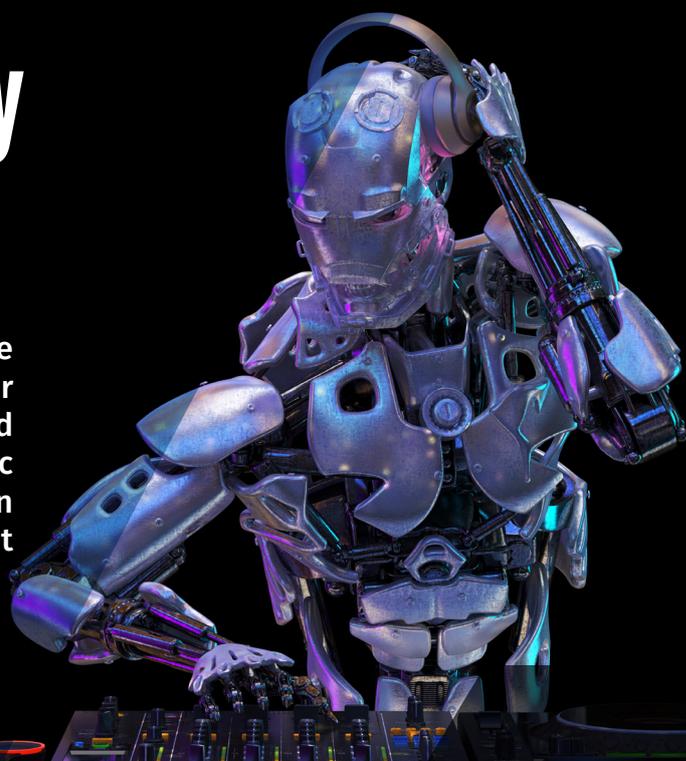


Success Story

RAI

All budding artists want to make sure they are being fairly rewarded for their efforts. With RAI, music publishers and other rights-holders can improve music metadata to ensure proper attribution for each and every download, imprint and stream.



Finance

The Company

A product of Exactuals LLC, and by extension City National Bank, RAI is based in Nashville, Tennessee, home to some of the most famous music talent this country has ever seen.

An open API, RAI enhances the supply chain by programmatically improving music metadata to meet digital distribution standards across multiple genres and music types, including major, independent and publishing, as well as more complex data like Classical and user generated content. These matches help ensure the right songwriters, publishers and other rights-holders are properly attributed when their works are consumed through digital platforms.

Led by Chris McMurtry (Head of Music Product), RAI's team of engineers contains many successful musicians in their own right. Together, they have created over 28 million links between ISRCs and ISWCs across the 73 million sound recordings and 12 million musical compositions managed by RAI clients. This helps music labels, publishers and distributors ensure that artists receive every penny that they are owed.

Highlights

- Industry:** Finance - Music Royalties
- Problem:** Push code from desktop to AWS, make all services Highly Available, retire on premise DB
- Solution:** Templated Terraform to deploy all AWS infrastructure, CI/CD for Docker, CloudEndure for workload migration
- AWS Tools:** AWS ECS, AWS RDS, AWS CodeBuild, AWS CodePipeline, Elasticsearch, Redshift, CloudWatch
- Outcome:** 200 Managed Resources deployed in 30mins. 5min build & deploy time for Docker.

Identifying the Challenges

At inception, RAI began by using home grown, manual deployments that had evolved over the past two years. Recognizing the needs of the industry, and more importantly their clients, the RAI team identified the need to go faster. In order to reduce development time to get new features into their product, the team wanted their development environments to become exactly like production. This would also give their developers a perfect environment to test in, without risking the production site.

In addition, to meet the stringent requirements of SOC1, RAI required an automated, audit-able deployment strategy to increase frequency of deployments and to meet internal standards set by their parent company, Exactuals.



Target

Decrease development time, and increase efficiency through automated QA processes. Release faster, and more reliably



Development

Development environments need to match production. Force frequent integration with other teams to drive a better release earlier



Reliability

Move all services to a High Availability infrastructure in preparation for supporting scale on demand use cases



The Solution

After evaluating RAI's needs, a solution was put in place to use ECS with AWS CodeBuild & AWS CodePipeline to auto deploy new docker builds on GIT Push to AWS CodeCommit. Utilizing ECS allowed RAI to build out micro-services for each of their API's that would run in their own containers.

Due to specific plugin requirements with RAI's on-prem PostgreSQL server, a solution was put in place to use CloudEndure to migrate the on-premise server to AWS. This allowed a seamless migration of data in the background allowing the teams to cut-over to the new server once any changes had been completed.

Extensive time and effort had been put into implementing data scraping, extraction, transformation and load scripts using Luigi. While AWS Glue was a good candidate, the team felt given their current skill-set, that it would be better to implement a Directed Acyclic Graph methodology. HATech implemented Apache Airflow to support the data pipeline automation running on docker inside AWS ECS. Apache Airflow was then taught how to scale on AWS ECS to accomodate workers as needed.

More Information?

Tab Wearing
Engagement Manager

e: twearing@hatech.io
c: 702 788 7150
a: 350 S Mill Ave, #B-201
Tempe, AZ, 85281



Chris McMurtry
Head Of Music Product

The Results

As moving to a new Highly Available and secure SOC1 platform was essential, HATech started off by creating a new AWS account. This allowed the implementation of Organizational policies that helped restrict access to regions and resources that were not able to be over-ridden through IAM.

The new infrastructure saw over 200 AWS resources deployed in 30 mins with Terraform providing the templated deployment models. With a simple push to GIT, developers were deploying their applications into their new AWS account, knowing that the controls and auditing had already been implemented using CloudTrail, Cloudwatch, Trusted Advisor to provide the reports necessary for SOC1 compliance.

The Postgresql server migration was seamless, allowing the team to retire an aging hardware server while still maintaining the performance needed to support their AI platform.

200
Managed

Resources

Amazon Web Services (AWS) offers reliable, scalable and inexpensive cloud computing services. Services include Virtual machines, messaging, Storage, Workload migration, Databases, Data Analytics, Machine Learning & AI and others. HATech manages and supports 1000's of resources.

1 Day
Setup Pipeline

CI/CD Pipeline

A deployment model in which code is continually pushed and integrated. Automated testing then validates the code and if successful pushes the code into production. HATech has templated CI/CD best practice methods for Jenkins as well as AWS CodePipeline, integrated with our test framework.

10
Days

Project Scope

HATech's agile and creative workshops turn our customers' vision into an actionable roadmap. An MVP is agreed, and key milestones are committed to. HATech construct an Agile Roadmap, creating the Epics and Stories required to execute. Daily we demonstrate momentum to our customers.

30
mins

Deployment

With AWS and templated deployment models, new environments can be deployed quickly from scratch. Once the patterns are built, it took only 30 mins to deploy an entire development environment including DB provisioning. Subsequent Docker container deployments took minutes.