Revealing the Detailed History of Script Outputs with Hybrid Provenance Queries

Yang Cao¹, Duc Vu², Qiwen Wang¹, Qian Zhang¹, Priyaa Ramesh³, Timothy McPhillips¹, Paolo Missier³, Bertram Ludäscher¹

Data

Liniveristy of Illinois at Chicago, ³School of Computing Science, Newcastle University, UK



Motivation

- Data- and Workflow-Provenance are crucial for transparency and reproducibility in computational and data-driven science.
- Scientific workflow systems provide both prospective provenance (the workflow graph) and retrospective provenance (runtime observables).

Challenges

- Most computational analyses and workflows are conducted using **scripts** (Python, R, MATLAB, bash, ...) rather than workflow systems.
- Retrospective Provenance Observables at different levels only yield isolated fragments of the overall data lineage and processing history.
- Prospective Provenance could be used to link and contextualize fragments into a meaningful and comprehensible workflow, but scripts alone do not reveal the underlying workflow graph.
- Provenance (like other metadata) appears to be useful "for others".

Fine-Grained Prospective Provenance Fig 1.1 Upstream subgraph for output data Fig 1.3 (1) Q2-Pro only shows upstream inputs of the Fig 1.2 YesWorkflow model of a MATLAB script of C3C4 C3 fraction data in the YesWorkflow model for the C3_fraction_data; (2) Q4-pro only shows downstream MATLAB script of C3C4. outputs of mean airtemp. Q2_Pro: List the script inputs that are upstream of a given data product D. q2_pro('C3_fraction_data',mean_precip). q2_pro('C3_fraction_data',mean_airtemp). q2_pro('C3_fraction_data','SYNMAP_land_cover_map_data'). Q4_Pro: List the outputs that depend on a particular script Input (downstream) q4_pro(mean_airtemp,'C4_fraction_data'). Fig 2.3 (1) Q2-Pro only shows upstream inputs of the Fig 2.1 Upstream subgraph for output data shifted_wavefile in the YesWorkflow model for the Fig 2.2 YesWorkflow model of a Python script of LIGO shifted_wavefile; (2) Q4-pro only shows downstream Q2 Pro: List the script inputs that are upstream of a given data product D. q2 pro(shifted wavefile,fs). q2_pro(shifted_wavefile,'FN_Sampling_rate') q2_pro(shifted_wavefile,'FN_Detector'). Q4 Pro: List the outputs that depend on a particular script Input (downstream). q4_pro(fs,'ASDs'). q4_pro(fs,spectrogram). q4_pro(fs,filtered_white_noise_data). q4_pro(fs,'WHITENED_strain_data'). q4_pro(fs,'H1_strain_filtered'). q4_pro(fs,whitened_bandpass_wavefile). q4_pro(fs,'H1_strain_unfiltered'). q4_pro(fs,spectrogram_whitened). strain_H1_shifted strain_L1_shifted

Approach

Simple YesWorkflow (YW) annotations allow users to reveal workflow (prospective provenance graph) implicit in scripts.

- Prospective provenance queries to expose and test data dependencies at the workflow level
- Hybrid provenance queries that situate runtime observables (retrospective provenance) in the overall workflow, yielding meaningful knowledge artifacts.
- Easily share comprehensible workflow graphs and customizable provenance reports for script runs, along with data, code in scientific studies ("provenance for self").

Run Reconstruction

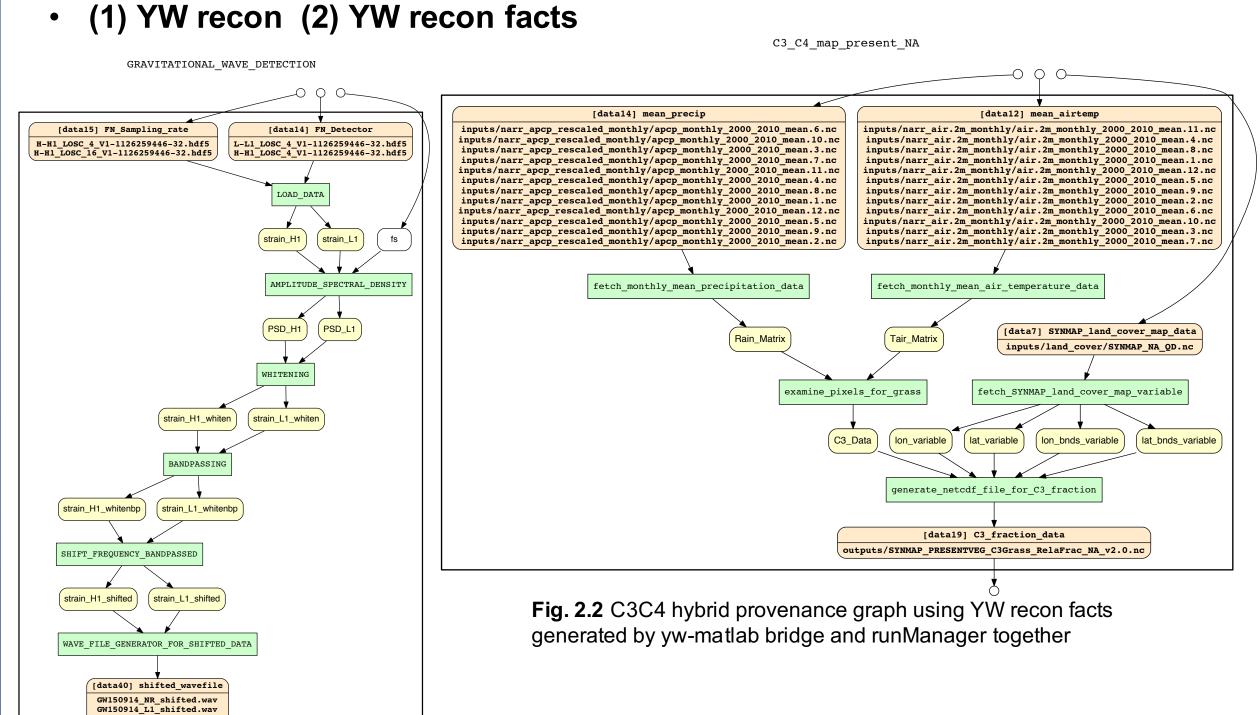
Fig. 2.1 LIGO hybrid provenance graph using yw-recon

- YW recon
- YW recon facts

Coarse and Fine-Grained Observations of Runs

- Matlab Run Manager -> list of files input or output
- YesWorkflow -> list of files matching @URI annotations
- noWorkflow -> values assigned to variables

Hybrid Queries for Fine-Grained Retrospective Provenance



>> mgr.view('runNumber', 2, 'sections', {'details', 'used', 'generated'}); JSED]: 25 Items used by this run 'SYNMAP NA QD.nc'

'52 MB' '15-Oct-2015 17:46:03' '15-Oct-2015 17:46:04' 'air.2m_monthly_2000_2010_mean.1.nc' '1 MB' 'air.2m_monthly_2000_2010_mean.2.nc' '1 MB' '15-Oct-2015 17:46:04 'air.2m monthly 2000 2010 mean.3.nc' '1 MB' '15-Oct-2015 17:46:04 'air.2m_monthly_2000_2010_mean.4.nc' '1 MB' '15-Oct-2015 17:46:04 'air.2m_monthly_2000_2010_mean.5.nc' '1 MB' '15-Oct-2015 17:46:04' 'air.2m monthly 2000 2010 mean.6.nc' '1 MB' '15-Oct-2015 17:46:04 'air.2m_monthly_2000_2010_mean.7.nc' '1 MB' '15-Oct-2015 17:46:04 'air.2m_monthly_2000_2010_mean.8.nc' '1 MB' '15-Oct-2015 17:46:04 'air.2m_monthly_2000_2010_mean.9.nc' '1 MB' '15-Oct-2015 17:46:04' 'air.2m_monthly_2000_2010_mean.10.nc' '1 MB' '15-Oct-2015 17:46:04' 'air.2m monthly 2000 2010 mean.11.nc' '1 MB' '15-Oct-2015 17:46:04' 'air.2m_monthly_2000_2010_mean.12.nc' '1 MB' '15-Oct-2015 17:46:04' 'apcp_monthly_2000_2010_mean.1.nc' '15-Oct-2015 17:46:04' 'apcp_monthly_2000_2010_mean.2.nc' '15-Oct-2015 17:46:04' 'apcp_monthly_2000_2010_mean.3.nc' '15-Oct-2015 17:46:04 '1 MB' 'apcp_monthly_2000_2010_mean.4.nc' '15-Oct-2015 17:46:04' 'apcp_monthly_2000_2010_mean.5.nc' '15-Oct-2015 17:46:04' 'apcp_monthly_2000_2010_mean.6.nc' '15-Oct-2015 17:46:04 'apcp_monthly_2000_2010_mean.7.nc' '15-Oct-2015 17:46:04' 'apcp_monthly_2000_2010_mean.8.nc' '15-Oct-2015 17:46:04' 'apcp_monthly_2000_2010_mean.9.nc' '15-Oct-2015 17:46:04' 'apcp_monthly_2000_2010_mean.10.nc' '1 MB' '15-Oct-2015 17:46:04' 'apcp_monthly_2000_2010_mean.11.nc' '1 MB' '15-Oct-2015 17:46:04' 'apcp_monthly_2000_2010_mean.12.nc' '1 MB' '15-Oct-2015 17:46:04' **GENERATED]: 7 Items generated by this run**

'SYNMAP_PRESENTVEG_C3Grass_RelaFrac_NA_v2.0.nc '1 MB' '09-Sep-2016 17:10:25' 'SYNMAP_PRESENTVEG_C4Grass_RelaFrac_NA_v2.0.nc' '1 MB' '09-Sep-2016 17:10:25' 'SYNMAP_PRESENTVEG_Grass_Fraction_NA_v2.0.nc' '1 MB' '09-Sep-2016 17:10:25'

Fig. 2.3 C3C4 RunManager Screenshot that shows the runtime file-level provenance observed by RunManager

Demo Queries

- Q1 (upstream prospective query): Render prospective upstream subgraph of the YW model of the script for a given output data
- Q2 (prospective query): **List** the script inputs that are upstream of a given data product D...
- Q3 (downstream prospective query): Render everything downstream of a given data product D, where D can be any one (input or intermediate) data element of the YW model of the script.
- Q4 (prospective query): **List** the script outputs that are downstream of a given data product D.
- Q5 (hybrid query): **Render** retrospective graph with with concrete filename for a given output data product D.

Conclusions and Future Work

- Provenance from script runs can be revealed graphically and made actionable (e.g., to yield customizable data lineage reports) via (1) simple YW user annotations, (2) linking runtime observables (e.g. DataONE RunManager, ReproZip, noWorkflow), and (3) sharing provenance artifacts and executable queries.
- Extend YW toolkit to support other workflow modeling constructs; to support graph pattern queries; to support project-level provenance.
- Evolve ProvONE to support project-level provenance and graph queries.

References

- Y Cao, D Vu, Q Wang, Q Zhang, P Ramesh, T McPhillips, P Missier, B Ludäscher (2016). DataONE AHM Provenance Demonstration: https://github.com/idaks/dataone-ahm-2016-poster
- YesWorkflow Project and Tools, https://github.com/yesworkflow-org
- T. McPhillips, T. Song, et al.(2015). YesWorkflow: A User-Oriented, Language-Independent Tool for Recovering Workflow Information from Scripts. Intl. Journal of Digital Curation 10, 298-313.
- T. McPhillips, S. Bowers, K. Belhajjame, B. Ludäscher (2015). Retrospective Provenance Without a Runtime Provenance Recorder. Workshop on the Theory and Practice of Provenance (TaPP).
- Cao, Y., Jones, C., Cuevas-Vicenttín, V., Jones, M.B., Ludäscher, B., McPhillips, T., Missier, et al., 2016, June. DataONE: A Data Federation with Provenance Support. Intl. Provenance and Annotation Workshop (IPAW). Springer.
- Pimentel, J.F., Dey, S., McPhillips, T., Belhajjame, K., Koop, D., Murta, L., Braganholo, V. and Ludäscher, B., 2016, June. Yin & Yang: demonstrating complementary provenance from noWorkflow & YesWorkflow. Intl. Provenance and Annotation Workshop (IPAW). Springer.

School of Information Sciences The iSchool at Illinois







