Formal Verification of Data Provenance Records

Szymon Klarman[1], Stefan Schlobach[1] and Luciano Serafini[2]

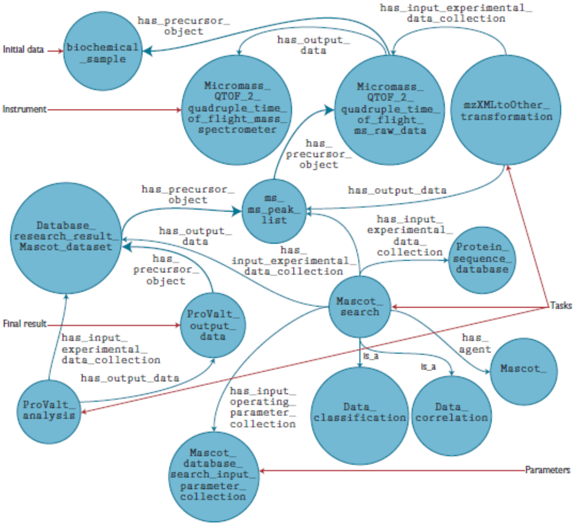
[1] Knowledge Representation & Reasoning Group
VU University Amsterdam

[2] Data & Knowledge Management Group
FBK Trento

November 14, 2012

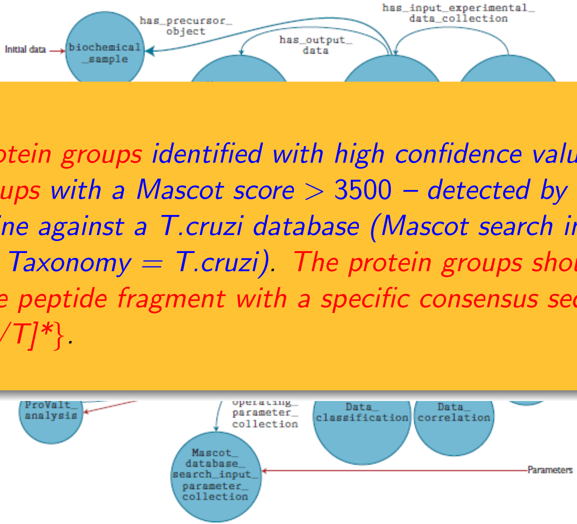
The 11th International Semantic Web Conference (ISWC-12)

Problem: reasoning over data provenance


Initial data
biochemical_sample
has_precursor_object
has_output_data
has_input_experimental_data_collection
Instrument
Micromass_QTOF_2_quadruple_time_of_flight_mass_spectrometer
Micromass_QTOF_2_quadruple_time_of_flight_ms_raw_data
mzXMLtoOther_transformation
has_precursor_object
has_precursor_object
ms_ms_peak_list
has_output_data
Database_research_result_Mascot_dataset
has_output_data
has_input_experimental_data_collection
has_precursor_object
has_input_experimental_data_collection
Protein_sequence_database
Mascot_search
Tasks
Final result
ProValt_output_data
has_input_experimental_data_collection
has_output_data
has_agent
Mascot_
is_a
is_a
ProValt_analysis
has_input_operating_parameter_collection
Data_classification
Data_correlation
Mascot_database_search_input_parameter_collection
Parameters

Sahoo, S., Sheth, A., Henson, C. **Semantic Provenance for eScience: Managing the Deluge of Scientific Data.** In *IEEE Internet Computing 12(4)*, 2008.

Problem: reasoning over data provenance



List the protein groups identified with high confidence value – that is, protein groups with a Mascot score > 3500 *– detected by the Mascot search engine against a T.cruzi database (Mascot search input parameter, Taxonomy* $=$ *T.cruzi). The protein groups should contain at least one peptide fragment with a specific consensus sequence of* {**N [P] [S/T]**}.

Sahoo, S., Sheth, A., Henson, C. **Semantic Provenance for eScience: Managing the Deluge of Scientific Data.** In *IEEE Internet Computing 12(4)*, 2008.

Overview

Problem:

How to *formally verify* data provenance records? This involves:

- adequately representing provenance records,
- defining a language for expressing relevant properties,
- ensuring that reasoning is "manageable".

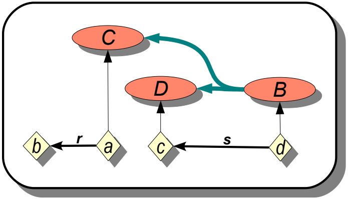
Approach:

- Provenance records resemble *transition systems*, which are typically verified using various *dynamic logics*.
- We develop *Provenance Specification Logic* for verifying and querying data provenance records, based on Propositional Dynamic Logic and standard query languages.

Data provenance records

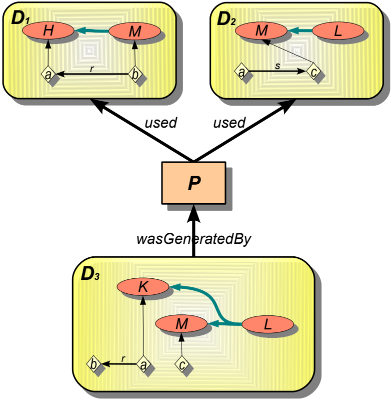
A data provenance record is the *history of derivation* of a *data artifact* from its sources.

Data artifact =
dataset / knowledge base = a set of axioms (DL/OWL/RDF(S))



Note: Particular representation languages come with dedicated query languages, e.g., conjunctive queries for DLs/OWL, datalog for OWL RL, SPARQL for RDF(S).

Data provenance records



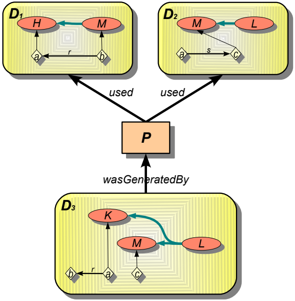
Provenance graphs:

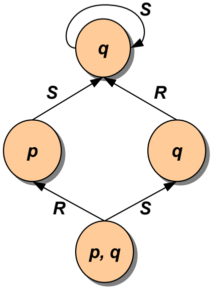
- *process* nodes: P
- *data artifact* nodes: D_1, D_2, D_3 (each corresponding to a data artifact)
- edges labeled with *relation names*, e.g.: *wasGeneratedBy*, *used*.
- directed, acyclic, finite.

L. Moreau, et al. **The open provenance model – core specification.** In *Future Generation Computer Systems 27*, 2010.

Verification as model-checking

Provenance graphs are very similar to *finite-state transition systems*.





- natural to analyze using the framework of modal logics, in particular *Propositional Dynamic Logic*,
- basic reasoning task is *model-checking*,
- we need to replace propositions with richer formulas — *queries* — and effectively work with *two-dimensional languages*.

Provenance specification logic

Object formulas:	q	$::=$	queries from a given class $\mathcal{Q}$
Path expressions:	π	$::=$	$r \mid \pi;\pi \mid \pi \cup \pi \mid \pi^- \mid \pi^* \mid v? \mid \alpha?$
Provenance formulas:	α	$::=$	$\{q\} \mid \langle\pi\rangle\alpha \mid \alpha \wedge \alpha \mid \neg\alpha \mid \top$

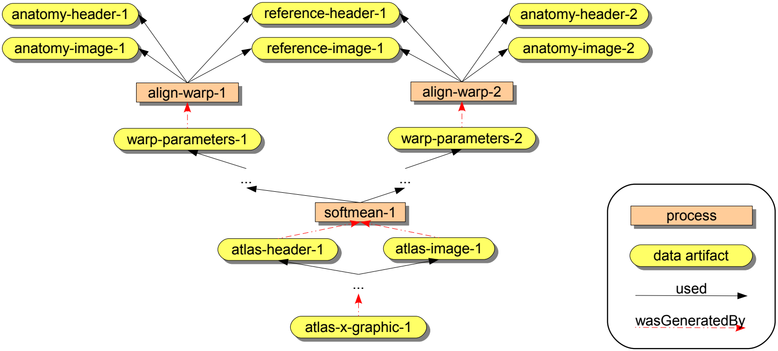
The semantics is a *combination of the semantics* of PDL and $\mathcal{Q}$-queries:

- a sequence of instances $\vec{a}$ is an *answer* to α *iff* $G, v \models \alpha[\vec{a}]$
- for a query $q(\vec{x})$ in α, and node v, $q(\vec{x})$ is *satisfied* in v for $\vec{a}$ *iff* $D(v) \models q[\vec{a}|_{\vec{x}}]$

Model-checking problem: given a provenance graph G, node v, provenance formula α, and a sequence $\vec{a}$, decide wether $G, v \models \alpha[\vec{a}]$.

The First Provenance Challenge

- a workflow for creating "atlases" of high resolution anatomical data
- 9 queries about the resulting provenance records



L. Moreau, et al. **Special issue: The First Provenance Challenge.** In *Concurrency and Computation: Practice and Experience 20*, 2008.

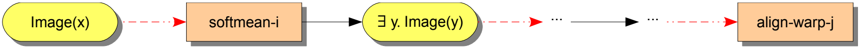
Example

Q: *Find all output averaged images of softmean (average) procedures, where the warped images taken as input were align warp'ed using a twelfth order nonlinear 1365 parameter model, i.e. where softmean was preceded in the workflow, directly or indirectly, by an align warp procedure with argument -m 12.*

$$\alpha := \{\mathsf{Image}(x)\} \wedge \langle \mathit{wasGeneretedBy}; \mathit{softmean}_{1\ldots n}; \mathit{used}\rangle(\{\exists y.\mathsf{Image}(y)\} \wedge \langle(\mathit{wasGeneratedBy}; \mathit{used})^*; \mathit{wasGeneratedBy}; \mathit{align\text{-}warp}_{1\ldots m}\rangle \top)$$

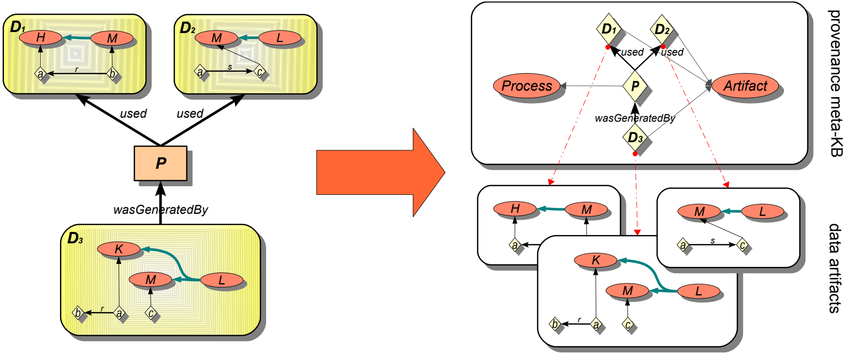
where: $\mathit{softmean}_{1\ldots n} := \mathit{softmean}\text{-}1? \cup \ldots \cup \mathit{softmean}\text{-}n?$

$\mathit{align\text{-}warp}_{1\ldots m} := \mathit{align\text{-}warp}\text{-}1? \cup \ldots \cup \mathit{align\text{-}warp}\text{-}m?$



Accommodating rich provenance metadata

Represent the provenance graph as a *separate* (meta-)*knowledge base*.



Add new *test operator $C?$*, for a concept C of the provenance language.

$$G, v \models C? \quad \textit{iff} \quad \text{meta-KB} \models C(v)$$

Example cntd.

Q: [...] *was preceded in the workflow, directly or indirectly, by an align warp procedure with argument -m 12*.

Provenance meta-KB:

- $\mathit{Align\text{-}warp} \sqsubseteq \mathit{Process}$
- $\mathit{Align\text{-}warp} \sqsubseteq \exists \mathit{argument}.\mathtt{String}$
- $\mathit{Align\text{-}warp}(\mathit{align\text{-}warp}_i)$, for every $1 \leq i \leq m$,
- $\mathit{argument}(\mathit{align\text{-}warp}_k, \text{“-m 12”})$, for every $\mathit{align\text{-}warp}_k$ with argument “-m 12”.

$$\alpha := ...\langle(\mathit{wasGeneratedBy}; \mathit{used})^*; \mathit{wasGeneratedBy}; \mathit{align\text{-}warp}_{1...m}\rangle\top$$

where: $\mathit{align\text{-}warp}_{1...m} := \mathit{align\text{-}warp}\text{-}1? \cup \ldots \cup \mathit{align\text{-}warp}\text{-}m?$
replace: $\mathit{align\text{-}warp}_{1...m}$
with: $\mathit{Align\text{-}warp} \sqcap \exists \mathit{argument}.\text{“-m 12”}?$

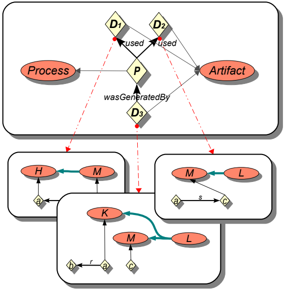
Observations

- we assume this collection is *representative of the problem* of reasoning with data provenance,
- the tasks consist of a *logical verification* component and a *search component*,
- the logical verification component *can be captured by PSL*, often by breaking down complex tasks into a number of model-checking problems,
- the queries are essentially *two-dimensional*,
- some patterns could be usefully compiled out as a syntactic sugar.

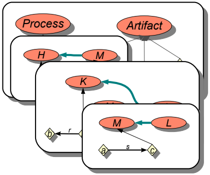
Reasoning

Reasoning in PSL is $\textsc{PTime}^{SW}$-complete, where:

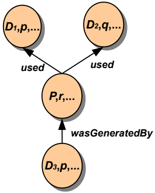
- $\textsc{PTime}$ is the complexity of model-checking in PDL,
- $\cdot^{SW}$ is an oracle performing reasoning with the Semantic Web representation/query languages used, of the respective complexity.



Reasoning with
data provenance records



Reasoning with data



Model-checking
transition systems

Summary

Our problem involved:

- adequately representing provenance records
  ⇒ *provenance graphs*, i.e. transition systems with rich data states. The approach is agnostic as to the choice of particular data and provenance languages,
- defining a language for expressing relevant properties
  ⇒ *PSL* = dynamic logic + query formulas as atoms,
- ensuring that reasoning is "manageable"
  ⇒ $\textsc{PTime}^{SW}$-completeness is good!

Conclusion:

A generic, declarative approach to reasoning with data provenance records.

Outlook:

Broader validation, implementation, study of most useful setups.