Progress Chef

Standard Bank: Our DevOps Journey

CASE STUDY

Standard Bank is the largest bank in Africa. As an established institution with a long history, it has a technology legacy that is the product of more than 40 years of development. It's a large mainframe shop, and its overall platform environment is quite heterogeneous. Some of the platforms include Red Hat, SUSE, Solaris, and AIX. In the front end of the technology stack there is, for example, Java and AngularJS, while you can find Cobol in the back end. Although its traditional release and management processes work, they are slow.

It became clear to Standard Bank that, to remain relevant, there needed to be changes. The goal was to transform its delivery process to provide new features and services at velocity.

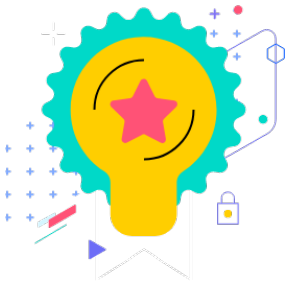
The Beginning

Approximately 18 months ago, Dawie Olivier, who was then Chief Information Officer for the retail bank, responded by trying to accelerate service delivery. His ultimate goal was a software pipeline that could support continuous deployment. Of course, introducing continuous deployment into a large, established organization is challeng- ing to say the least. Dawie and his team were able to implement continuous integra- tion but then ran into roadblocks. The team found it difficult to get everything they wanted to build into the production environment.

Then, a year ago, Dawie became the Executive Head of Group Technology Build, with a far-reaching mandate that took in everything from solution architecture to integration all the way down to just before production. He and his team reached the point where they were building a complete release train but it took too long.

One problem was in provisioning the different environments, from the ones that ran on developer machines to the ones that ran in production. Each of these environ- ments was provisioned and governed differently, so getting consistency was always difficult. Another problem was that getting a new virtual machine could take weeks.

It was time to start breaking down silos. A close partnership between Dev and Ops was essential if there was going to be any real improvement but it demanded a real change in the way work happened at Standard Bank. Dawie reached out to Mike Murphy, Head of IT Operations, his counterpart in the world of release infrastructure. Together they thought about the problem of how to release services and applications quickly. This was where Standard Bank's DevOps journey began.

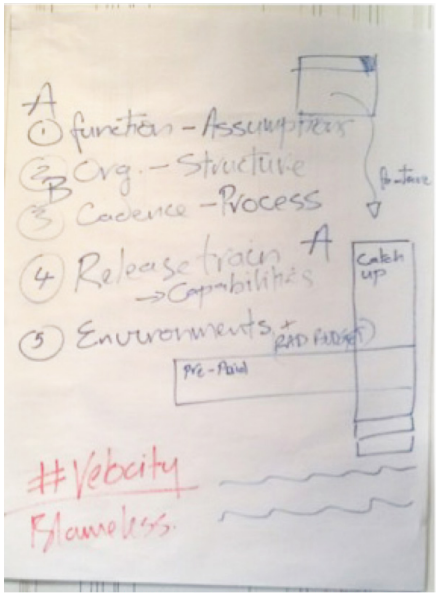


After studying case histories and success stories, and reaching out to DevOps practitioners around the world, Dawie and Mike wanted to expand the conversation even more. In the spirit of DevOps, Mike and Dawie decided against a top-down approach. Instead of delivering a plan they expected others to follow, they decided to put people from Dev, from Ops and from the business side together in a room, connect the group with other smart people, and let them define the journey. Not surprisingly, they also brought Chef into the conversation to help them plan and execute their DevOps journey.

Planning the Journey with Chef

The first of a series of planned visits from Chef was three days long and, during that time, Chef and Standard Bank worked together to develop an approach. Dawie Olivier, who represented the development side and Mike Murphy, who represented the operations side, brought together the sharpest minds at Standard Bank to answer these questions:

- Was the group willing to make the changes required to adopt DevOps?
- If not, was the group willing to accept the consequences (basically, irrelevance and obscurity) of continuing with the old ways?
- If the group did want to adopt DevOps, would they work with Chef?
- Would the group accept the responsibility of planning the journey?



On the first day, the Chef team spent time with about 35 Standard Bank execu- tives. For many of the participants, it was the first time they had worked together. Chef took everyone in the room through basic DevOps, an explanation of automa- tion, and how Chef fits into the story. A key point was that, in order to implement continuous deployment, some companies apply that process to infrastructure, while others apply it to applications. Chef integrates those two aspects. With Chef, you can create a single train for applications and infrastructure.

In the afternoon, the audience learned about the business reasons for using Chef and received a crash course in Chef fundamentals. For example, the group learned about resources, roles and environments. Although the people

in the meeting weren't the people who would write the cookbooks, the objective was to give everyone involved a common vocabulary. DevOps is about inclusion and a shared language makes that possible.

At the end of the day, Chef asked, "Where are you now?" "Where do you want to be?" The Standard Bank group went off on its own to devise a plan. The group want- ed to decide how best to use the next two days to answer those questions. Here is their agenda.

The group wanted to outline:

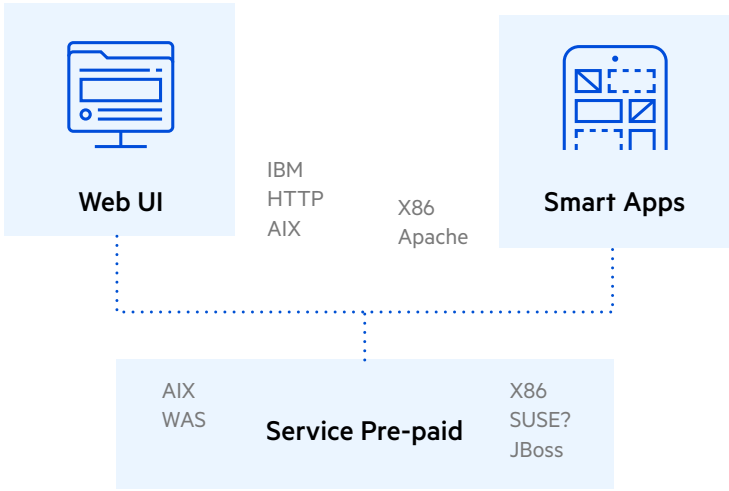
- The vision and scope of the project
- The function of the project and the assumptions behind it
- Release train capabilities
- The organization of the team that would work on the pilot project • The cadence of the project
- The environments used by the prepaid feature
- The budget

The group decided that part of the Chef team would spend the next two days training the engineers who would actually write cookbooks and oversee the release pipeline. At the same time, some of the executives would meet to decide what area of the business would be best for introducing Chef. The goal was to automate an entire slice of the stack and to build an entirely automated pipeline.

The executives wanted to select a feature that would provide customer value immediately. It would be pointless to put lots of effort into something that didn't ben- efit the bank's customers as soon as it was available. Secondly, the feature needed to belong to a group that already had experience with disciplines such as agile and continuous integration. These two criteria were best met by using a feature that was part of the bank's Internet portfolio. Those features are close to the customers and one of the teams that handled that portfolio—the mobile team—was one of the teams in the banks most truly driven by the need for velocity.

Of course, stability was also a concern. Although the feature needed to be a critical one, it shouldn't be something so central to the bank that, if it failed, the bank couldn't function.

From a range of options, the group decided to implement the prepaid feature, which would allow bank customers to prepay utilities such as electricity and broad- band. The pilot project would create the mobile app, with a web front end following shortly afterwards. Eventually, it would be possible to use the feature from an ATM.



The group also discussed the team that would work on the project and some guiding principles.

At the end of the three days, everyone came together and Dawie spoke to the group and summarized the situation. “We are going on a journey and we want to im- plement the prepaid feature. We want to use Chef. Every two weeks we will demon- strate success with showcases. If anyone thinks this strategy is incorrect, you have to speak up now.” No one disagreed and ev- eryone had enthusiasm for the project. The group was ready for change and ready to take risks. It was time to put aside the old way of doing things. It was time for some- thing new.

Guiding Principles and the Team

The planning group wanted some principles that would guide the team and the proj- ect as a whole. Dawie said, “The first principle of our journey is moving quality to the left.” By this he meant that everything had to have quality built into it from the outset, and the process had to be repeatable and as automated as possible. Speed would follow as a logical consequence of the automation. Everything the team did would be in service to that goal.

Here are the principles that Standard Bank adopted:

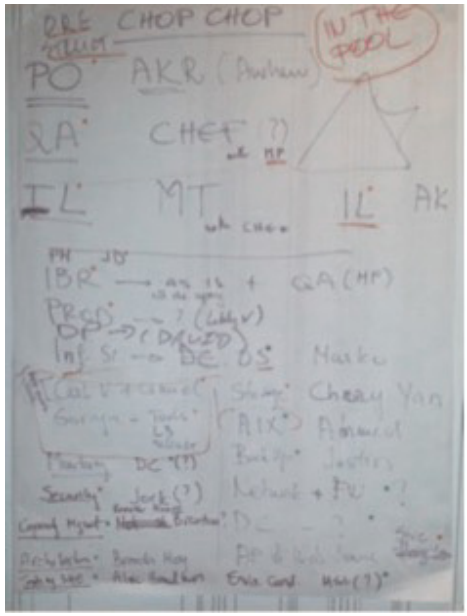
- Don’t be a chop.

> “Chop” is a South African term for someone who does something idiotic.

- Velocity is your guide—just do it. • Perform blameless postmortems.
- Think as a lean startup.
- No pets, only cattle.
- Fail fast, fail forward.
- Practice trust and respect. • This is a partnership.

The group put these principles in practice immediately. Some people were wor- ried that they didn't have enough resources and people to take on anything new. For example, there weren't enough firewall experts to run normal operations as well as be part of the DevOps team. Another distraction was that people would get into de- tailed discussions of how complex the infrastructure was and not make any progress.

Sometimes the group resolved problems quickly. Sometimes, Chef helped by bringing up the guiding principles, such as having blameless postmortems and think- ing like an agile organization. Problems with money or resources could be discussed with the executives who were committed to making the project a success. "How many people do you need to handle the firewalls? One hundred? Two hundred?" "Oh no, we need two people." "Done. Next question." The group was determined to move at velocity.



Structuring the team highlighted one of the challenges of creating a DevOps-centric organization within an existing organization where everyone is already very busy. How do you juggle everyone's responsibilities and are you willing to dedicate some people to the new project? If you do that, what happens to their existing workload?

The group decided to have a core team who would only work on the prepaid feature. Then, there would be people who, while still part of the core team, would when neces- sary go back to other parts of the organization. Finally, there were people who did the mainstream work who would be pulled in as required.

The team included people from many disciplines across Standard Bank. Just a few of these areas were se- curity, storage, monitoring, application services, infrastruc- ture and tools. You can see the complexity in this drawing, done during the meeting.

To emphasize the focus on velocity, the team's name became the Chop Chop team. Of course, "chop chop" means "quickly" or "without delay." To improve communica- tion,

people on the core team relocated to a space where they could all work together. To make their new home a welcoming place, they added a coffee maker and a popcorn machine. Of course, every great team needs a great t-shirt and the Chop Chop team was no exception.

Switching Gears

The introduction of a DevOps culture and automation had far-reaching effects on both development and operations. Creating consistent environments had long been diffi- cult for Standard Bank. Mike Murphy, Head of IT Operations, described the process.

"We could spin up VMs fairly quickly. That was never the real issue. The issue had more to do with creating the machines in a predictable, standard and consistent fashion. The machines spun up relied, to a degree, on humans doing the right thing and we know that, oftentimes, that doesn't work. Also, spinning up a cluster of ma- chines to create an environment was not something we contemplated. Machines were literally spun up one by one, on their own, and in their own ways. The consis- tency simply wasn't there.

"For example, if we had an application that was built from scratch and deployed onto a virtual environment in production (with its associated high-availability (HA) and disaster recovery (DR) elements), we'd sometimes encounter a problem when invoking either the HA or DR component. This was, more often than not, as a result of differences in the configuration of the three environments that was caused by re- liance on manual work. We didn't really have peace of mind that either the HA or DR capability would operate as designed."

Another issue was time. Mike says, “If there was a request from a project to spin up a “full-stack” (OS, DB, middleware, etc.), it would start with the VM itself getting spun up and when that piece was working, it was handed over to the database team, then to the middleware team, then to the backup team and then to the monitoring team. All of these teams, whilst housed within IT Infrastructure, operated relatively independently of one another and you’d have to hand off from team to team to team to get the activity completed. Because the hand off was manual and the configuration and deployment was manual, you could end up waiting weeks or longer before the solution was working.”

“Handoffs are a killer. In any lean environment, they’ll tell you that you lose between 15% and 20% of efficiency at each hand off. It only takes four or five handoffs before I should start paying the business for the right to develop the app. ”

Dawie Olivier,
Executive Head of Group Technology Build

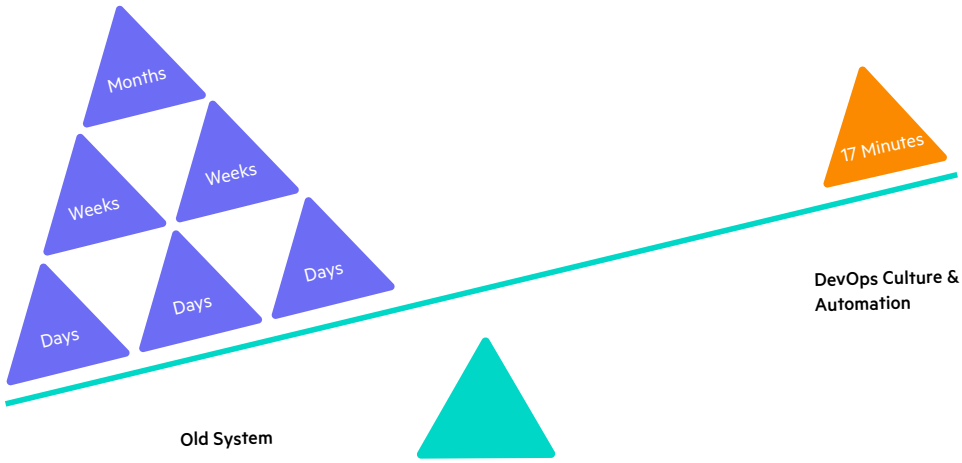
The solution was both technical and cultural. The Chop Chop team, responsible for the DevOps pilot project at Standard Bank, is multidisciplinary. Everyone in- volved with the prepaid feature, from oper- ational administrators to developers, sit together and work together. They benefit from each other’s insights and add to the discussion. In the past, if someone made a configuration tweak to improve perfor- mance, people would have found out about it by accident, if at all. Now, the change is a deliberate part of the design because it was a part of the conversation.

Crossing boundaries also shortened the learning cycle. Knowledge went straight into the solution instead of existing as documents that were handed over to another group. Nothing was lost in translation. The prepaid app was developed much more quickly than with the traditional approach.

After automation, deployments also happened more frequently. Team members participated in creating Chef recipes for automating the deployments. The Chop Chop team can now cycle through three different versions of a deployment in a day, or even half a day, just because they don’t have to wait for something to manifest in an environment.

Under the old system, creating an environment could take weeks. Now, thanks to a DevOps culture and automation, the team can spin up the entire Internet banking environment in 17 minutes.

Creating an environment Then and Now



Mike stressed the importance of cultural change and empowering the people on the team. “Chop Chop was conceived, planned and delivered by the team. My role and Dawie’s was really hands off. The only thing we did was give the team access to resources and clear obstacles out of their way. It was a self-discovery journey for them. They literally did everything themselves. That’s why we’re looking to rework some of the cultural elements in the organization. How do we shift the culture from being a man- agement-led culture to an engineering-led culture? How do we get the guys who do the actual work to drive the direction we should be going in?”

Testing Infrastructure

Members of the Chop Chop team found the DevOps journey to be both exhilarating and challenging. Derek Chung is the team’s iteration manager and manages the deliverables. Mark Figueira works in Quality Assurance. Marcus Talken is the techni- cal lead. Together they discussed what working on the team meant to them. Their discussion revolves around change—changes in process, changes in approaches to testing, changes in tools and changes in culture.

To set the stage, Mark describes the waterfall approach that Standard Bank has traditionally used to develop applications.

“Business had its requirements. Those got handed to a business analyst who drafted an FSS (functional system specifications). The FSS went to the technical teams. Depending on the organization, one team would deliver the infrastructure and the other would deliver the application. In parallel, someone would write the test cases based on the requirements within the functional spec.

"It would get to a point where development would complete some form of unit testing. Then, the application would be handed off to another organization for com- ponent integration testing. When that phase was complete, another organization per- formed system integration testing.

"There were three testing cycles and we were always picking up bugs, throwing the application back over the fence to development or, if there were other require- ments, back to the business analyst who would then confirm the requirements with business, update the functional spec, and update the test cases.You could be work- ing on a project for five months and still hit a bug that delayed the whole process."

In contrast, the Chop Chop team uses a DevOps approach. Mark says, "The product guy is on board, the technical guy, the QA guy, we all understand what we're going to deliver. "

Other changes include what is tested and how testing is done. The Chop Chop team's pilot project, the prepaid feature, is actually a part of a larger initiative that has adopted agile methods and testing strate- gies, so there are existing application tests that the team can use and adapt. What is new is test-driven infrastructure.

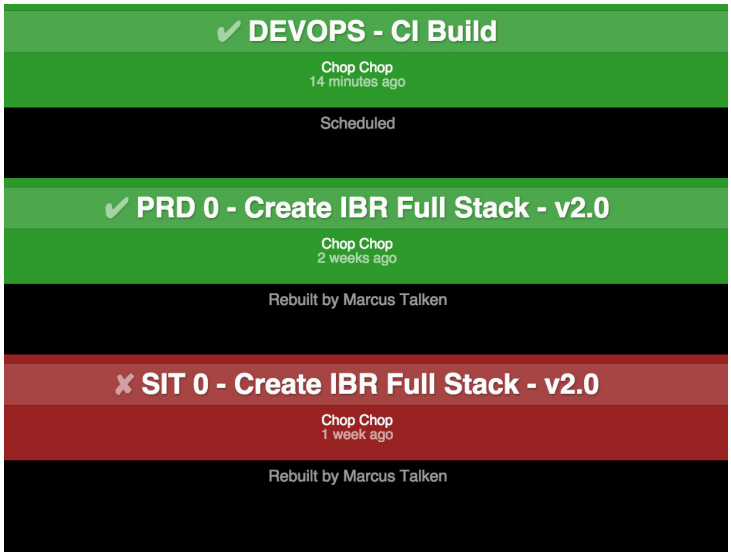
"With DevOps, you pick up issues up front rather than three months, six months down the line. It's a good change for us."

Mark Figueira,
Quality Assurance

Derek says, "Historically, we've never actually done any tests on infrastructure. We'd build a server, we'd deliver it, we'd build a database, we'd deliver it – we would subsequently get come backs around the quality of the server builds; they were not consistent. We never actually had the concept of testing an infrastructure component. On top of that, we're dealing with the whole concept of shifting left when you write your infrastructure tests."

Marcus adds, "For all of us, writing test cases up front was different. I'm from the traditional infrastructure world where testing comes last, if at all. We were really struggling with the concept of getting the test cases sorted out. It took a while to figure out the test cases first, do a build, and then run the tests again."

The team had to learn several new technologies. Along with Chef, they learned test kitchen and Serverspec. Mark describes what it was like when they first started to use test kitchen. “In kitchen, it was a case of not really knowing what we don’t know. Initially, we’d map out what we felt would be the tests we should pass. We started with a manual process and then from that, we automated it. By the end, what we delivered was not what we expected up front but it was a lot better.”



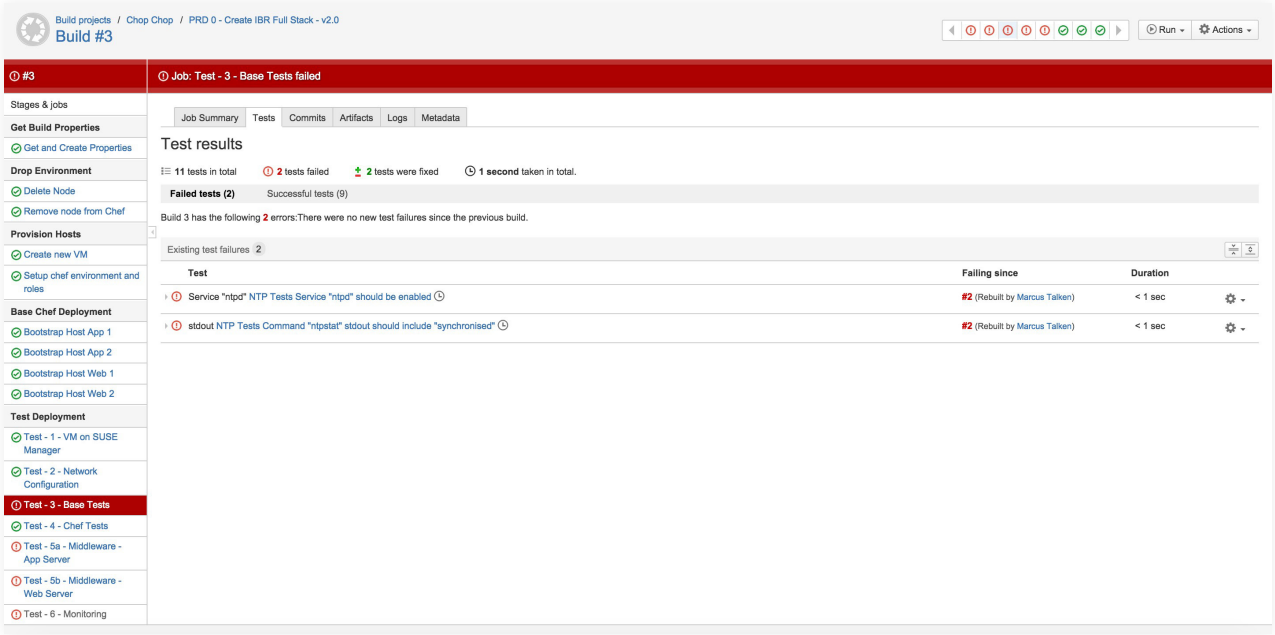
They also learned how to use Bamboo for their continuous integration server. The team used the Serverspec tests they developed to run in test kitchen and incorporated them into their Bamboo runs.

Marcus described the continuous inte- gration process. “We’ve got a continuous integration cycle for infrastructure that runs hourly. Coupled with that, we’ve embedded some of the infrastructure and application tests. Every hour we recreate the box and it runs all the infrastructure tests and applica- tion tests to make sure that any code changes we’ve made (the box is always built from the latest version of source avail- able, not necessarily the latest release available) is then run through all of those tests to make sure that they still pass.”

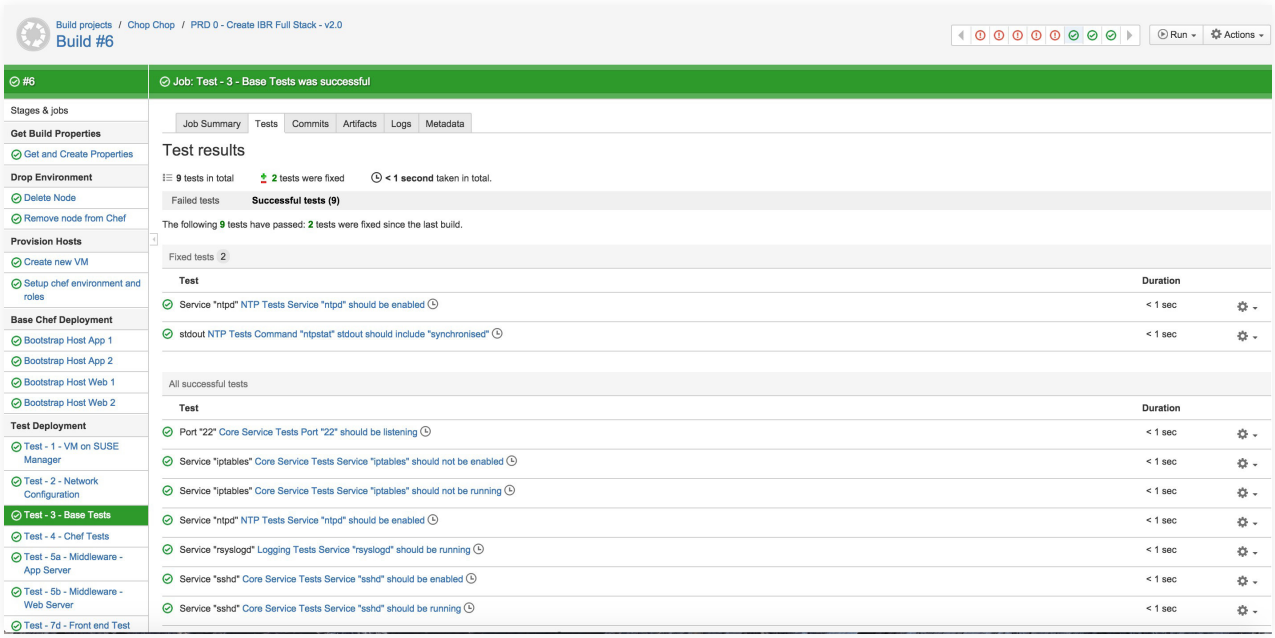
The Bamboo dashboard shows the status of the continuous integration build as well as the system integration tests and the production builds.

Marcus says, “The infrastructure tests make sure that, for example, the correct ports are open, the correct services are running, and the files are in the correct places. We then have contract tests, which are basically back-end server tests that exercise the service calls. The last test is a front-end functionality test that uses a tool to test the actual UI by mimicking what a user would do. We also check that, if you create a payment at the front end, all the necessary back-end calls happen to ensure that the payment goes through.”

Here is an example of what the Chop Chop team sees when tests fail.



Here is an example of what they see once all the tests pass.



Mark says, “When the build comes out the other side, we’re very confident that what was requested is what was delivered.”

It takes time to completely adopt a new approach. Marcus notes that, “Officially we write the tests before, unofficially it’s a bit of a mix and match. There are tests that are written before but there are times when the cookbook gets written and then we need to write the tests to make sure it works. “ Writing tests up front is still a challenge but is recognized by the team as a very important aspect of DevOps and a key driv- er to the future success of Chop Chop.

Derek points out that change doesn't happen immediately. "It's a behavioral thing, to get the guys to change from the usual way. It's something that we need to practice and get a little more discipline with so we're doing it all the time."

Another challenge was deciding who should actually write the tests. Derek says, "Just figuring out the roles and responsibilities is something new. Security might think that the server team should be writing the security tests to make sure the server is compliant. The server team might say it's the security team's requirements so they should be writing the tests. It got us talking. It's the whole point of DevOps. We're all debating it. We haven't fully answered that question."

Many factors have contributed to the team's success. On the technical side, Marcus cites automation as extremely important. Derek agreed and said, "I can't imagine if one of us broke a machine and had to wait two weeks for another one to be spun up. It's literally a one-button click to get things right."

Autonomy and a measured approach also mattered. Marcus says, "I think you need not take the enterprise's view on everything. I won't say we ignored enterprise standards, but we did things in a way that would work for us. As an example, we chose Red Hat as our operating system over SUSE. Although the bank standard is generally SUSE, Red Hat was just a better fit.

"Also, most organizations try and do everything perfectly from day one. We did it in baby steps. Can I build a vanilla server with nothing on it? Yes. Now, can I build a server with a little bit of security on it? Yes. Next, can I automatically deploy Chef? Yes. Now, can I automatically deploy Chef with a role? You can't do everything in one go.You've got to implement something, fix it, make sure it's working and then try and make it a little bit better."

On the non-technical side, everyone stressed the importance of teamwork, transparency and executive support. Derek said that the people on the team had more than expertise in their fields. They had a can-do attitude and worked together. Marcus said, "Everyone was core. There was no hero business where one person was doing everything. Everyone had a job to do, they knew what the goal was and they got together and delivered. If anyone deserves to get rewarded, it's the whole team."

Also important was support from the executives. Marcus said, "I think transpar- ency and exec support was very important. They believed that we could do it and never actually told us how to do it. They said, 'Don't burn the bank down, but go do it.'"

In terms of transparency, Marcus made sure that anyone who was interested knew what the team was doing. “With every team we engaged with, we tried to impart as much information as possible, even if they weren’t doing exactly what we were doing. We tried to get everyone on board.”

> **“It’s about having the right culture. With it, everything is possible”**
>
> **Marcus Talken,** Technical Lead

Finally, a blameless culture was essential. Marcus said, “When someone did something that broke a box, broke an environment, we didn’t go on a witch hunt. If something didn’t work, there was no finger pointing. It was, ‘OK, this is what broke. What can we learn from it? What can we do differently so we don’t have this problem next time?’ That was a really big change from the way I saw the organization work previously. Any environment where you’re going to experiment, you need to be able to make mistakes and not get into trouble for that.

The Future

On February 11, the Chop Chop team went live with their prepaid feature. Currently, it’s available internally, on the Standard Bank network, but anyone who’s a part of that network can use it. The next day, there was a large, internal IT conference at Standard Bank where executives discussed what they want to do in the coming year. Dawie Olivier gave a presentation, showing how the app was promoted through the different quality gates into production, and how a person could log on and use it.

The Numbers

The Chop Chop team gathered some metrics to demonstrate what they’ve accom- plished with Chef and DevOps.

- Time to build the stack: 26 minutes to build for production nodes (2 web servers, 2 app servers, with end to end deployment and testing)
- Number of automated tests embedded into Bamboo:
  - Test Environment – Total 209 Tests
  - 31 Infrastructure
  - 178 Application – already existed before Chop Chop
  - Production Environment – Total 39 Tests
    - 31 Infrastructure
    - 8 Application – written by Chop Chop

- Number of cookbooks – 12 (5 custom, 7 community)
- Time to market for pilot Internet Bank Refresh deployment – 12 weeks
- Number of catalogued automated services: 3
  - Redhat Linux VM
  - Enterprise Application Platform (EAP)/JBoss on Redhat
  - Apache on Redhat

Looking Forward

The demonstration was well received and there was much discussion about the implications of adopting a DevOps culture. The members of the Chop Chop team are now thinking about how to put what they've learned into broader practice and what else needs to happen to bring Chef and DevOps to Standard Bank.

Derek Chung, the iteration manager, says, "Whatever the organization is looking for, whatever we can automate, is basically what we're trying to do. Hopefully, it will reduce the amount of administration the bank faces at the moment and the amount of governance. The big question is, 'Can we replicate the success we've had in the last few months?'"

"The part that's powerful is when I can spin up a preconfigured and certified environment that's ready for my entire Internet banking solution to run in it. That requires a very significant change to our approach. That's what Chef does."

Dawie Olivier,
Executive Head of Group Technology Build

Marcus Talkien, the technical lead, wants to build on the team's growing library of Chef cookbooks to create a service catalog. "We've written our own JBoss cook- book, which has a whole bunch of functionality. The next project that comes along that needs JBoss, we can easily deploy it. That's true as well of Apache. The next project that needs Oracle, it may take us a while to get our heads around it, but going forward, once we've got that Chef cookbook filled out, it becomes relatively easy that any project that wants Oracle, we've got a cookbook. It's just about putting in the right parameters."

He also wants to make sure that they safeguard the tools they use, such as Chef and Bamboo. “These tools need high availability (HA) and disaster recovery (DR) environments. For example, we’ve put in the Chef server but it doesn’t have HA. We definitely want to have a backup server ready to go in the event we lost our Chef box.”

Another consideration is how to best take advantage of their enterprise CMDB. It contains a great deal of information, but how to use that information well is still under discussion.

The implications of DevOps are far reaching and affect many areas of the bank. People from a variety of teams want to know how automatically provisioning environ- ments will affect what they do. For example, how will they know when an environment is provisioned?

Because the bank has, of course, many auditing and compliance requirements to meet, the question of how to integrate the bank’s IT service management suite, BMC Remedy, into a DevOps workflow looms large.

Derek says, “From an ITIL perspective, there’s a lot of considerations that need to be taken into account. I don’t think the bank was prepared for such a disruptive concept. Before, we accepted the whole ‘four weeks to build a server’ idea. Slowly, the information would trickle down to the configuration management database (CMDB) and, obviously, the cost would get charged to a specific client.

“There have been a lot of questions asked by the team that’s looking after ITIL, primarily about how the ability to create an environment on demand will affect them. There’s a lot of detail around instant configuration changes and release manage- ment.

“Historically, we haven’t had a mature configuration management practice. We’re just setting that up. This is one of the big opportunities we have to influence things. It’s not just CMDB. It’s also total cost of ownership (TCO).

“We have a TCO model that only charges per core that you’re allocated. How do you charge on a consumption basis? At the moment, we don’t have answers. The way the bank charges back is very complicated and the financial chargeback models are kind of static. You’re only charged every three months for what you’ve used in the last three months for full allocation. How do you charge if I allocate you half a core for one day? How do you pay for it? We are coming to grips with it. It will be a part of a big theme we’re trying to drive.”

As always, culture is an important factor. Mike Murphy, head of IT Operations, discusses how to move from a pilot project to a more broad-based initiative. “How do you take the continuous delivery, DevOps concepts and scale them across a much larger organization? How do you approach it? We proved that what we thought was provable really works and the benefits we suspected we would see we absolutely do see. How do we make this single instance and the practices and culture associated with it much more widespread? I don’t

think it's a case that one can scale this in a linear fashion. Scaling is a complex thing to contemplate. Technology scaling can certainly happen but how do you scale and influence culture in a much broader way?

"I guess you could say that, at the moment, we're in kind of a contemplative, information gathering phase. We don't have the answer. We're speaking to a number of companies, including Chef. ChefConf at the end of March is going to be a great opportunity to talk to people to get a sense of what is possible and what has and hasn't worked.

"One other thing that's important in this new way of doing things is to actually ask the team itself to contemplate how they would do it, given their hands-on, personal experience. How would they influence their colleagues? We're trying to get the an- swer from the bottom up as opposed to a top-down, management approach. That's where we are at the moment. We certainly do not know. We're trying to work out what's next. Everyone was wowed by what they saw at the showcase. There's a general, kind of corridor buy in, so how do we make this real?"

About Progress

Dedicated to propelling business forward in a technology-driven world, Progress (NASDAQ: PRGS) helps businesses drive faster cycles of innovation, fuel momentum and accelerate their path to success. As the trusted provider of the best products to develop, deploy and manage high-impact applications, Progress enables customers to develop the applications and experiences they need, deploy where and how they want and manage it all safely and securely. Hundreds of thousands of enterprises, including 1,700 software companies and 3.5 million developers, depend on Progress to achieve their goals—with confidence. Learn more at www.progress.com.

Rev 2022/01 RITM0140982

Worldwide Headquarters

Progress, 14 Oak Park,
Bedford, MA 01730 USA
Tel: +1-800-477-6473
www.progress.com

- facebook.com/getchefdotcom
- twitter.com/chef
- youtube.com/getchef
- linkedin.com/company/chef-software
- learn.chef.io
- github.com/chef
- twitch.tv/chefsoftware

Progress®