• Topics
  – Submitting a patch to OMEC
  – OMEC CI/CD
Pull Request flow

1. Create a Pull Request
   - Fork the repo of interest
     - make changes and then push the change to a branch on your fork
1. Create a Pull Request as per github workflow [cont'd]

   - Submit a Pull Request from your fork to the original OMEC repo (see [https://help.github.com/en/articles/creating-a-pull-request-from-a-fork](https://help.github.com/en/articles/creating-a-pull-request-from-a-fork))

     a. Go to the original repo and click New Pull request

     b. compare across forks and select your fork / branch

     ![Diagram showing compare across forks and selecting fork/branch](image)

     c. Add title / description
     (if PR solves an issue, add “resolve #<issue no.>” in the description)

     d. Create Pull request
2. One of the OMEC Project maintainers (https://github.com/orgs/omec-project/people) will review your patch set. If it's approved, a comment with `ok to test` will be left on the pull request, and a OMEC Jenkins Job (https://jenkins.opencord.org/view/OMEC) will be started to test your patch set.

3. If the patch set passes tests, it can be merged into the project.

4. If tests fails, Jenkins can be inspected for details via Console output and log inspection.
• Topics
  – Submitting a patch to OMEC
  – OMEC CI/CD
A Pull Request for one of the three repos:
- c3po
- ngic-rtc
- openmme

will trigger a Jenkins job (omec_<repo>_combined) that will coordinate:

- VNFs installation (in parallel): (*)
  - omecc3po-hss_install
  - omecc3po-sgx_install
  - omeccopenmme_install
  - omeccngic-rtc_install

- and test:
  - omectc1 [ZMQ-KNI Regression TC1]
  - omectc2 [UDP-STATIC_ARP Regression TC2]

(*) all repos will be on master, except <repo>, which will be on pull/<PR number>/head
• Resources for CI/CD Jobs:
  – Jenkins scripts are hosted @ [https://github.com/omec-project/omec-project-ci](https://github.com/omec-project/omec-project-ci)
  – VNFs configuration for OMEC CI/CD is stored in OMEC VNFs repos (in .ci directory)
  – Migration of installation/test logic from Jenkins scripts to local bash scripts in .ci directories is ongoing
  – Jenkins scripts:
    • written as per pipeline architecture using both Declarative and Scripted pipeline
    • goal is to use only Declarative pipeline

• patch set involving 2 different repos at the same time
  – not manageable under current setup
  – shall be considered a corner case and handled ‘manually’ one patch at a time

• Pull request from HSS or SGX CDR Dealer/KMS will trigger a build/install job on whole c3po repo
ONF set-up is currently running in Intel vLabs, with plans to migrate to ONF
- VMs that are running in the ONF cluster for OMEC
  - ngic-cp1
  - ngic-dp1
  - c3po-hss1
  - c3po-mme1
  - c3po-db1
  - sgx-kms-cdr: SGX CDR Dealer/KMS components is running on bare metal
  - plrs2: polaris load tester is running on bare metal -> to be replaced by ng40

Activity is in progress to migrate from polaris to ng4t test system
- ilnperf7 has now replaced plrs2
- CI/CD is currently 'on-hold' to allow this migration to happen

Prerequisites for Jenkins CI/CD
- VMs up and running
- DB VM is left untouched as it is provisioned for 100,000 Ues
  (to check for the status run this command on DB VM: nodetool status)
- Ethernet connectivity between the servers and switches in the ONF Cluster environment

All existing processes of OMEC in ONF Cluster servers are killed on CI/CD execution/job trigger from Jenkins
• Traffic tests are set to default parameters for Jenkins CI/CD execution as follows:
  – nUE = 32000
  – attachRate = 250
  – detachRate = 250
  – curDataRate = 35000 Kbps

• ZMQ-KNI Regression TC1:
  – traffic test with ZMQ (Push/Pull) enabled in ngic-cp/dp and KNI enabled

• UDP-STATIC_ARP Regression TC2:
  – traffic test with UDP enabled in ngic-cp/dp and STATIC_ARP enabled

• Job stage “Polaris test” result is based on existing Polaris report (Success/Failure), will be updated when ng40 is up and running

• Enclave/ca_bundle.h in SGX CDR Dealer/KMS should maintain the schematic of the Enclave Measurements structure for editing the keys during Jenkins job execution
Concurrency of PRs

If a PR is being tested and a second PR is submitted or retested, Jenkins will queue the 2nd PR until the first one is completed.

E.g.:
- Job `omec_ngic-rtc-combined #21` will be executed only after `omec_c3po_combined #4` is completed.
CICD Flow

In Jenkins [https://jenkins.opencord.org/view/OMEC/](https://jenkins.opencord.org/view/OMEC/) results are visible/browsable via:

- the standard interface
e.g. for a PR on ngic-rtc repo
  [https://jenkins.opencord.org/view/OMEC/job/omec_ngic-rtc_combined/](https://jenkins.opencord.org/view/OMEC/job/omec_ngic-rtc_combined/)

- the Open Blue Ocean interface
e.g. for a PR on ngic-rtc repo
  [https://jenkins.opencord.org/blue/organizations/jenkins/omec_ngic-rtc_combined/activity/](https://jenkins.opencord.org/blue/organizations/jenkins/omec_ngic-rtc_combined/activity/)

Outputs:
- Build jobs Console Outputs are verbose and provide a step-by-step view, for easier debug in case of failure
- VNFs stdout / stderr consoles during installation and tests are logged to file and available in Jenkins
<table>
<thead>
<tr>
<th>S</th>
<th>W</th>
<th>Name</th>
<th>Last Success</th>
<th>Last Failure</th>
<th>Last Duration</th>
<th>Fav</th>
<th>Robot Results</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>docker-publish-github_c3po</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>docker-publish-github_ngic-rtc</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>docker-publish-github_openmm</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>docker-publish_omec-cni</td>
<td>3 days 11 hr - #3</td>
<td>N/A</td>
<td>4 min 37 sec</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>docker-publish_omec-pod-init</td>
<td>3 days 11 hr - #3</td>
<td>6 days 0 hr - #1</td>
<td>53 sec</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>omec_c3po-hss_install</td>
<td>3 days 8 hr - #228</td>
<td>1 mo 0 days - #153</td>
<td>6 min 28 sec</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>omec_c3po-sgx_install</td>
<td>3 days 8 hr - #193</td>
<td>13 days - #148</td>
<td>6 min 21 sec</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>omec_c3po_combined</td>
<td>3 mo 11 days - #23</td>
<td>3 days 8 hr - #48</td>
<td>26 min</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>omec_ci-test_pipelines</td>
<td>6 mo 6 days - #20</td>
<td>6 mo 7 days - #14</td>
<td>2 min 19 sec</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>omec_ngic-rtc_combined</td>
<td>6 days 21 hr - #204</td>
<td>6 days 23 hr - #203</td>
<td>28 min</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>omec_ngic-rtc_install</td>
<td>3 days 8 hr - #343</td>
<td>11 days - #302</td>
<td>6 min 2 sec</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>omec_openmmme_combined</td>
<td>5 days 12 hr - #72</td>
<td>3 days 14 hr - #75</td>
<td>31 min</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>omec_openmmme_install</td>
<td>3 days 8 hr - #358</td>
<td>9 days 16 hr - #346</td>
<td>37 sec</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>omec_tc1</td>
<td>5 days 12 hr - #446</td>
<td>3 days 8 hr - #456</td>
<td>11 min</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>omec_tc2</td>
<td>5 days 12 hr - #181</td>
<td>10 days - #174</td>
<td>11 min</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Example of PR -> CICD flow for PR #85 on ngic-rtc

- [https://github.com/omec-project/ngic-rtc/pull/85](https://github.com/omec-project/ngic-rtc/pull/85)

Fix issue #47 PGWU: Not able to launch, failed to create Rx/Tx rings #85

- This commit fixes the issue where the DP in PGWU configuration fails with error creating Rx/Tx rings. Please review

- **Review required**
  - At least 1 approving review is required by reviewers with write access. Learn more.

- **All checks have passed**
  - 1 successful check

- **Merging is blocked**
  - Merging can be performed automatically with 1 approving review.
Example of PR -> CICD flow

- https://jenkins.opencord.org/view/OMEC/job/omec_ngic-rtc_combined/204/console

Console Output

GitHub pull request #85 on commit 33229fc6c799c6e9b320e89a4378b8c0009e5f7, no merge conflicts.

Scheduling project: **omec_tcl**
Starting building: **omec_tcl #444**

Scheduling project: **omec_tcl2**
Starting building: **omec_tcl2 #180**

Setting status of 33229fc6c799c6e9b320e89a4378b8c0009e5f7 to SUCCESS

RTC combined/204/...and message: 'Build finished.'
Using context: CORD Jenkins
Finished: SUCCESS
Pipeline omec_ngic-rtc_combined

- https://jenkins.opencord.org/view/OMEC/job/omec_ngic-rtc_combined/

Stage View

<table>
<thead>
<tr>
<th>Stage</th>
<th>Average Time</th>
</tr>
</thead>
<tbody>
<tr>
<td>Declarative: Check out SCM</td>
<td>1s</td>
</tr>
<tr>
<td>Install</td>
<td>35ms</td>
</tr>
<tr>
<td>Install c3po-hss</td>
<td>6min 42s</td>
</tr>
<tr>
<td>Install c3po-sgx</td>
<td>6min 27s</td>
</tr>
<tr>
<td>Install ngic-rtc</td>
<td>6min 43s</td>
</tr>
<tr>
<td>Install openmme</td>
<td>45s</td>
</tr>
<tr>
<td>ZMQ, KNI Regression TC1</td>
<td>9min 30s</td>
</tr>
<tr>
<td>UDP STATIC ARP Regression TC2</td>
<td>5min 26s</td>
</tr>
</tbody>
</table>

#204

Aug 26
13:26
No Changes
Pipeline omec_ngic-rtc_install

- https://jenkins.opencord.org/job/omec_ngic-rtc_install/335/

Build #335 (Aug 26, 2019 11:20:29 AM)

Build Artifacts
- cical_cp1_install_stder.log 19.96 KB view
- cical_cp1_install_stdout.log 182.67 KB view
- cical_dp1_install_stder.log 23.10 KB view
- cical_dp1_install_stdout.log 231.55 KB view

Started by upstream project omec_ngic-rtc_combined build number 204 originally caused by:
- GitHub pull request #85 of commit 33229fc8f799ec6ae3820e89a4378b8c0009e9f7, no merge conflicts.

This run spent:
- 6.4 sec waiting;
- 7 min 1 sec build duration;
- 7 min 7 sec total from scheduled to completion.

Revision: 43b1f1e0539f2121d024344b67e6b535bdbc3e
- origin/master

Stage View
- VMs check 6s Install DP 4min 55s Install CP 1min 29s
- Aug 30 02:02 No Changes
- 4s 4min 32s 1min 20s
Pipeline omec_ngic-rtc_install

- https://jenkins.opencord.org/job/omec_ngic-rtc_install/2/console

Cloning into 'ngic-rtc'...
From https://github.com/omec-project/ngic-rtc
  * [new ref]  refs/pull/85/head -> jenkins_test

First, rewinding head to replay your work on top of it...
Applying: This commit fixes #47
commit 1894e105efc5988af6d752d8e1960e7a1717e28
Pipeline omec_tc1

https://jenkins.opencord.org/view/OMEC/job/omec_tc1/

Stage View

<table>
<thead>
<tr>
<th>VMs check</th>
<th>c3po-ctificdf</th>
<th>c3po-router</th>
<th>c3po-sgxcdr</th>
<th>c3po-hss1</th>
<th>c3po-mmne1</th>
<th>make CP</th>
<th>make DP</th>
<th>ZMQ [push/pull] start</th>
<th>run ngic-rtc</th>
<th>check processes</th>
<th>test polaris</th>
<th>test sgx-dp</th>
</tr>
</thead>
<tbody>
<tr>
<td>18s</td>
<td>10s</td>
<td>12s</td>
<td>27s</td>
<td>16s</td>
<td>36s</td>
<td>4s</td>
<td>12s</td>
<td>11s</td>
<td>38s</td>
<td>9s</td>
<td>7min 33s</td>
<td>2s</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30s</td>
<td>11s</td>
<td>11s</td>
<td>25s</td>
<td>18s</td>
<td>38s</td>
<td>4s</td>
<td>15s</td>
<td>11s</td>
<td>36s</td>
<td>9s</td>
<td>6min 55s</td>
<td>3s</td>
</tr>
</tbody>
</table>
Pipeline omec_tc1

- [Link](https://jenkins.opencord.org/job/omec_tc1/444/console) (similar report for tc2)
Examples of Failures (1/3)

Pipeline omec_ngic-rtc_combined
Created by omec-combined job-template from ci-management/job/omec-ci.yaml

Stage View

<table>
<thead>
<tr>
<th>Declarative: Checkout SCM</th>
<th>Install</th>
<th>Install c3po</th>
<th>Install ngic-rtc</th>
<th>Install openmme</th>
<th>ZMQ KNI Regression TC1</th>
<th>UDP STATIC ARP Regression TC2</th>
</tr>
</thead>
<tbody>
<tr>
<td>2s</td>
<td>36ms</td>
<td>9min 53s</td>
<td>13min 14s</td>
<td>40s</td>
<td>5min 33s</td>
<td>1min 38s</td>
</tr>
</tbody>
</table>

Average stage times:
(Average full run time: 28min 33s)

<table>
<thead>
<tr>
<th>#26</th>
<th>Mar 19, 2019 5:31 PM</th>
<th>Changes</th>
</tr>
</thead>
<tbody>
<tr>
<td>1s</td>
<td>33ms</td>
<td>9min 39s</td>
</tr>
<tr>
<td></td>
<td></td>
<td>12min 23s</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>#115</th>
<th>Mar 19, 2019 15:31</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Back to Dashboard
Status
Changes
Build with Parameters
Delete Pipeline
Configure
Full Stage View
GitHub
Job Config History
Open Blue Ocean
Rename
Pipeline Syntax
Examples of Failures

Add Dockerfile, Makefile targets and install_deps.sh #15

ksrnat729 commented on Feb 17 - edited

- Move to Ubuntu 18.04
- Added simple script to install all dependencies
  - Use packages for hyperscan
- Added a multi-stage Dockerfile to build SPGW-C/U
- Added docker related targets to top-level Makefile

Files: #6, #7
Based-on: #31
Signed-off-by: Saikrishna Edupuganti saikrishna.edupuganti@intel.com

Review requested
Review has been requested on this pull request. It is not required to merge. Learn more.

All checks have failed
1 failing check

CORD Jenkins — Build finished.

This branch has no conflicts with the base branch
Only those with write access to this repository can merge pull requests.
Examples of Failures (2/3)

- PR was submitted to ngic-rtc repo
- Job omec_ngic-rtc_combined #26 failed
- From there, it is possible to navigate to the failing point, in this case omec_ngic-rtc_install
- Pipeline view, Console Output and logs of omec_ngic-rtc_install will provide where the process stopped
Examples of Failures (3/3)

Pipeline omec_ngic-rtc_install
Created by omec-install job-template from ci-management/jb/omec-ci.yaml

Last Successful Artifacts
- icd_cp1_install stderr.log 19.96 KB
- icd_cp1_install stdout.log 182.32 KB
- icd_dp1_install stderr.log 23.12 KB
- icd_dp1_install stdout.log 231.22 KB

Recent Changes

Stage View

Average stage times:
VMs check 12min 5s 1min 29s
Install DP Install CP
5s 5s
12min 4s

Console Output

Started by upstream project "omec_ngic-rtc_combined" build number 26
originally caused by:
GitHub pull request #15 of commit 2c0588e2f53464cc1cccf57b94237e010ca4d4015,

/home/jenkins/ngic-rtc/dpdk/mk/ste.mk:936: recipe for target 'ngic_dataplane' failed
/home/jenkins/ngic-rtc/dpdk/mk/ste.mk:936: recipe for target 'all' failed

CC cc
CC gnu_echo.o
CC /home/jenkins/ngic-rtc/dp/./interface/udp/vep0_udp.o
CC /home/jenkins/ngic-rtc/dp/./interface/faceo.o
CC pipeline/pg1.o
CC pipeline/pgpcket_frameo.o
CC pipeline/pg_apas_dns.o
CC pipeline/pgp1.o
1D ngic_dataplane

34
Migration polaris -> ng40

- CI/CD is currently 'on-hold' to allow migration from polaris to ng40
- if a job is triggered now, an early failure of TC1 will be shown
Additional / backup slides
Pipeline `omec_c3po-hss_install`

- [https://jenkins.opencord.org/job/omec_c3po-hss_install/220/](https://jenkins.opencord.org/job/omec_c3po-hss_install/220/)

**Pipeline `omec_c3po-hss_install`**

*Created by `omec-install` job-template from `ci-management/job/omec-ci.yaml`*

### Last Successful Artifacts

<table>
<thead>
<tr>
<th>Artifact</th>
<th>Size</th>
<th>View</th>
</tr>
</thead>
<tbody>
<tr>
<td><code>hss_install.stderr.log</code></td>
<td>2.26 KB</td>
<td><img src="view" alt="View" /></td>
</tr>
<tr>
<td><code>hss_install.stdout.log</code></td>
<td>54.08 KB</td>
<td><img src="view" alt="View" /></td>
</tr>
<tr>
<td><code>hss_make.stderr.log</code></td>
<td>0 B</td>
<td><img src="view" alt="View" /></td>
</tr>
<tr>
<td><code>hss_make.stdout.log</code></td>
<td>6.59 KB</td>
<td><img src="view" alt="View" /></td>
</tr>
<tr>
<td><code>hsssec_make.stderr.log</code></td>
<td>0 B</td>
<td><img src="view" alt="View" /></td>
</tr>
<tr>
<td><code>hsssec_make.stdout.log</code></td>
<td>9.28 KB</td>
<td><img src="view" alt="View" /></td>
</tr>
</tbody>
</table>

**Recent Changes**

**Stage View**

<table>
<thead>
<tr>
<th>VMs check</th>
<th>Install HSS</th>
</tr>
</thead>
<tbody>
<tr>
<td>3s</td>
<td>6min 26s</td>
</tr>
</tbody>
</table>

### Build #220 (Aug 26, 2019 11:20:29 AM)

**Build Artifacts**

- `hss_install.stderr.log`: 2.26 KB
- `hss_install.stdout.log`: 54.08 KB
- `hss_make.stderr.log`: 0 B
- `hss_make.stdout.log`: 6.59 KB
- `hsssec_make.stderr.log`: 0 B
- `hsssec_make.stdout.log`: 9.28 KB

**Started by upstream project `omec-ngio-rtc_combined` build number 204**

- Originally caused by:
  - GitHub pull request #85 of commit 332299c8e799c8a8b3b20a39a94378b5c0009a8f7, no merge conflicts.

**This run spent:**

- 6.4 sec waiting
- 6 min 42 sec build duration
- 6 min 48 sec total from scheduled to completion

**Revision:** 43b11c85.39f292d0243446b67e6b535b6dcb31c
- `origin/master`
Pipeline omec_c3po_sgx_install

- https://jenkins.opencord.org/job/omec_c3po-sgx_install/185/

Create by omec-install job-template from ci-management/jb/omec-ci.yaml

**Stage View**

<table>
<thead>
<tr>
<th>VMs check</th>
<th>Install SGX. Dealer</th>
</tr>
</thead>
<tbody>
<tr>
<td>3s</td>
<td>6min 24s</td>
</tr>
</tbody>
</table>

- Started by upstream project omec_npc-rtc_combined build number 204
- Originally caused by
  - GitHub pull request #86 of commit 332296c8b79db7e6b382e8a59a437b8c30009e87, no merge conflicts.
Pipeline omec_openmme_install

- [https://jenkins.opencord.org/job/omec_openmme_install/350/](https://jenkins.opencord.org/job/omec_openmme_install/350/)

Started by upstream project [omec-ngc-rtc_combined](https://jenkins.opencord.org/job/omec-ngc-rtc_combined) build number 204 originally caused by:

- GitHub pull request #85 of commit 33229fc8a799c6aeb3820a89a4378b8c0009a87f, no merge conflicts.

This run spent:
- 6.4 sec waiting;
- 37 sec build duration;
- 43 sec total from scheduled to completion.

Revision: 43b11c8639f2f2f21d024344b67eb68b5635b4dcb3fc
- origin/master
Pipeline omec_tc2

- [https://jenkins.opencord.org/view/OMEC/job/omec_tc2/](https://jenkins.opencord.org/view/OMEC/job/omec_tc2/)

### Stage View

<table>
<thead>
<tr>
<th></th>
<th>VMs check</th>
<th>c3pocctiddf</th>
<th>c3pocrouter</th>
<th>c3pogxcdr</th>
<th>c3pommmse1</th>
<th>ZMQ [push/pull] stop</th>
<th>make CP</th>
<th>make DP</th>
<th>run nginx-rt</th>
<th>check processes</th>
<th>test polaris</th>
<th>test sgx-dp</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aug 27</td>
<td>34s</td>
<td>15s</td>
<td>17s</td>
<td>35s</td>
<td>24s</td>
<td>8s</td>
<td>6s</td>
<td>18s</td>
<td>28s</td>
<td>11s</td>
<td>6min 56s</td>
<td>4s</td>
</tr>
<tr>
<td>Aug 26</td>
<td>14s</td>
<td>9s</td>
<td>10s</td>
<td>20s</td>
<td>18s</td>
<td>6s</td>
<td>6s</td>
<td>12s</td>
<td>28s</td>
<td>9s</td>
<td>6min 50s</td>
<td>3s</td>
</tr>
</tbody>
</table>

(Average full run time: ~10m 28s)
Thank You

Q & A