CIS 602: Provenance & Scientific Data Management

Introduction

Dr. David Koop
About Me

• 1998-2002: B.S. in Mathematics, B.Comp.Sci. [Calvin College]
• 2002-2005: M.S. in Computer Science [University of Wisconsin-Madison]
• 2005-2006: Consulting [nVISIA, Inc.]
• 2006-2011: Ph.D. in Computing [University of Utah]
• 2011-2012: Senior software architect [VisTrails, Inc.]
• 2012-2014: Research assistant professor [New York University]
• Now: Assistant professor at UMass-Dartmouth
About Me

• Research Interests
  - Visualization
  - Computational Provenance
  - Data Science

• Research Projects
  - VisTrails: www.vistrails.org
  - VisComplete, UV-CDAT, SAHM

• Hobbies
  - Ultimate Frisbee
  - Hiking

• See my web page for more information
  - http://www.cis.umassd.edu/~dkoop/
About You

• Graduate students?

• Background:
  - Previous topics course (CIS 602)? (visualization, bioinformatics)
  - Research Papers?
  - Provenance?
  - Scientific Workflows?
  - Database Experience?
    • Relational
    • NoSQL
    • XML
    • Graph Databases
About this course

• Course web page is authoritative:
  - http://www.cis.umassd.edu/~dkoop/cis602/

• Topics course
  - A current research area the professor works in
  - A chance to be on the “cutting edge” of research

• No textbook
  - Use recent research papers

• Requires student participation
  - Reading responses
  - Reading presentation
  - Course project
About this course

• Course Registration:
  - Make sure you have registered in COIN for the course
  - Email me (dkoop@umassd.edu) if you need a permission code

• Review of course policies:
  - If you have any concerns or questions, please email me as soon as possible

• If you are not sure if this course is a good fit, please email me or talk to me
What is Provenance?
Provenance in Art

Rembrandt van Rijn
Dutch, 1606 - 1669

Self-Portrait, 1659
oil on canvas
Andrew W. Mellon Collection
1937.1.72

Provenance


[1] This early provenance is established by presence of a mezzotint after the portrait by R. Earlom (1743-1822), dated 1767. See John Charrington, A Catalogue of the Mezzotints After, or Said to Be After, Rembrandt, Cambridge, 1923, no. 49.

Associated Names

• Buccleuch, Henry, 3rd Duke of
• Buccleuch, John Charles, 7th Duke of
• Colnaghi & Co., Ltd., P. & D.
• Knoedler & Company, M.
• Mellon, Andrew W.
• Mellon Educational and Charitable Trust, The A.W.
• Montagu, and 4th Earl of Cardigan, George, 3rd Duke of
Provenance in Art

Rembrandt van Rijn
Dutch, 1606 - 1669

Self-Portrait, 1659
oil on canvas
Andrew W. Mellon Collection
1937.1.72

Provenance


[1] This early provenance is established by presence of a mezzotint after the portrait by R. Earlom (1743-1822), dated 1767. See John Charrington, A Catalogue of the Mezzotints After, or Said to Be After, Rembrandt, Cambridge, 1923, no. 49.

Associated Names

- Buccleuch, Henry, 3rd Duke of
- Buccleuch, John Charles, 7th Duke of
- Colnaghi & Co., Ltd., P. & D.
- Knoedler & Company, M.
- Mellon, Andrew W.
- Mellon Educational and Charitable Trust, The A.W.
- Montagu, and 4th Earl of Cardigan, George, 3rd Duke of

[National Gallery of Art]
Provenance in Science

- Provenance: the **lineage** of data, a computation, or a visualization

- **Provenance is as (or more) important as the result!**

- Old solution:
  - Lab notebooks

- New problems:
  - Large volumes of data
  - Complex analyses
  - Writing notes doesn’t scale

- Let computers handle this automatically!

[DNA Recombination, Lederberg]
Provenance in Science

• Provenance: the **lineage** of data, a computation, or a visualization

• **Provenance is as (or more) important as the result!**

• Old solution:
  - Lab notebooks

• New problems:
  - Large volumes of data
  - Complex analyses
  - Writing notes doesn’t scale

• Let computers handle this automatically!

[DNA Recombination, Lederberg]
Provenance-Rich Computational Science

Data Management

Provenance

Visualization

Publishing

ACKNOWLEDGMENTS

REFERENCES
Scientific Workflows

- **Problem:** Lots of different tools, often need to use more than one for reasonably complex analyses

- **Solutions:**
  - Rely on human memory
  - Scripts
  - Workflows
Scientific Workflows

```python
data = vtk.vtkStructuredPointsReader()
data.SetFileName('..\examples\data\head.120.vtk')
contour = vtk.vtkContourFilter()
contour.SetInput(data.GetOutput())
contour.SetValue(0, 67)
mapper = vtk.vtkPolyDataMapper()
mapper.SetInput(contour.GetOutput())
mapper.ScalarVisibilityOff()
actor = vtk.vtkActor()
actor.SetMapper(mapper)
cam = vtk.vtkCamera()
cam.SetViewUp(0, 0, -1)
cam.SetPosition(745, -453, 369)
cam.SetFocalPoint(135, 135, 150)
cam.ComputeViewPlaneNormal()
ren = vtk.vtkRenderer()
ren.AddActor(actor)
ren.SetActiveCamera(cam)
ren.ResetCamera()
renwin = vtk.vtkRenderWindow()
renwin.AddRenderer(ren)
style = vtk.vtkInteractorStyleTrackballCamera()
iren = vtk.vtkRenderWindowInteractor()
iren.SetRenderWindow(renwin)
iren.SetInteractorStyle(style)
iren.Initialize()
iren.Start()
```
Scientific Workflow Provenance

```xml
<module id="12" name="vtkDataSetReader"
   start_time="2010-02-19 11:01:05"
   end_time="2010-02-19 11:01:07"/>
<module id="13" name="vtkContourFilter"
   start_time="2010-02-19 11:01:07"
   end_time="2010-02-19 11:01:08"/>
<module id="15" name="vtkDataSetMapper"
   start_time="2010-02-19 11:01:09"
   end_time="2010-02-19 11:01:12"/>
<module id="16" name="vtkActor"
   start_time="2010-02-19 11:01:12"
   end_time="2010-02-19 11:01:13"/>
<module id="17" name="vtkCamera"
   start_time="2010-02-19 11:01:13"
   end_time="2010-02-19 11:01:14"/>
<module id="18" name="vtkRenderer"
   start_time="2010-02-19 11:01:14"
   end_time="2010-02-19 11:01:14"/>
...
```
Evolution Provenance: Photo Editing

• User Actions

- original
- darkened
- sharpened
- grayscale

• Undo/Redo History
Evolution Provenance: Photo Editing

- User Actions

original → darkened → sharpened → grayscale

- Undo/Redo History

whidbey.png

Open

Brightness/Contrast

Watercolor

watercolor
Workflow Evolution Provenance

- initial data
  - corrected data
    - November ff
      - sum of ffs
        - 30-D weekly
          - 161st-River
    - November 2 data
    - August 16 Tab
  - station locations
    - station map
    - added fares
  - difference

- GetFareData (Group)
  - DateRange (PythonSource)
    - BuildLabels (PythonSource)
      - MplFigure
        - MplFigureCell
      - MplBar
        - MplFigureProperties
      - MplAxesProperties
    - MplFigure
      - MplFigureCell
    - 161st-River
    - Join Table
      - ProjectTable
      - CSVFile
      - JSONFile
    - heatmap
      - filtered
      - concourse line
      - Broadway line
        - August 16
      - Broadway diff map
        - HTTPFile
          - HTTPFile

Workflow Evolution Provenance
Workflow Evolution Provenance
Workflow Evolution Provenance
delete module “GMapCell”
delete module “CellLocation”
delete module “ProjectTable”
delete module “SelectFromTable”

... 

add module “SelectFromTable”
add parameter “float_expr” to “SelectFromTable” with value “latitutde > 40.6”
delete parameter “float_expr” from “SelectFromTable”
add parameter “float_expr” to “SelectFromTable” with value “latitutde > 40.7”
delete parameter “float_expr” from “SelectFromTable”
add parameter “float_expr” to “SelectFromTable” with value “latitutde > 40.8”

...
System-Level Provenance

[PASS Architecture, Seltzer et al.]
Database Provenance

- Determine the raw data that contributed to a computed table

| Source database $D$: Employee | Department | Query $Q$: 
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>empid</td>
<td>dept</td>
<td>deptid</td>
</tr>
</tbody>
</table>
| e977 ($a_1$) CS ($a_2$) | BME ($b_1$) | SELECT $e$.empid, $e$.dept, $d$.budget 
FROM Employee $e$, Department $d$
WHERE $e$.dept = $d$.deptid 
PROPAGATE default |
| e132 ($a_3$) EE ($a_4$) | CS ($b_3$) | FROM |
| e657 ($a_5$) BME ($a_6$) | EE ($b_5$) | WHERE |
| | MATH ($b_7$) | PROPAGATE |
| | 230K ($b_8$) | |

- For aggregations, tracing back to original data can require many more annotations or schemes that track the mappings

[Provenance in Databases, Tan et al.]
Provenance Storage

Figure 2: Example of Basic Factorization. (a) ABC1 and LXR molecule data items. (b) Same data items after Basic Factorization.
Querying Provenance

- What process led to the output image?
- What input datasets contributed to the output image?
- What workflows include resampling and isosurfacing with isovalue 57?
- Graph traversal or graph patterns
  - How do we write such queries?
Querying Provenance by Example

- Provenance is represented as graphs: hard to specify queries using text!
- Querying workflows by example [Scheidegger et al., TVCG 2007; Beeri et al., VLDB 2006; Beeri et al. VLDB 2007]
  - WYSIWYQ -- What You See Is What You Query
  - Interface to create workflow is same as to query

[Scheidegger et al.]
Provenance Analytics

Activity Histograms by Date

- The data in the previous section shows that workflow evolution provenance data that can be used to identify "user signatures", other variables, such as the time between actions can help in understanding how different users approach a problem.
- Provenance analytics allow one to measure, summarize, and remember what they did. One of the really nice features of the unobtrusive way that VisTrails captures provenance is that there is no extra burden on the user; they can do their work without caring about it. This also makes it easier to have adoption in other places.
- The process of using the new module in an example can easily be turned into a tutorial on how to use the new functionality. The instructors can help students discover how to create workflows from scratch. This makes it possible for the instructors to share their own work and the students' learning experience. Due to the provenance information, it is possible for one person to see what another person did, and to easily compare their own work to it.
- We strongly believe that teaching is one of the killer applications of provenance-enabled systems. Provenance information can help instructors to be more effective and improve their teaching.
- We classified the actions involved in workflow development into: different types of work involved in a task, we classified the actions spent changing parameters has the greatest variance of provenance-enabled systems. Provenance information can help instructors to be more effective and improve their teaching.
Provenance Analytics

Comparing Paths to Solutions for Two Students

[Lins et al.]
Provenance Mining

Database of Workflows

[VisComplete, Koop et al.]
Secure Provenance

• Make sure no one can tamper with provenance after it is originally generated
• Potential issues: want to add to provenance chain after original generation
• Solution: Secure hashing
Provenance & Semantics

- Make it possible to reason about provenance using domain-specific language
- Use semantics/linked data approaches

(a) Domain-agnostic provenance graph

(b) Domain-aware provenance graph

[Janus, Missier et al.]
Provenance Standards

- PROV W3C Standard:
  - http://www.w3.org/TR/prov-overview/
- Preceded by Open Provenance Model (OPM)
- Allow different modes of provenance captures to be combined
- Very general, work to extend to specific domains
Visualization & Provenance

The Aruvi prototype allows analysts to track and emphasize interesting objects during exploration. DOI facilitates convergent analysis, while history tracking supports divergent analysis. Analysts can show or hide low DOI objects through the show only filtered data interface. The current selection interface displays high DOI objects, and the information bar shows details about selection and size encoding.

History Tracking can be configured with various heuristics to avoid excessive detail. The base model allows for varied choices in tracking visualization states.

USE CASE

We present a use case where a user explores a digital camera dataset using Aruvi. The user identifies trends and finds cameras that meet requirements, performing trend analysis by comparing attributes for different years. They record findings in a mind map, which is a radial diagram used for idea representation.[9] The central idea, trend analysis, is recorded in note 1. The user plots camera models with specific megapixel ranges and zoom ratios.

Recent camera features are based on digital TTL and have specific characteristics.

In recent cameras, Nikon or Canon models are preferred, but the user also wants to see other cameras.

CyberShot DSC-H5 is a recent camera with megapixels > 7.0 and zoom ratios available in the market.

Aruvi, Shrinivasan & van Wijk
Reproducibility

- Capture **how** results were achieved
- Includes **many** different items
- Improve **collaboration** and sharing
Scientific Data Management

• Relational Databases
  - Row-column format data, aggregation, grouping
  - Very useful
  - Time-tested

• Scientific Data
  - Different requirements
  - New types of users
  - Streaming
  - Often shared
  - Data is not the end of the story
Graph Databases

• Lots of data in the form of graphs: molecules, social networks
• Store structured data in a way that makes connectivity queries more efficient
Graph Indexing

- Subgraph Isomorphism is NP-Complete
- Use frequent graphs or other techniques to make searching graph databases more efficient

Figure 1: A Graph Database with Three Graphs

Figure 2: A Query Graph

Figure 6: Frequent Graphs of $G$

Figure 7: Discriminative Graphs

[Tree + Delta, Zhao et al.]
Scientific Databases

- Store scientific data like arrays (need locality)

<table>
<thead>
<tr>
<th></th>
<th>[ 0 ]</th>
<th>[ 1 ]</th>
<th>[ 2 ]</th>
<th>[ 3 ]</th>
<th>[ 4 ]</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ 0 ]</td>
<td>(2, 0.7)</td>
<td>(5, 0.5)</td>
<td>(4, 0.9)</td>
<td>(2, 0.8)</td>
<td>(1, 0.2)</td>
</tr>
<tr>
<td>[ 1 ]</td>
<td>(5, 0.5)</td>
<td>(3, 0.5)</td>
<td>(5, 0.9)</td>
<td>(5, 0.5)</td>
<td>(5, 0.5)</td>
</tr>
<tr>
<td>[ 2 ]</td>
<td>(4, 0.3)</td>
<td>(6, 0.1)</td>
<td>(6, 0.5)</td>
<td>(2, 0.1)</td>
<td>(7, 0.4)</td>
</tr>
<tr>
<td>[ 3 ]</td>
<td>(4, 0.25)</td>
<td>(6, 0.45)</td>
<td>(6, 0.3)</td>
<td>(1, 0.1)</td>
<td>(0, 0.3)</td>
</tr>
<tr>
<td>[ 4 ]</td>
<td>(6, 0.5)</td>
<td>(1, 0.6)</td>
<td>(5, 0.5)</td>
<td>(2, 0.15)</td>
<td>(2, 0.4)</td>
</tr>
</tbody>
</table>

Step 1: Vertically partition attributes in the logical array.

<table>
<thead>
<tr>
<th></th>
<th>{ A }</th>
<th>{ B }</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ J ]</td>
<td>2 5 4 2 1</td>
<td>0.7 0.5 0.9 0.8 0.2</td>
</tr>
<tr>
<td></td>
<td>5 3 5 5 5</td>
<td>0.5 0.5 0.9 0.5 0.5</td>
</tr>
<tr>
<td></td>
<td>4 6 6 2 7</td>
<td>0.3 0.1 0.5 0.1 0.4</td>
</tr>
<tr>
<td></td>
<td>4 6 6 1 0</td>
<td>0.25 0.45 0.3 0.1 0.3</td>
</tr>
<tr>
<td></td>
<td>6 1 5 2 2</td>
<td>0.5 0.6 0.5 0.15 0.4</td>
</tr>
</tbody>
</table>

Step 2: Decompose each attribute array into equal sized, and potentially overlapping, chunks.

<table>
<thead>
<tr>
<th></th>
<th>{ A₁ }</th>
<th>{ A₂ }</th>
<th>{ A₃ }</th>
<th>{ A₄ }</th>
</tr>
</thead>
<tbody>
<tr>
<td>[ J ]</td>
<td>2 5 4</td>
<td>4 2 1</td>
<td>4 6 6</td>
<td>6 2 7</td>
</tr>
<tr>
<td></td>
<td>5 3 5</td>
<td>5 5 5</td>
<td>4 6 6</td>
<td>6 1 0</td>
</tr>
<tr>
<td></td>
<td>4 6 6</td>
<td>6 2 7</td>
<td>6 1 5</td>
<td>5 2 2</td>
</tr>
</tbody>
</table>

[SciDB]
Assignments for next Tuesday (9/9/2014)

• Reading:
  - Provenance for Computational Tasks: A Survey by Freire et al.
  - No reading response, will do sample next class
  - Be prepared for a short quiz

• Reading Topics:
  - Think about which topics you might like to present on

• Course Projects:
  - Think about potential course projects that involve the areas discussed today

• Course Registration:
  - Make sure you have registered in COIN for the course
  - Email me (dkoop@umassd.edu) if you need a permission code