Simplifying the Design of Workflows for Large-Scale Data Exploration and Visualization

Juliana Freire
http://www.cs.utah.edu/~juliana

Claudio Silva
http://www.cs.utah.edu/~csilva

University of Utah
Workflows are emerging as a paradigm for representing and managing complex computations:
- Simulations, data analysis, visualization, data integration

They capture computation and analysis processes, enabling:
- Automation, reproducibility, result sharing

Workflows are rapidly replacing primitive *shell* scripts:
- Apple’s Mac OS X Automator, Microsoft Windows Workflow Foundation, and Yahoo! Pipes

Business Workflows ⇒ Scientific Workflows
- Important differences!
Workflows: Scientific vs. Business

- Express sequence of data transformations
- Dataflow: Stateless, functional
- Data intensive, computing intensive
- Cater to a broad set of users

- Ensure rules and prescribed processes are followed
- Control flow (e.g., BPEL): State and side effects
- Targeted to programmers
Workflows have been traditionally used to automate repetitive tasks.

In exploratory tasks, *change is the norm!*
- Data analysis and exploration are iterative processes.

Figure modified from J. van Wijk, IEEE Vis 2005.
Data Exploration and Workflows

raw data: CT scan

workflow

Files (workflow specifications)

- anon4877_voxel_scale_1_zspace_20060331.srn
- anon4877_textureshading_20060331.srn
- anon4877_textureshading_plane0_20060331.srn
- anon4877_goodxferfunction_20060331.srn
- anon4877_lesion_20060331.srn

Notes

- Initial visualization with z-scaling corrected
- Added texture and shading
- Added plane to visualize internal structure
- Found good transfer function
- Identified lesion tissue
Exploration and Creativity Support

- Exploratory processes require reflective reasoning
- "Reflective reasoning requires the ability to store temporary results, to make inferences from stored knowledge, and to follow chains of reasoning backward and forward, sometimes backtracking when a promising line of thought proves to be unfruitful. ...the process is slow and laborious"
  Donald A. Norman

- Need external aids—tools to facilitate this process
  - Creativity support tools [Shneiderman, CACM 2002]
- Need aid from people—collaboration
Data Exploration and Workflows: Issues

- Hard to assemble and iteratively refine workflows
- Combine many tools and libraries: Need in-depth knowledge to weave them together
- No support for reflective reasoning
  - E.g., history of the exploration trail maintained manually through file-naming conventions and detailed notes
  - Hard to understand the exploratory process and relationships among workflows
- Lack of support for collaboration

*Existing systems fail to provide the necessary infrastructure for exploratory tasks. As a result, the generation and maintenance of workflows is a major bottleneck in the scientific process*
VisTrails: Managing Scientific Exploration

- Goal: reduce time to insight
- Build infrastructure to streamline exploratory tasks such as data analysis and visualization
- Support for collaboration
- Usability—provide tools and intuitive interfaces
- The VisTrails System: an open-source provenance-enabled scientific workflow system
  - > 6,000 downloads since 2007
  - Used in many applications: environmental modeling (OHSU), physics simulation (Cornell, LANL), medical studies (University of Utah), ...
Outline

- Using provenance to support reflective reasoning
- Exploring and re-using provenance
  - Querying workflows by example
  - Creating workflows by analogy
  - Auto-completion for workflows
- Emerging applications
- Future work
Keeping Exploration Trails

Trail

- ColorMap
- Color and opacity
- Shifted 2
- Shifted
- Linear
- Hue
- z-space
- Texture with Shading
- TS and planes
- good transfer function
- xferFunc1
- baseImage
- lesionImage1
- Coronal View
- Axial View
- Default Camera
- Locked Coronal View

Workflows

Data Products

Microsoft eScience Workshop, 2008
Change-Based Provenance

- Captures provenance of workflow evolution
- Records user actions
- Provenance = changes to computational tasks
  - Add a module, add a connection, change a parameter value
Change-Based Provenance

- Records user actions
- Provenance = changes to computational tasks
  - Add a module, add a connection, change a parameter value
- Extensible change algebra
- A vistrail node $v_t$ corresponds to the workflow that is constructed by the sequence of actions from the root to $v_t$

$$v_t = x_n \circ x_{n-1} \circ \ldots \circ x_1 \circ \emptyset$$

[Freire et al, IPAW 2006]
Exploring the Change Space

- Scripting workflows: Parameter explorations are simple to specify and apply
- Exploration of parameter space for a workflow $v_t$
  
  $$(setParameter(id_{n}, value_{n}) \circ ... \circ (setParameter(id_{1}, value_{1}) \circ v_t))$$
Exploring the Change Space

- Scripting workflows: Parameter explorations are simple to specify and apply
- Exploration of parameter space for a workflow $v_t$
  
  $\text{setParameter}(id_{n}, value_{n}) \odot \ldots \odot (\text{setParameter}(id_{1}, value_{1}) \odot v_t)$

- Exploration of multiple workflow specifications
  
  $\text{addModule}(id_{i}, \ldots) \odot (\text{deleteModule}(id_{i}) \odot v_1)$

  $\ldots$

  $\text{addModule}(id_{i}, \ldots) \odot (\text{deleteModule}(id_{i}) \odot v_n)$

- Results can be conveniently compared in the VisTrails spreadsheet
- Can create animations too!
- Caching to avoid redundant computations [Bavoil et al., IEEE Vis 2005]
Computing Workflow Differences

- No need to compute subgraph isomorphism!
- A vistrail is a rooted tree: all nodes have a common ancestor—diffs are well-defined and *simple to compute*

\[ \text{vt}_1 = x_i \circ x_{i-1} \circ ... \circ x_1 \circ \emptyset \]
\[ \text{vt}_2 = x_j \circ x_{j-1} \circ ... \circ x_1 \circ \emptyset \]
\[ \text{vt}_1 - \text{vt}_2 = \{x_i, x_{i-1}, ..., x_1, \emptyset\} - \{x_j, x_{j-1}, ..., x_1, \emptyset\} \]

- Different semantics:
  - Exact, based on ids
  - Approximate, based on module signatures
Collaborative Exploration

- Collaboration is key to data exploration
  - Translational, integrative approaches to science
- Store provenance information in a database
- Synchronize concurrent updates through locking
  - Real-time collaboration [Ellkvist et al., IPAW 2008]
- Asynchronous access: similar to version control systems
  - Check out, work offline, synchronize
  - Users exchange patches
- Synchronization is simple—provenance is monotonic
- No need for a central repository—support for distributed collaboration
  - For details see Callahan et al, SCI Institute Technical Report, No. UUSCI-2006-016 2006
Change-Based Provenance: Summary

- General: Works with any system that has undo/redo!
- Concise representation
- Uniformly captures data and workflow provenance
  - Data provenance: where does a specific data product come from?
  - Workflow evolution: how has workflow structure changed over time?
- Results can be reproduced
- *Detailed information about the exploration process*
- **Provenance beyond reproducibility:**
  - Support for reflective reasoning
  - Scalable exploration of the parameter space—results can be compared side-by-side in the spreadsheet
  - Support for collaboration
  - Understand problem-solving strategies—knowledge re-use
Storing detailed information is important, but not enough!
Need appropriate user interface and operations to leverage information
  - Understand and re-use the history
Simplify the creation of new workflows
Looking for Examples

- Need to query workflow collection:
  - Find workflows that process a particular type of file
  - Find workflows that output a particular data product
  - Find workflows that contain a given module or sequence of modules

- Workflow are graphs: hard to specify queries using text
  - SQL, SparQL, Prolog....
Querying Workflows by Example

- WYSIWYG -- What You See Is What You Query
- Interface to create workflow is same as to query

[Scheidegger et al., TVCG 2007]
Refining Workflows

- Complex workflows are hard to create
  - Domain knowledge
  - Familiarity with different tools

Steep learning curve
Refining Workflows by Analogy

- Leverage the wisdom of the crowds in *shared provenance*
  - Some workflow refinements are common, e.g., change the rendering technique, publish image on the Web

- Apply refinements by analogy, automatically [Scheidegger et al, IEEE TVCG 2007]
Refining Workflows by Analogy

is to

as

is to

?
The Analogy Algorithm

1. Compute difference: $\Delta(A, B)$
   - Just like a patch!
   - But...
   $D = \Delta(A, B) \circ C$ may not be a valid workflow

2. Find correspondences between $A$ and $C$: $\text{map}(A, C)$
   - Diffuse similarity scores across the product graph $A \times C$ using Eigenvalue decompositions

3. Compute mapped difference $\Delta_{AC}(A, B) = \text{map}(A, C) \Delta(A, B)$

4. $D = \Delta_{AC}(A, B) \circ C$
The Analogy Operation

- Allows workflows to be refined without requiring users to directly modify the specification
- Basis for scalable updates
- Analogies are not foolproof
  - If it works, great. If it doesn't, it may help
  - User can edit and fix the new version
- Improve by
  - Using domain knowledge
  - Learning from user feedback
The Need for Guidance in Workflow Design
VisComplete: A Workflow Recommendation System

- Mine provenance collection: Identify graph fragments that co-occur in a collection of workflows
- Predict sets of likely workflow additions to a given partial workflow
- Similar to a Web browser suggesting URL completions

[Koop et al., IEEE Vis 2008]
VisComplete: A Workflow Recommendation System

- Identify graph fragments that co-occur in a collection of workflows
- Predict sets of likely workflow additions to a given partial workflow
- Similar to a Web browser suggesting URL completions
VisComplete: Demo

http://www.cs.utah.edu/~juliana/videos/viscomplete_h_264.mov

VisComplete: Data-driven Suggestions for Visualization Systems

[Koop et al., IEEE Vis2008]
Results Summary

- Eliminates over 50% of actions
- Selected completions are almost always in the first four suggestions
- A database of simple pipelines can aid users constructing more complex pipelines
- See [Koop et al., TVCG 2008] for details on how the path database is constructed and on the completion algorithm
Conclusions and Future Work

- Appropriate support for exploratory tasks is essential for a wider adoption and more effective use of scientific workflow systems
- Provenance can be used to support reflective reasoning
- Intuitive interfaces for simplifying the construction and refinement of workflows
- Sharing workflows provenance at a large scale creates new opportunities
  - Workflow/provenance repositories; provenance-enabled publications
  - Scientists can learn by example; expedite their scientific training; and potentially reduce their time to insight [Freire and Silva, CHI SDA, 2008]
Acknowledgments: Funding

- This work is partially supported by the National Science Foundation, the Department of Energy, an IBM Faculty Award, and a University of Utah Seed Grant.
Acknowledgments: People

- VisTrails Group
  - Claudio Silva
  - Erik Anderson
  - Jason Callahan
  - Steven P. Callahan
  - Lorena Carlo
  - David Koop
  - Lauro Lins
  - Emanuele Santos
  - Carlos E. Scheidegger
  - Huy T. Vo
  - Geoff Draper
For more info about VisTrails

Visit: http://www.vistrails.org