An Automated Infrastructure to Support High-Throughput Bioinformatics

Gianmauro Cuccuru, Simone Leo, Luca Lianas, Michele Muggiri, Andrea Pinna, **Luca Pireddu**, Paolo Uva, Andrea Angius, Giorgio Fotia, Gianluigi Zanetti

CRS4

July 23, 2014

What are we presenting?

- How we automated the analysis of data produced by our sequencing facility
- Solution in production use since July 2013
- Reduced human resources required from 4 to 1 full-time individuals
- Built from a combination of open-source tools and purpose-built components

Outline

Motivation

- A data-intensive revolution is underway in the biosciences (e.g., NGS)
  - due to cheaper and faster data acquisition
- Lower data acquisition costs bring larger datasets and larger studies
- Effectively processing in this context requires:
  - keeping track of data provenance
  - minimizing operational costs
  - high processing throughput

Current tools do not yet provide good support for this type of operation

Why at CRS4?

CRS4 Sequencing and Genotyping Platform

Currently the largest sequencing center in Italy

- Sequencing Equipment: 3 Illumina HiSeq2000, 2 GA IIx
- Sequencing Capacity: about 5 Tbases/month
- Computing facility (3200 cores, 4.5 PB parallel, shared storage)
- Lab has been used for complex large-scale genetic studies, producing:
  - over 2000 whole-genome and tumour samples
  - over 1000 RNA and exome samples
  - thousands of microarrays, Sanger capillary reactions and expression profiles, etc.

The challenge of scaling

Efficient data processing challenges are perhaps the most evident:

- high throughput
- storage capacity

But there are many more logistic and data management challenges:

- Increasing operator effort
- Increasing samples and datasets increases logistic complexity
  - tracking data provenance
  - reliably documenting analysis procedures
  - finding and accessing data files

Increasing complexity increases risk of errors!

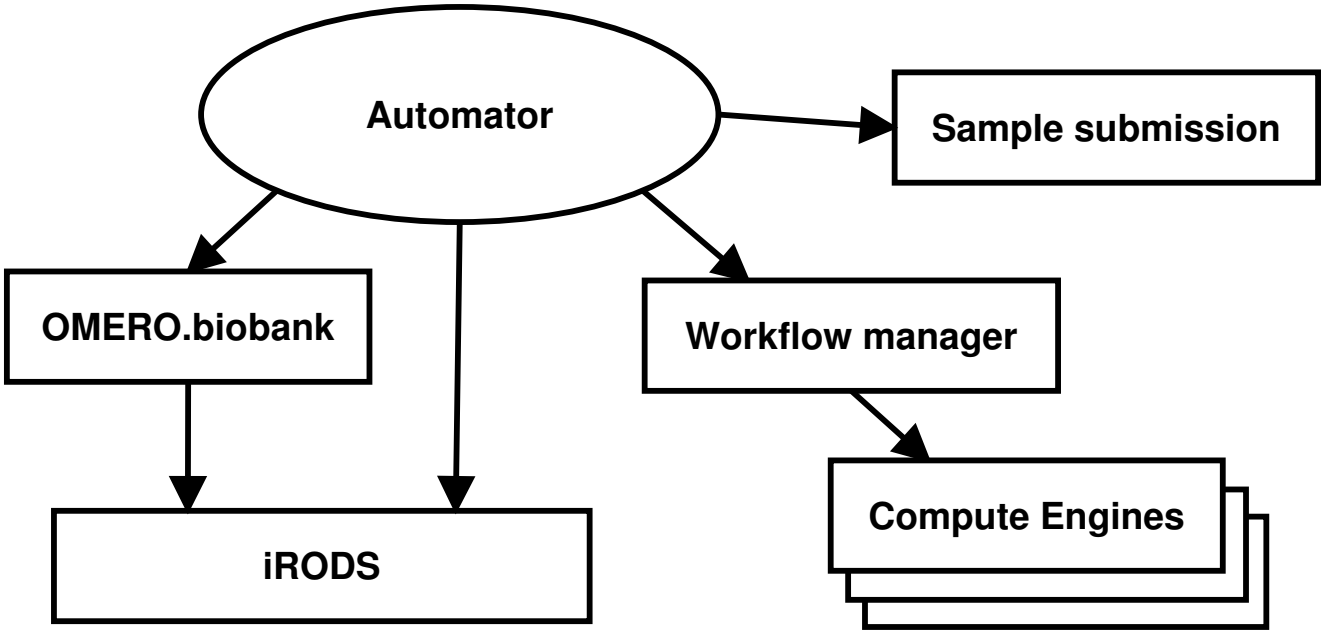
Outline

Our solution

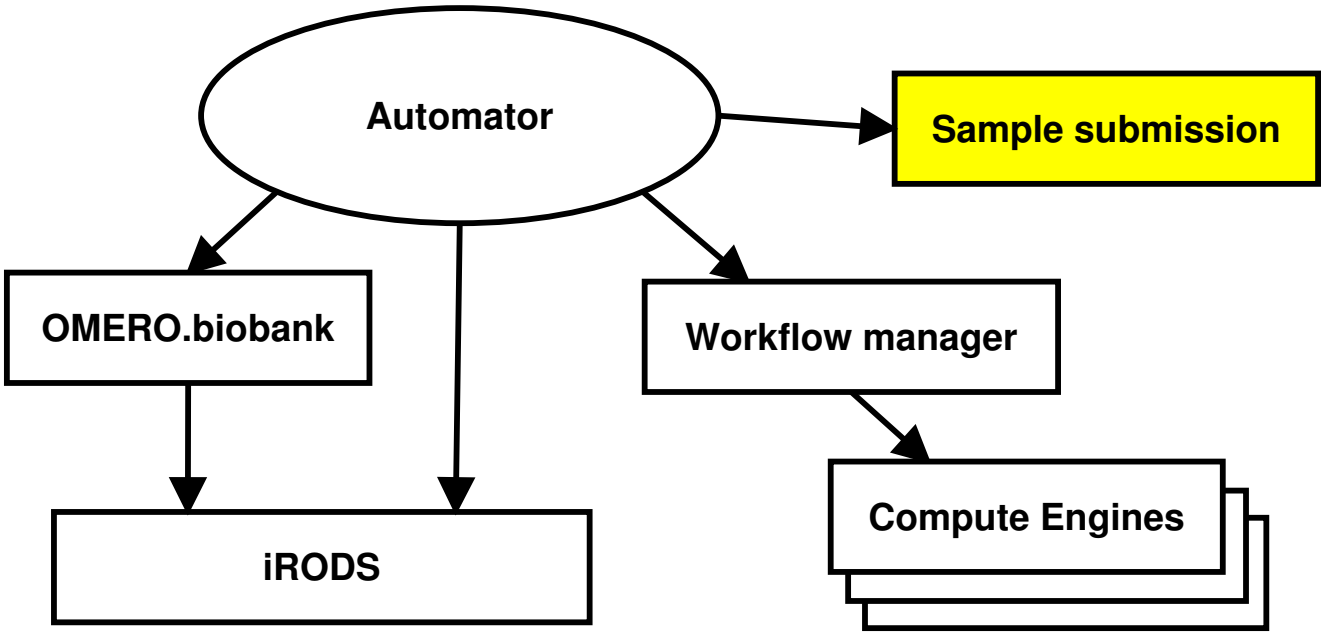
- A system to autonomously perform standard primary data processing
- Scales through these main features:
  - monitors data acquisition devices
  - automatically executes standard processing procedures
  - keeps provenance information for all generated datasets
  - achieves sufficient processing throughput

Overall architecture

Six main components:



Sample Submission

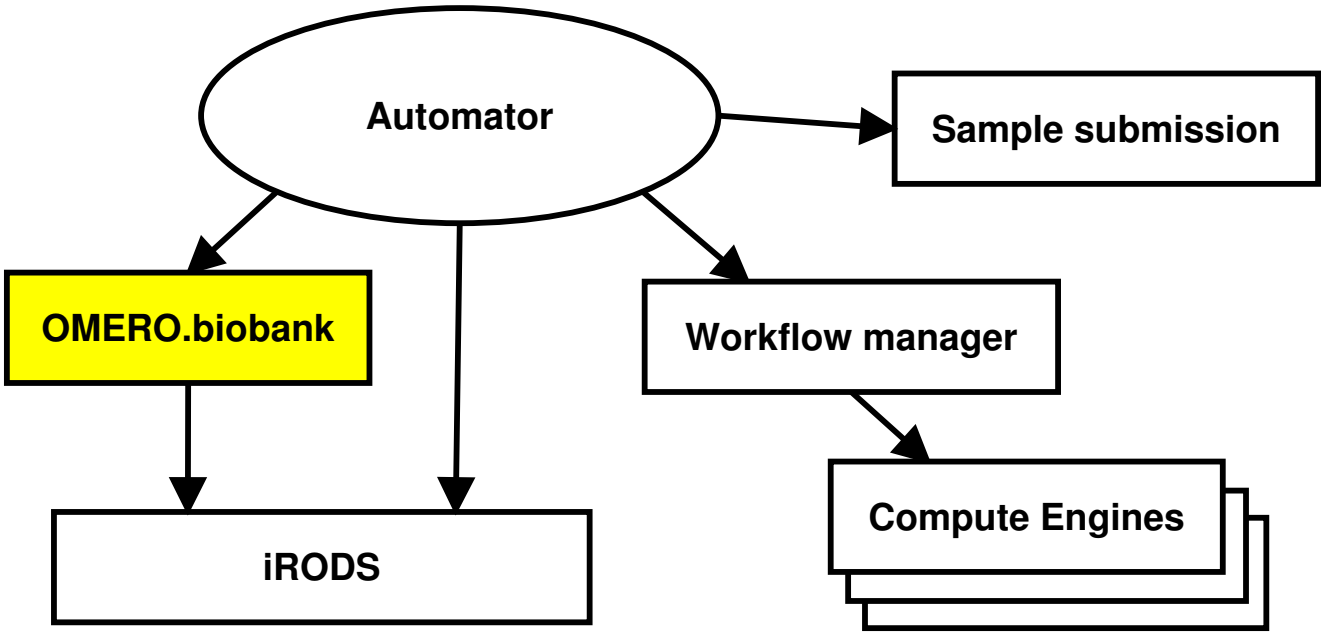


Sample Submission

Electronically track samples and the procedure to be applied to them

- We built a sample submission and tracking system to supplant manual tracking
  - based on the Galaxy sample tracking platform
- Reduces errors by catching them early through automated validation
- Web service interface to provide data for downstream operations
- Original Galaxy system extended in a number of ways:
  - most importantly, let personnel monitor the status of the sequencing and analysis operation
    - support "customer" relationship management

OMERO.biobank



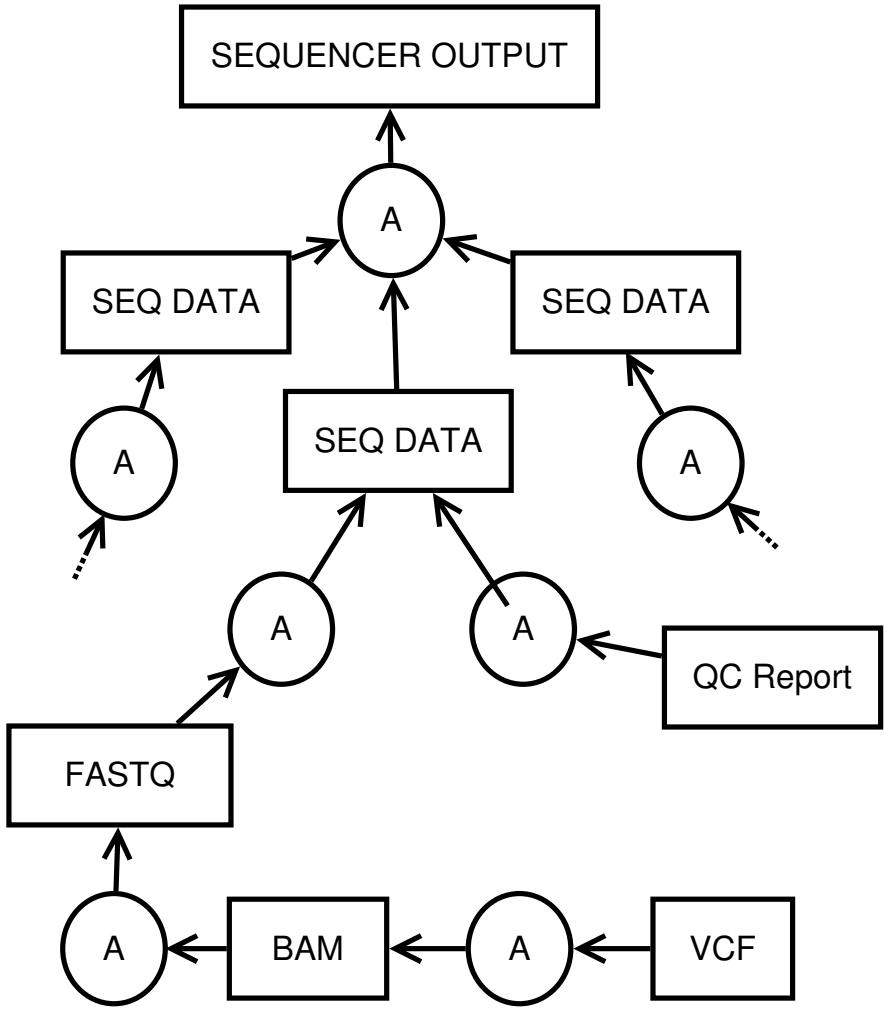
OMERO.biobank

Document all datasets created and all operations performed on them

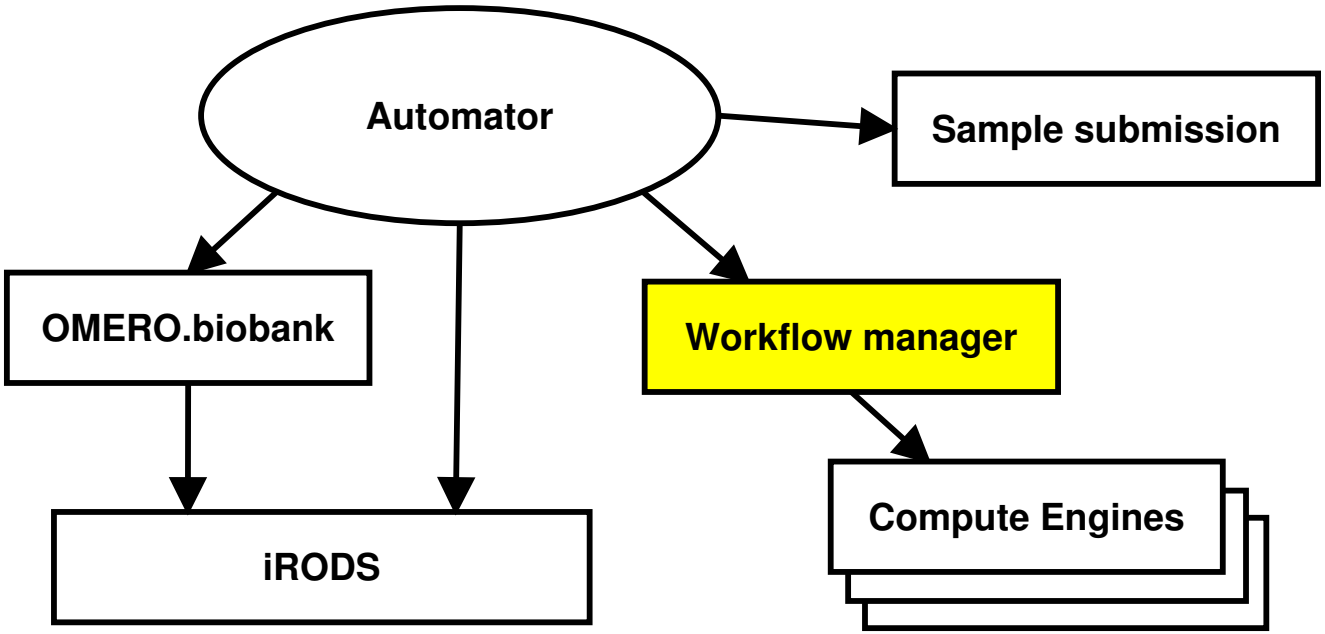
- Developed at CRS4 on top of OMERO and Neo4J
- Implements the model of a specialized graph database with two types of nodes
  - entities: metadata about datasets, including path/URI and data format
  - actions: operations which transform input entities into output ones
    - i.e., serialized workflow, with all parameters
- Simple and expressive data model keeps it from binding to specific data flow pattern

Example: part of the traceability graph produced by an exome processing workflow.

- Rectangles and circles represent entities and actions, respectively.



Workflow Manager



Workflow Manager

Execute and record operations on data

- Analyses require procedures consisting of many steps and parameters
- Subject to variation as knowledge is refined
- Workflow Manager can give us a serialized version of the history of operations performed
  - . . . which can be reinstantiated and executed

Workflow Manager

Our workflow management system is based on the Galaxy bioinformatics platform

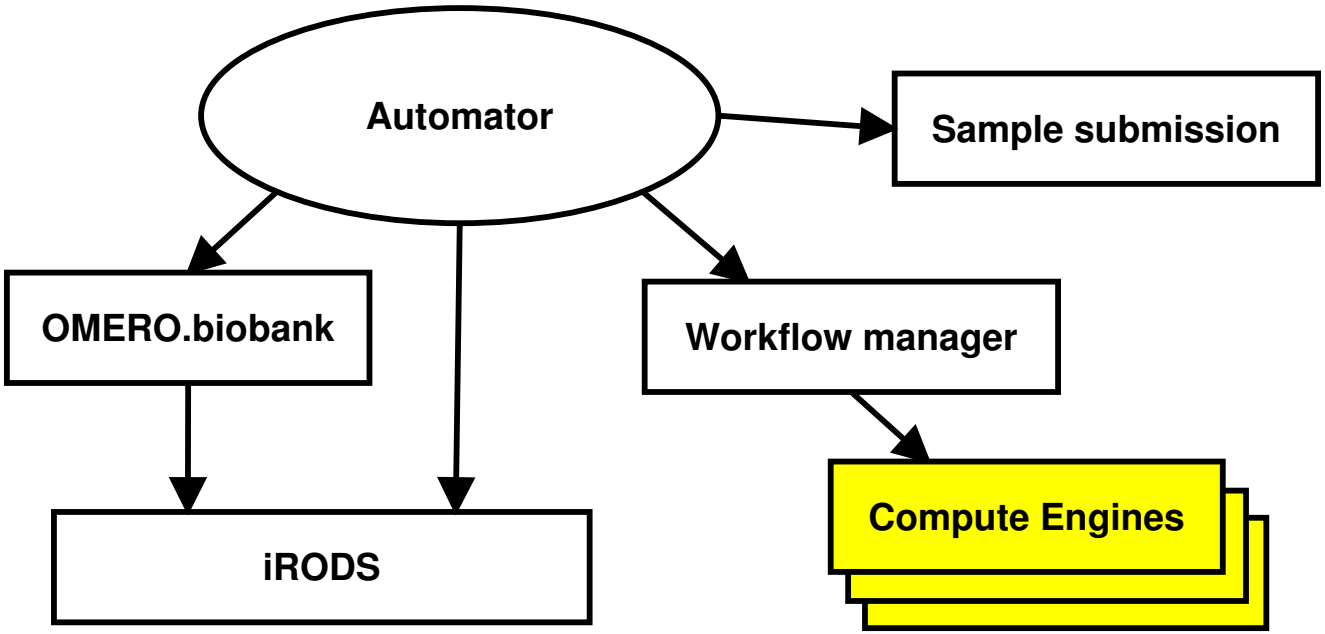
Orione: a highly customized Galaxy instance

- `http://orione.crs4.it/`

Coupled with:

- Hadoop-Galaxy
  - `https://github.com/crs4/hadoop-galaxy`
- Bioblend.objects: Python API to programmatically access and control Galaxy
  - `https://github.com/crs4/bioblend`

Compute Engines



Compute Engines

Scalable computational strategies for time-consuming steps

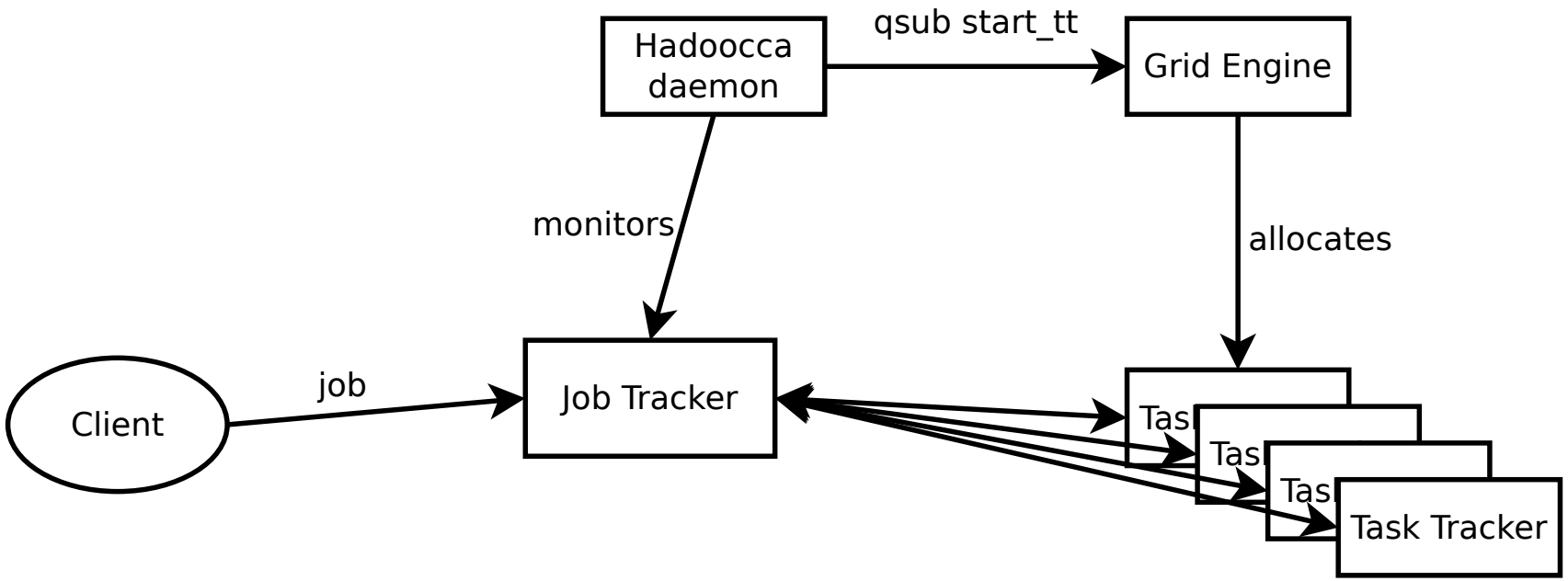
- High data production rates impose a scalable computation strategy
- We have reimplemented computational bottlenecks on the Hadoop platform
  - Standard sequence processing: Seal (`https://github.com/crs4/seal`)
  - Specialized or ad hoc data analysis and processing tasks:
    - Pydoop (`https://github.com/crs4/pydoop`)
    - SeqPig (`https://github.com/HadoopGenomics/seqpig`)

Hadoocca

- Hadoop helps us achieve computational scalability
- But... it doesn't play nice with our shared HPC cluster

- Using Hadoop in conventional batch queue HPC settings is an open issue
- CRS4 has just such a shared HPC cluster, and no possibility for a dedicated Hadoop installation
  - ... nor would the bursty workload justify it
- To helps us make these two worlds coexist we developed *Hadoocca*

On-demand Hadoop node allocation



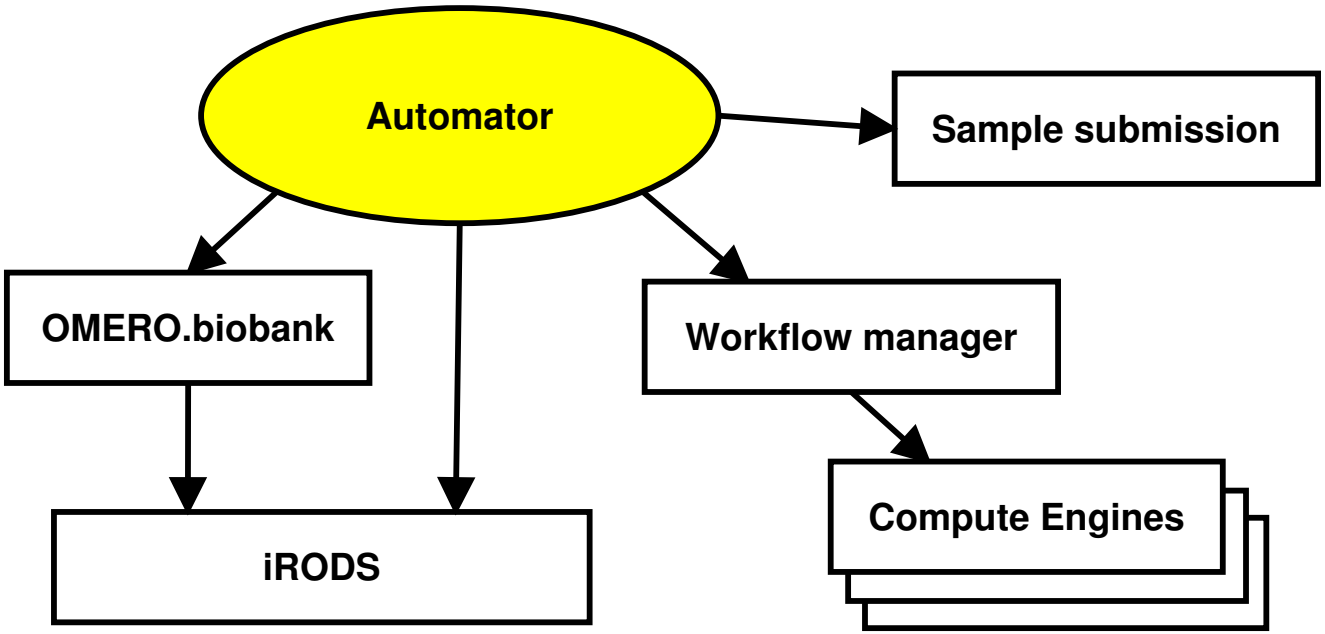
Hadoocca – About doing without HDFS

Note that our Hadoocca setup foregoes HDFS

- Instead, we rely on a conventional shared parallel file system

- Gives us the flexibility to deallocate nodes without moving data
- Also, storage capacity not dependent on allocated nodes
- For medium and small clusters, on our hardware, our tests showed comparable performance
  - parallel copying large files
  - only one disk per node
  - compute node I/O performance not spectacular
  - common situation in HPC clusters

Automator


Automator
Sample submission
OMERO.biobank
Workflow manager
iRODS
Compute Engines

Automator

Glue it all together and drive the system

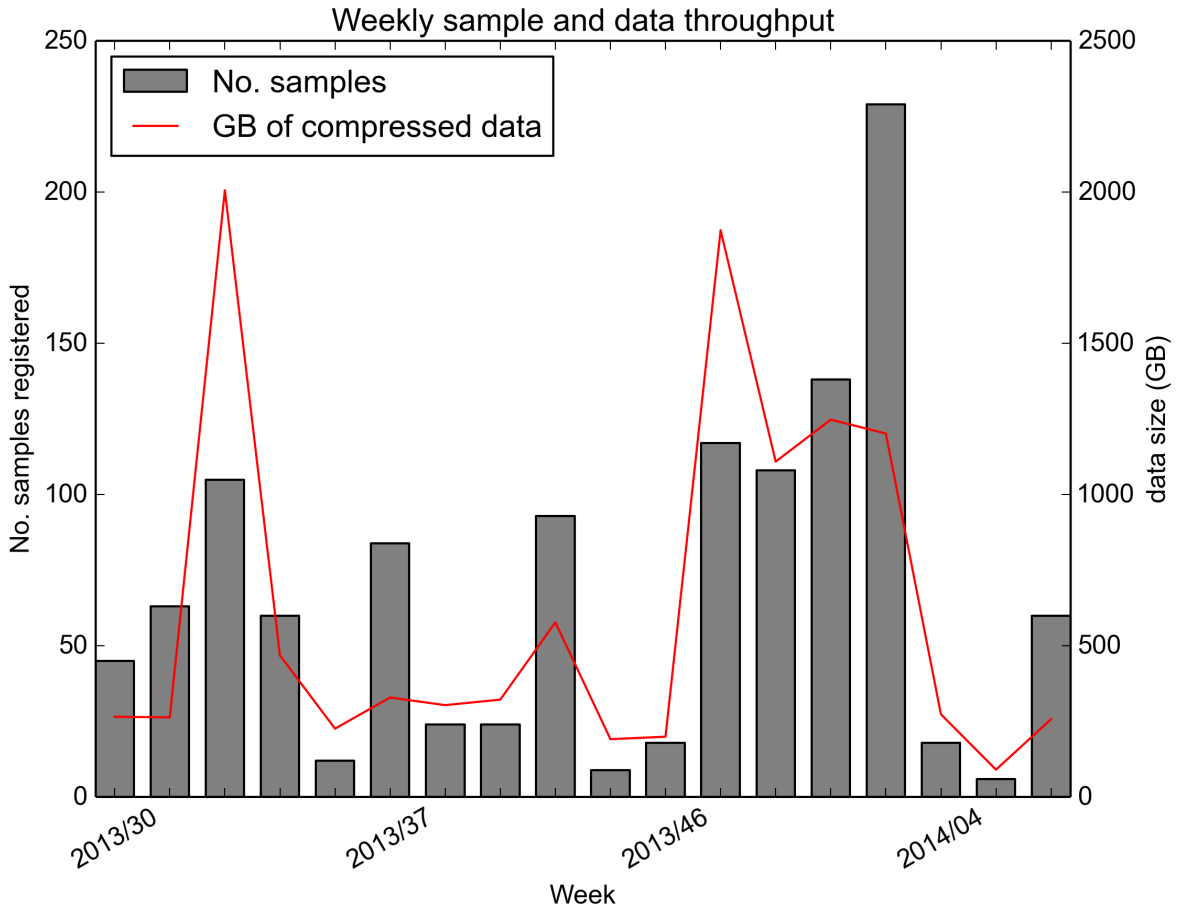
- Custom-made controller and middleware that drives the entire system
- Distributed event dispatching system
  - Central event queue built with RabbitMQ
  - Distributed design allows multiple instances to run concurrently
- Custom event handlers perform high-level data handling steps
  - e.g., launch workflow, move files, notify people, emit further events
- Includes modules to interface with other system components

The automator does not operate directly on the data!

- Data handling operations are expressed as workflows executed through the workflow manager.

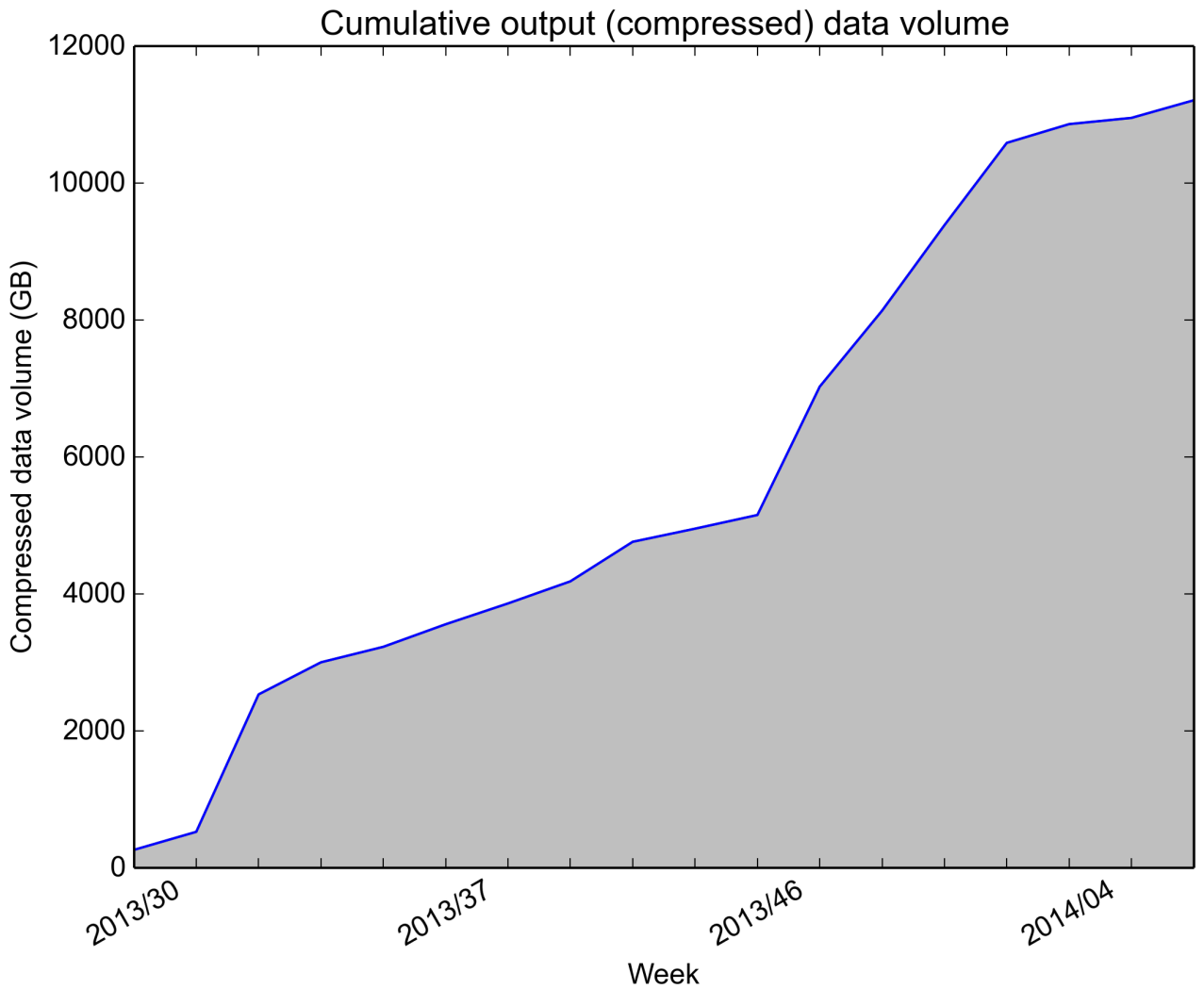
Outline

Throughput



- Output data is reduced to bare sequences with base quality scores and compressed
- Peaks of 2000 GB and almost 250 samples/week have been handled

Cumulative Data Growth



Accumulated gzip-compressed analysis-ready output data

Outline

Conclusion

- Large-scale biosci data acquisition has become accessible to small- and medium-sized facilities
- Automation is crucial to scale resulting data processing operation
  - automation and processing needs to be accessible to such smaller facilities
- It's all useless if we cannot document the provenance of our data
- The presented solution is the one in use at CRS4

Conclusion

- Large-scale biosci data acquisition has become accessible to small- and medium-sized facilities
- Automation is crucial to scale resulting data processing operation
  - automation and processing needs to be accessible to such smaller facilities
- It's all useless if we cannot document the provenance of our data
- The presented solution is the one in use at CRS4

That's all. Thank you!