GitHub Probot Vs. GitHub Actions

Ronen Levinson

DevOps Engineer

ronenl@codevalue.net

About Me

Ronen Levinson

- Investments enthusiastic

- Photographer in free time

- World traveler

Agenda

- Webhooks & Events

- GitHub Probot

- Probot app development

- GitHub Actions

- Actions structure

- Probot Vs. Actions

Challenges in code & app workflows

- Life cycle of new branches

- Master & dev relations

- Record of code changes

- Many systems to support

- Maintenance of VMs

Webhooks & Events

Webhooks & Events

- a way for notifications to be delivered to an external web server

- Each event:

    - corresponds to a certain set of actions

    - Each event type has a specific payload format with the relevant event information

- When event is triggered, GitHub will send a HTTP POST payload to the webhook's configured URL

Payload Delivery

Payload Delivery

```
POST /payload HTTP/1.1
Host: localhost:4567
X-GitHub-Delivery: 72d3162e-cc78-11e3-81ab-4c9367dc0958
X-Hub-Signature:
sha1=7d38cdd689735b008b3c702edd92eea23791c5f6
User-Agent: GitHub-Hookshot/044aadd
Content-Type: application/json
Content-Length: 6615
X-GitHub-Event: issues
{
  "action": "opened",
  "issue": {
    "url": "https://api.github.com/repos/octocat/Hello-
World/issues/1347",
    "number": 1347,
    ...
  },
```
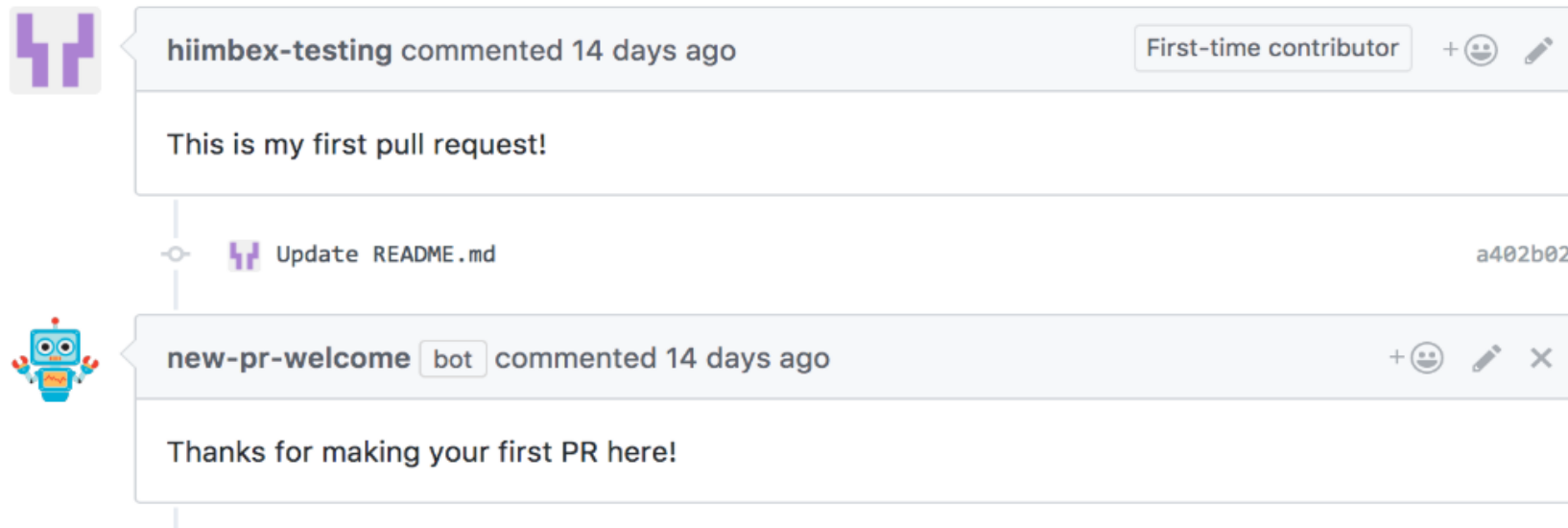
```
  "repository" : {
    "id": 1296269,
    "full_name": "octocat/Hello-World",
    "owner": {
      "login": "octocat",
      "id": 1,
      ...
    },
    ...
  },
  "sender": {
    "login": "octocat",
    "id": 1,
    ...
  }
}
```

GitHub Probot
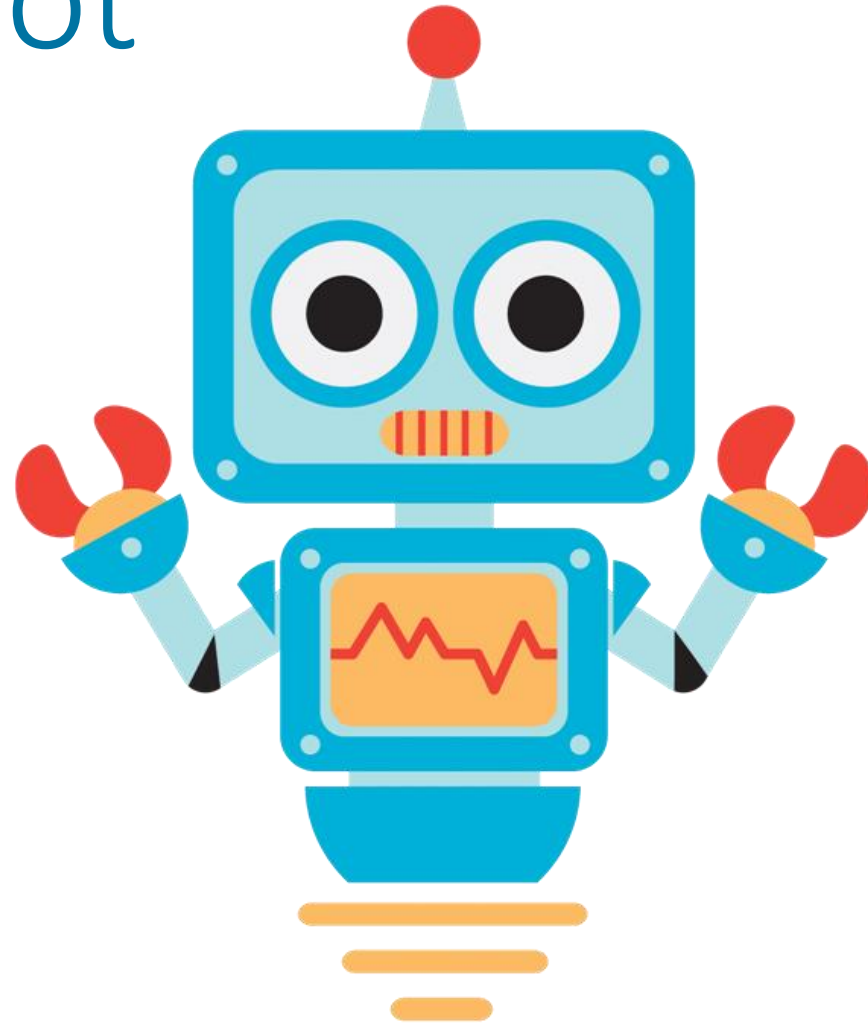
What is Probot?

- A framework for building GitHub Apps in Node.js

- The app will listen for webhook events triggered by GitHub

Demo:
Probot App Development

GitHub Actions

What is GitHub Actions?

- flexibility build an automated software development lifecycle workflow

- You can write individual tasks, called actions, and combine them to create a custom workflow

- Workflows are:
  - custom automated processes that you can set up to build, test, package, release, or deploy any code on GitHub

  - Configured by different triggers to start

Workflow.yml

```yaml
jobs:
  test:
    name: Test on node ${{ matrix.node_version }} and ${{ matrix.os }}
    runs-on: ${{ matrix.os }}
    strategy:
      matrix:
        node_version: [8, 10, 12]
        os: [ubuntu-latest, windows-latest, macos-latest]

    steps:
    - uses: actions/checkout@v1

    - name: Use Node.js ${{ matrix.node_version }}
      uses: actions/setup-node@v1
      with:
        version: ${{ matrix.node_version }}

    - name: npm install, build and test
      run: |
        npm install
        npm run build --if-present
        npm test
```

CODEVALUE

Workflow Log

GitHub Probot Vs. Actions
Which one to choose?

Strengths of Actions & Probot

GitHub Actions	GitHub Probot
GitHub runs your workflows in hosted VMs	Run fast and react to events quickly
Contains a secret store	Easy to create for your needs
SCM and CI/CD in one system	Can be deployed by the user
Build and test your code on different operating	Policy enforcement the "nice" way

GitHub Actions or Probot?

- To automate a task that makes use of your codebase - Actions

- Events that needs a quick response - Probot

- When you want to deploy your app by your own - Probot

- Interacting with external services that needs authentication - Actions

My Favorite Probot Apps

CODEVALUE

Delete merged branch

Request Info

My Favorite Actions

Upload artifact

Basic:

```
steps:
- uses: actions/checkout@v1

- run: mkdir -p path/to/artifact

- run: echo hello > path/to/artifact/world.txt

- uses: actions/upload-artifact@master
  with:
    name: my-artifact
    path: path/to/artifact
```

Where does the upload go?

In the top right corner of a workflow run, once the run is over, if you used this action, there will be a `Artifacts` dropdown which you can download items from. Here's a screenshot of what it looks like
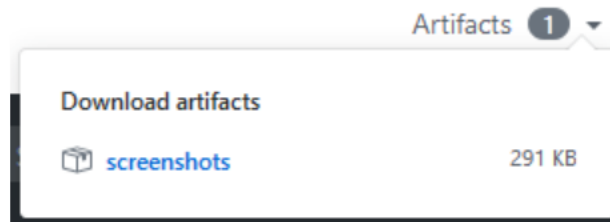
Kubernetes CLI - kubectl

Usage

`.github/workflows/push.yml`

```yaml
on: push
name: deploy
jobs:
  deploy:
    name: deploy to cluster
    runs-on: ubuntu-latest
    steps:
    - uses: actions/checkout@master
    - name: deploy to cluster
      uses: steebchen/kubectl@master
      env:
        KUBE_CONFIG_DATA: ${{ secrets.KUBE_CONFIG_DATA }}
      with:
        args: set image --record deployment/my-app container=${{ github.repository
          }}:${{ github.sha }}
    - name: verify deployment
      uses: steebchen/kubectl@master
      env:
        KUBE_CONFIG_DATA: ${{ secrets.KUBE_CONFIG_DATA }}
      with:
        args: '"rollout status deployment/my-app"'
```

CODEVALUE

And my most favorite one is…

Twitter, together!

GitHub Action

Twitter, together!

🏷 v1.0.4 Latest version

Twitter, together!

build mixed coverage 100% Greenkeeper enabled

Use latest version

Stars

★ Star 85

Contributors

Categories

Open Source management

Publishing

CODEVALUE

Places to find more Probot apps and Actions

Places to find more Probot apps and Actions

Thank you!

Ronen Levinson

DevOps Engineer

ronenl@codevalue.net

@Ronenrl