

# 20X Reduction in Testing Time Achieved with API Test Automation and Service Virtualization



## The Challenge: Accelerating Testing in Parallel and Agile Development Environments

Ignis Asset Management is a global asset management company, headquartered in London, with over \$100 billion (USD) in assets under management. Ignis recently embarked on a large project aimed at outsourcing the back office as well as implementing the architecture and applications required to support the outsourcing model.

“To meet the business’s needs, a number of projects have to be developed and delivered in parallel,” explained Aaron Martin, Programme Test Manager at Ignis. “However, we didn’t have the resources, budget, and management capacity required to create and maintain multiple test environments internally. This limited test environment access impeded our ability to validate each application under test’s (AUT) integration with third-party architectures. Moreover, our third-party providers also had limited test environment access, which restricted the time and scope of their joint integration testing.”

At the same time, the company was transitioning to an agile development methodology. To support this initiative, they needed to adopt an automated testing solution to provide faster feedback after each build.

It soon became apparent that the existing testing process had to be optimized in order to meet these new demands. Executing the core test plan required 10 man-days. This process involved manually entering transactions in the originating application, which wasn’t the primary AUT. Moreover, they were also manually building simple stubs to simulate interactions with third-party components that were not integrated. To enable complete testing to occur in more agile, parallel development—without requiring additional test environments to be built and maintained—they needed ways to:

- Enable applications (or parts of the target architecture) to be tested against the Ignis architecture before integration into the complete Ignis system.
- More efficiently simulate the AUT’s interactions with third-party systems not yet integrated into the Ignis system.

## Parasoft API Testing and Service Virtualization Enables Ignis to Begin Extensive Automated Testing Before Integration

Ignis implemented Parasoft’s API Testing and Service Virtualization solutions to establish a test automation framework that not only addressed the challenges outlined above, but also helped extend test automation across the SDLC.

Ignis’s initial implementation of the API Testing solution focused on automating the generation of order management traffic at the API level. The AUT was the message architecture, which interfaces with third-party components—both existing services provided by business partners as well as services being implemented in parallel by outsourcing providers. From the application initiating the order, live trade scenarios were used to form their basic test transactions. Using SOAtest (Parasoft’s API Testing tool), they were able to run the full transaction test plan, generating new instances of the message from a data source. This data-driven message building took advantage of features such as SOAtest’s ability to update attributes to create unique IDs, set dates, and perform calculations.

In parallel with the functional test automation, Parasoft Virtualize (Parasoft's Service Virtualization tool) was implemented to simulate the expected transaction response messages from third-party components. "First, we rapidly implemented a simple virtual asset that provided a positive response to all generated transactions, enabling us to simulate third-party responses without manually developing and managing stubs," Martin explained. "The virtual assets were then extended to handle more complex response scenarios."

Ignis also implemented automated tests and virtual assets to test outsourced components fully-decoupled from the Ignis environment. They used this to establish a "quality gate" that had to be passed before progressing to the integration phase. Martin remarked, "This was quite useful, since their code quality was poor and repeated testing in our integrated environment would have impacted other deliverables."

## Leveraging Supero to Transform a Manual Testing Process into an Automated One

Since Ignis test resources were not experienced in test automation or service virtualization, they enlisted the help of an automation developer to build out their test requirements in the Parasoft ecosystem. Ignis engaged Supero Solutions to manage the implementation and ongoing test requirements since they had extensive experience implementing and using Parasoft. Ignis has now replaced all the manual test resources in one location with Supero resources.

Supero's expertise has been critical for building automated tests within the scrum teams, which is a key factor in the success of the Ignis agile initiative. "Using Supero allows us to flex our resources to meet project requirements while still maintaining a consistent approach," Martin said.

Once the implementation proceeded, the value of having a Parasoft expert lay the proper foundation became clear. From this starting point, any resource can now run test plans via Parasoft and enable virtual assets in the test environment with a very minimal learning curve.

---

**"With Parasoft, we can focus on the core test requirements and get more value from our investment in automation."**

---

## Results: A 20X Reduction in Testing Time

"With Parasoft's integrated functional test automation and service virtualization, we were able to reduce the execution and verification time for our transaction regression test plan from 10 days to a half day," shared Martin. This testing is not only automated, but also quite extensive. For example, to test the Ignis system's integration with one business partner's trading system, Ignis's fully automated regression testing now covers 300 test scenarios in a near UAT-level approach—with 12,600 validation checkpoints per test run.

"Previous automation implementations focused on automating testing at the UI level—with varying levels of success," Martin continued. "We determined that we really needed to generate transaction scenarios and traffic at the API level instead. With Parasoft, we can focus on the core test requirements and get more value from our investment in automation."

Beyond addressing the original challenges posed by the project, the solution has also enabled automated testing to occur all the way from the component/unit level to system integration. To achieve this impressive level of automation, testers fostered close relationships with the development team. Now, testers' role within the organization is elevated, and collaboration between development and testing has reached an all-time high.

© Parasoft Corporation All rights reserved. Parasoft and all Parasoft products and services listed within are trademarks or registered trademarks of Parasoft Corporation. All other products, services, and companies are trademarks, registered trademarks, or servicemarks of their respective holders in the US and/or other countries.



**USA PARASOFT HEADQUARTERS**  
101 E. Huntington Drive, Monrovia, CA 91016  
Phone: (888) 305-0041, Email: [info@parasoft.com](mailto:info@parasoft.com)