

WorkflowGenerator

Workflow Generator

To facilitate evaluation of workflow algorithms and systems on a range of workflow sizes, we have developed a set of synthetic workflow generators. These generators use the information gathered from actual executions of scientific workflows on the Grid as well as our understanding of the processes behind these workflows to generate realistic, synthetic workflows resembling those used by real world scientific applications.

The code used to generate all of the synthetic workflows below, and many others, is available from [the GitHub repository](#). The java workflow generator sometimes generates negative task runtimes, so watch out for that.

Simulator

[WorkflowSim](#) can be used to simulate the workflows generated by the Workflow Generator.

Traces

Traces and execution logs from real workflows are available here: [here](#), [here](#), and [here](#). Data sets like these were used to parameterize the Workflow Generator.

Synthetic Workflows

Pegasus Workflows

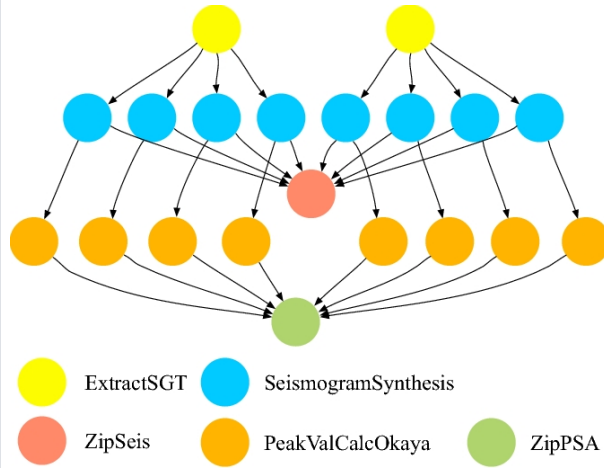
These workflows come from a [paper by Bharathi, et al. \[1\]](#). There is [another paper with more information about the workflows by Juve, et al. \[2\]](#).

[A large collection of DAXes similar to the ones listed below is available here](#). Note that it is about 375 MB.

Workflow Type	Example	DAX
Montage The Montage application created by NASA/IPAC stitches together multiple input images to create custom mosaics of the sky.	 <ul style="list-style-type: none">mProjectPPmDiffFitmConcatFitmBgModelmBackgroundmImgTblmAddmShrinkmJPEG	25 Node DAX 50 Node DAX 100 Node DAX 1000 Node DAX

CyberShake

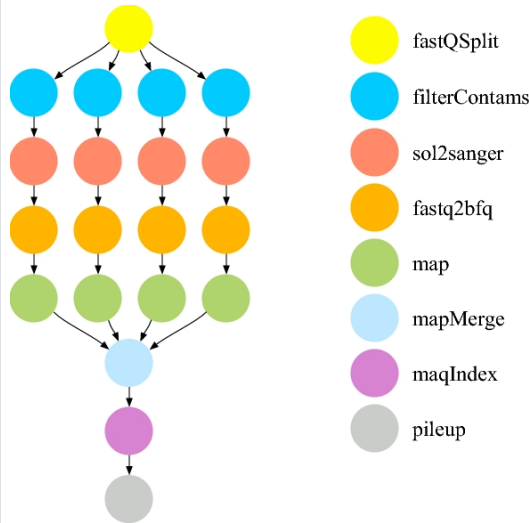
The CyberShake workflow is used by the Southern California Earthquake Center to characterize earthquake hazards in a region.



30 Node DAX
50 Node DAX
100 Node DAX
1000 Node DAX

Epigenomics

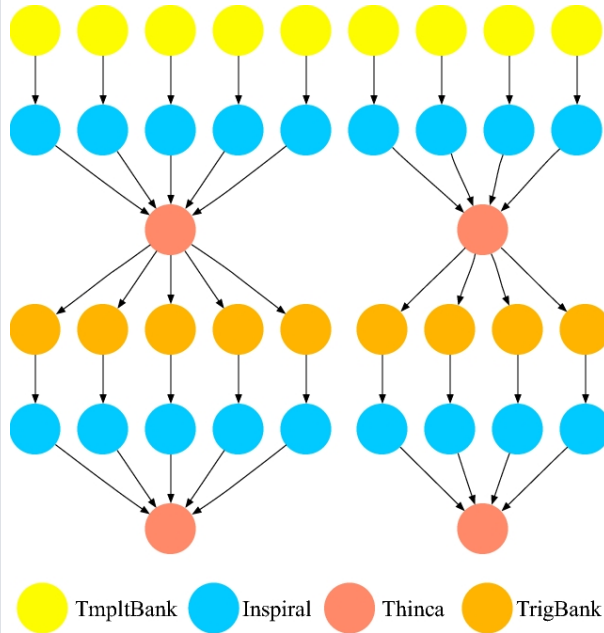
The epigenomics workflow created by the USC Epigenome Center and the Pegasus Team is used to automate various operations in genome sequence processing.



24 Node DAX
46 Node DAX
100 Node DAX
997 Node DAX

LIGO Inspiral Analysis

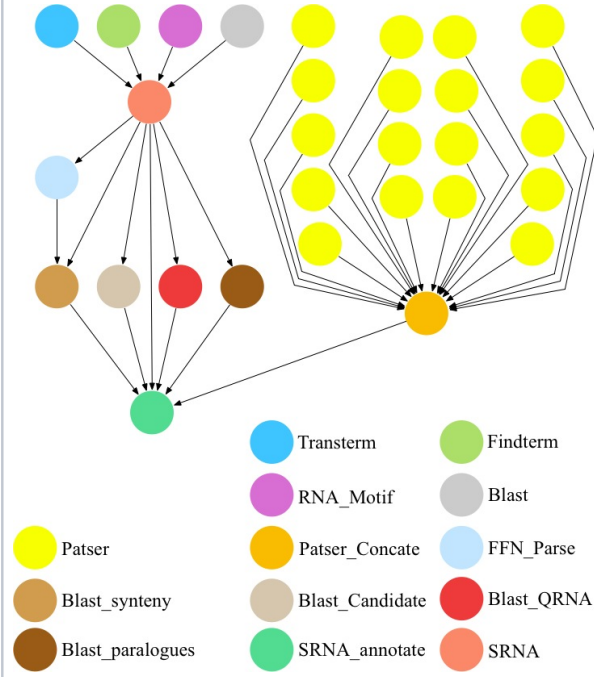
LIGO's Inspiral Analysis workflow is used to generate and analyze gravitational waveforms from data collected during the coalescing of compact binary systems.



30 Node DAX
50 Node DAX
100 Node DAX
1000 Node DAX

SIPHT

The SIPHT workflow, from the bioinformatics project at Harvard, is used to automate the search for untranslated RNAs (sRNAs) for bacterial replicons in the NCBI database.

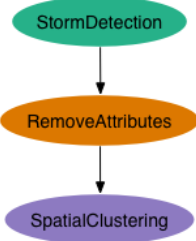
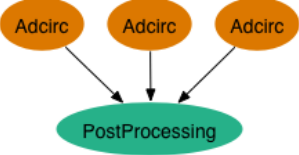
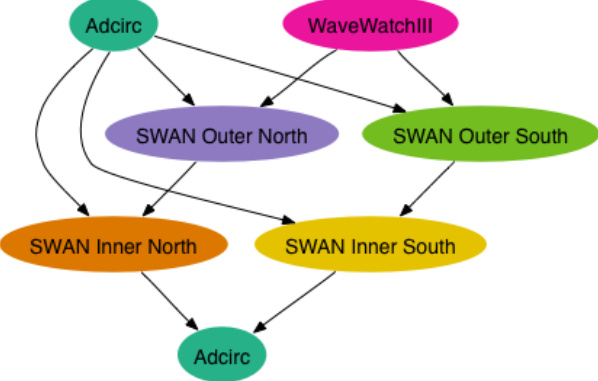
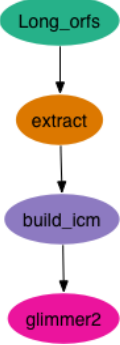
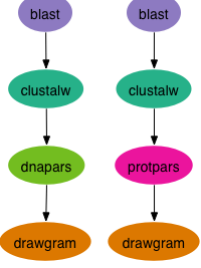
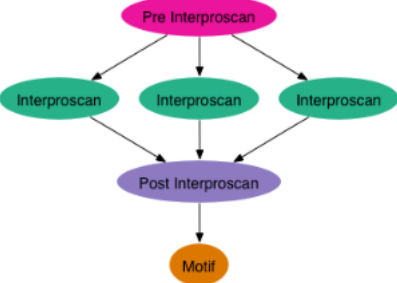


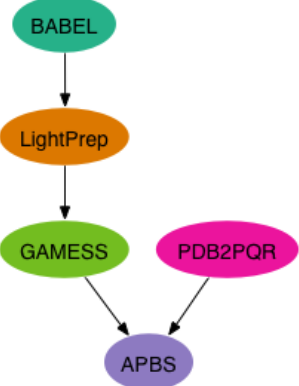
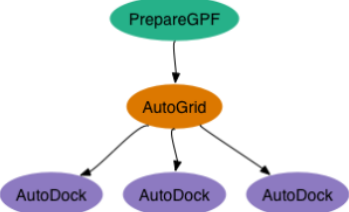
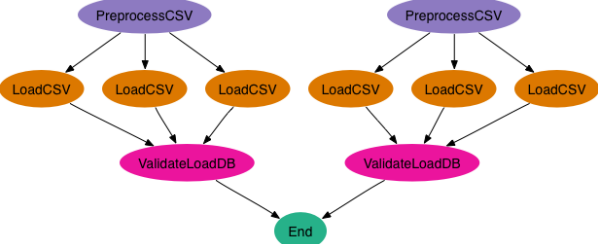
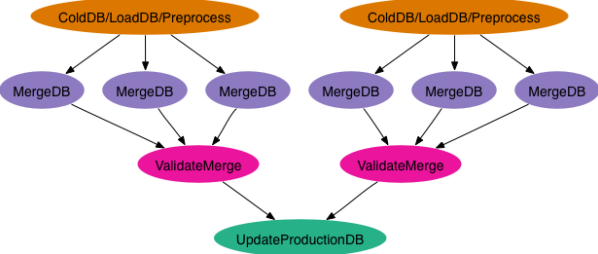
30 Node DAX
60 Node DAX
100 Node DAX
1000 Node DAX

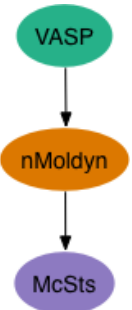
Ramakrishnan and Gannon Workflows

These workflows come from a [report](#) by Ramakrishnan and Gannon [3].

Workflow Type	Figure in Report	Example	DAX
LEAD Mesoscale Meteorology	Figure 1		leadmm.xml
LEAD ARPS Data Analysis System	Figure 2		leadadas.xml

LEAD Data Mining Workflow	Figure 3	 <pre> graph TD A(StormDetection) --> B(RemoveAttributes) B --> C(SpatialClustering) </pre>	leaddm.xml
Storm Surge SCOOP Workflow	Figure 4	 <pre> graph TD A1(Adcirc) --> C(PostProcessing) A2(Adcirc) --> C A3(Adcirc) --> C </pre>	scoop_small.xml scoop_medium.xml scoop_large.xml
Floodplain Mapping	Figure 5	 <pre> graph TD A1(Adcirc) --> B(SWAN Outer North) A1 --> C(SWAN Outer South) A1 --> D(SWAN Inner North) A1 --> E(Adcirc) F(WaveWatchIII) --> B F --> C B --> D C --> E D --> E </pre>	floodplain.xml
Glimmer	Figure 6	 <pre> graph TD A(Long_orfs) --> B(extract) B --> C(build_icm) C --> D(glimmer2) </pre>	glimmer.xml
Gene2Life	Figure 7	 <pre> graph TD A1(blast) --> B1(clustalw) B1 --> C1(dnapars) C1 --> D1(drawgram) A2(blast) --> B2(clustalw) B2 --> C2(protpars) C2 --> D2(drawgram) </pre>	gene2life.xml
Motif Network	Figure 8	 <pre> graph TD A(Pre Interproscan) --> B1(Interproscan) A --> B2(Interproscan) A --> B3(Interproscan) B1 --> C(Post Interproscan) B2 --> C B3 --> C C --> D(Motif) </pre>	motif_small.xml motif_medium.xml motif_large.xml

MEME-MAST	Figure 9	 <pre> graph TD MEME((MEME)) --> MAST((MAST)) </pre>	mememast.xml
Molecular Sciences	Figure 10	 <pre> graph TD BABEL((BABEL)) --> LightPrep((LightPrep)) LightPrep --> GAMESS((GAMESS)) LightPrep --> PDB2PQR((PDB2PQR)) GAMESS --> APBS((APBS)) PDB2PQR --> APBS </pre>	molsci.xml
Avian Flu	Figure 11	 <pre> graph TD PrepareGPF((PrepareGPF)) --> AutoGrid((AutoGrid)) AutoGrid --> AutoDock1((AutoDock)) AutoGrid --> AutoDock2((AutoDock)) AutoGrid --> AutoDock3((AutoDock)) </pre>	avianflu_small.xml avianflu_medium.xml avianflu_large.xml
caDSR	Figure 12	 <pre> graph TD findProjects((findProjects)) --> findClassesInProjects((findClassesInProjects)) findProjects --> findSemanticMetadata((findSemanticMetadata)) findClassesInProjects --> findSemanticMetadata findSemanticMetadata --> searchLogicConcept((searchLogicConcept)) </pre>	cadsr.xml
Pan-STARRS Load	Figure 13	 <pre> graph TD subgraph Path1 P1[PreprocessCSV] --> L1[LoadCSV] P1 --> L2[LoadCSV] P1 --> L3[LoadCSV] end subgraph Path2 P2[PreprocessCSV] --> L4[LoadCSV] P2 --> L5[LoadCSV] P2 --> L6[LoadCSV] end L1 --> V1[ValidateLoadDB] L2 --> V1 L3 --> V1 L4 --> V2[ValidateLoadDB] L5 --> V2 L6 --> V2 V1 --> End((End)) V2 --> End </pre>	psload_small.xml psload_medium.xml psload_large.xml
Pan-STARRS Merge	Figure 14	 <pre> graph TD subgraph Path1 C1[ColdDB/LoadDB/Preprocess] --> M1[MergeDB] C1 --> M2[MergeDB] C1 --> M3[MergeDB] end subgraph Path2 C2[ColdDB/LoadDB/Preprocess] --> M4[MergeDB] C2 --> M5[MergeDB] C2 --> M6[MergeDB] end M1 --> V1[ValidateMerge] M2 --> V1 M3 --> V1 M4 --> V2[ValidateMerge] M5 --> V2 M6 --> V2 V1 --> U[UpdateProductionDB] V2 --> U </pre>	psmerge_small.xml psmerge_medium.xml psmerge_large.xml

McStas	Figure 15	 <pre>graph TD; VASP([VASP]) --> nMoldyn([nMoldyn]); nMoldyn --> McSts([McSts]);</pre>	mcstas.xml
--------	-----------	---	----------------------------

[1] R. F. da Silva, W. Chen, G. Juve, K. Vahi, E. Deelman. Community Resources for Enabling Research in Distributed Scientific Workflows. 10th IEEE International Conference on e-Science (eScience 2014)

[2] S. Bharathi, A. Chervenak, E. Deelman, G. Mehta, M.-H. Su, and K. Vahi, "Characterization of Scientific Workflows", 3rd Workshop on Workflows in Support of Large Scale Science (WORKS 08), 2008.

[3] Gideon Juve, Ann Chervenak, Ewa Deelman, Shishir Bharathi, Gaurang Mehta, and Karan Vahi, "Characterizing and Profiling Scientific Workflows", *Future Generation Computer Systems*, 29:3, pp. 682–692, March 2013.

[4] L. Ramakrishnan and D. Gannon, "A Survey of Distributed Workflow Characteristics and Resource Requirements", Indiana University Technical Report TR671, 2008.