

Near-RT RIC (G release)

These are the planned versions to be used for the G release (G-release status: [link](#)).

Use the recipe file `RECIPE_EXAMPLE/example_recipe_oran_g_release.yaml` from ' `git clone "https://gerrit.o-ran-sc.org/r/ric-plt/ric-dep"` ' to deploy the near-RT RIC platform. This file is aligned with the versions below.

Component (docker image /library/module)	Image in release repo (10002) or in staging repo (10004)	Gerrit review (to be used to recreate image if disappeared from staging)
docker: ric-plt-e2mgr (E2 manager)	nexus3.o-ran-sc.org :10002/o-ran-sc/ric-plt-e2mgr:6.0.1	
docker: ric-plt-e2 (E2 termination)	nexus3.o-ran-sc.org :10002/o-ran-sc/ric-plt-e2:6.0.1	
docker: ric-plt-submgr	nexus3.o-ran-sc.org :10002/o-ran-sc/ric-plt-submgr:0.9.5	
docker: ric-plt-rtmgr	nexus3.o-ran-sc.org :10002/o-ran-sc/ric-plt-rtmgr:0.9.4	
docker: ric-plt-a1	nexus3.o-ran-sc.org :10002/o-ran-sc/ric-plt-a1:3.0.1	
docker: ric-plt-appmgr	nexus3.o-ran-sc.org :10002/o-ran-sc/ric-plt-appmgr:0.5.7	
docker: ric-plt-alarmmanager	nexus3.o-ran-sc.org :10002/o-ran-sc/ric-plt-alarmmanager:0.5.14	
docker: jaegeradapter	docker.io/jaegertracing/all-in-one :1.12	no changes. Same version as before from docker.io
docker: ric-plt-o1	nexus3.o-ran-sc.org :10002/o-ran-sc/ric-plt-o1:0.6.1	
docker: ric-plt-vespamgr	nexus3.o-ran-sc.org :10002/o-ran-sc/ric-plt-vespamgr:0.7.5 DONE	same as in E release, no new version
docker: xapp-onboarder (dms_cli)	<code>git clone -b g-release "https://gerrit.o-ran-sc.org/r/ric-plt/appmgr"</code>	
docker: ric-plt-dbaas (database)	nexus3.o-ran-sc.org :10002/o-ran-sc/ric-plt-dbaas:0.6.2	released on Apr-04-2022
docker: influxDB (optional)	docker.io/influxdb:2.2.0-alpine	Version as before from docker.io . Major upgrade compared to F release
docker: mgxapp (optional metrics gateway xApp)	not planned to be released in G release (See RIC-799)	
docker: ricdms	nexus3.o-ran-sc.org :10002/o-ran-sc/ric-plt-ricdms:2.0.0	
library: rmr	version 4.8.5 (Dec-07-2022)	https://gerrit.o-ran-sc.org/r/c/ric-plt/lib/rmr/+ /9998
library: mdclog	version 0.1.4-1 (Aug-25-2021)	same as in E release, now new version
library: ricxfcpp (xapp-frame-cpp)	version 2.3.8 (Dec-14-2022)	
library: riclibe2ap	not planned to be updated to E2APv.2.0 in G release due to lack of time (same as before: version 1.1.0 (Jun-04-2021))	
library: sdl (c++)	version 1.6.0 (Mar-14-2022) (no changes)	
library: rnib (c++ (=rnib... deb") and golang),	version 1.2.7 (Dec-12-2022)	
module: mdclogpy	version 1.1.4(Jun-17-2021)	same as in E and F release, now new version
module: ricsdl (sdl for python)	version 3.1.2 (Dec-12-2022)	
module xapp-frame-py (in pypi = ricxappframe)	version 3.2.1 (Dec-12-2022)	
module xapp-frame (go)	version 0.9.15 (Sep-01-2022)	
module xapp-frame-rust (rust)	<code>git clone -b g-release "https://gerrit.o-ran-sc.org/r/ric-plt/xapp-frame-rust"</code>	
module sdlgo (go)	version 0.10.2 (Sep-30-2022)	

End-of-release tasks

1. Complete code upload/update
 - a. **e.g. update the dockerfile to use the latest RMR library: 4.8.5 (includes a bug fix)**
2. Double-check from [Code coverage reports](#) (click on "primary link" in that page) that all **blocker** code smells have been addressed as per RC-5 in [Release criteria checklist template](#)
 - a. **status 2022-12-14: no further actions needed**
3. If applicable, release binary artifact(s) such as docker container image(s), or libraries.
 - a. Formal instructions: [O-RAN Developer's Guide to CI Resources and Processes at the LF](#). For container image releases we do check "container-tag.yaml" in the repo and from there you get the latest version of the container (*the expectation is that during development that version is updated as changes are made to the repo*). If it's the same as in the previous release (Cherry) because there were no changes, or because developers forgot to update that one, update it as well. This has to be committed and merged first. As a second step take the version from container-tag.yaml and update that into .../releases/container-release-**ric-plt-e2mgr**.yaml (in two places; in row "version" (skip this if the italics text above applied) and in row container_release_tag) also take the commit Id that is the latest and update that into /releases/container-release-**ric-plt-e2mgr**.yaml. The bold part obviously differs per component.
4. Update references to artifacts in the above table. Also fill in the last column with reference to the gerrit review that can be used to regenerate the artifact.
5. Update release notes: Make sure you have a release summary as per RC-1 in [Release criteria checklist template](#): "every component must update their rst release notes (in the doc subfolder in the repo)". Based on this per component description, the PTL will create a release summary.
 - a. Release Notes must be published to <https://docs.o-ran-sc.org/en/latest/>
 - b. Instructions for setting up documentation folder for repo: <https://wiki.o-ran-sc.org/display/DOC/Configure+Repo+for+Documentation>
6. After finishing all the above, branch "g-release" (all lower case with a dash between f and release).
 - a. Repo owners (committers) can create branch.
 - b. There are several ways to create branch. For example do this: Login to web UI <https://gerrit.o-ran-sc.org/r/admin/repos> > Find repo, click on repo name > "Branches" in right panel > "Create New"