

# API Virtualization - Reduce Project Cost

## Introduction

This white paper describes how teams can use lightweight API-specific virtualization to minimize the costs associated with mimicking service behavior during development, testing, and integration phases of a software delivery process.

## Background

As APIs become increasingly prevalent in digital systems, software teams look to speed their delivery process using virtualized versions of their APIs to simplify timing and resource logistics. Traditionally, enterprise service virtualization solutions require standing up a full-stack of resources, training and management over software suites, and incur an enormous financial burden with entry costs into the hundred-thousands.

The total cost of ownership over an enterprise service virtualization solution, typically in the range of \$50,000 - \$200,000, includes:

- Time identifying an appropriate solution
- Initial Licensing costs
- Implementation and professional services costs
- Cost to reconfigure virtualization strategy due to project changes
- Employee training

The magnitude of these costs can often be overwhelming for agile project teams either just getting started on implementation of a service oriented architecture (SOA) or formulating a larger enterprise API strategy. Delays in budgeting and purchasing such a costly solution prevent teams from being efficient in their projects long after the need for service virtualization is realized.

## Solution

For teams that use APIs, who don't require mainframe or database level virtualization, or who abstract away more complicated components of a full stack through web services, virtualization of specifically API dependencies (aka. "API virtualization") is a significantly more affordable alternative to full-stack enterprise service virtualization.

API virtualization is a ground-up attitude to mimicking system behaviors and does not assume that an entire environment is required in order to simulate realistic behavior, just what is necessary. Service virtualization itself is only a supportive task to building the actual system, and any time spent on it is time not spent on building or testing the actual system. Therefore, minimizing the amount of work and time it takes to configure virtual APIs improves both opportunity cost and minimizes unnecessary enterprise licensing costs.

Time is money. With API virtualization, front-end designers, testers, and system integrators can begin their work as soon as developers produce an initial specification (WSDL, RAML, Swagger, etc.) early on in the design process. Using virtual APIs:

- Testing teams can immediately begin to build out test artifacts
- Front-end designers can complete realistic front-ends faster
- System integrators can make sure that new changes do not cause loss of revenue

The time saved by using cost-effective virtual APIs to simplify logistics between team members and work independently also represents opportunity cost savings in that when teams finish faster, they can move on to the next project faster as well.

## Advertisement

### **SmartBear provides API virtualization through ServiceV, a tool in the Ready! API platform.**

ServiceV allows you to quickly stand up virtual APIs from description specification formats such as WSDL, WADL, RAML, Swagger, API-Blueprint and from 3<sup>rd</sup> party API management systems such as 3scale, Intel/Mashery I/O Docs, IBM API Management Portal, and WSO2. In addition to ServiceV for designing and running your “Virts” (virtual APIs) locally, a stand-alone VirtServer allows teams to share Virts between members of software teams. These indispensable tools allow you to develop and test in parallel, control static or dynamic response data on the fly without rebuild or rollback, and transform traffic to and from Virts and real-time systems, all for around \$1000 per seat (list price).

## Conclusion

The proliferation of APIs in web, mobile, desktop, and cloud solutions demands that teams look to optimize their capabilities around faster delivery. Lightweight virtual APIs overcome the burden of cost, skill, and implementation time associated with full stack service virtualization, and empower teams with the ability to begin parallel work earlier, exercise fine-grain control over service behavior, and share resources as necessary, to achieve fast results with minimal skills or cost.

# SMARTBEAR

## About SmartBear Software

SmartBear is the choice of more than two million software professionals and over 25,000 organizations in 90 countries that use its products to build and deliver the world’s best software applications. SmartBear’s user-centric application management solutions support key software delivery processes of development, testing, API readiness, and application performance management across desktop, web, and mobile platforms. Get started at [www.smartbear.com](http://www.smartbear.com)