How to Choose the Best
In-Memory Solutions for Your Apps

DMITRIY SETRAKYAN
GridGain Founder
Apache Ignite PMC Chair

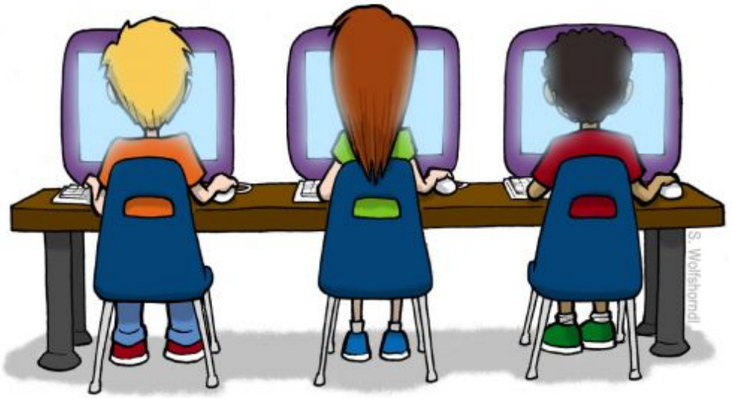
GridGain

Content

- IMC Introduction
- IMC Myths
- IMC Product Categories
- Key IMC Features
- Key IMC Use Cases & Applications

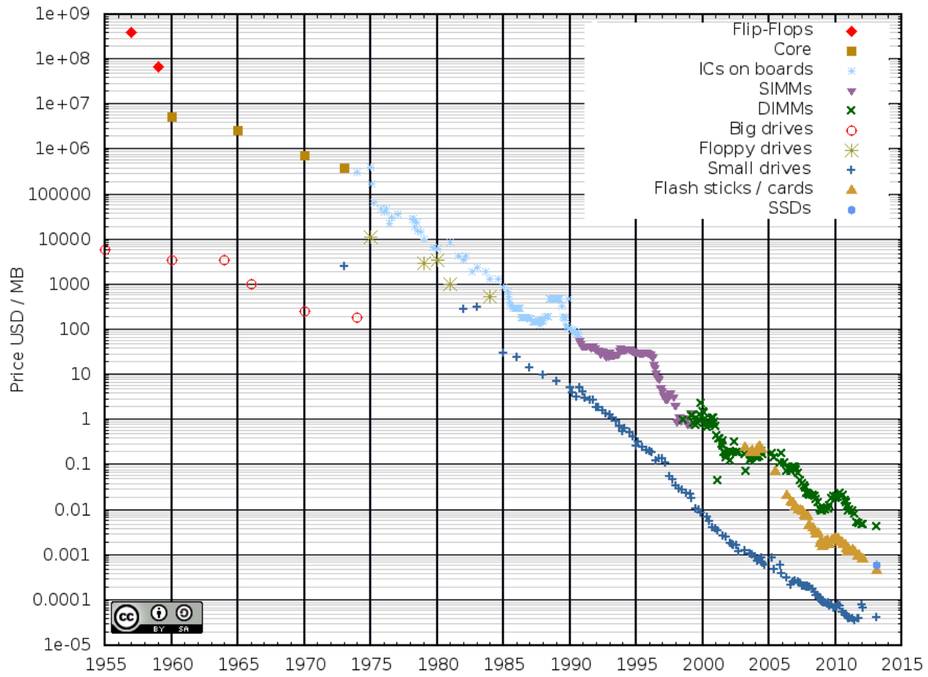
Apache Ignite: We Are Hiring!

- Very Active Community
- Great Way to Learn Distributed Computing
- How To Contribute:
  - https://ignite.apache.org/community/contribute.html#contribute
  - https://cwiki.apache.org/confluence/display/IGNITE/How+to+Contribute



Introduction

In-Memory Computing uses high-performance, distributed memory systems to compute and transact on large-scale data sets in real-time - orders of magnitude faster than disk-based systems.



GridGain

Paradigm Shift

- **1950s – Era of non-volatile storage begins**
  - External (relative to RAM) storage medium
  - 1951 Remington introduced 1st tape drive UNISERVO (224 KB)
- **1970s – Era of HDD(*)**
  - IBM released “Winchester” IBM 340 disk
  - SQL Revolution (structured data)
- **2000s – Era of Flash(*)**
  - Toshiba introduced NAND (1989)
- **2010s – Era of Memory**
  - 64-bit CPUs + DRAM prices drop 30% YoY
  - NoSQL + SQL (unstructured data)
  - **Last frontier for storage**

RAM is a new disk, disk is a new tape.

Gartner®

Memory First vs. Disk First

- **Disk First Architecture**
  - Disk as primary storage, memory for caching
  - Access chain: API call <> OS I/O <> I/O controller <> disk
  - Latency: milliseconds
- **Memory First Architecture**
  - Memory as primary storage, disc for backup
  - Access chain: API call <> pointer arithmetic
  - Latency: nanoseconds to microseconds

Myth #1: Too Expensive

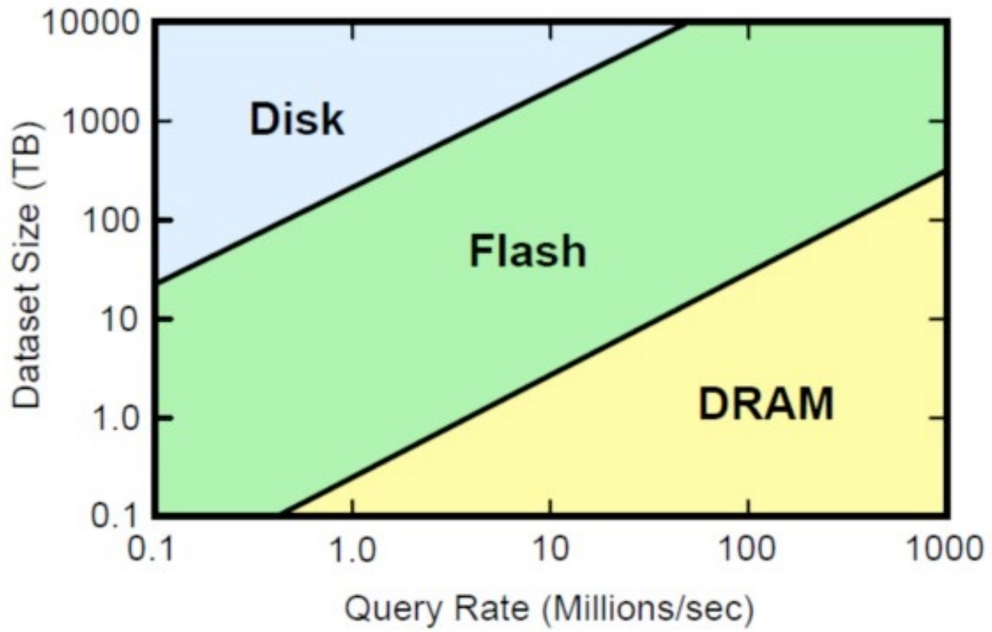
- **Facts:**
  - 2015: 1TB DRAM cluster ~ $20K
    - 30% YoY price reduction
  - Memory Channel Storage (MCS)
    - NAND in DRAM form factor, 2x speed of flash, same price as flash
  - Storage Class Memory (SCM)
    - ~10x slower than DRAM, Flash price, non-volatile

Myth #2: Not Durable

- **Facts:**
  - IMC have durable backups and disk storage
    - Active or passive replicas, transactional read-through and write-through
  - Mature IMC provide tiered storage
    - DRAM - Local Swap - RDBMS/HDFS
  - Operational vs. Historical datasets
    - 99% of operational datasets < 10TB

Myth #3: Flash Is Fast Enough

- **Facts:**
  - Flash on PCI-E is still... a block device.
    Still going through OS I/O, I/O controller, marshaling, buffering.



Myth #4: Only For Caching

- **Facts:**
  - Caching is important use case for yesterday
    Easiest adoption and a “low-hanging fruit”
  - In-Memory Data Grids & Fabric for today
    Main system of records moving to in-memory
  - Vertical and PnP products are the future
    Minimal integration, maximum benefit

IMC Product Categories

- **In-Memory "Options"**
  - Oracle Database 12c, Microsoft SQL Server
- **In-Memory RBDMS**
  - MemSQL, VoltDB, SAP HANA
- **In-Memory Data Grids**
  - Hazelcast, GigaSpaces, Coherence, GemFire, Infinispan
- **In-Memory Data Fabrics**
  - GridGain (Apache Ignite), Terracotta

Category: In-Memory "Options"

- Feature onto an **EXISTING** databases
- Ideal when only configuration change is possible:
  - No API changes
  - No code changes
  - No data migration
- Limited benefits
  - Basically a "marketing" for a modern caching

Fast Data & Big Data

- Fast Data
  - OLTP mostly
  - Smaller Operational Data Set
  - High Throughput (ops/sec)
  - Low Latencies
  - Consistent or Transactional

- Big Data
  - OLAP mostly
  - Larger Historical Data Set
  - Read-Mostly
  - Throughput Not Important
  - Low Query Latencies
  - Good-enough for interactive analytics

Fast Data & Big Data

- Fast Data
  - Streaming
    - Apache Flink
    - Apache Kafka
    - Apache Apex
  - In-Memory Data Grid / Fabric
    - Apache Ignite
    - Apache Geode
  - In-Memory Database
    - MemSQL
    - VoltDB
  - NoSQL
    - MongoDB
    - Apache Cassandra

- Big Data
  - Apache Hadoop
    - MapReduce
    - HDFS
    - HBase
  - Apache Spark
    - Machine Learning
    - Graph Processing
    - SQL
  - Warehouse/DB Vendors

Category: In-Memory Databases

- In-Memory Databases
  - MemSQL
    - Closed Source
    - Free Limited Community Edition
  - VoltDB
    - Open Source Community Edition (AGPL)
    - Closed Source Enterprise Edition
- Main Features
  - High-Throughput
  - Low Latencies
  - Full SQL Support
    - However, SQL is the only API
  - Disk Persistence
    - Disk is just a copy of memory

- **Complete replacement of existing databases! Good or Bad?**

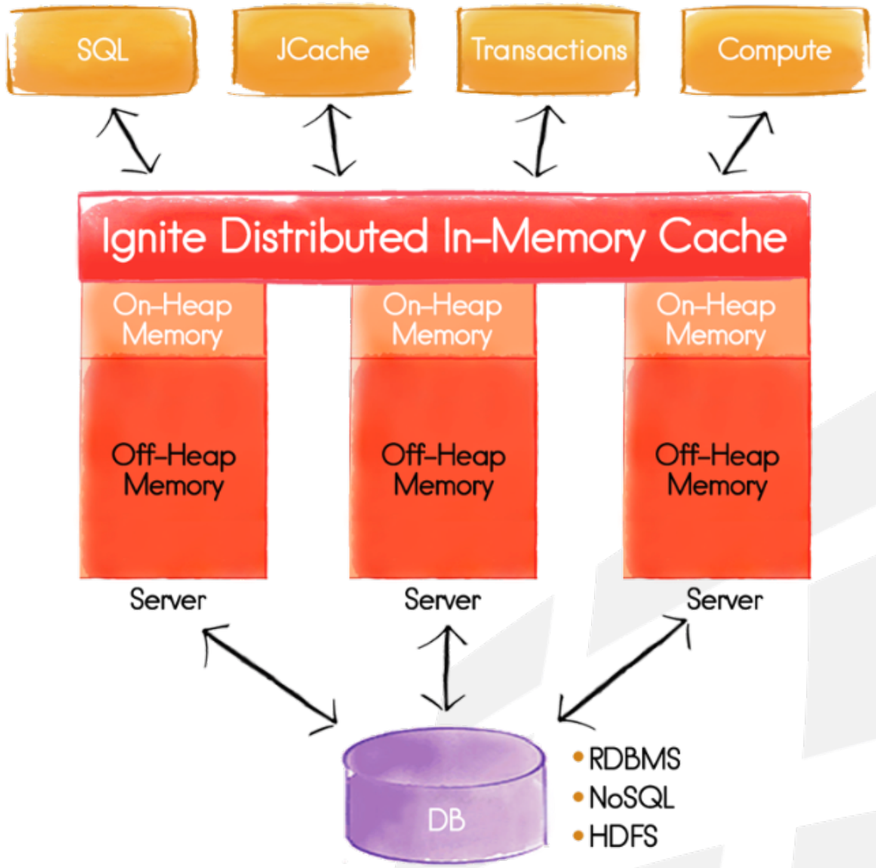
Category: **In-Memory Data Grids**

- In-Memory Data Grids
  - Apache Ignite – In-Memory Data Fabric
  - Apache Geode (incubating)
- Main Features
  - High Throughput
  - Low Latencies
  - Key-Value Store
  - Transactions
  - Data Querying Capability
  - Disk Persistence
    - Read & Write-through to databases
    - Keep your existing database

Apache Ignite Data Grid

- Based on JCache (JSR 107)
  - In-Memory Key-Value Store
  - Basic Cache Operations
  - ConcurrentMap APIs
  - Collocated Processing (EntryProcessor)
  - Events and Metrics
  - Pluggable Persistence
- Ignite Data Grid
  - ACID Transactions
  - SQL Queries (ANSI 99)
  - In-Memory Indexes
  - On-Heap & Off-Heap Memory
  - Automatic RDBMS Integration

JCache



GridGain

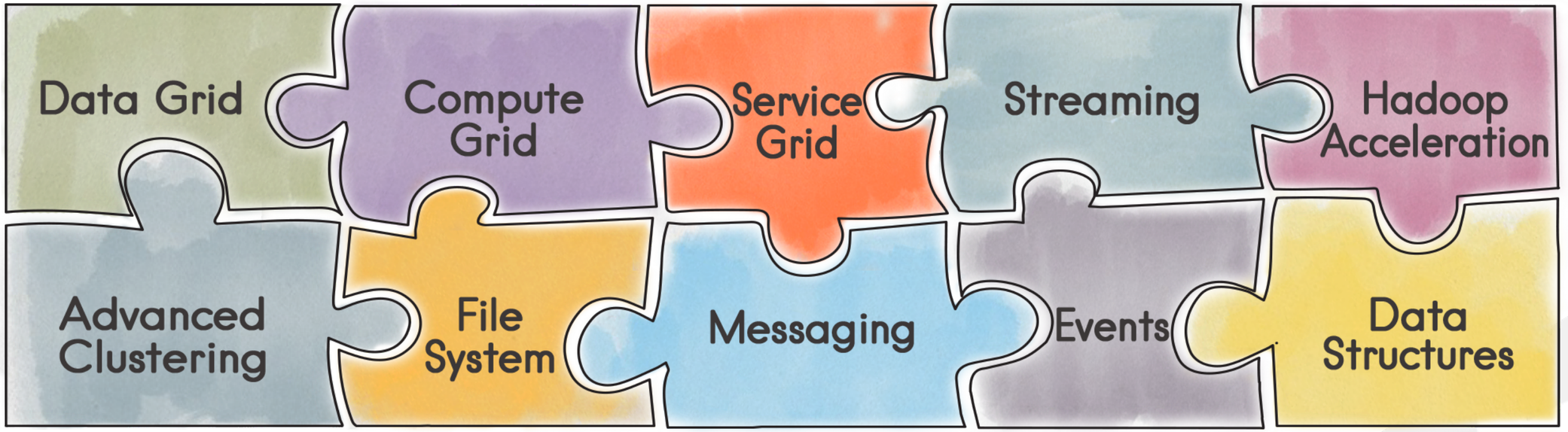
Apache Ignite: Ad-Hoc SQL (ANSI 99)

- ANSI-99 SQL
- JDBC & ODBC Drivers
- Always Consistent
- Fault Tolerant
- In-Memory Indexes (On-Heap and Off-Heap)
- Automatic Group By, Aggregations, Sorting
- Cross-Cache Joins, Unions, etc.
- Non-Collocated Joins
- Ad-Hoc SQL Support
- DDL & DML coming this year (IMDB?)

Category: In-Memory Data Fabric

- **Strategic view** on IMC processing
- Collection Of Integrated In-Memory Components
- Covers most IMC use cases: data management, compute, streaming, etc.
- May require significant refactoring
  - When moving from SQL-based processing

Apache Ignite In-Memory Data Fabric



apache Ignite

GridGain

Data Grids & Fabrics: Key Features

- Scale-Out Clustering
- Scale-Up Deployment
- **In-Memory Data Grids**
  - Fault tolerance, high availability
  - Transactions
  - Distributed SQL support
- Cross Language Interop:
  - C++, .NET, node.js, REST
- **In-Memory Compute Grids**
  - Distributed Closures
  - Fork/Join, MapReduce
- **In-Memory Streaming**
- Security
- Management & administration
- Big data stack integration
  - Hadoop, Spark
- Deployment options:
  - Mesos, AWS, YARN

THANK YOU!