

Near-RT RIC Cherry Plan

Thoralf Czichy (Nokia), Near-RT RIC PTL

<firstname>.<lastname>@nokia.com

O-RAN Alliance + O-RAN SC Kick-off discussion for Cherry, 2020-Jun-08

RIC items in release Cherry

- Jun-Aug 2020:
 - [filter10603](#) (incl. 13 stretch goals) = 40 items
 - [filter10605](#) (excl. 13 stretch goals) = 27 items
 - Generally how JIRA is used in near-RT RIC: [link](#)
- Sep 2020: two candidate items: RIC-136 (remove pre-compiled ASN.1) and RIC-375 remove usage of ASN.1 in E2 subscription manager interface used by xApps (only supporting the new REST interface)
- Work by Nokia, Samsung and AT&T
- Near-RT RIC all-we-know wiki page: [link](#), incl. [individual components](#)
- readthedocs.io on near-RT RIC: [link](#)
- Project meetings: [link](#)

Highlights (Cherry-picked)

- First actual platform alarm (framework exists only, xApp side already working), e.g. for E2 connection loss, message overload, RIC internal routing distribution timeouts - RIC-56
- HA: in all components as auto-restart, but database has replication between three instances RIC-139, RIC-76, RIC-421
- Operator-initiated E2 connection ping-pong healthcheck based on RIC SERVICE UPDATE - RIC-95
- Simple RIC-internal time-series database (RIC-114)
- RIC benchmarking (RIC-150)
- Reject new E2 connection requests in addition to closing existing ones (RIC-177)
- Actual platform statistics from A1 and E2 in addition to existing xApp stats - RIC-353, RIC-358, RIC-422
- xApp framework enhancements (go, python, C++) – RIC-365, RIC-367

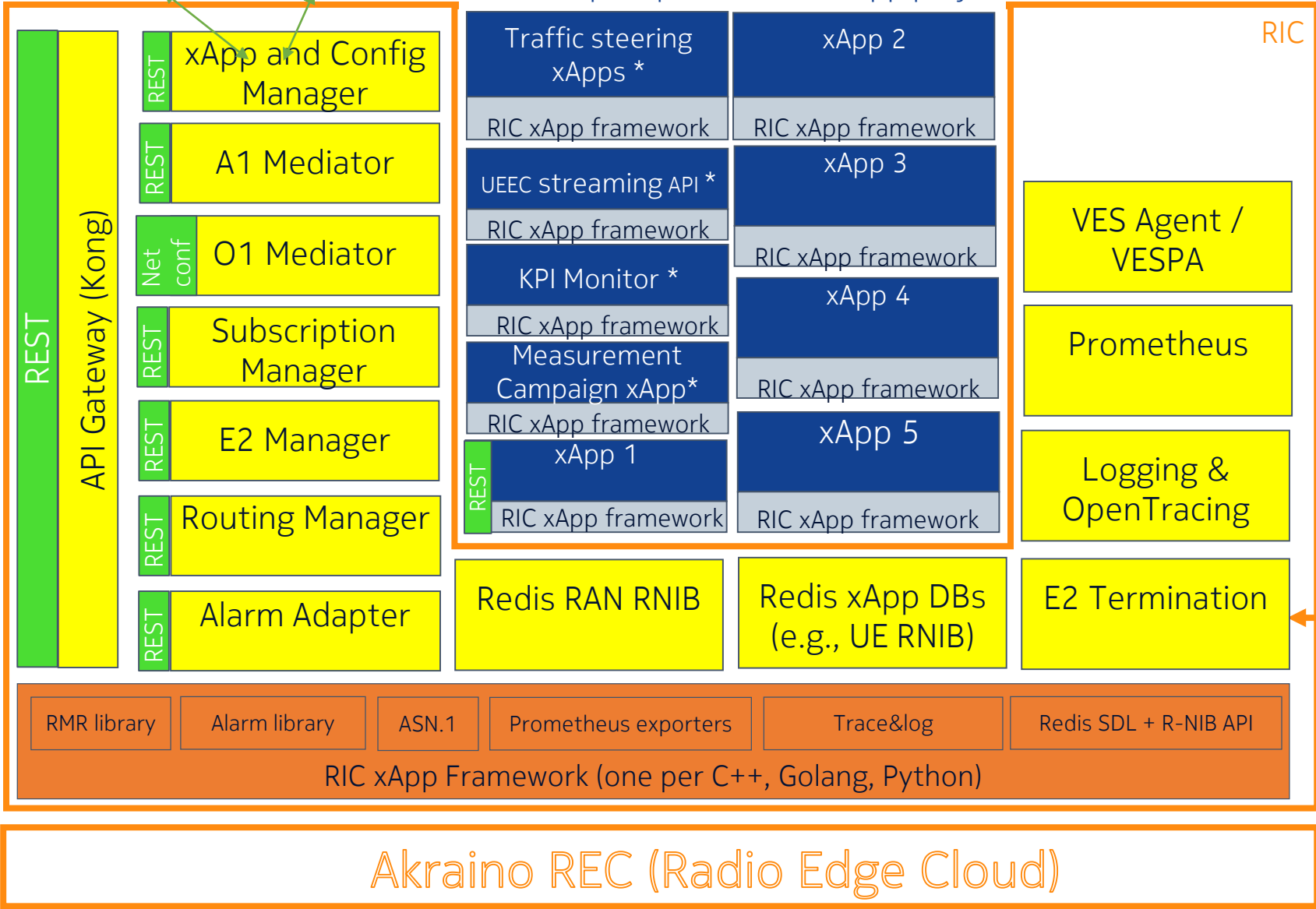
- Studies
 - Study on further message bus (NATS, Kafka, grpc, enhancements to RMR) - RIC-48
 - Study xApp scaling (RIC internally) – RIC-363
 - Study on helm and kubernetes upgrades – RIC-360

- Not yet
 - E2 RESET (not from RAN and not from near-RT RIC) - an issue of prioritization
 - Dynamic E2 scaling and E2 failover - waiting for WG3



* = example open src from xApp project

Management platform



RAN, i.e. gNB/eNB

E2

Bronze release statement in [link](#)

- The E2AP protocol implementation of the near-RT RIC platform has been adapted from one based on pre-specification assumptions to the E2AP protocol as defined by the O-RAN alliance. The new near-RT RIC O1 mediator implements the first O1 interfaces based on netconf and Yang models, e.g., to check the health status of xApps. The near-RT RIC internal messaging and message routing solution has been rewritten to significantly improve throughput, latency, and to behave more robustly in failure and high-load conditions. The Redis-based shared data layer now supports HA (high-available) deployments. Various Near-RT RIC platform components use it to store their state and recover that state in failure, restart and upgrade situations. xApps can use the same shared data HA later solution for their state. xApps that use E2AP subscriptions can now use the new E2AP subscription API instead of having to directly generate ASN.1 encoded E2AP subscription messages. In addition to a Go version, the Bronze release of the near-RT RIC platform now also provides a C++ and python framework to simplify the implementation of xApps.

Bronze release statement

same as previous slide, but as bullet list

- From pre-specification assumptions to the E2AP protocol as defined by the O-RAN alliance.
- New near-RT RIC O1 mediator implements the first O1 interfaces based on netconf and Yang models, e.g., to check the health status of xApps.
- The near-RT RIC internal messaging and message routing solution has been rewritten
 - significantly improves throughput, latency, and behave more robustly in failure and high-load conditions.
- The Redis-based shared data layer now supports HA (high-available) deployments.
 - Various Near-RT RIC platform components use it to store their state and recover that state in failure, restart and upgrade situations.
 - xApps can use the same shared data HA later solution for their state.
- xApps that use E2AP subscriptions can now use the new E2AP subscription API instead of having to directly generate ASN.1 encoded E2AP subscription messages.
- In addition to the existing Go version, the Bronze release of the near-RT RIC platform now also provides a C++ and python framework to simplify the implementation of xApps.