



## ブラウフィールド VM の展開

ここでは、ESCでの単純なブラウフィールド展開の展開方法を示します。VMが作成されてインポートされると、ブラウフィールド展開としてESCに展開されます。ブラウフィールド展開が成功すると、ESCはESCで作成された他のVMと同様にそのVMを監視および管理します。VMの展開の詳細については、[VMの展開 \(3 ページ\)](#) を参照してください。

- [VNFのインポート \(1 ページ\)](#)
- [VMの展開 \(3 ページ\)](#)
- [VMの作成 \(5 ページ\)](#)
- [VMの展開解除 \(5 ページ\)](#)
- [ブラウフィールドモードでのVMの操作 \(5 ページ\)](#)

### VNFのインポート

アクティブ VNF をインポートするには、次の手順を実行します。

1. RPC ロードを通じて、ランタイムデータを含む XML を ESC に送信します。

次に例を示します。

```
esc_nc_cli import-deployment-data CREATE admin name-dep /opt/cisco/existing_vms.xml
```

2. 次のコマンドを使用して、関連する dep.xml ファイルを ESC に展開します。

```
esc_nc_cli edit-config dep.xml
```

3. 必要に応じて、展開の問題を修正します。問題を修正するには、次の手順を実行します。

1. インポートしたデータを再発行します (ステップ 1)。

2. dep.xml ファイルを展開解除します。

次に例を示します。

```
esc_cli delete-dep aTenantName aDeploymentName
```

3. CLI を使用して dep.xml を再展開します。

次に例を示します。

```
esc_cli edit-config dep.xml
```

4. 必要に応じてステップ a ~ c を繰り返します。
4. RPC を呼び出して、ESC が VNF を完全に管理することを宣言します。  
次に例を示します。

```
esc_nc_cli import-deployment-data FINALIZE admin name-dep
```

サンプル import.xml の例

```
<import>
  <vms>
    <vm_details> <!--First VM details-->
      <flavor/>
      <host/>
      <host_id/>
      <image/>
      <port/>
      <uuid>1bb008e1-d4dd-4107-bc45-7e298a5ac510</uuid>
      <name>parvmoha-bf-vm</name>
      <attached_volume/>
    </vm_details>
    <vm_details>
      ..... <!--add more vms using vm_details tag-->
    </vm_details>
  </vms>
  <deployment_name>parvmoha-dep</deployment_name>
  <project_name>admin</project_name>
  <project_uuid>563fba7044c847a6a370cc10d5ef7d57</project_uuid>
</import>
```

dep.xml の例 :

```
<?xml version="1.0" encoding="UTF-8"?>
<esc_datamodel xmlns="http://www.cisco.com/esc/esc">
  <tenants>
    <tenant>
      <name>aTenantName</name>
      <managed_resource>>false</managed_resource>
      <deployments>
        <deployment>
          <name>brownfield-deployment</name>
          <vm_group>
            <name>g2</name>
            <vim_vm_name>vm-cirros</vim_vm_name>
            <image>Automation-Cirros-Image</image>
            <flavor>Automation-Cirros-Flavor</flavor>
            <bootup_time>100</bootup_time>
            <recovery_wait_time>0</recovery_wait_time>
            <interfaces>
              <interface>
                <nicid>0</nicid>
                <network>esc-net</network>
              </interface>
            </interfaces>
          </vm_group>
          <vim_interface_name>vm-cirros-interface</vim_interface_name>
        </deployment>
      </deployments>
      <kpi_data>
        <kpi>
          <event_name>VM_ALIVE</event_name>
          <metric_value>1</metric_value>
          <metric_cond>GT</metric_cond>
          <metric_type>UINT32</metric_type>
        </kpi>
      </kpi_data>
    </tenant>
  </tenants>
</esc_datamodel>
```

```

        <metric_occurrences_true>2</metric_occurrences_true>
        <metric_occurrences_true>3</metric_occurrences_true>
        <metric_collector>
          <nicid>0</nicid>
          <type>ICMPPing</type>
          <poll_frequency>3</poll_frequency>
          <polling_unit>seconds</polling_unit>
          <continuous_alarm>>false</continuous_alarm>
        </metric_collector>
      </kpi>
    </kpi_data>
    <rules>
      <admin_rules>
        <rule>
          <event_name>VM_ALIVE</event_name>
          <action>ALWAYS log</action>
          <action>TRUE servicebooted.sh</action>
          <action>FALSE recover autohealing</action>
        </rule>
      </admin_rules>
    </rules>
    <config_data />
    <scaling>
      <min_active>1</min_active>
      <max_active>1</max_active>
      <elastic>>true</elastic>
    </scaling>
    <recovery_policy>
      <recovery_type>AUTO</recovery_type>
      <action_on_recovery>REBOOT_THEN_REDEPLOY</action_on_recovery>
      <max_retries>1</max_retries>
    </recovery_policy>
  </vm_group>
</deployment>
</deployments>
</tenant>
</tenants>
</esc_datamodel>

```

### インポートの制限

- VM グループごとに 1 つの VM のみがサポートされます。
- スケーリングはサポートされていません。

## VM の展開

展開中にインポートされたコンポーネントが更新されていないことを確認することをお勧めします。たとえば、VM、サブネット、または OpenStack で実行されているポートです。

次の例は、OpenStack にブラウフィールドを展開する方法を示しています。

```

<?xml version="1.0" encoding="UTF-8"?>
<esc_datamodel xmlns="http://www.cisco.com/esc/esc">
  <tenants>
    <tenant>
      <name>aTenantName</name>
      <managed_resource>>false</managed_resource>
    </tenant>
  </tenants>
</esc_datamodel>

```

```

<deployment>
  <name>brownfield-deployment</name>
  <vm_group>
    <name>g2</name>
    <vim_vm_name>vm-cirros</vim_vm_name>
    <image>Automation-Cirros-Image</image>
    <flavor>Automation-Cirros-Flavor</flavor>
    <bootup_time>100</bootup_time>
    <recovery_wait_time>0</recovery_wait_time>
    <interfaces>
      <interface>
        <nicid>0</nicid>
        <network>esc-net</network>
        <vim_interface_name>vm-cirros-interface</vim_interface_name>
      </interface>
    </interfaces>
    <kpi_data>
      <kpi>
        <event_name>VM_ALIVE</event_name>
        <metric_value>1</metric_value>
        <metric_cond>GT</metric_cond>
        <metric_type>UINT32</metric_type>
        <metric_occurrences_true>2</metric_occurrences_true>
        <metric_occurrences_true>3</metric_occurrences_true>
        <metric_collector>
          <nicid>0</nicid>
          <type>ICMPPing</type>
          <poll_frequency>3</poll_frequency>
          <polling_unit>seconds</polling_unit>
          <continuous_alarm>>false</continuous_alarm>
        </metric_collector>
      </kpi>
    </kpi_data>
    <rules>
      <admin_rules>
        <rule>
          <event_name>VM_ALIVE</event_name>
          <action>ALWAYS log</action>
          <action>TRUE servicebooted.sh</action>
          <action>FALSE recover autohealing</action>
        </rule>
      </admin_rules>
    </rules>
    <config_data />
    <scaling>
      <min_active>1</min_active>
      <max_active>1</max_active>
      <elastic>>true</elastic>
    </scaling>
    <recovery_policy>
      <recovery_type>AUTO</recovery_type>
      <action_on_recovery>REBOOT_THEN_REDEPLOY</action_on_recovery>
      <max_retries>1</max_retries>
    </recovery_policy>
  </vm_group>
</deployment>
</deployments>
</tenant>
</tenants>
</esc_datamodel>

```



- (注) 展開モードがブラウフィールドの [ファイナライズされたインポート (Finalized Import)] としてマークされている場合、VNF が ESC によって完全に管理されていることを意味します。ただし、完全にインポート済みとしてマークされている場合、この展開をインポートモードに戻すことはできません。

## VM の作成

展開後、実際のコールをバイパスして、ブラウフィールドモードの OpenStack でポートと VM を作成します。

ESC では、dep.xml 内の展開およびテナント名だけが import.xml を使用してロードされた内容と一致する場合、ブラウフィールド展開が検出されます。ブラウフィールド展開が検出されると、ESC は VIM が稼働しているかどうか、dep.xml で指定されたイメージとフレーバーが使用可能かどうかをチェックします。

ESC は、dep.xml で指定されたカスタムポートがアクティブかどうかもチェックします。VIM ドライバは、VM の作成要求を受信すると、実際の OpenStack の作成操作をバイパスし、DB から読み取った VNF の UUID を返します。

## VM の展開解除

ブラウフィールド展開が正常に展開されると、ブラウフィールドモードを [ファイナライズされたインポート (Finalized Import)] に変更した後にのみ、ESC によって完全に管理されます。

展開がブラウフィールドモードの場合、VIM で VM を削除せずにブラウフィールド展開を ESC から展開解除できます。

展開が成功したら、ブラウフィールドモードを [ファイナライズされたインポート (Finalized Import)] に変更できます。ブラウフィールドモードが [ファイナライズされたインポート (Finalized Import)] の場合、VIM から VM が削除されます。

## ブラウフィールドモードでの VM の操作

ブラウフィールドモードで VM のすべての操作をブロックします。ブラウフィールド展開が発生すると、VM が展開され、VM\_ALIVE 状態がトリガーされます (該当する場合)。その後、展開は SERVICE\_ACTIVE 状態に移行します。

ブラウフィールド展開が展開されていて、ブラウフィールドモードが [オン (ON)] の場合、手動でトリガーされたすべての VM アクション (START、STOP、RECOVER、REBOOT、ENABLE/DISABLE MONITOR) がブロックされます。

ブラウフィールド展開の一部として展開された VM は、最初はブラウフィールドモードが [オン (ON)] の状態です。ブラウフィールドモードを手動でオフにした後にはのみ、VM は ESC によって完全に管理されます。したがって、ブラウフィールドモードでの VM 操作 (手動または自動トリガー) はブロックされます。

ユーザが、まだ完了していないブラウフィールド展開に対してサービスアクションまたは VM アクションを手動でトリガーしようとする、検証エラーが表示されます。次に、検証エラーメッセージの例を示します。

```
<?xml version="1.0" encoding="UTF-8"?>
<rpc-reply xmlns="urn:ietf:params:xml:ns:netconf:base:1.0" message-id="1">
  <rpc-error>
    <error-type>application</error-type>
    <error-tag>operation-failed</error-tag>
    <error-severity>error</error-severity>
    <error-path xmlns:esc="http://www.cisco.com/esc/esc"
xmlns:nc="urn:ietf:params:xml:ns:netconf:base:1.0">
      /nc:rpc/esc:recoveryVmAction
    </error-path>
    <error-message xml:lang="en">Exception from action callback: Recovery VM action
action not allowed in brownfield mode.</error-message>
    <error-info>
      <bad-element>recoveryVmAction</bad-element>
    </error-info>
  </rpc-error>
</rpc-reply>
```

## MONA トリガーイベントのブロックング

ブラウフィールドの展開が成功すると、VM\_DEPLOYED および VM\_ALIVE イベントが送信されます。展開が SERVICE\_ACTIVE 状態に移行します。この展開では、ブラウフィールドモードが [オン (ON)] になっています。

展開を手動で終了する前に MONA から RECOVERY イベントがトリガーされると、展開は SERVICE\_ERROR 状態に移行し、インポートが失敗したことを知らせる通知が ESC からノースバウンドに送信されます。イベントに影響するその他の VM はブロックされ、MONA によってトリガーされた VM アクションがブロックされたことを知らせる VM\_ACTION\_BLOCKED 通知が ESC からノースバウンドに送信されます。

例：

```
<?xml version="1.0" encoding="UTF-8"?>
<notification xmlns="urn:ietf:params:xml:ns:netconf:notification:1.0">
  <eventTime>2019-10-03T14:06:53.042+00:00</eventTime>
  <escEvent xmlns="http://www.cisco.com/esc/esc">
    <status>SUCCESS</status>
    <status_code>200</status_code>
    <status_message>VM Action [VM_RECOVERY_AUTO] Blocked for VM
[parvmiha-cirros-_Grp1_0_89192a9f-d624-46c7-a264-bb8326ec5d29].</status_message>
    <event>
      <type>VM_ACTION_BLOCKED</type>
    </event>
  </escEvent>
</notification>
```

ブラウフィールドモードを [オフ (OFF)] に切り替えて、VM を手動で回復できます。この場合、ESC の変更がブロックされたため、サービスは SERVICE\_ACTIVE 状態のままです。ブ

ブラウフィールドモードが [オフ (OFF) ] になると、サービスが `SERVICE_ACTIVE` 状態になるため、`CONFD` の `svc-action` は手動リカバリを拒否します。ブラウフィールドモードを [オフ (OFF) ] にした後、VM を手動で回復するには、`recovery-vm-action` コマンドを使用します。



---

(注) `VM_ALIVE` イベント通知は、`SERVICE_ACTIVE` 状態への最初の移行に必要なためブロックされません。

---

