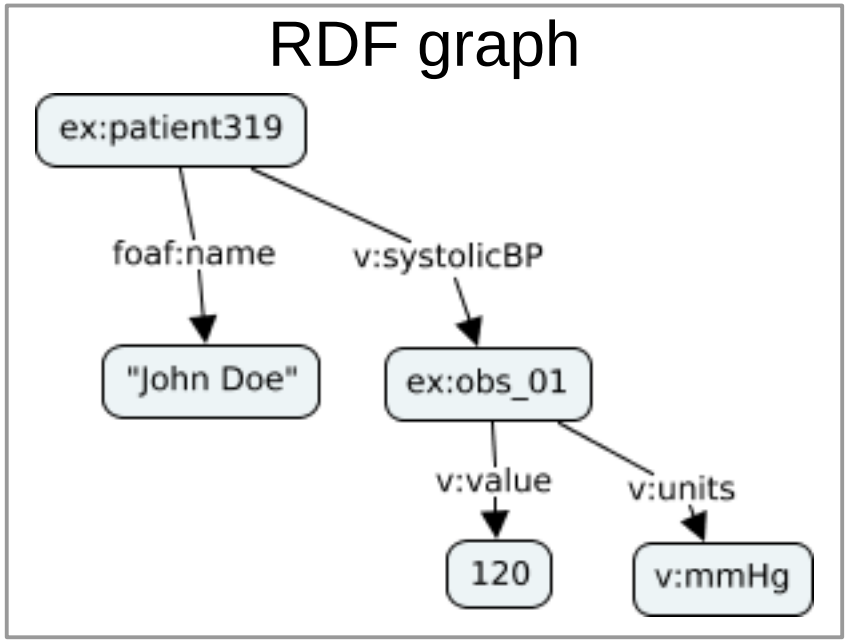
RDF Assertions (a/k/a "Triples")

```
PREFIX ex: <http://.../data/>
PREFIX v: <http://.../vocab/>
ex:patient319 v:fullName "John Doe" .
ex:patient319 v:systolicBP ex:obs_001 .
ex:obs_001 v:value 120 .
ex:obs_001 v:units v:mmHg .
```

Sets of assertions form an RDF graph . . .

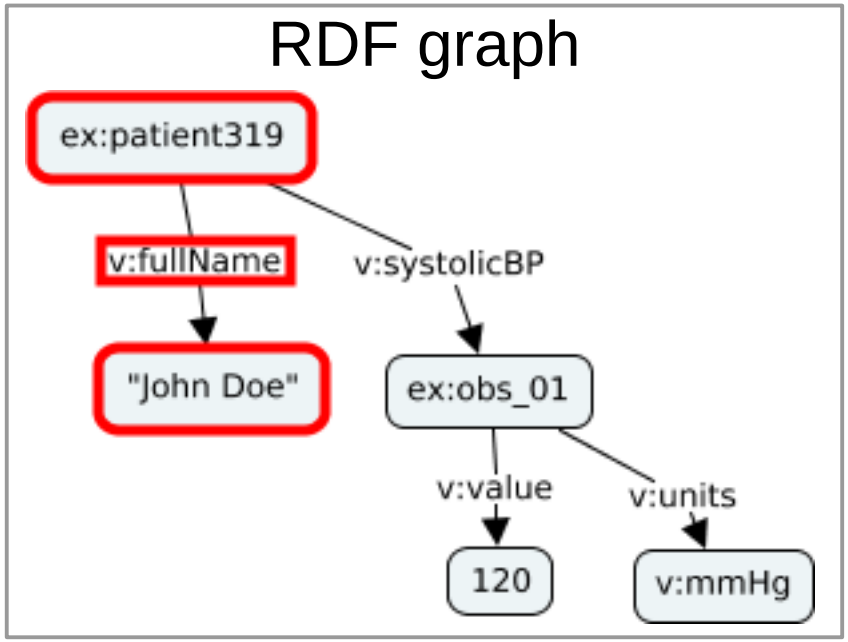
RDF Graph

PREFIX ex: <http://.../data/>
PREFIX v: <http://.../vocab/>
ex:patient319 v:fullName "John Doe" .
ex:patient319 v:systolicBP ex:obs_001 .
ex:obs_001 v:value 120 .
ex:obs_001 v:units v:mmHg .



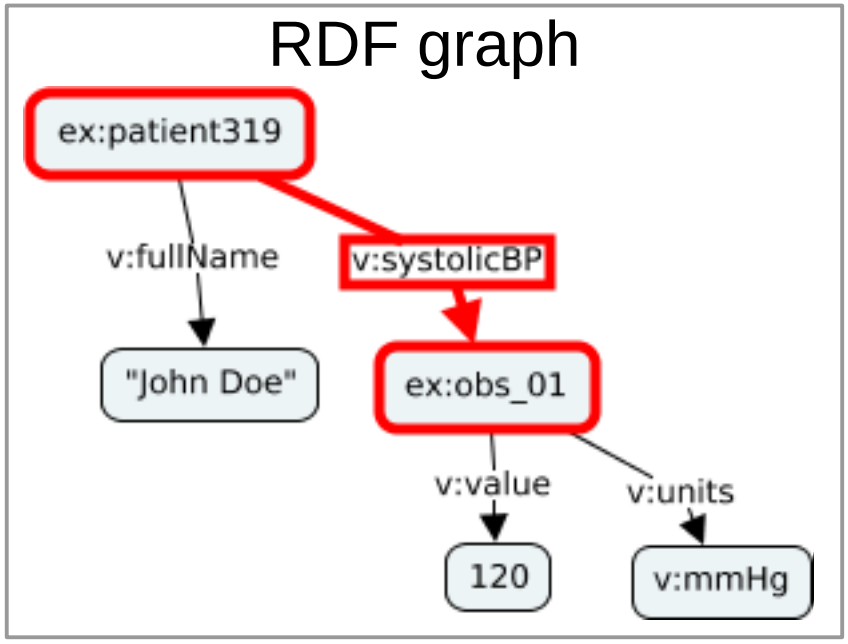
RDF Graph

```
PREFIX ex: <http://.../data/>
PREFIX v: <http://.../vocab/>
ex:patient319 v:fullName "John Doe" .
ex:patient319 v:systolicBP ex:obs_001 .
ex:obs_001 v:value 120 .
ex:obs_001 v:units v:mmHg .
```



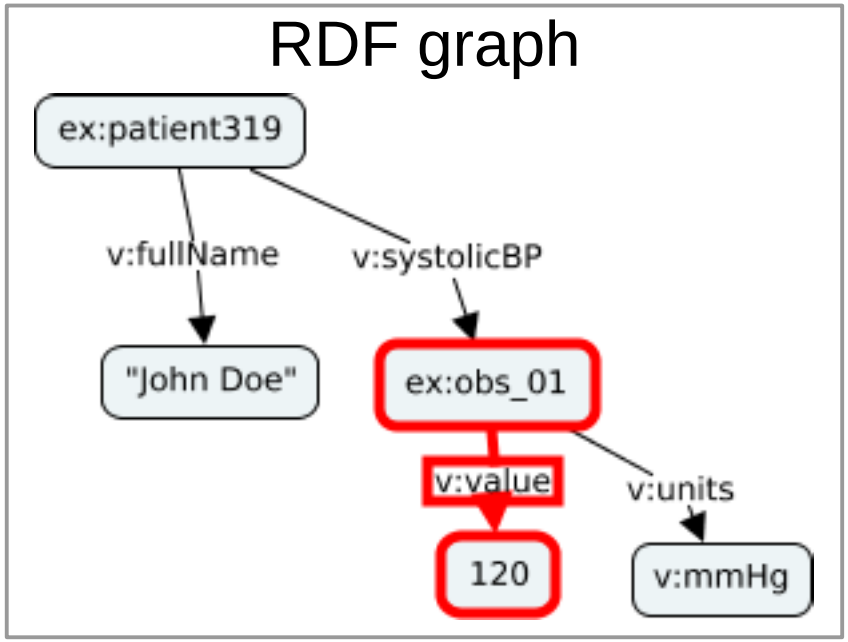
RDF Graph

```
PREFIX ex: <http://.../data/>
PREFIX v: <http://.../vocab/>
ex:patient319 v:fullName "John Doe" .
ex:patient319 v:systolicBP ex:obs_001 .
ex:obs_001 v:value 120 .
ex:obs_001 v:units v:mmHg .
```



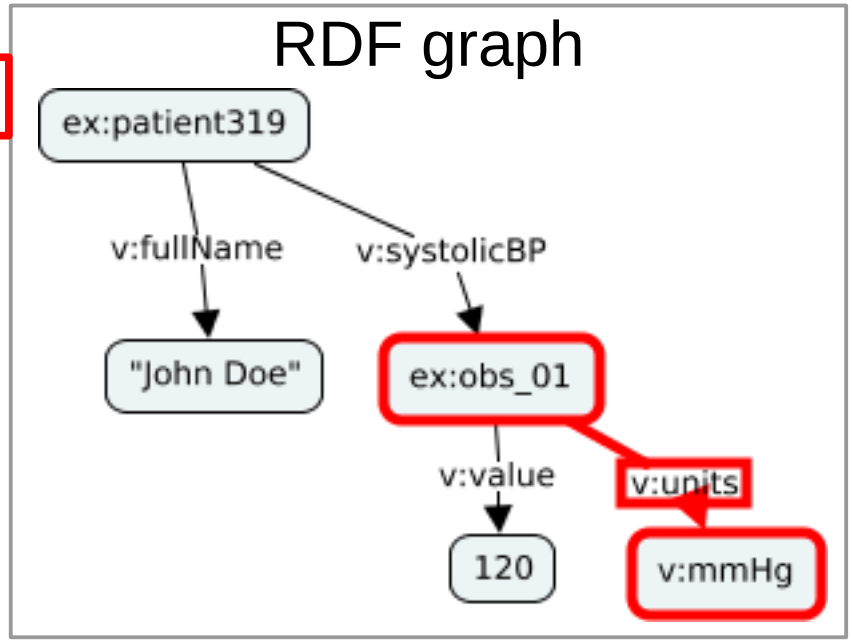
RDF Graph

PREFIX ex: <http://.../data/>
PREFIX v: <http://.../vocab/>
ex:patient319 v:fullName "John Doe" .
ex:patient319 v:systolicBP ex:obs_001 .
ex:obs_001 v:value 120 .
ex:obs_001 v:units v:mmHg .



RDF Graph

PREFIX ex: <http://.../data/>
PREFIX v: <http://.../vocab/>
ex:patient319 v:fullName "John Doe" .
ex:patient319 v:systolicBP ex:obs_001 .
ex:obs_001 v:value 120 .
ex:obs_001 v:units v:mmHg .



What is RDF good for?

- Large-scale information integration
- Semantically connecting diverse data models and vocabularies
- Translating between data models and vocabularies
- Smarter data use

Let's see why . . .

Key things you need to know about RDF

#5: RDF is self describing

- RDF uses URIs as identifiers

#4: RDF is easy to map from other data representations

- RDF data is made of assertions

#3: RDF captures information – not syntax

- RDF is format independent

#2: Multiple data models and vocabularies can be easily combined and interrelated

- RDF is multi-schema friendly

#1: RDF enables smarter data use and automated data translation

- RDF enables inference

#5: RDF is self describing

- Uses URIs as identifiers

http://www.drugbank.ca/drugs/DB00945

#5: RDF is self describing

- Uses URIs as identifiers

http://www.drugbank.ca/drugs/DB00945

drugbank: DB00945

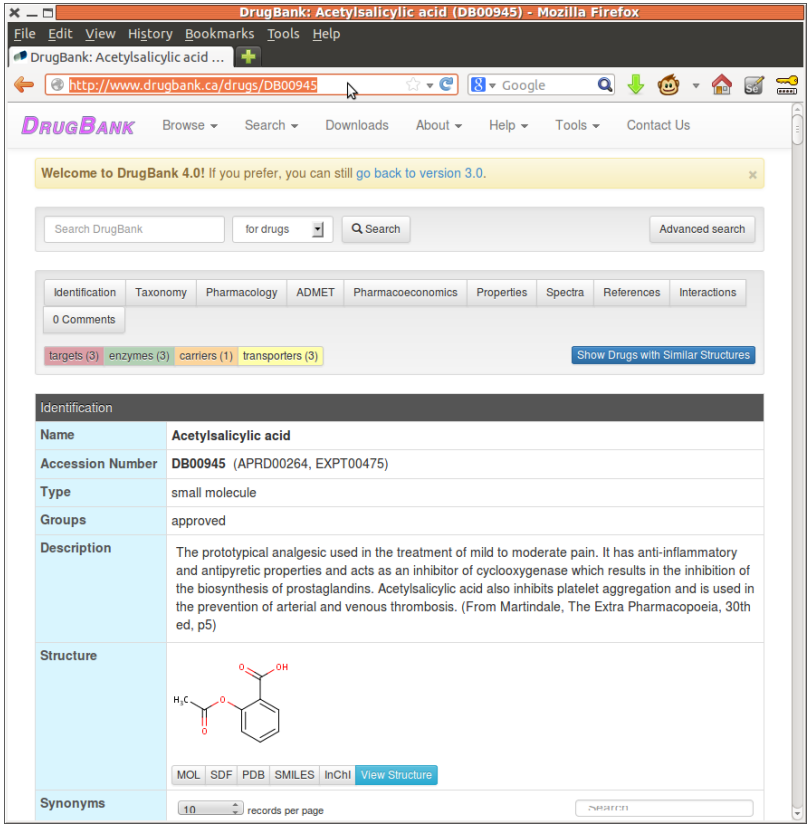
Often abbreviated in RDF:

PREFIX drugbank: <http://www.drugbank.ca/drugs/>
drugbank:DB00945 . . . .

#5: RDF is self describing

- Uses URIs as identifiers

http://www.drugbank.ca/drugs/DB00945



Why is this important?

- Terms, data models, vocabularies, etc., can be linked to definitions
- Definition can be found by any party
  - Reduces ambiguity
- Aids in bootstrapping new terms toward standardization

Supports standards and innovation

Terms are **self describing**?

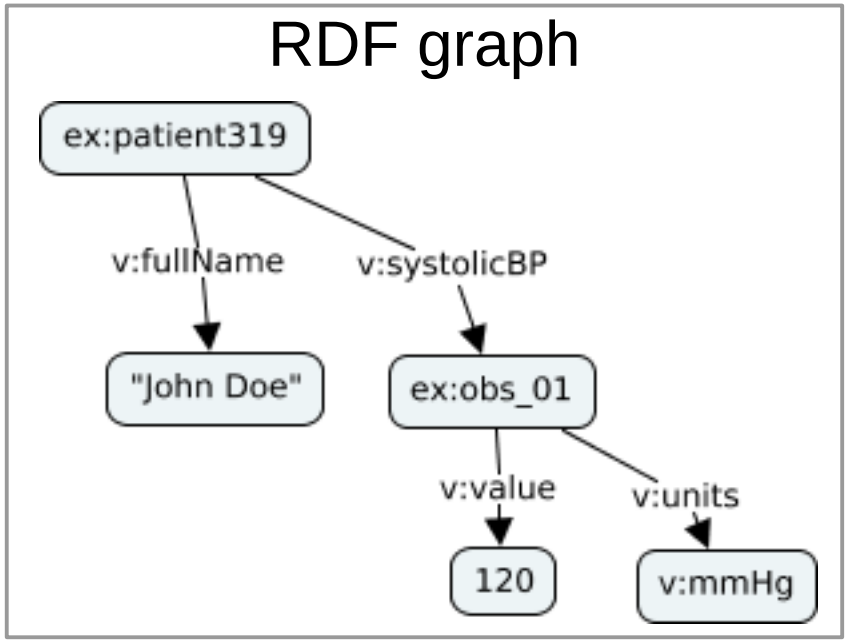
- XML:
  - Can be just as good as RDF if namespaces are properly used
  - In practice, namespaces are not always used or clickable to definitions
- JSON:
  - In theory, could be used like RDF
  - In practice, almost never done

#4: RDF is easy to map from other data representations

- RDF represents information as triples
- Triples form a graph

RDF Graph

PREFIX ex: <http://.../data/>
PREFIX v: <http://.../vocab/>
ex:patient319 v:fullName "John Doe" .
ex:patient319 v:systolicBP ex:obs_001 .
ex:obs_001 v:value 120 .
ex:obs_001 v:units v:mmHg .

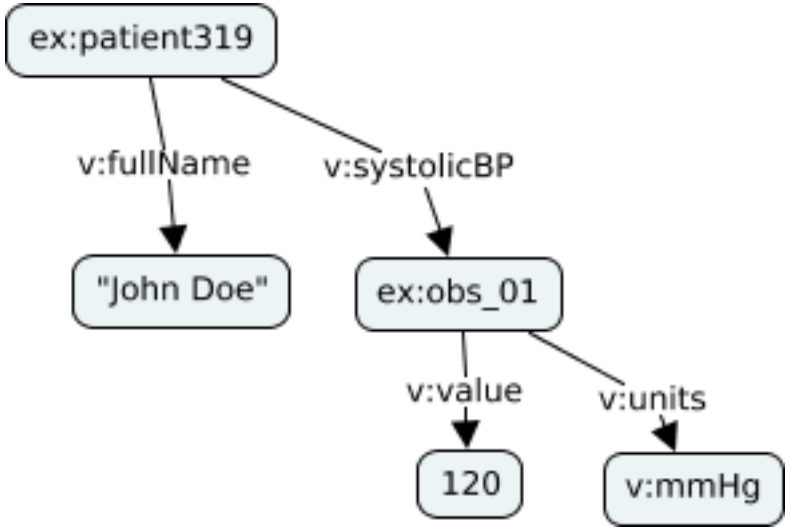


Why does this matter?

- Easy to represent any data model
  - Hierarchical, relational, graph, etc.
- Easy to map any data format to RDF
  - E.g., XML, JSON, CSV, SQL tables, etc.

Great for data integration!

Hierarchical data model in RDF



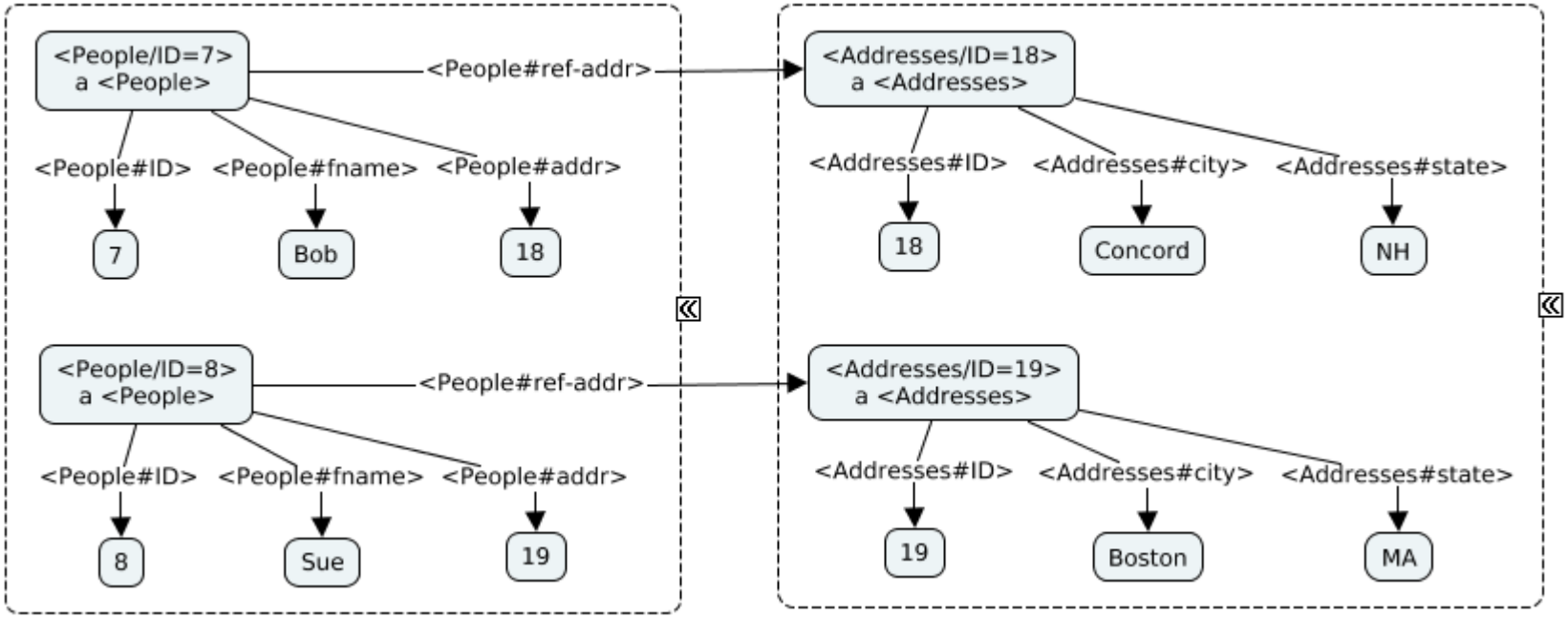
Relational data model in RDF

People

ID	fname	addr
7	Bob	18
8	Sue	19

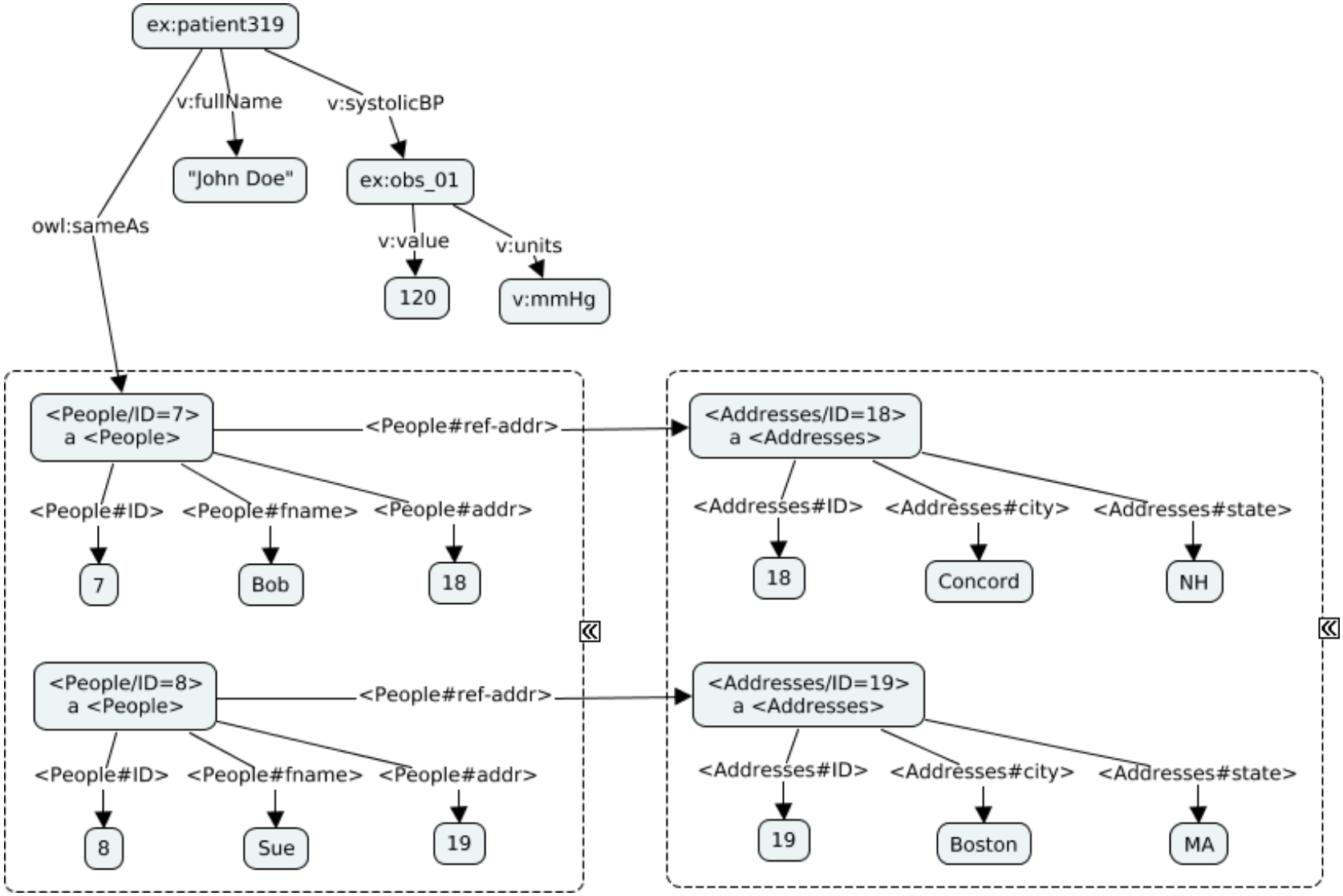
Addresses

ID	City	State
18	Concord	NH
19	Boston	MA

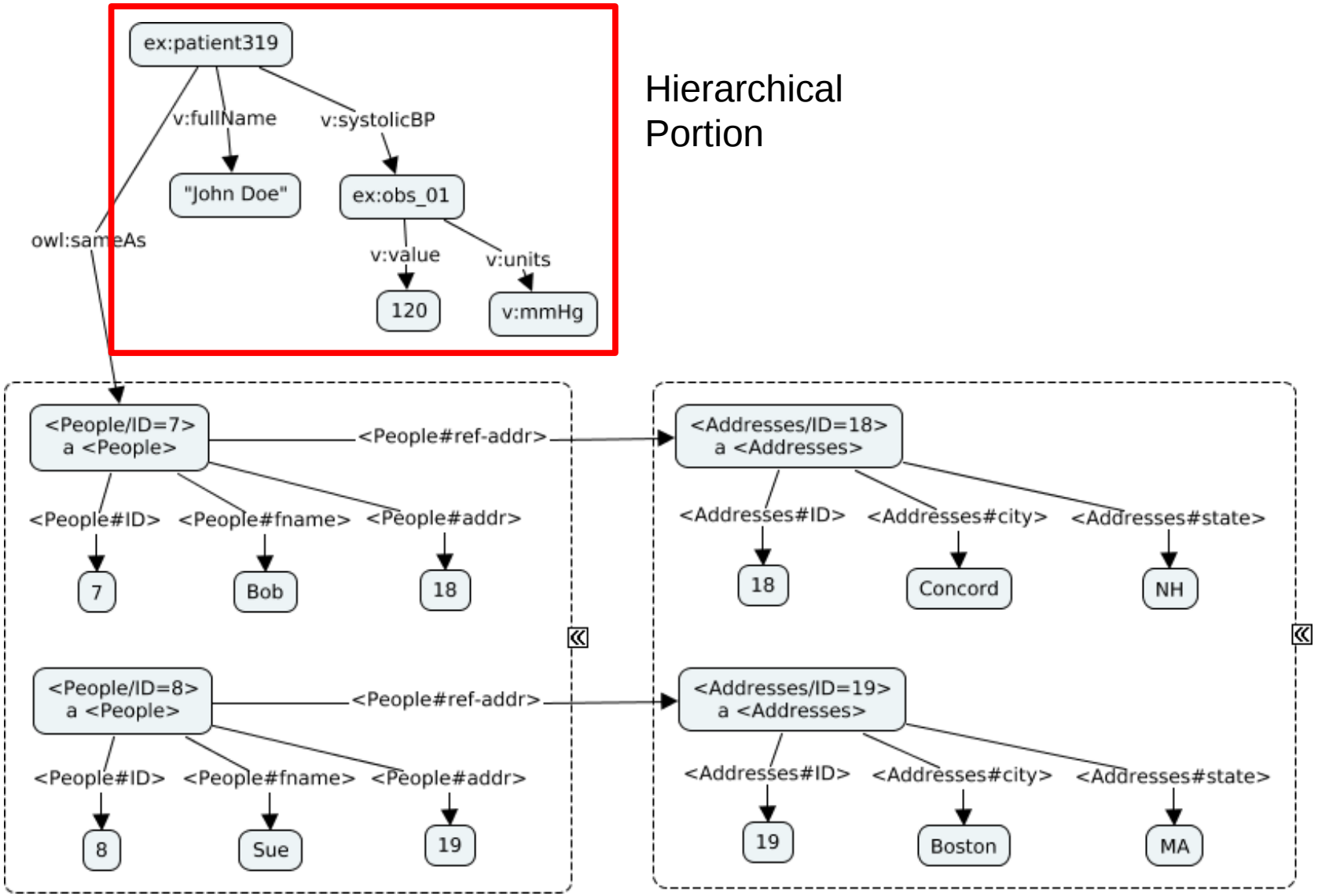


See W3C Direct Mapping of Relational Data to RDF:
http://www.w3.org/TR/rdb-direct-mapping/

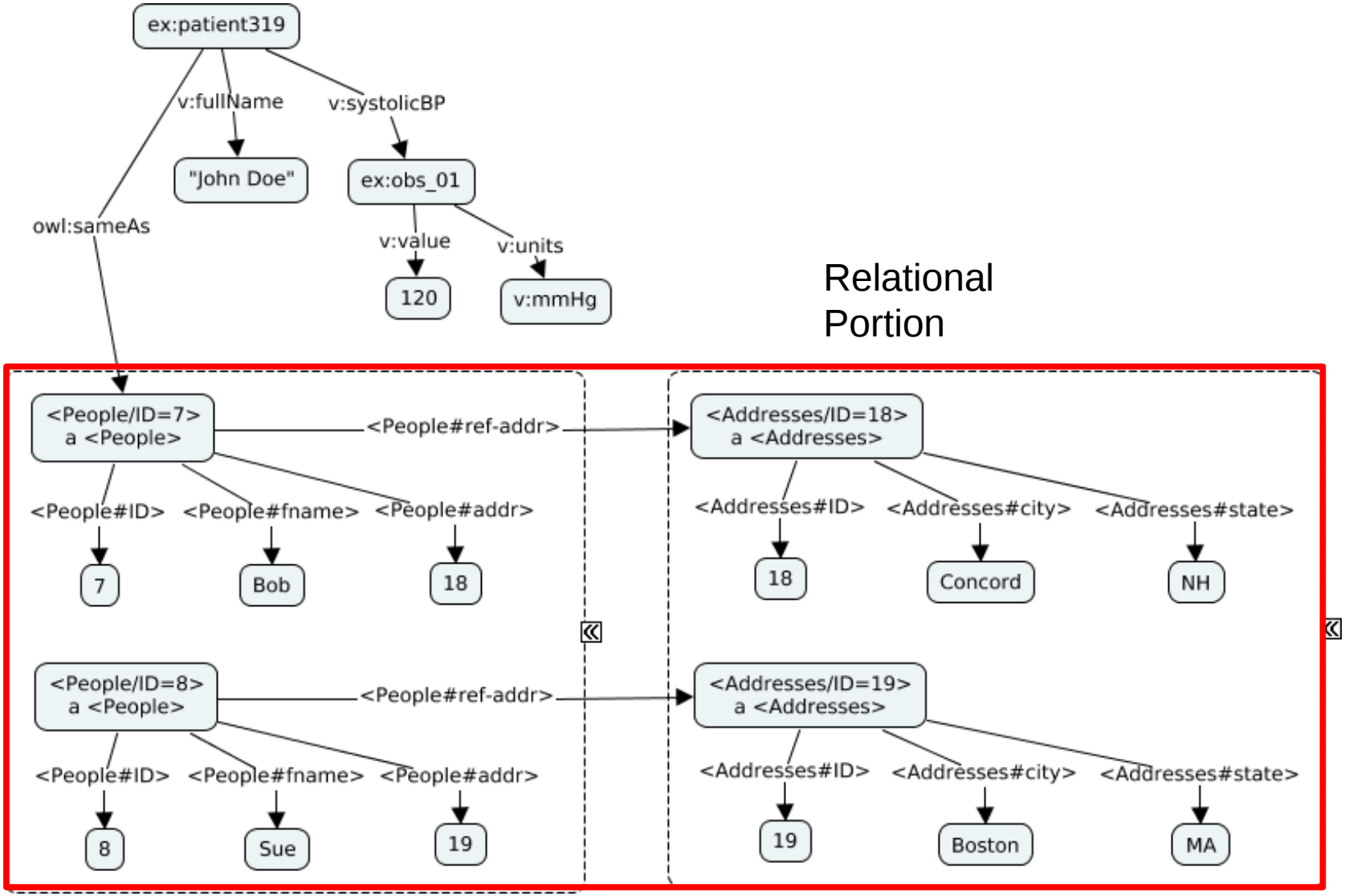
Combined: Hierarchical + Relational



Combined: Hierarchical + Relational



Combined: Hierarchical + Relational



Easy to map from other formats?

- XML:
  - Graphs are possible but messy

½

- JSON:
  - Except cyclic graphs

✓

#3: RDF captures information – not syntax

- RDF is format independent
- There are multiple RDF syntaxes: Turtle, N-Triples, JSON-LD, RDF/XML, etc.
- The same information can be written in different formats
- Any data format can be mapped to RDF

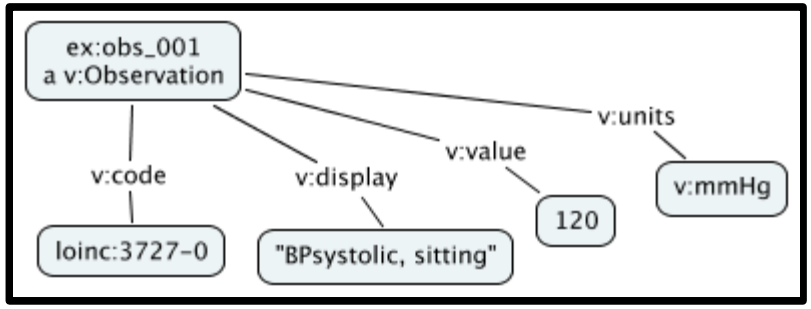
RDF examples

RDF (Turtle)

```
@prefix ex: <http://example/ex/> .
@prefix loinc: <http://loinc.org/> .
@prefix v: <http://example/v/> .

ex:obs_001 a v:Observation ;
 v:code loinc:3727-0 ;
 v:display "BPsystolic, sitting" ;
 v:value 120 ;
 v:units v:mmHg .
```

RDF graph



RDF (N-Triples)

```
<http://example/ex/obs_001> <http://www.w3.org/1999/02/22-rdf-syntax-ns#type> <http://example/v/Observation> .
<http://example/ex/obs_001> <http://example/v/code> <http://loinc.org/3727-0> .
<http://example/ex/obs_001> <http://example/v/display> "BPsystolic, sitting" .
<http://example/ex/obs_001> <http://example/v/value> "120"^^<http://www.w3.org/2001/XMLSchema#integer> .
<http://example/ex/obs_001> <http://example/v/units> <http://example/v/mmHg> .
```

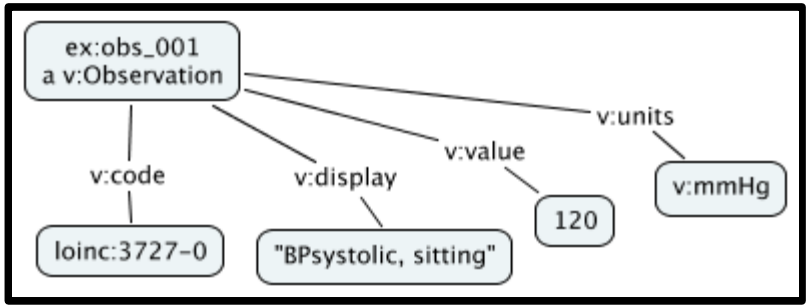
Same information!

RDF examples

RDF (JSON-LD)

```
{
  "@id": "http://example/ex/obs_001",
  "@type": "http://example/v/Observation",
  "http://example/v/code": {
    "@id": "http://loinc.org/3727-0"
  },
  "http://example/v/display": "BPsystolic, sitting",
  "http://example/v/units": {
    "@id": "http://example/v/mmHg"
  },
  "http://example/v/value": 120
}
```

RDF graph



RDF (RDF/XML)

```
<?xml version="1.0" encoding="utf-8"?>
<rdf:RDF xmlns:ex="http://example/ex/" xmlns:loinc="http://loinc.org/"
      xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#" xmlns:v="http://example/v/">
  <rdf:Description rdf:about="http://example/ex/obs_001">
    <rdf:type rdf:resource="http://example/v/Observation"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://example/ex/obs_001">
    <v:code rdf:resource="http://loinc.org/3727-0"/>
  </rdf:Description>
  <rdf:Description rdf:about="http://example/ex/obs_001">
    <v:display>BPsystolic, sitting</v:display>
  </rdf:Description>
  <rdf:Description rdf:about="http://example/ex/obs_001">
    <v:value rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">120</v:value>
  </rdf:Description>
  <rdf:Description rdf:about="http://example/ex/obs_001">
    <v:units rdf:resource="http://example/v/mmHg"/>
  </rdf:Description>
</rdf:RDF>
```

Same info!

Different source formats, same RDF

HL7 v2.x

```
OBX|1|CE|3727-0^BPsystolic,
sitting||120||mmHg|
```

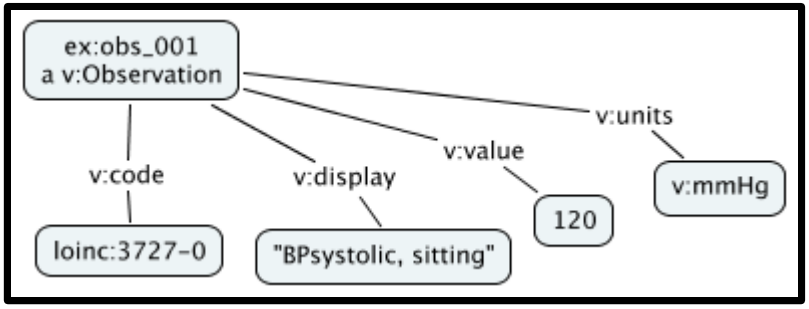
Maps to

FHIR

```
<Observation
      xmlns="http://hl7.org/fhir">
   <system value="http://loinc.org"/>
   <code value="3727-0"/>
   <display value="BPsystolic, sitting"/>
   <value value="120"/>
   <units value="mmHg"/>
</Observation>
```

Maps to

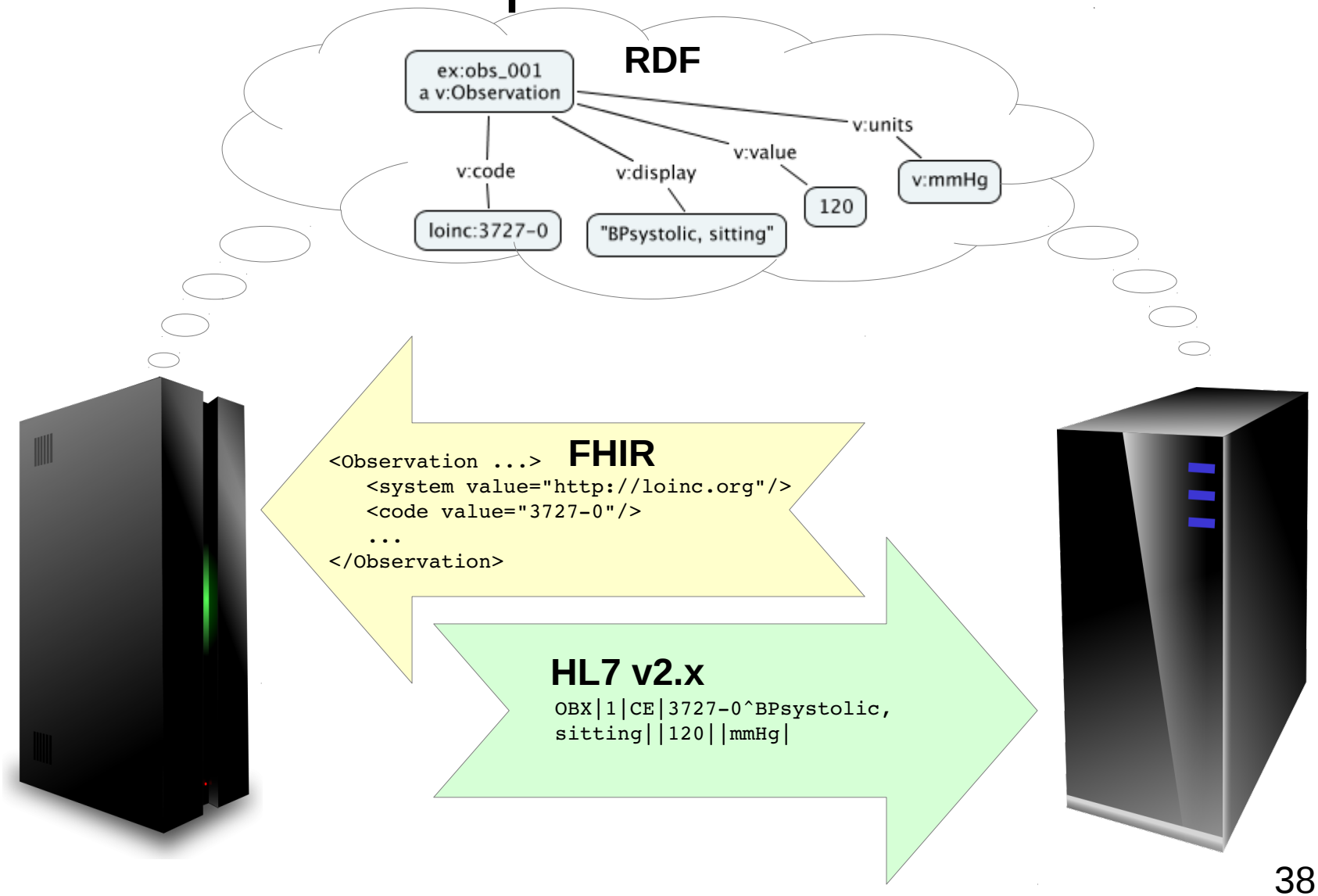
RDF graph



Why does this matter?

- Emphasis is on the meaning (where it should be)
- RDF acts as a universal information representation
  - Different formats can be exchanged with the same meaning

RDF as a universal information representation



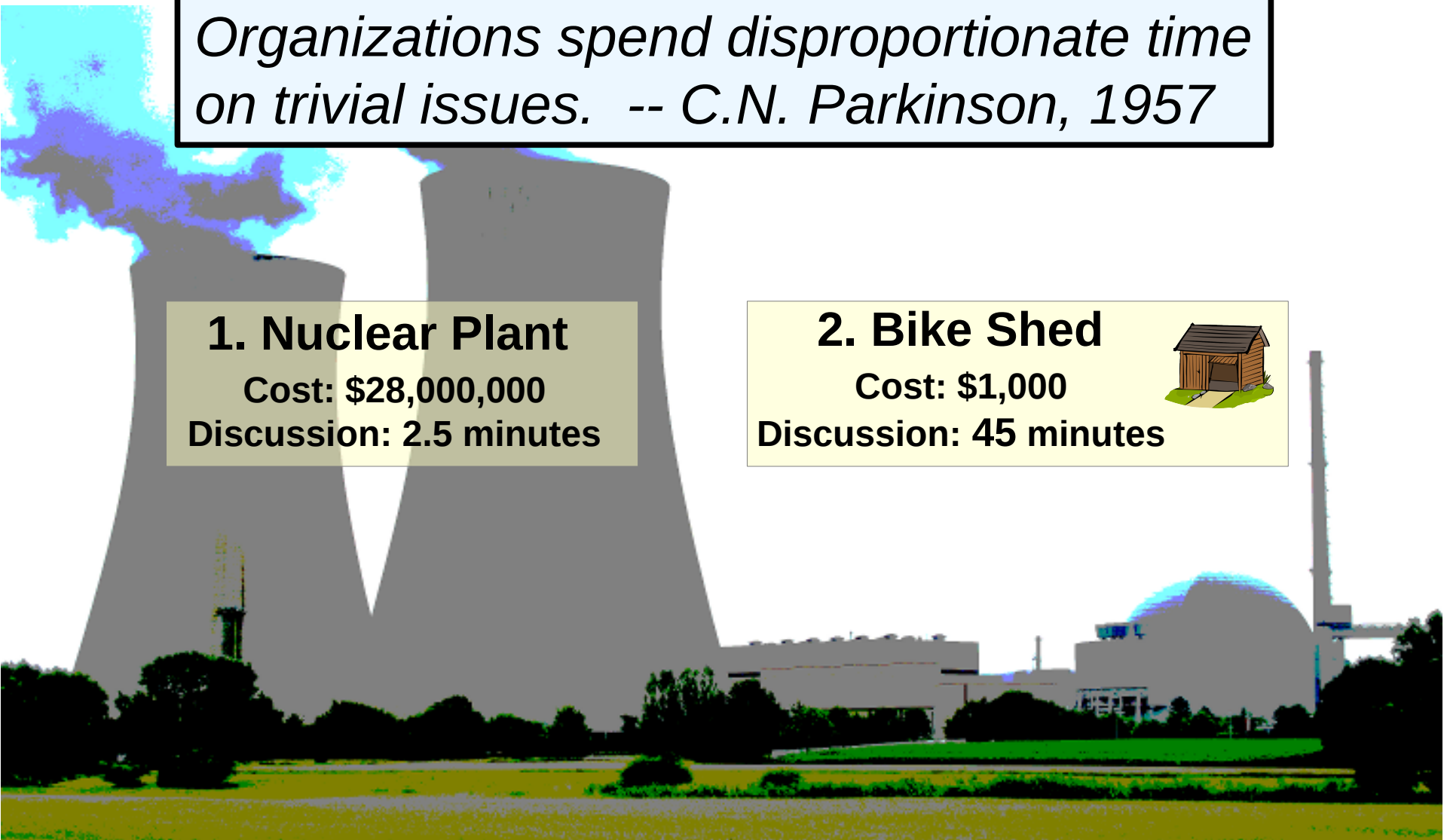
Why does this matter?

- Helps avoid the bike shed effect in standards, a/k/a Parkinson's Law of Triviality
  - Standards committees often spend hours arguing over syntax and naming -- irrelevant to computable information content

Bike shed effect

a/k/a Parkinson's Law of Triviality

Organizations spend disproportionate time on trivial issues. -- C.N. Parkinson, 1957



1. Nuclear Plant

Cost: $28,000,000

Discussion: 2.5 minutes

2. Bike Shed

Cost: $1,000

Discussion: 45 minutes

Captures meaning, not syntax?

- XML:
  - Syntax only

- JSON:
  - Syntax only

½

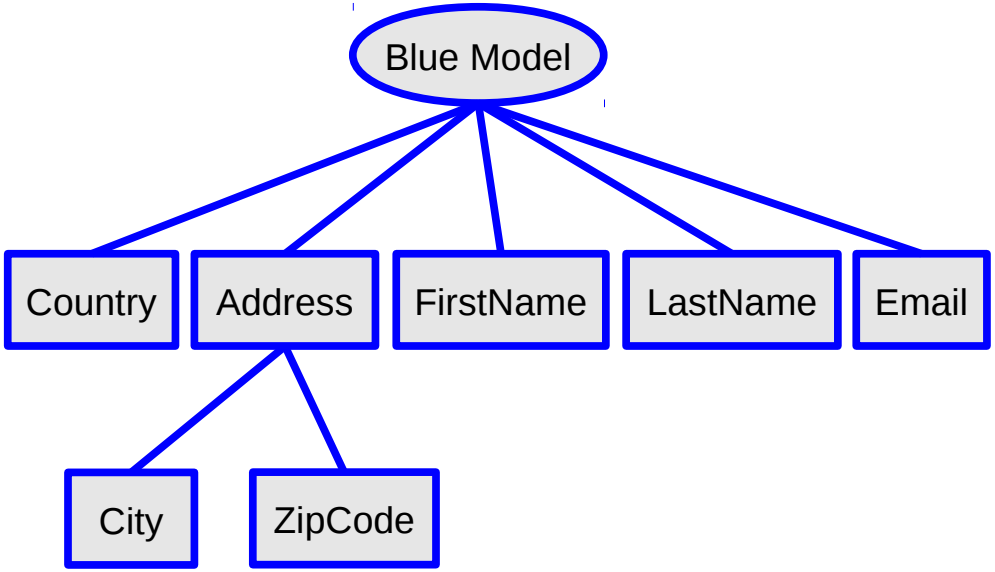
#2: Multiple data models and vocabularies can be easily combined and interrelated

- RDF is multi-schema friendly*
- Multiple data models/schemas and vocabularies can peacefully co-exist, semantically connected

*A/k/a schema-promiscuous, schema-flexible, schema-less, etc.

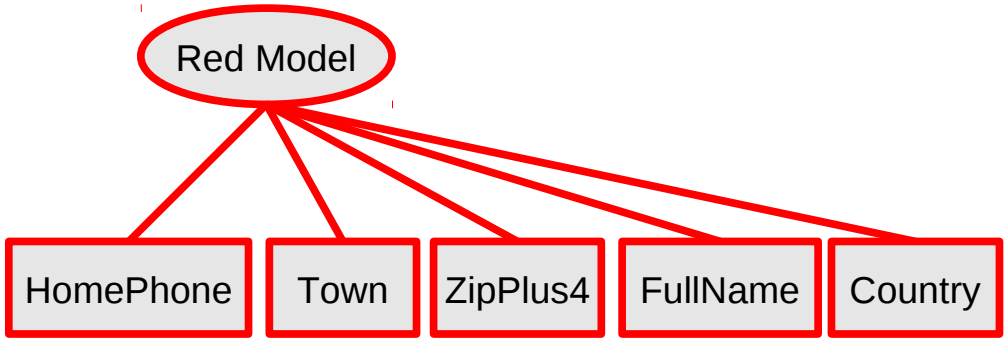
Multi-schema friendly

- Blue App has model



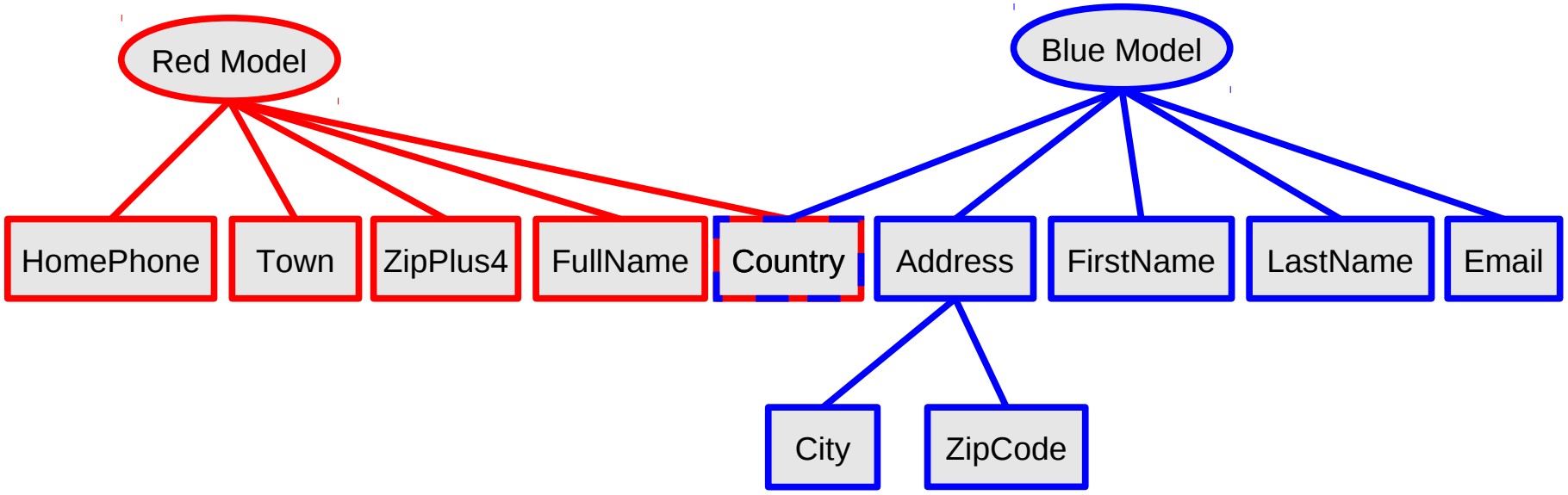
Multi-schema friendly

- Red App has model



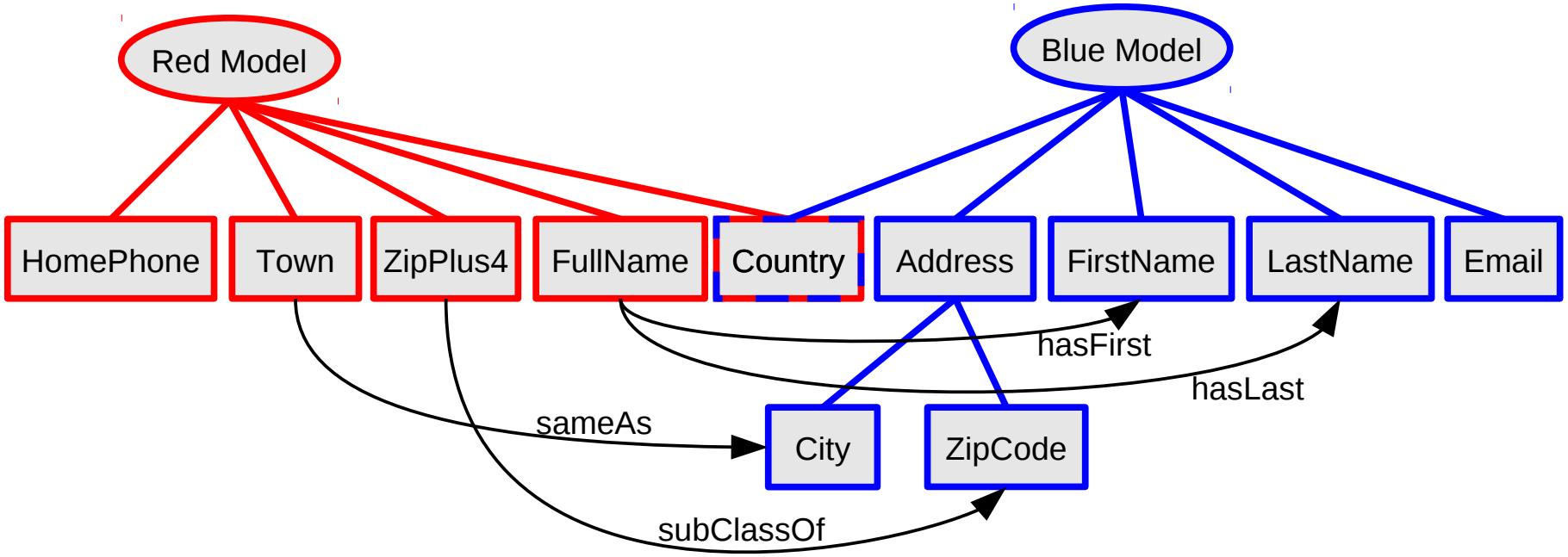
Multi-schema friendly

- Merge RDF data
- Same nodes (URIs) join automatically



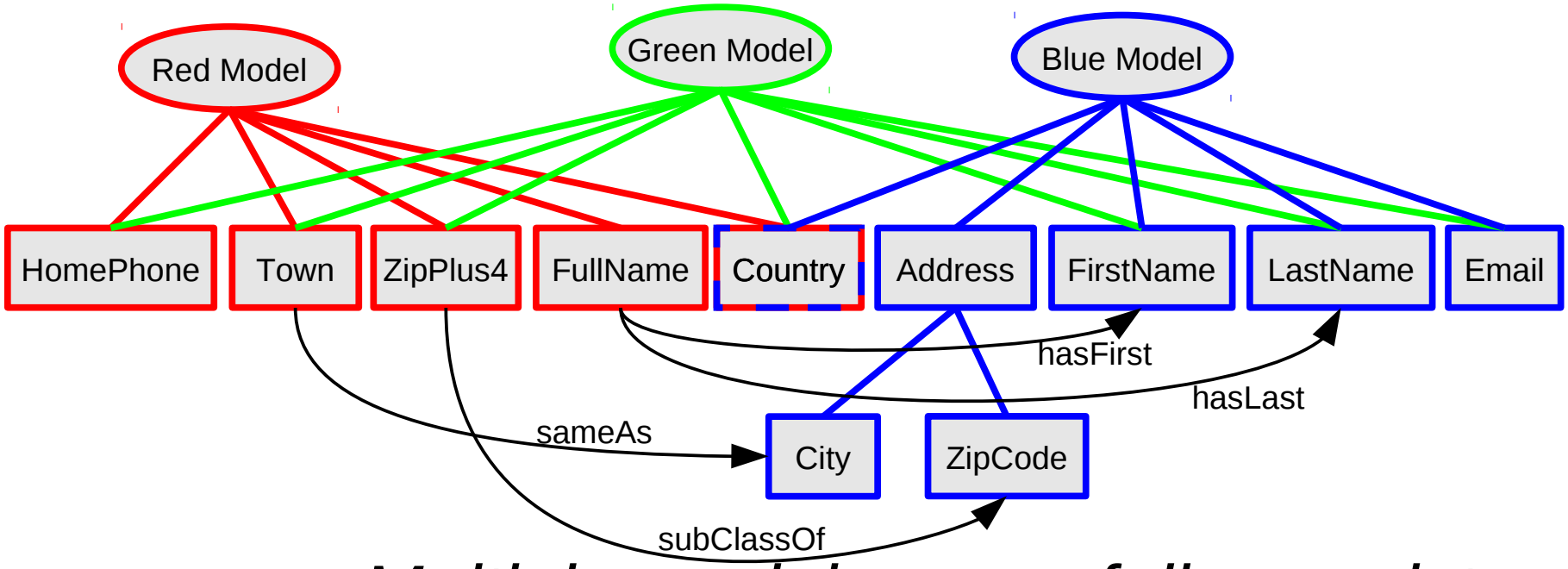
Multi-schema friendly

- Add relationships and rules
- (Relationships are also RDF)



Multi-schema friendly

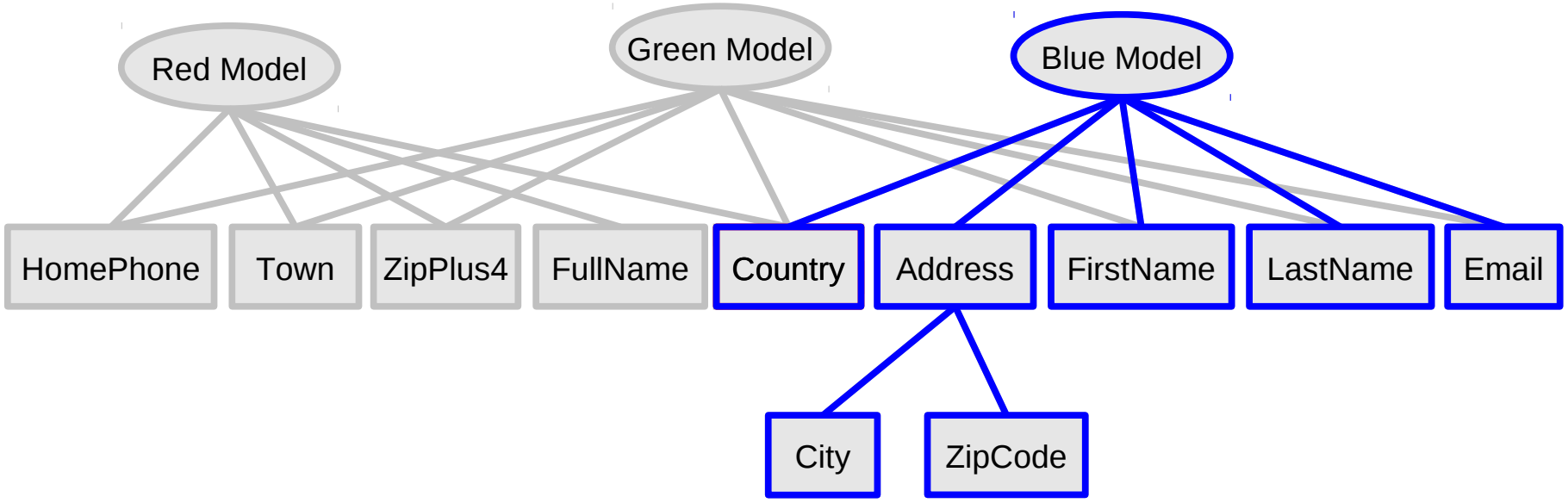
- Later add Green model
  (Using Red & Blue models)



Multiple models peacefully coexist

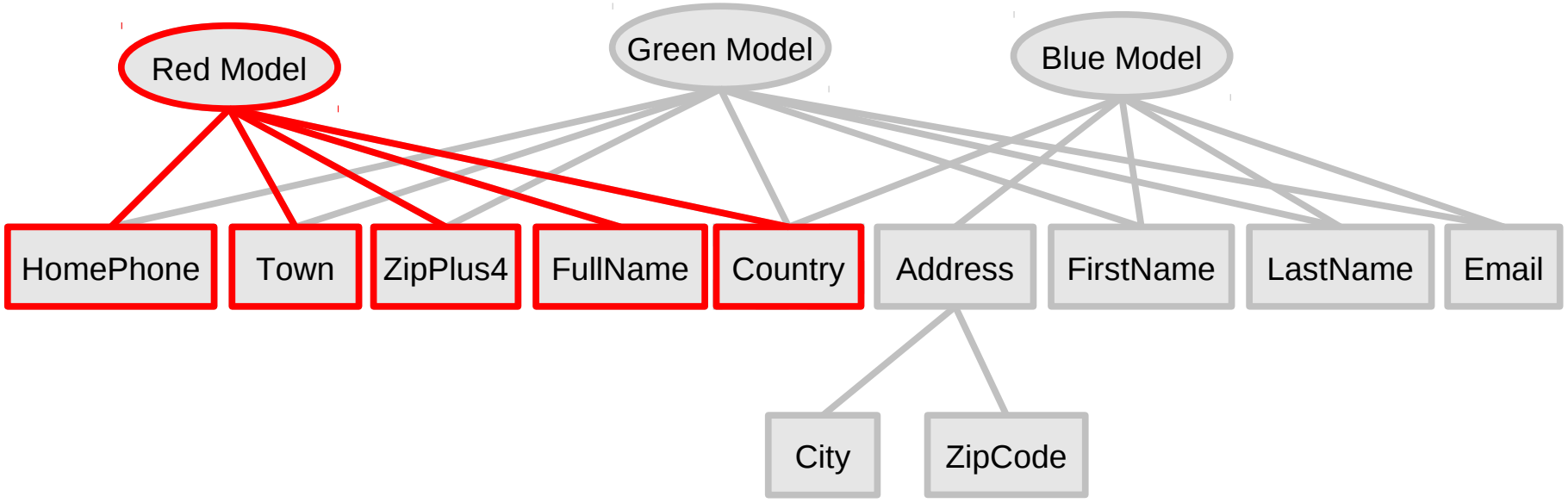
Multi-schema friendly

- Blue app sees Blue model



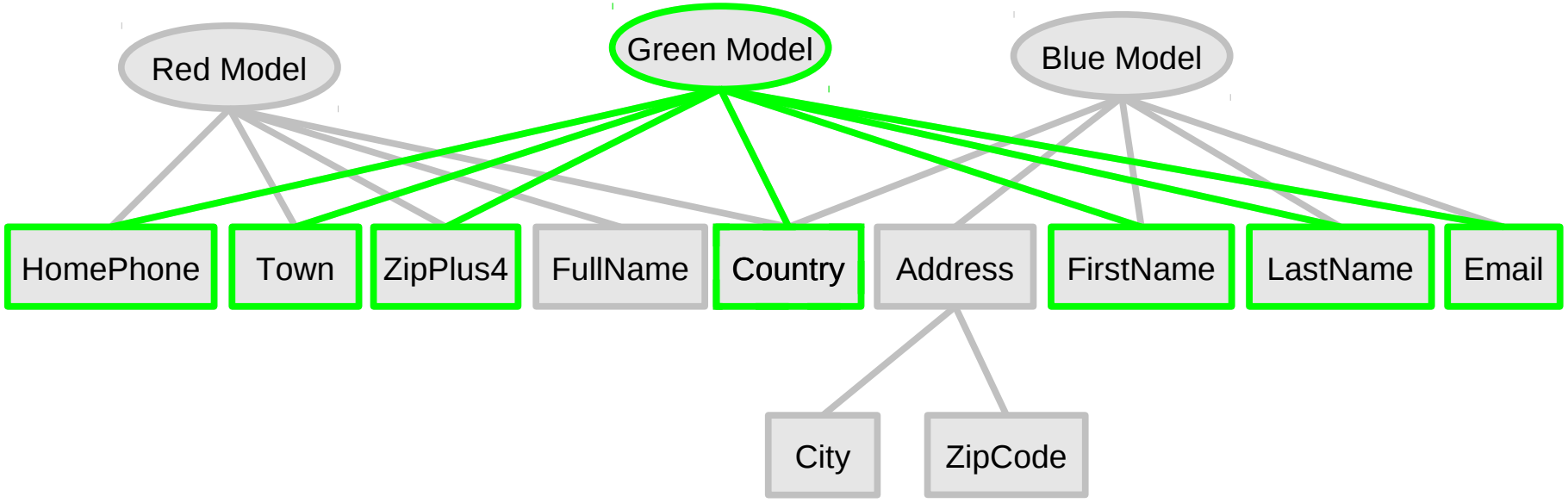
Multi-schema friendly

- Red app sees Red model

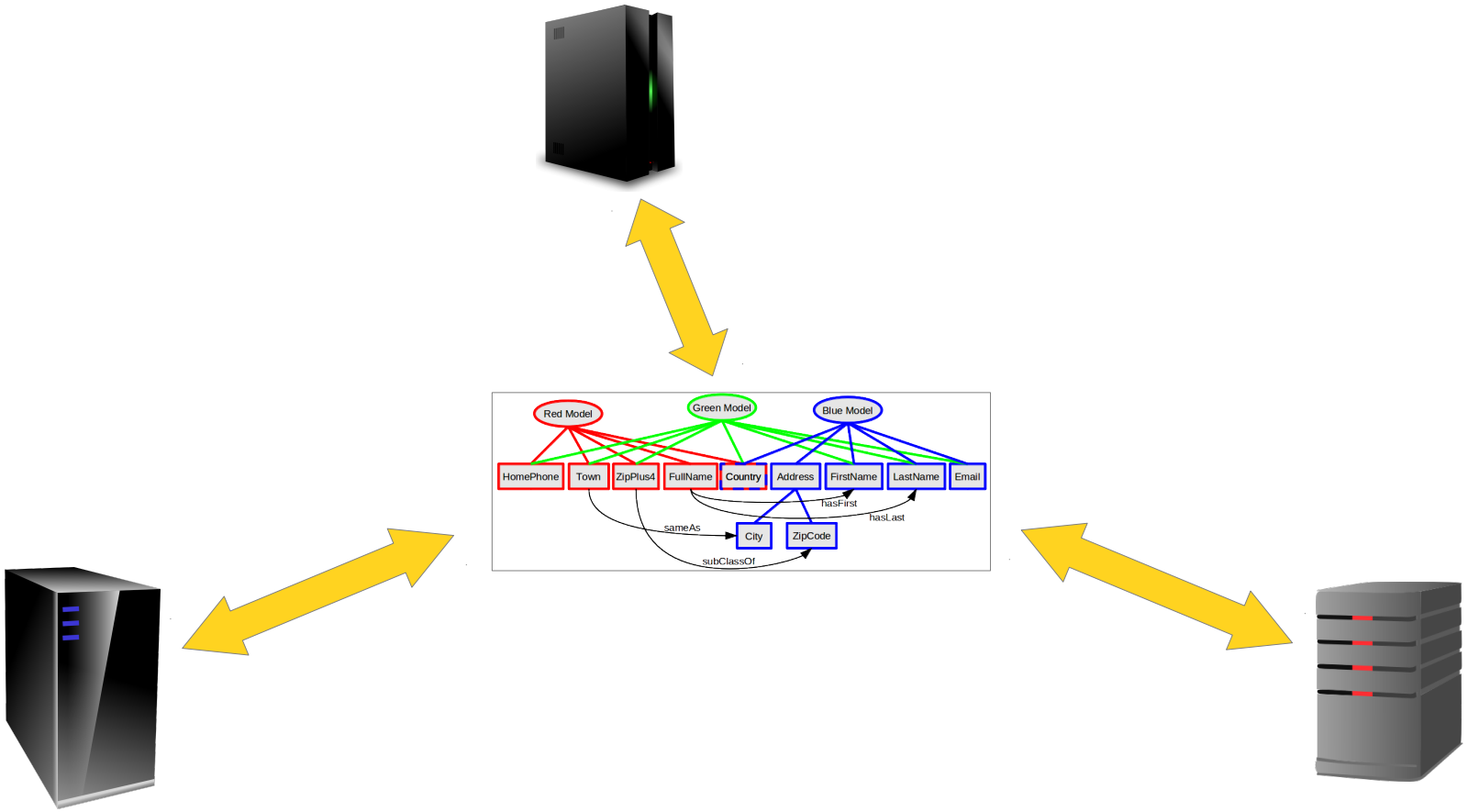


Multi-schema friendly

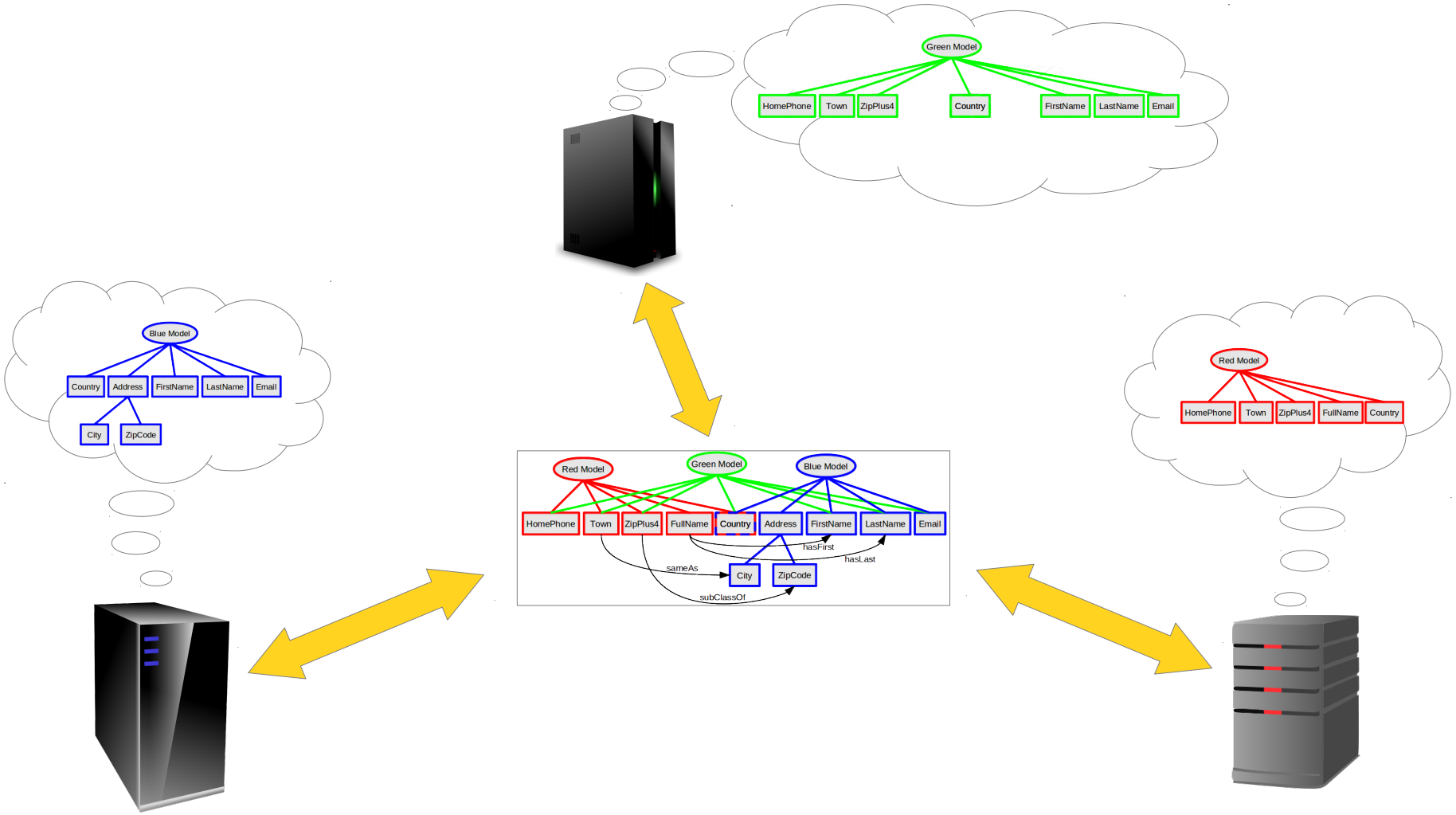
- Green app sees Green model



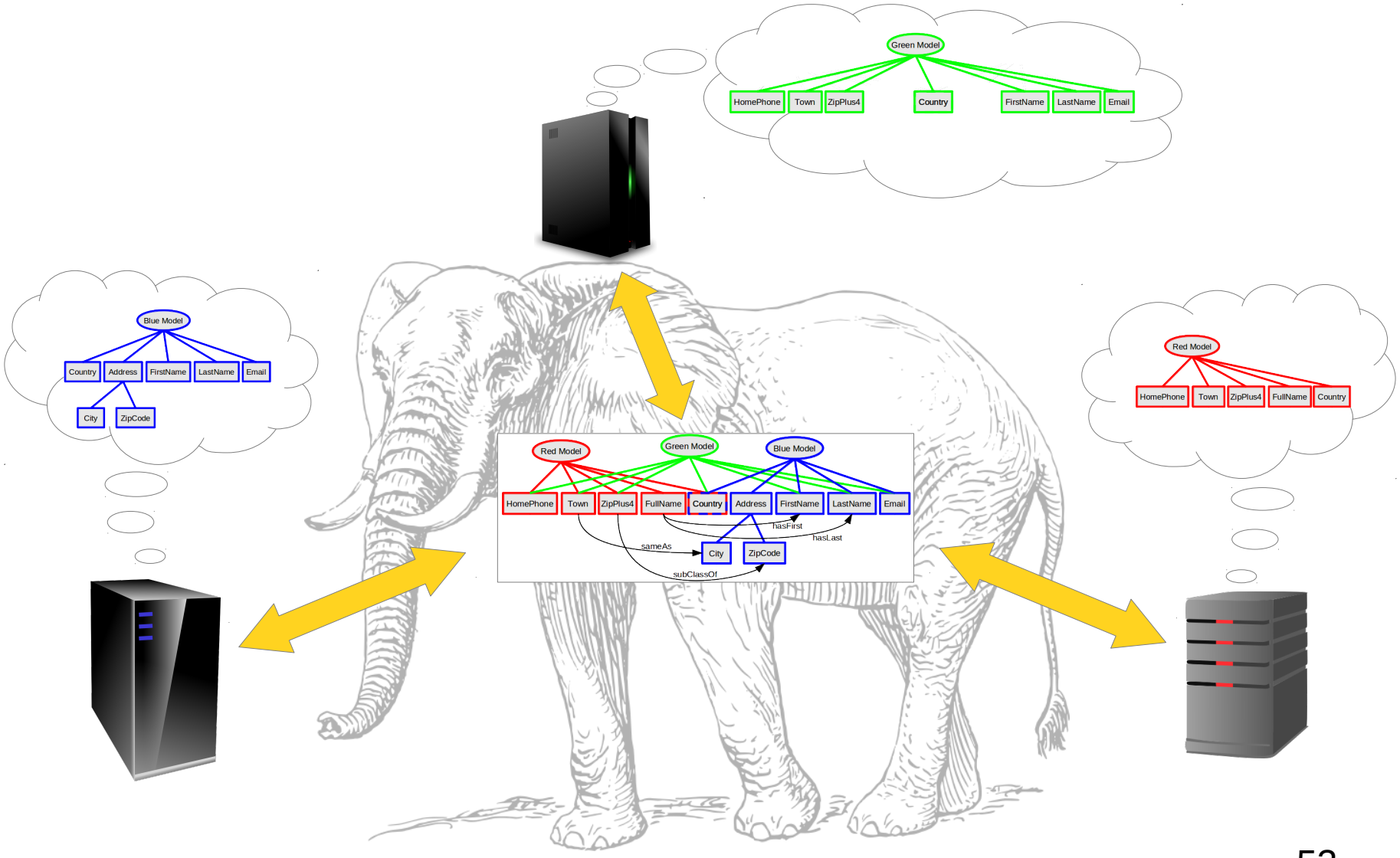
Different views for different systems



Different views for different systems



Different views for different systems



Why is this important?

- Multiple data models and vocabularies can be:
  - added dynamically
  - used together harmoniously
- This is critical in domains that involve many or changing data models/vocabularies
- Even standards change!
  - Standards are revised or they become obsolete

Easy to **combine and relate** data?

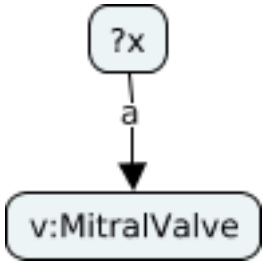
- XML:
  - Schemas compete to be "on top"
  - Meaningful merge requires new schema and manual mapping

- JSON:
  - A little easier than with XML
  - But meaningful merge still requires new model and manual mapping

#1: RDF enables smarter data use and automated data translation

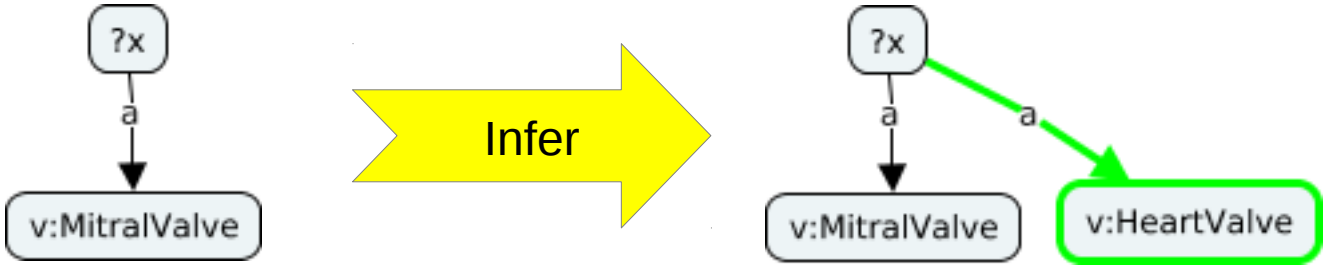
- RDF enables inference
- Inference derives new assertions from old
  - "Entailments"
- Query for v:HeartValve surgeries can find v:MitralValve surgeries

Inference example



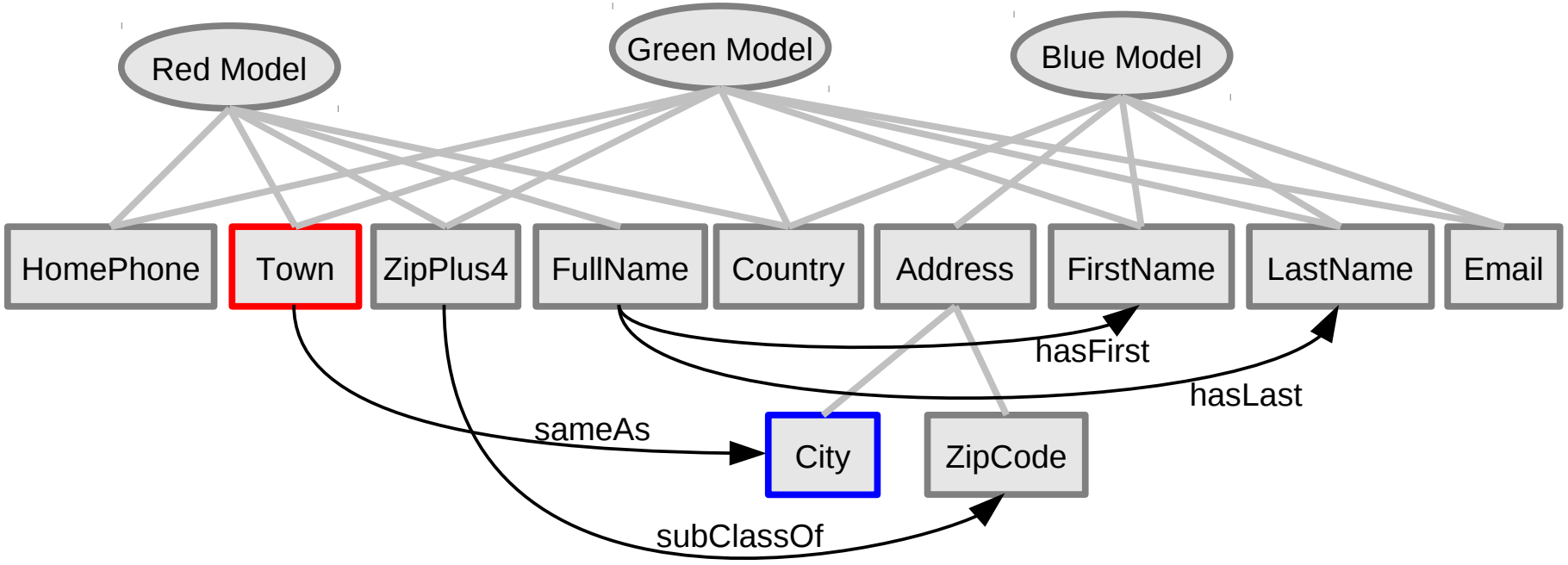
- **If you know:**
  ?x a v:MitralValve .
  v:MitralValve rdfs:subClassOf v:HeartValve .

Inference example



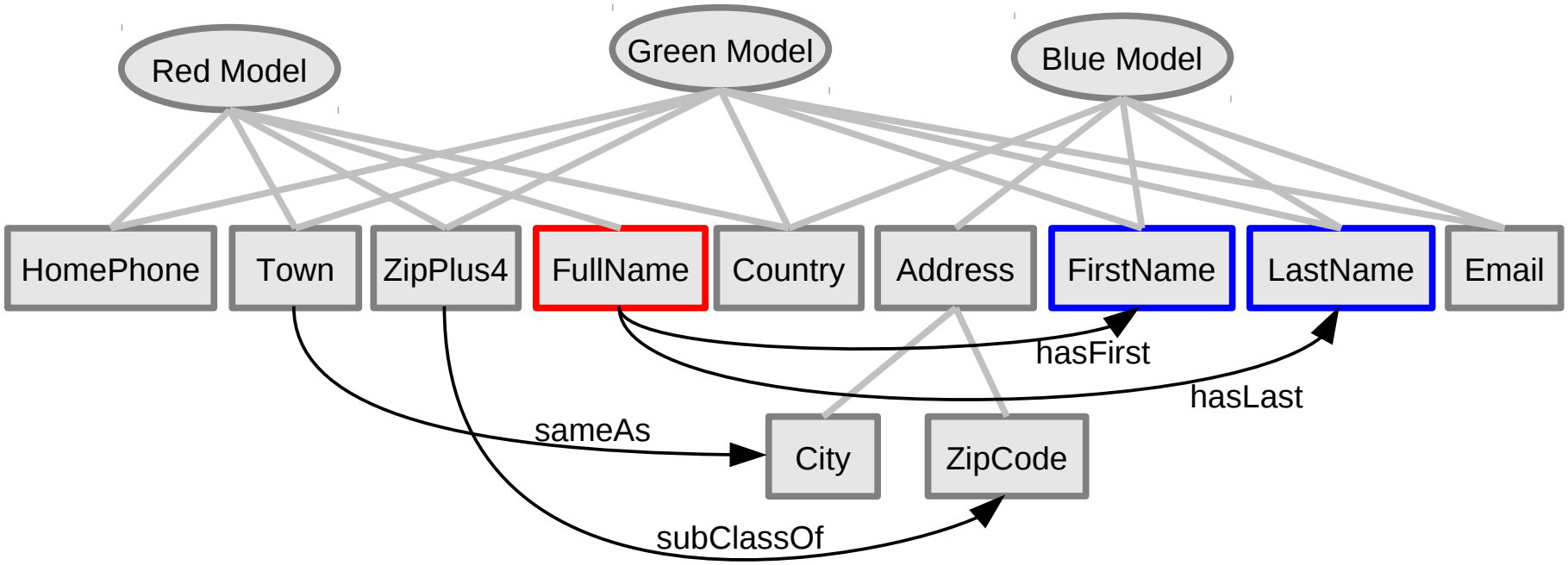
- **If you know:**
  ?x a v:MitralValve .
  v:MitralValve rdfs:subClassOf v:HeartValve .
- **You can infer:**
  ?x a v:HeartValve .

Inference example: sameAs



- **If you know:** Town
- **You can infer:** City (or vice versa)

Inference example: composition



- **If you know:** FirstName + LastName
- **You can infer:** FullName
  - But not necessarily vice versa

Why is this important?

- Smarter data use
  - Query for v:HeartValve surgeries can find v:MitralValve surgeries

Facilitates smarter queries?

- XML:
  - No inference

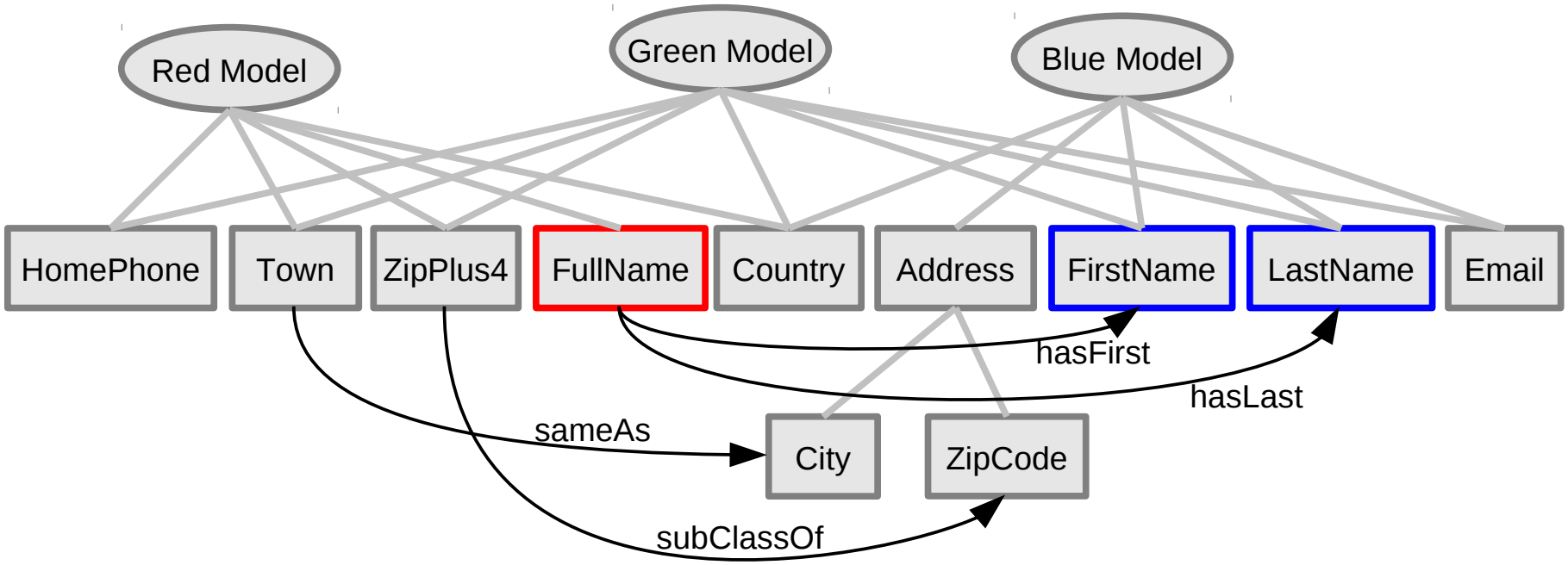
- JSON:
  - No inference

Why is this important?

- Data can be automatically **translated** between different data models and vocabularies
  - E.g., db:DB00945 => v:aspirin
  - Red Model data + Blue Model data => Green Model data

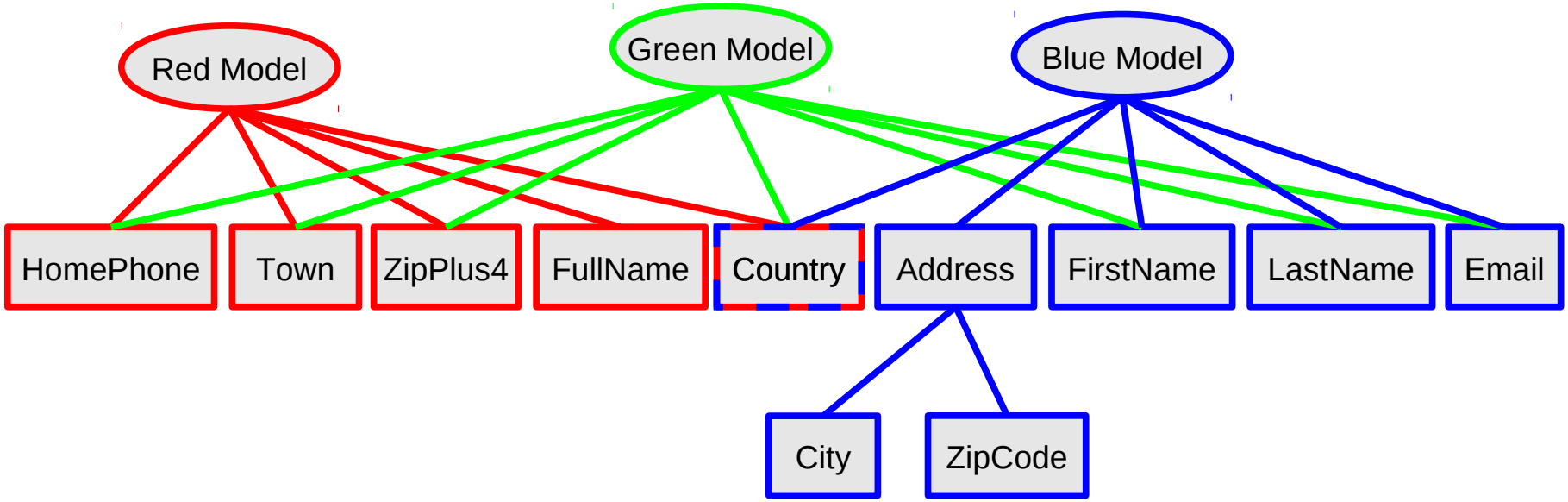
Very helpful for data integration!

Inference example: composition



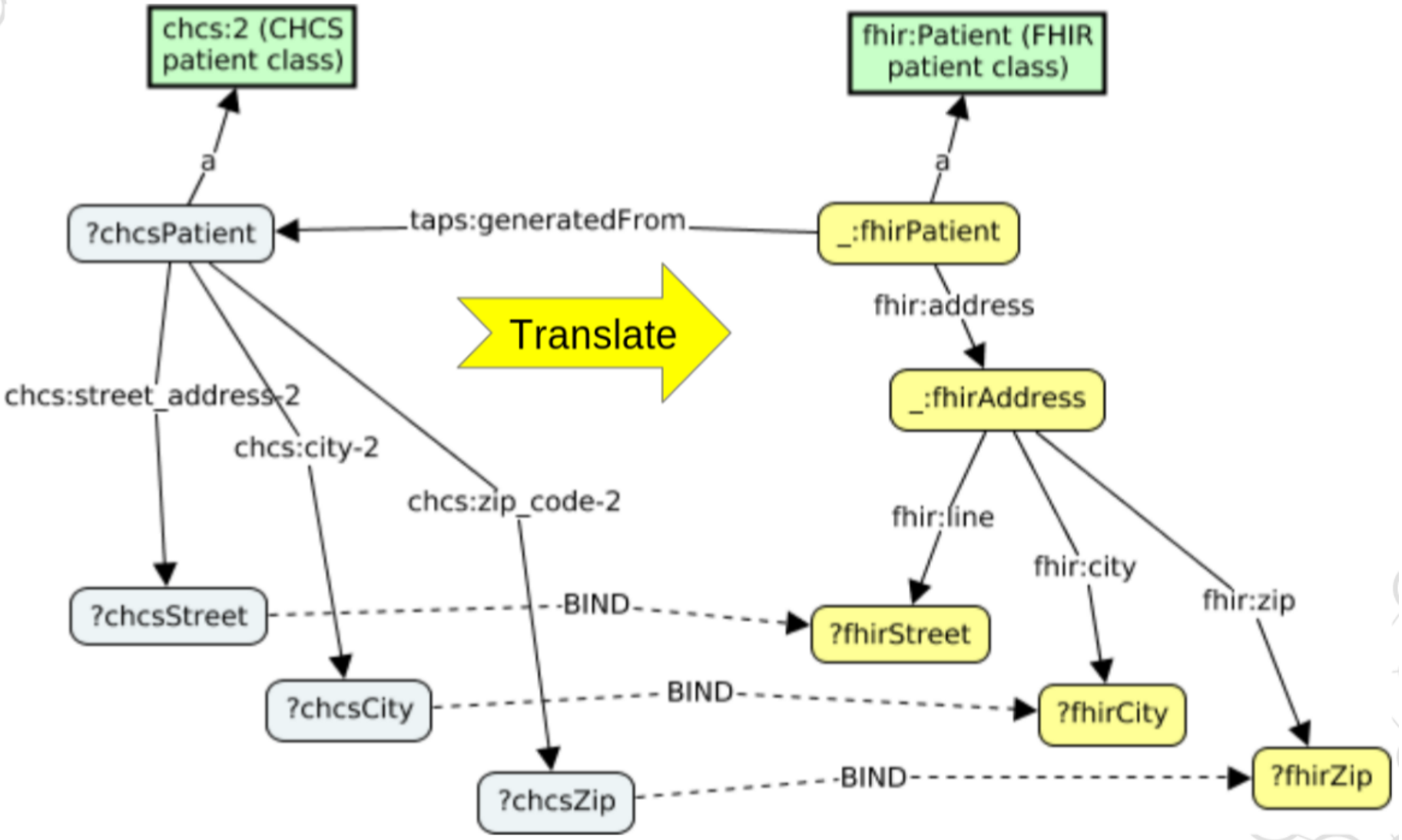
- **If you know:** FirstName + LastName
- **You can infer:** FullName
  - But not necessarily vice versa

Inference example: data translation

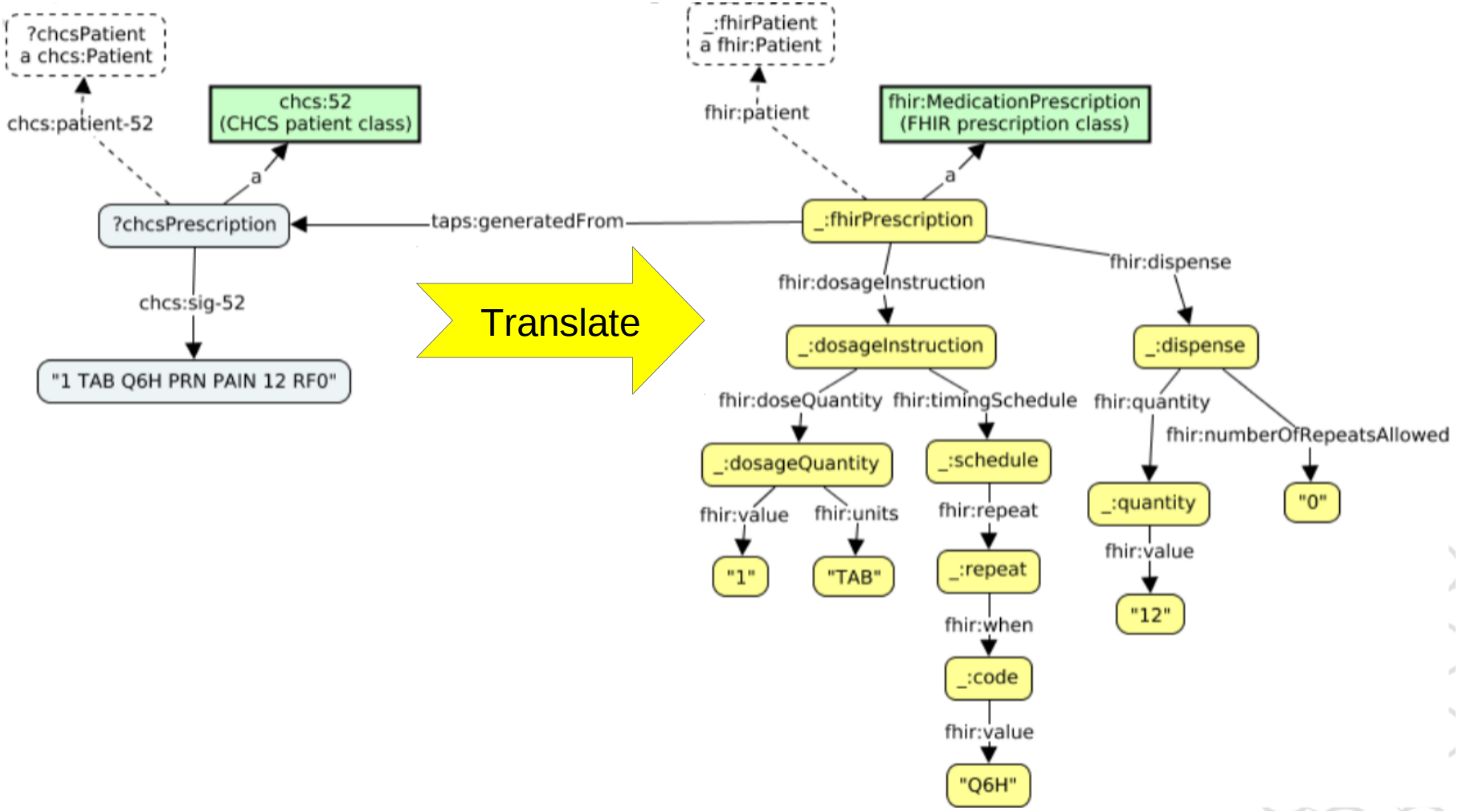


- If you know: Red Model data + Blue Model data
- You can infer: Green Model data

Translation as inference



Translation as inference



Facilitates data translations?

- XML:
  - Not by inference, but tools are available

½

- JSON:
  - Not by inference, but tools are available

½

Key things you need to know about RDF

#5: RDF is self describing

- RDF uses URIs as identifiers

#4: RDF is easy to map from other data representations

- RDF data is made of assertions

#3: RDF captures information – not syntax

- RDF is format independent

#2: Multiple data models and vocabularies can be easily combined and interrelated

- RDF is multi-schema friendly

#1: RDF enables smarter queries and automated data translation

- RDF enables inference

Weaknesses of RDF

- RDF tools are less mature; expertise is less widespread
- RDF has some annoyances:
  - "Blank nodes" have subtleties that add complication (Best to limit their use)
  - URI allocation – can be a hassle
- Weaknesses should be understood, but are not show stoppers

Conclusions

- RDF provides key benefits that distinguish it from other frequently used information representations
- RDF is best for problems that involve:
  - Large-scale information integration
  - Semantically connecting diverse vocabularies and data models
  - Changing vocabularies and data models
  - Inference and data translation

Questions?

BACKUP SLIDES

Key things you need to know about RDF

#5: RDF is self describing

- RDF uses URIs as identifiers

#4: RDF is easy to map from other data representations

- RDF data is made of assertions

#3: RDF captures information – not syntax

- RDF is format independent

#2: Multiple data models and vocabularies can be easily combined and interrelated

- RDF is multi-schema friendly

#1: RDF enables smarter queries and automated data translation

- RDF enables inference

If time permits . . .

- Ivan Herman's Semantic Web tutorial:
  http://www.w3.org/People/Ivan/CorePresentations/SWTutorial/Slides.pdf