EdgeX Foundry

DevOps WG Meeting 07/16/18

07 July 2018 / 1:00 PM CT

ATTENDEES

Jeremy Phelps, Brandon Forster, Keith Steele, Trevor Conn, Michael Hall, Alain Pulluelo, Jim White, Andrew Foster

AGENDA

Last Meeting Follow-up

1. Starting Fresh

Current State

1. Jenkins: There are 3 categories of jobs.
    a. Verify - Verify jobs are triggered when a PR is opened, updated, and when a comment of `recheck` is issued. These jobs do a code build and run unit tests generally. They are required to pass before a PR can be merged.
    b. Merge - Merge jobs are triggered when a PR is merged. These jobs do a code build, run unit tests, and push SNAPSHOT artifacts up to nexus. This builds on the tip of the branch that just got updated with your PR.
    c. Release/Stage - Release/Stage jobs are triggered on a daily cron. The do a code build, run unit tests, and push artifacts up to a staging repository; these should be considered release candidates. The staging repositories are also where integration testing is concentrated (black box tests). Note that we currently only run integration tests in a containers and not on raw artifacts.

2. Current services in California

   It might be that we do not actually need all of these and would be super nice to drop some of them from consideration.

   a. - device-virtual
      i. 0.6.0 at 23e45aea7e6c5841f57c3d9709de08525306f4cd
   b. - device-modbus
      i. 0.6.0 at a5cf756bbc2dc0774524ddbf4986069b7441542b
   c. - device-controller
      i. 0.6.0 at f8088636981b5baab2bbe049c540100e273b6f74
   d. - device-sdk
      i. 0.6.0 at d6be2505713bb48068f95a1826dc690f842e2518
   e. - device-snmp
      i. 0.6.0 at 70a26920595279b0fc0174007cec8a11e8c98db6
   f. - device-fischertechnik
      i. 0.6.0 at 5104e47e546883ab1edc2a094e9aa6c8ab39f611
   g. - device-scheduling
      i. 0.6.0 at 7d5359b1dcca3aac88d01bc8f88344dcb002eae2
   h. - device-bacnet
      i. 0.6.0 at 084c2496cb5c6b48d6bb95395cfdf04ca1cb0738
   i. - device-mqtt
      i. 0.6.0 at df08f3d9bddd161c46efa109bf1e543e64a8a2f7
   j. - device-domain
      i. 0.6.0 at ec951bba9d26888a76d3507f2ac56c8632ea8a8b
   k. - device-sdk-tools
      i. 0.6.0 at 976d60ca281ec18cf7d08c46945566f8ce3a3df8
   l. - device-bluetooth
      i. 0.6.0 at 9c3037d577ddd67f0a31b46157f2568b575f6fc7
   m. - core-config-seed-go
      i. 0.6.0 at b4f8859ef46b9479c5b652270079bbb23ce5a5c2
   n. - core-config-watcher
      i. 0.6.0 at 22f68b3b46392af84825e2d55dd041b25024863d
   o. - core-test
      i. 0.6.0 at 7080e523585cc150ea66c7e360171ca894277cb8
   p. - support-notifications
      i. 0.6.0 at 5d4757a97b3e21d74056f105995364d0fe505fcb
   q. - support-rulesengine

i. 0.6.0 at 0e12199853d3d3d8ddf0150254c255a48b1d23dd

r. - support-scheduler

i. 0.6.0 at ba39a322eda9eb3d1ec0a5d443f198cecfd3cde8

s. - edgex-go

i. 0.6.0 at 4efd350c59655db0695c3f3c60434de35da1bfc3

t. - export-domain

i. 0.6.0 at 1413e08b5b9af13b6c0acb3b695f9c443bdf74f3

u. - export-test

i. 0.6.0 at 54409f65703cdc04abca4cb180cdf684951ded0f

v. - docker-edgex-mongo

i. 0.6.0 at f4cdf4d182d5e8828cfa50e53ac93b923b3df3be

w. - docker-edgex-volume

i. 0.6.0 at 2d5a4ef3ce786cb5f07daf01b2c6572586d15ea9

x. - security-api-gateway

i. 0.1.0 at 7d48865cdca257d4c78deb841e962978da6c3d90

y. - security-secret-store

i. 0.1.0 at affe358cdba36094b96e809ff669d7ba030f95ea

z. - blackbox-testing

i. 0.6.0 at ae8c5bb968f6e4fb98ed969ad968b6e7e0ea389b

3. Java services with x86 and arm64 containers

a. Device-virtual

b. Device-modbus

c. Device-snmp

d. Device-fischertechnik

e. Device-bacnet

f. Device-mqtt

g. Device-bluetooth

h. Support-notifications

i. Support-rulesengine

j. Support-scheduler

4. Java services with Jar only
    a. Export-domain
    b. Export-test
    c. Core-config-watcher
    d. Core-test
    e. Device-controller
    f. Device-sdk
    g. Device-sdk-tools
    h. Device-scheduling
5. Container services only
    a. Docker-edgex-mongo
        i. X86 and arm64
    b. Docker-edgex-volume
        i. X86 and arm64
    c. Security-secret-store
        i. Vault and Vault worker x86 only.
6. Golang services
    a. Edgex-go
        i. No aarch64 Dockerfiles defined (maybe we don't need them)
    b. Core-config-seed-go
    c. Security-api-gateway (x86 Docker only)
7. Development workflow
    a. EdgeX has a common workflow but we need to formalize it and document it. I have some initial docs here https://wiki.edgexfoundry.org/display/FA/Contributor+Process and here https://wiki.edgexfoundry.org/display/FA/Proposal+for+CODEOWNERS
    b. We are also in need to formalize our release process. Currently it is myself asking Jim what services will be included in the release. Generally we branch cut before the release date to allow folks to get fixes in before tag and release of artifacts. IMO this is the correct way to do it, but to my earlier point, we need to formalize it. (Long discussion ensues)

8. Integration testing (black box)
    a. This is currently an area of concern. It is one of our main validations but seems very loose at the moment. California integration tests run daily on x86 and arm64 hardware and grab california containers from staging. Master integration tests will do the same but grab artifacts generated from the master branch (work to do on master)
    b. Currently there are black box verify and testing jobs which run against a matrix of branch and cpu architecture.
    c. Black box tests run daily with email notifications on failure.

9. ARM testing on Cavium hosted server.
    a. Grateful that Cavium was able to host this server for us, but we are growing integration tests to a volume where we will benefit from dynamically provisioned builders. Luckily our provider has recently acquired some arm hardware and I’m in the process of migrating the arm jobs to that infra.
10. Overlooking of postman/newman and consul arm64 containers
    a. Trevor and I discovered these had been completely overlooked. Luckily there is a published arm64 container for consul. I built a postman/newman arm64 container and published it in our nexus for consumption.

New Business

1. Release Manager Role
    a. I view this as administrative and possibly a good idea. I disagree that they would set the DevOps agenda (which should be guided by the community in order to ensure buy-in and consensus)

2. Formalize release process
    a. LF has some work going on to automate the signing and release of artifacts.
    b. I am working on similar automation for containers published to Docker Hub.
    c. Rethink branch cutting strategy?
    d. JIM 2 weeks in between branch and release Monday 8th for Delhi. (Stick to the schedule). And can we push out the release to November.
    e. Release manifest. We really need to enumerate what is to be released (We do this but is very informal). Info should include branch/repo, sha/version, and artifacts to be released.
3. Contribution Process
    a. Require unit tests and associated integration tests to be submitted with bugfix and features?
    b. JIM +2 We will have to run black box tests on edgex-go PR. How to do? Possibly put integration into code repos.
    c. TREVOR +2 Planning to occur in Core WG meeting, execution occur in DevOps.
4. Use of CODEOWNERS for monorepos
    a. This will enable us to require PR signoff for changes to specific areas in a monorepo. (changes in core will require sign-off from a core services committer for example).
    b. MICHAEL - If there are changes in multiple areas will it require PR sign-off from multiple folks?
    c. JIM and TREVOR - Leave this off for now (can revisit later if needed.)
5. Migration of arm jobs to LF infra
    a. I’ve got provisioning jobs for these images that are failing, ticket opened with provider today.
6. Arm32 infra
    a. Will also ask our provider about this. I think all I have to do is use arm32 images.

7. Reporting of Black Box Test Results
    a. Parsed from log or other mechanism?

   b. Published to where? (JUNIT or something similar JEREMY)
   c. Integration w/GitHub for issue creation. (JEREMY to explore)
   d. Metrics on test duration, pass/fail percentage (Jeremy to look at)
   e. Tests running for all of edgex
   f. Do we want to include black box testing run on every PR? Currently only the verify job runs which only brings up the containers.
   g. Infra driven by QA/Test group but tests from contributors?
   h. JIM - Reports post-merge.
   i. TREVOR - separate code build/test jobs from packaging jobs.(JEREMY)

8. Checklist for onboarding new projects.
   a. This will be after the project decides that we will in fact adopt a new project.
   b. What artifacts are desired (Binaries, docker containers, snaps), and all supported architectures.
   c. JIM - Is there a project in LF with a template to use?

9. More EdgeX community participation in DevOps
   a. These do not have to be but can include implementation details if you want (for example: anyone can submit a patch to the ci-management repo)
   b. More buy-in means we get consensus and the best approach going forward. (not a dictatorship)
   c. First start with this is re-starting the DevOps meeting singly. Before I had just tacked it onto the Core WG meeting. We may still discuss things in there but we need a meeting focused solely on DevOps.
   d. KEITH (Adjust time of the meeting..have Brett create poll)

10. 0.6.1 release of California

a. Some bugs were found in rulesengine and fixes have been made in california. In addition we made some changes to the Dockerfiles in all of the java services. As soon as we get blackbox and snap passing we will need to create a 0.6.1 release.

b. Snap builds are still failing on CI and locally.

c. Black box tests are still failing on CI.

11. Create repos for device-sdk-(go|c)

    a. Steve and Tony, if you could send over the build script for each of these as well and any tools required it will speed up the process.

12. Port java service Docker file refactors to master branch (only makes sense to do if we are going to release java services in Delhi). JEREMY port this to master.
13. Deprecation and archiving of java services. POST DELHI
14. Make sure all generated containers have the git-sha and VERSION interpolated into the build as a LABEL. This is currently done on edgex-go and core-config-seed-go containers. Need to do in java service repos and security* repos.
15. In an effort to make it easier for new folks to build, I'd like to propose that we use Make. This is relatively easy to set up and provide simple instructions for getting something up and running. If we do this we need to make sure that we have commonly named targets across repos.
16. EdgeX has the option of adopting ci-management as a community repository. Other projects have done this (Hyperledger for example). JIM and KEITH looking at bringing extra folks in.

NOTES

1. ...

ACTION ITEMS

1. ...

NEXT WEEK'S AGENDA

...