Accelerating Intelligence: Bigstream Hyper-Acceleration

Roop Ganguly
Hardware Accelerators Break Through the Processing Wall

- **Microsoft Azure**: On track to millions of servers w/FPGAs
- **Amazon Web Services**: Introduces FPGA-powered server instances
- **Google Cloud Platform**: TPUs help avoid cost of 12-15 data centers
- **Intel FPGA**: $16.7B acquisition of Altera
- **Xilinx**: $15 in datacenter revenues
- **NVIDIA**: 193% growth in datacenter segment
Inhibitor: Programming Model Gap for Hardware Accelerators

**BIG DATA PLATFORMS**

- presto
- Kafka
- Spark
- MySQL
- TensorFlow
- Elastic

**Data Science Programming Model**

**Programming Model Gap**

**Acceleration Programming Model**

**Focus on Macroarchitecture**

**Skills Gap**

**Focus on Microarchitecture**

- CPU
- GPU
- FPGA

Data Scientists & Developers

Performance Engineering
Introducing: Bigstream Hyper-acceleration Layer

Zero code change

Cross platform
Intelligent, automatic computation slicing
Cross acceleration hardware

3X to 30X acceleration

Not STAC Benchmarks
Big Data Platform Architecture with Bigstream
Spark.sql("SELECT name FROM People WHERE age BETWEEN 25 AND 40")

Bigstream Hyper-acceleration with Intel Multi-core CPUs and FPGAs

- Step 1: SQL Converted to Dataflow
- Step 2: Dataflow Slicing
- Step 3: Mapping of Execution

Acceleration occurs without code changes

Spark.sql(“SELECT name FROM People WHERE age BETWEEN 25 AND 40”)
Bigstream + Micron AC-520: 4.5X Faster than Standard Spark

Spark vs Spark/Micron AC520

Benchmark System: Xeon 6 core server, Apache Spark 2.1, results with and without Micron AC-520

Not STAC Benchmarks
Thank You,
Please tick the Bigstream box
About Bigstream

Established: July 2015
First Product Shipped: March 2017
Headquarters: Mountain View, CA

Based on patented technology
Core Competencies:
• Advanced compiler technology
• FPGA/GPU programming and strategy
• Apache Spark and Big Data platforms

Key Insight:
“Any big data/machine learning workload can be hyper-accelerated”

– M Lavasani, CEO
Customer Case: Hedge Fund Pricing Models

Challenge: Iterate faster on investment pricing models

Diverse Data Sources (Structured & Unstructured)

Complex Queries

Multiple Stages with UDFs

• Runtime Environment
  • 500+ node cluster
  • Amazon EMR
  • R3.2XLarge instances
  • Spark 2.X

• Business Goals
  • Intra-day price modeling
  • 3+ iterations/day
Customer Engagement

- Technical Review
  - Requirement analysis

- Deploy and xRay*
  - Run installation script
  - Assess accelerator potential

- POC
  - Acceleration Report
  - Recommendation Report

- Production Support
  - Incremental performance improvements

* xRay is a performance profiling tool built on the Bigstream Hyper-acceleration platform
The End of Moore’s Law

- Machine Learning
- Natural Language Processing
- AI/Deep Learning
- Behavioral Analytics
- Risk Management
- Security Analytics
Bigstream Spark Architecture
Spark + Bigstream Hyper-acceleration

**Input**
- Data Sources
  - Cloud
  - Enterprise Systems
  - Streaming Data
- Ingest
- (de)compress
- Parse

**Process**
- Transform, Cleanse, Enrich, Analytics
- Datasets, Dataframes
- User Defined Functions (UDFs)
- Batch/Stream Processing

**Output**
- Reporting
  - JSON
  - CSV
  - Parquet
- Dashboard
  - Qlik
  - Tableau
  - MicroStrategy

**Computing Resources**
- CPU
- GPU
- FPGA
Sentiment Analysis Pipeline

Different data sources
- Region A
- Region B
- Region C
- Region D
- Region E

KAFKA
User/transaction data

Tens of servers

Distributed messaging system

Hundreds of servers

Distributed computation system

Spark Streaming
Spark Streaming
Spark Streaming
Spark Streaming
Spark Streaming
Spark Streaming
Spark Streaming

Spark ML

Decisions engine
Acceleration Value Propositions

- **Faster Intelligence, Time To Insight**
  - Improved latencies for algorithmic trading
  - Real-time risk assessment at big data scale

- **Maximize Infrastructure ROI**
  - Free up computing resources
  - Reduce project backlog

- **Quant/Data Science Productivity**
  - Iterate faster on simulations and models
  - Include fast data (e.g., tick data) in models

- **Simplify Scalability**
  - Avoid forklift upgrades
  - Minimize floor space, power, HVAC
Spark + Bigstream Hyper-acceleration

- **Input**
  - Ingest
  - (de)compress
  - Parse
  - **Data Sources**
    - Cloud
    - Enterprise Systems
    - Streaming Data

- **Process**
  - **Transform, Cleanse, Enrich, Analytics**
  - Datasets, Dataframes
  - User Defined Functions (UDFs)
  - SQL
  - Batch/Stream Processing
  - **CPU**, **GPU**, **FPGA**

- **Output**
  - Reporting
  - **JSON**, **XML**
  - **Parquet**
  - Dashboard
  - **Qlik**, **Tableau**
  - **MicroStrategy**

---

**Compare to “point solutions”**:  
- New languages/APIs  
- Single hardware/software platform  
- Query plan-only optimizations  
- I/O-only optimizations  
- Hand-optimized UDFs  
- Library calls
Architecture for an User Behavior Pipeline In-line Accelerator
Bigstream Technology

- **Platform Agnostic Layer**: Big Data and Machine Learning Platforms
- **Hardware Agnostic Dataflow**: CPU, GPU, FPGA
- **In-line Acceleration**: Perform computation on the data while it is on the move
- **Dynamic Partitioning**: Dynamic partitioning of workloads between software & different accelerators
- **Faster Compilation**: Logical to physical plan mapping
- **Dynamic Refinement**: Variety of parallelism and locality optimizations
Spark Code for an User Behavior Pipeline

```scala
val lines = 
  KafkaUtils.createDirectStream
  [String, String, StringDecoder, StringDecoder]
  (ssc, kafkaParams.toMap, topicsSet).map(_._2)

lines.foreachRDD((jsonRDD: RDD[String]) => {
  val sparkSession = 
  SparkSessionSingleton.getInstance(sparkConf)
  val df = sparkSession.read.schema(schema).json(jsonRDD)

  df.filter("country is not null and price is not null")
  .filter(df("country").equalTo(30))
  .groupBy("country").agg(count("price"), min("price"),
                           max("price")).show()}
```
Expose simplistic, composable acceleration primitives to developers.
Gorilla-Generated Code for a User Behavior Pipeline

```plaintext
val depack = Engine("depacketizer.c")
val kafka_token = Engine("kafka_tokenizer.c")
val json_deserializer = Engine("json_deserializer.c")
val tkCntrl = Engine("tokenMemController.c")
val data = spMem(height = 64, width = 256)
val membuff = Offload(tkCntrl, data, "data")
val flit2mem = Engine("FM.c")
val mem2flit = Engine("MF.c")
val group_by = Engine("group_by_state.c")
val aggEngine = Engine("aggregate.c")
val aggMem = spMem(height=20, width = 192)
val aggregate = Offload(aggEngine, aggMem, "aggMem")
val array_serializer = Engine("array_serializer.c")
val fm_groupby = Chain(flit2mem, group_by)
val fm_groupby_mem = Offload(fm_groupby, membuff, "gBTMem0")
val fm_groupby_agg = Chain(fm_groupby_mem, aggregate)
val fm_groupby_agg_mem = Offload(fm_groupby_add, membuff, "gBTMem1")
val aggregate_nf = Chain(fm_groupby_agg_mem, mem2flit)
val aggregate_part = Offload(aggregate_nf, membuff, "gBTMem2")
val prepare_data = Chain(depack, kafka_token, json_deserializer)
val result = Chain(prepare_data, aggregate_part, array_serializer)
```

Packet processing and de-serialization parts

Token memory controller for passing variable length tokens

Two additional engines for storing streaming data into memory and fetch streaming data from memory

Data filtering, aggregation, and serialization into an array in program memory

Connecting the above components by "Chain" and "Offload" operations
User Behavior Pipeline Accelerator after Place & Route
Announcement – Bigstream on AWS EMR

Bigstream Hyper-Acceleration Layer
Sold by: Bigstream

14-DAY FREE TRIAL - Bigstream hyper-acceleration software provides a performance boost of up to 5x for your Spark SQL applications without changing a single line of code. Using native compilation, vectorization, locality optimization and custom data connectors, Bigstream provides faster time-to-insight and cost avoidance of up to 50% of your EMR infrastructure. The installation of our software is incredibly easy. After you have registered your Bigstream service we will send you a link that you can paste into the bootstrap field of your EMR cluster provisioning screen. And that’s it... Read more

Customer Rating

Customer Reviews)

Delivery Method
Software as a Service (SaaS) Subscriptions (Read more)

Support
See details below

Highlights
- Seamless hyper-acceleration: automatic performance gains without changing a single line of code
- Seamless installation: subscribe to the Bigstream service once, and anytime you want to accelerate your EMR cluster, simply add the Bigstream bootstrap URL to turn on hyper-acceleration
- If after the Free Trial Period, you want to continue using Bigstream, you are all set. Your usage fees will be added to your monthly AWS bill.

Product Description
14-DAY FREE TRIAL - Bigstream hyper-acceleration software provides a performance boost of up to 5x for your Spark SQL applications without changing a single line of code. Using native compilation, vectorization, locality optimization and custom data connectors, Bigstream provides faster time-to-insight and cost avoidance of up to 50% of your EMR infrastructure.

The installation of our software is incredibly easy. After you have registered your Bigstream service we will send you a link that you can paste into the bootstrap field of your EMR cluster provisioning screen. And that’s it! You can now run your job with state of the art hyper-acceleration.

Pricing Details

Software Fees

<table>
<thead>
<tr>
<th>Users</th>
<th>Cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>All small, medium, large, and xlarge instances</td>
<td>$0.100 / host / hour</td>
</tr>
<tr>
<td>all 2xlarge instances</td>
<td>$0.300 / host / hour</td>
</tr>
<tr>
<td>all 4xlarge instances</td>
<td>$0.600 / host / hour</td>
</tr>
<tr>
<td>all 8xlarge and 10xlarge instances</td>
<td>$1.200 / host / hour</td>
</tr>
<tr>
<td>all 16xlarge instances</td>
<td>$2.400 / host / hour</td>
</tr>
<tr>
<td>all 32xlarge instances</td>
<td>$4.800 / host / hour</td>
</tr>
</tbody>
</table>

Note: This software is priced along a consumption dimension. Your bill will be determined by the number of hosts you use per hour.

Recent Product Reviews
Create Your Own Review
Bigstream ON EMR

Add the Bigstream bootstrap URL and your cluster has hyper-acceleration

Setting the bootstrap script
Cross Platform
Cross Hardware
Intelligent, automatic computation routing

Dataflow Adaptation Layer
Bigstream Dataflow
Bigstream Hypervisor

OPEN-SOURCE BIG DATA PLATFORMS
presto
kafka
Spark
MySQL
TensorFlow
elastic

CUSTOM BIG DATA PLATFORMS
Custom libraries/Dataflow primitives

3X to 30X acceleration

Future Bigstream Customized Hyper Acceleration API

HYPER-ACCELERATION LAYER
OPEN-SOURCE BIG DATA PLATFORMS
CUSTOM BIG DATA PLATFORMS

Future Bigstream Customized Hyper Acceleration API

HYPER-ACCELERATION LAYER