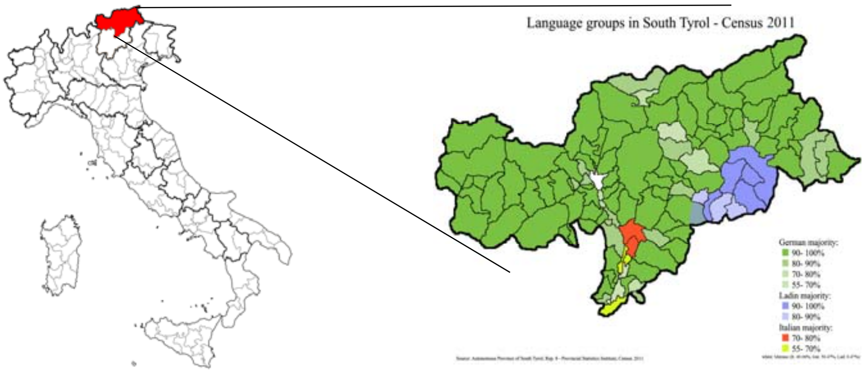
Why Databases Should Know How Much They Know

Werner Nutt

Free University of Bozen-Bolzano, Italy

unibz Fakultät für Informatik
Facoltà di Scienze e Tecnologie informatiche
Faculty of Computer Science

Bolzano is an Autonomous Province of Italy



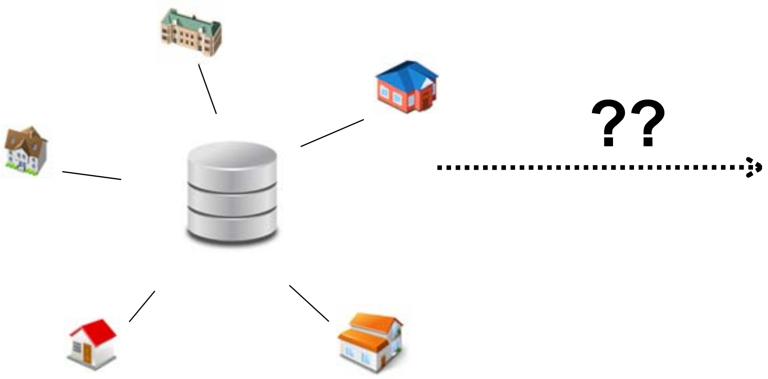
- Has its own school system

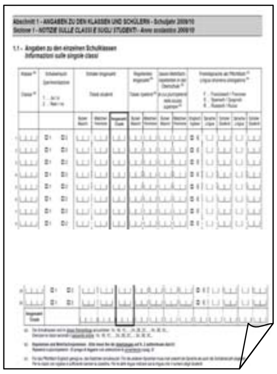
  … with its own data management

School Data Management in Bolzano …

Decentrally maintained database

Statistical reports





Abschnitt 1 - ANGABEN ZU DEN KLASSEN UND SCHÜLERN - Schuljahr 2009/10
Sezione 1 - NOTIZIE SULLE CLASSI E SUGLI STUDENTI - Anno scolastico 2009/10

1.1 - Angaben zu den einzelnen Schulklassen
Informazioni sulle singole classi

generally incomplete

require complete data

… Gave Rise to Work on Data Completeness

Simon Razniewski

Ognjen Savkovic

Fariz Darari

Radityo Eko Prasoyo

Main publications: VLDB 11, CIKM 12, ISWC 13, SIGSPATIAL 14, SIGMOD 15, ICDT 16, ICWE 16, ADMA 17, CIKM 18, ACM TWEB 18, K-CAP 19, SWJ (to appear)

Demos: CIKM 12, VLDB 13, COLD@ISWC 2016

Systems: COOL-WD, Recoin (plugins for Wikidata)

Papers can be retrieved from http://www.inf.unibz.it/~nutt/publications.html

… Gave Rise to Work on Data Completeness

Simon Razniewski

Ognjen Savkovic

Fariz Darari

Radityo Eko Prasoyo

Award for outstanding PhD thesis at International Semantic Web Conference 2018

Main publications: VLDB 11, CIKM 12, ISWC 13, SIGSPATIAL 14, SIGMOD 15, ICDT
CIKM 18, ACM TWEB 18, K-CAP 19, SWJ (to appear)

Demos: CIKM 12, VLDB 13, COLD@ISWC 2016

Systems: COOL-WD, Recoin (plugins for Wikidata)

Papers can be retrieved from http://www.inf.unibz.it/~nutt/publications.html

Incompleteness in the School Data

Facts in real world

result(Giulia, Math, A)
result(Paul, Math, A)

Facts in school database

result(Giulia, Math, NULL)

Missing information in the school database:
- no entry for Paul (missing record)
- no grade for Giulia (missing value)

Consequence: Query Answers are Incorrect

Query Q: *"How many pupils have grade A in Math?"*

In the real world: Q(result(Giulia, Math, A)
result(Paul, Math, A)) = 2

According to available database: Q(result(Giulia, Math, NULL)) = 0

→ If data is incomplete, query answers become incorrect

Data Completeness is a Key Issue in Data Quality

One of the core dimensions of data quality (besides consistency, accuracy, timeliness, …)

- 2,500,000 hits on Google for "Data Quality" + "Completeness"

Incompleteness typically occurs

- if data sets have many contributors

- in data integration
- in knowledge base construction

The General Approach to Data (In)completeness

Try to complete all the data:

- link business processes to computers
- add more sensors
- resort to data imputation
- compare with known complete data sets
- let humans add data

Shortcomings:

- does not always work …
- what do you do while data are still incomplete?
- or if they cannot be completed at all?

There can be Information about Partial Completeness!

... vocational schools
use the information system
of the province
to manage grades

checked by looking at a database alor

at is not there!

... primary schools
took part in a survey
of Math education

However, we may k whether parts of a database are com te, e.g.,

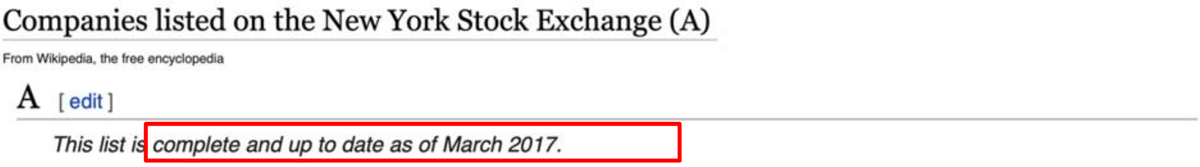
- *"The grades from vocational schools are com lete"*
- *"The Math grades from primary schools are complete"*

This information is not in the database …

… if it were, one could use to assess completeness of database queries

Statements about Completeness are not that Novel

- Wikipedia:



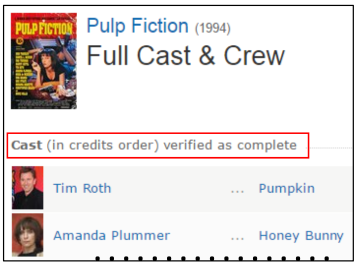
Companies listed on the New York Stock Exchange (A)

From Wikipedia, the free encyclopedia

A [edit]

This list is complete and up to date as of March 2017.

- Internet Movie DB:



Pulp Fiction (1994)

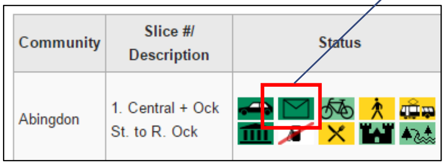
Full Cast & Crew

Cast (in credits order) verified as complete

Tim Roth	... Pumpkin
Amanda Plummer	... Honey Bunny

- OpenStreetmap:

Complete for all street names of Abingdon

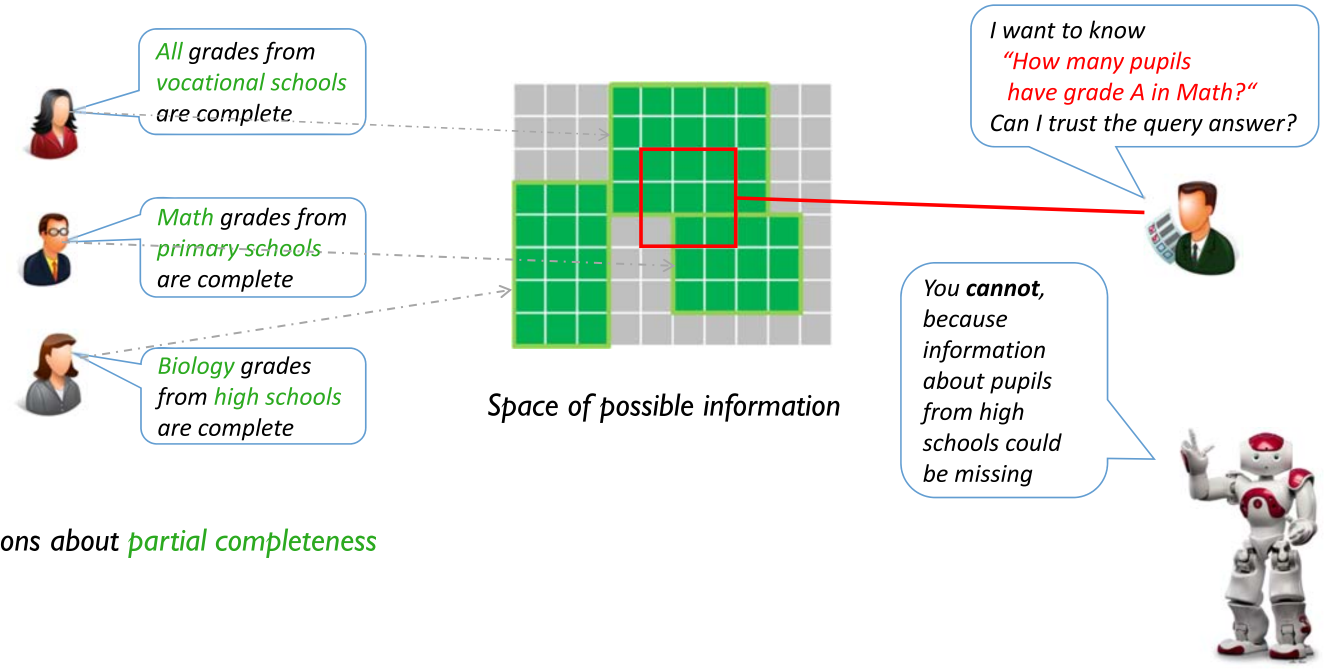


Community	Slice #/ Description	Status
Abingdon	1. Central + Ock St. to R. Ock	

... but they are not formal

If they were, we could use them
for automated reasoning about query completeness

Reasoning about Query Completeness



All grades from vocational schools are complete

Math grades from primary schools are complete

Biology grades from high schools are complete

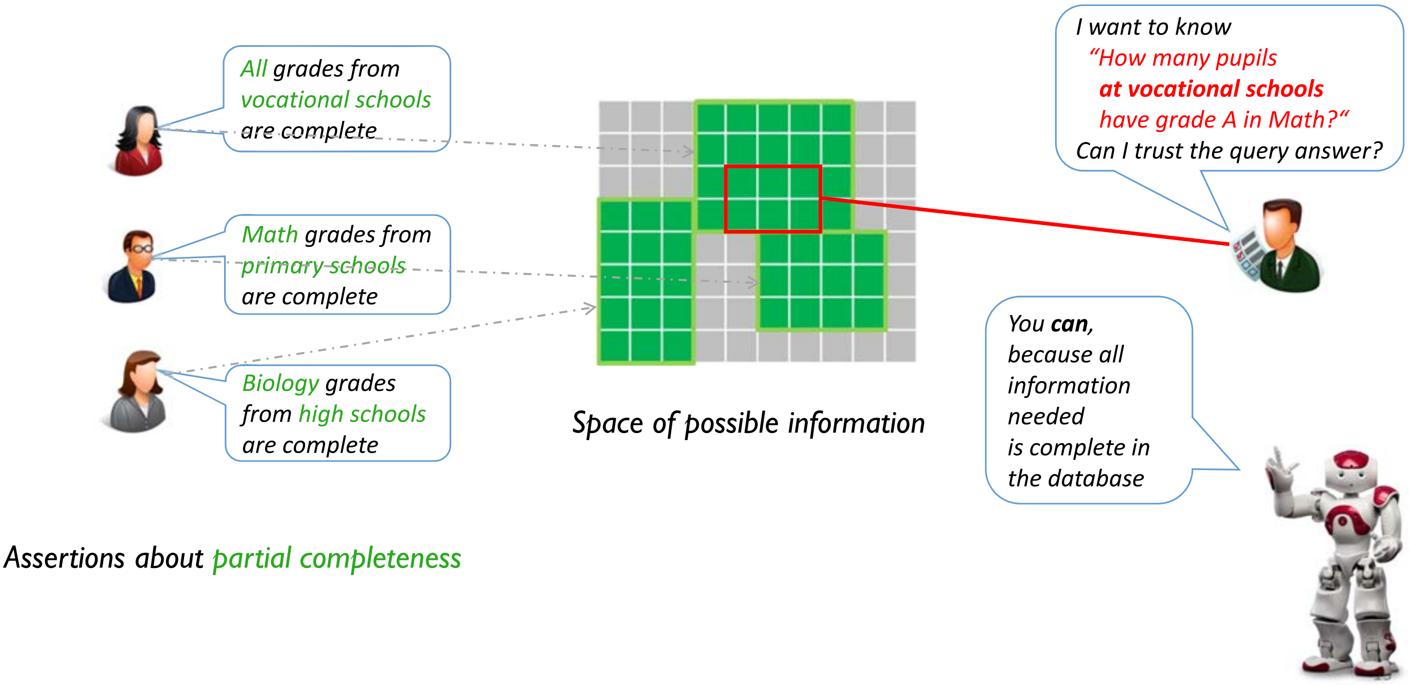
Space of possible information

I want to know "How many pupils have grade A in Math?" Can I trust the query answer?

*You **cannot**, because information about pupils from high schools could be missing*

Assertions about partial completeness

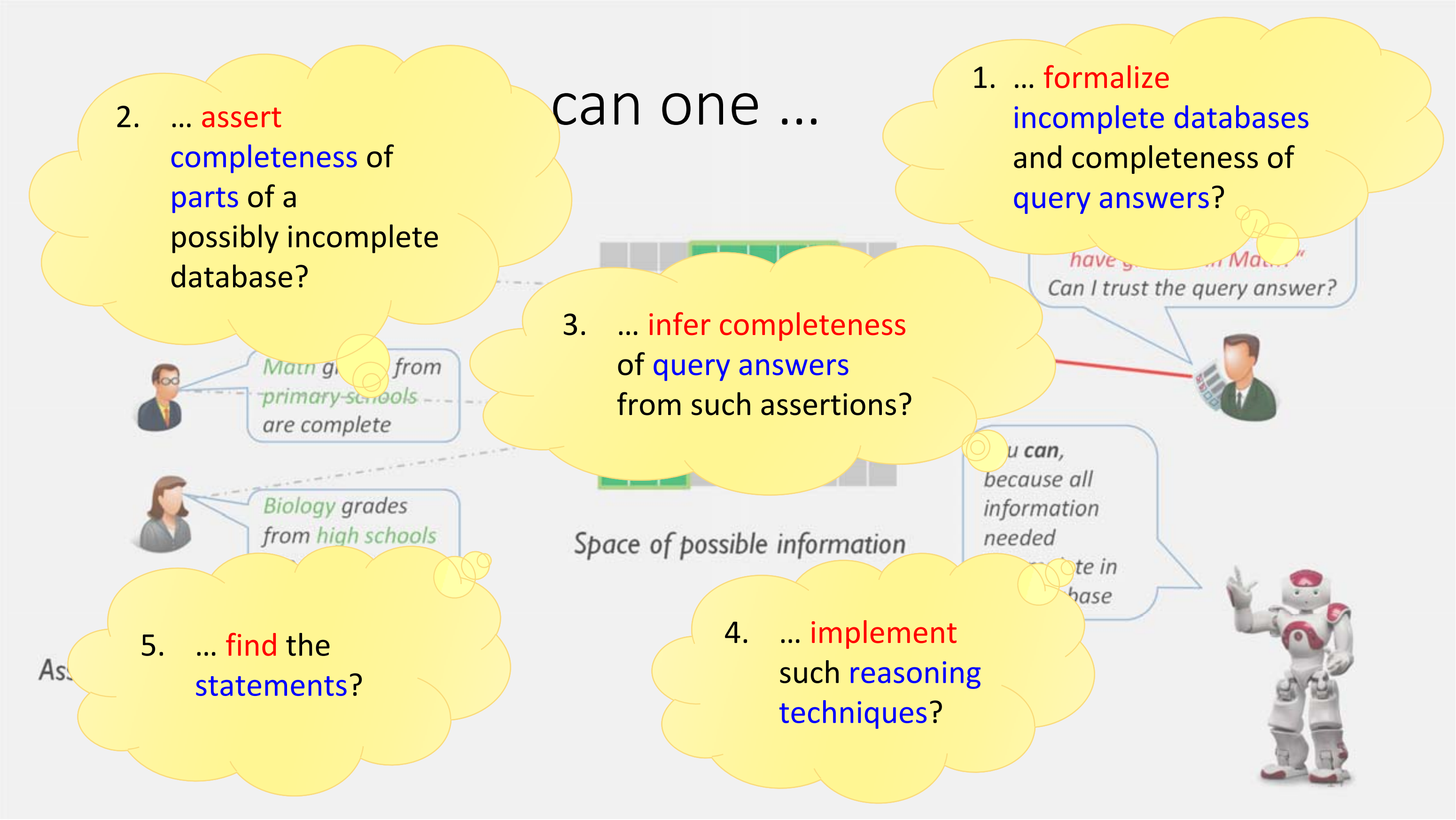
Reasoning about Query Completeness (2)


All grades from vocational schools are complete
Math grades from primary schools are complete
Biology grades from high schools are complete
I want to know “How many pupils at vocational schools have grade A in Math?“ Can I trust the query answer?
You can, because all information needed is complete in the database

Space of possible information

Assertions about partial completeness

… can one …


Math g… from primary schools are complete
Biology grades from high schools
Space of possible information
have g… Math?“ Can I trust the query answer?
u can, because all information needed …te in …base
As

1. … formalize incomplete databases and completeness of query answers?

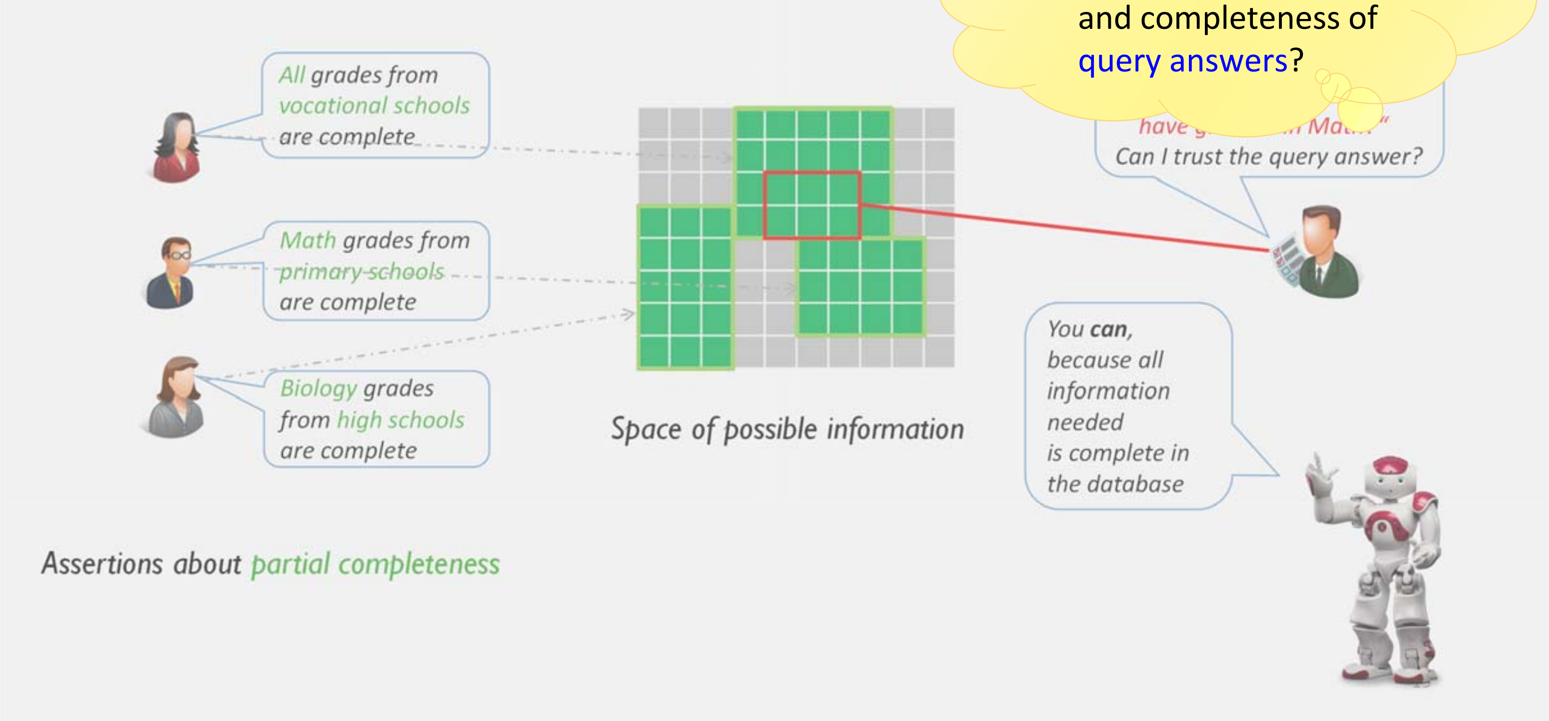
2. … assert completeness of parts of a possibly incomplete database?

3. … infer completeness of query answers from such assertions?

4. … implement such reasoning techniques?

5. … find the statements?

Questions: How can one …


All grades from vocational schools are complete
Math grades from ~~primary schools~~ are complete
Biology grades from high schools are complete
Assertions about partial completeness
Space of possible information
have g… …n Math"
Can I trust the query answer?
You **can**, because all information needed is complete in the database

1. … formalize incomplete databases and completeness of query answers?

A Database …

- has a schema

pupil(name, schoolName, schoolType)

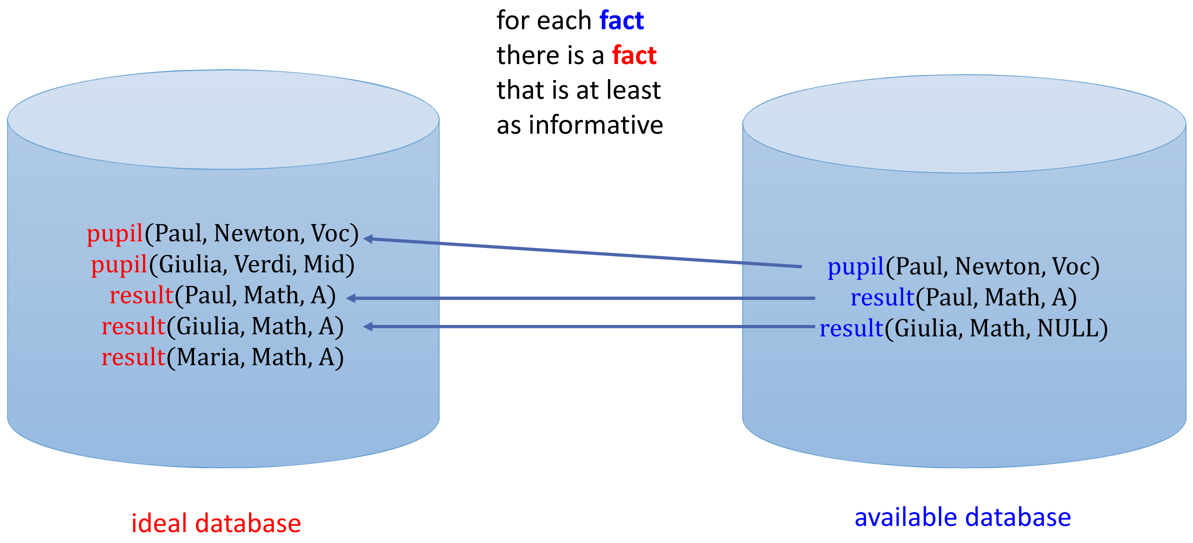
result(name, subject, grade)

- is a collection of records

pupil(Paul, Newton, Voc)
result(Paul, Math, A)
result(Giulia, Math, NULL)

Incomplete Databases

When talking about incompleteness, we implicitly refer to a complete reference



for each **fact**
there is a **fact**
that is at least
as informative

pupil(Paul, Newton, Voc)
pupil(Giulia, Verdi, Mid)
result(Paul, Math, A)
result(Giulia, Math, A)
result(Maria, Math, A)

pupil(Paul, Newton, Voc)
result(Paul, Math, A)
result(Giulia, Math, NULL)

ideal database

available database

Formalization: Incomplete Database

[Motro 1989]

An *incomplete database* $\boldsymbol{D}$ is a pair of

an ideal database D^i and

an available database D^a

$$\boldsymbol{D} = \left(D^i, D^a\right)$$

such that

for each record in D^a there is

a “more informative” record in D^i

For databases w/o Nulls,
this means

$$D^a \subseteq D^i$$

Queries

Reasoning for arbitrary SQL queries is impossible (undecidability of 1st order logic)

We concentrate on single block SQL queries (possibly with **DISTINCT**):

```
SELECT r.name
FROM   pupil p, result r
WHERE  r.name = p.name AND
       r.subject = ’Math’ AND
       r.grade = ’A’ AND
       p.schoolType = ’Voc’
```

“Which pupils at vocational schools had grade A?”

We write them as rules

$$Q(n) \leftarrow \text{result}(n, \text{Math}, \text{A}),\ \text{pupil}(n, sn, \text{Voc})$$

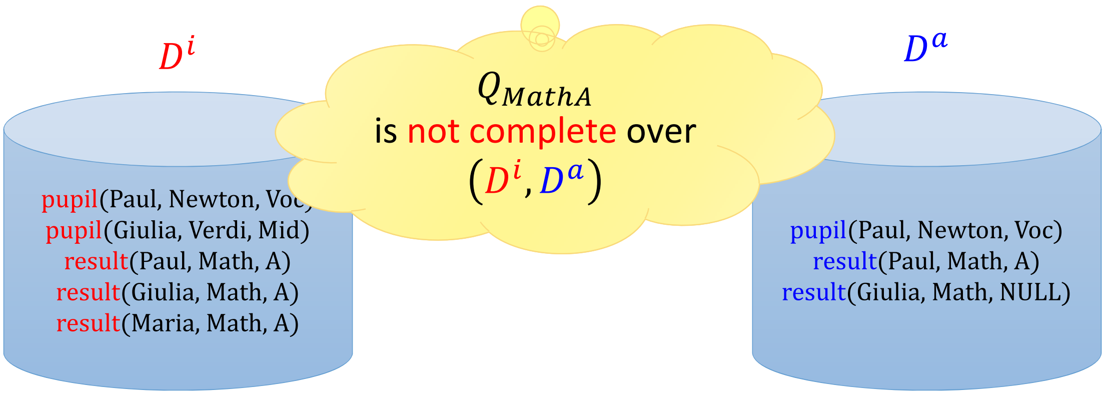
also called “conjunctive queries”

Query Completeness

$$Q_{MathA}(n) \leftarrow \text{result}(n, \text{Math}, \text{A})$$

$Q_{MathA}(D^i) = \{\text{Paul, Giulia, Maria}\}$

$Q_{MathA}(D^a) = \{\text{Paul}\}$



D^i

pupil(Paul, Newton, Voc)
pupil(Giulia, Verdi, Mid)
result(Paul, Math, A)
result(Giulia, Math, A)
result(Maria, Math, A)

Q_{MathA}
is not complete over
(D^i, D^a)

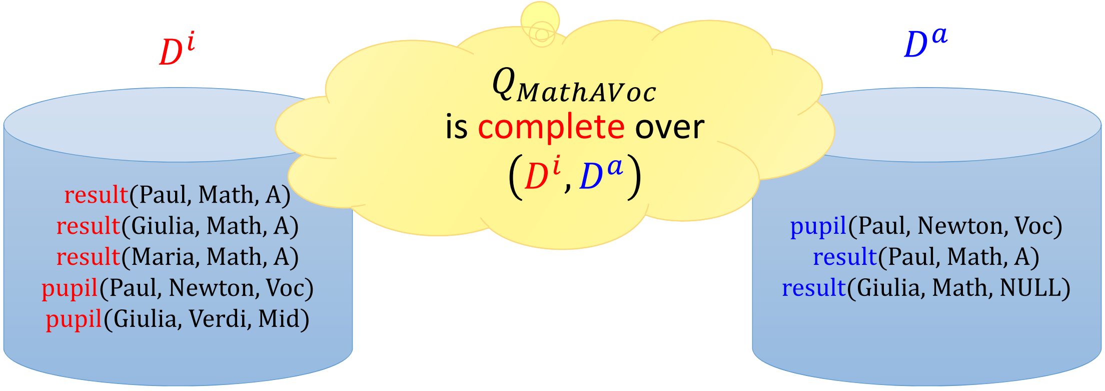
D^a

pupil(Paul, Newton, Voc)
result(Paul, Math, A)
result(Giulia, Math, NULL)

Example: Query Completeness (2)

$$Q_{MathAVoc}(n) \leftarrow \text{result}(n, \text{Math}, \text{A}),\ \text{pupil}(n, sn, \text{Voc})$$

$Q_{MathAVoc}(D^i) = \{\text{Paul}\}$ $Q_{MathAVoc}(D^a) = \{\text{Paul}\}$



D^i

result(Paul, Math, A)
result(Giulia, Math, A)
result(Maria, Math, A)
pupil(Paul, Newton, Voc)
pupil(Giulia, Verdi, Mid)

$Q_{MathAVoc}$ is complete over (D^i, D^a)

D^a

pupil(Paul, Newton, Voc)
result(Paul, Math, A)
result(Giulia, Math, NULL)

Formalization: Query Completeness

Query Q

"The answer to Q is complete"

Notation: $\text{Compl}(Q)$

To be precise, we must distinguish between queries w/ and w/o **DISTINCT**

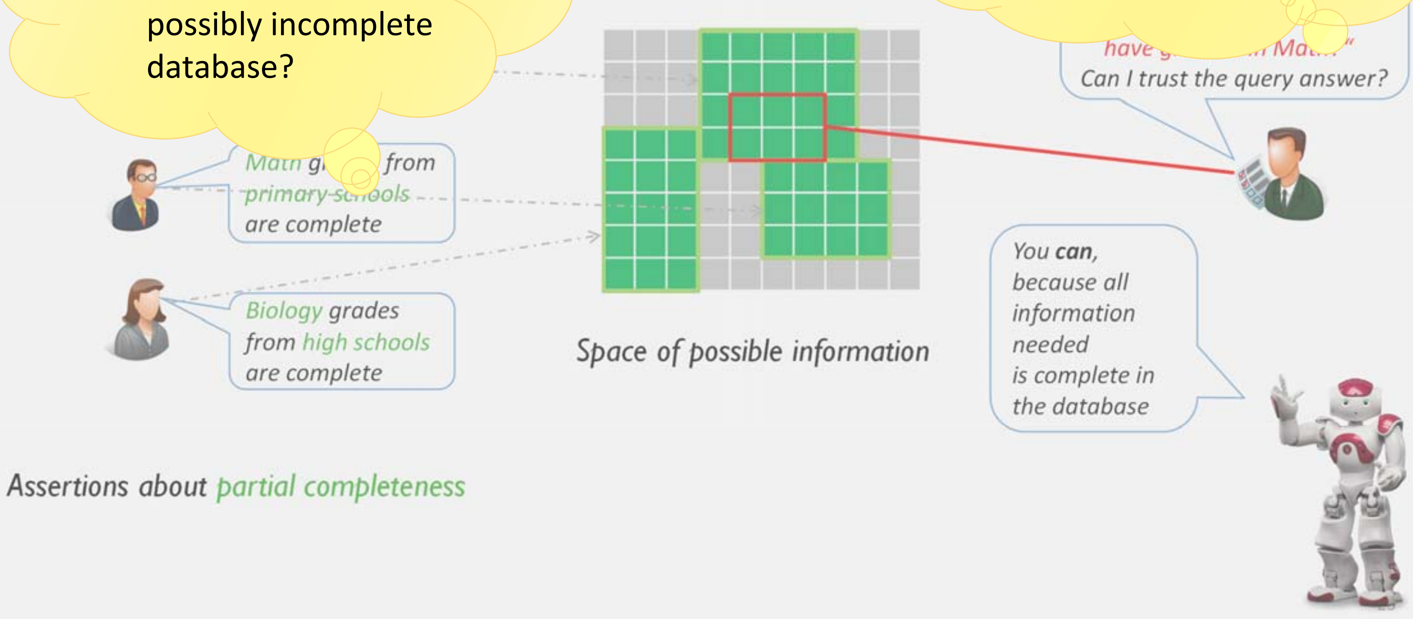
Semantics:

$$(D^i, D^a) \vDash \text{Compl}(Q) \quad \text{iff} \quad Q(D^i) = Q(D^a)$$

can one …

1. … formalize incomplete databases and completeness of query answers?

2. … assert completeness of parts of a possibly incomplete database?


Math g… from primary schools are complete
Biology grades from high schools are complete
Assertions about partial completeness
Space of possible information
have g… …Ma…"
Can I trust the query answer?
You can, because all information needed is complete in the database

Completeness Statements: Idea

Based on [Levy 96]

"The table result *contains all result records of pupils from vocational schools"*

means

"If there is a record $\mathrm{result}(n,s,g)$ in D^i,

and there is a record $\mathrm{pupil}(n,sn,\mathrm{Voc})$ in D^i,

then the record $\mathrm{result}(n,s,g)$ is also in D^a "

We treat here relations in D^i and D^a as different, by tagging them with i and a

This can be expressed by the rule

$$\mathrm{result}^i(n,s,g),\ \mathrm{pupil}^i(n,sn,\mathrm{Voc}) \rightarrow \mathrm{result}^a(n,s,g)$$

Idea: an incomplete db (D^i, D^a) *satisfies the statement iff it satisfies the rule*

Example: Satisfaction of Completeness Statements

$$\mathrm{result}^i(n, s, g),\ \mathrm{pupil}^i(n, sn, \mathrm{Voc}) \rightarrow \mathrm{result}^a(n, s, g)$$

D^i holds over (D^i, D^a) D^a

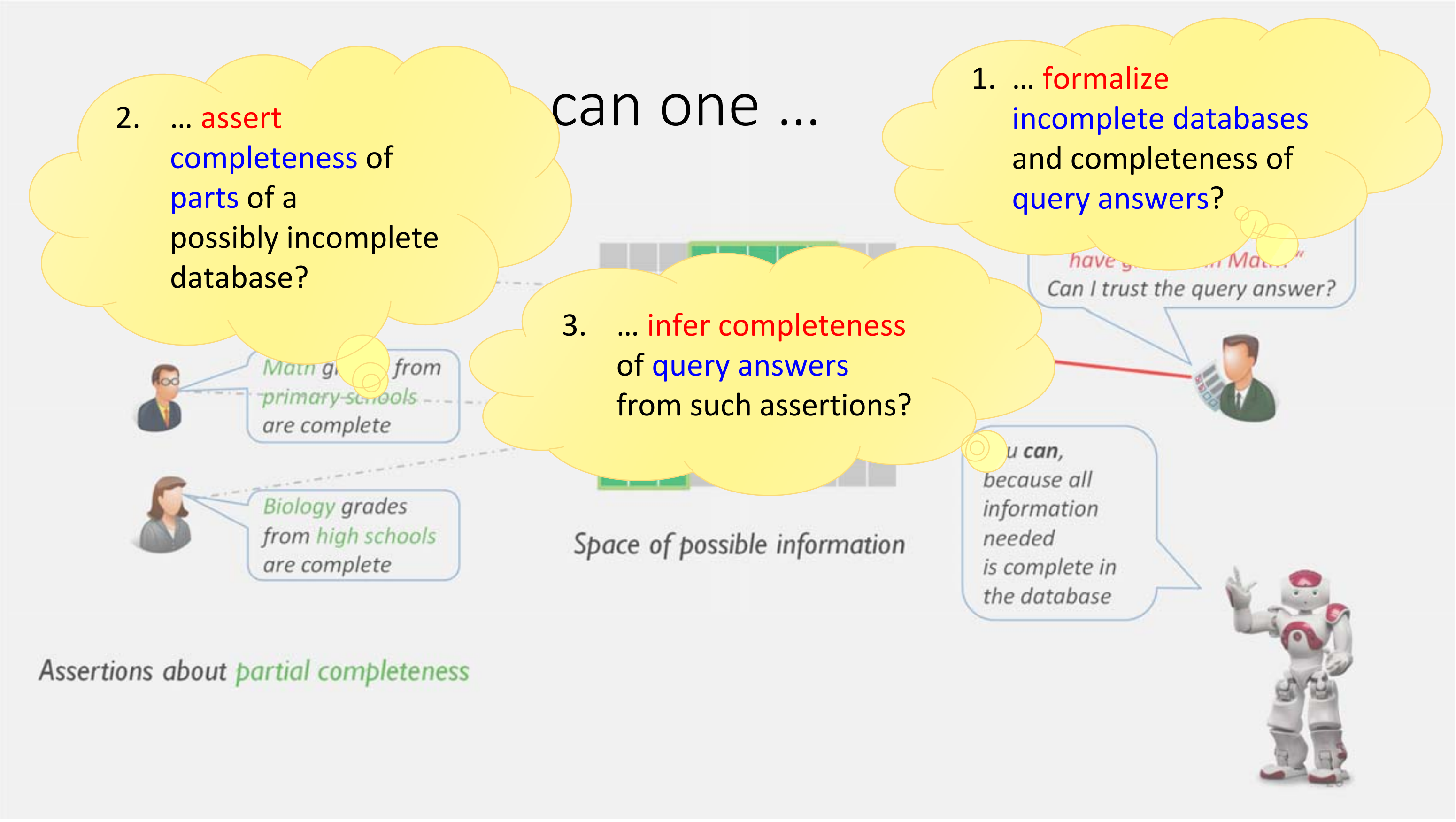
D^i:

pupil(Paul, Newton, Voc)
pupil(Giulia, Verdi, Mid)
result(Paul, Math, A)
result(Giulia, Math, A)
result(Maria, Math, A)

D^a:

pupil(Paul, Newton, Voc)
result(Paul, Math, A)
result(Giulia, Math, NULL)

because result(Paul, Math, A) is in D^a



can one …

1. … formalize incomplete databases and completeness of query answers?

2. … assert completeness of parts of a possibly incomplete database?

3. … infer completeness of query answers from such assertions?

have g… in Math"
Can I trust the query answer?
Math g… from primary schools are complete
Biology grades from high schools are complete
Space of possible information
…u can, because all information needed is complete in the database
Assertions about partial completeness

Data-agnostic Completeness-Reasoning

Given a query Q, a set of completeness statements $\mathbf{C}$

- Is $Q(D^i) = Q(D^a)$ for **every** incomplete db (D^i, D^a) that satisfies $\mathbf{C}$?

Idea: We don't know D^i (and cannot look at D^a),

but we know something about the relationship between D^i and D^a.

Can we conclude that Q gives the same answers over D^i and D^a ?

Data-aware Completeness-Reasoning

Given a query Q, a set of completeness statements $\mathbf{C}$, and a db instance D^a

- Is $Q(D^i) = Q(D^a)$ for **every** $D^i \supseteq D^a$ such that (D^i, D^a) satisfies $\mathbf{C}$?

Idea: We don't know D^i,

but we know D^a, and

we know something about the relationship between D^i and D^a.

Can we conclude that Q gives the same answers over D^i and D^a ?

Reasoning: Starting Point

Query: *"Which pupils at vocational schools had an A in Math?"*

$$Q_{MathAVoc}(n) \leftarrow \mathrm{result}(n, \mathrm{Math}, \mathrm{A}),\ \mathrm{pupil}(n, sn, \mathrm{Voc})$$

Completeness statements:

- *"All result records of pupils at vocational schools are available"*

$$\mathrm{result}^i(n, s, g),\ \mathrm{pupil}^i(n, sn, \mathrm{Voc}) \rightarrow \mathrm{result}^a(n, s, g)$$

- *"All pupils are available"*

$$\mathrm{pupil}^i(n, sn, st) \rightarrow \mathrm{pupil}^a(n, sn, st)$$

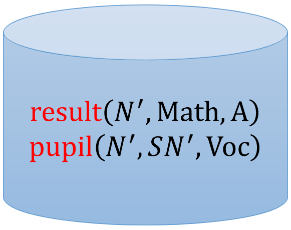
If the Query Returns an Answer over D^i, what is in D^i?

Query: *"Which pupils at vocational schools had an A in Math?"*

$$Q_{MathAVoc}(n) \leftarrow \mathrm{result}(n, \mathrm{Math}, \mathrm{A}),\ \mathrm{pupil}(n, sn, \mathrm{Voc})$$

1. Assume $Q_{MathAVoc}$ returns N' over D^i

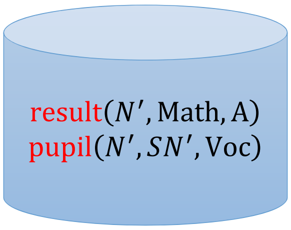
2. See which facts must be in D^i (invent constants N', SN' for variables n, sn)



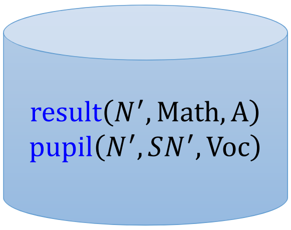
$\mathrm{result}(N', \mathrm{Math}, \mathrm{A})$

$\mathrm{pupil}(N', SN', \mathrm{Voc})$

Apply Completeness Statements to Derive D^a



$\mathrm{result}(N', \mathrm{Math}, \mathrm{A})$
$\mathrm{pupil}(N', SN', \mathrm{Voc})$



$\mathrm{result}(N', \mathrm{Math}, \mathrm{A})$
$\mathrm{pupil}(N', SN', \mathrm{Voc})$

- *“All* result *records of pupils at vocational schools are available“*

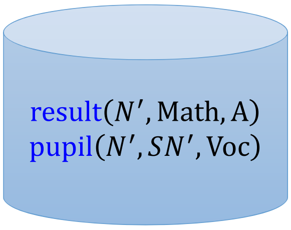
$$\mathrm{result}^i(n, s, g),\ \mathrm{pupil}^i(n, sn, \mathrm{Voc}) \rightarrow \mathrm{result}^a(n, s, g)$$

- *“All pupils are available“*

$$\mathrm{pupil}^i(n, a, sn, st) \rightarrow \mathrm{pupil}^a(n, sn, st)$$

Run Query on D^a

“Which pupils at vocational schools had an A in Math?“



$\mathrm{result}(N', \mathrm{Math}, \mathrm{A})$
$\mathrm{pupil}(N', SN', \mathrm{Voc})$

$Q_{MathAVoc}(D^a) = \{N'\}$ $\quad\Longrightarrow\quad$ N' is also returned over D^a

Summary: If

- an arbitrary N' is returned over the ideal db, and
- ideal and available db are related by the *completeness statements*

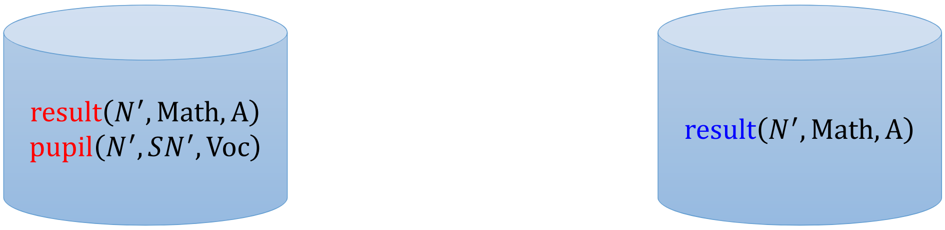
then

- N' is also returned over the available db

Hence: the query is complete

What if N' is not Returned?

Assume, completeness of pupils is not asserted



$\text{result}(N', \text{Math}, \text{A})$
$\text{pupil}(N', SN', \text{Voc})$
$\text{result}(N', \text{Math}, \text{A})$

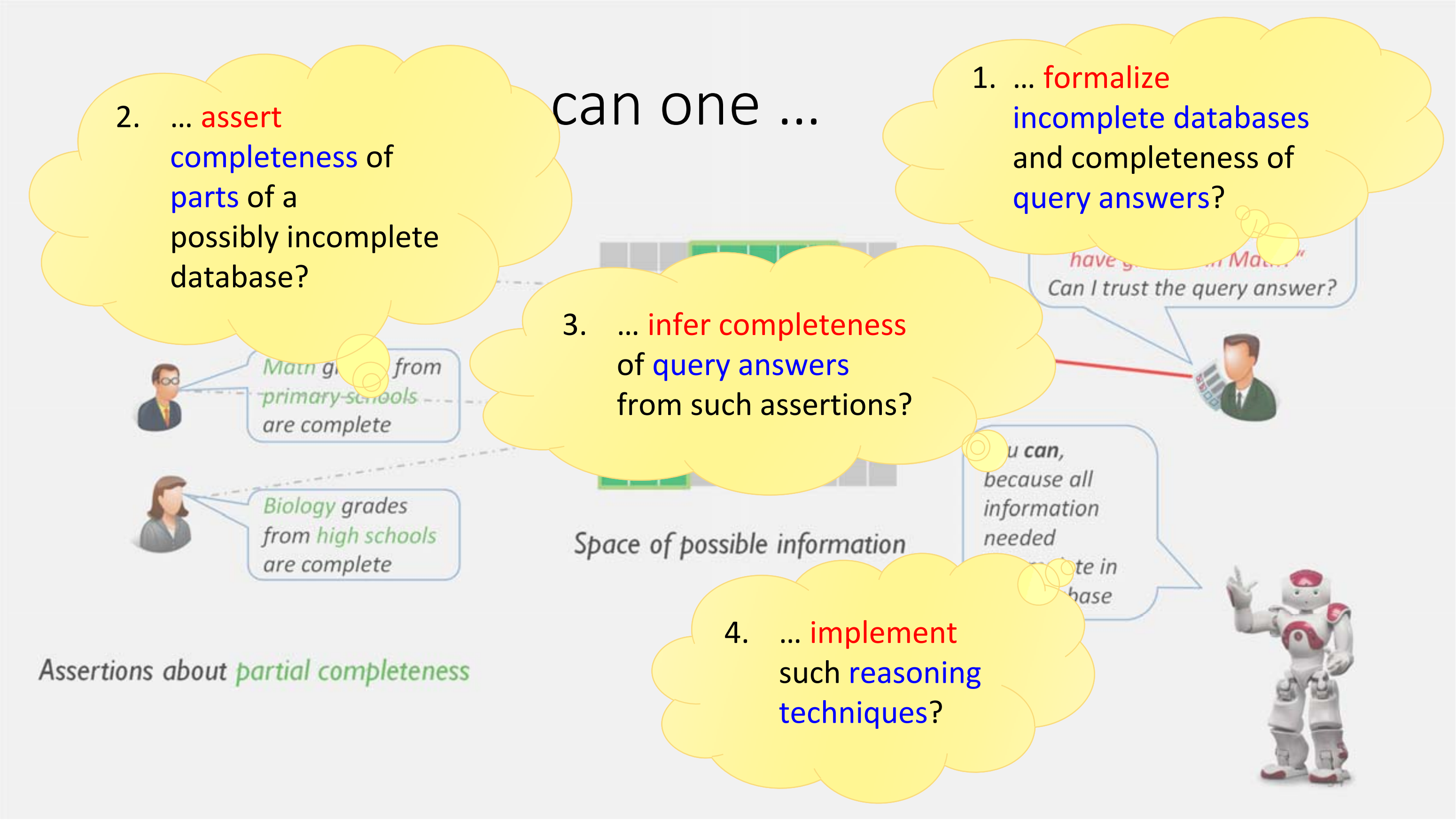
- *“All* result *records of pupils at vocational schools are available“*

$$\text{result}^i(n, s, g),\ \text{pupil}^i(n, sn, \text{Voc}) \rightarrow \text{result}^a(n, s, g)$$

Now,

- (D^i, D^a) satisfies the statement(s)
- but $Q(D^i) \neq Q(D^a)$

Counterexample!
The answer need not be complete …



have g... in Math“
Can I trust the query answer?
Math g... from primary schools are complete
Biology grades from high schools are complete
Space of possible information
...u can, because all information needed ...te in ...base
Assertions about partial completeness

can one …

1. … formalize incomplete databases and completeness of query answers?
2. … assert completeness of parts of a possibly incomplete database?
3. … infer completeness of query answers from such assertions?
4. … implement such reasoning techniques?

Implementation: Logic Programming

- Query ↦ "Ideal" facts

  $\text{result\_i}(n, '\text{Math}', '\text{A}').\quad \text{pupil\_i}(n, sn, '\text{Voc}').$

Run with
- Prolog
- Datalog engine

- Completeness statements ↦ Rules

  $\text{result\_a}(N, S, G) \leftarrow \text{result\_i}(N, S, G), \text{pupil\_i}(N, SN, '\text{Voc}').$

  $\text{pupil\_a}(N, SN, ST) \leftarrow \text{pupil\_i}(N, SN, ST).$

- Completeness check: Test query

  $\text{Q\_test} \leftarrow \text{result\_a}(n, '\text{Math}', '\text{A}'), \text{pupil\_a}(n, SN, '\text{Voc}').$

If Q_test succeeds, the query is complete, otherwise not

Alternative Implementation: CONSTRUCT Queries

To reason about the completeness of conjunctive SPARQL queries:

- BGP of query Q ↦ "ideal" graph G_Q
- Completeness statements ↦ CONSTRUCT queries Q_C
- Completeness check:
  - apply the Q_C to G_Q, resulting in a new graph $\text{G'} = \bigcup Q_C(Q_G)$
  - does $\text{Q}(\text{G'})$ contain the output variables of Q ?

Databases Can Have More Features

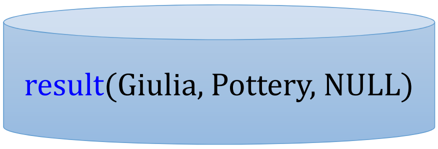
- Integrity Constraints
  - Finite domain constraints: *“School types are only primary, middle, and vocational”*

    $$\mathrm{pupil}[\mathrm{schoolType}] = \{\mathrm{Prim},\ \mathrm{Mid},\ \mathrm{Voc}\}$$

  - Keys: “pupil *records are identified by their* name,
    result *records by* name *and* subject”
  - Foreign keys: *“For every* result *there is a corresponding* pupil *with the same* name”

    $$\mathrm{result}[\mathrm{name}] \subseteq \mathrm{pupil}[\mathrm{name}]$$

- Null values



result(Giulia, Pottery, NULL)

Reasoning with Finite-Domain Constraints

Completeness Statements

- “**pupils** from **primary** schools are complete”
- “**pupils** from **middle** schools are complete”
- “**pupils** from **vocational** schools are complete”

Query: $Q(n) \leftarrow \mathrm{pupil}(n, sn, st)$ “Give me the names of all pupils”

Q complete? No! (There could be other types of schools)

Suppose $\mathrm{pupil}[\mathrm{schoolType}] = \{\mathrm{Prim},\ \mathrm{Mid},\ \mathrm{Voc}\}$ holds

Q complete? Yes!

Implement Reasoning With Finite Domains

$Q(n) \leftarrow \text{pupil}(n, sn, st)$ complete?

If $\text{pupil[schoolType]} = \{\text{Prim, Mid, Voc}\}$ holds,

we make a case analysis with three versions of $D_Q^i = \{\, \text{pupil}(N', SN', ST')\,\}$ substituting each possible school type for ST'

$$[ST' \mapsto \text{Prim}]D_Q^i = \{\, \text{pupil}(N', SN', \text{Prim})\,\}$$

$$[ST' \mapsto \text{Mid}]D_Q^i = \{\, \text{pupil}(N', SN', \text{Mid})\,\}$$

$$[ST' \mapsto \text{Voc}]D_Q^i = \{\, \text{pupil}(N', SN', \text{Voc})\,\}$$

Can be encoded into disjunctive logic programming

Then move on as before …

Reasoning With Foreign Keys

“For every result record, there is a corresponding pupil record with the same name”

$$\text{result[name]} \subseteq \text{pupil[name]}$$

can have two meanings

1. holds for D^i (= holds in the world)

2. holds for D^i and D^a (= the FK is enforced in our db)

Foreign Key Holding over D^i

Query Q: *"Give me all* result *records "*

Suppose *"*result *records are complete for pupils from **primary** schools"*
*"*result *records are complete for pupils from **middle** schools"*
*"*result *records are complete for pupils from **vocational** schools"*

Suppose $\text{pupil}[\text{schoolType}] = \{\text{Prim, Mid, Voc}\}$ holds

Q complete? No! (There could be result records without pupils …)

Suppose $\text{result}[\text{name}] \subseteq \text{pupil}[\text{name}]$ holds over D^i

Q complete? Yes! (Every result record in D^i belongs to some pupil,
which goes to a primary, middle, or vocational school.
result records are complete for each of the three cases.)

Foreign Key Holding over D^i and D^a

Query Q: "Give me all results for the pupils of primary schools"

Suppose "results are complete"

Q complete? No! (We have the results,
but we may not have the pupil records with the school type)

Suppose $\text{result}[\text{name}] \subseteq \text{pupil}[\text{name}]$ holds over D^i and D^a

Q complete? Yes! (Every result in D^i belongs to some pupil,
since the FK holds over D^i ,
and that pupil is in D^a ,
since FK holds over D^a)

asoning With Foreign Keys

For every result^i,
generate
a corresponding pupil^i,
inventing school name,
and school type
(works only if FKs are
(weakly) acyclic)

ings of

l there is a corresponding pupil record with the sam name”

For every result^a,
copy
the corresponding pupil^i
to a pupil^a

1. FK holds for D^i (= holds in the world):

$$\text{result}^i(n, s, g) \rightarrow \text{pupil}^i\left(n, f_{\text{pupil,sname}}(n), f_{\text{pupil,stype}}(n)\right)$$

2. FK holds also for D^a (= the FK is enforced in our db):

$$\text{result}^a(n, s, g), \text{pupil}^i(n, sn, st) \rightarrow \text{pupil}^a(n, sn, st)$$

Implement Reasoning with Finite Domains and Foreign Keys

Finite domains alone:

- Instantiate constrained query variables/generic constants in all possible ways
- Then apply rules as before

Foreign Keys

- Generate new records, inventing new generic constants

$$\text{result}^i(n, s, g) \rightarrow \text{pupil}^i\left(n, f_{\text{pupil,sname}}(n), f_{\text{pupil,stype}}(n)\right)$$

The new constants need to be instantiated as well!

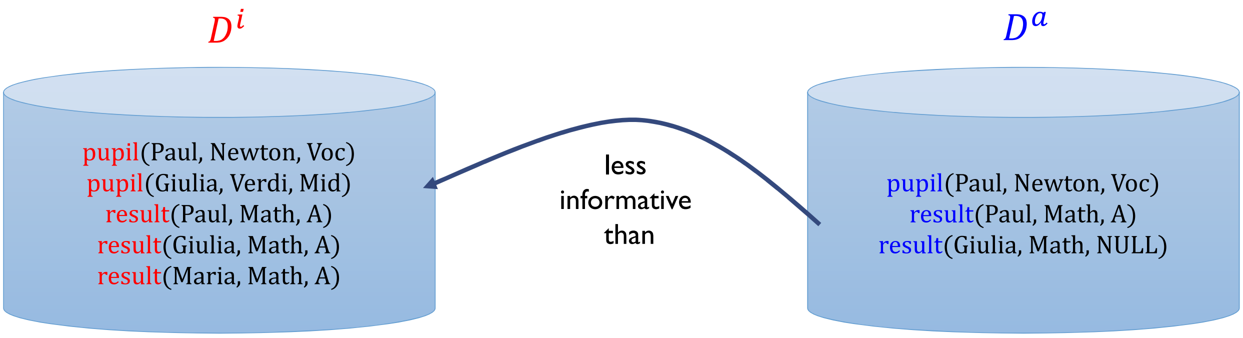
Tricky! Needs disjunctive rules, only possible with Answer Set Programming

(see [CIKM 15])

DBs w/ Nulls: Generalize Completeness Rules

“The available database contains all subjects taken by pupils at vocational schools, (but not necessarily the grades)“

$$\text{result}^i(n,s,g),\ \text{pupil}^i(n,sn,\text{Voc}) \rightarrow \exists g'\ \text{result}^a(n,s,g')$$



D^i

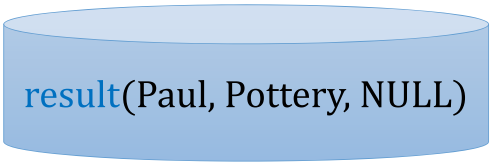
pupil(Paul, Newton, Voc)
pupil(Giulia, Verdi, Mid)
result(Paul, Math, A)
result(Giulia, Math, A)
result(Maria, Math, A)

less informative than

D^a

pupil(Paul, Newton, Voc)
result(Paul, Math, A)
result(Giulia, Math, NULL)

What is the Meaning of NULL?



result(Paul, Pottery, NULL)

- Paul received a grade, but the grade was not recorded? Unknown value
- No grades were given in the Pottery course? Non-existing value
- It is unknown, which of the two is the case? Ambiguous NULL

⟹ NULLs may indicate *incomplete information*, but *need not*

⟹ Usage of NULLs is *ambiguous*

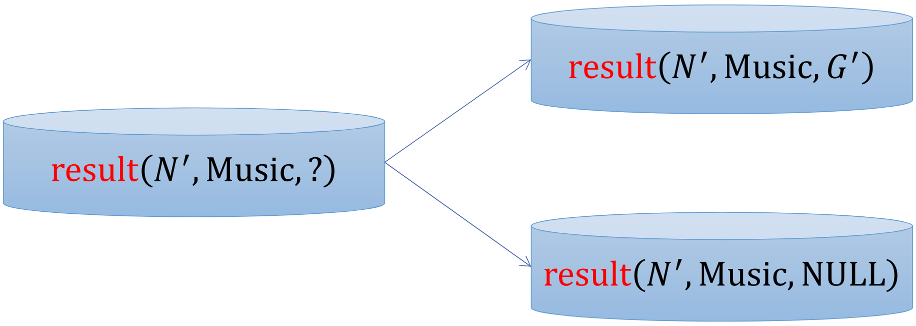
Reasoning about NULL Requires Case Analysis

Query: *"Which pupils took Music?"*

$$Q_{Music}(n) \leftarrow \text{result}(n, \text{Music}, g)$$

...then apply rules

Create prototypical D^i from Q_{Music}


$\text{result}(N', \text{Music}, ?)$
$\text{result}(N', \text{Music}, G')$
$\text{result}(N', \text{Music}, \text{NULL})$

Pupil N' may have a Music grade

Null vs. non-Null distinction increases worst-case complexity

(grade not applicable)

Data-Aware Reasoning

Schema: $\text{result}(\text{name}, \text{subject}, \text{grade})$
$\text{pupil}(\text{name}, \text{schoolName})$
$\text{school}(\text{schoolName}, \text{schoolType})$

Statements: Pupils *are complete for vocational schools*
Results *are complete for* Paul *and* Mary

$$Q(n, s, g) \leftarrow \text{result}(n, s, g),\ \text{pupil}(n, \text{Newton})$$

Data-agnostic reasoning: Q is incomplete

Let's look at the data!

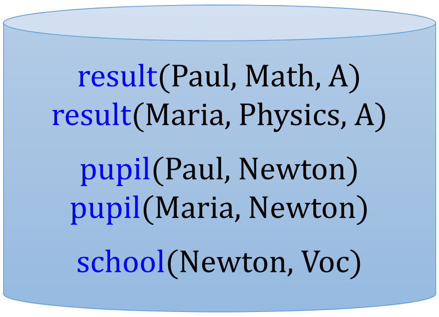
1. Match statements **against** both, **query and data**

Reasoning

Statements: Pupils *are complete for vocational schools*

Results *are complete for* Paul *and* Mary

$Q(n,s,g) \leftarrow \text{result}(n,s,g),\ \text{pupil}(n,\text{Newton})$



result(Paul, Math, A)
result(Maria, Physics, A)

pupil(Paul, Newton)
pupil(Maria, Newton)

school(Newton, Voc)

1. Pupils *are complete for* Newton (because Newton is vocational)
2. Paul *and* Maria *are **all*** pupils *from* Newton *school* (because pupils are complete for Newton)
3. *Check "*results of Paul*" and "results of* Maria*"*
4. Results *are complete for* Paul *and* Maria, *therefore for all* pupils *from* Newton

⇒ Q is complete

3. **Evaluate fragment** and **instantiate query** with the results

2. Identify **maximal complete fragment**

4. Original query is complete **if fragments are complete**

More Reasoning Services

- Aggregate queries:

"How many pupils have grade A in Math?"

Answer is correct if non-aggregate query *"pupils with A in Math"* is complete

⇒ correctness of aggregate queries

- Queries with negation:

"Which pupils never had grade F?"

Answer is sound for a pupil for whom we have complete results

⇒ soundness of queries with negation

More Reasoning Services

- Simple Ontology Languages (RDFS):

  “Which pupils attend a vocational school ?”

  “Vocational and professional schools are subclasses of each other”

Query is complete if *“pupils of professional schools”* are complete

⇒ completeness wrt RDF Schema Axioms

- SPARQL queries with OPTIONAL:

  “What are the names of pupils taking pottery, and optionally their grades?”

⇒ Check completeness of

“What are the names of pupils taking pottery?”

and

“What are the names and grades of pupils taking pottery and having a grade?”

More Reasoning Services

- Suppose query *“All Math results of pupils”* is incomplete:

  Are specializations of this query complete?

  “Data are complete for Math results from primary and vocational schools.”

⇒ query specialization

- “Movies of Tarantino are complete until 2016”

⇒ time-aware completeness reasoning

Overhead of Completeness Reasoning

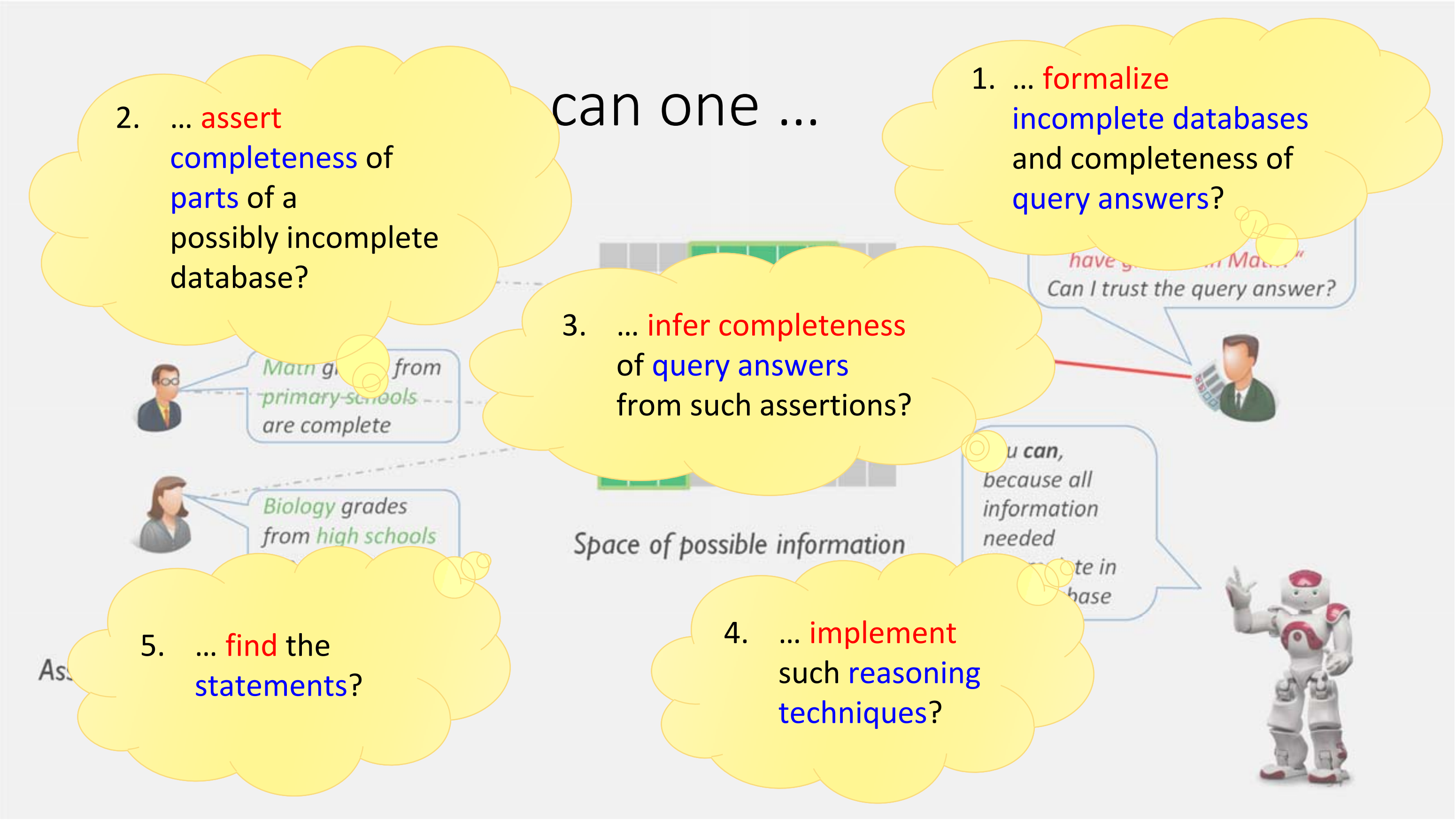
Optimize by indexing the completeness statements: get only relevant ones

Experiments with query logs from Web knowledge bases
(statements synthesized from queries):

Data Source	Number of Queries	Number of Completeness Statements	Query Evaluation Time	Completeness Reasoning Time	Overhead
DBpedia	334,000	331,000	18 ms	0.08 ms	0.44%
SWDF	108,000	44,000	36 ms	0.12 ms	0.33%
LGD	22,00,	21,000	8 ms	0.05 ms	0.60%

Data-aware reasoning lead to overheads of ~400 ms

[TWEB 18, SWJ 20]



can one ...

1. … formalize incomplete databases and completeness of query answers?

2. … assert completeness of parts of a possibly incomplete database?

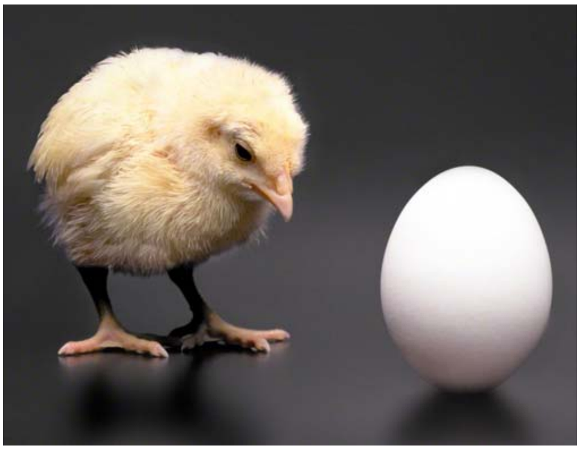
3. … infer completeness of query answers from such assertions?

4. … implement such reasoning techniques?

5. … find the statements?

have g... in Math"
Can I trust the query answer?
Math g... from primary schools are complete
Biology grades from high schools
Space of possible information
...u can, because all information needed ...te in ...base
As...

Where do Statements Come From?



- To benefit from completion reasoning, one needs completeness statements
- People will only provide completeness statements, if they experience benefits

Image Source: https://www.chronicle.com/blogs/linguafranca/2017/05/30/clear-skies-acts-abounding/

Bootstrap: Ideas for Web Knowledge Bases

Completeness Rule Mining:

Learn general completeness statements from individual statements:

$$\text{hockeyPlayer}(x) \rightarrow \text{Incomplete}(x, \text{hasChild})$$
$$\text{scientist}(x),\ \text{hasWonNobelPrize}(x) \rightarrow \text{Complete}(x, \text{graduatedFrom})$$

[Galarraga et al., WSDM 2017]

Cardinality Extraction:

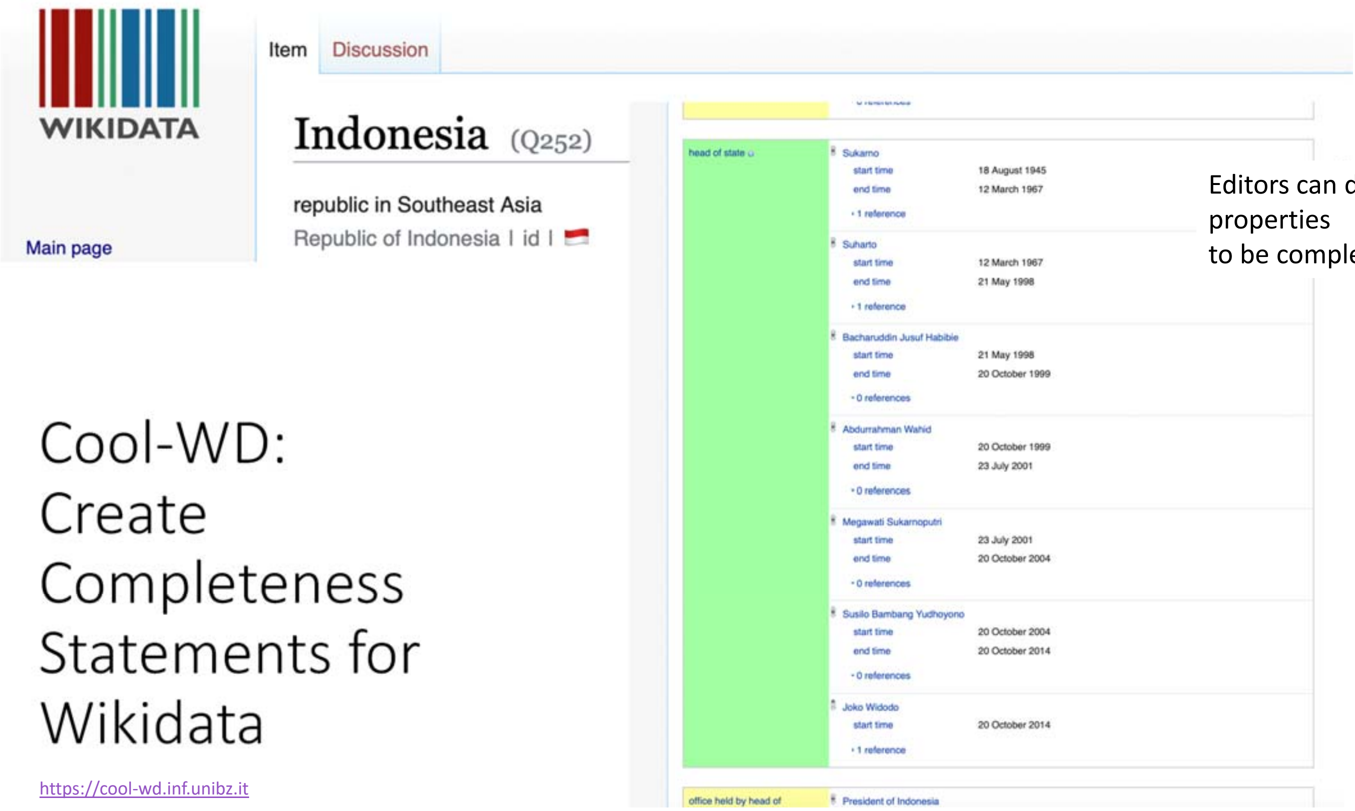
Extract count information from text and match with database

Wikipedia: *“Donald Trump has five children”*

Wikidata: contains 5 children of Trump

⇒ Wikidata is complete for children of Trump

[Mirza et al., ACL 2017,
Mirza et al. ISWC 2018]



Item | Discussion

Main page

Indonesia (Q252)

republic in Southeast Asia

Republic of Indonesia | id |

• 0 references

head of state

Sukarno
start time 18 August 1945
end time 12 March 1967
• 1 reference

Suharto
start time 12 March 1967
end time 21 May 1998
• 1 reference

Bacharuddin Jusuf Habibie
start time 21 May 1998
end time 20 October 1999
• 0 references

Abdurrahman Wahid
start time 20 October 1999
end time 23 July 2001
• 0 references

Megawati Sukarnoputri
start time 23 July 2001
end time 20 October 2004
• 0 references

Susilo Bambang Yudhoyono
start time 20 October 2004
end time 20 October 2014
• 0 references

Joko Widodo
start time 20 October 2014
• 1 reference

office held by head of | President of Indonesia

Cool-WD: Create Completeness Statements for Wikidata

https://cool-wd.inf.unibz.it

Editors can declare
properties
to be complete

Open Problems

Find more efficient implementation techniques
(e.g., for queries with $<, \leq$, or disjunctive information)

Lift completeness reasoning to quantitative completeness analysis

- Now: we are 100% complete (or not)
- Better: we are 50% complete with a probability/confidence of 80%

How collect completeness statements?

Can we generalize the approach to query-aware data quality reasoning?

Conclusions

- Data analysis usually relies on the assumption that data is complete, which often is not the case
- Techniques exist to annotate query results with completeness information if completeness statements are available
- Not only data matters, also (quality) metadata!

Thank you!