

Finance: Scaling Test Environment Access for Parallel Agile Development

The transition to Agile at a leading financial services company meant that their Development organization was reorganized into many smaller cross-functional (dev/test) groups. Ironically, this effort to speed up the SDLC actually introduced new delays. One example: a test environment that was once dedicated to a single team suddenly needed to be shared by 9 smaller teams. Due to the complex data setup required, the environment could be used by only one team at a time—the others had to wait. Since the test environment included a third-party application that was \$250K per instance, creating 9 separate instances of this physical test environment would have been prohibitively expensive. Service virtualization enabled them to establish 9 simulated test environments that gave each team instant, flexible access to the behavior of that system—with zero impact on the other teams.

The Challenge: Scaling Access to a Complete Test Environment Including an Expensive Third-Party Application

This company's mission is to provide personal investors direct access to investment and brokerage services. They focus their research and development efforts on making it easy for clients to research and select securities that suit their financial goals, as well as to monitor and optimize portfolio performance. Rather than “reinvent the wheel,” they leverage a proven third-party application to handle core industry-standard functionality, such as executing market purchases and sales.

Before the transition to Agile, the team responsible for trading functionality was able to complete their development and testing tasks using a shared test environment. However, once the team was split into 9 different teams—each of which was trying to complete different development and testing tasks in parallel—test environment access quickly emerged as a problem. Since each group had to have the test environment set up in a very specific manner, any attempts to access that test environment simultaneously meant that the groups were stepping on one another's toes—wasting time having to constantly configure and reconfigure the conditions and data needed to complete their particular tasks.

Restricting test environment access to one group at a time was not well-suited to their goal of accelerating the SDLC with parallel development. However, providing each group their own physical test environment was not feasible. Since each instance of the third-party trading application cost \$250K, this meant that they would have to spend \$2 million to make this application available in eight additional test environments. This option was deemed prohibitively expensive.

The Solution: Simulating the Constrained Dependency's Behavior and Data in Multiple Zero-Impact Sandboxes

The company was able to use Parasoft Service Virtualization and Parasoft Environment Manager to simulate the behavior and data of this third-party application and make it available on demand in 9 independent test environments that each team could configure and reconfigure as needed—with zero impact on the other teams.

Exercising the AUT's interactions with the third-party application, they were able to capture the behavior and data associated with their core use cases and make it available in "virtual assets." Parasoft Environment Manager was then used to design a master test environment template that included these virtual assets. From this template, any number of teams could instantly stamp out their own test environment—with the virtual asset configured to the appropriate state, and with the ability to easily add additional data to increase test coverage or to adjust response times as needed for performance testing. This way, each team could instantly access a preconfigured environment, then customize it for their own specialized testing needs—with zero disruption to other teams' dev/test activities.

Replacing the actual third-party system with virtual assets yielded additional benefits beyond enabling the Agile teams to develop and test in parallel. Previously, when their test environment included a real instance of the trading application, trading-related transactions could be tested only during trading hours—9:30 am to 4 pm Eastern time. Since the development and test teams were based in California, this meant that they could test only from 6:30am to 1 pm, which is only about 50% of their typical work day. With the virtual assets standing in for the actual system, testing could be performed 24/7, enabling the team to perform exploratory testing at their convenience as well as exercise these transactions as part of their continuous integration process.

Another benefit was that test execution time was significantly shortened. Tests against the actual system took over 20 minutes due to a delayed (asynchronous) response from the trading system. By adjusting the performance of the virtual asset, the team could get almost instantaneous responses, which expedited both automated and exploratory testing.

© 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