guavus
Go Decisively

BUILDING BIG DATA OPERATIONAL INTELLIGENCE PLATFORM WITH APACHE SPARK

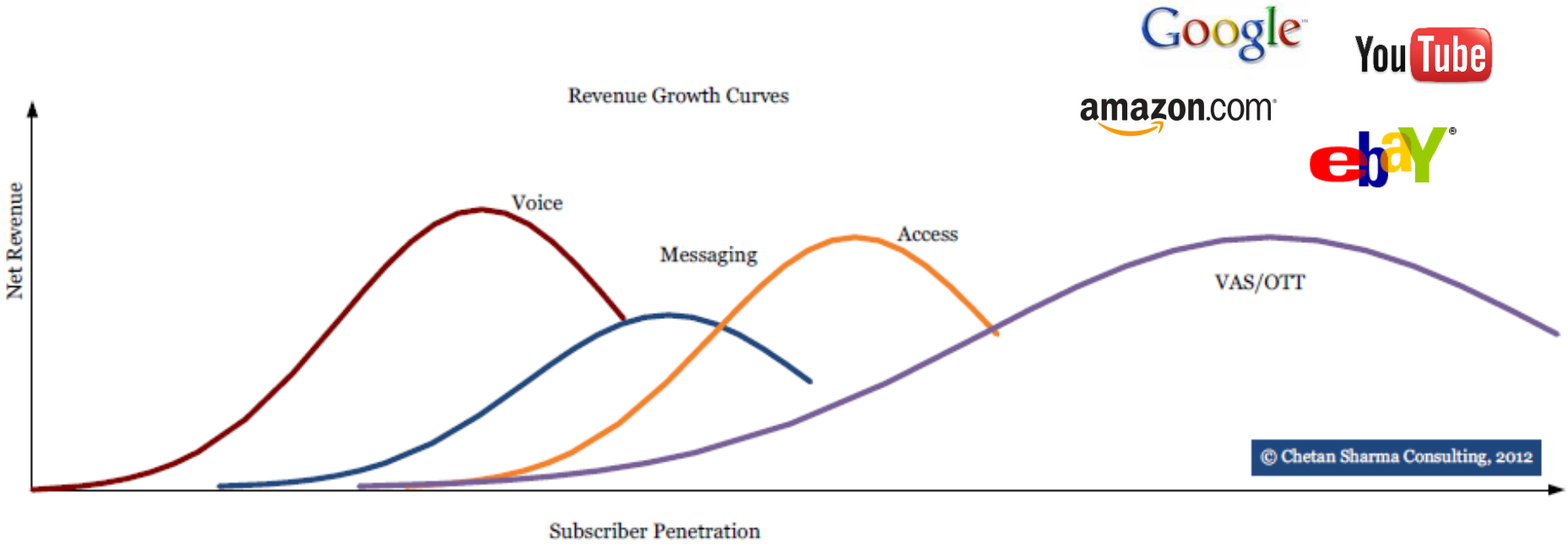
Eric Carr (VP Core Systems Group)

Spark Summit 2014

Communication Service Providers & Big Data Analytics

Market & Technology Imperatives

Industry Context for Communication Service Providers (CSPs)

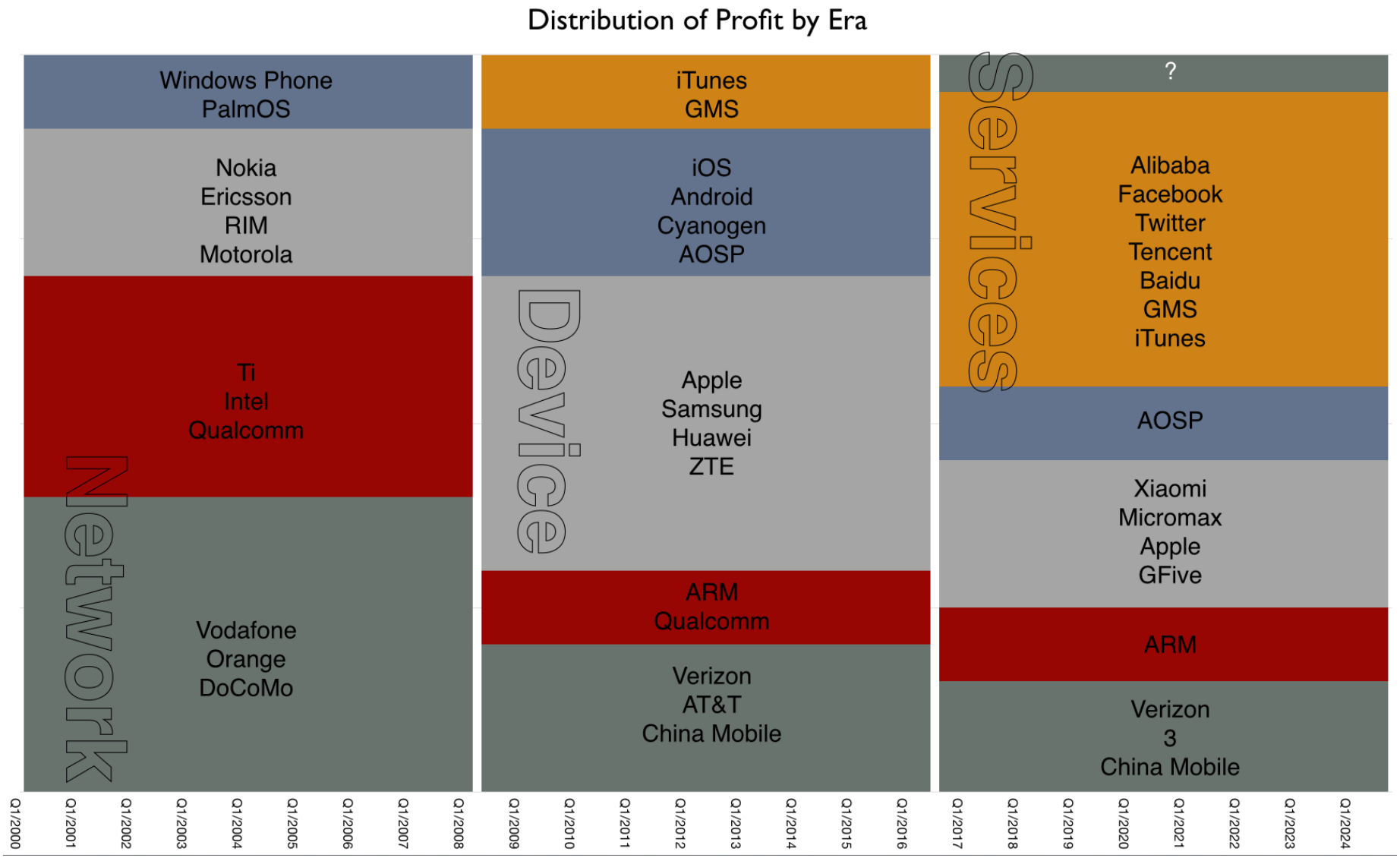


Big Data is at the heart of two core strategies for CSPs:

- Improve current revenue sources through greater operational efficiencies
- Create new revenue source with the 4th wave

Source: Chetan Consulting

CSPs - Industry Value Chain Shift



Source: asymco

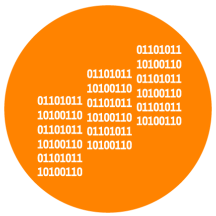
CSPs - A High Bar for Operational Intelligence

Exponential Data Growth

Petabytes of data per day; billions of records per day

Distributed Network

Dozens of locations for capturing data, scattered around a vast territory



Data Diversity

Hundreds of sources from different equipment types and vendors

Timely Insights

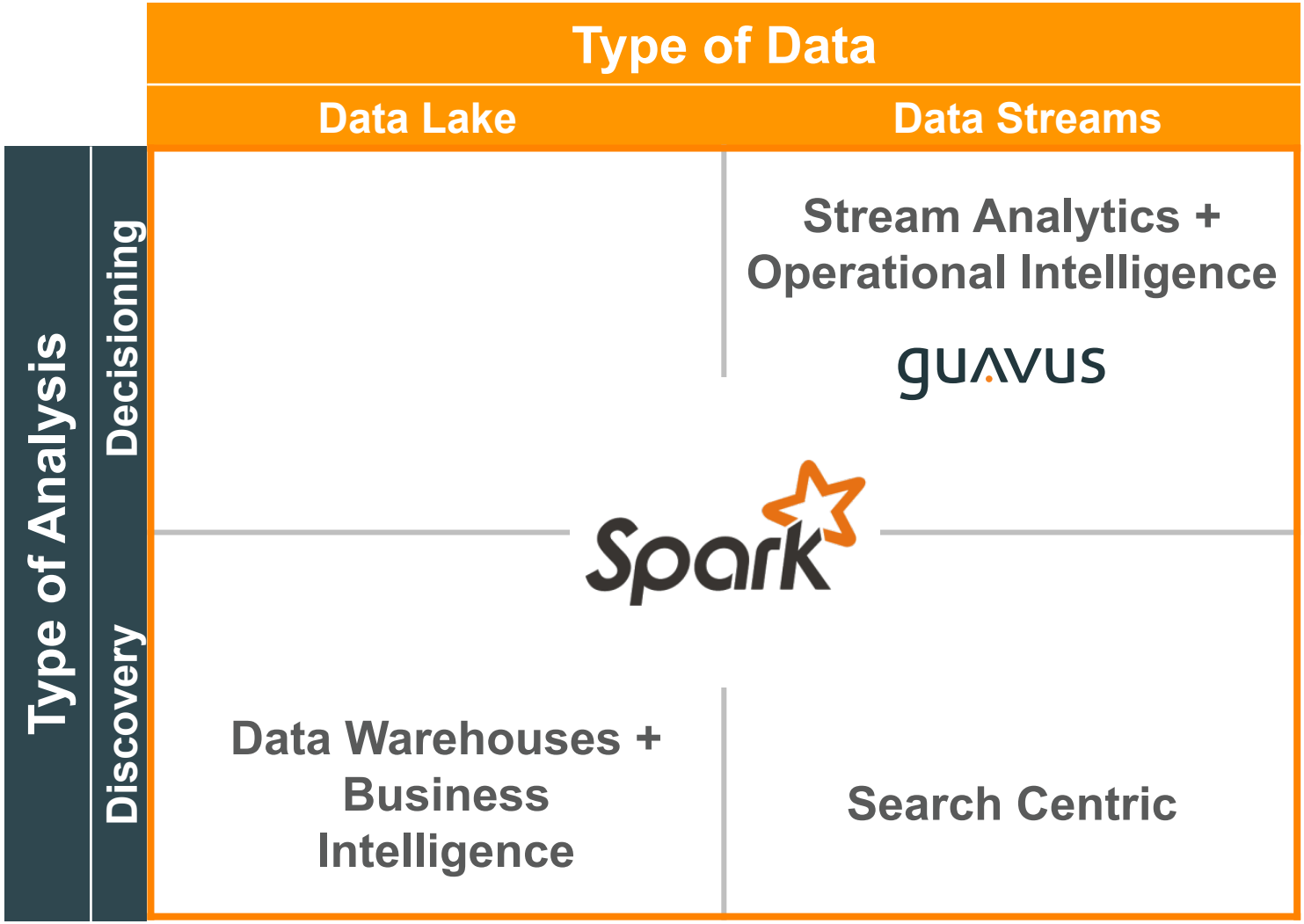
Automated reactions triggered in seconds

High Availability

No data loss; no down time

CSPs require solutions engineered to meet very stringent requirements

Different Platforms target Different Questions



Streaming Analytics & Machine Learning to Action

Driving Streaming Analytics to Action

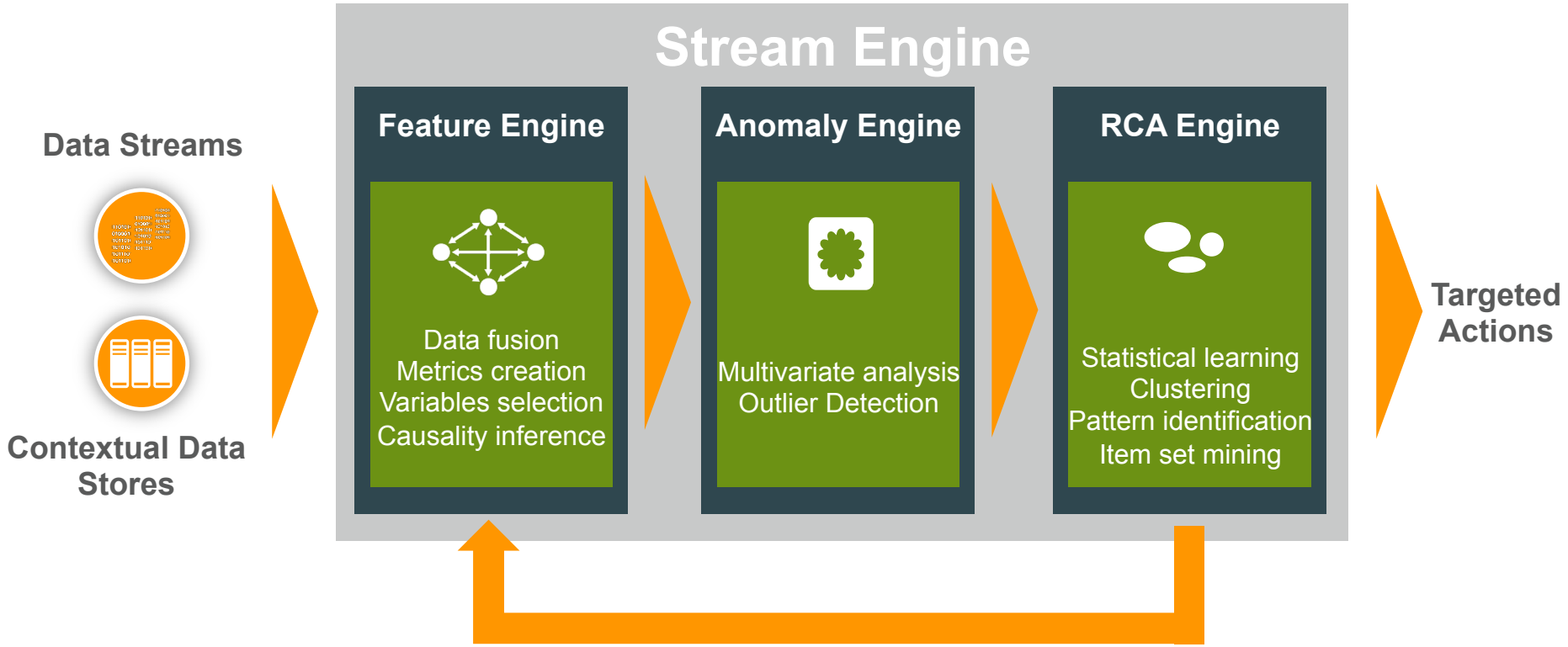
Network Flow Analytics	Usage Awareness	Operational Interactions	Real-Time Actions
	Layer 7 Visibility	Policy Profile Triggers	Small Cell / RAN / Backhaul Differentiation
NetFlow, Routing Planning	Operational Intelligence	Care & Experience Mgmt	SON / SDN / Virtualization
	Content & CDN Analytics	New Service Creation & Monetization	Content Optimization

Reflex 1.0 Pipeline – Timely Cube Reporting

Reflex 1.5 Pipeline – Spark / Yarn

Reflex 2.0 Pipeline – Spark Streaming core

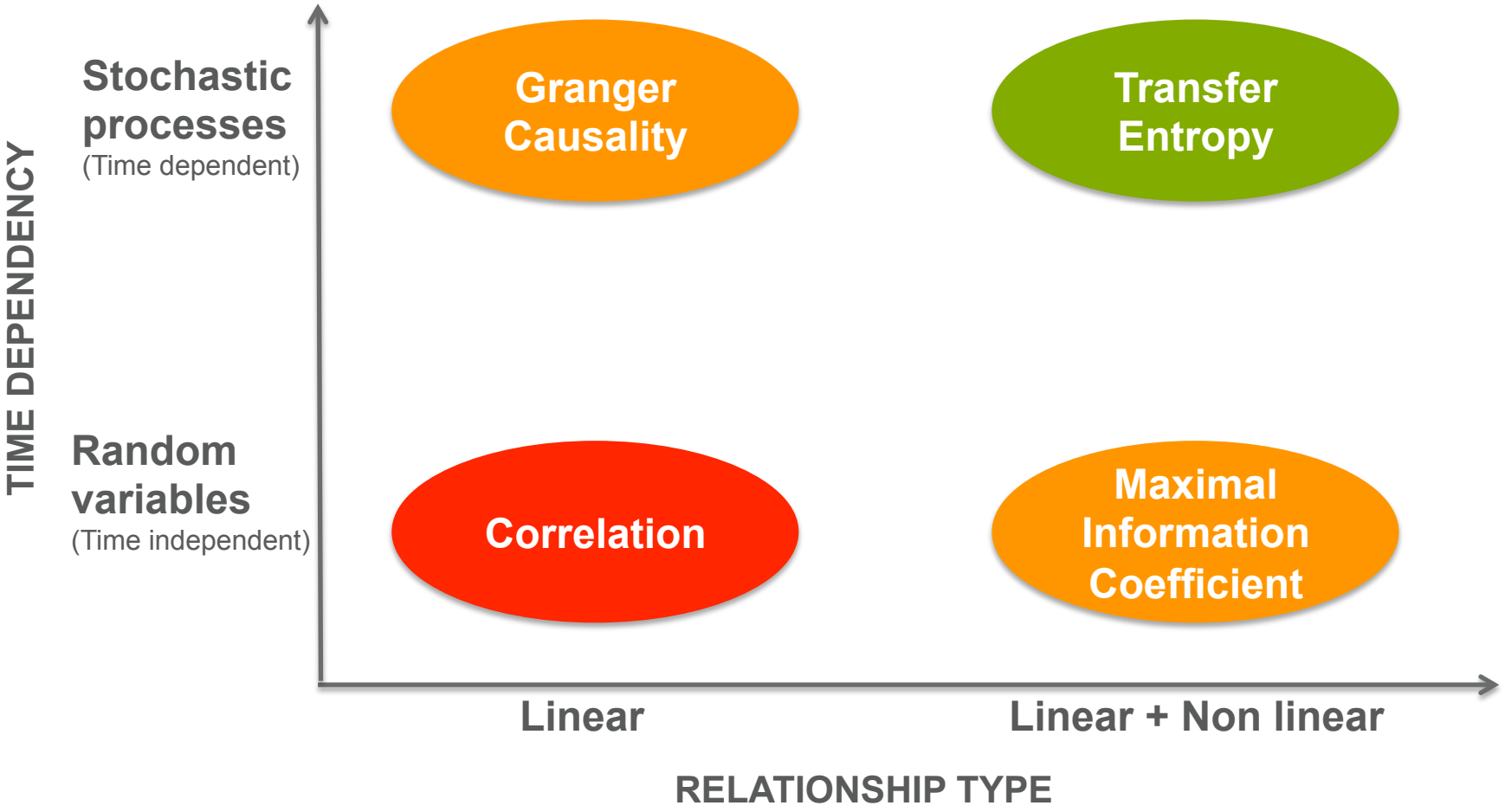
Stream Engine - Operational Intelligence Analytics



Optimized algorithmic support for common stream data processing & fusions

- Detect unusual events occurring in stream(s) of data. Once detected, isolate root cause
- Anomaly / outlier detection, commonalities / root cause forensics, prediction / forecasting, actions / alerts / notifications
- Record enrichment capabilities – e.g. URL categorization, device id, etc.

Example - State of the art causality analysis



Ranking of metrics:
1) Transfer Entropy
2) Maximal Information Coefficient, Granger Causality
3) Correlation

Example - Causality Techniques

Transfer entropy from a process *X* to another process *Y* is the amount of uncertainty reduced in future values of *Y* by knowing the past values of *X* given past values of *Y*.

PROS
Model free, information theory based approach
Most generic estimation of causality between two random processes

CONS
Challenging joint probability estimation
Large amount of data needed for calculation
Choice of time lags

Maximal Information Coefficient is a measure of the strength between two random variables based on mutual information. Methodology for empirical estimation based on maximizing the mutual information over a set of grids

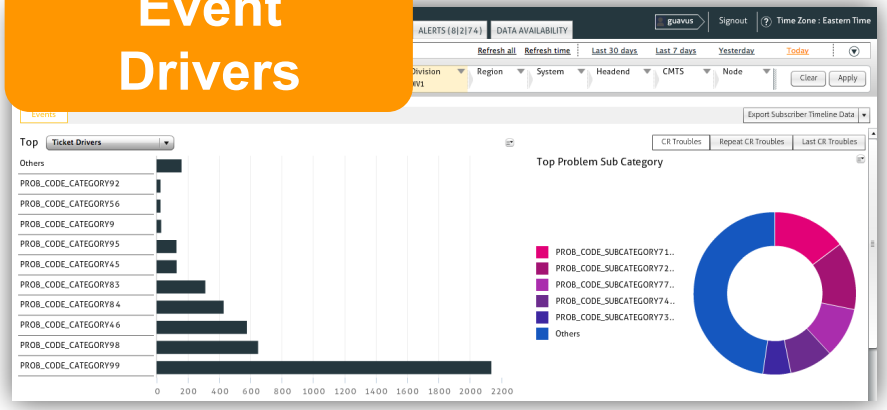
PROS
Model free, information theory based approach
Can find linear and non-linear relationships
Estimation possible with smaller dataset

CONS
No time information

Network Operations / Care Example

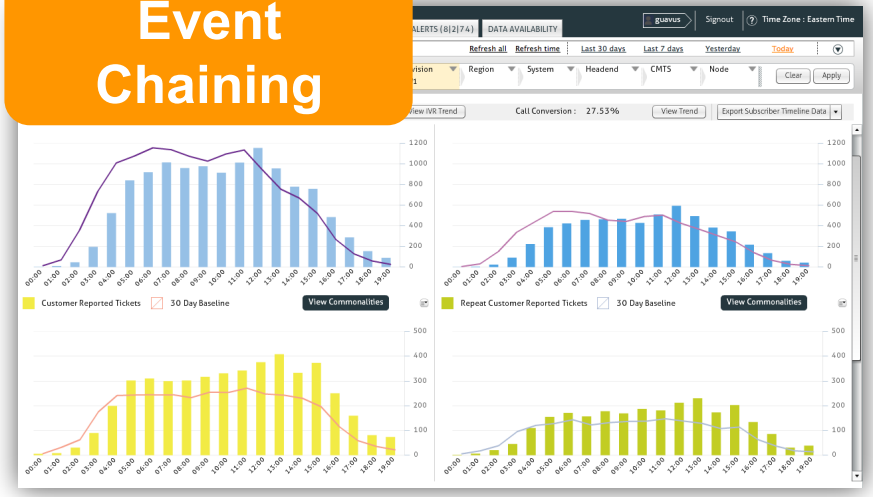
Identifying Commonalities, Anomalies, RCA

Event Drivers

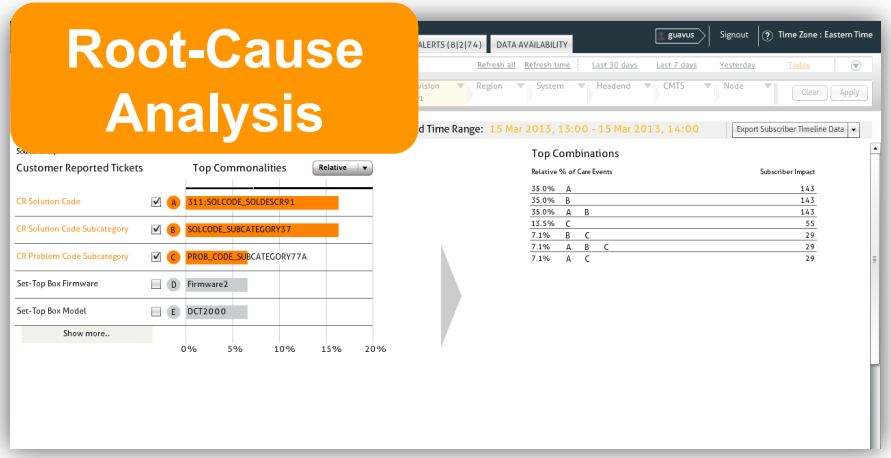


Anomaly Detection

Event Chaining



Root-Cause Analysis



BinStream Details

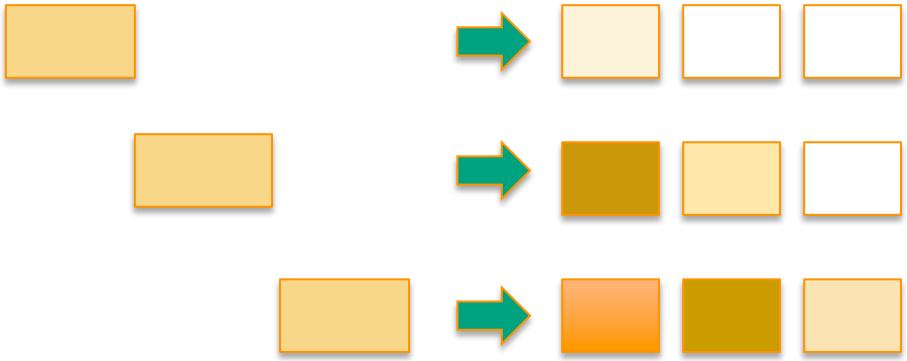
Use Case

- Use IP Traffic Records to calculate Bandwidth Usage as a Time Series (*continuously* …), *can't do* that based on the time the records are received by Spark.
  - In general for any record which has a timestamp, it important to analyze based on the time of event rather than the reception of the event *record*.

Challenges With

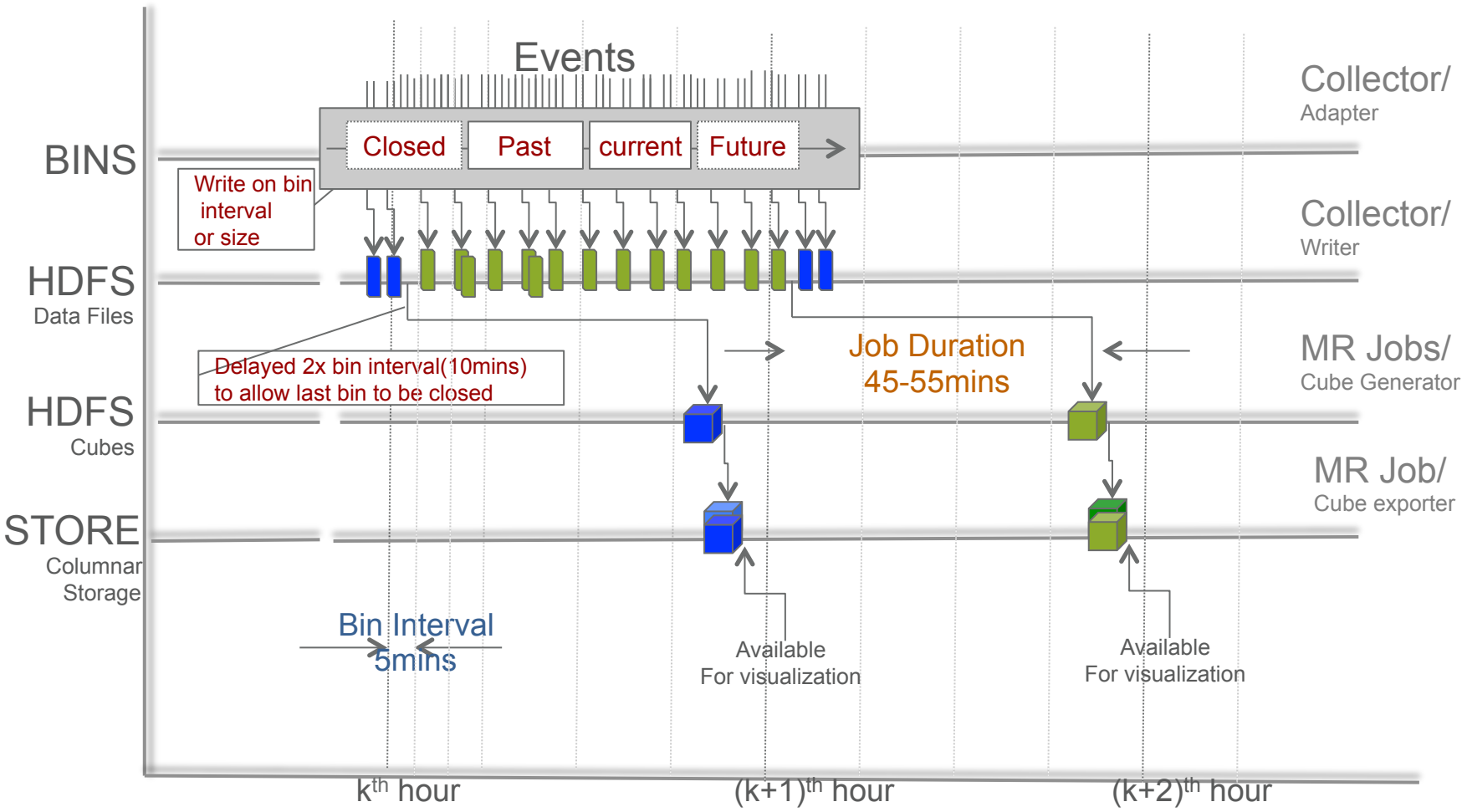
- For one dataset. Make the time stamp part of the key.

- For continuously streamed data sets.

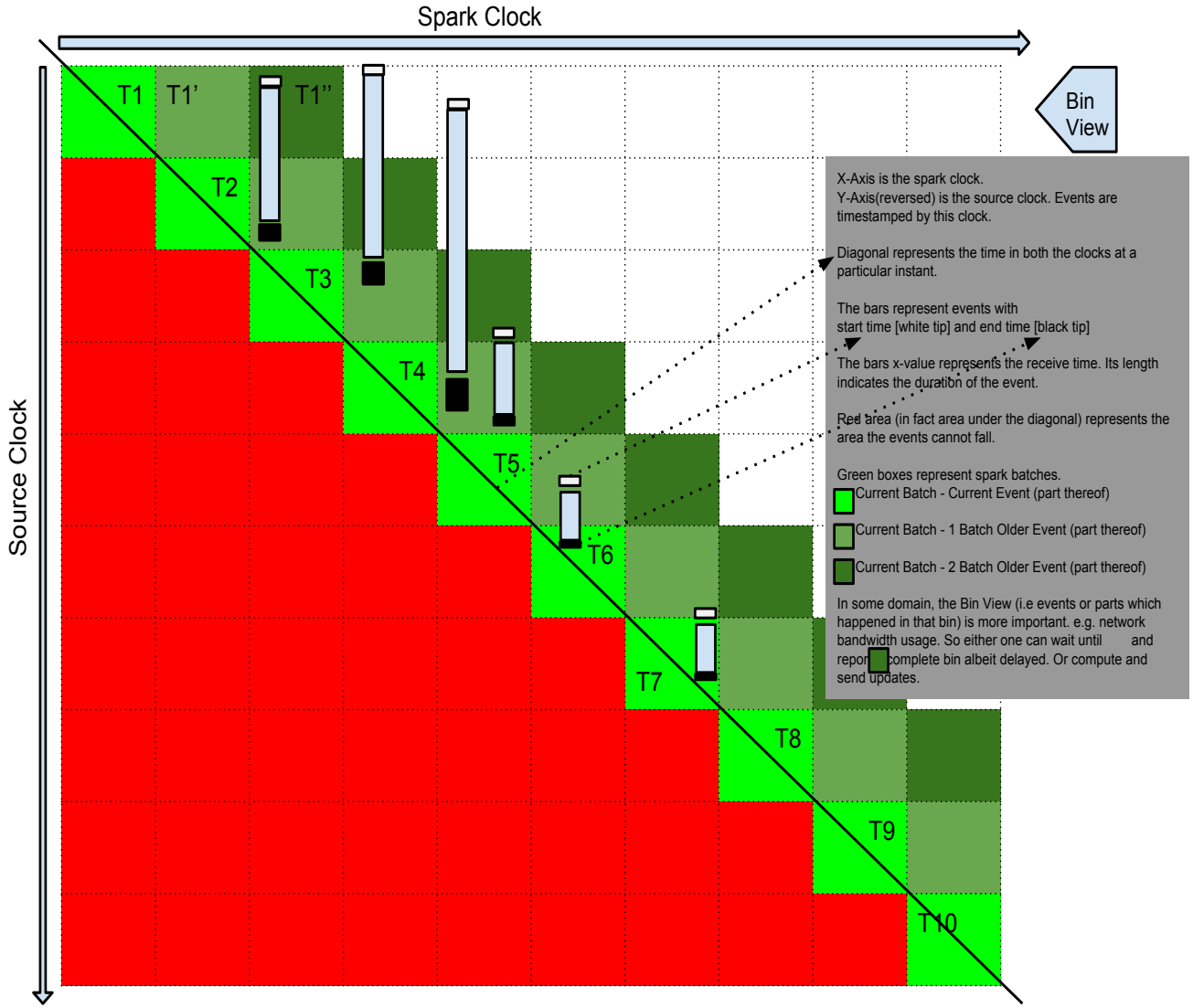


  - You do not know if you have received all data for a particular time slot. Caused by event delay, or event duration.
  - An event could span multiple time slots. Caused by event duration.

Data Processing – Timing (Map-Reduce world)



Proposed Binning & Proration Solution



Solution (cond.)

- The typical solutions are:
  - For event delay: Wait (Buffer).
  - For event duration: Prorate events across time slots.

- Introduce a concept of BinStream. An abstraction over the Dstream, which needs a
  - Function to extract the time fields from the records.
  - Function to prorate the records.

Note that this can be trivially achieved by using 'window' functionality, by having the batch equal to the time series interval and the window size equal to maximum possible delay.

Problems / Solutions

- window == wait & buffer.
  - *This has two issues.*
    - *A. Need memory for buffering.*
    - *B. Downstream needs to wait for the result (or any part of it)*

- *BinStream provides two additional options*
  - *Gets rid of delay for getting partial results, by sending regular latest snapshots for the old time slots. This does not solve the memory and increases the processing load.*
  - *If the client can handle partial results, i.e. if it can aggregate partial results, it can get updates to the old bins. This reduces the memory for the spark-streaming application.*

Limitations.

- The number of time series slots for which the updates can be generated is fixed, basically governed by the event delay characteristics.

guavus

Go Decisively

THANK YOU!