

Nova 2022

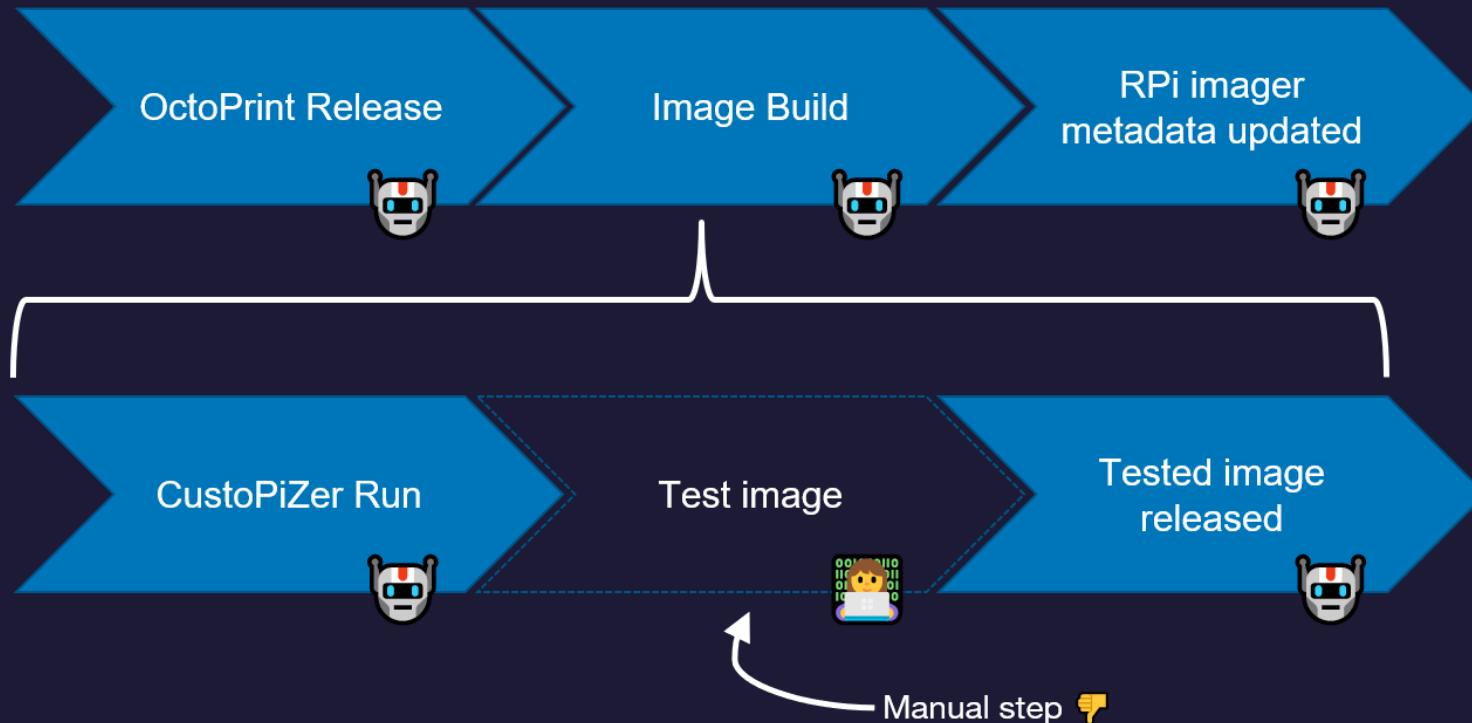
Tying custom hardware testing into GitHub Actions



Starting situation

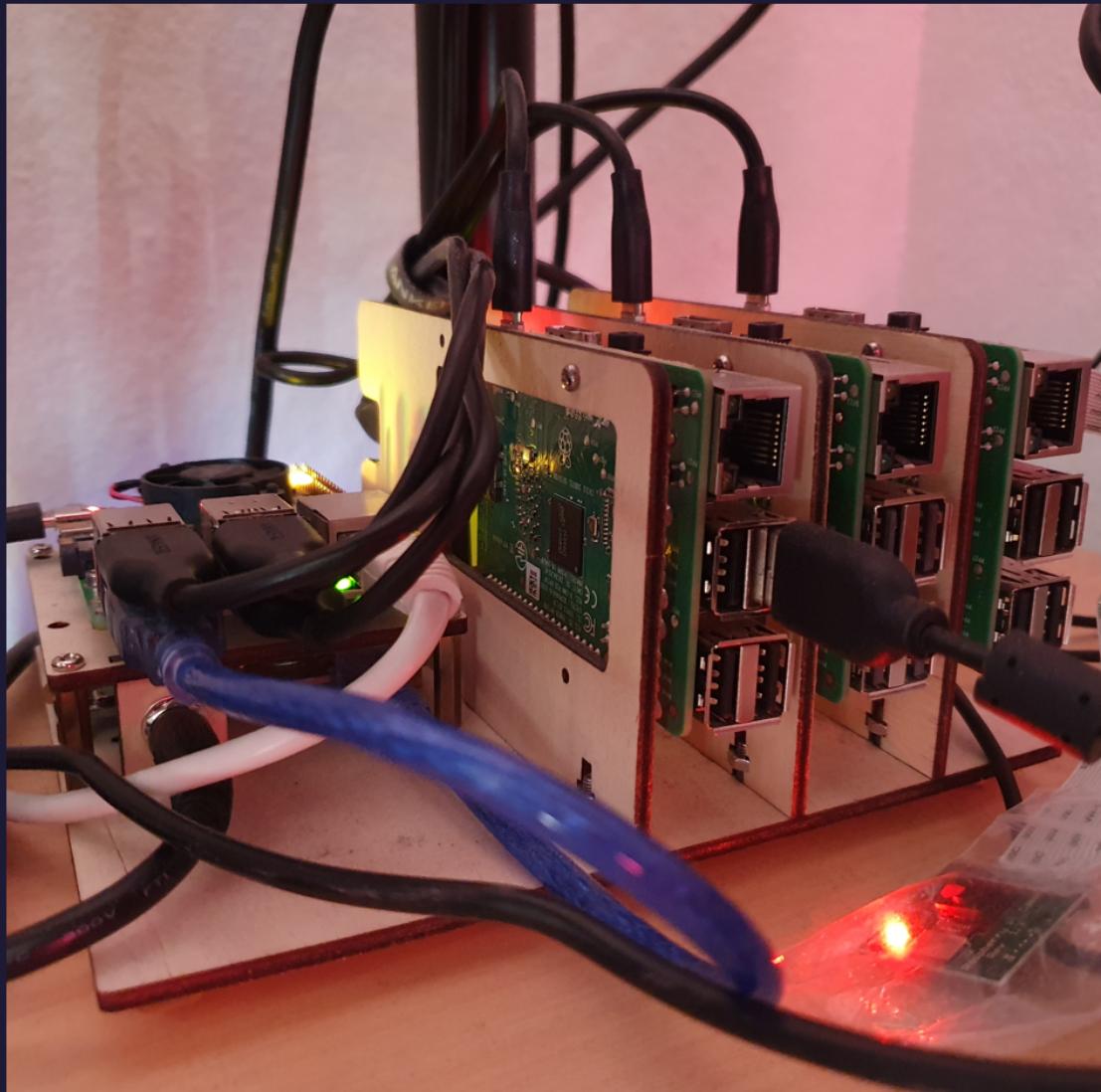
- Update existing RPi image for OctoPrint on each release
 - ➡️ OctoPrint/CustoPiZer
 - Image generation, preparation of input file for RPi image flashing tool
 - Talk @ Universe 2021: “RPi image customization with GitHub Actions”
 - Problem: Test before deployment? 😞





The Testrig™

- Built in 2020
- Used in OctoPrint's release testing
- One RPi 4 (flashhost), 3 RPi 3s (DUTs)
- ➡ OctoPrint/devtools



I've got an idea...



- Tailscale for network access
- Custom flashhost user & shell
- Workflow that gets remotely triggered by image build
 - Flashes & provisions image, preconfigures OctoPrint, runs available E2E test suite
- All green? Confirm & deploy in production!



Initial setup 1/2

- Tailscale network
 - flashhost & runner
 - port 22 & port 80
 - reusable & ephemeral auth key for runner

The screenshot shows the Tailscale interface for the domain `octoprint.org.github`. The top navigation bar includes links for Download, Support, Docs, and a user profile icon. Below the navigation is a menu bar with **Machines**, Services, Users, Access Controls, DNS, and Settings. The main content area is titled **Machines** and displays two entries:

MACHINE	IP	OS	LAST SEEN
<code>github-fv-az315-393</code> tag:github-runner No expiry: Ephemeral	100.78.105.162	Linux 1:14.0	Connected
<code>flashhost</code> fooseel@github No expiry	100.86.206.19	Linux 1:28.0	Connected

At the bottom, there is a link to "Connect more machines by [installing Tailscale](#)".



Initial setup 2/2

- Custom flashhost user & shell
- new user “fab”
- custom shell, only devtools
- SSH key used by runner

```
Linux flashhost 5.4.51-v7l+ #1327 SMP Thu Jul 23 11:04:39 BST 2020 armv7l

The programs included with the Debian GNU/Linux system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*copyright.

Debian GNU/Linux comes with ABSOLUTELY NO WARRANTY, to the extent
permitted by applicable law.
Last login: Wed Sep 21 11:57:25 2022 from 192.168.1.5

TARGET:
TAG:

fab shell. Hit Ctrl+D to exit

fab> flashhost_list_images
[pi@flashhost.lan] Executing task 'flashhost_list_images'
Available images:
  0.15.0
  0.15.1
  0.16.0
  0.17.0
  0.17.0-py3
  0.18.0-1.8.0rc5
  0.18.0-1.8.2
  0.18.0
  0.18.0-latest
  1.0.0rc1-32bit
  1.0.0rc1-64bit
  1.0.0rc2-32bit

Done.
Disconnecting from flashhost.lan... done.
fab> █
```



Tailscale & SSH

– □ ×

```
- name: "🔑 Install SSH key in agent"
  uses: shimataro/ssh-key-action@v2
  with:
    key: ${{ secrets.DEPLOY_KEY }}
    known_hosts: ${{ secrets.DEPLOY_KNOWN_HOSTS }}

- name: "🌐 Connect to tailscale network"
  uses: tailscale/github-action@v1
  with:
    authkey: ${{ secrets.TAILSSCALE_AUTHKEY }}
```



Deployment on Testrig

- □ ×

```
- name: "🔥 Flash & provision image"
  env:
    TARGET: ${{ github.event.inputs.target }}
    IMAGE: ${{ env.IMAGE }}
  run: |
    ssh -o SendEnv=TARGET fab@flashhost flashhost_flash_and_provision:$IMAGE octopi_wait:headless=1

- name: "🐙 Preconfigure OctoPrint"
  env:
    TARGET: ${{ github.event.inputs.target }}
  run: |
    ssh -o SendEnv=TARGET fab@flashhost octopi_test_provisioned:headless=1,config=configs/e2e
```



Summary

Jobs



Deploy



E2E

Deploy

succeeded 6 minutes ago in 7m 52s

- > Set up job
- > Install SSH key in agent
- > Connect to tailscale network
- > Check if tailscale is running
- > Determine image
- > Flash & provision image
- > Preconfigure OctoPrint
- > Clean up if needed
- > Complete job

E2E testing

— □ ×

```
- name: "⬇ Checkout latest OctoPrint"
  uses: actions/checkout@v3
  with:
    repository: OctoPrint/OctoPrint
    path: OctoPrint

- name: 🚀 Run Cypress
  uses: cypress-io/github-action@v2
  with:
    working-directory: OctoPrint/tests/cypress
    browser: chrome
  env:
    CYPRESS_BASE_URL: "http://flashhost/${{ github.event.inputs.target }}/"
```



✓ Test against testrig Test against testrig #38

Summary

Jobs

✓ 🚒 Deploy

✓ ✎ E2E

✎ E2E

succeeded 4 minutes ago in 2m 46s

- > ✓ Set up job
- > ✓ 🌐 Connect to tailscale network
- > ✓ ✅ Check if tailscale is running and make sure DUT is up
- > ✓ ⏪ Checkout latest OctoPrint
- > ✓ 🚀 Run Cypress
- ∅ ⏪ Upload screenshots
- > ✓ ⏪ Upload videos
- > ✓ ⏪ Post ⏪ Checkout latest OctoPrint
- > ✓ ⏪ Find DUT

Image build

- Build injects tag into release notes
- Testrig workflow is triggered
- Environment protection
- Tag is removed

```
test:  
  name: "Test"  
  runs-on: ubuntu-latest  
  needs: build  
  steps:  
    - name: 🚧 Run test against testrig  
      uses: jonas-schievink/workflow-proxy@v1  
      with:  
        token: ${{ secrets.DISPATCH_TOKEN }}  
        workflow: Test against testrig  
        repo: OctoPrint/devtools  
        ref: main  
        inputs: '{"image": "${{ needs.build.outputs.image-url }}"}'  
        repost-logs: true  
  
deploy:  
  name: "Deploy"  
  runs-on: ubuntu-latest  
  needs: [build, test]  
  environment: live  
  steps:  
    - uses: actions/github-script@v5  
      name: ✓ Mark release as tested"  
      env:  
        TAG: "${{ needs.build.outputs.tag }}"  
      with:  
        script: |  
          # remove untested tag
```



 **octoprint_release** CustoPiZe #29

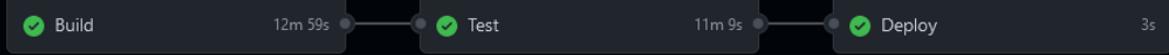
[Summary](#)

Re-run triggered 14 minutes ago Status:  foosel → ebdb552 Total duration: 12m 8s Artifacts: -

Jobs:

-  Build
-  Test
-  Deploy

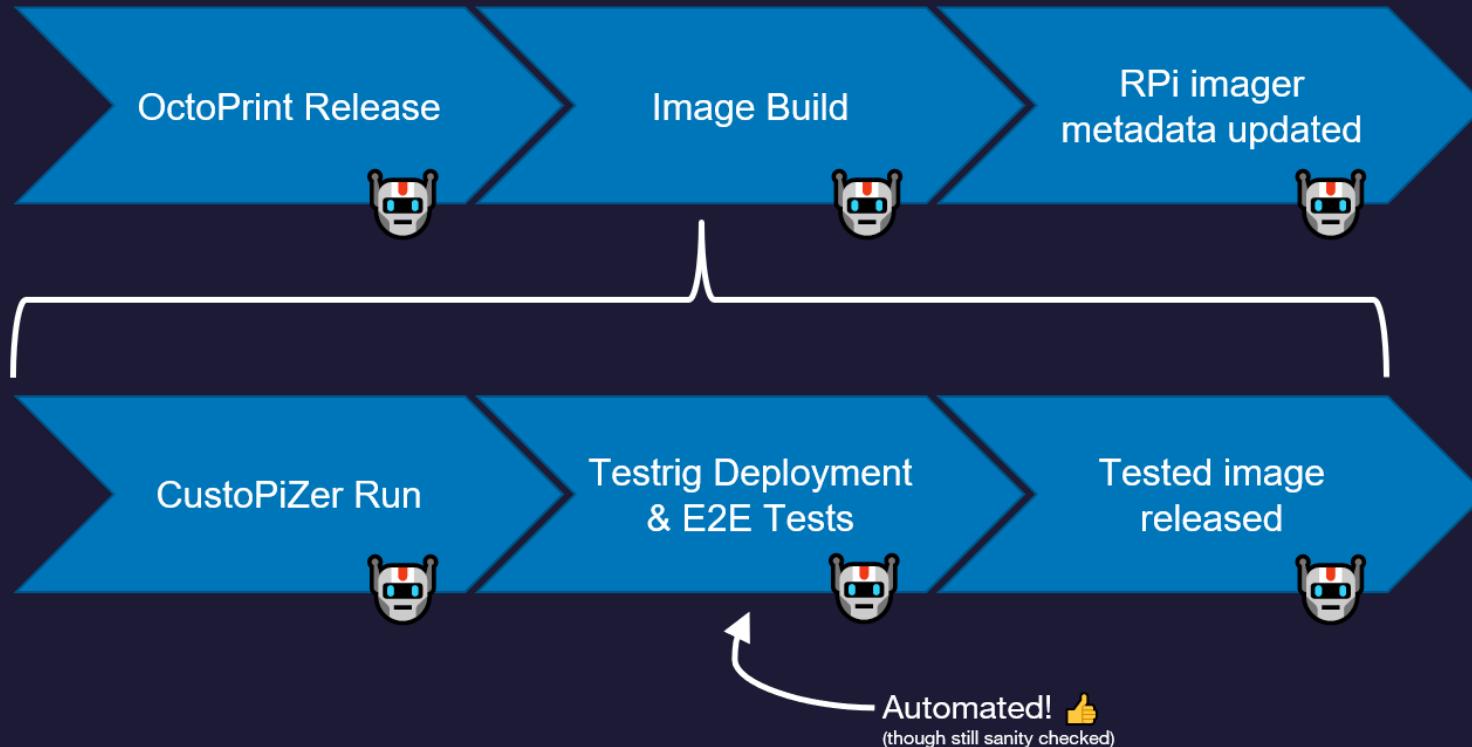
custopize.yml
on: repository_dispatch



Deployment reviews
Learn who reviewed the deployments in this run

Event	Environments	Comment
 foosel approved 2 minutes ago	live	Tests are green, let's ship this!





Where's the code?

- OctoPrint/CustoPiZer
- OctoPrint/devtools
- OctoPrint/OctoPi-UpToDate



