Improve ONAP tests
Use-case artifacts placement

Eric Debeau, Sylvain Desbureaux

June 13, 2019

Statement

- What do we really test ?

- Still some use-case artifacts embedded in ONAP components

244 PODs

Current tests

- Unitary tests
  - Code coverage

- Repeated tests
  - 61 HealthChecks
  - CSIT
  - Some use-cases automated (cf OOM Gating)

- Manuals or semi-automatic tests
  - Pair-wise testing
  - Use-case testing

But what do we test ?

- No information about POD involved in tests
  - Healthcheck
  - CSIT
  - Use-case testing
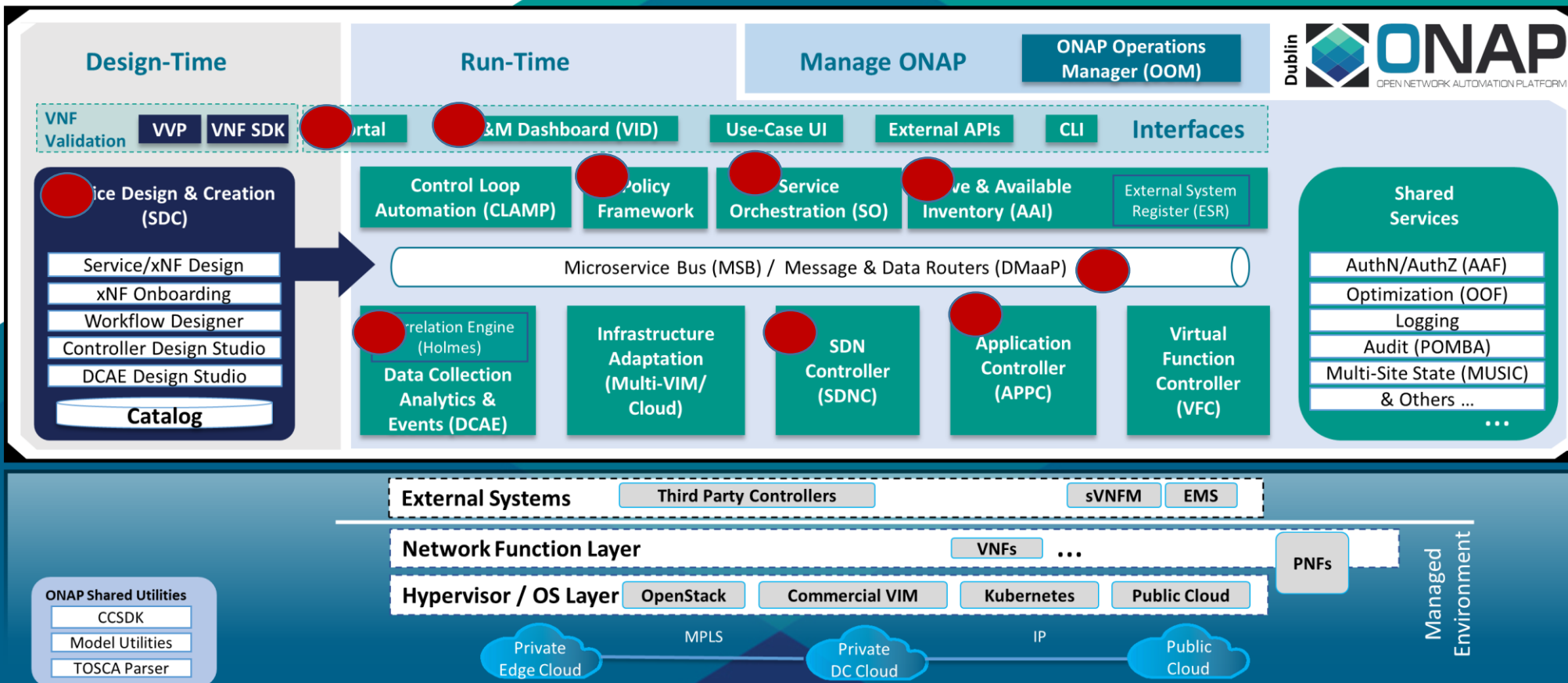
- Some PODs are not involved in CSIT or E2E

vFW example

vFW example: A&AI detail

14 PODs running for A&AI (+ shared Cassandra) which one involved in vFW use-case ?

onap-aai-aai-55ffc8764f-wtr8c

onap-aai-aai-babel-99bb655c5-tdm8l

onap-aai-aai-champ-6ddf64f8d8-jkzvf

onap-aai-aai-data-router-868db64f98-q5ljt

onap-aai-aai-elasticsearch-68f6cdd8bb-c6499

onap-aai-aai-gizmo-7cdcd9c67b-tzwkm

onap-aai-aai-graphadmin-5ddbb7cb5b-sfsgr

onap-aai-aai-modelloader-6695466c49-jwlkd

onap-aai-aai-resources-8466b5d59f-mf5z4

onap-aai-aai-schema-service-736a3c45e4-s32ae

onap-aai-aai-search-data-74c957d877-sdrlr

onap-aai-aai-sparky-be-7d778b7c6c-5jql8

onap-aai-aai-spike-5d74bdf5bc-qbg9t

onap-aai-aai-traversal-5f76756cf-stmrk

Feedbacks from ONAP Components 'users/developers'

- We discover bugs late in the Release

- We discover run-time dependencies when using use-cases

El Alto objective

- Ensure that every POD is tested in a use-case or end-to-end test case

- A POD not involved in a use-case or E2E test case should not be part of the Release

El Alto proposals

- Use-case owner
  - Define the list of projects impacted

- Project Technical Leader
  - Define the set of POD used for each use-case
  - Check that every POD involved in
    - Level0: HealthCheck
    - Level1: CSIT
    - Level2: Use-case, E2E test case

- To be added in Check-list and documented in readthedocs

Statement

- What do we really test ?

- Still some use-case artifacts embedded in ONAP components

ONAP Platform should be deployed with no use-case artifacts

- Many components embed some code specific to use-cases
    - Dedicated policy hardcoded
    - Specific DG
    - Specific entry in some internal database

- Consequences
    - Difficult to focus tests on core features

Migrate all the use-case artifacts in a dedicated repo

- usecases
  - vFW
    - artifacts
      - HEAT
      - Policy
    - doc
    - test
  - BBS
    - artifacts
      - PNF package
      - micro-service
      - policy
    - doc
    - test