Fast Data
Delivered through Open Source vs. VoltDB

The world is rapidly becoming API- and automation-driven, heading towards the effective viability of artificial intelligence in an enterprise environment. In these environments, events and data stream in via a reliable, highly-distributed, fault tolerant message queue like Kafka. These events initiate an analyzing and decisioning cycle whose outcomes cascade into notifications to actors. This cascading can be perceived as an event itself.

For a highly automated environment where events trigger decisions which in turn trigger notifications, a Kafka + VoltDB setup allows a single platform to handle both single event processing — similar to Storm — or a contextual processing similar to Spark. The unification of the processing and storage elements into a single VoltDB platform eliminates any probability of ever increasing base latency, as is the case with stitching multiple open source technologies together.

Fast Data: Delivered through Open Source vs. VoltDB

Applications, Message Queues, Data Sources			Open Source	vs.	VoltDB	
			Kafka		**Kafka**	
Ingest			Storm, Flume, Sqoop		**VoltDB**	
Analyze		Decide				
Counters Aggregations Time series Statistics	Store results Query and recombine Fast serving	Per-event policy evaluations Responses (synchronous) Side-effects (asynchronous)	Storm & Serving Layer Spark & Serving Layer	Cassandra, HBase	SQL, Java for Analytics	Transactions/ ACID
Export & Pipeline			Hadoop, Message queues		Hadoop, Message queues	

VOLTDB

Real-World Example

Nimble Storage is a strategic IoT customer of VoltDB — their Infosight solution collects sensor data from over 7,000 arrays at 5-minute intervals, which equate to over 70m data points per array per day. Each event is inspected to determine the health of the drive and if a possible failure is detected, a triage goes into effect which is all automated and rules based.

VoltDB is working in conjunction with Kafka importer to load the data from the thousands of storage arrays and perform real-time analysis and issue detection while consuming newly generated insights from Vertica. Rather than someone watching a dashboard and waiting for something to flash red, the database procedures that ingest the data are doing the watching, so someone is alerted whenever a problem is beginning to manifest and in many cases the rules engines automatically remedies the issue.

The results are quite remarkable — 90% of their service tickets are opened without a customer calling, 85% are resolved without human intervention and they achieved 99.999% uptime!