



Market Research

Application Troubleshooting Report

Abstract

The environment in which internal applications run has become increasingly more complex. Applications are being run both in-house as well as hosted in private and public clouds, with the number of these internally-developed applications on the rise, while support teams are becoming more diverse in location, function and responsibility.

This market research explore how well are application teams troubleshooting problems, and what's causing them the greatest challenge in finding the source and cause of an issue?

Stackify 2014 Application Troubleshooting Report

The environment in which internal applications run has become increasingly more complex. Applications are being run both in-house as well as hosted in private and public clouds, with the number of these internally-developed applications on the rise, while support teams are becoming more diverse in location, function and responsibility.

All of this complexity comes at a time when applications are most critical to the business, making it imperative that any issues arising be addressed as quickly as possible.

So how well *are* application teams troubleshooting problems, and what's causing them the greatest challenge in finding the source and cause of an issue?

To find out, we surveyed over 170 professionals involved with supporting internal applications to better understand who is responsible, what tools are being used, and what impact they have on quickly resolving application issues.

Who's Supporting Applications?

When thinking about organizations developing and supporting internal applications, thoughts normally turn to larger organizations. But of the 172 respondents representing organizations of every size, from the very small business, up to the very large enterprise, nearly **two-thirds** resided in the "500 or less" category, as shown in **Figure 1**. This demonstrates the adoption of internal application development down to even the very small business is in full swing.

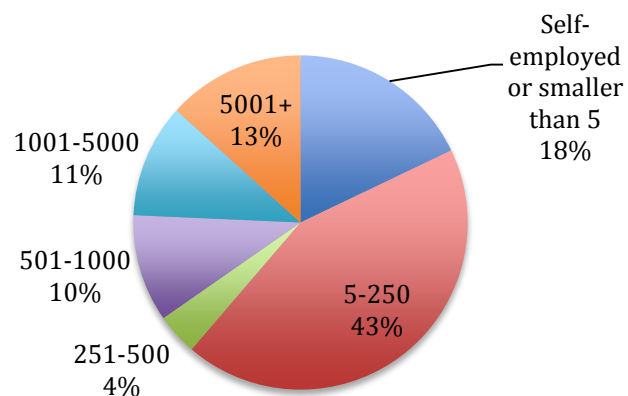


Figure 1: Organizations of all sizes build internal applications

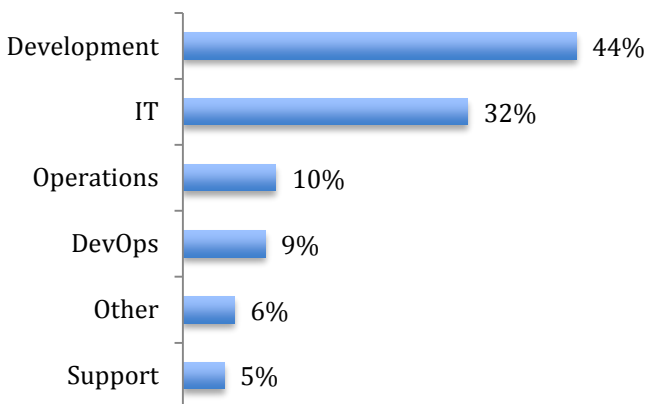


Figure 2: Roles of those supporting applications

Applications can't run without hardware and hardware is useless without applications. But with the DevOps movement certainly no longer in its infancy, it's surprising to see only **9%** of respondents identifying themselves as part of DevOps. Instead, IT and Development roles made up **76%** of the respondents, as shown in Figure 2

What's Being Supported?

Organizations today are reliant on more than a single critical application. As shown in Figure 3, **85%** of organizations are utilizing multiple applications. With more than one-third developing and supporting over 10 internally developed applications, it's surprising to note that this group is distributed across organizations of all sizes across the spectrum in this report.

In order to keep those applications running, organizations are required to make a significant investment in servers or in the cloud to support front end user interaction, back end data storage and retrieval, and everything in between. It is interesting to see that despite the majority of organizations responding as being below 250 employees, more than half of organizations use 10 or more servers, as shown in Figure 4.

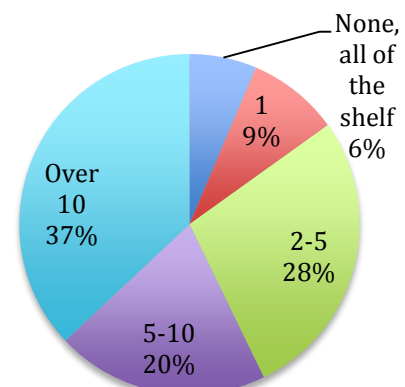


Figure 3: Number of in-house applications developed by organizations

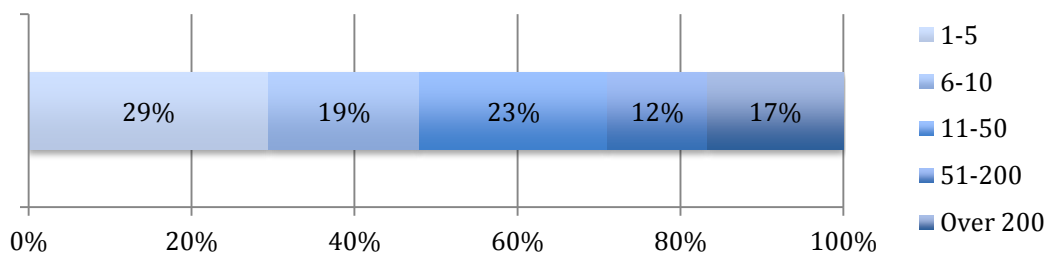


Figure 4: Number of servers used to run internally developed applications

As expected, the number of servers required roughly aligned the number of applications within an organization.

Mentioned earlier, organizations are looking beyond the organization walls to find viable hosting options in both public and private cloud offerings. As shown in **Figure 5**, organizations are using every hosting option to provide highly available and reliable applications to their end-users and customers.

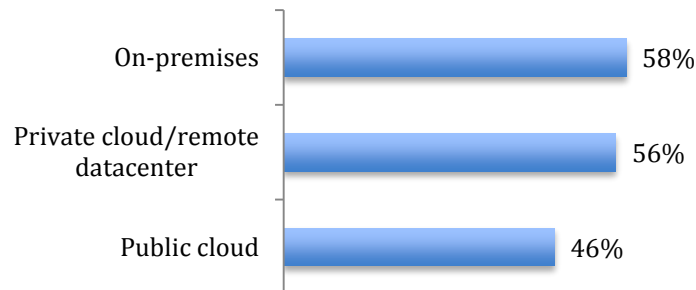


Figure 5: Where organizations are running their applications

With so many applications, on so many servers, on so many platforms, development organizations are stretched thin when application errors occur. *So, what are they doing when issues arise?*

What Happens When Apps Go Wrong?

It's a given that applications are going to have issues. When exceptions occur, the question of how application support professionals find out is the first indicator of how mature their application support model is.

While **46%** find out using application monitoring, **37%** are still finding out from application users calling the helpdesk, as shown in **Figure 6**.

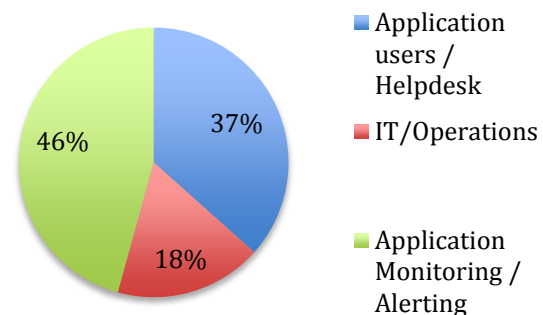


Figure 6: How organizations are notified of application issues

Once an issue is known to exist, organizations told us they gather and utilize a range of troubleshooting data to identify the cause of application issues. As shown in **Figure 7**, logs and error data top the charts as the primary sources of information leading to the resolution of issues.

Given the large amount of data encompassed by the various sources in, organizations must turn to solutions to collect, analyze, alert, and report on.

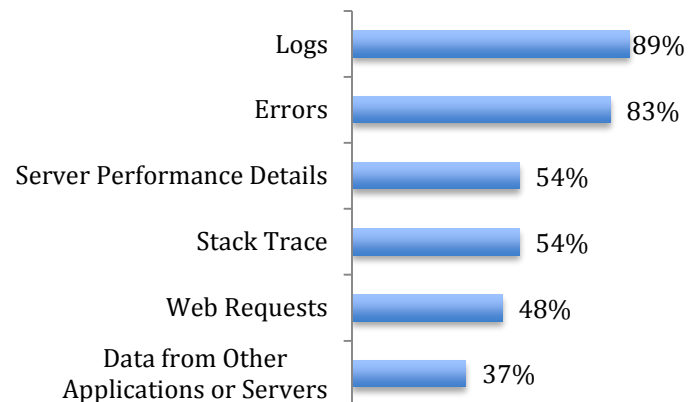


Figure 7: Information used to troubleshoot application issues

While logs and errors topped the list of *data sources* used to troubleshoot application issues, error aggregation tools fell behind infrastructure monitoring, and notification tools in a list of the top tools, shown in **Figure 8**, used by organizations to troubleshoot application issues.

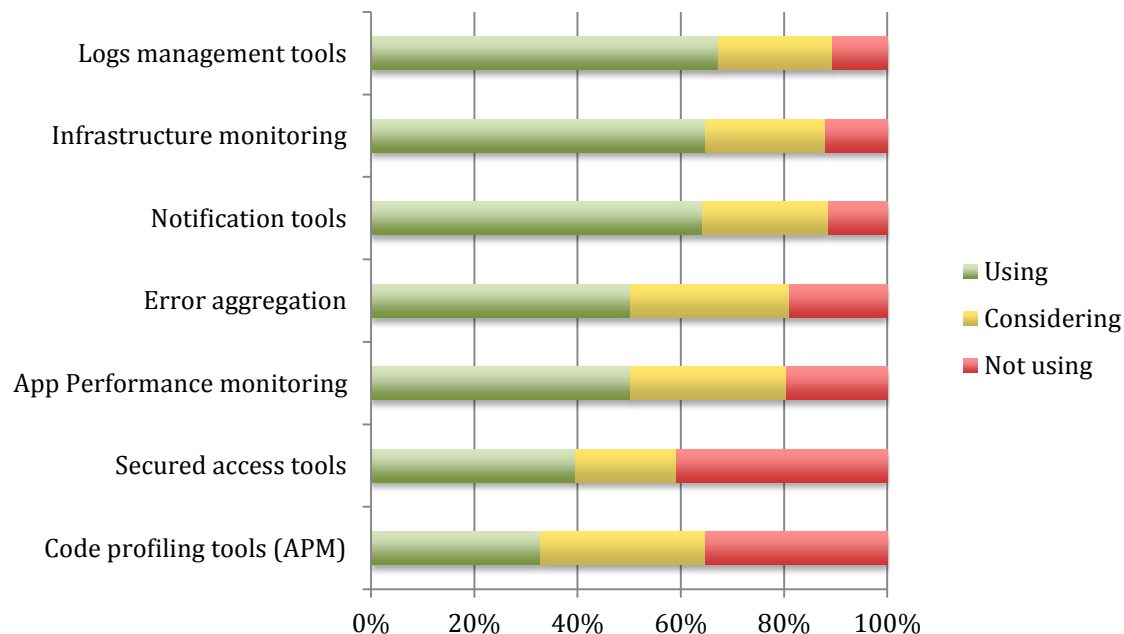


Figure 8: Application troubleshooting tools used and being considered by organizations

Even with log management tools at the top of the list, a full one-third of organizations or more aren't using any or all of the tools listed, making application troubleshooting largely a manual process of collecting and correlating error, log, and supporting data.

When asked about the nature of the tools being used, only **41%** of organizations are using an *integrated solution*, as shown in **Figure 9**, utilizing the best functionality of each tool type under a single pane of glass to simplify the searching for, and identification of error sources.

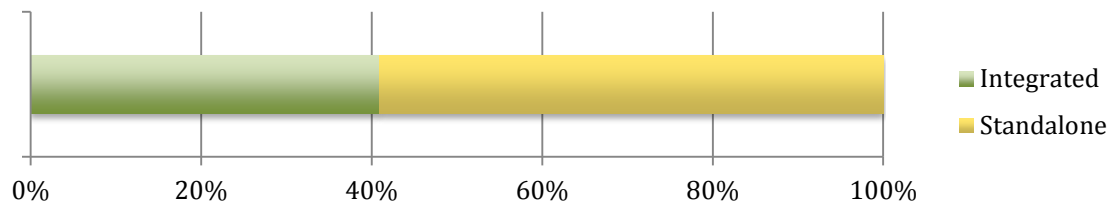


Figure 9: Troubleshooting toolset types users by organizations

With so many organizations using standalone tools, what is the impact on the organization's ability to quickly troubleshoot issues, which, in turn, has implications on the business itself?

The Impact of Integrated Troubleshooting Tools

Mature development organizations use integrated tools to simplify and speed up the process of troubleshooting application issues. *But do those tools really have a positive impact?*

How much time is needed?

When asked about the amount of time taken to identify the root cause of an issue, those with integrated tools showed a lower average amount of time needed. Organizations with *standalone tools* cited **52%** of issues taking a half of a day to find the root cause, whereas those with *integrated tools* only cited **37%**, as shown in **Figure 10**

This is further confirmed by the demonstrably higher **46%** of issues taking only an hour to identify root causes with integrated tools, compared with only **32%** of issues when using standalone tools. Integrated tools empower an organization to more quickly root out a problem.

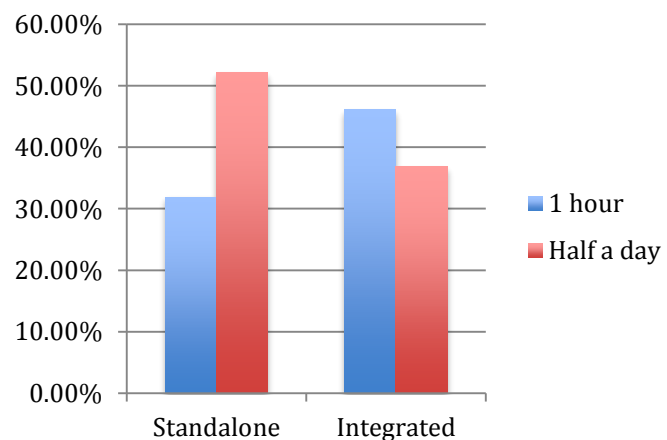


Figure 10: Average time to identify root cause of a problem

But even if a root cause is identified, it still takes time to resolve, so what's the impact on users.

How Are Users Affected?

This is the ultimate litmus test of a development organization – can they identify and resolve an issue before users are impacted. Granted finding the source of a problem and coding a workaround are two very different things, but should the troubleshooting duration take too long, users may be impacted well before a resolution can even be conceived. On average, as shown in **Figure 11**, organizations are able to resolve issues before users are impacted nearly 2 to 1.

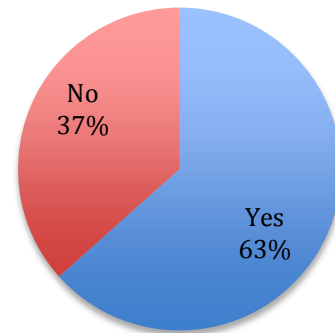


Figure 11: Are issues resolved before they impact users

When broken down into those using standalone vs. integrated tools, the gap widens significantly, as shown in **Figure 12**. Those using *integrated tools* are able to resolve issues **80%** of the time *without* impacting users, whereas those using *standalone tools* only do so **48%** of the time. This affirms the value of integrated tools in the troubleshooting process to both speed up the identification of the root cause, while minimizing the impact on application users.



Figure 12: % of issues resolved without impacting users

Conclusion

Internal development organizations face an ever-present challenge with each new or repeating error that rears its ugly head: find out why quickly without disrupting application availability.

Troubleshooting data is available in, arguably, massive quantities, making it even more difficult to find the source of an issue. Tools address the automation needed to minimize manual searches, but generally lack the integration needed to have a material impact on developments speed and accuracy of troubleshooting.

With 59% of organizations using standalone tools, they are only increasing the time it takes to identify the root cause of application issues and putting the business at risk. Until organizations adopt integrated application troubleshooting solutions, they will continue to face the same troubleshooting challenges, attempting to solve application errors with little intelligence or insight into why and how errors occur.

About Stackify

Stackify is the industry's only platform that combines error-aware log management and smart error aggregation. The solution optimized for dynamic environment, providing cross-server aggregation all errors and logs. Stackify also provide an optional comprehensive application monitoring, performance management, custom metrics, notification, and secured data access for better app troubleshooting. Stackify provides developers and operations with DevOps insight, allowing them to detect issues before they affect business and shorten time to resolution to ensure a better end user experience.

Try it now for [free](http://www.stackify.com/) at <http://www.stackify.com/>