

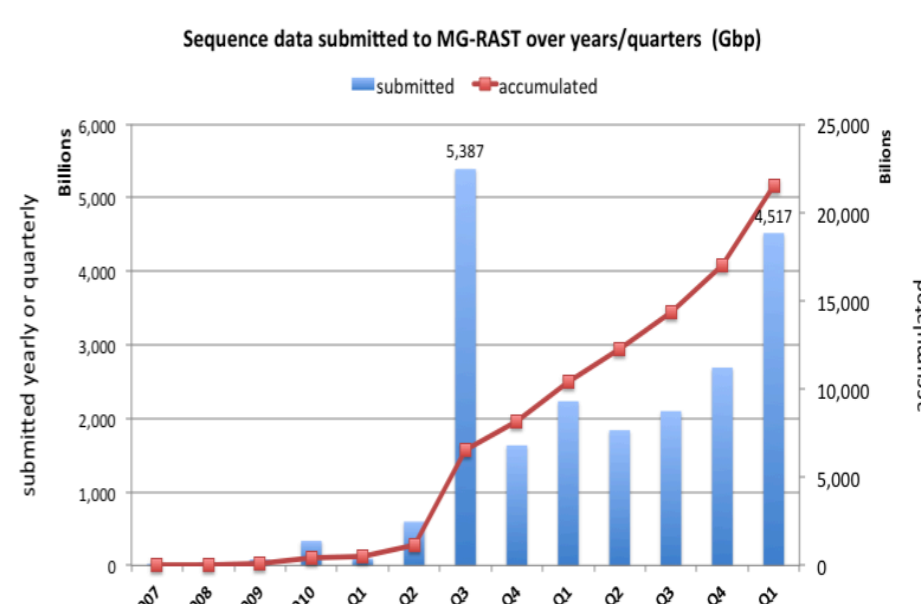
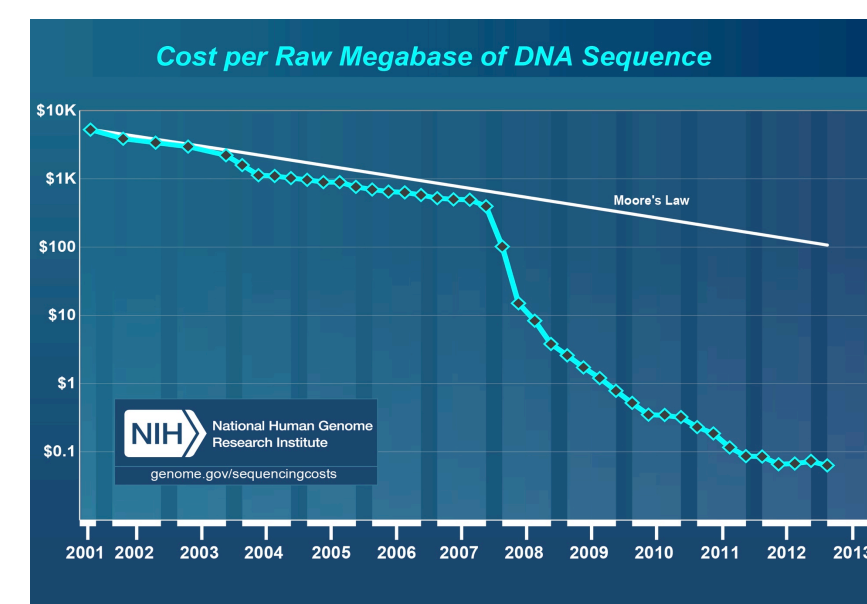
Building Scalable Data Management and Analysis Infrastructure for Metagenomics

Wei Tang, Jared Wilkening, Jared Bischof, Wolfgang Gerlach, Andreas Wilke, Narayan Desai, Folker Meyer

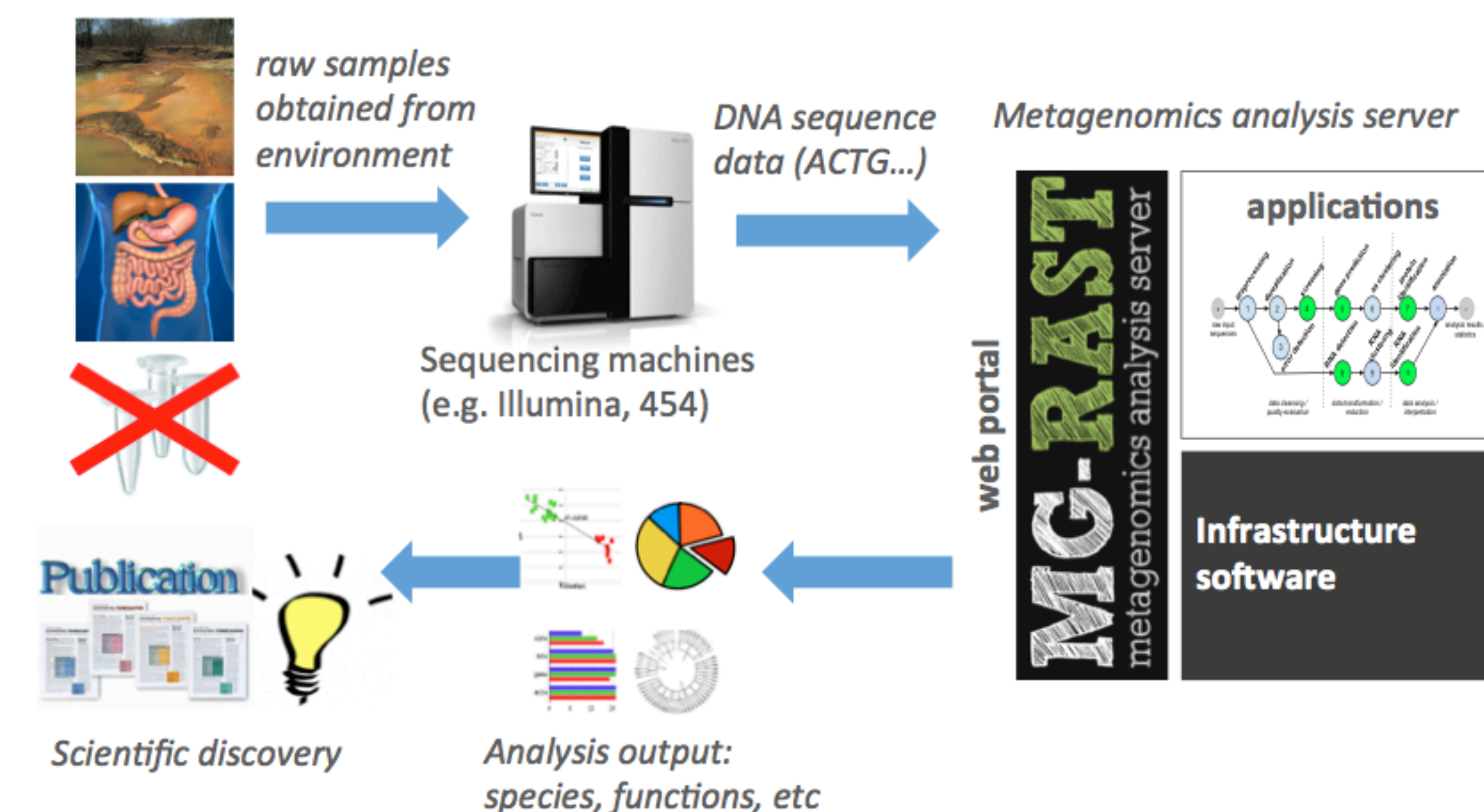
Argonne National Laboratory, USA

Motivation

With the advent of the next generation sequencing (NGS) technology, genomics and metagenomics data has become big, and the analysis and management of these data has become a bottleneck for scientific progresses. For example, MG-RAST, a public metagenome annotation server, has experienced tremendous growth of data submission for years, which has already impacted the efficiency of data process.



Typical procedure for metagenomics



MG-RAST: A free metagenome annotation server running at ANL, serving over 10,000 users worldwide since 2007.

As of 10/18/2013, MG-RAST has analyzed

# of metagenomes	95,283
# base pairs	36.04 Tbp
# of sequences	316.34 billion
# of public metagenomes	13,615

Problem caused by data deluge:

- Computing capability cannot keep up with the input data growth
- Neglecting reusable data product wastes lots of compute cycles

Goals of building new infrastructure

Scalability and Portability: Provide the capability of running analysis computation in scalable, distributed computing resources.

Reusability and reproducibility: Manage data in a proper way that all the data product can be shared, traced, reused, and reproduced.

Flexibility and customizability: Used for general data analysis workflows.

Building scalable infrastructure

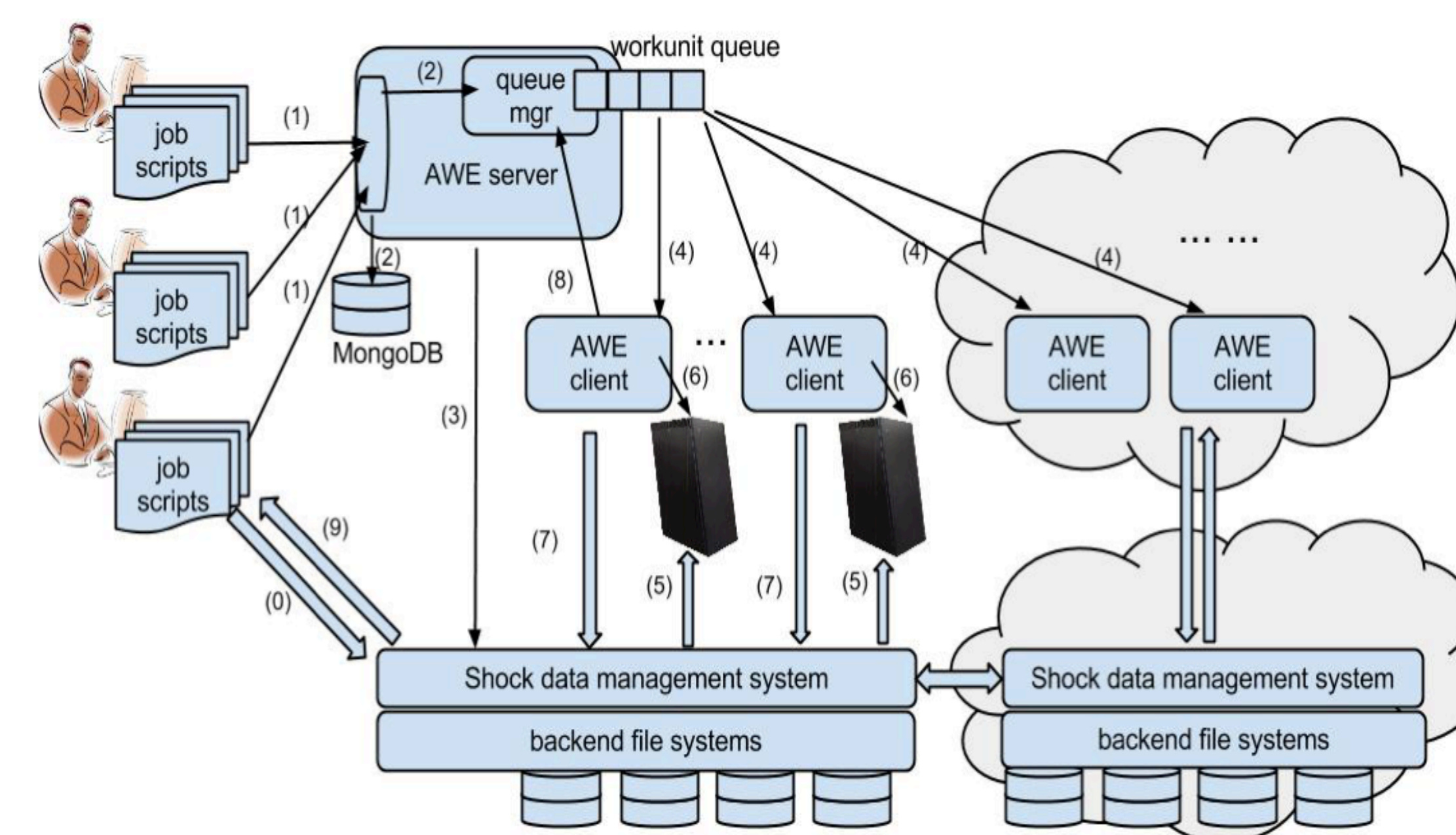
Shock data management system:

- Object-based data storage layer above filesystems
- Data object encapsulates data file, metadata, indexing information, etc.
- Useful interfaces for biological sequence data (e.g., subselection, splitting, and merging)
- Support metadata for both scientific and computational provenance
- Provide RESTful APIs for data management operations (query, upload, download, etc)

AWE workflow management system:

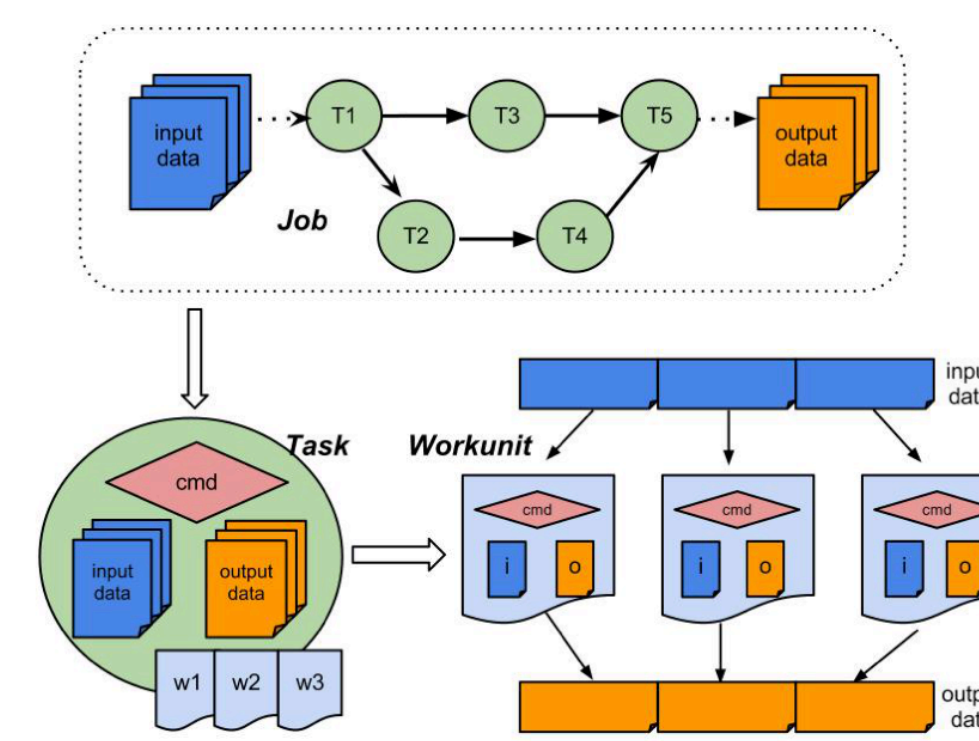
- Automatic execution of scientific workflow
- Support task parallelization, fault tolerance, automatic job recovery, etc
- Master/worker model, highly portable and scalable; flexible queue management
- Integrating workflow and resource management
- Use Shock layer for data management
- User programmable via REST APIs

Both Shock and AWE are open source software, available at [Github](https://github.com):



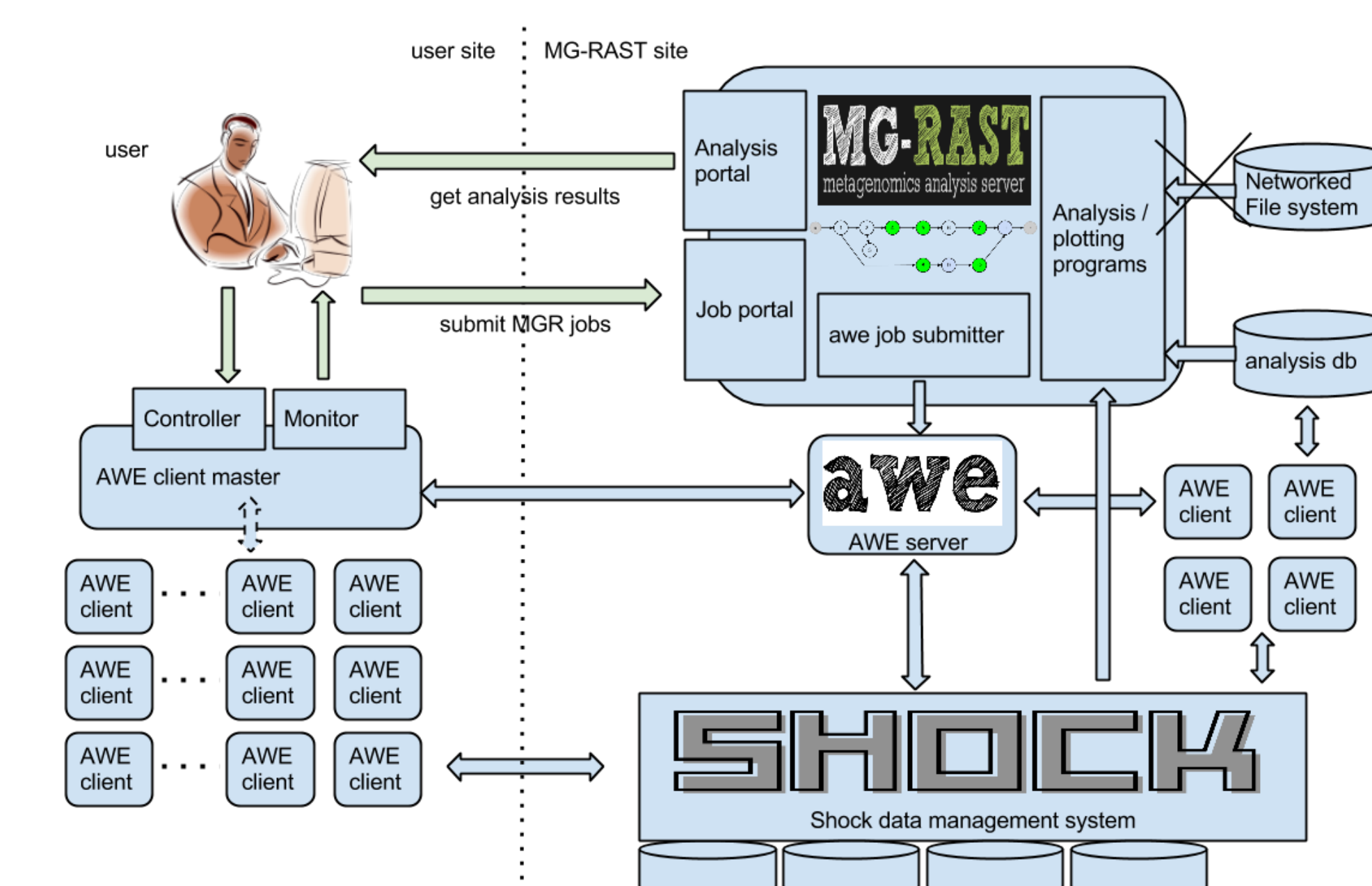
Working diagram. A centralized AWE server parses submitted jobs into tasks and workunits and manages them in a queue. The AWE clients, residing on distributed, heterogeneous, and scalable computing resources, keeps checking out workunits from the server and executing them locally. Shock is used for handling data, such as retrieval and storage, splitting and merge.

SHOCK
<https://github.com/MG-RAST/Shock>
awe
<https://github.com/MG-RAST/AWE>



Key elements. A job is parsed into tasks with data dependencies; a task can be split into multiple workunits running the same command on different data parts.

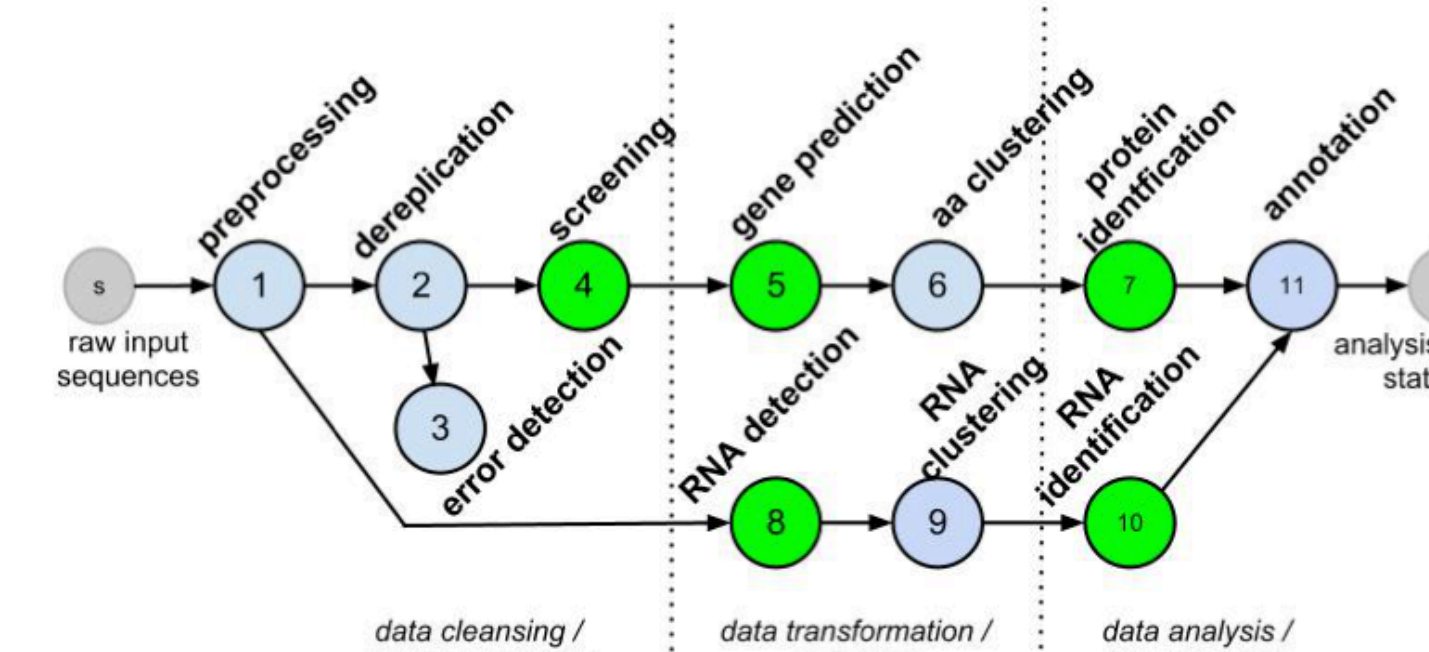
Accelerating analysis for MG-RAST using Shock & AWE



- User submits jobs to MG-RAST web portal and get results there
- AWE and Shock handles underlying workflow execution and data management
- Besides dedicated AWE clients, user can contribute their computing cycles by running AWE clients (clients delivered to users as VM images along with applications)
- Users with computing resources can achieve fast job turnaround

Case study

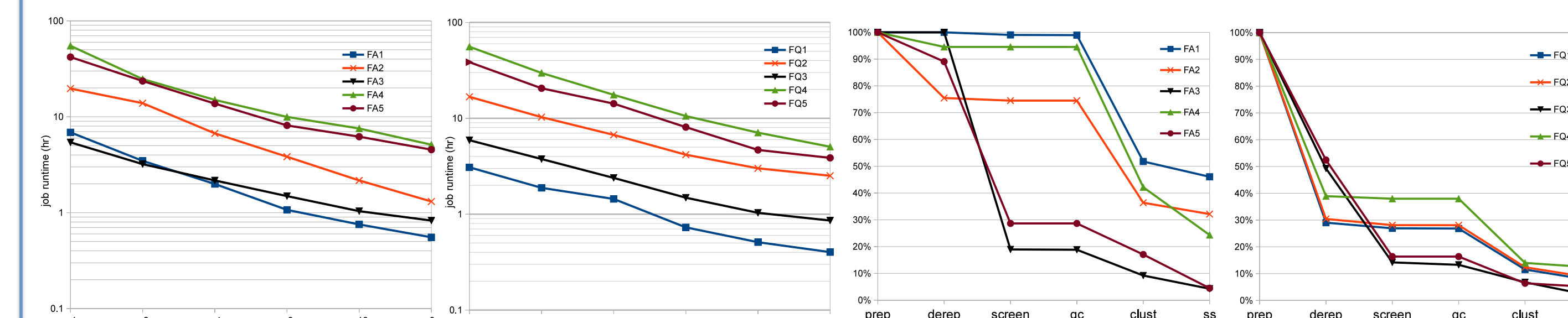
MG-RAST pipeline



Sample jobs from MG-RAST

Table 1: Job characteristics					
Format	Name	Size (Mbp)	Seqs	Avg Len	File Size
fasta	FA1	100	1.0	100	133 MB
	FA2	500	1.0	498	587 MB
	FA3	1000	10	100	1.36 GB
	FA4	2000	20	98	2.16 GB
	FA5	5000	51	98	8.15 GB
fastq	FQ1	100	0.99	100	337 MB
	FQ2	500	50	100	1.59 GB
	FQ3	1000	7.8	128	2.52 GB
	FQ4	2000	27	75	4.63 GB
	FQ5	5000	28	177	11.0 GB

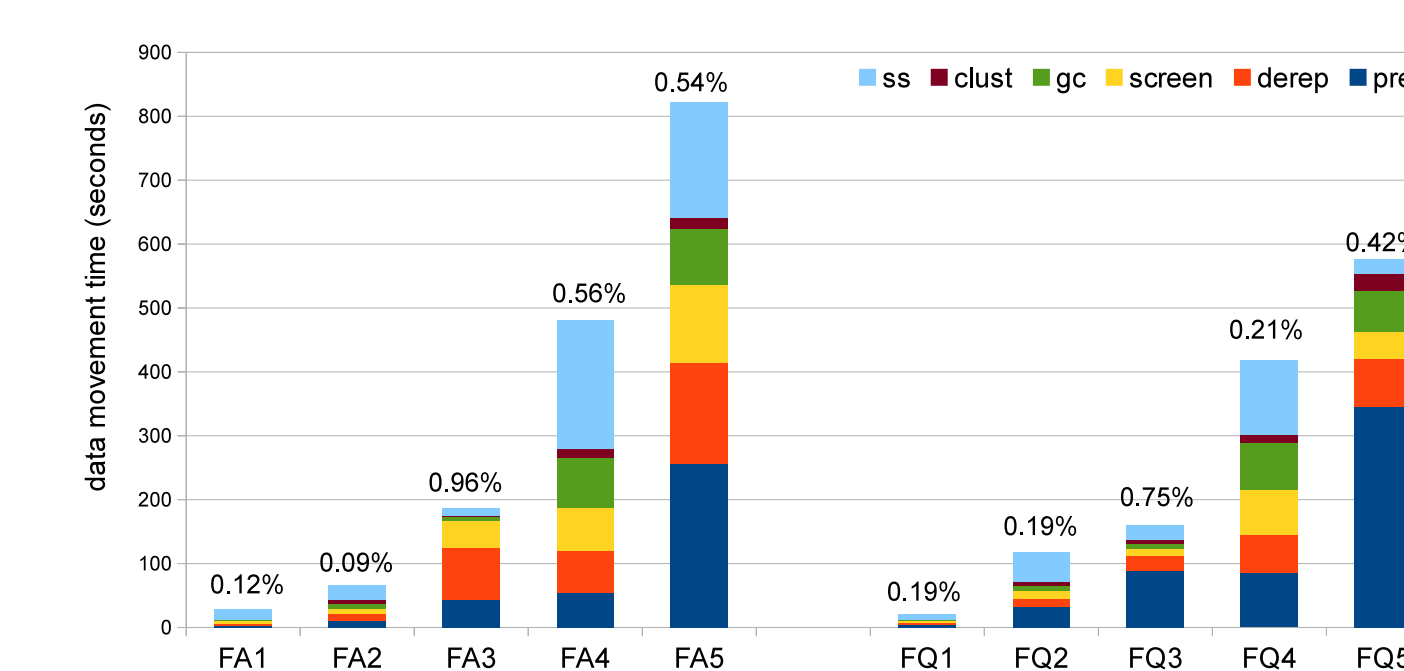
Single job evaluation (run individual sample job one by one)



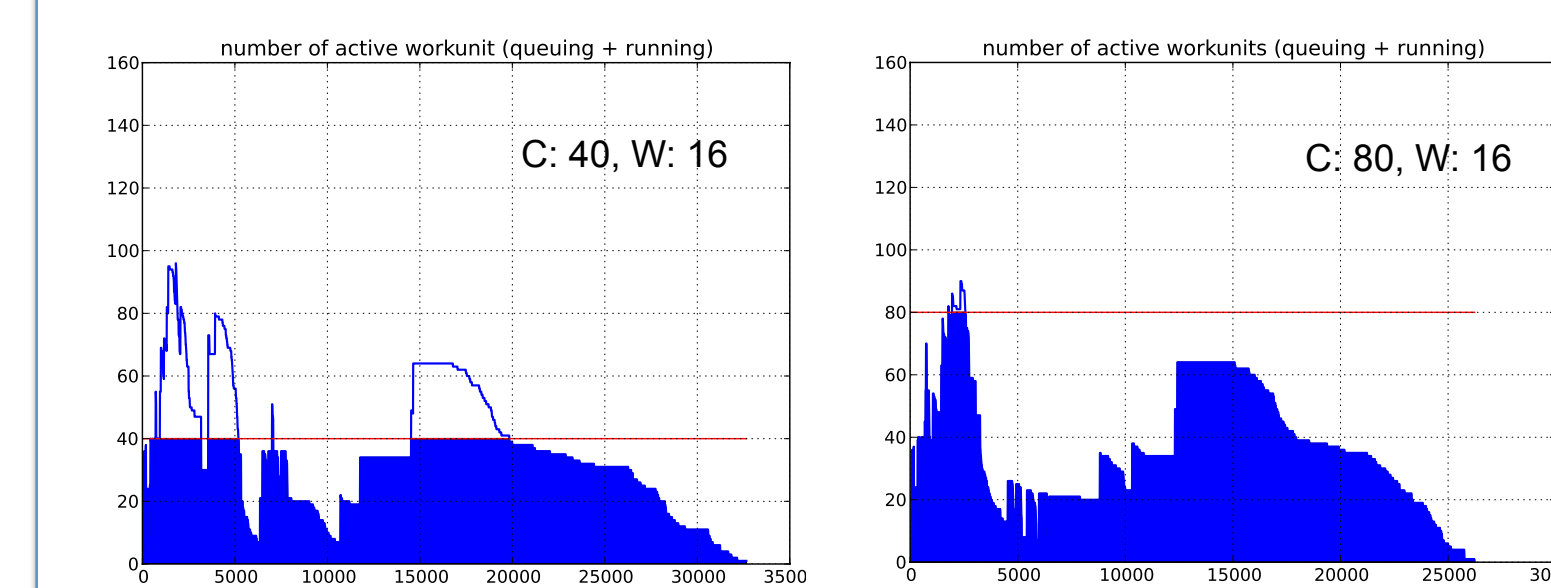
Job running time reduced as workunit splits increase. wN means splittable tasks (in green) will be split into N workunits

Data reduction at different task stages

Data movement overhead: data transfer time between AWE clients and Shock. (per task seconds and %job runtime)

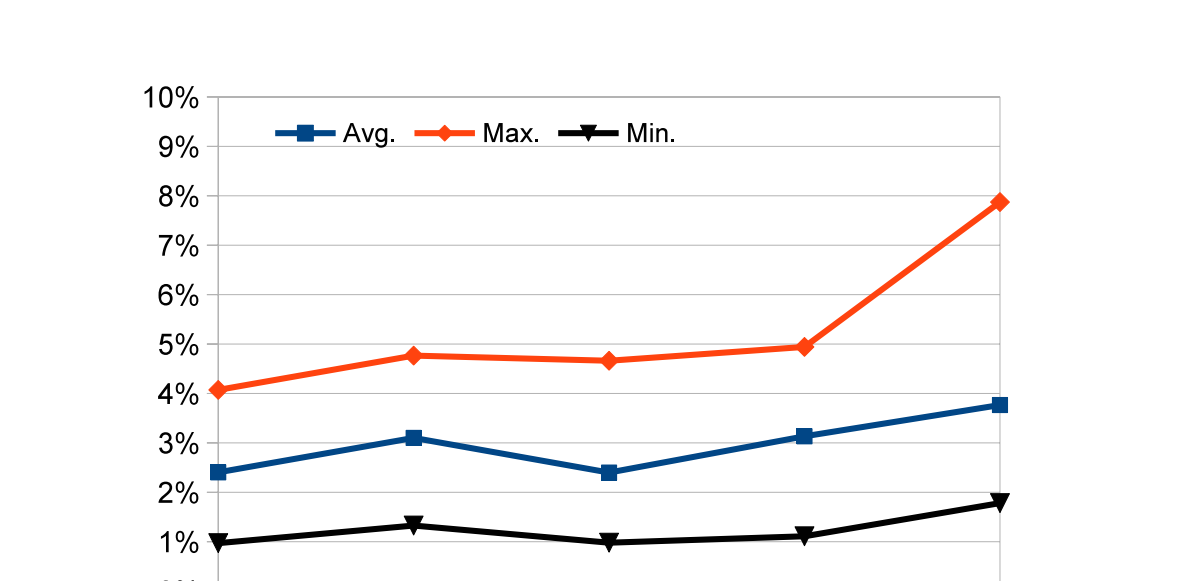
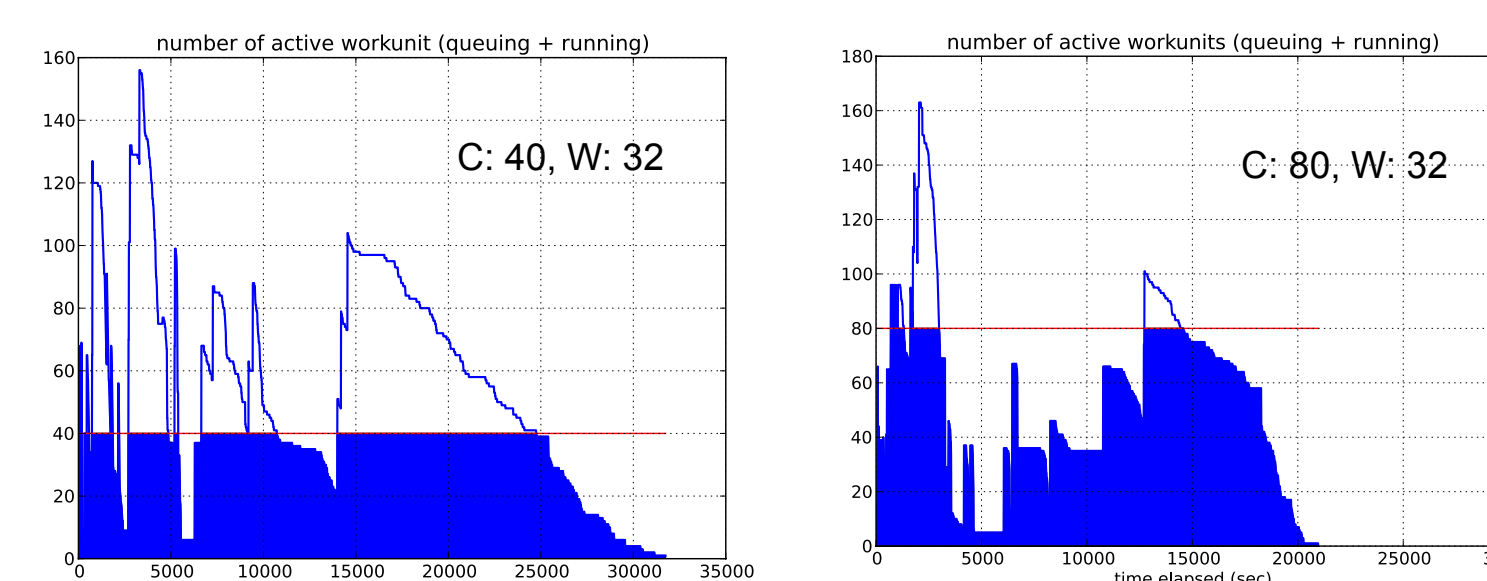


Multi-job workload evaluation (submitting all 10 sample jobs at time 0)



Active workunits by time (C: # of max clients, W: # of workunits split). Throughput can be increased by scaling up clients or splitting tasks.

Per job data movement overhead increases with C and W, which can be addressed by deploying multiple Shock, each serving a group of clients.



Future Work

- Workload characterization and dynamic computing resource provisioning to achieve optimized system throughput with budget control
- Data aware resource management and workflow scheduling

References

- [1] MG-RAST: <http://metagenomics.anl.gov>
- [2] F. Meyer et al., The metagenomics RAST server - a public resource for the automatic phylogenetic and functional analysis of metagenomes, BMC Bioinformatics, 2008
- [3] W. Tang, J. Wilkening, N. Desai, W. Gerlach, A. Wilke, and F. Meyer. A scalable data analysis platform for metagenomics. In Proc. IEEE International Conference on Big Data, 2013.
- [4] A. Wilke et al. A metagenomics portal for a democratized sequencing world. Methods Enzymol., 531:487–523, 2013.