Lecture 8

Artificial Intelligence: Knowledge Graphs

Part I: Intelligent Agents, Resource Description Framework (RDF)

COMP 6721, Fall 2023

Outline

1. Introduction
2. The Resource Description Framework (RDF)
3. Example: Data Integration with Knowledge Graphs
4. Notes and Further Reading

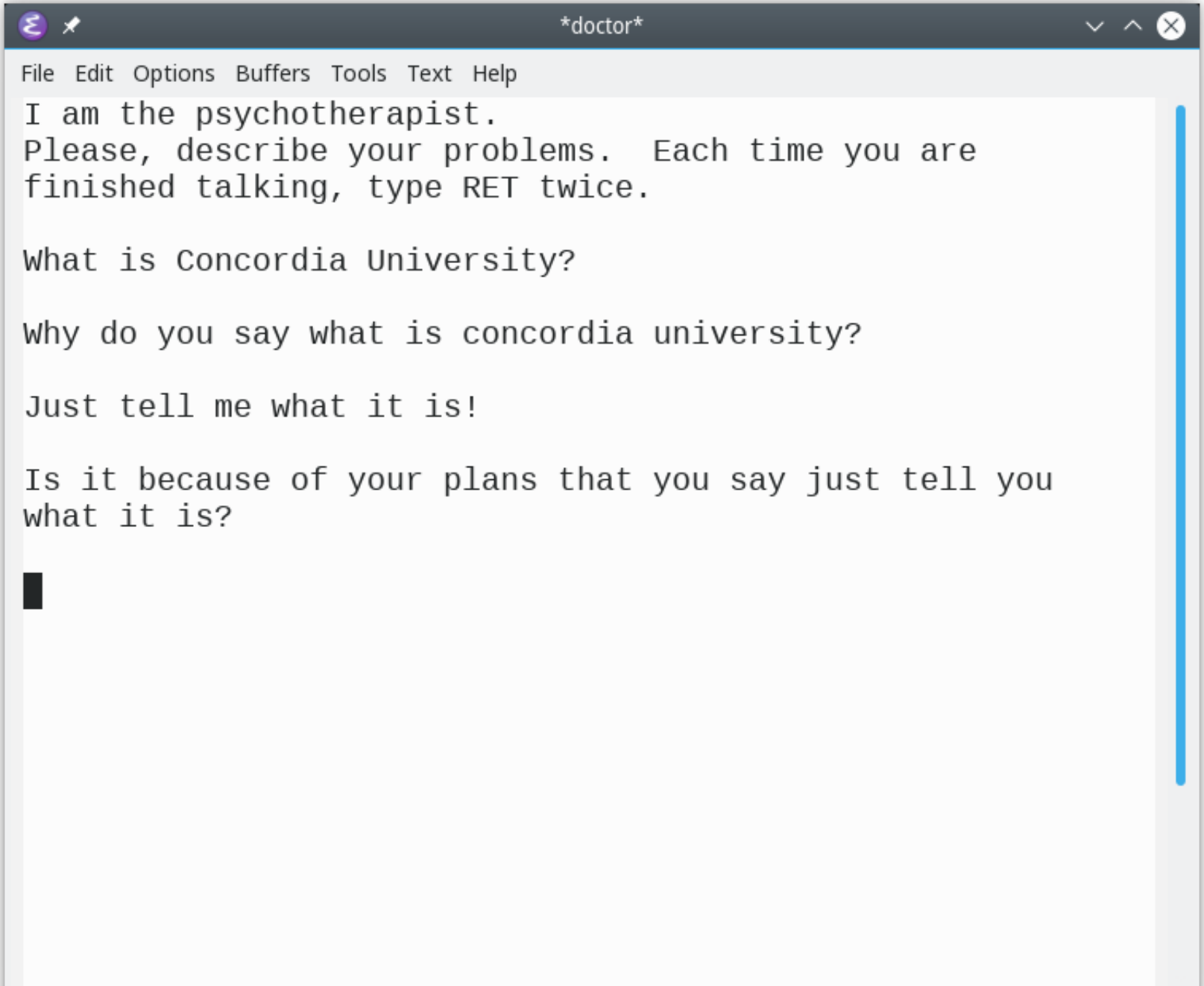
Slides Credit

- Includes slides from Jay Pujara & Sameer Singh, *Mining Knowledge Graphs from Text*, https://kgtutorial.github.io/
- Includes slides by Ivan Herman, W3C [Her]
- Includes slides from Hoifung Poon, Chris Quirk & Scott Wen-Tau Yih, *Machine Reading for Precision Medicine*, https://www.microsoft.com/en-us/research/uploads/prod/2018/01/1802_aaai-tutorial_precision-med.pdf

Outline

1. Introduction
   - Motivation
   - Why Knowledge Graphs?
   - Example: AI for Precision Medicine
   - History
2. The Resource Description Framework (RDF)
3. Example: Data Integration with Knowledge Graphs
4. Notes and Further Reading

Eliza


doctor
File Edit Options Buffers Tools Text Help
I am the psychotherapist.
Please, describe your problems. Each time you are
finished talking, type RET twice.
What is Concordia University?
Why do you say what is concordia university?
Just tell me what it is!
Is it because of your plans that you say just tell you
what it is?

Google Assistant

What is Concordia University?

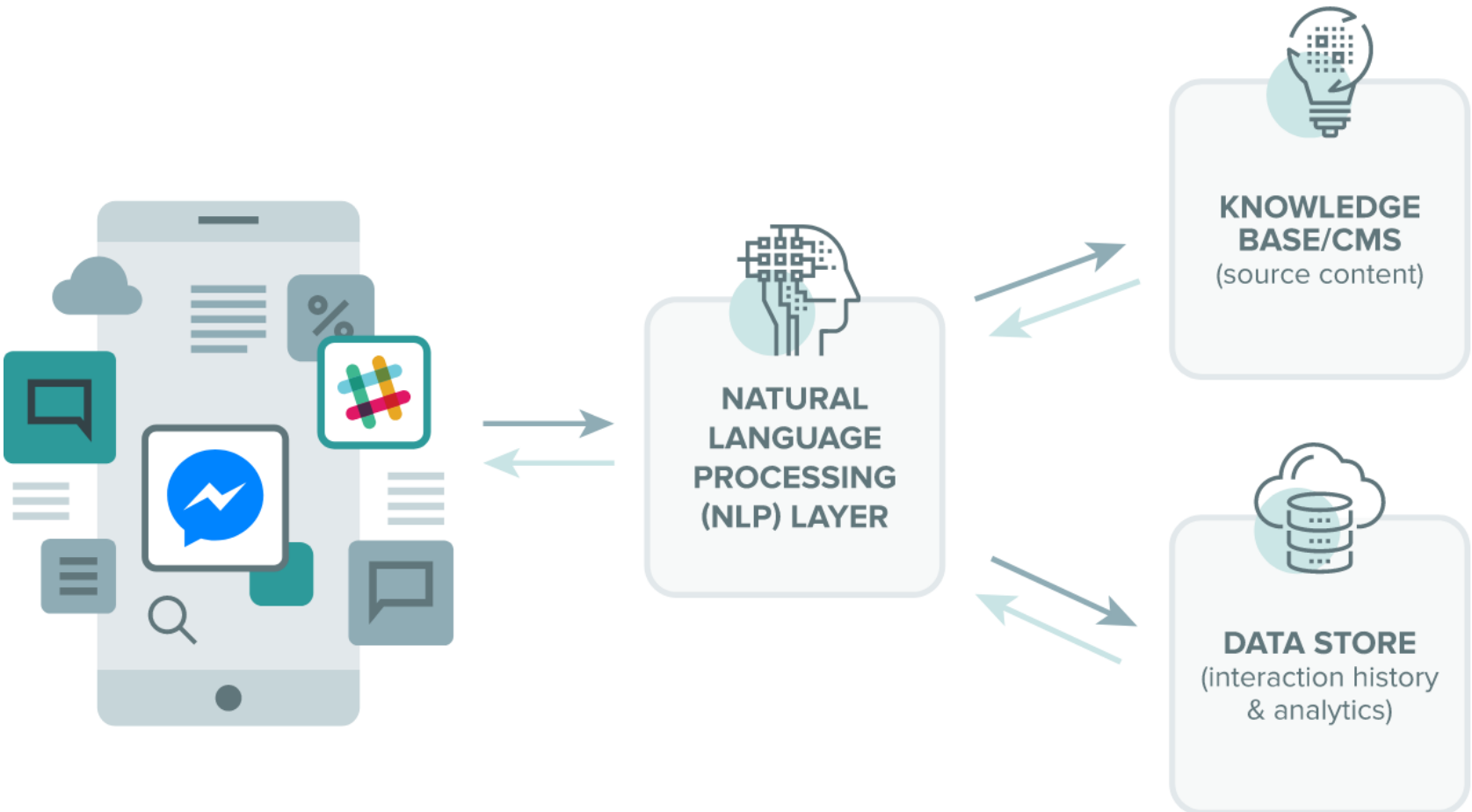


Concordia University (French: Université **Concordia**; commonly referred to as **Concordia**) is a public comprehensive research **university** located in Montreal, Quebec, Canada. ... **Concordia** is a non-sectarian and coeducational institution, with more than 215,000 alumni worldwide.

Athletics brand: Concordia Stingers
Subsidiary or constituent schools: Loyola Campus, Faculty of Arts
Date founded: August 24, 1974
Geographic scope: Canada

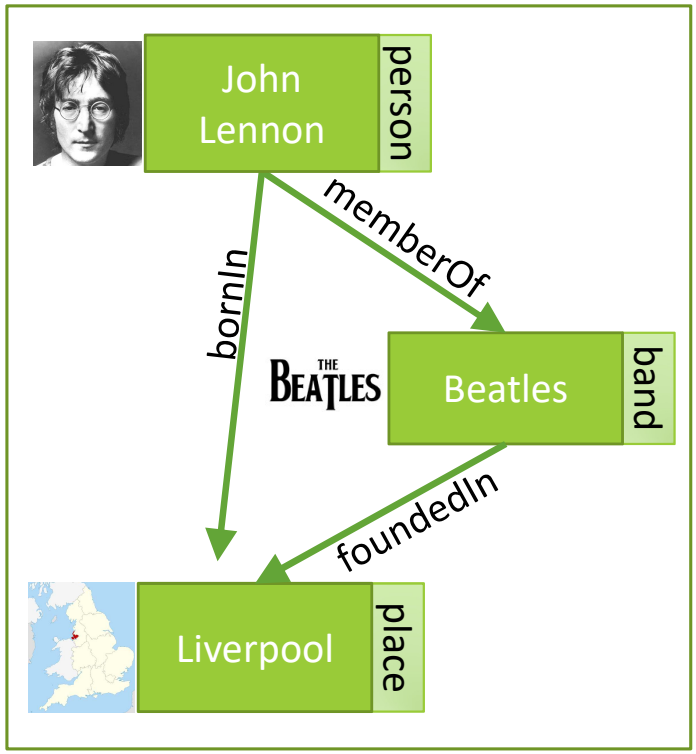
IBM Watson

Generic Intelligent Assistant Architecture



Example knowledge graph

- Knowledge in graph form!
- Captures entities, attributes, and relationships
- Nodes are entities
- Nodes are labeled with attributes (e.g., types)
- Typed edges between two nodes capture a relationship between entities

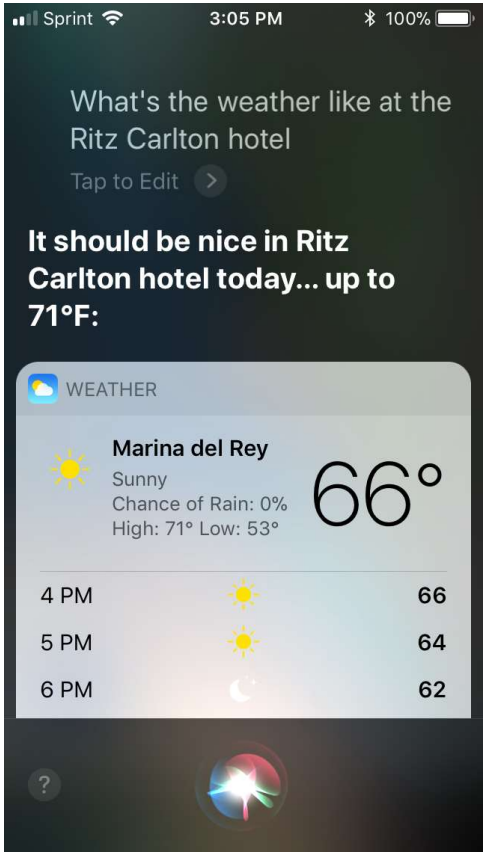


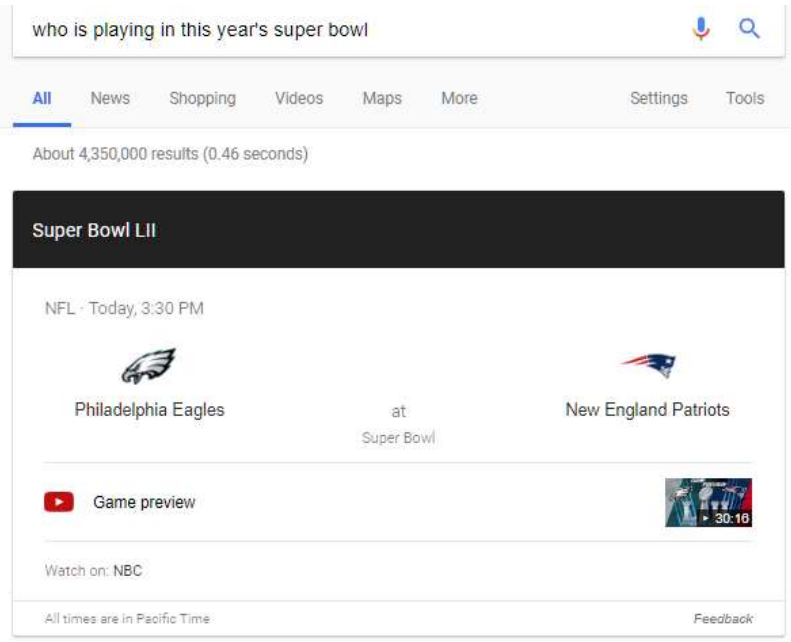
Why knowledge graphs?

- Humans:
  - Combat information overload
  - Explore via intuitive structure
  - Tool for supporting knowledge-driven tasks

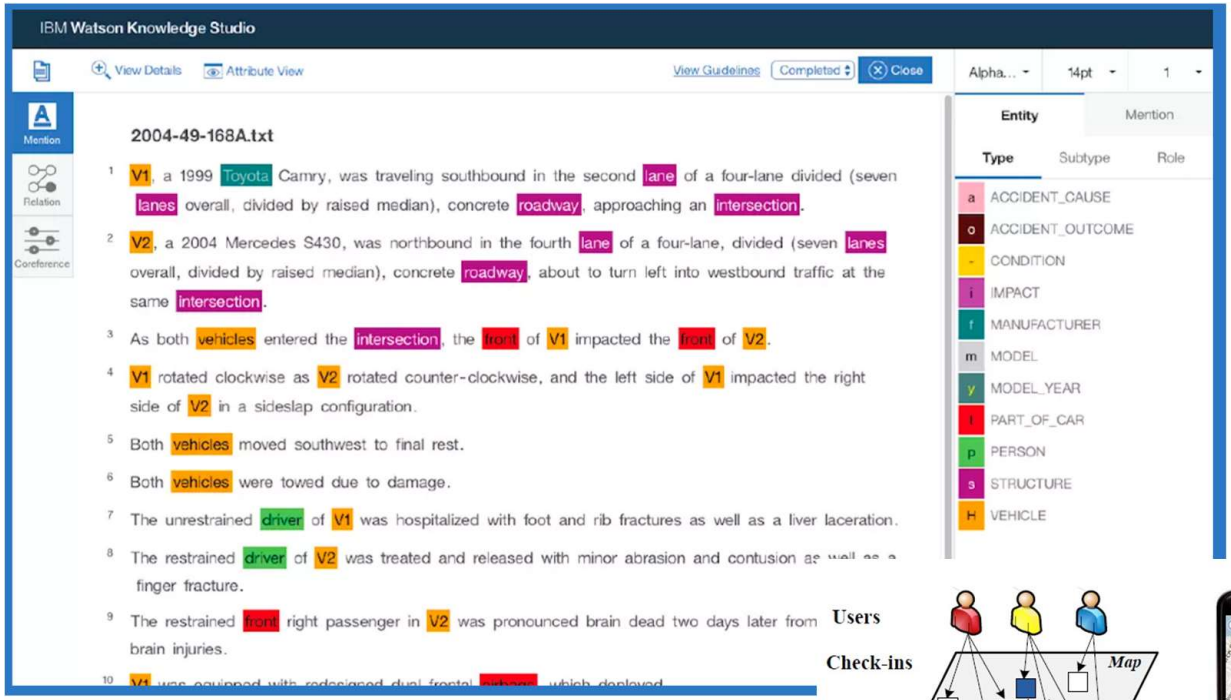
- AIs:
  - Key ingredient for many AI tasks
  - Bridge from data to human semantics
  - Use decades of work on graph analysis

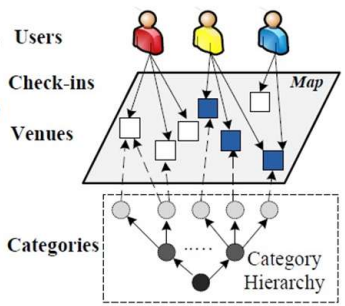
Applications 1: QA/Agents





Applications 2: Decision Support





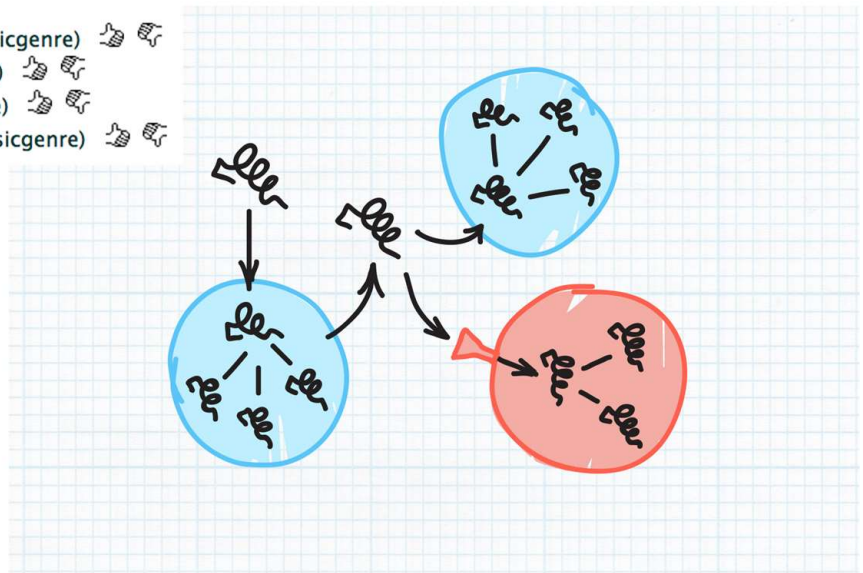
Applications 3: Fueling Discovery

beatles (musicartist)
literal strings: BEATLES, Beatles, beatles

Help NELL Learn!

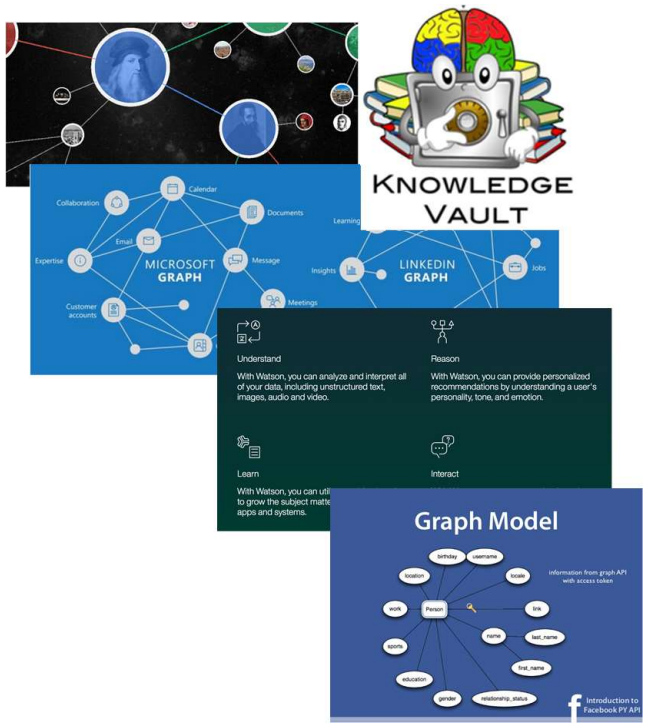
NELL wants to know if these be
If they are or ever were, click thumbs-up. O

- beatles is a musical artist
- beatles is a musician in the genre classic_pop (musicgenre)
- beatles is a musician in the genre pop (musicgenre)
- beatles is a musician in the genre rock (musicgenre)
- beatles is a musician in the genre classic_rock (musicgenre)

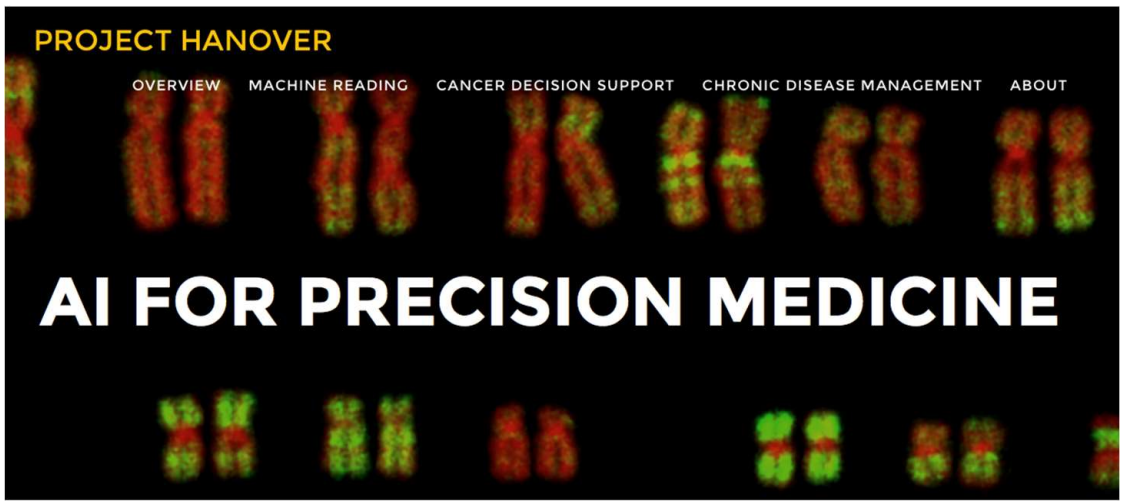


Knowledge Graphs & Industry

- Google Knowledge Graph
  - Google Knowledge Vault
- Amazon Product Graph
- Facebook Graph API
- IBM Watson
- Microsoft Satori
  - Project Hanover/Literome
- LinkedIn Knowledge Graph
- Yandex Object Answer
- Diffbot, GraphIQ, Maana, ParseHub, Reactor Labs, SpazioDati



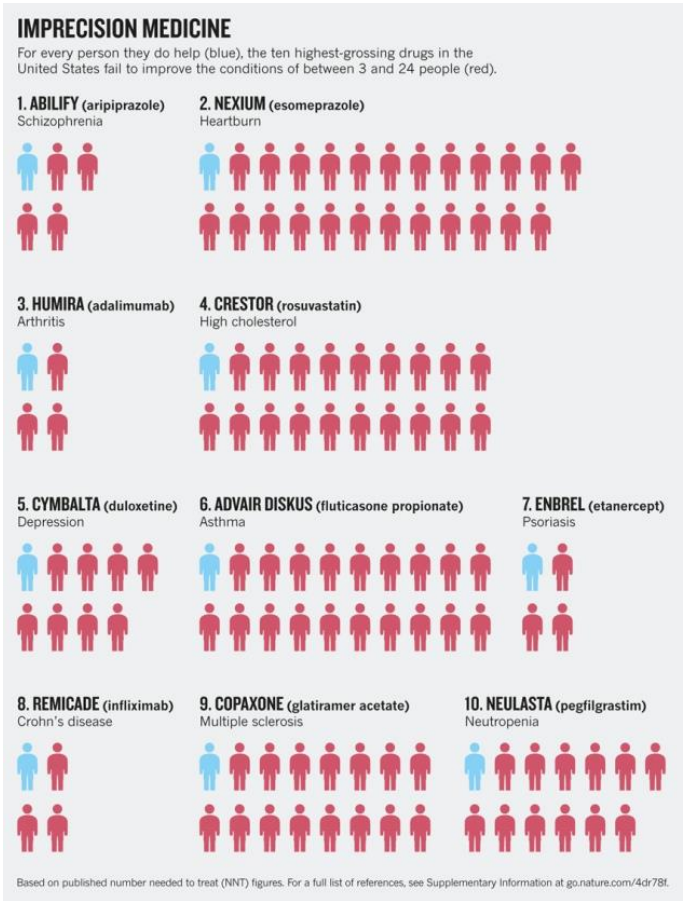
Interesting application of Knowledge Graphs



Microsoft Research

Chronic disease management:
develop AI technology for predictive and preventive personalized medicine to reduce the national healthcare expenditure on chronic diseases
(90% of total cost)

Medicine Today Is Imprecise



Top 20 drugs
80% non-responders

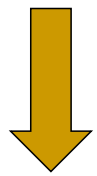
Wasted
1/3 health spending
$1 Trillion / year

5

Example: Tumor Board KB Curation

The deletion mutation on exon-19 of EGFR gene was present in 16 patients, while the L858E point mutation on exon-21 was noted in 10.

All patients were treated with gefitinib and showed a partial response.



Gefitinib can treat tumors w. EGFR-L858E mutation

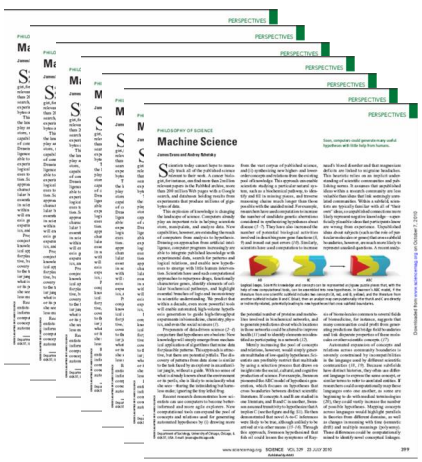
21

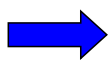
PubMed

27 million abstracts

Two new abstracts every minute

Adds over one million every year





Machine Reading

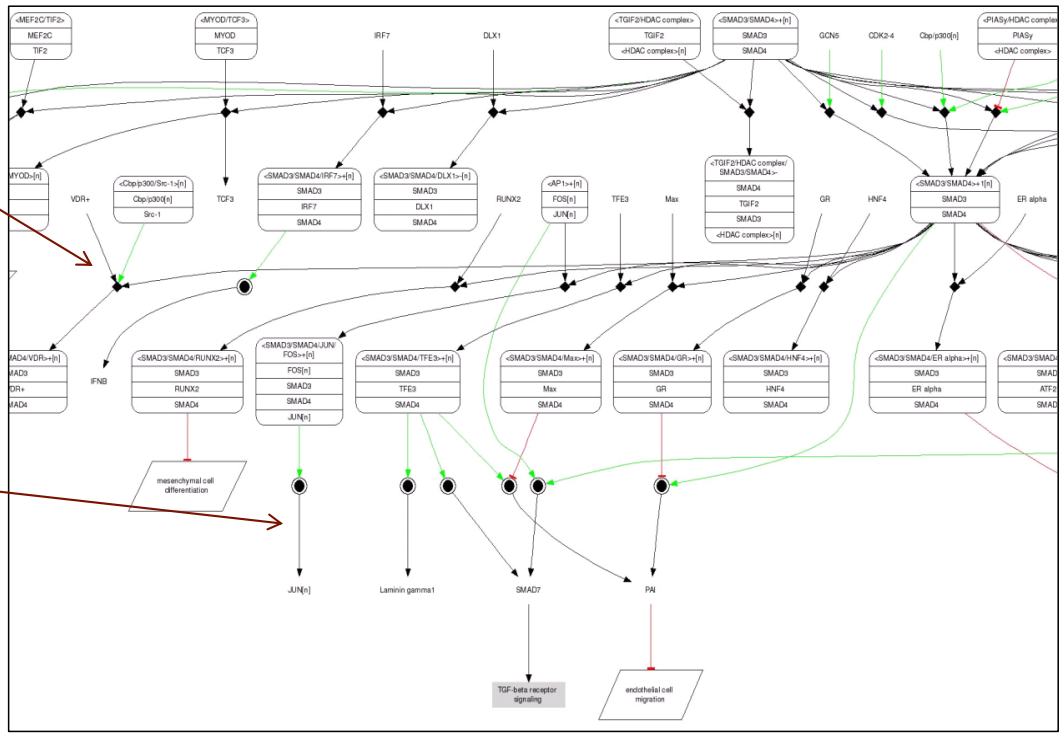
PMID: 123

…
VDR+ binds to SMAD3 to form
…

PMID: 456

…
JUN expression is induced by SMAD3/4
…

⋮



Knowledge Base

Complex Semantics

Involvement of p70(S6)-kinase activation in IL-10 up-regulation in human monocytes by gp41 envelope protein of human immunodeficiency virus type 1 ...

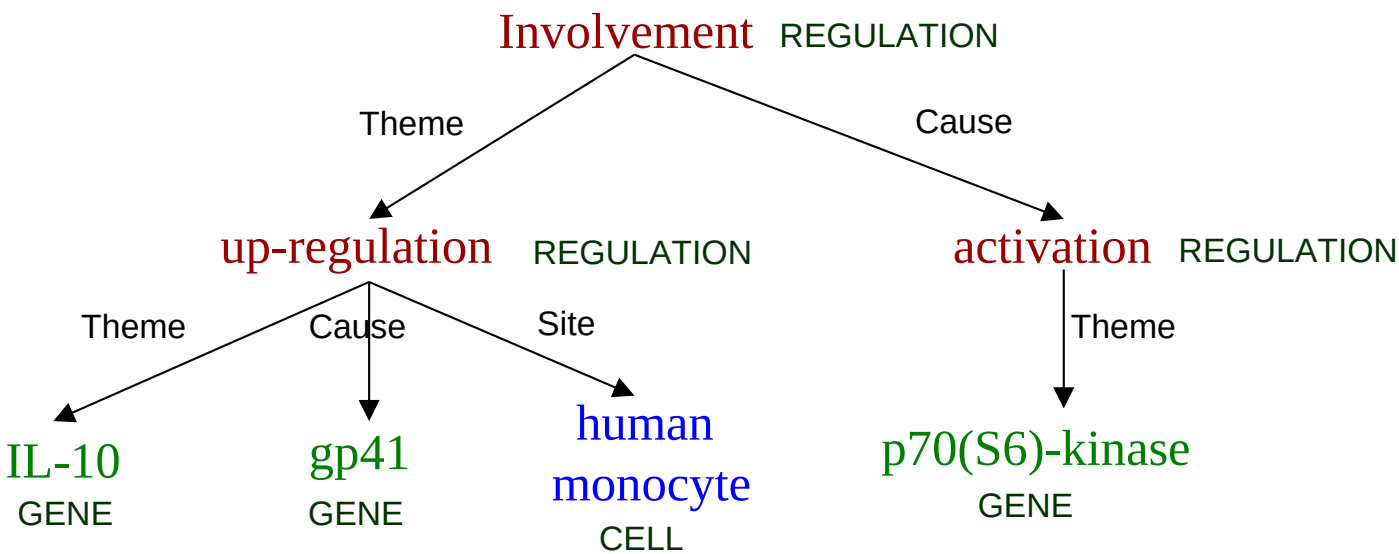
Complex Semantics

Involvement of p70(S6)-kinase activation in IL-10 up-regulation in human monocytes by gp41 envelope protein of human immunodeficiency virus type 1 ...

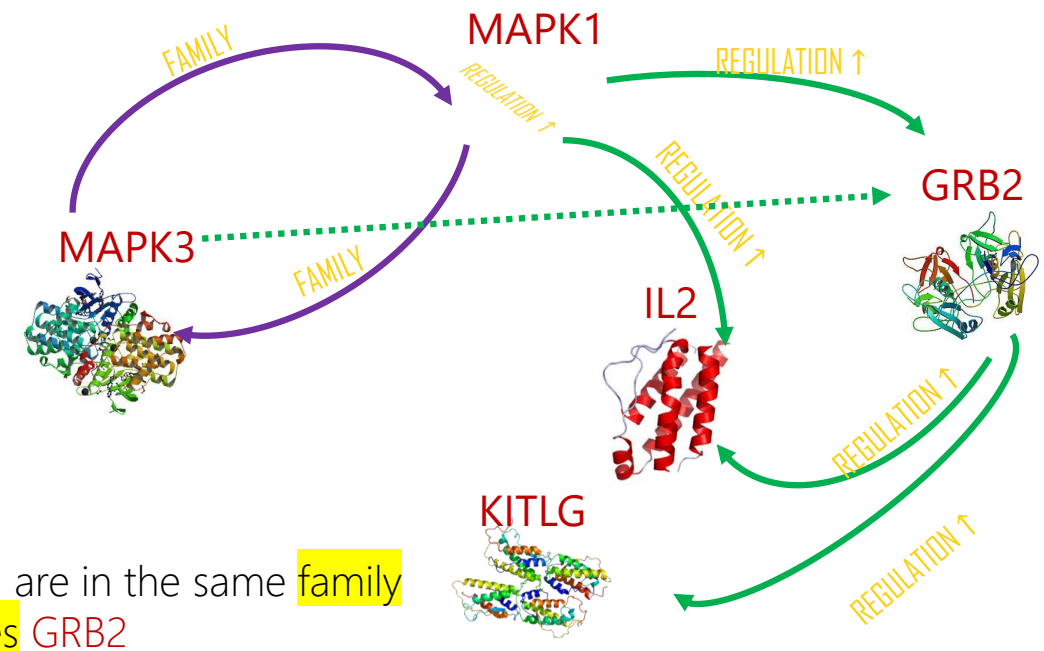
IL-10	gp41	human monocyte	p70(S6)-kinase
GENE	GENE	CELL	GENE

Complex Semantics

Involvement of p70(S6)-kinase activation in IL-10 up-regulation in human monocytes by gp41 envelope protein of human immunodeficiency virus type 1 ...



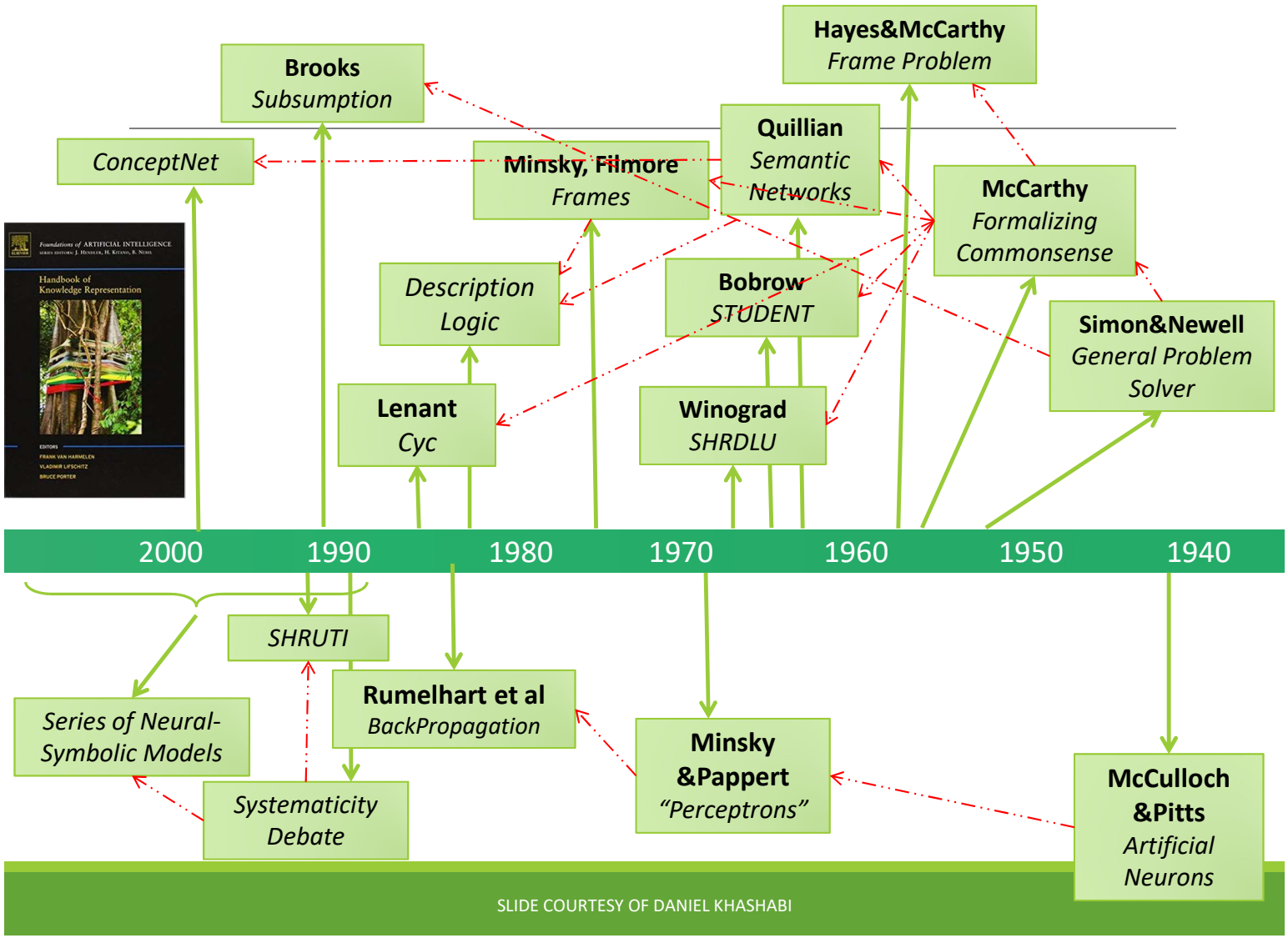
Genomics Knowledge Base (Network)



MAPK3 and MAPK1 are in the same family
MAPK1 up-regulates GRB2

Likely that MAPK3 up-regulates GRB2

History of Knowledge Representation (KR)



Today

From 1950–2020...

- Concepts have been around for a long time (Semantic Networks, Frames, Description Logic, ...)

Today

From 1950–2020...

- Concepts have been around for a long time (Semantic Networks, Frames, Description Logic, ...)

1980s/90s

- AI/IS systems suffer from the *Knowledge Acquisition Bottleneck*
- One of the reasons for the AI Winter at that time

Today

From 1950–2020...

- Concepts have been around for a long time (Semantic Networks, Frames, Description Logic, ...)

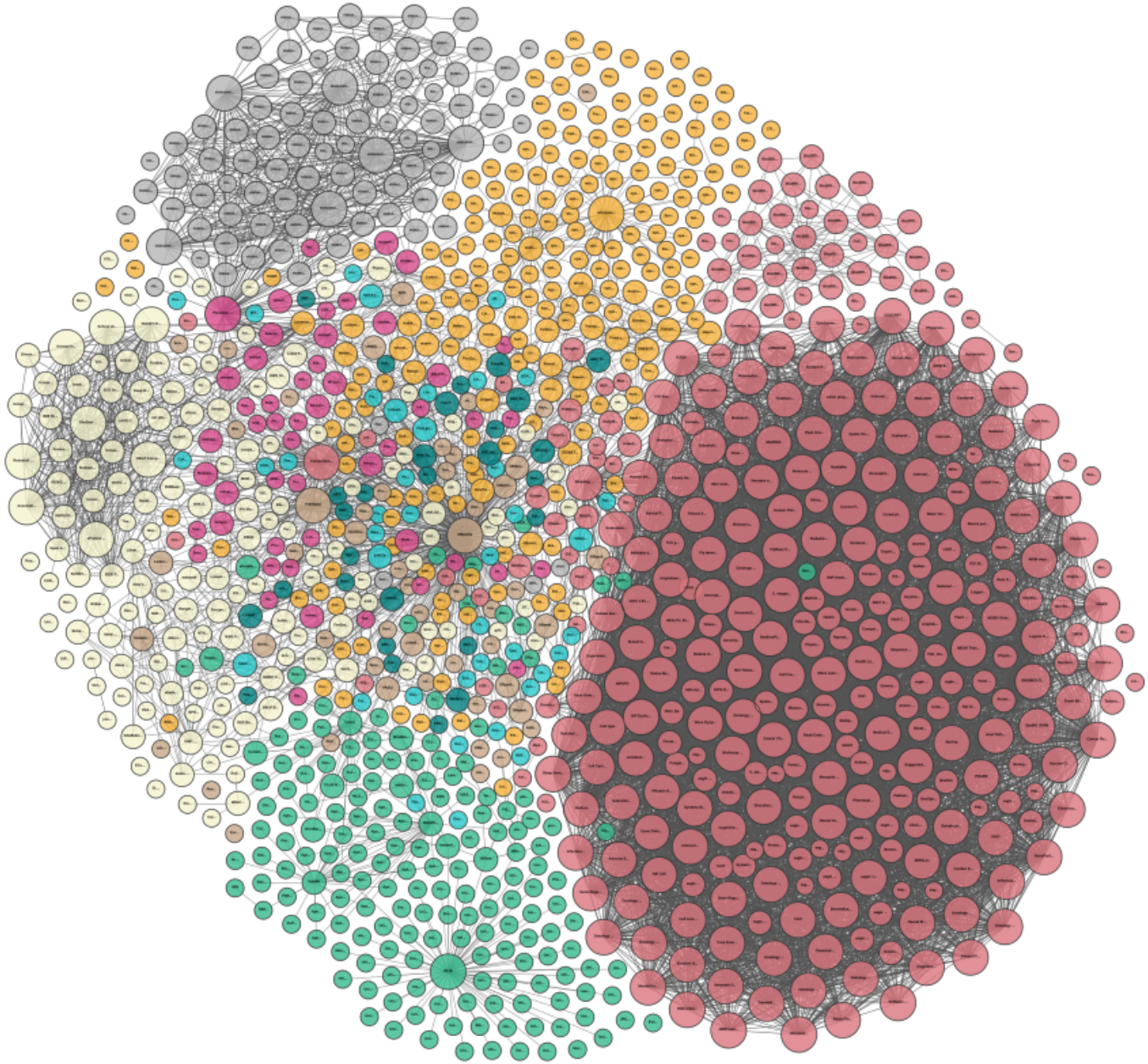
1980s/90s

- AI/IS systems suffer from the *Knowledge Acquisition Bottleneck*
- One of the reasons for the AI Winter at that time

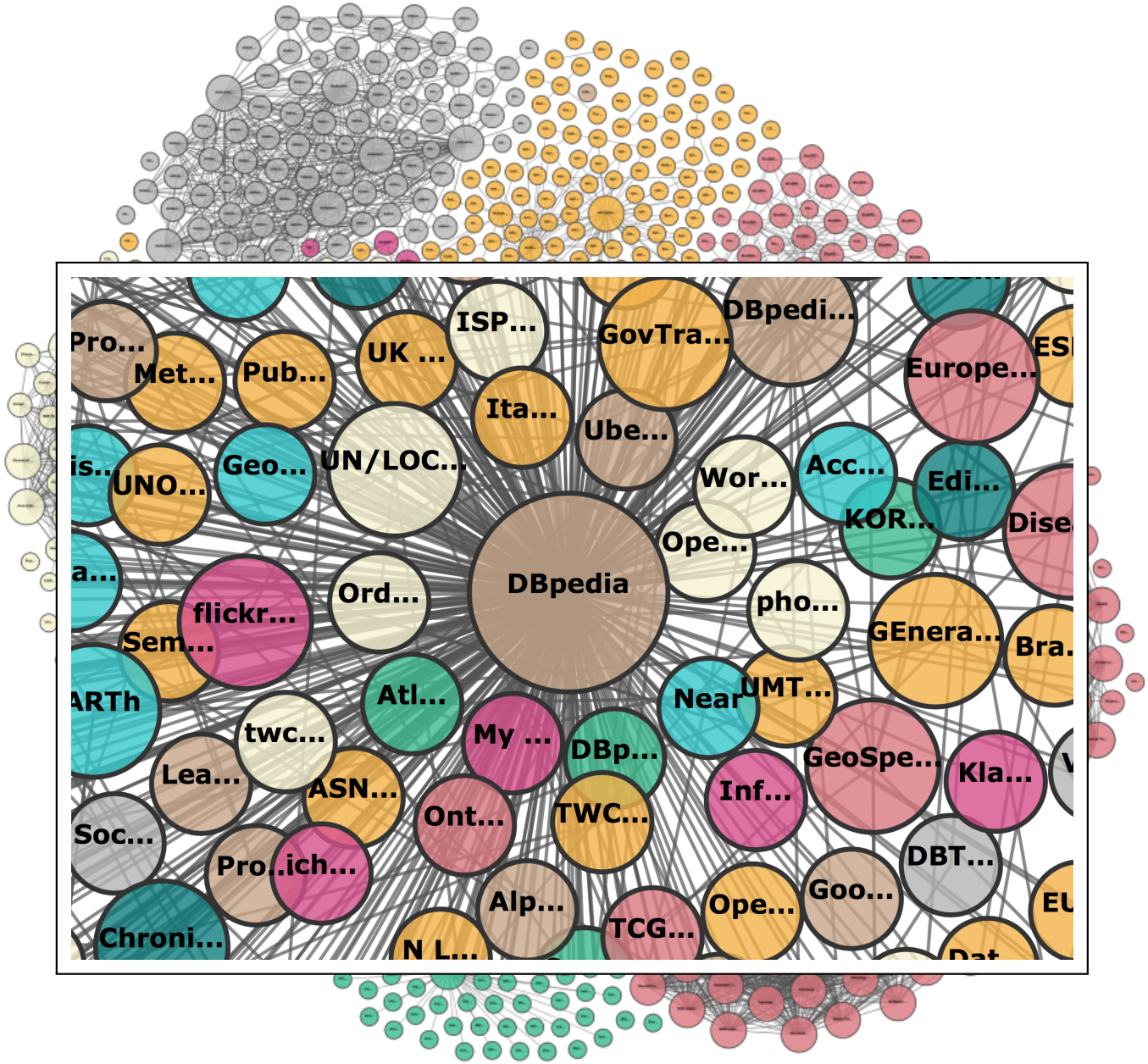
Technology

- Open standards, based on W3C recommendations, e.g., RDF
- Proprietary products, e.g., Neo4J or Oracle Spatial and Graph
- We now have substantial knowledge bases available, both proprietary
  (e.g., Facebook Graph Search, Google Knowledge Graph) and open access (e.g., Wikidata, DBpedia, YAGO)

The Linked Open Data Cloud



The Linked Open Data Cloud



TBL at TED on "The Next Web" (2009)

Tim Berners-Lee: The next Web of open, linked data

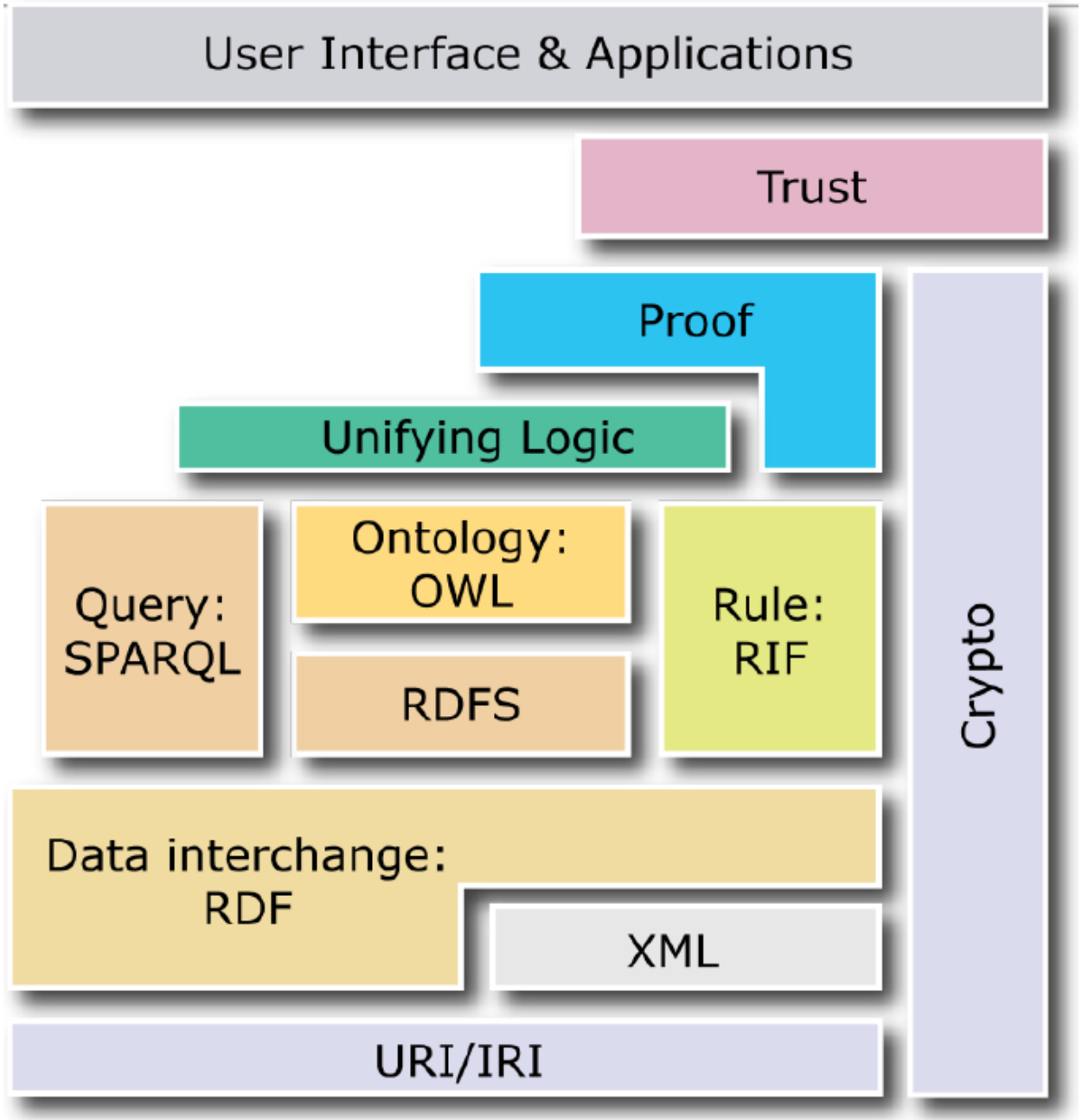
https://www.youtube.com/watch?v=OM6XIICm_qo

Outline

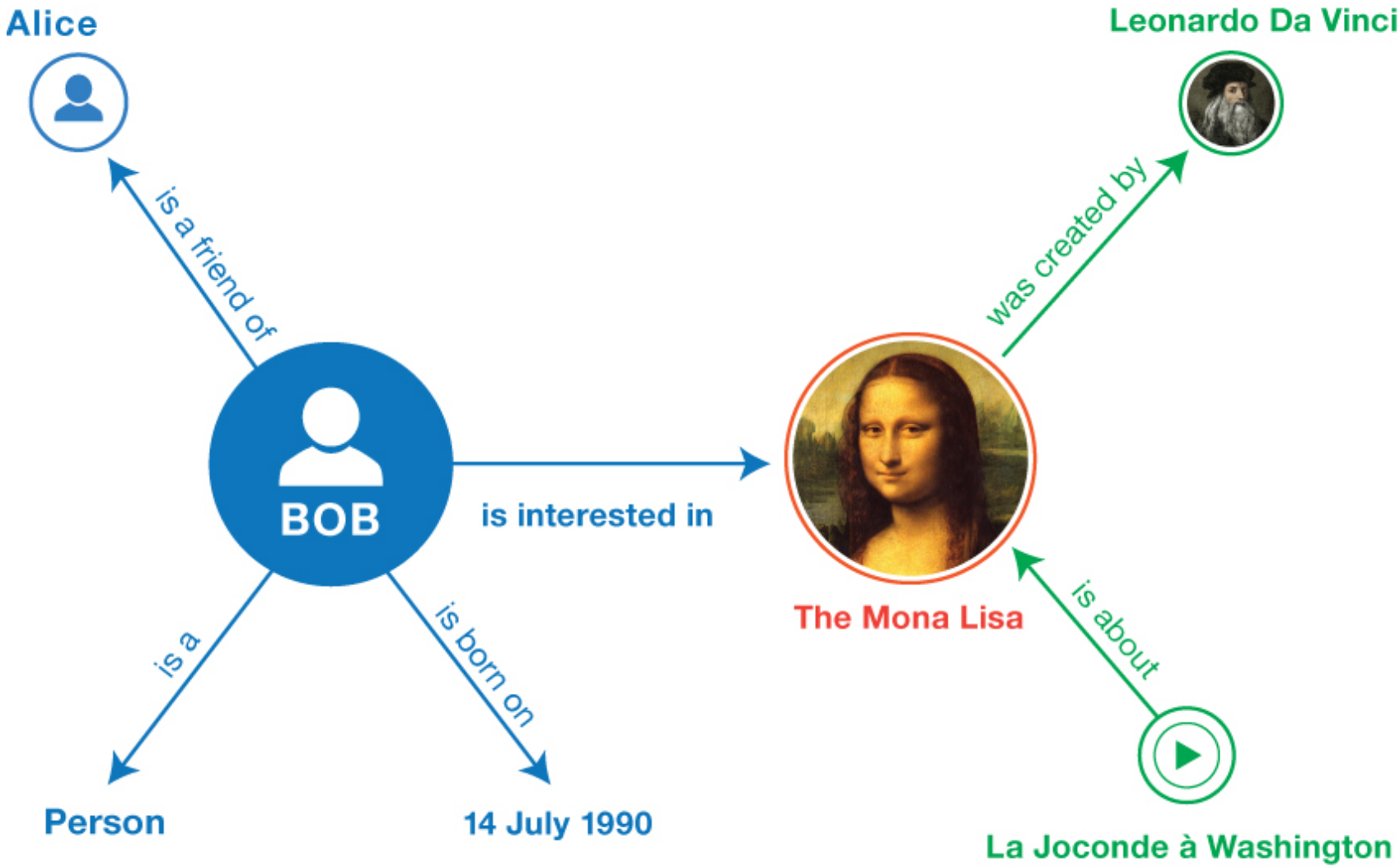
1. Introduction
2. The Resource Description Framework (RDF)
   - Introduction
   - RDF Triples
   - Literals
   - Namespaces
   - Serialization
   - Programming
3. Example: Data Integration with Knowledge Graphs
4. Notes and Further Reading


The Basis: RDF

The W3C "Layer Cake"



Knowledge as Graphs



https://www.w3.org/TR/rdf11-primer/

→ Worksheet #7: "Your first Knowledge Graph" & "Graph Updates"

Triples

Representation of Knowledge Graphs

In a system, we represent graphs in form of triples:

```
<subject> <predicate> <object>
```

(The *predicate* is sometimes called *property*.)

Triples

Representation of Knowledge Graphs

In a system, we represent graphs in form of triples:

```
<subject> <predicate> <object>
```

(The *predicate* is sometimes called *property*.)

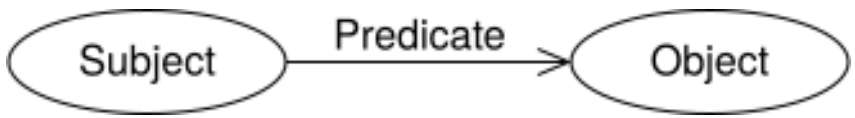
Examples

```
<Bob> <is a> <person>.
<Bob> <is a friend of> <Alice>.
<Bob> <is born on> <the 14th of July 1990>.
<Bob> <is interested in> <the Mona Lisa>.
<the Mona Lisa> <was created by> <Leonardo da
Vinci>.
```

→ **Worksheet #7: "Triples"**

Graphs vs. Triples

<subject> <predicate> <object>



→ Worksheet #7: "More Triples"

RDF Triples

The Resource Description Framework (RDF)

W3C (World Wide Web Consortium) standard ("recommendation")

- first public draft 1997
- RDF 1.0 in 1999; revised in 2004
- RDF 1.1 in 2014 (current version)

Family of standards: RDF, RDFS, RDFa, Turtle, N3, SPARQL, ...

RDF Triples

Format of triples

In RDF,

- Subject and predicate must be URIs (IRIs)
- Object can be IRI or literal

RDF Triples

Format of triples

In RDF,

- Subject and predicate must be URIs (IRIs)
- Object can be IRI or literal

Examples

```
<http://www.wikidata.org/entity/Q12418>
    <http://purl.org/dc/terms/title>
    "Mona Lisa" .

<http://www.wikidata.org/entity/Q12418>
    <http://purl.org/dc/terms/creator>
    <http://dbpedia.org/resource/Leonardo_da_Vinci> .
```

→ Worksheet #7: “Wikidata” & “Using URIs”

RDF Literals

"Mona Lisa"

In this triple:

```
<http://www.wikidata.org/entity/Q12418>
    <http://purl.org/dc/terms/title> "Mona Lisa" .
```

"Mona Lisa" is a string literal

RDF Literals

"Mona Lisa"

In this triple:

```
<http://www.wikidata.org/entity/Q12418>
    <http://purl.org/dc/terms/title> "Mona Lisa" .
```

"Mona Lisa" is a string literal

Things to know about literals

- Literals have a datatype, e.g., `string` or `int`
- Strings can have a language tag, e.g.,

  ```
  "Leonardo da Vinci"@en
  "Léonard de Vinci"@fr
  ```

- Strings are often used to provide human-readable labels
- For strings only, datatype can be omitted:
  `"Mona Lisa"` is equivalent to `"Mona Lisa"^^xsd:string`
- Again, literals can only appear in the object position of a triple

All the details about datatypes:

https://www.w3.org/TR/2014/REC-rdf11-concepts-20140225/#section-Datatypes

Namespaces

Shortening URIs

Instead of always writing full URIs (IRIs), we can split them into a prefix and suffix, e.g.:
<http://dbpedia.org/resource/Leonardo_da_Vinci>

- We define a prefix dbpedia:
  `PREFIX dbpedia: <http://dbpedia.org/resource/>`
- and now we can simple write:
  `dbpedia:Leonardo_da_Vinci`
- Note: angle brackets <> only for full IRIs

→ reduces dataset sizes, easier to read

Namespaces

Shortening URIs

Instead of always writing full URIs (IRIs), we can split them into a prefix and suffix, e.g.:
<http://dbpedia.org/resource/Leonardo_da_Vinci>

- We define a prefix dbpedia:
  `PREFIX dbpedia: <http://dbpedia.org/resource/>`
- and now we can simple write:
  `dbpedia:Leonardo_da_Vinci`
- Note: angle brackets <> only for full IRIs

→ reduces dataset sizes, easier to read

Conventions

Commonly used URLs use the same namespace prefix

- E.g., FOAF (friend-of-a-friend):
  `PREFIX foaf: <http://xmlns.com/foaf/0.1/>`
- Lookup a prefix at https://prefix.cc/

→ Worksheet #7: "More URIs" & "Namespaces"

Serialization

Formats

There is no single format `.rdf` (like `.xml`), commonly used are:

- RDF/XML for data exchange (somewhat deprecated)
- RDFa for embedding RDF into web pages
- N-Triples (N3) for streaming RDF data and bulk dataset up-/download
- Turtle for human-readable files
- JSON-LD for web applications

plus some variations/extensions.

Serialization

Formats

There is no single format `.rdf` (like `.xml`), commonly used are:

- RDF/XML for data exchange (somewhat deprecated)
- RDFa for embedding RDF into web pages
- N-Triples (N3) for streaming RDF data and bulk dataset up-/download
- Turtle for human-readable files
- JSON-LD for web applications

plus some variations/extensions.

N-Triples

So far, we've mostly used the N-Triples format:

```
<http://www.wikidata.org/entity/Q12418>
      <http://purl.org/dc/terms/title> "Mona Lisa"
.
```

each line in a file is one triple, full IRIs only (no namespace prefixes) and ended by a period '.'

Turtle

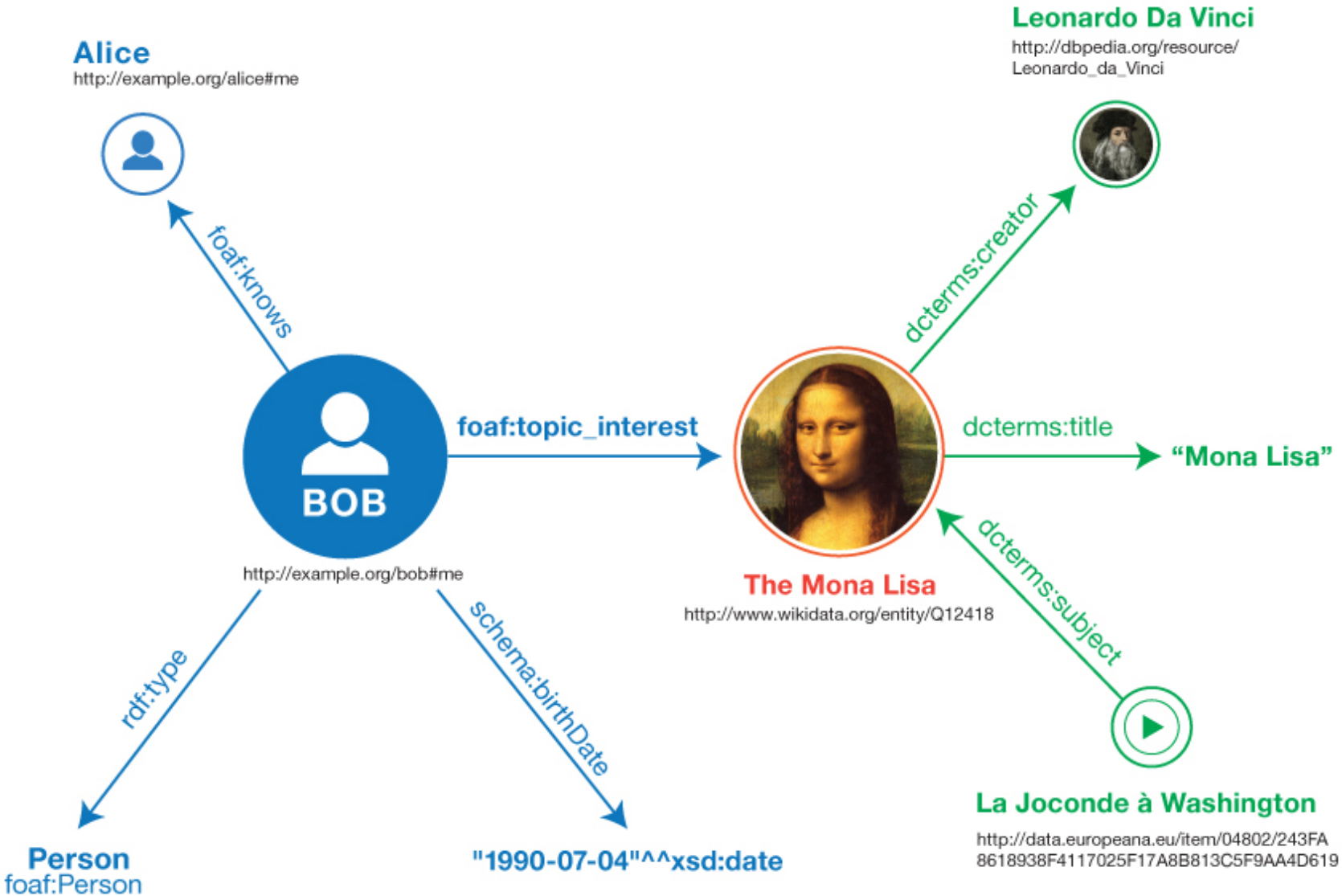
```
BASE  <http://example.org/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema>
PREFIX schema: <http://schema.org/>
PREFIX dcterms: <http://purl.org/dc/terms/>
PREFIX wd: <http://www.wikidata.org/entity/>

<bob#me>
    a foaf:Person ;
    foaf:knows <alice#me> ;
    schema:birthDate "1990-07-04"^^xsd:date ;
    foaf:topic_interest wd:Q12418 .

wd:Q12418
    dcterms:title "Mona␣Lisa" ;
    dcterms:creator <http://dbpedia.org/resource/Leonardo_da_Vinci> .

<http://data.europeana.eu/item/04802/243FA8618938F4117025F17A8B813C5F9A
    dcterms:subject wd:Q12418 .
```

Graph corresponding to the Turtle example



https://www.w3.org/TR/rdf11-primer/

RDF in programming practice

- For example, using Python+RDFLib:
  - a “Graph” object is created
  - the RDF file is parsed and results stored in the Graph
  - the Graph offers methods to retrieve (or add):
    - triples
    - (property,object) pairs for a specific subject
    - (subject,property) pairs for specific object
    - etc.
  - the rest is conventional programming…
- Similar tools exist in Java, PHP, etc.

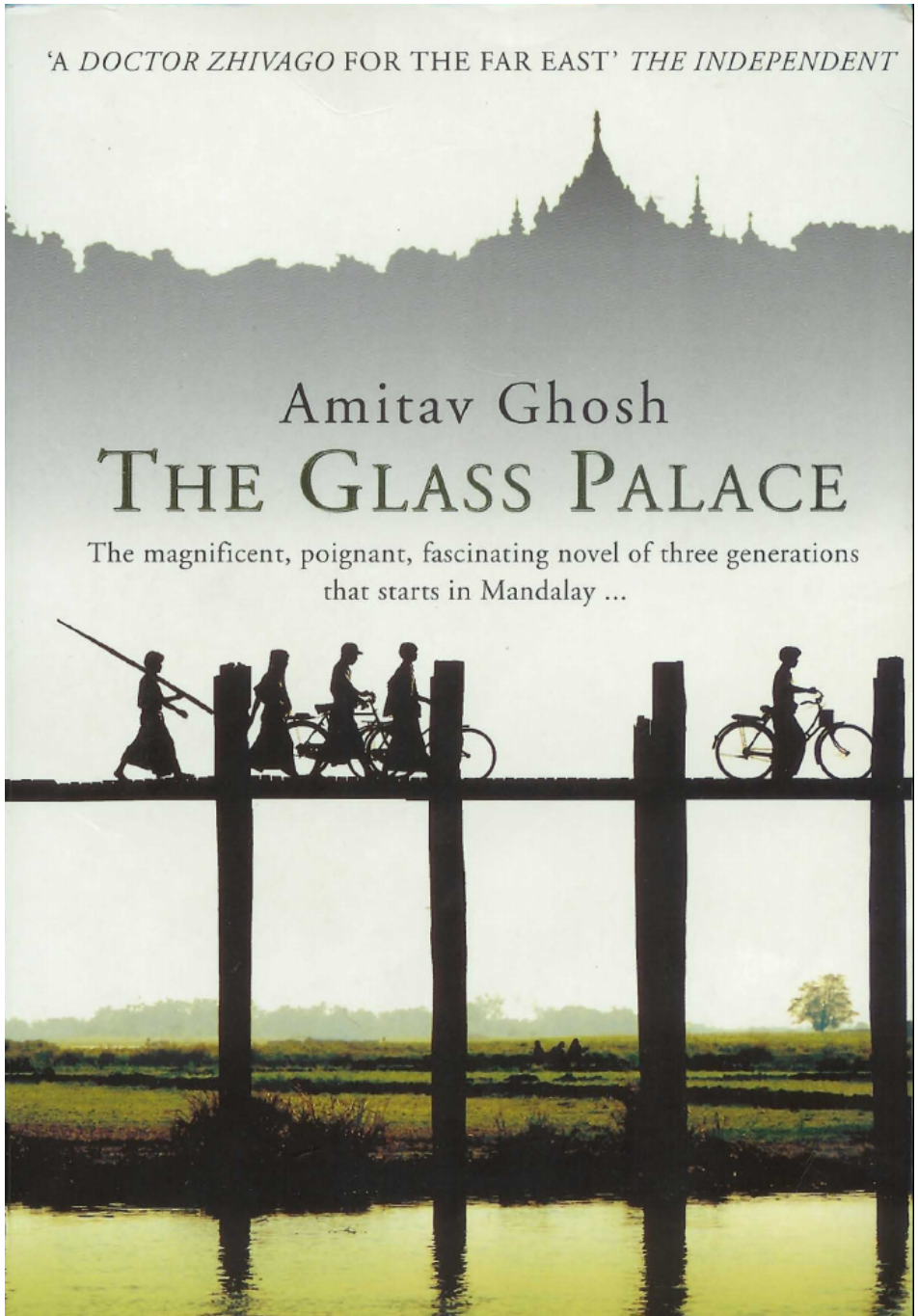
Python example using RDFLib

```
# create a graph from a file
graph = rdflib.Graph()
graph.parse("filename.rdf", format="rdfxml")
# take subject with a known URI
subject = rdflib.URIRef("URI_of_Subject")
# process all properties and objects for this subject
for (s,p,o) in graph.triples((subject,None,None)) :
  do_something(p,o)
```

Outline

1. Introduction
2. The Resource Description Framework (RDF)
3. Example: Data Integration with Knowledge Graphs
4. Notes and Further Reading

Example: Let's start with a Book...



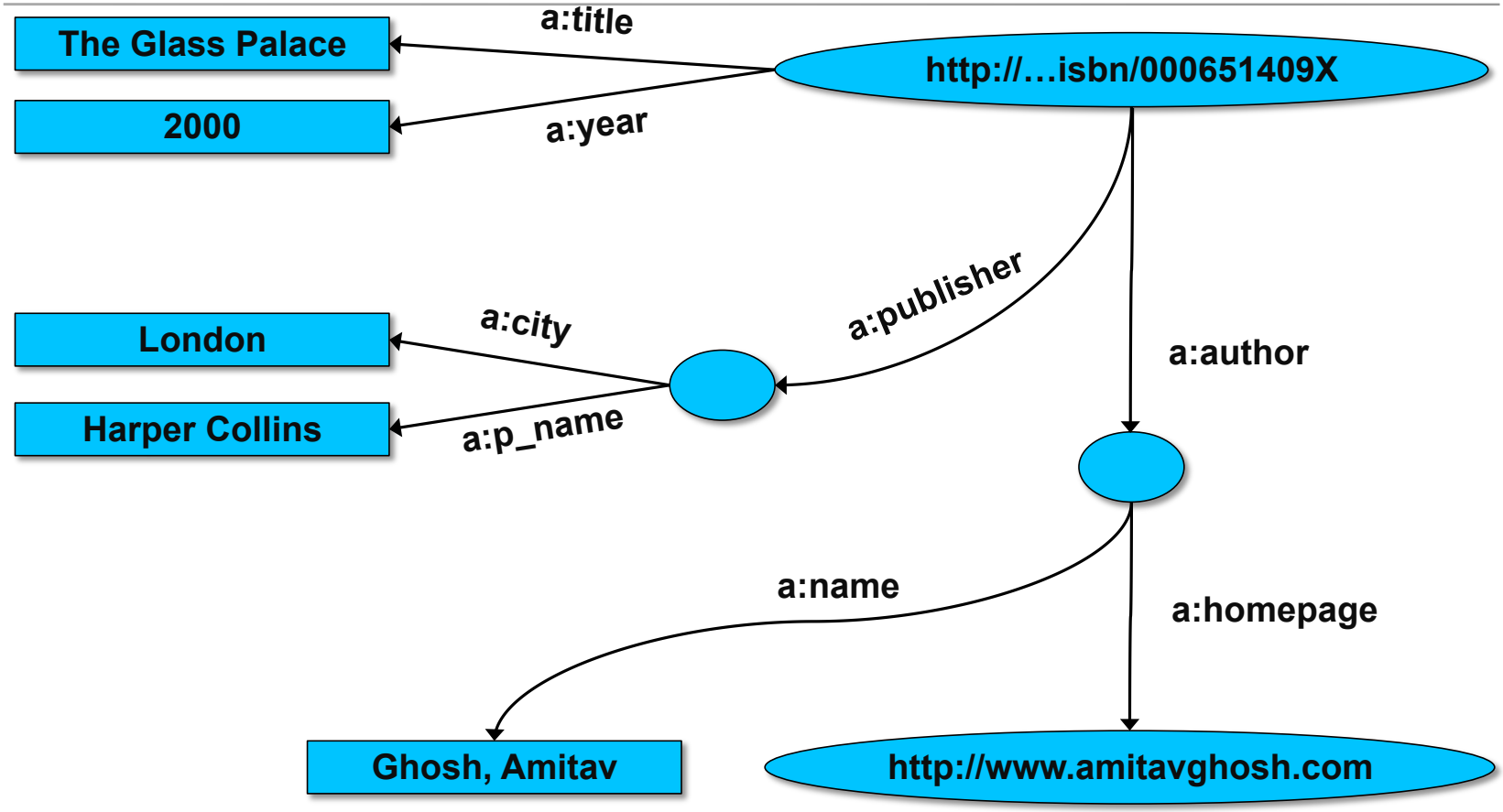
A simplified bookstore data (dataset “A”)

ISBN	Author	Title	Publisher	Year
0006511409X	id_xyz	The Glass Palace	id_qpr	2000

ID	Name	Homepage
id_xyz	Ghosh, Amitav	http://www.amitavghosh.com

ID	Publisher’s name	City
id_qpr	Harper Collins	London

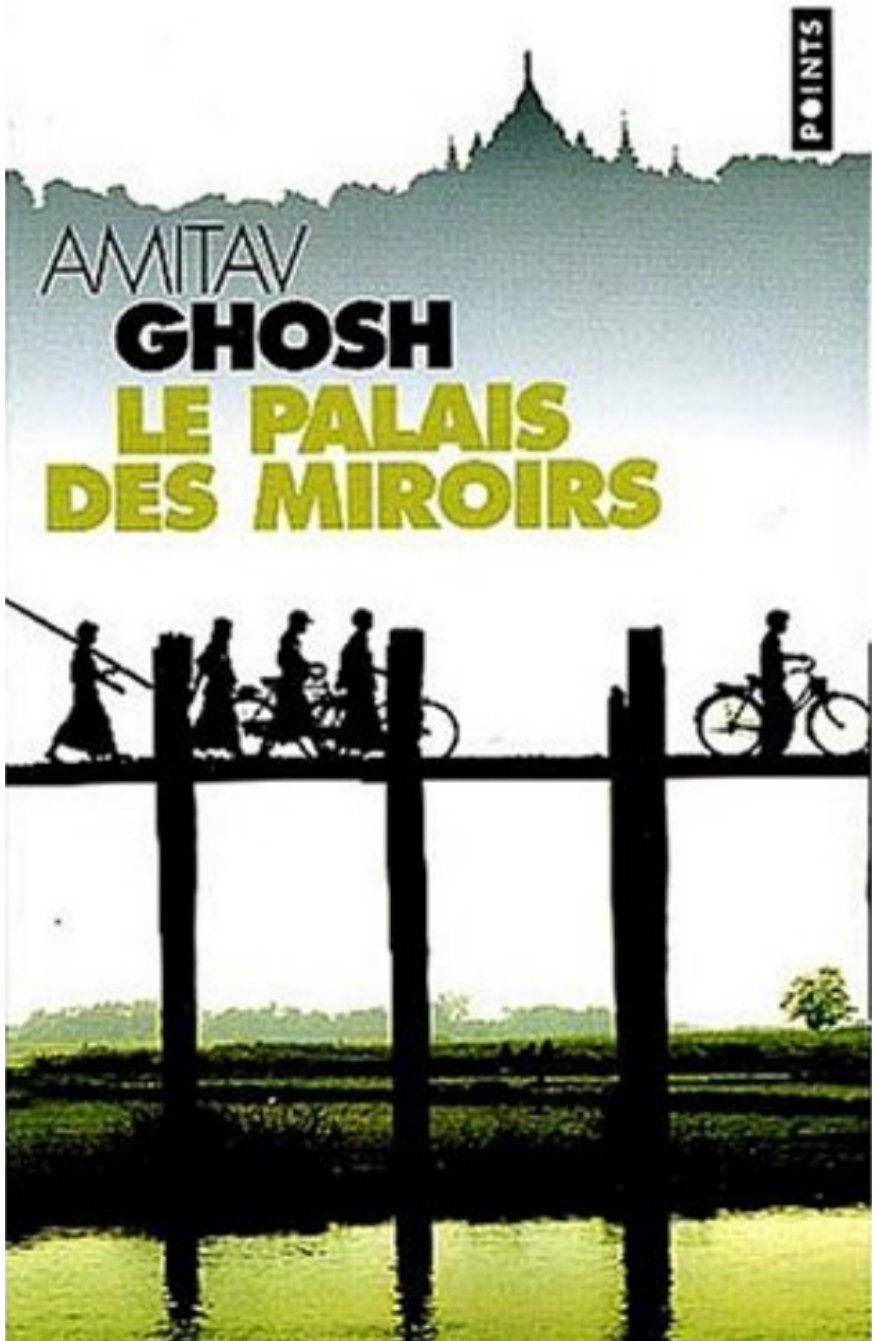
1st: export your data as a set of *relations*



Some notes on the exporting the data

- Relations form a graph
  - the nodes refer to the “real” data or contain some literal
  - how the graph is represented in machine is immaterial for now

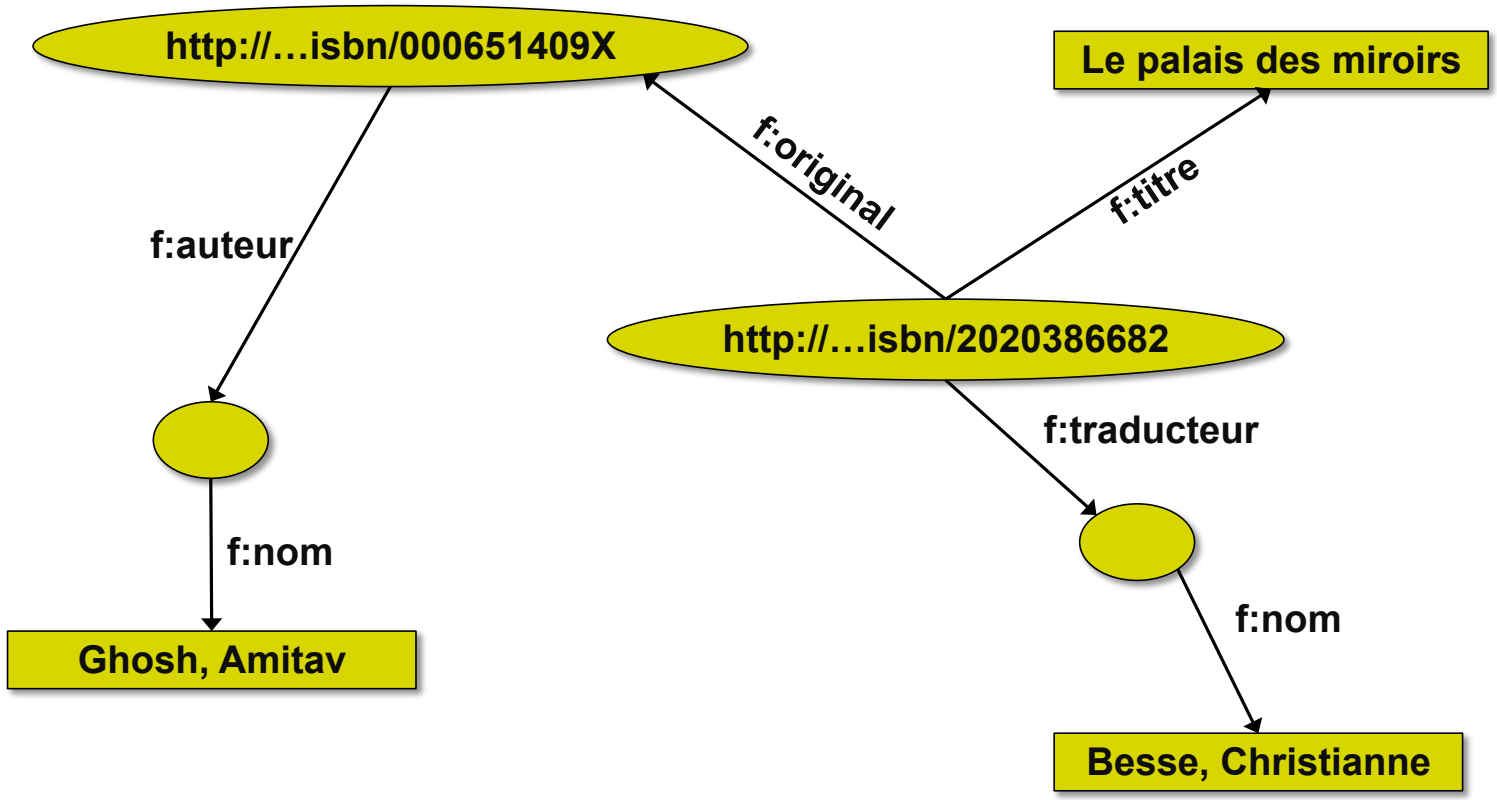
Now the same book in French...



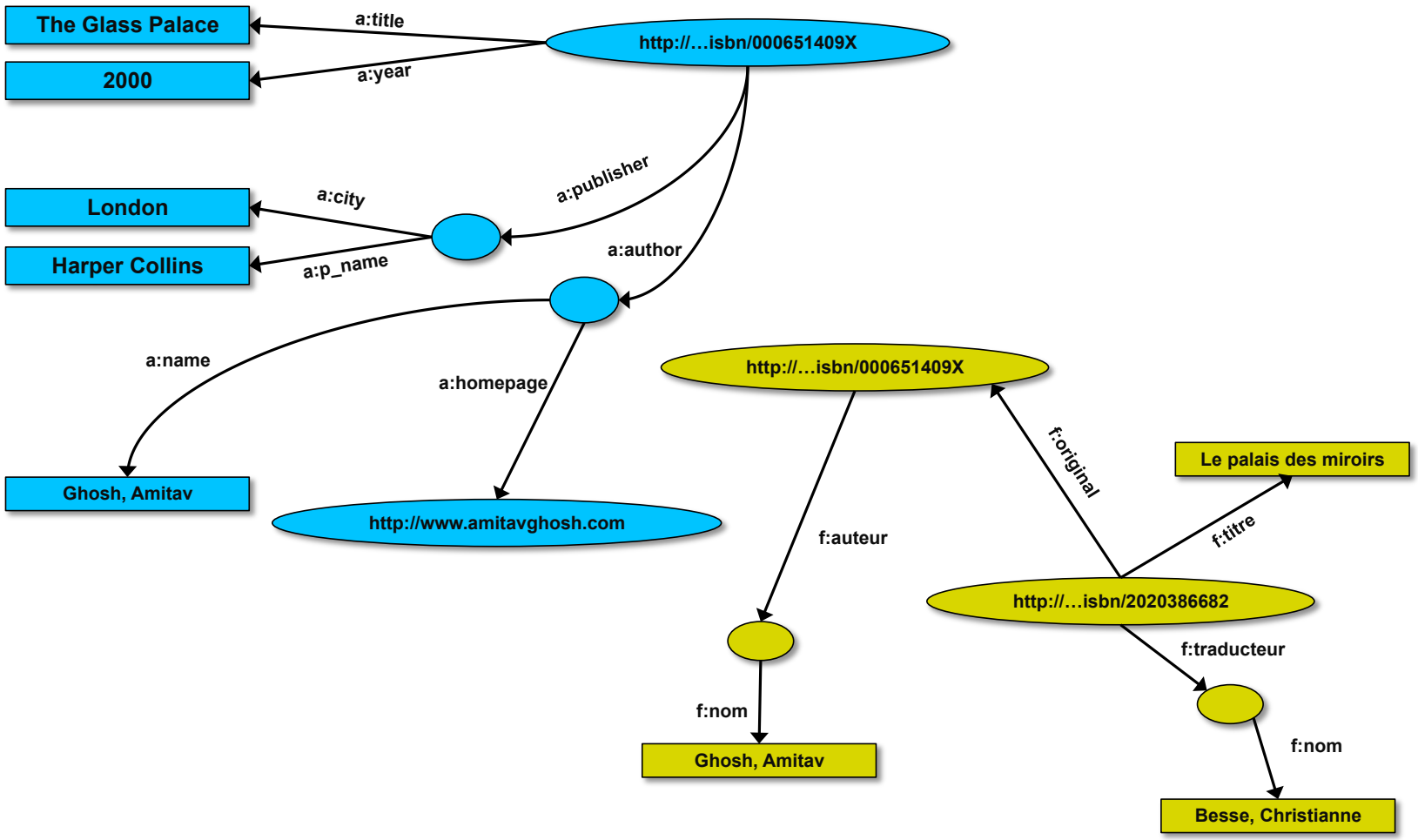
Another bookstore data (dataset “F”)

	A	B	C	D
1	**ID**	**Titre**	**Traducteur**	**Original**
2	ISBN 2020286682	Le Palais des Miroirs	$A12$	ISBN 0-00-6511409-X
3				
4				
5				
6	**ID**	**Auteur**		
7	ISBN 0-00-6511409-X	$A11$		
8				
9				
10	**Nom**			
11	Ghosh, Amitav			
12	Besse, Christianne			

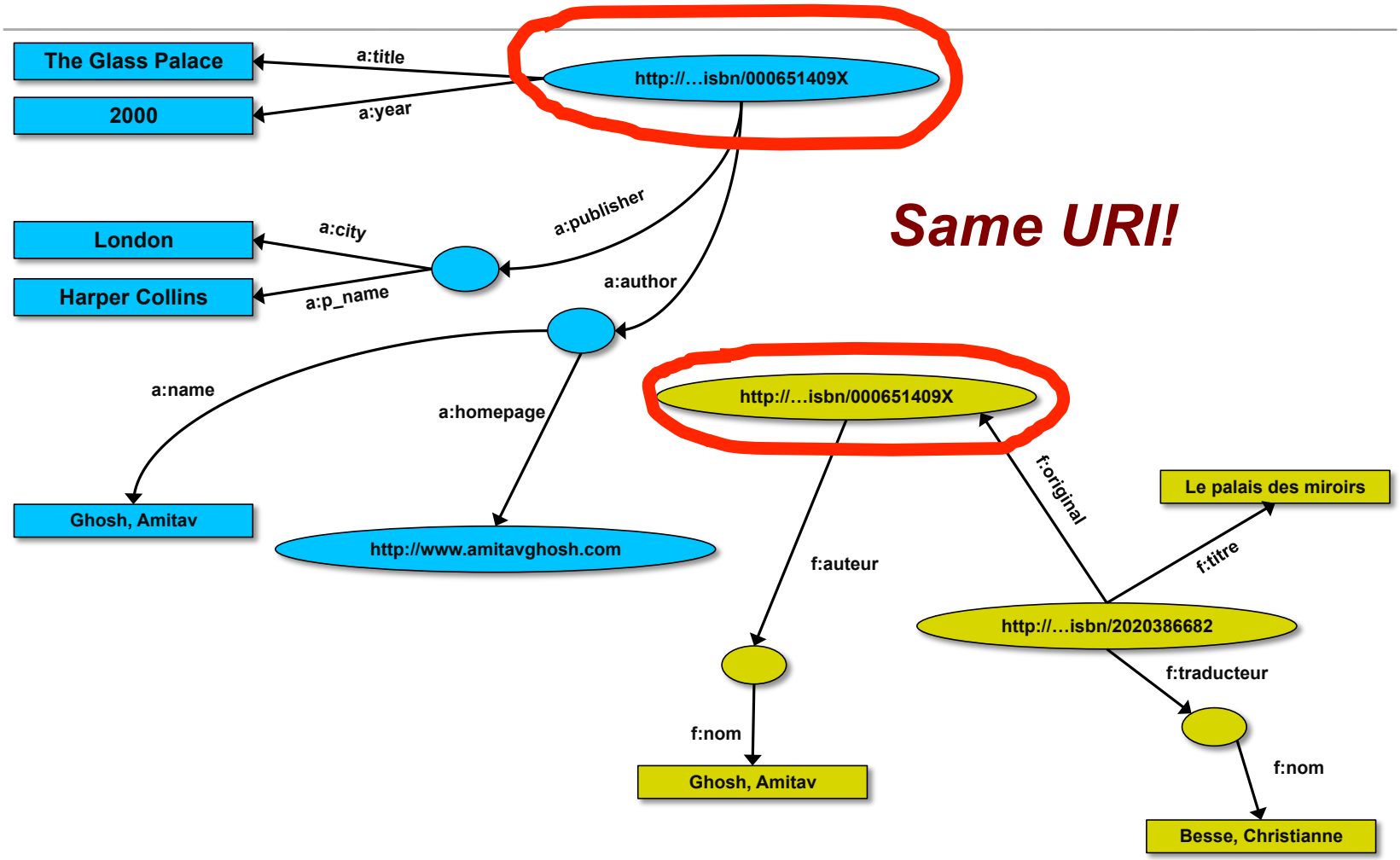
2nd: export your second set of data



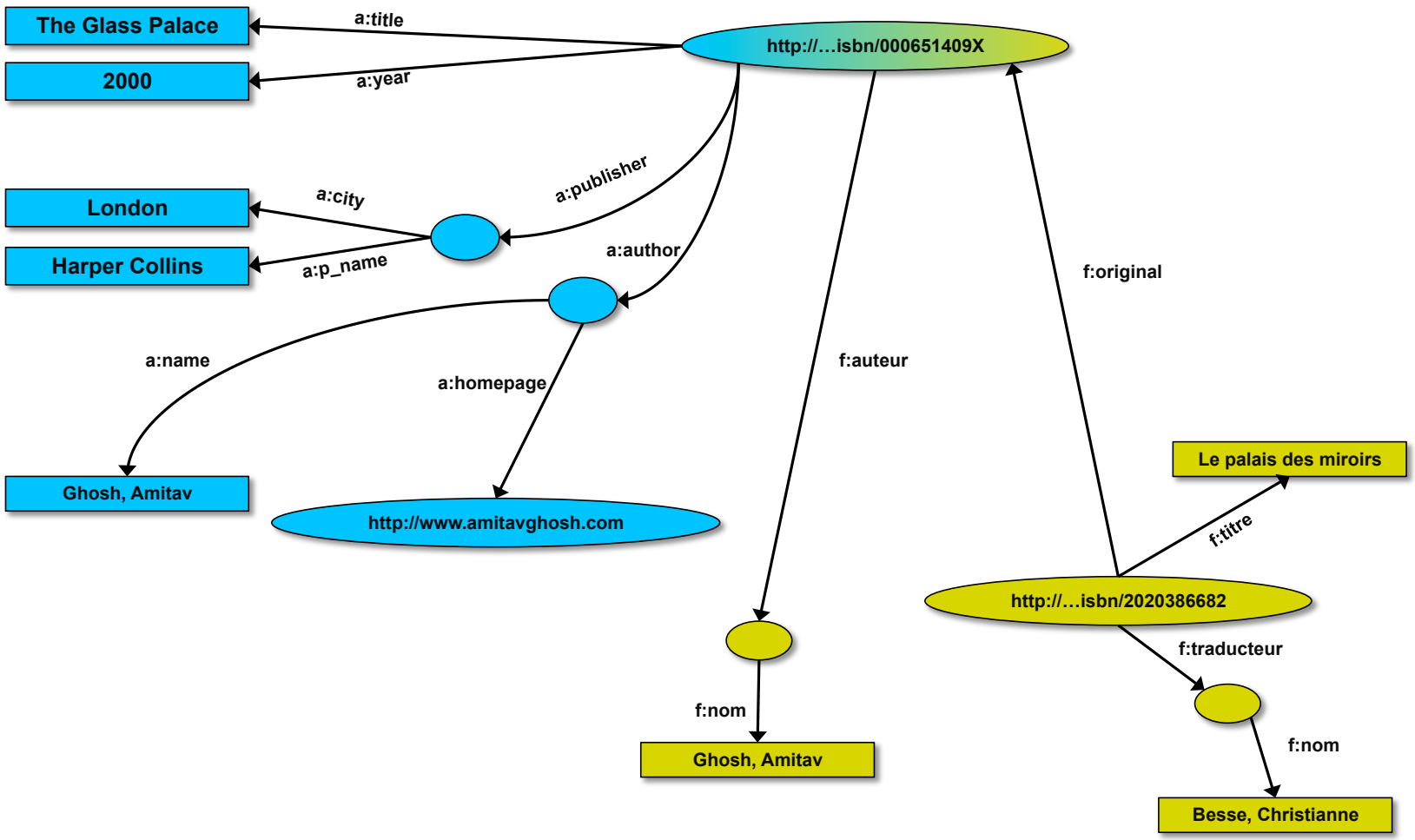
3rd: start merging your data



3rd: start merging your data (cont)

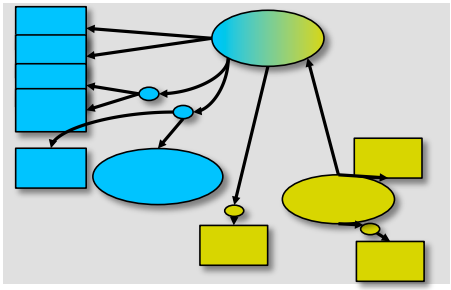


3rd: start merging your data



Start making queries…

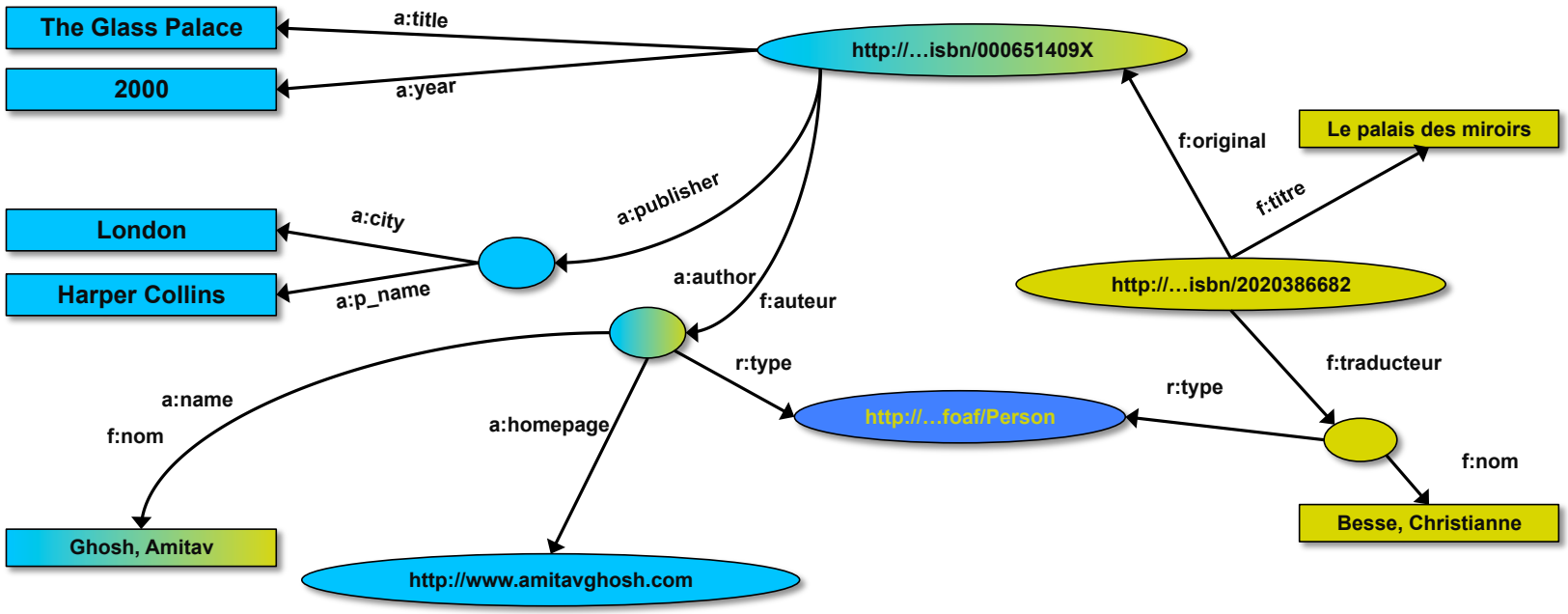
- User of data “F” can now ask queries like:
  - “give me the title of the original”
    - well, … « donnes-moi le titre de l’original »
- This information is not in the dataset “F”…
- …but can be retrieved by merging with dataset “A”!

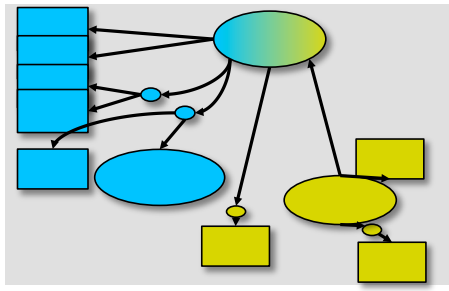


However, more can be achieved…

- We “feel” that a:author and f:auteur should be the same
- But an automatic merge doest not know that!
- Let us add some extra information to the merged data:
  - a:author same as f:auteur
  - both identify a “Person”
  - a term that a community may have already defined:
    - a “Person” is uniquely identified by his/her name and, say, homepage
    - it can be used as a “category” for certain type of resources

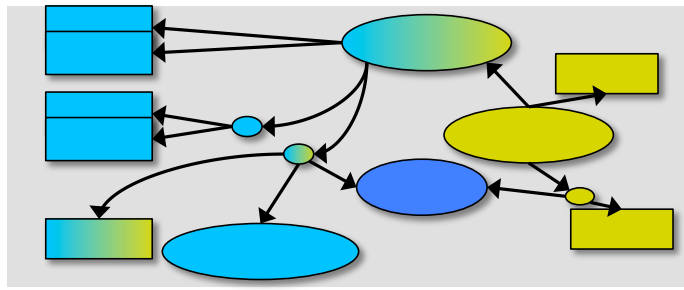
3rd revisited: use the extra knowledge





Start making richer queries!

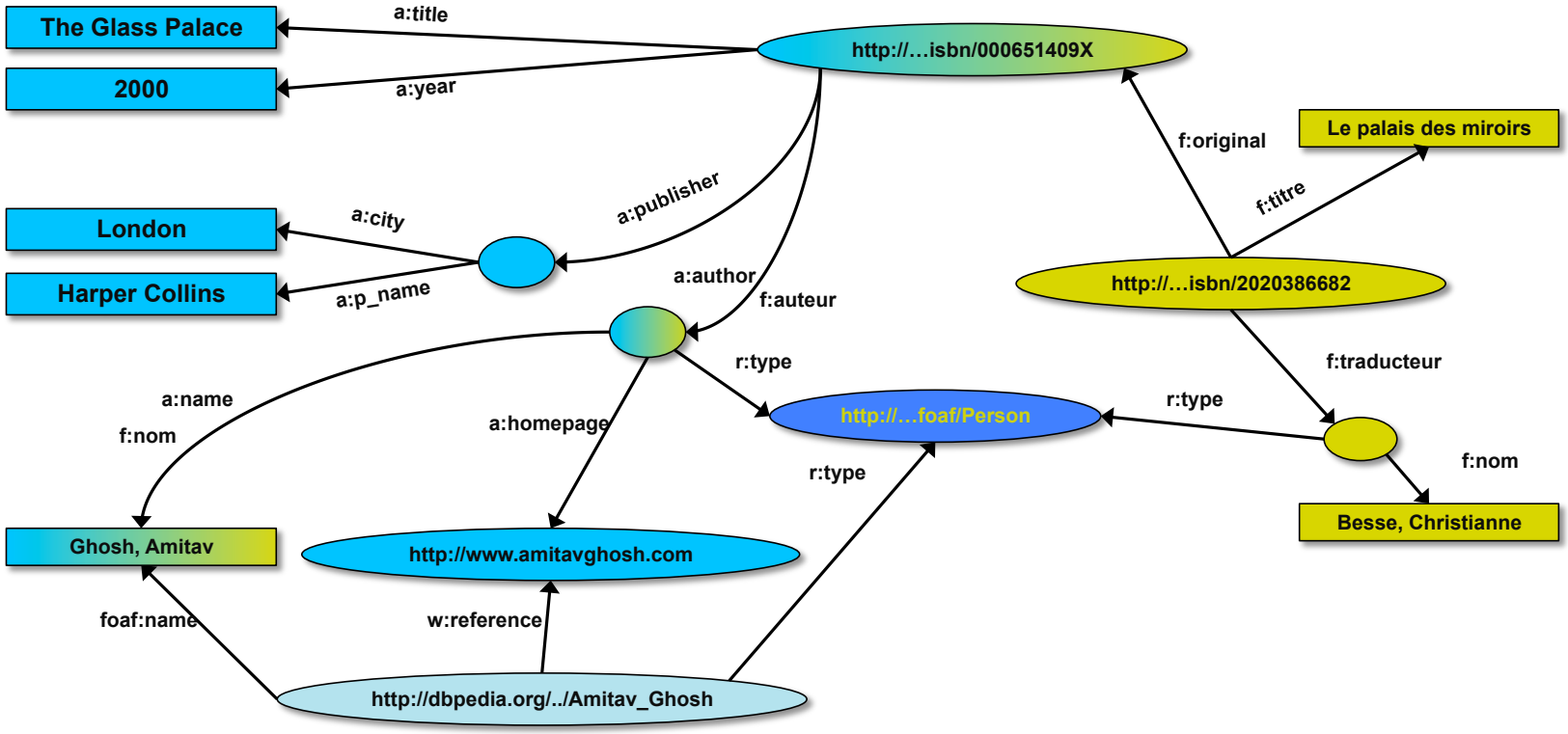
- User of dataset “F” can now query:
  - “donnes-moi la page d’accueil de l’auteur de l’original”
    - well… “give me the home page of the original’s ‘auteur’”
- The information is not in datasets “F” or “A”…
- …but was made available by:
  - merging datasets “A” and datasets “F”
  - adding three simple extra statements as an extra “glue”



Combine with different datasets

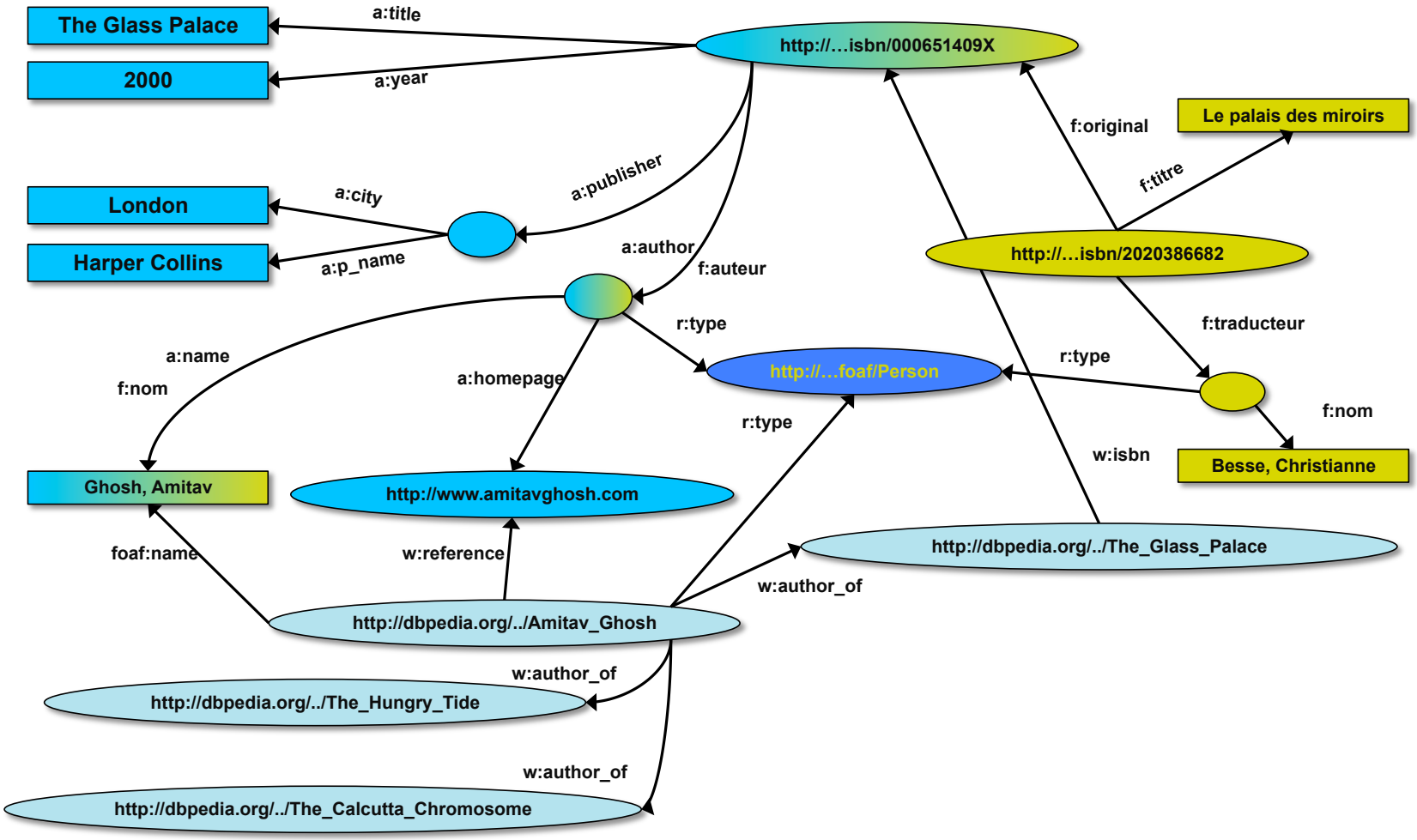
- Using, e.g., the “Person”, the dataset can be combined with other sources
- For example, data in Wikipedia can be extracted using dedicated tools
  - e.g., the “dbpedia” project can extract the “infobox” information from Wikipedia already…

Merge with Wikipedia data

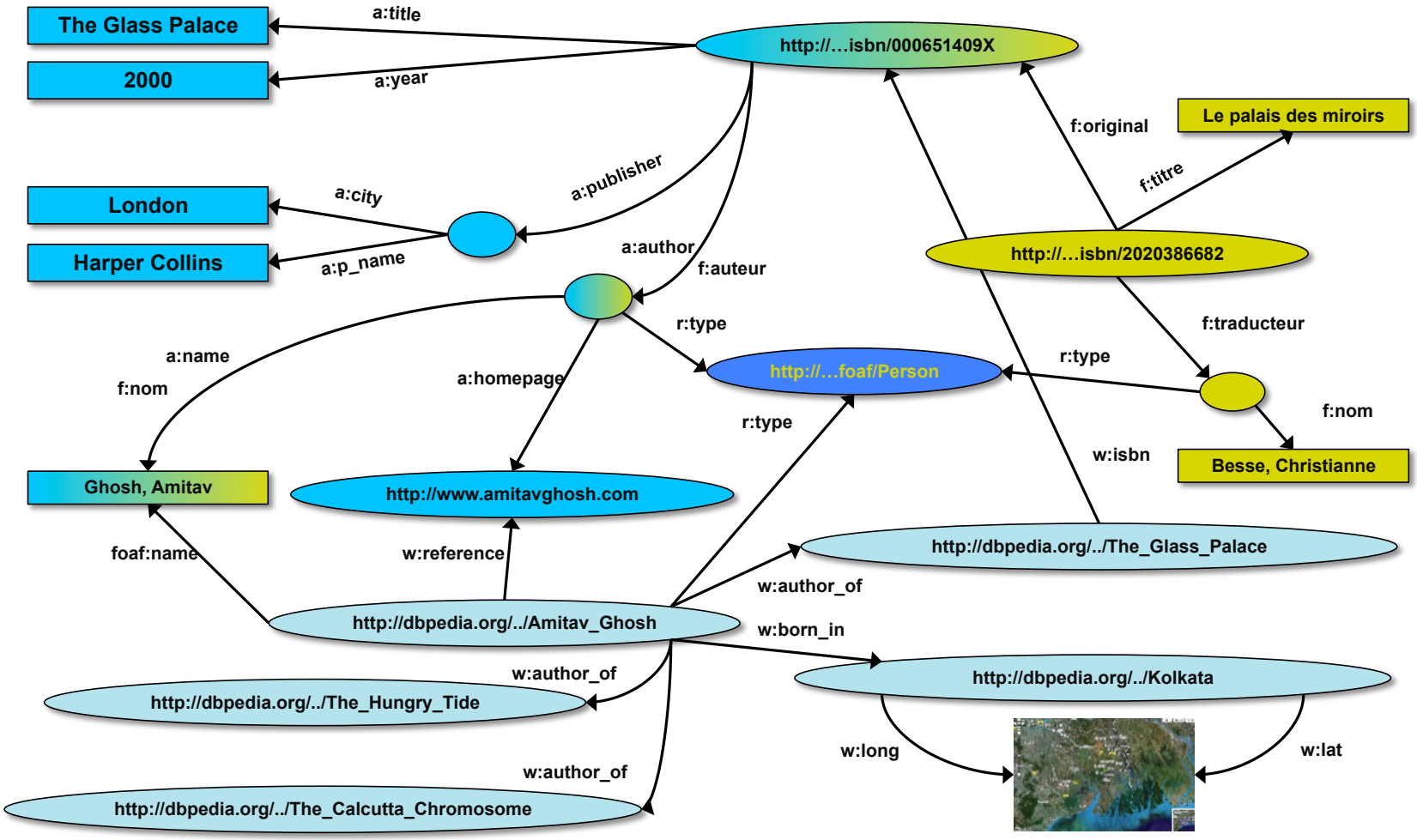


→ Worksheet #7: "Enriching Graphs"

Merge with Wikipedia data



Merge with Wikipedia data



Is that surprising?

- It may look like it but, in fact, it should not be…
- What happened via automatic means is done every day by Web users!
- The difference: a bit of extra rigour so that machines could do this, too

→ Worksheet #7: "AI Agent"

Outline

1. Introduction
2. The Resource Description Framework (RDF)
3. Example: Data Integration with Knowledge Graphs
4. Notes and Further Reading

Reading Material

Required

- [Wor14, Sections 1–3] (RDF Primer)

Supplemental

- [Yu14, Chapters 1, 2] (Introduction, RDF)

References

[Her] Ivan Herman.
Tutorial on Semantic Web Technologies.
http://www.w3.org/People/Ivan/CorePresentations/RDFTutorial/.

[Wor14] World Wide Web Consortium (W3C).
RDF 1.1 Primer.
http://www.w3.org/TR/rdf11-primer/, 24 June 2014.

[Yu14] Liyang Yu.
A Developer's Guide to the Semantic Web.
Springer-Verlag Berlin Heidelberg, 2nd edition, 2014.
https://concordiauniversity.on.worldcat.org/oclc/897466408.