Chandra Krintz

Computer Science Dept.

Univ. of California, Santa Barbara

KDT Mind Meld

Mar. 5, 2012

UCSB

AppScale

- A cloud platform
  - Distributed system that provides a complete runtime stack
    - Upload (Python, Java, Go, …) programs and web services
  - Providing scalable program-level abstractions via well-defined interfaces
    - Different storage options, user management, tasking, messaging,…
    - Automates configuration, distributed deployment
- Executes over
  - Virtualized cluster resources (requires manual VM instantiation)
  - Infrastructure-as-a-Service (automatically)
    - Amazon EC2, Eucalyptus, any IaaS with euca2ools support
  - Isolation at either/both the VM and process level

What AppScale Is/Does and Why

- Open source
  - Facilitate research into the next generation of
    - Cloud runtime systems, services, applications, technologies
  - Engender a community of users
  - Leverage and integrate wide variety of popular open source technologies
    - HTTP proxies, applications servers, load balancers, databases, multimedia, communications, distributed locking, messaging, programming models (mapreduce)…
    - Emerging research: StochKit, KDT, …

What AppScale Is/Does and Why

- Open source
- Automate configuration/deployment of distributed apps
  - Broaden participation in use of cloud systems
  - Increase programmer productivity: cloud/web/distributed apps
    - Reduce the overhead of using popular distributed open source technologies

What AppScale Is/Does and Why

- Open source
- Automate configuration/deployment of distributed apps
- Portability for cloud applications
  - Provide a universal set of APIs
  - Over different cloud fabrics
    - Without application modification
    - "Write Once, Run Anywhere (WORA)" for the cloud
  - Eliminate *lock-in* to any single public cloud vendor
    - Encourage cost competition
  - Investigate and facilitate hybrid cloud use

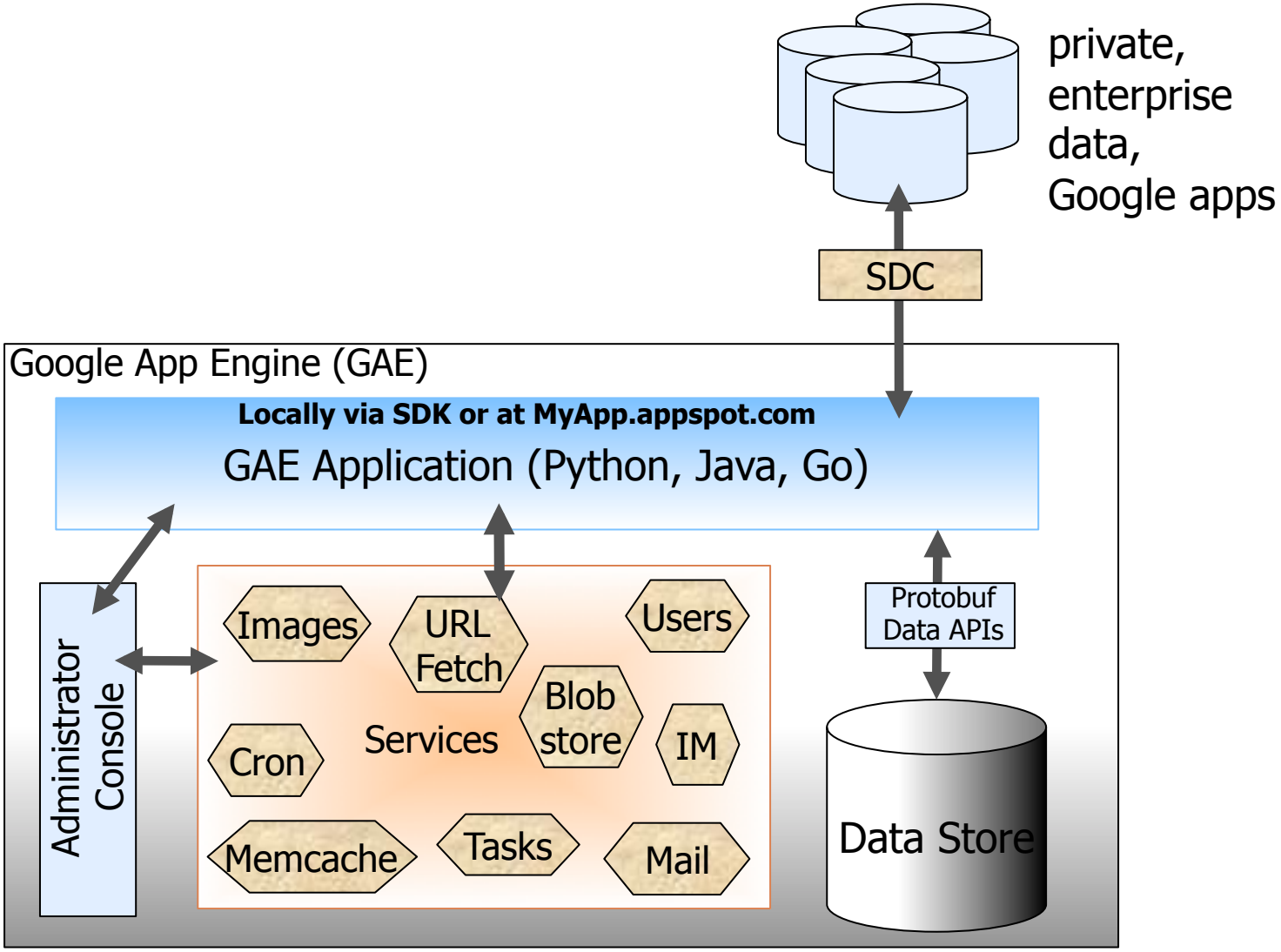
What AppScale Is/Does and Why

- Open source
- Automate configuration/deployment of distributed apps
- Portability for cloud applications
- For a wide range of applications and application domains
  - Using different programming languages & programming models
  - Multiple domains:
    - Web service based
    - Computationally intensive (HPC)
    - Data analytics

What AppScale Is/Does and Why

- Open source
- Automate configuration/deployment of distributed apps
- Portability for cloud applications
- For a wide range of applications and application domains
- Mirror public cloud technology "standard"
  - Scalable, efficient, fault resilient, and elastic
  - API compatiblity with "standard"
    - Google App Engine (GAE)
    - Engender user community from an existing one
    - Real applications written by others to evaluate
  - Portable API for popular cloud services
    - Data access/storage, mail/messaging, task execution, monitoring
    - VM management (coordination with IaaS layer)

Google App Engine (GAE)



GAE: Upload to Google Public Cloud

private, enterprise data, Google apps

SDC

GAE app users via the Internet

Google App Engine (GAE)

appcfg.py update MyApp/ → MyApp.appspot.com

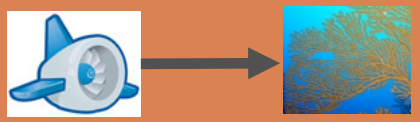
GAE Application (Python, Java, Go)

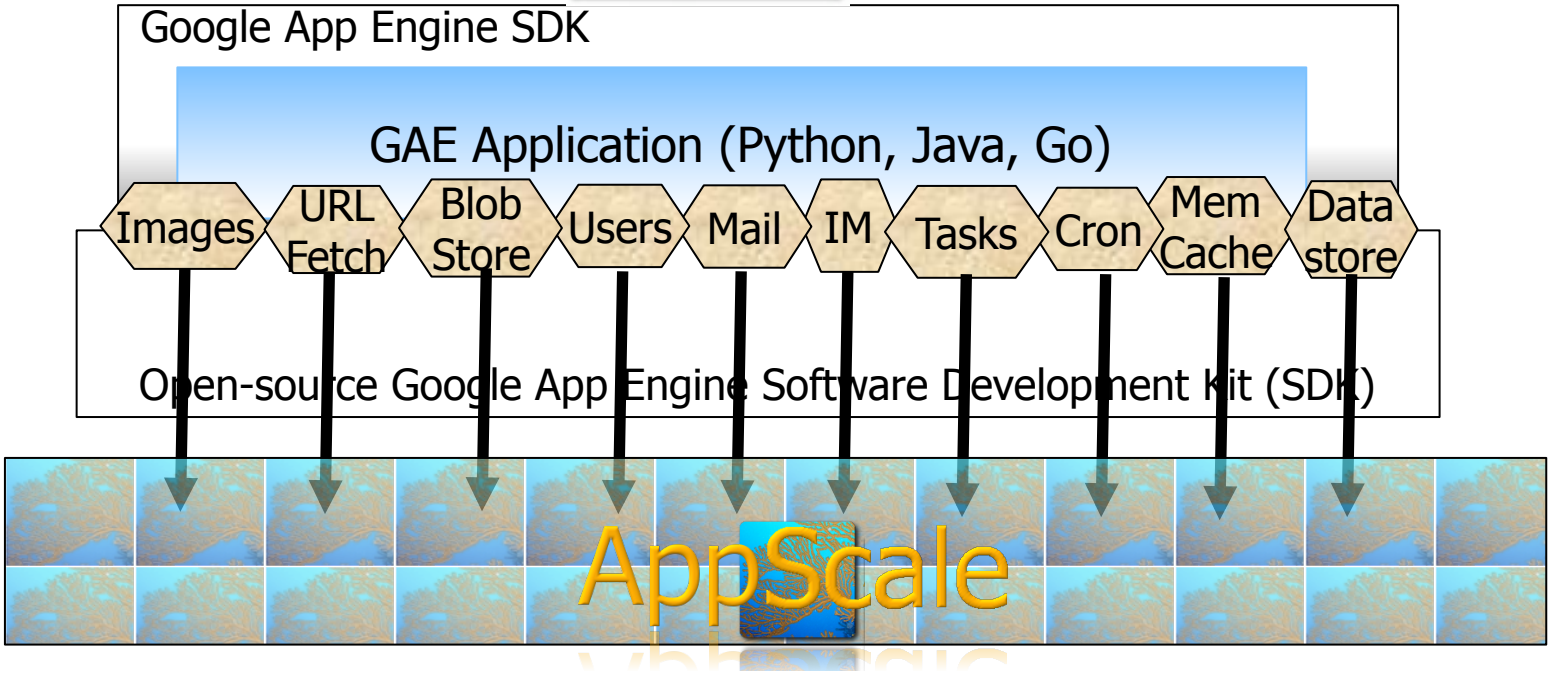
Images | URL Fetch | Blob Store | Users | Mail | IM | Tasks | Cron | Mem Cache | Data store

Administrator Console

Free w/ quotas
Pay for additional scale:
CPU, BW, emails, data
BigTable/MegaStore (HRD):
Automatic scaling
High availability

From GAE to AppScale





Your local cluster resources (virtualized and/or managed via Eucalyptus) or Amazon EC2

APIs

	GAE	**AppScale**
Datastore	BigTable/Megastore	Cassandra HBase Hypertable MemcacheDB MongoDB MySQL Cluster Redis Voldemort Amazon SimpleDB

APIs

	GAE	**AppScale**
Datastore	BigTable/Megastore	Open Source Alternatives
Blobstore	Proprietary	Tornado + Datastore
Memcache	Proprietary Memcache	memcached
XMPP	Google Talk	ejabberd
Channel API	Google Talk	ejabberd and Strophe.js
Images	Picassa	Google SDK
URL Fetch	Proprietary	Google SDK
Task Queues, Cron	Proprietary	RabbitMQ
Mail	Google Mail	Sendmail
MapReduce	Task Queues, Pipeline	Task Queues, Hadoop
VMInstance Control	--	Euca2ools
Cloud configuration	--	**Neptune** ♆
HPC and data analytics	--	**Neptune** enabling MPI, UPC, Erlang, X10, StochKit, KDT,…

Neptune

- Domain-specific programming language and runtime
  - Extensions to the Ruby programming language
- Automating deployment of non-GAE (non-web-service) application code
- Facilitates deployment of apps from other app domains
  - HPC Data analysis Graph processing
  - Written in any programming language
    - Python, Java, Go, Ruby, C/C++, Erlang, …
  - Supported toolkits (Neptune *types*)
    - MPI, UPC, X10, StochKit, KDT
    - MapReduce (Hadoop, Hive)
- Runtime integrates with and leverages AppScale

Using Neptune

- Deploy an AppScale cloud; place code
- On any machine with Ruby and Neptune (gem) write/run job specification
  - Storage (code and data) can be local filesystem, AppScale datastore, Google Bigtable, Amazon S3, Walrus (Eucalyptus), …

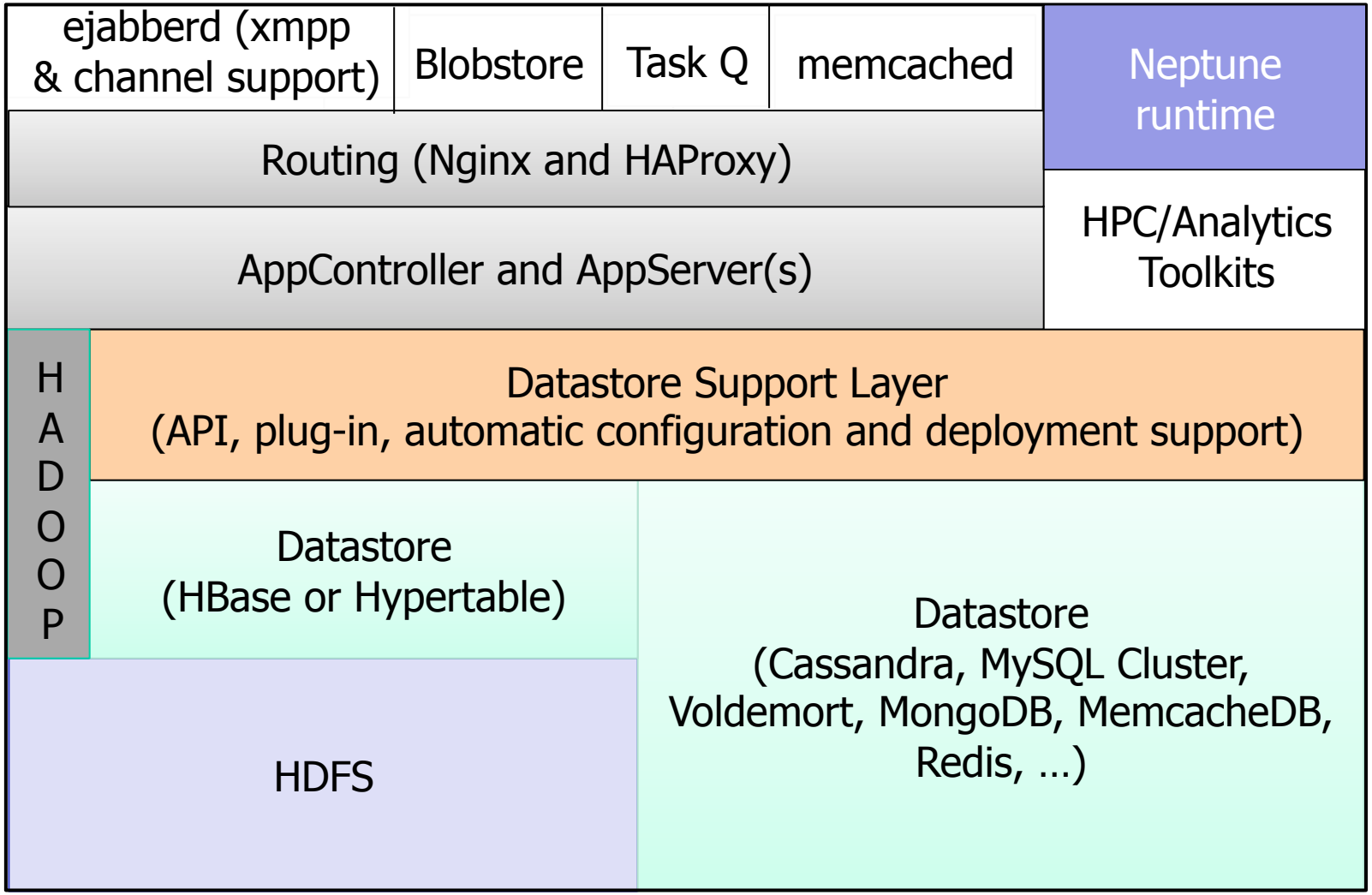
```
neptune :type => :mpi,
        :code => "/code/ring",
        :nodes_to_use => 32,
        :output => "/output/ring"
```

Using Neptune

- Deploy an AppScale cloud; place code
- On any machine with Ruby and Neptune (gem) write/run job specification
  - Storage (code and data) can be local filesystem, AppScale datastore, Google Bigtable, Amazon S3, Walrus (Eucalyptus), …

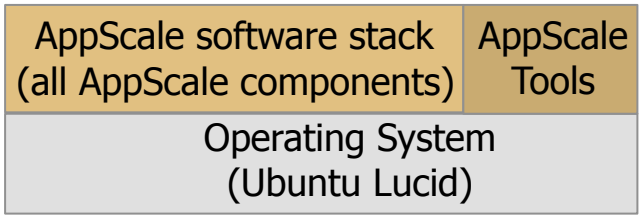
```
neptune :type => :kdt,

 :code => "/code/Graph500.py",

 :nodes_to_use => 16,

 :output => "/output/graph500"
```

AppScale Software Stack



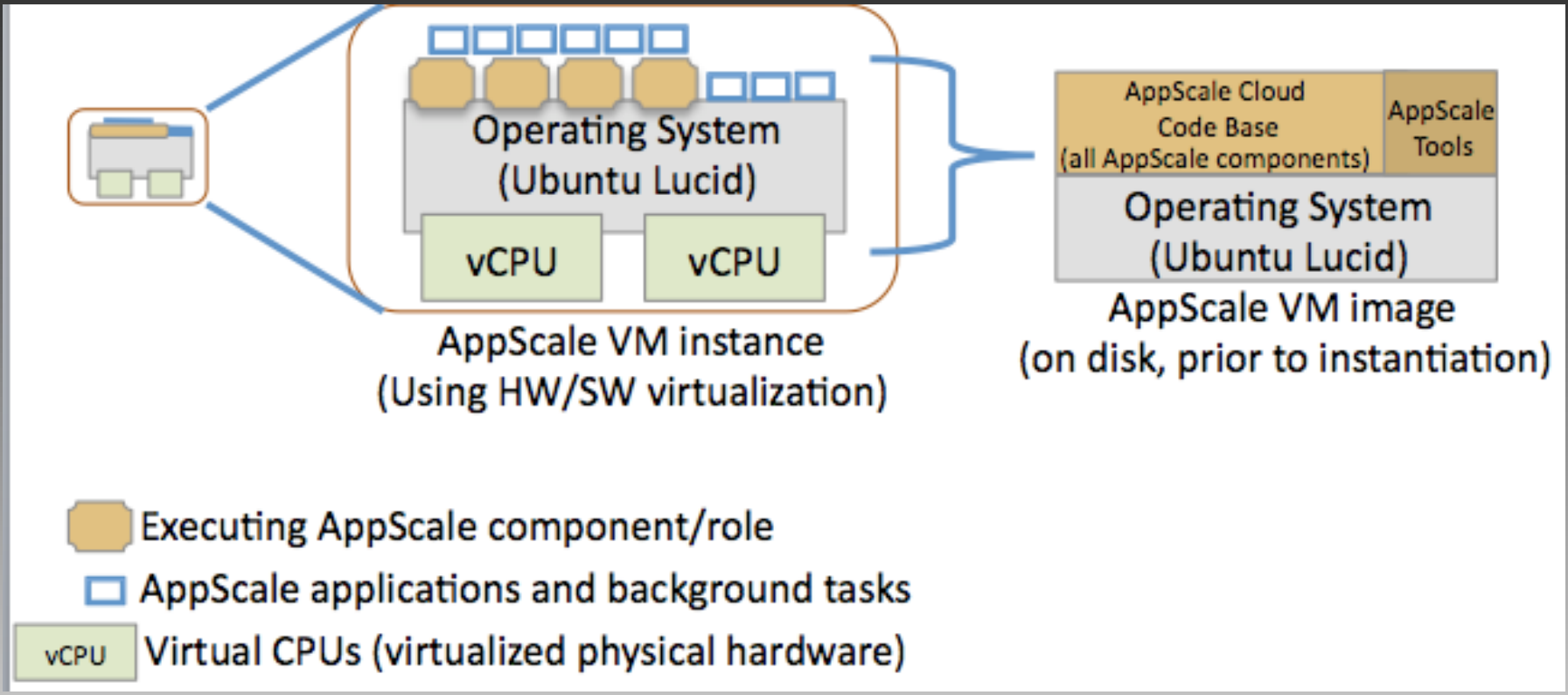
AppScale Deployment

- Available as
  - A virtual machine image
    - Eucalyptus compatible: precluding our need to support all Linux distributions and virtualization layers
  - An Amazon EC2 AMI
  - Open source (automated AppScale image installation)
- Tool set for command-line cloud deployment
  - Inspired by AWS tools for instance management
    - Run, describe, terminate instances
  - Automates deployment & configuration of all components



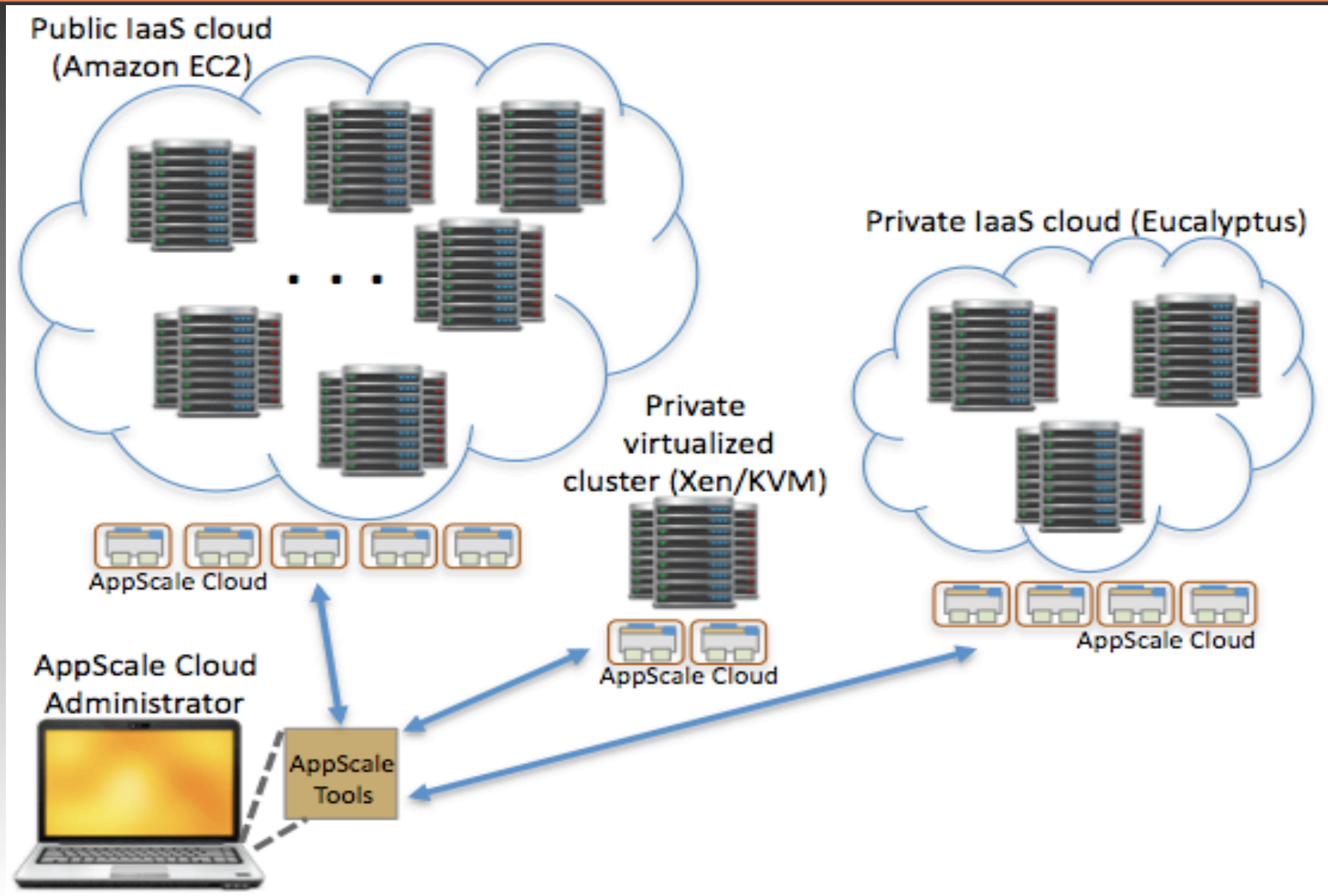
AppScale VM image
(on disk, prior to instantiation)

AppScale as a Deployed VM Instance



- AppScale roles (specify via a YAML config file, or use defaults)
  - AppController, AppLoadBalancer, AppServer, AppDB
  - Cloud status monitor, ZooKeeper, Neptune runtime/agent, TaskQ master/agent, logging master/agent
  - Custom, Open (for fast elasticity)

AppScale Cloud Deployment



Summary

- AppScale is an open source distributed execution platform for a wide range of applications
  - Automatically deploys complex distributed technologies
  - Investigate different domains, programming systems, cloud technologies and services
- API-compatibility with GAE
  - GAE apps run over AppScale without modification
- IaaS-portable API/deployment for HPC and data analytics
- Hybrid cloud support
  - Automatic use of different public clouds and their services
- Many research directions to investigate
  - Runtime systems, OS/virtualization, dist computing, optimization
  - Cloud (cost models/estimation, scaling/elasticity, hybrids)

Thanks!

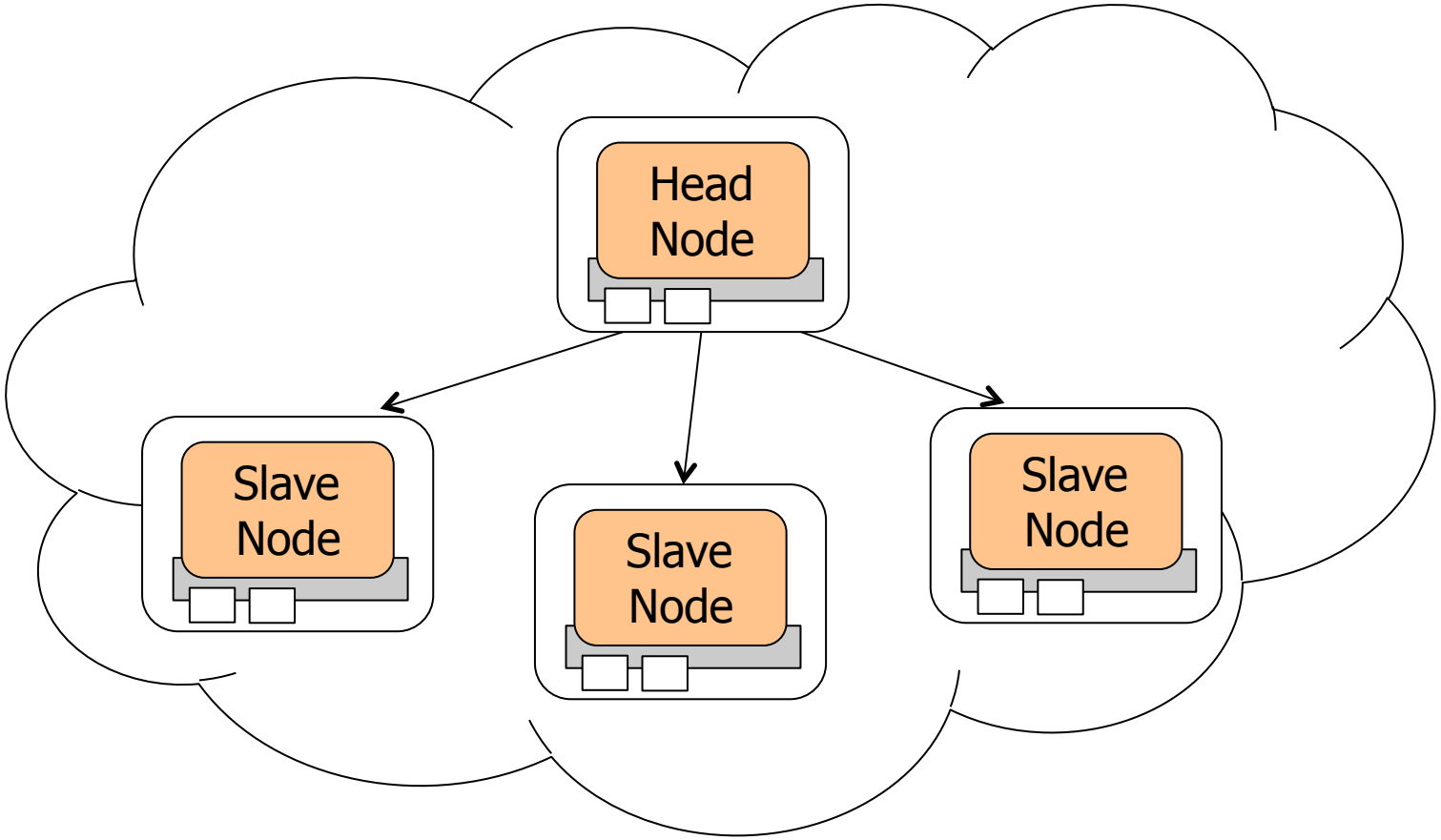
- Students and Visitors!
  - Vaibhav Arora, Maciej Baranski, **Chris Bunch**, Navyasri Canumalla, Jovan Chohan, **Navraj Chohan**, Nupur Garg, Anand Gupta, Shashank Hedge, Matt Hubert, Jonathan Kupferman, Puneet Lakhina, Yiming Li, Nagy Mostafa, Yoshihide Nomura (Fujitsu), Kowshik Prakasam, Raviprakash Ramanujam, Andres Riofrio, Sujay Sundaram, Bing Wei, Michal Weigel
- Support
  - Google, IBM, NSF, NIH

http://www.cs.ucsb.edu/~ckrintz

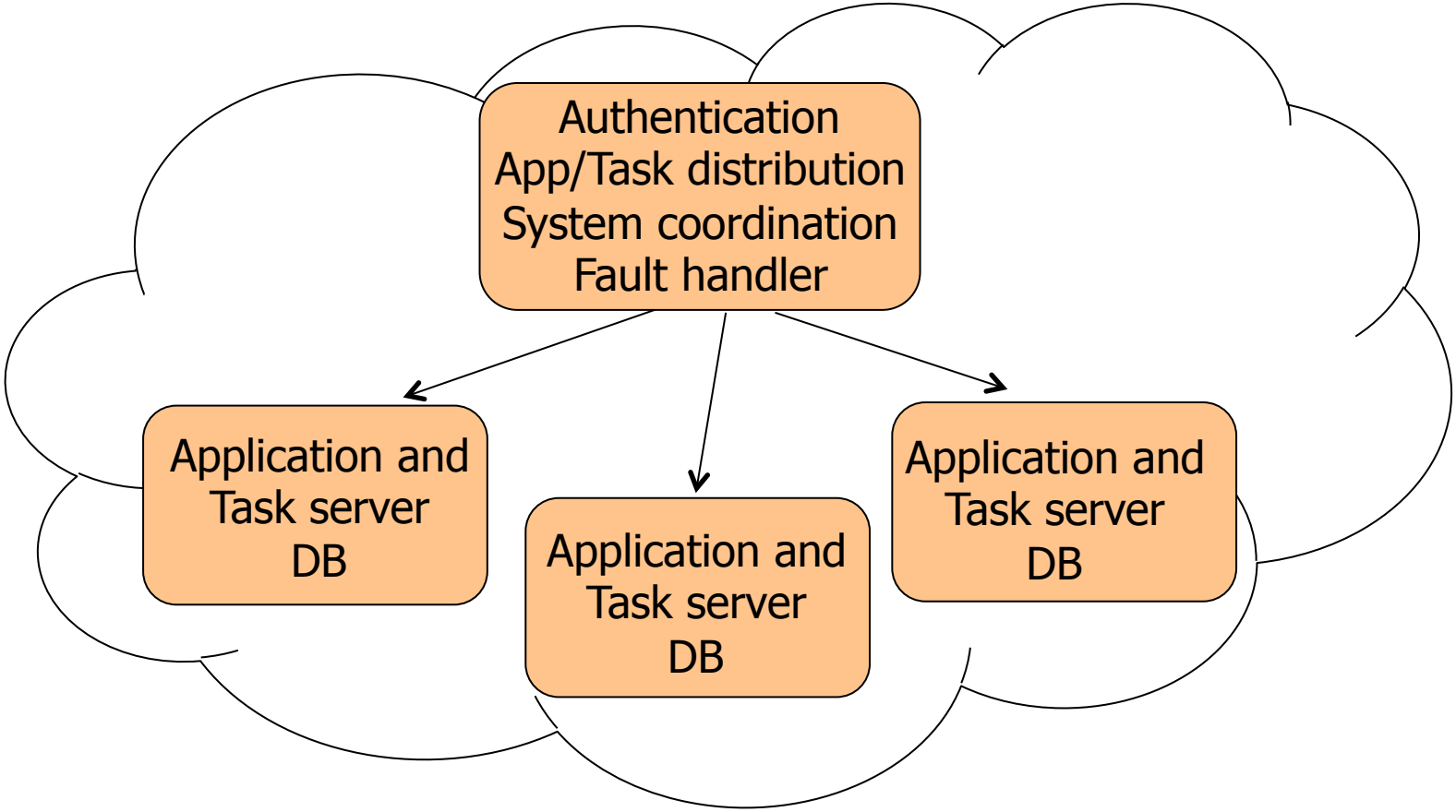
http://appscale.cs.ucsb.edu/

Backup Slides

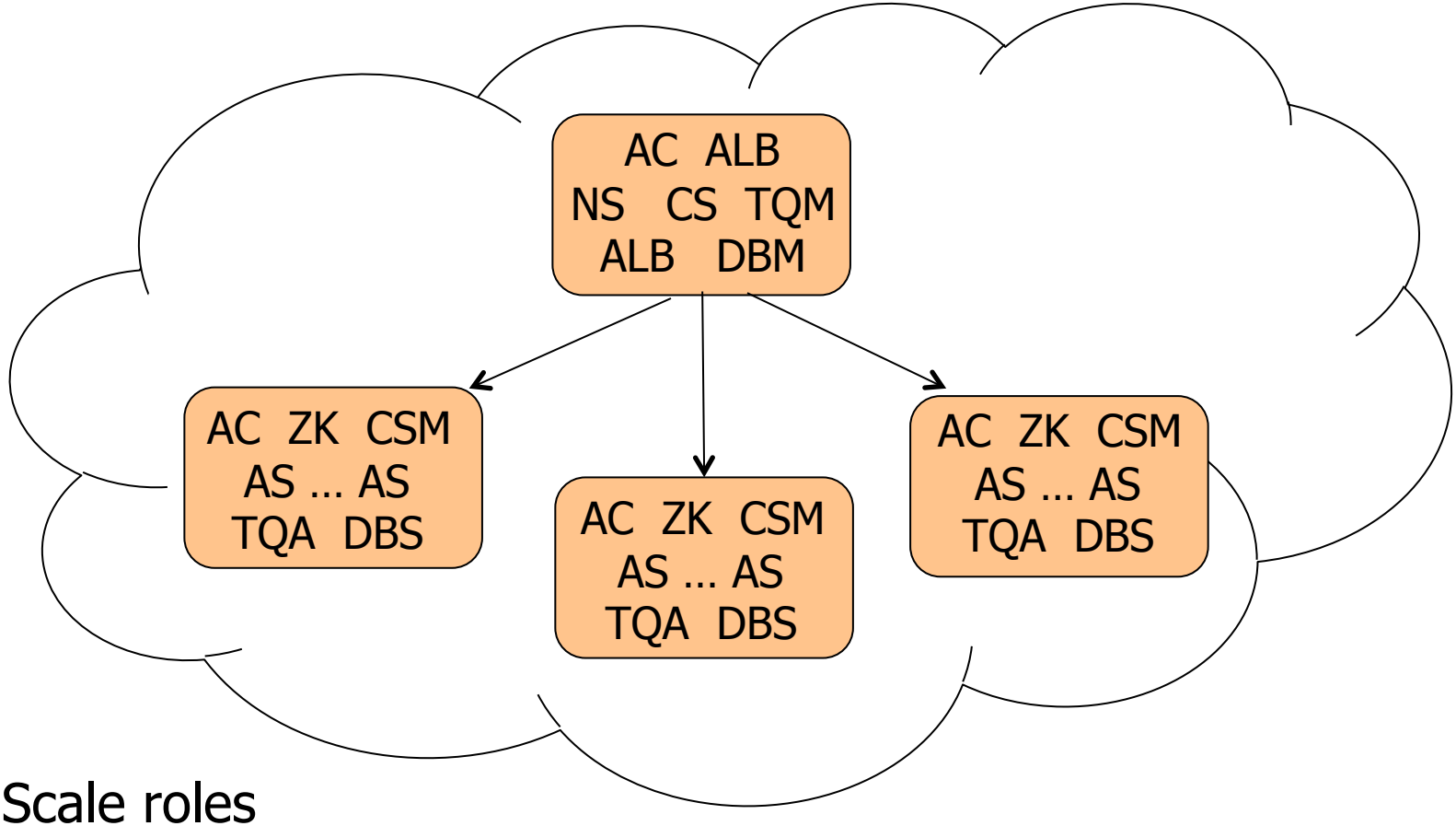
AppScale Cloud Default Deployment



AppScale Cloud Default Deployment

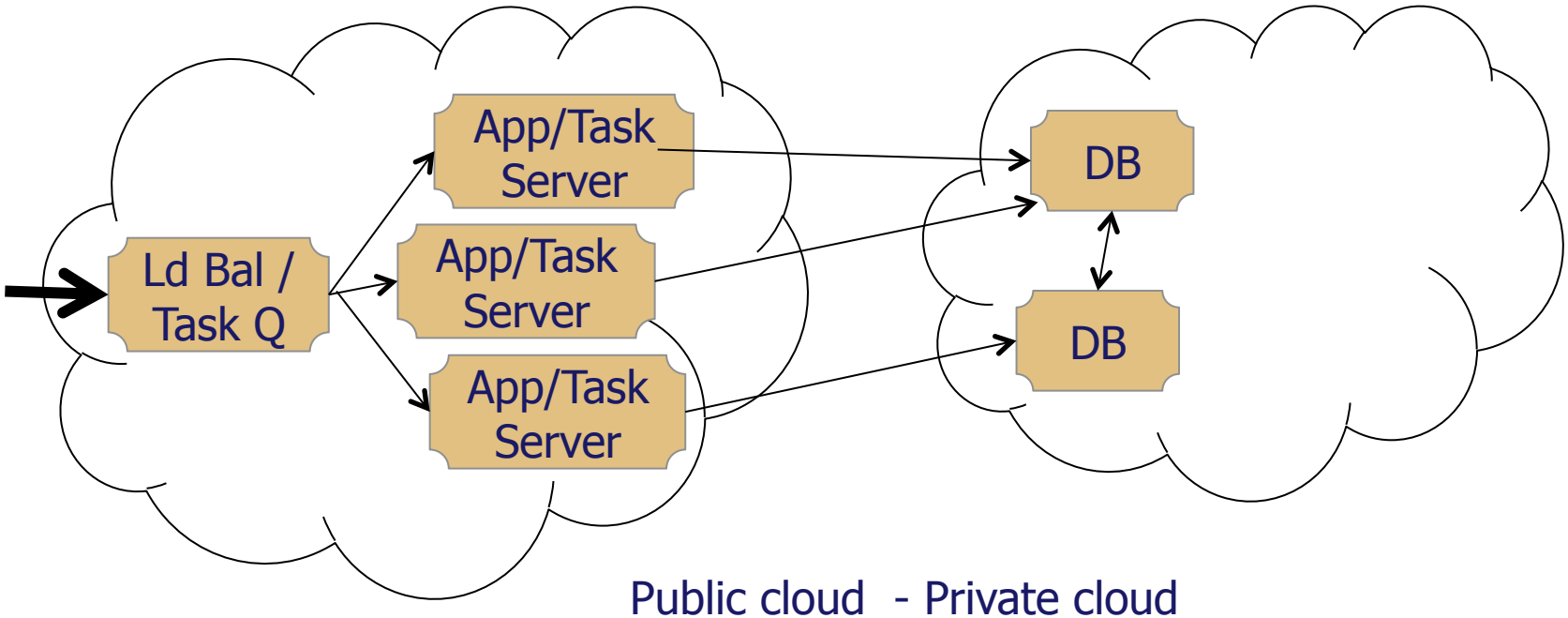


AppScale Cloud Default Deployment



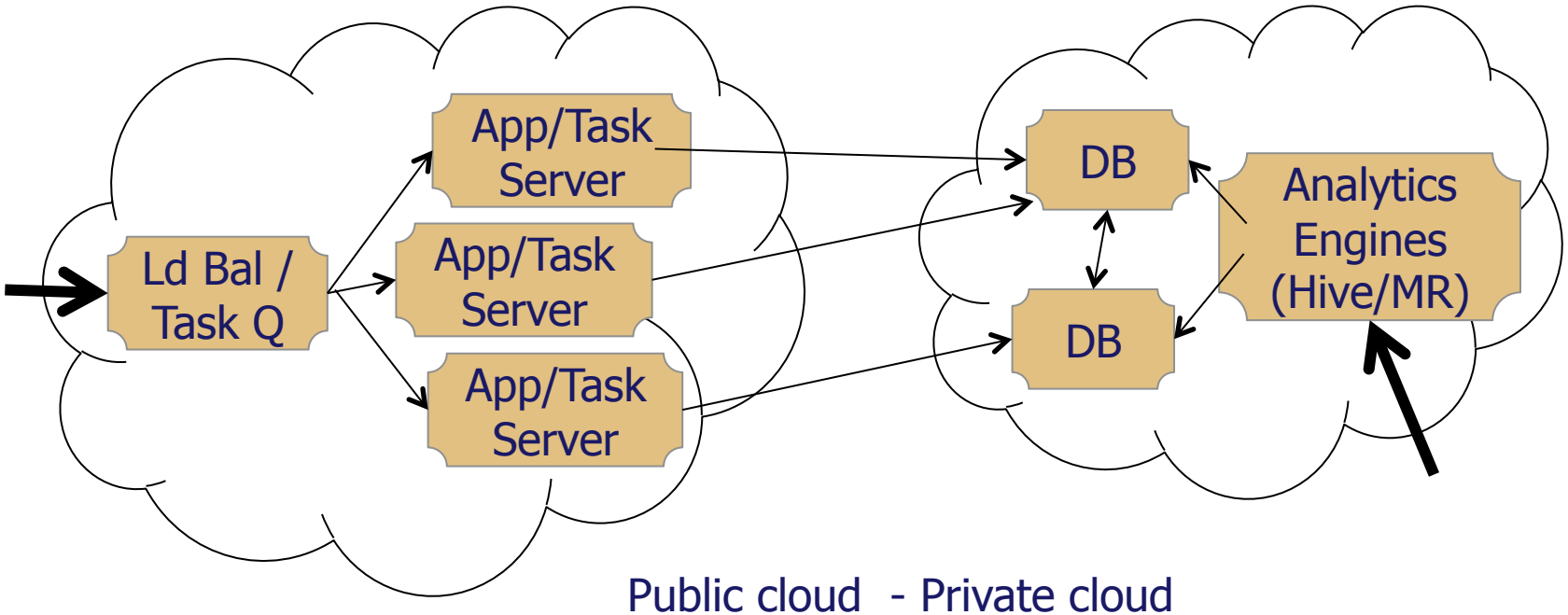
- AppScale roles
  - AppController (AC), AppLoadBalancer (ALB), AppServer (AS), AppDB (DBM,DBS)
  - Cloud status/monitors (CS), ZooKeeper (ZK), Neptune server (NS), TaskQ master/agent (TQM,TQA)
  - Custom, Open

AppScale Hybrid Clouds



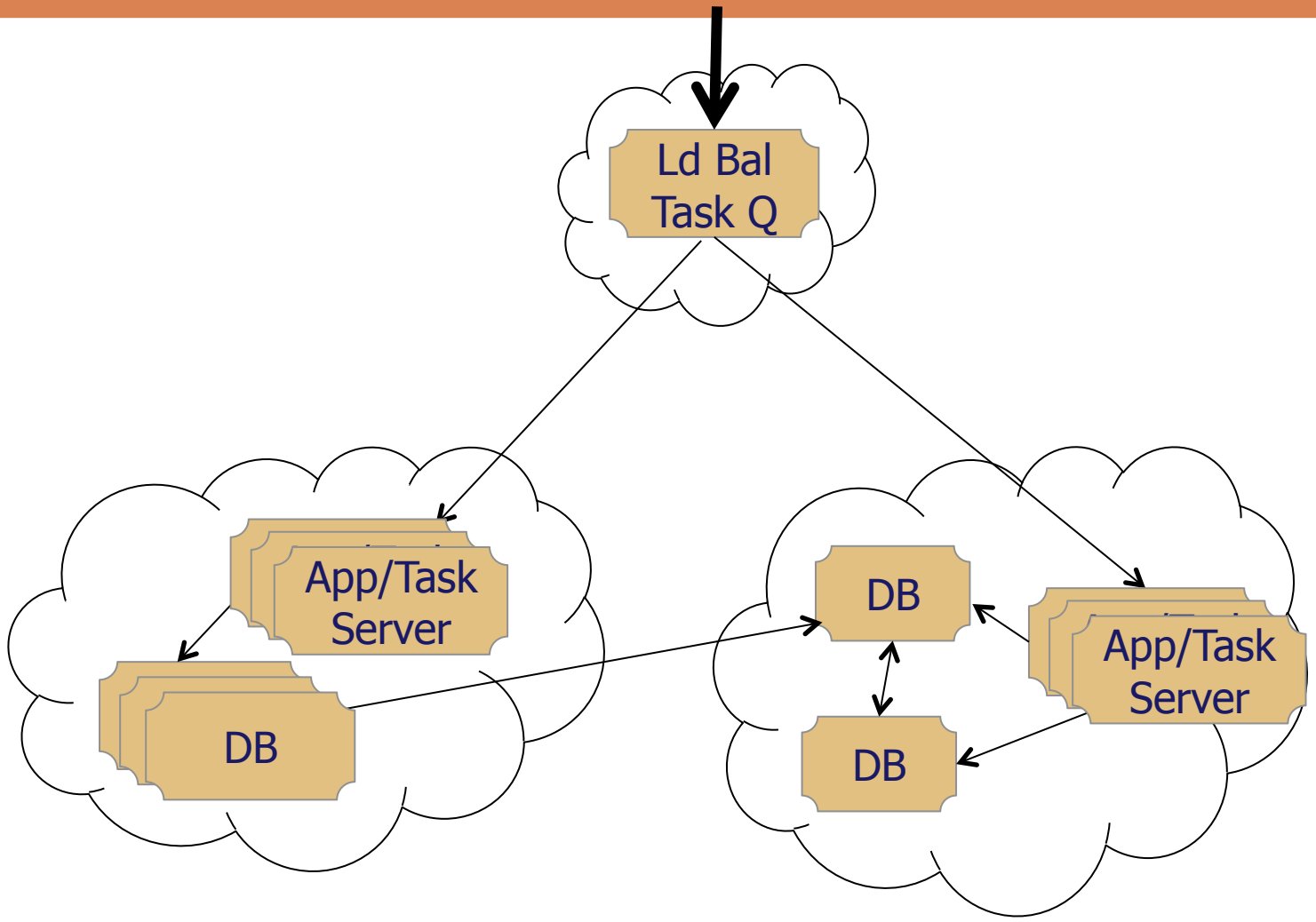
Public cloud - Private cloud
Private cloud - Public cloud
Private cloud K - Private cloud J
Public cloud (zone A) - Public cloud (zone B)
Public cloud vendor X - Public cloud vendor Y

AppScale Hybrid Clouds

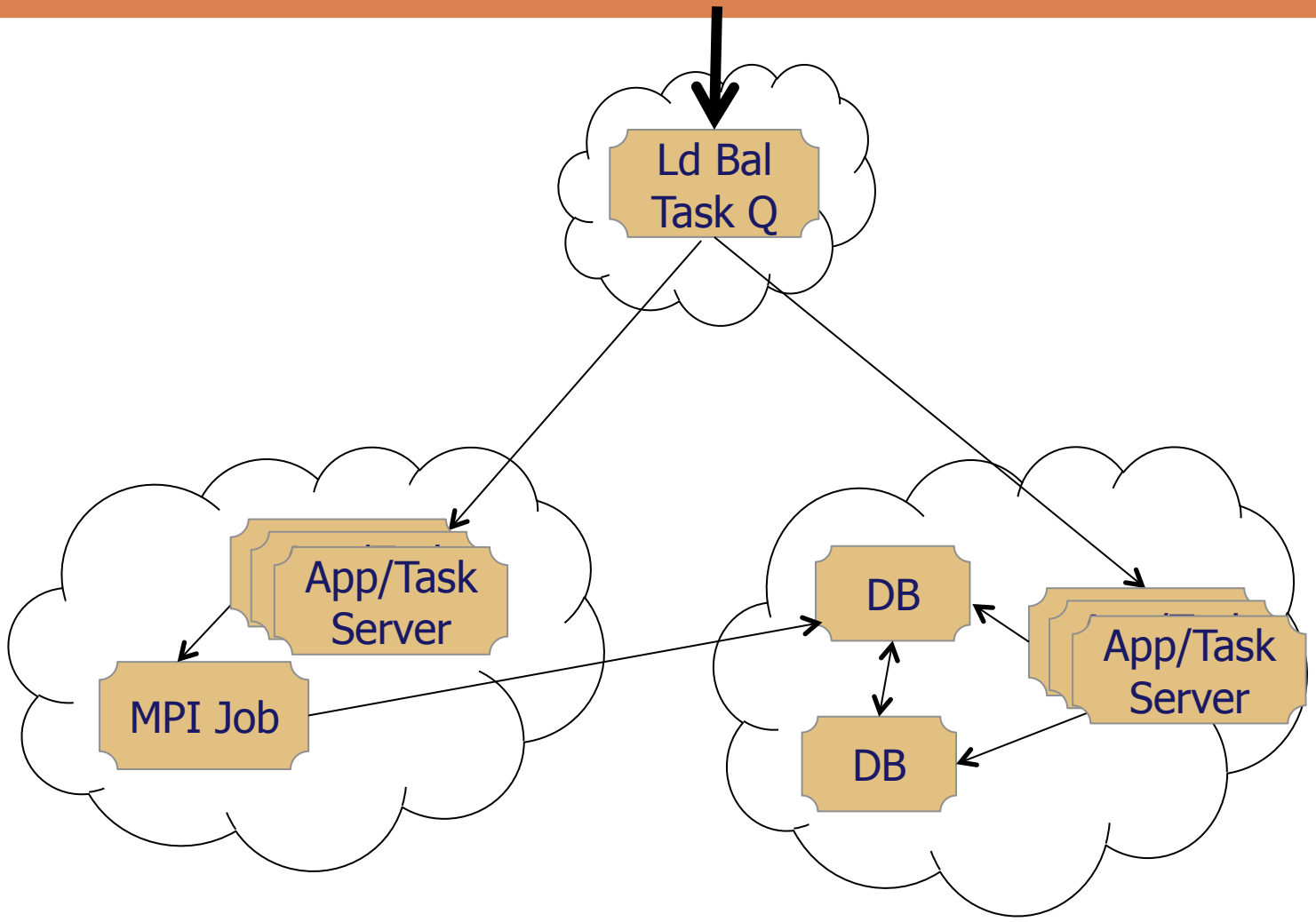


Public cloud - Private cloud
Private cloud - Public cloud
Private cloud K - Private cloud J
Public cloud (zone A) - Public cloud (zone B)
Public cloud vendor X - Public cloud vendor Y

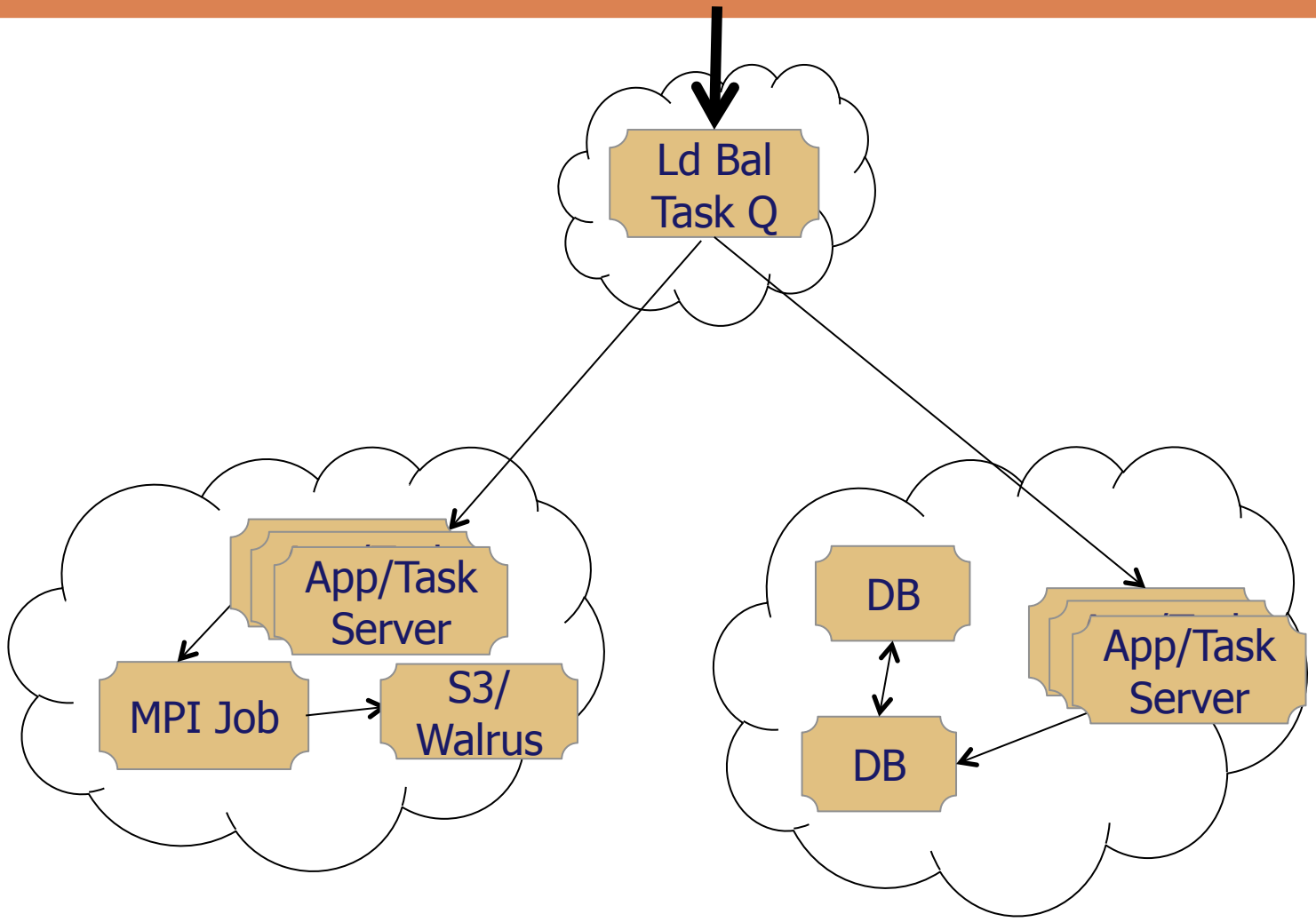
AppScale Hybrid Clouds



AppScale Hybrid Clouds



AppScale Hybrid Clouds



AppScale Interoperability & Hybrid Support

- GAE applications: over AppScale or GAE
- Datastore support via database support layer
  - AppScale native (plugins), AWS SimpleDB, AWS S3, GAE Bigtable
- Task Q support
  - AppScale native (rabbitMQ), AWS SQS, Microsoft Azure, GAE
- Task agent support
  - Web service and background (non-GAE) process — on-premise
  - Python, Ruby, Java, Go, R, C/C++ — & over AWS
  - Python, Java, Go — over GAE
- High-performance toolkits (not GAE compatible)
  - MPI, UPC, X10, StochKit, KDT
  - MapReduce (Hadoop), Hive support (offline analytics)