Meeting the Challenges of Managing Large-Scale Scientific Workflows in Distributed Environments

Ewa Deelman
Yolanda Gil
USC Information Sciences Institute
Scientific Workflows

- Current workflow approaches are exploring specific aspects of the problem:
  - Creation, reuse, provenance, performance, reliability

- New requirements are emerging
  - Streaming data, from batch to interactive steering, event-driven analysis, collaborative design of workflows

- Need to develop a science of workflows
  - A more comprehensive treatment of workflow lifecycle
  - Understand current and long-term requirements from science applications
    - reproducibility
  - Workflows as first-class citizens in CyberInfrastructure
Workflow Lifecycle

- **Data Products**
  - Adapt, Modify
  - Workflow and Component Libraries
  - Workflow Template
  - Populate with data
  - Workflow Instance
  - Workflow Execution
  - User Experiences
  - Data, Metadata, Provenance Information
  - Scheduling/Execution
  - Compute, Storage and Network Resources
  - Planning
  - Mapping to available resources
  - Resource, Application Component Descriptions

Ewa Deelman
www.isi.edu/~deelman
Outline

- Rendering the workflow lifecycle
  - Wings/Pegasus/DAGMan
- Challenges across the various aspects of workflow management
  - User experiences
  - Planning/Mapping
  - Execution
- Workflows—what are they good for?
- Research issues
- Conclusions
Workflow Lifecycle

Data Products

WINGS

Adapt, Modify

Workflow and Component Libraries

Workflow Template

Workflow Instance

Map to available resources

Executable Workflow

Resource, Application Component Descriptions

Data, Metadata, Provenance Information

Execute

DAGMan

Compute, Storage and Network Resources

WINGS

Populate with data

Pegasus

Data, Metadata Catalogs
WINGS/Pegasus: Workflow Instance Generation and Selection, Using semantic technologies for workflow generation

“Show me workflows that generate hazard maps”

“Run that with the USGS data set”

“Validate this workflow based on the component specs”

“Here is a new wave propagation model, takes in a series of fault ruptures, is compiled for MPI”

Wings for Pegasus: A Semantic Approach to Creating Very Large Scientific Workflows
Yolanda Gil, Varun Ratnakar, Ewa Deelman, Marc Spraragen, and Jihie Kim, *OWL: Experiences and Directions* 2006
Pegasus: Planning for Execution in Grids

- Maps from workflow instance to executable workflow
- Automatically locates physical locations for both workflow components and data
- Finds appropriate resources to execute the components
- Augments the workflow with data staging and registration
- Reuses existing data products where applicable
- Publishes newly derived data products
Condor DAGMan (University of Wisconsin)

- Follows dependencies in workflow
- Releases nodes to execution (to Condor Q)
- Provides retry capabilities

```
executing
waiting
done OK
```
Challenges in user experiences

- Users’ expectations vary greatly
  - High-level descriptions
  - Detailed plans that include specific resources
- Users interactions can be exploratory
  - Modifying portions of the workflow as the computation progresses
- Users need progress, failure information at the right level of detail
Portals, Providing high-level Interfaces

Earthworks Project (SCEC), lead by with J. Muench P. Maechling, H. Francoeur, and others


TG Science Gateway, Washington University
SCEC CyberShake Workflow, not a one shot workflow

Needs to run before rest of the workflow is instantiated
Iterative workflow instantiation, mapping and execution

Wings for Pegasus: A Semantic Approach to Creating Very Large Scientific Workflows
Yolanda Gil, Varun Ratnakar, Ewa Deelman, Marc Spraragen, and Jihie Kim, in submission
Some challenges in workflow mapping

- Automated management of data
  - Through workflow modification
- Efficient mapping the workflow instances to resources
  - Performance
  - Data space optimizations
  - Fault tolerance (involves interfacing with the workflow execution system)
    - Recovery by replanning
    - plan “B”
- Providing feedback to the user
  - Feasibility, time estimates
Execution Environment

Pegasus

DAGMan

Condor Q

LOCAL SUBMIT HOST

Condor-G

Condor-C

Condor-G

GRAM

PBS

LSF

Condor

GridFTP

HTTP

Storage

Ewa Deelman, deelman@isi.edu
www.isi.edu/~deelman
pegasus.isi.edu
Node clustering

- **Vertical clustering**
- **Level-based clustering**
- **Arbitrary clustering**

Useful for small granularity jobs

Ewa Deelman, deelman@isi.edu

www.isi.edu/~deelman

pegasus.isi.edu
Montage application
~7,000 compute jobs in instance
~10,000 nodes in the executable workflow
same number of clusters as processors
speedup of ~15 on 32 processors

Small 1,200 Montage Workflow

Total Time (in minutes) for the end-to-end execution of the concrete DAG for M16 6 degrees at NCSA cluster

Wall Clock Time (minutes)

no. of processors
Efficient data handling

- Input data is staged dynamically, new data products are generated during execution
- For large workflows 10,000+ files

- Similar order of intermediate and output files
- Total space occupied is far greater than available space—failures occur

Solution:
- Determine which data is no longer needed and when
- Add nodes to the workflow do cleanup data along the way

Issues:
- minimize the number of nodes and dependencies added so as not to slow down workflow execution
- deal with portions of workflows scheduled to multiple sites
- deal with files on partition boundaries
- Benefits: preliminary results show up to 50% space improvements for a gravitational-wave physics applications
Challenges in Workflow Execution

- Provide fault tolerance
  - Mask errors, Interact with the workflow planner
- Support resource provisioning
- Provide monitoring information
- Providing execution-level provenance
- Support debugging
  - Provide workflow traces for easy replay
Southern California Earthquake Center (SCEC) workflows on the TeraGrid

Gurmeet Singh et al. “Application-level Resource Provisioning”, Wednesday, M15, 14:30-16:00 session

SCEC on the TeraGrid Fall 2006

Number of jobs per day (23 days), 261,823 jobs total, Number of CPU hours per day, 15,706 hours total (1.8 years), 10TB of data
Benefits of Scientific Workflows
(from the point of view of an application scientist)

- Conducts a series of computational tasks.
  - Resources distributed across Internet.
- Chaining (outputs become inputs) replaces manual hand-offs.
  - Accelerated creation of products.
- Ease of use - gives non-developers access to sophisticated codes.
  - Avoids need to download-install-learn how to use someone else's code.
- Provides framework to host or assemble community set of applications.
  - Honors original codes. Allows for heterogeneous coding styles.
- Framework to define common formats or standards when useful.
  - Promotes exchange of data, products, codes. Community metadata.
- Multi-disciplinary workflows can promote even broader collaborations.
  - E.g., ground motions fed into simulation of building shaking.
- Certain rules or guidelines make it easier to add a code into a workflow.

Slide courtesy of David Okaya, SCEC, USC
Workflows for education and sharing

- Application specialists design individual application components
- Domain experts compose workflows using application components
  - Set correct parameters for components
  - Pick appropriate data sets
- Students run sophisticated workflows on training data sets
- Young researchers run sophisticated workflows on data sets of interest to them
- Scientist share workflows across collaborations to validate a hypothesis
- Need to develop tools, workflow libraries, component libraries
Current and Future Research

- Resource selection
- Resource provisioning
- Workflow restructuring
- Adaptive computing
  - Workflow refinement adapts to changing execution environment
- Workflow provenance (including provenance of the mapping process) – new collaboration with Luc Moreau
- Management and optimization across multiple workflows
- Workflow debugging
- Streaming data workflows
- Automated guidance for workflow restructuring
- Support for long-lived and recurrent workflows
General Conclusions

- Workflows are recipes for CyberInfrastructure
- Need to support the dynamic nature of science
- Support for long-lived and recurrent workflows
- Many challenges and many workflow tools out there
  - Interoperability is desired
- Need common representations that can be used by various workflow management systems
  - Maybe semantic technologies?
- Need common provenance tracking capabilities
  - See IPAW 06, and the Provenance Challenge
- To make forward progress
  - collaboration with application scientists is essential
  - collaboration between workflow system designers is essential
Scientific Workflows—a very active area

- Many workshops
- Book on e-Science Workflows (Taylor, Deelman, Gannon, Shields eds.) *to appear 2006*
- Bill Gate’s SC 2005 Keynote
Acknowledgments

- Pegasus is being developed at ISI by Gaurang Mehta, Mei-Hui Su, and Karan Vahi
  - [http://pegasus.isi.edu](http://pegasus.isi.edu)
- Wings is lead by Yolanda Gil, Jihie Kim, Varun Ratnakar
  - [www.isi.edu/ikcap/wings/](http://www.isi.edu/ikcap/wings/)
- DAGMan is lead by Miron Livny
  - [www.cs.wisc.edu/condor/](http://www.cs.wisc.edu/condor/)
- Many application scientists made the workflows happen (GriPhyN, NVO, LIGO, Telescience, SCEC)