Modeling and Querying Scientific Workflow Provenance in the D-OPM

Víctor Cuevas-Vicenttín, Saumen Dey, Michael Li Yuan Wang, Tianhong Song, Bertram Ludäscher

7th Workshop on Workflows in Support of Large-Scale Science
WORKS´12
Salt Lake City, UT, November 12 2012

University of California at Davis – University of New Mexico – DataONE
Scientific workflows and provenance

- *Provenance* is concerned with the origin, context, derivation, ownership or history of some artifact [CFLV12]
  - How was this data created and by whom?
  - Which tools were employed to generate this result?
  - Which intermediate products were affected by this dataset?

→ **Model and query scientific workflow provenance**
The Open Provenance Model

Object properties implementing OPM
- wasDerivedFrom
- wasGeneratedBy
- wasTriggeredBy
- wasControlledBy
- used
- wasGeneratedAt
- time:TemporalEntity

Object properties not as exactly defined in OPM
- rdfs:subClassOf relationships

1 prefix time: http://www.w3.org/2006/time#

owl:Thing
Agent
Artifact
Process
time:Instant
time:Interval
D-OPM for scientific workflows

• Prospective and retrospective provenance
  – Possible-future and past workflow executions
• Compatible with different workflow models
  – Kepler, Taverna, Vistrails, etc.
• Highly informative though not comprehensive
• Supports various querying mechanisms
Aspects to cover with D-OPM

- Workflow modeling and structure (WF)
- Composition and subworkflows (SW)
- Workflow execution and traces (TR)
- Data representation and structure (DS)
- Workflow evolution (WE)*
- Temporal data and constraints (TE)*
- User and execution context information (CX)*

* Work on process
D-OPM UML diagram
Regrid and rescale climate data WF

... Task SCRIPTool(in source_grid str, in target_grid str, out factor_matrix str);
... Process NARR_Precipitation

  Connectors sf1 sf2 bsf csf tgbsf tgcsf sgbcsv sgccsv tgbcsv tgccsv sgnet targnet regfactmat
  sgi tgi mwf ornarrregnarr gpcp resnarr;
  ArcGIS_Reproject(sf1, bsf);
  ArcGIS_Reproject(sf2, csf);
  GDAL_Reformat(bsf, sgbcsv);
  GDAL_Reformat(csf, sgccsv);
  GDAL_Reformat(tgbsf, tgbcsv);
  GDAL_Reformat(tgcsf, tgccsv);
  IDL_CreateNetcdf(sgbcsv, sgccsv, sgnet);
  IDL_CreateNetcdf(tgbcsv, tgccsv, targnet);
  SCRIPTool(sgnet, targnet, regfactmat);
  Matlab_Reformat(regfactmat, sgi, tgi, mwf);
  RegridAWA(sgi, tgi, mwf, ornarr,regnarr);
  Matlab_Rescale(regnarr, gpcp, resnarr);

Endprocess
D-OPM reference implementation

- Workflow spec. [Kahn PN]
- PN language parser
- D-PROV model manager [Hibernate]
- Workflow Engine [Akka]
- GUI
- D-PROV DB
- Task classes [Reflection]
- Akka actors
Regrid and rescale climate data WF
Regular path queries for WF provenance

- Labeled directed graph \( G = (V, E, L) \), \( E \subseteq V \times L \times V \)
- Paths \( \pi = x_0 \xrightarrow{l_1} x_1 \xrightarrow{l_2} x_2 \xrightarrow{l_3} \ldots \xrightarrow{l_n} x_{n-1} \xrightarrow{l_n} x_n \)
- A RPQ returns *pairs of nodes* for which there is a *path* between them that satisfies a *regular expression*
- RPQ/2 and RPQ/4 variants
- \( R ::= R \cdot R \mid R \mid R^* \mid R^+ \mid R^{-1} \mid R? \)
Regular path query example

What are the data artifacts in CSV format that should be updated if we run a new version of the ArcGIS tool?

ArcGIS.toolsUsedBy.(wasGeneratedBy.used?)*.csv
Regular path queries evaluation

ArcGIS toolsUsedBy \cdot \text{wasGeneratedBy used ? \cdot * . \cdot csv .}

\begin{equation}
g(V, \text{subexpression}, V)
\end{equation}

ArcGIS toolsUsedBy . \text{wasGeneratedBy used ? \cdot * . csv .}

\begin{equation}
g(V, L, V)
\end{equation}

RDBMS

Transitive closure script

SQL queries

RPQ Engine
Experimentation

• Testbed: Binary Decision Diagrams
  – Propositional directed acyclic graphs
  – Instances of the N-queens problem
• Queries involving different operators
  – Concatenation: 0.0.0
  – Transitive closure: 0+, 0*
  – Inverse: $0^{-1}.0^{-1}.0^{-1}$
• Graphs of different sizes
  – Between 1.1 and 95K nodes, 2.2 and 190K edges
Experimental results

Query: 0.0.0 (RPQ/2)

Query 0.0.0 (RPQ/4)

Query 0+ (RPQ/2)

Query 0+ (RPQ/4)
Conclusions

- Extension of OPM for scientific workflow provenance
  - Covers multiple aspects and supports querying
  - Initial validation and experimentation
- Provenance graph querying mechanism based on regular path queries
  - DBMS-based implementation enables interoperability and extensibility
  - Viable performance for moderately sized graphs
Related work

• OPM extension for scientific workflows and collection framework [Lim-Lu-Chebotko-Fotouhi(2010)]

• OPMW: OPM profile to represent abstract workflows and enable SPARQL querying [Garijo-Gil (2011)]

• Regular path queries
  – Characterization and complexity analysis [Mendelzon-Wood(1995)]
Perspectives and future work

• Characterization and implementation of temporal aspects
• Alignment with W3C’s PROV
• Evaluation of large scale storage and processing infrastructures
  – DataONE Cyberinfrastructure, MapReduce, Pregel
• Extensions to the query language


References


