CloudBees

CASE STUDY

Delta Dental of California Automates CI/CD; Gains Control and Insight with CloudBees Platform

DELTA DENTAL®

Industry

Insurance

Geography

United States

Summary

Delta Dental of California moved beyond Jenkins and adopted CloudBees CI to rapidly scale its DevOps environment and improve the speed, quality, security and compliance of its software delivery.

Challenge

Implement repeatable best practice governance policies to speed development cycles and increase collaboration.

Solution

Adopted CloudBees CI and DevOps tools and methodologies to support CI/CD automation, audit and compliance.

Results

- Rapidly scale software development environment
- Create a standardized, predictable CI/CD pipeline
- Automate rigorous audit and compliance requirements
- Establish trusted, automated processes
- Create tight collaboration between teams
- Set the stage for migration to the cloud

Product

CloudBees CI

Delta Dental was founded in 1954 to increase access to oral health care by offering insurance plans to companies and their employees. Today, it is the largest dental insurance plan in the U.S., comprising 39 independent member companies.

Delta Dental of California (DDC) is the largest member company in the network with more than 3,000 employees and approximately $9 billion in annual revenue. Always a pioneer, DDC played a key role in the development of Delta Dental's national provider file, which centralized account management across the entire organization. The national provider file paved the way for Delta Dental to offer insurance plans to organizations with employees in multiple states.

It's with that same pioneering spirit that DDC implemented DevOps. According to Adrian Blakey, a former senior DevOps engineer, "When we got serious about CI/CD, the natural place to start was with open source and Jenkins. It was a way to put some structure around the work the developers were doing."

CHALLENGE

About five years ago, DDC got serious about forming a DevOps team. According to Blakey, who was an enterprise architect at the time, "We were using Jenkins all over the place formally and informally." As the 400+-person IT department rallied around DevOps, they formalized DevOps into a full-fledged initiative with procedures and governance. The team quickly realized they needed to integrate tools so their developers could automate manual tasks.

Around the same time as DevOps was kicking into gear, DDC's IT strategy was shifting to containers. Although the DevOps team introduced Docker to the IT organization, DDC reached a ceiling with Docker after about two years. "We needed container orchestration and management, which led us to Kubernetes," Blakey said. "We also realized we needed a proper high availability installation for Jenkins. Therefore, we adopted CloudBees to provide 24x7 capability that we could depend on."

SOLUTION

A natural evolution

When Blakey thinks back on the path to CloudBees CI, he says. "There was no reason to consider any other vendors. From open source to a supported solution, we had so much invested in Jenkins that it wouldn't have been tenable to try anything else." The team also used CloudBees CI as a pilot for Kubernetes to facilitate its container strategy. "Back then we weren't quite ready for Kubernetes, so we backed out of it and instead installed several Linux servers and built something fairly conventional, which has worked just fine, and that's how we're still running it today," he said.

Adoption to expectation

The DDC DevOps team started creating jobs using CloudBees CI and then began building pipelines. There was an eager project team willing to push the envelope with CloudBees CI to the limits. "This team wanted to do things differently and break the norms," he said. "They were starting to develop microservices and Node. js applications on the front end. It was the first development team that adopted CloudBees CI. It was greenfield, and we could start fresh with them. We were all learning at the same time and it went really well."

With that successful learning experience under its belt, the DDC DevOps team evaluated the project and mapped out how they could create a repeatable practice. The realization that builds for the many dev teams weren't that different led to a breakthrough: templates. The DevOps team created a build request form in Confluence so developers could request a build and, from there, the DevOps team would work their magic.

Shortly after implementing CloudBees CI, the DDC DevOps team acquired a GitHub appliance. "We started doing everything with Git," Blakey said. "From that point on, collaboration and building pipelines with the development teams got even easier."

While Blakey admits that it's been challenging for some teams, user adoption has been swift and strong. "We're so far into it that our developers actually take it for granted. The fact that there aren't any issues, that things work correctly, means that people have become used to builds magically appearing on servers. Things just work."

> *"We couldn't survive without CloudBees CI. It's like a Swiss Army penknife. There are so many tools that you can integrate."*
>
> – Adrian Blakey, Former
> Senior DevOps Engineer

A build job for everything

Once all the development teams were fully functional in the CI/CD pipeline, the DevOps team fine-tuned processes to ensure strict compliance. "There has to be a build in CloudBees for everything," Blakey said. "If a team is organizing to deliver anything, whether it's an innovation effort or a development project, that's the route it has to follow. That discipline has proved time and time again that it crystalizes things."

Those meticulous processes have helped the DevOps team provide the audit trail that DDC release managers want. According to Blakey, "They want control and insight. Because we follow precise processes, we can link things together, write logs and put things into our change management system. We can provide a complete audit trail of what's going on." The net result is that release managers trust the DevOps process and compliance team members have confidence in the process.

The Swiss Army penknife of tools

The DDC DevOps team decided to manage its own tool chain. Blakey said the approach has taught the team how to configure tools so they can deploy them as quickly as possible for optimal use by the developers. "We couldn't survive without CloudBees CI. It's like a Swiss Army penknife. There are so many tools that you can integrate," said Blakey. They typically program tools using shell scripts, Pearl, Python or Groovy - having REST APIs makes integration a breeze.

An iterative process

Blakey readily admits that their DevOps journey is a work in progress. "You put some reasonable processes in place, set some guidelines and stick to them. Then, we modified processes as our circumstances changed."

Over time, the team was able to augment their processes with quality gate checks and security scanning. "Integrating automated testing with CloudBees CI was added later. Change management tickets are submitted to Jira on completion of a successful build and the artifacts staged for release. Releases are planned and the artifacts are deployed. We know they are verified and the code has been checked, so we close the ticket. There's still some work to do to get to a fully closed-loop process, but we're almost there," said Blakey.

Path to success and beyond

From initially dabbling in Jenkins to a CloudBees CI platform to a full-fledged DevOps culture, Blakey is very positive about where DDC is on its CI/CD journey. "I feel we've been successful with the path we took," said Blakey. What's Blakey's advice to other companies considering DevOps? "Do some research. Read some books and then just do something. Start small. Put some reasonable processes and guidelines around your CI/CD initiative. Then, stick to them, and iterate when you need to make changes," he said.

In just a few short years, the DDC DevOps team made a significant impact on the development organization's ability to deliver quality products in a shorter time. As he reflected on his team, their role and the culture, he said, "I think of DevOps as the glue that holds everything together. My team is the facilitator, and we get to know first-hand what works and what doesn't. It's very much a team effort."

When asked what's on the horizon for the DDC DevOps team, Blakey said, "We're moving everything to the cloud. We're going to create a complete set of pipelines for microservices on Azure."

RESULTS

Productivity boost

"We can turn builds very, very quickly and there's no friction, no latency," Blakey says.

Predictability

"Our developers have gotten used to builds magically appearing on servers," Blakey says. "They fill out a form, and the build appears. We judge our success by the lack of complaints. The fact that no one really complains, I think is fantastic!"

Process improvements

"It's very, very infrequent when something doesn't work. We realize that the developers totally depend on this," Blakey says.

Cost savings

"The one thing that's going to save us money is automating manual tasks," Blakey says.

Governance

"We don't allow anything to go into an environment unless there's a repeatable process and that means there has to be a build in CloudBees CI for everything," Blakey says. "Otherwise, it never finds its way out the door."

Audits

"Release managers are a constituency we're serving much better because we can provide a complete audit trail of what's going on," Blakey says. "We've built trust into the system, so the release managers can say, 'Now, I can trust that the changes have been made, and the code has been tested.' They can press a button to give their approval. We've automated the process in a way that builds trust."

Attracting talent

"We look for DevOps engineers that have a few core competencies," Blakey says. "Can you program? Do you know CloudBees? Do you know Artifactory? Do you know Jenkins?"

> *"We don't allow anything to go into an environment unless there's a repeatable process and that means there has to be a build in CloudBees CI for everything. Otherwise, it never finds its way out the door."*
>
> – Adrian Blakey, Former Senior DevOps Engineer

CloudBees CI is built on top of Jenkins, an independent community project. Read more about Jenkins at: www.cloudbees.com/jenkins/about

0122v00

CloudBees.

CloudBees, Inc.
4 North Second Street | Suite 1270
San Jose, CA 95113
United States
www.cloudbees.com
info@cloudbees.com