Provenance for Computational Tasks: A Survey

Juliana Freire, David Koop, Emanuele Santos,
and Cláudio T. Silva
University of Utah

Victor Alencar {victoralencar@id.uff.br}
Apresentação para a disciplina **E-Science 2019.1**

Tópicos

- Definição
- Motivação
- Formas de Proveniência
- Componentes-Chave
- Captura de Informação
- Representação / Modelos
- Infraestrutura
- Panorama dos Sistemas
- Conclusão

Definição

- *"The source or origin of an object; its history and pedigree; a record of the ultimate derivation and passage of an item through its various owners."* **(The Oxford Dictionary)**
- Na ciência, ajuda a interpretar e entender resultados, a partir da análise da sequência de passos que levaram àquele resultado.

Motivação

- Capacidade e facilidade na reprodução de experimementos é uma grande vantagem ao se fazer ciência;
- Permitir reflexão para além das perguntas simples; e
- Tornar mais intuivo e abstratas análise de tarefas que antes demandavam grande esforço manual. Exemplo: script versus workflow.

Script vs Workflow

```
1  import vtk
2
3  data = vtk.vtkStructuredPointsReader()
4  data.setFileName("../../../examples/data/head.120.vtk")
5
6  contour = vtk.vtkContourFilter()
7  contour.SetInput(0, data.GetOutput())
8  contour.SetValue(0, 67)
9
10 mapper = vtk.vtkPolyDataMapper()
11 mapper.SetInput(contour.GetOutput())
12 mapper.ScalarVisibilityOff()
13
14 actor = vtk.vtkActor()
15 actor.SetMapper(mapper)
16
17 cam = vtk.vtkCamera()
18 cam.SetViewUp(0,0,-1)
19 cam.SetPosition(745,-453,369)
20 cam.SetFocalPoint(135,135,150)
21 cam.ComputeViewPlaneNormal()
22
23 ren = vtk.vtkRenderer()
24 ren.AddActor(actor)
25 ren.SetActiveCamera(cam)
26 ren.ResetCamera()
27
28 renwin = vtk.vtkRenderWindow()
29 renwin.AddRenderer(ren)
30
31 style = vtk.vtkInteractorStyleTrackballCamera()
32 iren = vtk.vtkRenderWindowIneractor()
33 iren.SetRenderWindow(renwin)
34 iren.SetInteractorStyle(style)
35 iren.Initialize()
36 iren.Start()
```

(a)

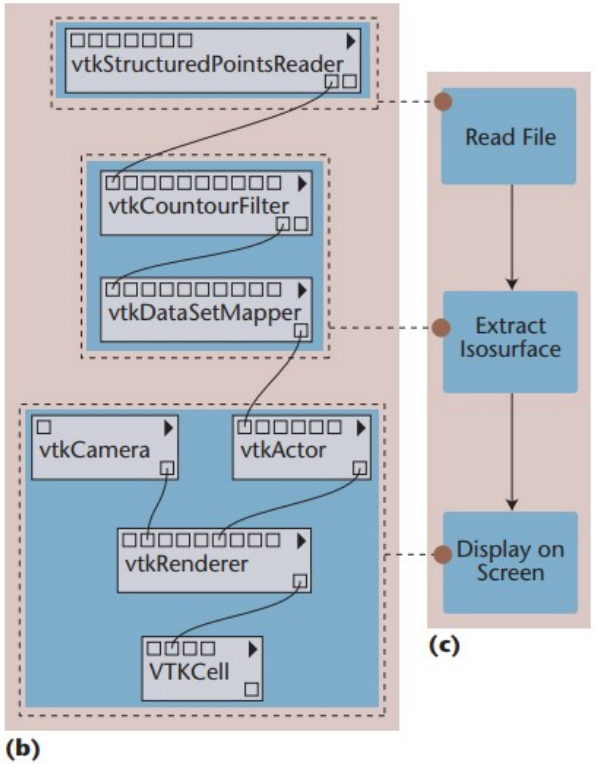
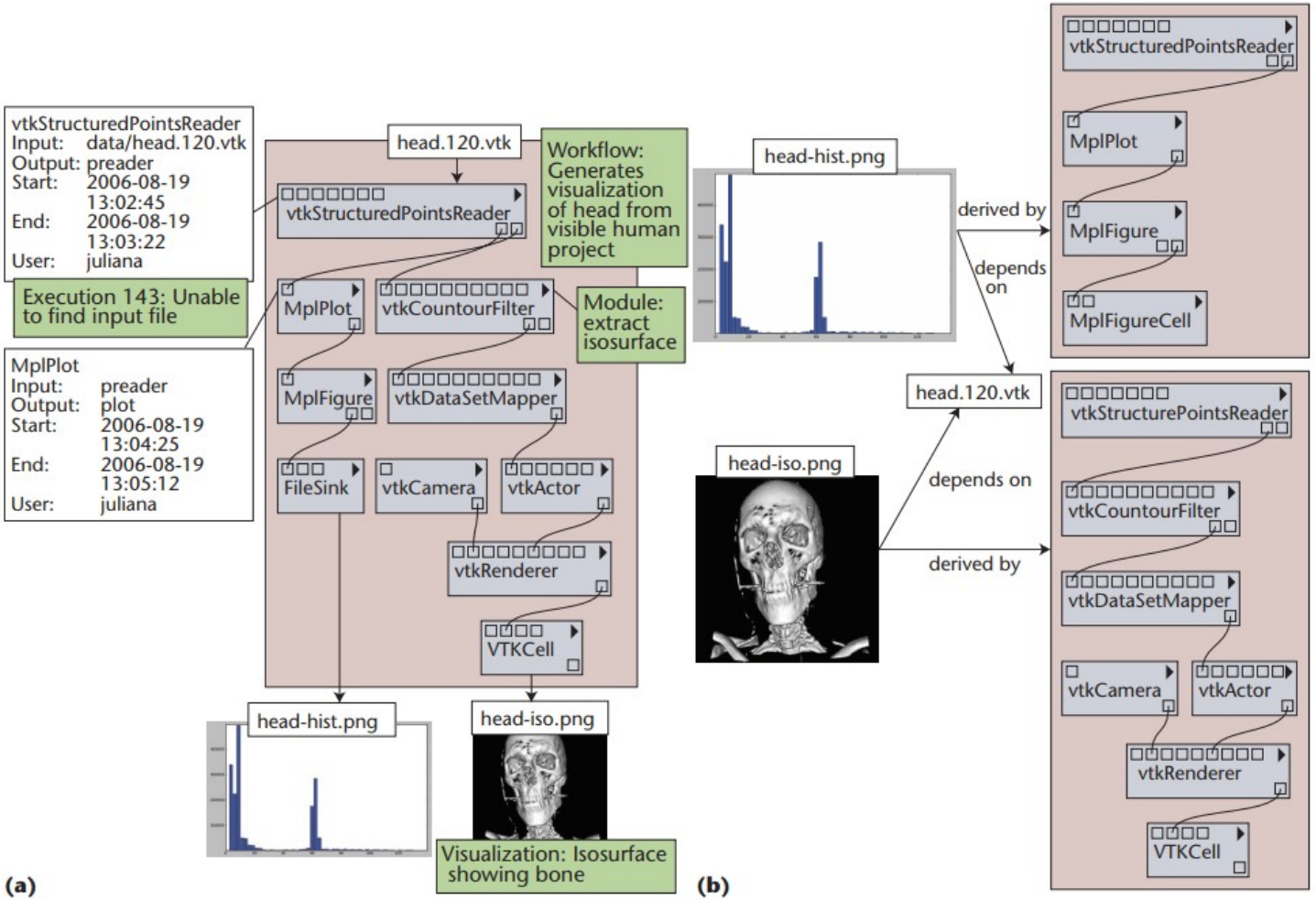


Figure 1. Different abstractions for a data flow. (a) A Python script containing a series of Visualization Toolkit (VTK) calls; (b) a workflow that produces the same result as the script; and (c) a simplified view of the workflow that hides some of its details.

Formas de Proveniência

- Prospectiva
  - Captura a especificação da tarefa computacional;
  - Quais passos são necessários para a tarefa?
  - Em que ordem?
- Retrospectiva
  - Captura os passos efetivamente executados;
  - Captura o contexto/ambiente no qual esses passos foram executados.
  - Quem? Quando? Alguma intercorrência?

Prospectiva vs Retrospectiva



Componentes-Chave

- Captura
  - Workflow;
  - Processo; e
  - Sistema Operacional.
- Representação/Modelo
  - Desafios de Proveniência
- Infraestrutura
  - Armazenamento;
  - Acesso e consultas;

Captura

- A etapa de captura de uma tarefa computacional precisa:
  - Acessar os passos executados;
  - Metadados(quem, quando, ambiente, etc) da execução;
  - Permitir anotações dos usuários;
  - Quaisquer outras informações relevantes para a tarefa;

Captura: Workflow

- Plugáveis ou integrados ao sistema de workflow;
- Vantagem: permitem captura direta por meio de APIs;
- Desvantagem: podem ser domain-biased;
- Exemplos: VisTrails, REDUX, Swift, Kepler, Taverna (bioinformática), Pegasus.

Captura: Processo

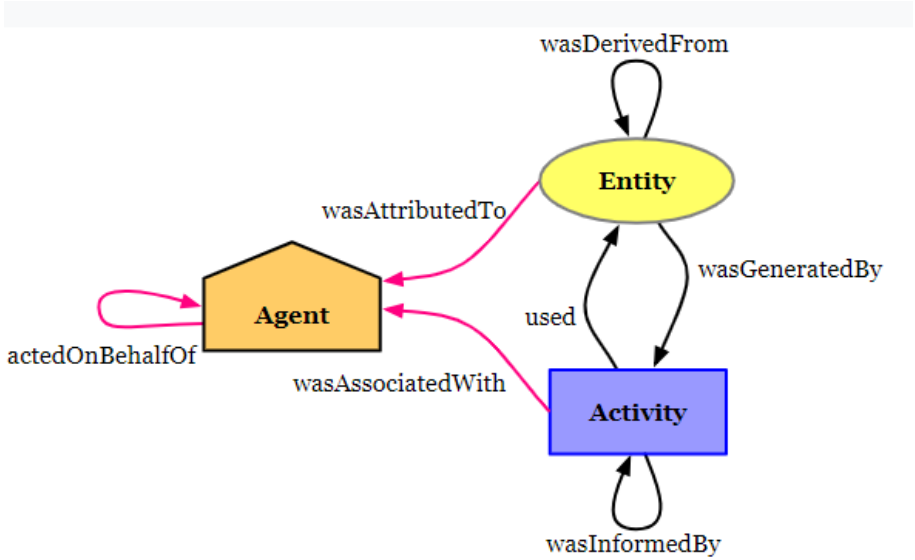
- Desvantagem: demandam algum esforço adicional;
- Cada serviço deve ser instrumentado para capturar proveniência;
- Vantagem: domain-bias mitigado pelo processo;
- Exemplos: Karma, PASOA/PreServ

Captura: SO

- Telemetria nativa do sistema operacional;
- Vantagem: não são acoplados a processos ou workflows, são agnósticos;
- Desvantagem: damandam pós-processamento para extrair informação, podem gerar muitos dados;
- Exemplos: PASS, ES3;

Representação/Modelo

- Vários esforços e modelos;
- Assim, surgiram os Desafios de Proveniência:
  - Quatro edições de desafios consolidaram o OPM:
  - https://openprovenance.org/



PROV Specifications

- Overview of PROV
- PROV Model Primer
- PROV-O
- PROV-DM
- PROV-N
- PROV-JSON
- PROV Constraints
- Provenance Working Group at W3C

Camadas dos Modelos

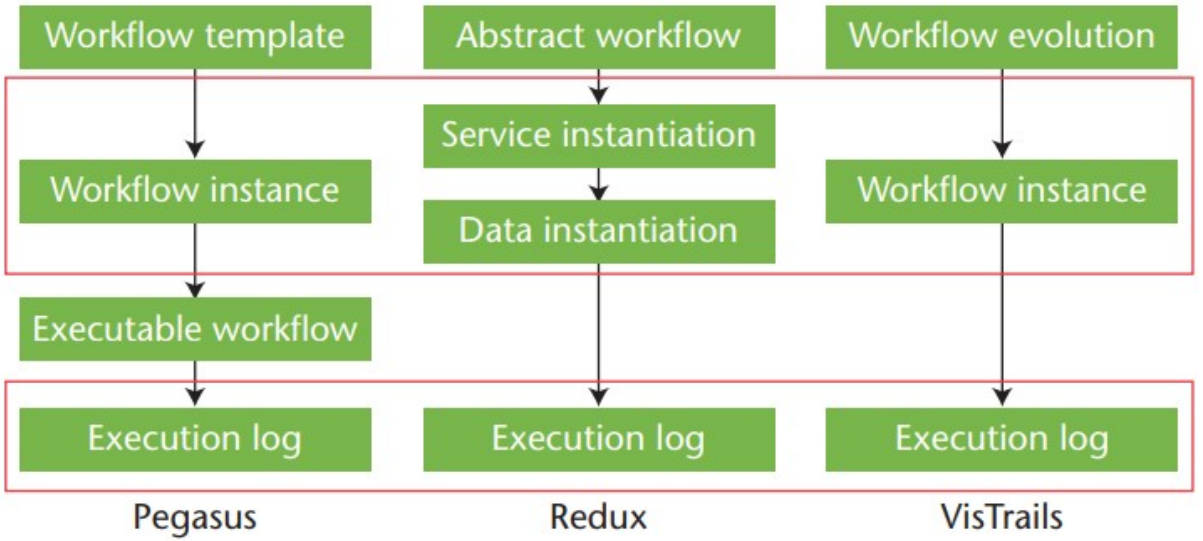


Figure 3. Layered provenance models. For REDUX, the first layer corresponds to an abstract description, the second layer describes the binding of specific services and data to the abstract description, the third layer captures runtime inputs and parameters, and the final layer captures operational data. Other models use layers in different ways. The top-layer in VisTrails captures provenance of workflow evolution, and Pegasus uses an additional layer to represent the workflow execution plan over grid resources.

Infraestrutura

- A infraestrutura encapsula:
  - Armazenamento;
  - Acesso; e
  - Consultas;
- Apenas recentemente pesquisadores começaram a dirigir atenção a essa área;
- Existência de trade-offs:
  - flexibilidade vs eficiência;
  - custo de armazenamento.

Infraestrutura

- Arquivos vs Bancos de Dados Relacionais
  - Arquivos não necessitam de infraestrutura adicional;
  - Bancos de dados Relacionais são centralizados, permitem fácil colaboração e são, em geral, mais eficientes.
- Efetividade e eficiência nas consultas à dados de proveniência transformaram-se em fator essencial;
- Sobrecarga de dados pode ser um problema para sistemas baseados em workflows.

Infraestrutura

- A capacidade de fazer consultas em proveniência possibilidade reuso de conhecimento;
- Consultas precisam ser eficientes, sobretudo quando os dados são massivos;
- Podem ser associados a linguagens de consultas como SQL, PROLOG, SPARQL;
- É comum que as informações de proveniência sejam associadas a imagens ou grafos;
- Web Semântica funde representação e consulta.

Panorama dos sistemas

Table 1. Provenance-enabled systems.

System	Capture mechanism	Prospective provenance	Retrospective provenance	Workflow evolution
REDUX	Workflow-based	Relational	Relational	No
Swift	Workflow-based	SwiftScript	Relational	No
VisTrails	Workflow-based	XML and relational	Relational	Yes
Karma	Workflow- and process-based	Business Process Execution Language	XML	No
Kepler	Workflow-based	MoML	MoML variation	Under development
Taverna	Workflow-based	Scufl	RDF	Under development
Pegasus	Workflow-based	OWL	Relational	No
PASS	OS-based	N/A	Relational	No
ES3	OS-based	N/A	XML	No
PASOA/PreServ	Process-based	N/A	XML	No

Conclusão

- Proveniência é uma área nova que avança rapidamente;
- Pesquisadores tem perseguido diversas direções dessa área:
  - Integração de proveniência com diferentes sistemas;
  - Melhoria de mecanismos de análise e visualização;
- Proveniência tem pavimentado caminhos para colaboração científica.