



Apache Ignite™ (Incubating) - In-Memory Data Fabric

Fast Data Meets Open Source

DMITRIY SETRAKYAN

Founder, PPMC

<http://www.ignite.incubator.apache.org>

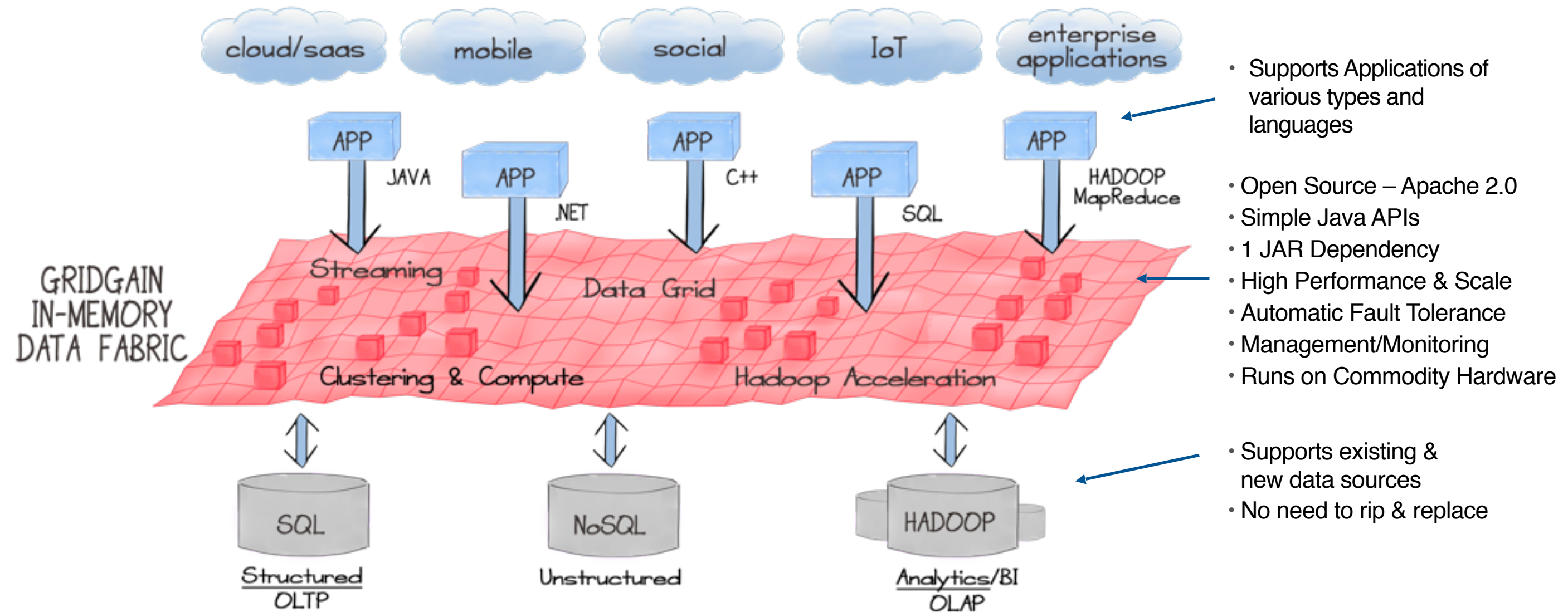


#apacheignite

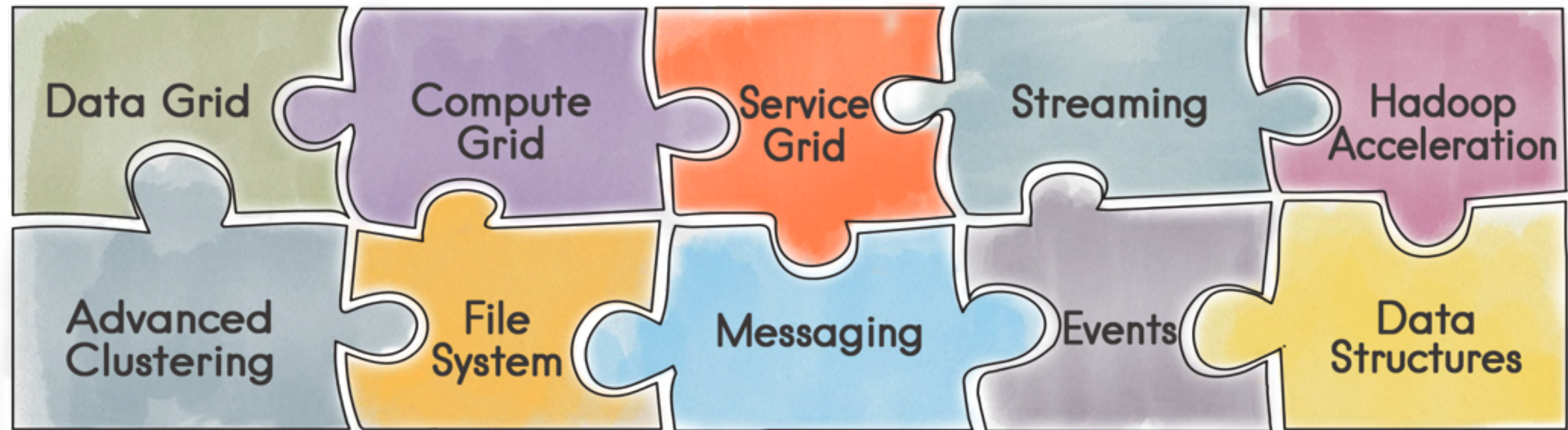
Agenda

- Apache Ignite^(tm) In-Memory Data Fabric
 - Advanced Clustering
 - Data Grid
 - Compute Grid
 - Service Grid
 - Streaming & CEP
 - Plug-n-Play Hadoop Accelerator
- Benchmarking
- Q & A

Apache Ignite™ In-Memory Data Fabric: Strategic Approach to IMC

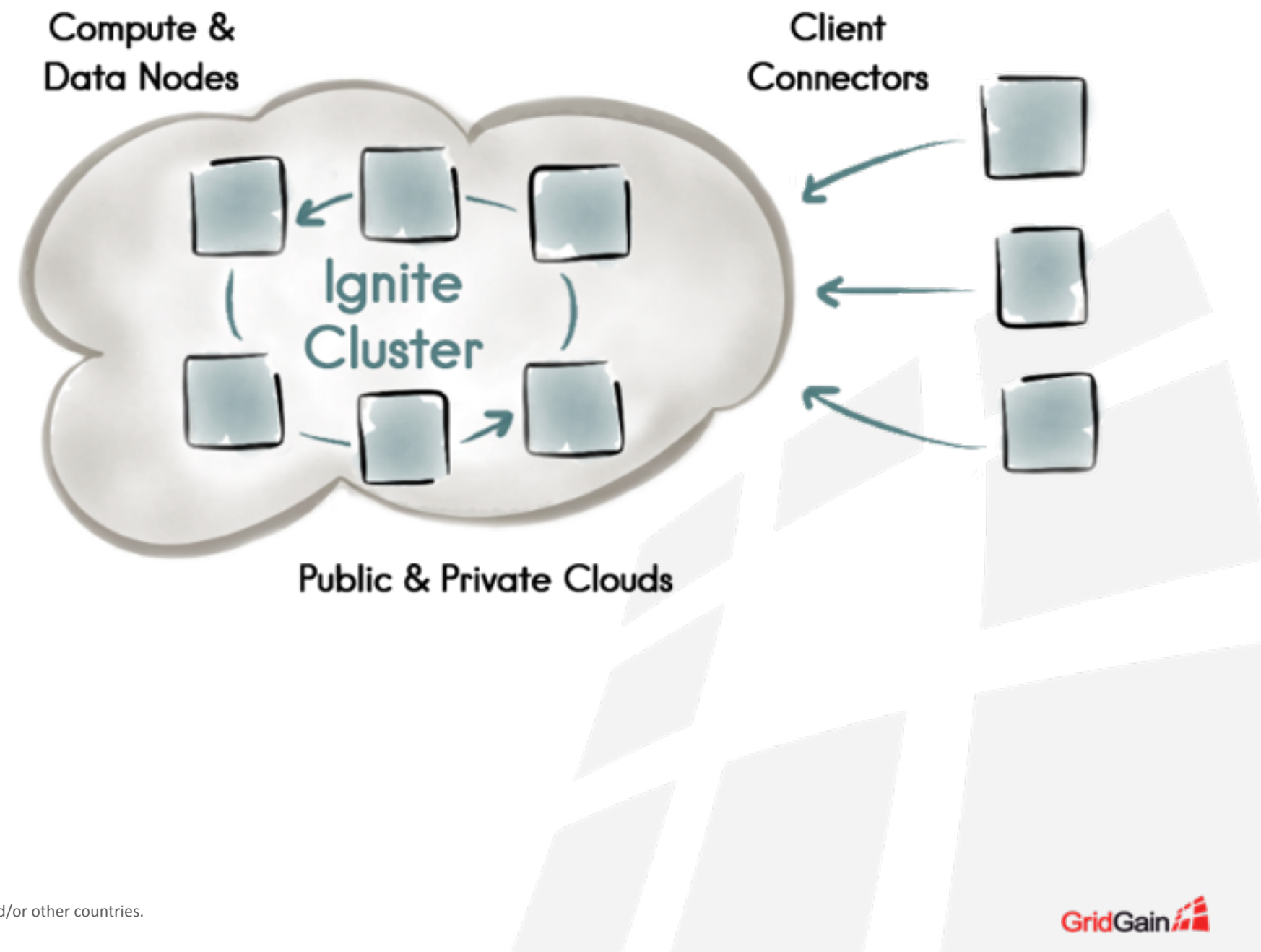


In-Memory Data Fabric: More Than Data Grid



In-Memory Data Fabric: Clustering

- Ease of Getting Started
 - Automatic Discovery
- Any Environment
 - Public Cloud (AWS, OpenStack)
 - Private Cloud
 - Hybrid Cloud
 - Local Laptop
- Zero-Deployment
 - Auto-Deploy Code
- Pluggable Design



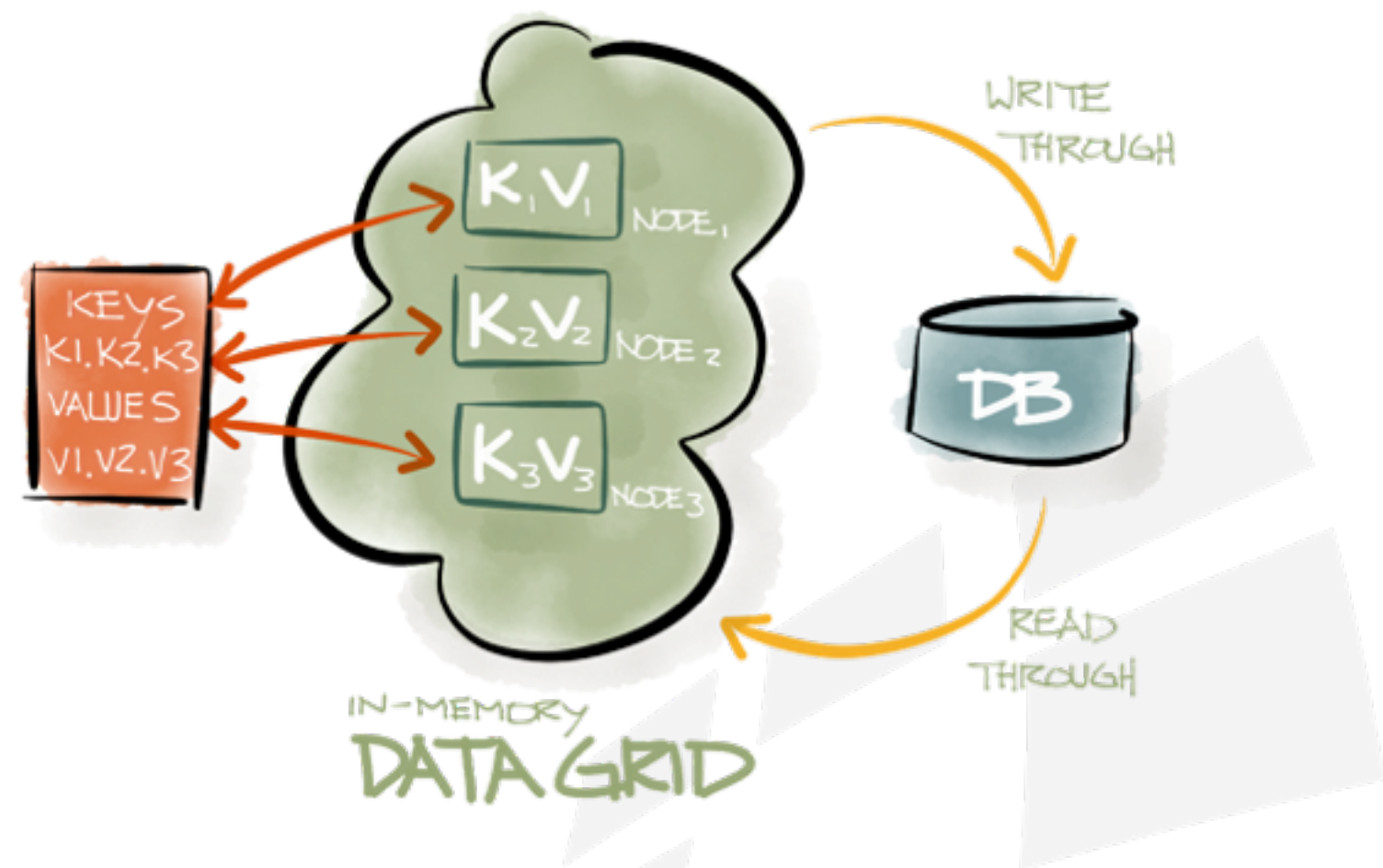
In-Memory Data Fabric: Clustering

```
[14:04:22]
[14:04:22] /____/____( )____/____/____( )____
[14:04:22] / ( _// _// _// ( _// _// _// _// _//
[14:04:22] \____// _// _// \, _\____/ \, _// _// _//
[14:04:22]
[14:04:22] ver. 6.5.0-os#20140925-sha1:6dc3d773
[14:04:22] 2014 Copyright (C) GridGain Systems
[14:04:22]
[14:04:22] Quiet mode.
[14:04:22] ^-- Logging to file '/Users/Dmitriy/GridGain/release/6.5.0/gridgain-fabr
ic-os-6.5.0/work/log/gridgain-cacf70c7.0.log'
[14:04:22] ^-- To see **FULL** console log here add -DGRIDGAIN_QUIET=false or "-v"
to ggstart.{sh|bat}
[14:04:22]
[14:04:23] Failed to initialize HTTP REST protocol (consider adding gridgain-rest-htt
p module to classpath).
[14:04:24] If running benchmarks, see http://bit.ly/GridGain-Benchmarking
[14:04:24] To start Console Management & Monitoring run ggvisorcmd.{sh|bat}
[14:04:24]
[14:04:24] GridGain node started OK (id=cacf70c7)
[14:04:24] Topology snapshot [ver=1, nodes=1, CPUs=8, heap=1.0GB]
[14:38:21] Topology snapshot [ver=2, nodes=2, CPUs=8, heap=2.0GB]
```

```
dsetmac-2:gridgain-fabric-os-6.5.0 $ bin/ggstart.sh
[14:38:20]
[14:38:20]  /____/____( )____/____/____( )____
[14:38:20] / ( // // // _ / ( // _ ' // _ \
[14:38:20] \____// // // \, \____/ \, \____// //
[14:38:20]
[14:38:20] ver. 6.5.0-os#20140925-sha1:6dc3d773
[14:38:20] 2014 Copyright (C) GridGain Systems
[14:38:20]
[14:38:20] Quiet mode.
[14:38:20] ^-- Logging to file '/Users/Dmitriy/GridGain/release/6.5.0/gridgain-fabr
ic-os-6.5.0/work/log/gridgain-c7b4932f.0.log'
[14:38:20] ^-- To see **FULL** console log here add -DGRIDGAIN_QUIET=false or "-v"
to ggstart.{sh|bat}
[14:38:20]
[14:38:21] Failed to initialize HTTP REST protocol (consider adding gridgain-rest-htt
p module to classpath).
[14:38:22] If running benchmarks, see http://bit.ly/GridGain-Benchmarking
[14:38:22] To start Console Management & Monitoring run ggvisorcmd.{sh|bat}
[14:38:22]
[14:38:22] GridGain node started OK (id=c7b4932f)
[14:38:22] Topology snapshot [ver=2, nodes=2, CPUs=8, heap=2.0GB]
```

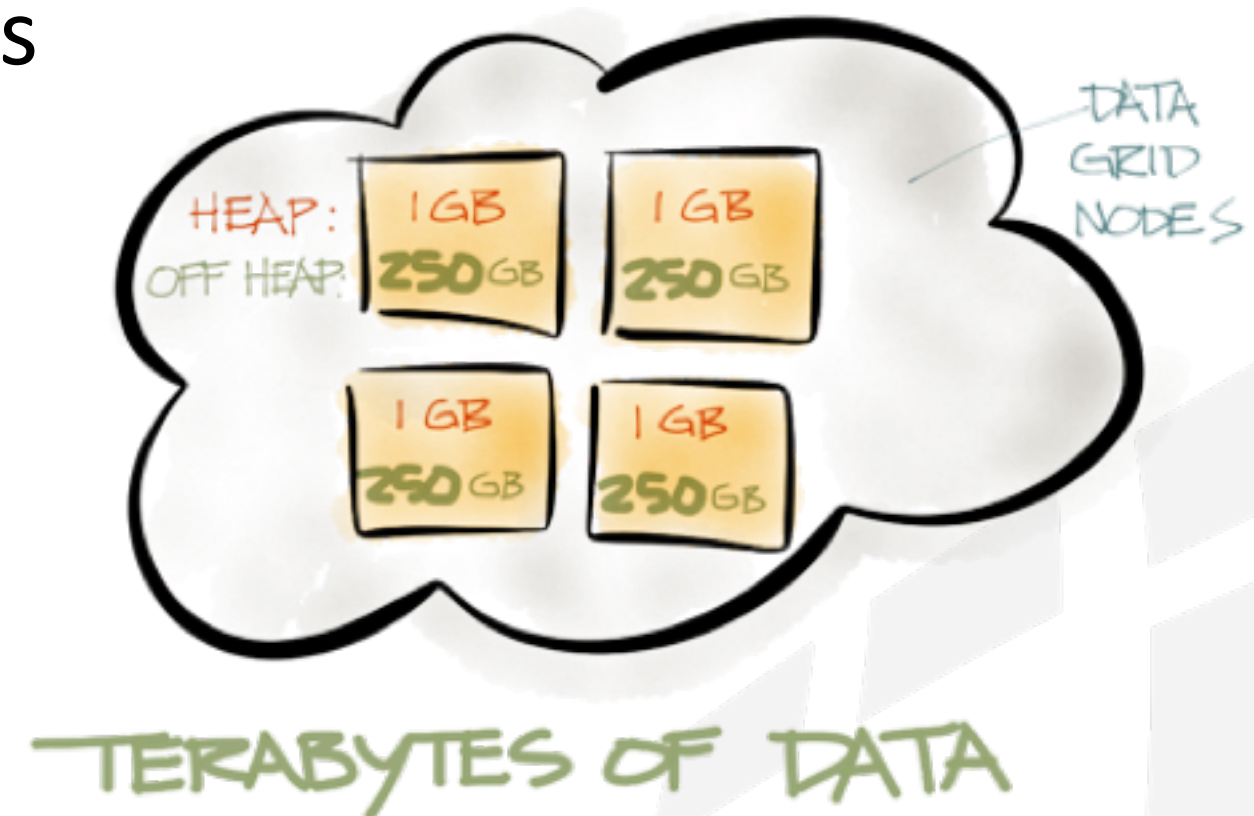
In-Memory Data Fabric: Data Grid

- Distributed In-Memory Key-Value Store
- Replicated and Partitioned data
- TBs of data, of any type
- Redundant Backups
- Distributed ACID Transactions
- SQL queries and JDBC driver (ANSI 99)
- Data Structures (Queue, AtomicLong, etc.)
- Collocation of Compute and Data



In-Memory Data Fabric: Off-Heap Memory

- Unlimited Vertical Scale
- Avoid Java Garbage Collection Pauses
- Small On-Heap Footprint
- Large Off-Heap Footprint
- Off-Heap Indexes
- Full RAM Utilization
- Simple Configuration



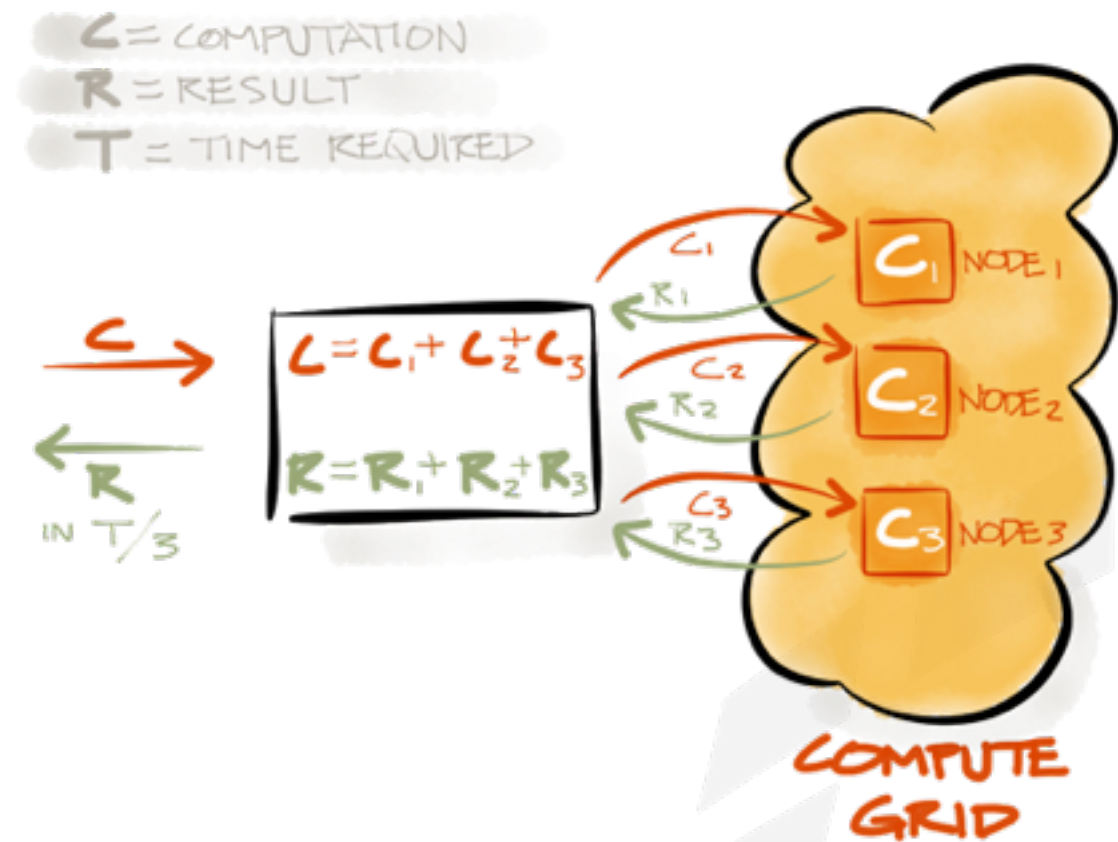
Distributed Java Structures

- Distributed Map (cache)
- Distributed Set
- Distributed Queue
- CountdownLatch
- AtomicLong
- AtomicSequence
- AtomicReference
- Distributed ExecutorService

```
GridCacheQueue<Integer> queue =  
    dataStructures.queue( "myQ",  
  
    // Distribute queue elements  
    // across grid.  
    for (int i = 0; i < 20; i++)  
        queue.add(i);  
  
    // Poll queue elements.  
    for (int i = 0; i < 20; i++)  
        queue.poll();
```

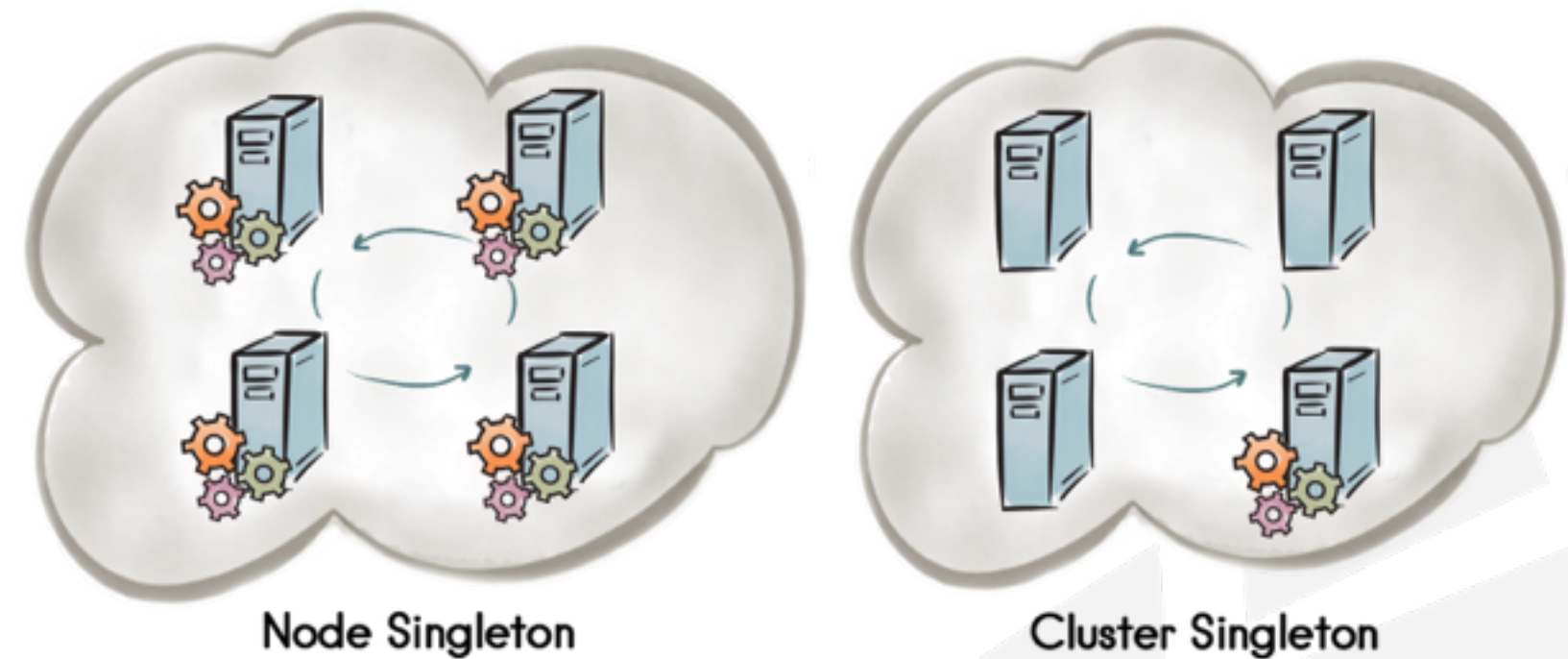
In-Memory Data Fabric: Compute Grid

- Direct API for MapReduce
- Direct API for ForkJoin
- Zero Deployment
- Cron-like Task Scheduling
- State Checkpoints
- Load Balancing
- Automatic Failover
- Full Cluster Management
- Pluggable SPI Design



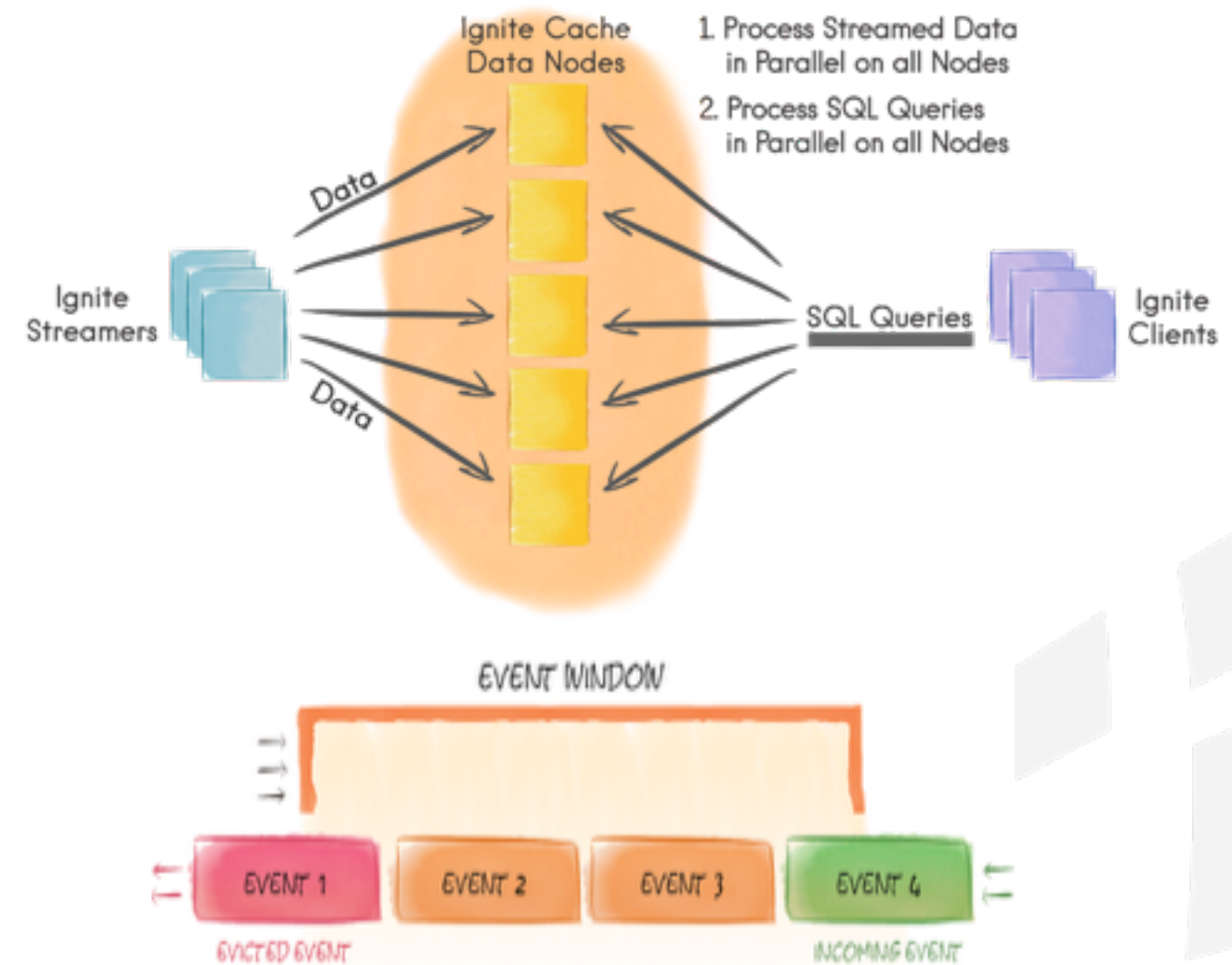
In-Memory Data Fabric: Service Grid

- **Singletons on the Cluster**
 - Cluster Singleton
 - Node Singleton
 - Key Singleton
- **Distribute any Data Structure**
 - Available Anywhere on the Grid
 - Access Anywhere via Proxies
- **Guaranteed Availability**
 - Auto Redeployment in Case of Failures



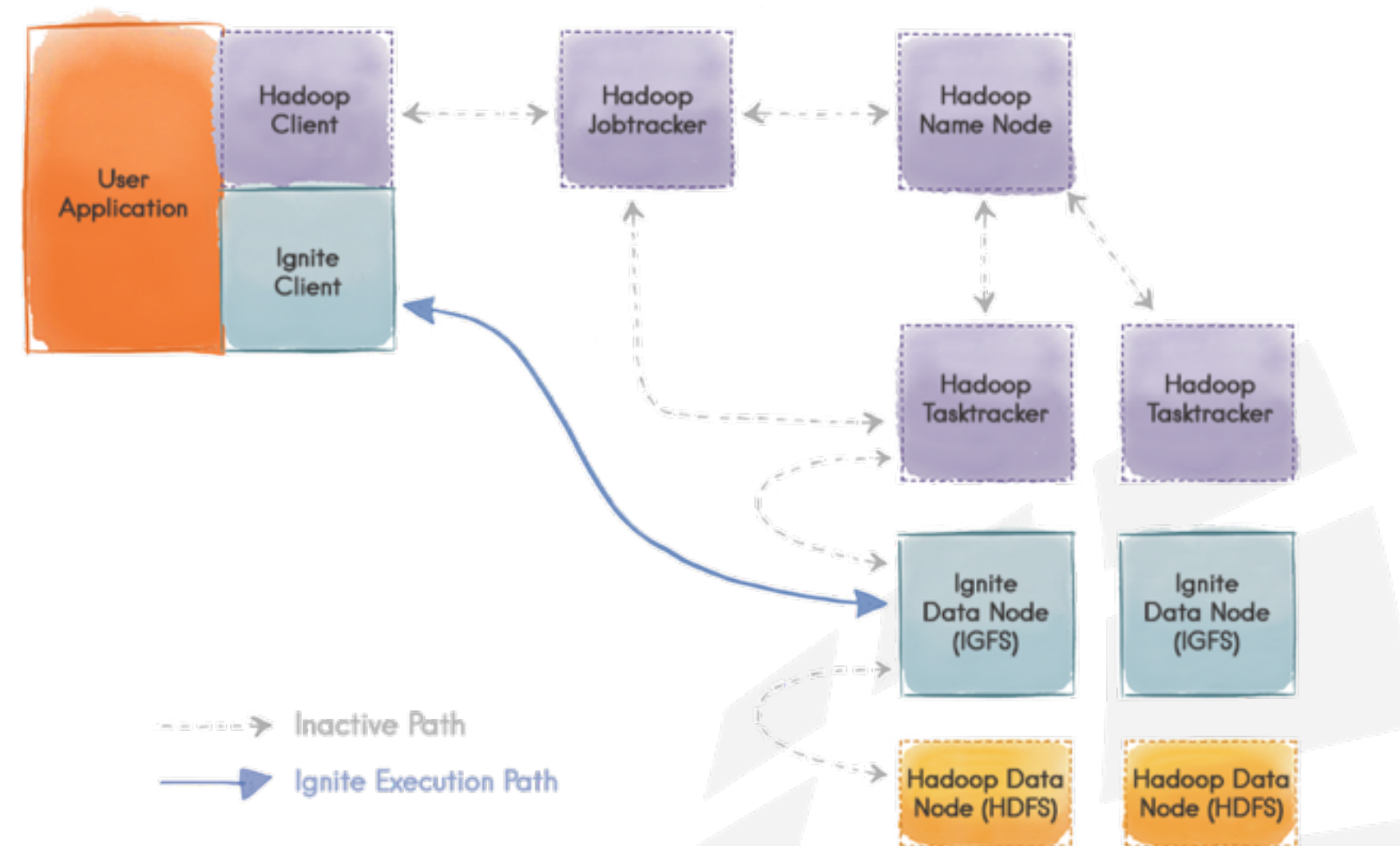
In-Memory Data Fabric: Streaming and CEP

- Streaming Data Never Ends
- Branching Pipelines
- Pluggable Routing
- Sliding Windows
- CEP/Continuous Query
- SQL Query
- Real Time Analysis



In-Memory File System & Accelerated Map Reduce

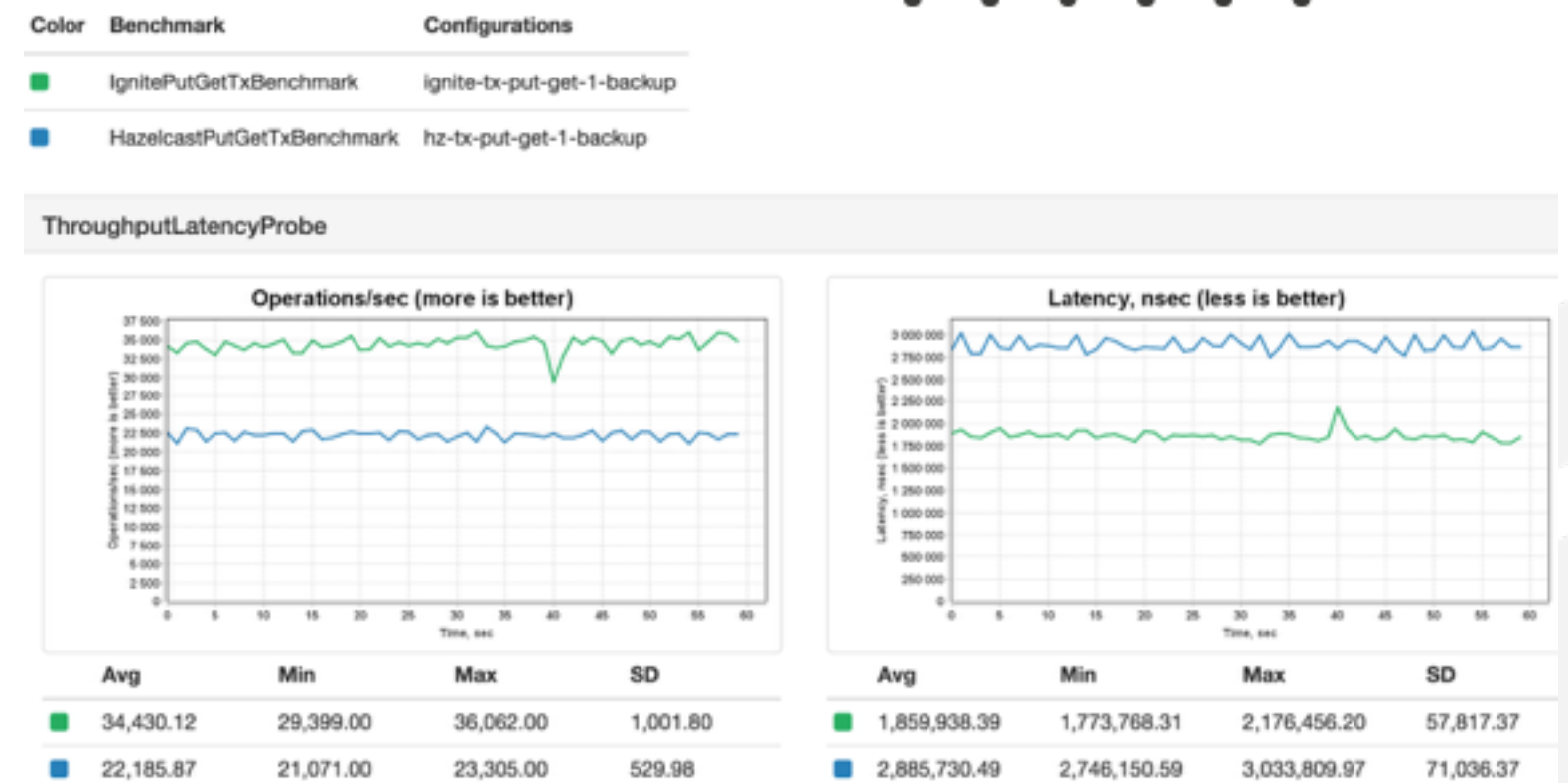
- IGFS - In-Memory File System
- Zero Code Change
- Use existing MR code
- Use existing Hive queries
- No Name Node
- Write-through to HDFS
- In-Process Data Colocation
- Eager Push Scheduling



Yardstick: Distributed Benchmarking



- Use Yardstick Docker Containers
 - Apache Ignite
 - Hazelcast
 - Apache Spark (coming)
 - Apache Cassandra (coming)
- AWS Images Ready to Go
- View Results in S3 Bucket



<https://github.com/yardstick-benchmarks/yardstick-docker>

ASF: Spark vs Ignite

Apache Ignite

- Computational analytics
 - In-Memory Indexes
 - Real Streaming
- Transactional data processing
- Classic High Performance Compute



Apache Spark

- Interactive analytics
 - Full Scans (no indexes)
 - Discretized Streaming
- Machine learning
- Classic data science

Coding Examples

- Compute Grid
- Data Grid



Join us This Year at the Industry-First



JUNE 29-30 - SAN FRANCISCO, CA

For networking, education and exchange of ideas

Use promo code “IMCS629M” to register and get 30% off with the early bird rate at

www.imcsummit.org/register-meetup/



ANY QUESTIONS?

Thank you for joining us. Follow the conversation.

<http://www.ignite.incubator.apache.org>



#apacheignite