YesWorkflow: A User-Oriented, Language-Independent Tool for Recovering Workflow Information from Scripts

Timothy McPhillips¹, Tianhong Song², Tyler Kolisnik³, Steve Aulenbach⁴, Khalid Belhajjame⁵, Kyle Bocinsky⁶, Yang Cao¹, Fernando Chirigati⁷, Saumen Dey², Juliana Freire⁷, Deborah Huntzinger¹¹, Christopher Jones⁸, David Koop⁹, Paolo Missier¹⁰, Mark Schildhauer⁸, Christopher Schwalm¹¹, Yaxing Wei¹², James Cheney¹³, Mark Bieda³, Bertram Ludäscher¹,¹⁴

¹Graduate School for Library and Information Science (GSLIS), University of Illinois at Urbana-Champaign (UIUC); ²Dept. of Computer Science, University of California, Davis; ³University of Calgary; ⁴University Corporation for Atmospheric Research (UCAR) and U.S. Global Change Research Program (USGCRP); ⁵Paris Dauphine University, LAMSADE; ⁶Department of Anthropology, Washington State University, Pullman, WA; ⁷New York University; ⁸University of California, Santa Barbara; ⁹University of Massachusetts, Dartmouth; ¹⁰University of Newcastle, UK; ¹¹Northern Arizona University; ¹²Oak Ridge National Laboratory; ¹³University of Edinburgh, Scotland; ¹⁴National Center for Advanced Supercomputing Applications (NCSA), UIUC.
Overview

• **Enter Scientific Workflows ...**
  – Kepler, Taverna, VisTrails, ...

• **... Exeunt**

• **Enter Scripts ...**
  – Python, R, Matlab, ...

• **Enter YesWorkflow**
  – Complements noWorkflow
    • ... *runtime* (= *retrospective*) provenance from scripts
  – Combines the best of both worlds:
    • Scripts + Comments
      => **Workflow Views** (*prospective* provenance) from scripts
Scientific Workflows: ASAP!

- **Automation**
  - wfs to **automate** computational aspects of science

- **Scaling** (exploit and optimize **machine cycles**)
  - wfs should make use of **parallel compute resources**
  - wfs should be able handle **large data**

- **Abstraction, Evolution, Reuse** (**human cycles**)
  - wfs should be easy to (re-)**use, evolve, share**

- **Provenance**
  - wfs should capture **processing history, data lineage**
    - traceable data- and wf-evolution
  - **Reproducible Science**
Science Example: Paleoclimate Reconstruction

- **Kohler & Bocinsky**: study rain-fed maize of *Ancestral Pueblo, Anasazi*
  - Four Corners; AD 600–1500
  - *Climate change* influenced *Mesa Verde Migrations*; late 13th century AD.
  - Uses **network of tree-ring chronologies** to **reconstruct a spatio-temporal climate** field at a fairly high resolution (~800 m) from AD 1–2000
  - Algorithm estimates joint information in tree-rings and a climate signal to identify “best” tree-ring chronologies for reconstructing climate at a given time and place.


... implemented as an R Script ...
... Paleoclimate Reconstruction ...

Map showing the "selected" trees for reconstructing precipitation at four sites in our CAR regression approach (Correlation-Adjusted correlation).

Reconstructions for AD 1247

K. Bocinsky, T. Kohler, A 2000-year reconstruction of the rain-fed maize agricultural niche in the US Southwest. Nature Communications. doi:10.1038/ncomms6618
YesWorkflow = Scripts + Comments

• Scripts can be hard to digest, communicate

• Idea:
  – Add structured comments (cf. JavaDoc)

=> reveal workflow structure and dataflow

=> obtain some scientific workflow benefits

• ... ASAP ...
Get 3 views for the price of 1!

**Process view**

**Data view**

**Combined view**
User Comments: YW Annotations

```r
# @begin GO_Analysis
# @in hgCutoff @as GO_stats_p_value_cutoff
# @in higheridrlinkedtogenes @as DEG_list_higher_in_test_condition
# @in loweridrlinkedtogenes @as DEG_list_lower_in_test_condition
# @out gostatshigher @as GO_stats_gene_list_higher_in_test_condition
# @out BP_SummmH_File @as GO_stats_BP_higher_in_test_condition
# @out CC_SummmH_File @as GO_stats_CC_higher_in_test_condition
# @out MF_SummmH_File @as GO_stats_MF_higher_in_test_condition
# @out gostatslower @as GO_stats_gene_list_lower_in_test_condition
# @out BP_Summl_File @as GO_stats_BP_lower_in_test_condition
# @out CC_Summl_File @as GO_stats_CC_lower_in_test_condition
# @out MF_Summl_File @as GO_stats_MF_lower_in_test_condition

# GO_Analysis Block

# Gene Ontology Statistics are Calculated Here.

# Gene Ontology Categories that were shown to be relatively Higher (more expressed) in the Experimental Condition.
gostatshigher <- higheridrlinkedtogenes[1]
higherstatsfilename <- paste(outputDirectory, "/", runName, ".", conditions[1], "_GOStatsHigher_", mytestcond[1], "_vs_", baseline, "\n", write.table(gostatshigher, file=higherstatsfilename, row.names=FALSE, col.names=FALSE, quote=FALSE, sep="\t")
geneListHigherCHR <- gostatshigher$SYMBOL
geneListHigherLinkedtoEntrezIds <- select(hgu133plus2.db, keys= geneListHigherCHR, "ENTREZID", "SYMBOL")
```

@begin GO_Analysis

@in hgCutoff

@in ...

@out BP_Summl_file

@out ...

@end GO_Analysis

...
Paleoclimlate Reconstruction ...

- ... explained using YesWorkflow

Kyle B., (computational) archeologist:
"It took me about 20 minutes to comment. Less than an hour to learn and YW-annotate, all-told."
YesWorkflow Architecture: KISS!

- **YW-Extract**
  - ... structured comments

- **YW-Model**
  - Program Block, Workflow
  - Port (data, parameters)
  - Channels (dataflow)

- **YW-Graph**
  - ... using GraphViz/DOT files

- **YW-Query, YW-Validate, YW-CLI**
Gene Expression Microarray Data Analysis

- **[Normalize]**
  - Normalization of data across microarray datasets
- **[SelectDEGs]**
  - Selection of differentially expressed genes between conditions
- **[GO Analysis]**
  - Determination of gene ontology statistics for the resulting datasets
- **[MakeHeatmap]**
  - Creation of a heatmap of the differentially expressed genes.

Tyler Kolisnik, Mark Bieda
Multi-Scale Synthesis and Terrestrial Model Intercomparison Project (MsTMIP)

Christopher Schwalm, Yaxing Wei
Summary: Scientific Workflows

Scientific Workflows

- [+] Automation
- [+] Scalability
- [+] Abstraction
- [+] Provenance
- ...
- [+/-0] Easy to use
  - [0] learning a new paradigm
- [-] Teaching resources
- [-] Special expertise needed for deep changes
e.g. new Java actors, shims, ...

B. Ludäscher
Summary: Scripts + YesWorkflow

Scripts: [+] Automation, [0] Scalability, [-] Abstraction, [0/-] Provenance

Now: Scripts + YesWorkflow Annotations

- [+] Abstraction
  - explain your methods to mere mortals
  => encourage (re-)use

- [+] Provenance:
  - noWorkflow (retrospective provenance)
  - YesWorkflow (prospective provenance)

- [+] Language independent (R, Matlab, Python, ...)

- [+] Empower tool makers (script programmers): give them ...
  - ... some immediate benefits (workflow views)
  - ... some medium term improvements (provenance integration)
  - ... some long term benefits: think about your methods differently
  => dataflow programming => [+ ] Scalability
YesWorkflow: Acknowledgements

- With support from NSF awards DBI-1356751 (*Kurator*), ACI-0830944 (*DataONE*), SMA-1439603 (*SKOPE*).

- Thanks to the IDCC organizers for choice of venues!