Edge Scoping MVP for Casablanca - ONAP Enhancements

Summary: Edge Architecture & Work Items#ONAPEdgeMVP

Distributed Edge Cloud Infrastructure Object Hierarchy (Stretch Goal - Beyond Casablanca)

Value:

- **Fine grained resource management & analytics for Distributed Edge Clouds**

References:

- Infrastructure Modelling: ONAP R3+ Cloud Infrastructure Modeling; Cloud Infrastructure Aggregate Representation Classes

MULTICLOUD-153 - Getting issue details... STATUS

ONAP Component	Life Cycle Phase	Enhancements
Multi-Cloud	Deploy	Support Distributed Cloud Infrastructure Capability Discovery (Note 1, Note 2)
A&AI	Deploy	Support Standardized Distributed Cloud Infrastructure Object Hierarchy & Capability Database (Ref. 1) • Loose coupling between HW objects (private cloud) and SW objects (private and public clouds) • Includes Standardized Capabilities across clouds & Capabilities unique to certain clouds • Note: ◦ Multi-Cloud Distributed Cloud Infrastructure Capability Discovery process will populate the aforementioned database

OOF	Deploy	Execute Distributed Cloud Infrastructure Placement Policies for Optimized Service/VNF Placement across Cloud Regions (Note 3, Note 4)
SO	Deploy	Extend SO OOF API to support data opaque to SO (Note 5) Extend SO MC API to support data opaque to SO (Note 6)

Assumption for Policy, SO, OOF:

- This uses the current Generic VNF workflow in SO

Note 1:

- Configured Capacity and Utilized (or Currently Used) Capacity are managed by the specific cloud.

Note 2:

- Cloud SW Capability example
  - Cloud region "x" with SR-IOV, GPU, Min-guarantee support
  - Cloud region "y" with SR-IOV support
- Cloud HW Capability example
  - Resource cluster "xa" in Cloud region "x" with SR-IOV and GPU support
  - Resource cluster "xb" in Cloud region "x" with GPU support
  - Resource cluster "ya" in Cloud region "y" with SR-IOV support

Note 3:

- 5G Service/VNF placement example
  - Constraints used by Optimization Framework (OOF)
    - 5G CU-UP VNF location to be fixed to a specific physical DC based on 5G DU, bounded by a max distance from 5G DU
  - Optimization Policy used by OOF
    - Choose optimized cloud region (or instance) for the placement of 5G CU UP for subscriber group based on the above constraints

Note 4:

- For the 5G Service/VNF placement example in Note 3
  - 5G CU-UP VNF preferably maps to a specific Cloud region & Physical DC End Point

Note 5:

- For the 5G Service/VNF placement example in Note 3
  - OOF will pass the Physical DC End Point to SO as a opaque data

Note 6:

- For the 5G Service/VNF placement example in Note 3
  - SO passes the Physical DC End Point to Multi-Cloud as a opaque data, besides the Cloud Region

Cloud-agnostic Placement/Networking & Homing Policies (Phase 1 - Casablanca MVP, Phase 2 - Stretch Goal)

End-to-end use case Applicability:

- All (especially the data plane VNFs with fine-grained VNF placement and high performance networking requirements)

Value:

- Improve "workload deployability" by avoiding exposure of "cloud specific" capabilities to several ONAP components and addressing "separation of concerns"
- Applicable to all workloads - VM-based or Container-based

MULTICLOUD-272 - Getting issue details... STATUS

Phase 1 (Casablanca MVP) Summary:

- Multi-Cloud Policy Framework
  - Assist OOF in target cloud region selection for VNF placement (aka homing) through intent
    - Cloud Agnostic Intent (Policy) Execution Workflow - Steps 1- 4

    - Dynamically modify the cloud specific VNF deployment template based on cloud-specific realization of the specified intent (e.g. Infra HA for VMs within a VNF could have different realizations across different clouds)
        - Cloud Agnostic Intent (Policy) Execution Workflow - Step 5
- Intent Support
    - Single realization option per Cloud Region for the specified Intent
- Impact Projects:
    - Multi-Cloud (Highest), OOF, SO (Minimal)
- End-to-end use case demonstration:
    - vCPE (no additional implementation dependency), vDNS
- Types of intent supported (through OOF Policy)
    - "Infrastructure High Availability for VNF"
    - "Infrastructure Resource Isolation for VNF": "Burstable QoS"
    - "Infrastructure Resource Isolation for VNF": "Guaranteed QoS"
- Related Specs/Jiras:
    - OOF
        - OOF Policy spec -- https://wiki.onap.org/display/DW/OOF+Cloud+Agnostic+Policies; Policy Specification and Retrieval for OOF
    - SO
        - SO Casablanca HPA design spec with cloud agnostic intent -- https://wiki.onap.org/display/DW/SO+Casablanca+HPA+Design
    - Multi-Cloud
        - Generic API for SO to talk to different Multi cloud plugins to be updated with cloud agnostic intent -- https://gerrit.onap.org/r/#/c/60691/
        - HPA Cloud specific (flavor etc.) Mapping for R3 – HPA Policies and Mappings
        - Intent Cloud specific (flavor etc.) Mapping for R3 – Cloud Agnostic Intent and Mappings
    - End-to-end use case
        - Support homing through OOF for vFW, vDNS – **SO-745** - Getting issue details... STATUS
- Useful Links:
    - R2 HPA Integration testing – vCPE Use Case + OOF + HPA Tutorial: Design and Deploy based on ONAP#PrepareHEATtemplates

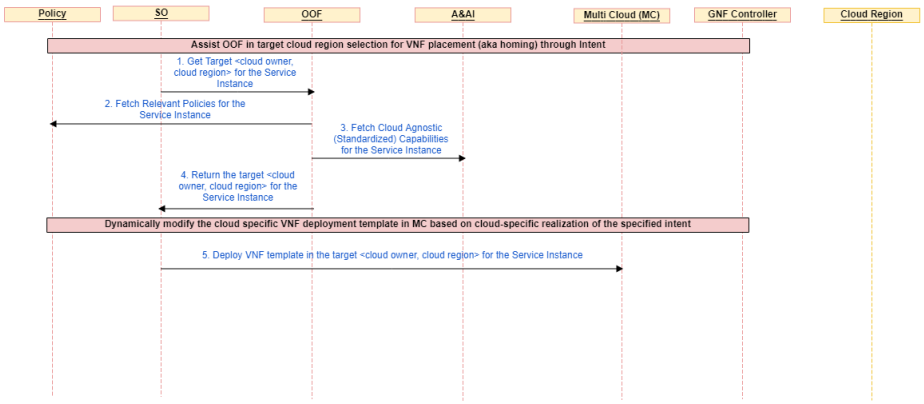
Phase 2 (Casablanca Stretch Goal) Summary (Build on Phase 1 Work):

- Multi-Cloud Policy Framework
    - Dynamically modify the cloud specific VNF deployment template based on cloud-specific realization of the specified intent – Impact to VNF configuration
        - E.g. High performance Intra-DC data plane networking with several realization choices
- Intent Support
    - Multiple realization options per Cloud Region for the specified Intent
- Major Impact Projects:
    - Multi-Cloud
- Minor Impact Projects:
    - SO, OOF, GNF Controller
- Wiki Link:
    - Edge Scoping - Casablanca Stretch Goal/Beyond Casablanca

References:

⚠️ The sequence diagram below expands "Multi-Cloud/VNFM Deploy Apps" in Edge Architecture & Work Items Sequence Diagram

Cloud Agnostic Intent (Policy) Workflow Summary (Phase 1 - Casablanca MVP):



Cloud Agnostic Intent (Policy) Workflow Details (Phase 1 - Casablanca MVP):

Private Cloud Setup - OpenStack-based

- Pre-defined (including custom) **flavors** map to **Instance types** in Public Clouds
  - Pre-defined flavors are created by the Cloud Admin before the Cloud is used by ONAP for workload deployment
- VMware VIO Configuration for Min Guarantee feature
  - Ref: https://docs.vmware.com/en/VMware-Integrated-OpenStack/5.0/com.vmware.openstack.admin.doc/GUID-D2C17D33-EC6C-4B8C-A9A3-25417917461F.html
  - Ref: https://docs.openstack.org/horizon/latest/admin/manage-flavors.html
- Create necessary tenants per <cloud owner, cloud region>
  - Mapping of VNFC, VNF (VF module), Service (e.g. vCPE) to the corresponding tenant happens in the respective Multi-Cloud plugin.

VNFC to Instance Type Mapping

- One or more VNFCs (e.g. vCPE VGW) could map to an **Instance Type**
  - Use Case: Residential Broadband vCPE (Approved)
    - Ref: https://github.com/onap/demo/blob/a19bd423c1e4d2f5552bab090ba2650c38fa7192/heat/vCPE/vgw/base_vcpe_vgw.yaml
    - Ref: https://github.com/onap/demo/blob/ab98f6a0db3df8f5232395d899e96462dbc11159/heat/vCPE/vgw/base_vcpe_vgw.env
  - OpenStack-based Clouds
    - Instance type maps to pre-defined Flavors
  - Microsoft Azure
    - Pre-defined Instance Types
      - Ref: https://docs.microsoft.com/en-us/azure/virtual-machines/linux/sizes?toc=%2fazure%2fvirtual-machines%2flinux%2ftoc.json

Step 1. SO OOF - Get Target <Cloud Owner, Cloud Region> for the Service Instances (no code changes for R3)

Step 2. OOF Policy - Fetch Cloud Selection Policy for Homing

2a) OOF Processing - the fetched Policy (example below) is stored in a local data structure and is available for further use **(need OOF code changes for R3)**

`OOF Homing Enhanced Cloud Selection Policy based on Intent -- Schema with Use Case Examples as runnable python code:`

OOF Homing Enhanced Cloud Selection Policy Example (Step 2a)

```
#
#Spec Reference: https://wiki.onap.org/display/DW/Edge+Scoping+MVP+for+Casablanca+-
+ONAP+Enhancements#EdgeScopingMVPforCasablanca-ONAPEnhancements-Cloud-agnosticPlacement
/Networking&HomingPolicies(Phase1-CasablancaMVP,Phase2-StretchGoal)
#

from jsonschema import validate

oof_cloud_selection_policy_schema = {
        "service": {"type": "string"},
        "policyName": {"type": "string"},
        "policyDescription": {"type": "string"},
        "templateVersion": {"type": "string"},
        "version": {"type": "string"},
        "priority": {"type": "string"},
        "riskType": {"type": "string"},
        "riskLevel": {"type": "string"},
        "guard": {"type": "string"},

        "content": {
                "type": "object",
                "required": ["cloud-deployment-intent"],
                "properties" : {

                        # VNFC is not used in the OOF->MC path for R3
                        # This is kept to be consistent with the SO-> MC path
                                    # As an example, vDNS VNF in ONAP has 3 VNFCs - DNS, Packet Gen & Load
Balancer --
                                                # Each of the VNFCs could have different
policies

                        "vnfc": {"type": "string"},

                        # cloud-specific realization of the specified deployment intent
                        # happens in  multi-cloud in the cloud-specific plugin
                        "cloud-deployment-intent": {
                                "type": "object",
                                "properties" : {
```

```
                                        # Cloud Type -- Azure, K8S, OpenStack, VMware VIO, Wind River Titanium
                                                        # Optionally Accomodate policies per Cloud Type
                                                        "Cloud Type (Cloud Provider)": {"type", "array"},

                                        "Infrastructure High Availability for VNF": {"type", "boolean"},

                                        "Infrastructure Resource Isolation for VNF": {"type", "string"},

                                        # Infrastructure Resource Isolation for VNF
                                        # Only certain pre-defined over-subscription values are allowed to
                                        # reflect practical deployment and simplify implementation for R3
                                        "Infrastructure Resource Isolation for VNF - Burstable QoS
Oversubscription Percentage": {"type": "int"},
                                    },
                            },
                },
        },

        "resources": {"type", "array"}, #"vgw" is also interchangeably used as "vg"
        "applicableResources": {"type", "string"},
        "identity": {"type", "string"},
        "policyScope": {"type", "array"},
        "policyType": {"type", "string"}
}

#
#Example 1: vCPE, Burstable QoS
#vCPE: Infrastructure Resource Isolation for VNF with Burstable QoS
#
oof_cloud_selection_policy_instance1 = {
        "service": "cloudSelectionPolicy",
        "policyName": "oofMulti-cloudCasablanca.cloudSelectionPolicy_vCPE_VNF",
        "policyDescription": "Cloud Selection Policy for vCPE VNFs",
        "templateVersion": "0.0.1",
        "version": "oofMulti-cloudCasablanca",
        "priority": "3",
        "riskType": "test",
        "riskLevel": "2",
        "guard": "False",

        "content": {
                "vnfc": "vgw",
                "cloud-deployment-intent": {
                        "Cloud Type (Cloud Provider)": {"VMware VIO"},
                                    "Infrastructure Resource Isolation for VNF": "Burstable QoS",
                                    "Infrastructure Resource Isolation for VNF - Burstable QoS Oversubscription
Percentage": 25,
                },
        },

        "resources": ["vgw"], #"vgw" is also interchangeably used as "vg"
        "applicableResources": "any",
        "identity": "cloud-atrributes",
        "policyScope": ["vCPE", "US", "INTERNATIONAL", "ip", "vgw", "vgmux"],
        "policyType": "AllPolicy"
}

#
#Example 2:
#vCPE: Infrastructure Resource Isolation for VNF with Guaranteed QoS
#
oof_cloud_selection_policy_instance2 = {
        "service": "cloudSelectionPolicy",
        "policyName": "oofMulti-cloudCasablanca.cloudSelectionPolicy_vCPE_VNF",
        "policyDescription": "Cloud Selection Policy for vCPE VNFs",
        "templateVersion": "0.0.1",
        "version": "oofMulti-cloudCasablanca",
        "priority": "3",
        "riskType": "test",
        "riskLevel": "2",
        "guard": "False",
```

```
        "content": {
                "vnfc": "vgw",
                "cloud-deployment-intent": {
                                "Infrastructure Resource Isolation for VNF": "Guaranteed QoS",
                },
        },

        "resources": ["vgw"], #"vgw" is also interchangeably used as "vg"
        "applicableResources": "any",
        "identity": "cloud-atrributes",
        "policyScope": ["vCPE", "US", "INTERNATIONAL", "ip", "vgw", "vgmux"],
        "policyType": "AllPolicy"
}

#
#Example 3:
#vDNS: Infrastructure HA for VNF & Infrastructure Resource Isolation for VNF with Burstable QoS
#
oof_cloud_selection_policy_instance3 = {
        "service": "cloudSelectionPolicy",
        "policyName": "oofMulti-cloudCasablanca.cloudSelectionPolicy_vDNS_VNF",
        "policyDescription": "Cloud Selection Policy for vDNS VNFs",
        "templateVersion": "0.0.1",
        "version": "oofMulti-cloudCasablanca",
        "priority": "3",
        "riskType": "test",
        "riskLevel": "2",
        "guard": "False",

        "content": {
                "vnfc": "vdns",
                "cloud-deployment-intent": {
                        "Cloud Type (Cloud Provider)": {"VMware VIO", "Azure"},
                        "Infrastructure High Availability for VNF": True,
                        "Infrastructure Resource Isolation for VNF": "Burstable QoS",
                        "Infrastructure Resource Isolation for VNF - Burstable QoS Oversubscription
Percentage": 25,
                },
        },

        "resources": ["vDNS"],
        "applicableResources": "any",
        "identity": "cloud-atrributes",
        "policyScope": ["vDNS", "US", "INTERNATIONAL", "vDNS"],
        "policyType": "AllPolicy"
}

#
# Example 4:
# vDNS: Infrastructure HA for VNF & Infrastructure Resource Isolation for VNF
# with Guaranteed QoS
#
oof_cloud_selection_policy_instance4 = {
        "service": "cloudSelectionPolicy",
        "policyName": "oofMulti-cloudCasablanca.cloudSelectionPolicy_vDNS_VNF",
        "policyDescription": "Cloud Selection Policy for vDNS VNFs",
        "templateVersion": "0.0.1",
        "version": "oofMulti-cloudCasablanca",
        "priority": "3",
        "riskType": "test",
        "riskLevel": "2",
        "guard": "False",

        "content": {
                "vnfc": "vdns",
                "cloud-deployment-intent": {
                        "Infrastructure High Availability for VNF": True,
                        "Infrastructure Resource Isolation for VNF": "Guaranteed QoS",
                },
        },
```

```
        "resources": ["vDNS"],
        "applicableResources": "any",
        "identity": "cloud-atrributes",
        "policyScope": ["vDNS", "US", "INTERNATIONAL", "vDNS"],
        "policyType": "AllPolicy"
}

validate(oof_cloud_selection_policy_instance1, oof_cloud_selection_policy_schema)
validate(oof_cloud_selection_policy_instance2, oof_cloud_selection_policy_schema)
validate(oof_cloud_selection_policy_instance3, oof_cloud_selection_policy_schema)
validate(oof_cloud_selection_policy_instance4, oof_cloud_selection_policy_schema)
```

Step 3. OOF A&AI - Fetch Cloud-Agnostic (Standardized) Capabilities for the Service Instance (no code changes for R3)

3a) OOF Processing - Perform Cloud Agnostic Capability check for each <cloud owner, cloud region>. OOF will prune any <cloud owner, cloud region> which is not satisfying the standardized capabilities.

Step 4. OOF SO - Return the target <cloud owner, cloud region> for the Service Instance + deployment-intent per vnfc (code changes in OOF for R3)

OOF SO API extension - aligned to the OOF/SO API defined by SO Casablanca HPA Design to minimize the terminology set. The data between OOF to SO and SO to MC is identical -- details of the API are in section 5.

Step 5. SO MC - Deploy VNF template in the target <cloud owner, cloud region> for the Service Instance (code changes in Multi-Cloud for R3)

5) MC Processing (need MC code changes)

- **Parse Template** (e.g. OpenStack Heat Template)
  - For each VNFC, instance type in the template
    - Parse Policy JSON coming in the SO MC directives API
    - Modify template (if needed) according to Intent
      - Intent examples of interest for R3
        - "Infrastructure High Availability (HA) for VNF"
        - "Infrastructure Resource Isolation for VNF"
          - "Burstable QoS"
        - "Infrastructure Resource Isolation for VNF"
          - "Guaranteed QoS"
- **Policy (Intent) Realization**
  - Determining the flavor (OpenStack-based VIMs) # same logic applies for instance type in Azure
    - Each VNFC uniquely maps to a Flavor - for e.g. VNFC "vgw" maps to "vgw-base", "vDNS" maps to "vDNS-base"
    - Beyond Casablanca
      - VNFC intent to realization mapping happens through A&AI.
  - "Infrastructure High Availability (HA) for VNF"
    - OpenStack-based Cloud realization
      - For R3, Host-based anti-affinity using server groups //Beyond R3, Support other anti-affinity models at availability zone level etc.
      - *Implementation Notes:*
        - *Instance "count" in heat template specifies VNFC scale out factor*
        - *While dynamic injection of server group into heat template is ideal, a simple starting point could be just switching to an alternate heat template which is identical to the deployment template and additionally has server group*
    - Azure realization
      - Availability Set?
  - "Infrastructure Resource Isolation for VNF" – { "qosProperty": { {"Burstable QoS": "TRUE", "Burstable QoS Oversubscription Percentage": "25"} } }
    - OpenStack-based VMware VIO Cloud realization
      - This can be achieved through min guarantee -- Max or limit (upper bound) & Min or Reservation (guarantee) are part of OpenStack flavor metadata
        - https://docs.openstack.org/horizon/latest/admin/manage-flavors.html
        - https://blogs.vmware.com/openstack/resources-guarantee-openstack-and-vmware-integrated-openstack/
      - Example
        - VNFC "vgw" with "Guaranteed QoS"
          - vCPU (Min/Max) - 16, Mem (Min/Max) - 32GB
          - Maps to "vgw-Guaranteed-QoS" flavor for OpenStack-based VIMs
          - Same VNFC with "Burstable QoS", 25% over-subscription
            - vCPU (Min) - 16, Mem (Min) - 32GB
            - vCPU (Max) - 20, Mem (Max) - 40GB

                                - Maps to "vgw-Burstable-QoS-25-percent-oversubscription" flavor for OpenStack-based VIMs
                        - VNFC "vDNS" with "Guaranteed QoS" & "Infrastructure High Availability"
                            - Maps to "vDNS-Guaranteed-QoS" flavor and "vDNS-infrastructure-high-availability" heat template
                - **Only certain pre-defined over-subscription values are allowed to simplify implementation**
                - *Implementation Notes:*
                    - *While dynamic injection of limit/reservation into flavor is ideal, a simple starting would be to be to switch to a pre-defined flavor in the environment file*
                        - *For aforementioned example*
                            - *Original flavor - "flavor-xyz-no-oversubscription"*
                            - *Modified flavor based on Policy - "flavor-xyz-25-percent-oversubscription"*
        - *Implementation Notes:*
            - *From an implementation stand point, MC would be exposing a Workload Deployment Policy (Intent) API*
                - *Input : deployment-intent, cloud owner, cloud region, deployment template, deployment environment file, ...*
                - *Output : Success or Failure with reason, modified deployment template, modified deployment environment file, ...*

SO MC API extension - aligned to the SO/MC API defined by **SO Casablanca HPA Design** to minimize the terminology set

(This data is sent from OOF to SO. SO transparently echoes this data to MC)

SO <-> MC Cloud-Agnostic Workload Deployment Policy API

```
#
#The same information is opaquely passed from OOF to SO
#

#
#Example 1: vCPE, Burstable QoS
#vCPE: Infrastructure Resource Isolation for VNF with Burstable QoS
#
   "oof_directives":{
      "directives":[
         {
            "vnfc_directives":[
               {
                  "vnfc_id":"vgw",
                  "directives":[
                     {
                        "directive_name":"Resource-Isolation-Intent-directive",
                        "attributes":[
                                                {
                              "attribute_name": "Infrastructure Resource Isolation for VNF",
                              "attribute_value": "Burstable QoS",
                           },
                                                {
                              "attribute_name": "Infrastructure Resource Isolation for VNF - Burstable QoS
Oversubscription Percentage",
                              "attribute_value": "25",
                           },

                        ]
                     },
                  ]
               },
            ]
         },
      ]
   }


#
#Example 2:
#vCPE: Infrastructure Resource Isolation for VNF with Guaranteed QoS
#
   "oof_directives":{
      "directives":[
         {
            "vnfc_directives":[
               {
                  "vnfc_id":"vgw",
                  "directives":[
                     {
```

```
                        "directive_name":"Resource-Isolation-Intent-directive",
                        "attributes":[
                                                {
                              "attribute_name": "Infrastructure Resource Isolation for VNF",
                              "attribute_value": "Guaranteed QoS",
                           },
                        ]
                     },
                  ]
               },
            ]
         },
      ]
   }

#
#Example 3:
#vDNS: Infrastructure HA for VNF & Infrastructure Resource Isolation for VNF with Burstable QoS
#
   "oof_directives":{
      "directives":[
         {
            "vnfc_directives":[
               {
                  "vnfc_id":"vdns",
                  "directives":[
                     {
                        "directive_name":"Resource-Isolation-Intent-directive",
                        "attributes":[
                                                {
                              "attribute_name": "Infrastructure Resource Isolation for VNF",
                              "attribute_value": "Burstable QoS",
                           },
                                                {
                              "attribute_name": "Infrastructure Resource Isolation for VNF - Burstable QoS
Oversubscription Percentage",
                              "attribute_value": "25",
                           },
                        ]
                     },
                                    {
                        "directive_name":"Infrastructure-HA-Intent-directive",
                        "attributes":[
                                                {
                                                       "attribute_name": "Infrastructure High Availability
for VNF",
                           },
                        ]
                                    },
                  ]
               },
            ]
         },
      ]
   }

#
# Example 4:
# vDNS: Infrastructure HA for VNF & Infrastructure Resource Isolation for VNF
# with Guaranteed QoS
#
   "oof_directives":{
      "directives":[
         {
            "vnfc_directives":[
               {
                  "vnfc_id":"vdns",
                  "directives":[
                     {
                        "directive_name":"Resource-Isolation-Intent-directive",
                        "attributes":[
```

```
                                        {
                           "attribute_name": "Infrastructure Resource Isolation for VNF",
                           "attribute_value": "Guaranteed QoS",
                        },
                     ]
                  },
                                 {
                     "directive_name":"Infrastructure-HA-Intent-directive",
                     "attributes":[
                                        {
                                               "attribute_name": "Infrastructure High Availability
for VNF",
                        },
                     ]

                                        },
               ]
            },
         ]
      },
   ]
}
```

Follow ups:

- Use Cases for Integration testing
  - vCPE
    - In the current state, this use case cannot support the intent "Infra HA for VMs in a VNF"
    - This use case has been tested in R2 with OOFMC capacity check API
  - vDNS
    - Can support intent "Infra HA for VMs in a VNF" and "Infrastructure Resource Isolation for VNF"
    - Nothing additional needed in OOF or MC
    - Changes needed in SO to call OOF API
      - **Marcus from Intel is driving this**
- Policy DB – is there any restriction on the type of json objects that can be stored?
  - **Matti to follow up with Ankit**

Implementation trade offs for Casablanca (R3) and potential Dublin (R4) plan:

- Deployment-Intent
  - 1. "Infrastructure Resource Isolation for VNF" – { "qosProperty": { {"Burstable QoS": "TRUE", "Burstable QoS Oversubscription Percentage": "25"} } }
    - Casablanca Plan
      - Only certain pre-defined over-subscription values are allowed to reflect practical deployment and simplify implementation
    - Dublin & Beyond Potential Plan
      - Creating instance types on demand for private clouds - to study
  - 2. Cloud-agnostic Workload Deployment Policy (Intent)
    - Casablanca Plan
      - Cloud-Agnostic Workload Deployment Policy (Intent) can be directly mapped to specific realization (e.g. OpenStack Flavor, Azure Instance Type) to simplify implementation.
    - Dublin & Beyond Potential Plan
      - VIM Capability Discovery to populate Intent in A&AI aligning taking into account Cloud selection policy based on cost specific to Intent (leverage similarities to HPA label discovery supported since R2)
        - VIM selection – Intent to be populated in A&AI for capability matching
        - VIM Deployment realization - Intent(s) to specific realization mapping (e.g. OpenStack Flavor, Azure Instance Type) to be populated in A&AI

Cloud Resource Partitioning for Differentiated QoS (Combined with Previous)

Value:

- Applicable to all use cases
- Casablanca Targets:
  - vCPE (Enable Tiered service offering); 5G Network Slicing (Stretch Goal)

References:

- **https://wiki.onap.org/download/attachments/28377610/ONAP-mc-of-resource-reservation.pptx?version=1&modificationDate=1526771712000&api=v2**

Edge Automation Requirement:

Support three types of slices in the Cloud Infrastructure (Definition Reference: https://kubernetes.io/docs/tasks/configure-pod-container/quality-service-pod/)

- Guaranteed Resource Slice (hard isolation) for various infra Resources (CPU/Memory/Network)
  - Max (limit), Min (request) are the same; resource guarantee is "Max"
  - Maps to 5G Applications such as Connected Car which fall in the category of ultra-reliable machine-type communications (ref. 1)
- Burstable Resource Slice (soft isolation) for various infra Resources
  - Min (request) <= Max (limit); resource guarantee is "Min"
  - Maps to Burstable Network Slice such > 1Gbps broadband which fall in the category of extreme mobile broadband (ref. 1)
- Best Effort Resource Slice (no isolation) for various infra Resources
  - No Min (request) ; resource guarantee is "None"
  - Maps to 5G Applications such as IoT which fall in the category of massive machine-type communications (ref. 1)

Implementation:

- Leverage current HPA framework with appropriate extensions

References:

- https://metis-ii.5g-ppp.eu/wp-content/uploads/white_papers/5G-RAN-Architecture-and-Functional-Design.pdf
- Driving Superior Isolation for Tiered Services using Resource Reservation -- Optimization Policies for Residential vCPE -https://jira.onap.org/browse/OPTFRA-240

Note:

- Any VMs/Containers which are part of a resource slice will adhere to the specs of the resource slice

ONAP Component	Life Cycle Phase	Enhancements
Policy	Design	Configuration Policies for Guaranteed, Burstable & Best Effort Cloud Infrastructure Resource Slices (this will apply to VMs/Containers also) Placement Policies for Resource Slices • Higher (programmable) weight to Cloud Region which supports all three types of resource slices vs only two types of resource slices (Guaranteed/Best Effort)
Multi-Cloud	Deploy	Resource Slice Capability Discovery
A&AI	Deploy	Resource Slice Capability per Cloud Region • Guaranteed/Burstable/Best Effort Resource Slice Type • Guaranteed/Burstable/Best Effort
OOF	Deploy	Execute Resource Slice Placement Policies for Optimized Service/VNF Placement across Cloud Regions

Aggregated Infrastructure Telemetry Streams (Aligns with HPA requirements, Combining efforts with HPA)

Value

- Edge Infrastructure Analytics complementing VNF Analytics
- Increase the accuracy of placement decisions
- Addresses gap in cloud provider solution – e.g. open source OpenStack does not have a comprehensive telemetry solution

Casablanca MVP

- HPA metrics visualization
- End-to-end use cases: vCPE, vDNS

Casablanca Stretch Goal

- OOF to use aggregated telemetry information for fine-grained optimization

MULTICLOUD-254 - Getting issue details... STATUS

ONAP, as in R2, collects the statistics/alarms/events from workloads (VMs) and take any close loop control actions such as Heal a process, scale-out, restart etc.. In R3 and beyond, infrastructure related statistics/alarms/events will be collected, generate actionable insights and take life cycle actions on the workloads. Infrastructure statistics normally include performance counters, NIC counters, IPMI information on per physical server node basis. To reduce the load on the ONAP, it is necessary that aggregated (summarized) information is sent to the ONAP from edge-clouds.

As part of this activity, intention is to create aggregation micro-service that collects the data from physical nodes (over collected and other mechanisms), aggregate the information (time based aggregation, threshold based aggregation, silencing etc.,..) based on the configurable rules and export the aggregate data to DCAE. This micro service can be instantiated by ONAP itself - one or more instances for edge-clouds at the ONAP-central itself using OOM, it could be instantiated at the edge-cloud using their own deployment tools or it could be deployed edge service providers at the regional site level.

In R3, functionality is limited to HPA features and visualization. R3 stretch goal: It collects information from each compute node for all HPA features and keeps track of health and resource information. It would use this information in placement decisions by OOF for accurate results.

Even though the aggregation service is being developed in Multi-Cloud project, it is expected that this can be deployed at various places. The decision to deploy at various levels can be due to performance and regulatory reasons. Following deployments are envisaged at this time:

- At the edge site level.
- At the regional site level (on behalf of set of edge sites).
- At the ONAP level (on behalf of set of edge sites)

Impacted projects (development activities)

ONAP Component	Enhancements
Overall	• Define models to represent summation information (Alerts/statistics/Events) for various groups • Defining various groups such as CPU usage, Memory usage, file descriptor usage, NIC utilization, various HPA features etc...

Multi-Cloud	• Development activities:
	  ◦ Prometheus based monitoring & summation
	  ◦ Support for collectd for statistics collection from NFVI nodes.
	  ◦ Support for VES agent to send the aggregate data to DCAE (Used when the aggregate service is instantiated outside of ONAP control)
	  ◦ Support for DMAAP agent to send the aggregate data to DCAE (Normally used if the aggregate service is instantiated at the ONAP-Central.
	  ◦ Provide ability to add new plugins (to collect statistics as well as to export aggregation information)
	  ◦ Provide ability to upload the recording and alert rules (on per edge-cloud basis or set of edge-clouds basis)
	  ◦ Ability to auto-cleanup of time series DB (based on size allocated for this micro-service)
	• Edge-Cloud registration time (as part of ESR)
	  ◦ Check whether registration data indicates whether the aggregation service to be brought up). If so, inform the aggregation micro service to authentication and listen for statistics from that edge-cloud.
	• Run time
	  ◦ Collects the information (support for both pull/push).
	  ◦ Apply rules
	  ◦ Generate alarms
	  ◦ Export them via VES or DMAPP or any other plugins in future.
	  ◦ Update A&AI HPA (Resources and health)

AAI & ESR	• ESR Development activities (Need to be done when Edges started to send aggregated data, so future requirement)
	  ◦ Enhancements to ESR to indicate whether aggregation service is required for this edge-cloud at the ONAP
	  ◦ Enhancements to ESR to indicate Multi-Cloud for Multi-Cloud to listen for connections and statistics requests from the edge-clouds. Information such as CA cert to use to authenticate the remote party or any other UN/PWD method.
	• HPA Enhancements
	  ◦ On per cloud-region, provide a a way to indicate whether a given HPA feature (that needs resources) resources are available and if so, the number of resources available.
	  ◦ On per cloud-region, provide a way to indicate the health of the HPA feature.

PORTAL	ESR portal related changes to take information about the edge-cloud (CA Cert and UN/PWD information) - Future when the edges started to send aggregate data)
OOF	HPA Enhancements • Current HPA filter only consider whether the cloud-region supports HPA capabilities via profiles. Enhancements to consider "availability of resources" and "health of HPA" on the cloud-region.

Life Cycle stages related functions

ONAP Component	Life cycle phase	Activities
AAI and ESR	Deploy & Run time	▪ Add/Modify/Delete recording and alerting rules
AAI and ESR	Run time	• Add/Modify/Delete Edge-cloud information
Multi-Cloud	Run time	• Get Edge information from A&AI whenever Edge-Cloud is added or removed. • Prepare to wait for information from that Edge-cloud • Receive information from edge-cloud and put it in the time series DB. • Summation based on recording & alerting rules ○ Export information to DCAE via DMAPP or VES
OOF	Run time	• HPA filter changes to accommodate health of the hardware/software on a feature and availability of resources.

High level architecture slides:

ONAP Edge Analytics with DCAE/DMaaP independent of closed loop (Stretch Goal - Beyond Casablanca)

Value

- 5G Analytics

ONAP Component	Life cycle phase	Enhancements

OOM - ONAP Central	Deploy	• Separate ONAP-edge Instance per 'edge domain', (ie., separate from onap-central instance, of course)
		  ○ Note: Independent of any Edge CP's Orchestration components.
		• SP uses a central-OOM with a 'policy' for deployment of an onap-edge instance, e.g., xyz edge provider with abc components, etc.
		  ○ However, onap-edge instance can be 'lighter weight' with subset of components needed (per MVP discussed below)
		  ○ Desirable to managed as a separate K8s cluster (ie., separate from onap-central instance, of course) and, only for onap-edge use, ie., don't use for other 'workloads' like network apps or 3rd party apps
		• Central OOM to deploy the following ONAP edge instance
		  ○ DMaaP with mirror capability

Multi-Cloud Deployment in Edge Cloud (Stretch Goal - Beyond Casablanca)

MULTICLOUD-262 - Getting issue details... STATUS

Value:

- Multi-Cloud service to assist in central A&AI scaling by caching A&AI data locally and syncing up with A&AI periodically