

## Stream Ingestion, Processing and Analytics Using In-Memory Computing

Denis Magda Apache Ignite PMC Chair GridGain Product Management

© 2018 GridGain Systems, Inc.





## Agenda

- Streaming Analytics
- Ignite Native Streaming APIs
- Ignite Streamers Ecosystem
- Ignite and Spark: Better Together
- GridGain Kafka Connector Integration
- Q & A



## Streaming Analytics

© 2018 GridGain Systems, Inc.







## Ignite Native Streaming APIs

© 2018 GridGain Systems, Inc.





© 2018 GridGain Systems, Inc.



#### **Ignite Streaming APIs**

- Ignite Data Streamer
  - Partitioning of streams of data
  - Ignite streaming powerhouse
- Stream Receivers and Transformers
  - Last-call data updates and analysis
- Continuous Queries
  - Data updates notifications and postprocessing

| lgnite Data<br>Streamer |  |
|-------------------------|--|
|                         |  |





## Ignite Streamers Ecosystem

© 2018 GridGain Systems, Inc.



#### **Ignite Streamers Ecosystem**

- Various Streaming Technologies
  - Kafka, Spark, Flink, Storm, etc.
  - Process, Enrich and push to Ignite
- Ignite as a final store for streaming data
  - Streaming Analytics













## Ignite and Spark: Better Together



### **Comparing Ignite and Spark**





Distributed memory-centric **database**  Ingests data from HDFS or another ullet

- Fully fledged compute platform: SQL, transactions, key-value, collocated processing, ML/DL
- Streaming and compute engine

storage

OLAP and OLTP

MR payloads



#### Inclined towards OLAP and focused on



#### **Ignite and Spark Together**

#### Ignite is a memory-centric store for Spark

- No data movement from Ignite to Spark
  - In-place query execution
- Boost DataFrame and SQL performance
- Share state and data among Spark jobs
- Faster data and streaming analytics





### **Ignite and Spark Integration**





# GridGain Kafka Connector Integration



#### **GridGain Confluent Integration**





## **GridGain Connector Advantages over Ignite Kafka Integration**

- Advanced parallelism
- Exactly Once processing semantics
- Single connector per multiple caches/topics
- Filtering of source and sink connectors
- Enterprise Ready
  - Supported by GridGain, certified by Confluent.

# GridGain

# Econfluent



# Thank You!!!

Thank you for joining us. Follow the conversation. https://ignite.apache.org

@denismagda#apacheignite#gridgain

