Editor: **Robert Blumen**
Katana Graph
robert@robertblumen.com

Dave Cross on GitHub Actions

Gavin Henry

From the Editor
In Episode 567 of "Software Engineering Radio," Dave Cross, owner of Magnum Solutions and author of *GitHub Actions Essentials*, speaks with Software Engineering Radio host Gavin Henry about GitHub actions. Topics covered include what developers can do with GitHub actions, continuous integration/continuous deployment, events that drive GitHub fine-grained action events, Action Marketplace, contexts, YAML, Docker base images, self-hosted runners, identity management, permissions, and dependency management. We provide summary excerpts below; to hear the full interview, visit http://www.se-radio.net or access our archives via RSS at http://feeds.feedburner.com/se-radio.—*Robert Blumen*

Gavin Henry: What is continuous integration (CI)?

Dave Cross: It's automating the bits of your project so you can measure quality. Every time you commit new code or change code into your code base, you can run processes that run unit tests, run a linter against your code base, and perform other quality metrics, such as measuring the complexity of the code or the coverage of your tests. In this way, CI produces visible numbers indicative of code quality.

The first thing you'd look at would be the coverage, to see how well the test suite matches the amount of code actually in the project. Having a test suite that covers the code base well means that it's easier to change code and know that you're not breaking things.

What is continuous deployment (CD)?

That's what comes after continuous integration. Once you're happy that your code is good or better than it was previously, you can automatically take that code from your source-code system and move it into production in a manner that is easy to reproduce, hopefully just pressing a button, and after some processes run, the code is up on your production server and running.

What are GitHub actions?

GitHub actions were introduced a couple of years ago. Many projects were using products like Jenkins, Travis-CI, or CircleCI to do CI and CD. GitHub actions is GitHub's answer to that. It allows you to define workflows, and the definition of those workflows sits within your code repository. Then, in response to various events, GitHub fires up a container and runs through the steps in your process, which allows you to do CI and CD.

But it isn't limited to that. It's an event-driven architecture to the extent

Digital Object Identifier 10.1109/MS.2023.3322339
Date of current version: 20 December 2023

that you can trigger your workflow to run in response to a source-code control event, such as when you get a pull request to your code base or when someone raises an issue in your code base. It also basically gives you a complete cron job implementation. You can trigger events or workflows on time purely, trigger workflows manually, get a button on the workflow page and say "just run this now." Or you can use it as a web hook: just make an http request to GitHub, and it will trigger your workflow.

Does GitHub offer cross-platform support?

The GitHub-supplied runners, the *containers*, are available in three different flavors: Ubuntu, Windows, and Mac OS. For each, there are some different versions available. When you set up a workflow definition, you say what it runs on. One of the easiest ways to get a workflow up and running is just to say that it runs on Ubuntu, and they will just pull down the latest version of their lightly modified Ubuntu container, and you can run it on there. But that also works for Windows and Mac OS. Because you can run your own containers too, there's nothing to stop you running on a container that runs a completely different OS.

Can you give examples of projects that have benefited from GitHub actions?

The software developer Simon Willison came up with the concept of *Git scraping*, which is powered by actions. He uses his Datasette software, which is good for looking at SQLite databases, and Git scraping is a way of building these databases. He uses the cron job functionality for triggering things, and he'll find a website that has interesting data in the form of a JSON file. In the GitHub workflow, he scrapes that JSON file, and then uses Git to do a *diff* between that and the previous version. Git will give him a history of the changes in the data. He is applying this to websites that are monitoring forest fires in California and things like that. Taking the differences and putting them in a SQLite database and Datasette builds websites that enable you to plot that data on a graph or depict interesting ways that the data have changed over time. That's a fun and different use of GitHub actions.

GitHub is giving you free access to running cron tabs on their service. So anything that you can think of that is a schedule—do some stuff and then store it either in GitHub or in a database—you can do from GitHub actions. I've also worked with CPAN, the repository of add-on code for Perl programs. The Perl community in CPAN does a lot of unit testing. So, I built a site called *CPAN Dashboard*. Anyone who writes CPAN modules can create a pull request to my site. All we need is their CPAN username to add them. The site then uses GitHub actions to run some software that pulls information about all of their CPAN modules from CPAN. It uses the meta CPAN API [application programming interface] and then produces a list of all of their modules.

SOFTWARE ENGINEERING RADIO

Visit www.se-radio.net to listen to these and other insightful hour-long podcasts.

RECENT EPISODES

- 574—Chad Michel, coauthor of *Lean Software Systems Engineering for Developers*, joins host Jeff Doolittle to discuss treating software development as an engineering discipline.
- 573—Varun Singh, chief products and technology officer at Daily.co, speaks with host Nikhil Krishna about the 30-year evolution of web protocols.
- 571—Jeroen Mulder, author of *Multi-Cloud Strategy for Cloud Architects*, discusses cloud, private cloud, and multicloud computing architectures and trends with host Robert Blumen.
- 570—Stanisław Barzowski of XTX Markets and a committer on the jsonnet project joins Software Engineering Radio's Robert Blumen for a conversation about the jsonnet programming language.

UPCOMING EPISODES

- Ori Mankali of Akeyless discusses the complexities and solutions of secrets management with host Nikhil Krishna.
- Host Brijesh Ammanath discusses how software engineers can master business communication with Josh Doody.
- Zach Lloyd and host Gregory M. Kapfhammer discuss how to implement and effectively use command-line terminals.

They also need to tell me which CI tools they are using, whether it's GitHub actions or Travis-CI or CircleCI, and it then goes away on a schedule and interrogates all of those services and builds badges for all of those modules on all of the CI services that that author uses. It produces a nice visual representation of all the modules that the authors have written and how well they are doing on the various CI services.

What would you want a software engineer to remember from our show?

A lot of teams already have a lot of resource invested in existing CI/CD solutions. GitHub has produced the GitHub Actions Importer, which allows you to easily move your workflows from a different system into GitHub actions. That's a good, easy way to try things out.

The main thing is, CI and CD are great and everyone should be using them, but GitHub actions aren't just that. They give you access to containers running on GitHub hardware, and the sky is the limit in what you can do with it.

ABOUT THE AUTHOR

GAVIN HENRY is the CIO of TelcoSwitch, founder of the recently acquired SureVoIP, and a Software Heritage Ambassador. Contact him at ghenry@surevoip.co.uk.

IEEE Software (ISSN 0740-7459) is published bimonthly by the IEEE Computer Society. IEEE headquarters: Three Park Ave., 17th Floor, New York, NY 10016-5997. IEEE Computer Society Publications Office: 10662 Los Vaqueros Cir., Los Alamitos, CA 90720; +1 714 821 8380; fax +1 714 821 4010. IEEE Computer Society headquarters: 2001 L St., Ste. 700, Washington, DC 20036. Subscribe to *IEEE Software* by visiting www.computer.org/software.

Postmaster: Send undelivered copies and address changes to *IEEE Software*, Membership Processing Dept., IEEE Service Center, 445 Hoes Lane, Piscataway, NJ 08854-4141. Periodicals Postage Paid at New York, NY, and at additional mailing offices. Canadian GST #125634188. Canada Post Publications Mail Agreement Number 40013885. Return undeliverable Canadian addresses to PO Box 122, Niagara Falls, ON L2E 6S8, Canada. Printed in the USA.