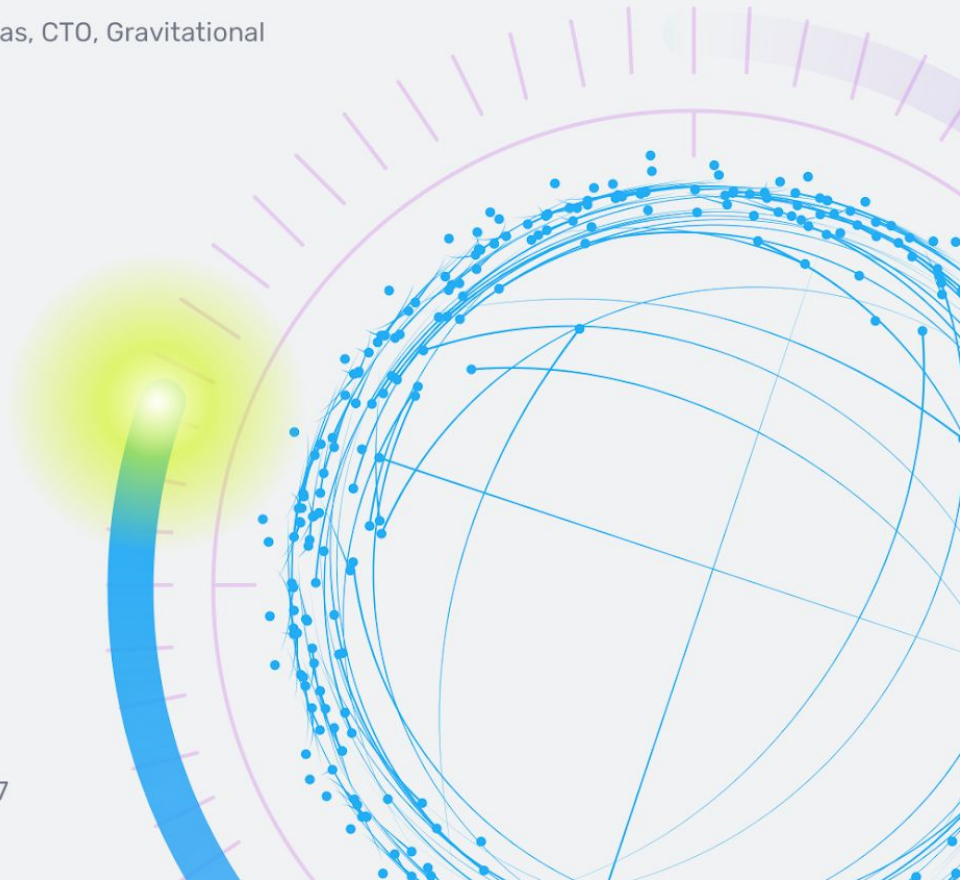
influxdata

How Gravitational Uses InfluxDB to Monitor Kubernetes

AN INFLUXDATA CASE STUDY

Sasha Klizhentas, CTO, Gravitational



December 2017

Company in brief

Based in Oakland with offices in Toronto and Munich, Gravitational is an infrastructure startup that makes cloud applications portable. Its technology allows developers to deploy and remotely manage applications on any infrastructure, anywhere in the world. Gravitational bridges the gap between software vendors and enterprise customers. Its mission is to simplify the management of cloud software. Gravitational's Gravity offers multi-region automation for distributed applications. It combines a production-hardened deployment of Kubernetes with Teleport, Gravitational's multi-region SSH server, enabling clients to manage multiple deployments of Kubernetes applications across regions, data centers and cloud providers. Using Gravitational, companies can take code on public cloud servers and deploy it on multiple private servers. Their applications run as a service on private infrastructure, allowing sales to enterprises that require private cloud editions of their services.

Case overview

Gravitational needed to meet SaaS providers' demand for deploying and managing their applications across multiple environments, including on-prem, private cloud and public cloud. SaaS providers needed such infrastructure flexibility to service enterprises requiring particular security and regulatory requirements. Through its Gravity product, formerly called Telekube (multi-regional Kubernetes for deploying and managing distributed applications), Gravitational sought to solve the SaaS portability and security problem by onboarding SaaS providers to Kubernetes, and helping them to deliver Kubernetes to various remote environments. Gravitational's team chose InfluxData as the high availability remote monitoring and alerting platform for the Kubernetes clusters they offer SaaS providers. Gravitational uses InfluxData to monitor Kubernetes both as a system component and applications running on top of it. Using InfluxData, Gravitational enabled SaaS providers to deliver portability and security, and thereby gain clients they would otherwise have been unable to service.

> *"Suddenly, they [SaaS companies] are faced with dozens of different deployments of their microservice architecture across the globe that they don't have access to. That's a problem we helped to solve."*
>
> ***Sasha Klizhentas,*** *CTO*

The business problem

There are two use cases that Gravitational wanted to solve for their customers — **private SaaS** and **multi-region Kubernetes**. Private SaaS is where Software vendors (including SaaS applications) need to deploy their complex software into private data centers or 3rd party cloud accounts owned by their enterprise customers (often due to regulatory or security measures). And multi-region Kubernetes is where Operations teams in large companies with many distributed product teams need to provide Kubernetes-as-a-Service within their organization across multiple hosting regions and multiple hosting providers. To support these use cases, Gravitational needed the solution to be portable and secure to avoid their customers being required to deploy different versions of their infrastructure and thereby face the complex problem of running infrastructure versions they can't control. The solution needed to provide multi-region deployment and management for complex, multi-tier applications across distributed infrastructure, as well as a means to monitor system and application performance.

The technical journey

> *"It's a unique product that provides features like clustering, commercial support, and data retention that matter a lot for us. That's why we picked InfluxData and that's why several years ago, we rolled out our first integration with InfluxData for the Kubernetes clusters that we deploy."*

To solve its business problem, Gravitational decided to use Kubernetes as a portable runtime for complex applications. It developed Gravity (which combines a production-hardened deployment of Kubernetes with Teleport, Gravitational's multi-region SSH server) but needed to monitor Gravity and its different components, to set up real-time performance alerts and enable high-level metrics and measurements analysis after the fact:

- Gravitational had about 100 production clusters (with 3-20 nodes per cluster) to monitor for clients such as banks and regulated companies with strict security requirements.
- Gravitational did not have full access to those clusters or their data centers.
- Keeping clusters running involved a lot of effort from Gravitational and its customers.
- Gravitational needed the clusters to be resilient, highly available, and engineered correctly from both their own and their customers' side.

Gravitational set out to evaluate several time series databases to assess their suitability for configurable monitoring and alerting, among them Prometheus and OpenTSDB:

- **Prometheus** - Gravitational found it to be performant, with first-class Kubernetes integration and low operational footprint, but also found it not suitable for durable long-term storage and limited to the size of a machine. Gravitational could not tolerate the possibility of data loss and instead needed to implement data retention policies.
- **OpenTSDB** - Gravitational found it to be performant and able to scale linearly but the high operational footprint, with dependencies on HBase and Cassandra as a storage backend, and on Zookeeper and Java, did not provide the out-of-the-box performance they needed.

Finally, Gravitational picked InfluxData because its features, as a platform purpose-built for metrics and events, met their needs:

- Very performant, with 350,000 writes per second
- Very low operational footprint (written in Go and has low memory usage even on a bigger footprint and higher scale)
- Retention policies and rollups (to guarantee retention of certain data for a certain period)

Gravitational also valued InfluxData's upgrade flexibility, with an Enterprise on-premise offering (InfluxDB Enterprise) and a managed cloud version (InfluxDB Cloud). Both come with clustering support. This meant that Gravitational's data center customers can upgrade to InfluxDB Enterprise if they want high availability, and can go to InfluxDB Cloud if they are on their private AWS account.

Technical architecture

When Gravitational's team deploy Kubernetes, they roll out several monitoring tools:

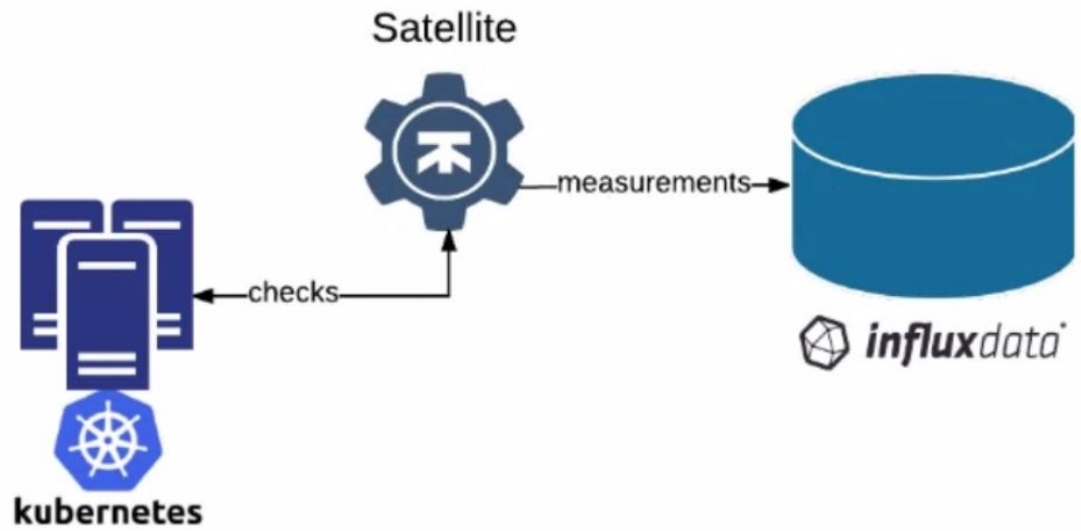
1- **Kubernetes itself**, which collects the data and has a lot of checkers internally that can provide interesting information about the cluster state.

2- **A distributed checker called Satellite**, which is a stable system deployed in the cluster and designed to combine with checks from Kubernetes, from the operating system, and from all the various components. Its purpose is to send the measurements to InfluxData and alert the Gravitational team if low-level components encounter a problem.

3- **InfluxData**, to which all monitoring information collected is sent for high-level analysis.

InfluxData (InfluxDB, Kapacitor, and Telegraf) is used on the backend, and Grafana is used on the frontend to display monitoring dashboards. Gravitational's monitoring infrastructure allows not only a cluster-wide view but also an application-centered view built around pods (groups of containers running together in Kubernetes).

Monitoring Kubernetes with InfluxData



Since Kubernetes is a distributed system that runs on standard Linux setups, Gravitational monitors the operating system itself as well as Kubernetes components:

- **Filesystem storage layer**: On this layer, Docker consumes a large number of inodes and often brings down clusters. To monitor that, Gravitational simply collects filesystem inodes measurements and also has overall inodes available, so whenever the number of inodes exceeds a certain threshold, Gravitational receives an alert.
- **Networking:** Kubernetes uses Bridge netfilter plugin, which allows filtering the packets in one of the Linux bridges in four different configurations. If a client's security team turns off this plugin, and if a cluster goes down, it loses overnight network functioning. That's why Gravitational wired out a special checker in Kapacitor (called deadman) that alerts on absence of certain data, to know if Bridge netfilter is on.

- **Etcd:** Etcd is the distributed, highly available database used for Kubernetes itself. It's usually deployed in the clustering mode but is sensitive to certain parameters of filesystem. Gravitational also developed a Kapacitor alert for Etcd that triggers a warning on high latency spikes.
- **Kubernetes itself:** Gravitational monitors Kubernetes itself, and the Satellite tool sends self-assessment information from Kubernetes. Whenever a Kubernetes node is not ready, it can send an alert that is displayed in the dashboard. Kubernetes itself is used to distribute those alerts — all such alerts are structured as ConfigMap objects.

> *"Thanks to InfluxData, we can say: Here's the overall cluster memory available to us through the aggregate measurements needed by Telegraf or the nodes."*

Monitoring applications with InfluxData

Using InfluxData, Gravitational ensures applications run smoothly on any modern Linux kernel. The application, in Kubernetes terms, is a container. In a given 100-node cluster, powerful automation using Kapacitor helps spot the application that doesn't behave correctly:

- Creating the deployment in Kubernetes creates containers and groups them in pods.
- A Kapacitor alert is created inside the Kubernetes cluster to understand performance of individual pods.
- Kapacitor measurements and Kapacitor streams help identify a problematic pod.

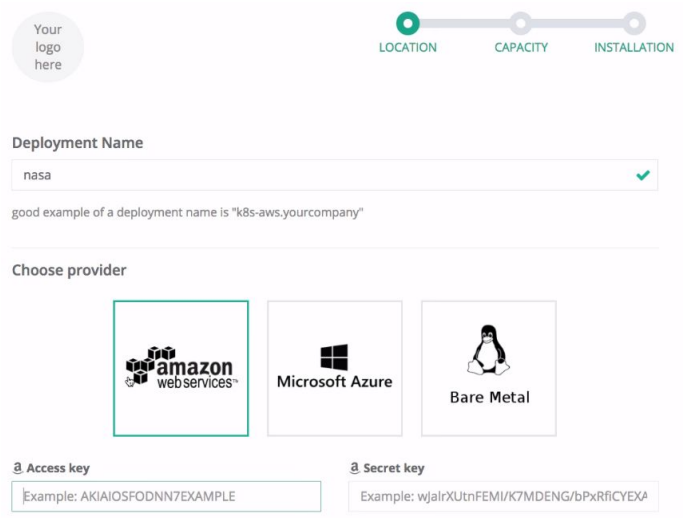
- Threshold-based alerts trigger a warning (for example, if pods running in a Kubernetes cluster consume more than a certain percentage of the CPUs available on a node).

> *"If you're working with a system like Kubernetes, it really raises the bar for your monitoring game. And you have to spend a lot of time instrumenting this. So I want to show you how easy it is with Kapacitor to spot the application that doesn't behave correctly."*

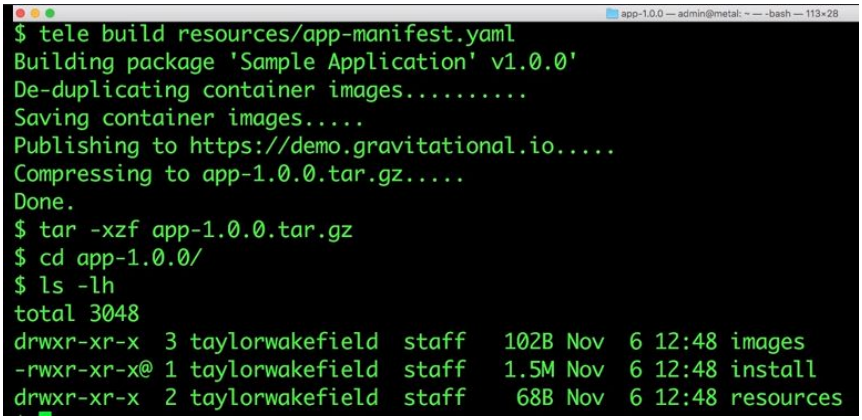
Results

Gravitational's Gravity "Kubernetes-on-Autopilot" allows SaaS companies to deploy their complex applications into on-premise and private environments.

Screenshot Showing Gravity Deployment Choices



Gravity achieves portability for multi-tier applications by packaging a client's applications, along with Kubernetes (and all dependencies) and creating a single, autonomous tarball that can be installed across server clusters.



The built-in Gravity 4.x monitoring service includes a configurable monitoring and alerting system using Heapster, InfluxDB, Kapacitor, and Grafana. It comes with built-in alerts for typical issues such as disk outages, networking problems and server misconfigurations. This allows for more scalable and proactive operational management across multiple clusters. In addition, customers can upgrade to InfluxDB Enterprise or managed InfluxDB Cloud for production-ready, scalable metrics storage with enterprise-level support.

Using InfluxData, Gravitational gained the competitive advantage of appealing to companies who:

- Are really sensitive about audit logging and data security, made possible through InfluxData's built-in flexible retention policies and rollups feature
- Care about high availability resilience and could easily upgrade to InfluxDB Cloud to cut their operational cost and completely off-load cluster support to the InfluxData team

By choosing InfluxData, Gravitational was able to make cloud applications portable for its clients and implement improvements that extended the power of Kubernetes and served their own need to scale the operational management of applications across many clusters.

About InfluxData

InfluxData is the creator of InfluxDB, the open source time series database. Our technology is purpose-built to handle the massive volumes of time-stamped data produced by IoT devices, applications, networks, containers and computers. We are on a mission to help developers and organizations, such as Cisco, IBM, PayPal, and Tesla, store and analyze real-time data, empowering them to build transformative monitoring, analytics, and IoT applications quicker and to scale. InfluxData is headquartered in San Francisco with a workforce distributed throughout the U.S. and across Europe.

Learn more.

InfluxDB documentation, downloads & guides

Download InfluxDB
Get documentation
Additional case studies
Join the InfluxDB community

799 Market Street
San Francisco, CA 94103
(415) 295-1901
www.InfluxData.com
Twitter: @InfluxDB
Facebook: @InfluxDB