

A1 RIC Health-Check (Flow #3)

The A1 (A1-AP) interface is used to support the Non-RealTime RIC creating and managing policies for near-RealTime RICs in the RAN. The nearRT RICs can then use/pass these A1 policies/requests to hosted xAPPs for enforcement.

A1-related RIC Health-Check must ensure that: 1) the A1 interface is operational, and 2) policy changes (create, change, delete) can be processed.

The A1 RIC Health-Check flow supports the following requirements/Epics:

- Provide E2E Health-Check - Policy Changes via A1 interface
- Non-RT RIC support of A1 messages - Policy list queries
- RIC Support of A1 Policy Test Message Generation and Mediation
 - "HelloWorld" Test Policy Instance Creation and Deletion
 - xAPP process of "HelloWorld" Policy Creation and Deletion


For A1 interface health in the nearRT RIC, the nearRT RIC Platform is already defining a self-check mechanism (see [RIC Self-Check page](#)) in which the nearRT RIC A1 Mediator will support internal health-check requests (*get_healthcheck* request defined). And if there are any anomalies, an alarm will be generated via O1VES and the RIC alarm-list will be updated. So there is no need for the NB client issue a specific A1 health-check; NB client can just query for the alarm-list for A1 mediator health and other component/xAPP health that might hinder the policy configuration from being processed.

For A1 interface health in the NONRTRIC A1 Policy Management Service constantly monitors the consistency of the policies in all nearRT RICs and maintains a consistent view. If Inconsistencies occur a warning is logged and the inconsistency is resolved according to the last know intents specified by the actor creating/deleting/updating policies.

To verify whether policy configuration requests can be processed, it is recommended that a "HelloWorld" xAPP be defined that supports a unique "HelloWorld" policy type. New instances of this policy type can then be created and deleted by the test driver. The creation, deletion, status and consistency of these instances can then be checked to ensure healthy operation. These checks would then confirm that the following functions operate as expected: 1) NONRTRIC A1 Policy Management Service, 2) the NONRTRIC A1 Adapter, 3) the A1 connections, 4) the nearRT RIC A1 Mediator, and 5) the nearRT RIC "HelloWorld" xAPP.

Here is the sequence of operations:

- Get current policy list (to ensure "HelloWorld" policy instance is not present)
- Send via A1 a "HelloWorld" policy instance creation request
- Upon receiving creation request successful, check status, and get the current policy list again (to verify "HelloWorld" policy instance is present as expected)
- Send via A1 a "HelloWorld" policy instance deletion request
- Upon successful response, check status, and get the current policy list (to verify "HelloWorld" policy instance removal)

Near-RT RIC angle:  [RIC-46](#) - Support A1 health check (granularity/depth of check to be decided) DONE


NONRTRIC angle:  [NONRTRIC-95](#) - NONRTRIC - Supporting Health-Check Use Cases DONE

Figure below shows the corresponding sequence flow diagram.

- Please click on the image for a larger version - or [here](#)
- The PlantUML source is available [here](#)

A1 O-RAN Healthcheck (Bronze)

