signal fx

ELASTICSEARCH + SIGNALFX INTEGRATION

What is Elasticsearch?

Elasticsearch is an open-source search server based on Apache Lucene. It is highly distributed and designed for easy implementation, fast query against large data volumes, multi-tenant availability, and horizontal scale. Elasticsearch is used primarily as a NoSQL storage, indexing, and search utility for unstructured documents and can also serve as a log analysis tool as part of the Elastic Stack.

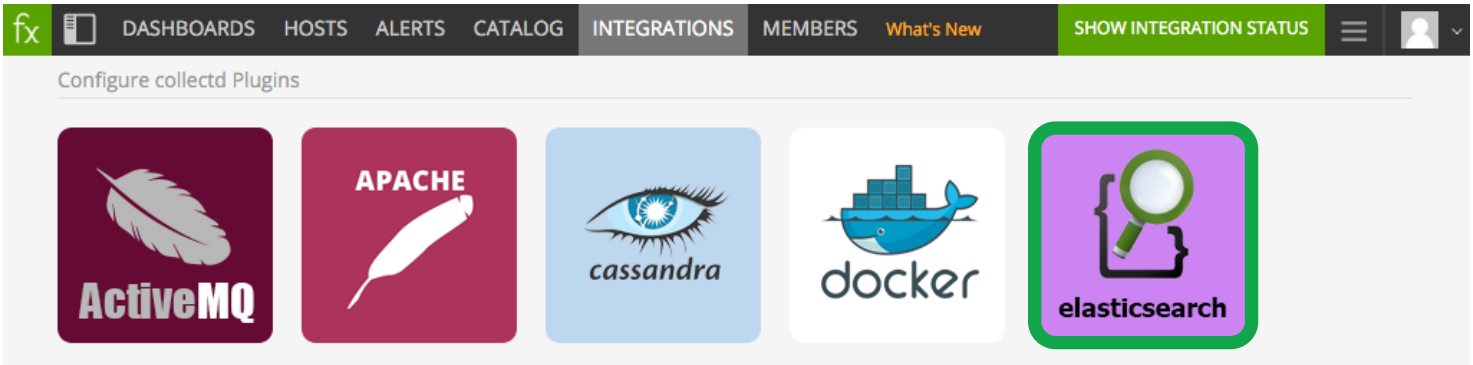
Sending Elasticsearch Metrics to SignalFx

Use collectd with the collectd-elasticsearch plugin to capture Elasticsearch metrics and track data by index and cluster. SignalFx provides built-in dashboards displaying the most useful metrics for running Elasticsearch in production at the node, cluster, and cross-cluster levels. You can also track metrics per index and add dimensions to the metadata to easily aggregate, filter, and group metrics by any properties you choose.

Monitoring Elasticsearch

From SignalFx's experience monitoring a large number of Elasticsearch nodes in production, we have learned that the first step in monitoring at scale is to find out whether an issue is cluster-wide or machine-specific. There are basically three sources of problems with which to contend — cluster, shard, and node — and keeping an eye on all three is the best way to manage availability and performance.

INFRASTRUCTURE (NOT JUST APP): When monitoring Elasticsearch, many performance issues come down to a noisy neighbor, or network I/O bugginess, or other problems with the Amazon machine image or virtual machine, underlying a given node on AWS. Although modern architecture relies on modern approaches to monitoring, it's still useful to know when less-modern problems arise.



FIELD DATA VS. DOC VALUES:
Large spikes in memory consumption can cause problems during garbage collection. A look at individual caches often reveals a problem with the field data cache, which is effective cluster-wide and per-node (on heap). In this case, move to using doc values (on disk).

SPIKES IN THREAD POOL REJECTIONS:
At some point, you'll need to re-shard your index, and it's not uncommon to run into thread pool rejections on the indexing queue as you batch re-index your documents shard-by-shard. To avoid extra load, randomly distribute the query order as you build the batch of documents to index, rather than asking for query results in shard order.

Elasticsearch Metrics

CPU Load	Disk IOPs
Memory Utilization	Search Requests / Sec
Heap Utilization	Indexing Requests / Sec
GC Time %	Merges / Sec
Avg Query Latency	File Descriptors
Requests/Sec	Segments
Doc Growth Rate %	Thread Pool Rejections
Top Indexes by Search Requests	Top Indexes by Indexing Requests
Top Indexes by Index Growth	Top Clusters by Search Requests
Top Clusters by Query Latency	Top Clusters by Index Growth
Deleted Docs %	Filter Cache Size
Active Merges	Field Data Cache Size
# Clusters	Top Indexes by Query Latency
# Nodes	Top Clusters by Indexing Requests
Nodes / Cluster	

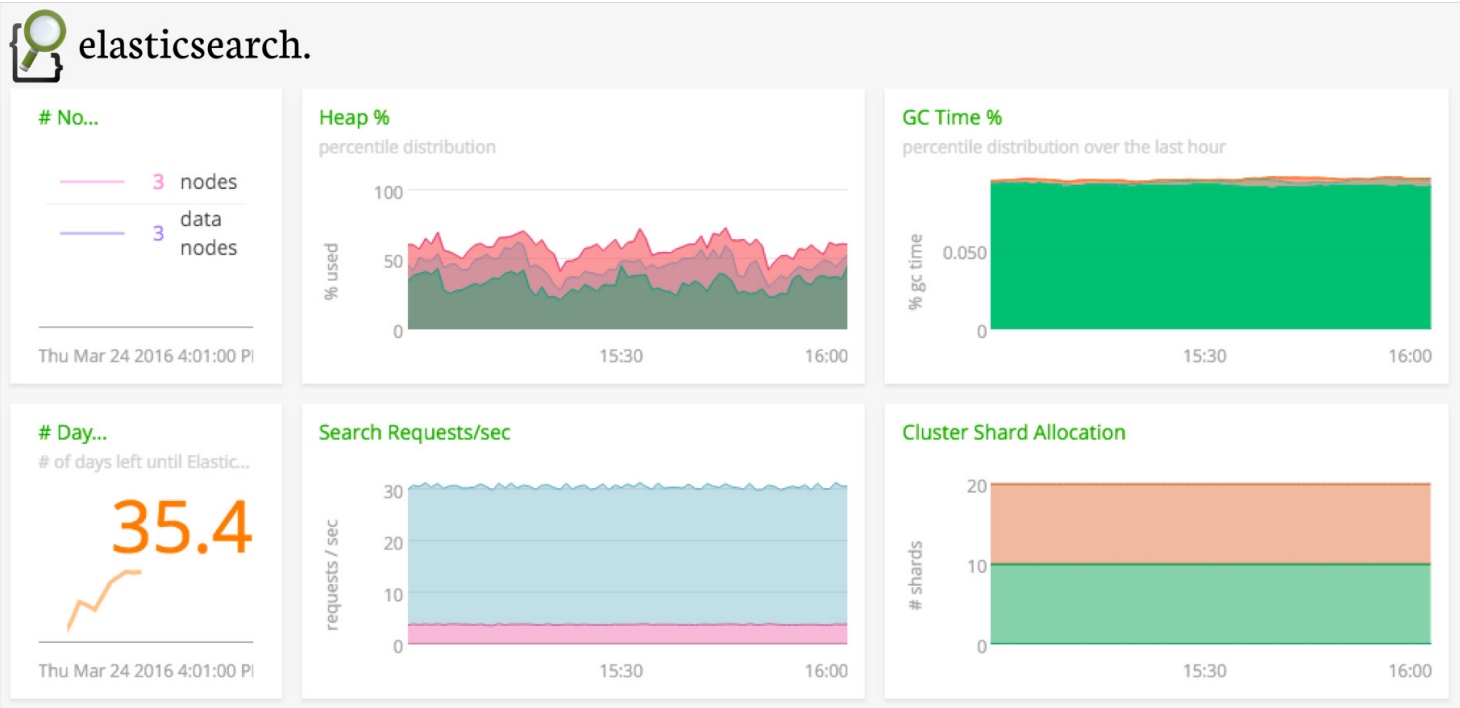
The SignalFx Difference

MONITORING CLUSTER STATUS: Noisy alert storms pose a huge problem for monitoring Elasticsearch. On its own, an Elasticsearch cluster experiencing issues would send a discrete notification on every node, one after the next. Under a best-case scenario, this is simply annoying, but a meaningful alert of a real problem. A common situation, however, is that an automatic scale-down has made nodes appear to be offline, even though there's plenty of capacity to handle the current workload. The alert is pure noise, and the result is a fatigued operations team with no issue to address. SignalFx makes it easy to isolate signal based on actual cluster status by assigning a score to the host and alerting on maximum value. If all the nodes in the cluster change to yellow or red, you only get one, meaningful notification.

DURATION CONDITIONS: Applying time requirements to alert rules helps determine whether an issue actually requires attention. Elasticsearch can recover a failed machine by restarting replicas on another node. SignalFx helps set duration thresholds so that you know if a problem persists longer than the window required to restart and self-heal. You aren't alerted of and forced to troubleshoot an issue that Elasticsearch auto-recovery will fix before you can reach it.

SCALING AND CAPACITY PLANNING: Knowing when to scale Elasticsearch requires plenty of runway to manage re-sharding, which can be challenging if you don't want to lose availability. Re-sharding is a complex process, and doing it while still writing to the old index makes it even harder. SignalFx's pre-built dashboards make it easy to compare document growth to storage growth and absolute storage consumption in real time, essentially modeling remaining capacity so you can plan for the future before you suffer performance problems. An alert on storage consumption above a specific threshold is a trustworthy indicator that large merges will cripple the service, and it's time for scale planning.

CURATING METRICS AND GETTING VISIBILITY: There are many metrics specific to Elasticsearch, and knowing where to start and what to monitor can be difficult. Similarly, Elasticsearch doesn't operate in isolation, so analyzing a single service's data without the context of other application and system metrics from across the infrastructure restricts value. SignalFx curates the Elasticsearch metrics that matter right out of the box based on experience with large, diverse architectures in production. SignalFx also provides pre-built dashboards, meaningful alerts, and automatic configuration that give you a running start on monitoring your modern environment.



About SignalFx

SignalFx is the most advanced monitoring and alerting solution for modern infrastructure. Our mission is to help cloud-ready organizations drive high levels of availability in today's elastic, agile, distributed environments. With SignalFx, development and operations teams gain a real-time view of, interact with, and take action on the infrastructure and application metrics that matter. We have enterprise customers including Yelp, Cisco, Zuora, and Hubspot and thousands of users analyzing billions of metrics every day. SignalFx was founded in 2013 by former Facebook and VMware executives, launched in 2015, and is backed by Andreessen Horowitz and Charles River Ventures.