HYBRID CLOUD SUPPORT FOR LARGE SCALE ANALYTICS AND WEB PROCESSING

Navraj Chohan, Anand Gupta, Chris Bunch, Kowshik Prakasam, and Chandra Krintz

UCSB

Overview

- Google App Engine (GAE)
- GAE Analytics Libraries
- AppScale
- Hybrid
  - Data synchronization
  - Hive Analytics
- Evaluation
- Conclusion

Google App Engine

Google App Engine

- Platform-as-a-Service
  - Developers focus on their applications
  - Test locally then deploy on Google's infrastructure
- Language Support
  - Python 2.5 and 2.7
  - Java
  - Go
- 500,000 existing apps
- Auto-scaling, pay-as-you-go
  - Web requests, background tasks, and storage

Google App Engine APIs

Name	Description
Datastore	Key/Value object storage
Memcache	Distributed caching service
Blobstore	Storage of large files
Channel	Long lived JavaScript connections
Images	Simple image manipulation
Mail	Sending and receiving email
Users	Login service with Google Accounts
Task Queues	Background tasks
URL Fetch	Resource fetching with HTTP request
XMPP	XMPP-compatible messaging service

Google App Engine

- Restrictions to enforce scalability and security
  - Limited query support
  - Runtime restrictions
  - No socket access, no file system access
 - White-list of libraries

Data Analytics in GAE

Task Queues

- Background tasks of 10 minutes
  - `taskqueue.add(url='/path/to/my/worker')`
- Task names to prevent fork bombs
- Task Queue Chaining
  - Splitting up large background jobs is the burden of the developer
- GAE Analytics libraries abstract away the Task Queue (TQ)

Fantasm

- Based on the Task Queue
  - Uses memcache and DB to manage state
- State machine driven
  - Specified in YAML
- Iterate over a large dataset
- Fan-in to join data
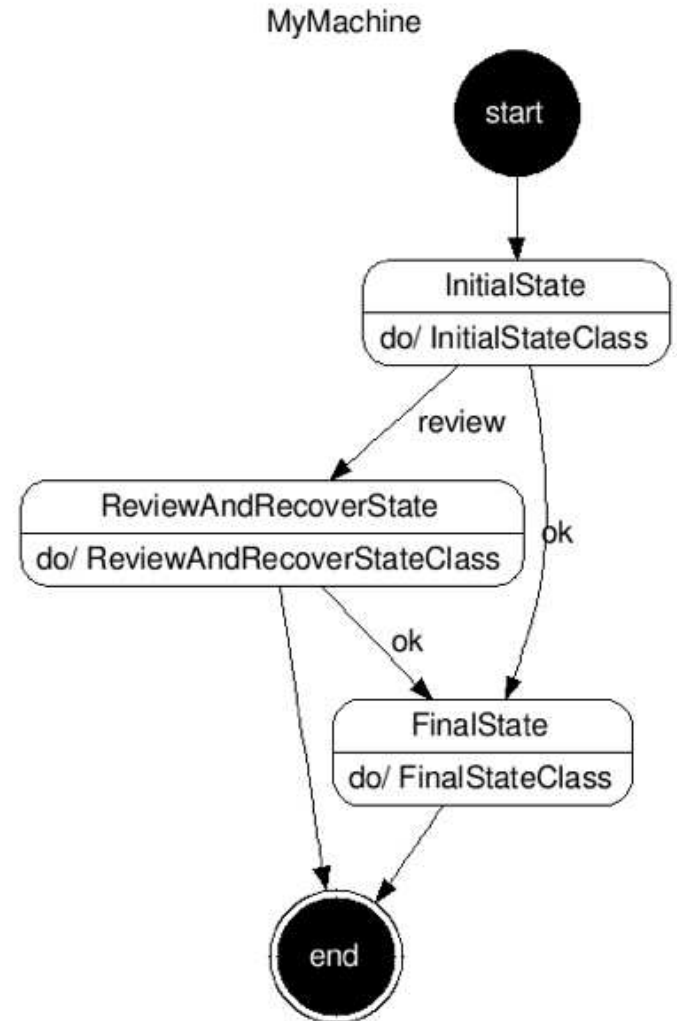
Pipeline

- Task Queue based
- Chains tasks into a workflow

```python
class Add(pipeline.Pipeline):
  def run(self, a, b):
    return a + b
class Multiply(pipeline.Pipeline):
  def run(self, a, b):
    return a * b
class LinearFunc(pipeline.Pipeline):
  def run(self, x, slope=1, offset=0):
    # y = m*x + b
    mx = yield Multiply(x, slope)
    yield Add(mx, offset)
```

GAE MapReduce

- Built on top of GAE infrastructure
  - Not Google's internal MapReduce or Hadoop
- Parallel processing and reductions on large datasets
- Map across a particular type of object
  - Must scan the entire type (no subsets)
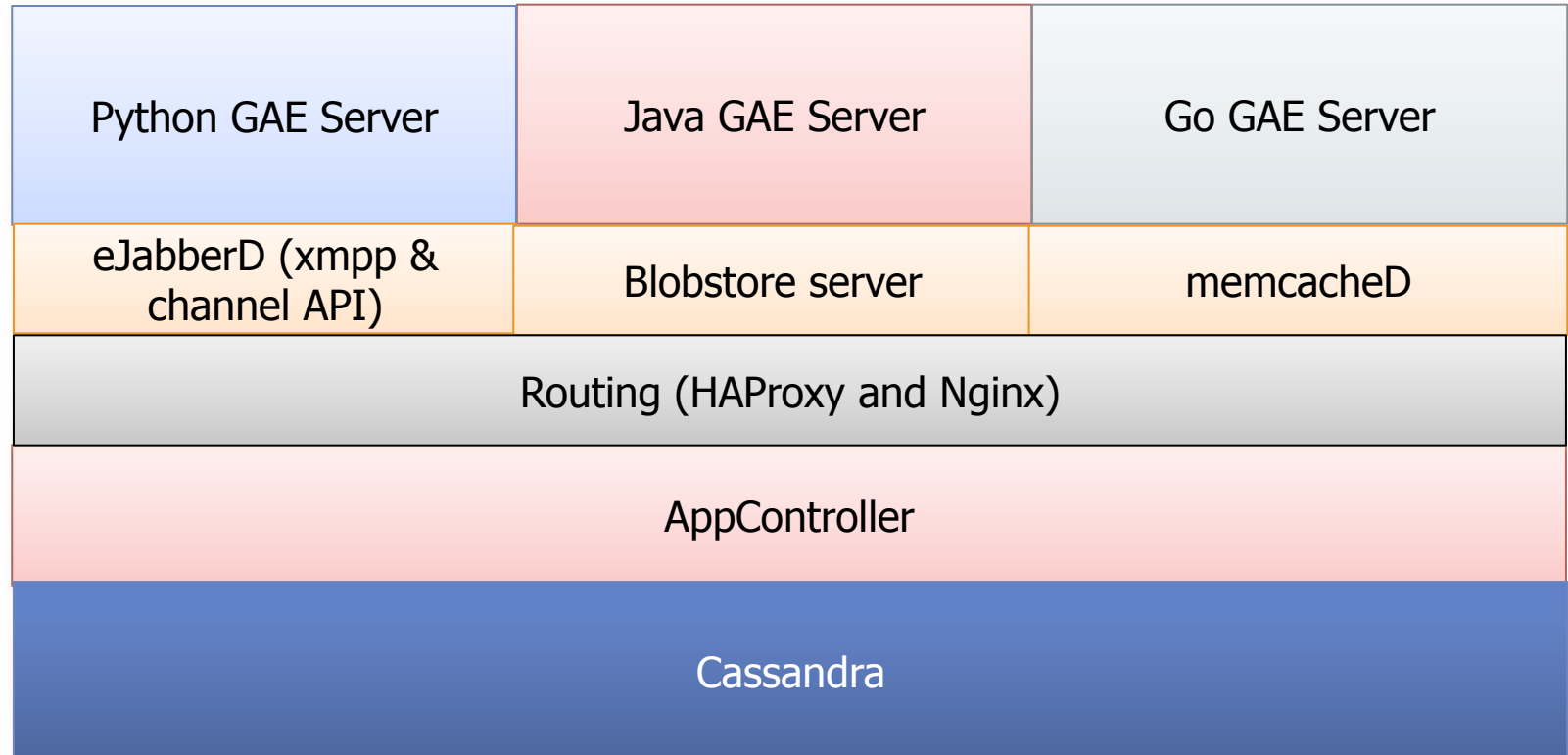- Multiple MR jobs can be linked with Pipeline
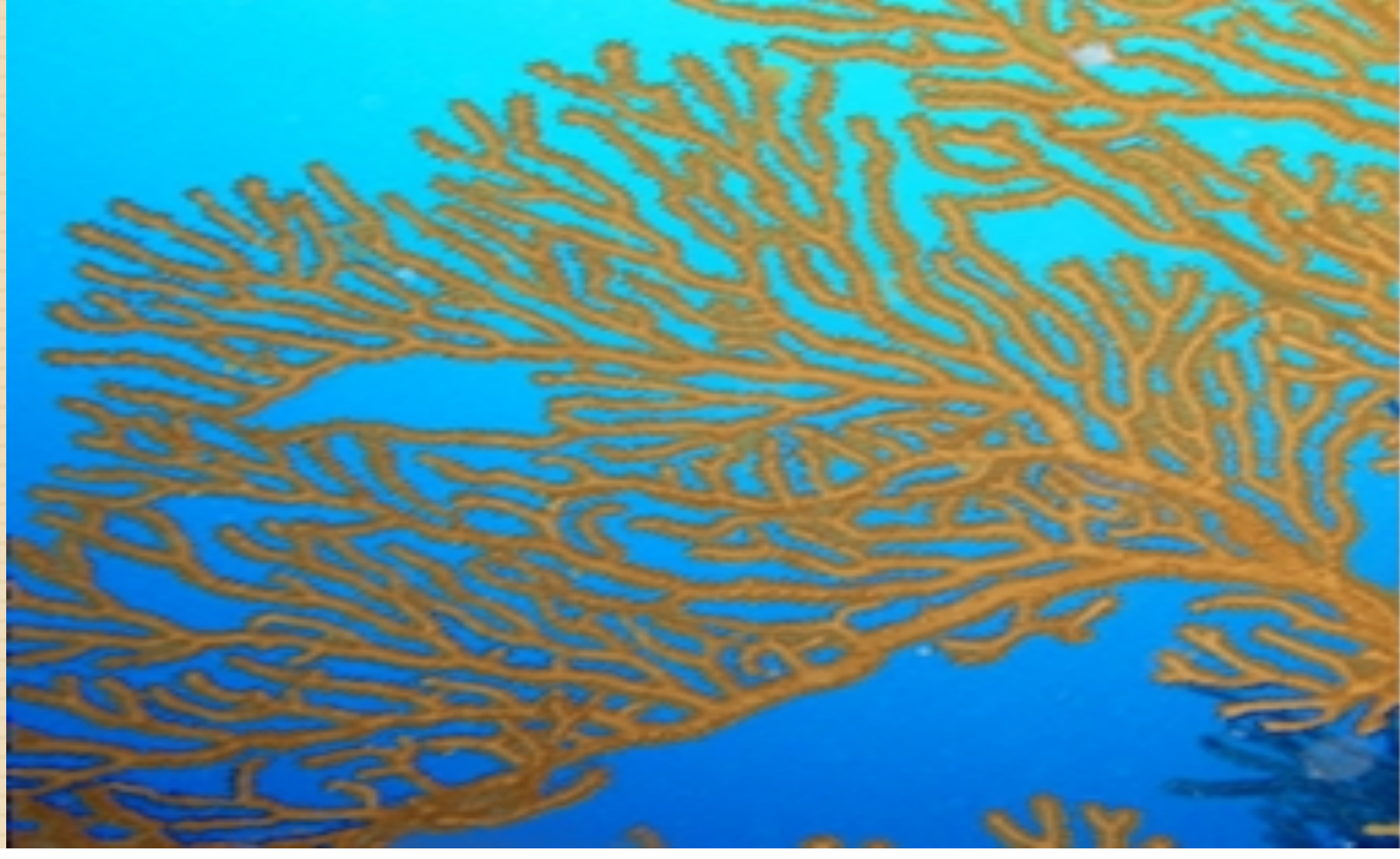
AppScale

AppScale

- Private PaaS with GAE API compatibility
  - Application portability
  - Engenders a developer community
- Distributed and fault tolerant API implementations
  - Leverages open source and new software systems
- Supports Python, Java, and Go languages
- Infrastructure agnostic
  - One virtual machine with all components
  - KVM, Xen, EC2, Eucalyptus, Openstack, etc
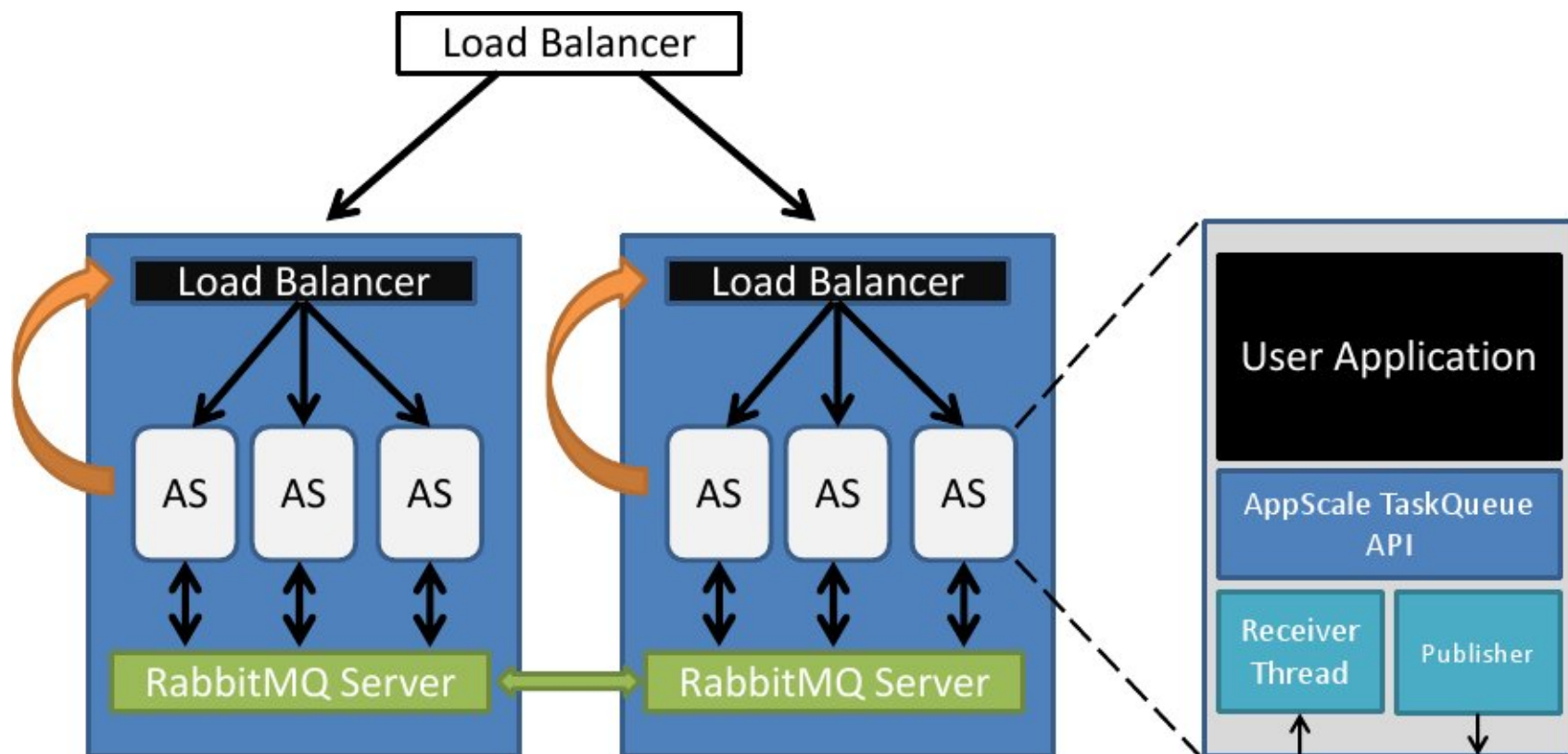- Datastore agnostic
  - Cassandra, HBase, Hypertable, etc

The AppScale Stack

Python GAE Server	Java GAE Server	Go GAE Server
eJabberD (xmpp & channel API)	Blobstore server	memcacheD
Routing (HAProxy and Nginx)		
AppController		
Cassandra		

GAE Analytics in AppScale

□ Task Queue with RabbitMQ

Analytic Libraries in GAE

- Learning curve for libraries
- Analytics must be part of application code
  - Can introduce bugs
  - Can disrupt the user experience
 - May require significant code and time to implement

AppScale and GAE Hybrid

Hybrid Solution

- Connect GAE to AppScale
  - Scale of GAE for OTAP
  - Flexibility of AppScale for OLAP
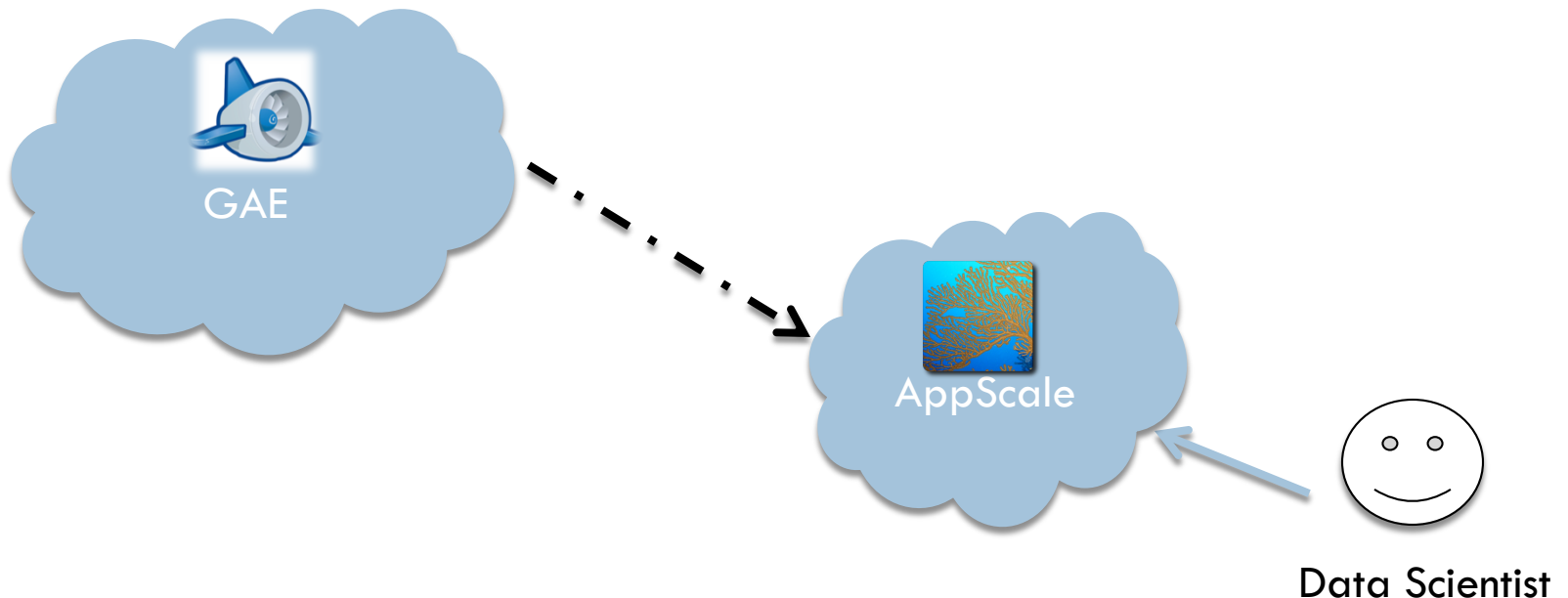- Contributions
  - Datastore mirroring
  - Hive queries

Datastore Library

- Datastore library for asynchronous updates
  - Best Effort (BE) with async URL Fetch
  - Eventual Consistency (EC) with Task Queue
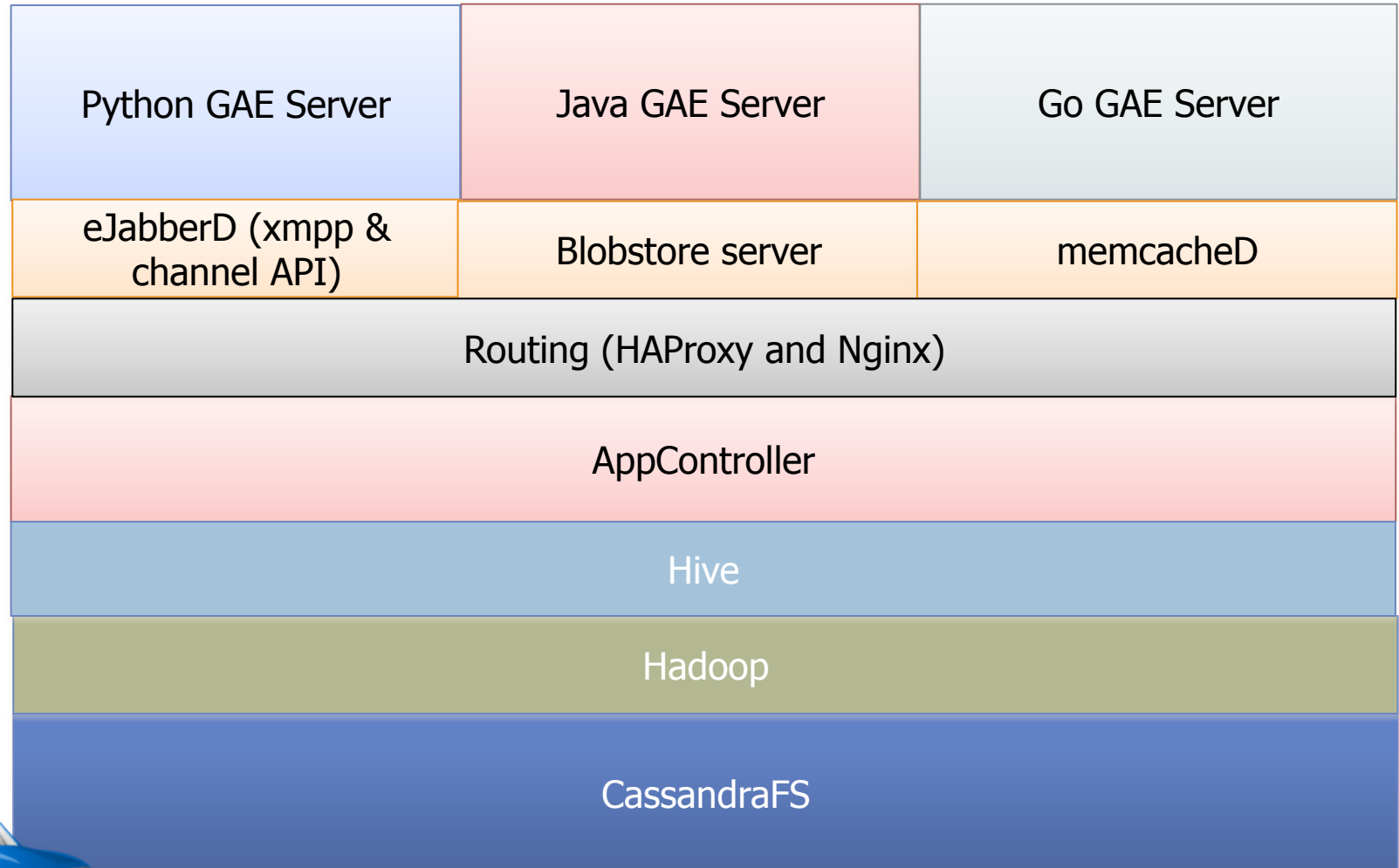 - No transaction support

GAE

AppScale

Service User
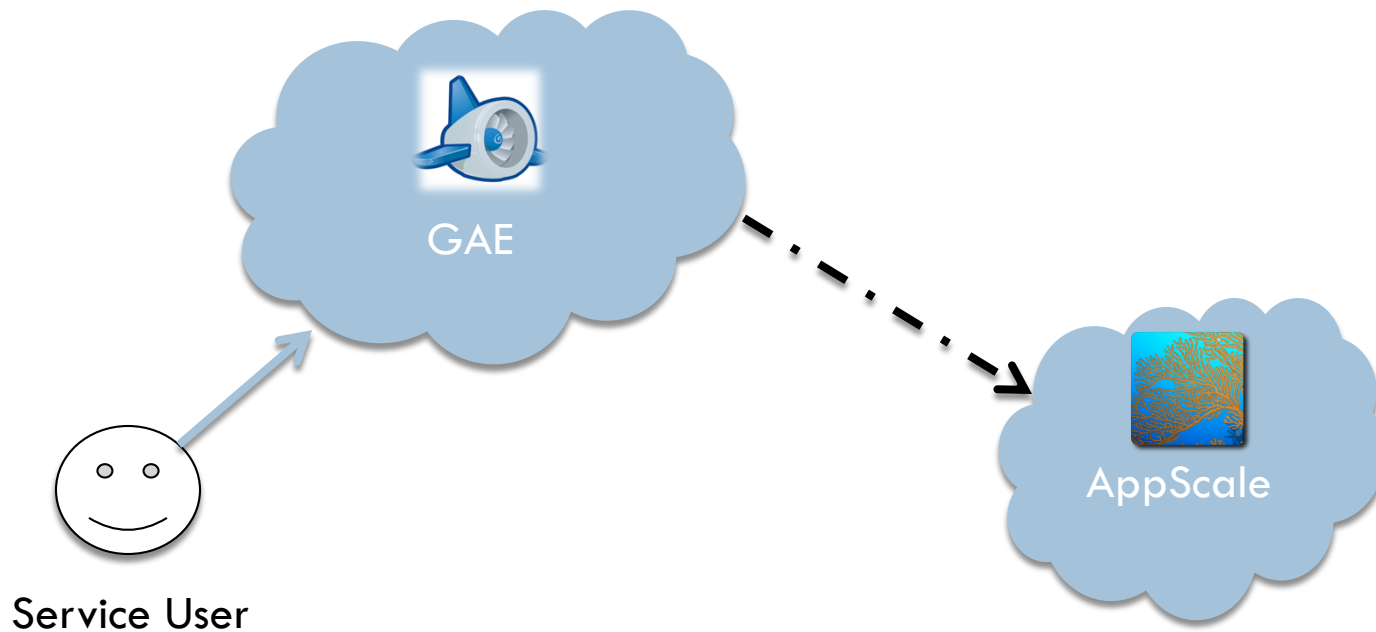
Hive Support in AppScale

- Run SQL statements which translate to Hadoop MR jobs

- We provide the mapping interface from GAE data to Hive queries

GAE

AppScale

Data Scientist

AppScale Stack with Hive Support

| Python GAE Server	Java GAE Server	Go GAE Server
eJabberD (xmpp & channel API)	Blobstore server	memcacheD
Routing (HAProxy and Nginx)		
AppController		
Hive		
Hadoop		
CassandraFS		

Hive Queries

- Simple, short, and fast on-demand queries
- Offline processing of online data
- Does not impact user experience
- Does not introduce bugs or code bloat
- No ETL, data processing in-place
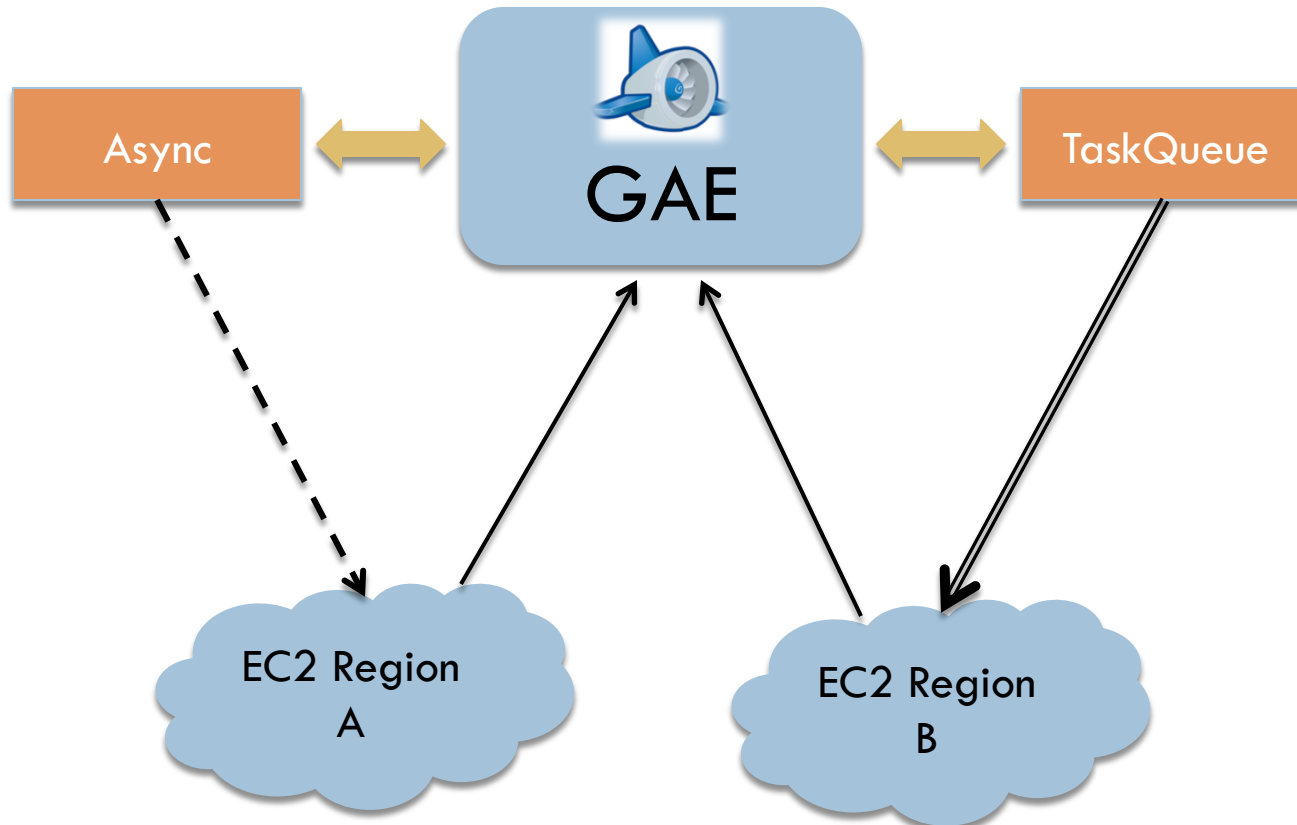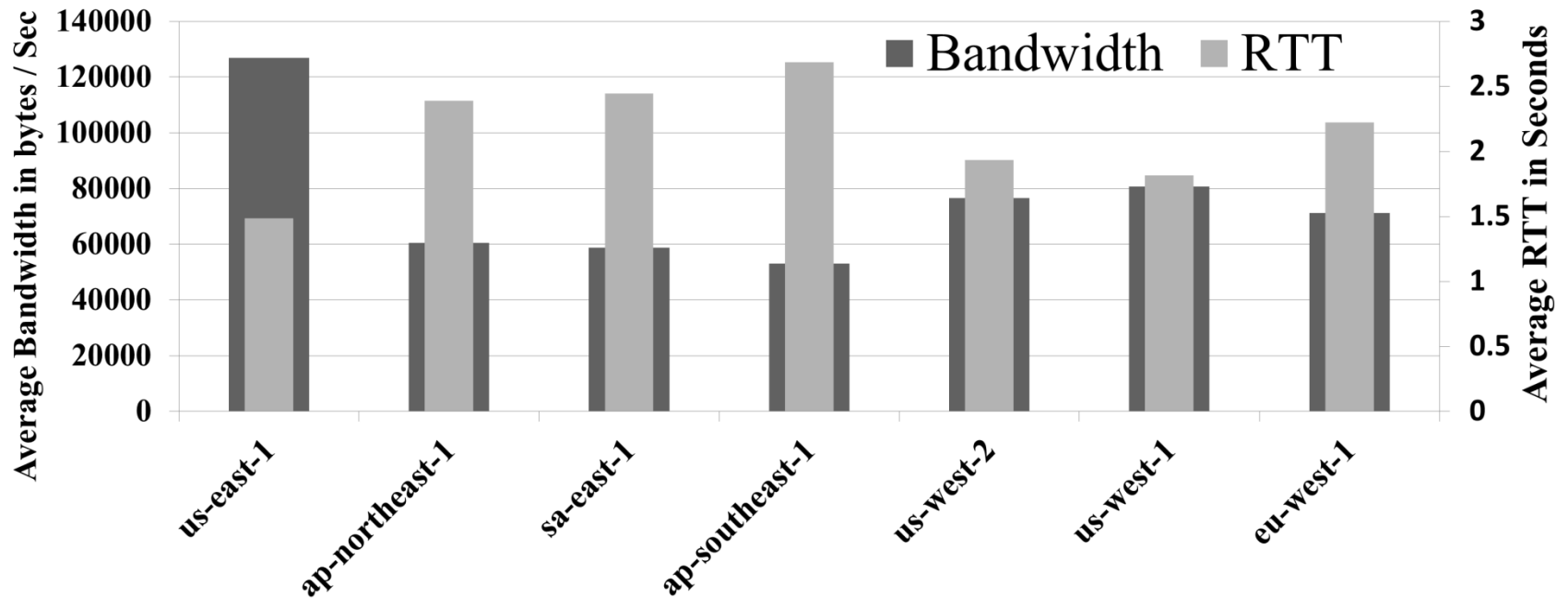
```
SELECT COUNT(*) FROM appid_kind;
```
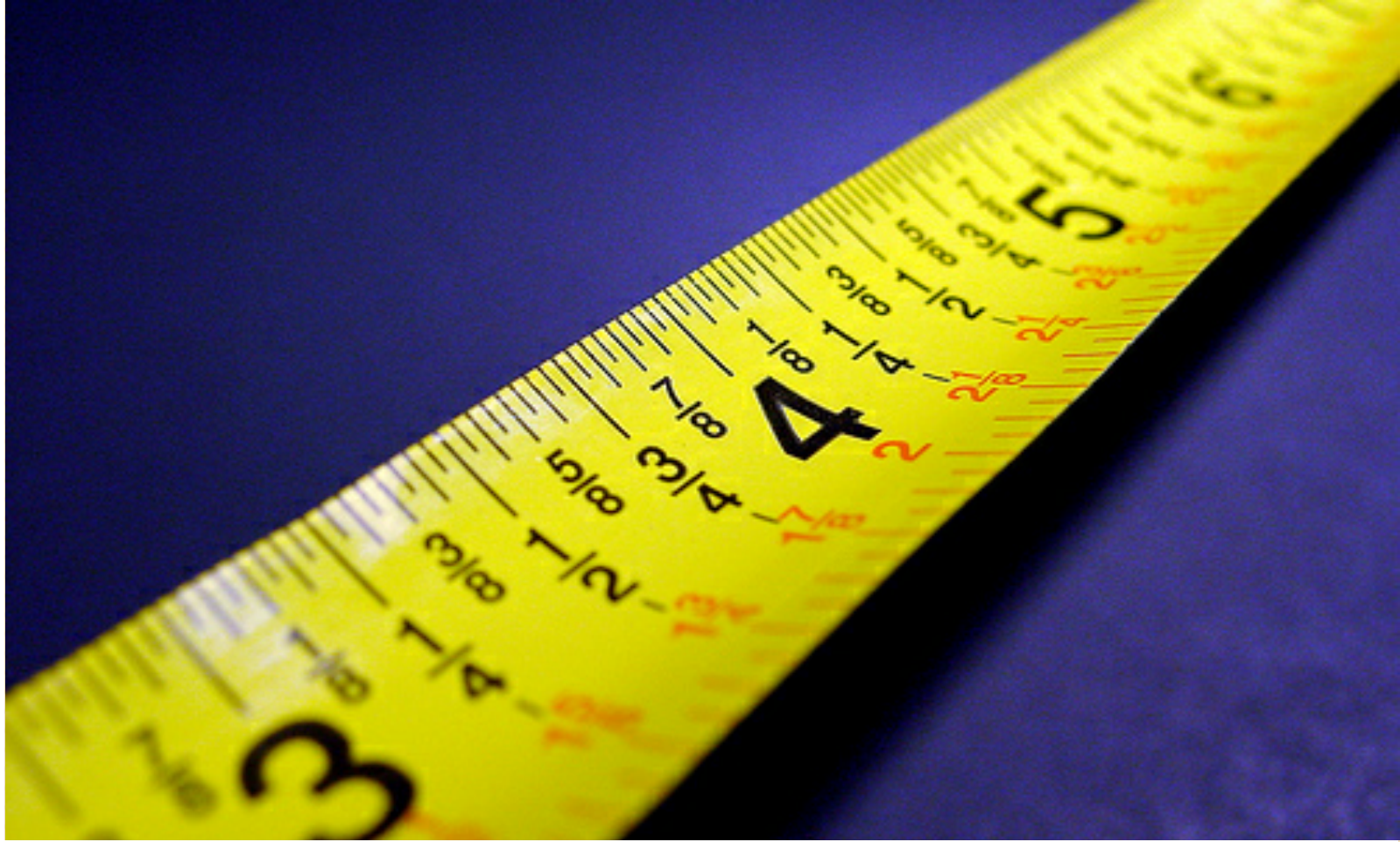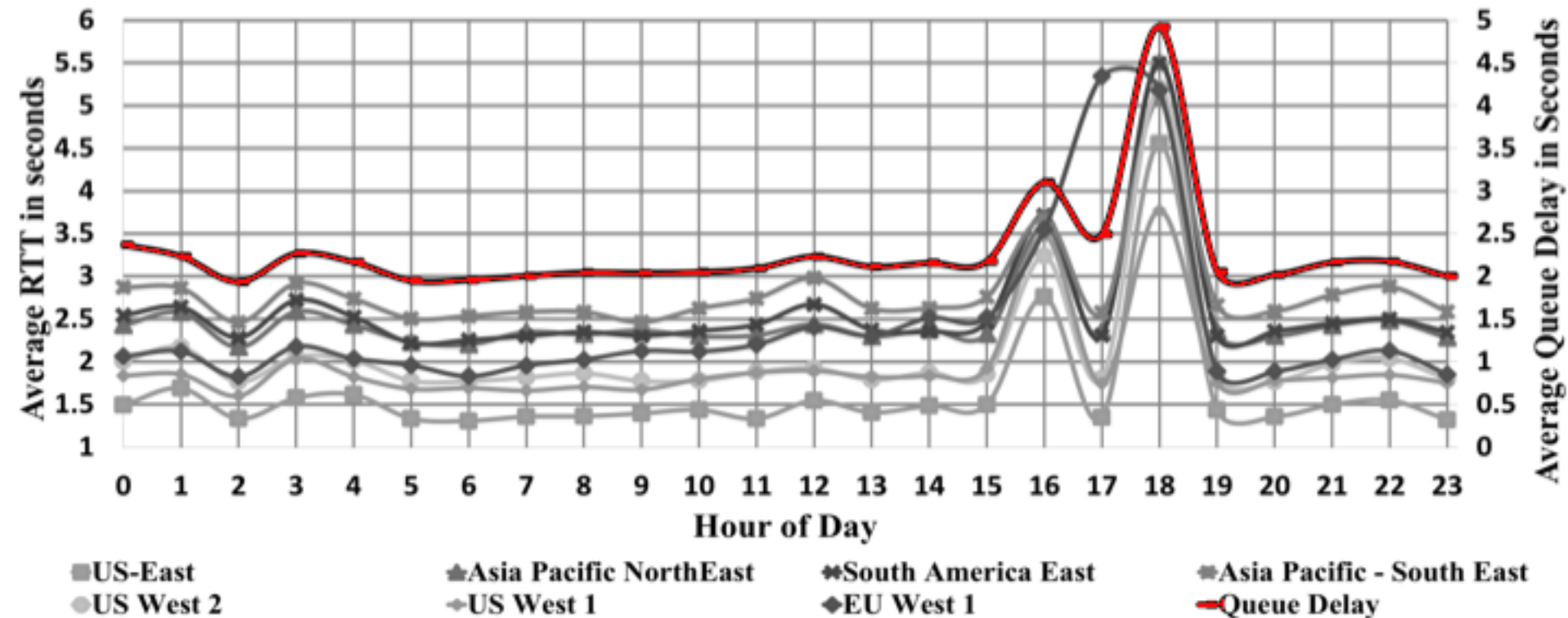
Measurements

Cross Cloud Measurements

Cross Cloud Data Transfer

Round-trip Time and Bandwidth Between a GAE
Application and Different EC2 Regions

Cross Cloud Task Queue Delay

Round-trip time from multiple regions to a deployed GAE application with task queue delay

Analytic Execution Time

	Fantasm (GAE)	Pipeline (GAE)	MapReduce (GAE)	Hive (AppScale)
Aggregate	11334.59	98.34	377.70	20.94
Grep	10360.40	98.89	227.57	10.69
Join	10147.75	159.96	256.40	23.41
Subset	78.28	3.81	237.75	20.66
Wordcount	10977.50	222.14	840.71	21.54

Processing 100K Entities (5 trials)

Cost Analysis

- Data synchronization
  - Bandwidth: $0.12/GB
  - Data backup
- Benefits
  - Programmer productivity
  - Ad-hoc analysis

Related Work

- Private PaaS offerings
  - Red Hat's OpenShift
  - VMWare's CloudFoundry
- GAE Compatible
  - TyphoonAE
- MapReduce
  - Amazon Elastic MapReduce
  - Mesos framework

Thanks

Check out AppScale at:

appscale.cs.ucsb.edu