How to Choose an Enterprise Agile Platform

Handling Complexity and Scale in Application Development
# Table of Contents

Enterprise Agile: Extending the Agile Process Beyond Team Level .................. 1  
Key Differences Between Small Agile Projects and Enterprise Agile .......... 1  
Support Full Enterprise Agile Development Cycle ................................. 2  
Identify Trends from Data ................................................................. 8  
Introducing ALM Octane ................................................................. 8  
What Makes ALM Octane Special Today ............................................. 9  
Conclusion—Enterprise Agile Readiness with ALM Octane ....................... 10
Enterprise Agile: Extending the Agile Process Beyond Team Level

Without a doubt, Agile development methodology is becoming the norm. The State of Agile Report, the largest, longest-running Agile survey in the world, recently published its 11th annual report (April 6, 2017). The survey polled thousands of IT organizations and collected insights on the most used Agile tools, Agile maturity in organizations across the world, and challenges faced when scaling.

Among the key findings:

- **Agile adoption is still growing**—While 94% of respondents said their organizations practiced Agile, they also stated that more than half of their organizations’ teams are not yet practicing it. 80% of respondents also said their organization was at or below a "still maturing" level with Agile.

- **Enterprise agility is increasing**—Enterprise agility is growing throughout organizations and across almost all industries at an accelerated rate.

- **Organizations are succeeding with Agile**—Respondents stated that the top benefits of adopting agile were accelerated delivery, better project visibility, improved team productivity, and management of changing priorities.

In the recent Micro Focus® Quality 2020 report, 16% of respondents reported that their organization uses a pure Agile development methodology, 51% said they lean towards an agile development methodology, and 24% indicated they use a "hybrid" approach. This implies that while awareness and use of Agile are widespread, there is still a transition happening from traditional processes to fully Agile ones. Many organizations are struggling with the transition.

Key Differences Between Small Agile Projects and Enterprise Agile

Agile development methodology originated in small teams with relatively small projects. There are several key challenges when adopting Agile practices at the enterprise level.

<table>
<thead>
<tr>
<th>Area</th>
<th>Small Agile project</th>
<th>Enterprise-level Agile project</th>
</tr>
</thead>
<tbody>
<tr>
<td>Application architecture</td>
<td>The application architecture is expected to emerge as the project progresses.</td>
<td>Leverage and reuse successful architecture patterns. Upfront architecture design.</td>
</tr>
<tr>
<td>Program portfolio management</td>
<td>No need to provide a mechanism for higher-level integration to fulfill typical corporate needs.</td>
<td>Adopt a hybrid approach to application development. Manage and prioritize all business demands for IT, consolidate legacy and agile projects in program and portfolio views to continuously focus on business goals.</td>
</tr>
<tr>
<td>Organization</td>
<td>Single co-located Agile teams work on their own.</td>
<td>Align many Agile teams to the same mission. Typically, these are distributed teams.</td>
</tr>
</tbody>
</table>

Continued on the next page
Agile practice excels in finding the right way by experimenting and getting quick feedbacks. However, when rolling agile practice across the entire organization, it doesn’t mean everything is experimental. Gartner coined the word “bi-modal” IT to describe that an enterprise has two modes of IT, mode 1 is governed by best practices and predictable, and mode 2 is experimental. The co-existence of the two modes are essential to Enterprise Agile practice.

The keys to successful Enterprise Agile lie in aligning teams of both modes, managing the scale and complexity, correlating the data from all places and phases of application development lifecycle, identifying trends from the data, and moving fast based on that intelligence.

## Support Full Enterprise Agile Development Cycle

Typically, application development lifecycle involves four key phases: Plan, Build, Test, Release. When choosing an enterprise agile ALM solution, check its capabilities from the perspective of each of these phases.

### 1. Plan for Agile Project Success

Planning in Enterprise Agile means managing demands, Agile projects and programs, and their prioritization against resources, budget, and business goals. That’s why Enterprise Agile requires program portfolio management (PPM) for the highest-level of strategy and decision-making. Though many vendors claim to support portfolio management, some provide limited functionality. Advanced PPM features such as portfolio prioritization, optimization, and financial management capability are what you need to reduce costs for quality IT operations. Idea & Demand Management, Enterprise Resource Management, advanced analytics, and reporting are all important in portfolio management.
It is said the roadblock for Enterprise Agile adoption is culture. An Enterprise Agile platform that fully supports all personas across the enterprise will lower the hurdle of adoption. Starting from CEO/CIO, the tool should facilitate them to participate enterprise backlog prioritization and clean-up. It should also have highly flexible dashboard features that allow executives to create their own dashboards, providing immediate insight into the overall application portfolios and individual project status. Some vendors rely on integration with third-party tools to have this capability, or they provide this capability separately and require additional fees for it. Choosing a tool with this built-in not only reduces costs but also gives executives more control of how to explore the data.

In large enterprise projects, the project management office (PMO) enforces best practices guidance that each project should follow. With a proven Enterprise Agile platform in place, your PMO can define templates for requirement/user story, defect or test, and configure workflow for these them. It can also embed role-based business rules into these work items.

2. Build Quality Code Continuously
To realize the benefit of Enterprise Agile, it is essential to have end-to-end agility. This requires putting a DevOps infrastructure in place. But setting up this infrastructure often takes a lot of effort and time. Your Enterprise Agile solution should minimize the steps needed to connect to the most popular continuous integration servers (such as Jenkins, TeamCity, Bamboo, TFS) and allow you to manage pipelines from within the solution and gather code changes and build data.

By using a tool with built-in DevOps support especially build failure analysis capability, you enable your teams to:

- Decrease the waiting time for test environment provisioning
- Discover issues and address them early and quickly
- Eliminate delays in hand-off between groups, such as between development and QA, by making it obvious to everybody whether the hand-off criteria are met
This effectively reduces your overall project costs.

**Figure 1. DevOps Support**

During project execution, project managers and scrum masters closely track the work progress of teams. Look for a solution that provides a Team Backlog module where a project manager or scrum master tracks all the work in his management scope: user stories, code changes, tests, defects, and charts to visualize the data.
Out-of-the-box integration with popular IDEs such as IntelliJ and Eclipse allows developers to access their tasks without leaving the IDE, which greatly boosts developer productivity.

In addition, code changes are automatically linked to stories or defects, which means you don’t waste time chasing status report from developers while always knowing the progress in real time. It also enforces best practices like commenting changes properly, which is very important in making an application maintainable. In large enterprise projects, those who are in specific roles have their favorite tools. Out-of-the-box integration with popular IDEs such as IntelliJ and Eclipse allows developers to access their tasks without leaving the IDE, which greatly boosts developer productivity.

On your Enterprise Agile platform, project teams can use Slack or Skype to collaborate smoothly and give Bots commands to perform operations within these tools. The platform itself should also make collaboration at hand, that in any context of the platform, relevant users are shown (such as who are the committers relevant to a build), and one can notify any of them by mentioning the user in a comment.

Nowadays some best-of-breed development tools come from open-source, your Enterprise Agile tool should have strong open-source support which enables you to take advantage of the innovations these tools provide. Your Enterprise Agile platform should also have an OData REST API along with SDK that makes it easy to integrate any other tool needed.

3. Test Consistently for Application Quality at Scale

Many vendors don’t have test management capabilities built into their Enterprise Agile tools. What they do instead is to pull back the automated development build result to mark the success/failure of the whole test case, without the concept of test step and which step failed. Some even require manual input of test
result. They don’t handle end-to-end tests, user acceptance tests, or manual tests—let alone performance tests and security tests. Look for a solution that manages all these types of tests, automatically collects test results and generates reports that give you a 360-degree view of application quality.

Another important thing to check is whether it facilitates the reuse of test cases. The value of reuse lies in not only efficiency but also test consistency across different run environments. Reusing test cases in Regression test ensures that new developments don’t break existing code.

**The Importance of Behavior-Driven Development (BDD) in Testing**

In the era of automating everything, why is a manual testing feature still important? One of the key tenets of the agile process is to pull together teams with various skill sets to create items that add value to the company at a high velocity. As such, teams need a way that business users can create test scenarios. It is critical in involving business users into test-scenario creation. Behavior Driven Development (BDD) support allows testing teams to describe business scenarios with natural language and perform an initial manual run of the test. After business users have verified that these scenarios make sense. Automation can be done using frameworks like Cucumber or SpecFlow.

**Figure 3. Quality**

BDD tools define test scenarios and help align all stakeholders (developers, testers, quality assurance, and business analysts) on test requirements that everyone understands (typically in a format of “given [context], when [event occurs], then [outcome]”). A byproduct of this tight collaboration on test requirements among all stakeholders is often a better understanding of the business requirements themselves. BDD also
enables test-driven development, because test scenario requirements are written first and coding follows. This process facilitates test automation, with traceability between test scenarios and the automation code. Whether your Enterprise Agile tool has a native framework to support BDD makes a critical difference.

Another convenience the tool should provide in line with agility is to allow different roles to submit defects directly in the context where they are performing testing related tasks. Switching context simply for submitting a defect incurs a high percentage of overhead and lowers user satisfaction.

4. Track the Development Lifecycle for Actionable Insights

Look for an Agile platform that is designed to deliver continuous feedback. It should automatically capture and render the following metrics better than products from other competing vendors:

- Traceability from requirements/defects to related code changes, builds, and tests.
- Test results of a wide range of leading test automation tools.
- Visibility to code and developer activities. There is no need to manually collect progress and status in daily stand meetings and sort the data in a Microsoft Excel file.
- Dedicated “quality” view showing the real-time status of quality and progress of the application under development, with granularity from epic to application modules and features. With the insight, VP Apps and VP QA can quickly understand whether development is progressing healthily, and the application is ready to be released.

The correlated data from all sources lets application development and delivery leaders truly gain confidence in application quality—so that they can release more frequently and get customer feedback faster. What makes the application module view so important to the complete lifecycle is that it is where the test assets created during your agile sprints and releases are stored and mapped to your application. This enables you to easily manage and understand your regression testing process. The importance becomes greater and greater as your agile teams are adding more features and completing more user stories that you need to know that you are not breaking old code and incurring technical debt.

You can quickly understand every application module’s development status by looking at its dashboard. Enterprises can define their application in modules that reflect architecture, define user stories and tests that align with application modules, and track defects by module.

In addition to that, The OData REST API makes it easy to leverage external reporting tools, to meet your special needs of measuring and analysis.

The global search capability is also important. It lets you quickly find any piece of information across projects and across information types by plain text search. For example, searching for “performance” gives you all the items relate to this keyword, including requirements, tests, and defects.

“Modern software metrics—speed, quality, and value—are based on continuous feedback from business partners and customers.”

Forrester: Build The Right Software Better And Faster With Agile And DevOps Metrics, August 2016.
Identify Trends from Data

According to a 2016 Forrester Research report, “In the future, machine-learning algorithms will correlate data to produce insights where today’s leaders simply see randomness. These capabilities will equip AD&D leaders with a deeper understanding of how to win, serve, and retain customers through better software delivery capabilities that they can improve over time.”*

In 2017 today, this is no longer a future statement. Choose your Enterprise Agile tool which provides predictive analytics capability.

Predictive analytics uses artificial intelligence statistical algorithms to identify patterns in the data that you have, and then accurately forecast what approaches might be most successful in the future. To benefit from your investment in software development, you need to understand how to make better decisions to improve outcomes. Predictive analytics delivers actionable insights into the software development and delivery processes, helping you more accurately meet delivery timeframes, evaluate the risks of upcoming releases, and improve the quality of software you produce.

Plan: Accurate planning and estimates help set up Agile projects for success, by analyzing past performance data for better development timelines.

Build: Improve efficiency and accuracy while writing code, to identify problematic code and avoid rework by identifying existing code for reuse. The same efficiency applies when finding causes of build failures. Predictive analysis is able to learn how to identify the root cause from the huge build log and tells humans where the problem is.

Test: Accelerate for continuous testing, predicting defects and reducing time to fix with recommendations for reusing existing tests.

Track: Promote test efficiencies and collaboration across development, testing, and operations with enterprise scale, using actual production data to minimize risks and costs.

Introducing ALM Octane

Micro Focus ALM Octane is part of the Micro Focus ADM Portfolio. It delivers the essential elements for supporting Enterprise Agile and DevOps. It is a tool that meets all the criteria described above.

With ALM Octane, enterprises can quickly and easily kick start their projects, whether Agile or non-Agile. There is no need to select and purchase add-ons from various vendors, and no need for complex integration configurations.

*ALM Octane provides a new approach to delivering applications with the speed, quality, and scale to accelerate business success. It enables agile organizations to make better decisions with a centralized, management platform that delivers quality applications across four key phases: Plan, Build, Test, Track.
ALM Octane provides a new approach to delivering applications with the speed, quality, and scale to accelerate business success. It enables agile organizations to make better decisions with a centralized, management platform that delivers quality applications across four key phases: Plan, Build, Test, Track.

What Makes ALM Octane Special Today

Working at enterprise scale to meet the application delivery needs of tens/hundreds of distributed teams in a business, ALM Octane manages quality and test management across complex application portfolios. An integrated toolchain with visibility and traceability in end to end lifecycle management reporting, ALM Octane facilitates business processes and rules for governance and compliance. Besides helping businesses transition with confidence from traditional waterfall development to hybrid application development, here are just some of the reasons that make ALM Octane special today:

- **Support for bimodal and transitional IT with a single product.** With ALM Octane, both Agile and non-Agile teams see the same single source of record.
- **Enterprise agility.** Seamless execution across planning, building, testing, and running applications reduces miscommunication and latency stemming from manual handoffs and IT silos.
- **Unified Project and Portfolio Management.** ALM Octane closely integrates with Micro Focus PPM, a leader in the Gartner Magic Quadrant for Integrated IT Portfolio Analysis Application.
- **Allows you to manage across apps, projects, teams, and geographies.**
- **Native SAFe (Scaled Agile Framework) support.**
- **Supports teams in various maturity levels and Agile practices (Scrum, Kanban, SAFe).**
- **ALM/QC sync ensures a smooth transition from traditional IT.**
- **Ability to capture, correlate and visualize testing and quality data.** It can enable a wide array of prebuilt and custom graphs and reports that are created and rendered to align with KPIs, trends, and highest impact areas.
- **Supports Agile and non-Agile teams.**
- **A simplified user experience; accessible on any platform or browser.**
- **Open architecture, integrated and extensible.**
- **Easily share and re-use assets, adding velocity, reducing technical debt.**
- **Tightly integrated quality ensures teams address issues sooner, saving time and money.**
- **Built with a DevOps management center—enabling lean, continuous process and pipeline management.**
- **Support enterprise large scale deployment.**
- **Support strong need for regulatory compliance.**
Conclusion—Enterprise Agile Readiness with ALM Octane

Don't assume the complexity, security, and scale of Enterprise Agile can be easily handled by any Agile project management solution. Before you even start to undertake Enterprise Agile, consider these key questions:

■ Do you have to spend a lot of time to make it scale, to figure out what plugins to add in order to do what you want, and to set up DevOps pipeline? Some of the seemingly cheap solutions cost large amounts of your time and make you lag behind the competition.

■ Does the solution simply give you a lot of test results but not the indications of application quality? Does it manage development activities and data aligned with application architecture, so that resources can be reused?

■ Does it let you learn the trend from existing data, and avoid unnecessary work?

■ Does it allow you to oversee the status of your company’s related projects together, and optimize your investment in application portfolio?

■ Does it enable all stakeholders to collaborate smoothly, yet secure your project data with authentication, encryption, and role-based access control?

ALM Octane addresses these questions and provides a means for your Enterprise Agile project to immediately hit the ground running—with the confidence that you can succeed.

Learn more at
www.microfocus.com/alm-octane