Enterprise Service Virtualisation

MarketUpdate

Introduction

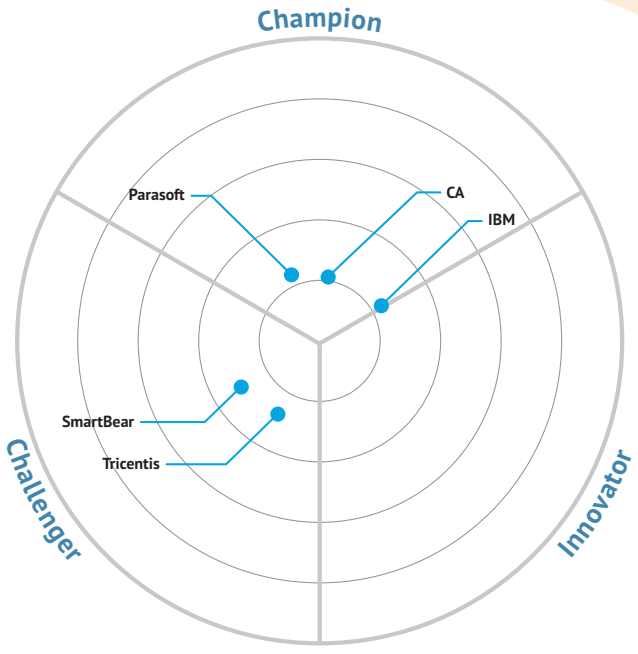
In recent years, service and microservice oriented software architectures have become more popular than ever. This has led to a growing number of applications that dwell within these architectures, and therefore rely extensively on external services: services that are developed and maintained either within the parent company, but outside of the application itself, or by a third party entirely. Although these architectures have significant advantages, and are generally considered good practice, they pose a challenge when it comes to software testing.

When an application is entirely contained within itself, end-to-end testing is relatively simple, since every part of the application is there for you to test. Integration testing is often necessary in addition, but only as a last step after the bulk of testing has been carried out. However, this falls apart when the application is not a self-contained unit, as in service-oriented architectures. External services suffer from downtime, whether scheduled or unscheduled, and third-party services may incur fees with each use. When your application only relies on one or two of these services, this is not usually an issue. More than that, and it can easily become a big problem. Perhaps the worst-case scenario is in a microservices architecture: imagine an application that connects with dozens of different services, all of which must be online and functional for your tests to run. This can slow down the testing process immensely, create a bottleneck in your development lifecycle, and make continuous testing impossible.

Service virtualisation offers a solution. It allows you to create virtual services that imitate the behavior of real services, but without any of the functionality. These can then be managed and maintained internally and used for testing purposes, intercepting requests to the real service and issuing appropriate, pre-recorded responses within your testing environment. Since they are internal rather than external, they are under your control, and you can make sure they are available for testing as and when you need them. Moreover, since they do not have any real functionality, they are very lightweight and are not prone to breakage. The end result is that by creating virtual copies of the services your application interacts with, you can test the application in isolation of any external services. This also means that any fault your tests find is almost certainly within the app itself, rather than a service.

This leads into an additional point. When using service virtualisation to test your application in isolation of any external services, it behooves you to test the behavior of those external services. This is where API testing comes in, and although it is not the focus of this report, it is still worth mentioning, as it is a necessary component for testing applications within many service-oriented architectures (it is also featured in several leading service virtualisation solutions). Between service virtualisation and API testing, you can test both your app and your services in isolation of each other. This should, ultimately, make integration (and integration testing) fast and easy.

Figure 1: The highest scoring companies are nearest the centre. The analyst then defines a benchmark score for a domain leading company from their overall ratings and all those above that are in the champions segment. Those that remain are placed in the Innovator or Challenger segments, depending on their innovation score. The exact position in each segment is calculated based on their combined innovation and overall score. It is important to note that colour coded products have been scored relative to other products with the same colour coding.



In this report we compare and evaluate several best of breed, enterprise-level service virtualisation solutions. It should be noted that there is a significant split in the service virtualisation market, between the aforementioned enterprise products on the one hand, and a variety of smaller-scale solutions on the other. The former tend to be very full-featured and offered as part of a suite of testing products, and are universally proprietary, while the latter are more often than not open source, and focus on providing a minimalistic and lightweight (both in terms of footprint and features offered) service virtualisation solution. The products on either side of this split cater to very different use cases, to the point that we have concluded that they are not truly comparable: any metric we chose to compare them by would do one side or the other a disservice. Accordingly, we have chosen to focus this report exclusively on service virtualisation products that are fully-featured and enterprise-ready.

Market Trends

The service virtualisation space, at least as far as the enterprise vendors are concerned, is relatively mature, and the majority of products have been tried and tested over a number of years. In this sense, you can be fairly sure that, whichever vendor you choose, you'll be provided with a solution that will be competent at creating and managing virtual services. The real differentiators are in the additional features provided on top of that, as well as the depth of functionality for service creation and management that is on offer. Innovation is not uncommon, but disruption is occurring primarily in the form of the creation of a wide variety of (usually) open source tools that offer basic service virtualisation capabilities. These solutions allow even very small organisations access to service virtualisation but lack the scalability and additional features offered by the solutions covered in this report. Consequently, they serve a very different use case, and for the most part we do not feel they are in real competition with enterprise service virtualisation products.

The exception to this is when it comes to the freemium products and licensing models that several enterprise vendors are beginning to offer in addition to their standard proprietary offerings. These tend to be more flexible in pricing – they are usually priced using some variety of service model – and more lightweight than their older, more expensive siblings. This makes them more suited to smaller organisations, or to an individual team within an enterprise organisation, that do not require all the bells and whistles present in the original product. However, this does mean that they are competing directly with the open source products and smaller vendors mentioned in the previous section. This is no bad thing, as these freemium offerings are generally going to be more well supported than any given open source product. Whether this support is worth the additional cost is left to the reader's discretion.

Enterprise service virtualisation products are increasingly offered as part of a suite of products, usually focused around testing (and often a specific form of testing, at that, such as performance or functional testing). This has some significant benefits when it comes to deploying and integrating an end-to-end testing solution, assuming you buy into the suite as a whole. This is not a given: testing environments are often built organically and from the bottom-up, meaning that an organisation buying into an entire suite at once is unlikely. Some service virtualisation vendors even offer their products exclusively as part of a suite, meaning they are not available as a standalone product at all. We cannot help but feel that by using this approach these vendors are narrowing their customer pool unnecessarily.

In terms of the products themselves, the most notable and interesting trend is that of API testing. As mentioned in our introduction, API testing is extremely complementary to service virtualisation, as one allows you to test your application in isolation, while the other allows you to test its interactions with other services. It's clear that the vendors in this report have noticed this, as they all offer API testing in one form or another. In fact, at least one vendor puts it on effectively even footing with service virtualisation. An additional technological trend is the increasing tendency for service virtualisation solutions to support the IoT (Internet of Things). Given its increasing prevalence, this seems wise. A number of products in the space also support containerisation, typically using technologies such as Docker or Kubernetes. This may be useful if you have a microservices architecture, as they often utilise containers extensively.

Although not strictly part of the service virtualisation space, it's also worth noting that service-oriented and, particularly, microservice architectures, although not a new concept, have been growing and continue to grow in popularity. This has given organisations additional impetus to invest in service virtualisation, as the number of services in these architectures is immensely larger than it would be otherwise. In our opinion, these architectures are often good development practice, and are unlikely to go away. Thus, we believe that

service virtualisation will become increasingly vital (perhaps even necessary) for many organisations in the coming years.

Vendors

In this report, we have evaluated several best of breed service virtualisation offerings that are suitable for deployment at the enterprise level. To wit, and in no particular order, these are CA Service Virtualisation, Parasoft Virtualize, IBM Rational Test Virtualization Server, SmartBear ServiceV Pro and Tricentis Tosca Orchestrated Service Virtualization.

In addition, a handful of enterprise service virtualisation products have not been covered, including Cavisson Systems NetOcean, Micro Focus Service Virtualization and Cognizant SmartStub. The reasons for this vary. Cognizant operates primarily as a service, and accordingly SmartStub is not available as a product. Cavisson has not been included because of what we see as major flaws in the product, to the point that we could not in good conscience describe it as best of breed. Finally, as far as Micro Focus is concerned, we have repeatedly asked for information regarding their product and failed to get a useful response. From a historical perspective, this is unusual for the company. However, a number of key people appear to have recently left and have not been replaced. This is, perhaps, a result of difficulties stemming from their recent acquisition of HPE. Until this situation is resolved, we cannot recommend the adoption of Micro Focus products.

Scoring

To score the various vendors/products discussed in this report we have used the following metrics:

- **Service creation and maintenance –** how easy is it to create and maintain virtual services with the product? This includes the range of methods on offer for carrying out these processes (for example, in many products you can create a virtual service by either monitoring a real service or by importing an existing API definition) as well as how much of these processes can be automated.
- **Requests and responses –** how much functionality the product provides for intercepting a request and sending out an appropriate response. A high scoring product in this category should be able to match requests to responses in a variety of ways, as well as respond in an intelligent fashion. This might include, for example, state handling, parameterisation, or desensitisation.
- **Service management –** the ease and efficiency of managing virtual services within the product. This includes any advanced configuration options on offer, such as passthrough to the original service and user configurable response time.
- **Application testing –** how well the solution supports application testing and testing products, particularly within its own testing suite (assuming it belongs to such a suite). This will usually cover functional testing, but in some cases performance testing may be relevant as well.
- **API testing –** the effectiveness, comprehensiveness and sophistication of API testing that the product offers.
- **Deployment –** the variety of deployment, onboarding and pricing options made available for the solution. One of the bigger considerations here is the existence (or not) of a freemium solution, and how well integrated it is with the main product. For example, how difficult is it to upgrade from a freemium to a proprietary solution?
- **Ease of use –** how easily the solution can be used. This is primarily from a user interface perspective, but also takes into account the amount of work a user has to put in to get things done. Therefore, the amount of automation provided also plays a significant factor.
- **Integration –** the quantity and variety of message formats, protocols (particularly IoT specific protocols), third-party products and so on that the solution will integrate with, as well as the quality (where applicable) of said integration. Note that this does not include integration with testing products, as they are covered by their own category.

We recognise that some aspects of these requirements will be more important for some users than others. So, while all of the scores for individual products are included in the detailed descriptions that follow later, the tables below represent the comparative scoring for each of the areas set out above. Note that each score is out of 5 but, unlike Amazon or Trip Advisor, it is impossible to score 5 on any topic. A score of 5 would represent a *"perfect"* product at this time. As we do not believe in perfection no product can be awarded a maximum score.

The scores below are solely related to the products under evaluation. However, the positioning on the Bullseye diagram, as well as the *"mutable"* diagrams accompanying each vendor evaluation, also encompasses company issues such as support, geographic presence, stability and so on; as well as factors like innovation and the ability to support moves towards a data-driven enterprise.

Scores for service virtualisation solutions

VENDOR	API TESTING
SmartBear	4.5
Parasoft	4.5
IBM	4.5
Tricentis	4.5
CA	3.5

VENDOR	INTEGRATION
IBM	4.5
Parasoft	4.5
CA	4.0
SmartBear	4.5
Tricentis	3.0

VENDOR	APPLICATION TESTING
CA	4.5
Parasoft	4.5
IBM	4.0
Tricentis	4.0
SmartBear	3.0

VENDOR	REQUESTS AND RESPONSES
CA	4.0
Parasoft	4.0
IBM	4.5
Tricentis	4.5
SmartBear	3.0

VENDOR	DEPLOYMENT
Parasoft	4.0
CA	4.5
IBM	4.5
SmartBear	3.0
Tricentis	3.0

VENDOR	SERVICE CREATION AND MAINTENANCE
IBM	4.5
CA	4.0
Parasoft	4.0
Tricentis	4.5
SmartBear	4.5

VENDOR	EASE OF USE
Tricentis	4.5
SmartBear	4.0
CA	4.5
Parasoft	4.5
IBM	4.5

VENDOR	SERVICE MANAGEMENT
CA	4.0
IBM	4.0
Parasoft	4.0
SmartBear	4.5
Tricentis	4.5

Conclusion

As we mentioned at the beginning of this report, all of these products are best of breed. However, they each have their own areas of specialty: CA and Parasoft both provide a broad range of capabilities and functionality, especially when it comes to application testing; SmartBear is unparalleled when it comes to APIs and API testing; Tricentis is very easy to use and integrates particularly well with its parent testing suite; and finally, IBM provides an innovative and effective way to create and maintain virtual services. The bottom line is that they are all good solutions, and your preference will most likely depend on which of the aforementioned strengths most appeals to you.

About the authors

DANIEL HOWARD
Senior Researcher

Daniel started in the IT industry relatively recently, in only 2014. Following the completion of his Masters in Mathematics at the University of Bath, he started working as a developer and tester at IPL (now part of Civica Group). His work there included all manner of software and web development and testing, usually in an Agile environment and usually to a high standard, including a stint working at an 'innovation lab' at Nationwide.

In the summer of 2016, Daniel's father, Philip Howard, approached him with a piece of work that he thought would be enriched by the development and testing experience that Daniel could bring to the table. Shortly afterward, Daniel left IPL to work for Bloor Research as a researcher and the rest (so far, at least) is history.

Daniel primarily (although by no means exclusively) works alongside his father, providing technical expertise, insight and the 'on-the-ground' perspective of a (former) developer, in the form of both verbal explanation and written articles. His area of research is principally DevOps, where his previous experience can be put to the most use, but he is increasingly branching into related areas.

Outside of work, Daniel enjoys latin and ballroom dancing, skiing, cooking and playing the guitar.

Bloor overview

Technology is enabling rapid business evolution. The opportunities are immense but if you do not adapt then you will not survive. So in the age of Mutable business Evolution is Essential to your success.

We'll show you the future and help you deliver it.

Bloor brings fresh technological thinking to help you navigate complex business situations, converting challenges into new opportunities for real growth, profitability and impact.

We provide actionable strategic insight through our innovative independent technology research, advisory and consulting services. We assist companies throughout their transformation journeys to stay relevant, bringing fresh thinking to complex business situations and turning challenges into new opportunities for real growth and profitability.

For over 25 years, Bloor has assisted companies to intelligently evolve: by embracing technology to adjust their strategies and achieve the best possible outcomes. At Bloor, we will help you challenge assumptions to consistently improve and succeed.

Copyright and disclaimer

Bloor Research International Ltd
20–22 Wenlock Road
LONDON N1 7GU
United Kingdom

Tel: **+44 (0)20 7043 9750**
Web: **www.Bloor.eu**
Email: **info@Bloor.eu**