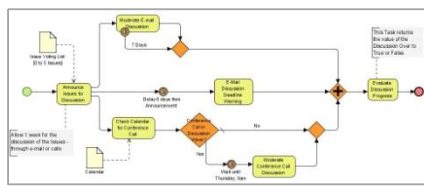
Database Completeness Reasoning and Completeness Verification over Processes

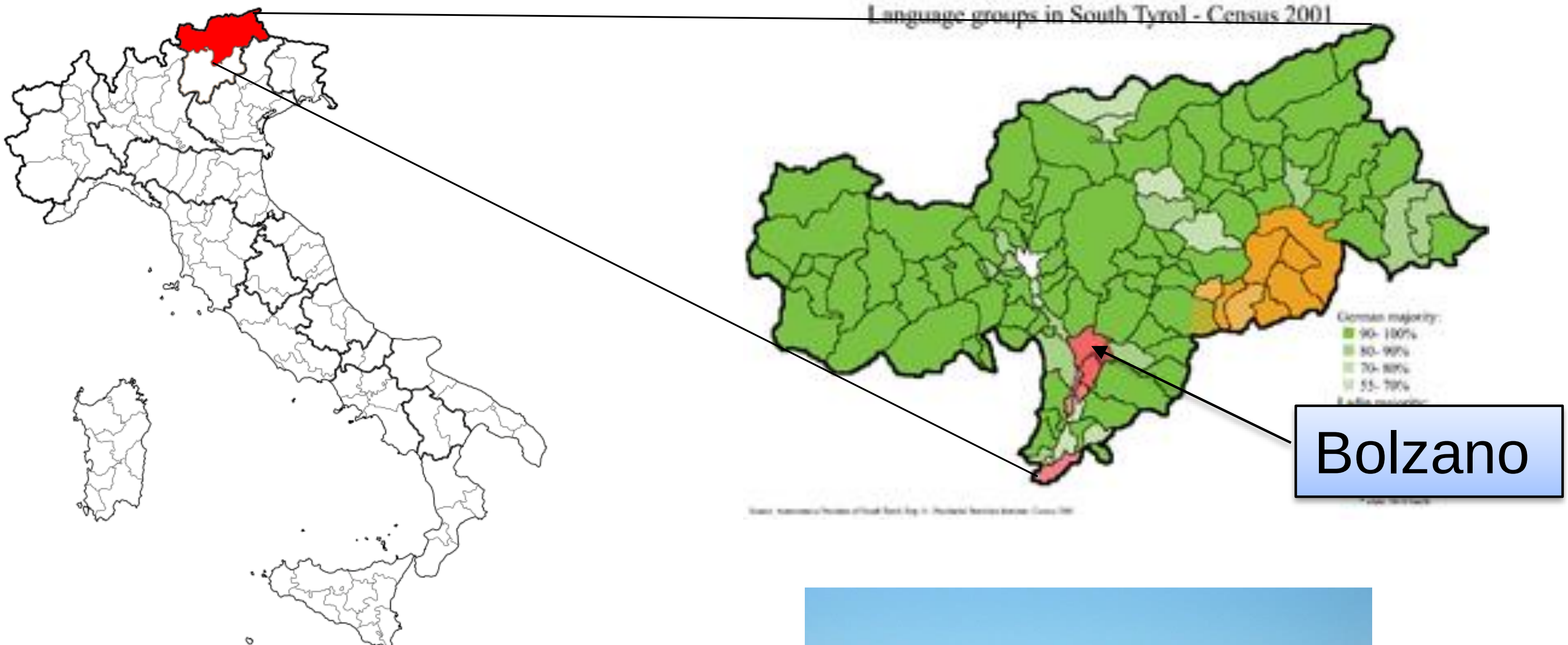


Simon Razniewski

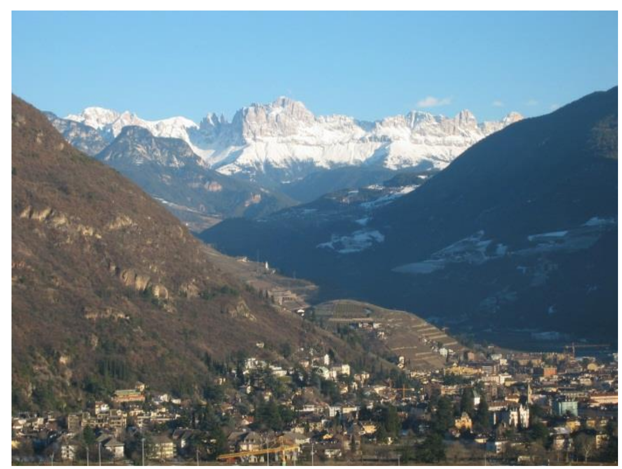
Joint work with Marco Montali and Werner Nutt

Free University of Bozen-Bolzano

Free University of Bozen-Bolzano, Italy



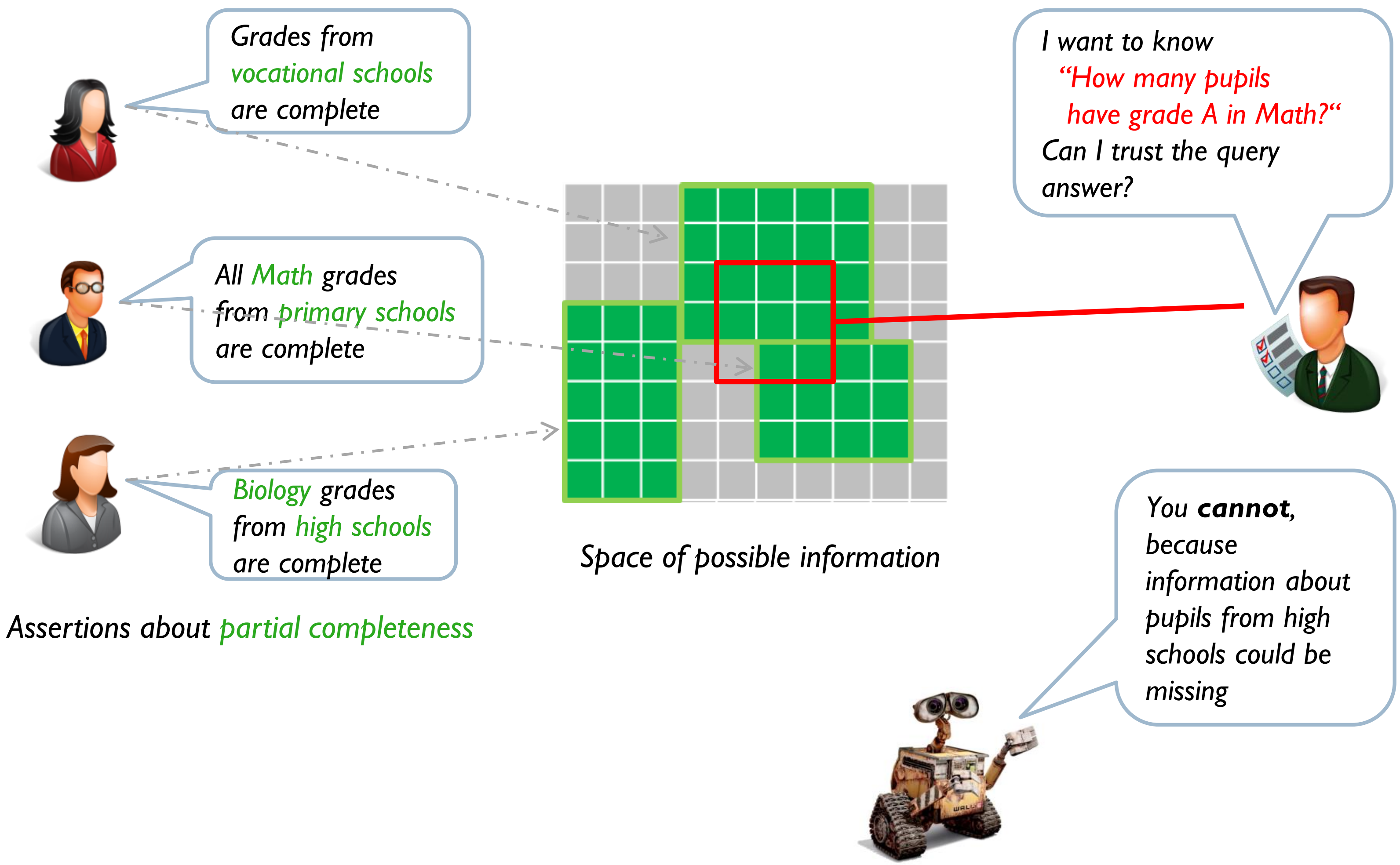
- founded in 1997
- 3500 students



Background: School Management in the Province of Bolzano

- Province has central database about pupils, teachers, etc.
- Would like to answer statistical queries
- Problem: Data often entered with delays or not at all
- Thus, administrators would like to know
  whether a query is currently reliable (complete)

Completeness Problem



Relational Database Setting

- Completeness reasoning can be reduced to query containment [VLDB 2011]
- Null values increase complexity due to ambiguity (Unknown values versus non-existing values) [CIKM 2012]

Linked Data/Semantic Web

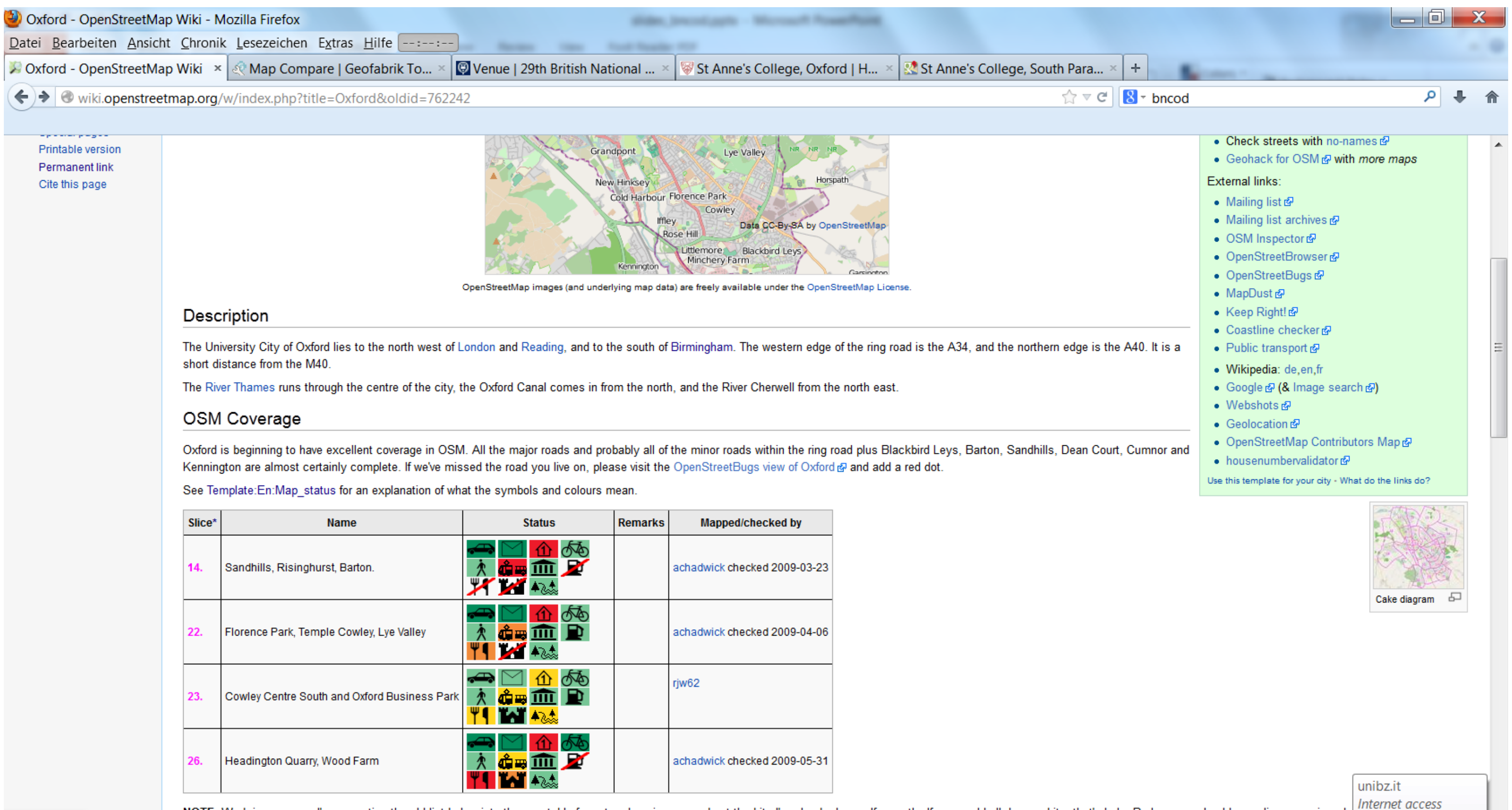
- Generally, the semantic web is assumed to be open-world
- Completeness statements can allow closed-world reasoning [ISWC 2013]

Geographic Data

- Spatial predicates allow to conclude query completeness in certain areas

  “All hotels near a train station and a museum”

- Completeness statements exist in OpenStreetMap [BNCOD 2013]



Verifying Completeness over Processes

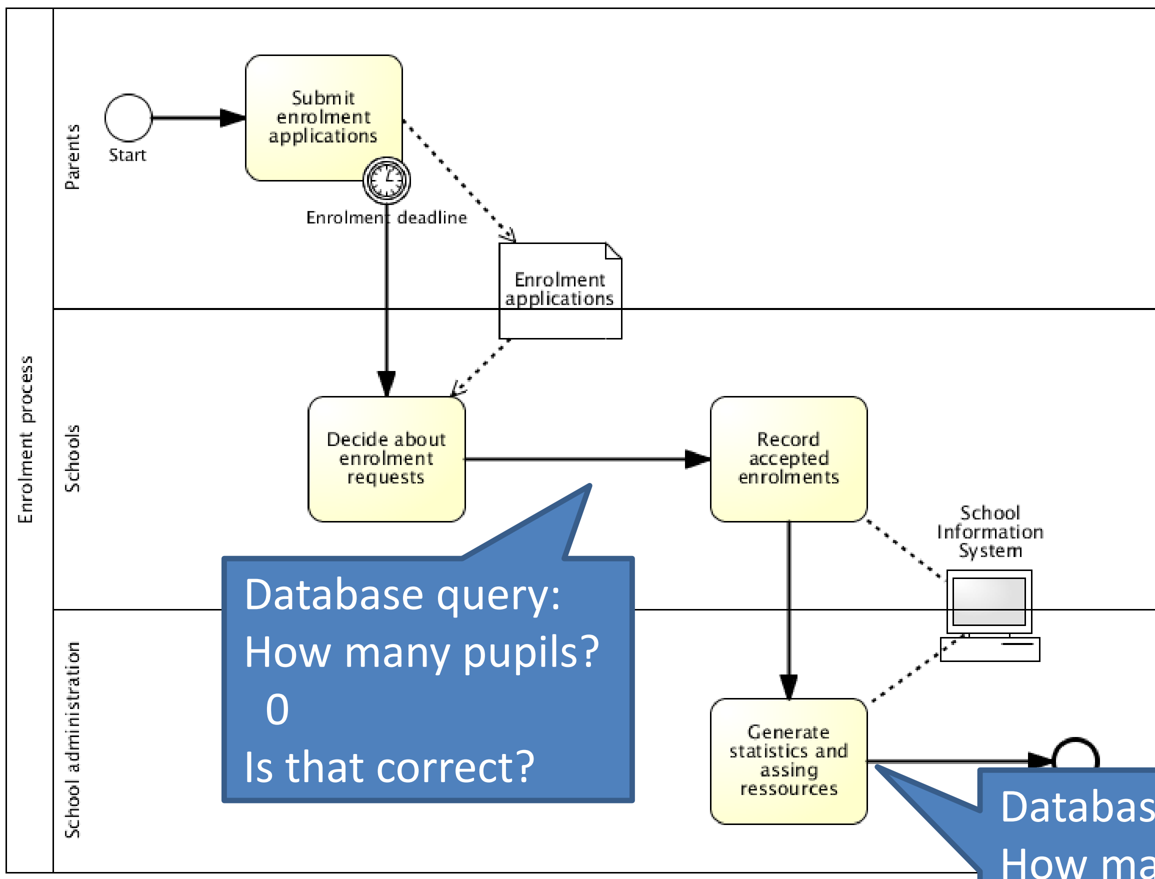
- Data often created following processes
- Many processes are executed only partially formal
  (pen&paper, email, phone, …)

➔ Valid information may be stored in databases
with delays

➔ Database content is of questionable completeness

N.B.: Completeness and timeliness are equivalent problems, if database is monotonic

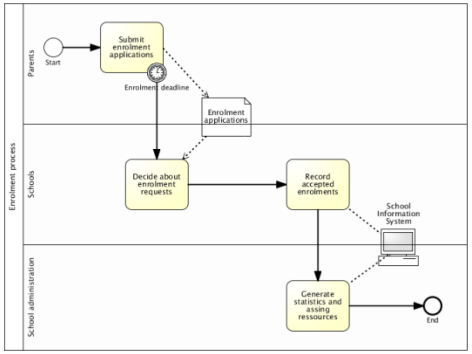
Enrolment Process in a School



Database query:
How many pupils?
0
Is that correct?

Database query:
How many pupils?
137
Is that correct?

Observation



- At some points, new facts in the real world have not yet been stored

  → queries may give wrong answers
- At other points, all facts that hold in the real world have been stored

  → queries give correct answers

Formalization: Two Databases

Conceptually, there are
- the state of the information system
- the state of the real world

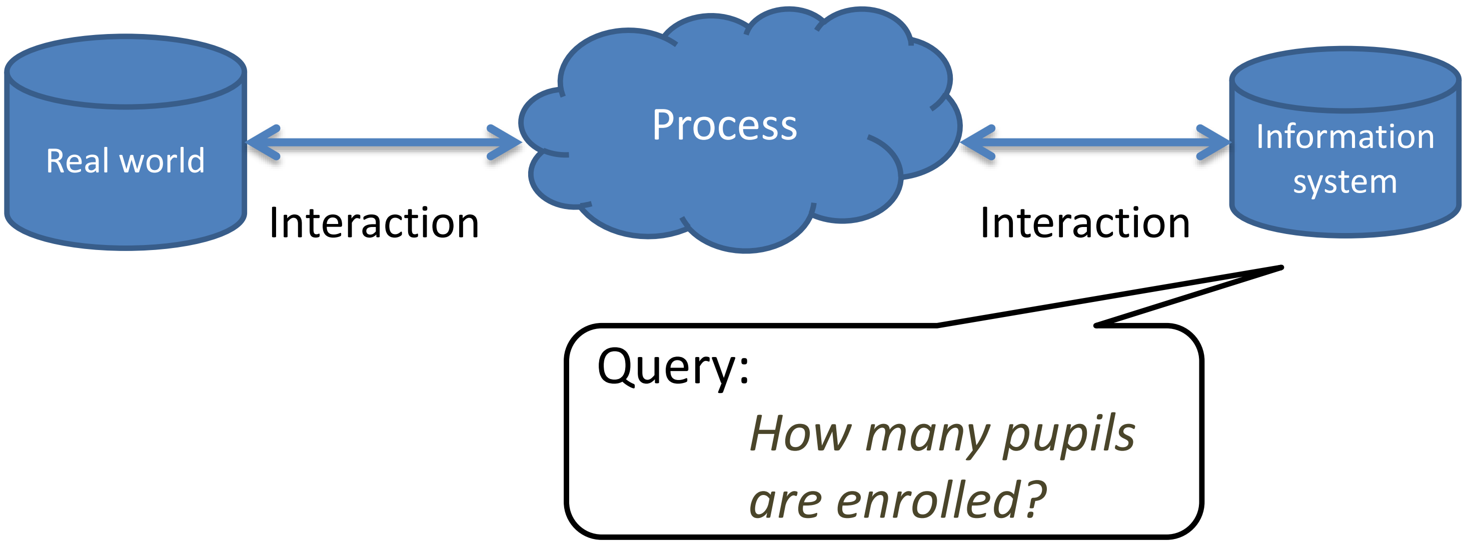
We model
- each state as a database
- the process interacting with both

Two databases: Example

- Deciding about enrolments:
  - read from and write into real-world world database

- Recording accepted enrolments into the information system:
  - read from real-world database
  - write into information system database

Completeness Problem



Is the information system up-to-date wrt. the state of the real world?

Research Questions

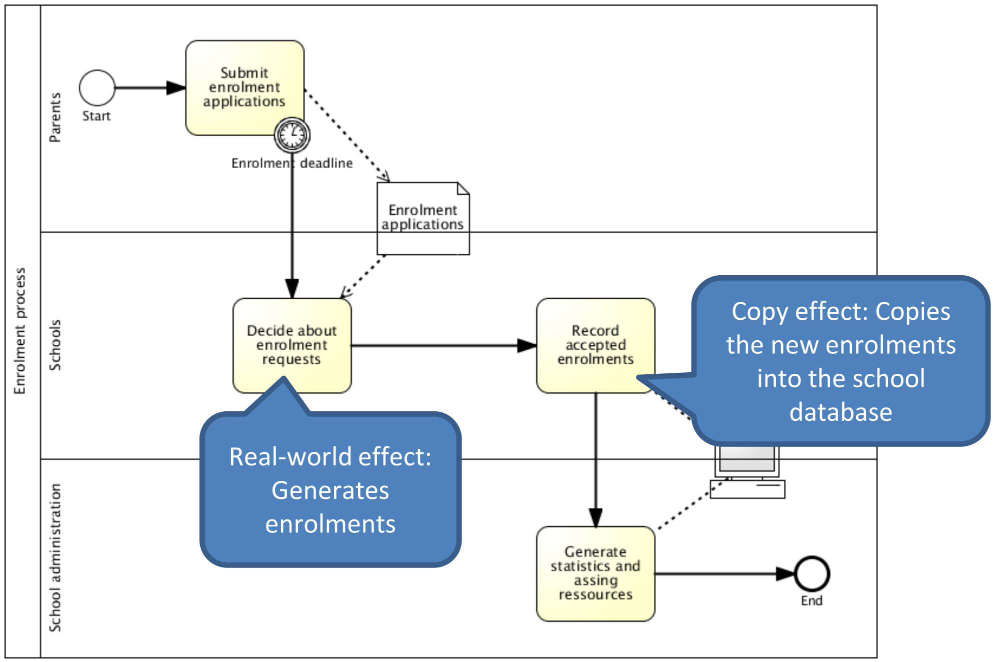
- How can we express which data a process generates?
- What does completeness mean?
- How can we find out whether a query is complete?

Model: Quality-aware Transition Systems (QATS)

- Goal: Technique applicable to different modeling languages
- Low-level formalism for process instances: Transition systems
  - *Petri nets can be encoded using their reachability graphs (possibly exponential encoding due to parallelism)*
- Actions in a QATS can be labeled with two kinds of effects:
  - Real-world effects: allow to create new data in the real world
  - Copy effects: store information that holds in the real world into the information system

QATS are data-monotonic

Example Revisited



Real-world effect: $\text{pupil}^{rw}(n, s) \leftsquigarrow \text{request}^{rw}(n, s)$

Copy effect: $\text{pupil}^{rw}(n, s) \rightarrow \text{pupil}^{is}(n, s)$

Real-world and Copy Effects

Real-world effect: $\text{pupil}^{rw}(n, s) \leftsquigarrow \text{request}^{rw}(n, s)$

Copy effect: $\text{pupil}^{rw}(n, s) \rightarrow \text{pupil}^{is}(n, s)$

In general, a real-world effect has the form

$$R^{rw}(X,Y) \leftsquigarrow G^{rw}(X,Z)$$

where G is a condition, X are bound variables and Y are unbound variables.

It allows to introduce new facts $R^{rw}(X,Y)$, if $G^{rw}(X,Z)$ holds for some Z

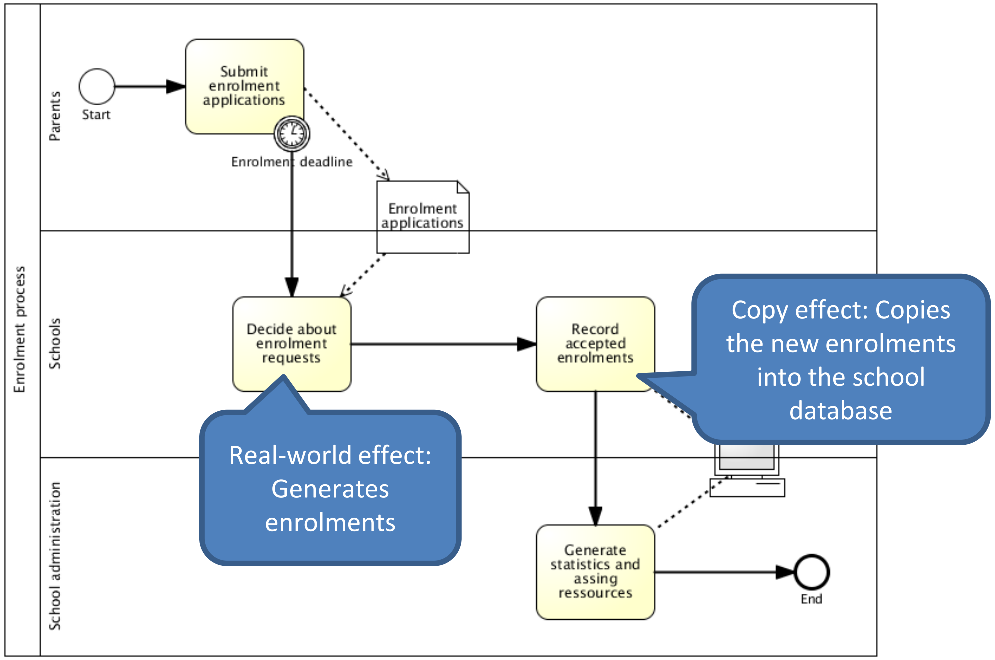
A copy effect has the form

$$R^{rw}(X), G^{rw}(X,Y) \rightarrow R^{is}(X)$$

It copies all facts in R^{rw} that satisfy G^{rw} into R^{is}

Real-world effects are nondeterministic, copy effects are deterministic

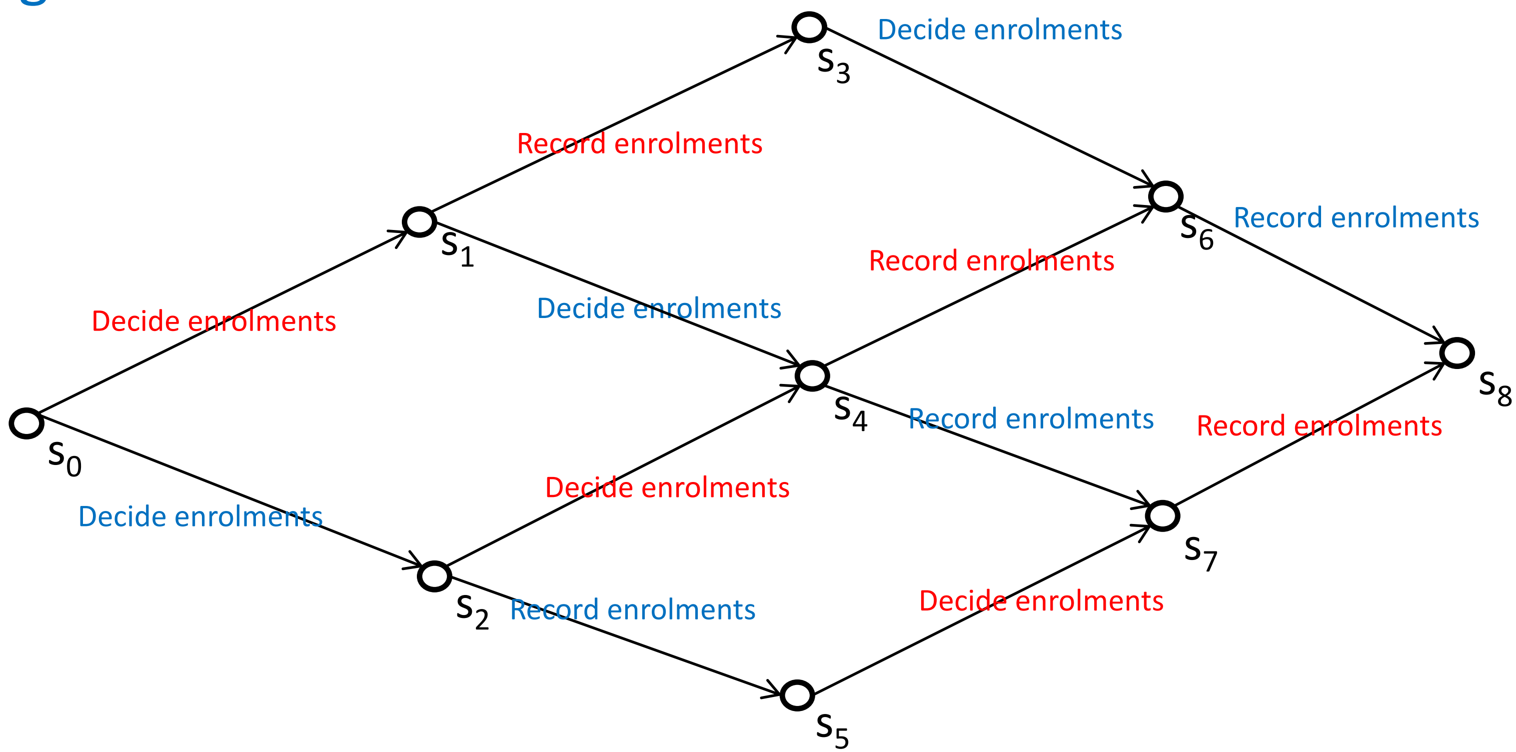
Transition Systems for Process Instances



Transition Systems for Process Instances

Two concurrent process instances:

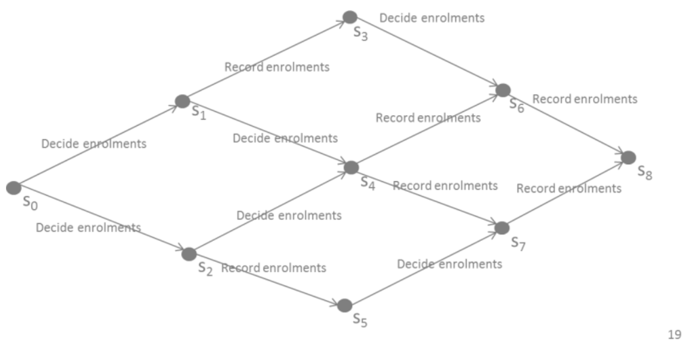
- Middle School A
- High School B



Completeness Verification

Given

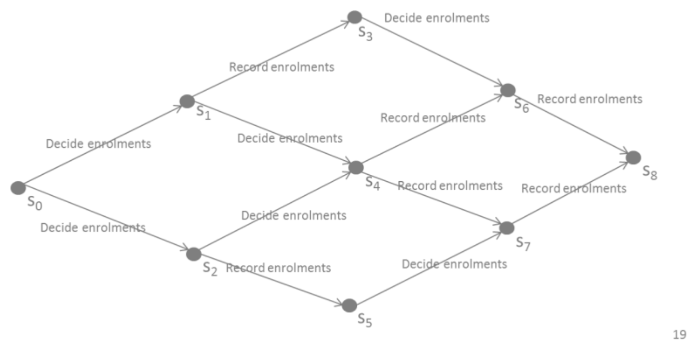
- Process description
- State S
- Query Q



Question

Is it safe to pose the query Q in state S against the information system database?

Completeness Verification (2)



A state S of a QATS satisfies completeness for a query Q,

if

for all paths leading to S,

for all process-compliant database developments $((D^{rw}_0, D^{is}_0), .., (D^{rw}_n, D^{is}_n))$,

$$\mathbf{Q(D_n^{is}) = Q(D_n^{rw})}$$

Meaning: In state S the information system gives the same result as holds in the real world

Decision Problem

Compliance

When does a development $((D^{rw}_0, D^{is}_0), .., (D^{rw}_n, D^{is}_n))$ comply to a sequence of real-world and copy effects?

Compliance to Real-world Effects

Real-world database

request(John,HS)
request(Mary,HS)

Real-world effect: $\text{pupil}^{rw}(n, HS) \leftsquigarrow \text{request}^{rw}(n, HS)$

Because "$\leftsquigarrow$" is nondeterministic

Possible successive real-world databases:

request(John,HS)
request(Mary,HS)

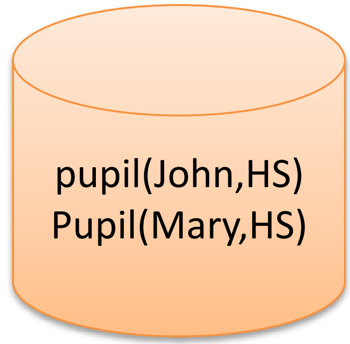
request(John,HS)
request(Mary,HS)
pupil(John,HS)

request(John,HS)
request(Mary,HS)
Pupil(Mary,HS)

request(John,HS)
request(Mary,HS)
pupil(John,HS)
Pupil(Mary,HS)

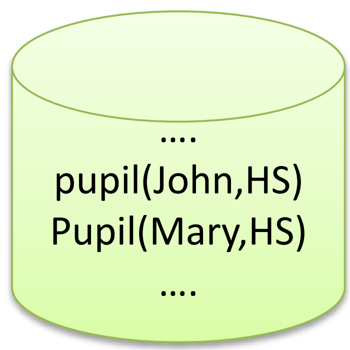
Compliance to Copy Effects

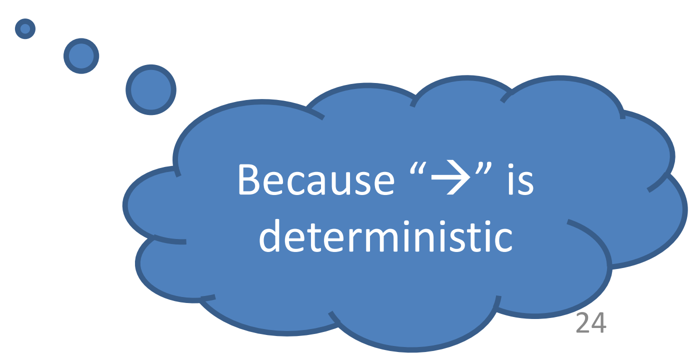
Real-world database



Copy effect: $\mathrm{pupil}^{\mathrm{rw}}(\mathrm{n}, \mathrm{HS}) \rightarrow \mathrm{pupil}^{\mathrm{is}}(\mathrm{n}, \mathrm{HS})$

Resulting information system database





Results – Completeness over Paths

- A real-world effect is risky wrt. a query,
  if it has the potential to change the query result

  Adding pupils in class 1A is risky wrt. a query for all pupils,
  but not wrt. a query for all pupils in level 2

- Copy effects can repair a risky effect,
  if they copy all data that has the potential to change the query result

  Copying all pupils in level 1 into the information system
  repairs the risky effect.

- Result: A query is complete over all developments of a path,
  if all risky effects in the path are repaired

Theorem: Repair checking can be reduced to query containment

  – Query containment for conjunctive queries (SELECT … FROM … WHERE …) has been well studied in database research

Results – Completeness in States

- Completeness holds in a state, if it holds for all paths that lead to that state
- A priori, infinitely many paths (due to cycles)

Theorem: Repeated actions can be ignored

  - Thus, only finitely many paths to consider
  - Still, number of paths can be exponential wrt. the QATS

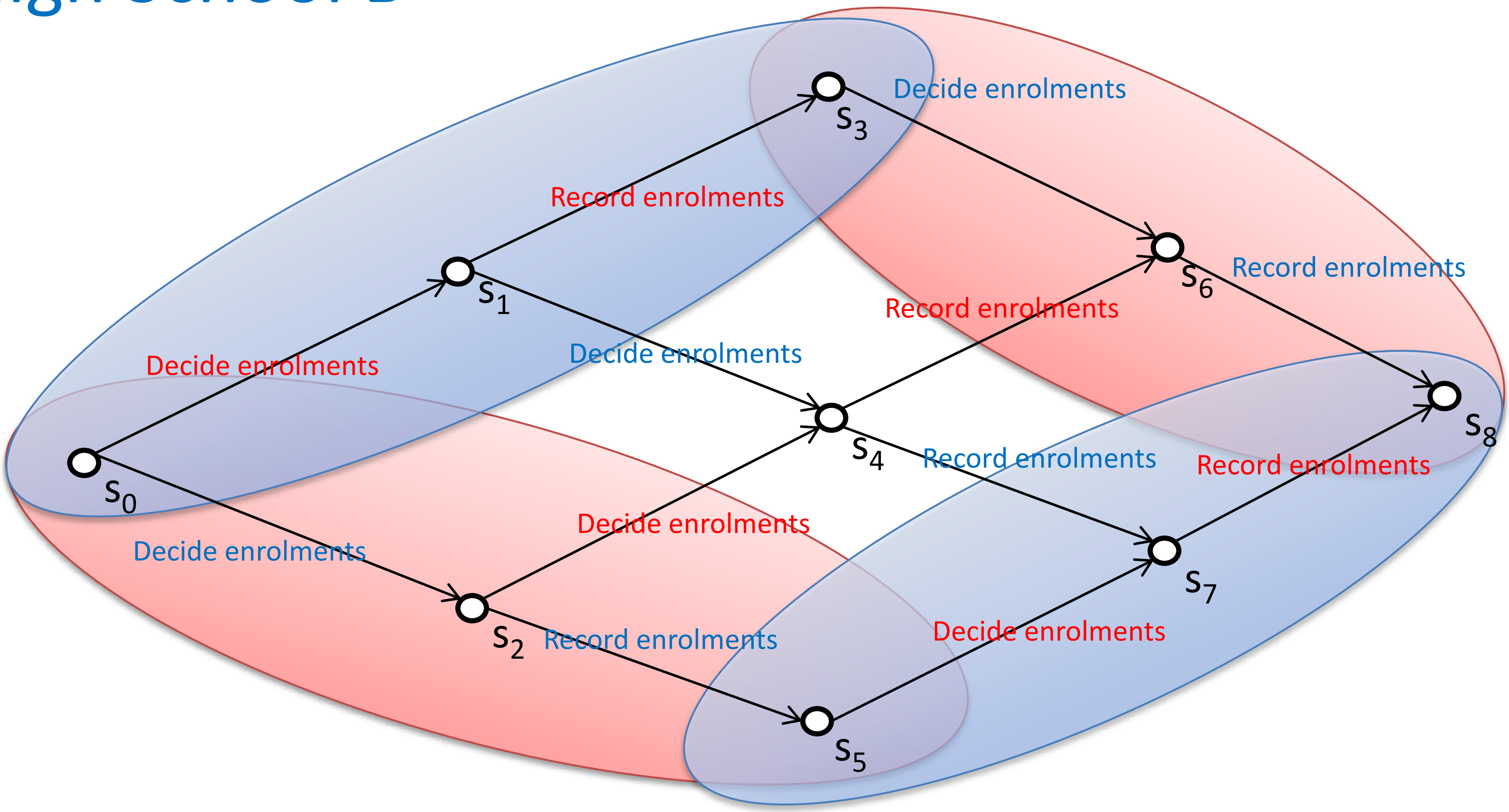
Example Revisited

Middle School A

High School B

How many middle school pupils?

How many high school pupils?



Complexity

Query and effect language	Complexity of completeness checking for a path	Complexity of completeness checking for a state
Arbitrary conjunctive queries (CQ)	Π^P_2-complete	Π^P_2-complete
CQs without <, ≤	NP-complete	In Π^P_2
CQs without selfjoins	coNP-complete	coNP-complete
CQs without selfjoins and without <, ≤	PTIME	in coNP

Applications

- Annotation of statistics and KPI with completeness information (see next slide)
- Process mining (trace analysis) - to validate whether queries over traces return the real state of the process
- Auditing – to verify whether the information about the real-world is properly stored

Possible Use: Statistical Reports

School Report

	Total	Change
Pupils in primary schools:	548	+2.3%
Pupils in middle schools:	390	-17.1%
Pupils in high schools:	242	+1.4%
Pupils taking English:	957	-0.8%
Pupils taking French:	685	+3.7%
Pupils taking Chinese:	52	-23.8%

…….

Data from the Da Vinci School and the Gherdena School was not submitted

The Hofer School did not enter its language course attendance yet

Conclusion

- Introduced the problem of query completeness due to delays between real-world events and their recording in a database
- Modelling of the problem using quality-aware transition systems that interact both with the real world and with an information system
- Showed how to verify query completeness over such models
- Future work: Demo for a high-level process language (BPMN or YAWL)

Thank you!

Questions?

Acknowledgment

This research has been supported by the project "MAGIC", funded by the Province of Bozen-Bolzano