

Elastic Streaming

Spark Streaming + Dynamic Provisioning + Dynamic Allocation

Neelesh Shastry, Architect

Shaun Klopfenstein, CTO



The image features a stylized globe in the background, constructed from a network of glowing blue and purple dots connected by thin lines, creating a mesh-like structure. The globe is centered and occupies most of the frame. Overlaid on the center of the globe is the text "The Vision" in a large, white, sans-serif font. The text is slightly shadowed to stand out against the complex background.

The Vision



Requirements

Business Requirements

- Near real-time activity processing
- Billions activities per customer per day
- Improve cost efficiency of operations while scaling up
- Global enterprise grade security and governance

SAAS Requirements

- Customers are added and removed
- Fairness and throttling *per customer*
- Strict sequential event processing for some applications
- Temporarily suspend a customer, when errors occur

A top-down view of a person's hands kneading a large ball of dough on a dark, floured wooden surface. To the left, there is a small glass pitcher of milk, a bowl of butter, and a whisk. A rolling pin is visible on the right. The scene is dimly lit with a blueish tint. A semi-transparent dark rectangle is overlaid in the center, containing the text "Technology Selection".

Technology Selection

Use Cases

- React to activities
 - Send an email when someone visits a web page
 - Change the score when someone fills a form
- Replicate data
 - Build Solr Indexes, near real-time
 - Update DataXChange – an internal lead cache
 - Sync to/from CRM Systems
- Analytics
 - Incrementally update email reports
 - Enrich activities and feed to Druid for advanced email/web reports

Why Spark Streaming?

- Micro-batching provides sink-side efficiencies
- Great integration with Kafka
- No strict real time processing requirements
- Great community, industry adoption

Challenges with Spark + Kafka

- No way to add/remove topics on the fly
- No out of the box support for sequencing RDDs
- No support for turning off topics under errors
- Does not play well with scaling Kafka partitions up/down, when ordering is required

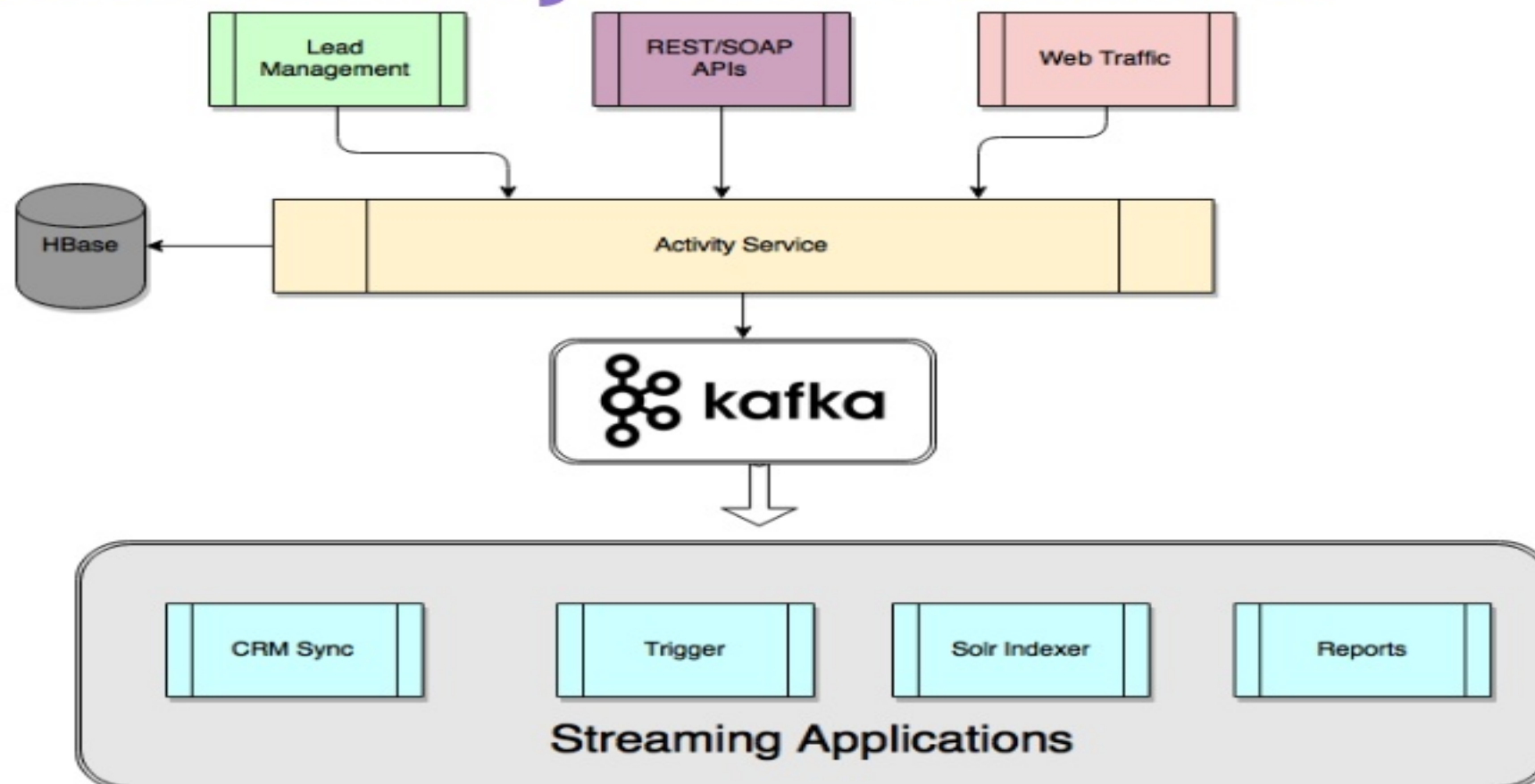
Challenges - Stragglers

- A batch can't complete until the slowest operation finishes
- Many of our batches include slow operations
 - Sometimes don't complete within the batch time
- Batches are multitenant
 - one customer's operation can delay processing for other customers in the same batch
- **Severe impact on utilization & batch delay**



Architecture & Design

Marketo Activity Architecture



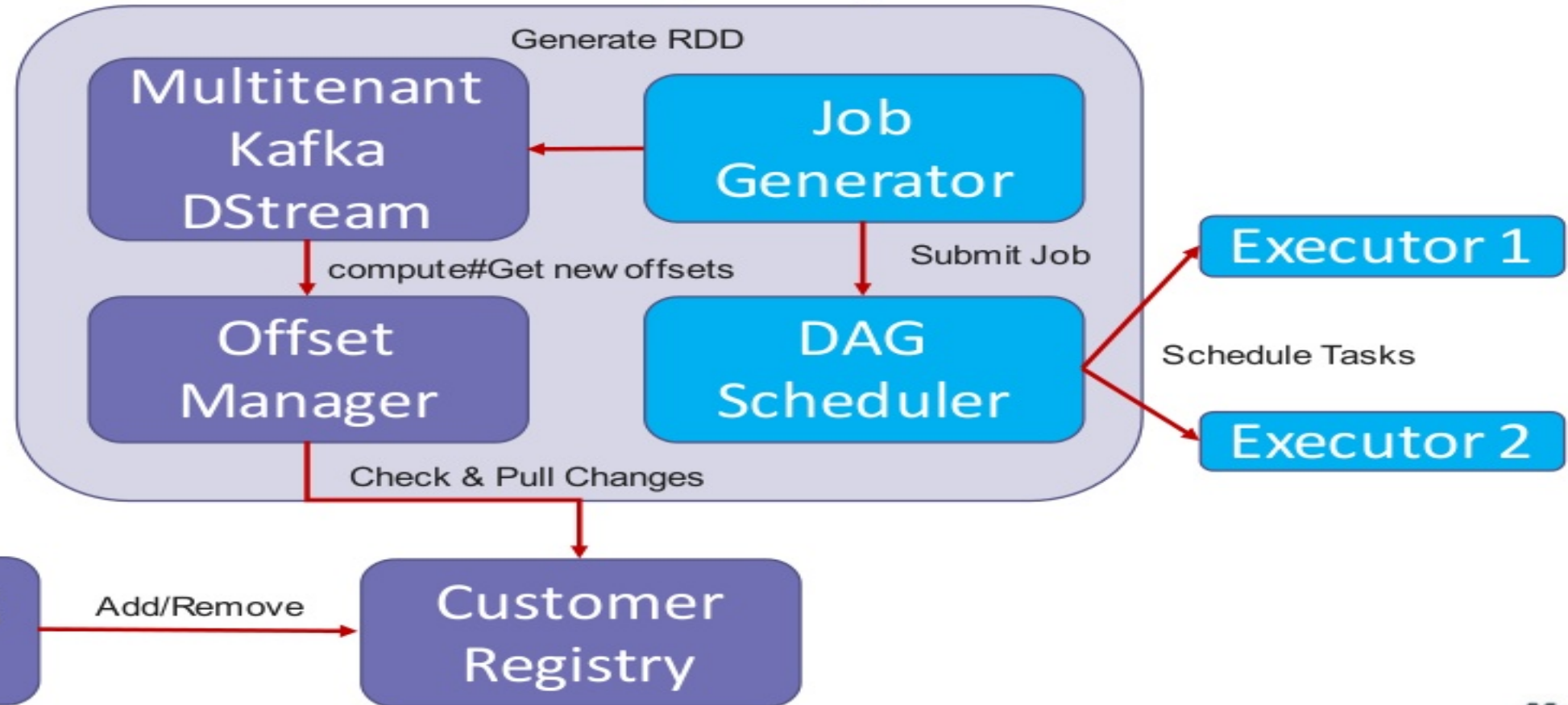
Kafka Topics Organization

- One topic per use case, data from all customers
 - Easy to manage
 - A single customer can create backlogs for others during activity storms
 - Fairness/throttling is hard to implement
- One topic per use case, per customer
 - Storms are isolated to the customer
 - Fairness/throttling is easy to control, by tweaking the topic
 - Pressure on Kafka ZK – so far not a problem



Solutions

Dynamic provisioning capacity



Marketo Offset Manager

- Tracks multitenancy
- Streaming Jobs process data for many customers
 - Accessing multiple Kafka topics and partitions
- Adds new topics
- Remove/Deactivate/Suspend topics

Multitenant DStream

- Enables efficient multitenant RDDs
- Controlled sequencing of RDDs
- Coalesce Kafka partitions
 - Bin-packing for efficiency
- Maintains partition lineage for offset management

Provisioning

- Manages allocating customers to a spark streaming application
- round robin + resource affinity
- Enables rebalancing of customers across spark streaming jobs
- Oozie based framework

Dynamic Resource Allocation

- SPARK-12133
 - Goal – “make processing time infinitely close to duration”
 - Assumes tasks are roughly similar
- Stragglers throw this goal off
- What we really want :
 - DRA + Safe concurrent job execution

Results so far

- ~ 10 different use cases
- > 100 Spark Executors
- >1000 Kafka Partitions
- Processing latencies < 5s (99th %)
- Rolled out to ~20% customers

A man and a woman are working in a modern office. The man is standing and looking at a smartphone, while the woman is sitting at a desk, typing on a keyboard. The image has a purple overlay and a network of white lines connecting dots, suggesting a digital or technological theme. The text "Future Work" is centered in a white, bold, sans-serif font.

Future Work

Application Scheduling

- Scheduling within an application to handle stragglers
- `spark.streaming.concurrentJobs`
- Exploring scheduler pools
- Changes to Streaming Job Scheduler, to execute multiple RDDs safely

Scaling Up Kafka Partitions

- Our customers grow in size over a period of time
- Ordering requirements mean we cannot alter topic on the fly
- Coordination required on both producer & consumer fronts
- Enhance provisioner to manage partition up/down scaling

Move to 2.x and Open Source!



We're Hiring!
[Http://Marketo.Jobs](http://Marketo.Jobs)
Q & A



A man with light brown hair, wearing a blue button-down shirt, is sitting at a desk and looking down at a laptop. The background is a blurred office environment with shelves, papers, and other people. The entire image is covered with a semi-transparent purple overlay. Overlaid on the purple background are white, thin, interconnected lines forming a network or web-like structure. On the left side, the text 'Q & A' is written in a white, sans-serif font. In the bottom right corner, the Marketo logo is visible, consisting of a stylized bar chart icon above the word 'Marketo' in a bold, italicized sans-serif font.

Q & A



Architecture Requirements

- Maximize utilization of hardware
- Multitenancy support with fairness
- Encryption, Authorization & Authentication
- Applications must scale horizontally



Deploying It

A woman with long, wavy red hair is looking down at a white smartphone she is holding in her hands. She is wearing a light-colored, textured cardigan over a white top. The background is a blurred office environment with other people working at desks. The entire image has a purple tint and is overlaid with a network of white lines and dots, suggesting a digital or technological theme.

Running It