

# Fender + Honeycomb:

## Faster Answers, Happier Customers



### ABOUT

Fender Musical Instruments Corporation (FMIC), whose portfolio of brands includes Fender®, Squier®, Gretsch® guitars, Jackson®, EVH® and Charvel®, follows a player-centric approach to crafting the highest quality instruments and musical solutions across genres. FMIC is dedicated to unlocking the power of music through electric and acoustic guitars, amplifiers, pro audio, accessories and digital products that inspire and enable musical expression at every stage, from beginners to history-making legends.

### ENVIRONMENT

AWS Lambda microservices in Go

AWS Lambda sending cloudwatch logs to Honeycomb

### GOALS

"Why would you enter into an [online space as a guitar company](#)? It's to know more about players: understand who they are, talk to them, and guide them along the way." —**Ethan Kaplan, Digital GM, Fender**

Fender's customer focus makes observability a key requirement: to understand and support your users, you have to have visibility into their experience of your service.

"The goal of the platform team is to create and support services that support our web and mobile applications and provide data and analytics to the business. In that way we are able to understand and support players." —**Michael Garski, Director of Software Engineering, Fender**

### WHAT THEY NEEDED:

- A powerful and intuitive interface for debugging and troubleshooting problems
- Fast search results across high-cardinality fields such as unique customer IDs

# Fender + Honeycomb: Faster Answers, Happier Customers

## HONEYCOMB @ FENDER

The Fender Platform team maintains their own ELK environment to aggregate logs and investigate problems, but have found the platform slow and difficult for team members to learn to use.

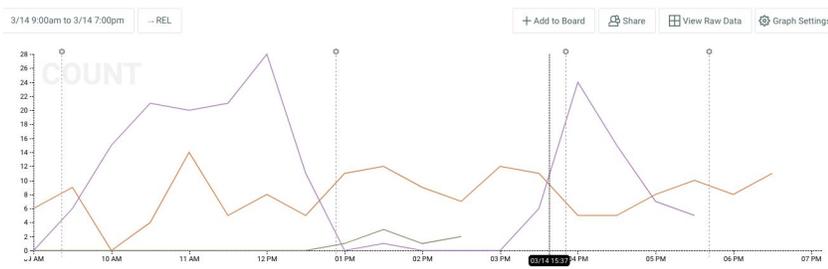
"We have an ELK stack, but the query language is cumbersome and the UI is not as easy to use as Honeycomb." **–Michael Garski**

They installed Honeycomb specifically to get the benefit of searching across all their Cloudwatch logs from Lambda at once, and have been especially pleased with the speed at which they can zero in on an issue.

"Honeycomb and Lambda work very well together. Without Honeycomb, it was very difficult to get all the Cloudwatch logs correlated, but now we can just do a quick search by request or userID with no difficulties." **–Michael Garski**

Recently, the Fender Platform team deployed an update to the subscription management service for Fender Play users. Soon afterward, they noticed errors related to the billing systems, which started to spike over the course of the morning.

Using Honeycomb, they determined that the issue was related to differences between their test and production environments and could confirm, thanks to the [Honeycomb Marker](#) they'd set, that this problem was definitely caused by their recent deployment. They were able to roll back to a stable environment within five minutes.



The graph shows the Lambda function status codes during the timeframe of the deploy. The spike in 500s occurred after the first Marker, then dropped after the second, which was the rollback. Subsequent Markers show that a third deploy still had the issue, but it was fixed in the 4th.

Even more critically, they were able to drill into their data and identify the exact users impacted by the issue, allowing their Customer Support team to contact those users before they contacted Customer Support to report an issue.

And all of this would have taken significantly longer using their existing ELK setup.

"If we did not have Honeycomb it may have taken an additional 30 minutes or so to determine what the issue was by poking around in ELK or Cloudwatch logs. Honeycomb's visualization and honeymarkers made it obvious that the issue was related to a recent release. On top of that, Honeycomb allowed us to determine the affected users and pass that information on to our support team." **–Michael Garski**

The team finds it easier and faster to get results with Honeycomb.