CIS 602: Provenance & Scientific Data Management

Provenance Standards

Dr. David Koop
Reminders

- Project Emails:
  - Almost everyone has now received my feedback on their projects
  - If you do not email me back, I will assume that you concur with my understanding of your project
  - That means, what I wrote is what I expect to see in the final project!

- Project Progress Reports Due November 13
  - Can serve as the start to your final project report
Related Work and Citations

- Related work is meant to provide context and background as well as give credit for past work.
- Citations point to other papers that a reader can look at for more information.
  - Citations should immediately the use of the concept, summary, argument, or position in the paper.
  - Citations that reference examples can be as simple as (e.g. [21, 41]) or (e.g. (Smith, 1994), (Jones, 2001a)).
- Direct quotes are rare in computer science research papers.
  - Reuse an exact definition.
  - Point to a position that you are concerned about misrepresenting.
History of Provenance (in Computer Science)

• Data Derivation & Provenance Workshop, 2002

• International Provenance & Annotation Workshop (IPAW)
  - Meeting for community to discuss provenance and provenance-specific issues
  - Published proceedings, papers, and posters
  - Held every other year, 2006-present

• Workshop on the Theory and Practice of Provenance (TaPP)
  - Developed to focus on new ideas, short papers on new ideas, vision statements
  - Held every year, 2009-present

• This year IPAW and TaPP were co-located and integrated into a “Provenance Week” event
Provenance Challenges (2006-2009)

- Developed to “understand the different representations used for provenance, its common aspects, and the reasons for its differences” [http://twiki.ipaw.info/bin/view/Challenge/]

- Led to development of Open Provenance Model (OPM) and eventually PROV.
First Provenance Challenge (2006)

- Focus on representations, capabilities, and scope of provenance-aware systems
- Developed a simple example workflow from Functional MRI
- 17 Teams contributed:
  - Representations of the workflow in their system
  - Representations of provenance for the example workflow
  - Representations of the result of the core (and other) queries
  - For each query, whether (1) the query can be answered by the system, (2) the system cannot answer the query now but considers it relevant, (3) the query is not relevant to the project.
1. Find the process that led to Atlas X Graphic / everything that caused Atlas X Graphic to be as it is.
2. Find the process that led to Atlas X Graphic, excluding everything prior to the averaging of images with softmean.
3. Find the Stage 3, 4 and 5 details of the process that led to Atlas X Graphic.
4. Find all invocations of procedure align_warp using a twelfth order nonlinear 1365 parameter model that ran on a Monday.
5. Find all Atlas Graphic images outputted from workflows where at least one of the input Anatomy Headers had an entry global maximum=4095.
6. Find all output averaged images of softmean procedures where the warped images taken as input were align_warped using a 12th order nonlinear 1365 parameter model.
7. A user has run the workflow twice, in the second instance replacing each procedures (convert) in the final stage with two procedures: pgmtoppm, then pnmtojpeg. Find the differences between the two workflow runs.
8. A user has annotated some anatomy images with a key-value pair center=UChicago. Find the outputs of align_warp where the inputs are annotated with center=UChicago.
9. A user has annotated some atlas graphics with key-value pair where the key is studyModality. Find all the graphical atlas sets that have metadata annotation studyModality with values speech, visual or audio, and return all other annotations to these files.
## First Provenance Challenge Results

<table>
<thead>
<tr>
<th>Characteristic</th>
<th>Redux</th>
<th>Mindswap</th>
<th>Karma</th>
<th>JP</th>
<th>myGrid</th>
<th>VisTrails</th>
<th>ES3</th>
<th>ZOOM</th>
<th>RWS</th>
<th>COMAD</th>
<th>PASS</th>
<th>SDG</th>
<th>NCSD2K</th>
<th>NCSCI</th>
<th>VDL</th>
<th>OPA</th>
<th>USC/ISI</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>1. Characteristics of Provenance Systems</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>1.1 Execution Environment</strong></td>
<td>Workflow system</td>
<td>Web</td>
<td>Workflow system</td>
<td>Workflow system</td>
<td>Operating system</td>
<td>Workflow system</td>
<td>Workflow system</td>
<td>Operating system</td>
<td>Workflow system</td>
<td>Visual Program. Env.</td>
<td>Workflow system</td>
<td>Workflow system</td>
<td>Technological Workflow system</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>(actual system)</td>
<td>WWF</td>
<td>XBaya and SPEL</td>
<td>EGEE glide (inc. Condor, Dagman and JDL)</td>
<td>Taverna and Xcuff</td>
<td>VisTrails system</td>
<td>IDL and Bash</td>
<td>Technology independent</td>
<td>Kepler, Polonius</td>
<td>Kepler, Polonius</td>
<td>Linux</td>
<td>Kepler, Polonius</td>
<td>D2K</td>
<td>Cyber Integrator</td>
<td>VDS</td>
<td>Wings and Pegasus</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>1.2 Execution Environment (for the challenge)</strong></td>
<td></td>
<td></td>
<td>EGEE VOCE VO</td>
<td>VisTrails system and shell</td>
<td>SQL scripts</td>
<td>Shell script</td>
<td>RDF (external) and SAM/Web (internal)</td>
<td>RDF</td>
<td>RDF</td>
<td>RDBMS</td>
<td>Internal and XML view</td>
<td>OWL and RDSMS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>1.3 Provenance Representation</strong></td>
<td>RDBMS</td>
<td>OWL</td>
<td>XML View and RDBMS</td>
<td>RDF</td>
<td>XML View and RDBMS</td>
<td>XML view</td>
<td>RDBMS</td>
<td>Internal and XML view</td>
<td>Internal</td>
<td>OWL and RDSMS</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>1.4 Query Language</strong></td>
<td>SQL</td>
<td>SPARQL</td>
<td>Java</td>
<td>Java</td>
<td>Java</td>
<td>Java</td>
<td>Java</td>
<td>Java</td>
<td>Java</td>
<td>Java</td>
<td>Java</td>
<td>Java</td>
<td>Java</td>
<td>Java</td>
<td>Java</td>
<td>Java</td>
<td>Java</td>
</tr>
<tr>
<td><strong>1.6 Challenge Implementation</strong></td>
<td>Run</td>
<td>Run</td>
<td>Run</td>
<td>Run</td>
<td>Run</td>
<td>Simulated</td>
<td>Run</td>
<td>Run</td>
<td>Run</td>
<td>Partial</td>
<td>Run</td>
<td>Partial</td>
<td>Run</td>
<td>Partial</td>
<td>Run</td>
<td>Partial</td>
<td>Run</td>
</tr>
</tbody>
</table>

### 2. Properties of Provenance Representation

#### 2.1 Includes workflow representation?
- Yes
- No

#### 2.2 Data Derivation vs Causal Flow of Events
- Yes
- No

#### 2.3 Arbitrary annotations in scope/implemented?
- Yes
- No

#### 2.4 Time supported/required
- Yes
- No

#### 2.5 Naming required (if yes, then what?)
- Keys for ports and data
- No

#### 2.6 Tracked data, and granularity
- Port level (I/O) but not their contents
- All data
- Any GUID assignable data

#### 2.7 Abstraction mechanisms
- Layered provenance model
- Script or job
- User view of composite steps
- N/A

---


Prepared using cpeauth.cls

First Provenance Challenge Results
Second Provenance Challenge (2007)

- First challenge helped identify common characteristics as well as differences
- Want to better understand how the systems relate
- Focus on **interoperability**
- Teams share provenance of different components of workflow, run queries over compositions of provenance data (cross-model provenance queries)
- Same workflow and queries as in first challenge
- Goals:
  - Understand where data in one model is translatable to or has no parallel in another model.
  - Understand how the provenance of data can be traced across multiple systems, so adding value to all those systems.
Second Provenance Challenge Results

• Interoperability was fairly straightforward: teams could understand each other’s provenance without much interaction

• Issues:
  - Naming: identifying inputs/outputs across different systems’ provenance can be challenging without any agreed-upon identification scheme
    • Artificial problem?
    • Could compare data by checksums/hashes
    • Defer to actual names used in executions
  - Extraneous Information: other systems capture provenance information that isn’t used by another system’s query system

• Different levels of abstraction for provenance
• General agreement on annotated causality graph representation
Open Provenance Model (2007)

- **http://twiki.ipaw.info/bin/view/Challenge/OPM**
- **Requirements:**
  - To allow provenance information to be exchanged between systems, by means of a compatibility layer based on a shared provenance model.
  - To allow developers to build and share tools that operate on such provenance model.
  - To define the model in a precise, technology-agnostic manner.
  - To support a digital representation of provenance for any “thing”, whether produced by computer systems or not.
  - To define a core set of rules that identify the valid inferences that can be made on provenance graphs.

[L. Moreau et al.]
Open Provenance Model

• Non-requirements

- It is not the purpose of this document to specify the internal representations that systems have to adopt to store and manipulate provenance internally; systems remain free to adopt internal representations that are fit for their purpose.

- It is not the purpose of this document to define a computer-parseable syntax for this model; model implementations in XML, RDF or others will be specified in separate documents, in the future.

- We do not specify protocols to store such provenance information in provenance repositories.

- We do not specify protocols to query provenance repositories.

[L. Moreau et al.]
OPM Components

- **Artifact**: Immutable piece of state, which may have a physical embodiment in a physical object, or a digital representation in a computer system.
- **Process**: Action or series of actions performed on or caused by artifacts, and resulting in new artifacts.
- **Agent**: Contextual entity acting as a catalyst of a process, enabling, facilitating, controlling, affecting its execution.

[L. Moreau et al.]
OPM Edges

A \xrightarrow{\text{used}(R)} P

P \xrightarrow{\text{wasGeneratedBy}(R)} A

Ag \xrightarrow{\text{wasControlledBy}(R)} P

P1 \xleftarrow{\text{wasTriggeredBy}} P2

P1 \xleftarrow{\text{wasDerivedFrom}} A1

A1 \xleftarrow{\text{wasDerivedFrom}} A2

[L. Moreau et al.]
OPM Cake Example

John

wasControlledBy(cook)

Bake

wasGeneratedBy(cake)

100g Butter

used(butter)

2 eggs

used(egg)

100g Sugar

used(sugar)

100g Flour

used(flour)

Cake

[L. Moreau et al.]
OPM Inferences

A1 ← used(R1) Acc1

P1 ← wasGeneratedBy(R2) Acc2

A2 ← used(R3) Acc3

P2

wasTriggeredBy

wasDerivedFrom (asserted)

mayHaveBeenDerivedFrom (inferred)

[L. Moreau et al.]
Other OPM Notes

- Model is timeless but may have time annotations
- Has overlapping and hierarchical descriptions (Accounts)
- Roles are mandatory in certain edges
- Has support for collections
Third Provenance Challenge (2009)

- Main Goal: test the efficacy of OPM in representing causality of real world applications and, importantly, its usefulness for interoperating across systems
- Other goals:
  - Build momentum around the proposed standard, integrate support for OPM within existing provenance systems
  - Spawn novel OPM-based tools for the community

[Y. Simmhan et al.]
Third Provenance Challenge Workflow & Queries

- Pan-STARRS sky survey project
- Loads CSV files and converts raw image data into query-able observations that are stored in a relational database

Queries:
- For a given detection, which CSV files contributed to it?
- The user considers a table to contain unexpected values. Was the range check performed for this table?
- Which operation executions were strictly necessary for the Image table to contain a particular (non-computed) value?
Third Provenance Challenge Results

<table>
<thead>
<tr>
<th>Team</th>
<th>Workflow/Provenance System</th>
<th>Native Provenance Format</th>
<th>OPM Binding Format</th>
<th>Native Provenance Storage</th>
<th>PC3 Query Format</th>
<th>Teams Interoperated With (Query Success Status)</th>
</tr>
</thead>
<tbody>
<tr>
<td>NCSA</td>
<td>Java/Tupelo</td>
<td>RDF</td>
<td>XML, RDF</td>
<td></td>
<td>Java API</td>
<td>NCSA (Success)</td>
</tr>
<tr>
<td>Univ of Chicago</td>
<td>Swift</td>
<td>Relational</td>
<td>XML</td>
<td>DBMS</td>
<td></td>
<td>UChicago (Partial)</td>
</tr>
<tr>
<td>Microsoft Research</td>
<td>Trident</td>
<td>Relational</td>
<td>XML</td>
<td>DBMS/MS SQL Server</td>
<td>SQL</td>
<td>MSR (Success), Soton (Fail), UoM (Fail), UCD (Partial), IU (Success), UUtah (Fail)</td>
</tr>
<tr>
<td>UC-Davis</td>
<td>COMAD-Kepler</td>
<td>Relational</td>
<td>XML</td>
<td>DBMS/ MySQL</td>
<td>SQL</td>
<td>UCD (Success), Soton, RPI, KCL, Harvard</td>
</tr>
<tr>
<td>Univ of Soton, USC-ISI</td>
<td>Java</td>
<td>XML</td>
<td>XML, RDF</td>
<td>File</td>
<td>Java API</td>
<td>Soton (Success)</td>
</tr>
<tr>
<td>Univ of Manchester</td>
<td>Taverna</td>
<td>RDF</td>
<td>XML, RDF</td>
<td></td>
<td>SPARQL</td>
<td>UoM, UCD, Soton, NCSA</td>
</tr>
<tr>
<td>RPI/Tetherless</td>
<td>Java/ProtoProv</td>
<td>RDF</td>
<td>XML</td>
<td>Jena</td>
<td>SPARQL</td>
<td>RPI (Success)</td>
</tr>
<tr>
<td>UvA/VL-e</td>
<td>WS-VLAM/PLIER</td>
<td>Relational</td>
<td>XML</td>
<td>DBMS</td>
<td>SQL</td>
<td>UvA (Success)</td>
</tr>
<tr>
<td>SDSC</td>
<td>Kepler</td>
<td></td>
<td>XML</td>
<td></td>
<td>XQuery</td>
<td>SDSC (Success)</td>
</tr>
<tr>
<td>Univ of Utah</td>
<td>VisTrails, Python</td>
<td>XML</td>
<td>XML</td>
<td></td>
<td>XQuery</td>
<td>UUtah (Partial), SDSC (Partial)</td>
</tr>
<tr>
<td>Kings College, London</td>
<td>Java</td>
<td>XML</td>
<td></td>
<td></td>
<td></td>
<td>KCL (Success)</td>
</tr>
<tr>
<td>Harvard Univ</td>
<td>Bash, Java/Pass</td>
<td>Path Query Language Graphs</td>
<td>XML</td>
<td>PQL DB Store</td>
<td>PQL</td>
<td>Harvard (Success)</td>
</tr>
<tr>
<td>Indiana Univ</td>
<td>ODE/Karma</td>
<td>Relational</td>
<td>XML</td>
<td>DBMS/ MySQL</td>
<td>SQL</td>
<td>IU (Success)</td>
</tr>
<tr>
<td>UTEP</td>
<td>WDO-It!</td>
<td>Proof Markup Language</td>
<td></td>
<td>File</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
PROV

- At Third Provenance Challenge Workshop, connections made with W3C and PROV incubator was created
- Some similarities to OPM, some of the same people involved, but also tried to involve a wider group
PROV Model Primer


Presented by: Srinidhi Pesara
PROV Tutorial

L. Moreau, P. Groth, T. D. Huynh
PROV Discussion

• What are the benefits of PROV?
• What are potential problems with PROV?
• PROV to promote “inter-operable interchange”, not necessarily the primary storage of provenance for each system?
• Extensions of the core model for specific systems and/or uses
  - ProvONE: http://vcvcomputing.com/provone/provone.html
  - Wf4ever provenance: http://wf4ever.github.io/ro/
• Other examples:
  - Extracting PROV from Wikipedia: [P. Missier & Z. Chen, 2013]
  - Git2Prov: [T. de Nies et al., 2013]
Reminders

• Project Emails:
  - Almost everyone has now received my feedback on their projects
  - If you do not email me back, I will assume that you concur with my understanding of your project
  - That means, what I wrote is what I expect to see in the final project!

• Project Progress Reports Due November 13
  - Can serve as the start to your final project report

• Next Class: Visualization & Provenance