



# CHAPTER 3

## Cisco VNMC XML API メソッドの説明

この章では、次の内容を説明します。

- 「Cisco VNMC XML API メソッド」(P.3-1)

API メソッドの詳細は以下に示されています。

### Cisco VNMC XML API メソッド

これらのメソッドは、GUI コンソールからも呼び出されます。ここでは、API メソッド、構文（要求および応答）、使用例について説明します。Cisco VNMC の API メソッドは次のように定義されます。

- [aaaGetRemoteUserRoles](#)
- [aaaGetUserLocales](#)
- [aaaKeepAlive](#)
- [aaaLogin](#)
- [aaaLogout](#)
- [aaaRefresh](#)
- [configConfFiltered](#)
- [configConfMo](#)
- [configConfMoGroup](#)
- [configConfMos](#)
- [configFindDnsByClassId](#)
- [configConfMoGroup](#)
- [configMoChangeEvent](#)
- [configResolveClass](#)
- [configResolveClasses](#)
- [configResolveDn](#)
- [configResolveDns](#)
- [configResolveParent](#)
- [configScope](#)
- [eventSendHeartbeat](#)
- [eventSubscribe](#)
- [eventSubscribeApps](#)
- [faultAckFault](#)
- [faultAckFaults](#)
- [faultResolveFault](#)
- [loggingSyncOcns](#)
- [methodVessel](#)
- [orgResolveElements](#)
- [orgResolveInScope](#)
- [orgResolveLogicalParents](#)
- [policyEstimateImpact](#)
- [poolResolveInScope](#)

### aaaGetRemoteUserRoles

この例は、リモート ロケーションのユーザ特権を返します。

#### 要求構文

```
<xs:element name="aaaGetRemoteUserRoles" type="aaaGetRemoteUserRoles"
substitutionGroup="externalMethod"/>
  <xs:complexType name="aaaGetRemoteUserRoles" mixed="true">
    <xs:attribute name="inRemoteUserName">
      <xs:simpleType>
```

```

        <xs:restriction base="xs:string">
            <xs:pattern value="[a-zA-Z][a-zA-Z0-9_@-]{0,31}"/>
        </xs:restriction>
    </xs:simpleType>
</xs:attribute>
<xs:attribute name="cookie" type="xs:string"/>
<xs:attribute name="response" type="YesOrNo"/>
</xs:complexType>

```

## 応答構文

```

<xs:element name="aaaGetRemoteUserRoles" type="aaaGetRemoteUserRoles"
substitutionGroup="externalMethod"/>
    <xs:complexType name="aaaGetRemoteUserRoles" mixed="true">
        <xs:attribute name="outRemoteUserPriv">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:pattern
value="((policy|aaa|read-only|admin|tenant|operations|res-config|fault),){0,7}(policy|aaa|
read-only|admin|tenant|operations|res-config|fault){0,1}"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="cookie" type="xs:string"/>
        <xs:attribute name="response" type="YesOrNo"/>
        <xs:attribute name="errorCode" type="xs:unsignedInt"/>
        <xs:attribute name="errorDescr" type="xs:string"/>
        <xs:attribute name="invocationResult" type="xs:string"/>
    </xs:complexType>

```

## 例

### 要求

```

<aaaGetRemoteUserRoles
  cookie="<real_cookie>"
  inRemoteUserName="adminuser"
  outRemoteUserPriv/>

```

### 応答

```

<aaaGetRemoteUserRoles
  cookie="<real_cookie>"
  commCookie="11/15/0/2964"
  srcExtSys="10.193.33.109"
  destExtSys="10.193.33.109"
  srcSvc="sam_extXMLApi"
  destSvc="mgmt-controller_dme"
  response="yes"
  outRemoteUserPriv="admin">
</aaaGetRemoteUserRoles>

```

## aaaGetUserLocales

この例は、認可されたユーザ ロケーションのリストを返します。

## 要求構文

```
<xs:element name="aaaGetUserLocales" type="aaaGetUserLocales"
substitutionGroup="externalMethod"/>
  <xs:complexType name="aaaGetUserLocales" mixed="true">
    <xs:attribute name="inUserName">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:pattern value="[a-zA-Z][a-zA-Z0-9_@-]{0,31}"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="inIsUserRemote">
      <xs:simpleType>
        <xs:union memberTypes="xs:boolean">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="no"/>
              <xs:enumeration value="yes"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:union>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
  </xs:complexType>
```

## 応答構文

```
<xs:element name="aaaGetUserLocales" type="aaaGetUserLocales"
substitutionGroup="externalMethod"/>
  <xs:complexType name="aaaGetUserLocales" mixed="true">
    <xs:attribute name="outUserLocales">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:minLength value="0"/>
          <xs:maxLength value="512"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
  </xs:complexType>
```

## 例

### 要求

```
<aaaGetUserLocales
  cookie="<real_cookie>"
  inUserName="john"
  inIsUserRemote="no"
  outUserLocales/>
```

### 応答

```
<aaaGetUserLocales
```

```

    cookie="<real_cookie>"
    commCookie="11/15/0/2962"
    srcExtSys="10.193.33.109"
    destExtSys="10.193.33.109"
    srcSvc="sam_extXMLApi"
    destSvc="mgmt-controller_dme"
    response="yes"
    outUserLocales="TestSanity">
</aaaGetUserLocales>

```

## aaaKeepAlive

この例では、デフォルトのセッション時間が経過し、メソッド呼び出しの後で同じ Cookie が使用されるまでセッションがアクティブなままになります。

### 要求構文

```

<xs:element name="aaaKeepAlive" type="aaaKeepAlive" substitutionGroup="externalMethod"/>
  <xs:complexType name="aaaKeepAlive" mixed="true">
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
  </xs:complexType>

```

### 応答構文

```

<xs:element name="aaaKeepAlive" type="aaaKeepAlive" substitutionGroup="externalMethod"/>
  <xs:complexType name="aaaKeepAlive" mixed="true">
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
  </xs:complexType>

```

## 例

### 要求

```

<aaaKeepAlive
  cookie="<real_cookie>" />

```

### 応答

```

<aaaKeepAlive
  cookie="<real_cookie>"
  commCookie="11/15/0/2969"
  srcExtSys="10.193.33.109"
  destExtSys="10.193.33.109"
  srcSvc="sam_extXMLApi"
  destSvc="mgmt-controller_dme"
  response="yes">
</aaaKeepAlive>

```

## aaaLogin

この例は、セッションを開始するために必要で、クライアントと VNMCM 間で認証された HTTPS セッションを確立するログインプロセスを示しています。

### 要求構文

```
<xs:element name="aaaLogin" type="aaaLogin" substitutionGroup="externalMethod"/>
  <xs:complexType name="aaaLogin" mixed="true">
    <xs:attribute name="inName">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:pattern value="[¥-¥.:_a-zA-Z0-9]{0,16}"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="inPassword">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:minLength value="0"/>
          <xs:maxLength value="512"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
  </xs:complexType>
```

### 応答構文

```
<xs:element name="aaaLogin" type="aaaLogin" substitutionGroup="externalMethod"/>
  <xs:complexType name="aaaLogin" mixed="true">
    <xs:attribute name="outCookie" type="xs:string"/>
    <xs:attribute name="outRefreshPeriod" type="xs:unsignedInt"/>
    <xs:attribute name="outPriv">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:pattern
value="((policy|aaa|read-only|admin|tenant|operations|res-config|fault),){0,7}(policy|aaa|
read-only|admin|tenant|operations|res-config|fault){0,1}"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="outDomains" type="xs:string"/>
    <xs:attribute name="outChannel">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="fullssl"/>
          <xs:enumeration value="noencssl"/>
          <xs:enumeration value="plain"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="outEvtChannel">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="fullssl"/>
          <xs:enumeration value="noencssl"/>
          <xs:enumeration value="plain"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:complexType>
```

```

        </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="outSessionId">
        <xs:simpleType>
            <xs:restriction base="xs:string">
                <xs:minLength value="1"/>
                <xs:maxLength value="64"/>
            </xs:restriction>
        </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="outVersion" type="xs:string"/>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
</xs:complexType>

```

## 例

### 要求

```

<aaaLogin
  inName="admin"
  inPassword="Nbv12345"/>

```

### 応答

```

<aaaLogin cookie=""
  commCookie=""
  srcExtSys="0.0.0.0"
  destExtSys="0.0.0.0"
  srcSvc="" destSvc=""
  response="yes"
  outCookie="<real_cookie>"
  outRefreshPeriod="600"
  outPriv="admin"
  outDomains=""
  outChannel="fullssl"
  outEvtChannel="fullssl"
  outSessionId="web_49019"
  outVersion="1.0 (0.39938) ">
</aaaLogin>

```

## aaaLogout

次に、現在のセッションを終了するログアウト プロセスの例を示します。デフォルトのセッション時間が経過すると、このプロセスは自動的に呼び出されます。

### 要求構文

```

<xs:element name="aaaLogout" type="aaaLogout" substitutionGroup="externalMethod"/>
<xs:complexType name="aaaLogout" mixed="true">
  <xs:attribute name="inCookie" type="xs:string"/>
  <xs:attribute name="cookie" type="xs:string"/>
  <xs:attribute name="response" type="YesOrNo"/>
</xs:complexType>

```

## 応答構文

```
<xs:element name="aaaLogout" type="aaaLogout" substitutionGroup="externalMethod"/>
  <xs:complexType name="aaaLogout" mixed="true">
    <xs:attribute name="outStatus">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="success"/>
          <xs:enumeration value="failure"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
  </xs:complexType>
```

## 例

### 要求

```
<aaaLogout
  inCookie="<real_cookie>"
  outStatus/>
```

### 応答

```
<aaaLogout cookie=""
  commCookie=""
  srcExtSys="0.0.0.0"
  destExtSys="0.0.0.0"
  srcSvc=""
  destSvc=""
  response="yes"
  outStatus="success">
</aaaLogout>
```

## aaaRefresh

セッションはユーザ アクティビティによってアクティブに保つことができます（デフォルトのセッション時間枠内）。デフォルトでは、アクティビティがない時点から 7200 秒カウントダウンされます。7200 秒が経過すると、Cisco VNMC はスリープ モードを開始し、再度サインインを要求します。これにより、カウントダウンが再起動されます。セッションは同じセッション ID を使用し続けます。



(注) このメソッドを使用すると、以前の Cookie の有効期限が切れ、新しい Cookie が発行されます。

## 要求構文

```
<xs:element name="aaaRefresh" type="aaaRefresh" substitutionGroup="externalMethod"/>
  <xs:complexType name="aaaRefresh" mixed="true">
    <xs:attribute name="inName">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:pattern value="[¥-¥.:_a-zA-Z0-9]{0,16}"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:attribute>
  </xs:complexType>
```

```

        </xs:restriction>
    </xs:simpleType>
</xs:attribute>
<xs:attribute name="inPassword">
    <xs:simpleType>
        <xs:restriction base="xs:string">
            <xs:minLength value="0"/>
            <xs:maxLength value="512"/>
        </xs:restriction>
    </xs:simpleType>
</xs:attribute>
<xs:attribute name="inCookie" type="xs:string"/>
<xs:attribute name="cookie" type="xs:string"/>
<xs:attribute name="response" type="YesOrNo"/>
</xs:complexType>

```

## 応答構文

```

<xs:element name="aaaRefresh" type="aaaRefresh" substitutionGroup="externalMethod"/>
    <xs:complexType name="aaaRefresh" mixed="true">
        <xs:attribute name="outCookie" type="xs:string"/>
        <xs:attribute name="outRefreshPeriod" type="xs:unsignedInt"/>
        <xs:attribute name="outPriv">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:pattern
value="( (policy|aaa|read-only|admin|tenant|operations|res-config|fault), ) {0,7} (policy|aaa|
read-only|admin|tenant|operations|res-config|fault) {0,1}"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="outDomains" type="xs:string"/>
        <xs:attribute name="outChannel">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:enumeration value="fullssl"/>
                    <xs:enumeration value="noencssl"/>
                    <xs:enumeration value="plain"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="outEvtChannel">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:enumeration value="fullssl"/>
                    <xs:enumeration value="noencssl"/>
                    <xs:enumeration value="plain"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:attribute>
        <xs:attribute name="cookie" type="xs:string"/>
        <xs:attribute name="response" type="YesOrNo"/>
        <xs:attribute name="errorCode" type="xs:unsignedInt"/>
        <xs:attribute name="errorDescr" type="xs:string"/>
        <xs:attribute name="invocationResult" type="xs:string"/>
    </xs:complexType>

```

## 例

### 要求



```
<aaaRefresh
  cookie="<real_cookie>"
  inName="admin"
  inPassword="Nbv12345"
  inCookie="<real_cookie>"/>
```

### 応答

```
<aaaRefresh
  cookie="<real_cookie>"
  commCookie="" srcExtSys="0.0.0.0"
  destExtSys="0.0.0.0"
  srcSvc=""
  destSvc=""
  response="yes"
  outCookie="<real_cookie>"
  outRefreshPeriod="600"
  outPriv="admin"
  outDomains=""
  outChannel="fullssl"
  outEvtChannel="fullssl">
</aaaRefresh>
```

## configConfFiltered

次に、設定されたポリシーに基づいて、データおよびアクティビティが制限されている例を示します。

### 要求構文

```
<xs:element name="configConfFiltered" type="configConfFiltered"
  substitutionGroup="externalMethod"/>
  <xs:complexType name="configConfFiltered" mixed="true">
    <xs:all>
      <xs:element name="inFilter" type="filterFilter" minOccurs="0"/>
      <xs:element name="inConfig" type="configConfig" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="inHierarchical">
      <xs:simpleType>
        <xs:union memberTypes="xs:boolean">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="no"/>
              <xs:enumeration value="yes"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:union>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="classId" type="namingClassId"/>
  </xs:complexType>
```

### 応答構文

```
<xs:element name="configConfFiltered" type="configConfFiltered"
  substitutionGroup="externalMethod"/>
  <xs:complexType name="configConfFiltered" mixed="true">
```

```

<xs:all>
  <xs:element name="outConfigs" type="configSet" minOccurs="0"/>
</xs:all>
<xs:attribute name="cookie" type="xs:string"/>
<xs:attribute name="response" type="YesOrNo"/>
<xs:attribute name="errorCode" type="xs:unsignedInt"/>
<xs:attribute name="errorDescr" type="xs:string"/>
<xs:attribute name="invocationResult" type="xs:string"/>
<xs:attribute name="classId" type="namingClassId"/>
</xs:complexType>

```

## 例

### 要求

```

<configConfFiltered
  cookie="<real_cookie>"
  inHierarchical="false"
  classId="orgTenant">
  <inFilter>
    <eq class="orgTenant"
      property="name"
      value="Cisco" />
  </inFilter>
  <inConfig>
    <orgDatacenter
      dn="org-HR"
      descr="HR (Human Resources- new Descr)"/>
  </inConfig>
</configConfFiltered>

```

### 応答

```

<configConfFiltered
  cookie="<real_cookie>"
  commCookie="5/15/0/617"
  srcExtSys="10.193.33.206"
  destExtSys="10.193.33.206"
  srcSvc="sam_extXMLApi"
  destSvc="resource-mgr_dme"
  response="yes"
  classId="orgTenant">
  <outConfigs>
    <orgDatacenter
      descr="HR (Human Resources- new Descr)"
      dn="org-root/org-Cisco/org-HR"
      fltAggr="0"
      level="2"
      name="HR"
      status="modified"/>
  </outConfigs>
</configConfFiltered>

```

## configConfMo

次に、単一のサブツリー（DN など）で、指定された管理対象オブジェクトを設定する例を示します。

## 要求構文

```
<xs:element name="configConfMo" type="configConfMo" substitutionGroup="externalMethod"/>
  <xs:complexType name="configConfMo" mixed="true">
    <xs:all>
      <xs:element name="inConfig" type="configConfig" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="inHierarchical">
      <xs:simpleType>
        <xs:union memberTypes="xs:boolean">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="no"/>
              <xs:enumeration value="yes"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:union>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="dn" type="referenceObject"/>
  </xs:complexType>
```

## 応答構文

```
<xs:element name="configConfMo" type="configConfMo" substitutionGroup="externalMethod"/>
  <xs:complexType name="configConfMo" mixed="true">
    <xs:all>
      <xs:element name="outConfig" type="configConfig" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
    <xs:attribute name="dn" type="referenceObject"/>
  </xs:complexType>
```

## 例

### 要求

```
<configConfMo
  dn=""
  cookie="<real_cookie>"
  inHierarchical="false">
  <inConfig>
    <aaaLdapEp
      attribute="CiscoAvPair"
      basedn="dc=pasadena,dc=cisco,dc=com"
      descr=""
      dn="sys/ldap-ext"
      filter="sAMAccountName=$userid"
      retries="1"
      status="modified"
      timeout="30"/>
    </inConfig>
  </configConfMo>
```

### 応答

```

<configConfMo
  dn=""
  cookie="<real_cookie>"
  commCookie="11/15/0/28"
  srcExtSys="10.193.33.101"
  destExtSys="10.193.33.101"
  srcSvc="sam_extXMLApi"
  destSvc="mgmt-controller_dme"
  response="yes">
</outConfig>
  <aaaLdapEp
    attribute="CiscoAvPair"
    basedn="dc=pasadena,dc=cisco,dc=com"
    childAction="deleteNonPresent"
    descr=""
    dn="sys/ldap-ext"
    filter="sAMAccountName=$userid"
    fsmDescr=""
    fsmPrev="updateEpSuccess"
    fsmProgr="100"
    fsmStageDescr=""
    fsmStamp="2010-11-22T23:41:01.826"
    fsmStatus="nop"
    fsmTry="0"
    intId="10027"
    name=""
    retries="1"
    status="modified"
    timeout="30"/>
  </outConfig>
</configConfMo>

```

## configConfMoGroup

次に、設定されたポリシーに基づいて、管理対象オブジェクトのグループを設定する例を示します。

### 要求構文

```

<xs:element name="configConfMoGroup" type="configConfMoGroup"
  substitutionGroup="externalMethod"/>
  <xs:complexType name="configConfMoGroup" mixed="true">
    <xs:all>
      <xs:element name="inDns" type="dnSet" minOccurs="0"/>
      <xs:element name="inConfig" type="configConfig" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="inHierarchical">
      <xs:simpleType>
        <xs:union memberTypes="xs:boolean">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="no"/>
              <xs:enumeration value="yes"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:union>
      </xs:attribute>
      <xs:attribute name="cookie" type="xs:string"/>
      <xs:attribute name="response" type="YesOrNo"/>
    </xs:complexType>

```

```
</xs:complexType>
```

## 応答構文

```
<xs:element name="configConfMoGroup" type="configConfMoGroup"
substitutionGroup="externalMethod"/>
  <xs:complexType name="configConfMoGroup" mixed="true">
    <xs:all>
      <xs:element name="outConfigs" type="configSet" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
  </xs:complexType>
```

## 例



(注) orgDataCenter (org-root/org-Cisco および org-root/org-Cola 下) の descr プロパティが変更されます。descr プロパティは暗黙的でないため、変更できます。暗黙的な場合、変更は適用されず、新しい orgDataCenter が作成されます。

### 要求

```
<configConfMoGroup
  cookie="<real_cookie>"
  inHierarchical="false">
  <inDns>
    <dn value="org-root/org-Cisco" />
    <dn value="org-root/org-Cola" />
  </inDns>
  <inConfig>
    <orgDatacenter
      dn="org-HR"
      descr="HR (Human Resources)"/>
    </inConfig>
  </configConfMoGroup>
```

### 応答

```
<configConfMoGroup
  cookie="<real_cookie>"
  commCookie="5/15/0/600"
  srcExtSys="10.193.33.206"
  destExtSys="10.193.33.206"
  srcSvc="sam_extXMLApi"
  destSvc="resource-mgr_dme"
  response="yes">
  <outConfigs>
    <orgDatacenter
      descr="HR (Human Resources) "
      dn="org-root/org-Cola/org-HR"
      fltAggr="0"
      level="2"
      name="HR"
      status="modified"/>
    <orgDatacenter
      descr="HR (Human Resources) "
```

```

        dn="org-root/org-Cisco/org-HR"
        fltAggr="0"
        level="2"
        name="HR"
        status="modified"/>
    </outConfigs>
</configConfMoGroup>

```

## configConfMos

次に、複数のサブツリーで DN を使用して管理対象オブジェクトを設定する例を示します。

### 要求構文

```

<xs:element name="configConfMos" type="configConfMos" substitutionGroup="externalMethod"/>
  <xs:complexType name="configConfMos" mixed="true">
    <xs:all>
      <xs:element name="inConfigs" type="configMap" minOccurs="0">
        <xs:unique name="unique_map_key_2">
          <xs:selector xpath="pair"/>
          <xs:field xpath="@key"/>
        </xs:unique>
      </xs:element>
    </xs:all>
    <xs:attribute name="inHierarchical">
      <xs:simpleType>
        <xs:union memberTypes="xs:boolean">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="no"/>
              <xs:enumeration value="yes"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:union>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
  </xs:complexType>

```

### 応答構文

```

<xs:element name="configConfMos" type="configConfMos" substitutionGroup="externalMethod"/>
  <xs:complexType name="configConfMos" mixed="true">
    <xs:all>
      <xs:element name="outConfigs" type="configMap" minOccurs="0">
        <xs:unique name="unique_map_key_4">
          <xs:selector xpath="pair"/>
          <xs:field xpath="@key"/>
        </xs:unique>
      </xs:element>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
  </xs:complexType>

```

## 例

## 要求

```

<configConfMos
  cookie="<real_cookie>"
  <inConfigs>
    <pair key="org-root/logprof-default">
      <policyLogProfile dn="org-root/logprof-default"
        name="default"
        level="debug1"
        size="10000000"
        backupCount="4"/>
    </pair>

    <!-- Update Controller Device Profile -->
    <pair key="org-root/controller-profile-default">
      <policyControllerDeviceProfile
        dn="org-root/controller-profile-default"
        adminState="enabled">

        <commDnsProvider hostip="171.70.168.183" order="1"/>
        <commDnsProvider hostip="171.68.226.120" order="2"/>
        <commDnsProvider hostip="64.102.6.247" order="3"/>
      </policyControllerDeviceProfile>
    </pair>
  </inConfigs>
</configConfMos>

```

## 応答

```

<configConfMos
  cookie="<real_cookie>"
  commCookie="7/15/0/1a74"
  srcExtSys="10.193.34.70"
  destExtSys="10.193.34.70"
  srcSvc="sam_extXMLApi"
  destSvc="policy-mgr_dme"
  response="yes">
<outConfigs>
  <pair key="org-root/logprof-default">
    <policyLogProfile
      adminState="enabled"
      backupCount="4"
      descr="the log level for every process"
      dn="org-root/logprof-default"
      intId="10065"
      level="debug1"
      name="default"
      size="10000000"/>
  </pair>
  <pair key="org-root/controller-profile-default">
    <policyControllerDeviceProfile
      adminState="enabled"
      coreFilePolicy=""
      descr="default profile for management server virtual machine"
      dn="org-root/controller-profile-default"
      dnsPolicy=""
      faultPolicy="default"
      httpPolicy="default"
      httpsPolicy="default"
      intId="10057"
      logProfilePolicy="default"
      name="default"

```

```

        snmpPolicy=""
        syslogPolicy=""
        telnetPolicy=""
        timezone=""/>
    </pair>
</outConfigs>
</configConfMos>

```

## configFindDnsByClassId

次に、識別名を検索し、クラス ID でソートされた識別名を返す例を示します。

### 要求構文

```

<xs:element name="configFindDnsByClassId" type="configFindDnsByClassId"
substitutionGroup="externalMethod"/>
  <xs:complexType name="configFindDnsByClassId" mixed="true">
    <xs:all>
      <xs:element name="inFilter" type="filterFilter" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="classId" type="namingClassId"/>
  </xs:complexType>

```

### 応答構文

```

<xs:element name="configFindDnsByClassId" type="configFindDnsByClassId"
substitutionGroup="externalMethod"/>
  <xs:complexType name="configFindDnsByClassId" mixed="true">
    <xs:all>
      <xs:element name="outDns" type="dnSet" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
    <xs:attribute name="classId" type="namingClassId"/>
  </xs:complexType>

```

### 例

#### 要求

```

<configFindDnsByClassId
  classId="computeBlade"
  cookie="<real_cookie>" />

```

#### 応答

```

<configFindDnsByClassId
  cookie="<real_cookie>"
  response="yes"
  classId="computeBlade">
  <outDns>
    <dn value="sys/chassis-1/blade-7"/>
    <dn value="sys/chassis-1/blade-5"/>
  </outDns>
</configFindDnsByClassId>

```



```

    <dn value="sys/chassis-1/blade-3"/>
    <dn value="sys/chassis-1/blade-1"/>
  </outDns>
</configFindDnsByClassId>

```

## configMoChangeEvent

次に、指定された管理対象オブジェクトに対する変更イベントを返す例を示します。

### 要求構文

```

<xs:element name="configMoChangeEvent" type="configMoChangeEvent"
substitutionGroup="externalMethod"/>
  <xs:complexType name="configMoChangeEvent" mixed="true">
    <xs:all>
      <xs:element name="inConfig" type="configConfig" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="inEid" type="xs:unsignedLong"/>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
  </xs:complexType>

```

### 応答構文

```

<xs:element name="configMoChangeEvent" type="configMoChangeEvent"
substitutionGroup="externalMethod"/>
  <xs:complexType name="configMoChangeEvent" mixed="true">
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
  </xs:complexType>

```

### 例

#### 要求

```

<configMoChangeEvent
  cookie=""
  commCookie=""
  srcExtSys="10.193.33.120"
  destExtSys="10.193.33.120"
  srcSvc="service-reg_dme"
  destSvc="sam_extXMLApi"
  inEid="71449">
  <inConfig>
    <extpolRegistry
      dn="extpol/reg"
      lastPollTs="2010-11-12T20:33:51.071"
      status="modified"/>
    </inConfig>
  </configMoChangeEvent>

```

#### 応答

```

Event (12:35:20:675):
<eventSendHeartbeat

```

```

    cookie="0/0/0/2a74"
    commCookie=""
    srcExtSys="0.0.0.0"
    destExtSys="0.0.0.0"
    srcSvc="" destSvc=""
    response="yes"
    outSystemTime="2010-11-12T20:34:19.630">
</eventSendHeartbeat>

Event (12:35:20:690):
<eventSendHeartbeat
  cookie="0/0/0/2a74"
  commCookie=""
  srcExtSys="0.0.0.0"
  destExtSys="0.0.0.0"
  srcSvc="" destSvc=""
  response="yes"
  outSystemTime="2010-11-12T20:34:19.630">
</eventSendHeartbeat>

```

## configResolveChildren

次に、管理対象情報ツリーで特定の DN 下にある管理対象オブジェクトの子を取得する例を示します。返される子の数を減らすためにフィルタを使用できます。

### 要求構文

```

<xs:element name="configResolveChildren" type="configResolveChildren"
substitutionGroup="externalMethod"/>
  <xs:complexType name="configResolveChildren" mixed="true">
    <xs:all>
      <xs:element name="inFilter" type="filterFilter" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="inDn" type="referenceObject"/>
    <xs:attribute name="inHierarchical">
      <xs:simpleType>
        <xs:union memberTypes="xs:boolean">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="no"/>
              <xs:enumeration value="yes"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:union>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="classId" type="namingClassId"/>
  </xs:complexType>

```

### 応答構文

```

<xs:element name="configResolveChildren" type="configResolveChildren"
substitutionGroup="externalMethod"/>
  <xs:complexType name="configResolveChildren" mixed="true">
    <xs:all>
      <xs:element name="outConfigs" type="configSet" minOccurs="0"/>
    </xs:all>

```

```

<xs:attribute name="cookie" type="xs:string"/>
<xs:attribute name="response" type="YesOrNo"/>
<xs:attribute name="errorCode" type="xs:unsignedInt"/>
<xs:attribute name="errorDescr" type="xs:string"/>
<xs:attribute name="invocationResult" type="xs:string"/>
<xs:attribute name="classId" type="namingClassId"/>
</xs:complexType>

```

## 例

### 要求

```

<configResolveChildren
  cookie="<real_cookie>"
  classId="aaaUser"
  inDn="sys/user-ext"
  inHierarchical="false">
  <inFilter>
  </inFilter>
</configResolveChildren>

```

### 応答

```

<configResolveChildren
  cookie="<real_cookie>"
  commCookie="11/15/0/2a59"
  srcExtSys="10.193.33.120"
  destExtSys="10.193.33.120"
  srcSvc="sam_extXMLApi"
  destSvc="mgmt-controller_dme"
  response="yes"
  classId="aaaUser">
  <outConfigs>
    <aaaUser descr="" dn="sys/user-ext/user-doe"
      email="" expiration="never" expires="no" firstName="John" intId="12999"
      lastName="Doe" name="doe" phone="" priv="admin,read-only" pwdSet="yes"/>
    <aaaUser descr="" dn="sys/user-ext/user-jacks" email="" expiration="never"
      expires="no" firstName="Play" intId="12734" lastName="Jacks" name="jacks"
      phone="" priv="fault,operations,policy,read-only,res-config,tenant"
      pwdSet="yes"/>
    <aaaUser descr="" dn="sys/user-ext/user-admin" email="" expiration="never"
      expires="no" firstName="" intId="10052" lastName="" name="admin" phone=""
      priv="admin,read-only" pwdSet="yes"/>
    <aaaUser descr="" dn="sys/user-ext/user-over" email="" expiration="never"
      expires="no" firstName="Roll" intId="12711" lastName="Over" name="over"
      phone="" priv="fault,operations,policy,read-only,res-config,tenant"
      pwdSet="yes"/>
    <aaaUser descr="" dn="sys/user-ext/user-fun" email="" expiration="never"
      expires="no" firstName="Have" intId="12708" lastName="Fun" name="fun" phone=""
      priv="read-only" pwdSet="yes"/>
    <aaaUser descr="testuser" dn="sys/user-ext/user-aaa" email="" expiration="never"
      expires="no" firstName="a" intId="10620" lastName="aa" name="aaa" phone=""
      priv="aaa,read-only" pwdSet="no"/>
  </outConfigs>
</configResolveChildren>

```

## configResolveClass

次に、該当するクラスの要求された管理対象オブジェクトを返す例を示します。*inHierarchical* が true の場合は、オブジェクトに子が含まれます。

## 要求構文

```
<xs:element name="configResolveClass" type="configResolveClass"
substitutionGroup="externalMethod"/>
  <xs:complexType name="configResolveClass" mixed="true">
    <xs:all>
      <xs:element name="inFilter" type="filterFilter" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="inHierarchical">
      <xs:simpleType>
        <xs:union memberTypes="xs:boolean">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="no"/>
              <xs:enumeration value="yes"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:union>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="classId" type="namingClassId"/>
  </xs:complexType>
```

## 応答構文

```
<xs:element name="configResolveClass" type="configResolveClass"
substitutionGroup="externalMethod"/>
  <xs:complexType name="configResolveClass" mixed="true">
    <xs:all>
      <xs:element name="outConfigs" type="configSet" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
    <xs:attribute name="classId" type="namingClassId"/>
  </xs:complexType>
```

## 例

### 要求

```
<configResolveClass
  cookie="<real_cookie>"
  classId="pkiEp"
  inHierarchical="false">
  <inFilter>
  </inFilter>
</configResolveClass>
```

### 応答

```
<configResolveClass
  cookie="<real_cookie>"
  commCookie="11/15/0/2a5b"
  srcExtSys="10.193.33.120"
  destExtSys="10.193.33.120"
  srcSvc="sam_extXMLApi"
```

```

destSvc="mgmt-controller_dme"
response="yes"
classId="pkiEp">
<outConfigs>
  <pkiEp descr=""
    dn="sys/pki-ext"
    intId="10037"
    name=""/>
</outConfigs>
</configResolveClass>

```

## configResolveClasses

次に、複数のクラスの要求された管理対象オブジェクトを返す例を示します。*inHierarchical* が *true* の場合は、オブジェクトに子が含まれます。

### 要求構文

```

<xs:element name="configResolveClasses" type="configResolveClasses"
substitutionGroup="externalMethod"/>
  <xs:complexType name="configResolveClasses" mixed="true">
    <xs:all>
      <xs:element name="inIds" type="classIdSet" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="inHierarchical">
      <xs:simpleType>
        <xs:union memberTypes="xs:boolean">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="no"/>
              <xs:enumeration value="yes"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:union>
      </xs:attribute>
      <xs:attribute name="cookie" type="xs:string"/>
      <xs:attribute name="response" type="YesOrNo"/>
    </xs:complexType>

```

### 応答構文

```

<xs:element name="configResolveClasses" type="configResolveClasses"
substitutionGroup="externalMethod"/>
  <xs:complexType name="configResolveClasses" mixed="true">
    <xs:all>
      <xs:element name="outConfigs" type="configSet" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
  </xs:complexType>

```

## 例

## 要求

```
<configResolveClasses
  cookie="<real_cookie>"
  inHierarchical="false">
  <inIds>
    <Id value="computeBlade"/>
    <Id value="equipmentChassis"/>
  </inIds>
</configResolveClasses>
```

## 応答

(この応答は、省略されています)。

```
<configResolveClasses
  cookie="<real_cookie>"
  response="yes">
  <outConfigs>
    <computeBlade
      adminPower="policy"
      adminState="in-service"
      dn="sys/chassis-1/blade-1"
      .
      .
    <computeBlade
      adminPower="policy"
      adminState="in-service"
      dn="sys/chassis-1/blade-3"
      .
      .
    <computeBlade
      adminPower="policy"
      adminState="in-service"
      dn="sys/chassis-1/blade-5"
      .
      .
    <computeBlade
      adminPower="policy"
      adminState="in-service"
      dn="sys/chassis-1/blade-7".
      .
      .
    <equipmentChassis
      adminState="acknowledged"
      configState="ok"
      .
      .
  </outConfigs>
</configResolveClasses>
```

## configResolveDn

次に、指定された DN について単一の管理対象オブジェクトを取得する例を示します。

## 要求構文

```
<xs:element name="configResolveDn" type="configResolveDn"
  substitutionGroup="externalMethod"/>
```

```

<xs:complexType name="configResolveDn" mixed="true">
  <xs:attribute name="inHierarchical">
    <xs:simpleType>
      <xs:union memberTypes="xs:boolean">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="no"/>
            <xs:enumeration value="yes"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:union>
    </xs:attribute>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="dn" type="referenceObject"/>
  </xs:complexType>

```

## 応答構文

```

<xs:element name="configResolveDn" type="configResolveDn"
substitutionGroup="externalMethod"/>
  <xs:complexType name="configResolveDn" mixed="true">
    <xs:all>
      <xs:element name="outConfig" type="configConfig" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
    <xs:attribute name="dn" type="referenceObject"/>
  </xs:complexType>

```

## 例

### 要求

```

<configResolveDn
  cookie="<real_cookie>"
  dn="vmmEp/vm-mgr-vcenter1" />

```

### 応答

```

<configResolveDn dn="vmmEp/vm-mgr-vcenter1"
  cookie="<real_cookie>"
  commCookie="9/15/0/1c0d"
  srcExtSys="10.193.34.70"
  destExtSys="10.193.34.70"
  srcSvc="sam_extXMLApi"
  destSvc="vm-mgr_dme"
  response="yes">
  <outConfig>
    <vmManager
      adminState="enable"
      descr=""
      dn="vmmEp/vm-mgr-vcenter1"
      fltAggr="0"
      fsmDescr="AG registration with
vCenter (FSM: sam:dme:VmManagerRegisterWithVCenter)"
      fsmPrev="RegisterWithVCenterRegistering"
      fsmProgr="13"
    </vmManager>
  </outConfig>

```

```

        fsmRmtInvErrCode="none"
        fsmRmtInvErrDescr=""
        fsmRmtInvRsIt=""
        fsmStageDescr="AG registration with
vCenter (FSM-STAGE:sam:dme:VmManagerRegisterWithVCenter:Registering)"
        fsmStamp="2010-11-11T21:37:15.696"
        fsmStatus="RegisterWithVCenterRegistering"
        fsmTry="1"
        hostName="savbu-vpod-dev-31.cisco.com"
        intId="21959"
        name="vcenter1"
        operState="unknown"
        stateQual=""
        type="vmware"
        version=""/>
    </outConfig>
</configResolveDns>

```

## configResolveDns

次に、DN のリストについて管理対象オブジェクトを取得する例を示します。

### 要求構文

```

<xs:element name="configResolveDns" type="configResolveDns"
substitutionGroup="externalMethod"/>
  <xs:complexType name="configResolveDns" mixed="true">
    <xs:all>
      <xs:element name="inDns" type="dnSet" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="inHierarchical">
      <xs:simpleType>
        <xs:union memberTypes="xs:boolean">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="no"/>
              <xs:enumeration value="yes"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:union>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
  </xs:complexType>

```

### 応答構文

```

<xs:element name="configResolveDns" type="configResolveDns"
substitutionGroup="externalMethod"/>
  <xs:complexType name="configResolveDns" mixed="true">
    <xs:all>
      <xs:element name="outConfigs" type="configSet" minOccurs="0"/>
      <xs:element name="outUnresolved" type="dnSet" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
  </xs:complexType>

```



```
<xs:attribute name="invocationResult" type="xs:string"/>
</xs:complexType>
```

## 例

### 要求

```
<configResolveDns
  cookie="<real_cookie>"
  inHierarchical="false">
  <inDns>
    <dn value="sys/chassis-1" />
    <dn value="sys/chassis-1/blade-1/board/cpu-1" />
    <dn value="sys/chassis-1/blade-1/board/t-stats" />
  </inDns>
</configResolveDns>
```

### 応答

```
<configResolveDns
  cookie="<real_cookie>"
  response="yes">
  <outConfigs>
    <processorUnit
      arch="Xeon"
      cores="4"
      dn="sys/chassis-1/blade-1/board/cpu-1"
      id="1"
      model="Intel(R) Xeon(R) CPU E5520 @ 2.27GHz"
      operState="operable"
      operability="operable"
      perf="not-supported"
      power="not-supported"
      presence="equipped"
      revision="0"
      serial=""
      socketDesignation="CPU1"
      speed="2.266000"
      stepping="5"
      thermal="ok"
      threads="8"
      vendor="Intel(R) Corporation"
      voltage="ok"/>
    <equipmentChassis
      adminState="acknowledged"
      configState="ok"
      connPath="A,B"
      connStatus="A,B"
      dn="sys/chassis-1"
      fabricEpDn="fabric/server/chassis-1"
      fltAggr="2"
      fsmDescr=""
      fsmPrev="PsuPolicyConfigSuccess"
      fsmProgr="100"
      fsmRmtInvErrCode="none"
      fsmRmtInvErrDescr=""
      fsmRmtInvRslt=""
      fsmStageDescr=""
      fsmStamp="2009-09-13T21:34:37"
      fsmStatus="nop"
      fsmTry="0"
      id="1"
      lcTs="1969-12-31T16:00:00"
```

```

        managingInst="A"
        model="N20-C6508"
        operQualifier="fabric-conn-problem"
        operState="fabric-conn-problem"
        operability="operable"
        power="ok"
        presence="unknown"
        revision="0"
        serial="FOX1252GG84"
        thermal="ok"
        vendor="Cisco Systems Inc"
        versionHolder="yes"/>
    </outConfigs>
    <outUnresolved>
        <dn value="sys/chassis-1/blade-1/board/t-stats"/>
    </outUnresolved>
</configResolveDns>

```

## configResolveParent

次に、指定された DN について管理対象オブジェクトの親を取得する例を示します。

### 要求構文

```

<xs:element name="configResolveParent" type="configResolveParent"
substitutionGroup="externalMethod"/>
  <xs:complexType name="configResolveParent" mixed="true">
    <xs:attribute name="inHierarchical">
      <xs:simpleType>
        <xs:union memberTypes="xs:boolean">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="no"/>
              <xs:enumeration value="yes"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:union>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="dn" type="referenceObject"/>
  </xs:complexType>

```

### 応答構文

```

<xs:element name="configResolveParent" type="configResolveParent"
substitutionGroup="externalMethod"/>
  <xs:complexType name="configResolveParent" mixed="true">
    <xs:all>
      <xs:element name="outConfig" type="configConfig" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
    <xs:attribute name="dn" type="referenceObject"/>
  </xs:complexType>

```

## 例

## 要求

```
<configResolveParent
  cookie="<real_cookie>"
  inHierarchical="false"
  dn="sys/chassis-1/blade-1/adaptor-1">
</configResolveParent>
```

## 応答

```
<configResolveParent dn="sys/chassis-1/blade-1/adaptor-1"
  cookie="<real_cookie>"
  response="yes">
<outConfig>
  <computeBlade
    adminPower="policy"
    adminState="in-service"
    assignedToDn=""
    association="none"
    availability="available"
    availableMemory="10240"
    chassisId="1"
    checkPoint="discovered"
    connPath="A,B"
    connStatus="A,B"
    descr=""
    diagnostics="complete"
    discovery="complete"
    dn="sys/chassis-1/blade-1"
    fltAggr="0"
    fsmDescr=""
    fsmFlags=""
    fsmPrev="DiscoverSuccess"
    fsmProgr="100"
    fsmRmtInvErrCode="none"
    fsmRmtInvErrDescr=""
    fsmRmtInvRslt=""
    fsmStageDescr=""
    fsmStamp="2009-09-23T23:44:30"
    fsmStatus="nop"
    fsmTry="0"
    intId="768052"
    lc="discovered"
    lcTs="1969-12-31T16:00:00"
    managingInst="B"
    model="N20-B6620-1"
    name=""
    numOfAdaptors="1"
    numOfCores="8"
    numOfCpus="2"
    numOfEthHostIfs="2"
    numOfFcHostIfs="0"
    numOfThreads="16"
    operPower="off"
    operQualifier=""
    operState="unassociated"
    operability="operable"
    originalUuid="e3516842-d0a4-11dd-baad-000bab01bfd6"
    presence="equipped"
    revision="0"
    serial="QCI12520024"
    slotId="1"
```

```

totalMemory="10240"
uuid="e3516842-d0a4-11dd-baad-000bab01bfd6"
vendor="Cisco Systems Inc"/>
</outConfig>
</configResolveParent>

```

## configScope

次に、管理対象オブジェクトとその設定に関する詳細を返す例を示します。

### 要求構文

```

<xs:element name="configScope" type="configScope" substitutionGroup="externalMethod"/>
  <xs:complexType name="configScope" mixed="true">
    <xs:all>
      <xs:element name="inFilter" type="filterFilter" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="inClass" type="namingClassId"/>
    <xs:attribute name="inHierarchical">
      <xs:simpleType>
        <xs:union memberTypes="xs:boolean">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="no"/>
              <xs:enumeration value="yes"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:union>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="inRecursive">
      <xs:simpleType>
        <xs:union memberTypes="xs:boolean">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="no"/>
              <xs:enumeration value="yes"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:union>
      </xs:simpleType>
    </xs:attribute>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="dn" type="referenceObject"/>
  </xs:complexType>

```

### 応答構文

```

<xs:element name="configScope" type="configScope" substitutionGroup="externalMethod"/>
  <xs:complexType name="configScope" mixed="true"> <xs:all>
    <xs:element name="outConfigs" type="configSet" minOccurs="0"/>
  </xs:all>
  <xs:attribute name="cookie" type="xs:string"/>
  <xs:attribute name="response" type="YesOrNo"/>
  <xs:attribute name="errorCode" type="xs:unsignedInt"/>
  <xs:attribute name="errorDescr" type="xs:string"/>
  <xs:attribute name="invocationResult" type="xs:string"/>

```

```
<xs:attribute name="dn" type="referenceObject"/>
</xs:complexType>
```

## 例

### 要求

```
<configScope
  dn="org-root"
  cookie="<real_cookie>"
  inClass="orgOrgRes"
  inHierarchical="false"
  inRecursive="false">
  <inFilter>
  </inFilter>
</configScope>
```

### 応答

```
<configScope dn="org-root"
  cookie="<real_cookie>"
  commCookie="2/15/0/2a53"
  srcExtSys="10.193.33.120"
  destExtSys="10.193.33.120"
  srcSvc="sam_extXMLApi"
  destSvc="service-reg_dme"
  response="yes">
  <outConfigs>
    <orgOrgCaps
      dn="org-root/org-caps"
      org="512"
      tenant="64"/>
    <orgOrgCounts
      dn="org-root/org-counter"
      org="36"
      tenant="7"/>
  </outConfigs>
</configScope>
```

## eventSendHeartbeat

次に、現在のセッションがまだアクティブであることを示すイベントを送信する例を示します。

### 要求構文

```
<xs:element name="eventSendHeartbeat" type="eventSendHeartbeat"
  substitutionGroup="externalMethod"/>
  <xs:complexType name="eventSendHeartbeat" mixed="true">
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
  </xs:complexType>
```

### 応答構文

```
<xs:element name="eventSendHeartbeat" type="eventSendHeartbeat"
  substitutionGroup="externalMethod"/>
  <xs:complexType name="eventSendHeartbeat" mixed="true">
    <xs:attribute name="outSystemTime" type="dateTime"/>
```

```

<xs:attribute name="cookie" type="xs:string"/>
<xs:attribute name="response" type="YesOrNo"/>
<xs:attribute name="errorCode" type="xs:unsignedInt"/>
<xs:attribute name="errorDescr" type="xs:string"/>
<xs:attribute name="invocationResult" type="xs:string"/>
</xs:complexType>

```

## 例

### 要求

クライアントアプリケーションが `eventSubscribeApps` または `eventSubscribe` を使用してイベントをサブスクライブする場合、Cisco VNMC は `eventSendHeartbeat` を定期的（デフォルトでは 120 秒）に送信します。

### 応答

```

<eventSendHeartbeat cookie="0/0/0/2a76"
  commCookie=""
  srcExtSys="0.0.0.0"
  destExtSys="0.0.0.0"
  srcSvc=""
  destSvc=""
  response="yes"
  outSystemTime="2010-11-12T20:38:19.630">
</eventSendHeartbeat>

```

## eventSubscribe

次に、アクティビティに対するサブスクライブ要求を送信する例を示します。

### 要求構文

```

<xs:element name="eventSubscribe" type="eventSubscribe"
  substitutionGroup="externalMethod"/>
  <xs:complexType name="eventSubscribe" mixed="true">
    <xs:all>
      <xs:element name="inFilter" type="filterFilter" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
  </xs:complexType>

```

### 応答構文

```

<xs:element name="eventSubscribe" type="eventSubscribe"
  substitutionGroup="externalMethod"/>
  <xs:complexType name="eventSubscribe" mixed="true">
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
  </xs:complexType>

```

## 例

## 要求

```
<eventSubscribe
  cookie="<real_cookie>"
  <inFilter>
</inFilter>
</eventSubscribe>
```

## 応答

NO RESPONSE OR ACKNOWLEDGEMENT.

## eventSubscribeApps

次に、指定されたアプリケーションのアクティビティに対するサブスクライブ要求の例を示します。クライアントアプリケーションは、さまざまなアプリケーションからイベントを受け取るために Cisco VNM システムをサブスクライブできます。eventApplication では、ip はアプリケーション (DME) が実行されている VM の IP アドレスです。クライアントアプリケーションは Cisco VSG からのイベントも受け取るようサブスクライブできます。この場合、ip は Cisco VSG の IP アドレスである必要があります、タイプは管理対象エンドポイントです。

## 要求構文

```
<xs:element name="eventSubscribeApps" type="eventSubscribeApps"
  substitutionGroup="externalMethod"/>
  <xs:complexType name="eventSubscribeApps" mixed="true">
    <xs:all>
      <xs:element name="inAppList" type="configSet" minOccurs="0"/>
      <xs:element name="inFilter" type="filterFilter" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
  </xs:complexType>
```

## 応答構文

```
<xs:element name="eventSubscribeApps" type="eventSubscribeApps"
  substitutionGroup="externalMethod"/>
  <xs:complexType name="eventSubscribeApps" mixed="true">
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
  </xs:complexType>
```

## 例

## 要求

```
<eventSubscribeApps
  cookie="<real_cookie>"
  commCookie=""
  srcExtSys="0.0.0.0"
  destExtSys="0.0.0.0"
```

```

srcSvc=""
destSvc="">
<inAppList>
  <eventApplication
    ip="10.193.33.101"
    type="service-reg"/>
  <eventApplication
    ip="10.193.33.101"
    type="policy-mgr"/>
  <eventApplication
    ip="10.193.33.101"
    type="mgmt-controller"/>
</inAppList>
<inFilter>
</inFilter>
</eventSubscribeApps>

```

**応答**

IF SUCCESSFUL, NO RESPONSE OR ACKNOWLEDGEMENT.

## faultAckFault

次に、障害が記録されたときに確認応答を送信する例を示します。

### 要求構文

```

<xs:element name="faultAckFault" type="faultAckFault" substitutionGroup="externalMethod"/>
  <xs:complexType name="faultAckFault" mixed="true">
    <xs:attribute name="inId" type="xs:unsignedLong"/>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
  </xs:complexType>

```

### 応答構文

```

<xs:element name="faultAckFault" type="faultAckFault" substitutionGroup="externalMethod"/>
  <xs:complexType name="faultAckFault" mixed="true">
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
  </xs:complexType>

```

### 例

**要求**

```

<faultAckFault
  inHierarchical="false"
  cookie="<real_cookie>"
  inId="10120" />

```

**応答**

```

<faultAckFault
  cookie="<real_cookie>"

```



```

commCookie="5/15/0/6c"
srcExtSys="10.193.33.214"
destExtSys="10.193.33.214"
srcSvc="sam_extXMLApi"
destSvc="resource-mgr_dme"
response="yes">
</faultAckFault>

```

## faultAckFaults

次に、複数の障害が記録されたときに確認応答を送信する例を示します。

### 要求構文

```

<xs:element name="faultAckFaults" type="faultAckFaults"
substitutionGroup="externalMethod"/>
  <xs:complexType name="faultAckFaults" mixed="true">
    <xs:all>
      <xs:element name="inIds" type="idSet" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
  </xs:complexType>

```

### 応答構文

```

<xs:element name="faultAckFaults" type="faultAckFaults"
substitutionGroup="externalMethod"/>
  <xs:complexType name="faultAckFaults" mixed="true">
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
  </xs:complexType>

```

## 例

### 要求

```

<faultAckFaults
  cookie="<real_cookie>"
  <inIds>
    <id value="10656"/>
    <id value="10660"/>
  </inIds>
</faultAckFaults>

```

### 応答

```

<faultAckFaults
  cookie="<real_cookie>"
  commCookie="11/15/0/505"
  srcExtSys="10.193.34.70"
  destExtSys="10.193.34.70"
  srcSvc="sam_extXMLApi"
  destSvc="mgmt-controller_dme"
  response="yes">

```

```
</faultAckFaults>
```

## faultResolveFault

次に、障害が解決されたときに応答を送信する例を示します。

### 要求構文

```
<xs:element name="faultResolveFault" type="faultResolveFault"
substitutionGroup="externalMethod"/>
  <xs:complexType name="faultResolveFault" mixed="true">
    <xs:attribute name="inId" type="xs:unsignedLong"/>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
  </xs:complexType>
```

### 応答構文

```
<xs:element name="faultResolveFault" type="faultResolveFault"
substitutionGroup="externalMethod"/>
  <xs:complexType name="faultResolveFault" mixed="true">
    <xs:all>
      <xs:element name="outFault" type="configConfig" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
  </xs:complexType>
```

### 例

#### 要求

```
<faultResolveFault
  inHierarchical="false"
  cookie="<real_cookie>"
  inId="10120" />
```

#### 応答

```
<faultResolveFault
  cookie="<real_cookie>"
  commCookie="5/15/0/6a"
  srcExtSys="10.193.33.214"
  destExtSys="10.193.33.214"
  srcSvc="sam_extXMLApi"
  destSvc="resource-mgr_dme"
  response="yes">
  <outFault>
    <faultInst
      ack="yes"
      cause="empty-pool"
      changeSet=""
      code="F0135"
      created="2010-11-19T11:02:41.568"
      descr="Virtual Security Gateway pool default is empty"
```

```

dn="org-root/fwpool-default/fault-F0135"
highestSeverity="minor"
id="10120"
lastTransition="2010-11-19T11:02:41.568"
lc=""
occur="1"
origSeverity="minor"
prevSeverity="minor"
rule="fw-pool-empty"
severity="minor"
tags=""
type="equipment"/>
</outFault>
</faultResolveFault>

```

## loggingSyncOcms

次に、DME からイベント ID を取得する例を示します。

### 要求構文

```

<xs:element name="loggingSyncOcms" type="loggingSyncOcms"
substitutionGroup="externalMethod"/>
  <xs:complexType name="loggingSyncOcms" mixed="true">
    <xs:attribute name="inFromOrZero" type="xs:unsignedLong"/>
    <xs:attribute name="inToOrZero" type="xs:unsignedLong"/>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
  </xs:complexType>

```

### 応答構文

```

<xs:element name="loggingSyncOcms" type="loggingSyncOcms"
substitutionGroup="externalMethod"/>
  <xs:complexType name="loggingSyncOcms" mixed="true">
    <xs:all>
      <xs:element name="outStimuli" type="MethodSet" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
  </xs:complexType>

```

### 例

#### 要求

```

<loggingSyncOcms
  cookie="<real_cookie>"
  inFromOrZero="0"
  inToOrZero="4567000"/>

```

#### 応答

イベント ID のリスト。

## methodVessel

次に、単一のペイロードに複数の configMoChangeEvent を含むバッチ イベント通知の例を示します。

### 要求構文

```
<xs:element name="methodVessel" type="methodVessel" substitutionGroup="externalMethod"/>
  <xs:complexType name="methodVessel" mixed="true">
    <xs:all>
      <xs:element name="inStimuli" type="MethodSet" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
  </xs:complexType>
```

### 応答構文

```
<xs:element name="methodVessel" type="methodVessel" substitutionGroup="externalMethod"/>
  <xs:complexType name="methodVessel" mixed="true">
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
  </xs:complexType>
```

### 例

#### 要求

```
<methodVessel
  cookie=""
  commCookie=""
  srcExtSys="10.193.77.66"
  destExtSys="10.193.77.66"
  srcSvc="resource-mgr_dme"
  destSvc="sam_extXMLApi">
  <inStimuli>
    <configMoChangeEvent
      cookie=""
      commCookie=""
      srcExtSys="10.193.77.66"
      destExtSys="10.193.77.66"
      srcSvc="resource-mgr_dme"
      destSvc="sam_extXMLApi"
      inEid="434920">
      <inConfig>
        <resstateNormalizedHealthState
          dn="fw/inst-1010/normalized-health-state"
          operState="config"
          status="modified"/>
      </inConfig>
    </configMoChangeEvent>

    <configMoChangeEvent
      cookie=""
      commCookie=""
      srcExtSys="10.193.77.66"
      destExtSys="10.193.77.66"
```

```

srcSvc="resource-mgr_dme"
destSvc="sam_extXMLApi"
inEid="434921">
<inConfig>
  <fwComputeFirewall
    configState="applying"
    dn="org-root/org-tenant_d3337/org-dcl/cfw-vsn-d3340"
    status="modified"/>
  </inConfig>
</configMoChangeEvent>

<configMoChangeEvent
  cookie=""
  commCookie=""
  srcExtSys="10.193.77.66"
  destExtSys="10.193.77.66"
  srcSvc="resource-mgr_dme"
  destSvc="sam_extXMLApi"
  inEid="434922">
<inConfig>
  <faultInst
    dn="org-root/org-tenant_d3337/org-dcl/cfw-vsn-d3340/fault-F0117"
    lc="flapping,soaking-clear"
    status="modified"/>
  </inConfig>
</configMoChangeEvent>
</inStimuli>
</methodVessel>
Event (10:32:2:807):

<methodVessel
  cookie=""
  commCookie=""
  srcExtSys="10.193.77.66"
  destExtSys="10.193.77.66"
  srcSvc="resource-mgr_dme"
  destSvc="sam_extXMLApi">
  <inStimuli>

  <configMoChangeEvent
    cookie=""
    commCookie=""
    srcExtSys="10.193.77.66"
    destExtSys="10.193.77.66"
    srcSvc="resource-mgr_dme"
    destSvc="sam_extXMLApi"
    inEid="434923">
  <inConfig>
    <eventLog
      dn="event-log"
      size="1412"
      status="modified"/>
  </inConfig>
</configMoChangeEvent>

  <configMoChangeEvent
    cookie=""
    commCookie=""
    srcExtSys="10.193.77.66"
    destExtSys="10.193.77.66"
    srcSvc="resource-mgr_dme"
    destSvc="sam_extXMLApi"
    inEid="434924">
  <inConfig>

```

```

    <eventRecord
      affected="fw/inst-1009"
      cause="transition"
      changeSet=""
      code="E4194509"
      created="2010-11-19T18:32:02.622"
      descr="[FSM:STAGE:REMOTE-ERROR]: WRONG STATE:Result: not-applicable Code:
unspecified Message: (sam:dme:FwInstanceAssociate:configPA)"
      dn="event-log/83081"
      id="83081"
      ind="state-transition"
      severity="info"
      status="created"
      trig="special"
      txId="219376"
      user="internal"/>
  </inConfig>
</configMoChangeEvent>
</inStimuli>
</methodVessel>

```

### 応答

```

<methodVessel
  cookie=""
  commCookie=""
  srcExtSys="10.193.77.66"
  destExtSys="10.193.77.66"
  srcSvc="resource-mgr_dme"
  destSvc="sam_extXMLApi">
  <inStimuli>
    <configMoChangeEvent
      cookie=""
      commCookie=""
      srcExtSys="10.193.77.66"
      destExtSys="10.193.77.66"
      srcSvc="resource-mgr_dme"
      destSvc="sam_extXMLApi"
      inEid="434925">
      <inConfig>
        <resstateNormalizedHealthState
          dn="fw/inst-1010/normalized-health-state"
          operState="config-failure"
          status="modified"/>
      </inConfig>
    </configMoChangeEvent>

    <configMoChangeEvent
      cookie=""
      commCookie=""
      srcExtSys="10.193.77.66"
      destExtSys="10.193.77.66"
      srcSvc="resource-mgr_dme"
      destSvc="sam_extXMLApi"
      inEid="434926">
      <inConfig>
        <fwComputeFirewall
          configState="failed-to-apply"
          dn="org-root/org-tenant_d3337/org-dcl/cfw-vsn-d3340"
          status="modified"/>
      </inConfig>
    </configMoChangeEvent>

    <configMoChangeEvent

```

```

cookie=""
commCookie=""
srcExtSys="10.193.77.66"
destExtSys="10.193.77.66"
srcSvc="resource-mgr_dme"
destSvc="sam_extXMLApi"
inEid="434927">
<inConfig>
  <faultInst
    dn="org-root/org-tenant_d3337/org-dc1/cfw-vsn-d3340/fault-F0117"
    lastTransition="2010-11-19T18:32:02.680"
    lc="flapping"
    status="modified"/>
  </inConfig>
</configMoChangeEvent>
</inStimuli>
</methodVessel>

```

## orgResolveElements

この例では、指定された DN 内で、クエリー フィルタを満たす管理対象オブジェクトが取得され、組織（オプションでは子組織）で開始された管理対象オブジェクトが検索されます。

この DN を持つ組織がない場合は、空のマップが返されます。この DN を持つ組織がある場合は、指定されたクラスとフィルタで管理対象オブジェクトが検索されます。

*inHierarchical* が **true** の場合は、一致するすべてのオブジェクトとその子孫が返されます。**false** の場合は、一致するオブジェクトだけが返されます。*inSingleLevel* が **true** の場合は、検索が組織レベルで停止します。**false** の場合は、子組織を含めます。

## 要求構文

```

<xs:element name="orgResolveElements" type="orgResolveElements"
substitutionGroup="externalMethod"/>
  <xs:complexType name="orgResolveElements" mixed="true">
    <xs:all>
      <xs:element name="inFilter" type="filterFilter" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="inClass" type="namingClassId"/>
    <xs:attribute name="inSingleLevel">
      <xs:simpleType>
        <xs:union memberTypes="xs:boolean">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="no"/>
              <xs:enumeration value="yes"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:union>
      </xs:attribute>
    <xs:attribute name="inHierarchical">
      <xs:simpleType>
        <xs:union memberTypes="xs:boolean">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="no"/>
              <xs:enumeration value="yes"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:union>
      </xs:attribute>
    </xs:complexType>
  </xs:element>

```

```

        </xs:union>
    </xs:simpleType>
</xs:attribute>
<xs:attribute name="cookie" type="xs:string"/>
<xs:attribute name="response" type="YesOrNo"/>
<xs:attribute name="dn" type="referenceObject"/>
</xs:complexType>

```

## 応答構文

```

<xs:element name="orgResolveElements" type="orgResolveElements"
substitutionGroup="externalMethod"/>
  <xs:complexType name="orgResolveElements" mixed="true">
    <xs:all>
      <xs:element name="outConfigs" type="configMap" minOccurs="0">
        <xs:unique name="unique_map_key_5">
          <xs:selector xpath="pair"/>
          <xs:field xpath="@key"/>
        </xs:unique>
      </xs:element>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
    <xs:attribute name="dn" type="referenceObject"/>
  </xs:complexType>

```

## 例

### 要求

```

<orgResolveElements
  dn="org-root/org-Cola"
  cookie="<real_cookie>"
  commCookie="7/15/0/19"
  srcExtSys="10.193.33.221"
  destExtSys="10.193.33.221"
  srcSvc="sam_extXMLApi"
  destSvc="policy-mgr_dme"
  inClass="policyPolicySet"
  inSingleLevel="no"
  inHierarchical="no">
  <inFilter>
  </inFilter>
</orgResolveElements>

```

### 応答

```

<orgResolveElements
  dn="org-root/org-Cola"
  cookie="<real_cookie>"
  commCookie="7/15/0/19"
  srcExtSys="10.193.33.221"
  destExtSys="10.193.33.221"
  srcSvc="sam_extXMLApi"
  destSvc="policy-mgr_dme"
  response="yes"
  errorCode="0"
  errorDescr="">
  <outConfigs>

```



```

<pair key="pset-default">
  <policyPolicySet
    adminState="enabled"
    descr="The default Policy Set"
    dn="org-root/pset-default"
    intId="10082"
    name="default"/>
</pair>
<pair key="pset-myPolicySet3">
  <policyPolicySet
    adminState="enabled"
    descr=""
    dn="org-root/org-Cola/pset-myPolicySet3"
    intId="12289"
    name="myPolicySet3"/>
</pair>
<pair key="pset-policySetSanity">
  <policyPolicySet
    adminState="enabled"
    descr=""
    dn="org-root/org-Cola/pset-policySetSanity"
    intId="24627"
    name="policySetSanity"/>
</pair>
<pair key="pset-pci_compliance_f">
  <policyPolicySet
    adminState="enabled"
    descr=""
    dn="org-root/pset-pci_compliance_f"
    intId="24539"
    name="pci_compliance_f"/>
</pair>
<pair key="pset-pci_compliance_h">
  <policyPolicySet
    adminState="enabled"
    descr=""
    dn="org-root/pset-pci_compliance_h"
    intId="24541"
    name="pci_compliance_h"/>
</pair>
</outConfigs>
</orgResolveElements>

```

## orgResolveInScope

次に、システムが、該当する DN を持つ組織と親組織（オプション）をルートまで再帰的に検索する例を示します。組織がない場合は、空のマップが返されます。組織がある場合は、指定されたクラスとフィルタですべてのプールが検索されます。



(注) *inSingleLevel* が *false* の場合は、親組織が、ルートディレクトリまで検索されます。

### 要求構文

```

<xs:element name="orgResolveInScope" type="orgResolveInScope"
  substitutionGroup="externalMethod"/>
  <xs:complexType name="orgResolveInScope" mixed="true">
    <xs:all>
      <xs:element name="inFilter" type="filterFilter" minOccurs="0"/>

```

```

</xs:all>
<xs:attribute name="inClass" type="namingClassId"/>
<xs:attribute name="inSingleLevel">
  <xs:simpleType>
    <xs:union memberTypes="xs:boolean">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="no"/>
          <xs:enumeration value="yes"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:union>
  </xs:simpleType>
</xs:attribute>
<xs:attribute name="inHierarchical">
  <xs:simpleType>
    <xs:union memberTypes="xs:boolean">
      <xs:simpleType>
        <xs:restriction base="xs:string">
          <xs:enumeration value="no"/>
          <xs:enumeration value="yes"/>
        </xs:restriction>
      </xs:simpleType>
    </xs:union>
  </xs:simpleType>
</xs:attribute>
<xs:attribute name="cookie" type="xs:string"/>
<xs:attribute name="response" type="YesOrNo"/>
<xs:attribute name="dn" type="referenceObject"/>
</xs:complexType>

```

## 応答構文

```

<xs:element name="orgResolveInScope" type="orgResolveInScope"
substitutionGroup="externalMethod"/>
<xs:complexType name="orgResolveInScope" mixed="true">
  <xs:all>
    <xs:element name="outConfigs" type="configMap" minOccurs="0">
      <xs:unique name="unique_map_key_6">
        <xs:selector xpath="pair"/>
        <xs:field xpath="@key"/>
      </xs:unique>
    </xs:element>
  </xs:all>
  <xs:attribute name="cookie" type="xs:string"/>
  <xs:attribute name="response" type="YesOrNo"/>
  <xs:attribute name="errorCode" type="xs:unsignedInt"/>
  <xs:attribute name="errorDescr" type="xs:string"/>
  <xs:attribute name="invocationResult" type="xs:string"/>
  <xs:attribute name="dn" type="referenceObject"/>
</xs:complexType>

```

## 例

### 要求

```

<orgResolveInScope
  cookie="<real_cookie>"
  dn="org-root/org-Cola"
  inClass="policyVirtualNetworkServiceProfile"
  inHierarchical="true"

```

```

InSingleLevel="false" >
<inFilter>
  <eq class="policyVirtualNetworkServiceProfile"
    property="name"
    value="spsanity" />
</inFilter>
</orgResolveInScope>

```

### 応答

```

<orgResolveInScope
  dn="org-root/org-Cola"
  cookie="<real_cookie>"
  commCookie="7/15/0/1c35"
  srcExtSys="10.193.34.70"
  destExtSys="10.193.34.70"
  srcSvc="sam_extXMLApi"
  destSvc="policy-mgr_dme"
  response="yes">
<outConfigs>
  <pair key="vnsp-spsanity">
    <policyVirtualNetworkServiceProfile
      childAction="deleteNonPresent"
      descr=""
      dn="org-root/org-Cola/vnsp-spsanity"
      intId="82018"
      name="spsanity"
      policySetNameRef="policySetSanity"
      vnspId="41">
    <policyVnspAVPair
      childAction="deleteNonPresent"
      descr=""
      id="1"
      intId="82019"
      name=""
      rn="vnsp-avp-1">
    <policyAttributeValue
      childAction="deleteNonPresent"
      id="1"
      rn="attr-val1"
      value="DEV"/>
    <policyAttributeDesignator
      attrName="dept"
      childAction="deleteNonPresent"
      rn="attr-ref"/>
    </policyVnspAVPair>
  </policyVirtualNetworkServiceProfile>
</pair>
</outConfigs>
</orgResolveInScope>

```

## orgResolveLogicalParents

この例は、システムが、指定された DN を持つ論理的な親をルート ディレクトリまでどのように検索するかを示しています。

### 要求構文

```

<xs:element name="orgResolveLogicalParents" type="orgResolveLogicalParents"
  substitutionGroup="externalMethod"/>

```

```

<xs:complexType name="orgResolveLogicalParents" mixed="true">
  <xs:attribute name="inSingleLevel">
    <xs:simpleType>
      <xs:union memberTypes="xs:boolean">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="no"/>
            <xs:enumeration value="yes"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:union>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="inHierarchical">
    <xs:simpleType>
      <xs:union memberTypes="xs:boolean">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="no"/>
            <xs:enumeration value="yes"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:union>
    </xs:simpleType>
  </xs:attribute>
  <xs:attribute name="cookie" type="xs:string"/>
  <xs:attribute name="response" type="YesOrNo"/>
  <xs:attribute name="dn" type="referenceObject"/>
</xs:complexType>

```

## 応答構文

```

<xs:element name="orgResolveLogicalParents" type="orgResolveLogicalParents"
substitutionGroup="externalMethod"/>
  <xs:complexType name="orgResolveLogicalParents" mixed="true">
    <xs:all>
      <xs:element name="outConfigs" type="configMap" minOccurs="0">
        <xs:unique name="unique_map_key_7">
          <xs:selector xpath="pair"/>
          <xs:field xpath="@key"/>
        </xs:unique>
      </xs:element>
    </xs:all>
    <xs:attribute name="cookie" type="xs:string"/>
    <xs:attribute name="response" type="YesOrNo"/>
    <xs:attribute name="errorCode" type="xs:unsignedInt"/>
    <xs:attribute name="errorDescr" type="xs:string"/>
    <xs:attribute name="invocationResult" type="xs:string"/>
    <xs:attribute name="dn" type="referenceObject"/>
  </xs:complexType>

```

## 例

### 要求

```

<orgResolveLogicalParents
  dn="org-root/org-Cisco/org-HR/zone-clients"
  cookie="<real_cookie>"
  commCookie="7/15/0/12"
  srcExtSys="10.193.33.221"
  destExtSys="10.193.33.221"

```

```

srcSvc="sam_extXMLApi"
destSvc="policy-mgr_dme"
inSingleLevel="no"
inHierarchical="no">
</orgResolveLogicalParents>

```

### 応答

```

<orgResolveLogicalParents
  dn="org-root/org-Cisco/org-HR/zone-clients"
  cookie="<real_cookie>"
  commCookie="7/15/0/12"
  srcExtSys="10.193.33.221"
  destExtSys="10.193.33.221"
  srcSvc="sam_extXMLApi"
  destSvc="policy-mgr_dme"
  response="yes"
  errorCode="0"
  errorDescr="">
<outConfigs>
  <pair key="org-root/org-Cisco/zone-clients">
    <policyZone
      descr=""
      dn="org-root/org-Cisco/zone-clients"
      intId="24841"
      name="clients"/>
    </pair>
  </outConfigs>
</orgResolveLogicalParents>

```

## policyEstimateImpact

この例では、管理対象オブジェクト (MO) が取得され、この変更により影響を受ける MO のリストが返されます。これは、ポリシー変更 (作成、変更、または削除) の影響度を推定するために使用します。

MO に影響を与える可能性があるポリシー変更は次のとおりです。

- 名前参照が変更されています (つまり、解決された名前が変更されます)。
- 削除時に、このオブジェクトを参照する別のオブジェクト参照を、ツリーの上にある同じ名前の異なるオブジェクト (または、デフォルトのオブジェクト) に解決できるようになりました。
- 作成時に、オブジェクト参照は新しく作成されたオブジェクトを参照できます。応答では、**outDeleteAllowed** パラメータは、指定されたオブジェクトを削除できるかどうかを示します。削除できる場合は、このオブジェクトを参照する名前参照はツリーの上部の別のオブジェクトを参照します。
- 指定されたオブジェクトがオブジェクト参照の参照されたオブジェクトであり、これをオーバーライドする他のオブジェクトがツリーにない場合は、指定されたオブジェクトを削除できません。

### 要求構文

```

<!--
  - Method: policy:EstimateImpact
  -->
  <xs:element name="policyEstimateImpact" type="policyEstimateImpact"
substitutionGroup="externalMethod"/>

  <xs:complexType name="policyEstimateImpact" mixed="true">

```

```

<xs:all>
  <xs:element name="inConfig" type="configConfig" minOccurs="0"/>
</xs:all>

<xs:attribute name="cookie" type="xs:string"/>
<xs:attribute name="response" type="YesOrNo"/>

</xs:complexType>

```

## 応答構文

```

<!--
  - Method: policyEstimateImpact
  -->
<xs:element name="policyEstimateImpact" type="policyEstimateImpact"
substitutionGroup="externalMethod"/>

<xs:complexType name="policyEstimateImpact" mixed="true">
  <xs:all>
    <xs:element name="outImpactedMoSet" type="configMap" minOccurs="0">
      <xs:unique name="unique_map_key_8">
        <xs:selector xpath="pair"/>
        <xs:field xpath="@key"/>
      </xs:unique>
    </xs:element>
  </xs:all>

  <xs:attribute name="outDeleteAllowed">
    <xs:simpleType>
      <xs:union memberTypes="xs:boolean">
        <xs:simpleType>
          <xs:restriction base="xs:string">
            <xs:enumeration value="no"/>
            <xs:enumeration value="yes"/>
          </xs:restriction>
        </xs:simpleType>
      </xs:union>
    </xs:simpleType>
  </xs:attribute>

  <xs:attribute name="cookie" type="xs:string"/>
  <xs:attribute name="response" type="YesOrNo"/>
  <xs:attribute name="errorCode" type="xs:unsignedInt"/>
  <xs:attribute name="errorDescr" type="xs:string"/>
  <xs:attribute name="invocationResult" type="xs:string"/>

</xs:complexType>

```

## 例

### 要求

```

<policyEstimateImpact
  cookie="1309385844/5d9899e6-5b5b-4ab0-b12c-cb6e50b5a7db"
  commCookie="7/12/0/1d18"
  srcExtSys="10.193.33.221"
  destExtSys="10.193.33.221"
  srcSvc="sam_extXMLApi"
  destSvc="policy-mgr_dme">
  <inConfig>

```

```

    <policyRuleBasedPolicy
      dn="org-root/org-tenant0/pset-myPolicySet1"
      status="deleted,modified"/>
  </inConfig>
</policyEstimateImpact>

```

### 応答

```

<policyEstimateImpact
  cookie="1309385844/5d9899e6-5b5b-4ab0-b12c-cb6e50b5a7db"
  commCookie="7/12/0/1d18"
  srcExtSys="10.193.33.221"
  destExtSys="10.193.33.221"
  srcSvc="sam_extXMLApi"
  destSvc="policy-mgr_dme"
  response="yes"
  errorCode="0"
  errorDescr=""
  outDeleteAllowed="yes">
  <outImpactedMoSet>
    <pair key="org-root/org-tenant0/vnsp-VNSP1">
      <policyVirtualNetworkServiceProfile
        descr=""
        dn="org-root/org-tenant0/vnsp-VNSP1"
        intId="40930"
        name="VNSP1"
        policySetNameRef="myPolicySet1"
        vnspId="17"/>
    </pair>
  </outImpactedMoSet>
</policyEstimateImpact>

```

## poolResolveInScope

次に、システムが、該当する DN があるプールと親プール（オプション）をルートまで再帰的に検索する例を示します。プールがない場合は、空のマップが返されます。プールがある場合は、指定されたクラスとフィルタですべてのプールが検索されます。



(注) *inSingleLevel* が *false* の場合は、親プールがルート ディレクトリまで検索されます。

### 要求構文

```

<xs:element name="poolResolveInScope" type="poolResolveInScope"
  substitutionGroup="externalMethod"/>
  <xs:complexType name="poolResolveInScope" mixed="true">
    <xs:all>
      <xs:element name="inFilter" type="filterFilter" minOccurs="0"/>
    </xs:all>
    <xs:attribute name="inClass" type="namingClassId"/>
    <xs:attribute name="inSingleLevel">
      <xs:simpleType>
        <xs:union memberTypes="xs:boolean">
          <xs:simpleType>
            <xs:restriction base="xs:string">
              <xs:enumeration value="no"/>
              <xs:enumeration value="yes"/>
            </xs:restriction>
          </xs:simpleType>
        </xs:union>
      </xs:simpleType>
    </xs:attribute>
  </xs:complexType>

```

```

        </xs:union>
    </xs:simpleType>
</xs:attribute>
<xs:attribute name="inHierarchical">
    <xs:simpleType>
        <xs:union memberTypes="xs:boolean">
            <xs:simpleType>
                <xs:restriction base="xs:string">
                    <xs:enumeration value="no"/>
                    <xs:enumeration value="yes"/>
                </xs:restriction>
            </xs:simpleType>
        </xs:union>
    </xs:simpleType>
</xs:attribute>
<xs:attribute name="cookie" type="xs:string"/>
<xs:attribute name="response" type="YesOrNo"/>
<xs:attribute name="dn" type="referenceObject"/>
</xs:complexType>

```

## 応答構文

```

<xs:element name="poolResolveInScope" type="poolResolveInScope"
substitutionGroup="externalMethod"/>
    <xs:complexType name="poolResolveInScope" mixed="true">
        <xs:all>
            <xs:element name="outConfigs" type="configMap" minOccurs="0">
                <xs:unique name="unique_map_key_9">
                    <xs:selector xpath="pair"/>
                    <xs:field xpath="@key"/>
                </xs:unique>
            </xs:element>
        </xs:all>
        <xs:attribute name="cookie" type="xs:string"/>
        <xs:attribute name="response" type="YesOrNo"/>
        <xs:attribute name="errorCode" type="xs:unsignedInt"/>
        <xs:attribute name="errorDescr" type="xs:string"/>
        <xs:attribute name="invocationResult" type="xs:string"/>
        <xs:attribute name="dn" type="referenceObject"/>
    </xs:complexType>

```

## 例

### 要求

```

<poolResolveInScope
  dn="org-root/org-Cisco"
  cookie="<real_cookie>"
  class=fwPool />

```

### 応答

```

<poolResolveInScope
  dn="org-root/org-Cisco"
  cookie="<real_cookie>"
  commCookie="5/15/0/5bf"
  srcExtSys="10.193.33.221"
  destExtSys="10.193.33.221"
  srcSvc="sam_extXMLApi"
  destSvc="resource-mgr_dme"
  response="yes">
  <outConfigs>

```



```
<pair key="fwpool-default">
  <fwPool
    assigned="0"
    descr="Default Pool of Virtual Security Gateway resources"
    dn="org-root/fwpool-default"
    fltAggr="65536"
    id="1"
    intId="10065"
    name="default"
    size="0"/>
  </pair>
</poolResolveInScope>
```

