Increasing Portability of Scientific Workflows with Linking

Casey Robinson and Douglas Thain
University of Notre Dame

Abstract
Dependency management remains a major challenge for all forms of software. A program implemented in a given environment typically has many implicit dependencies on programs, libraries, and other objects present within that environment, so that is virtually guaranteed to break when moved to another environment. Workflows particularly suffer from dependency management problems, precisely because they tie together multiple independent programs into a coherent whole. To address this problem, we propose applying the old idea of a linker into the new context of workflow systems. We have implemented a linker for the Makeflow workflow system, and extended the concept to apply recursively to executables and scripted languages within the workflow. We evaluate the system by applying it to a selection of bioinformatics workflows including BLAST, BWA, and SHRiMP, enabling them to be moved across multiple computation environments.


Solution

Known Working Environment

Blast.mf
input
job.params
blastx

Database

finish.sh
dist.sh
split.pl
localtime
$PATH
perl

scp

Workflow Linker

Blast.bundle
input
job.params
blastx

Database

finish.sh
dist.sh
split.pl
localtime
$PATH
perl

Where to get it?
http://www.nd.edu/~ccl

or
https://github.com/cooperative-computing-lab/ctools

References
5. Short Read Mapping Package http://compbio.cs.toronto.edu/shrimp/

Acknowledgement
This work was supported in part by the Department of Energy and the National Science Foundation via grants OCI-1148330 and CBET-0941565

Additional Examples
BWA, Biocompute[6], Matlab, SHRiMP, SSAHA

Challenges
- Which files should be collected?
- Minimizing data transfer
- Requirements for scripting languages

Biocompute
http://biocompute.cse.nd.edu

Workflow

linker

input
job.params
localtime
$PATH
perl

Additional Examples
BWA, Biocompute[6], Matlab, SHRiMP, SSAHA

Challenges
- Which files should be collected?
- Minimizing data transfer
- Requirements for scripting languages