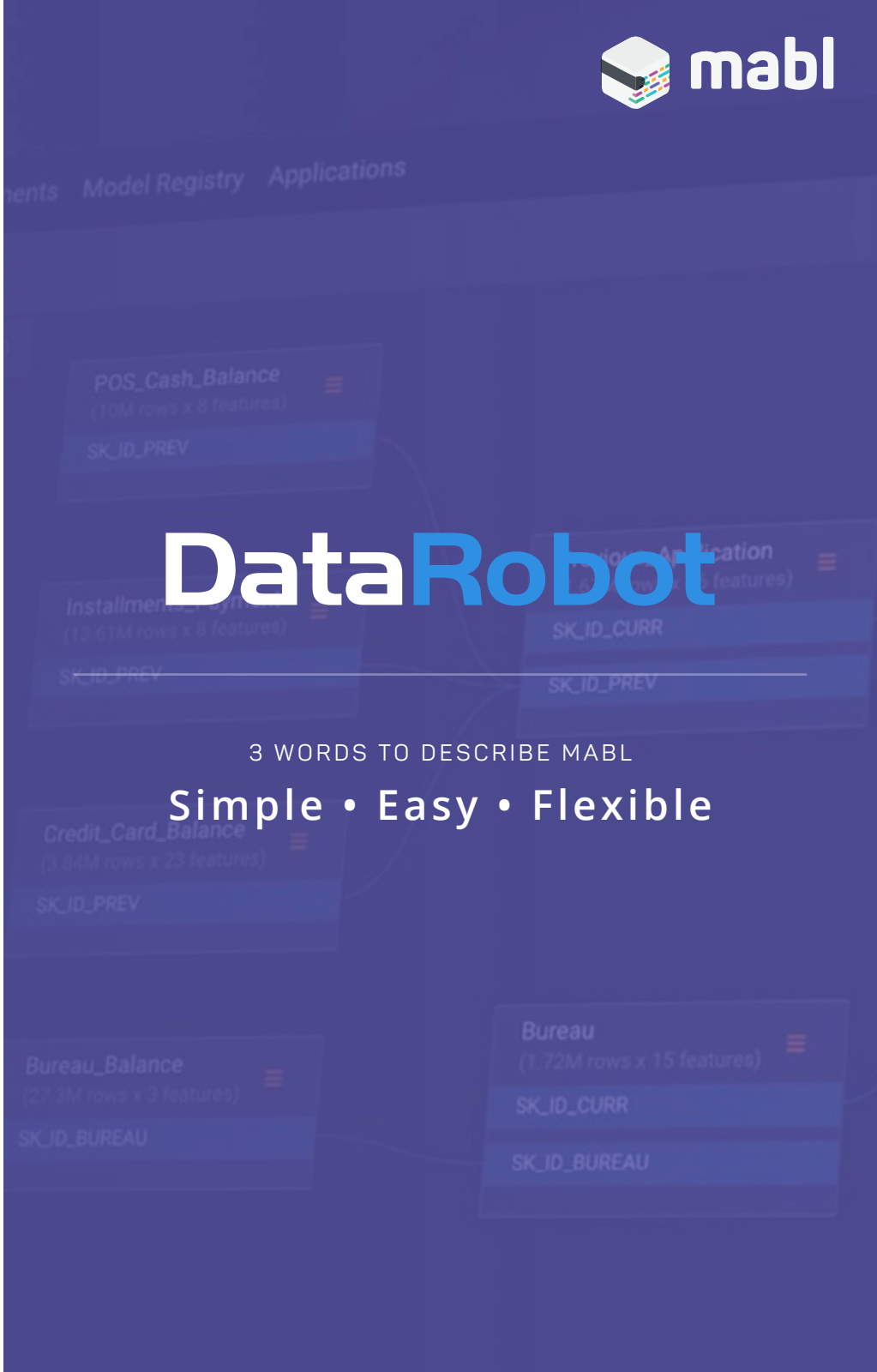
CUSTOMER SUCCESS STORY

mabl enables DataRobot to innovate at the cutting edge

DataRobot is an enterprise AI platform that makes it easy for organizations to leverage the power of artificial intelligence. Their platform democratizes data science by automating data analysis, giving businesses the power to prepare, build, deploy, and maintain AI-driven applications at scale.

Mabl is the leading intelligent test automation platform built for CI/CD. It's the only SaaS solution that tightly integrates automated end-to-end testing into the entire development lifecycle. With mabl, it's easy to create, execute, and maintain reliable tests, allowing software teams to increase test coverage, accelerate development, and improve application quality.



Automation for AI Applications

It's hard to find a higher velocity startup than DataRobot. The enterprise AI company raised $320 million in 2020, bringing their valuation to $2.8 billion and cementing their status as one of the top-funded AI startups in the world. Over 2.18 billion data models have been built on their automated data science platform and their engineering teams have logged over 1.4 million hours building the product used by customers across the Fortune 50.

To build their platform, DataRobot has a large engineering division with a number of teams working independently in domains that govern different functionalities. Though much smaller, the QA team is responsible for working with each team simultaneously to ensure each new feature or update works. As a result, they're often managing a wide variety of projects with different timelines and testing needs.

Further adding to the QA workload is that DataRobot's platform enables businesses to prepare, build, deploy, and maintain their own AI-driven applications, creating a number of unique cases that need extremely fine-tuned testing.

At a startup dedicated to automation in the data science field, automating testing is a no-brainer. Director of Engineering Meghan Elledge, who leads the QA team, says: "If I have to do something twice, I'm already thinking about how to automate the third time."

But despite the need for automated testing, few tools were able to keep up with the pace of innovation at DataRobot. Most automated testing platforms needed significant maintenance or lacked the controls necessary to test the specific needs of DataRobot's advanced capabilities. Newer features in particular have unusual flows that are challenging to fit into existing automated testing tools like Cypress.

> *"If I have to do something twice, I'm already thinking about how to automate the third time."*
>
> **Meghan Elledge**
> Director of Engineering, DataRobot

Data-driven to the core

When a friend suggested mabl to Elledge, she was immediately interested in the platform's ability to produce actionable information to better inform DataRobot's testing strategy. As a data science company, DataRobot is continuously applying analytics to their own product, often building their own internal tools to leverage their team's expertise in machine learning and data science to improve performance. The fact that mabl would not only streamline testing operations through automation, but also with analytics, strongly aligned with Elledge's mission of "quality assistance."

Quality assistance shifts the focus from pure software testing to a broader culture of quality throughout the engineering process. This mindset encourages development teams to play an active role in testing, so Elledge and Quality Engineer Troy Carter prioritize making testing available to developers earlier in the development process with tools that allow engineers to test their branches in different ways. QA can then focus their efforts on helping engineers test their work, write test plans, perform exploratory testing, and analyzing results from both automated and manual tests. With mabl's digestible reports and simple UI, engineers were more engaged in testing at an earlier stage of development.

Mabl's easy-to-navigate UI makes it easy to onboard new users, especially with remote work. As they moved through their proof of concept, Carter and Elledge found that mabl was a valuable teaching tool for QAs and engineers alike. When a user is given a login, they're able to essentially "press play" and watch the entire testing process, see how complex tests are built, and understand why that test was used. QAs are able to move between domain groups more easily and engineers are able to quickly understand how to test their own code. Elledge is even starting to use mabl to onboard new team members since remote work makes traditional teaching methods more difficult.

Of course, having the product team as active users sets a high standard. As Elledge notes "We have lots of machine learning engineers and data scientists doing research on the next big thing. People want to know what's really going on and how we're gathering information from our own product for improvements."

Stabilizing a Fast-Moving Product

Even when starting new domains, mabl allows QAs and developers to import existing tests and adjust as necessary, saving them the effort of starting from scratch. Mabl's reusable flows then make it easy to start a complete test by simply importing flows and adding another building block. Team members working in newer domains have a particular interest in using mabl because they have unique introductory workflows that don't fit in with other existing test tools. Rather than spend valuable time trying to adjust that existing tool to meet a new set of requirements, they just want to write a quick test to ensure a new feature runs on a frequently changing system - which is easily accomplished with mabl.

Overall, features like reusable flows, JavaScript snippets, and autohealing give Elledge's team more time to think about how to test better and why issues are happening. As Elledge explained it: **"You want your QA people thinking about why weird things are happening, not pointing and clicking at stuff. You want results and to be able to analyze those results rather than revising steps."**

Using mabl to write end-to-end flows that test DataRobot's UI, the QA team is able to increase test coverage, iterate more quickly, and implement changes faster. With the ability to fine-tune tests for innovative new features that don't fit the mold of other automated testing tools, QA is able to test more of the product with less effort.

With a year of using mabl under their belts, DataRobot is expanding adoption throughout the QA and product teams. While Elledge and Carter are strategically expanding their own team's use of mabl, they also find that many people in the engineering groups are expressing interest in bringing mabl into their own work. This makes sense to Elledge: "It's about making use of what automation does best and what people do best. Giving people back their time and their productivity [with automation] gives them the freedom to actually think about other higher value things."

Few companies are better positioned to understand that better than DataRobot and mabl, which continue to be at the forefront of democratizing testing with automation and machine learning in the enterprise.