PNF software upgrade for Frankfurt release

Zu Qiang (Ericsson)
Current Development Status

✓ VNF in place software upgrade is supported
  ✓ using Ansible and Chef
  ✓ with LCM API
  ✓ ‘generic’ SO building blocks

✓ PNF in place software upgrade is supported in Casablanca and updated in Dublin
  ➢ With the support of an EM
  ➢ Ansible protocol only
  ➢ Plan to use LCM API with existing SO building blocks
  ➢ Impacts on SDNC only (not E2E solution yet)
PNF software upgrade use cases

• Use case 1:
  - A solution is required to support PNF software update without impacts on PNF schema and service template
  - This solution can be used for PNF software correction update

• Use case 2:
  - A solution is required to support PNF software upgrade and PNF schema update (i.e. new onboarded PNF artifacts), considering service level impacts
PNF software upgrade is one aspect of Software Management. The purpose of this procedure is to upgrade the software currently running on the PNF to a target software version without impacts on PNF schema and service template.

Details
- Enhancement and additions of PNF in-place software update.
- Support direct PNF NETCONF interface with the vendor-specific YANG model.
- Enhance SO in-place software upgrade workflow with generic SO building blocks, which can be used for workflow design in the design time.
- Using CDS self-service API between SO and controller with the support of PNF in-place software upgrade
- Enhance VID to demonstrate single PNF in-place software upgrade
- Enhance SO procedure to support AAI update after the software upgrade completion.
✓ ONAP is ready to use
✓ SO upgrade workflows are ready to use
✓ A SDC service template with one PNF resource is designed (including CBA association) and it is distributed to run time
✓ Service instantiation is completed, including PNF PnP.
  ➢ A PNF instance is in operation with connectivity between PNF-ONAP, PNF-SFTP
1. Vendor delivers the new software image to the operator and stored in the SFTP server

2. At the VID, operator
   - selects a work-flow, and a PNF instance,
   - provides the target software version, and
   - initiates the upgrade procedure

3. SO executes the workflow
   a) SO sends CDS request(s) with action-identifier \{\text{actionName}, \text{blueprintName}, \text{blueprintVersion}\} to the blueprint processor inside the controller using CDS self-service API
   b) Controller/blueprint processor executes the blueprint scripts including sending Netconf request(s) to the PNF instance
   c) SO updates the A&AI with the active software-version when the upgrade is completed
Download the software only if the software image is not available in the PNF yet.

Split the upgrade task into two subtasks, download and activation with 3GPP TS 32.532 operation names.

Activate the downloaded software in the PNF.

Triggering rollback by controller or node, only when software activation has failure.

The expected result is
- If activation is not ok, the PNF is running at old version
- If activation is ok, the PNF is running at new version.

Possibility to download the software image before upgrade.
Operator can do
- Preparation first, and do Software Upgrade later, or
- Software Upgrade only

Software Upgrade Workflow

Preparation Workflow
Upgrade one PNF instance (Frankfurt)

**Pre-condition**
- Vendor
- Operator

**Design Time**
- SDC catalog
  - Onboard PNF resource A
  - Upload blueprint scripts
    - Add repository folder URL and userID / password
  - Resource A
    - invariantUUID
    - UUID
    - Revision
  - Service template A
    - Name
    - invariantUUID
    - UUID
    - Revision

**Run Time**
- VID
- SO
- AAI
- SDNC

**PNF**
- SFTP

**Service Instantiation (PnP)**
- PNF instance ID
  - model-invariant-id
  - model-version-id
  - software-version

**Selected workflow**
- Target software version
- PNF instance ID
- actionIdentifier
- blueprintName
- blueprintVersion

**Updates**
- Software version is updated

**Retrieve image**
API impacts in Frankfurt

Retrieve all PNF instance: `GET /network/pnfs`

Filter PNF instance list

User selects a PNF instance

Retrieve workflows: `GET /workflowSpecifications/v1/workflows?vnfModelVersionId={UUID}`

List of all workflows

User selects a workflow

User inputs the target software version

Execute the workflow with service instance UUID / PNF UUID / workflow UUID / target software version:

`POST /instanceManagement/v1/serviceInstances/{serviceInstanceId}/pnfs/{pnfInstanceId}/workflows/{workflow_UUID}?targetSoftwareVersion="2.0"`

Retrieve workflows:

`GET /workflowSpecifications/v1/workflows?vnfModelVersionId={UUID}`

List of all workflows

Update A&AI with active software version:

`PATCH /network/pnfs/pnf/{pnf-name}/software-versions/software-version/{software-version-id}`

Completion notification

Enhancement

SO Activity: SO activities request using self-service API

SO activities request using self-service API

Netconf / Yang

Retrieve image

Netconf / Yang

Execution

Query

ack
## ONAP Impacts in Frankfurt

<table>
<thead>
<tr>
<th>Story</th>
<th>components</th>
</tr>
</thead>
<tbody>
<tr>
<td>Support generic workflow design</td>
<td>SDC/SO</td>
</tr>
<tr>
<td>Support creating of SO upgrade workflows for PNF upgrade, including Upgrade and Preparation</td>
<td>SDC/SO</td>
</tr>
<tr>
<td>Create or modify SO activity building block for PNF upgrade, including downloadNESw, activateNESw, updateAAI, preCheck, postCheck</td>
<td>SDC/SO</td>
</tr>
<tr>
<td>Support PNF upgrade UI</td>
<td>VID</td>
</tr>
<tr>
<td>Update VID-SO API to execute the workflow of PNF software upgrade with target software version: POST /instanceManagement/v1/serviceInstances/{serviceInstanceId}/pnfs/{pnfInstanceId}/workflows/{workflow_UUID}?targetSoftwareversion=&quot;2.0&quot;</td>
<td>VID/SO</td>
</tr>
<tr>
<td>PNF and CBA association enhancement to support PNF upgrade</td>
<td>SDC/CDS</td>
</tr>
<tr>
<td>Support PNF upgrade with CDS self-service API</td>
<td>SO/CCSDK</td>
</tr>
<tr>
<td>Implement updateAAI activity for A&amp;AI updates with active software-version</td>
<td>SO</td>
</tr>
<tr>
<td>Documentation</td>
<td>VNFRQTS</td>
</tr>
<tr>
<td>integration / testing and demo</td>
<td>Integration</td>
</tr>
</tbody>
</table>
Next steps to support UC2

- Enable service level LCM operations
  - PNF software upgrade with updated service template
  - PNF schema update with updated service template
  - Service template update with multiple resource instances

- Including:
  - Updating the design time service template using vendor provided onboarding package
  - Upgrading a run time service instance based on the updated service template
  - Updating the run time catalog at software upgrade completion
Key issues to support next steps

- **Modelling/SDC/AAI:**
  - Onboarding the software version information using vendor provided onboarding package
  - Supporting software version in internal model

- **SDC/SO/VID:**
  - Resource upgrade path
  - Service upgrade options

- **SO:** Executing the workflow at service level

- **AAI:**
  - Support updating resource model ID/version
  - Support updating service model ID/version
Service level operation example for next steps

**Pre-condition**

**Vendor**

**Operator**

- SDC
- CDS

**SDC catalog**

**SDC**

**CDS**

**Onboard PNF Create resource A**

**Upload blueprint scripts Add repository URL and userID / password**

**Associate CBA and the PNF resource A**

**Create Service A with PNF resource A**

**Resource A**

- invariantUUID 1
- UUID 1
- swVersion 1.0
- Revision 1.0

**Service template A**

- Name A
- invariantUUID 1
- UUID 1
- Revision 1.0

**Distribution**

**Service Instantiation (PNP)**

**SDC catalog**

**VID**

**SO**

**AAI**

**SDNC**

**PNF**

**SFTP**

**Onboard PNF Create resource B**

**Upload blueprint scripts Add repository URL and userID / password**

**Associate CBA and the PNF resource B**

- _sdnc_model_name_
- _sdnc_model_version_

**Update Service A by including PNF resource B and excluding PNF resource A**

**Resource B**

- invariantUUID 2
- UUID 2
- swVersion 2.0
- Revision 2.0

**Service template A**

- Name A
- invariantUUID 1
- UUID 1
- Revision 1.0

**Distribution**

**Service Level LCM operation**

**SO workflow execution per resource instance**

- Instance ID & actionIdentifer
- Netconf / Yang

**Updates**

**Retrieve image**

**Service instance object A**

- service-instance-id 1
- service-instance-id 1
- model-invariant-id 1
- model-version-id 1

**Service instance object A**

- service-instance-id 1
- service-instance-id 1
- model-invariant-id 1
- model-version-id 1

**Design Time**

**Run Time**

**Service level operation example for next steps**

**Vendor**

**Operator**

- SDC
- CDS

**SDC catalog**

**SDC**

**CDS**

**Onboard PNF Create resource A**

**Upload blueprint scripts Add repository URL and userID / password**

**Associate CBA and the PNF resource A**

**Create Service A with PNF resource A**

**Resource A**

- invariantUUID 1
- UUID 1
- swVersion 1.0
- Revision 1.0

**Service template A**

- Name A
- invariantUUID 1
- UUID 1
- Revision 1.0

**Distribution**

**Service Instantiation (PNP)**

**SDC catalog**

**VID**

**SO**

**AAI**

**SDNC**

**PNF**

**SFTP**

**Onboard PNF Create resource B**

**Upload blueprint scripts Add repository URL and userID / password**

**Associate CBA and the PNF resource B**

- _sdnc_model_name_
- _sdnc_model_version_

**Update Service A by including PNF resource B and excluding PNF resource A**

**Resource B**

- invariantUUID 2
- UUID 2
- swVersion 2.0
- Revision 2.0

**Service template A**

- Name A
- invariantUUID 1
- UUID 1
- Revision 1.0

**Distribution**

**Service Level LCM operation**

**SO workflow execution per resource instance**

- Instance ID & actionIdentifer
- Netconf / Yang

**Updates**

**Retrieve image**

**Service instance object A**

- service-instance-id 1
- service-instance-id 1
- model-invariant-id 1
- model-version-id 1

**Service instance object A**

- service-instance-id 1
- service-instance-id 1
- model-invariant-id 1
- model-version-id 1

**Design Time**

**Run Time**

**Service level operation example for next steps**

**Vendor**

**Operator**

- SDC
- CDS

**SDC catalog**

**SDC**

**CDS**

**Onboard PNF Create resource A**

**Upload blueprint scripts Add repository URL and userID / password**

**Associate CBA and the PNF resource A**

**Create Service A with PNF resource A**

**Resource A**

- invariantUUID 1
- UUID 1
- swVersion 1.0
- Revision 1.0

**Service template A**

- Name A
- invariantUUID 1
- UUID 1
- Revision 1.0

**Distribution**

**Service Instantiation (PNP)**

**SDC catalog**

**VID**

**SO**

**AAI**

**SDNC**

**PNF**

**SFTP**

**Onboard PNF Create resource B**

**Upload blueprint scripts Add repository URL and userID / password**

**Associate CBA and the PNF resource B**

- _sdnc_model_name_
- _sdnc_model_version_

**Update Service A by including PNF resource B and excluding PNF resource A**

**Resource B**

- invariantUUID 2
- UUID 2
- swVersion 2.0
- Revision 2.0

**Service template A**

- Name A
- invariantUUID 1
- UUID 1
- Revision 1.0

**Distribution**

**Service Level LCM operation**

**SO workflow execution per resource instance**

- Instance ID & actionIdentifer
- Netconf / Yang

**Updates**

**Retrieve image**

**Service instance object A**

- service-instance-id 1
- service-instance-id 1
- model-invariant-id 1
- model-version-id 1

**Service instance object A**

- service-instance-id 1
- service-instance-id 1
- model-invariant-id 1
- model-version-id 1

**Design Time**

**Run Time**

**Service level operation example for next steps**

**Vendor**

**Operator**

- SDC
- CDS

**SDC catalog**

**SDC**

**CDS**

**Onboard PNF Create resource A**

**Upload blueprint scripts Add repository URL and userID / password**

**Associate CBA and the PNF resource A**

**Create Service A with PNF resource A**

**Resource A**

- invariantUUID 1
- UUID 1
- swVersion 1.0
- Revision 1.0

**Service template A**

- Name A
- invariantUUID 1
- UUID 1
- Revision 1.0

**Distribution**

**Service Instantiation (PNP)**

**SDC catalog**

**VID**

**SO**

**AAI**

**SDNC**

**PNF**

**SFTP**

**Onboard PNF Create resource B**

**Upload blueprint scripts Add repository URL and userID / password**

**Associate CBA and the PNF resource B**

- _sdnc_model_name_
- _sdnc_model_version_

**Update Service A by including PNF resource B and excluding PNF resource A**

**Resource B**

- invariantUUID 2
- UUID 2
- swVersion 2.0
- Revision 2.0

**Service template A**

- Name A
- invariantUUID 1
- UUID 1
- Revision 1.0

**Distribution**

**Service Level LCM operation**

**SO workflow execution per resource instance**

- Instance ID & actionIdentifer
- Netconf / Yang

**Updates**

**Retrieve image**

**Service instance object A**

- service-instance-id 1
- service-instance-id 1
- model-invariant-id 1
- model-version-id 1

**Service instance object A**

- service-instance-id 1
- service-instance-id 1
- model-invariant-id 1
- model-version-id 1

**Design Time**

**Run Time**

**Service level operation example for next steps**

**Vendor**

**Operator**

- SDC
- CDS

**SDC catalog**

**SDC**

**CDS**

**Onboard PNF Create resource A**

**Upload blueprint scripts Add repository URL and userID / password**

**Associate CBA and the PNF resource A**

**Create Service A with PNF resource A**

**Resource A**

- invariantUUID 1
- UUID 1
- swVersion 1.0
- Revision 1.0

**Service template A**

- Name A
- invariantUUID 1
- UUID 1
- Revision 1.0

**Distribution**

**Service Instantiation (PNP)**

**SDC catalog**

**VID**

**SO**

**AAI**

**SDNC**

**PNF**

**SFTP**

**Onboard PNF Create resource B**

**Upload blueprint scripts Add repository URL and userID / password**

**Associate CBA and the PNF resource B**

- _sdnc_model_name_
- _sdnc_model_version_

**Update Service A by including PNF resource B and excluding PNF resource A**

**Resource B**

- invariantUUID 2
- UUID 2
- swVersion 2.0
- Revision 2.0

**Service template A**

- Name A
- invariantUUID 1
- UUID 1
- Revision 1.0

**Distribution**

**Service Level LCM operation**

**SO workflow execution per resource instance**

- Instance ID & actionIdentifer
- Netconf / Yang

**Updates**

**Retrieve image**

**Service instance object A**

- service-instance-id 1
- service-instance-id 1
- model-invariant-id 1
- model-version-id 1

**Service instance object A**

- service-instance-id 1
- service-instance-id 1
- model-invariant-id 1
- model-version-id 1