MapR Streams
A global pub-sub event streaming system for big data and IoT

Ben Sadeghi – Data Scientist, APAC
IDA Forum on IoT – Jan 18, 2016
MapR Streams: Vision

To enable continuous, globally scalable streaming of event data, allowing developers to create real-time applications that their business can depend on.

Converged
Continuous
Global
MapR Converged Data Platform

Open Source Engines & Tools
- Hive
- Spark
- Apache Drill
- Search & Others

Commercial Engines & Applications
- Cloud & Managed Services
- Custom Apps

Utility-Grade Platform Services
- Enterprise Storage
  MapR-FS
  - Global Namespace
- Database
  MapR-DB
  - High Availability
  - Data Protection
  - Self-healing
  - Unified Security
- Event Streaming
  MapR Streams
  - Real-time
  - Multi-tenancy
Big Data is Generated One Event at a Time

“time” : “6:01.103”,
“event” : “RETWEET”,
“location” :
 “lat” : 40.712784,
 “lon” : -74.005941

“time” : “5:04.120”,
“severity” : “CRITICAL”,
“msg” : “Service down”

“card_num” : 1234,
“merchant” : ”Apple”,
“amount” : 50
Batch Processing Has Many Use Cases

- Customer 360
- Sentiment analysis

- Clickstream analysis
- Predictive maintenance

- Fraud detection
- Coupon offers
- Risk models
Real-time Processing is Complementary

- Trending now
- News feed
- Ops dashboards
- Failure alerts
- Breach detection
- Real-time fraud detection
- Real-time offers
- Push notifications
 Streams Simplify Data Movement

**Streams**
Reliable publish/subscribe transport between sources and destinations.

- Filtering & Aggregation
- Alerting
- Processing
Legacy Systems: Message Queues
IBM MQ, TIBCO, RabbitMQ

Usage/Requirements
- Tight, transactional conversations between systems
- 1:1 or Few:Few
- Low data rates
- Mission-critical delivery

Approach
- Queue-oriented design
  - Each message replicated to N output queues
  - Messages popped when read
- Scale-up, master/slave

Doesn’t Do
- High message rates (>100K/s)
- Slow consumers
- Queue replay/rewind
Evolving “big data” Event Streams: Distributed Logs
Kafka, Hydra, DistributedLog

Usage/Requirements
- High throughput data transferred from decoupled systems
  - Many -> 1
  - 1 -> Many
  - Different speeds

Approach
- Log-oriented design
  - Write messages to log files
  - Consumers pull messages at their own pace
  - Scale-out

 Doesn’t Do
- Global applications
- Message persistence
- Integrated analytics (data movement required)
MapR: Rethinking a Platform for Event Streams

- “Big data” scale and performance
- Global applications and data collection
- Multi-tenant and multi-application
- Secure
- Analytics-ready (no movement)
- Converged: no cluster sprawl
MapR Streams
Converged, Continuous, Global
Producers publish **billions** of messages/sec to a topic in a stream.

**Guaranteed, immediate delivery** to all consumers.

Tie together geo-dispersed clusters. **Worldwide.**

**Standard real-time API** (Kafka).
Integrates with Spark Streaming, Storm, Apex, and Flink

**Direct data access** (OJAI API) from analytics frameworks.
MapR Streams - Converged, Continuous, Global

**Converged**
- **Converged platform** with file storage and database
- OJAI API - **Direct access** from analytics tools
- Unified **security** framework with files and database tables
- **Multi-tenant** - topic isolation, quotas, data placement control

**Continuous**
- **Integrated** with Spark Streaming, Flink, Apex, others
- **Message persistence** for up to infinite time span
- **Guaranteed** delivery (at least once)
- Consistent, synchronous replication & no single point of failure

**Global**
- Native, **global data and metadata replication** with arbitrary topology
- Millions of streams, 100K topics/stream
- Billions of events per second
- Millions of producers & consumers
Global

Provides
- Arbitrary topology of thousands of clusters
- Automatic loop prevention
- DNS-based discovery
- Globally synchronized message offsets and consumer cursors

Enables
- Global applications & data collection
- Producer & consumer failover
- Analysis/filtering/aggregation at the edge
- “Occasional” connections
Top Differentiators

**Converged**
- Single cluster for files, tables, and streams.
- Data persistence and direct data access to batch processing frameworks.
- Authentication, authorization, encryption. **Unified policy** with all other platform services.

**Global**
- Global, IoT-scale “fabrics” with failover.

**Secure & Multi-Tenant**
- Tenant-owned streams, logical grouping of topics and messages.
Life Without a Converged Platform

- **Sources**
  - Streaming (TIBCO, IBM, Kafka)
  - Batch Loads
  - Operational Cluster (HBase, Cassandra)
  - Enterprise Storage (system of record)
  - Real-time

- **Stream Processing**
  - Spark

- **Analytics**
  - Open Source Analytics (Hadoop, Spark)

- **Apps**
  - Apache Drill
  - Enterprises Storage (system of record)
Life With a Converged Platform

Utility-Grade Platform Services

Data

- Enterprise Storage
  MapR-FS
  - Global Namespace

- Database
  MapR-DB
  - High Availability
  - Data Protection
  - Self-healing
  - Unified Security

- Event Streaming
  MapR Streams
  - Real-time
  - Multi-tenancy

Only full-stack “big data” platform.
Part of a Converged Reference Architecture

Source → Capture → Store → Process → Serve

- Source: NFS, Flume
- Capture: MapR-FS, ElasticSearch, MapR-DB, Streams
- Store: Spark
- Process: Drill, Spark Streaming
- Serve: Kibana Dashboard, Ops Dashboard
IoT Data Transport & Processing

Business Results
- New revenue streams from collecting and processing data from “things”.
- Low response times by placing collection and processing near users.

Why Streams
- IoT is event-based, and needs an event streaming architecture.

Why MapR
- Converged platform gives single cluster, single security model for data in motion and at rest.
- Reliable global replication for distributed collection, analysis, and DR.
Q&A
Engage with us!

@bensadeghi, @mapr
maprtech

mapr-technologies
MapR

bsadeghi@mapr.com
maprtech