SMARTBEAR
ReadyAPI

Case Study

Major American Airline Leveraging SmartBear API Testing Tools

SMARTBEAR

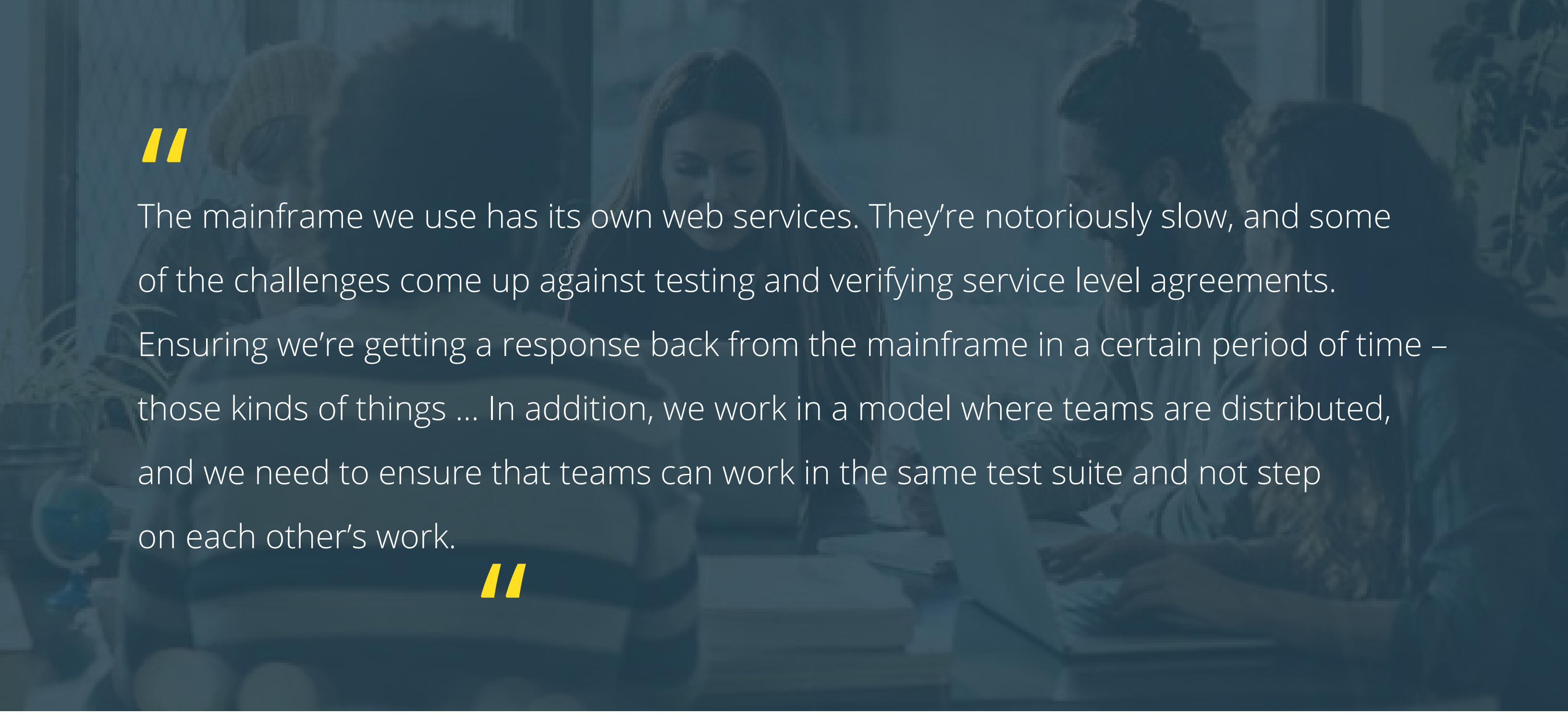
Introduction

It's no secret that the role of technology is changing. From marginally driving efficiency to fundamentally innovating the airline industry, it's caused a lot of disruption. In fact, a recent world economic forum report stated, "Digitalization represents an exciting opportunity for the aviation, travel, and tourism ecosystem, with the potential to unlock approximately $1 trillion of value for the industry and wider society over the next decade."

In this industry, Application Programming Interfaces (APIs) play a large role in digital transformation and are key tools that allow travel businesses to provide value to customers and, ultimately, compete better. From hotels and flights to cars and buses, there is an API for virtually every area of travel. Our Airline customer is no different.

"More and more teams are moving to REST services and APIs," says Mark Ruggles, Developer – IT Applications. "Many teams in our company are moving to build out whole microservices frameworks – we're building services to ensure proper booking, e-ticket confirmation, and ensuring we communicate to a mainframe. Just to name a few."

But with that transformation comes challenges.



> "The mainframe we use has its own web services. They're notoriously slow, and some of the challenges come up against testing and verifying service level agreements. Ensuring we're getting a response back from the mainframe in a certain period of time – those kinds of things ... In addition, we work in a model where teams are distributed, and we need to ensure that teams can work in the same test suite and not step on each other's work."

ReadyAPI as a Solution

QA teams throughout our airline customer found the answer to their challenges in ReadyAPI.

"To say that the Pro version of ReadyAPI offers comprehensive features is understood. To say that it offers more features that are essential to accomplishing the goals of building quality API software at our airline is a gross understatement."

Mark pointed out that ReadyAPI helped them with the following goals:

- Building automated test suites for API services in a DevOps environment
- Running those test suites in a Continuous Integration/Continuous Delivery (CI/CD) Pipeline
- Storing those test suites in a repository to be shared across distributed teams

Open Source vs. Pro Evaluation

As ReadyAPI was gaining popularity amongst QA teams within our airline customer, teams started to evaluate the Pro tool against the Open Source (OS) tool and found the pro tool to have additional features that were mission critical and essential. Our customer was able to highlight the following differences:

Environments:

"ReadyAPI environments are like containers that house all the preconfigured endpoints and more. They let users reuse test suites across different physical environments. The benefit comes from being able to change the endpoints, etc., across an entire test suite by simply making a selection in a drop-down menu. In other words, redirecting my entire test suite from the TEST to the STAGE environment is as simple as a click.

And since they can be passed in as a command line argument, your CI/CD pipeline test executions occur in one physical environment all the way through to production.

*This feature, **not available in the OS version**, is the foundation for test case reusability and maintainability across physical environments. And **it's an essential building block for pipeline test execution."***

Test Steps:

"There are many more test step types available in the Pro Version … I'll highlight a few of the more important ones" "…the Open Source (OS) version is missing some key sub-features that make working with these steps very difficult in the OS version and in some cases may even lead to using other third-party tools to fill in the gaps of what's missing in the OS version."

Step Name	*Description*
DataSource	Provides the ability to connect test cases to an external data source like an Excel sheet, database, flat file, etc. There is also a Data Generator DataSource variation that builds random data of all kinds. The DataSource step is the foundation and beginning of all data-driven testing (automation) in the Pro version.
DataSink	Provides the ability to target specific pieces of data to output to various types of files including Excel sheets, flat file, databases, etc. The output of this step could be used to build a DataSource for a subsequent test suite/case or could be used for reporting purposes.
DataSource Loop	This step is used in conjunction with the DataSource step to create test case loops. This step causes the test to repeat until the data coming from the DataSource is exhausted. This step is the trailing bookend for data-driven testing.

Composite Projects:

"Available in the Pro version, composite projects allow teams to store their test suites (project files) in a repository segmented so that test teams can work in the same test suite. It's like developers working in branched code, keeping teammates from stepping on each other's work. It's a must for distributed teams."

CI/CD Pipeline Integration:

"ReadyAPI builds the complex command line string for you that's necessary to run any set of tests in a pipeline. With the OS version, the command line has to be built manually, which introduces a margin for error."

ReadyAPI A Trusted Tool for the Present and Future:

ReadyAPI has now been in use with our airline customer for over seven years and continues to play a key role in the overall testing and software development lifecycle.

"I know that there are teams here who have integrated it with their CI/CD pipelines. And they've not only saved time, they've increased their release cycles exponentially. "

And even when in the face of uncertain times, the value of ReadyAPI was certain.

"Even with the pandemic, and the need to cut redundancies, we're not only keeping ReadyAPI, we've extended our license through the next several years. That's a testament to the value of this tool. We're entrenched with it. You simply can't do the same things with the open source that you can with ReadyAPI or any other testing tool."

SMARTBEAR

ReadyAPI

SMARTBEAR