GitHub Actions adds CI/CD to Augment DevOps Pipeline Collaboration

August 08, 2019

By: [Melinda-Carol Ballou](), [Jim Mercer](), [Arnal Dayaratna]()

IDC's Quick Take

GitHub the leading source control management solution built on top of the Git open source project and home to the majority of the world's active code announced it is expanding GitHub Actions capabilities to include Continuous Integration/Continuous Deploy (CI/CD). GitHub Actions will be available as a limited beta and is targeted to go GA in November of 2019 during the upcoming GitHub Universe conference. The key focus for GitHub with this release is to improve teaming, reuse and collaboration beyond individual developers and groups and to expand earlier, limited DevOps orchestration capabilities by enabling broader and deeper CI/CD and incorporating that functionality as part of GitHub Actions to benefit the community.

Product Announcement Highlights

GitHub announced they are expanding their GitHub Actions capabilities to include CI/CD. GitHub Actions will be available as a limited beta and is targeted to go GA in November of 2019 during the upcoming GitHub Universe conference. GitHub Actions CI/CD development features include:

- Community – Actions and CI/CD pipelines can be shared and cloned within GitHub and will benefit the larger GitHub community.
- Platform Agnostic – Developers can create workflows build test and deploy on the platform and cloud
- Parallel testing – Allows concurrent testing that can be hosted on Linux, macOS, and Windows
- Live Streaming logs – Developers can monitor logs showing real-time activity

GitHub CI/CD will be made available free of charge for public repositories and self-hosted environments. Hosted private repositories will be charged per minute of usage with the actual cost varying by platform.

IDC's Point of View

At GitHub Universe 2018, GitHub first announced the launch of GitHub Actions. GitHub Actions automate the execution of workflow-based applications on the GitHub Platform and enable users to create applications using a drag-and-drop interface. Actions are defined as code that run in a Docker container. The code invocation is event-driven and can be combined to define a workflow-based application in GitHub Actions. Users have the option of creating their own Actions in a private repository or a public repository or using one of GitHub's prebuilt Actions. Notably, Actions can be sourced from open source or public repositories, thereby enabling developers to use Actions that have been contributed by the broader GitHub community.

GitHub indicates that although the uptake of GitHub Actions has been strong, many customers and community members felt there was a gap and that Actions should provide a proper CI/CD capability. Without full CI/CD capabilities developers have had to build their own complex CI/CD workflows or resort to integrating another CI/CD tool into GitHub. The GitHub team indicated that the new Action

CI/CD capabilities are in direct response to the feedback from the overall community. We expect that once Actions CI/CD goes GA, that due to the simplicity and efficiency of building code within GitHub, many customers will progressively choose Actions as their preferred CI/CD solution over time.

Developers can use CI/CD on GitHub Actions to automate continuous delivery to any cloud, push images to container registries, create multi-phased builds featuring release gates, run parallel jobs and execute tests. The availability of CI/CD functionality on GitHub Actions enables developers to focus on building applications as opposed to managing manual tasks related to application updates, releases and testing. As such, the support for CI/CD on GitHub Actions marks an important step toward transforming GitHub Actions into a full-fledged platform for application development.

Moreover, the ability of developers to use the graphical user interface of GitHub Actions in conjunction with CI/CD functionality illustrates the maturation of the platform's visually guided development functionality insofar as it now embodies the development-related automation that developers expect and desire. IDC views the support for CI/CD on GitHub Actions as a significant milestone in GitHub's transformation from a code repository to a platform that supports end to end application development more generally. GitHub Actions workflows can be reused as well as being able to be easily created, shared, and forked (similar to open source). This can enable users to both share and reuse best practices across the GitHub community, and also to share and standardize workflows and practices across organizations.

GitHub was able to respond promptly to this demand for CI/CD by leveraging the proven Azure Pipelines technology from its parent company, Microsoft. Last year at this time there was much GitHub community concern regarding Microsoft's corporate influence on GitHub and how that might impact the developer community and trajectory for GitHub's popular open source collaboration platform. This is a case where appropriate and helpful leverage of synergistic Microsoft technology exemplifies benefits of the Microsoft acquisition and indicates how proper boundaries between the two organizations and strong GitHub leadership is working out well for GitHub and the larger GitHub community to deliver needed capabilities for GitHub Actions. Microsoft's Azure Pipelines enabled ready access to a proven, broadly established and well adopted CI/CD solution; not only would it have taken GitHub considerably longer to develop their own CI/CD tool, it certainly would not have incorporated all the capabilities that GitHub attained via the Azure Pipelines codebase.

GitHub actions CI/CD will empower the world's largest open source community to share CI/CD configurations and workflows. This capability to share CI/CD configurations and workflows has the potential to accelerate development with respect to the construction of CI/CD pipelines by rendering CI/CD workflows reusable. Moreover, the sharing of CI/CD workflows promises to enhance innovation specific to CI/CD-related development that derives from improved transparency into how other developers and organizations leverage CI/CD in practice. (Separately, GitHub also determined that GitHub Actions scripts will no longer use HashiCorp's HCL scripting language – this will now be replaced with YAML, due to a variety of factors, including the increased complexity demanded by new features for the Actions language. GitHub indicated that preexisting HCL scripts will not need to be converted to YAML and they will continue to run with Actions.)

GitHub's direct competitor GitLab (among others), which is also based on the open source Git project, has had CI/CD capabilities since 2016. GitLab is smaller than GitHub but has focused on including a full DevOps stack with their solution. Having built-in CI/CD capabilities can be a huge time-saver for

development teams with the GitHub pull request being at the center to the developer workflow allowing faster identification of coding issues without the need to plug-in an external tool. Given that GitHub Actions CI/CD incorporates Azure Pipelines, it will be interesting to see how this might affect the competitive equilibrium between the two organizations as well as competitive marketing messages.

The industry has experienced some consolidation amongst the CI/CD DevOps tools vendors. All the major cloud providers are offering CI/CD solutions as part of their platforms and this is beginning to commoditize these capabilities. There is already a saturation in the market of small start-ups and open source solutions that are providing CI/CD capabilities. The net effect has been a series of acquisitions of CI/CD providers. Some recent examples are Travis CI purchased by Idera, CloudBees' acquisition of Codeship and more recently, of Electric Cloud (see *CloudBees Acquires Electric Cloud to Add Release Automation* IDC #lcUS45023019, April 2019), Oracle acquiring Werker, and JFrog's recent acquisition of Shippable. Further leading CI/CD players such as Perforce expanded its platform by acquiring Perfecto Mobile and Roguewave to augment CI/CD with quality and continuous testing functionality (see *Perforce Acquires Perfecto Mobile Seeking to Perfect Its Mobile and Digital Quality Strategy for DevOps*, IDC #lcUS44391818, October 2018). All these players are broadening their portfolio to differentiate themselves from the litany of DevOps point solutions. These vendors are trying to play a larger role in the overall DevOps pipeline and open up new market opportunities to play a more pivotal role in the enterprise. We can speculate that this commoditization of CI/CD capabilities is one of the market forces driving Microsoft and GitHub to offer Actions CI/CD, based upon a full-featured CI/CD solution, and at such an affordable price point. This also serves to expand GitHub's role in the larger DevOps tools pipeline, which may put additional pressure on the consolidating collection of vendors in the market.

Key Takeaways

- GitHub Customers will reap the benefits of having a full-featured CI/CD tool tightly integrated into their source control management such that they can trigger a build right from a pull request. This will further strengthen GitHub's competitive position with developers and enterprise customers looking to reduce the litany of DevOps tools they need to maintain and procure.
- This injection of Azure Pipelines technology into the new GitHub Actions CI/CD stands as an example of how well Microsoft has managed this acquisition and their commitment to GitHub and the larger development community.
- Benefits to the GitHub Developer Community include a free beta for all users now, and suggested workflows (based on language context and other factors) that can help enable a transition onramp for those evolving their CI/CD usage. Collaboration leveraging CI/CD is key to evolving beyond disparate individualized development and GitHub platform use to coordinate as part of the broader DevOps pipeline.
- Longer term, next year GitHub plans to make Actions available for GitHub Enterprise Server customers with an on-premises hybrid option (to enable users to retain their code and packages in their own data centers as GitHub orchestrates workflow). Also planned are self-hosted runners for VMs to enable workflows in that context as well.

Subscriptions Covered:

[Agile Application Life-Cycle, Quality and Portfolio Strategies](#), [Application Security and DevSecOps](#), [Developer Demographics, Communities and Skills](#), [DevOps Solutions: Software, Infrastructure and Transformation](#)