Migrating the Monitoring Agent

Migrating the Monitoring Agent

Each ESC instance has an agent to monitor it to enable ESC to control recovery and scaling operations. Following are the various scenarios that need migration of the monitoring agent:

1. Migrating from **local** to **distributed**

   For example:

   When introducing a new D-MONA into a data center.

2. Migrating from **distributed** to **local**

   For example:

   When performing a software upgrade.

3. Migrating from **distributed** to **distributed**

   For example:

   When performing load balancing.

4. Migrating many instances in quick succession from **distributed** to **distributed**

   For example:

   Disaster recovery

This section covers API that will enable migrating the monitoring agent without impacting the primary function of the VNF instance and also minimizing the impact on virtualisation (recovery/scaling).

The following three steps are performed by this API to process the monitoring update:

- Disable monitoring
- Service model update

- Re-enable monitoring

Executing the Monitoring Migration API

Method Type:

GET

VNFM Endpoint:

```
{http_scheme}://{api_root}/vnflcm/v2/ext/vnf_instances/{vnfInstanceId}/monitoring/migrate
```

HTTP Request Header:

```
Content-Type: application/json
```

Following are the examples for JSON payload:

Sample VnfMonitoring payload for migrating monitoring to a D-MONA instance (dmona1):

```
{
  "monitoring_agent": "dmona://dmona1",
  "key": "MONITORING_AGENT"
}
```

Sample for VnfMonitoring payload migrating monitoring to local MONA

```
{
  "monitoringAgent": "dmonaName://local_mona",
  "key": "MONITORING_AGENT"
}
```

Note A new string value is introduced to represent the central MONA component within ESC. It is used for the migration to local MONA by the previous API.

The following are the supported attribute names and data types for the migration request:

Table 1:

Attribute Name	Data Type	Description
`monitoring_agent`	Identifier	Deployment identifier of the monitoring agent. In the event the agent is local to ESC, the string must be set to `dmonaName://local_mona`.

Attribute Name	Data Type	Description
`key`	IdentifierInVnfd	This is the key in which the value for the monitoring agent should be stored. It **must** match the key used to identify the monitoring agent in the initial deployment. However, if the VNFD contained no agent definition then the key will reference a new KeyValue pair against which the agent reference should be stored, else update the existing value. **Note** If the key supplied does not match the initial Key used to specify a monitoring agent, a new key will be created to store the new value against the VnfInstance. If the deployment is terminated and then re-instantiated without a new value for the monitoring agent, then the old value is used, which may not be the required outcome.

VNF Notifications During Migration

Once a request received for migration, ESC sends notifications for LCM operations for a particular VNF.

Following is the example for Starting Notification:

```
{
    "vnfInstanceId": "fd0bcc11-3f22-4c91-b363-1def72619db8",
    "timeStamp": "2020-07-23T08:38:47.876Z",
    "isAutomaticInvocation": false,
    "notificationType": "InfrastructureOperationOccurrenceNotification",
    "operationState": "STARTING",
    "notificationStatus": "START",
    "vnfLcmOpOccId": "143cfc34-cc14-414d-9374-d70d01ae7b5a",
    "_links": {
        "vnfInstance": {
            "href":
"https://172.16.235.30:8251/vnflcm/v2/vnf_instances/fd0bcc11-3f22-4c91-b363-1def72619db8"
        },
        "vnfLcmOpOcc": {
```

```
            "href":
"https://172.16.235.30:8251/vnflcm/v2/vnf_lcm_op_occs/143cfc34-cc14-414d-9374-d70d01ae7b5a"

        },
        "subscription": {
            "href":
"https://172.16.235.30:8251/vnflcm/v2/subscriptions/e54d546a-6753-4f35-86fa-6ef8ac07a9de"
        }
    },
    "subscriptionId": "e54d546a-6753-4f35-86fa-6ef8ac07a9de",
    "operation": "MONITORING_MIGRATION",
    "id": "6b737d3f-a485-46d9-9276-6802eb48decd"
}
```

If required, you can subscribe for other notifications.

Note The migration API is an extension for the existing subscription endpoint, `VNFM-preferred` for all other LCM operations .

For more information on the Subscription, see the Subscribing to Notifications section in the Alarms and Notifications for ETSI LCM Operations chapter.

Error Scenarios

ETSI invokes the following error handling procedures for all its ETSI VNF lifecycle management (LCM) operations:

For more information on the VNF Lifecycle Management Error Handling Procedures, see Error Handling Procedures chapter.

A new property, `monitorMigration.terminalStateOnError`, is added to the ETSI service to determine what happens in the event of an error when ESC is performing the migration.

Error / Interrupt	ESC Behaviour	ETSI-VNFM Behaviour	Resulting LcmOpOcc state	ETSI-VNFM Behaviour Resulting LcmOpOcc state with * [1]

Validation Failure	• Send validation error • Rejects service update request	• Move operation to FAILED_TEMP • Send notification with problem details containing error message from ESC Manager.	FAILED_TEMP	**ETSI-VNFM Behaviour** • Move operation to FAILED • Send notification with problem details containing error message from ESC. **Resulting LcmOpOcc state** FAILED
Monitoring already unset	• ESCManager will reject service update for monitoring migration if any of the VM is in VM_MONITOR_UNSET_STATE	• Move operation to FAILED_TEMP • Send notification with problem details containing error message from ESC Manager.	FAILED_TEMP	**ETSI-VNFM Behaviour** • Move operation to FAILED • Send notification with problem details containing error message from ESC Manager. **Resulting LcmOpOcc state** FAILED

Unset monitor fails	• Unset monitor fails silently. • Deleting rule from existing monitoring agent failed. • Update deployment. • Sends service update success notification. • Set monitor on the new monitoring agent. • Send VM_SET_MONITOR_STATUS and SVC_SET_MONITOR_STATUS notifications.	• Move operation to COMPLETED • Send notification	COMPLETED	**ETSI-VNFM Behaviour** • Move operation to COMPLETED • Send notification **Resulting LcmOpOcc state** COMPLETED
Service Update fails	• Unset monitor on existing monitoring agent. • Deployment update failed. • Send service update failure notification. • Set monitor on the existing/previous monitoring agent based on if the deployment was actually updated. • Send VM_SET_MONITOR_STATUS notification. • Send SVC_SET_MONITOR_STATUS notification.	• Move operation to FAILED_TEMP • Send notification with problem details containing error message from ESC Manager.	FAILED_TEMP	**ETSI-VNFM Behaviour** • Send notification with problem details containing error message from ESC Manager. • Start rollback process (ROLLING_BACK) **Resulting LcmOpOcc state** ROLLING_BACK → ROLLED_BACK

Set monitor fails	• Unset monitor from existing monitoring agent. • Update deployment. • Send service update success notification. • Set monitor failed - Adding rule to new monitoring agent failed. • Send VM_SET_MONITOR_STATUS notification with failure state. • Skips set monitor for other VMs with same monitoring agent. • Send SVC_SET_MONITOR_STATUS notification with partial failure/failure notification.	• Move operation to FAILED_TEMP • Send notification with problem details containing error message from ESC Manager.	FAILED_TEMP	• Send notification with problem details containing error message from ESC Manager. • Start rollback process (ROLLING_BACK) **Resulting LcmOpOcc state** ROLLING_BACK → ROLLED_BACK
Unset monitor fails (rollback)	• ETSI should not rollback on unset monitor failure.	N/A	N/A	N/A

Service Update fails (rollback)	• If the deployment config was updated with the new monitoring agent during the service update failure, then a service update rollback will restore the previous monitoring agent and a set monitor is attempted on the previous monitoring agent. • If the deployment config was not updated due to service update failure, then a service update rollback will not be accepted by ESCManager (service update will not be accepted unless there is something to be updated).	• Move operation to FAILED_TEMP • Send notification with problem details containing error message from ESC Manager.	FAILED_TEMP	**ETSI-VNFM Behaviour** • Move operation to FAILED_TEMP • Send notification with problem details containing error message from ESC Manager. **Resulting LcmOpOcc state** FAILED_TEMP

Set monitor fails (rollback)	• Unset monitor on new monitoring agent (because deployment config was already updated successfully). • Update deployment with the previous monitoring agent. • Send service update success. • Set monitor on the previous monitoring agent. • Send VM_SET_MONITOR_STATUS notification with success/failure state. • Send SVC_SET_MONITOR_STATUS notification with success/failure/partial-failure state.	• Move operation to ROLLED_BACK • Send notification **Note:** Rollback only checks for the service update notification not the service level set monitor notification.	ROLLED_BACK	• Move operation to ROLLED_BACK • Send notification **Note:** Rollback only checks for the service update notification not the service level set monitor notification. **Resulting LcmOpOcc state** ROLLED_BACK
Cancel operation (during unset)	Since the request to ESC Core is atomic, cancel cannot be serviced.	N/A	N/A	N/A
Cancel operation (during service update)	Since the request to ESC Core is atomic, cancel cannot be serviced.	N/A	N/A	N/A
Cancel operation (during set)	Since the request to ESC Core is atomic, cancel cannot be serviced.	N/A	N/A	N/A

[1] monitorMigration.terminalStateOnErrorOutcome flag true