Scientific workflow management a way to enable e-science on both Grids and Clouds

Adam Belloum
Outline

- Introduction
- Lifecycle of an e-science workflow
- Workflow management Systems
- Scientific workflows Applications
- Provenance
- Examples of Scientific workflow managements
Parallel programming

In the previous lecture we have discussed one way to create parallel and distributed programs:

- need to do something to your program to use multiple processors
- need to incorporate commands into your program which allow multiple threads to run
  - one thread per processor
  - each thread gets a piece of the work

- several ways (APIs) to do this …
  - MPI
  - OpenMP
  - Web services
A workflow is a model to represent a reliably repeatable sequence of operations/tasks by showing explicitly the interdependencies among them.

Source: **SigWin-Detector workflow** has been developed in the VL-e project to detect ridges in for instance a Gene Expression sequence or Human transcriptome map, BMC Research Notes 2008, 1:63 doi:10.1186/1756-0500-1-63.
Workflow management system

- Workflow management system is a computer program that manages the execution of a workflow on a set of computing resources.

The user interface of the WS-VLAM a workflow management system developed in the VL-e project to execute application workflow on geographically distributed computing resources.

Deployed as service on Dutch super Computer (DAS3), and Dutch NGI (BigGrid) Clusters
Distributed enabled workflow engines

Workflow composition
Workflow execution

Grid /cloud Middleware:
Data management
High level Data management services
Network & storage Resources

Grid /cloud Middleware:
Process & resource management
Workflow Engine service
other Services
Application server
Web Service Interface

Source: Bob Hertberger keynote talk at 2nd IEEE Conf on eScience & grid computing, Amsterdam 2006
The automation of a **business process**, in whole or parts, where **documents, information or tasks** are passed from one participant to another to be processed, according to a set of **procedural rules**. (WFMC definition of a Workflow)
Workflow Management systems can use various types of computing resources

Standalone Computers, Clusters, Grids and clouds

- **co-allocate** resources needed for workflow enactment across multiple domains?

- achieve **QoS** for data centric application workflows that have special requirements on network connections?

- achieve **Robustness** and **fault tolerance** for workflow running across distributed resources?

- increase **re-usability** of Workflow, workflow components, and refine workflow execution?
Real World Workflows

Epigenomics

https://confluence.pegasus.isi.edu/display/pegasus/WorkflowGenerator

LIGO Inspiral Analysis

https://confluence.pegasus.isi.edu/display/pegasus/WorkflowGenerator
Real World Workflows

(Structured) fork-join pattern
Structured: All branches of one fork are merged at one join

Epigenomics

https://confluence.pegasus.isi.edu/display/pegasus/WorkflowGenerator

CyberShake

LIGO Inspiral Analysis

https://confluence.pegasus.isi.edu/display/pegasus/WorkflowGenerator
Business vs Scientific Workflows (Similarities)

• Capturing knowledge/best practices
  – Capture business process based on the company policy
  – Capture best practices of scientist, expert from a specific domain

• Series of structured activities and computations
  – Both involves repeated execution of certain procedures, and both describes tasks within this procedures.

• Incorporate human decision in the process
  – There are exceptional cases that can not be automated both in business and scientific workflow

Source: http://www.csc.ncsu.edu/faculty/mpsingh/papers/databases/workflows/sciworkflows.html
Business vs Scientific Workflows (Differences)

Business Process
- Information, task, procedural rules of a certain company
- Driven by business profit goals

- **Static Procedures**
  - Reflecting certain policy within a company
  - Rigid, any changes require approval from management

- **Closed Environment**
  - Managed own resources
  - Within company, actual organization

- **Documents, task descriptions**
  - Flight reservation, credit approval, supply chain, billing, resource planning

Scientific process
- Data analysis, experiment, data manipulation recipes
- Driven by problem solving goal

- **Dynamic**
  - Exploratory and speculative
  - Flexible, scientist manage their own business (they are their own user/manager).

- **Open Environment**
  - Non Centralized grid environment
  - Across boundary, Virtual Organizations

- **Large Data**
  - High energy physics data, bioinformatics micro array/ genomic data etc.
What makes workflow management systems useful?

- workflow management systems offer a number of services:
  - **large** data flows support
  - **parameterize** execution of large number of jobs
  - monitor and control workflow execution including **ad-hoc** changes
  - execute in **dynamic** environment where resources are not known a priori and may need to adapt to changes
  - Support hierarchical **execution** with sub-workflows created and destroyed when necessary
What makes workflow management systems useful?

Provenance/ reproducibility

- **Provenance**: The recording of metadata and provenance information during the various stages of the workflow lifecycle

- “A complete provenance record for a data object allows the possibility to reproduce the result and reproducibility is a critical component of the scientific method”

wave propagation model applications

[Biomedical engineering Cardiovascular biomechanics group TUE]

wave propagation model of blood flow in large vessels using an approximate velocity profile function:

a biomedical study for which 3000 runs were required to perform a global sensitivity analysis of a blood pressure wave propagation in arteries

Query interface for the provenance data collected from 3000 simulations of the “wave propagation model of blood flow in large vessels using an approximate velocity profile function”
What makes workflow management systems useful?

Help in most of the Scientific experiments lifecycle phases

1. Problem investigation:
   - Look for relevant problems
   - Browse available tools
   - Define the goal
   - Decompose into steps

2. Experiment Prototyping:
   - Design experiment workflows
   - Develop necessary components

3. Experiment Execution:
   - Execute experiment processes
   - Control the execution
   - Collect and analysis data

4. Results Publication:
   - Annotate data
   - Publish data

Shared repositories

Source: A.S.Z. Belloum, Vladimir Korkhov, Spiros Koulouzis, Marcia A Inda, and Marian Bubak Collaborative e-Science experiments: from scientific workflow to knowledge sharing JULY/AUGUST, IEEE Internet Computing, 2011
workflow management systems ...

- Applications
  - Stream oriented applications
  - Data parallel application
  - Parameter sweep applications

- Infrastructure
  - Desktops
  - Clusters
  - Grids
  - Clouds

- Storage
  - Federated Cloud Storage

- Scaling
  - Automatic Task farming for grid jobs and web services
  - MapReduce ...

- Provenance
  - Open Provenance model
  - Xml history Tracing

source: https://ivi.fnwi.uva.nl/sne/wsvlam2/
Example of Scientific workflow

Image processing workflow

Core modules are Matlab functions

Input: set of images
Output: histogram distance

Images converted to different color spaces

Histogram difference is calculated between color spaces
Example of Scientific workflow

Color coded workflow to better understand the result graphs

- Directory Reader
- Normalize
- Converter1
- Parameters
- Histogram Difference
- Converter2
- Results
Workflow Without Scaling

Slow task causing a **bottleneck** in the workflow
Bottleneck task in Scientific workflow

Part of workflow **stalled** because of bottleneck in workflow

Scaling Converter1 can circumvent the bottleneck

Bottleneck causes other tasks to lay idle waiting for data
Zooming into the Task Converter 1

Data organized in atomic parcels (messages)

Converter 1
Zooming into the Task Converter 1

Converter 1

Data organized in atomic parcels (messages)

Task processes data sequentially
Scaling Concepts

Data organized in atomic parcels (messages)

Task processes data sequentially

Converter 1

Part of workflow stalled because of bottleneck in workflow

Scaling Converter 1 can circumvent the bottleneck

Bottleneck causes other tasks to lay idle waiting for data
Zooming into the Task Converter 1

Data organized in atomic parcels (messages)

Task processes data sequentially

Converter 1
Zooming into the Task Converter 1

Data organized in atomic parcels (messages)

Tasks processes data **concurrently**

Adding more tasks increases `message consumption` rate

Part of workflow stalled because of bottleneck in workflow

Scaling Converter1 can circumvent the bottleneck

Bottleneck causes other tasks to lay idle waiting for data
Zooming into the Task Converter 1

Data organized in atomic parcels (messages)

Task processes data sequentially

Adding more tasks increases message consumption rate

Challenge: How many tasks to create?

Too many and tasks get stuck on queues. Too few and optimal performance not achieved.
Load Prediction

$\text{Task} + \text{Time Slot} = x \text{ seconds}$

$\text{Task} = k \text{ seconds}$

Simplified Load $= \frac{6x}{k}$ time slots

Assumption: Size of data directly proportional to computation time. May not always be the case
Workflow execution with Scaling
Workflow execution with Scaling
Workflow execution with Scaling Task - I
Workflow execution with Scaling Task -2
ImageCollector was set to a **fixed** amount (4)
Auto Scaling Task -2

HistogramDifference was set to one-to-one scaling. Each parameter generates a new task.
Resource management

• Within a single workflow services are competing for resources.
• Scaling one service without any regard to the whole workflow may starve parts of the workflow and hamper progress.
• It would be ideal to have a mechanism to greedily consume resources if no one is using them but donate back resources once they are requested.
• Some workflow management systems might also help.
Scale with the increase of input load

Running Service instances

Workflow as a Service (WFaaS)

- Once a workflow is initiated on the resources it stays alive and process data/jobs continuously
- Reduce the scheduling overhead

Workflow management systems useful features

• **Workflow description**
  How to **capture knowledge** of expert while still **hiding** complexity of underlying system.

  – **Workflow Models**: allow to **model** the tasks and **dependencies** between them (DAG, Petri Net)

  – **Workflow languages**: provide the required support to express the workflow model.

• **Workflow Enactment**: The functions provided by enactment are **scheduling**, **fault management** and **data movement**.

  – In the context of Grid environment workflow enactment service can be built on the top of low level Grid middleware
Workflow management systems useful features

• **Workflow Refinement**
  – Modification from the workflow description
  – **Reduction** of workflow if some data already exist
  – Additional data movement preparation if needed

• **Mapping** to actual resource
  – Resource discovery, allocation and management
  – **Bind** to real computing resource

• **Workflow Fault Tolerance & Monitoring of Execution**
  – Two level failure recovery techniques
    • Task Level
    • Workflow Level
Workflow management systems useful features: (Model of computation)

- Model of computation: stream-based process network.
  - Engine co-allocates all workflows.
  - Components waste time idling.
  - Co-allocation difficult.

- Communication: time coupled
  - Assumes components are running
  - Simultaneously
  - Synchronized p2p
  - Fixed TCP/IP

Model of computation

- Model of computation: dataflow network
  - components scheduled depending on data
  - components only activated when data is available
  - no need for co-allocation

- Communication: time decouples
  - messaging communication system.
  - components not synchronized
  - communication not strictly TCP/IP
Workflow Taxonomy

For application workflows using Grid/cloud resources,
- the input files of tasks need to be staged to a remote site before processing the task.
- Similarly, output files may be required by their children tasks which are processed on other resources.

The intermediate data has to be staged out to the corresponding Grid/Cloud sites.

Source: A Taxonomy of Workflow Management Systems for Grid Computing
Component Based Workflow Description: Triana

Feature Highlights

- Modular Java Workflow Environment
- Triana comes with a wide variety of built-in tools.
  - There is an extensive signal-analysis toolkit, an image-manipulation toolkit, a desk-top publishing toolkit, and many more
- Triana Cloud Job Queuing
- Sophisticated Drag & Drop Composition
- Web Services
- Comprehensive Toolbox Libraries

Source [http://www.trianacode.org](http://www.trianacode.org)
Component Based Workflow Description: Kepler

- Kepler provides a graphical user interface and a run-time engine that can execute workflows either from within the graphical interface or from a command line.

- Kepler workflows can be nested, allowing complex tasks to be composed from simpler components.

- Kepler workflows can leverage the computational power of grid technologies (e.g., Globus, SRB, Web Services)

- Kepler workflows and customized components can be saved, reused, and shared with colleagues using the Kepler archive format (KAR)

- Kepler ships with a searchable library containing over 350 ready-to-use processing components ('actors') that can be easily customized,

Source: https://kepler-project.org
Program/Application workflow Based: Taverna

- Access to local tools/scripts and remote resources and analysis tools, Web
- Not restricted to predetermined services – rapid incorporation of new services without coding
- Up-to-date R support (version 3.1.0)
- Excel and csv spreadsheet support Interaction with a running workflow from your Web browser
- Creating and sharing workflow fragments as reusable components
- Standards-compliant workflow run provenance collection

Source http://www.taverna.org.uk
Pegasus

- **Portability / Reuse**
  - Pegasus mapper can reorder, group, and prioritize tasks in order to increase overall workflow performance
- **Scalability**
  - from just a few computational tasks up to 1 million
- **Provenance**
  - all jobs are launched using a wrapper that captures runtime provenance of the job and helps in debugging
- **Data Management**
  - handles replica selection, data transfers and output registrations in data catalogs
- **Reliability**
  - Jobs and data transfers are automatically retried in case of failures

Source: [http://pegasus.isi.edu](http://pegasus.isi.edu)
Pumpkin

- Automatic scaling of workflow components based
  - Resource load
  - Application load
  - Provenance data

- Scaling across various infrastructures
  - Desktop
  - Grids
  - Clouds

Source: https://ivi.fnwi.uva.nl/sne/wsvlam2/?page_id=36
History-tracing XML (FH Aachen)

• provides data/process provenance following an approach that
  – maps the workflow graph to a layered structure of an XML document.
  – This allows an intuitive and easy processable representation of the workflow execution path,
  – which can be, eventually, electronically signed.

Blast Application

The aim of the application is the alignment of DNA sequence data with a given reference database. A workflow approach is currently followed to run this application on distributed computing resources.

For Each workflow run
The provenance data is collected and stored following the XML-tracing system
User interface allows to reproduce events that occurred at runtime (replay mode)
User interface can be customized (User can select the events to track)
User Interface show resource usage

[Department of Clinical Epidemiology, Biostatistics and Bioinformatics (KEBB), AMC]
Summary

• Workflow research especially in the grid environments are rapidly growing research subject

• VOs in Grid can benefits from the experience of workflows in the business community

• Scientific Workflow in Grid Environment have their own characteristics that need to be dealt with new approach

• Scientific Workflow research is highly related with various other research topics: resource management, fault tolerance, application performance, ontology.
More References

1. A.S.Z. Belloum, V. Korkhov, S Koulouzis, M. A Inda, and M. Bubak Collaborative e-Science experiments: from scientific workflow to knowledge sharing JULY/AUGUST, IEEE Internet Computing, 2011

2. Ilkay Altintas, Manish Kumar Anand, Daniel Crawl, Shawn Bowers, Adam Belloum, Paolo Missier, Bertram Ludascher, Carole A. Goble, Peter M.A. Sloot, Understanding Collaborative Studies Through Interoperable Workflow Provenance, IPAW2010, Troy, NY, USA

