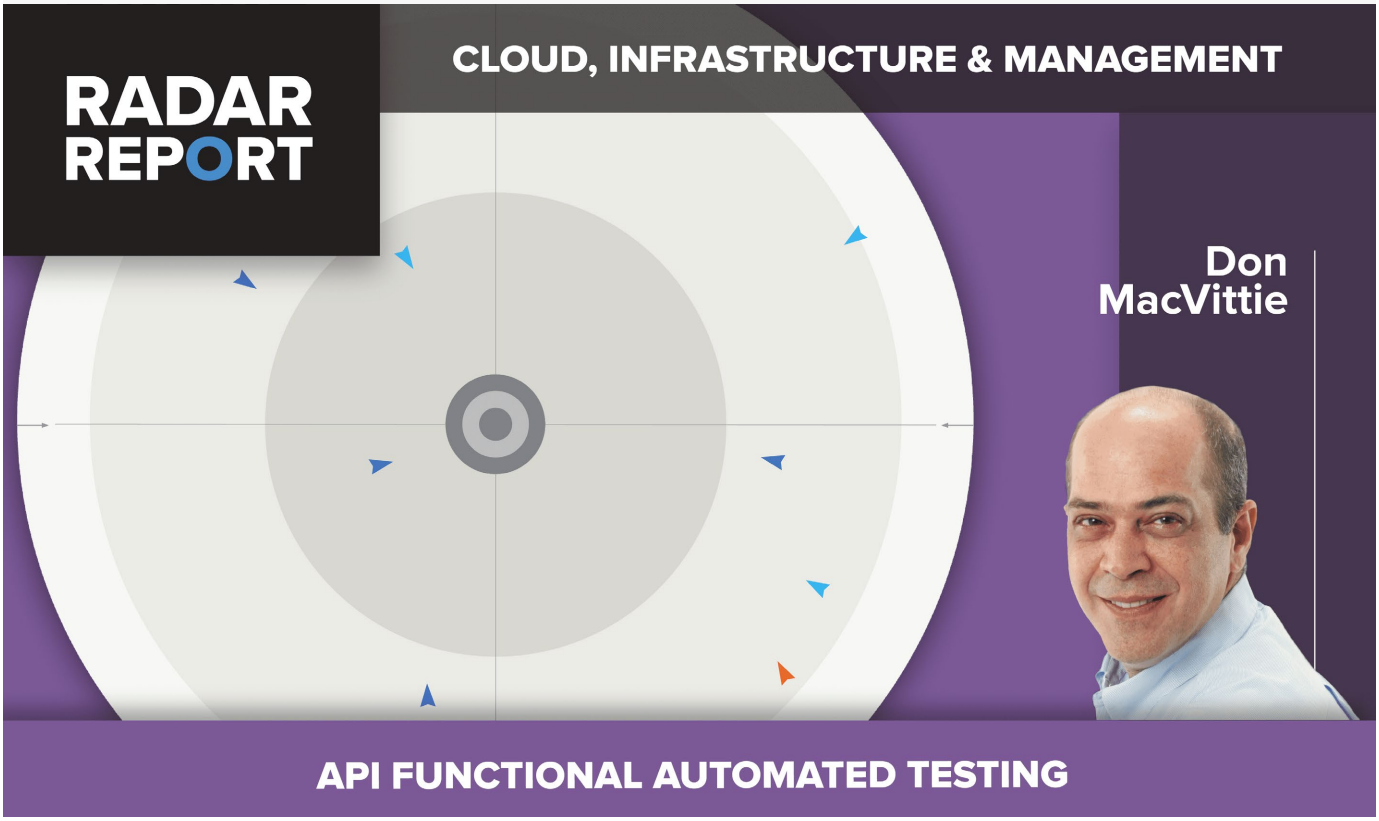


Don MacVittie
Dec 11, 2023

GigaOm Radar for API Functional Automated Testing v2.01

Table of Contents

1. Executive Summary
2. Market Categories and Deployment Types
3. Decision Criteria Comparison
4. GigaOm Radar
5. Solution Insights
6. Analyst's Outlook
7. Methodology
8. About Don MacVittie
9. About GigaOm
10. Copyright

1. Executive Summary

Functional testing has been a standard part of application development for a very long time. However, traditional functional methods can't be applied to many current projects or products, which are increasingly composed solely of APIs performing specific functions or accessing specific data, because that testing is based on the use of UIs or subroutines as entry points, neither of which can work for APIs.

Automation has long been key to the adoption of testing software. Slowing deployments while a test team runs testing manually could never be an option for most busy IT departments. Such constraints became even more of a burden as agile development and DevOps increased the cadence of software delivery and, by extension, increased the burden of testing. Fortunately, now there are tools that can run test suites automatically, filter results without user input, and offer results that are manageable and actionable for both testing and development teams.

The growth in both API usage and DevOps drove the need for automated API functional testing. With increasing numbers of applications turned out at an ever-quicker rate, the need for quality assurance of all kinds is also growing. The best tools evaluated in this analysis can offer testing, support for shifting testing and resolutions left, and functionality unique to API testing.

An API client that facilitates test development, for example, is relatively standard in these tools but not necessary in other functional testing toolsets. This feature can help developers and testers to develop tests interactively and save those tests for repeatable testing.

Tools in this market facilitate test-driven development (TDD) for organizations wishing to use this methodology. The tools enable developers to use the API contract as the starting point for tests, then add test functionality as the project progresses and the code behind the contract is developed. While a full suite of functional testing tools is not necessary to perform TDD, they are useful tools to have.

Our reasoning about what makes a product in this space innovative, and what constitutes a platform, has changed a bit since our last iteration. Our use of the "Maturity" designation more heavily relies upon release cadences and product stability than in the last iteration. Likewise, nearly every tool in this analysis uses AI at this point, so we've reduced its bearing on the "Innovation" rating. On the other axis, declaring a group of products a platform does not make it so, and we were more discerning as to what we accepted as a true Platform Play.

Application quality is important to all organizations, and today, applications are often built from a collection of API calls with user interfaces overlaid to enable them on different targets. API functional automated test tools help ensure the quality of the underlying APIs and to increase stability and quality of the application itself, regardless of where it is deployed.

This is our second year evaluating the API functional automated testing space in the context of our Key Criteria and Radar reports. This report builds on our previous analysis and considers how the market has evolved over the last year.

This GigaOm Radar report examines nine of the top API functional automated testing solutions in the market and compares offerings against the capabilities (table stakes, key features, and emerging features) and non-functional requirements (business criteria) outlined in the companion Key Criteria report. Together, these reports provide an overview of the category and its underlying technology, identify leading API functional automated testing offerings, and help decision-makers evaluate these solutions so they can make a more informed investment decision.

GIGAOM KEY CRITERIA AND RADAR REPORTS

The GigaOm Key Criteria report provides a detailed decision framework for IT and executive leadership assessing enterprise technologies. Each report defines relevant functional and non-functional aspects of solutions in a sector. The Key Criteria report informs the GigaOm Radar report, which provides a forward-looking assessment of vendor solutions in the sector.

2. Market Categories and Deployment Types

To help prospective customers find the best fit for their use case and business requirements, we assess how well API functional automated testing solutions are designed to serve specific market segments and deployment models (**Table 1**).

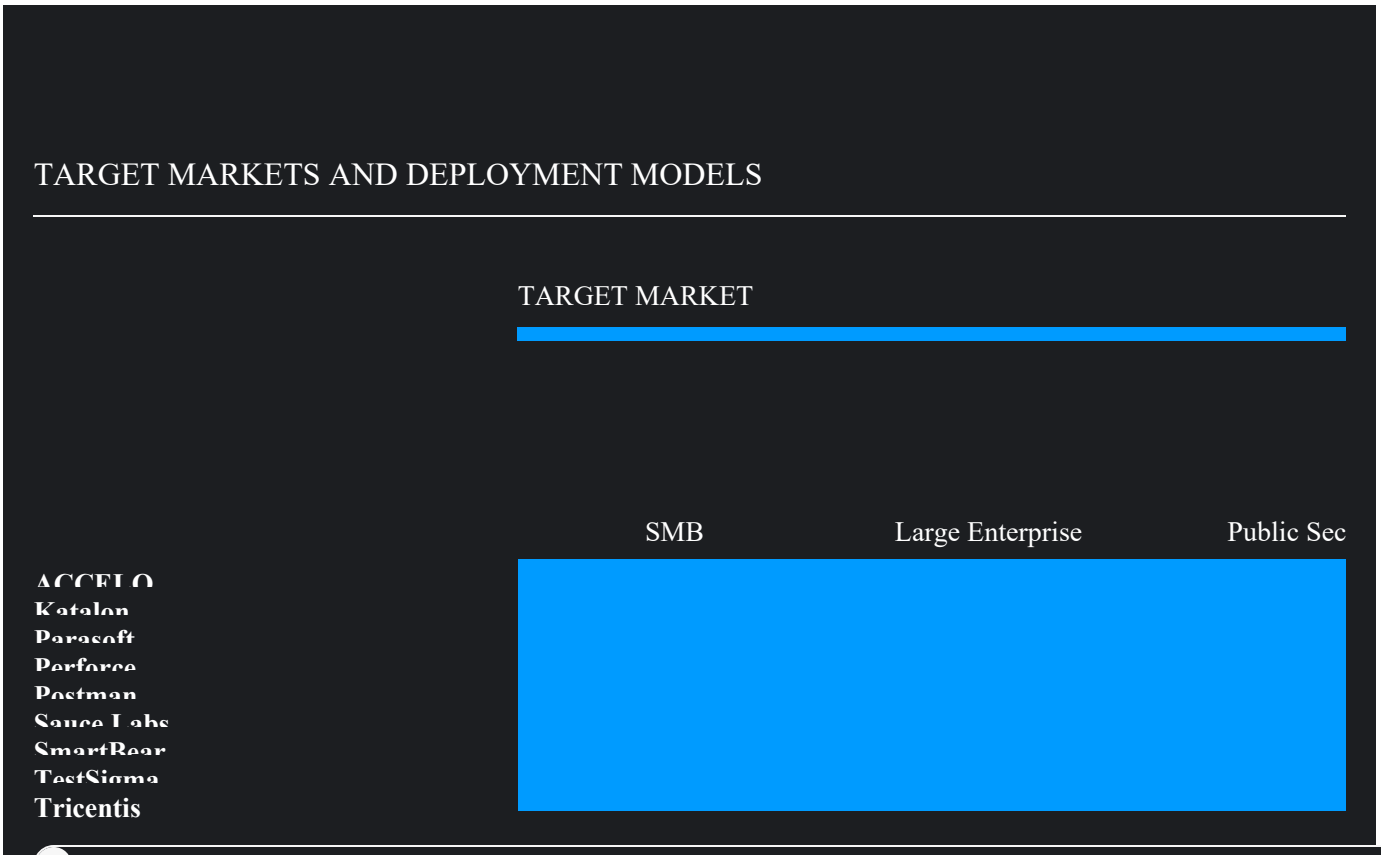
For this report, we recognize the following market segments:

- **Small-to-medium business (SMB):** In this category, we assess solutions on their ability to meet the needs of organizations ranging from small businesses to medium-sized companies. Also assessed are departmental use cases in large enterprises, where ease of use and deployment are more important than extensive management functionality, data mobility, and feature set.
- **Large enterprise:** Here, offerings are assessed on their ability to support large and business-critical projects. Optimal solutions in this category will have a strong focus on flexibility, performance, data services, and features to improve security and data protection. Scalability is another big differentiator, as is the ability to deploy the same service in different environments.
- **Public sector:** Very similar to large enterprises, public sector users are generally more concerned with compliance and reporting. For many public sector customers, data and information security is also paramount—even taking priority over business value in some scenarios.

In addition, we recognize the following deployment models:

- **Local:** The solution is available as software or containers for local management. The big advantages of this type of solution are the ability to run limitless tests without concern for usage fees and the ability to test local APIs without requiring network connectivity.
- **Public cloud:** These solutions are meant to be installed in the cloud, allowing customers to build cloud, hybrid, or multicloud infrastructures. They allow customers more flexibility, and users typically have more control over the entire stack with regard to resource allocation and tuning.
- **SaaS:** These solutions are implemented via a centralized system with interfaces (normally API-based) to interconnect with customer's data. UIs and tools in the SaaS application enable users to test, and testing can target any accessible API. SaaS is more easily scaled up and down than other options.
- **Variant self-managed:** Install packages enable the installation of the solution at the user's preferred location.

Table 1. Vendor Positioning: Target Market and Deployment Model



Source: GigaOm 2023

Table 1 is evaluated in a binary yes/no manner and do not factor into a vendor's designation as a Leader, Challenger, or Entrant on the Radar chart (**Figure 1**).

"Target Market" reflects which use cases each solution is recommended for, not simply whether it can be used by that group. For example, if it's possible for an SMB to use a solution but doing so would be cost-prohibitive, that solution would be rated "no" for that market segment.

3. Decision Criteria Comparison

All solutions included in this Radar report meet the following table stakes—capabilities widely adopted and well implemented in the sector:

- Import functionality
- Response validation
- Record and replay test runs
- Scripting languages support
- Authentication support

Tables 2, 3, and 4 summarize how each vendor included in this research performs in the areas we consider differentiating and critical in this sector. The objective is to give the reader a snapshot of the technical capabilities of available solutions, define the perimeter of the relevant market space, and gauge the potential impact on the business.

- Key features differentiate solutions, outlining the primary criteria to be considered when evaluating an API functional automated testing solution.
- Emerging features show how well each vendor is implementing capabilities that are not yet mainstream but are expected to become more widespread and compelling within the next 12 to 18 months.
- Business criteria provide insight into the non-functional requirements that factor into a purchase decision and determine a solution's impact on an organization.

These decision criteria are summarized below. More detailed descriptions can be found in the corresponding GigaOm report, "Key Criteria for Evaluating API Functional Automated Testing Solutions."

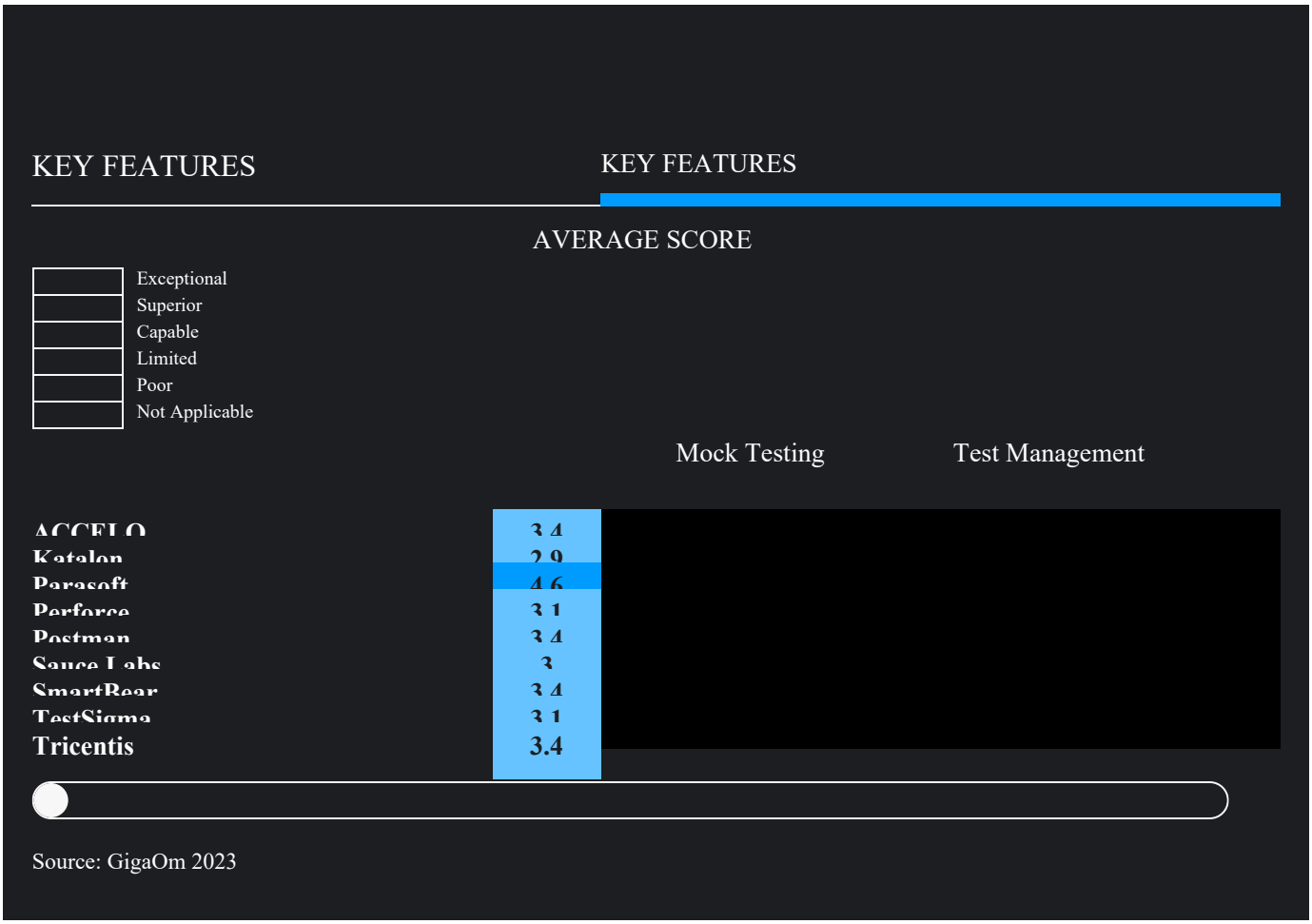
Key Features

- **Mock testing:** API development requires both a client and a server to be useful. The ability to answer API calls without writing code or to make an API call without processing the results allows developers to move forward with development when either a client or a server is not present. Mocking simulates the missing side of an API exchange.
- **Test management:** The ability to manage the breadth of tests required to validate APIs is mandatory. Creating a one-time-only test is easy in any of these products, but the ability to make changes, execute as needed, and reuse tests is central to professional solutions.
- **Mock data generation:** APIs, and by extension API tests, require data, but data is not always available. The ability to create fake data or mask existing valid data is imperative to all testing, but such data has often been generated by hand. We are past that point; large, complex projects require enough realistic data to exercise them, and this data must be generated automatically.
- **API client:** When developing APIs and the tests that an API needs to pass, using an API client is the standard method. The ability to dynamically modify tests, run them, and look at the results is as imperative as being able to write code, run it, and look at the results.
- **Integrations:** Testing that integrates with CI/CD regimes helps ensure consistency of testing. Automated testing is most effective when tests spawn automatically as needed throughout the software design lifecycle (SDLC). Integration into core CI/CD tools like Jenkins, Atlassian Bamboo, and GitLab is the way that test suites become

consistently applied, no matter the time pressures.

- **Functional test parameters:** While APIs are generically available from a client development language perspective, the fact is that developers also use some data, such as the user agent and the location of the caller's IP address, throughout their infrastructures and code bases. The ability to modify these fields and test results is important to overall API functional testing. Look for tools that offer support for data from the interface, data exchange, and environment, both on call and on response.
- **Low-code/no-code testing:** There are two scenarios where low-code and no-code capabilities become important: when the developer wants to create a thorough set of tests but lacks the time to write each one; and when testing is turned over to other people. The ability to drag-and-drop calls enables those users to augment the rudimentary test cases that developers contributed.

Table 2. Key Features Comparison



Emerging Features

- **Call chaining:** Call chaining involves nested API calls as data moves through the system. If an API is built to be an underlying bit of the application infrastructure and it is called from different places with different inputs, understanding those relationships and seeing, for example, how data that came into API #1 eventually gets to API #2, which consumes that data, is important to understanding testing—particularly of data values but also of functionality based upon those values.
- **Environment testing:** APIs do not run in a vacuum, and tools are beginning to address things that are normally outside the control of the API developer, like environment variables and runtime feature flags. This capability means being able to answer questions like "Will this test fail if there are changes in the environment (OS, hosting, environment variables, and the like)?" And if so, "Can we find the specific cause of the failure, even if it is not in the code under test?" This type of functionality will grow in use as applications and the teams that build and run them become more platform diverse.
- **AI/ML for results filtering:** The most significant inhibitor of adoption for testing has been the large volume of tests and results that must be processed. If there's no time to evaluate testing results, then testing is not worthwhile.

This emerging feature looks at the ways AI is being used to filter results to make what test teams must evaluate a manageable subset of all the results of the overall tests performed.

Table 3. Emerging Features Comparison



Business Criteria

- **Flexibility:** Flexibility encompasses the ability of a tool to meet a range of use requirements and use cases. This includes a number of factors, such as whether a tool can be deployed in many types of environments, integrate with existing business systems (with minimal work on the part of the customer), offer a range of license and subscription options, provide broad support for the API standards an organization is using and is likely to use in the future (REST APIs, SOAP, gRPC, and GraphQL), and support a number of report formats as well (XML, JSON, plain text, YAML, and so on).
- **Scalability**: This criterion looks at a test suite's ability to scale up or out, depending on the needs of the customer, because no matter how thorough and complex the testing is, it needs to finish in a timely manner. Keep in mind that scaling at runtime requires a totally different architecture, so the scalability of toolsets with a runtime component will also need to be considered.
- **Ease of Use**: Ease of use considers the factors that make API functional automated testing easier (IDE integrations, the API test client, and so on), as well as how easily tools slide into developers' workflows. Tools should make it easy for various stakeholders to find information, provide a variety of reports, and offer a range of test generation options (scripts, UI, drag-and-drop interfaces, and so on).
- **Cost:** This criterion encompasses various factors, including the cost to purchase or license the tool, the resources required to host a test environment, and the ability of prospective customers to perform cost comparison (transparent versus hidden pricing). While some costs are likely to be low (such as for hosting the API UI and the report interface), others may well be more significant, such as the cost of any test runners required and the cost of analyzing results—which might include how easily results can be filtered and prioritized and the time required to build tests and evaluate those results.
- **Enablement:** This criterion looks at how a given tool enables test teams to reuse tests. It also considers how results

filtering is implemented and used within the product, enabling test teams to do more if their attention isn't overwhelmed by excessive data. Keep in mind, however, that results filtering can also cause false negatives if implemented too liberally. Most test teams would rather spend a bit of extra time evaluating a suspicious result than save time but have errors slip through. Evaluators should consider how tools strike this balance and what adjustments are available to configure the level of filtering and/or reporting.

Table 4. Business Criteria Comparison



4. GigaOm Radar

The GigaOm Radar plots vendor solutions across a series of concentric rings, with those set closer to the center judged to be of higher overall value. The chart characterizes each vendor on two axes—balancing Maturity versus Innovation and Feature Play versus Platform Play—while providing an arrowhead that projects each solution's evolution over the coming 12 to 18 months.

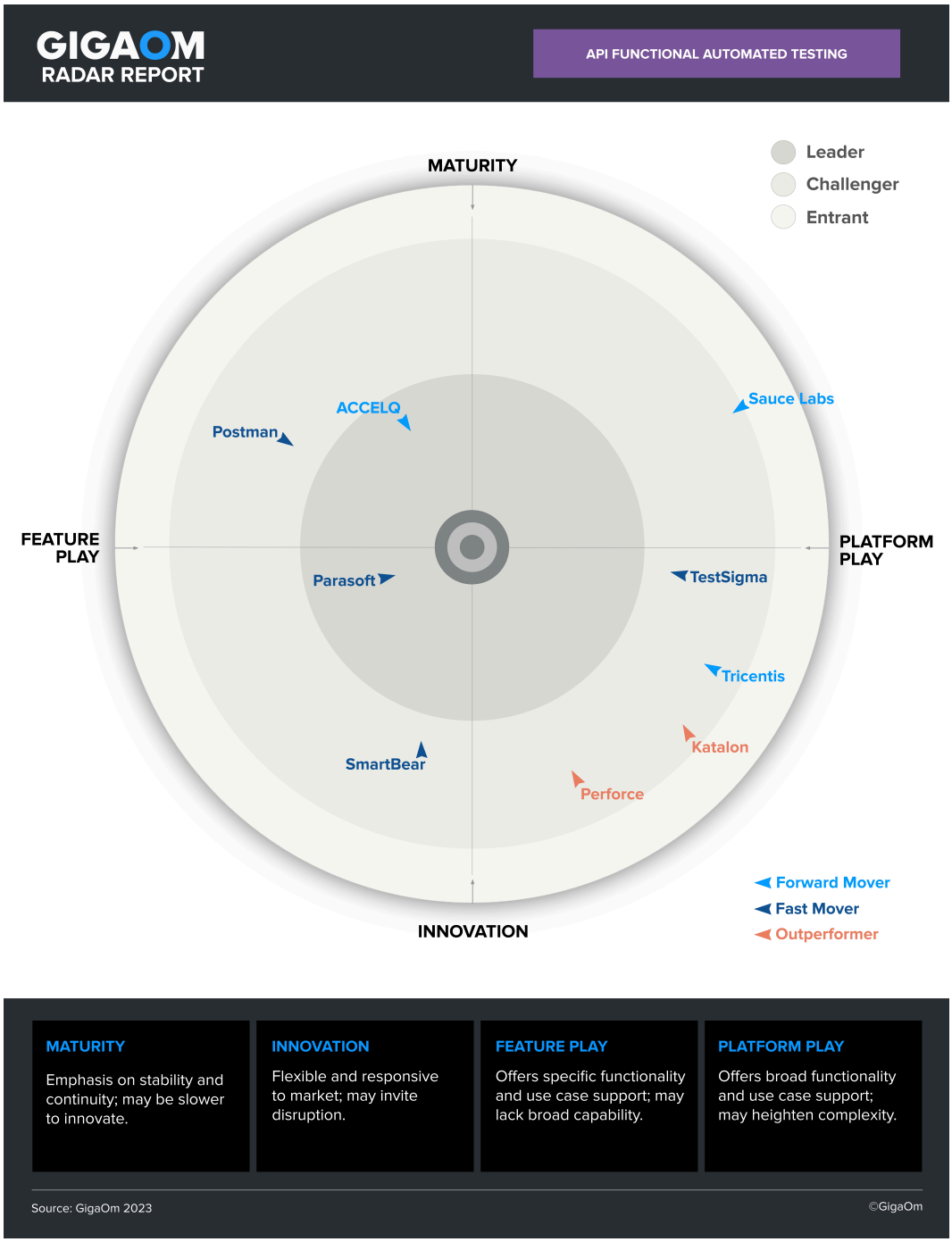


Figure 1. GigaOm Radar for API Functional Automated Testing

As you can see in the Radar chart in Figure 1, vendors run the gamut from Feature Play to Platform Play and from Innovation to Maturity. In this iteration of the Radar, we tightened up what constitutes a Platform Play, and that change impacted the placement of a couple of products. Tools that are offered as a single product with a single installation but are part of a larger offering are pure Platform Plays, while tools that are a collection of apps for solving different pieces of the puzzle that can be purchased separately are Feature Plays.

On the Innovation side, we find vendors whose tools have recently undergone a greater breadth of changes and have many more changes in their roadmaps. Vendors developing really quickly are highly innovative, while those that are focused on maintaining a stable product with few new features being introduced belong in the Maturity half of the chart. It is a testament to the diversity of the market that these products are distributed broadly, both among all four quadrants and in terms of their distance from center. This market is seeing more and more new features and technology, which makes vendors cross over between Innovation and Maturity more often as they implement new capabilities, then pause to assess the quality and popularity of those changes. We expect steady improvement all around as time goes on.

TestSigma is a new addition to this iteration of the Radar and debuts as a Challenger. The product is on par with other vendors in the space and more focused on AI than most, giving it an Innovation position. It is a Platform Play for those needing a broader solution.

In reviewing solutions, it's important to keep in mind that there are no universal "best" or "worst" offerings; there are aspects of every solution that might make it a better or worse fit for specific customer requirements. Prospective customers should consider their current and future needs when comparing solutions and vendor roadmaps.

INSIDE THE GIGAOM RADAR

To create the GigaOm Radar graphic, key features, emerging features, and business criteria are scored and weighted. Key features and business criteria receive the highest weighting and have the most impact on vendor positioning on the Radar graphic. Emerging features receive a lower weighting and have a lower impact on vendor positioning on the Radar graphic. The resulting chart is a forward-looking perspective on all the vendors in this report, based on their products' technical capabilities and roadmaps.

Note that the Radar is technology-focused, and business considerations such as vendor market share, customer share, spend, recency or longevity in the market, and so on are not considered in our evaluations. As such, these factors do not impact scoring and positioning on the Radar graphic.

For more information, please visit our Methodology.

5. Solution Insights

ACCELQ, Automate API

Solution Overview
ACCELQ is a testing software company that offers an array of products under the ACCELQ Unified umbrella. Aside from API testing, the company also has solutions for Web, Mobile, Desktop, Packaged Application testing and a test management tool. All are available either separately or bundled.

ACCELQ Unified encompasses the breadth of web and mobile application testing, but our analysis focuses on ACCELQ API and ACCELQ Manual, the two products that cover API testing. These solutions are offered as either SaaS or a completely local installation, allowing organizations to choose the deployment model that best suits their environment and security posture. ACCELQ comes at the problem of API testing from a test team perspective, offering more in test management and running than in TDD, though as with every vendor, the way it approaches the problem does not limit its ability to solve the problem, but is descriptive of its particular solution.

Strengths
ACCELQ scores well on our integrations key feature, with the tools needed in the build chain, security, and notifications areas covered.
Other vendors have some of each of these integrations, but ACCELQ tailors them to meet enterprise needs. This also boosted its ease of use business criterion score, as integrations make it easier to both use and deploy the tool. Test management is another area of strength for ACCELQ, with a dedicated product providing overall test management that includes all that users look for. From test planning to organization and execution and issue reporting, ACCELQ covers management with its ACCELQ Manual Test product.

Challenges
While ACCELQ is able to create and switch to and from mock server instances, the process is nowhere near as advanced as the best tools in the market. While most tools offer traffic monitoring to gather mocking information, ACCELQ requires users to enter that information. Considering that a microservices architecture can have hundreds of endpoints and thousands of potential interfaces, we would like to see more automation for this feature. We also prefer that client stubs can be generated via the same toolset, but ACCELQ, like most vendors, has chosen to leave that to API tools. Like nearly all other vendors, ACCELQ uses AI for test generation, but the ability to filter results with AI— basically rudimentary call chain analysis—is limited. This impacted all such products' scores.

Purchase Considerations
This solution can be implemented locally or as SaaS. Just as this is a purchase consideration, it is also an implementation choice. Users should make certain they know how the tool best fits their environment.

For organizations looking for a tool aimed more at testers than at the development process, ACCELQ should be a consideration. This solution has little for developers that have to test as they build solutions, but it has an exhaustive set of features that can help test teams ensure the quality of the code. If the test team is experienced, this toolset will serve the important purposes, and other needs might be met by the extended toolset.

Radar Chart Overview
ACCELQ's API testing solution is a point product zeroed in on the problem of APIs, making it a functional Feature Play solution more than a platform. However, it's closer to the vertical axis because the other products in the line can work with API testing, so it leans toward the platform side. At this time, ACCELQ is not working on innovating the product and is instead maintaining a stable toolset for test teams to use. Our analysis rates it as a Leader not because of a couple of standout features, but because the breadth of the product's capabilities manages the tasks at hand very well.

Katalon, The Katalon Platform

Solution Overview
Katalon is a testing software company with the Katalon Platform its only product. The Katalon Platform is a testing solution that includes functional testing for modern applications, with Katalon API a subset of the platform. The system consists of several pieces that can be deployed locally or in a public cloud and a SaaS component that can be multitenant or private SaaS. The local tools notably include Katalon Studio for test development, Katalon Runtime Engine for test execution, and Katalon TestOps for Orchestration and results analysis. The product is very much a broader testing tool that performs API testing well.

Strengths
The Katalon Platform offers both AI test generation and visual test development. This makes its low code/no code score one of the highest in the market. Similarly, the breadth of test management options and the tools to assist in reusing tests earned Katalon one of the better scores on our test management criterion.

Challenges
Where mocking is concerned—both API virtualization and test data generation— Katalon falls a bit short. For API virtualization, it focuses on the testing and leaves it to other tools to virtualize. If having a server to test client code against is important, another tool will be needed. For data generation, it's able to make use of other generation tools and keep that data but does not generate or mask data.

Purchase Considerations
At this time, there is no web-based API test editor. While Katalon studio is full featured and easy to use, those who prefer to do their test creation and management in a browser should be aware of this fact.

Katalon will most appeal to organizations looking for a complete application testing solution that comes from a testing perspective and has tools to make less technical users productive.

Radar Chart Overview
The Katalon Platform is close to a true platform: one product with license differences that reflect only how much API testing can be done. Extra charges apply for non-API functionality, and the tool requires a separate install of Katalon Studio. Since the introduction of the Katalon Platform, releases have been frequent, with new functionality introduced regularly. This led us to position the tool on the Innovation half of the Radar.

Parasoft, Parasoft Continuous Quality Platform

Solution Overview
Parasoft is an application testing company with a focus on the developer side. Like most vendors today, it aims to fulfill the needs of both test and development teams but focuses more on development, including language-specific test tools.

The Parasoft Continuous Quality Platform is a collection of tools that together solve modern testing problems, including of APIs. Its core elements include SOATest for API testing, Parasoft Virtualize for service virtualization, Parasoft Continuous Testing Platform for test management, and Parasoft Development Testing Platform for results analysis and reporting. The solution can be installed on any platform that an enterprise is likely to use, from Windows and Linux locally to cloud instances.

Strengths
Parasoft offers some of the best data creation capabilities in the market, with masking and generation that includes automated detection of the data model to create a test data model that can then be customized. Whatever an organization's test data needs, the Parasoft solution provides. Parasoft also offers a solid code solution, which is important because this is an area where many tools with developer roots lag behind. We consider the quality of this implementation to be indicative of the steps Parasoft has taken to provide a test team focus without losing the benefit of its developer roots.

Challenges
Parasoft's results filtering and management are some of the best on the market, but the solution does not as yet offer AI-assisted issue management or filtering. We expect this to change relatively soon, but it is currently the weakest part of the product. Similarly, while most developer-focused tools have API call stub generation to complement API virtualization, this functionality is absent in the Parasoft offering, so customers who hope to speed development by generating interface code should consider OpenAPI tools for that functionality.

Purchase Considerations
Parasoft is a local install solution, so it's probably not the best option for organizations that prefer SaaS.

Parasoft is one of the few products that truly caters to both development and testing teams. Organizations can purchase the tool for either use and will get the most from their investment by standardizing on Parasoft for testing APIs (and possibly more) across the SDLC.

For organizations looking to merge disparate testing environments, BlazeMeter's ability to make use of other test tools might also be appealing as a step in that process.

Radar Chart Overview
Parasoft is sold as a platform, but like so much of the market these days, it's actually a collection of tools that work together. It is better described as a portfolio, so we've placed it on the Feature Play side. Moreover, the amount of change in the tools is less than in some competitors' releases. While updates are regular, new features are slower to appear, and stability is important to Parasoft's enterprise customer base. Both of these axis positions are a change from our last iteration of this report.

Perforce, BlazeMeter

Solution Overview
Perforce is a DevOps tools company that differs from many of the vendors in this space in that the company offers both developer- and test-oriented tooling. Both offerings are expansive, but this analysis focuses on the testing tools.

BlazeMeter is Perforce's SaaS application testing solution. For this analysis, we focused on the product's API testing and data mocking capabilities. When considering TCO and ROI, evaluators should recognize that this product consists of more in a single purchase than what we evaluate, and more than most competitors offer in the core API testing product. Local agents are available for scenarios where SaaS is not appropriate.

Strengths
Recent AI updates to mock data generation improve the ease with which testing data can be created, so we rated the tool the best possible on this criterion. All licenses of BlazeMeter come with data creation tools on par with the market, but AI data generation requires an enterprise license.

When this licensing requirement is met, the large language model (LLM) used for data generation simplifies the need to set values. Instead of generating lists of sample values for fields, test teams can offer a general term such as "Car Model," which would return values of that type. The permutations are what one expects of AI generation; users could modify that request by saying "more like Jetta or Ranger," and get models from other manufacturers. The usual LLM caveats exist: the operator has to be able to narrow the scope to just what the test needs, but then randomized lists of data customized to the application or portfolio or organization are available to tests without recreation.

Additionally, BlazeMeter's API monitoring capabilities significantly enhance the testing processes, allowing for the efficient reuse of tests. This feature is designed for both technical and non-technical users, improving ease of use.

Scalability is also a strong point of this product. SaaS offloads scalability concerns to the vendor, which is a plus if licensing is not tied to throughput, which it is not for BlazeMeter. To push BlazeMeter's score into the highest range, an actual documented use case is scaling up JMeter, a tool that most testing organizations use and often find to be a

bottleneck.

Challenges
The tool does not offer a FAT testing client, using a browser plug-in instead. Developers, and even a fair number of testers, still prefer to have a standalone tool alongside a browser-based solution. This garnered BlazeMeter a below-average rating in the API client key feature.

While we would like to see a FAT testing client for the tool, if it otherwise fits the bill, BlazeMeter is able to import from most competing FAT clients, so a workaround could be to work in the tool of choice (which is generally very different for dev and test anyway), and then import.

While BlazeMeter's test management functionality gets the job done, it is not as thorough as its competitors. Reuse is one area where test management is possible, but the tool isn't designed for it or for doing things like snippet repurposing.

Purchase Considerations
While not fully explored in this report, BlazeMeter's robust API chaining options are designed to support complex testing scenarios. Prospective customers should inquire with the vendor if this is a feature they're looking for in a testing solution.

BlazeMeter also provides the ability to convert API testing products into its monitoring product. This is useful for teams looking to transition from different platforms, ensuring seamless integration into BlazeMeter's ecosystem while preserving their prior work. This also allows BlazeMeter to accommodate a wide range of user preferences.

BlazeMeter is SaaS. For some organizations, this is still a major issue. The browser plugin architecture will also be an issue for some teams.

BlazeMeter is a full application test tool. Organizations that prefer their testing to be SaaS and might need more testing than just APIs will be the happiest with this product.

Radar Chart Overview
Perforce is primarily a true platform with a single configuration and a single login, but some tools useful in API testing like Runscope are configured separately, so the vendor is on the right side of the Radar but not the extreme right. The product is also undergoing a fair number of innovations at this time, with the Perforce toolset being one of only two we analyzed that offer AI-assisted data generation for automation. Because of the recent changes, we classify Perforce as an Innovation vendor.

Postman, Postman API Platform

Solution Overview
Postman is one of the original API development tool companies, and API development naturally lends itself to API testing; APIs involve more effort for developers because the normal "write code, test results" of the development cycle has the added "write code to call the API." Postman has always been more focused on API testing than development itself, though the two are intertwined.

Postman Enterprise is offered as an application for Windows, Linux, or Mac desktops, or as a SaaS. Like most solutions in this space, the product is sold as a platform, but it does not meet our requirements to be classified as a Platform Play. The tool comes in several licensing models, including open source, though this analysis is focused on the enterprise license version.

Strengths
Postman has one of the oldest and most expansive FAT clients on the market, having largely invented the concept of "API client," alongside a few other vendors. This has several implications, both good and bad. Given a choice, nearly every developer would choose Postman, originally because there was an open source version, but as time went on, because the open source project and the company in general have been extremely quick to respond to user needs. This means that the tool is one of the oldest, and it has also been enhanced non-stop to keep it as one of the better API testing clients available. Other tools are often copies of it or a subset of its functionality, and nearly every vendor in this analysis can use tests saved in the Postman format because users expect Postman compatibility.

The pervasive use of Postman on development teams is the product's biggest advantage. Experience with the tools and trust in the product is already embedded in most organizations. Postman's API virtualization is also among the best in our analysis. Organizations that need to test clients without actual API calls should put this solution high on their list of

prospective tools.

Server mocking is exceptional in the Postman product, enabling mocks to be constructed from specification, traffic, or sample docs. As with most other vendors, generation of client stubs is not supported, but there are development tools able to perform that specific function.

Challenges
The less wonderful implication of Postman's longevity and reputation is that it has less motivation than competitors to include a truly excellent web-based editor. A browser-based editor is available for the SaaS solution but not for local use, for example. Of course, this is a far bigger issue for testing teams than for development teams.

Test data generation has some of the worst support in this analysis. While Postman can generate sample responses in mocking, actual data masking or generation is not supported. Organizations that have a data source from other testing will have no issues with this lack, but others will want to look to other data creation options.

If Postman's other functionality and advantages largely suit an organization's needs, there are dedicated test data tools on the market. However, this involves added expense and setup that should be considered during evaluation.

Purchase Considerations
Testing teams looking for a solution should consider Postman only if they are technically skilled or already use Postman. SaaS is offered, but it is newer, so we recommend teams look closely at both deployment models before installing.

Most development teams already have Postman in-house. Organizations looking to formalize a currently *ad hoc* API testing environment should consider starting with Postman due to familiarity and expand its use to the entire development staff.

The tool is more exciting to developers than to test teams, and which teams will be using the tool should be a consideration.

Radar Chart Overview
As mentioned above, the product is very much focused on API testing and does not come with a broader testing toolset that would make it a platform. The collection of features supported make it more than a pure play single-feature product, so it is positioned on the left side of the Radar but not at the edge because there is more than simple API testing in the toolset.

Postman provides a rock-solid product that has been testing APIs almost from their introduction. While new features are being added, stability appears to be more important at this time.

SauceLabs

Solution Overview
SauceLabs is a software test tool company whose primary product is the SauceLabs Testing Platform. The SauceLabs Testing Program is a broad testing solution with subsystems focused on most application testing features. Our analysis will focus on the Sauce API Quality subsystem, but we have included other parts of the platform in our evaluation because they are all available from a single login. The solution is SaaS, with a local Sauce Connect Proxy to run tests on APIs that are behind a firewall.

Strengths
SauceLabs has excellent low code and no code features, with auto generation of tests from API spec, low code editing, generation of tests from traffic, and JavaScript for parameter modification. Few vendors can match this level of support for less-technical test team members.

The product offers methods to generate data, but the data is not based upon the test dataset and likely will have to be modified to fall within (or for negative testing, outside) the domain and range of the variables being filled. This feature enables on-the-fly test data to keep testing moving along but may require a bit of work to make it truly useful to the application.

Challenges
Integrations are not a particular strength of SauceLabs, as it relies on a small local client being installed, then webhooks being written against it, to get most tools integrated. For organizations with complex environments, an evaluation of this integration process should be undertaken as part of any assessment.

While SauceLabs offers API virtualization for testing clients without relying on the server's availability, options to spin up virtual endpoints are not as broad as most vendors offer. While SauceLabs generates only from API specification, most vendors now support generation from several possible sources, including specification, documentation, and/or observed traffic.

Additionally, while data generation allows the adaptability we expect in API test data generation, we do wish it was more versatile in sources and saving options.

Purchase Considerations
Users looking for tools to enable test teams will be happy with SauceLab's offering, but those looking for development test tools or purely internal tools should consider other options.

Organizations that are looking for a broad application testing platform that includes API functional automated testing will find SauceLabs to be a good choice.

Radar Chart Overview
SauceLabs is a true Platform Play using a single sign-on with extra installs required only to extend beyond the firewall. This places it further to the right on our chart than any other vendor.

Recently, SauceLabs has been mostly implementing incremental changes, like adding support for new browsers. While it has typically been found near the front of the implementation curve for new features, we base our analysis on the current state, which places SauceLabs in the Maturity half of the Radar chart, meaning it is a good option for stable tooling.

SmartBear, ReadyAPI

Solution Overview
SmartBear is an API development and test software vendor with a wide selection of tools covering both spaces. For this analysis, we considered SmartBear's ReadyAPI and PactFlow products primarily, with some consideration lent to SwaggerHub for some features. ReadyAPI is a suite of three products, but for this analysis, we considered only ReadyAPI Test for functional testing features and ReadyAPI Virtualization for API virtualization functionality. PactFlow is SmartBear's solution for contract testing, and SwaggerHub is for API design testing and exploration. SwaggerHub and PactFlow are available as SaaS, while ReadyAPI is as a local installation.

Strengths
SmartBear's hierarchy of reusable test parts with parameterization make the tool's capability to manage tests and build up test variants one of the better ones in the market. The ability to assemble and reassemble both tests and test parts will be appreciated by test teams and developers alike.

SmartBear has more expansive access to functional parameters than most vendors we analyzed. This covers most possible parameters: headers, including TCP, and associated payloads. This expands the test opportunities and allows tests to report supporting data that other tools may not have access to. We consider the tool's programmability for parameters in testing to be among the best in this space.

Challenges
For organizations that want test data generation to be a part of the API test tool, SmartBear will not be as appealing as some other options. Production data masking is not included. As with the other vendors that are weak in this functionality, we recommend considering the total cost of implementing data generation in terms of both work and expense before choosing this solution.

A web-based API client is not available at this time. To support developers and test teams, having both web-based and local clients is preferred. However, the current SwaggerHub implementation is rapidly overcoming this deficiency.

Purchase Considerations
The ReadyAPI product comes as a local install suite, while PactFlow and SwaggerHub are SaaS. Organizations may find this disparate solution set an issue.

The SmartBear product is a broad solution set, of which API testing is one part. For organizations that need API management and a full application test suite, it will serve particularly well.

Radar Chart Overview

SmartBear sells ReadyAPI as a platform, but the API testing solution set is more a collection of tools than an overall platform, so we place SmartBear in the Innovation/Feature Play quadrant, but ReadyAPI's three products are implemented as unified modules. In addition, SwaggerHub shows SmartBear's movement toward a unified platform for API lifecycle management that includes testing. Given all of this, we kept the ReadyAPI solution close to the y-axis of the chart to indicate that it is a Platform Play close to being a Feature Play. SmartBear is implementing new features and products at a respectable pace, while maintaining its large existing portfolio, placing them well into the Innovation half of the Radar chart.

Testsigma

Solution Overview
Testsigma is a dedicated application testing company with products for Web, API, desktop, SaaS, and mobile testing that cover a variety of standard testing types.

The Automated API Testing solution is part of the Testsigma platform. It is available for local installation, as a SaaS, or as a self-managed instance. The tool is available in several simple-priced plans that make it easy to choose what level an organization needs. Testsigma leans more to the testing team side of application testing than the development side.

Strengths
The Testsigma solution is one of the best low-code/no-code solutions we analyzed. The tool offers complete creation and editing of API tests without requiring a line of code. Indeed, the test generation functionality is so thorough that this might be the only product in this analysis that does not require scripting at all and is probably best utilized for test generation via the low-code interfaces.

The platform is also strong in test data generation. The platform has a variety of options for test data generation and offers the ability to use masked production data. This scenario is ahead of most vendors in the space and means testers will have the data to test every case in their test plan.

Challenges
Testsigma does not offer API virtualization or client stub generation. While not the only vendor to be missing this functionality, testing sometimes requires access to APIs that currently do not exist, so creation of some kind of API interface will have to happen somewhere else.

The API test client is always in a browser. For some organizations, this will not be a problem, but to garner good scores for API editing, we expect both a local installable client and a browser interface.

Purchase Considerations
Server mocking is also not available in Testsigma. If it's not available elsewhere in the organization, that will have to be considered when purchasing a solution. API development and testing sometimes requires mock servers, so a solution will have to be implemented.

Organizations shopping for a tool for test teams will be attracted to Testsigma. Teams with less-technical staff will be interested because of the low/no code possibilities.

Radar Chart Overview
Testsigma is a platform. There is no option to purchase API testing separately and the toolset shares a single login and interface. This places them far out on the Platform Play side of the chart. Testsigma has a history of frequent updates with new features released regularly. This cadence seems to have slowed down in 2023, so we placed it in the Innovation half, but close to the Maturity line.

Tricentis, Tosca

Solution Overview
Tricentis is an enterprise application testing company. Its tools cover the breadth of application testing from web to mobile and beyond; if a feature is used in enterprise applications, they likely have a product that tests it.

Tricentis Tosca is the overall product we evaluated. It is an integrated testing platform aimed at enterprise applications, and as with other solutions in this category, our analysis focuses on the API testing features. The product is a SaaS with the ability to perform local testing via locally installed agents.

Strengths
The API editing tools available are some of the best on the market, for both web and local clients, with editing and saving of tests, snippets, and variables. Since the primary interface for developers and the more technical testers is the API client, this is very important to productivity.

Along the same lines, the ability to access and edit functional parameters is one of the best in the market. It covers the spectrum of possible parameters: headers, even async headers, and payloads in any string-based format (some require work; it doesn't know your special format but can use it for searching and parameterization). We consider the tool to be best in class.

Challenges
Like a few other tools, Tosca does not include mock data generation. Evaluators should consider the cost of implementing data generation and/or masking outside the scope of a Tosca implementation.

The product is also a bit behind the competition in low/no code capabilities. Low code tools are available, but unlike the best products in the space, they are not yet pervasive. Specifically for Tricentis Tosca, the API client tool is lacking in low code options, while the more testing-focused toolset offers good low/no code options. Organizations with a professional testing staff will be happier with the Tosca solution than teams hoping to extend test creation and results evaluation to less-technical staff.

Purchase Considerations
At the time of this analysis, less technical test teams will be happier with some of the other solutions available, though this is changing, and Tosca is better for those teams than it was a year ago.

More technical test teams and DevOps teams will find this to be one of the best solutions on the market. The inclusion of an entire suite of application testing means that organizations who need more than simply API testing should also have Tosca on their short list.

Radar Chart Overview
Tricentis is more of an actual platform than most vendors' products, with Tosca being a platform for application security, of which API security is but one small piece. The product has two different release cycles, short- and long-term support (STS, LTS). LTS offers the solid stability that most enterprises crave, but STS includes more-frequent updates for organizations that do not mind upgrading often. This is literally the best of both worlds and garnered Tricentis an Innovation placement, but not too far from the center line border with Maturity.

6. Analyst's Outlook

The API functional automated testing market is still coalescing. With some vendors coming from the testing market and others coming from the development and DevOps markets, the tools are moving to support both sets of testing. Given that API testing is far more technical than other application testing, evaluators should seriously consider the level of technical expertise required by products being evaluated. Similarly, evaluators should approach the market with an awareness of which side of the testing/devops needs spectrum their organization falls on. Some vendors solve both sides of the equation very well, but most are stronger on one side or the other.

This space is clearer than most with regards to Feature versus Platform Plays, with all of the vendors having a distinct focus on one or the other or an intentional mix. Innovation is less clear in this particular Radar for a variety of reasons, but mostly because of the overall rate of change. Testing is one of the few markets in which the usefulness of AI can't be overstated, and these tools are all implementing AI in one or several places and have plans to keep pushing it forward.

Organizations seriously considering implementing API functional automated testing should first define their specific needs. Since these tools mostly approach the problem from the DevOps or testing perspective, determining which products are most appealing will be heavily influenced by the organization's own requirements.

Moving forward, this market will continue to include both approaches. AI usage will continue to increase; indeed, we were surprised that these products were not using AI on the back end yet, but results processing for functional testing is relatively straightforward, so the vendors first focused on test and data generation for their AI efforts. These tools are continuing to move toward a sweet spot where they are useful for both test and DevOps teams. We believe that half or more of the vendors will be fully suited for both use cases in the next year. We also see API testing being increasingly integrated with the whole of application testing. While several vendors offer integrated tools for the entire solution set, we expect all vendors to move more in this direction because the entire application and application stack need testing, and

increasingly, APIs are just a part of that stack.

To learn about related topics in this space, check out the following GigaOm Radar reports:

- GigaOm Radar for API Security
- GigaOm Radar for Application Security Testing
- GigaOm Radar for Cloud Security Posture Management (CSPM)

7. Methodology

For more information about our research process for Key Criteria and Radar reports, please visit our Methodology.

8. About Don MacVittie

Don MacVittie has more than 25 years of technology industry experience, working in nearly every IT position in a wide variety of organizations – from tax software developer to insurance industry strategic architect and IT manager at a utility. Don has been an analyst for the last decade, is named on two networking patents, and has co-authored ANSI and ISO standards in geographic information systems. His love is learning about technology, and he will happily discuss technology with anyone.

9. About GigaOm

GigaOm provides technical, operational, and business advice for IT's strategic digital enterprise and business initiatives. Enterprise business leaders, CIOs, and technology organizations partner with GigaOm for practical, actionable, strategic, and visionary advice for modernizing and transforming their business. GigaOm's advice empowers enterprises to successfully compete in an increasingly complicated business atmosphere that requires a solid understanding of constantly changing customer demands.

GigaOm works directly with enterprises both inside and outside of the IT organization to apply proven research and methodologies designed to avoid pitfalls and roadblocks while balancing risk and innovation. Research methodologies include but are not limited to adoption and benchmarking surveys, use cases, interviews, ROI/TCO, market landscapes, strategic trends, and technical benchmarks. Our analysts possess 20+ years of experience advising a spectrum of clients from early adopters to mainstream enterprises.

GigaOm's perspective is that of the unbiased enterprise practitioner. Through this perspective, GigaOm connects with engaged and loyal subscribers on a deep and meaningful level.

10. Copyright

© Knowingly, Inc. 2023 *"GigaOm Radar for API Functional Automated Testing"* is a trademark of Knowingly, Inc. For permission to reproduce this report, please contact sales@gigaom.com.

Knowingly Corporation
3905 State Street #7-448
Santa Barbara, CA 93105-5107