

A group of four people, three men and one woman, are shown in profile, looking intently at a large screen. The screen displays a complex interface with lines of code and data visualizations. The scene is dimly lit, with the primary light source being the screen itself, creating a focused and professional atmosphere.

Power Continuous Testing Through Service Virtualization



Continuous Testing Hurdles Magnify Application Development Pressures

Your application development and delivery teams face constant pressure from customers to accelerate the pace of software innovation. The reason is simple: Digital experiences are becoming the de facto way for organizations to differentiate themselves—and innovative apps are key.

Like other leaders, you may be employing Agile and DevOps practices like continuous testing to support this effort across mobile, web, core and legacy apps.

Yet, despite these more advanced practices, **you may be experiencing multiple challenges where manual testing and other environmental constraints impede your efforts:**

- The testing environments you need are inaccessible.
- You're waiting on other teams to complete their work.
- Your testers can't recreate negative scenarios in your production environment.
- You're racking up costly fees using third-party systems to support regression testing.

Let's look at how you can evolve your continuous testing approach so it's truly an enabler of high-speed and high-quality app delivery.

Even DevOps adopters have limits



73%

have adopted DevOps Process¹



54%

of the adoptees still experience a software testing bottleneck²

Here's how you get started

Enable Service Virtualization to Support Continuous Testing

Shift Left to Speed Testing and Boost Quality

Neutralize Dependencies and Work in Parallel

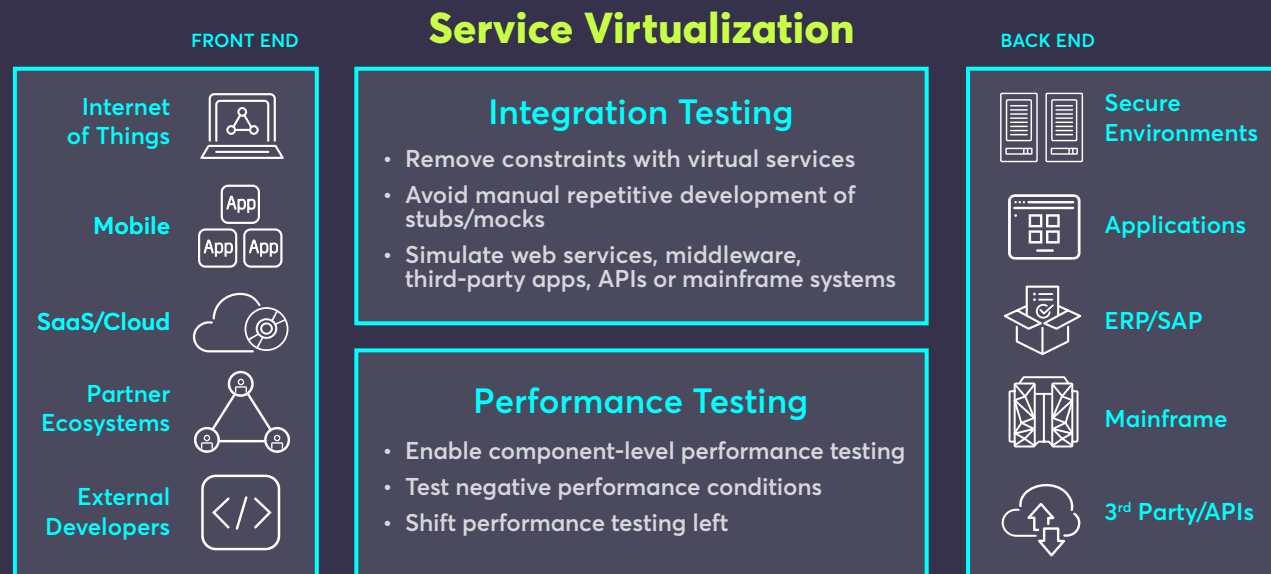
Enable Service Virtualization and Support Continuous Testing

App development can come to a standstill when you face any number of restrictions related to critical back-end components, such as when mainframe maintenance is underway or a needed system is being used by others.

For example, partners involved in integration testing may be working on a common API—so it's off limits until the task is complete. Or your ERP environment could have reached its capacity level, so performance testing grinds to a halt.

When you employ service virtualization, you can eliminate these kinds of roadblocks in your DevOps and Test environment by simulating previously unavailable services, systems, apps or APIs. Through an accurate rendering of dependent systems, your teams can perform coding and component-level testing whenever needed for tremendous gains in efficiency and greater responsiveness to the demands of the app economy.

With its ability to support continuous testing, service virtualization is a must-have technology for the modern software lifecycle.



Replicate reality

Unlike brittle stubs and mocks, service virtualization enables you to simulate the desired app behavior, data and performance of the actual dependent systems, so you get a real-world representation of how an app will function. This includes the ability to insert negative scenarios, such as error conditions and system delays, so your developers can see how they impact performance and make immediate changes to code instead of waiting for the last mile of performance testing.

Shift Left to Speed Testing and Boost Quality

In order to speed app development, you need to test sooner and more frequently as part of a continuous testing approach. By doing so, you'll be able to test more scenarios, more often, so you're able to identify and address problems earlier in the software development lifecycle (SDLC).

Service virtualization supports your ability to "shift left," so that your DevOps team can initiate testing as soon as possible. With greater insight into the performance of their code, your developers and testers can improve the quality of code as it moves from one step to the next—and ultimately deliver higher-performing apps.

When your teams "shift testing left," they can build in quality at a component-, development- and system-integration level far before conventional user acceptance testing activities.



Up to 90%
the number of defects detected at least 1-step earlier in the SDLC.³

Neutralize Dependencies and Work in Parallel

Are your developers and testers constantly waiting for environments to be available, or for sub-components or APIs to be ready? And how often have they had to wait until off-hours in order to connect to previously unavailable systems, such as the mainframe?

Being held captive to these kind of dependencies runs counter to your goal of bringing new apps to market faster and increasing customer satisfaction.

With service virtualization, your ability to simulate and access a virtual copy of everything you need makes former dependencies non-existent. As a result, your teams can work in parallel to perform their specific testing functions—avoiding downstream delays.

In this continuous testing reality, you can also improve test coverage because of reduced dependency on other components or test environments. This means you can do valid test scenarios 24/7 for multiple development and test teams.



**the amount of time
by which developers
were able to reduce
the application
dev/test cycle⁴**

CA Service Virtualization Gets Results

"Our organization has been using CA Service Virtualization for the last four years. Now, it is an essential part of the SDLC and stress-test programs for the largest and most critical applications."⁵

Global 500 Banking Company

"Generic stubs during development did not give us the capability of testing well beyond development. CA Service Virtualization allowed us to extend our testing and improve on delivery times."⁶

Global 500 Financial Services Company

"CA Service Virtualization is the best tool for quality, and it reduced the time it takes to get our applications to market."⁷

Capgemini

Learn more about CA Service Virtualization
ca.com/service-virtualization



1,2 <http://www.eweek.com/development/devops-adoption-high-in-2016-but-some-users-hit-qa-wall>

3,4 CA Service Implementation scorecard results

5 TechValidate Research Published: Aug. 25, 2016, TVID: A0D-5E2-F21

6 TechValidate Research Published: Sep. 15, 2016, TVID: 163-993-0AD

7 TechValidate Research Published: Aug. 12, 2016, TVID: 458-B6F-726

CA Technologies (NASDAQ: CA) creates software that fuels transformation for companies and enables them to seize the opportunities of the application economy. Software is at the heart of every business, in every industry. From planning to development to management and security, CA is working with companies worldwide to change the way we live, transact and communicate—across mobile, private and public cloud, distributed and mainframe environments. Learn more at ca.com.

Copyright © 2018 CA. All rights reserved. All trademarks, trade names, service marks and logos referenced herein belong to their respective companies. This document is for your informational purposes only. To the extent permitted by applicable law, CA provides this document "As Is" without warranty of any kind, including, without limitation, any implied warranties of merchantability or fitness for a particular purpose, or non-infringement. In no event will CA be liable for any loss or damage, direct or indirect, from the use of this document including, without limitation, lost profits, business interruption, goodwill or lost data, even if CA is expressly advised of such damages.