Cloud-based Data Management

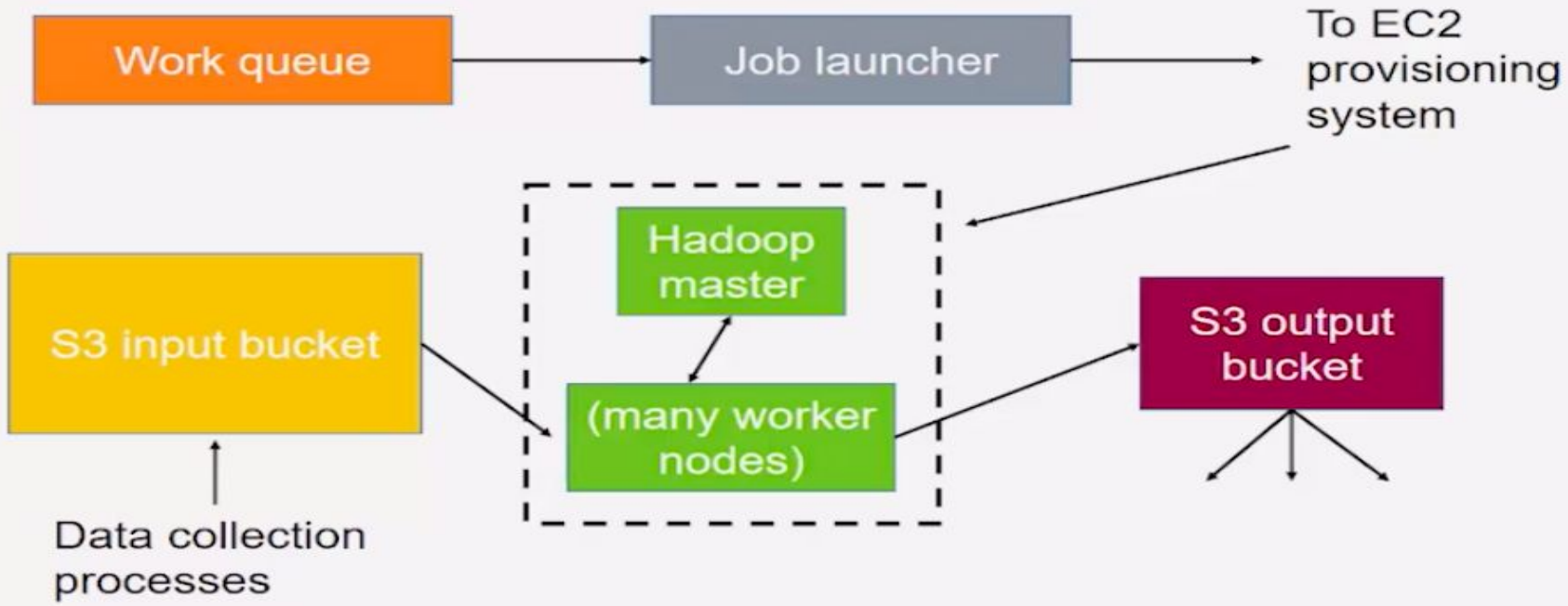
Scalable Cloud Data Processing

Ira A. Fulton Schools of Engineering
Arizona State University

Ming Zhao, Ph.D.

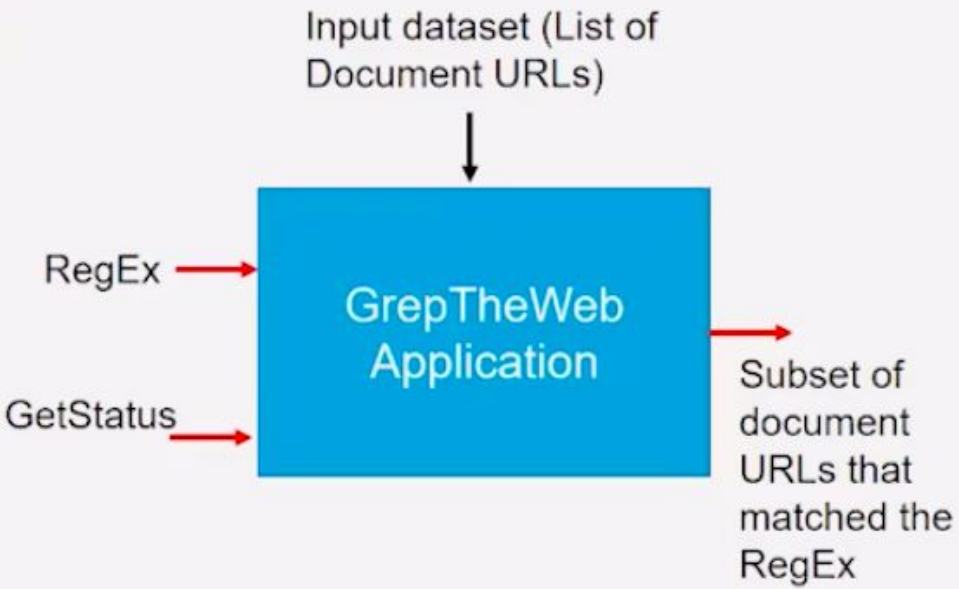
School of Computing, Informatics, & Decision Systems Engineering

Scalable Cloud Data Processing



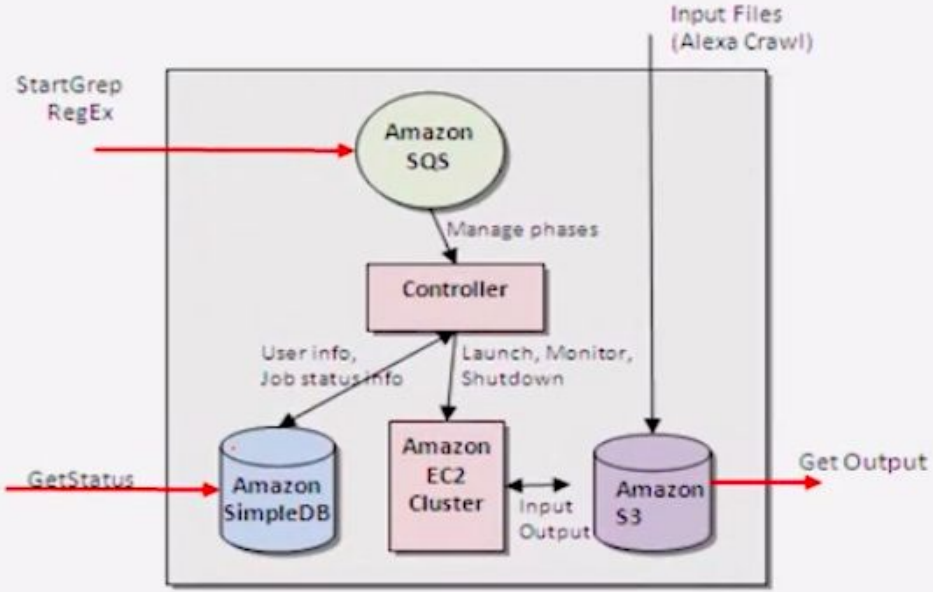
Example: GrepTheWeb

- Run regular expression against massive amount of web documents
- Challenges
  - Regular expressions could be complex
  - Dataset could be large
  - Unknown request patterns



High-level Architecture

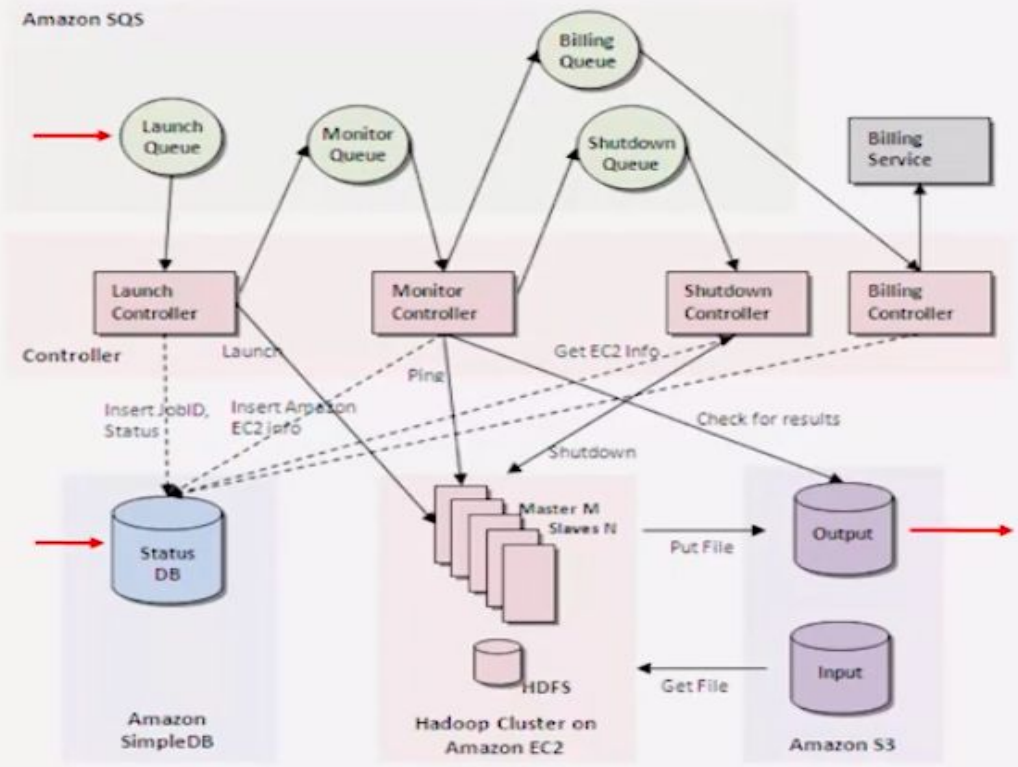
- **Amazon S3** for retrieving input datasets and for storing the output dataset
- **Amazon SQS** for durably buffering requests acting as a "glue" between controllers
- **Amazon SimpleDB** for storing intermediate status, log, and for user data about task



Workflow

- **Launch**: upon user request, execute the launch task
  - Update status in SimpleDB, start EC2 instances, start MapReduce
- **Monitor**: check Hadoop status periodically
- **Shutdown**: execute shutdown task
  - Kill Hadoop processes, terminate EC2 instances
  - The billing task calculates the billing
- **Cleanup**: archive SimpleDB data

Detailed Architecture



How EC2 was Used

- All the controllers run on EC2 instances
- The master and slave instances for Hadoop
  - Launched from preconfigured AMI

How Amazon S3 Was Used

Input

- The web documents stored on S3
- Huge (in terabytes) and always growing

Output

- The grep results

How Amazon SQS Was Used

Buffer

- Bridge the speed difference between sender and receiver
- Decouple sender and receiver and smooth out bursty traffic

Isolation

- Make entire system loosely coupled
- Provide a uniform way of transferring information between different components
- No component directly calls another component

Asynchrony

- One slow or failing component does not affect any other component
- Make the entire system stable and available

How Amazon SimpleDB Was Used

- Track the state of the system
  - SimpleDB is schema-less; every controller can define its own structure and append data to a "job" item
  - Launch controller adds/updates "launch_status"
  - Monitor controller adds/updates "monitor_status" and "hadoop_status"
  - …
  - Any component can query SimpleDB at anytime
- Store active Request IDs for historical and auditing/billing purposes