# UVA HPC & BIG DATA COURSE

## 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

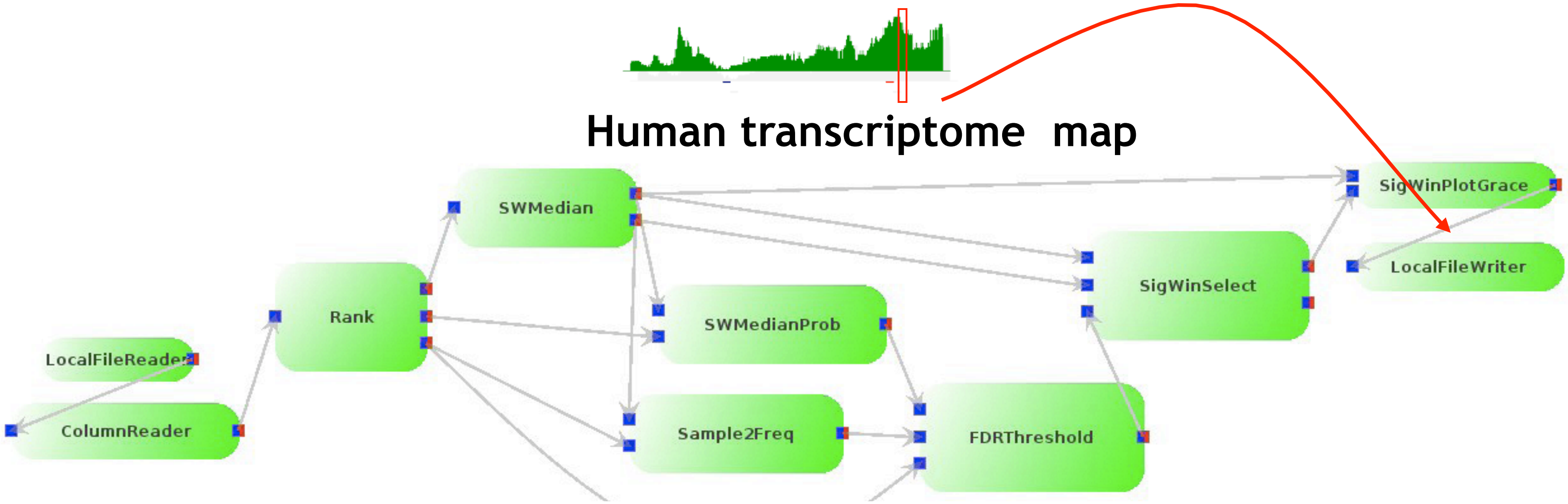
In the previous lecture we have discussed one way to create parallel and distributed programs:

- 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

# (Scientific) Workflow

A workflow is a model to represent a reliably repeatable sequence of operations/tasks by showing explicitly the interdependencies among them.

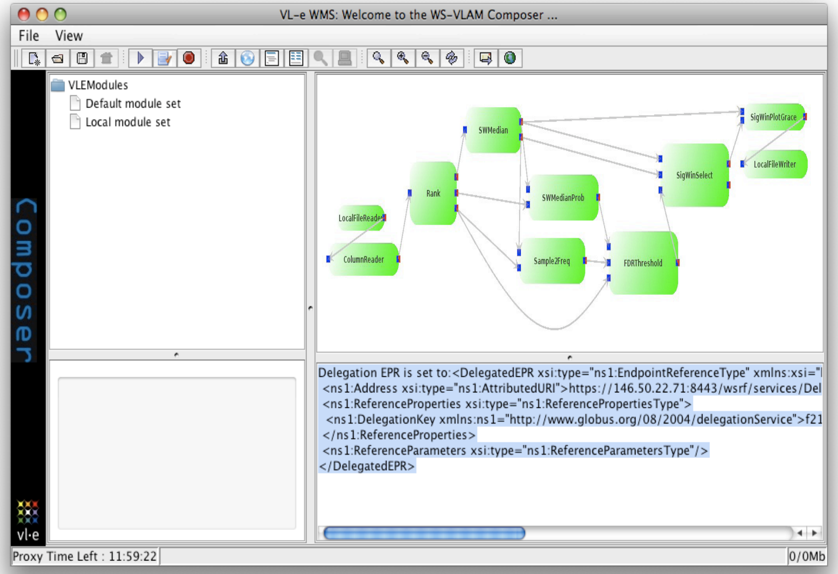



http://www.youtube.com/watch?v=R6bTFrzaR_w&feature=player_embedded

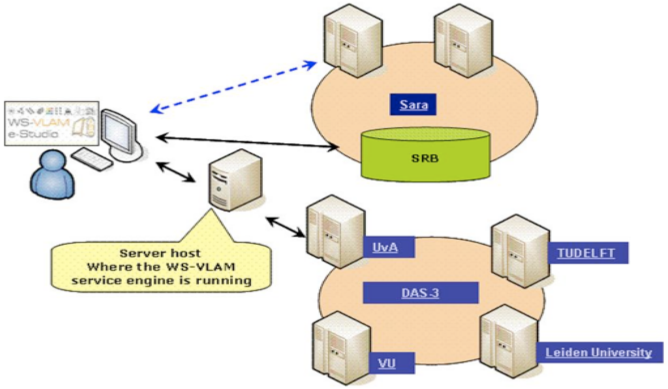
*Source: **SigWin-Detector workflow*** has been developed in the VL-e project to detect ridges in for instance a Gene Expression sequence or Human transcriptome map, BMC Research Notes 2008, 1:63 doi:10.1186/1756-0500-1-63.

# Workflow management system

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

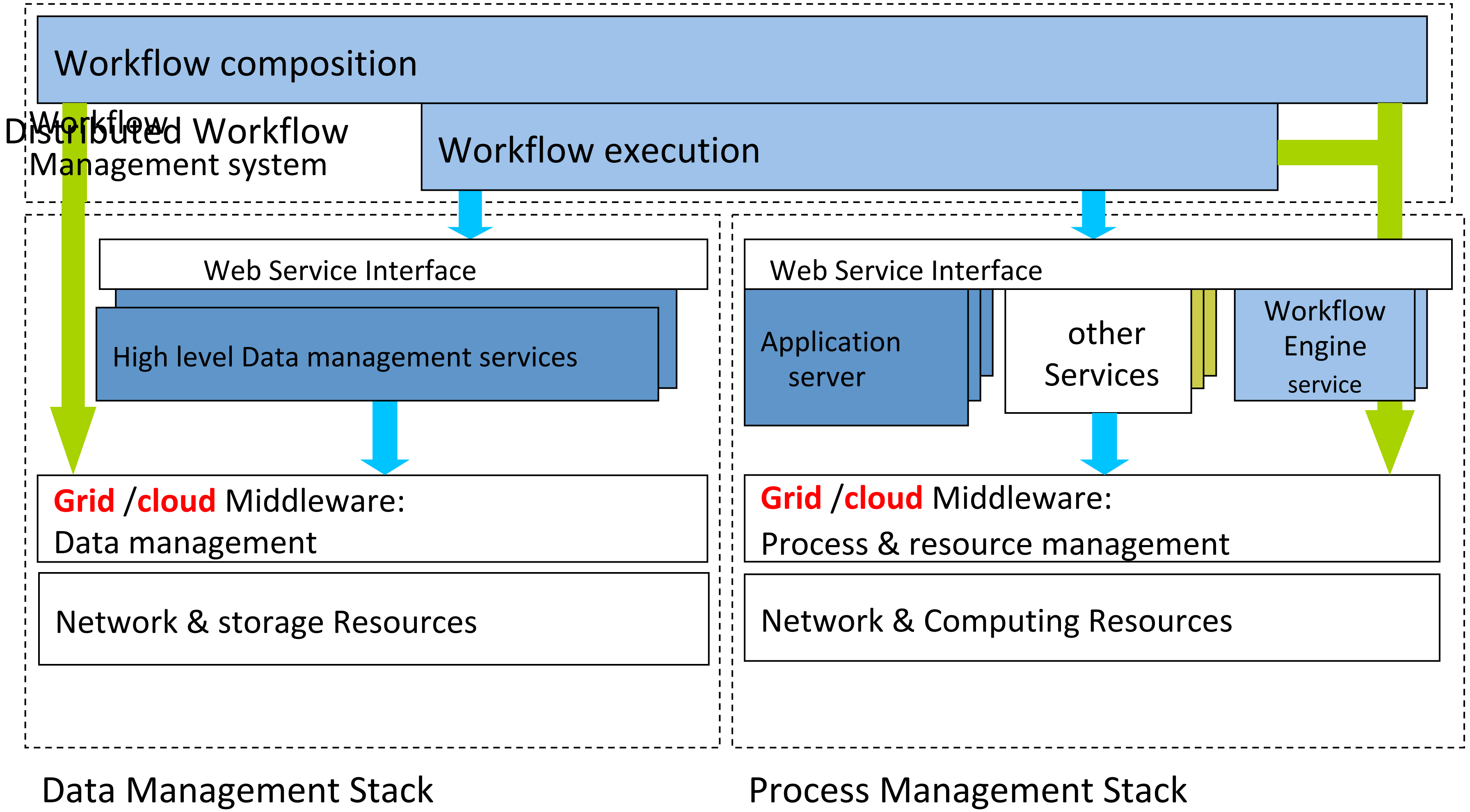



***The user interface of the WS-VLAM*** a workflow management system developed in the VL-e project to execute application workflow on geographically distributed computing resources



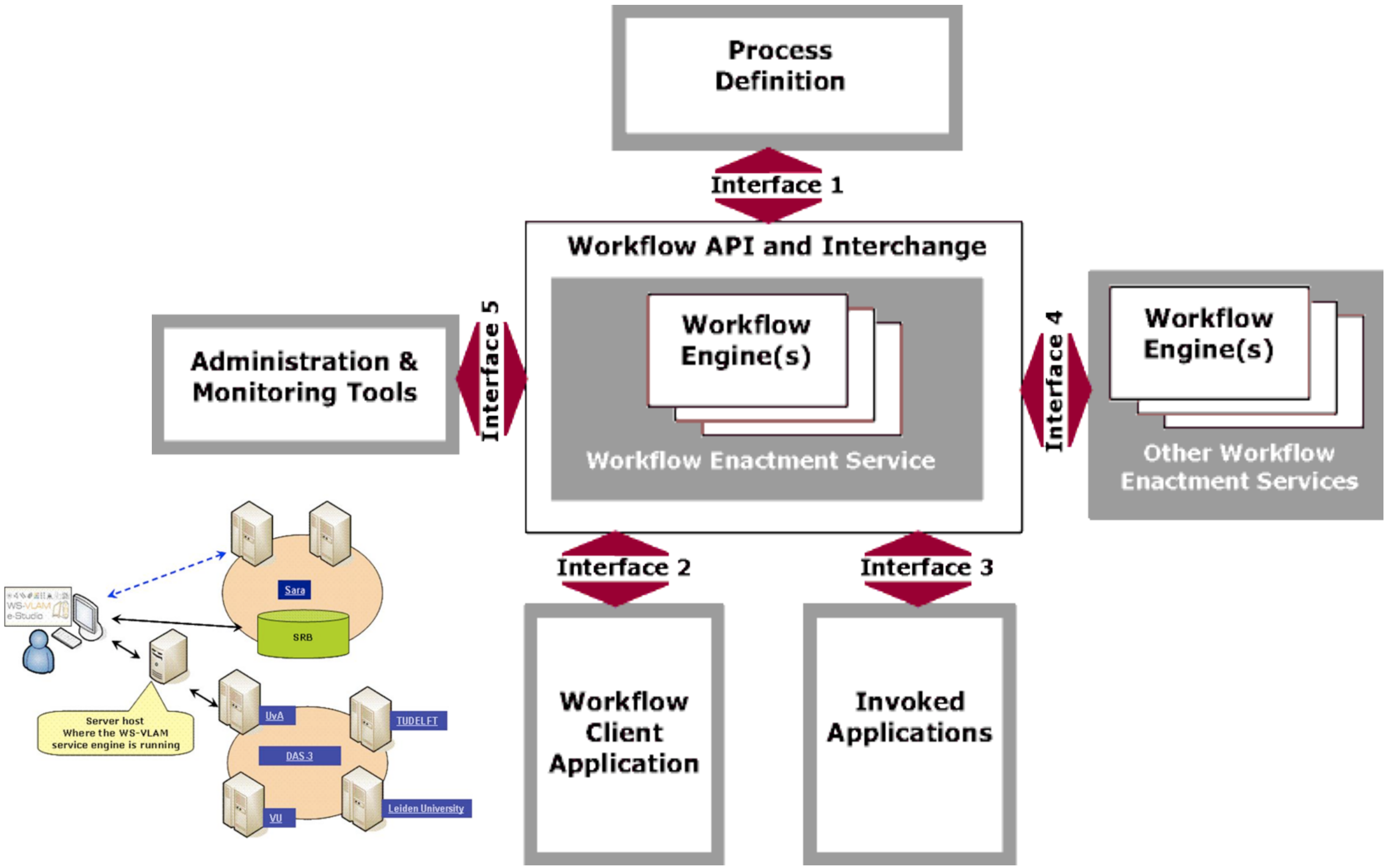

Deployed as service on Dutch super Computer (DAS3), and Dutch NGI (BigGrid) Clusters

# Distributed enabled workflow engines




Source: Bob Hertberger keynote talk at *2nd IEEE Conf on eScience & grid computing , Amsterdam 2006*

# Reference Model From WFMC




*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)*

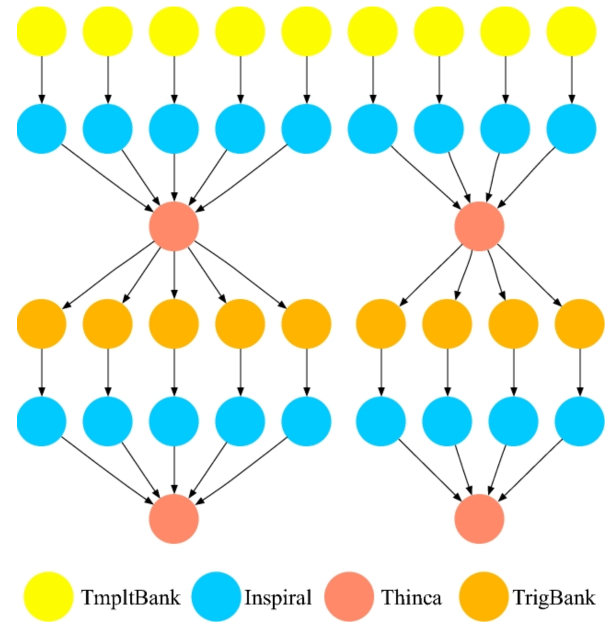
# Workflow Management systems can use various types of computing resources

Standalone Computers, Clusters, Grids and clouds

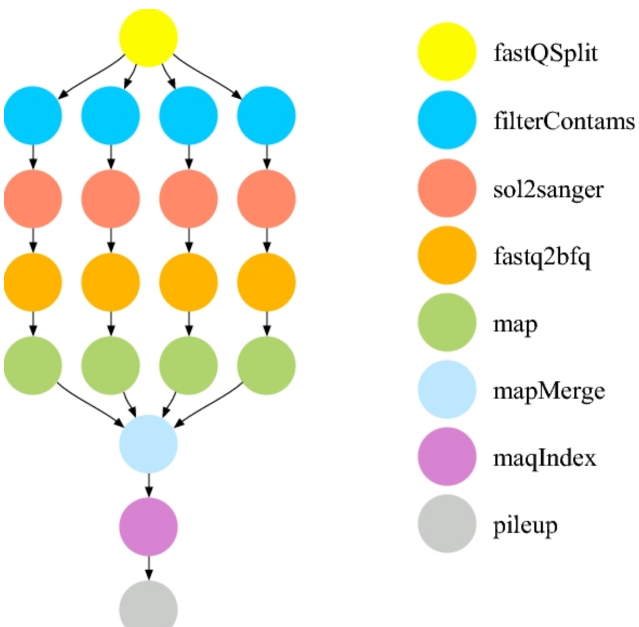
- co-allocate resources needed for workflow enactment across multiple domains?
- achieve QoS for data centric application workflows that have special requirements on network connections?
- achieve Robustness and fault tolerance for workflow running across distributed resources?
- increase re-usability of Workflow, workflow components, and refine workflow execution?

# Real World Workflows

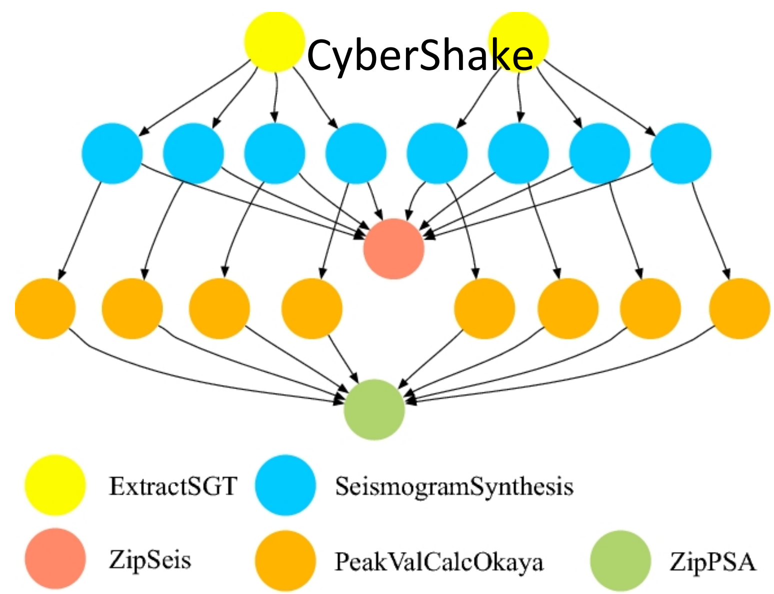
LIGO Inspiral Analysis




Epigenomics




CyberShake

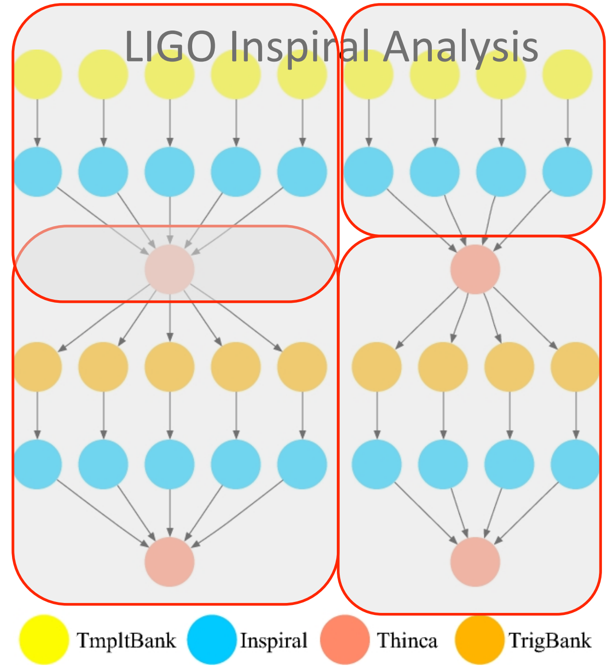


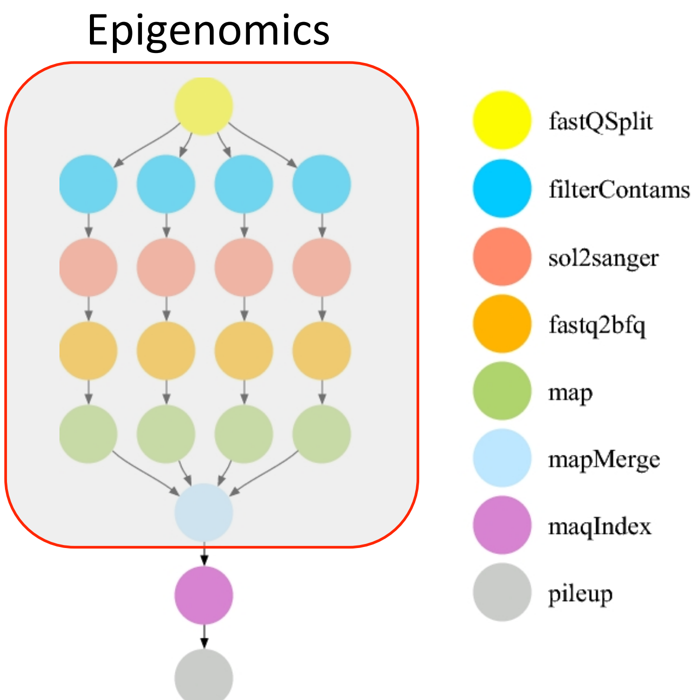

https://confluence.pegasus.isi.edu/display/pegasus/WorkflowGenerator

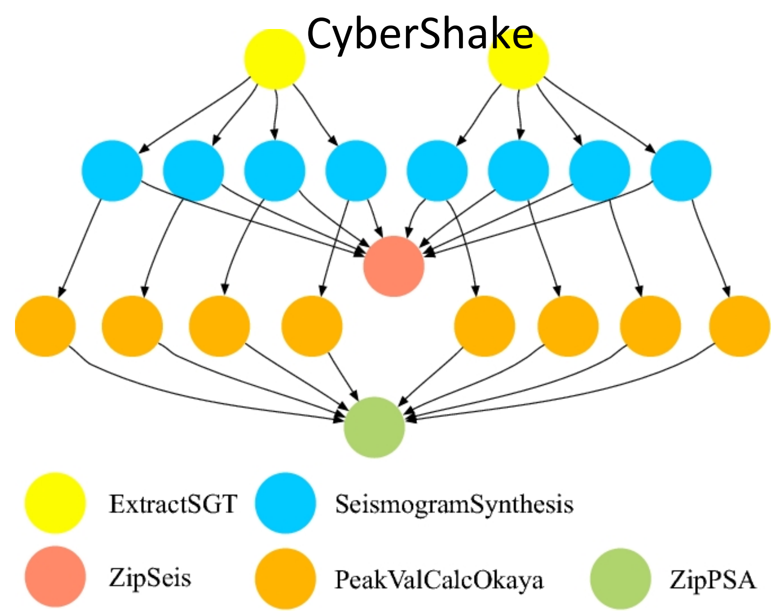
# Real World Workflows

(Structured) fork-join pattern

Structured: All branches of one fork are merged at one join




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?

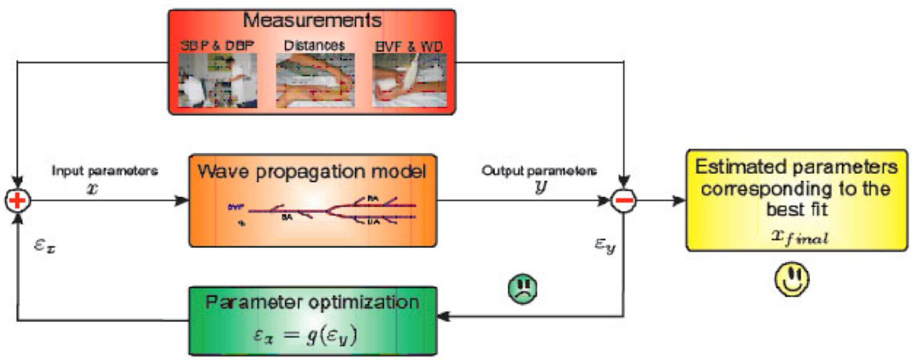
Provenance/ reproducibility

- **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"

Source: Workflows and e-Science: An overview of workflow system features and capabilities Ewa Deelmana, Dennis Gannonb, Matthew Shields c, Ian Taylor, Future Generation Computer Systems 25 (2009)

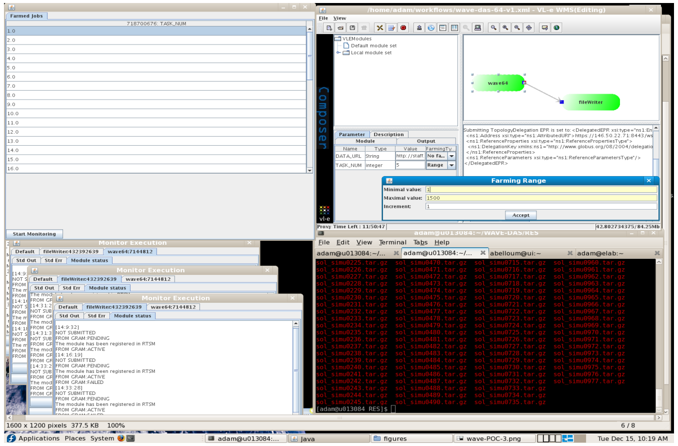
# wave propagation model applications



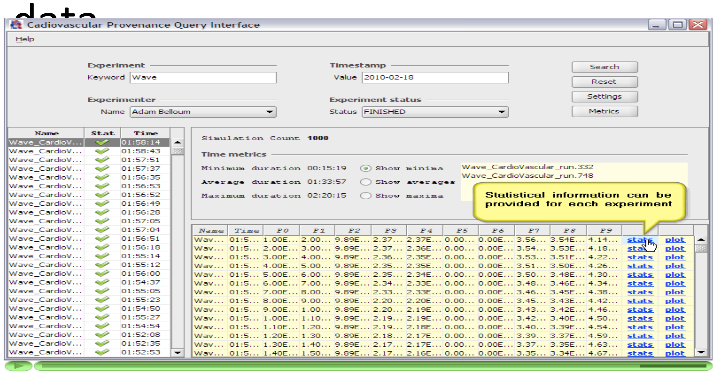

***[Biomedical engineering Cardiovascular biomechanics group TUE])***

*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



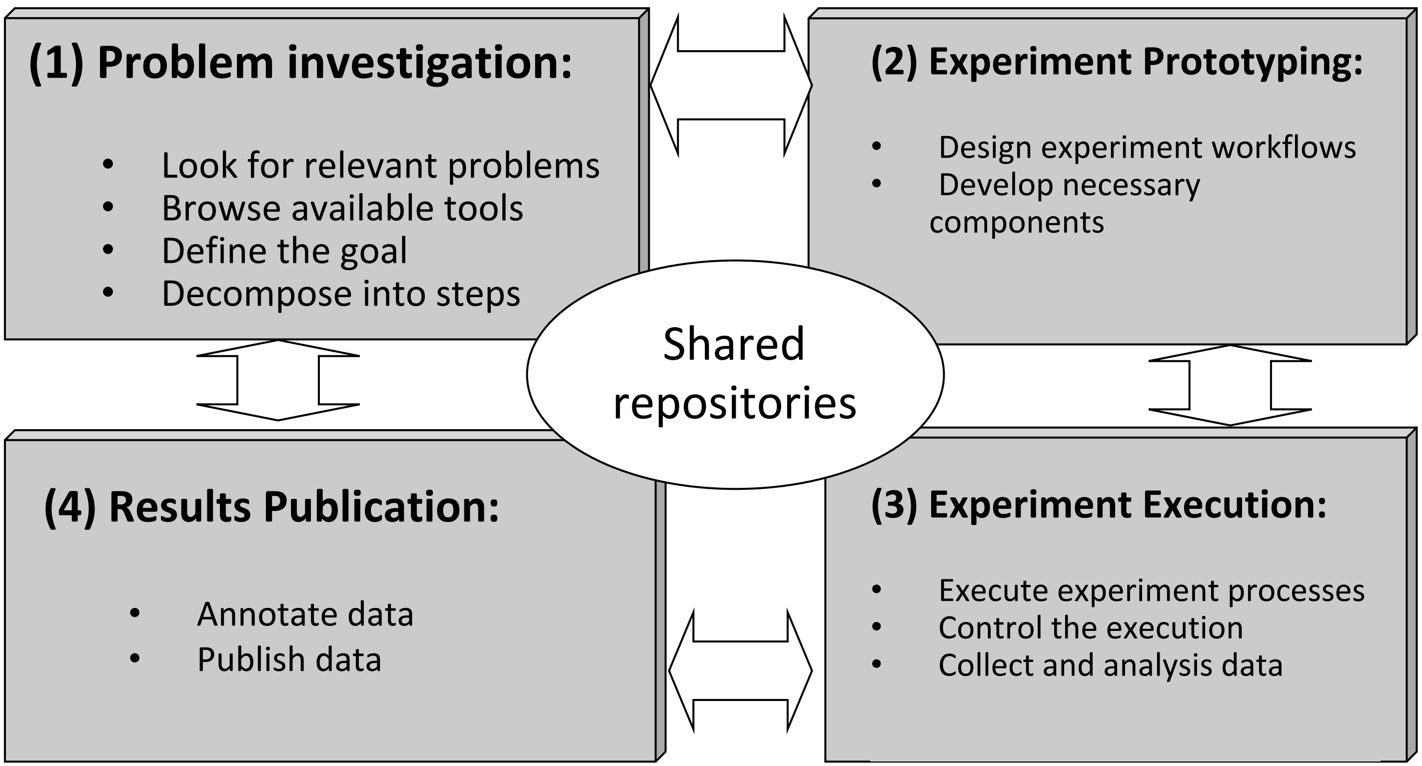
User Interface to compose workflow (top right), monitor the execution of the farmed workflows (top left), and monitor each run separately (bottom left) data




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?

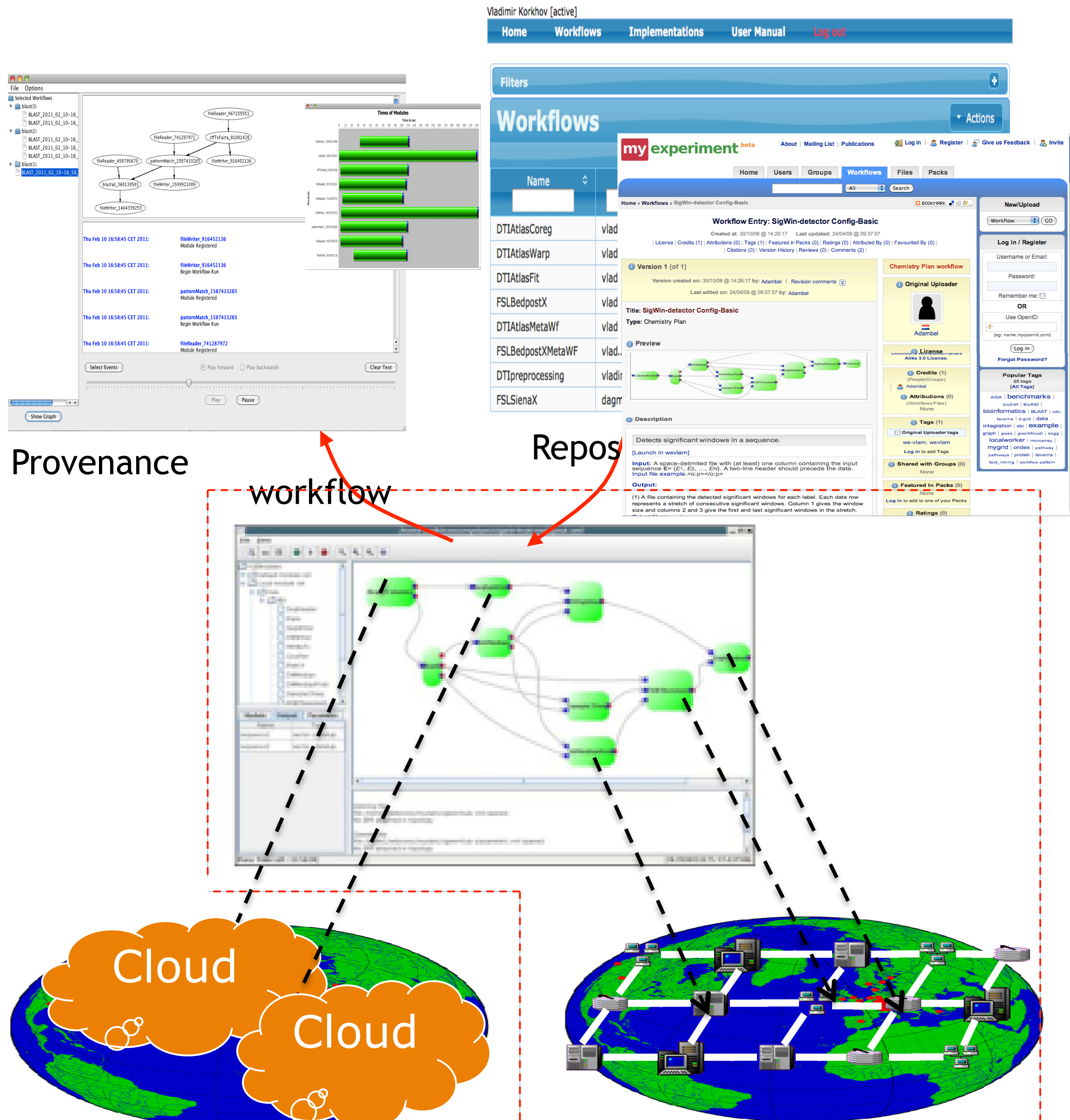
Help in most of the Scientific experiments lifecycle phases




*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

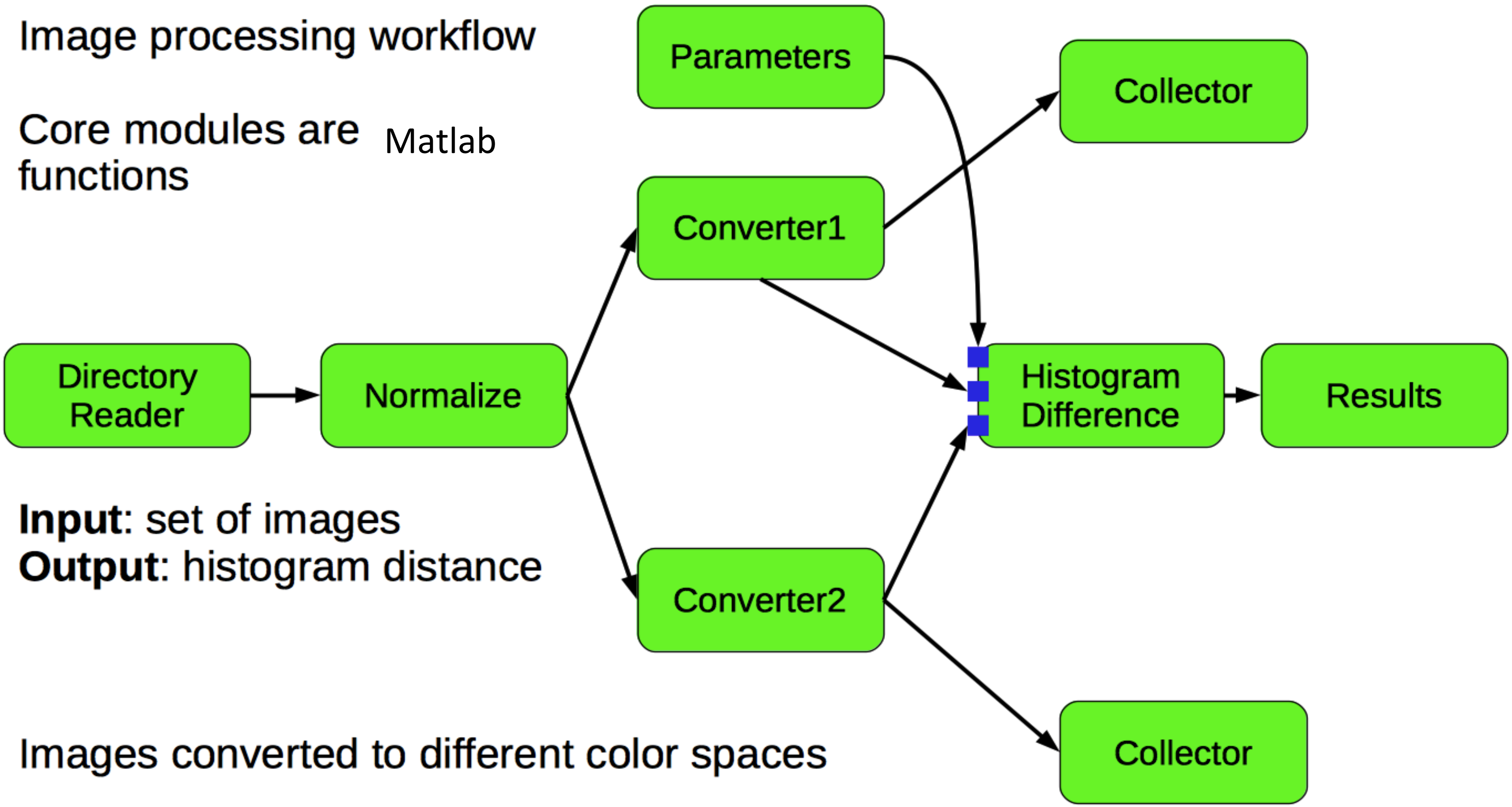



**source:** [https://ivi.fnwi.uva.nl/sne/wsvlam2/](https://ivi.fnwi.uva.nl/sne/wsvlam2/)

# Example of Scientific workflow

Image processing workflow

Core modules are Matlab functions



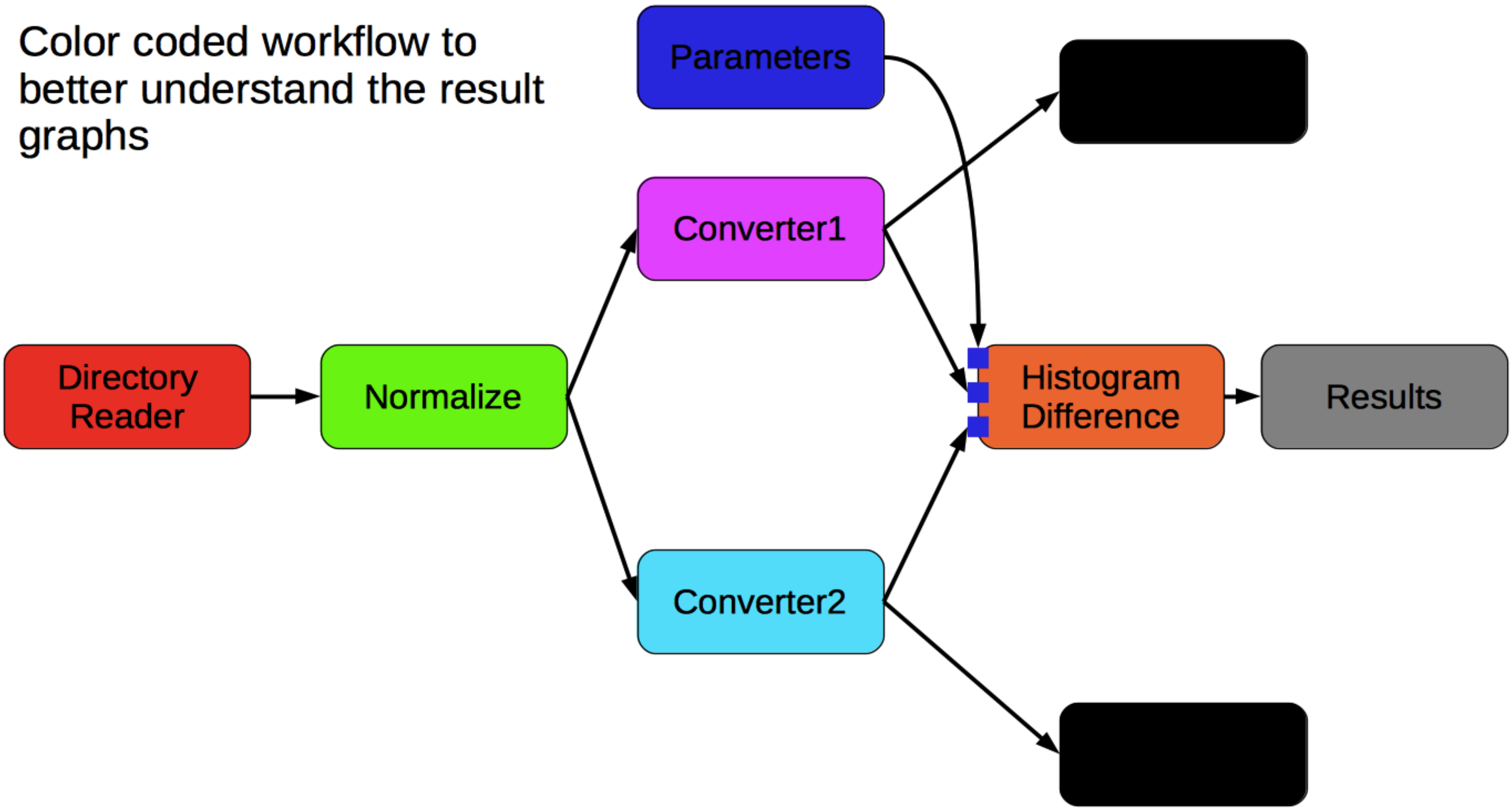

**Input**: set of images
**Output**: histogram distance

Images converted to different color spaces

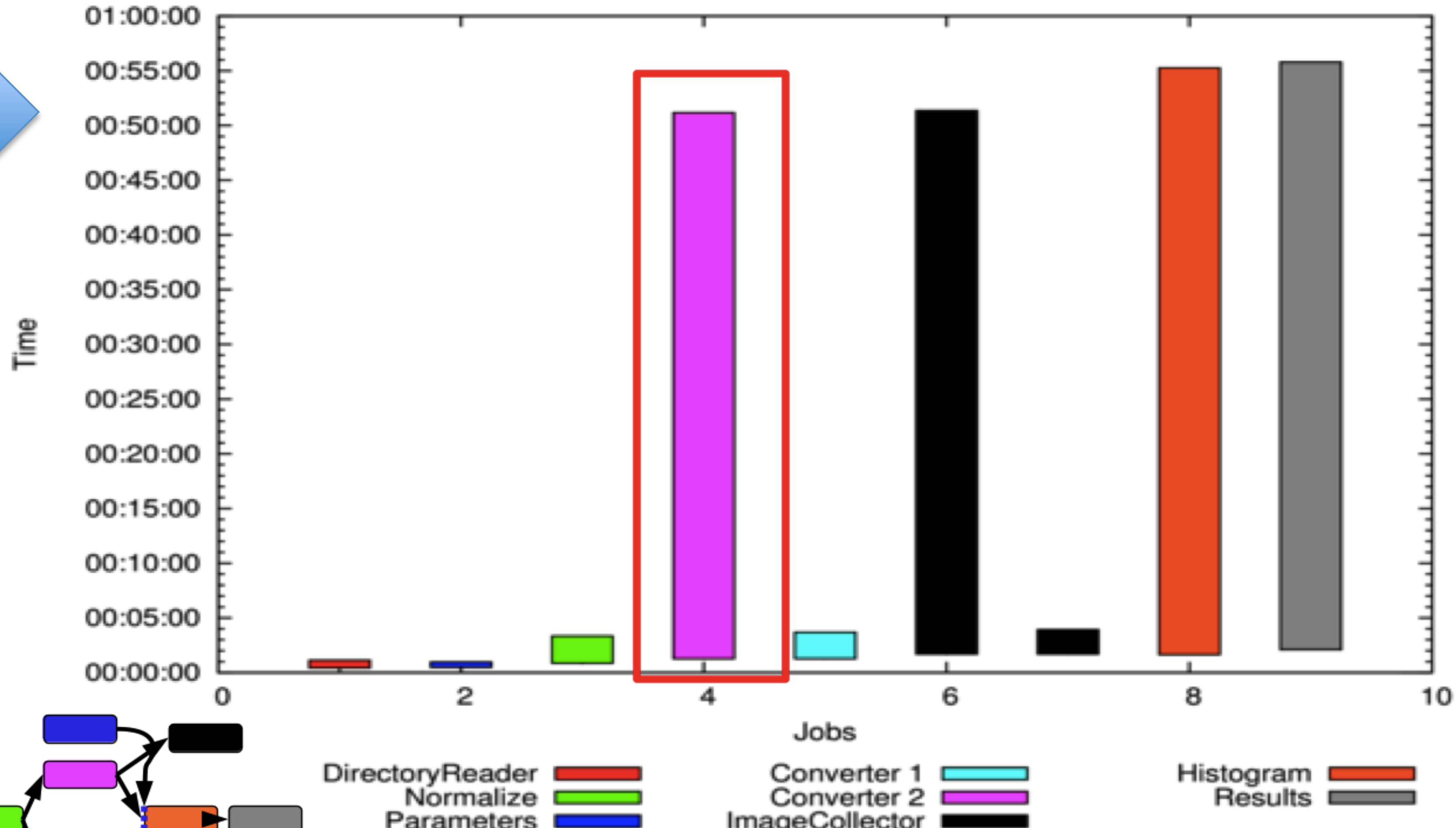
Histogram difference is calculated between color spaces

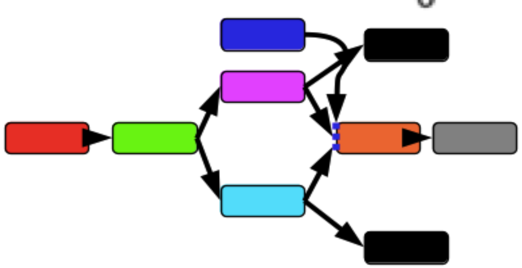
# Example of Scientific workflow

Color coded workflow to better understand the result graphs



# Workflow Without Scaling

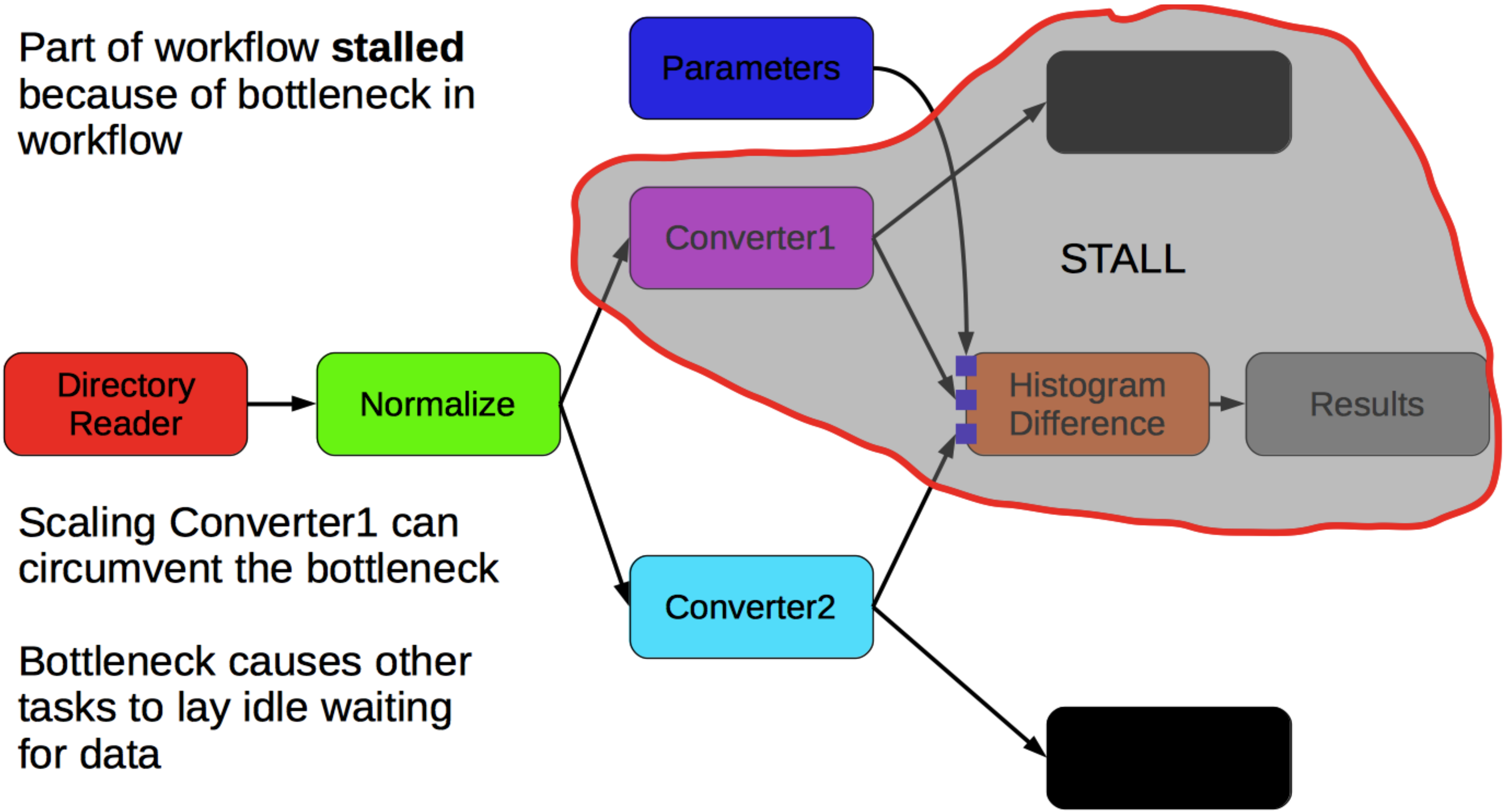


Slow task causing a **bottleneck** in the workflow

# Bottleneck task in Scientific workflow

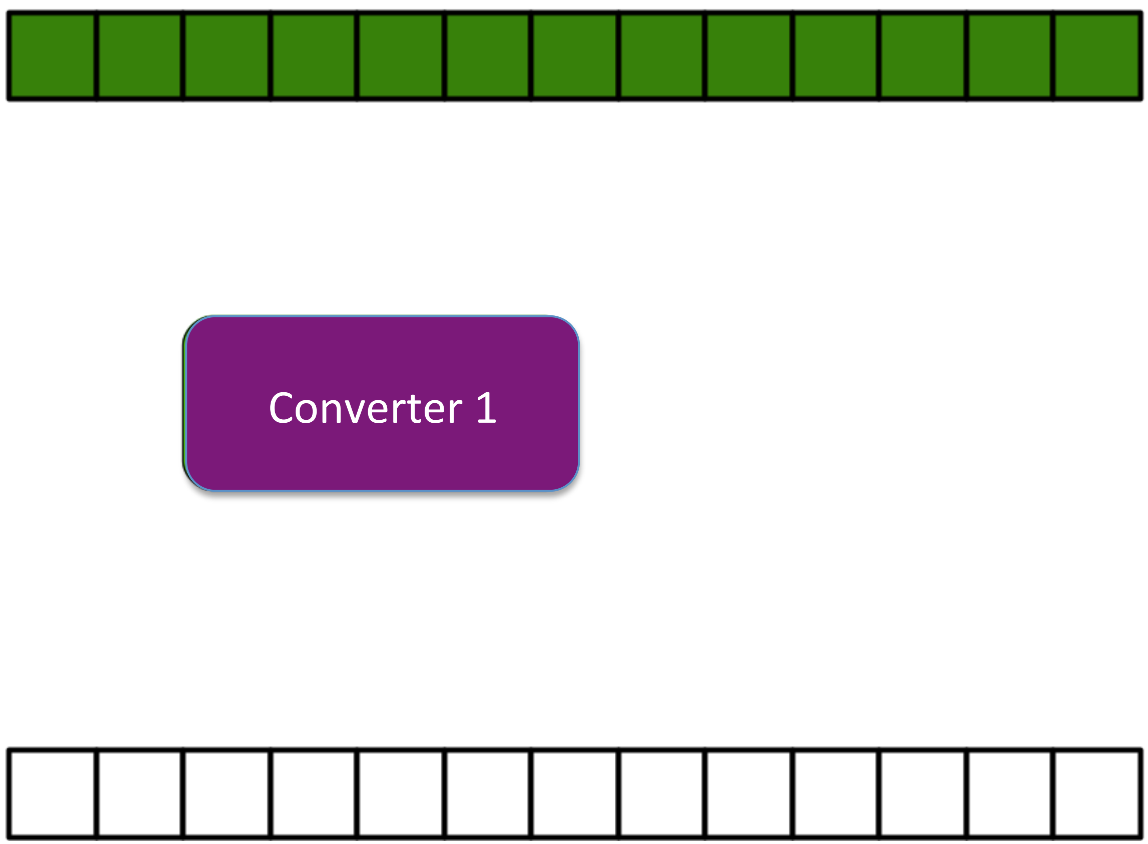
Part of workflow **stalled** because of bottleneck in workflow



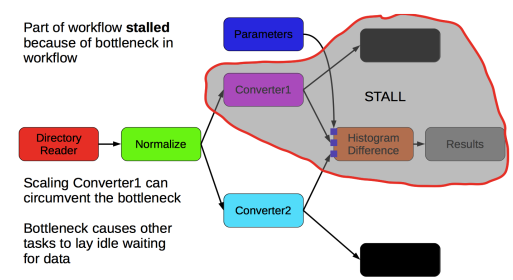

Scaling Converter1 can circumvent the bottleneck

Bottleneck causes other tasks to lay idle waiting for data

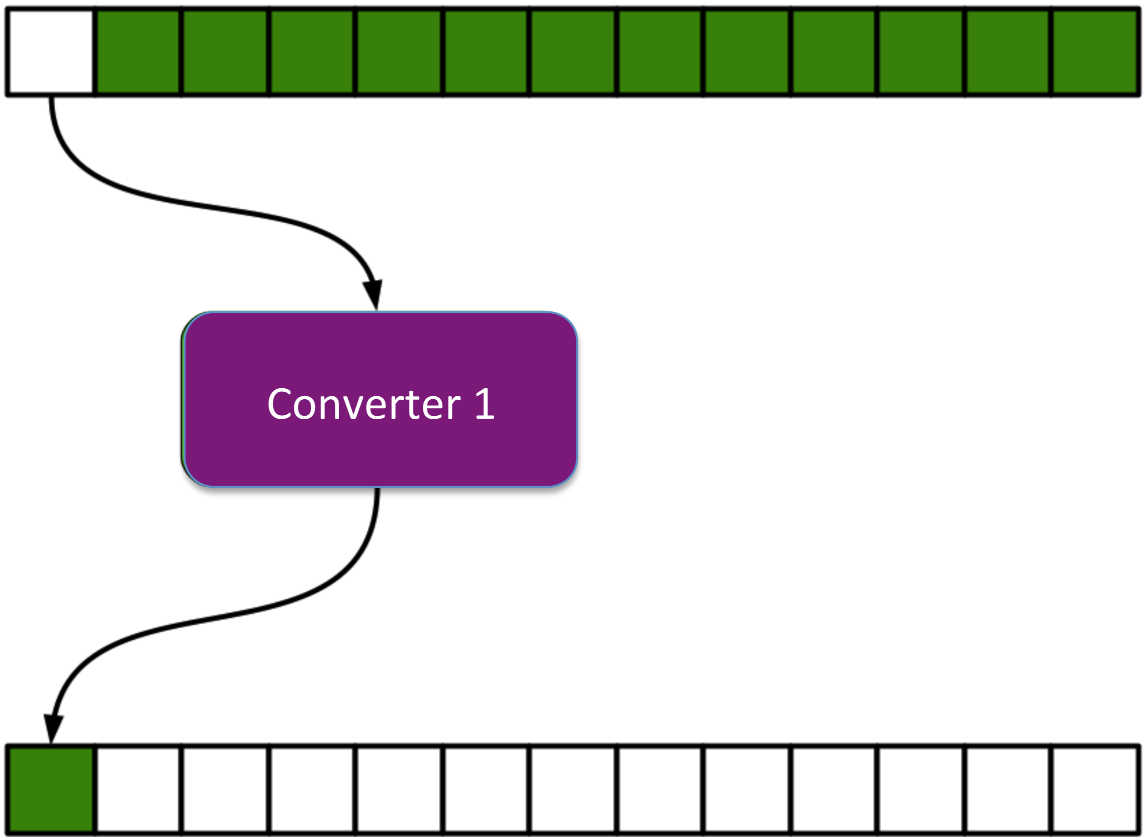
# Zooming into the Task Converter 1




Data organized in atomic parcels(messages)

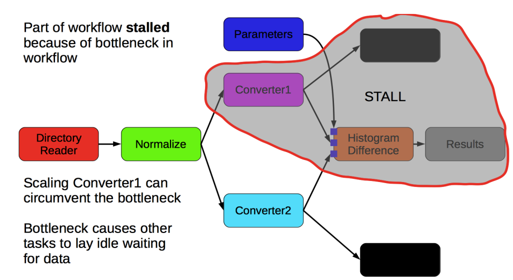


# Zooming into the Task Converter 1

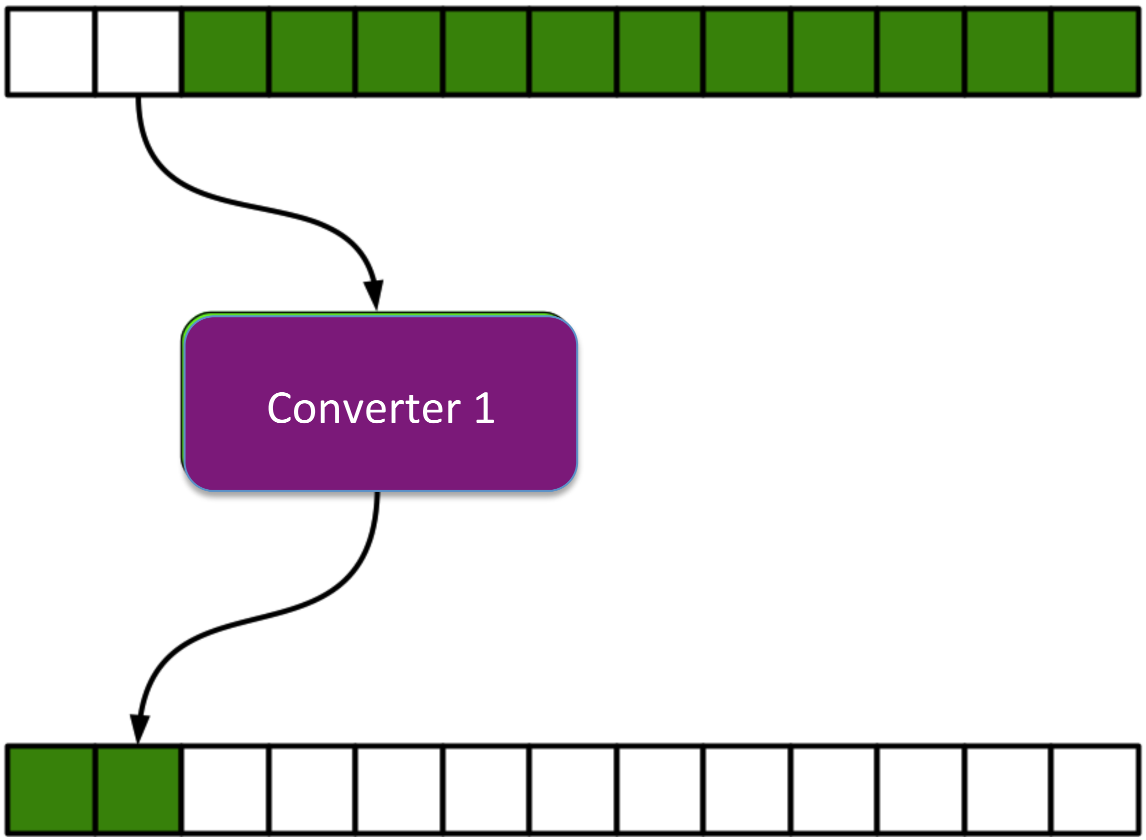



Data organized in atomic parcels(messages)

Task processes data sequentially

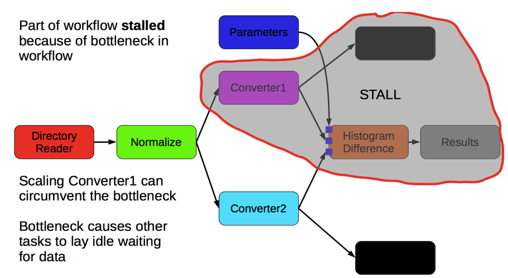


# Scaling Concepts

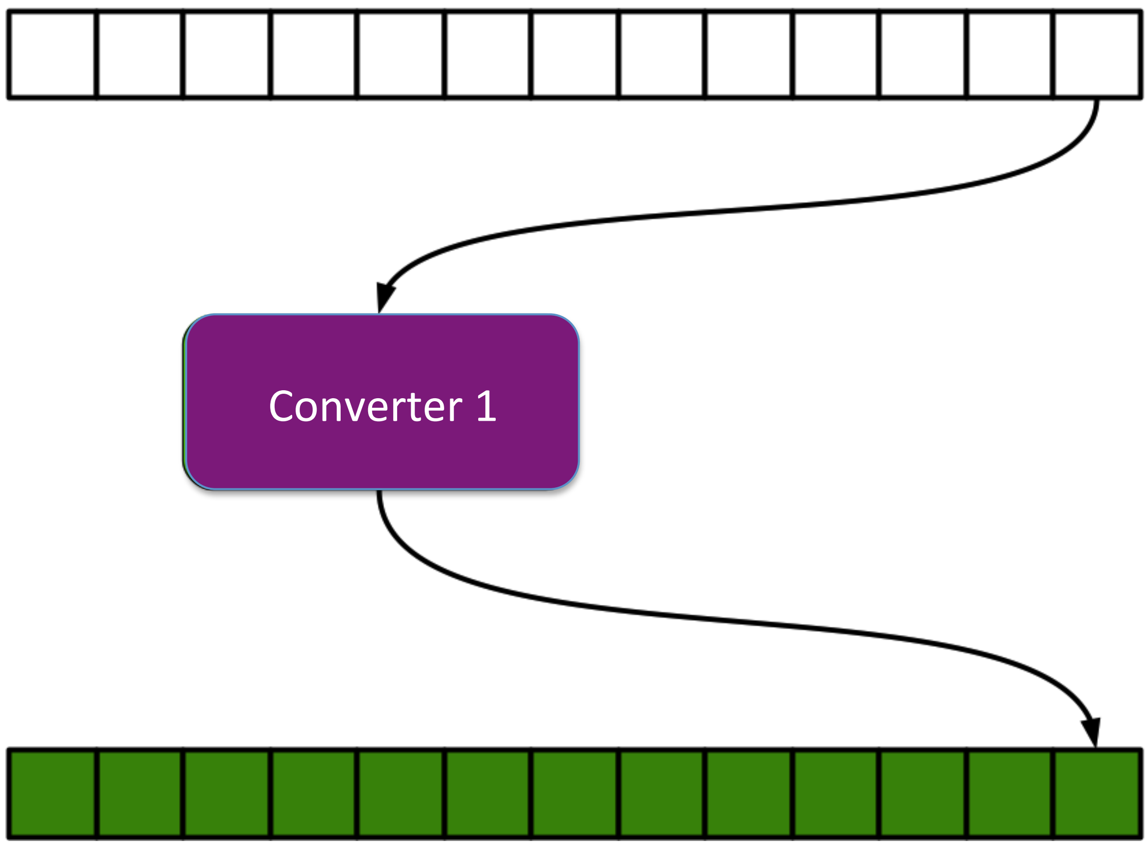



Data organized in atomic parcels(messages)

Task processes data sequentially

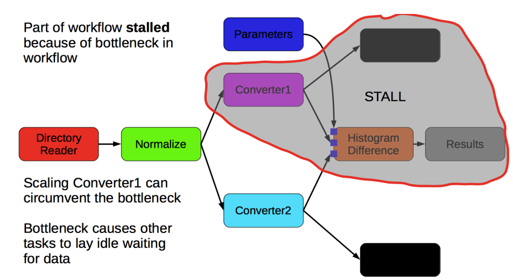


# Zooming into the Task Converter 1

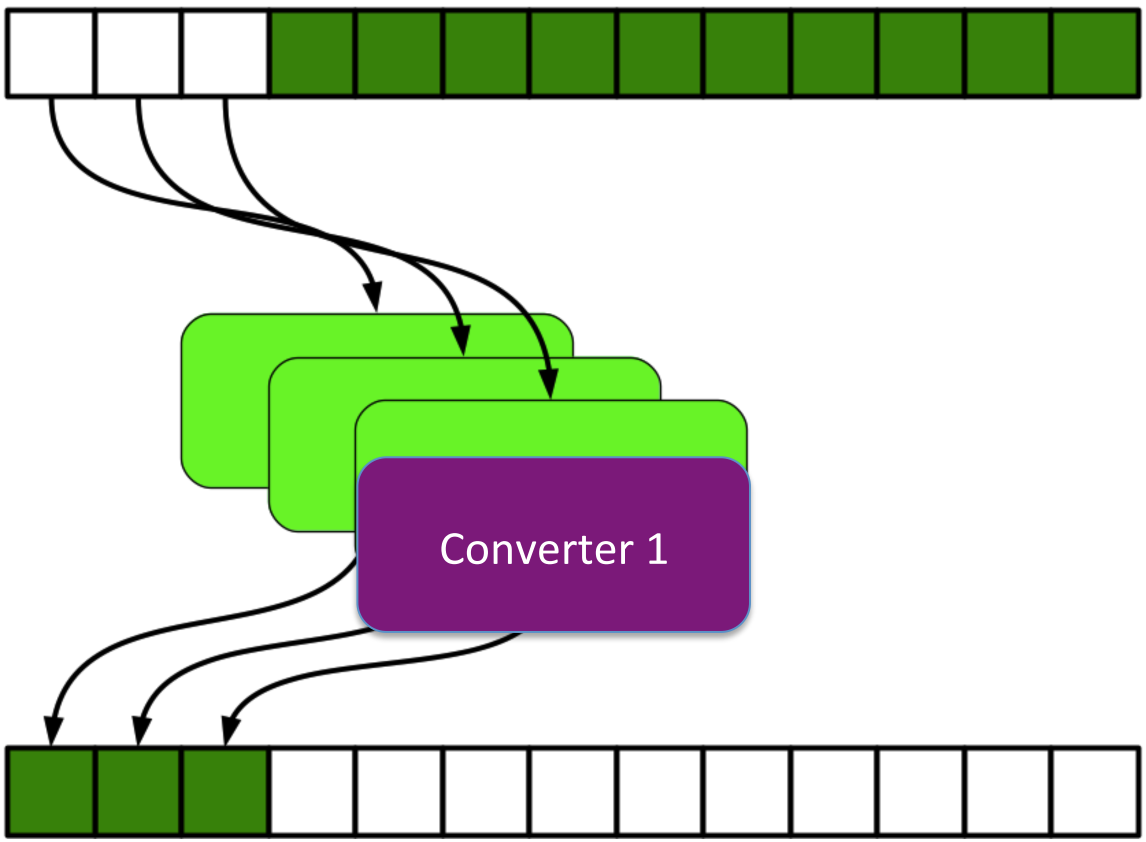



Data organized in atomic parcels(messages)

Task processes data sequentially



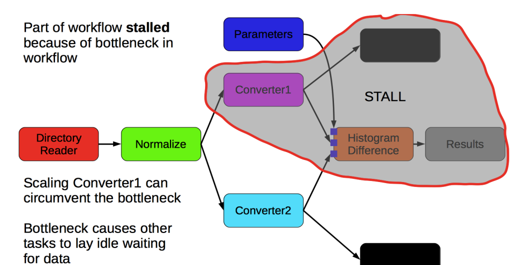
# Zooming into the Task Converter 1



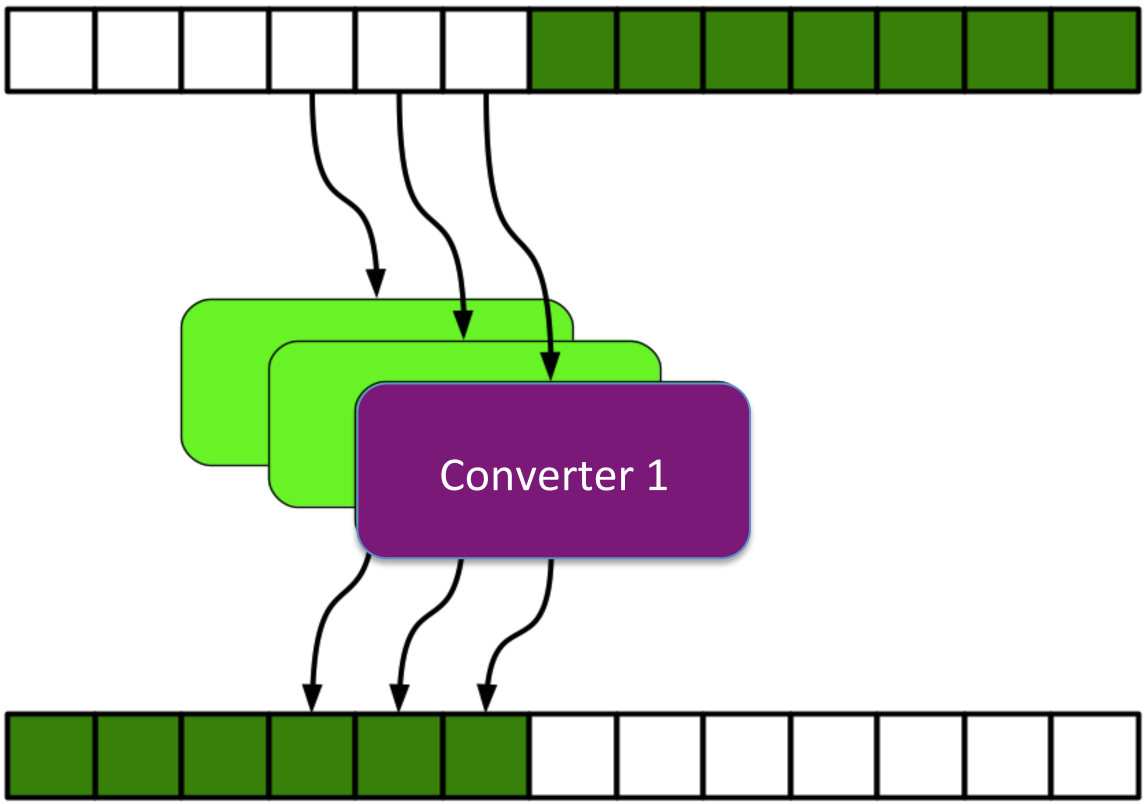

Data organized in atomic parcels(messages)

Tasks processes data **concurrently**

Adding more tasks increases **message consumption** rate



# Zooming into the Task Converter 1




Data organized in atomic parcels(messages)

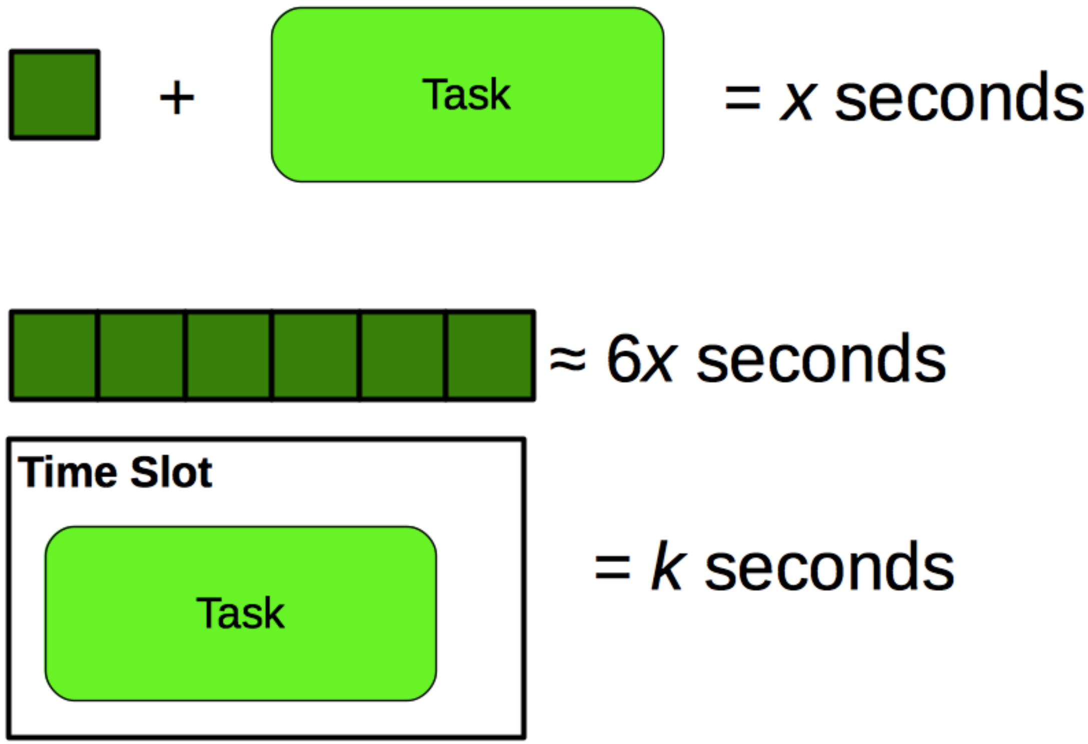
Task processes data sequentially

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

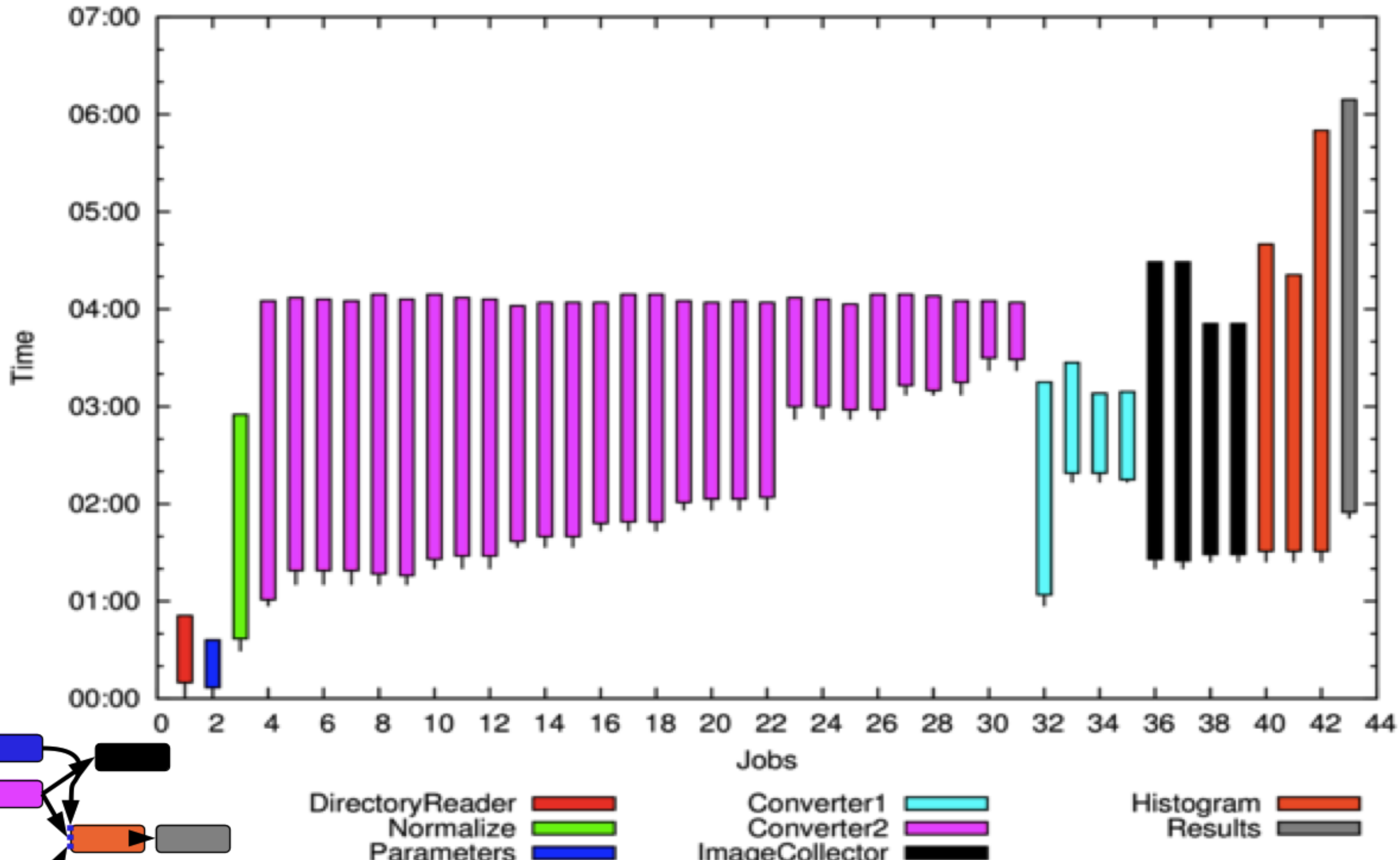
# Load Prediction

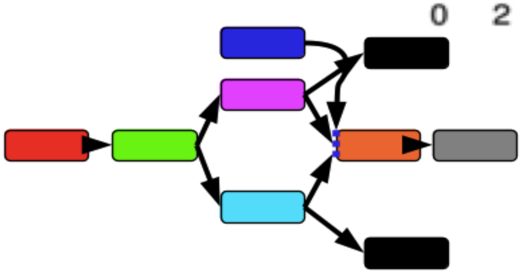



**Simplified Load** = $6x/k$ time slots

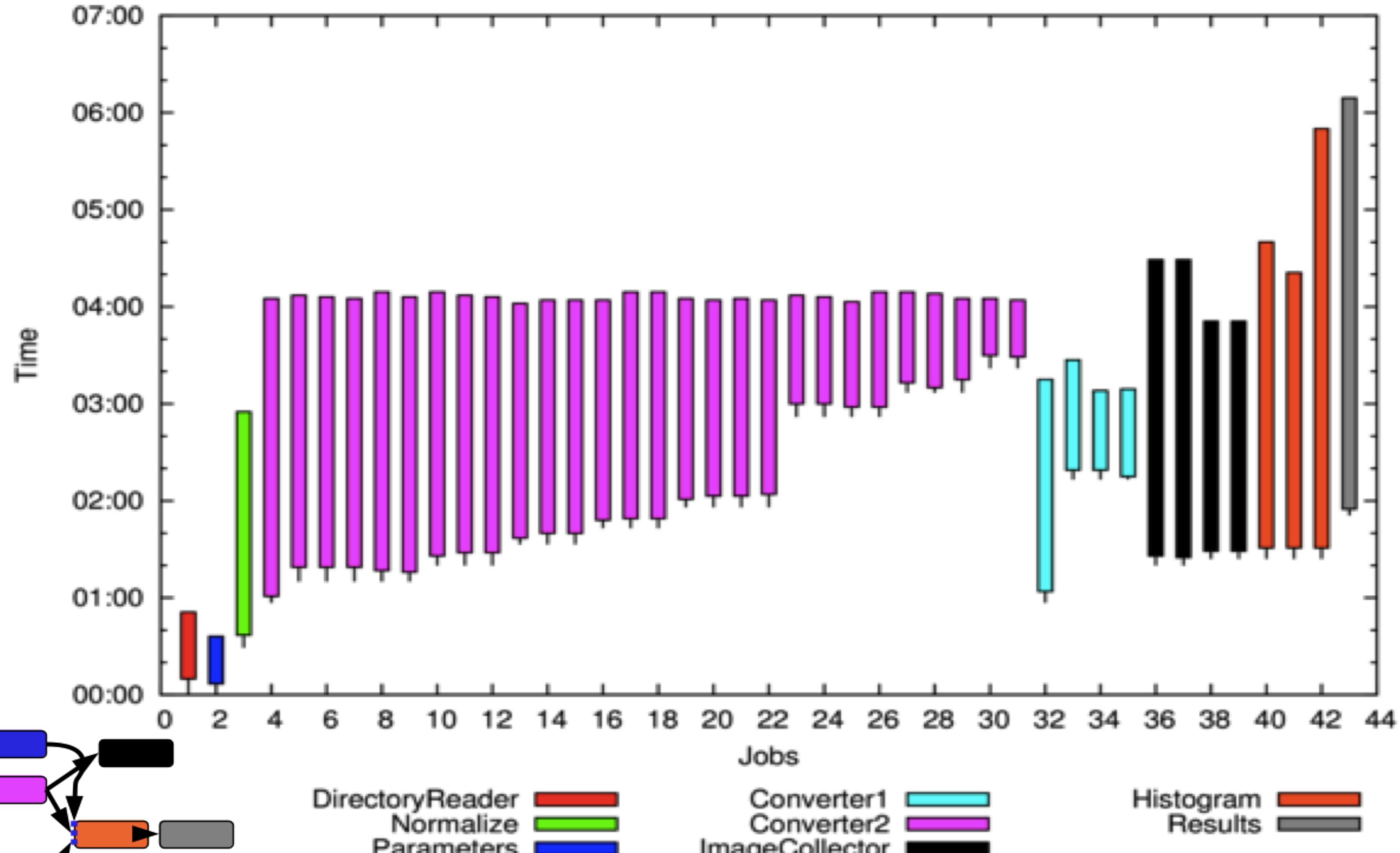
**Assumption**: Size of data directly proportional to computation time. May not always be the case

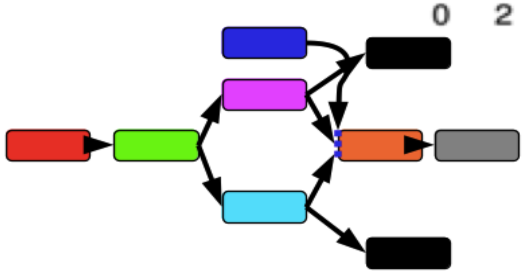
# Workflow execution with Scaling



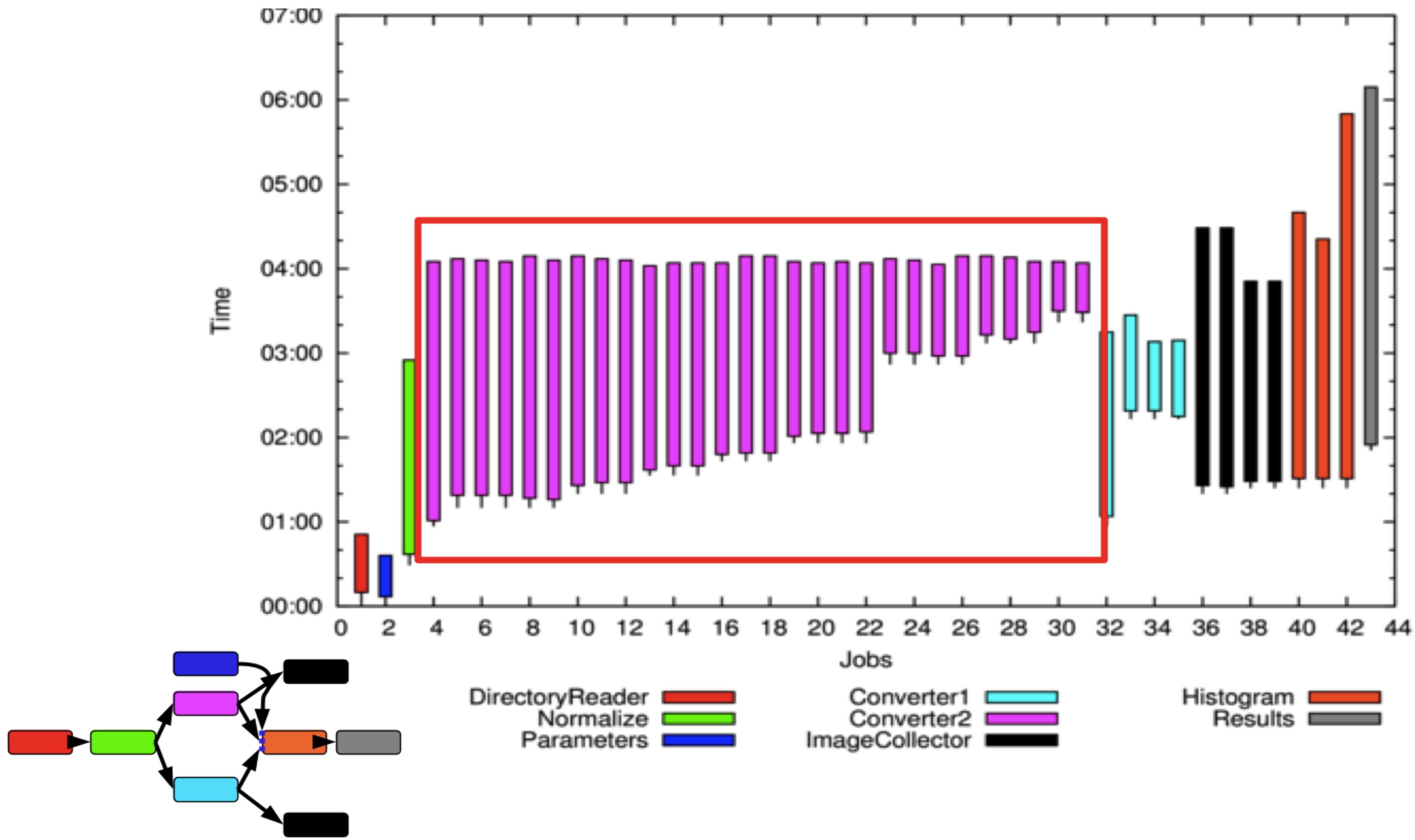



# Workflow execution with Scaling

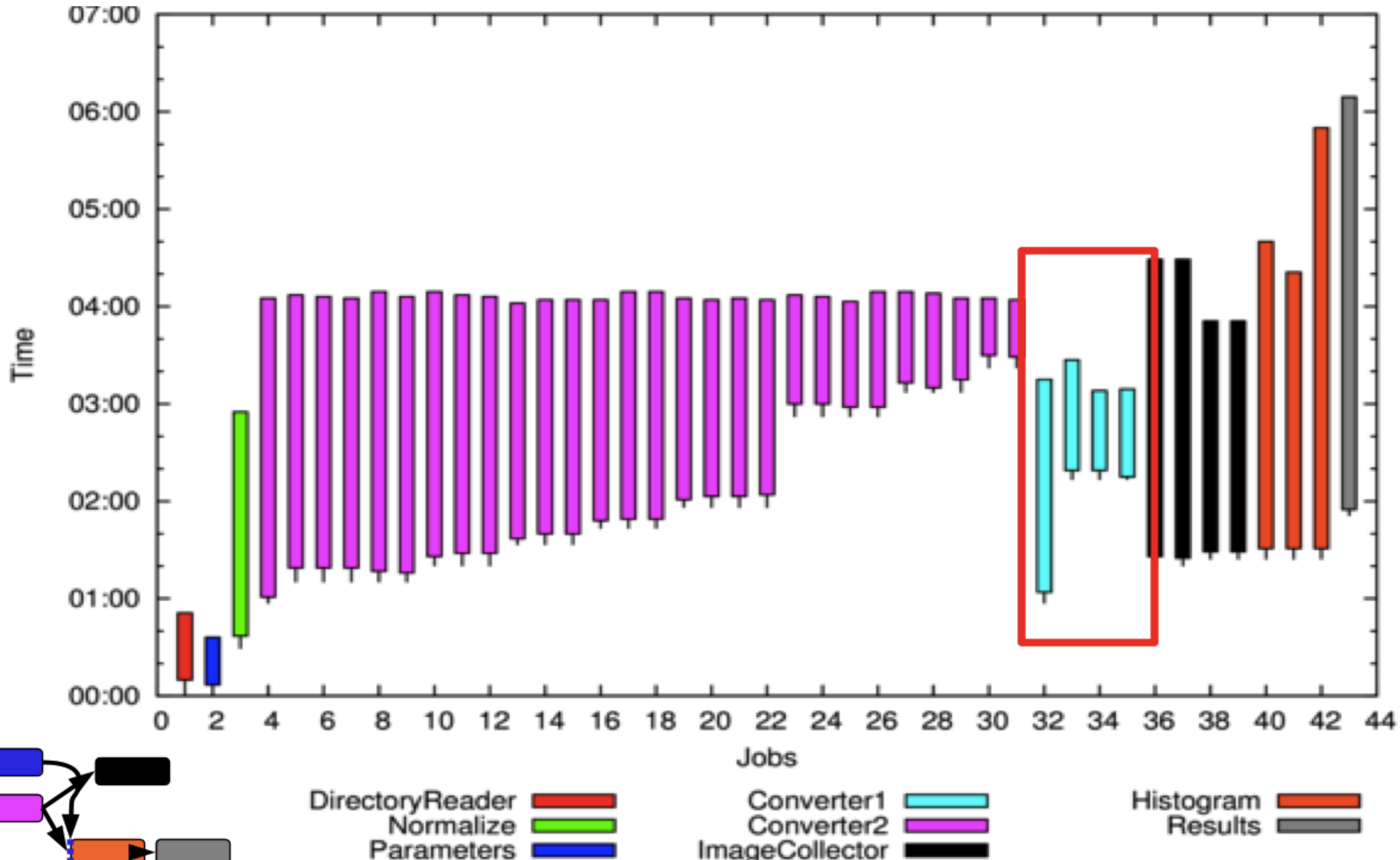


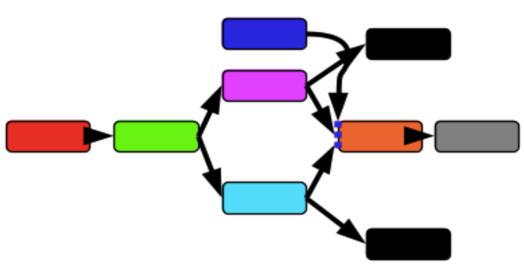



# Workflow execution with Scaling Task - I

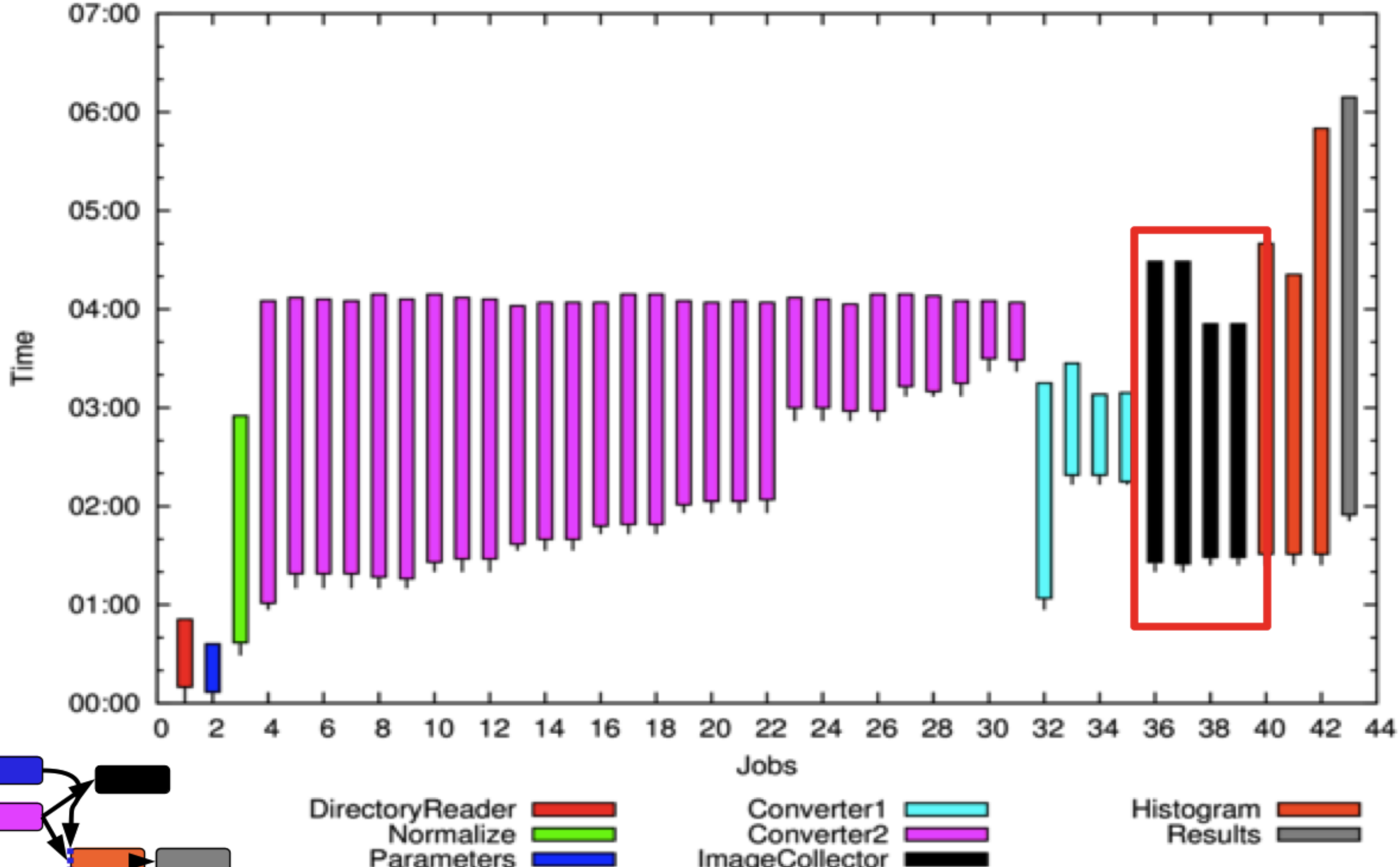


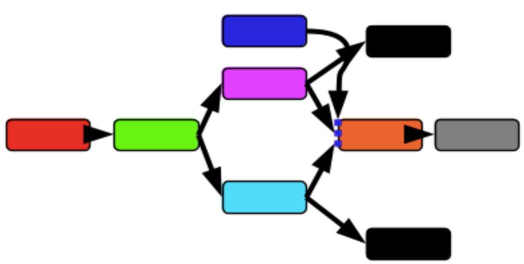
# Workflow execution with Scaling Task -2



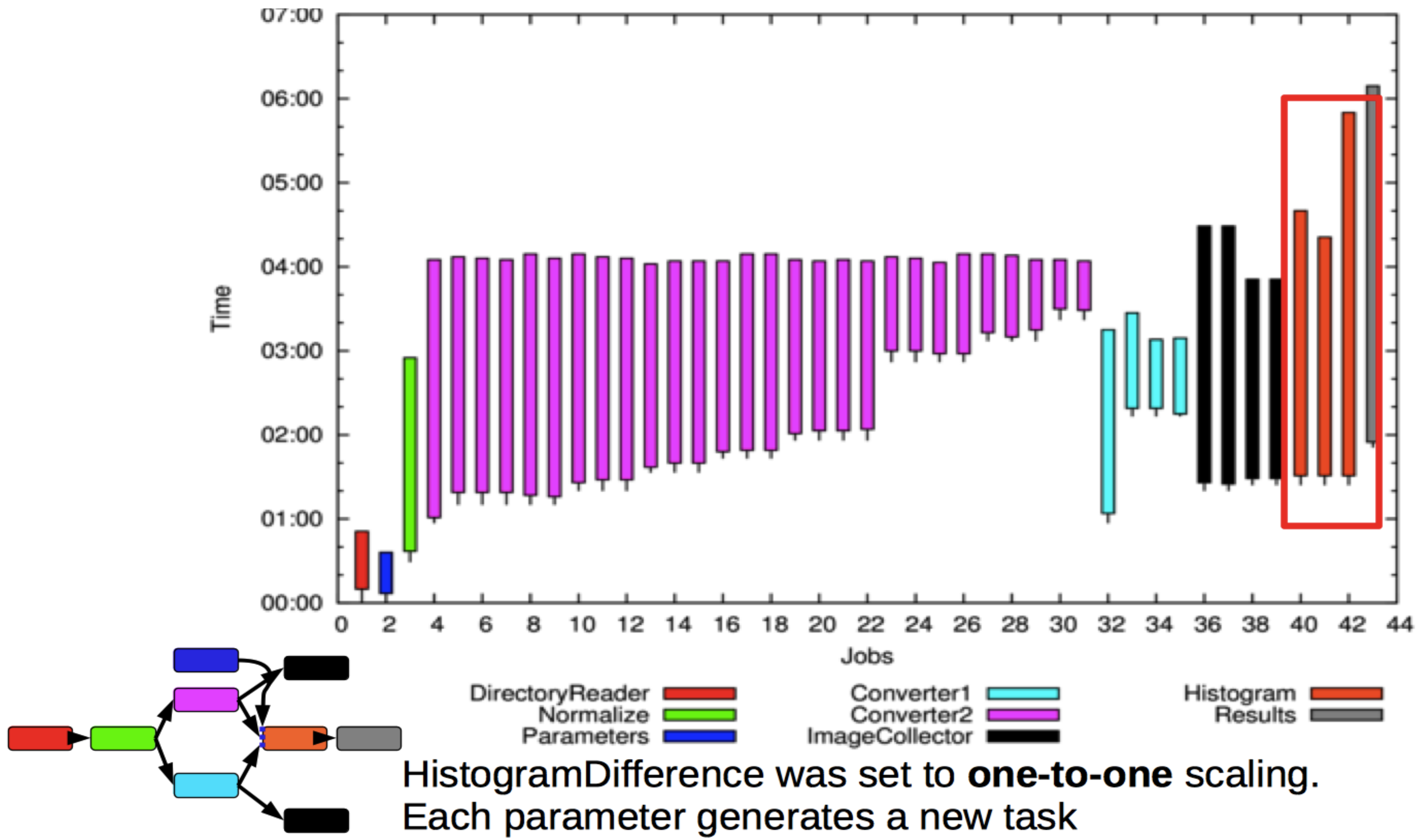
# Other Scaled Task - I



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

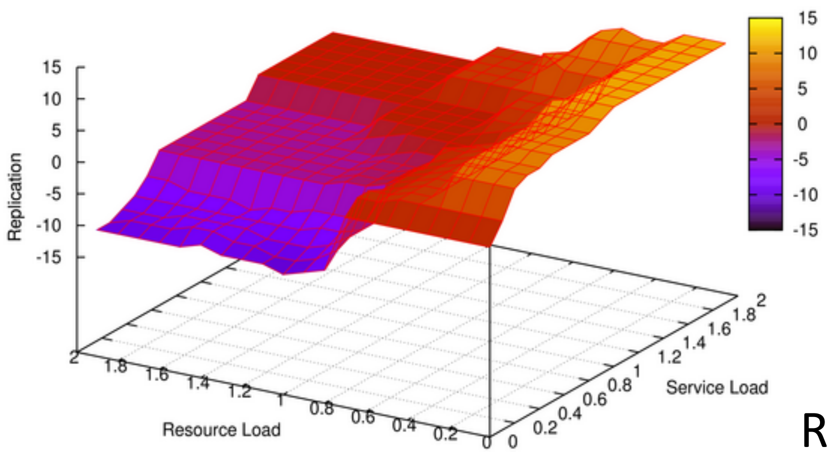
# Resource management

- 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**

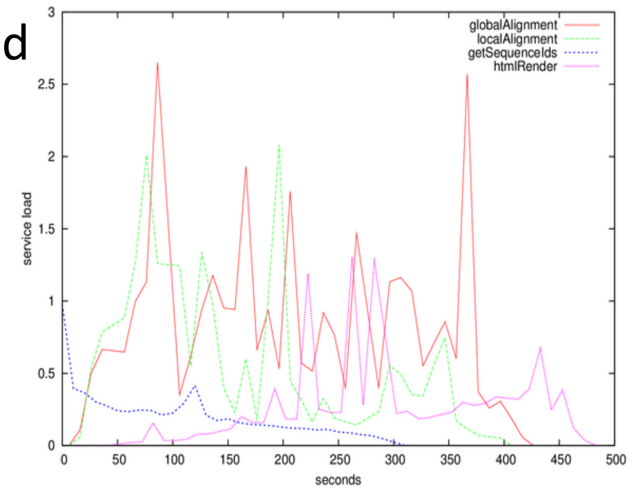
- It would be ideal to have a mechanism to **greedily** consume resources if **no one** is using them but **donate back** resources once they are requested.

- Some workflow management systems might also help

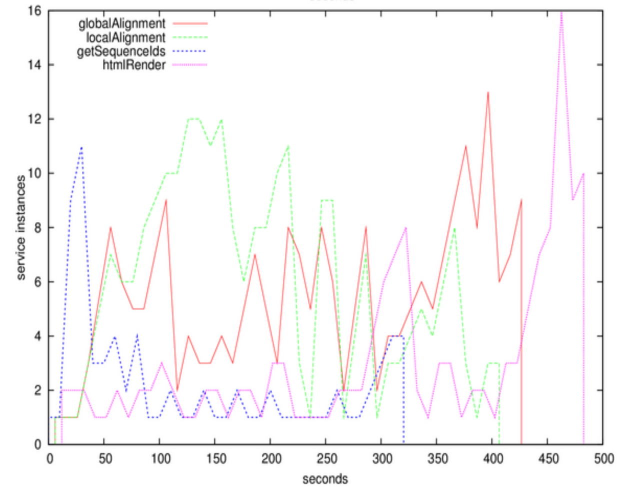
# Scale with the increase of input load

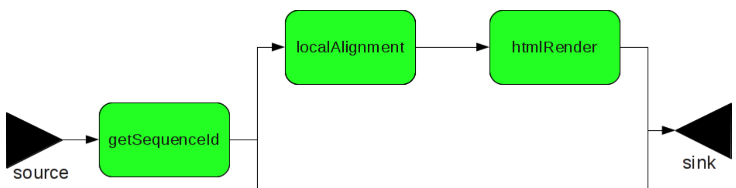



Service Load




Running Service instances

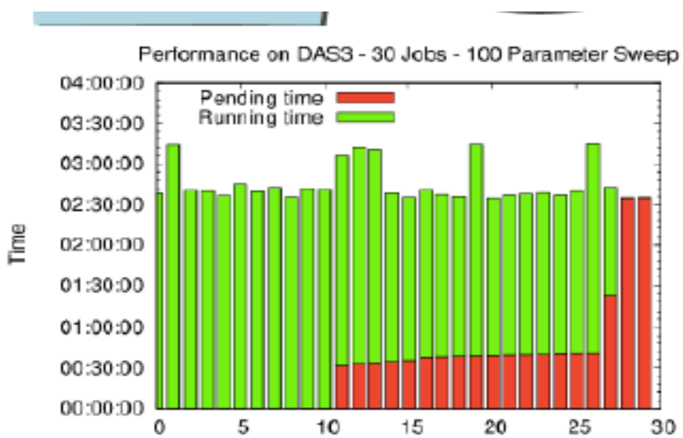


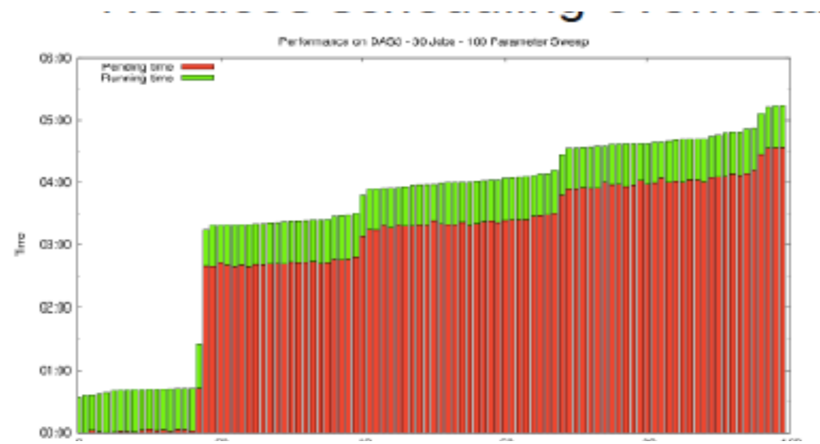




Source: Reginald Cushing, Spiros Koulouzis, Adam S. Z. Belloum, Marian Bubak, **Dynamic Handling for Cooperating Scientific Web Services**, 7th IEEE International Conference on e-Science, December 2011, Stockholm, Sweden

# 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



Source: Reginald Cushing, Adam S. Z.  Belloum, V. Korkhov, D. Vasyunin, M.T. Bubak, C. Leguy ECMLS’12, June 18, 2012, Delft, *Workflow as a Service: An Approach to Workflow Farming*, The Netherlands

# 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)

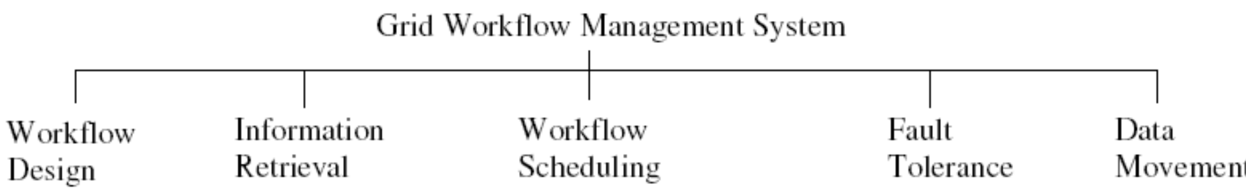
- Model of computation: stream-based process network.
  - Engine **co-allocates** all workflows.
  - Components waste time idling.
  - Co-allocation difficult.
- Communication: time **coupled**
  - Assumes components are running
  - Simultaneously
  - Synchronized p2p
  - Fixed TCP/IP

V. Korkhov et al. VLAM-G: Interactive data driven workflow engine for Grid-enabled resources, Scientific Programming 15 (2007) 173–188 173 IOS Press

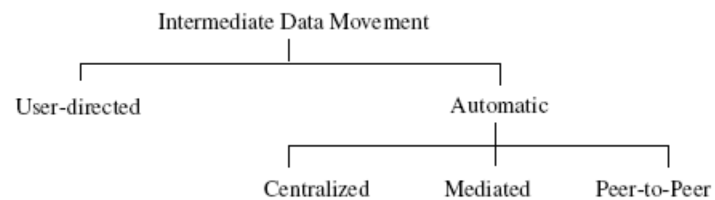
# Model of computation

- 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.



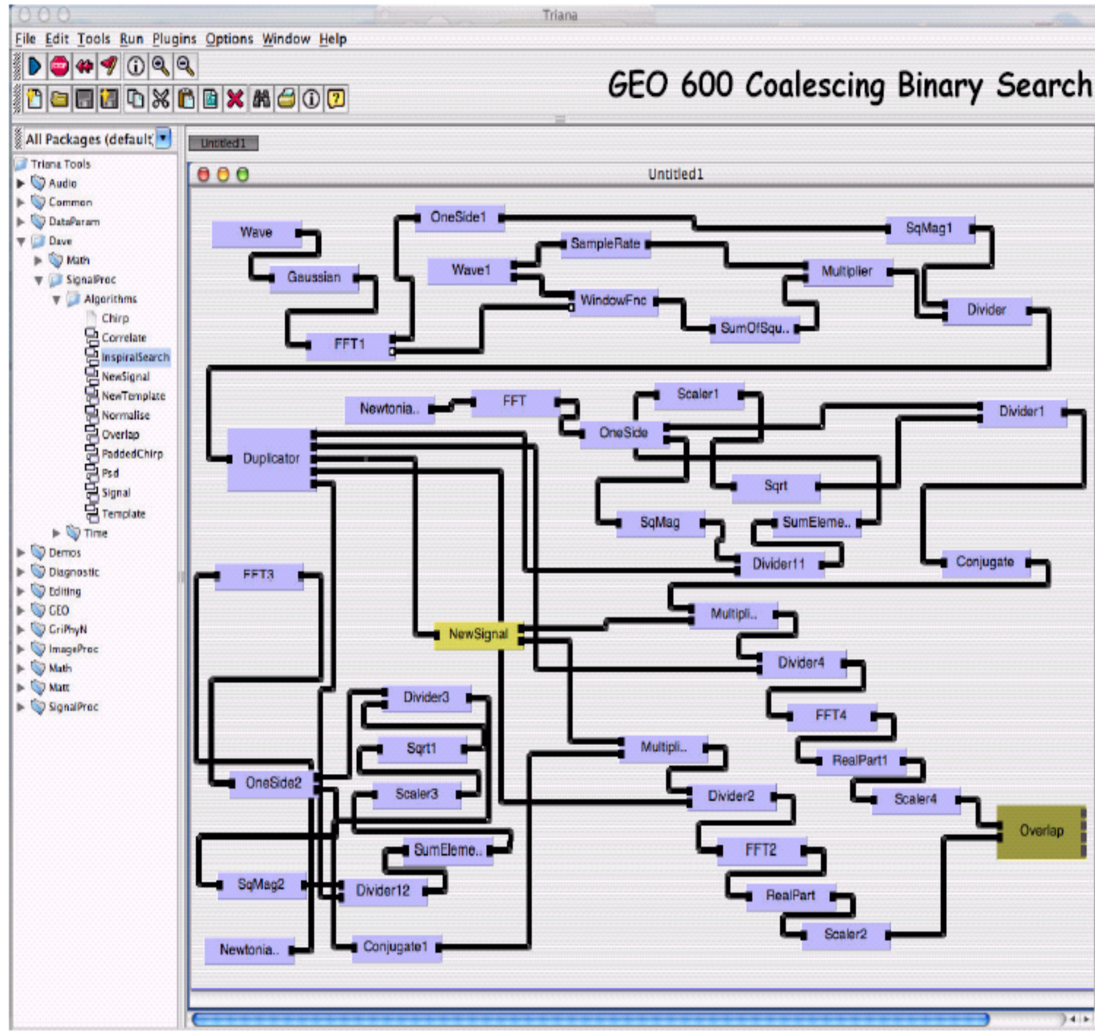

- The intermediate data has to be staged out to the corresponding Grid/Cloud sites.

**Source**: A Taxonomy of Workflow Management Systems for Grid Computing
Jia Yu and Rajkumar Buyya, http://www.cloudbus.org/reports/GridWorkflowTaxonomy.pdf

# Component Based Workflow Description: Triana

Feature Highlights

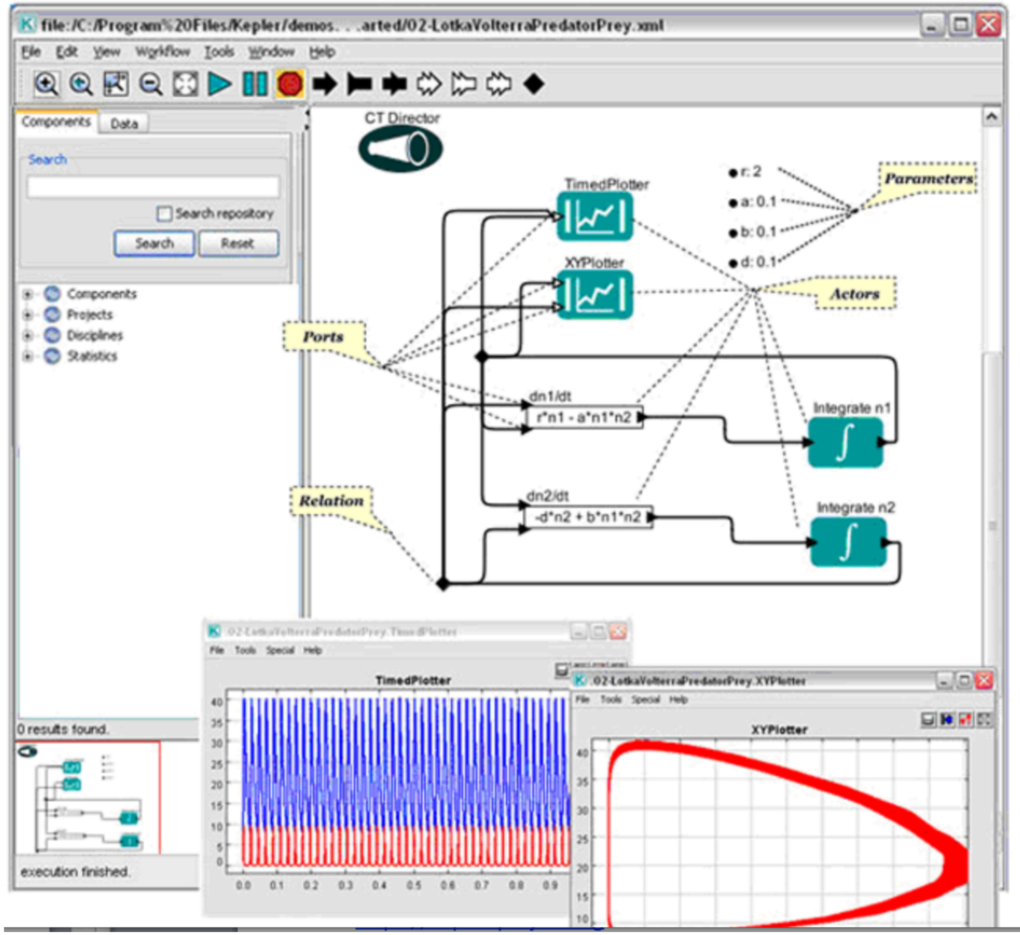
- Modular Java Workflow Environment
- Triana comes with a wide variety of built-in tools.
  - There is an extensive signal-analysis toolkit, an image-manipulation toolkit, a desk-top publishing toolkit, and many more
- Triana Cloud Job Queuing
- Sophisticated Drag & Drop Composition
- Web Services
-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




Source http://www.trianacode.org

# Component Based Workflow Description: Kepler

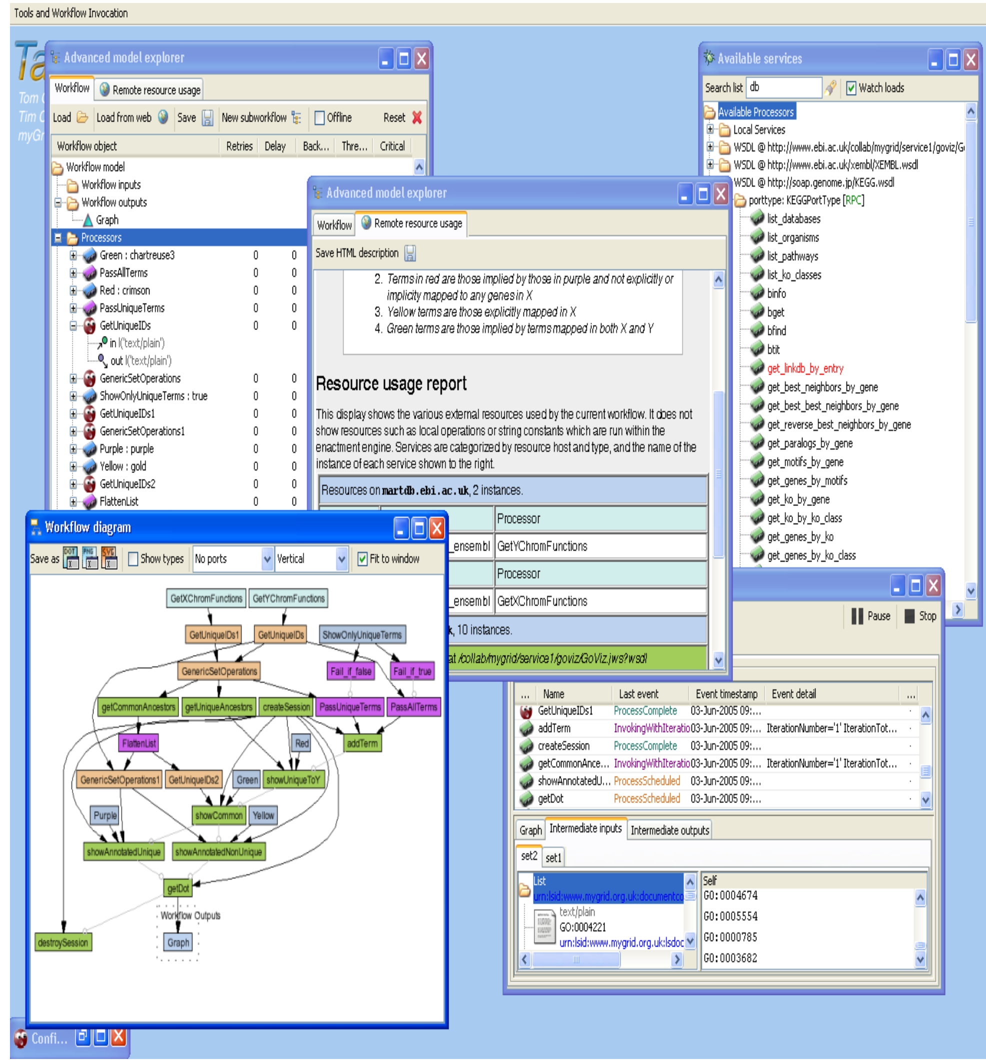
- Kepler provides a graphical user interface and a run-time engine that can execute workflows either from within the graphical interface or from a command line.
- Kepler workflows can be nested, allowing com-plex tasks to be composed from simpler components.
- Kepler workflows can leverage the compu-tational 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

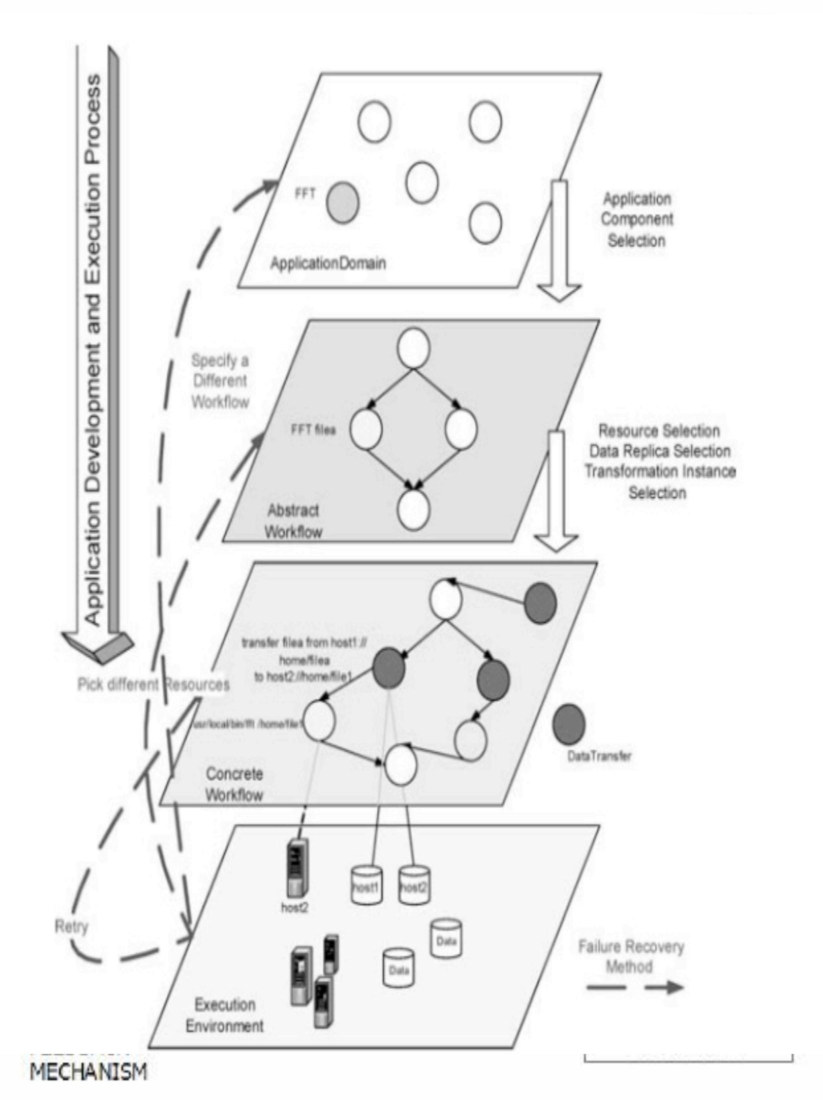
- Access to local tools/scripts and remote resources and analysis tools, Web
- Not restricted to predetermined services – rapid incorporation of new services without coding
- Up-to-date R support (version 3.1.0)
- Excel and csv spreadsheet support Interaction with a running workflow from your Web browser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




Source [http://www.taverna.org.uk](http://www.taverna.org.uk)

# Pegasus

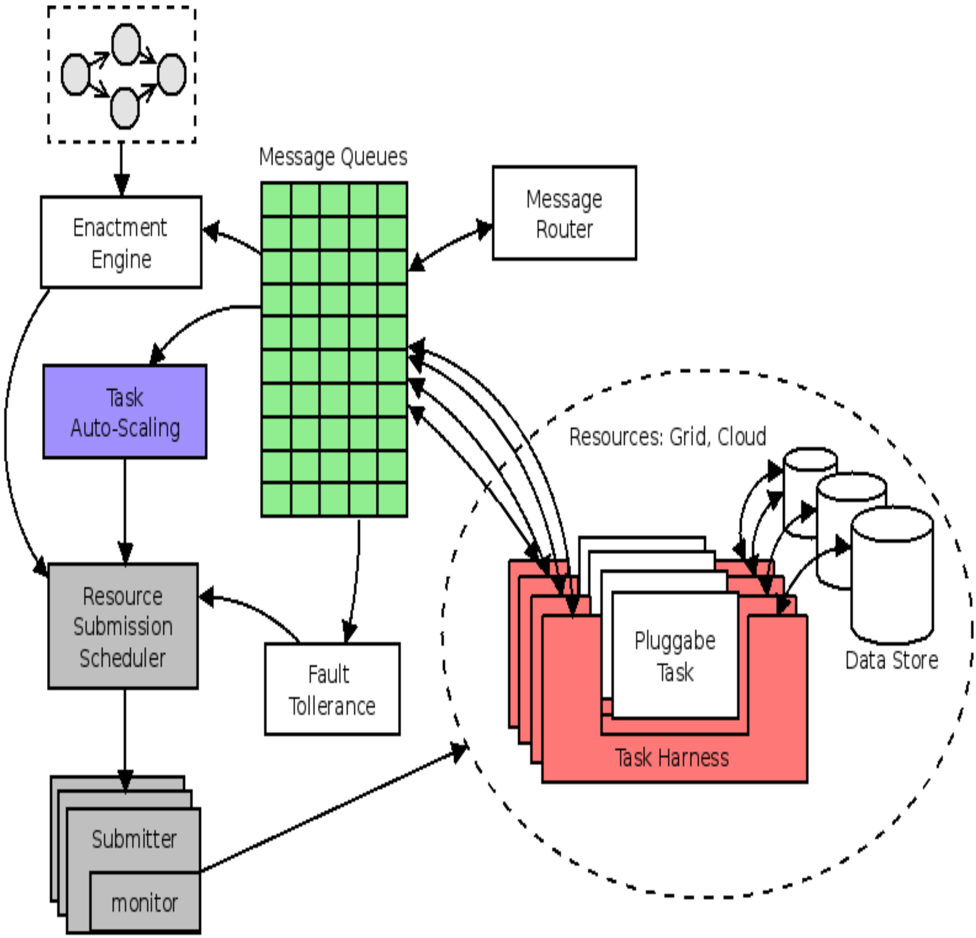
- Portability / Reuse
- Performance
  - Pegasus mapper can reorder, group, and prioritize tasks in order to increase overall workflow performance
- Scalability
  - from just a few computational tasks up to 1 million
- Provenance
  - all jobs are launched using a wrapper that captures runtime provenance 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

# 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

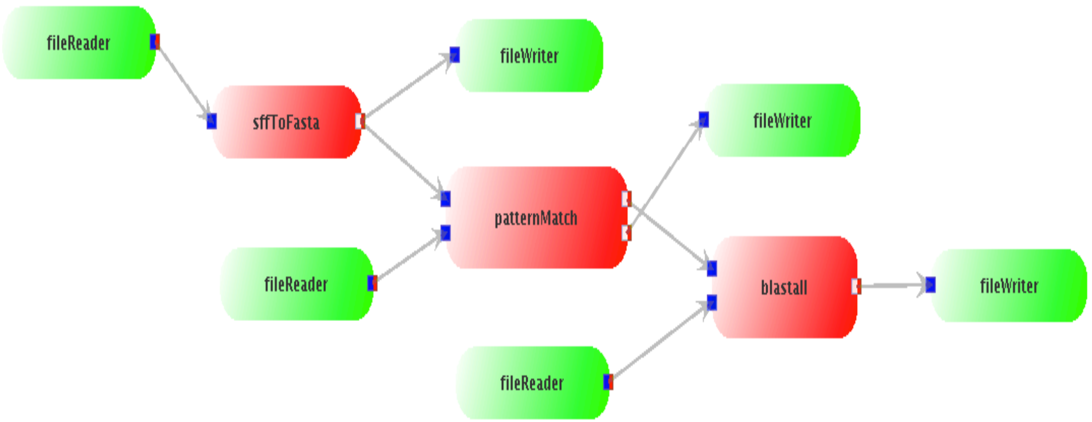
# History-tracing XML (FH Aachen)

- 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.

```
<patternMatch>
  <events>
    <PortResolved> provenance
data</PortResolved>
    <ConDone>provenance data
             </ConDone>
    ...
  </events>
  <fileReader2>
    <events> ... </events>
    <sign-fileReader2> ...
          </signfileReader2>
  </fileReader2>
  <sffToFasta>
    Reference
  </sffToFasta>
  <sign-patternMatch> ...
         </sign-patternMatch>
</patternMatch>
```

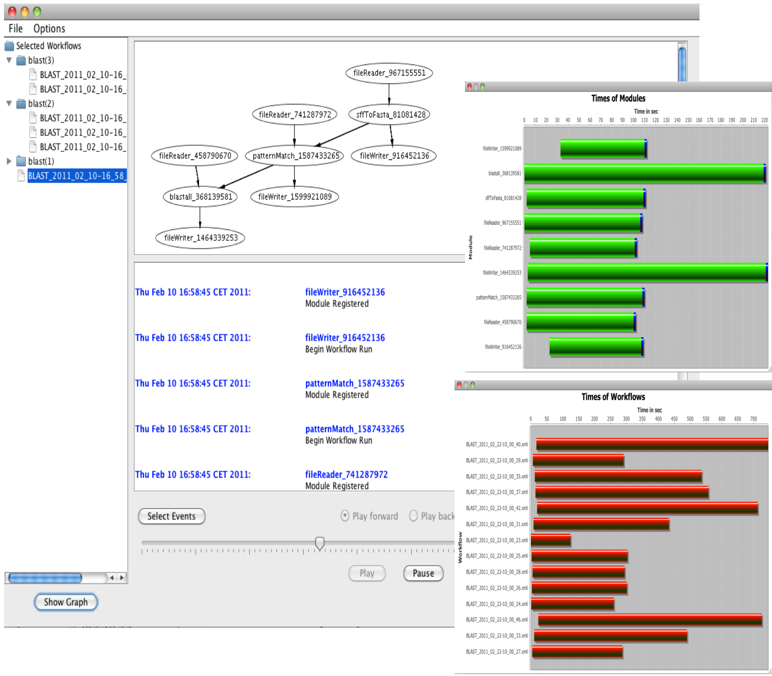
M. Gerards,  Adam S. Z. Belloum, F. Berritz, V. Snder, S. Skorupa, **A History-tracing XML-base Proveannce Framework for workflows**, WORKS 2010, New Orleans, USA, November 2010

# 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.

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




For Each workflow run

The provenance data is collected an 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

on-going work UvA-AMC-fh-aachen

# 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
3. A. Belloum, Z. Zhao, and M. Bubak Workflow systems and applications , Future Generation Comp. Syst. 25 (5): 525-527 (2009)
4. Z. Zhao, A.S.Z. Belloum, et al., Distributed execution of aggregated multi domain workflows using an agent framework The 1st IEEE International Workshop on Scientific Workflows, Salt Lake City, U.SA, 2007
5. Zhiming Zhao, Adam Belloum, Cees De Laat, Pieter Adriaans, Bob Hertzberger Using Jade agent framework to prototype an e-Science workflow bus Authors Cluster Computing and the Grid, 2007. CCGRID 2007