Grid Dynamics
Experts in *Extreme* Applications

Project Convergence:

Integrating Data Grids and Compute Grids

Eugene Steinberg, CTO
Grid Dynamics

May, 2008

Data-Driven Scalability Challenges in HPC

- Data is far away
    - Latency of remote connection
    - Latency of data movement through pipes
    - Chatty algorithms are expensive
- Data is centralized
    - HW Resources are limited
    - Inevitable disk I/O due to limited RAM
    - Connections are limited
    - Highly concurrent access doesn't scale well

Usual Solution: Compute Grid + Data Grid

- Classic Data Grid
  - Data is partitioned
  - Partitions are stored in memory of data grid
  - Data grid is deployed near to compute grid
  - Search is parallelized over partitions
  - Build-in replication, persistence, coherence, failover
- What is Achieved?
  - Reduced latency and data moving cost
  - Improved connection scalability
  - Reduced data contention
  - No Disk I/O – 100% memory speed

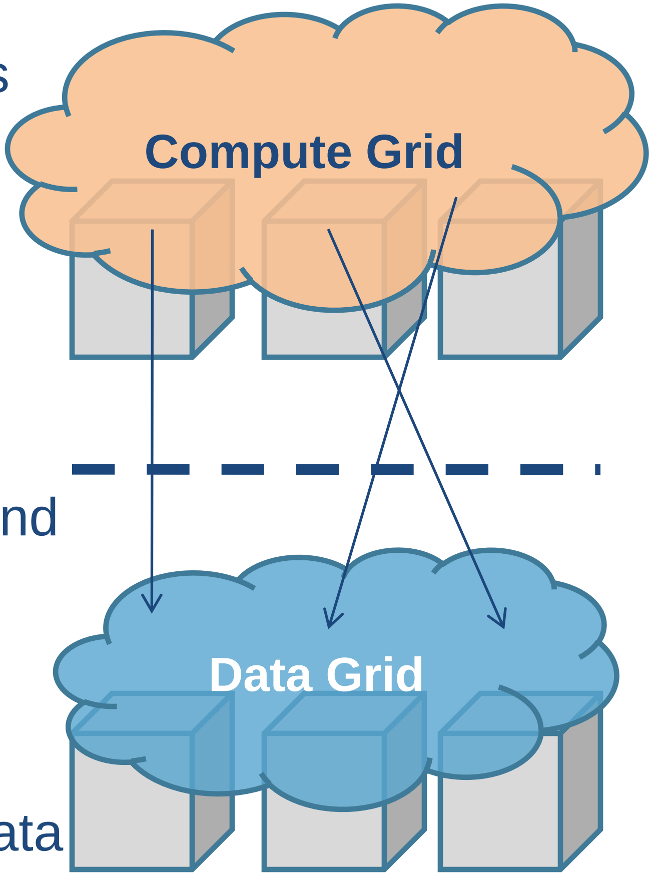
Is this the best we can do?

Limitations of Compute Grid + Data Grid

- Two separate grid environments
  - Hardware, footprint and management costs of dual infrastructure
  - Segregated infrastructures cannot share resources

- Sub-optimal resource utilization
  - Compute grid is CPU-bound, not RAM-bound
  - Data grid is RAM-bound, not CPU-bound

- Still sub-optimal performance
  - Still paying for remote network calls and data movement



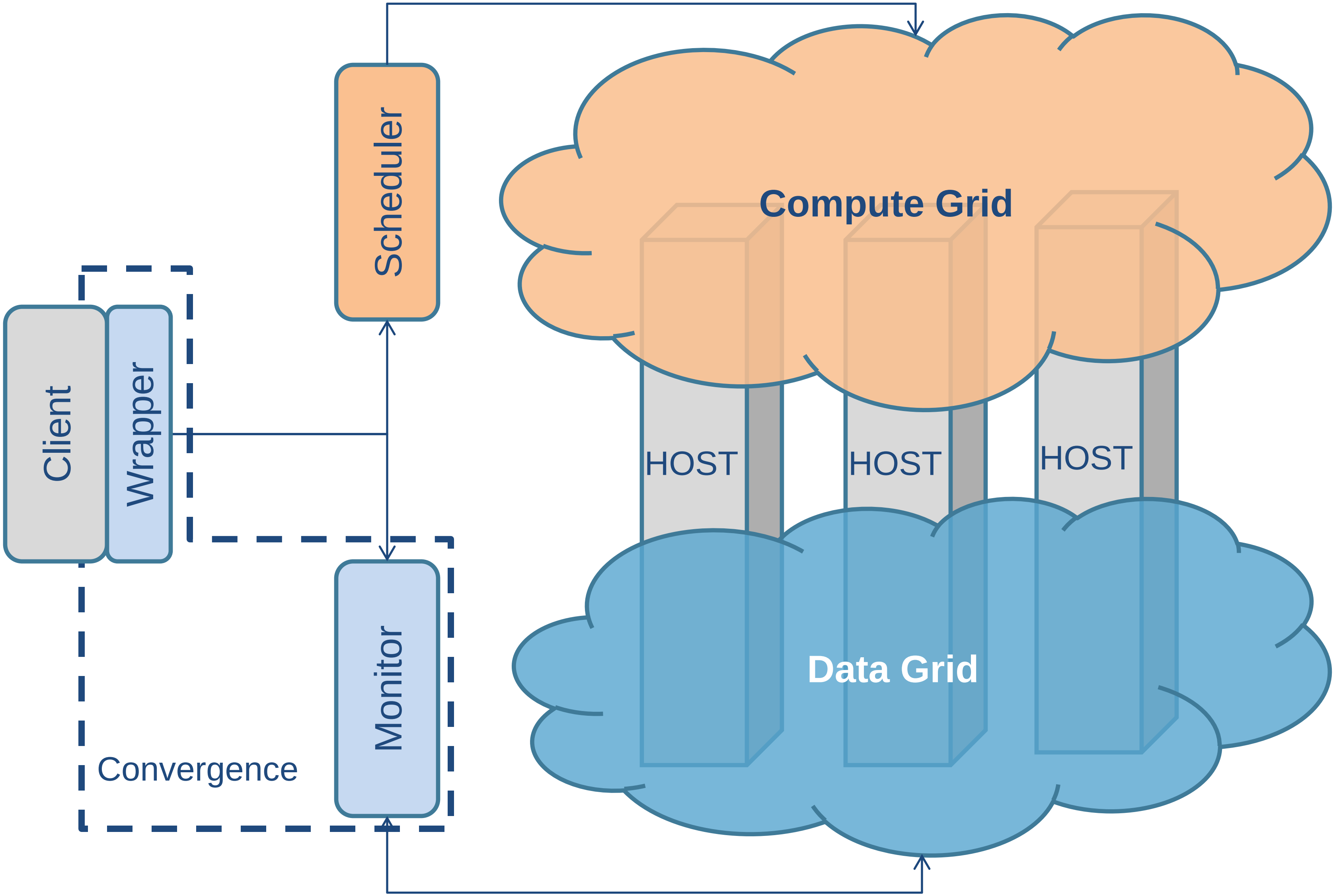
Better Answer: “Compute-Data Grid”

- Shared hardware between compute & data grid
  - Data grid resides in RAM of host machines
  - Compute grid runs HPC jobs on the same host machines
- Opportunity to collocate processing with data
  - Many applications support compute-data affinity
  - No network overhead on remote calls and data movement
- New recipe for scalability
  - As HPC application needs to scale in and out, data partitions are spread over larger or smaller pool of hosts

Project Convergence

- Open source reference architecture for Compute-Data Grid
- Goals
  - Pluggable architecture to support adapters for many grid products
  - Non-intrusive compute-data grid coordination
  - Library of adapters for popular commercial and open source grids
- Key Use Cases
  - Data-aware job scheduling
  - Dynamic data grid right-scaling

Logical Architecture



Implementation

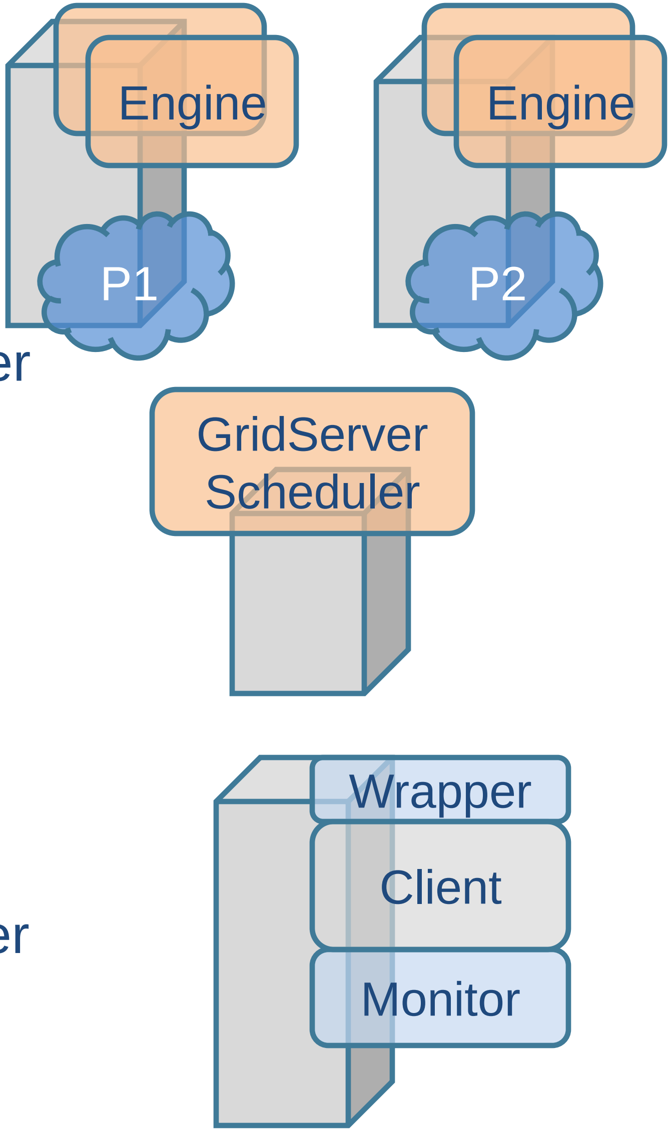
- Core Components
  - Data Grid Monitor: service responsible for knowing Data Grid’s topology and state
  - Data Aware Wrapper: client side library which extends Compute Grid’s scheduling API to support data-aware job scheduling

- Main Workflow
  - Client code submits the job using Data Aware Wrapper
  - Data Aware Wrapper consults Data Grid Monitor
  - Data Grid Monitor returns a set of hosts that are nearest to the data
  - Wrapper submits the job to the Scheduler, requesting specific hosts

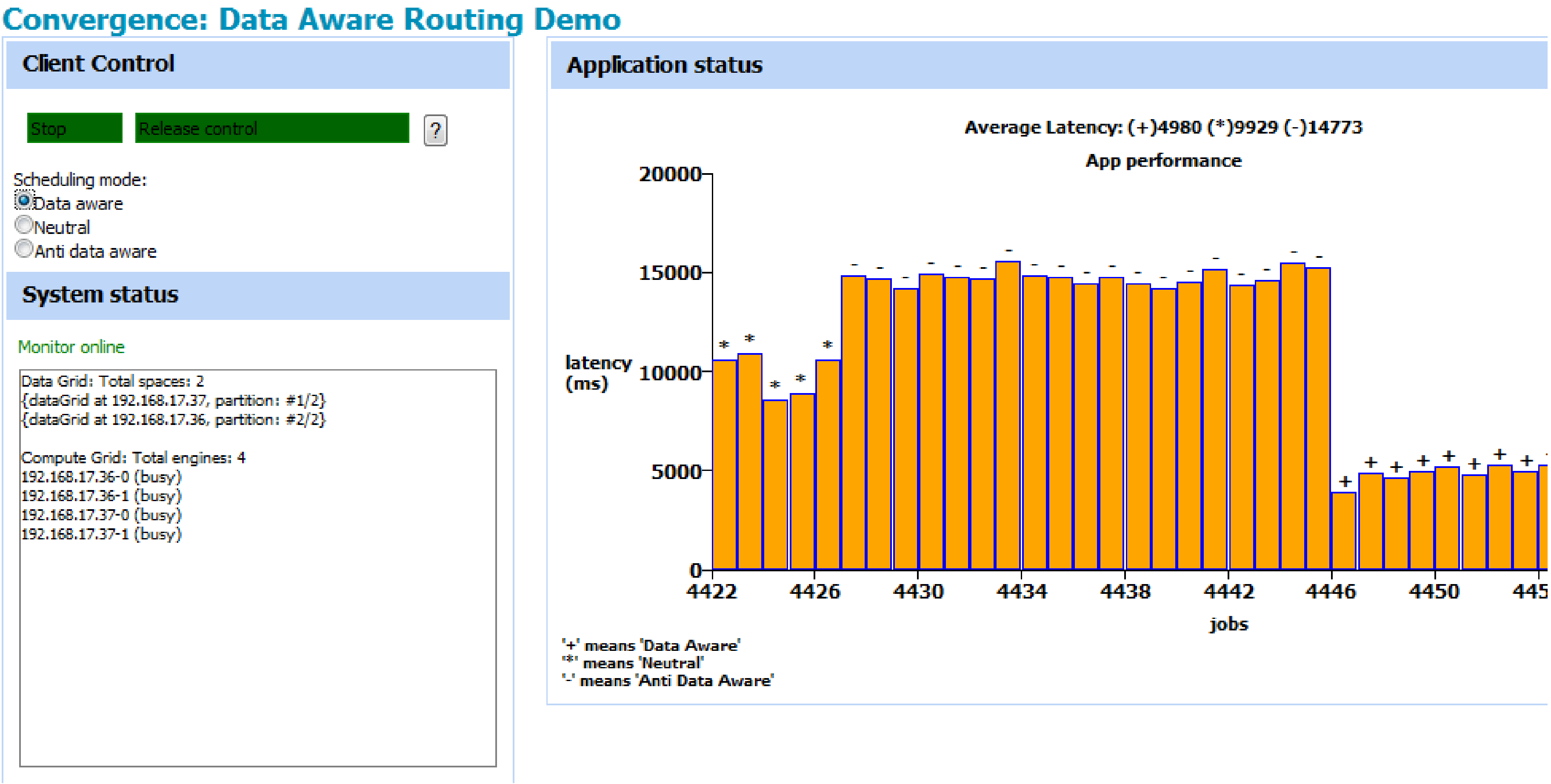
- Variation on Configuration
  - Monitor can be a network service or embedded as a library

Demo – “Hello, World” Trading Analytics

- Setup
  - 4 DataSynapse Engines, 2 per host
  - 2 GigaSpaces partitions
  - Scheduler: DataSynapse GridServer
  - Client app + embedded Monitor + Wrapper
- Test data
  - Stores100,000 trades for 10 stock tickers
  - Partitioned by ticker
- Job
  - Computes simple statistics about trades
  - A Job spawns 10 tasks, one task per ticker
- Task Scheduling Control Functions
  - Data-aware, random, or anti-data-aware



Demo Screenshot



Convergence: Data Aware Routing Demo

Scheduling mode:
● Data aware
○ Neutral
○ Anti data aware

Monitor online

```
Data Grid: Total spaces: 2
{dataGrid at 192.168.17.37, partition: #1/2}
{dataGrid at 192.168.17.36, partition: #2/2}

Compute Grid: Total engines: 4
192.168.17.36-0 (busy)
192.168.17.36-1 (busy)
192.168.17.37-0 (busy)
192.168.17.37-1 (busy)
```

Average Latency: (+)4980 (*)9929 (-)14773

'+' means 'Data Aware'
'*' means 'Neutral'
'-' means 'Anti Data Aware'

Current Project State

- Hosted by OpenSpaces.org
- Licensed under Apache 2.0
- Latest version 0.1.1 (Apr 2008 release)
- Use case supported: data-aware job scheduling
- Available plug-ins:
  - Compute Grids: Data Synapse GridServer 5.0
  - Data Grid: GigaSpaces XAP

Project Roadmap

- Support Additional Adapters
  - Convergence 0.2: GridGain (under development)
  - Convergence 0.3: Oracle Coherence
  - Convergence 0.4: Sun Grid Engine
- Support Additional Use Cases
  - Dynamic data grid right-sizing
- Call for Action
  - Please, join the project to help test and extend the system, or provide additional adapters

http://www.openspaces.org/display/CVG/Convergence

Q & A

Grid Dynamics

Experts in *Extreme* Applications

Thank You!

Eugene Steinberg,
CTO
esteinberg@griddynamics.com