ASTERISM

AN INTEGRATED, COMPLETE, AND OPEN-SOURCE APPROACH FOR RUNNING SEISMOLOGIST CONTINUOUS DATA-INTENSIVE ANALYSIS ON HETEROGENEOUS SYSTEMS

Rafael Ferreira da Silva
rafsilva@isi.edu

Rafael Ferreira da Silva[1], Rosa Filgueira[2], Ewa Deelman[1], Malcolm Atkinson[3]

[1]University of Southern California, Information Sciences Institute, USA
[2]NERC-British Geological Survey, Lyell Centre, UK
[3]University of Edinburgh, DIR, UK

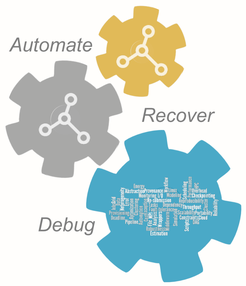
WHY SCIENTIFIC WORKFLOWS?

General Features and Workflow Management Systems

Abstraction, scientists can focus on their research and not computation management

Easy composition and execution

Enables parallel, distributed computations



Many scientific workflow management systems (WMS) have been developed, and they have been intensively used by various research communities, e.g., Earth sciences, astronomy, biology, computational engineering, climate modeling, etc.

ASTERISM FRAMEWORK

Open-source Platform-independent Framework

Greatly simplifies the effort required to develop **data-intensive applications** that run across multiple heterogeneous resources distributed in the wide area:

- Manages the data-distribution across systems
- Parallelizes the user's methods
- Co-places and schedules methods with computing resources
- Stores and transfers large/small volumes of data

How ?

By combining the strengths of

- Traditional Workflow Management Systems
- New stream-based data-flow systems

dispel4py + Pegasus

Automation
Automated pipeline executions
Parallel, distributed computations
Automated data transfers
Heterogeneous resources

Recovery & Debug
Job failure detection
Job Retry
Real-time monitoring

Optimization
Job clustering
Data cleanup

Pegasus

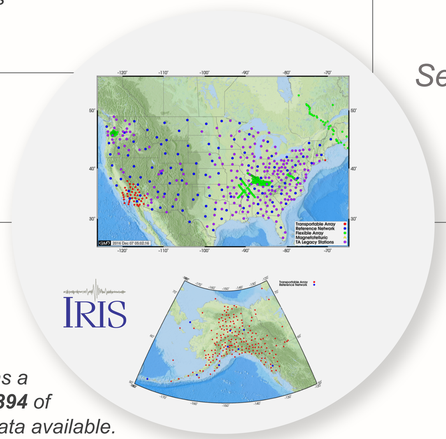
dispel4py

Mapping
Sequential
Multiprocessing
MPI
Apache Storm and Spark (Prototype)

Automation
Automated pipeline executions
Concurrent, distributed computations
Stream-based model

Optimization
Multiple streams
Avoids I/O operations

*Scientific workflow requests data from **IRIS services** (USArray TA)*



Seismic Ambient Noise Cross-Correlation

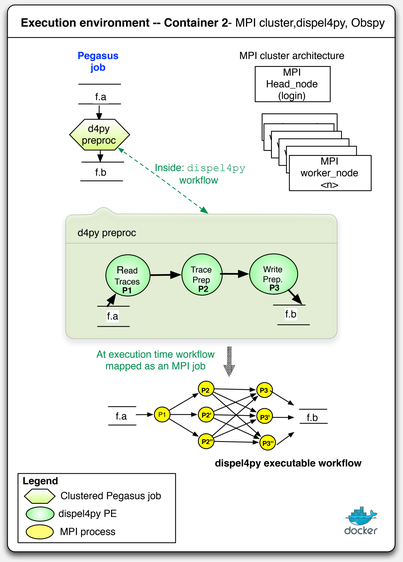
*The **USArray TA** has a list of 836 stations–**394** of them have online data available.*

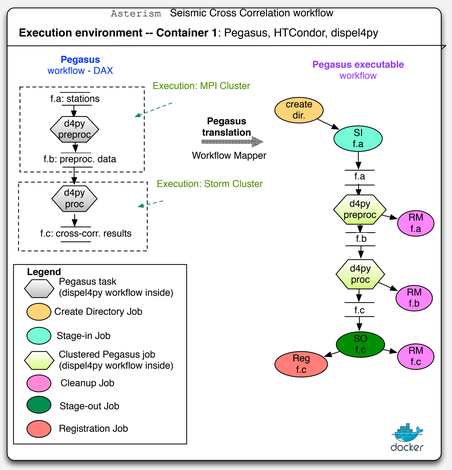
EXECUTION OF THE SEISMIC AMBIENT NOISE WORKFLOW

IRIS Data Services, Containers, and Cloud Computing

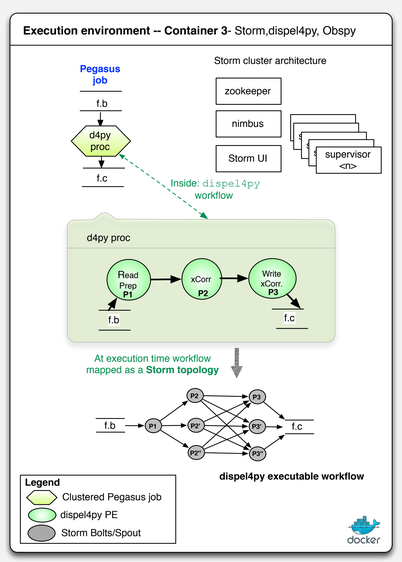
1 instance as Container 2 (MPI head node)

16 instances as Container 2 (MPI workers)





1 instance as Container 1



3 instances as Container 3 (zookeper, nimbus, Storm UI)

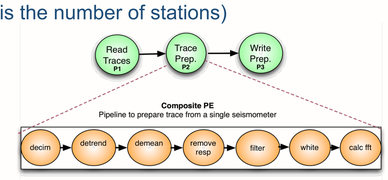
16 instances as Container 3 (Supervisors)

WORKFLOW APPLICATION

Seismic Ambient Noise Cross-Correlation

Pre-Process (Phase 1)

Each time series from a seismic station (each trace), is subject to a series of data- preparation treatments chosen and parameterized by seismologists; and the processing of each trace is independent from other traces. (complexity O(n), where n is the number of stations)



Cross-Correlation (Phase 2)

For all pairs of stations compute the correlation, essentially identifying the time for signals to travel between them, and hence infer some, as it turns out time varying, properties of the intervening rock. The complexity of this phase is $O(n^2)$

ObsPy
A Python Framework for Seismology

EVALUATION

IRIS Data Services, Containers, and Cloud Computing

Experiment 1

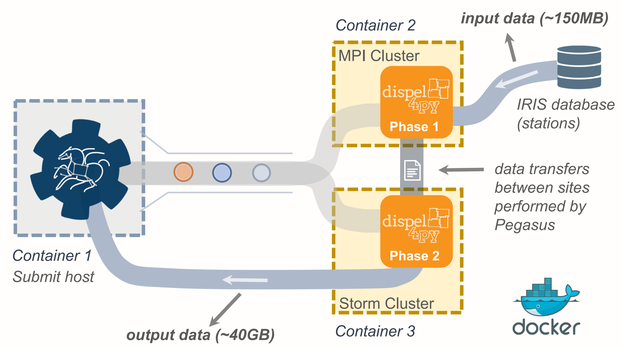
Data from IRIS services (394 stations)

Performance
Phase 1: 8 minutes
Phase 2: 2 hours
Data Movement: less than 1 minute

Data Statistics
Input Data: 150 MB
Output Data: 40GB

Experiment 2

Workflow runs for 3 days requesting data every 2 hours



Scope of the Work

Automated execution and parallelization of data-intensive applications in heterogeneous systems with different enactment engines

LEARN MORE

Pegasus Website
https://pegasus.isi.edu

Dispel4py GitHub
https://github.com/dispel4py