A streaming analytics platform for real-time business decisions

Alok Pareek, Bhushan Khaladkar, Rajkumar Sen, Basar Onat, Vijay Nadimpalli, Manish Agarwal, Nicholas Keene
Striim is an Intel & Dell funded company
Striim Safe Harbor

This following is for information only and represents Striim Inc.'s current view of its product development cycle.

Features and release dates are best estimates and should be considered provisional and subject to change without notice. There can be no guarantee that the release dates will be met or that the product or enhancements will be released at all.

Striim, INC. MAKES NO WARRANTIES, EXPRESS OR IMPLIED, IN THIS DOCUMENT.
Example problem statements.. In the real world Enterprise!

- For the BIRTE audience...
- Real Time Validation of Replacement of an aircraft part (Request spans multiple geographies and vendor systems)
  - Insider threat detection that leaks across lots of SIEM monitors
    - Involves lots of logs, correlation, SIEM monitors, data, visualization
  - Real Time Sharing of information with compliance and privacy
    - Involves device data collection, storing, obfuscation
  - Process Optimization (Factory floor through Enterprise)
    - Multiple communication levels (Device level – ERP)
Agenda

Striim – An Integrated Streaming Platform
Real-World Streaming Application - Demo
Key Technical Components/Contributions
Performance
What’s Ahead…?
Q&A
Striim Platform Overview

Real-Time Insights & Actions
- Alerts
- Triggers
- Ad-Hoc Queries

Machine Learning / AI

Databases & Data Warehouses

Messaging / Kafka

Big Data & NOSQL

Cloud

Files

Databases & Data Warehouses

Continuous Data Collection
- Databases
- Log files
- Messaging / Kafka
- Sensors

Continuous Queries / Time Series / Windowing

Transformation
- Filtering
- Enrichment

Aggregation

External Context

Real-Time Data Integration

Advanced Streaming Analytics
- Anomaly Detection
- Pattern Matching
- Multi-Stream Correlation

Real-Time Insights & Actions
- Real-time Dashboards
- Alerts
- Triggers
- Ad-Hoc Queries

Real-time Insights & Actions
- Real-time Insights & Actions
- Real-time Insights & Actions

Feed the Enterprise

© 2017 Striim, Inc. All rights reserved.
Data Ingestion – Collect Streaming Data

Including Non-Traditional un/semi-structured data

- Message Queues / Kafka Inherently Streaming
- Sensors / Devices Data is High Velocity And Might Need Edge Processing
- Files Need Continuous Parallel Collection
- Databases Need Change Data Capture (CDC)

Streaming Data Collection Allows Data to Move at its Own Speed
Data Preparation – Process

- Filter Out Unnecessary Data
- Transform to the Format You Need
- Aggregate to Remove Redundancy and Obtain Trends Over Time
- Simple & Easy to Use with all Processing through SQL
Deliver To Multiple Places (based on Need)

- Databases / ODS / EDW for data integration
- Files for up-stream processing
- Message Queues / Kafka for data as a service
- Cloud for elastic storage and scalability
- Hadoop / NOSQL for data lake

Facilitates self-service data access in DBs, Lake, etc.
Striim: A unified platform for streaming intelligence

- **A Single unified platform that combines**
  - Data Ingest/Capture
  - Real-time Event-Driven Analytics *using SQL*
  - Persistent Event Storage
  - Real-time visualizations
  - External Data Delivery

- **Benefits**
  - Build Fast Streaming Applications (reuse components)
  - Reduce Complexity - HA, Scalable, Declarative, Reliable, Manageable
  - Lowered TCO – Integrated, Replaces disparate stitched products
Demo
Striim: Core Components

- Real-time data capture
- Storage Manager
- Query Engine
- Recovery & Persisted Streams
- Real-time visualization Engine
Real-time Data Capture
Real-time data capture

- Built-in adapters (Parsers) to capture real-time events from a wide variety of data sources
  - Correlate /Join data without integrating (and paying for) third-party libraries

- **CDC Adapters**
  - Real-time transactional data from legacy databases: Oracle, SQL Server, HP NonStop, DB2 etc.

- **IoT Adapters**
  - Data from IoT devices using MQTT, OPC UA

- **File Capture** – Sequenced Coordination (Batched, Streaming)
Storage Manager Components

- **Stream**: Distributed data pipe across multiple components
  - Could be in-memory or persisted
- **Window**: Bound streaming data by time or count or both
  - Sliding, Jumping, Session
- **Cache**: In-Memory (refreshable) cache of historical data
  - Used to enrich real-time streaming data
- **Event Table**: Cache with Upsert Semantics
- **Result Store**: Persistent store for result events
  - Write result event to a fault-tolerant distributed store
Windows

- Low-latency storage layer for Windows
  - Lock-free in-memory Skip-lists to store window data
  - Bucketed Skip-list (batch neighboring events)
Caches

• Low-latency storage/indexing layer for Caches
  – Distributed in-memory Hash Table to store and manage cache data
  – Periodically refreshed from external data source using MVCC semantics
  – Optimized for O(1) key lookup access
  – Node Locality
  – Replication Factor, Partitionig
Persistent Event Store

- Event Store: A low-latency & reliable store
  - To persistently store result events
  - Query Engine continuously writes results to tables in this store
  - Can ingest high-velocity data
    - Micro-batch in certain cases
  - Serve interactive SQL queries from visualization engine
Query Engine
Query Engine Components

• Application:
  – Written in a SQL-like declarative language

• CQ (Continuous queries)
  – Part of an application
  – Filter, aggregate, search, join over Stream, Window, Cache, Event Store Tables
  – Java-based User-Defined Functions (UDF) for custom processing
  – Flexible integration with Machine Learning Libraries like h2o, Apache Spark etc.
Query Compilation

• Organic cost-based query optimizer & compiler
  – Performs rule-based SQL query rewrites
  – Join order for inner and outer joins
  – Generates run-time Java byte code for every distinct query
    • Code saved in repository to avoid expensive recompilation
  – Generates multiple plans for window-window joins
  – (Key based vs. scan based – Partitioned/non-partitioned)
Query Execution

- Continuous Query Execution Engine
  - K/V based data structure used to deliver **Window Snapshots** from Skip-List storage to Query Engine
  - Execution Schedule is a DAG of execution operators
  - Parallel and distributed execution
Recovery & Persisted Streams
Recovery

- Application Level
  - Application level asynchronous check-pointing
- Global fault-tolerance (Components)
- Replay from check-pointed state
- Exactly-Once-Processing
- Works across Applications using Persisted Streams
Persisted Streams

- Persist raw stream data to stable storage
- Solves two major enterprise use cases
  - Non-replayable data sources
    - IoT data sources
  - Application de-coupling
    - Streaming Analytics platform spanning multiple business groups
- Striim supports Exactly-Once-Processing across applications without requiring developers to write custom code
Benchmarks
Gearpump Performance Benchmark

Results

- SOL Event Processing App
- 100 Byte Message
- 4 Nodes
- 32 cores per node
- Intel Xeon 2.9 GHz
- 64 GB of RAM

~18m per second on 4 Nodes

Near Linear Scalability

Figure 6: Gearpump Throughput Measurement
Yahoo Cloud Serving Benchmark

Results

Node - 1 Intel Xeon CPU (4 cores with hyper-threading)
32 GB memory

Striim cache to store the ad campaign information and perform the joins inline

2.8 Million Events/sec
On 10 Nodes (w Kafka)

Figure 7: YCSB Throughput Measurement
The top-\(k\) customers per geographic region based on the total number of bytes delivered in that minute

The output information is used in real-time to straggle the customer

The input data is collected as part of the CDN Edge Devices and sent over to the Striim platform through a Kafka stream

- Gearpump hw
- The event type consists of the following fields: customer_code, timestamp_of_data, geographic_region, ip_address and bytes

**Figure 8: Top20 Throughput Measurement**

```sql
CREATE CQ Top20
INSERT INTO Top20Stream
SELECT w.subcustomer_id, w.geographic_region, sum(w.bytes) as sbytes
FROM RecWindow1min w
GROUP BY w.subcustomer_id,w.geographic_region
ORDER BY sbytes DESC
LIMIT 20;
```
6 Node Cluster - Persisted Streams

200 PARTITIONS WITH -> 2, 3, 4 AND 6 BROKERS

Throughput in Events/sec for a single Source

~ 4 Million events/sec
700k*6
Configuration Details – c3.xlarge EC2 Instance

- 6 EC2 Nodes
- 6 Data Sources

Intel Xeon E5-2680 v2 (Ivy Bridge) Processors

<table>
<thead>
<tr>
<th>Model</th>
<th>vCPU</th>
<th>Mem (GiB)</th>
<th>SSD Storage (GB)</th>
</tr>
</thead>
<tbody>
<tr>
<td>c3.xlarge</td>
<td>4</td>
<td>7.5</td>
<td>2 x 40</td>
</tr>
</tbody>
</table>

*No Tuning - OOTB EC2 Instance
Summary - Integrated Streaming Platform

Design Flows

Analyze

Deploy

Visualize

Monitor

© 2017 Striim, Inc. All rights reserved.
Thank You

Alok Pareek