ONAP xNF software upgrade status of art and extendibility point

Zu Qiang <Zu.Qiang@Ericsson.com>
Michela Bevilacqua <michela.bevilacqua@ericsson.com>
James Cuddy <james.cuddy@est.tech>
## ONAP xNF Software Upgrade Roadmap

<table>
<thead>
<tr>
<th>Release</th>
<th>Milestone</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rel A/B</td>
<td>✓ Ansible based VNF in-place software upgrade with Ansible adaptor in APPC</td>
<td></td>
</tr>
<tr>
<td>Rel C/D</td>
<td>✓ Partially complete PNF software upgrade procedure by leveraging legacy Ansible LCM API to support PNF software upgrade</td>
<td></td>
</tr>
<tr>
<td></td>
<td>✓ Ansible adaptor for PNF software upgrade in SDNC including Pre-check, Upgrade software, and Post-check</td>
<td></td>
</tr>
<tr>
<td></td>
<td>✓ Additional adaptor including Extend upgrade software operation by leveraging sub-operations from 3GPP</td>
<td></td>
</tr>
<tr>
<td>Rel F</td>
<td>✓ Complete PNF software upgrade procedure with support of three scenarios using the SO decision tree</td>
<td></td>
</tr>
<tr>
<td></td>
<td>✓ Using direct Netconf/Yang interface with PNF: using CDS self-service API, with direct Netconf/Yang interface with the PNF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>✓ Using Ansible protocol with EM: using legacy Ansible LCM API, with EM between ONAP and PNF, and leveraging the development from release C and D,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>✓ Using Netconf/Yang interface with EM: using CDS self-service API, with EM between ONAP and PNF, and Netconf/Yang interface</td>
<td></td>
</tr>
<tr>
<td>Rel G</td>
<td>✓ Partially complete xNF software upgrade procedure with schema update by enabling xNF software upgrade at service level</td>
<td></td>
</tr>
<tr>
<td></td>
<td>✓ A generic service level upgrade workflow in SO</td>
<td></td>
</tr>
<tr>
<td></td>
<td>✓ A generic PNF software upgrade workflow leveraging the development from release F</td>
<td></td>
</tr>
<tr>
<td></td>
<td>✓ New SO APIs for service level workflow retrieving and execution</td>
<td></td>
</tr>
<tr>
<td>Rel H</td>
<td>✓ Additional enhancements on the generic PNF software upgrade workflow</td>
<td></td>
</tr>
<tr>
<td></td>
<td>✓ Adding PNF software version at onboarding and SO catalog</td>
<td></td>
</tr>
<tr>
<td></td>
<td>✓ Support SDC CSAR upgrade with multiple xNFs in a service template</td>
<td></td>
</tr>
<tr>
<td>Rel ?+</td>
<td>✓ Support generic VNF software upgrade workflow using CDS self-service API</td>
<td></td>
</tr>
</tbody>
</table>
A new xNF software version, with/without interface changes, is available.

**xNF Resource Software Upgrade Use Case**

Upgrade the software version of the xNF instance

**Network Service Software Upgrade Use Case**

In addition to sw version upgrade of the xNF instance, an ONAP update is required to support new interface capabilities
PNF Resource Sw upgrade (in-place) scenarios:
- a) Using direct Netconf/Yang interface with PNF
- b) Using Ansible protocol with EM
- c) Using Netconf/Yang interface with EM

VNF Resource Sw upgrade (in-place):
- New VNF in-place software upgrade procedure is planned (CDS based)
- Existing (APPC based) VNF in-place software upgrade procedure to be discontinued

CNF software upgrade, under discussion
ONAP SO Workflows for xNF Software Upgrade

- SO Workflows can be customized (i.e. add/remove Building Blocks)
- Building block actions are xNF independent
- Building block actions execution demanded to ONAP controller blueprints
- Blueprint script per xNF type
- xNF specific protocol, data model, and actions managed by blueprint

PNF HealthCheck
- preCheck

PNF Preparation
- preCheck
- downloadNESw

PNF Software Upgrade
- preCheck
- downloadNESw
- activateNESw
- postCheck

VNF Software Upgrade
- UpgradePreCheck
- UpgradeSoftware
- UpgradePostCheck

planned
O-RAN O1 Requirements and ONAP implementations

PNF Software Management Services

**Requirements in O1**

- **Software Inventory**: retrieves information about the software packages on the PNF.
- **Software Download**: triggers software package downloading onto PNF,
  - to specify the location of software,
  - to verify if a software download is in progress,
  - to deny download of software if request is not valid,
  - to download needed files from specified location,
  - to perform integrity checks on downloaded software,
  - to install the software into the software slot.
- **Software Activation Pre-Check**: confirms that PNF is in a good state to activate the software,
  - to confirm that the software in the passive slot targeted for activation is good,
  - to determine whether the activation of the targeted software requires a reset and/or data migration.
- **Software Activation**: triggers activation of a software package on the PNF,
  - to allow its authorized consumer to activate valid software in a specific softwarePackage,
  - to verify whether a software activation is in progress and deny a concurrent activation of software,
  - to deny in invalid activation request,
  - to activate the softwarePackage,
  - to reset and send a reset reason notification,
  - to perform data migration,
  - to fallback to previously version and factory set.

**PNF sw upgrade procedure in ONAP**

- **Pre-Check**: verify PNF is in good condition before any upgrade actions.
- **DownloadNESw**: Triggers a download of a software package into the PNF, including:
  - package downloading if needed,
  - package integrity check,
  - progress report,
  - etc.
- **ActivateNESw**: Triggers activation of a software package on the PNF, including:
  - activation,
  - data migration,
  - fallback in case of activation failure,
  - etc.
- **Post-Check**: verify PNF is in good condition after upgrade.

ONAP defines only generic upgrade workflow and basic building blocks. Building block implementation (including definition of xNF actions / protocols / interface) is realized by blueprint script, and the blueprint script is onboarded per PNF type.
Update one PNF instance without schema update

**Vendor**

- PNF Package X
  - Descriptor
  - Artifacts

**Operator**

- SDC
- CDS

**Design Time**

- SDC catalog
- Upload blueprint scripts
  - Add repository folder URL and userID / password
- Onboard package & Create resource A
- Associate additional artifacts (e.g. CBA) with the resource A
- Create Service A with resource A

**Run Time**

- VID
- SO
- AAI
- SDNC
- PNF
- SFTP

**Service Instantiation (PnP)**

- Service instance object A
  - service-instance-id: 1
  - model-invariant-id: 1
  - model-version-id: 1

**Selected workflow, Target software version, and PNF instance ID**

- SW version is updated

**Software Image**

- software Image

**Pre-condition**

- Software version is updated

**Post-condition**

- Software version is updated
Service level software upgrade example with one PNF resource

**Design Time**

- **Onboard package & Create resource A**
  - Upload blueprint scripts and parameters
  - Resource Instance name M
    - Resource Name A invariant:UUID 1
    - UUID 1
    - swVersion 1.0
    - Artifacts

- **Associate additional artifacts (e.g. CBA) with the resource A**
  - Add additional service artifacts (e.g. wf)

- **Create Service A with one resource instance**
  - Additional service artifacts (e.g. wf)

**Run Time**

- **Upload/update blueprint scripts and parameters**
  - Resource Instance name M
    - Resource Name A invariant:UUID 1
    - UUID 3
    - swVersion 2.0
    - Artifacts

- **Update Service template by replacing the existing resource A with a new revision**
  - Update additional service artifacts (e.g. wf)

- **Update additional artifacts (e.g. CBA) with the resource A**

- **Update Service template with the new revision of the same resource template**

- **Select an existing service instance, a target service template revision and a service level upgrade workflow**

**Step 1:** Provide new image and new package to the operator

**Step 2:** Update the resource template using the new onboarding package

**Step 3:** Update service template with the new revision of the same resource template

**Step 4:** Select an existing service instance, a target service template revision and a service level upgrade workflow

**Step 5:** Execute the workflow on the service instance

**Step 6:** Update the management service instance object
SO Service level upgrade workflow

- **Service Level Preparation**
  - Creating resource template instance upgrade list by comparing the service templates
  - Select a resource level health check workflow based on the resource type
  - Execute the selected resource level health check workflow on all resource instances within the service

- **Service Level Upgrade**
  - Select a resource level upgrade workflow based on the resource type
  - Execute the selected resource level upgrade workflow on each upgrading resource instances
  - Update the software version, model-invariant-id and model-version-id of the resource template in the A&AI entry at end of each Resource level upgrade workflow

- **Service Level Update**
  - Update the model-version-id of the service template in the A&AI entry

- **Service Level postCheck**
  - Select a resource level health check workflow based on the resource type
  - Execute the selected resource level health check workflow on all resource instances
SO Service level upgrade workflow

- Service Level Preparation
- Service Level Upgrade
- Service Level Update
- Service Level postCheck

Example of resource level upgrade workflow:

- **PNF software upgrade**
  - precheck
  - downloadNESw
  - activateNESw
  - postCheck

- **VNF software upgrade**
  - UpgradePreCheck
  - UpgradeSoftware
  - UpgradePostCheck

Note: the service level workflow is network function type independent. When upgrade one resource instance, the subsequent resource level upgrade workflow is selected based on the network function type.

Repeat on each to-be-upgraded resource instance.
Demo: Service level software upgrade example with one PNF resource

Query PNF status from AAI and PNF node before software upgrade

Demo of Service template Upgrade with a new PNF onboarding package

Onboard the new PNF package → Update the PNF resource template → Update the service template → distribution

Demo of Service level Software Upgrade workflow with one PNF instance

Service Level Preparation → Service Level Upgrade: run PNF software upgrade workflow to upgrade the PNF instance to software version 3.0.0 → Service Level Update: update A&AI → Service Level postCheck

Query PNF status from AAI and PNF node after software upgrade

Operational level

Vendor

Operator

SDC

CDS

SDNC (blueprint processor)

AAI

SO

VID

SFTP

PNF

THE LINUX FOUNDATION

ONAP
Thank you