



# Understanding RME Device State Transition

---

The device participation in RME application tasks is no longer dependent on the device being in the Managed State as in RME 3.x. Instead, it will depend upon the actual device information required by the application task.

This allows RME to become flexible in meeting requirements to perform tasks on devices that have not yet been installed on the network.

Examples of such use-cases are pre-provisioning devices, managing spares, asset tracking, etc.

This section describes:

- The old State Management Behavior.
- The new State Management Architecture. These states are visible to you using the RME Device Management GUI.
- Examples of RME Application Tasks with description of the device information required for the task.

## RME 3.x Behavior

In RME 3.x, devices that were added by you to RME could fall into one of the following states:

- **Managed**—All inventory data for the device was collected from the device. Application tasks were initiated only on devices in this state, and you could only see these Managed devices in the device selector.

- **Suspended**—You have explicitly barred these devices from being included in Application tasks. These devices will not be seen in the device selector.
- **Not Responding**—When the device was added, Inventory data could not be collected for the device either because the device was unreachable or the credentials to access the device were incorrect. You have to correct any errors and manually resubmit the device for management.
- **Aliased Devices**—If the duplicate detection module in RME finds that this added device is a duplicate of an already managed device, this device is put into the Aliased state.
- **Conflicting State**—When the devices were added, a new device might have the same name as an already managed device but differences in one or more device passwords. A conflict might be found, for example, in the Read or Write community strings, Telnet password, or console-enable password.

Only devices in the managed state can be selected from the device selector for application tasks. Also, automated actions are taken only on Syslogs from managed devices. Most Syslog reports are also restricted to this set of devices.

Thus, in RME 3.x, a device had to be in managed state for most application tasks. However, some application tasks allowed devices that were not in the managed state. They were:

- **Import Status Report**—Displays the number of devices in the various RME 3.x states.
- **Syslog Unexpected Device Report**—Displays a list of Syslogs received from devices that are not in RME 3.x managed state.

## States in RME 4.x

The following describes the states a device can be in, and some sample application scenarios to explain the state.

The device states in RME, visible to all application are:

- **Normal**—Device has been successfully contacted by RME or the device has contacted RME at least once (polling, successful job completion, Syslog receipt etc.). This indicates that this is a real device in the network (at one point in time). Only the user can move a device out of this state.

This state does not guarantee that there was a successful Inventory Collection. Also note that the Pre-deployed and Normal states are meant primarily for the information of the end-user.

As in the Pre-deployed state, RME should allow/disallow application tasks purely based upon the presence/absence of the information required for that task.

In the device selector, devices in the normal state will appear in the appropriate MDF –based groups, and in custom groups, if the data necessary to resolve group membership is available.

Application tasks initiation and execution will behave exactly as described for devices in the pre-deployed state.

- Pre-deployed—Device has never ever been contacted by RME by any protocol (SNMP, SSH, etc.). Any successful contact with the device (be it SNMP polling, pre-provisioned job completion, etc.) will take it out of this state and into the Normal State.

This state helps you to identify the devices that are to be deployed. These devices will appear in the device selector in a separate group, Pre-deployed. These devices also appear under the appropriate MDF-based groups, based on the information entered in DCR.

The operator can initiate application tasks (includes jobs) using devices in this state by using the device selector in the same way as Normal devices.

However, some application tasks may not execute or be scheduled, since the information required for that task is not yet available in RME.

Thus, Software Management's Distribution Method by devices [Advanced] flow job creation task succeeds, since there is no data required from the device either current or cached.

On the other hand, Software Management Device Centric job creation task that requires Image Recommendation fails since the required Flash and Image data is not yet available in RME.

Application tasks will succeed or fail on a best-effort basis without depending on whether there is a full Inventory Collection performed on the device.

RME will have inventory and Configuration polling and collection jobs for all devices in the pre-deployed state. If any of these jobs successfully contact the device, the device is moved to Normal state.

- Pending—This is a transient state, and, in an operational RME, no device will be in this state for any significant time.

These devices appear in the device selector in a separate group, Pending. The operator can initiate application tasks (including jobs) by selecting these devices from the device selector.

When the device is added to RME, device management puts the device into this state, and invokes all the registered application tasks such as Inventory Collection, Configuration collection, etc. Based on the results of the tasks, the device goes into Pre-deployed, Normal or Aliased states.

- Suspended—You may suspend a device because you do not want to manage it, have it participate in any RME application tasks. Only you can move a device out of this state. RME application does not allow devices in this state to participate in normal RME application tasks (except for special cases like the Suspended Device Report).

Suspended devices do not appear in the device selector. Syslogs from these devices do not cause any automated actions to happen. However, syslogs from devices in this state are collected and reported on.

- Aliased—Devices that have been detected as duplicates, at the time of Device Addition, as done in RME 3.x. These devices cannot participate in RME application tasks (except for special cases like the Duplicate Device Report).

These states are visible to you in the Device Management GUI and the Device Summary Report. Here you can see how many devices are in each of the states.

There is a report that allows you to select a set of devices and view the states that they are in.

- Conflicting State—Devices that have the same name/IP Address as an already managed device, but the sysObjectID for the two conflict. The user has to resolve this conflict. Note that the already managed device continues to be in the managed state, and participates in all application tasks. The new entry with conflicting credentials will go to the conflicting state.

These states will be made visible to the user via the Device Management GUI and the device summary report to allow the user to see how many devices are in each of the states.

# State Transitions

The following defines how the state of a device can change in RME 4.0

## Pending

1. Inventory Collection for the device has succeeded, and the alias detection algorithm has determined that this device is an alias of a device that is already in the Normal state. The device is moved to the Aliased state requiring user interaction.
2. If:
  - History for the device exists, the device is directly moved from pending to normal state, without waiting for the results of the registered tasks.
  - There is no history for the device, tasks registered with Device Management (such as Inventory Collection or Config Collection) for the device has succeeded in contacting the device.

Alias detection algorithm either determined this was not an alias of an existing device, or did not have sufficient information to make this decision. Device goes to the Normal state.

3. None of the tasks registered with Device Management were successful in contacting the device. The device goes to the Pre-deployed state.

## Pre-deployed

1. Any RME task succeeds in contacting the device. The device is moved to Normal state.
2. Operator picks the device in Pre-deployed state, and suspends it. The device moves to Suspended state.
3. Operator picks the device in Pre-deployed state and deletes it from RME. The device goes out of RME.

## Normal

1. This is a rare case: Device d1 and d2 are aliases, d1 is added to RME, goes to pre-deployed state. The device d2 also goes into pre-deployed state. RME does not know that they are aliased. The d1 and d2 devices are in separate jobs. When job j1 runs, d1 is contacted and moves to Normal state. RME still does not know that d2 is an alias. Now, job j2 runs on d2 device, and d2 device is moved to Normal state. Now alias detection should catch that d1 and d2 are aliases, and should move d2 to Aliased state from Normal state.
2. Operator picks the device in Normal state and deletes it from RME. The device goes out of RME
3. Operator picks the device in Normal state, and suspends it. The device moves to Suspended state

## Aliased

1. User manually re-submits the device for RME management. The device goes to the Pending state.
2. Operator picks the device in Aliased state and deletes it from RME. The device goes out of RME

## Suspended

1. Operator picks the device in Suspended state and deletes it from RME. The device goes out of RME.
2. User manually re-submits the device for RME management. The device goes to the Pending state.

## Conflicting

1. Operator picks the device in Conflicting state and deletes it from RME. The device goes out of RME.
2. User manually updates the device credentials for RME management. The device goes to the Pending state.

## RME 4.x Scenarios

The following section describes the RME 4.x scenarios:

### Device Addition

This describes the application task where RME does not have the automatic synchronization with Device and Credential Repository (DCR) enabled.

1. Operator adds one or more devices to DCR, providing all the required attributes and credentials.

The mandatory fields from DCR are:

- a. Management IP address or Host Name or Device ID, Display Name.
  - b. Device MDF category or SysObjectID is expected to be valid value for the Device Type field. This is required to instantiate the correct device package for RME jobs (for example, NetConfig job creation).
2. DCR informs RME that new devices have been added, and RME updates its picker list.
  3. Operator selects devices from the picker list to add to RME.
  4. For each selected device, Device Management puts the device into pending state. It then invokes the registered RME application tasks such as Inventory Collection, Config Collection etc.
    - If no task succeeds in contacting the device, the device is put into the Pre-deployed state by Device Management.
    - If RME's alias detection is successful and finds that this device is a duplicate of an existing device in either the Pre-deployed or Normal states, the new device is put into the Aliased state.
    - If alias detection does not mark this device as being Aliased (either because it did not have sufficient information to make the decision or the algorithm identifies the device to be unique)and
    - If any of the application tasks in step 4 succeed in contacting (exchanging any packets) with the device, the device is moved to the Normal state. Note that no application task has to succeed in order for this to happen – Merely contact is sufficient.

5. The device selector displays all devices that are in the Pending, Pre-deployed as well as the Normal states.

The devices in Pre-deployed state appear in the system-defined MDF groups as well as a special group called Pre-deployed.

Similarly, devices in the pending state will also appear in the special group Pending as well as in the relevant locations in the MDF grouping. This helps you to quickly select all such devices for some specific task.

Device Management's Device Summary report displays the number of devices in each of the above states.



#### Note

---

Inventory Collection and Inventory Polling will have a system defined periodic job that will operate on all devices in both the Pre-Deployed and the Normal states. The actual set of devices to be collected or polled on will be determined at the job execution time.

---

## Inventory Detailed Device Report

1. The device selector for this task displays all devices in the Normal and Pre-Deployed states.
2. You can select a set of devices for their detailed report.
3. The application will make use of the application local states to determine the display format of the details for each device. Those devices that have no details collected at all will simply display `No Details Available`.
4. Reports for devices with partial data display this partial data in the appropriate locations.

## Configuration Deployment Using Config Editor

1. A base-lined Config or an external file is selected for editing.
2. After editing is completed, the file is saved.
3. You can create a deployment job, by selecting this recently saved file.

You can select one or more devices from the Device Selector. These selected devices could be in pending, pre-deployed or normal states This task does not need any additional info from the Inventory DB (except for device type).

4. Download mode can be either Merge or Diff if the device is in the normal state and at least one version of the configuration for the device has been archived. Only merge mode is possible in the case of Pending or Pre-deployed devices or for those devices in the Normal state for which no configuration is archived yet.
5. Job Option: Different Configuration Versions Considered Failure: This requires at least one configuration version to be archived.
6. Job Option: Sync Archive before Job execution: Applicable to devices in all of pending, Pre-deployed and Normal states. Failure to get the configuration at job execution time fails the job.
7. Job Option: Write running to startup configuration: Applicable to devices in all Pending, Pre-Deployed and Normal States.
8. You can enter the job policies and scheduling information, and submit the job.
9. At the scheduled time, the job is executed. (For Job Execution, there is no mandatory requirement for the existence of any inventory or Config Archive data) There are two cases:
  - Device is reachable: Config attempts to fulfill the Job options chosen, and deploys the Config to the device. If the device was in the pending or pre-deployed state, Device Management is informed that the device now is contactable.
  - Device not reachable: The Config deployment to this device is marked as failed, and no state change is requested for.

## Configuration Changes Using NetConfig

1. Select a set of devices from the device selector that contains all of Pending, Normal and Pre-deployed devices.
2. Select a set of task types to be executed on these devices (Banner change, SNMP Credentials change etc.).
3. For each device type and task type selected, you must enter the relevant parameters.
4. If a particular task requires inventory data (example, Cable templates that require Interface details) and if the required inventory data is not available, you will be prompted with a warning that the job cannot include that particular task. You can continue with job creation.

5. The following job options are available:
  - Different Configuration Versions Considered Failure—Requires at least one configuration version to be archived.
  - Rollback failure policies—At job creation time, all the rollback policies for failure are applicable to devices in any of pending, Pre-Deployed and Normal states.
  - Sync Archive before Job execution—Applicable to devices in any of Pending, Pre-deployed and Normal states. Failure to get the Config at job execution time fails the job.
  - Write running to startup configuration—Applicable to devices in any of Pending, Pre-Deployed and Normal States.
6. You can submit the job after selecting the job policies and entering the job scheduling information.
7. At the job execution time, before deploying the Config changes:
  - If sync archive policy is selected, and if archival fails (at the time of execution) the job fails. (Current behavior).
  - If rollback policy is selected, and no configuration has been collected for this device, the job execution will proceed with a warning in the job results.
  - If the device is reachable: NetConfig attempts to fulfill the selected job options, and deploys the configuration to the device. If the device was in the Pre-deployed or Pending states, Device Management is informed that the device is now contactable.
  - If the device not reachable: The configuration deployment to this device is marked as failed, and no state change is requested for.

## Software Image Upgrade

Only the Image Distribution by Device (advanced) flow is considered:

1. Select the Image Distribution by Device (advanced) task. The inventory data is not required.
2. Enter the image, target device and the target destination on the device. The devices can be in the Normal, Pending, or Pre-deployed states.

3. For each device Software Management attempts to get required data to do the verification

Examples of the data required for this verification include the boot loader version for IOS devices, or the Version running on the supervisor for a Card to be upgraded, and physical attributes like free flash on the given partition. The exhaustive data list is device type specific and will be provided in the Software Management design documents. This data is got from Inventory via ADI.

If verification is possible with the data for the device, Software Management does the verification and marks the device as verification passed or failed. If there is insufficient data for this operation, Software Management marks the device as skipped due to insufficient data, however if the user had specified best-effort verification, the job gets scheduled for execution with the skipped devices included. If the user had chosen mandatory verification, the job creation will fail.

4. At the execution time, Software Management attempts to contact each device. If the device is contactable, and Software Management's internal checks work out, the image is downloaded to the device, and this device is marked as successful. If the device was in pending or pre-deployed states at the time of job execution, Software Management will inform Device Management that the device was contacted.
5. If the device is not contactable at the job execution time, Software Management will mark this download as failed, and will continue on with the rest of the devices in the job. No state changes will be effected.

