A Hitchhiker's Guide to Ontology

Fabian M. Suchanek

Télécom ParisTech University,

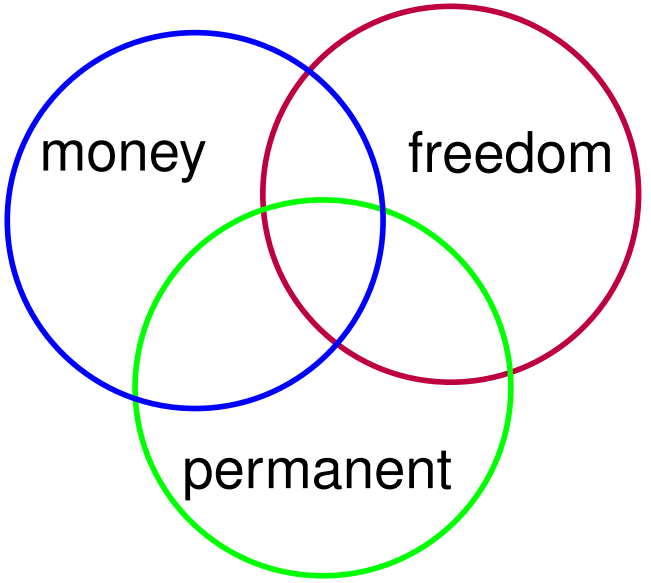
Paris, France

Fabian M. Suchanek

2003

2005

2008



Fabian M. Suchanek

2003: BSc in Cognitive Science

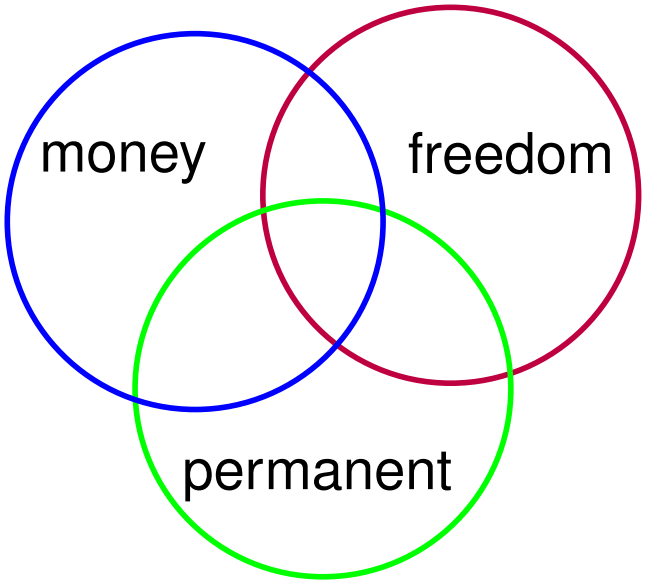
University of Osnabrück/DE

2005: MSc in Computer Science

Saarland University/DE

2008: PhD in Computer Science

Max Planck Institute/DE



Fabian M. Suchanek

2003: BSc in Cognitive Science

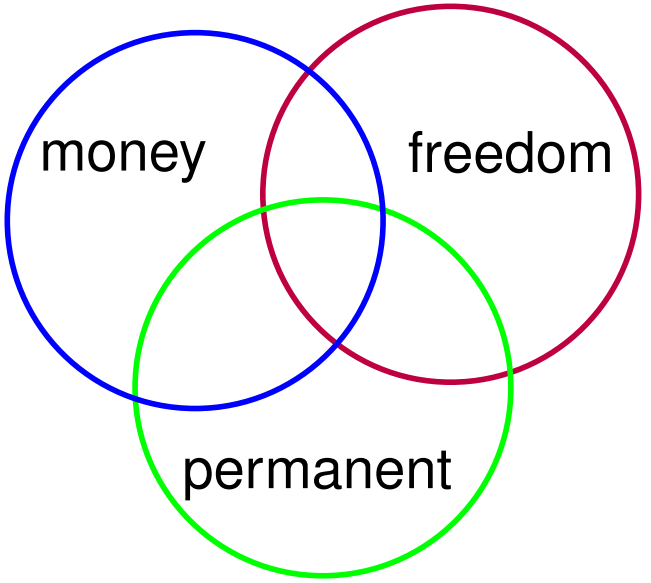
University of Osnabrück/DE

2005: MSc in Computer Science

Saarland University/DE

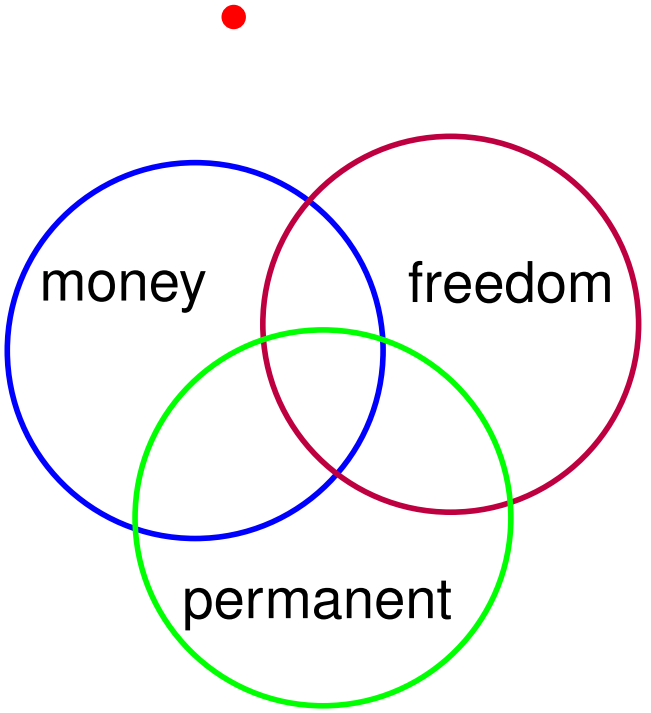
2008: PhD in Computer Science

Max Planck Institute/DE



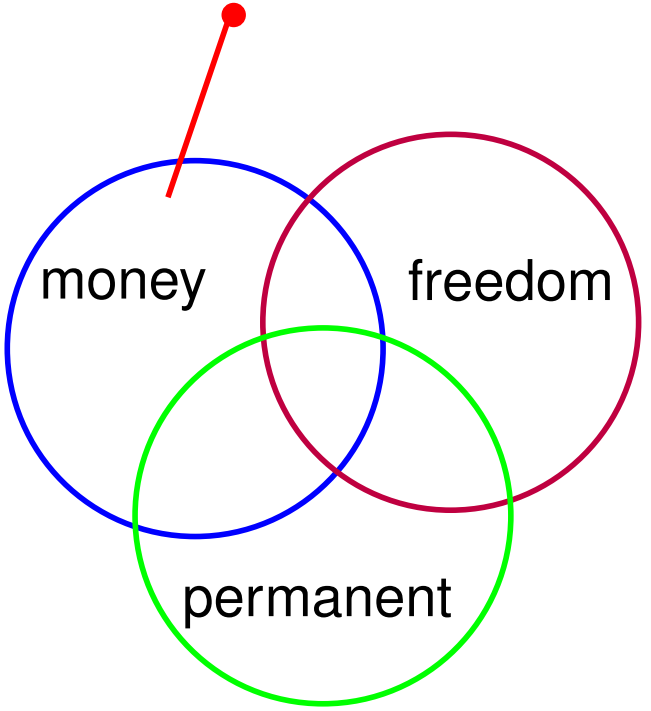
Fabian M. Suchanek

2009: PostDoc at Microsoft Research
Silicon Valley/US



Fabian M. Suchanek

2009: PostDoc at Microsoft Research
Silicon Valley/US



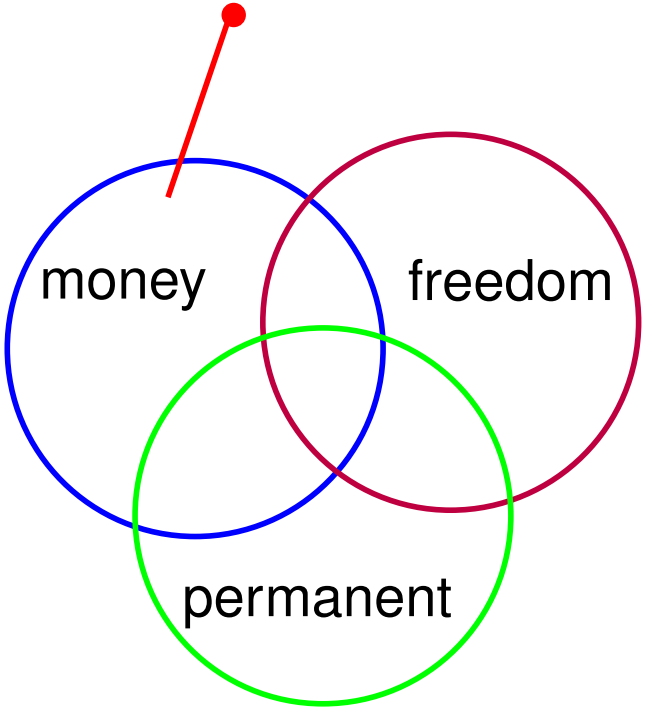
Fabian M. Suchanek

2009: PostDoc at Microsoft Research

Silicon Valley/US

2010: PostDoc

INRIA Saclay/FR



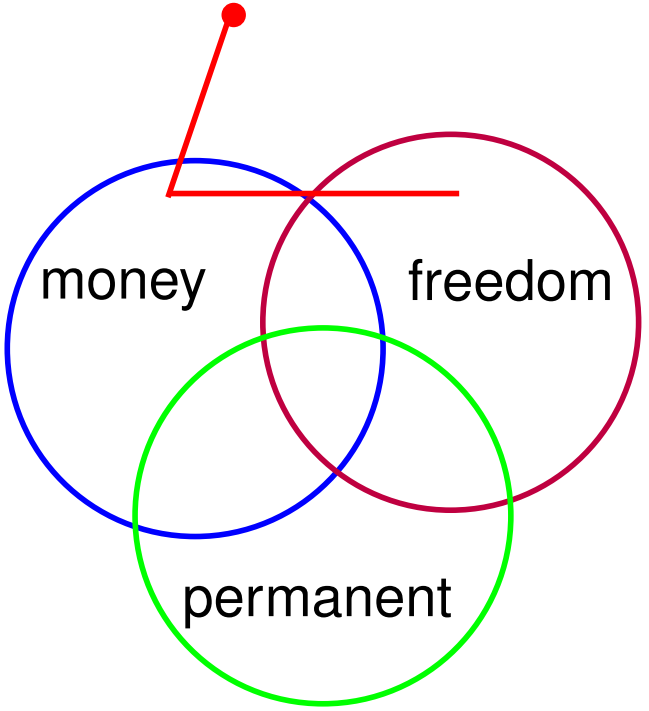
Fabian M. Suchanek

2009: PostDoc at Microsoft Research

Silicon Valley/US

2010: PostDoc

INRIA Saclay/FR



Fabian M. Suchanek

2009: PostDoc at Microsoft Research

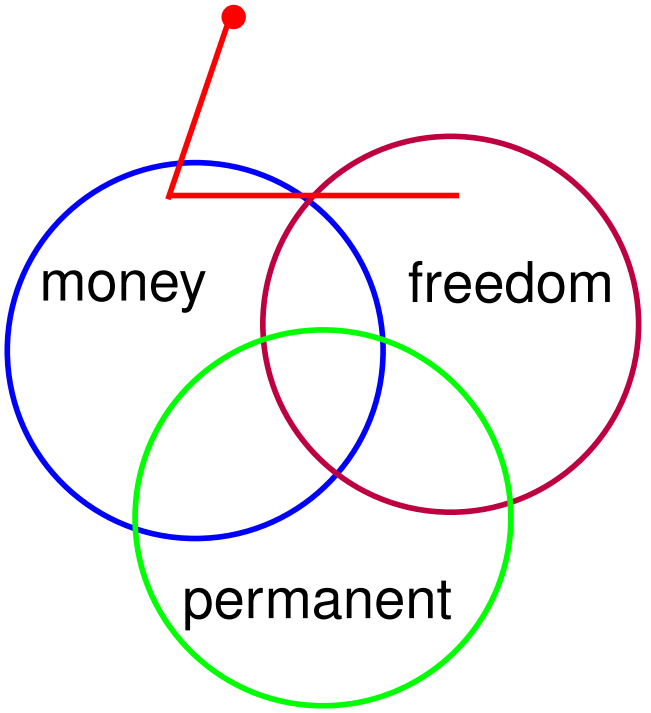
Silicon Valley/US

2010: PostDoc

INRIA Saclay/FR

2012: Research group leader

Max Planck Institute/DE



Fabian M. Suchanek

2009: PostDoc at Microsoft Research

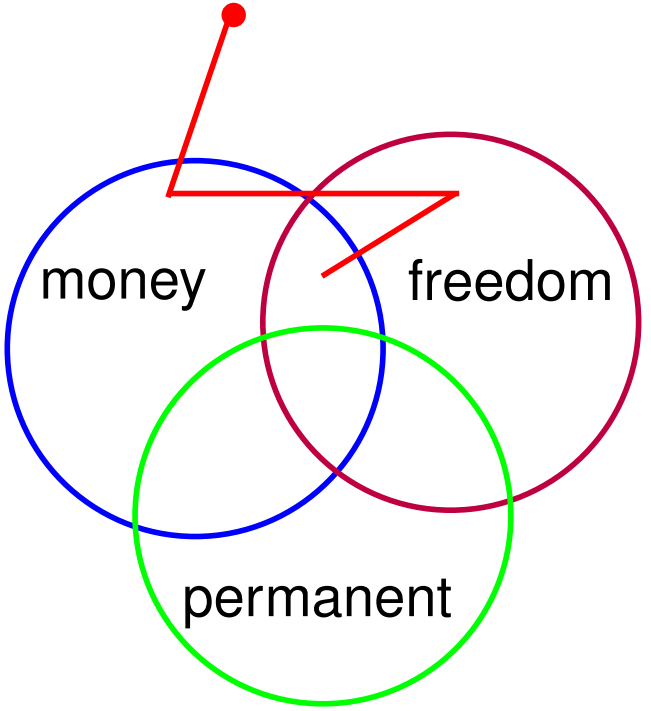
Silicon Valley/US

2010: PostDoc

INRIA Saclay/FR

2012: Research group leader

Max Planck Institute/DE



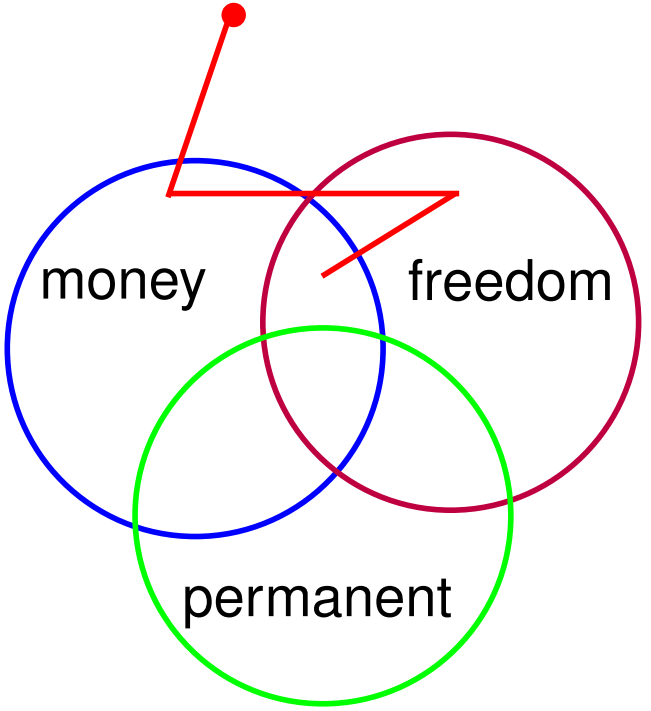
Fabian M. Suchanek

2009: PostDoc at Microsoft Research
Silicon Valley/US

2010: PostDoc
INRIA Saclay/FR

2012: Research group leader
Max Planck Institute/DE

2013: Maître de Conférences
Télécom ParisTech/FR



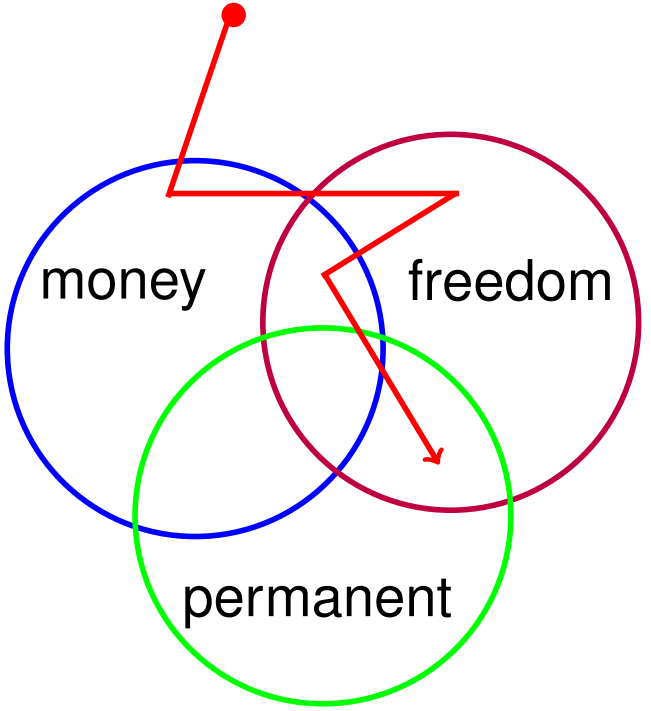
Fabian M. Suchanek

2009: PostDoc at Microsoft Research
Silicon Valley/US

2010: PostDoc
INRIA Saclay/FR

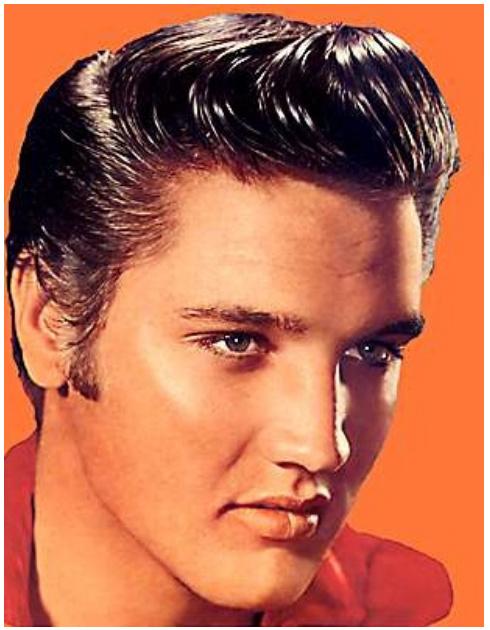
2012: Research group leader
Max Planck Institute/DE

2013: Maître de Conférences
Télécom ParisTech/FR



Fabian M. Suchanek

I am an Elvis fan!



Elvis, when I need you, I can hear you!

Will there ever be someone like him?

Searching with Google

Another singer called Elvis, young

Searching with Google

Google

Another singer called Elvis, young

Carol Connors talks about her first boyfriend

Carol Connors talks about her first boyfriend

- a young singer called Elvis Presley.

Searching with Google

Another singer called Elvis, young

Younger Elvis, singer

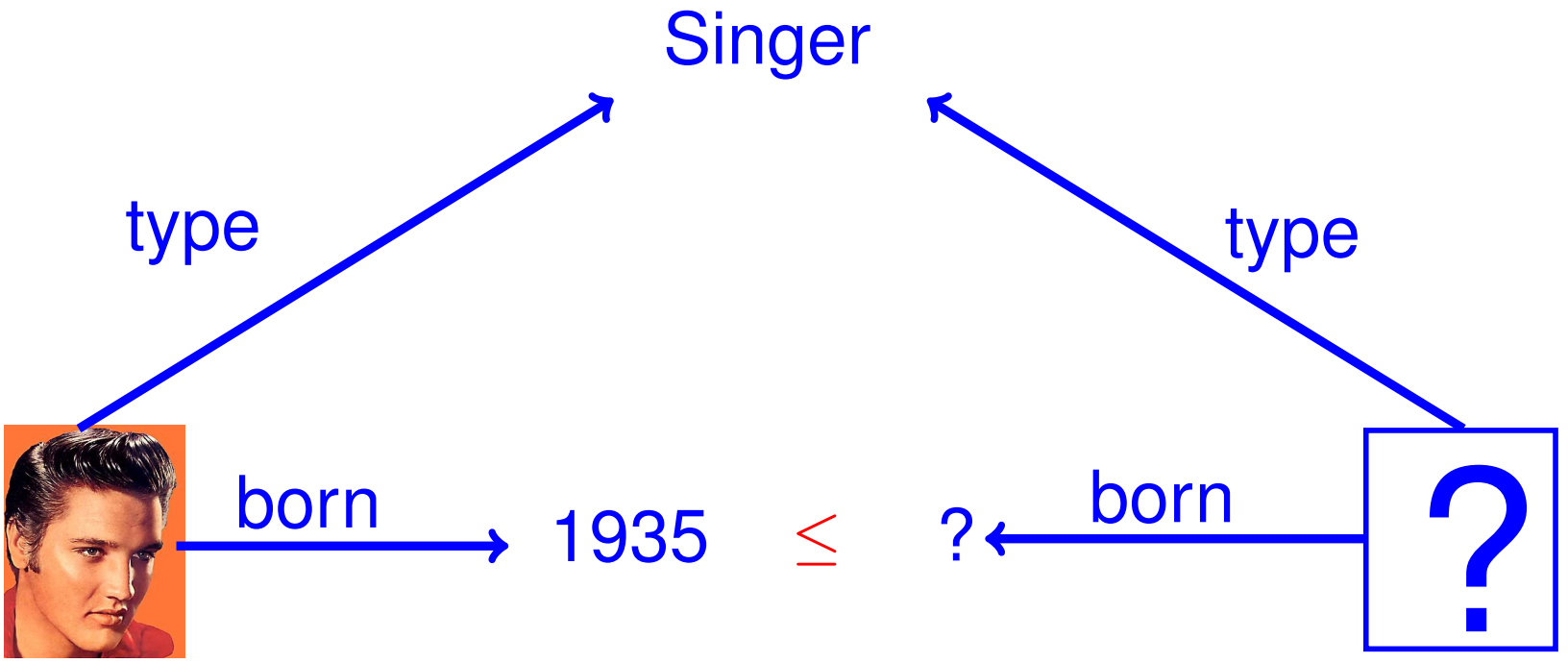
Rephrasing does not help

Another singer called Elvis, young

Younger Elvis, singer

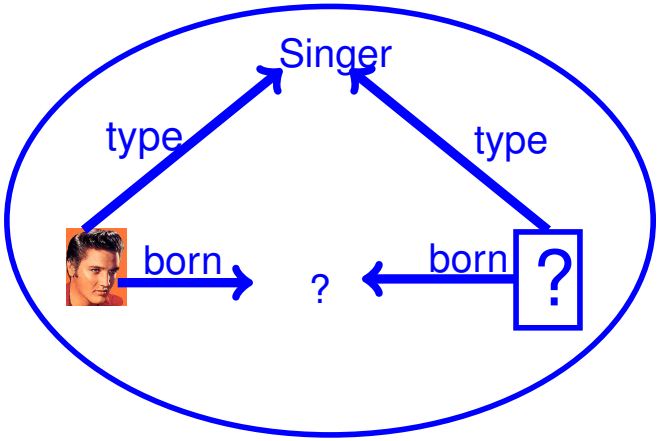
Google, you don’t understand! I want

We need structured knowledge



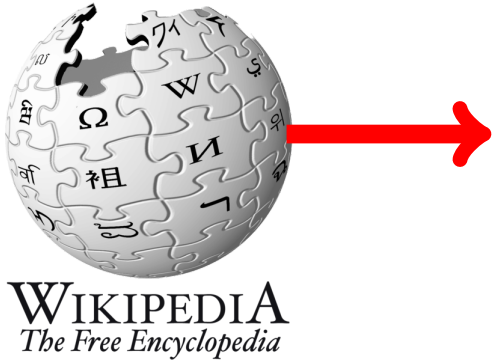
To answer the question, the computer would need structured knowledge: an “ontology”

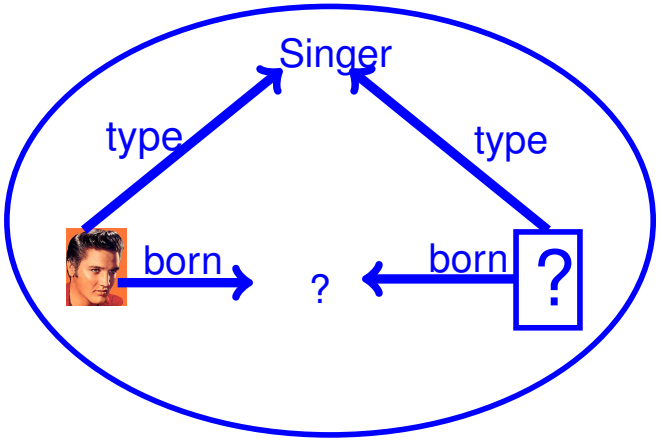
I work on Ontologies



I work on Ontologies

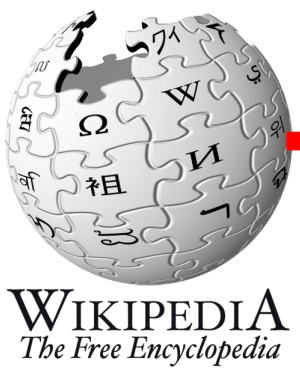
Constructing ontologies

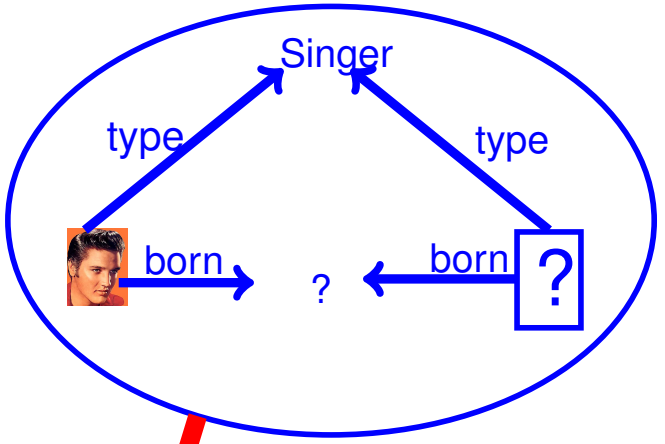




I work on Ontologies

Constructing ontologies



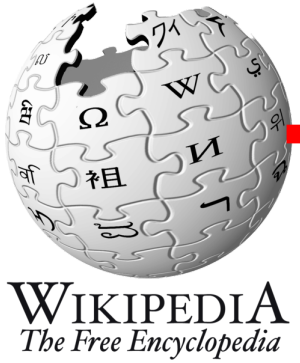


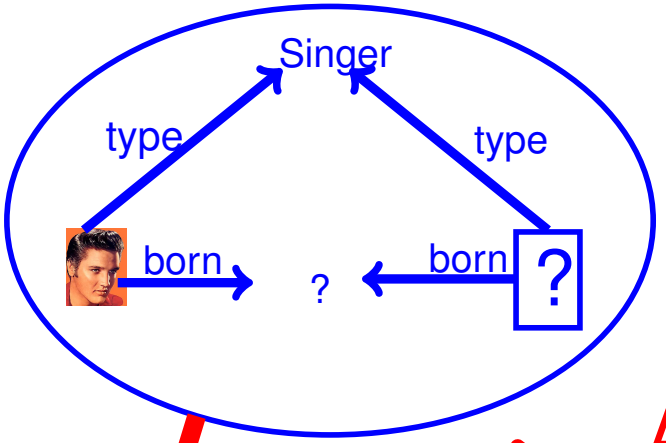
Mining ontologies

$$A \wedge B \Rightarrow C$$

I work on Ontologies

Constructing ontologies

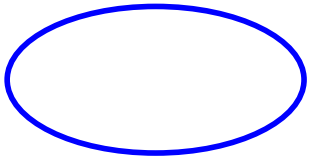




Mining ontologies

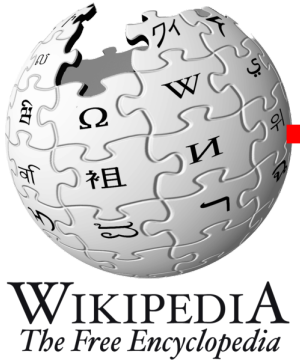
$$A \wedge B \Rightarrow C$$

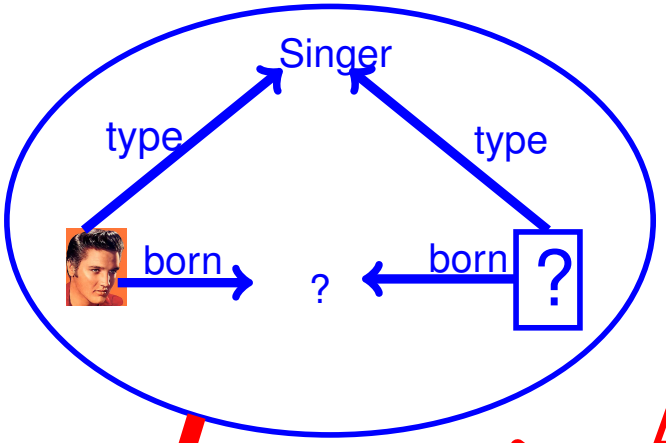
Aligning ontologies



I work on Ontologies

Constructing ontologies



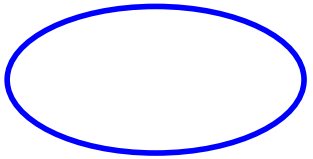


Protecting ontologies

Mining ontologies

$$A \wedge B \Rightarrow C$$

Aligning ontologies



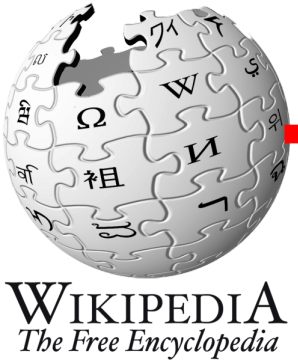
I work on Ontologies

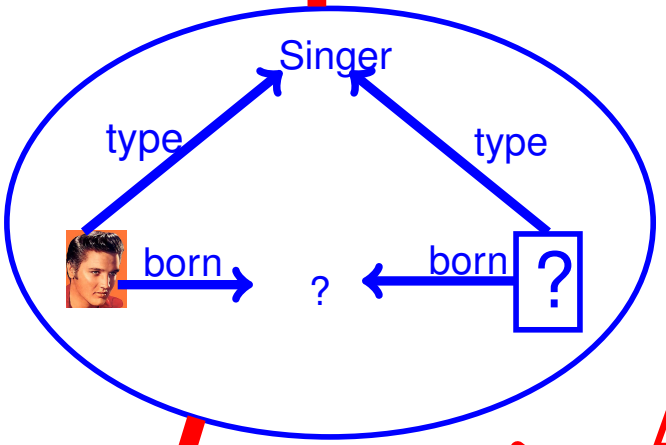
Applying ontologies

Le Monde

Constructing ontologies

Protecting ontologies





Mining ontologies

Aligning ontologies

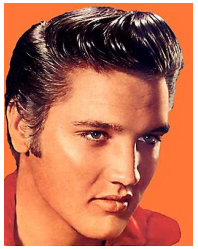
$$A \wedge B \Rightarrow C$$

Extracting from Wikipedia

Elvis Presley

WIKIPEDIA
The Free Encyclopedia

Elvis Presley was one of the best blah blah blub blah don't read this, listen to the speaker! blah blah blah blubl blah you are still reading this! blah blah blah blah blabbel blah



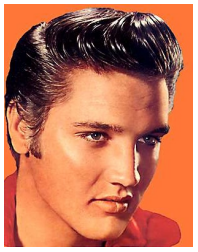
Born: 1935
In: Tupelo
...

Categories:
Rock&Roll, American Singers, Academy Award winners...

Extracting from Wikipedia

Elvis Presley

Elvis Presley was one of the best blah blah blub blah don't read this, listen to the speaker! blah blah blah blubl blah you are still reading this! blah blah blah blah blabbel blah



Born: 1935
In: Tupelo
...

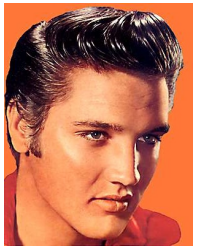
Categories:
Rock&Roll, American Singers,
Academy Award winners...

ElvisPresley

Extracting from Wikipedia

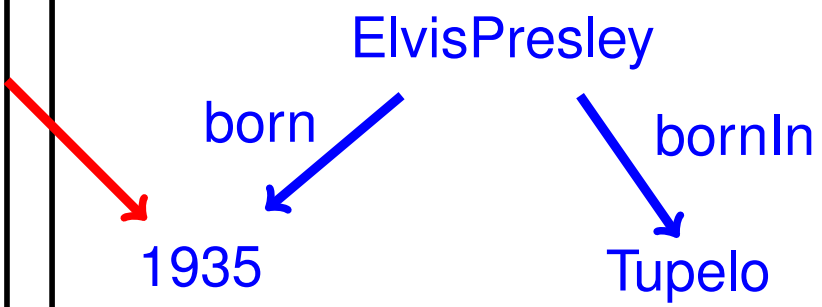
Elvis Presley

Elvis Presley was one of the best blah blah blub blah don't read this, listen to the speaker! blah blah blah blubl blah you are still reading this! blah blah blah blah blabbel blah



Born: 1935
In: Tupelo
...

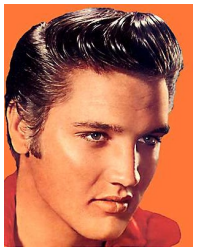
Categories:
Rock&Roll, American Singers,
Academy Award winners...



Extracting from Wikipedia

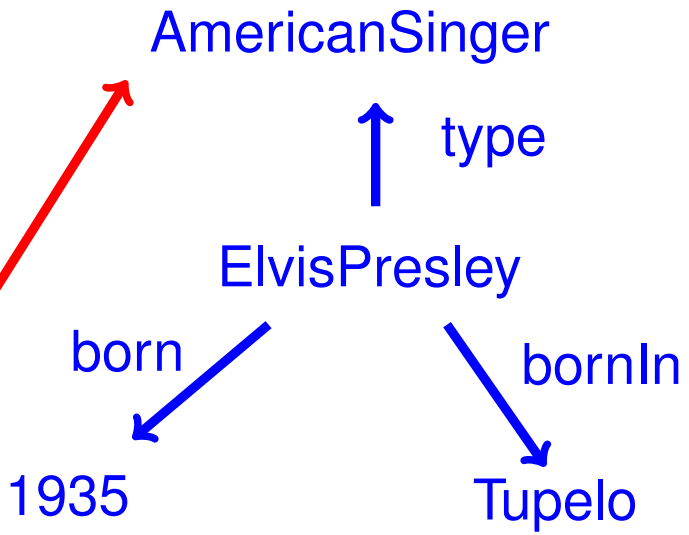
Elvis Presley

Elvis Presley was one of the best blah blah blub blah don't read this, listen to the speaker! blah blah blah blubl blah you are still reading this! blah blah blah blah blabbel blah



Born: 1935
In: Tupelo
...

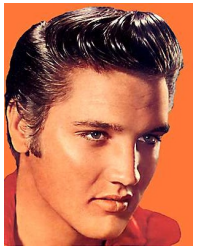
Categories:
Rock&Roll, American Singers,
Academy Award winners...



Extracting from Wikipedia

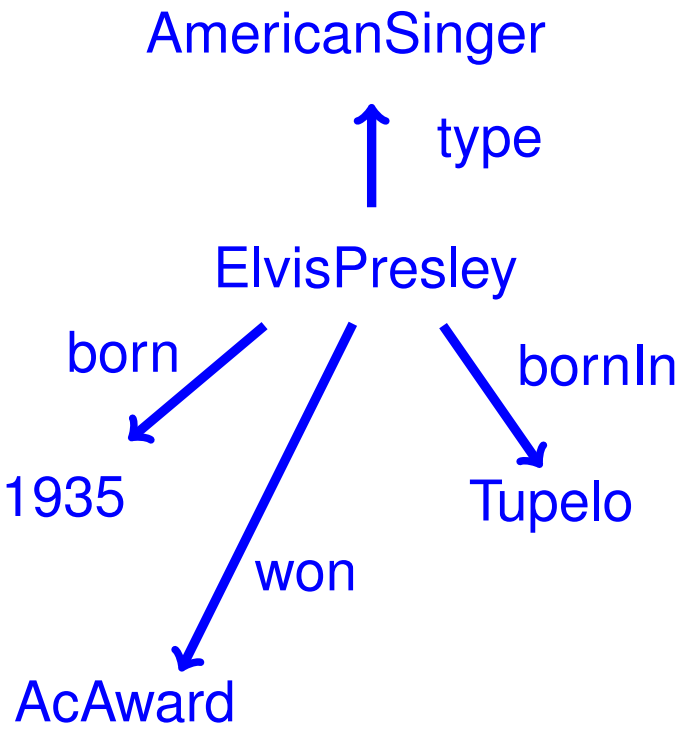
Elvis Presley

Elvis Presley was one of the best blah blah blub blah don't read this, listen to the speaker! blah blah blah blubl blah you are still reading this! blah blah blah blah blabbel blah



Born: 1935
In: Tupelo
...

Categories:
Rock&Roll, American Singers,
Academy Award winners...

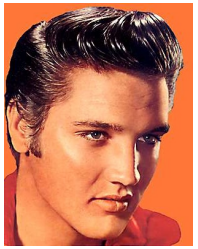


Extracting from Wikipedia

Elvis Presley

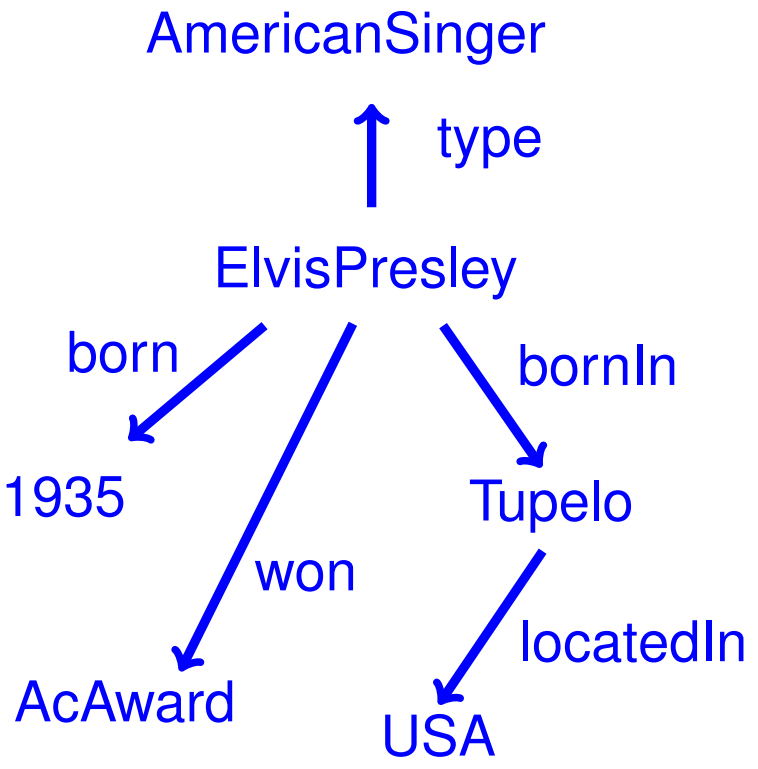
WIKIPEDIA
The Free Encyclopedia

Elvis Presley was one of the best blah blah blub blah don't read this, listen to the speaker! blah blah blah blubl blah you are still reading this! blah blah blah blah blabbel blah



Born: 1935
In: Tupelo
...

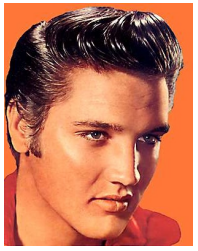
Categories:
Rock&Roll, American Singers, Academy Award winners...



Extracting from Wikipedia

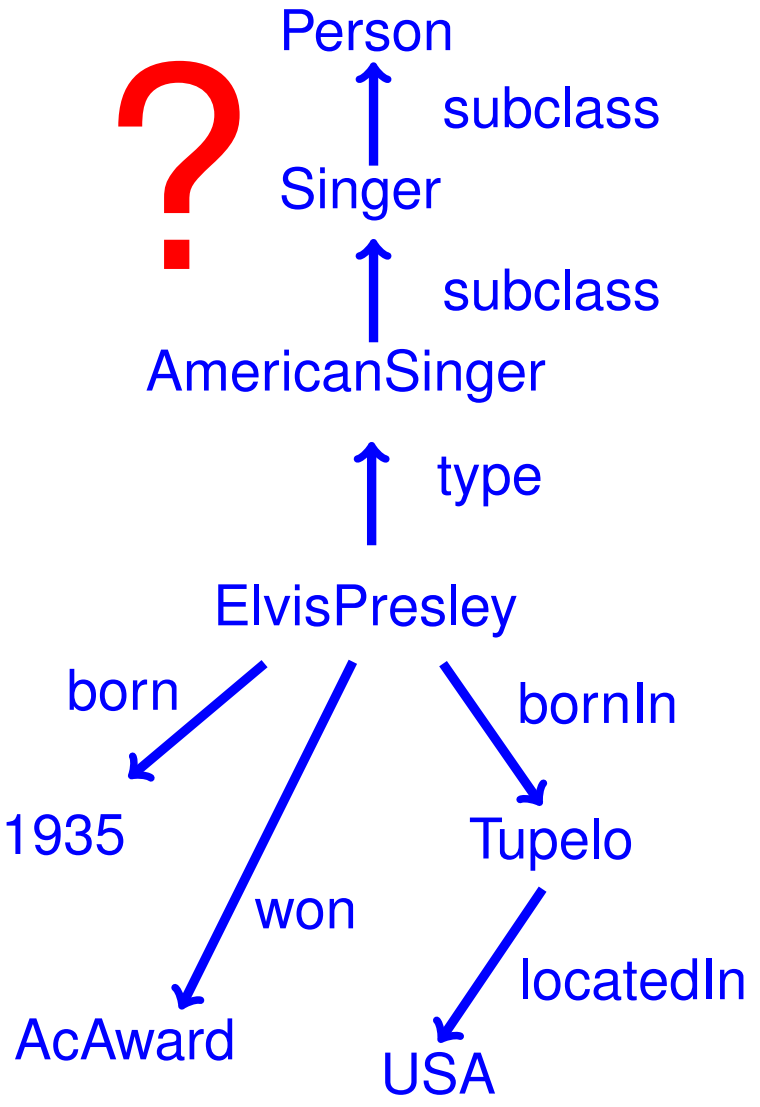
Elvis Presley

Elvis Presley was one of the best blah blah blub blah don't read this, listen to the speaker! blah blah blah blubl blah you are still reading this! blah blah blah blah blabbel blah

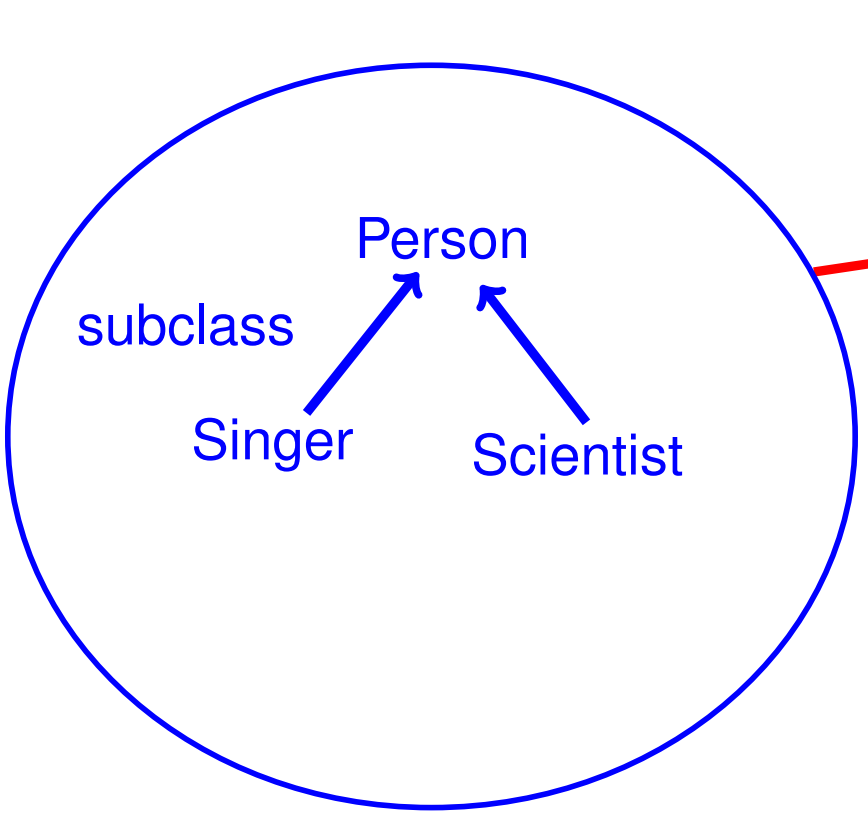


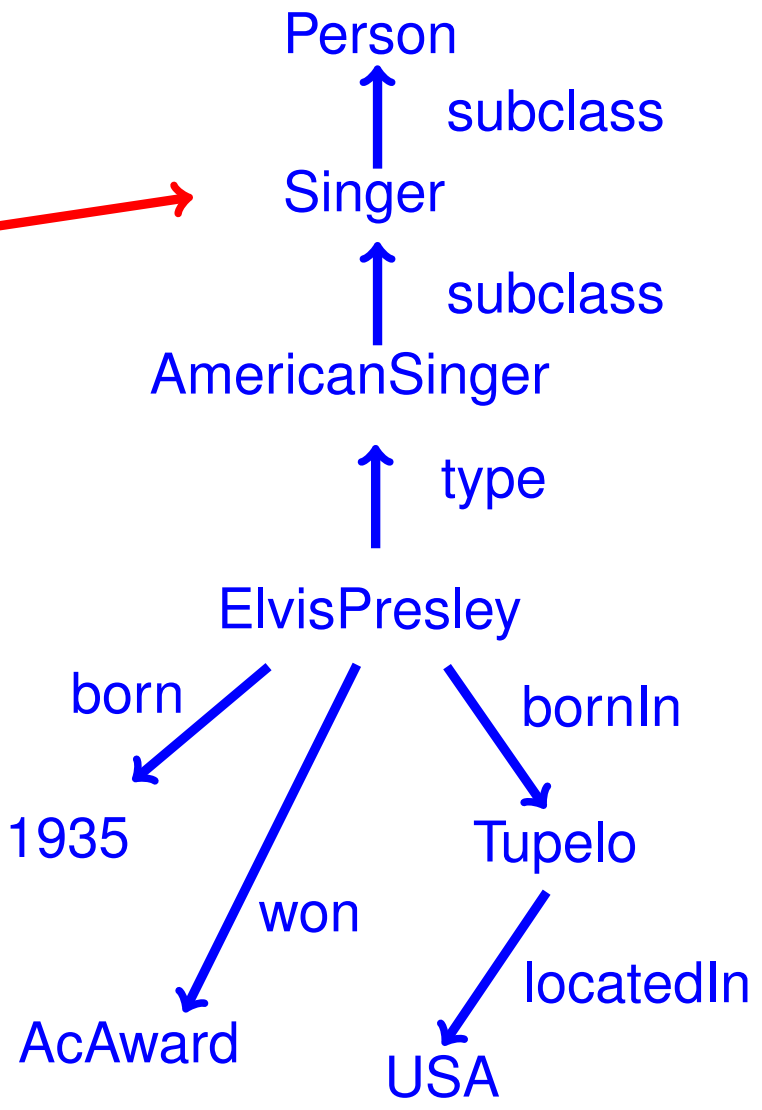
Born: 1935
In: Tupelo
...

Categories:
Rock&Roll, American Singers, Academy Award winners...



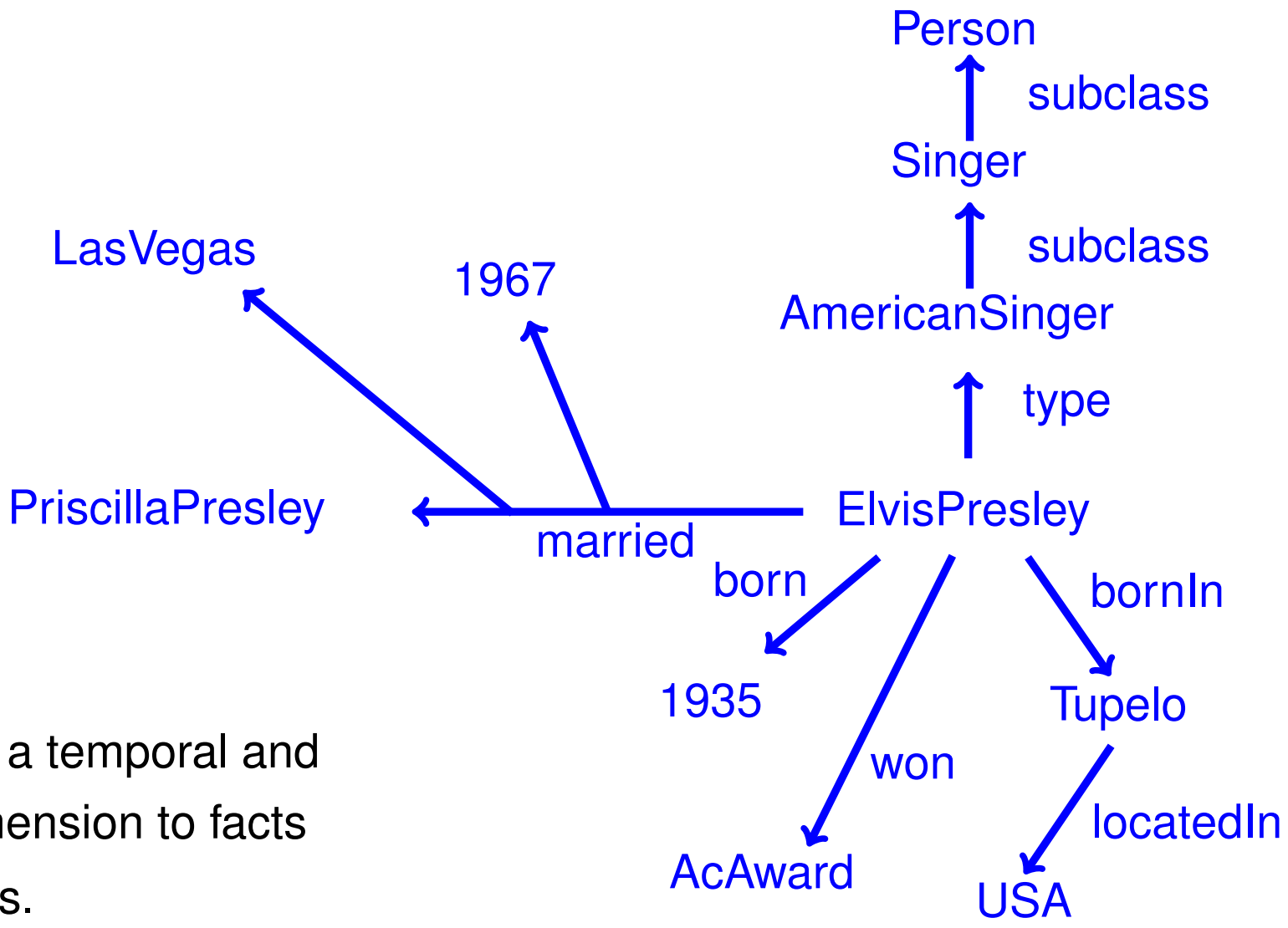
Adding WordNet





WordNet
a lexicon of the English language developed at Princeton

Adding Time and Space

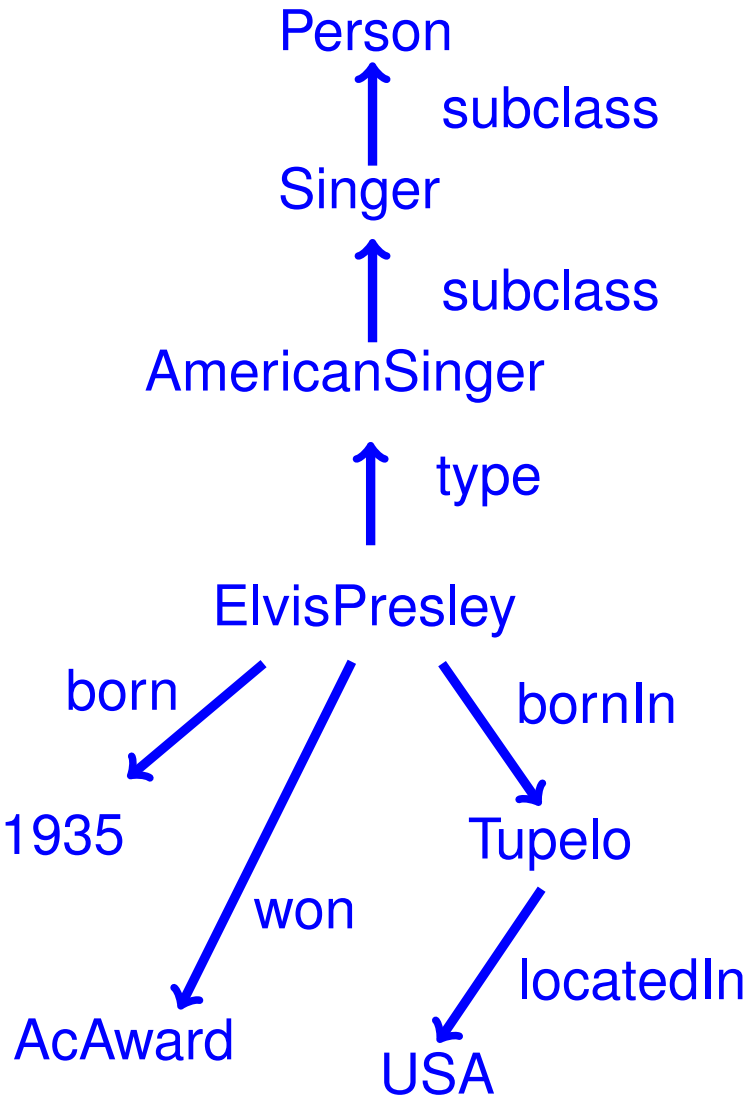


We added a temporal and spatial dimension to facts and entities.

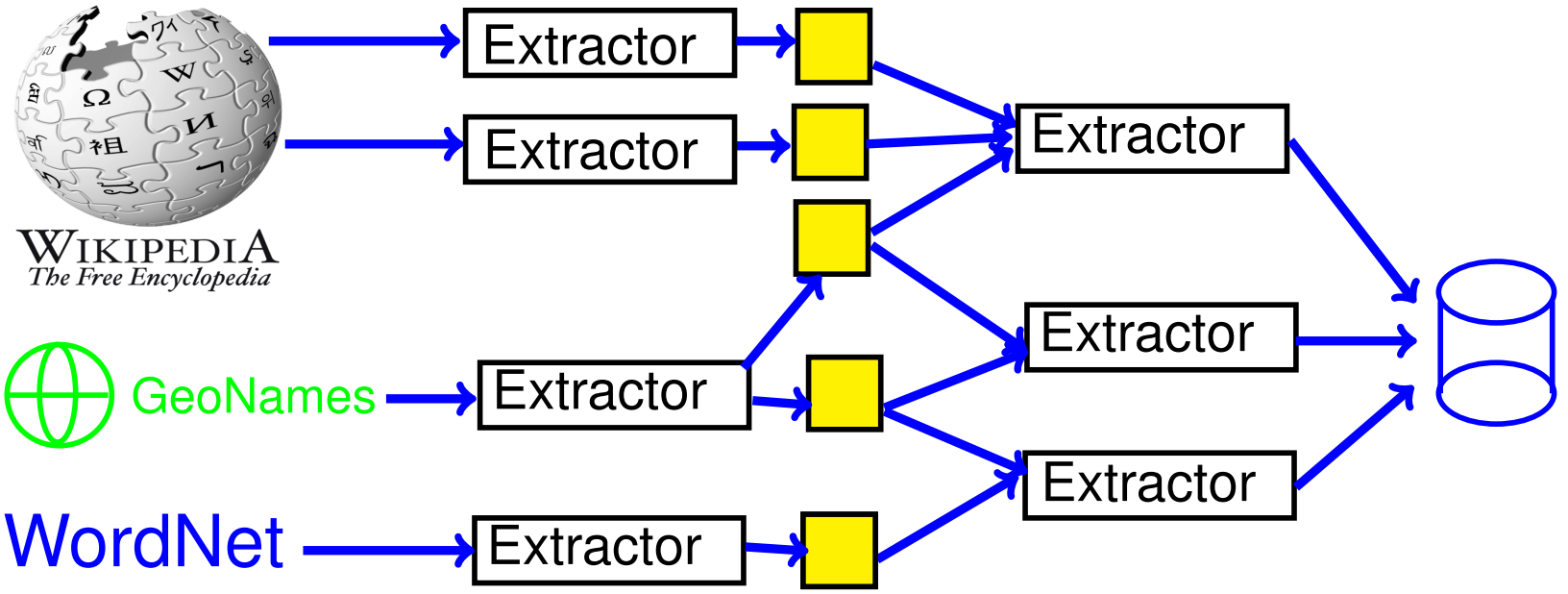
YAGO: a large knowledge base

YAGO is a knowledge base that

- combines WordNet classes and Wikipedia instances
- has time and space
- has a manually evaluated accuracy of 95%



YAGO: a large knowledge base



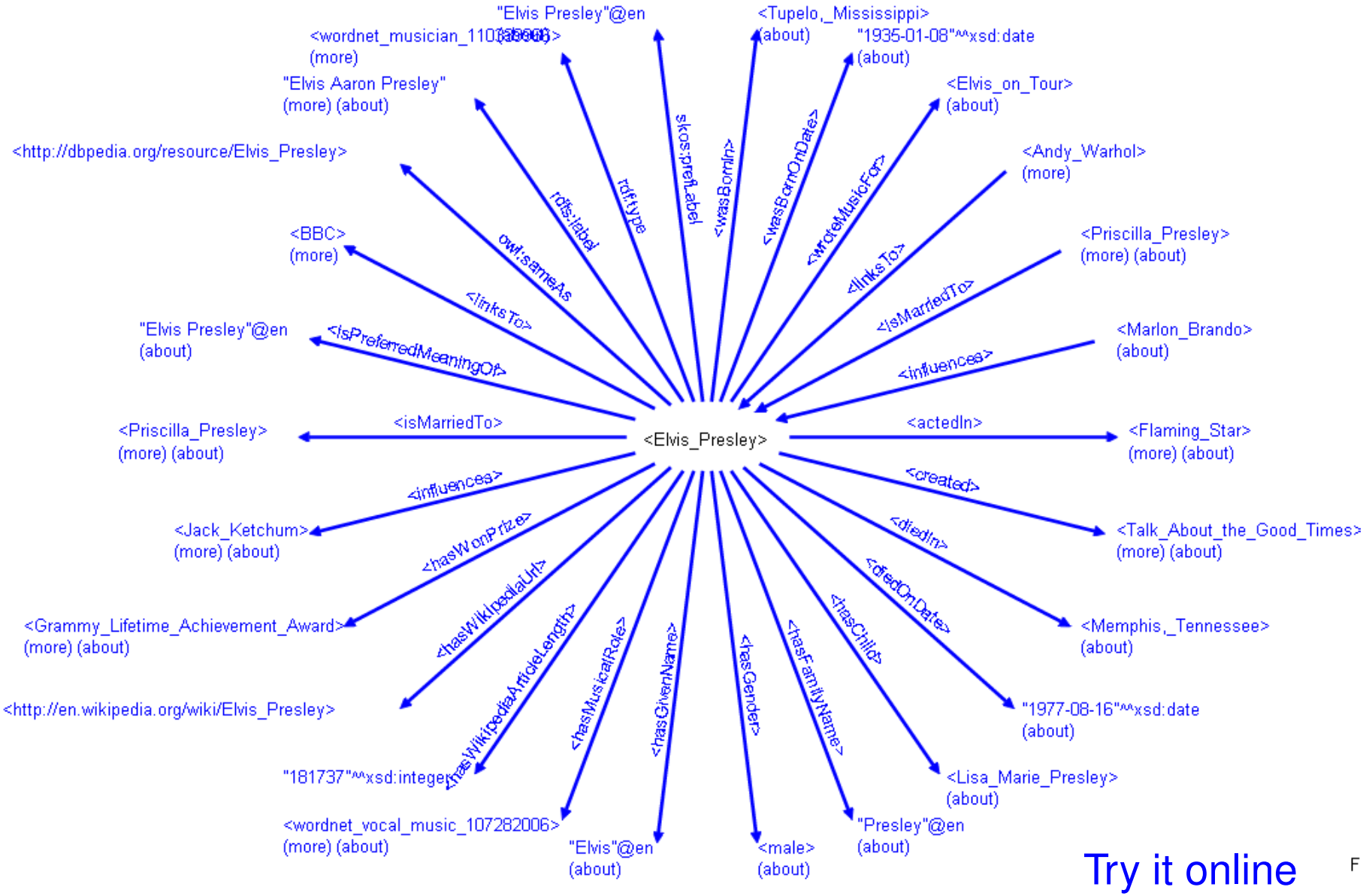
Try it online

Try locally

2007 —— 2013 ⟶

[WWW 2007, JWS 2008, WWW 2011 demo, AIJ 2013, WWW 2013 demo]

Example: YAGO about Elvis



Try it online

YAGO: a large knowledge base

100m facts

10m entities

95% accuracy

100 Web site visitors/day

1000 citations

linked to Freebase and DBpedia

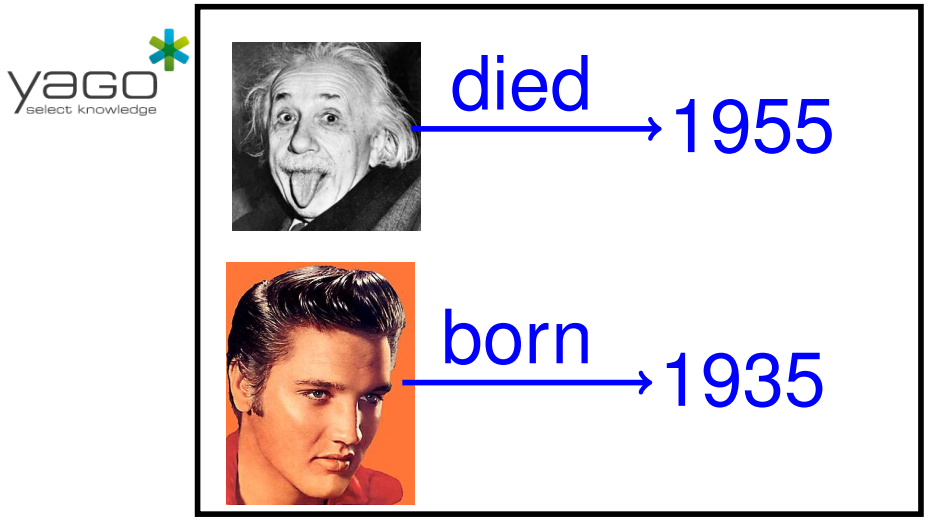
http://yago-knowledge.org





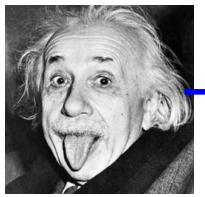

SOFIE extracts by Reasoning

Elvis died in 528

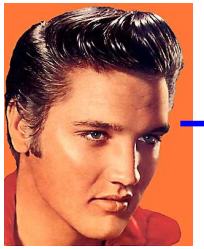


SOFIE extracts by Reasoning

Elvis died in 528



died → 1955



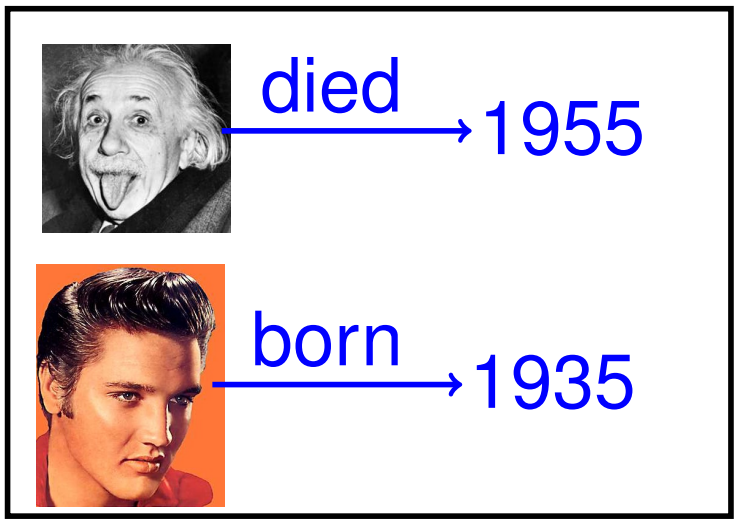
born → 1935

occurs(“Elvis”,“died in”,528)

SOFIE extracts by Reasoning

Elvis died in 528

Einstein died in 1955



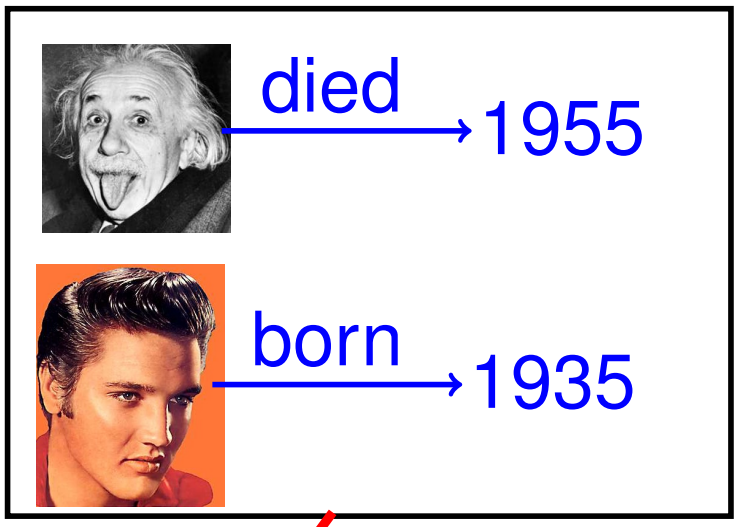
occurs(“Elvis”,“died in”,528)

occurs(“Einstein”,“died in”,1955)

SOFIE extracts by Reasoning

Elvis died in 528

Einstein died in 1955



occurs("Elvis","died in",528)

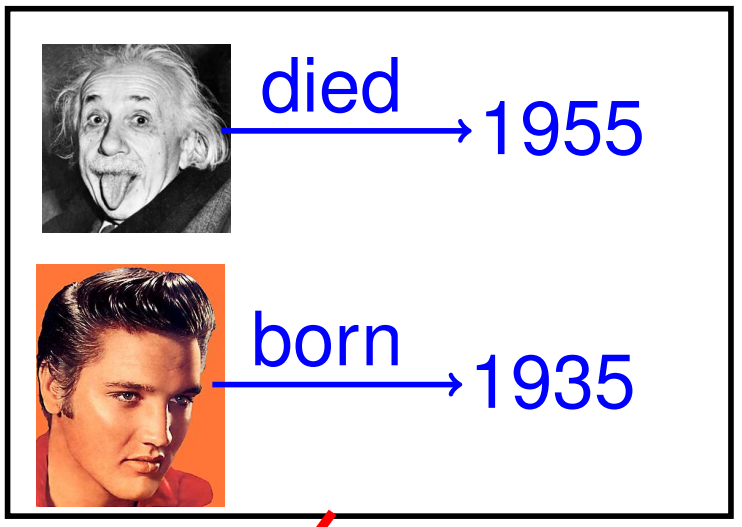
occurs("Einstein","died in",1955)

died(Einstein,1955), born(Elvis, 1935)

SOFIE extracts by Reasoning

Elvis died in 528

Einstein died in 1955



occurs("Elvis","died in",528)

occurs("Einstein","died in",1955)

died(Einstein,1955), born(Elvis, 1935)

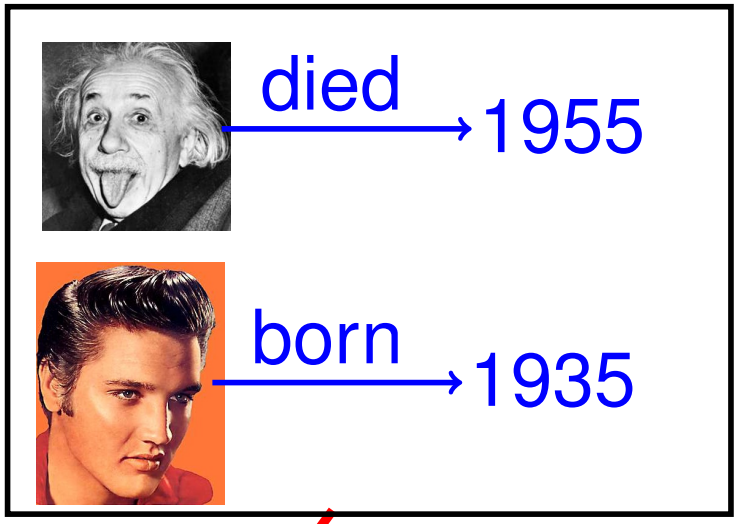
occurs(X',P,Y) & means(X',X) & R(X,Y) => pattern(P,R)

occurs(X',P,Y) & means(X',X) & pattern(P,R) => R(X,Y)

SOFIE extracts by Reasoning

Elvis died in 528

Einstein died in 1955



occurs("Elvis","died in",528)

occurs("Einstein","died in",1955)

died(Einstein,1955), born(Elvis, 1935)

occurs(X',P,Y) & means(X',X) & R(X,Y) => pattern(P,R)

occurs(X',P,Y) & means(X',X) & pattern(P,R) => R(X,Y)

born(X,Y) & died(X,Z) => Z>Y

...

SOFIE extracts by Reasoning

Elvis died in 528

Solving a
Weighted MAX SAT
problem at scale

occurs(“Elvis”,“died in”,528)

occurs(“Einstein”,“died in”,1955)

died(Einstein,1955), born(Elvis, 1935)

occurs(X’,P,Y) & means(X’,X) & R(X,Y) => pattern(P,R)

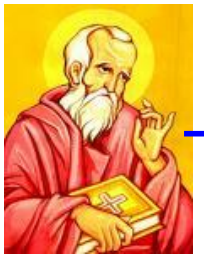
occurs(X’,P,Y) & means(X’,X) & pattern(P,R) => R(X,Y)

born(X,Y) & died(X,Z) => Z>Y

...

SOFIE extracts by Reasoning

Elvis died in 528



died → 528

[WWW 2009]

occurs(“Elvis”,“died in”,528)

occurs(“Einstein”,“died in”,1955)

died(Einstein,1955), born(Elvis, 1935)

occurs(X’,P,Y) & means(X’,X) & R(X,Y) => pattern(P,R)

occurs(X’,P,Y) & means(X’,X) & pattern(P,R) => R(X,Y)

born(X,Y) & died(X,Z) => Z>Y

...

Product extraction

yago
select knowledge

Work on Ontologies

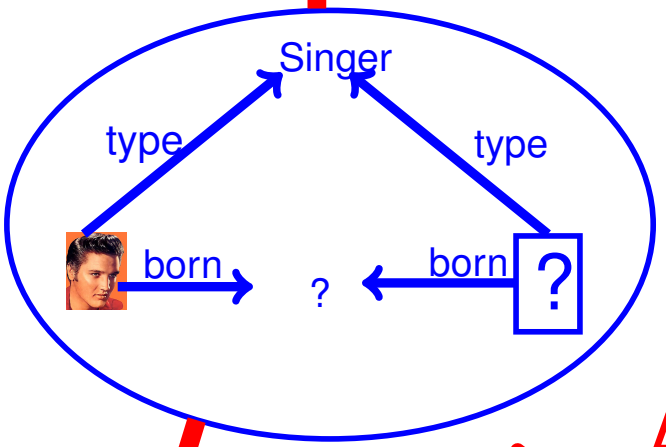
Applying ontologies

Le Monde

Constructing ontologies

✓

Protecting ontologies

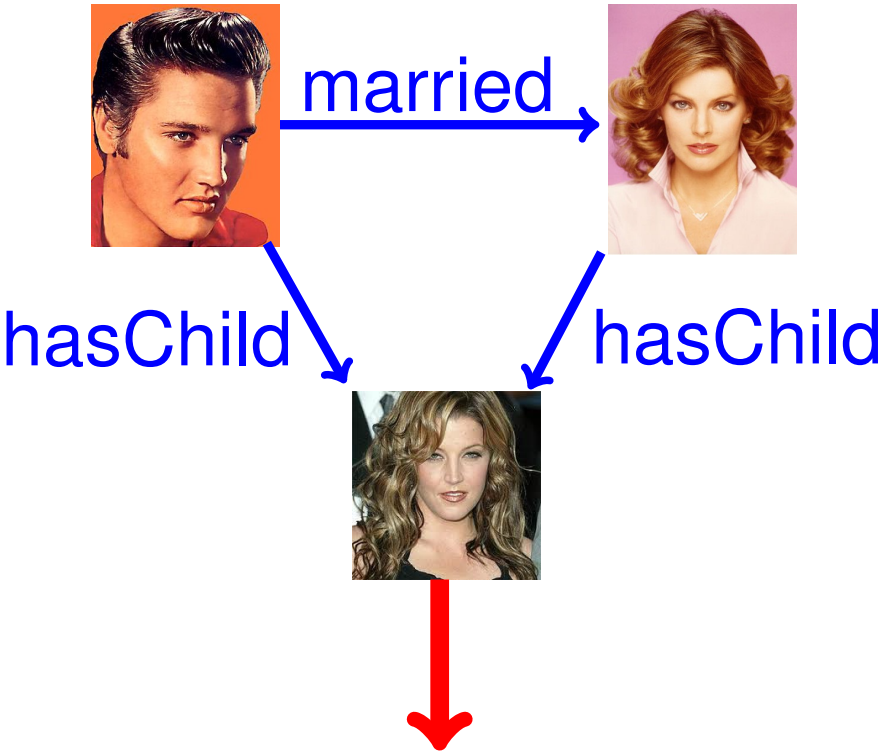


Mining ontologies

Aligning ontologies

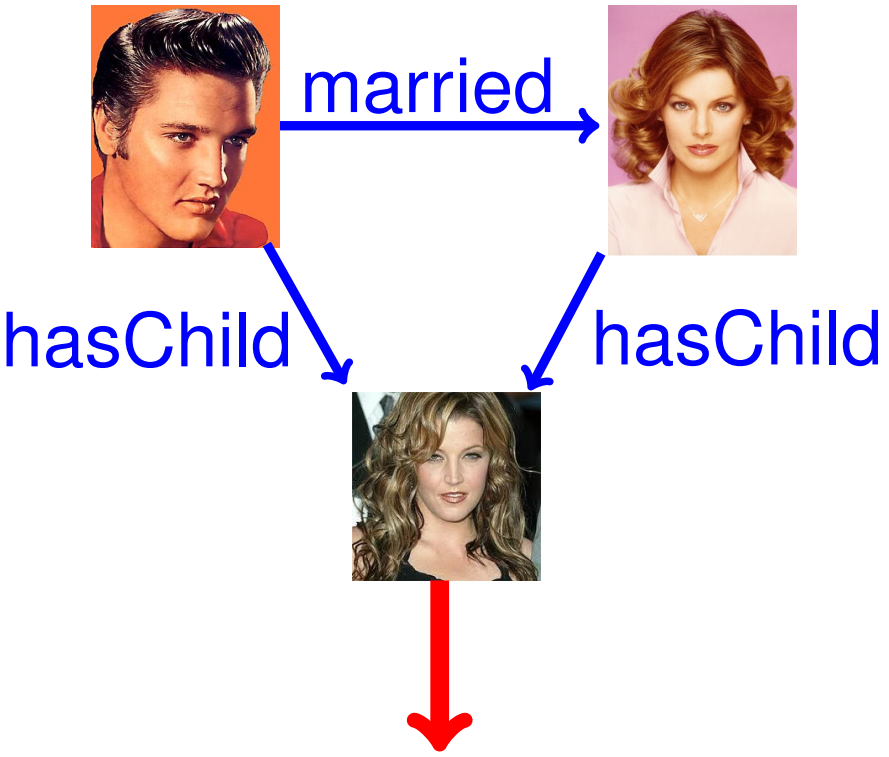
$A \wedge B \Rightarrow C$

Rule Mining finds patterns



$$married(x, y) \wedge hasChild(x, z) \Rightarrow hasChild(y, z)$$

Rule Mining finds patterns



$$married(x, y) \wedge hasChild(x, z) \Rightarrow hasChild(y, z)$$

But: Rule mining needs counter examples and RDF ontologies are positive only

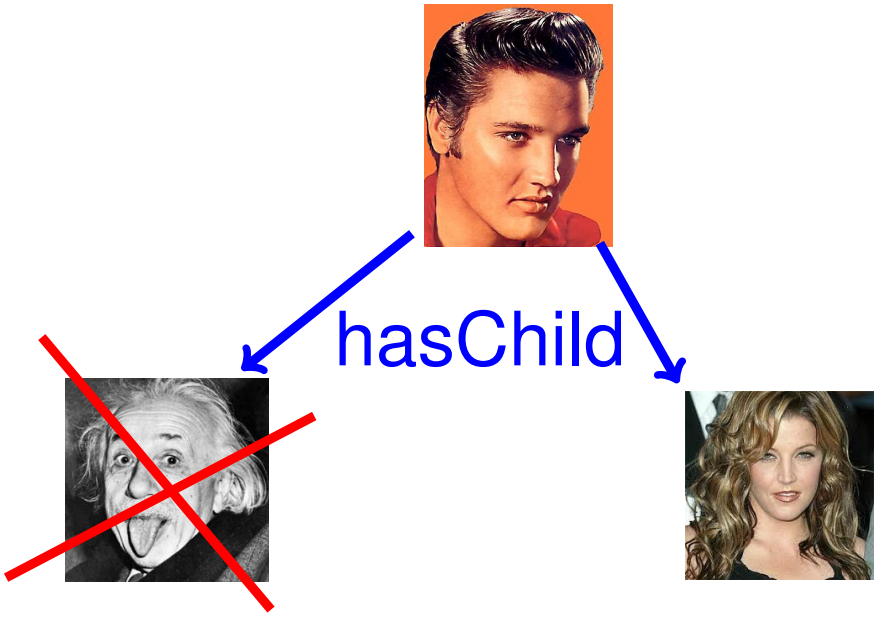
Partial Completeness Assumption

Assumption:

If we know r(x,y1),..., r(x,yn),

then all other r(x,z) are false.

Partial Completeness Assumption

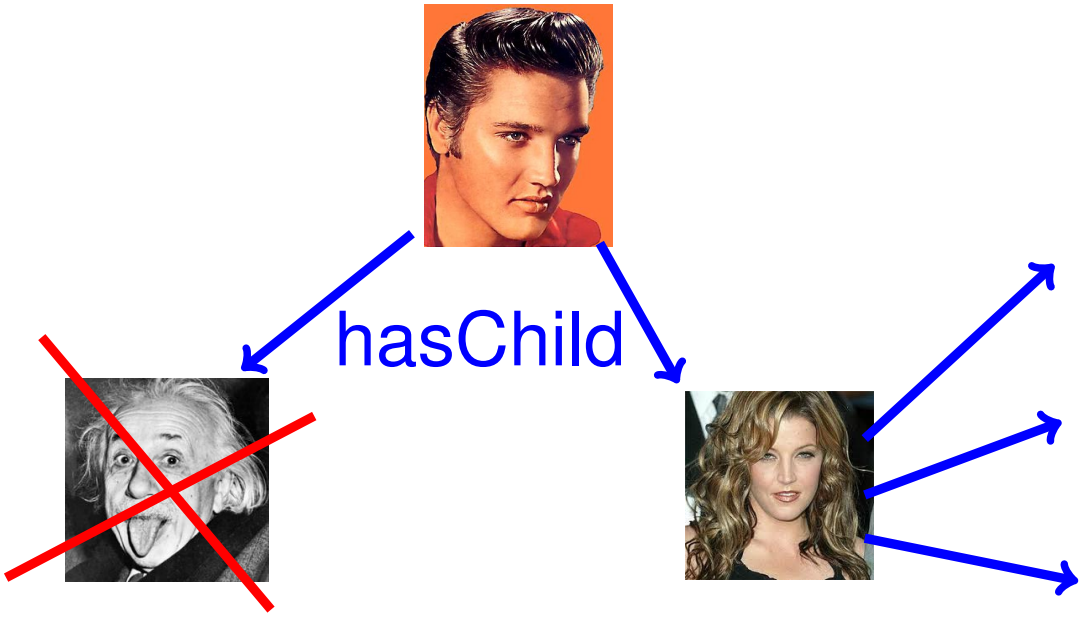


Assumption:

If we know r(x,y1),..., r(x,yn),

then all other r(x,z) are false.

Partial Completeness Assumption

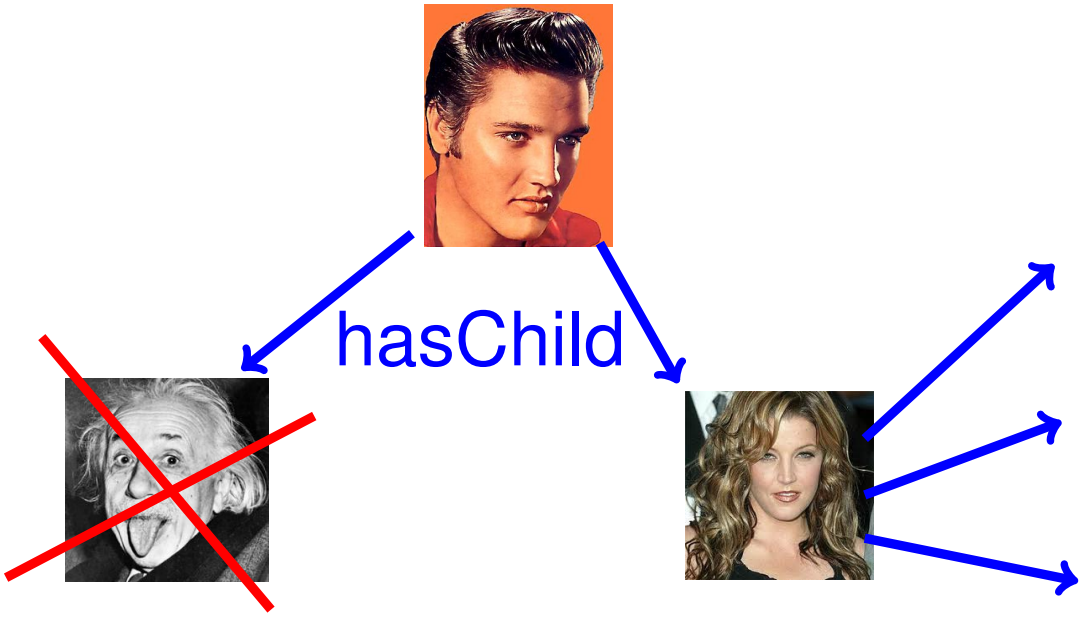


Assumption:

If we know r(x,y1),..., r(x,yn),

then all other r(x,z) are false.

Partial Completeness Assumption



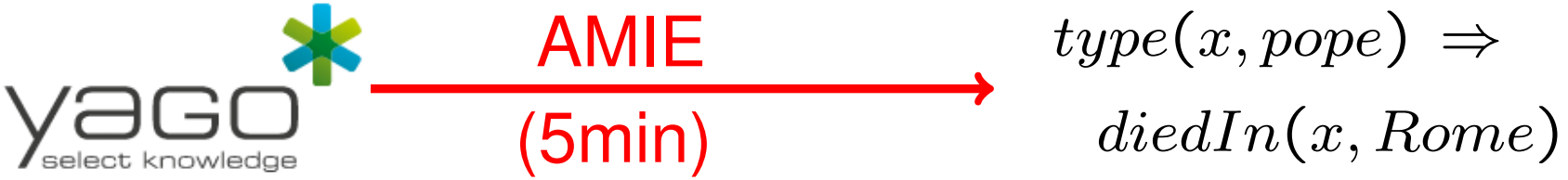
Assumption:

If we know r(x,y1),..., r(x,yn),

then all other r(x,z) are false.

+ an efficient implementation

AMIE finds rules in ontologies



AMIE finds rules in ontologies

AMIE
(5min)

$$type(x, pope) \Rightarrow$$
$$diedIn(x, Rome)$$

[WWW 2013]

Work on Ontologies

Applying ontologies

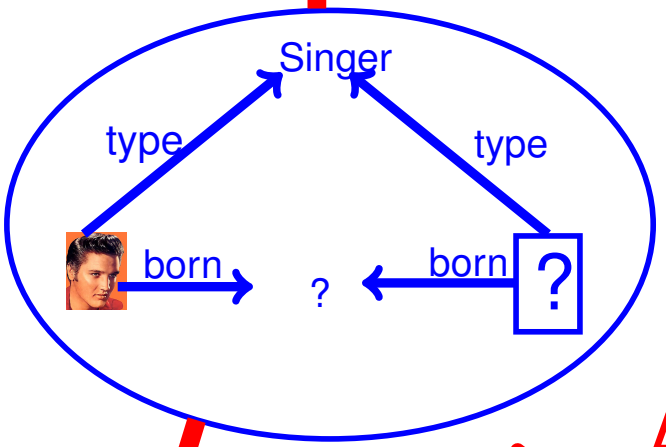
Le Monde

Constructing ontologies

✓

Protecting ontologies

WIKIPEDIA
The Free Encyclopedia



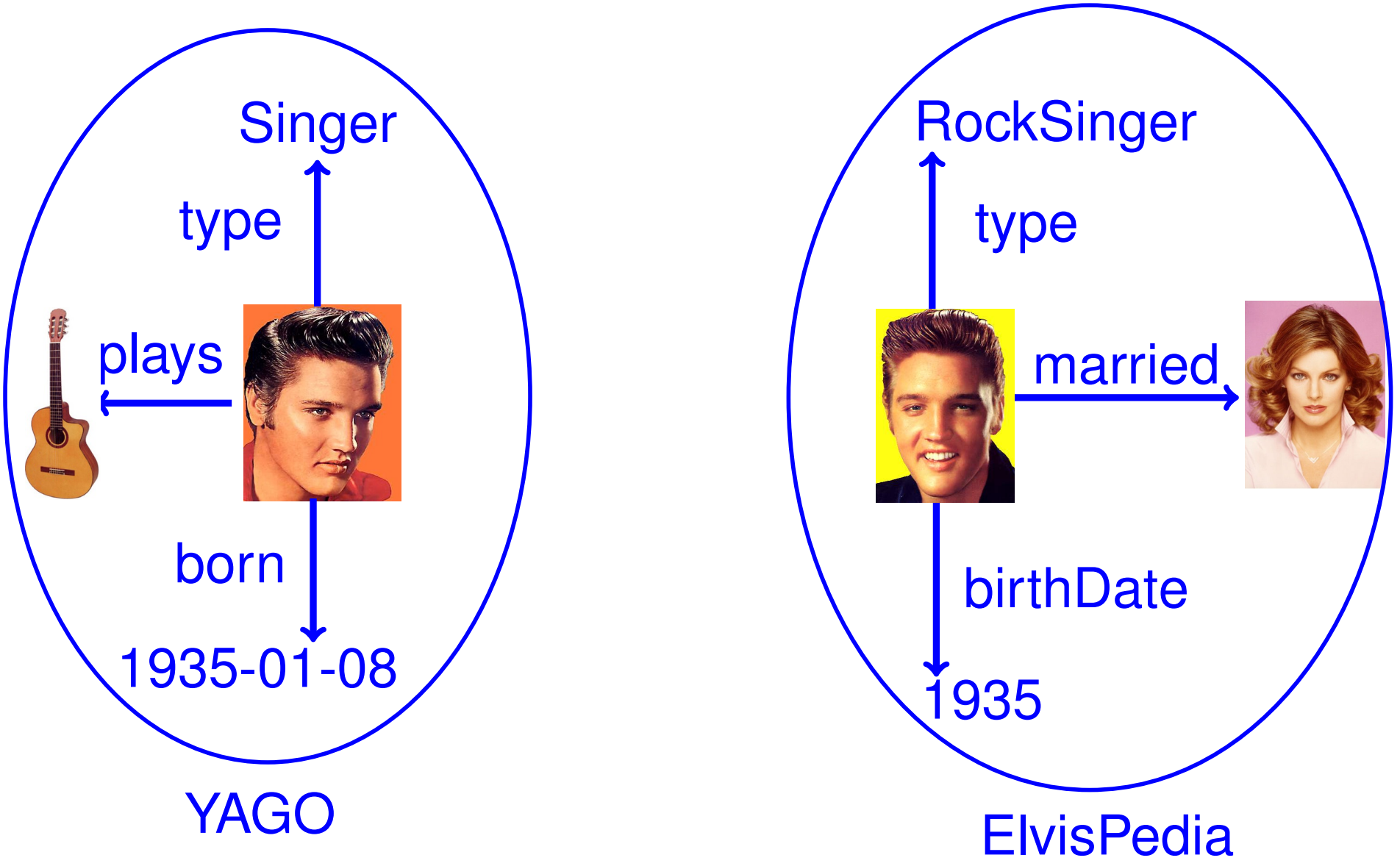
Mining ontologies

✓

Aligning ontologies

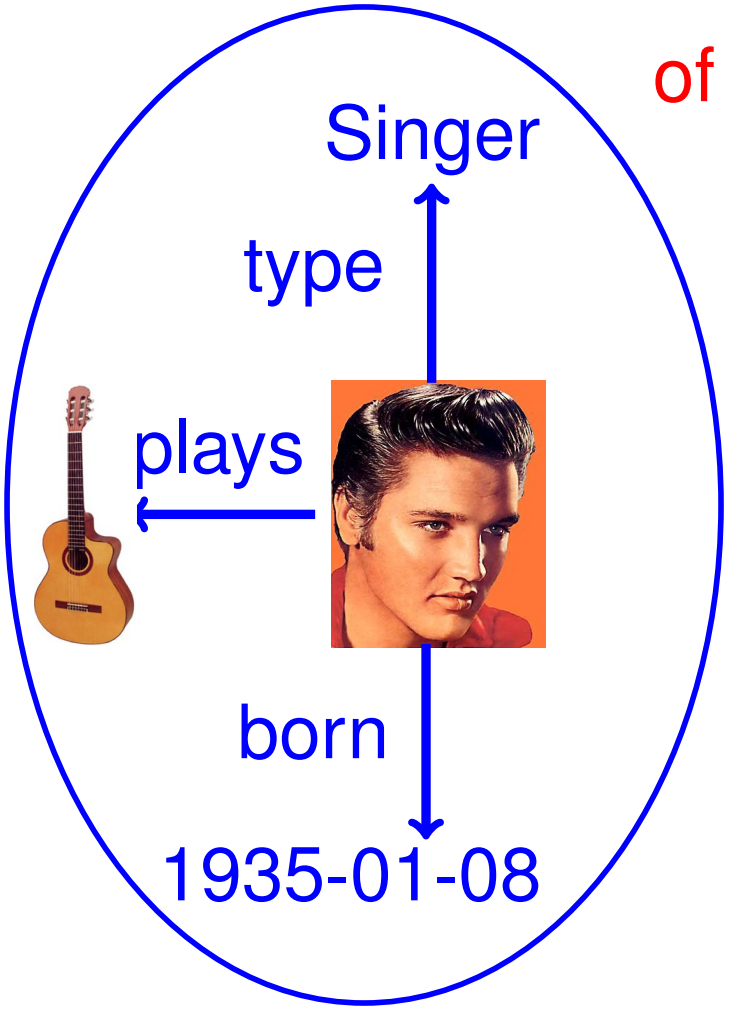
$A \wedge B \Rightarrow C$

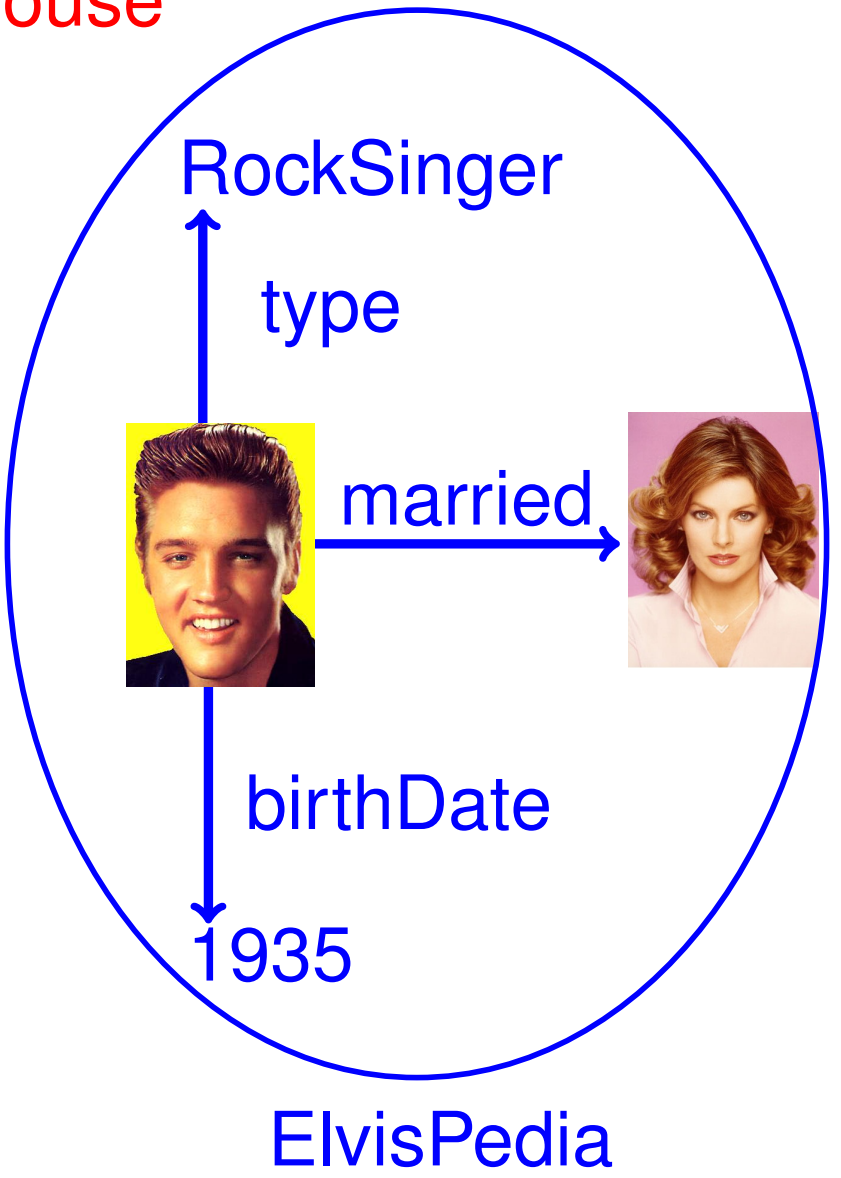
Ontologies are complementary



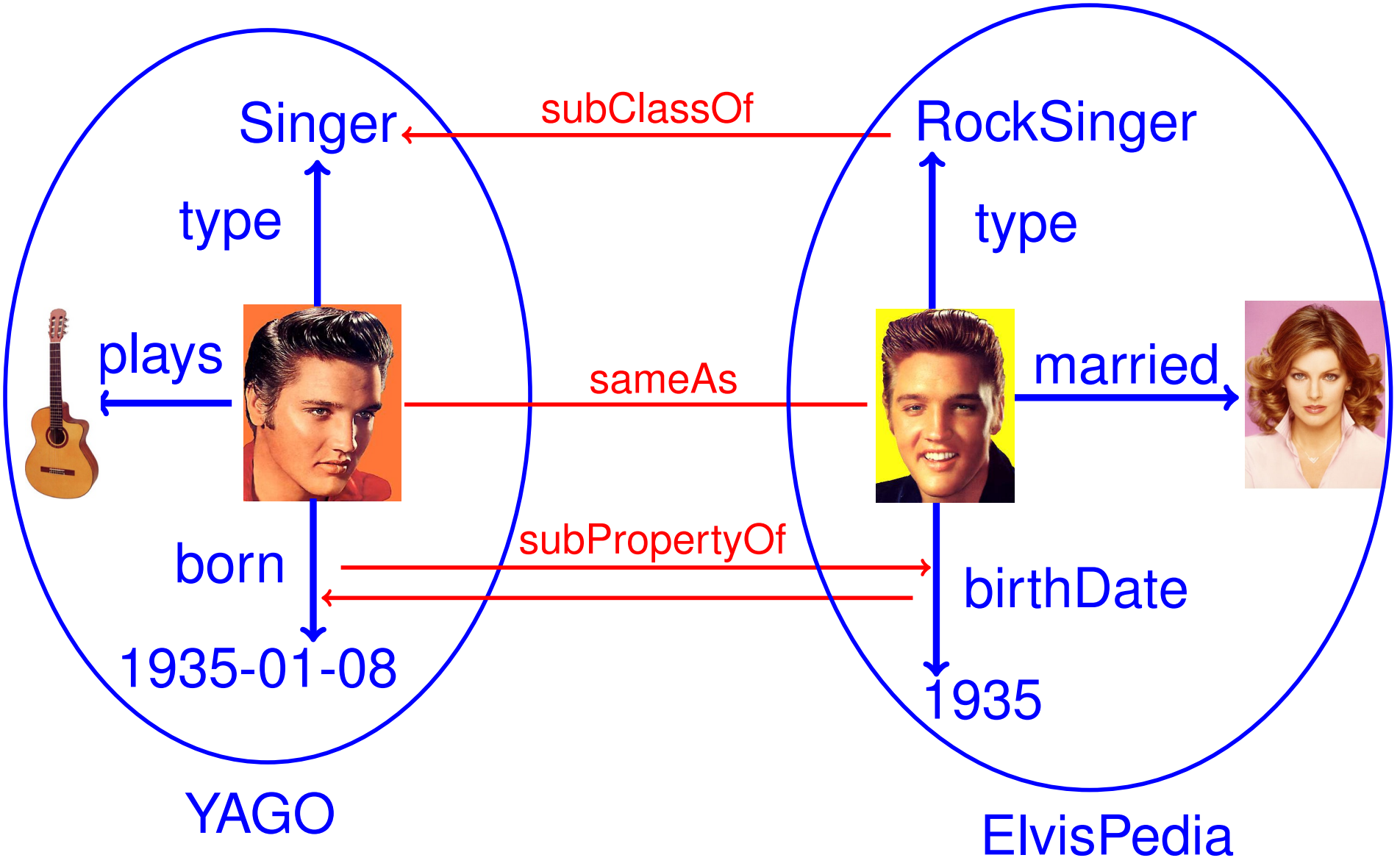
No Links => No Use

Wo is the spouse of the guitar player?

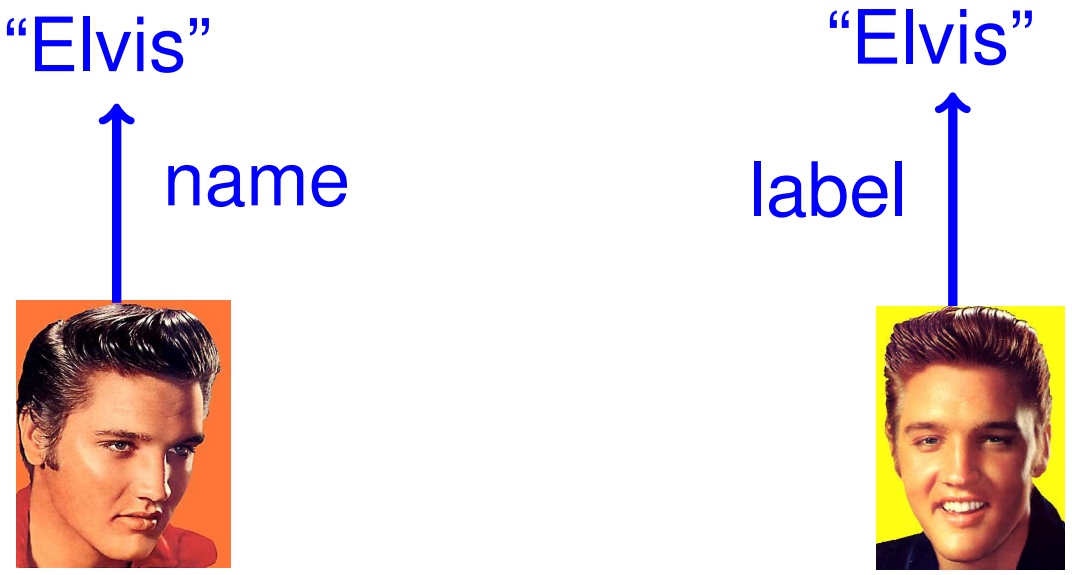




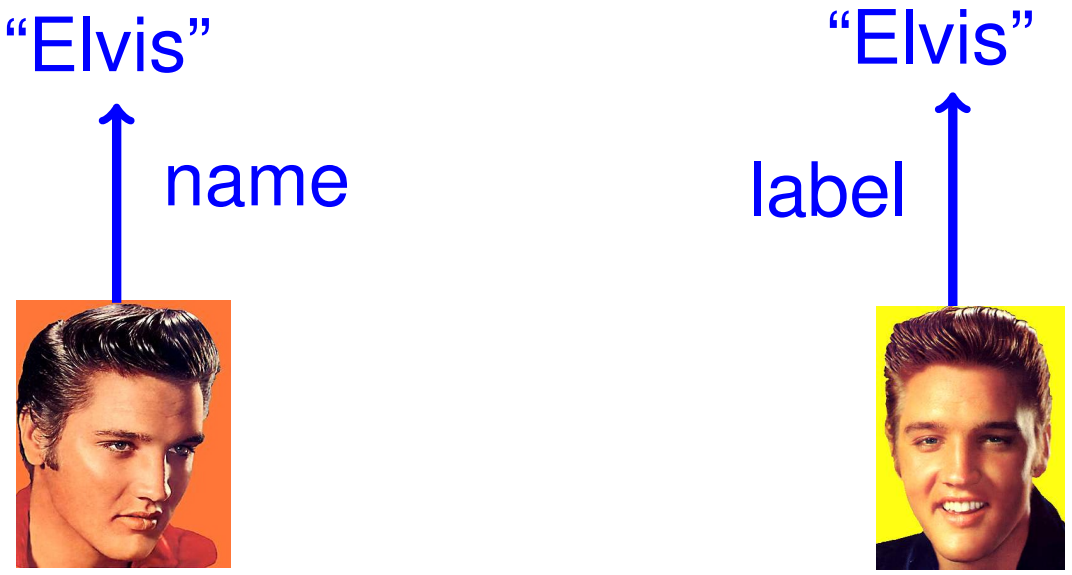
Match classes, entities, & relations



Match classes, entities, & relations

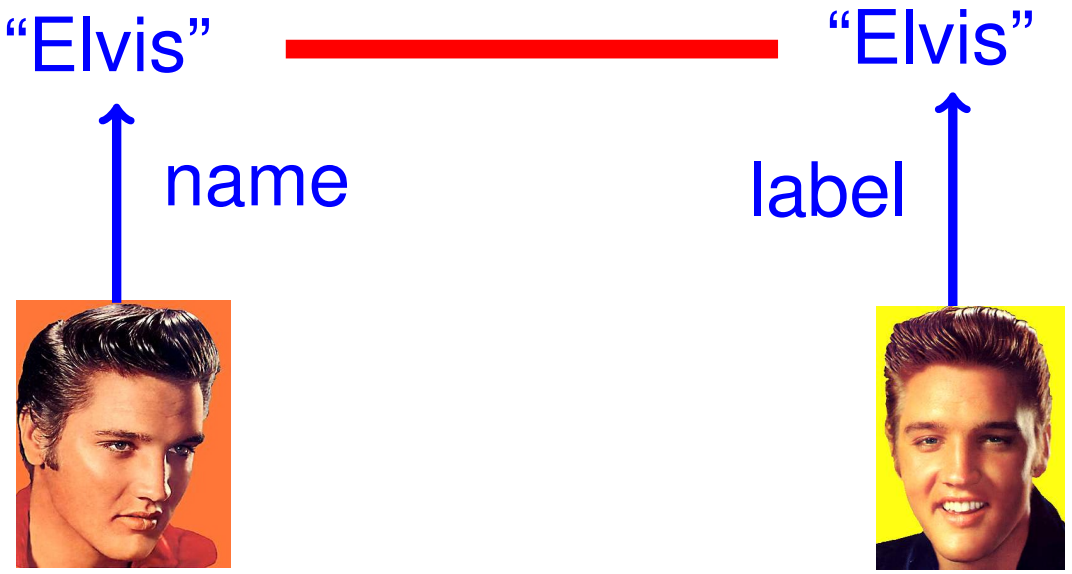


Match classes, entities, & relations



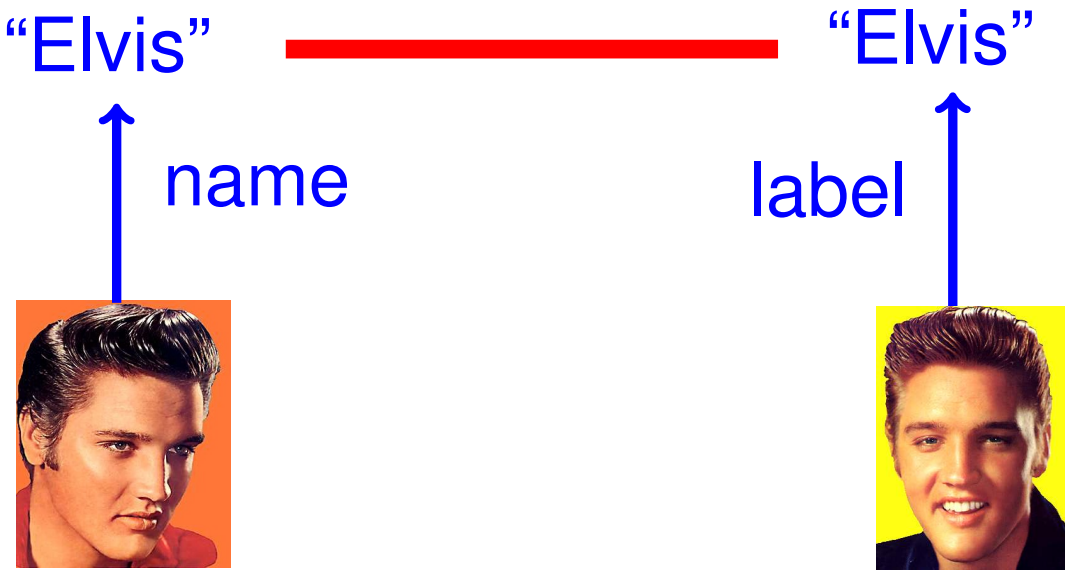
1. Match literals

Match classes, entities, & relations



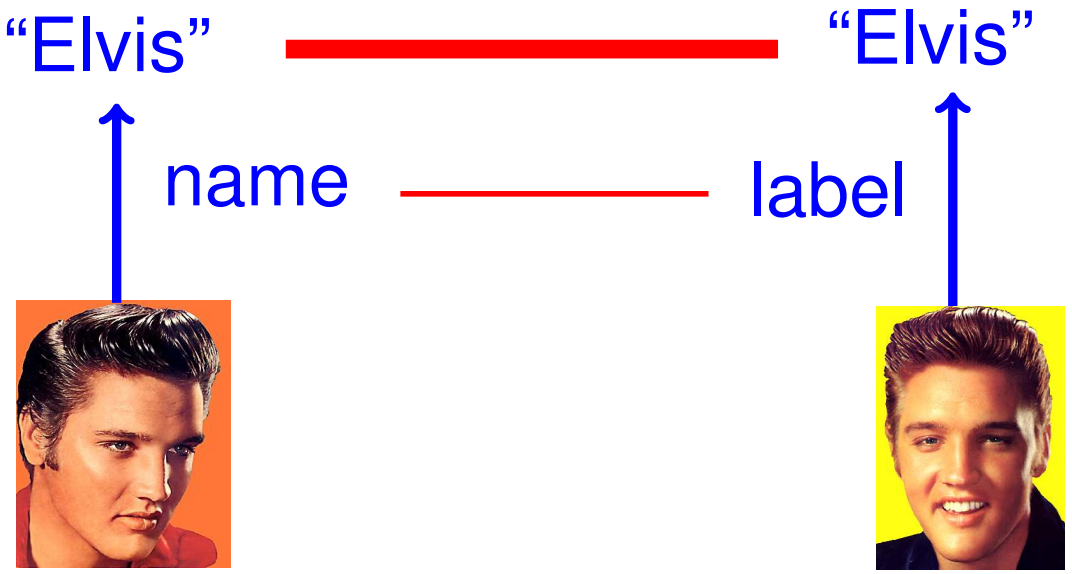
1. Match literals

Match classes, entities, & relations



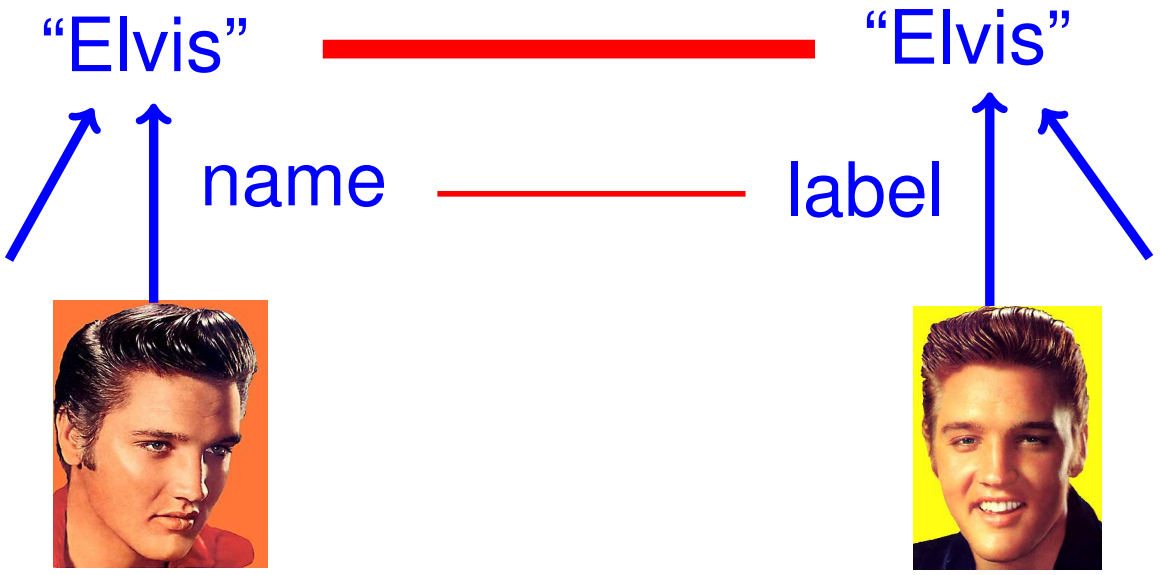
2. Assume small equivalence of all relations

Match classes, entities, & relations



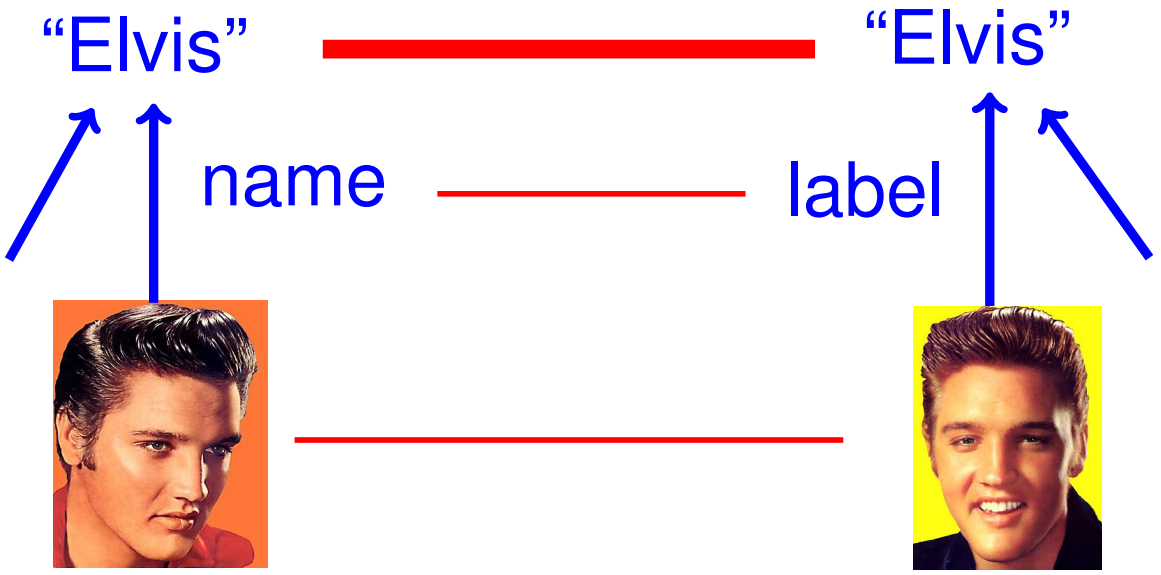
2. Assume small equivalence of all relations

Match classes, entities, & relations



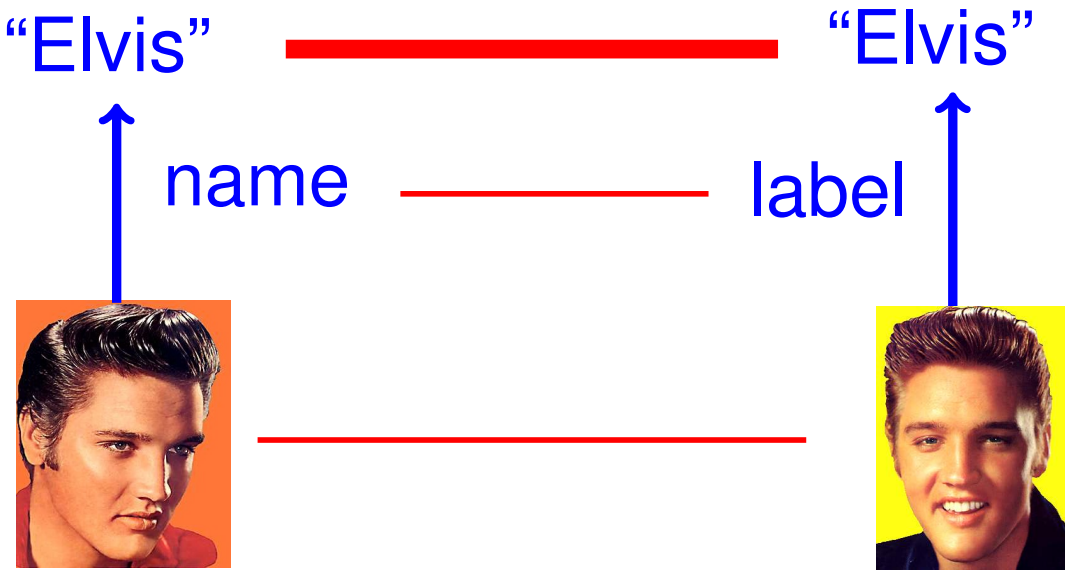
3. If entities share a relation that is highly inverse functional, and object is matched, match them.

Match classes, entities, & relations



3. If entities share a relation that is highly inverse functional, and object is matched, match them.

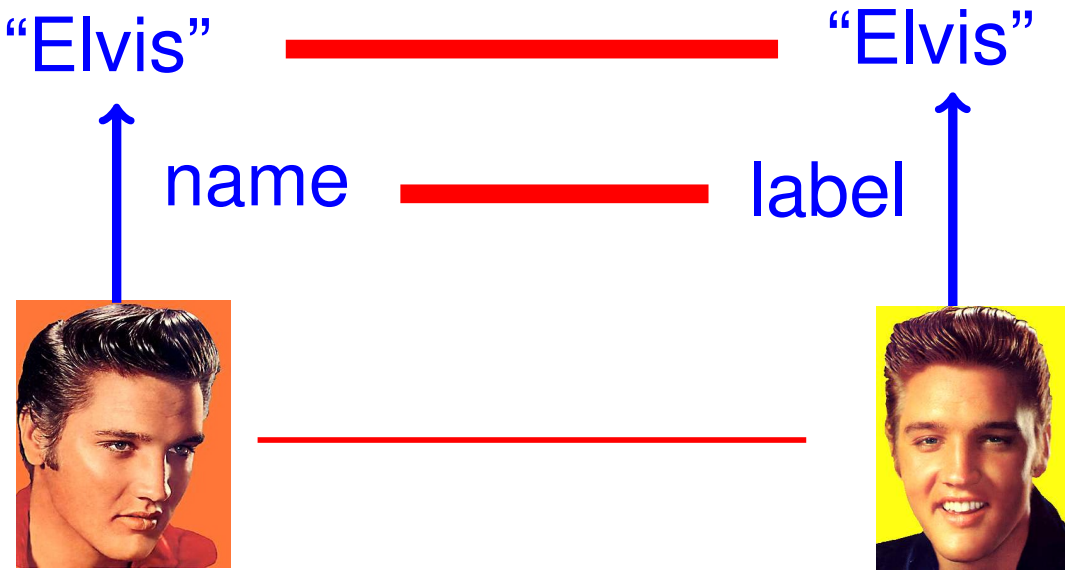
Match classes, entities, & relations



4. If relations share many pairs,

   increase their match

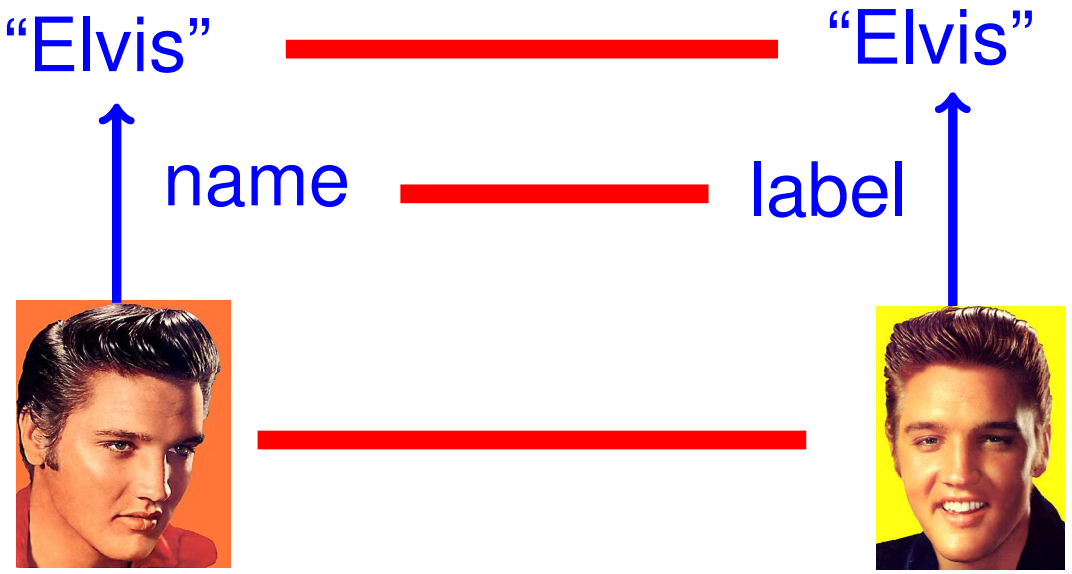
Match classes, entities, & relations



4. If relations share many pairs,

   increase their match

Match classes, entities, & relations

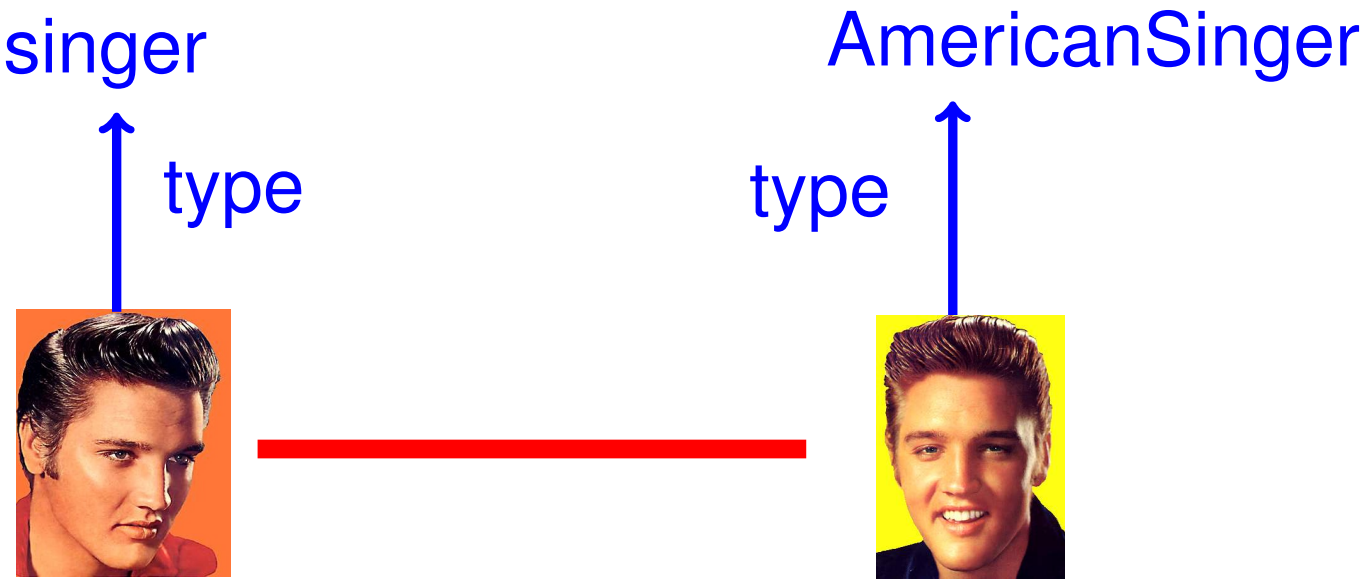


5. Iterate

$$P(e_1 \equiv e_2) = \Pi_{42}^{1}\, \alpha^{\beta} \ldots P(r_1 \subseteq r_2) \ldots$$

$$P(r_1 \subseteq r_2) = 42\phi \ldots P(e_1 = e_2) \ldots$$

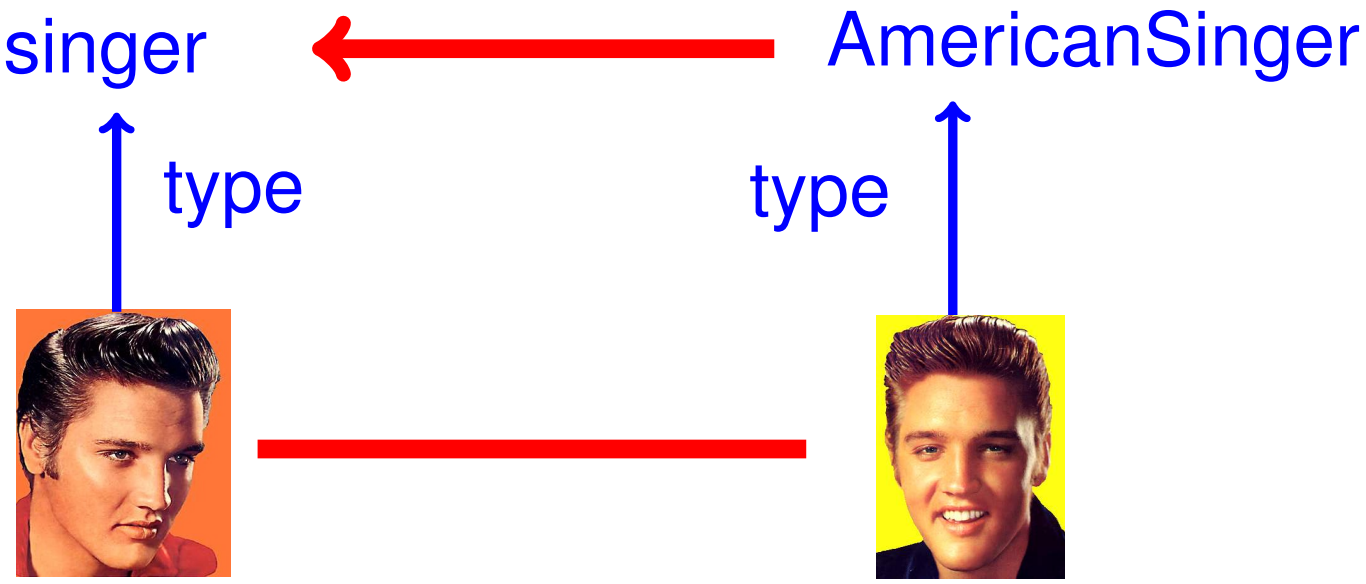
Match classes, entities, & relations



6. Compute class subsumption

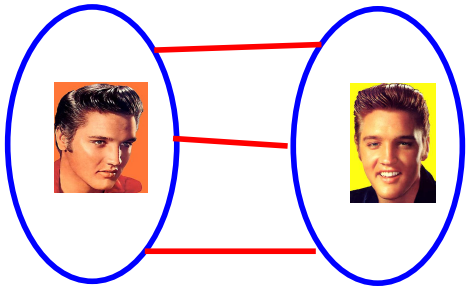
$$P(c_1 \subseteq c_2) = arcsin(4.1125) \times P(e_1 \equiv e_2) \times .$$

Match classes, entities, & relations



6. Compute class subsumption

PARIS:match entities,classes,relations

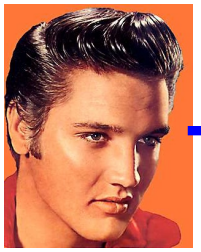


PARIS matches DBpedia & YAGO

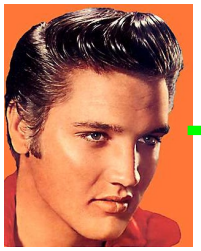
- in 2 hours
- with 90% accuracy

[VLDB 2012]

Matching heterogeneous KBs

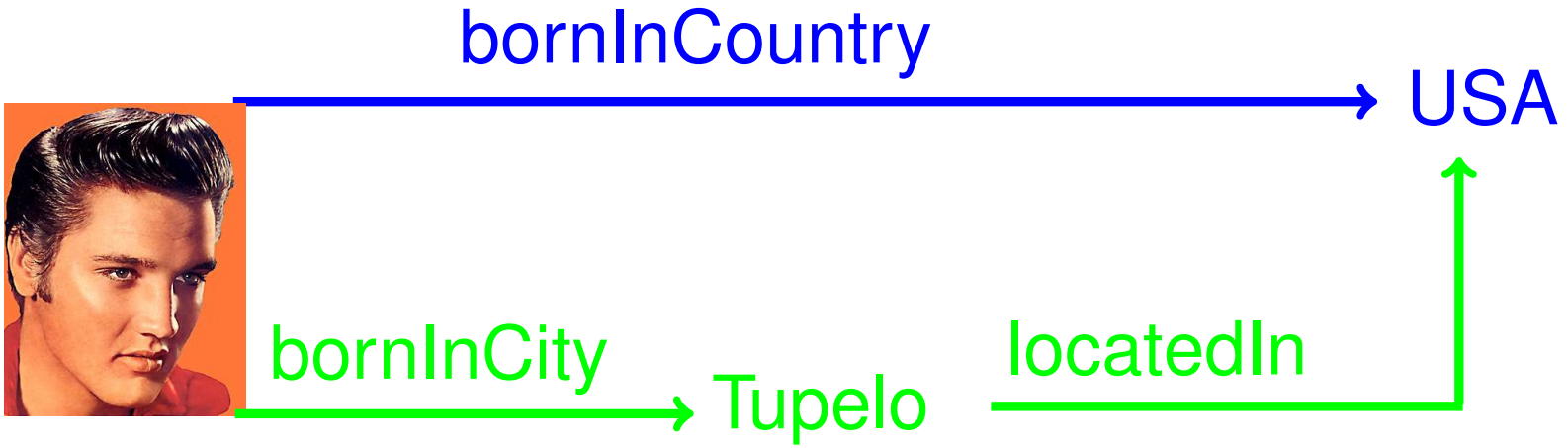


bornInCountry → USA

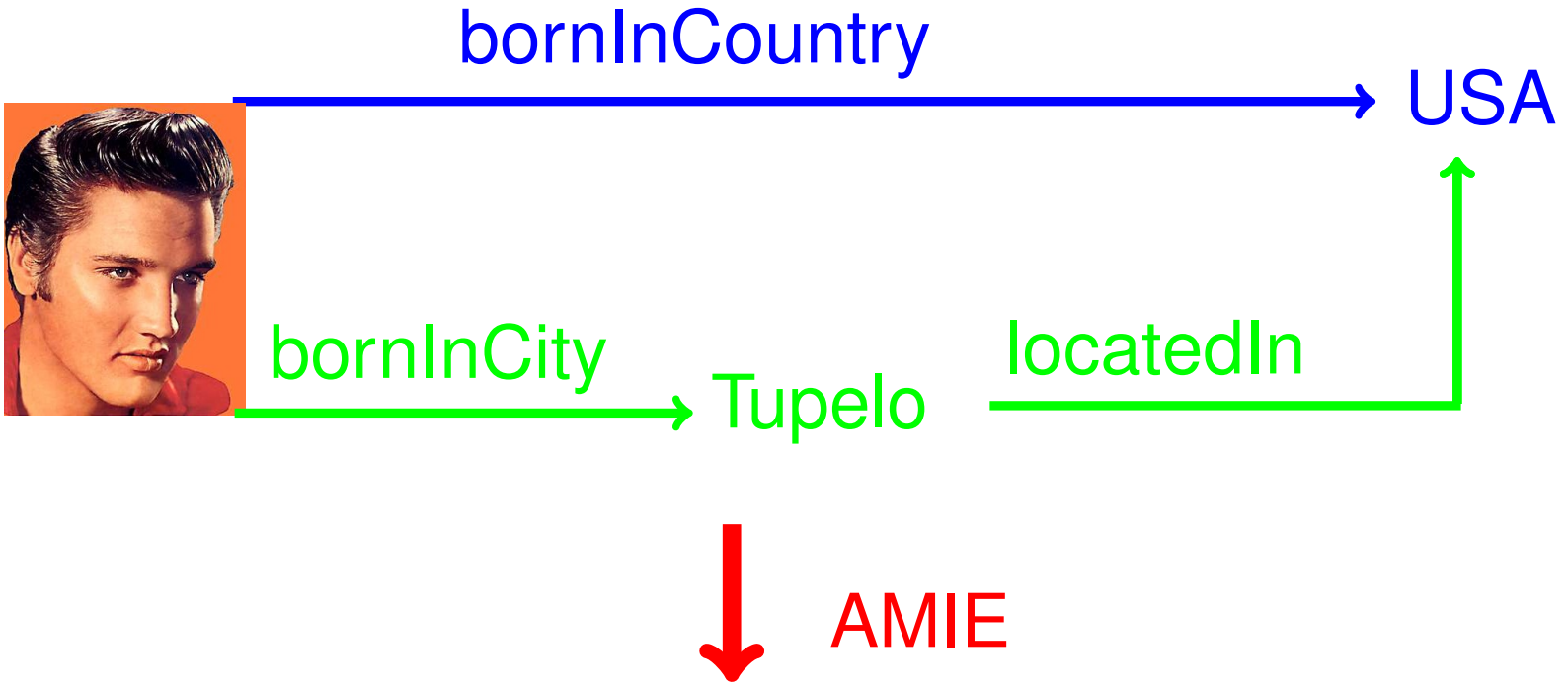


bornInCity → Tupelo → locatedIn → USA

1. Coalesce the KBs



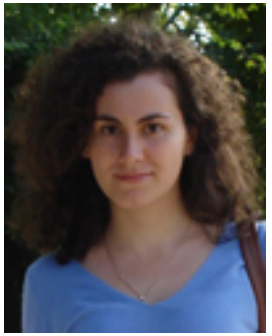
2. Mine rules



$$bornInCity(x, y) \wedge locatedIn(y, z) \Rightarrow bornInCountry(x, z)$$

"ROSA rule"

ROSA rules match ontologies



[AKBC 2013]

$$bornInCity(x, y) \wedge locatedIn(y, z) \Rightarrow bornInCountry(x, z)$$

"ROSA rule"

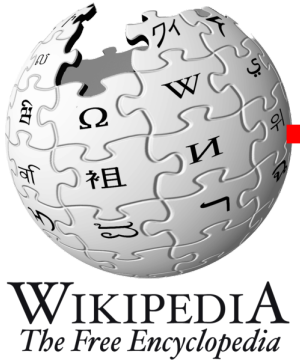
Work on Ontologies

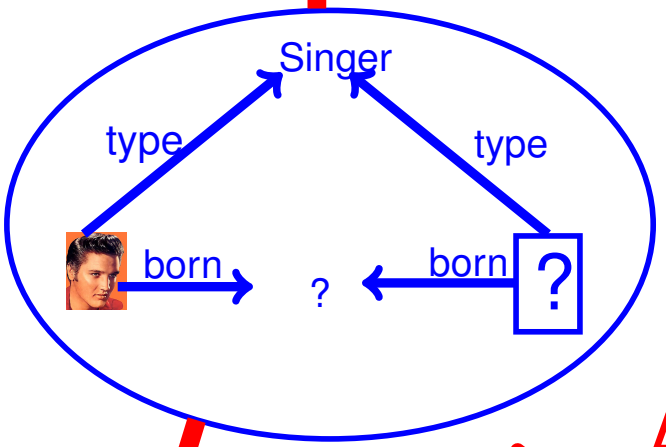
Applying ontologies

Le Monde

Constructing ontologies

✓





Protecting ontologies

Mining ontologies

✓

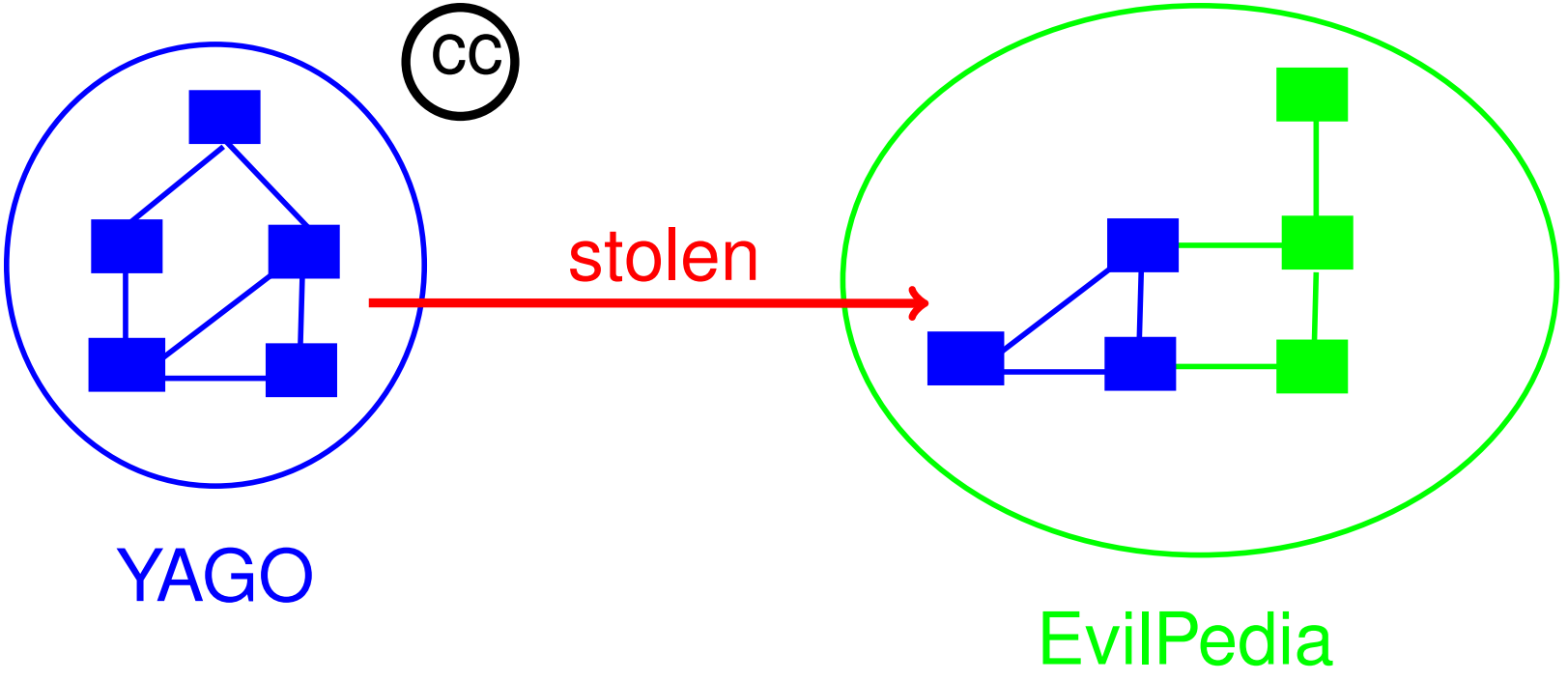
$A \wedge B \Rightarrow C$

Aligning ontologies

✓

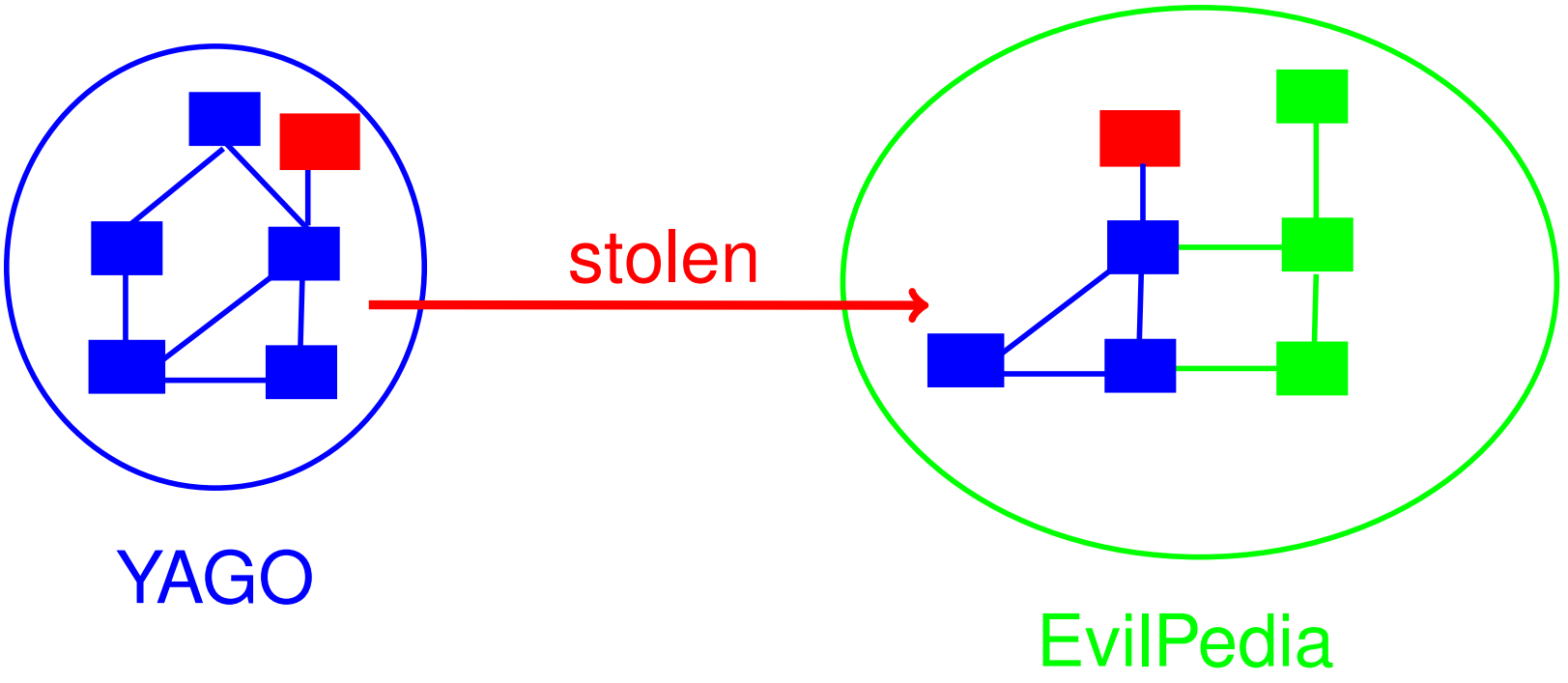
Plagiarism

People may “steal” from other ontologies without giving due credit. Most ontologies have licenses that require attribution.



Additive Watermarking

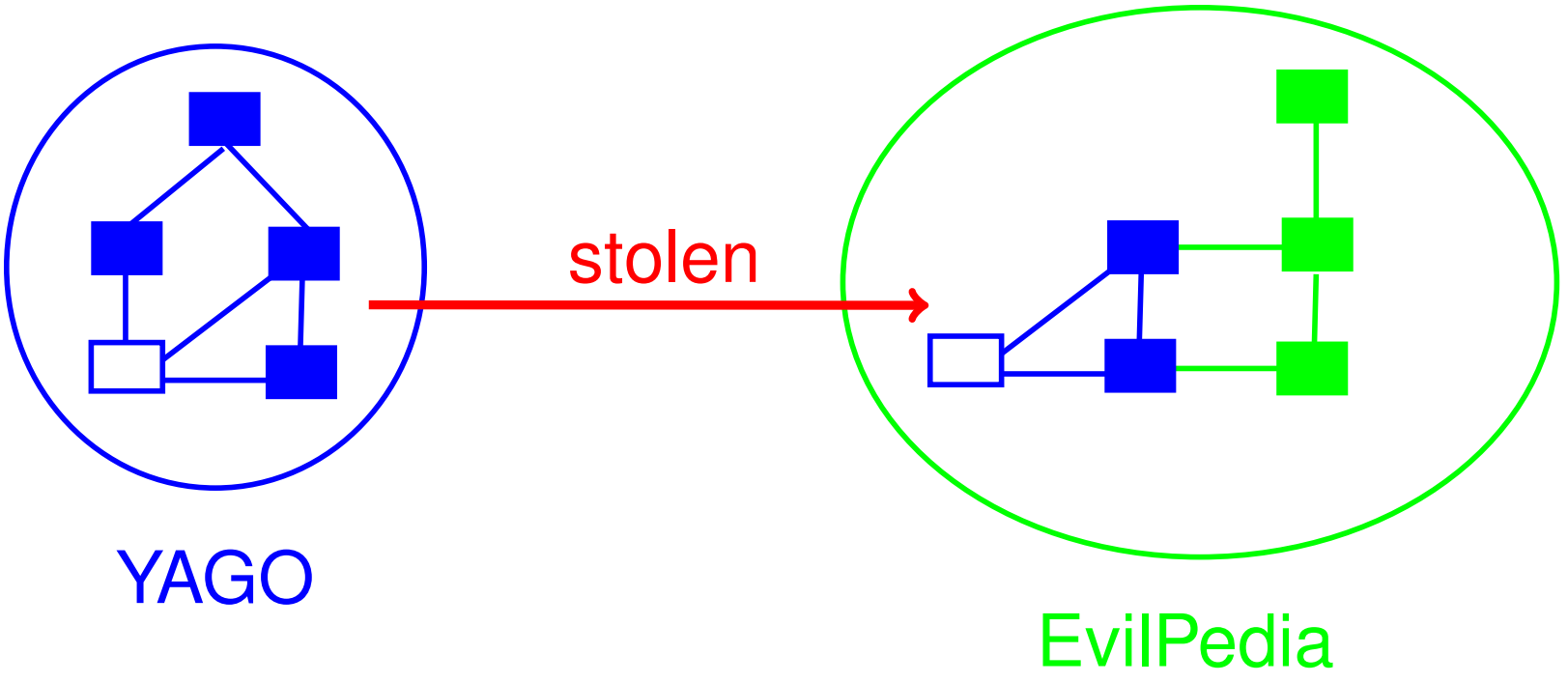
By adding a few fake facts to the source ontology, one can prove theft in the target ontology.



[WWW 2012 demo]

Subtractive Watermarking

One can also prove theft by selectively removing facts from the source ontology.



[ISWC 2011]

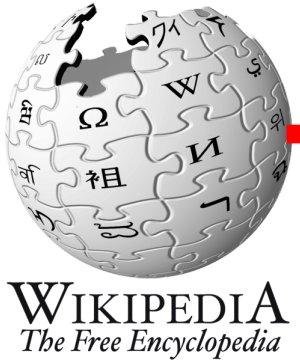
Work on Ontologies

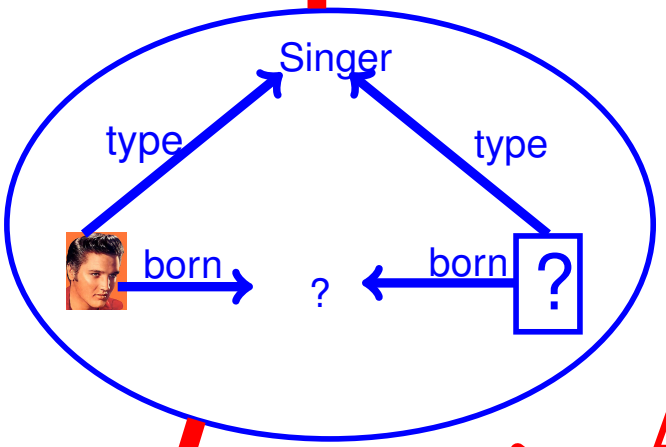
Applying ontologies

Le Monde

Constructing ontologies ✓

Protecting ontologies ✓



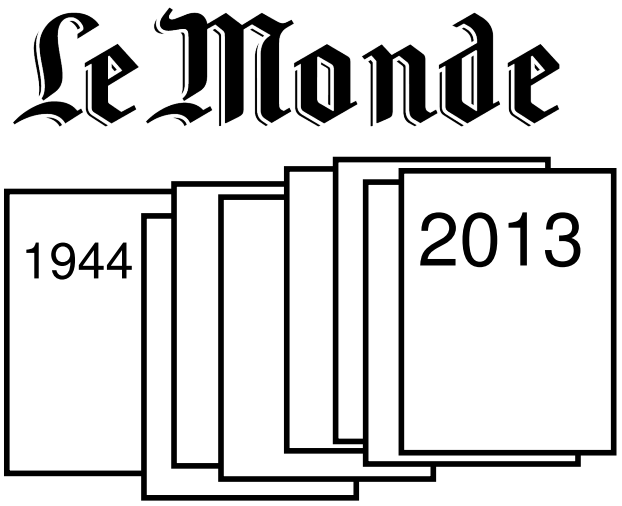


Mining ontologies ✓

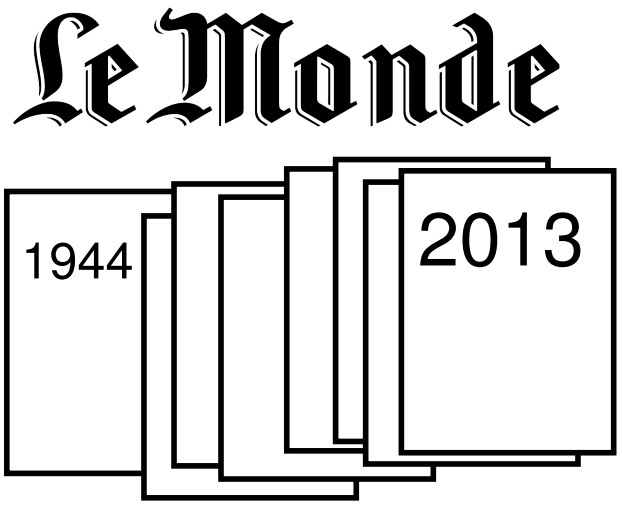
Aligning ontologies ✓

$A \wedge B \Rightarrow C$

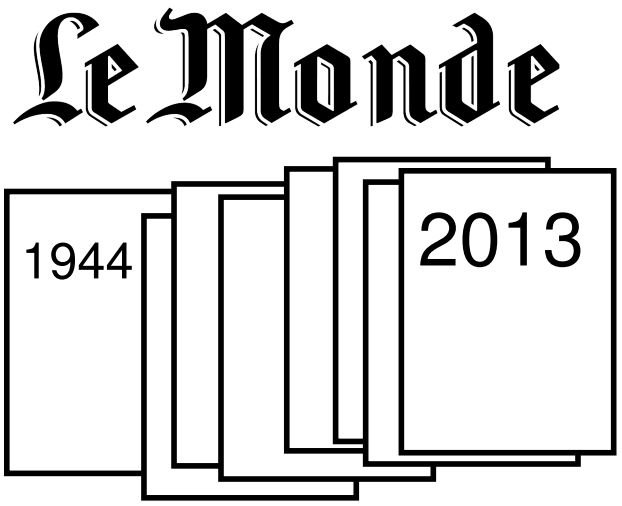
Mining Le Monde

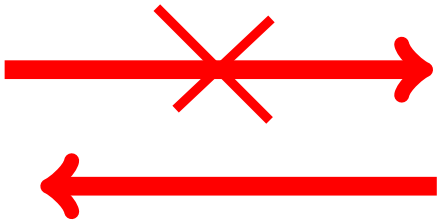


Mining Le Monde



Mining Le Monde





Mining Le Monde

Le Monde

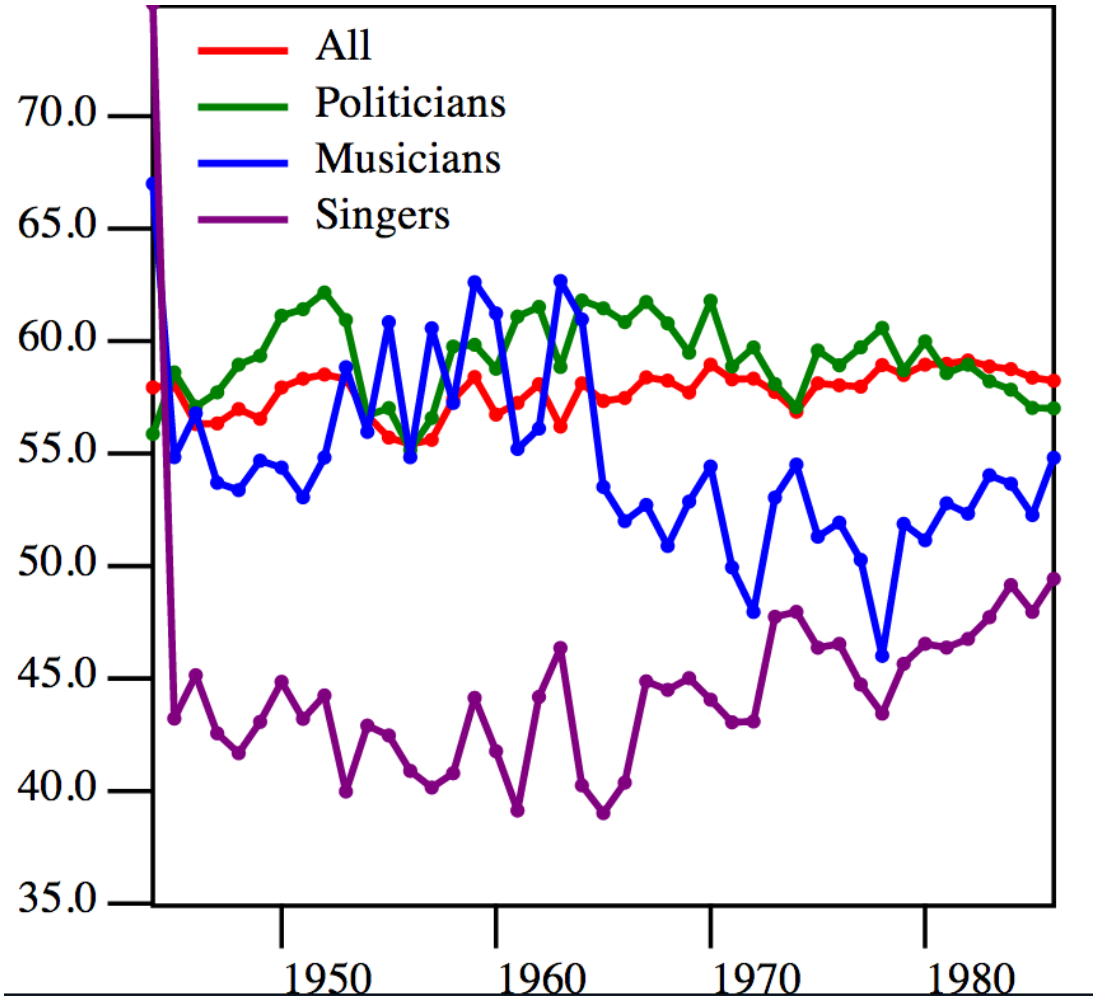
time	place	entity
1967	USA	

yago
select knowledge

Mining Le Monde

Le Monde

yago
select knowledge

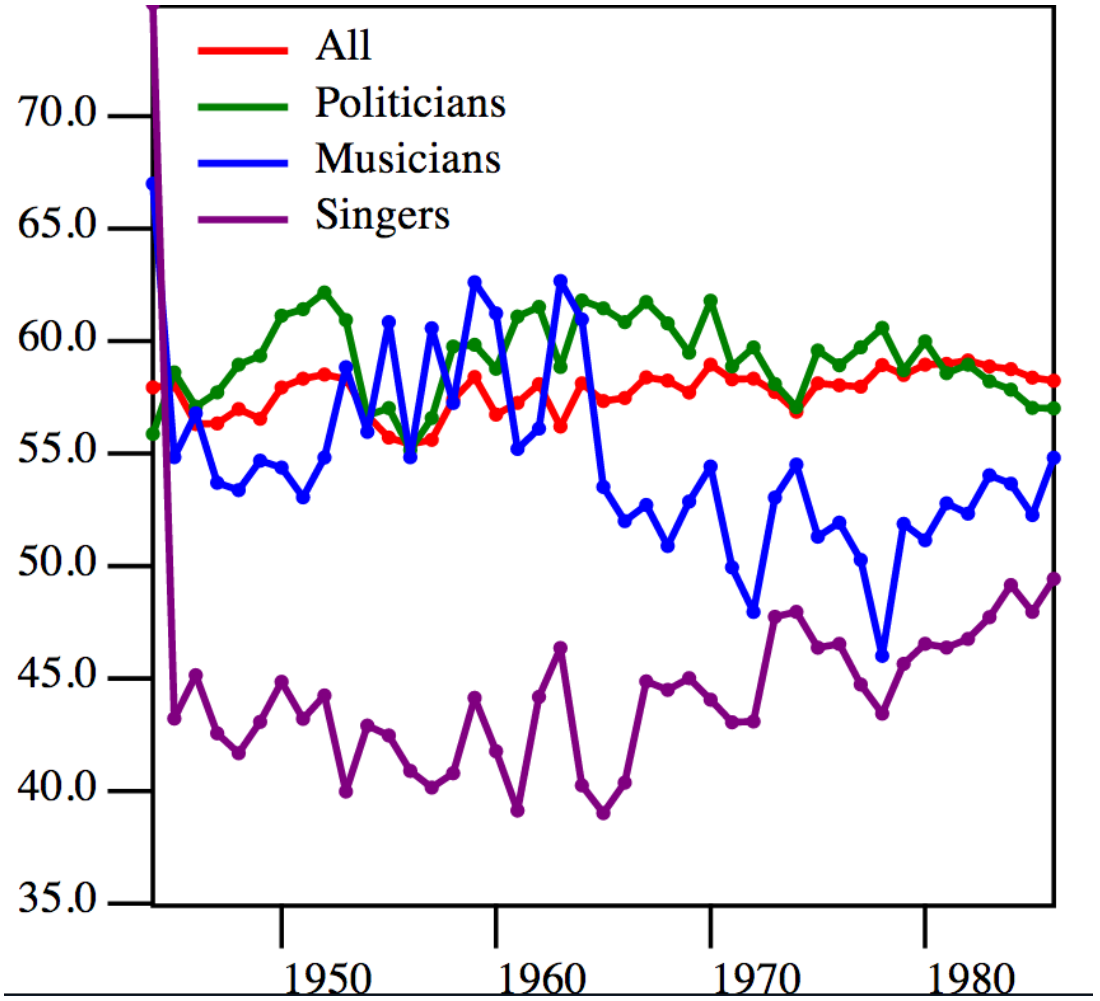


Average age of people mentioned

Mining Le Monde

Le Monde

yago
select knowledge



[AKBC 2013]

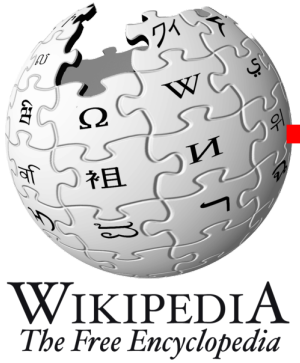
Work on Ontologies

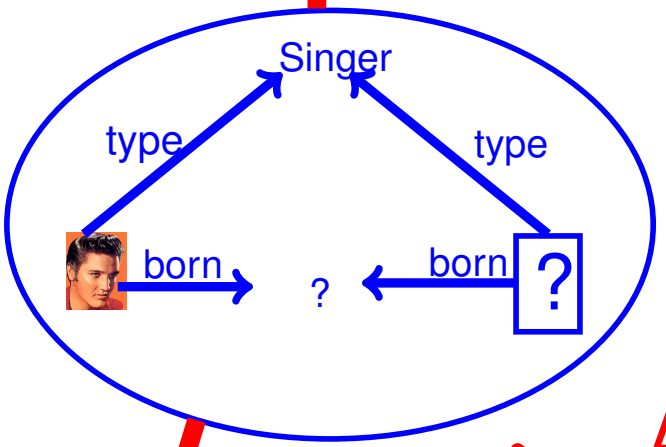
Applying ontologies ✓

Le Monde

Constructing ontologies ✓

Protecting ontologies ✓





Mining ontologies ✓

Aligning ontologies ✓

$A \wedge B \Rightarrow C$

YAGO can answer our question

Find x such that

x is called “Elvis”

x is a singer

x was born after 1970

http://yago-knowledge.org

YAGO can answer our question

Find x such that

x is called “Elvis”

x is a singer

x was born after 1970

Elvis Crespo,

singer,

born 1971

http://yago-knowledge.org

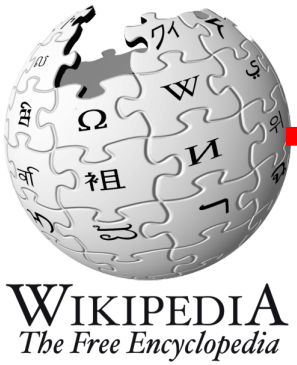
Thank you for your attention!

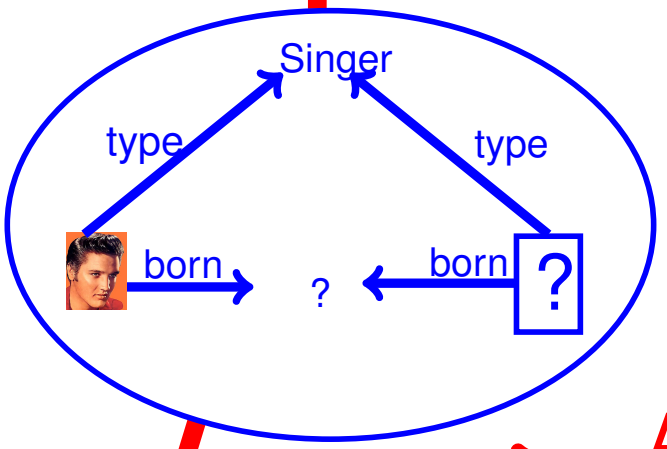
Applying ontologies ✓

Le Monde

Constructing ontologies ✓

Protecting ontologies ✓





Mining ontologies ✓

Aligning ontologies ✓

$$A \wedge B \Rightarrow C$$

Slides done with Powerline, my free SVG slide editor with LaTex support