Developing Software: Methodologies, Design Principles, and Technology Adoption

by Megan Sheehan, Ph.D.
# Table of Contents

- About the Author ................................................................. 1
- Introduction ........................................................................ 1
- Respondent Job Title ......................................................... 1
- Type of Application ............................................................ 2
- Development Methodologies ............................................. 3
- Important Software Development Capabilities ....................... 5
- DevOps Maturity .................................................................. 7
- Emerging Technology Adoption .......................................... 8
- ALM Octane .......................................................................... 9
The purpose of this white paper is to explore the survey results in more detail and to offer suggestions for organizations as they pursue their own Agile journey.

Respondents were professional developers, testers, and IT managers who work in a mix of technical roles.

---

About the Author

Megan Sheehan, Ph.D.
megan.sheehan@microfocus.com

Megan Sheehan is a Senior Product Manager in Micro Focus® Application Delivery Management focused on predictive analytics. She has a Ph.D. in Communications Sciences with an emphasis on quantitative research (SEM, CFA, Multivariate Regression, Data Mining, Segmentation, Meta-Analysis, Conjoint Analysis, Interaction Analysis, Experimental Design, Time Series Analysis, and Survey Design). In addition, she has over 15 years of experience in product management, program management, technical product planning, product innovation, and market research on a range of enterprise software products.

Previously, Megan led efforts around the HP LES Enterprise Innovation research on hardware, software, and services. Before joining HP, Megan worked at Microsoft on Windows 7, Visual Studio, and Windows Embedded. Furthermore, she served as the owner of Habits & Practices Pillar as well as redesigned Microsoft’s approach to capturing, quantifying, and prioritizing user needs and translating them into product features.

Introduction

Dimensional Research recently conducted an online survey with over 800 professional developers, testers, and IT managers in companies with more than 500 employees. The survey was intended to measure software development methodology, importance of key design principles, and technology adoption. The full Dimensional Research report can be found here.

The purpose of this white paper is to explore the survey results in more detail and to offer suggestions for organizations as they pursue their own Agile journey.

Respondent Job Title

Respondents were professional developers, testers, and IT managers who work in a mix of technical roles. IT Manager or IT Director was the most common job (32%). 12% of respondents were developers. 11% of respondents reported that they were IT Executives and 8% of respondents were Architects.
Web and mobile applications are the most common types of development projects. 39% of respondents built a Web application and 24% indicated that their most recent project was a mobile application. 18% of respondents worked on business or package applications. 10% reported working on mainframe or legacy applications. 7% worked on a desktop application in their most recent project.

**Type of Application**

Web and mobile applications are the most common types of development projects.
Most projects were deployed to Cloud Environments: 39% to private clouds, 22% to public clouds, and 18% to hybrid cloud environments. 20% of projects were not deployed to a Cloud environment. Micro Focus ALM Octane SaaS can be run on a datacenter or AWS infrastructure.

When asked what development methodology their organizations were using, most respondents (77%) reported that they were using a hybrid approach or leaning towards Agile on most projects.

**Development Methodologies**

When asked what development methodology their organizations were using, most respondents (77%) reported that they were using a hybrid approach or leaning towards Agile on most projects. The results indicate that the market is in flux. A minority (9%) report using pure Agile development methods across all projects, but most organizations are in transition to Agile. 11% of respondents report that they are leaning towards Waterfall but have adopted some Agile methods across projects. Only 3% of respondents reported that their organizations were using a pure Waterfall methodology (completely traditional process across all projects).

![Deployment types](image_url)
Moving to Agile is a cultural shift for an organization. It’s a journey, usually beginning with people—either assembling the right teams of people or training existing personnel on how to use Agile processes. There are many good training programs available, including SAFE, Scrum, DoD, LeSS, and Spotify.

A transition from a traditional Waterfall methodology to an Agile approach is not something that happens overnight. Moving to Agile is a cultural shift for an organization. It’s a journey, usually beginning with people—either assembling the right teams of people or training existing personnel on how to use Agile processes. There are many good training programs available, including SAFE, Scrum, DoD, LeSS, and Spotify.

Next, after people understand what Agile best practices are, it’s time to re-define their organization’s processes. This can also be very challenging since change is difficult and teams may be fighting organizational inertia. Change represents uncertainty; sometimes it’s easier to get started with an Agile transition if you label it a “pilot.” People are generally less resistant to something that’s called a “pilot.” If you can show success in the pilot, it’s easier to build on that and expand the scope of Agile—either by adopting more Agile processes or by gaining traction with new teams. As the survey data indicates, frequently there is a mix of methodologies within an organization. Some teams may be further along the Agile path, while others are closer to a Waterfall methodology.

After the new processes are established, it’s time to select the right tools. Your organization may already have tools that can support Agile processes or you may look to enhance your toolchain with applications that are optimized for Agile development. If possible, leveraging what you already own would reduce the costs and time needed to transition to Agile rather than purchasing new tools. Existing ALM and QC customers are entitled to ALM Octane as part of their active support contracts. They are also able to share licenses between ALM/QC and ALM Octane. This flexibility enables customers to support projects using a Hybrid or Agile methodology with ALM Octane and continue to use ALM/QC to support existing projects.

You may also find that a mix of open source and commercial tools is the right balance for you. When selecting tools, you should consider the features that are most critical to your organization. Do you need to have visibility across multiple projects or teams? Do you have distributed development teams? Are your teams using a mix of methodologies (from pure Waterfall to pure Agile or somewhere in the middle, like Hybrid)?

In your opinion, across typical development projects, to what extent is your organization using Agile or Waterfall methodologies or a more hybrid approach?**

Figure 4. Development methodologies
Planning is the first step in any Agile development project. One of the key principles of an Agile approach is delivering useable, high-quality features and products that come from thoughtful planning and a clear understanding of what needs to be delivered.

Respondents were asked to share their perceptions on the importance of four key software development capabilities:

- Being able to easily access and reuse tests, source code, and other project assets
- Ability to easily access key data on project performance
- Ability to save different versions of tests
- Flexibility to integrate with commercial testing tools and open source solutions

The list goes on. The new ALM Octane is designed to be flexible enough to support the full continuum of development methodologies from pure Agile, to Hybrid, and Waterfall in one tool, on a unified platform. In addition to its inherent flexibility, another one of the main benefits of using ALM Octane is visibility across teams, projects, and releases. Wherever your organization is on the continuum from Agile to Waterfall or if you have teams that are using different methodologies in parallel, ALM Octane can support it. If you’re like many teams and trying to adopt more Agile processes over time or using a Hybrid approach, ALM Octane has features to support you.

The ALM Octane platform provides teams with the ability to Plan, Build, Test, and Release in an Agile manner at enterprise scale for dozens of teams and thousands of users simultaneously. Planning is the first step in any Agile development project. One of the key principles of an Agile approach is delivering useable, high-quality features and products that come from thoughtful planning and a clear understanding of what needs to be delivered. Accurate planning is essential. ALM Octane supports the creation of releases, sprints, workflows, and backlog items with ranking, and calculates the expected velocity and capacity. ALM Octane supports teams through the development and build phases, including tracking tasks, and the overall release progress.

Continuous testing of delivered content in conjunction with the feature and story development is another key component of Agile development. Tests can be added directly to backlog items, test results can be analyzed, build processes can be specified and evaluated, and the quality of the entire release can also be quantified.

ALM Octane also supports the essential retrospective evaluation after the release or sprint is completed. Teams present the work that was done, evaluate how well the release or sprint was executed, and discuss what could be done differently/improved on in the next iteration. ALM Octane also allows easy tracking of the metrics for the sprint or release.

Important Software Development Capabilities

Respondents were asked to share their perceptions on the importance of four key software development capabilities. The first, “Being able to easily access and reuse tests, source code, and other project assets” was rated as very important by 86% of the respondents. The ability to “easily access key data on project performance (e.g., number of pass/failed tests, percentage of automated tests, injection rate of defects, etc.)” was rated as very important by 81% of the survey respondents. Seventy-eight percent of respondents indicated that the ability to “save different versions of tests” was very important. Three quarters of the respondents reported that the “flexibility to integrate with commercial testing tools and open source solutions” was very important.
In summary, all of these capabilities were all very important to the survey participants. Asset reuse (tests, source code, etc.) was rated as the most important capability and likely reflects a desire to increase team productivity and efficiency. Across industries, software development teams are under increased pressure to get to market faster. A logical strategy is to leverage existing assets to avoid rework whenever possible. But it can be very difficult to find the right tests, code, etc. ALM Octane makes it easier to locate existing project components through linkages to application modules and backlogs. The high importance ratings for the ability to "save different versions of tests" are related to test reuse. By saving different versions of tests, QA engineers and developers can reuse existing assets and alter them to test specific changes in code without losing the original. ALM Octane test versioning provides the ability to compare test versions, examine revisions, determine who made changes, assign a specific test version as the "current" one, and manage the releases assigned to each script revision. You also have traceability and an audit trail, so if there is any question about what was tested or when changes were made, ALM Octane is the system of record and provides that history. In addition, the ability to access key data on project performance (e.g., number of pass/failed tests, percentage of automated tests, injection rate of defects, etc.) was also deemed very important by survey participants. Data trumps opinion.

The high importance ratings on key data indicate a desire to have visibility into a project. It is easier to dismiss one person's opinion about a risky release and harder to refute a data point such as, "80% of automated tests are failing" or "a 60% gap in test coverage." The ALM Octane Dashboard leverages Elastic Search to enable users to explore key project data. Information is tailored to the user's context and provides highly customizable reports, graphs, and charts. The Dashboard is the control center for analysis of an application's development and quality. Dozens of out of box widgets are available, which users can customize.

In addition, users can create their own widgets, share Dashboard versions with select team members, or publish their Dashboard so the whole team can access it. The Dashboard provides a visual, customizable display of the application's development progress and what the level of quality is. The Dashboard also provides data to track the progress of development and testing, and the status of milestone goals; content
planning for deliverables; test coverage and pass/fail rates; and DevOps insights on data collected from external servers such as pipeline runs, production coverage, and tracking committed changes. 75% of the respondents rated “flexibility to integrate with commercial testing tools and open source solutions” as very important. Respondents probably want to leverage their existing software assets, use open source solutions, and select third party commercial applications to drive efficiency and cost optimization. Most enterprise organizations have heterogeneous environments; very few have standardized on a single vendor’s offerings. Rather than forcing customers to rip and replace the applications they already have, ALM Octane integrates with both open source and third party commercial software applications including TFS, JIRA, UFT Pro, Jenkins, TeamCity, Git, SVN, Bamboo, Elastic Search, Gherkin/Cucumber, Selenium, Slack, IntelliJ IDEA, Eclipse, and many more. Wherever possible ALM Octane has embraced open source and optimized for adding value in addition to the open source offering rather than competing with it.

In addition to the out of box integrations, ALM Octane provides a REST API, which enables connection with additional planning, development, test, management, and CI tools. This powerful capability provides visibility across the entire software development lifecycle to assess the overall quality of a release from inception to production.

DevOps Maturity

Another key topic of the survey was on adoption of DevOps practices, specifically continuous delivery, IaaS, PaaS, microservices, and OS level container technology. Most survey respondents (74%) report that their organizations are using a Continuous Delivery approach in their product development process. Nearly a quarter (22%) indicated that they use Continuous Delivery on all or most projects. Another 52% of respondents use CD on at least some projects. Almost one fifth of the participants (19%) report they are not currently using CD, but plan to use it in the future. Very few respondents (7%) have no plans to use CD or don’t know what CD is.

![Figure 6. DevOps practices](image-url)
These findings indicate that a continuous delivery approach has become mainstream. ALM Octane’s integration with Jenkins, TeamCity, and Bamboo makes it easier for teams getting started with a continuous delivery approach to be successful and accelerates teams who are already using CD because they can integrate their existing pipelines with ALM Octane. ALM Octane integration provides visibility into the release pipeline and source code management systems like GIT or SVN, which enables better collaboration among teams and supports a faster path from manual testing to test automation.

**Emerging Technology Adoption**

As for emerging technologies, Infrastructure-as-a-Service (IaaS) had the highest usage rate with 50% of respondents currently using it. There was also high interest in IaaS, as another 33% of respondents are planning to use it in the future. Only 13% have no plans to implement IaaS.

Platform as a Service is another popular technology solution. 47% of the respondents are currently using PaaS and another 38% are planning to use it in the future. Only 15% of participants are not using PaaS and have no plans to implement it. There is also strong interest in OS Level Container Technology, but only 29% of respondents are currently using it; 41% of respondents indicate that they are planning to use it in the future. The remaining 30% of respondents have no plans to implement OS Level Container Technologies.

Use of Microservice Architecture is mixed. About a third (34%) of respondents are currently using Microservice Architecture and another 39% are planning to use it in the future, while 27% of respondents have no plans to use it.

We want to thank the more than 800 survey respondents who provided valuable input to Dimensional Research on their software development methodologies, technology adoption, and the key design principles they value. The details provide us with a number of concrete suggestions for organizations as they pursue their own Agile journey.
If you are interested in learning more about ALM Octane, an overview of the product features, integrations, and capabilities is available [here](#). You also have the opportunity to use ALM Octane for yourself with a free trial. The trial sign up is available [here](#).

**Learn More At**

[www.microfocus.com/alm-octane](http://www.microfocus.com/alm-octane)