

CA 性能管理 Data Aggregator

使用 REST Web 服务管理指南

2.4



本文档包括内嵌帮助系统和以电子形式分发的材料（以下简称“文档”），其仅供参考，CA 随时可对其进行更改或撤销。

未经 CA 事先书面同意，不得擅自复制、转让、翻印、透露、修改或转录本文档的全部或部分内容。本文档属于 CA 的机密和专有信息，不得擅自透露，或除以下协议中所允许的用途，不得用于其他任何用途：(i) 您与 CA 之间关于使用与本文档相关的 CA 软件的单独协议；或者 (ii) 您与 CA 之间单独的保密协议。

尽管有上述规定，但如果您为本文档中所指的软件产品的授权用户，则您可打印或提供合理数量的本文档副本，供您及您的雇员内部用于与该软件相关的用途，前提是所有 CA 版权声明和标识必须附在每一份副本上。

打印或提供本文档副本的权利仅限于此类软件所适用的许可协议的有效期内。如果该许可因任何原因而终止，您应负责向 CA 书面证明已将本文档的所有副本和部分副本已退还给 CA 或被销毁。

在所适用的法律允许的范围内，CA 按照“现状”提供本文档，不附带任何保证，包括但不限于商品适销性、适用于特定目的或不侵权的默示保证。CA 在任何情况下对您或其他第三方由于使用本文档所造成的直接或间接的损失或损害都不负任何责任，包括但不限于利润损失、投资受损、业务中断、信誉损失或数据丢失，即使 CA 已经被提前明确告知这种损失或损害的可能性。

本文档中涉及的任何软件产品的使用均应遵照有关许可协议的规定且根据本声明中的条款不得以任何方式修改此许可协议。

本文档由 CA 制作。

仅提供“有限权利”。美国政府使用、复制或透露本系统受 FAR Sections 12.212、52.227-14 和 52.227-19(c)(1) - (2) 以及 DFARS Section 252.227-7014(b)(3) 的相关条款或其后续条款的限制。

版权所有 © 2014 CA。保留所有权利。 此处涉及的所有商标、商品名称、服务标识和徽标均归其各自公司所有。

CA Technologies 产品引用

本文档引用以下 CA Technologies 产品：

- CA 性能管理 Data Aggregator (Data Aggregator)
- CA 性能管理 Data Collector (Data Collector)
- CA Performance Center

联系技术支持

要获取在线技术帮助以及办公地址、主要服务时间和电话号码的完整列表，请联系技术支持：<http://www.ca.com/worldwide>。

目录

第 1 章：简介和基本操作	7
关于 REST Web 服务.....	7
使用基本操作.....	8
第 2 章：关于 OpenAPI	8
管理 OpenAPI 凭据.....	8
访问 OpenAPI.....	9
关于 OpenAPI 查询.....	9
OpenAPI 控件	10
创建 OpenAPI 查询.....	11
OpenAPI 查询故障排除.....	13
度量标准值不显示在表中.....	13
查询生成空表.....	13
第 3 章：常规 REST Web 服务	15
常规 REST Web 服务.....	15
管理常规 REST Web 服务的关系.....	15
第 4 章：数据驱动的 REST Web 服务	17
数据驱动的 REST Web 服务.....	17
查看数据驱动的 REST Web 服务的 XSD 架构.....	17
筛选属性.....	18
管理数据驱动的 REST Web 服务的基本关系.....	21
第 5 章：使用 Web 服务	23
将数据驱动的范围限制在承租方域.....	23
没有性能影响的轮询敏感和关键设备	23
更改数据保持期间.....	27
排定数据清除.....	29
更改计算同一天同一小时的基准平均值的时间	31
执行累加处理和基准计算时修改	32
更改受监视设备上的主 IP 地址	34
如何创建自定义度量标准系列.....	35
查看度量标准系列.....	36
下载度量标准系列架构和示例文件.....	37

创建自定义度量标准系列 XML 文件	37
在测试环境中确认您的自定义度量标准系列结果	63
导入自定义度量标准系列 XML 文件	64
自动移除已报废的组件	64

第 6 章：故障排除 69

查找有关错误消息的详细信息	69
故障排除：度量标准系列不完整	69
故障排除：度量标准系列不受支持	70

第 1 章：简介和基本操作

此部分包含以下主题：

[关于 REST Web 服务](#) (p. 7)

[使用基本操作](#) (p. 8)

关于 REST Web 服务

Data Aggregator 使用 REST Web 服务对管理操作进行管理，如检索数据或管理配置文件与承租方或组之间的关系。这些 REST Web 服务支持 CA Performance Center 用户界面。或者，您可以使用 API 执行操作。作为管理员，您可以使用任何 REST 客户端工具来执行请求，也可以使用 HTTP 工具来发送请求并获得响应。

有三种类型的 REST Web 服务：

OpenAPI

用作公共 API，以供出于自身目的而想要访问 Data Aggregator 配置信息和度量标准数据的客户使用。OpenAPI 的目的在于，即使内部数据存储结构有所更改，它也能在日后保持不变。

重要信息：OpenAPI 使用 QueryBuilder 接口。该接口的这一版本处于测试阶段，因此不建议在生产环境中使用该接口。在可扩展性和授权选项中存在限制（单个用户和密码）。

数据驱动

被 CA Performance Center UI 用来读取和修改 Data Aggregator 配置信息，如“监视配置文件”和“组”。

常规

被 CA Performance Center UI 用来管理度量标准系列和受限的 SNMP 供应商认证支持。常规 REST Web 服务是自我筛选的，不在 URL 中使用参数来管理关系。

使用基本操作

端点是各种类型的项，例如组、监视配置文件、承租方和设备认证。通过下列请求方法或基本操作来使用特定端点以返回结果列表，或者创建、更新或删除项。

注意：单个度量标准系列和 SNMP 供应商认证项使用名称而非 ID 进行指定。

重要说明！使用 Data Aggregator REST Web 服务执行操作时，请将上下文类型文件设置为 **application/xml**。

基本操作包括：

GET http://.../endpoint

返回指定类型的所有项的列表。getlist.xsd 架构定义返回数据的格式。

GET http://.../endpoint/[id | name]

为具有指定 ID 或认证名称的单个项返回详细信息。XSD 架构定义返回数据的格式。

POST http://.../endpoint

使用指定的方面创建指定类型的对象。XSD 架构定义返回数据的格式。

PUT http://.../endpoint/[id | name]

更新指定项的属性。update.xsd 架构定义格式和预期的字段。

DELETE http://.../endpoint/[id | name]

删除使用 ID 或认证名称指定的项。

第 2 章： 关于 OpenAPI

管理 OpenAPI 凭据

默认情况下会禁用对 OpenAPI 的访问。要启用访问，请设置 OpenAPI 的用户名和密码。用户名和密码将存储在外部文件中，且无法使用 OpenAPI 接口进行更改。

请执行以下步骤：

1. 打开以下文件：
`/opt/IMDataAggregator/apache-karaf-2.3.0/etc/com.ca.im.odata.auth.OpenAPIAuthenticationFilter.cfg`
2. 更改用户名和密码。
删除最后一个字符之后的所有空格。如果存在空格，则应将其输入以便密码生效。
3. 保存文件。
开始使用新的用户名和密码，而无需重新启动 Data Aggregator。Data Aggregator 可立即识别新的凭据。

访问 OpenAPI

在系统管理员设置用户名和密码之前，访问 OpenAPI 会一直受到限制。一旦使用提供的用户名和密码登录，您不需要在相同的会话中再次提供凭据。在您清除浏览器 cookie 或重新启动计算机之前，该会话不会超时。

OpenAPI 查询生成器的基本 URL 如下：
`http://hostname:port/odataquery`

hostname:port

指定 Data Aggregator 主机名和端口号。

默认端口：8581

关于 OpenAPI 查询

OpenAPI 可用于导出配置和轮询数据以便进行复查。OpenAPI 让您运行用于创建自定义查询 URL 的查询，您可以随时访问这些 URL 以便查看自定义的数据。当内部数据存储结构更改时，OpenAPI 将保持不变。

与数据驱动的服务和常规 Web 服务不同，OpenAPI 为公共 API 并可供客户使用。OpenAPI 采用 OData 2.0 行业标准。此标准的相关文档可在 OData 2.0 标准 <http://www.odata.org/documentation/odata-version-2-0/> 网页上找到。

注意：请参考元数据以查看架构 XML 说明。此说明提供有关 OpenAPI 的详细信息。要查看元数据，请导航到以下 URL：
`http://<您的 DA>:8581/odata/api/$metadata`

如果您将新的度量标准系列添加到 Data Aggregator 中，请重新启动 Data Aggregator 以更新 OpenAPI 中的 OData 架构。要重新启动 Data Aggregator，请运行以下命令：
`/etc/init.d/dadaemon restart`

重要信息：该 QueryBuilder 接口的这一版本处于测试阶段，因此不建议在生产环境中使用该接口。在可扩展性和授权选项中存在限制（单个用户和密码）。

OpenAPI 控件

OpenAPI 查询生成器使用若干类型的控件。使用这些控件来创建并导出查询 URL：

标记

标记是构成了完整查询的 OpenAPI 查询语法的逻辑要素。单击“指定查询”字段时，会出现标记。通过标记可以选择要包含在查询中的实体类型、列、度量标准系列、度量标准或时间范围。选定每个标记时，会相应地更新查询 URL。

示例：

列、筛选、时间范围、度量标准系列、组件

筛选控件

让您可以创建基于逻辑表达式的自定义筛选。通过筛选控件可以使用 AND 和 OR 操作来创建表达式。您可以管理单个规则或一组规则，还可以选择要对各个规则应用的操作（AND 或 OR）。

示例：

列筛选、度量标准筛选

格式控件

让您可以设置数据的格式样式，包括显示的列和输出的格式。选定每个格式控件时，会相应地更新查询 URL。选择“选择性地指定格式”。

示例：

排序、结果限制、结果格式

可以使用以下两种类型的数据表：

带有额外度量标准的 HTML 表

让您可以查看聚合的最小、最大和平均度量标准。单击放大镜图标可查看相应项的详细度量标准数据。

带有导出的 HTML 表

度量标准列显示为空，但是您可以导出配置信息。

按钮

让您可以在 UI 上执行操作，如将查询 URL 复制到剪贴板。

示例：

复制 ()、添加规则、添加组、重置

预定义的查询筛选（可选）

让您可以使用即用型选择项，作为可以对其进行自定义以创建 OpenAPI 查询的模板。这些模板为构建您自己的查询提供了起点。选择预定义的查询筛选，然后选择更多的标记和筛选控件来创建自定义查询。添加模板之后，修改筛选控件以适合您的需求。

示例：

特定组中的路由器

创建 OpenAPI 查询

OpenAPI 查询生成器让您可以创建查询 URL。对于查询中的每个参数，OpenAPI 均会将相应的参数添加到查询 URL 中。参数可用性取决于组件、设备、路由器或是要查询的其他项目。

请执行以下步骤：

1. 单击“指定查询”字段开始查询。

此时显示标记列表。

2. 选择标记或处方。

此时在“指定查询”字段中显示标记或处方。

Web 浏览器的地址栏会进行更新以显示您在“指定查询”字段中所做的更改。如果您或其他用户想在稍后继续编辑此查询，请复制并保存此 URL 以留作记录。


3. 使用可用的标记完成查询。

4. 单击“格式”和“有效负载”选项以自定义数据的格式设置。

选择排序方式、要返回的项数和数据的格式。

5. （可选）单击“运行”按钮。

OpenAPI 会运行查询，并以您选定的格式显示结果。

6. 单击“复制”() 按钮。

查询 URL 被复制到剪贴板，并可粘贴到 Web 浏览器或 REST 工具中。

所有 OpenAPI 查询的基本 URL 如下：

`http://hostname:port/odata/api/`

重要信息：返回大量结果集的查询会对您的系统产生负面影响。建议对您的查询进行细化以便仅返回与您的需求相关的结果。

OpenAPI 查询故障排除

度量标准值不显示在表中

症状:

我运行查询以便生成具有度量标准值的表。但是，当我打开表时，结果中却没有显示任何度量标准值。

解决方案:

OpenAPI 当前不支持将某个特殊时间范围的多个数据样本聚合为单个值。

要预览您的查询结果，请选择适当的输出格式，如：

- 带有额外度量标准的 HTML 表
- JSON
- XML

查询生成空表

症状:

运行查询之后，生成的查询 URL 显示空表。

解决方案:

当选择以表格式显示度量标准列时，请选择至少一个配置列，如项名称。另一个解决方案是选择除“带有导出的 HTML 表”之外的格式。

第 3 章： 常规 REST Web 服务

此部分包含以下主题：

[常规 REST Web 服务](#) (p. 15)

[管理常规 REST Web 服务的关系](#) (p. 15)

常规 REST Web 服务

常规 REST Web 服务用于管理度量标准系列和有限的 SNMP 供应商认证支持。常规 REST Web 服务是自我筛选的，不在 URL 中使用参数来管理关系。

常规 Web 服务的基本 URL 为：

`http://hostname:port/genericWS`

hostname:port

指定 Data Aggregator 主机名和端口号。

默认端口：8581

在浏览器中访问以下 URL，以获得有关常规 Web 服务的详细信息，包括 XSD、URI、支持的 HTTP 方法、属性和关系。

- 显示有关系统中每个用户面对的常规 Web 服务的详细信息：

`http://hostname:port/genericWS/`

- 显示有关指定端点的详细信息：

`http://hostname:port/genericWS/endpoint/documentation`

管理常规 REST Web 服务的关系

常规 REST Web 服务不在 URL 中使用参数来管理关系。相反，常规 REST Web 服务仅依靠基本操作来管理关系。端点对自己进行筛选以显示信息。这些方法可用于管理度量标准系列和 SNMP 供应商认证之间的关系。

使用下列方法来查看、创建和删除常规 REST Web 服务的关系：

GET `http://hostname:port/genericWS/endpoint/name/endpoint`

PUT `http://hostname:port/genericWS/endpoint/name/endpoint`

DELETE `http://hostname:port/genericWS/endpoint/name/endpoint`

参数

hostname:port

指定 Data Aggregator 主机名和端口号。

默认端口：8581

endpoint/name

指定度量标准系列或 SNMP 供应商认证端点的名称。

示例：列出与 SNMP 供应商认证有关的度量标准系列

要返回与指定 SNMP 供应商认证有关的度量标准系列的列表，请使用以下 URL 语法：

GET `http://hostname:port/genericWS/certifications/snmp/name/metricfamilies`

第 4 章： 数据驱动的 REST Web 服务

此部分包含以下主题：

[数据驱动的 REST Web 服务](#) (p. 17)

[查看数据驱动的 REST Web 服务的 XSD 架构](#) (p. 17)

[筛选属性](#) (p. 18)

[管理数据驱动的 REST Web 服务的基本关系](#) (p. 21)

数据驱动的 REST Web 服务

数据驱动的 REST Web 服务可用于大多数 Data Aggregator Web 服务，如监视配置文件和组。

数据驱动的 Web 服务的基本 Web 服务 URL 为：

`http://hostname:port/rest`

hostname:port

指定 Data Aggregator 主机名和端口号。

默认端口：8581

在浏览器中访问以下 URL，以获得有关端点和数据驱动的 Web 服务的详细信息，包括 XSD、URI、支持的 HTTP 方法、属性和关系。

- 显示有关系统中每个用户面对的数据驱动的 Web 服务的详细信息：

`http://hostname:port/rest/`

- 显示有关指定端点的详细信息：

`http://hostname:port/rest/endpoint/documentation`

查看数据驱动的 REST Web 服务的 XSD 架构

在执行 HTTP 请求之前执行以下下列步骤：

- 验证端点的 XML 架构定义 (XSD)。
- 检查服务提供的返回或上载 XML 的格式。

XML 文档中包含的每项内容都必须遵守端点的说明。

要获取端点的 XSD，请将以下路径与数据驱动的 Web 服务 URL 结合使用：

`http://hostname:port/rest/endpoint/XSD/operation.xsd`

操作

指定要执行的操作的类型。

总的来说支持下列操作，但其中有些操作并非每个端点都支持：

get

获得单项 get 的 XSD。

getlist

获得端点项列表的 XSD。

filterselect

允许使用 GET 通道指定高级筛选条件和返回 XML 格式。

创建

获得试图创建时任何输入 XML 必须匹配的 XSD。

update

获得试图更新时任何输入 XML 必须匹配的 XSD。

注意：如果某一操作不受支持，Web 服务失败并返回“403 禁止访问”消息。

自动生成的 create、update、get、getlist 和 filterselect 的 XSD 文件包含说明属性和度量标准系列目的的标记。

筛选属性

您可以筛选数据驱动的 Web 服务的 XSD 架构中所列的属性，如项名称、说明和其他此类属性。例如，按所包含的度量标准系列筛选监视配置文件。您可以使用该信息确定是将度量标准系列添加到监视配置文件还是从中删除。

请执行以下步骤：

1. 在 Web 浏览器中输入以下 URL：

`http://hostname:port/rest/`

此时将显示一个列表，其中显示数据驱动的可用 Web 服务。

2. 单击一个 Web 服务。

将显示该 Web 服务的文档页面。

3. 单击“筛选后的 get 列表”方法下的 URL。
将打开 XSD 架构。
4. 查找具有 substitutionGroup="AttributeFilterTypeSubstitution" 的元素，如下例所示。使用此信息可确定要筛选的属性。

```
- <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:include schemaLocation="basefilterselect.xsd" />
  <xs:element name="Item.CreateTime" substitutionGroup="AttributeFilterTypeSubstitution" />
  <xs:element name="Item.Name" substitutionGroup="AttributeFilterTypeSubstitution" />
  <xs:element name="Item.Description" substitutionGroup="AttributeFilterTypeSubstitution" />
  <xs:element name="MonitoringProfile.PollGroupIDs" substitutionGroup="AttributeFilterTypeSubstitution" />
  <xs:element name="MonitoringProfile.PollRate" substitutionGroup="AttributeFilterTypeSubstitution" />
  <xs:element name="MonitoringProfile.Description" substitutionGroup="AttributeFilterTypeSubstitution" />
  <xs:element name="MonitoringProfile.FacetTypes" substitutionGroup="AttributeFilterTypeSubstitution" />
</xs:schema>
```

5. 打开 REST 客户端编辑器或 HTTP 工具（用于发送请求并获得响应），并将上下文类型设置为 **application/xml**。
6. 输入以下筛选条件：

- URL: `http://hostname:port/rest/endpoint/filtered/`

- HTTP 方法 = POST

此方法必须定义筛选条件。

- “HTTP 请求”窗格中“正文”选项卡上的基本筛选选择条件：

```
<FilterSelect xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="filter.xsd">
  <Filter>
    <elementName
type="CONTAINS">filter-criteria</elementName>
  </Filter>
</FilterSelect>
```

filter-criteria

指定属性的实际值。

elementName

指定要筛选的元素名称（属性）。

注意：您也可以仅指定选择条件，如轮询比例。有关详细信息，请参阅以下示例。

将在“HTTP 响应”窗格的“正文”选项卡中返回结果。

示例：返回包含使用筛选和选择条件的度量标准系列的监视配置文件列表。

输入以下 URL，以便返回包含使用轮询比例作为选择条件的度量标准系列的监视配置文件：

- 方法和 URL：

POST `http://hostname:port/rest/monitoringprofiles/filtered/`

- 正文：

```
<FilterSelect xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:noNamespaceSchemaLocation="filter.xsd">
  <Filter>
    <MonitoringProfile.FacetTypes
type="CONTAINS">{http://im.ca.com/normalizer}NormalizedPortInfo</MonitoringProfile.FacetTypes>
  </Filter>
  <Select use="exclude" isa="exclude">
    <MonitoringProfile use="exclude">
      <PollRate use="include"/>
    </MonitoringProfile>
  </Select>
</FilterSelect>
```

下图反映了前述的示例。响应显示包含 NormalizedPortInfo 度量标准（筛选条件）的监视配置文件，这些配置文件仅包含指定的 PollRate 属性（选择条件）。

The screenshot displays an HTTP client interface with two main sections: 'HTTP Request' and 'HTTP Response'.

HTTP Request Section:

- URL: `http://host:port/rest/monitoringprofiles/filtered`
- Method: `GET`
- Headers: `application/xml; charset=UTF-8`
- Body (XML):

```
1 <FilterSelect xmlns:xsl="http://www.w3.org/2001/XMLSchema-instance" xsl:noNamespaceSchemaLocation="filter.xsd">
2   <Filter>
3     <MonitoringProfile.FacetTypes type="CONTAINS">{http://m.ca.com/normalizer}NormalizedPortInfo</MonitoringProfile.FacetTypes>
4   </Filter>
5   <Select use="exclude" isa="exclude">
6     <MonitoringProfile use="exclude">
7       <PollRate use="include"/>
8     </MonitoringProfile>
9   </Select>
10 </FilterSelect>
```

HTTP Response Section:

- Status: `HTTP/1.1 200 OK`
- Body (XML):

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <MonitoringProfileList>
3   <MonitoringProfile version="1.0.0">
4     <ID>33</ID>
5     <PollRate>NORMAL</PollRate>
6     <RelatesTo>
7       <GroupIDList relatesURL="relatesto/groups" rootURL="groups">
8         <ID>18</ID>
9         <ID>22</ID>
10        <ID>25</ID>
11      </GroupIDList>
12    </RelatesTo>
13    <OutOfBox version="0.0.0">
14    </OutOfBox>
15  </MonitoringProfile>
</MonitoringProfileList>
```

管理数据驱动的 REST Web 服务的基本关系

使用下列方法来创建和删除数据驱动的 REST Web 服务的关系：

PUT `http://hostname:port/rest/endpoint/id/relatesto/endpoint/id`

DELETE `http://hostname:port/rest/endpoint/id/relatesto/endpoint/id`

也可以使用下列方法查看嵌套的关系：

`http://hostname:port/rest/endpoint/id/relatesto/endpoint/endpoint`

示例

使用以下 URL 查看与特定监视配置文件 ID 781 相关联的所有组和设备：

GET `http://hostname:port/rest/monitoringprofile/781/relatesto/groups/devices`

响应 URL 列出与指定的监视配置文件 ID 有关的所有组和设备，并在当前的 `getlist.xsd` 文件中列出属性。

第 5 章： 使用 Web 服务

此部分包含以下主题：

[将数据驱动的范围限制在承租方域](#) (p. 23)

[没有性能影响的轮询敏感和关键设备](#) (p. 23)

[更改数据保持期间](#) (p. 27)

[排定数据清除](#) (p. 29)

[更改计算同一天同一小时的基准平均值的时间](#) (p. 31)

[执行累加处理和基准计算时修改](#) (p. 32)

[更改受监视设备上的主 IP 地址](#) (p. 34)

[如何创建自定义度量标准系列](#) (p. 35)

[自动移除已报废的组件](#) (p. 64)

将数据驱动的范围限制在承租方域

您可以将操作的数据驱动的范围限制为某些 Data Aggregator 功能的特定承租方域，而不是在整个全局存储库中访问信息。

使用以下基本方法访问具有特定承租方域的端点的信息：

```
GET http://hostname:port/rest/tenant/id/endpoint
```

可以将一些（但非全部）Web 服务与承租方域结合使用。

没有性能影响的轮询敏感和关键设备

作为管理员，您要知道哪些关键设备对多次轮询敏感，可能产生性能问题。但要确保关键设备的性能，您必须在某种程度上监视这些敏感设备。通过配置 SNMP 轮询控件，您可以限制 SNMP 轮询请求，以使敏感设备崩溃。

默认情况下，SNMP 轮询通过两种方式控制：

- **SNMP 流量阈值** -- 同时可以将不超过 15 SNMP 请求发送给设备。超过 15 个的轮询和发现 SNMP 请求可以排队并在轮询周期可能的情况下发送给设备。最多可以有 600 个请求排队。
- **SNMP 超时阈值**：当 15 个或更多 SNMP 请求超时，轮询将根据当前轮询周期的提醒挂起。事件生成，向您通知情况。

注意：轮询在每个轮询周期开始时恢复。在超时不超过 15 个阈值时，生成“清晰的”事件。

两个阈值是为了防止轮询请求破坏设备。然而，您可以在需要时配置这些 **SNMP** 轮询阈值。

例如，您的较旧的路由器对轮询格外敏感。但是，此路由器非常关键并且必须尽可能频繁地被轮询。已调整您的监视配置文件，以便从轮询删除不必要的度量标准系列。也在监视配置文件中应用筛选，以减少轮询接口的数目。然而，调查仍然会导致此路由器崩溃。因此，您唯一的选择是调整敏感路由器的默认 **SNMP** 轮询参数。

某些参数（如下所示）可能会添加到 **IPRangeList** 中“**IPRange**”部分的个别 IP 或 IP 范围的策略：

MaxOutstandingRequests

发送到所示 IP 范围内所有设备的未完成请求的最大数量。

MaxRequestSize

限制传出 **SNMP** 请求中的 **OID** 数量。如果 **SNMP** 请求中的 **OID** 数量超过 **MaxRequestSize** 的值，则会将传出请求分为两个或更多较小的请求。

“**IPRange**”部分未包含某些 IP 范围。对于全局设置，请使用 **MaxRequestSizeDefault** 参数来设置 **OID** 限制。

请执行以下步骤：

1. 通过打开以下网址查找 IP 域 的 ID：

`http://hostname:port/rest/ipdomains`

主机名:端口

指定用于访问 REST Web 服务的 Data Aggregator 主机名和端口号。

2. 在下列 **SNMP** 限制策略列表中找到您的 IP 域 ID，并且注意相应的 policy ID：

`http://hostname:port/rest/snmpthrottlepolicies`

3. 确定要包含在单个传出 **SNMP** 请求中的 **OID** 数量。某些设备会忽略过大的请求而不发送错误。因此，**SNMP poller** 无法访问该设备。使用 **MaxRequestSize** 值以允许 Data Collector 监控这些设备。

示例

如果接口 **SNMP** 请求包含 27 个 **OID** 且将 **MaxRequestSizeDefault** 设成 15，则会将传出请求分为两个较小的请求。一个请求包含 14 个 **OID**，另一个请求则包含 13 个 **OID**。

示例：来自 SNMP 限制策略的下列示例显示 IP 域“2”的策略 ID 为“601”，且对 OID 的数量不做限制：

```
<SnmpThrottlePolicy version="1.0.0">
  <ID>601</ID>
  <MaxOutstandingRequestsDefault>15</MaxOutstandingRequestsDefault>
  <QueueLength>600</QueueLength>
  <TimeoutFailSafeThrottleDefault>15</TimeoutFailSafeThrottleDefault>
  <MaxRequestSizeDefault>0</MaxRequestSizeDefault>
  <IPDomainID>2</IPDomainID>
</SnmpThrottlePolicy>
```

4. 打开 REST 客户端编辑器或 HTTP 工具（用于发送请求并获得响应），并将上下文类型设置为 application/xml。
5. 通过输入以下条件来打开并编辑您的 IP 域的 SNMP 限制策略：

- URL: `http://hostname:port/rest/snmpthrottlepolicies/policyID`

指定唯一标识编号，这是分配给包含您的敏感设备的 IP 域的 SNMP 限制策略。

示例： `http://hostname:port/rest/snmpthrottlepolicies/601`

- HTTP 方法 = PUT
- 调整以下 HTTP 请求窗格“正文”选项卡上的“IP 范围”的值：
 - <MaxOutstandingRequests> – SNMP 流量阈值
 - <TimeoutFailSafeThrottle> – SNMP 超时阈值

注意：每个“IP 范围”实体都需要这两个值。您可以通过将值设置为“0”来禁用任何一个参数。

- 删除以下行：
 - <ID>
 - <IPDomainID>

将在“HTTP 响应”窗格的“正文”选项卡中返回结果。

示例：在此示例中，阈值仅针对设备 10.231.41.7 降至“10”。对于此设备，OID 的数量限制为 50。默认阈值和其他 IP 范围阈值将继续使用默认值“15”。对于设备 10.231.41.1-10.231.41.255，SNMP 请求最多可包含 30 个 OID。

```
<SnmpThrottlePolicy version="1.0.0">
  <IPRangeList>
    <IPRange>
      <IPRangeText>10.231.41.7</IPRangeText>
      <MaxOutstandingRequests>10</MaxOutstandingRequests>
      <TimeoutFailSafeThrottle>10</TimeoutFailSafeThrottle>
      <MaxRequestSize>50</MaxRequestSize>
    </IPRange>
    <IPRange>
      <IPRangeText>10.231.41.1-10.231.41.255</IPRangeText>
      <MaxOutstandingRequests>15</MaxOutstandingRequests>
      <TimeoutFailSafeThrottle>15</TimeoutFailSafeThrottle>
      <MaxRequestSize>30</MaxRequestSize>
    </IPRange>
  </IPRangeList>
  <MaxOutstandingRequestsDefault>15</MaxOutstandingRequestsDefault>
  <QueueLength>600</QueueLength>

  <TimeoutFailSafeThrottleDefault>15</TimeoutFailSafeThrottleDefault>
</SnmpThrottlePolicy>
```

注意：您可以调整单个设备或一系列设备的阈值。“IP 范围”定义和“IP 范围”顺序确定适用的阈值。“IP 范围”按优先级顺序列出。即，适用于设备的第一个 IP 范围确定要应用的阈值。

6. 在根级别目录下，应始终在更新/POST XML 中包括 *MaxOutstandingRequestsDefault*、*TimeoutFailSafeThrottleDefault* 和 *QueueLength* 参数。即便值与默认值不同，也应包括这些参数。

示例：

该 PUT 命令将生成下列策略。

```
更新 XML: PUT on URL DA-HOST:8581/rest/snmpthrottlepolicies/21
<SnmpThrottlePolicy version="1.0.0">
  <IPRangeList>

    <IPRange>

      <IPRangeText>130.119.103.8</IPRangeText>
      <MaxOutstandingRequests>10</MaxOutstandingRequests>
      <TimeoutFailSafeThrottle>10</TimeoutFailSafeThrottle>
      <MaxRequestSize>20</MaxRequestSize>
    </IPRange>
  </IPRangeList>
  <MaxRequestSizeDefault>50</MaxRequestSizeDefault>
  <MaxOutstandingRequestsDefault>15</MaxOutstandingRequestsDefault>
  <TimeoutFailSafeThrottleDefault>15</TimeoutFailSafeThrottleDefault>
  <QueueLength>600</QueueLength>
</SnmpThrottlePolicy>
```

该命令将生成以下策略：

```
<SnmpThrottlePolicy version="1.0.0">
  <ID>21</ID>
  <QueueLength>600</QueueLength>
  <TimeoutFailSafeThrottleDefault>15</TimeoutFailSafeThrottleDefault>
  <IPDomainID>2</IPDomainID>
  <IPRangeList>
    <IPRange>
      <IPRangeText>130.119.103.8</IPRangeText>
      <MaxOutstandingRequests>10</MaxOutstandingRequests>
      <TimeoutFailSafeThrottle>10</TimeoutFailSafeThrottle>
      <MaxRequestSize>20</MaxRequestSize>
    </IPRange>
  </IPRangeList>
  <MaxRequestSize>50</MaxRequestSize>
  <MaxOutstandingRequestsDefault>15</MaxOutstandingRequestsDefault>
</SnmpThrottlePolicy>
```

更改数据保持期间

您可以更改 **Data Repository** 轮询数据、每小时累加数据、每日累加数据和每周累加数据的保留率。例如，您可以将轮询的数据保留值更改为 30 天以节省磁盘空间。寻找最适合需求和环境的平衡。

默认情况下，数据在 Data Repository 中保留以下天数：

- 轮询的数据：45 天

注意：如果您从 Data Aggregator 的以前版本升级到此版本，轮询数据保留期将仍为以前的默认值 10 天。

- 每小时累加数据：90 天
- 每日累加数据：365 天
- 每周累加数据：730 天

Data Repository 可以保留数据的最短天数如下所示：

- 轮询数据：2 天
- 每小时累加数据：8 天
- 每日累加数据：31 天
- 每周累加数据：366 天

请执行以下步骤：

1. 在 Web 浏览器中输入以下信息：

2. `http://hostname:port/rest/globalretentiondefinition`

hostname:port

指定 Data Aggregator 主机名和端口号。

默认端口：8581

将打开 globalretentiondefinition Web 服务端点 URL。

3. 请记下分配给 globalretentiondefinition 的 ID。
4. 查找具有 GtdRollupDataRetentionPeriod、DailyRollupDataRetentionPeriod、PolledDataRetentionPeriod 以及 HourlyRollupDataRetentionPeriod 的元素。此信息帮助您确定您要为哪些类型的数据修改保留期。
5. 打开 REST 客户端编辑器或 HTTP 工具（可以发送请求并获得响应），并将上下文类型设置为 application/xml。
6. 输入以下条件：
 - URL: `http://hostname:port/rest/globalretentiondefinition/ID`
ID
是分配给 globalretentiondefinition 的唯一标识号。
 - HTTP 方法 = PUT
 - 在“HTTP 请求”窗格的“正文”选项卡中，输入您希望更改的保持期。

例如：

```
<GlobalRetentionDefinition version="1.0.0">
<PolledDataRetentionPeriod>4</PolledDataRetentionPeriod>
</GlobalRetentionDefinition>
```

重要说明！ 确认在每一行的开头没有空格，否则 PUT 将失败。

在本示例中，轮询的数据的保持期间已更改为四天。

将在“HTTP 响应”窗格的“正文”选项卡中返回结果。

例如：

```
<GlobalRetentionDefinitionList>
<GlobalRetentionDefinition version="1.0.0">
  <ID>4</ID>
  <GtdRollupDataRetentionPeriod>730</GtdRollupDataRetentionPeriod>
  <DailyRollupDataRetentionPeriod>365</DailyRollupDataRetentionPeriod>
  <PolledDataRetentionPeriod>4</PolledDataRetentionPeriod>
  <HourlyRollupDataRetentionPeriod>90</HourlyRollupDataRetentionPeriod>
</Item version="1.0.0">
  <CreateTime>Thu Dec 08 16:03:05 CST 2011</CreateTime>
  <Name>Global Retention Definition</Name>
</Item>
</GlobalRetentionDefinition>
</GlobalRetentionDefinitionList>
```

在本示例中，轮询的数据的保持期间已更改为四天。每周累加数据、每日累加数据和每小时累加数据的默认保留期将保持不变。

排定数据清除

可以排定数据存储库清除早于指定保持期的所有数据的频率。您可以修改开始小时数，开始分钟数，也可以修改开始秒数。默认情况下，Data Aggregator 每天凌晨 2:00:00 清除数据。

请执行以下步骤：

1. 在 Web 浏览器中输入以下信息：
2. `http://hostname:port/rest/globalretentionscheduledefinition`
hostname:port

指定用于访问 REST Web 服务的 Data Aggregator 主机名和端口号。

将打开 globalretentionscheduledefinition Web 服务端点 URL。

3. 请记下分配给 globalretentionscheduledefinition 的 ID。
4. 查找具有 StartMinute、StartHour 和 StartSecond 的元素。使用此信息确定清除旧数据时是否要修改开始小时数、开始分钟数或开始秒数。

5. 打开 REST 客户端编辑器或 HTTP 工具（可以发送请求并获得响应），并将上下文类型设置为 `application/xml`。
6. 输入以下条件：

- URL:
`http://hostname:port/rest/globalretentionscheduledefinition/ID`

ID 是分配给 `globalretentionscheduledefinition` 的唯一标识号。

- HTTP 方法 = PUT
- 输入要在“HTTP 请求”窗格的“正文”选项卡上更改的时间值。

例如：

```
<GlobalRetentionScheduleDefinition version="1.0.0">
  <StartMinute>28</StartMinute>
  <StartHour>17</StartHour>
  <Enabled>true</Enabled>
  <Status>Scheduled to run everyday at 17:28:00</Status>
</GlobalRetentionScheduleDefinition>
```

重要说明！ 确保在每一行的开头没有空格，否则 PUT 操作将失败。

在本示例中，开始小时数已更改为 17，开始分钟数已更改为 28。

注意：要禁用清除作业，请将 `<Enabled>` 设置为 `false`。要重新启用清除作业，请将 `<Enabled>` 设置为 `true`。

将在“HTTP 响应”窗格的“正文”选项卡中返回结果。

例如：

```
<GlobalRetentionScheduleDefinitionList>
<GlobalRetentionScheduleDefinition version="1.0.0">
  <ID>9</ID>
  <StartMinute>28</StartMinute>
  <StartHour>17</StartHour>
  <Enabled>true</Enabled>
  <JobStatus>Has never run</JobStatus>
  <Status>Scheduled to run everyday at 17:28:00</Status>
  <StartSecond>0</StartSecond>
</Item version="1.0.0">
  <CreateTime>Thu Dec 15 15:52:20 EST 2011</CreateTime>
  <Name>Global Retention Schedule Definition</Name>
</Item>
</GlobalRetentionScheduleDefinition>
</GlobalRetentionScheduleDefinitionList>
```

在本示例中，将在每天 17:28:00 清除早于指定保持期间的数据。

更改计算同一天同一小时的基准平均值的时间

最初，会在收集到有限数量的数据后，于每一周前一天的同一小时计算基准平均值。

有更多数据后，会自动切换计算方法。**Data Aggregator** 通过计算一周内同一天前可用天数的每小时样本的平均值，来建立“标准”。

默认情况下，在至少一周内 3 天中同一小时的数据样本可用于过去 12 个周时，会出现这种自动改变。

可以更改发生此自动改变的时间。

请执行以下步骤：

1. 在 Web 浏览器中输入以下信息：

`http://hostname:port/rest/sdshbaselineconfig`

hostname:port

指定 Data Aggregator 主机名和端口号。

默认端口：8581

将打开 sdshbaselineconfig Web 服务端点 URL。

2. 查看所需的数据点数最小值和之前周数的最大值的当前值。
3. 打开 REST 客户端编辑器或 HTTP 工具（可以发送请求并获得响应），并将上下文类型设置为 `application/xml`。
4. 输入以下条件：
 - HTTP 方法 = PUT
 - 输入要在“HTTP 请求”窗格的“正文”选项卡上更改的最大数量的之前周数内所需的数据点数最小值（触发基准计算方式切换）。

例如：

```
<SdshBaselineConfiguration version="1.0.0">
  <SDSHSettings>

    <MinimumNumberOfRequiredDataPoints>5</MinimumNumberOfRequiredDataPoints>

    <MaximumNumberOfWeeks>10</MaximumNumberOfWeeks>

  </SDSHSettings>
</SdshBaselineConfiguration>
```

在此示例中，已将用于基准平均值计算的数据点最小值更改为 5。已将查找这些数据点的之前周数更改为 10。

在计算一周内同一天同一小时的基准平均值时，请考虑有关进行更改的以下信息：

- 您可以选择更改两个属性或其中一个属性。
- 两个属性值都必须是大于或等于 1 的数值。
- 没有上限。但是，每小时累加的保留策略定义了上限。默认情况下，每小时保留期是 90 天（大约是 12 周的数据）。如果您要加大之前周数的最大值，请同时增加每小时累加保留率。
- `MinimumNumberOfRequiredDataPoints` 属性值必须小于或等于 `MaximumNumberOfWeeks` 值。

执行累加处理和基准计算时修改

管理员可在执行累加处理和基准计算时进行修改。通过更改这些操作的运行时间，管理员可以将这些 **Vertica** 敏感型操作排定为在非工作时间进行。在非工作时间执行这些操作后，就不会在工作时间影响生成报告的用户。

默认情况下，累加处理和基准计算在每天每小时的半点进行。

请执行以下步骤：

1. 在 Web 浏览器中输入以下 URL：

`http://主机名:端口/rest/rollups/config`

hostname:port

指定 Data Aggregator 主机名和端口号。

默认端口：**8581**

2. 请记录指定给配置项的 ID。
3. 打开 REST 客户端编辑器或可以发送请求和获得响应的 HTTP 工具。输入以下条件：

- URL: `http://hostname:port/rest/rollups/config/ID`

主机名:端口

指定 Data Aggregator 主机名和端口号。

默认端口：8581

ID

是指定给配置项的唯一标识号。您在上个步骤中记录的编号。

- HTTP 方法: PUT
- 在 HTTP 请求窗格的“正文”选项卡中,输入您希望累加处理开始和结束的当日时间。

默认情况下,将会返回下列结果:

```
<RollupsConfigurationList>
  <RollupsConfiguration_version="1.0.0">
    <ID>8</ID>
    <StartHour>0</StartHour>
    <EndHour>23</EndHour>
  </RollupsConfiguration>
</RollupsConfigurationList>
```

StartHour

以 24 小时的时间格式定义累加处理开始时的当日时间（当地时间）。

EndHour

以 24 小时的时间格式定义累加处理结束时的当日时间（当地时间）。在结束之后,将不会启动新的累加,但可以完成任何正在进行的累加。

注意: 有关这些属性的详细信息,请参见
<http://hostname:port/rest/rollups/config/documentation>.

示例: 在此示例中,您可以更改时间表,以便仅在 20:00 到 7:00 期间允许累加处理和基准计算。

```
<RollupsConfigurationList>
  <RollupsConfiguration_version="1.0.0">
    <ID>8</ID>
    <StartHour>20</StartHour>
    <EndHour>6</EndHour>
  </RollupsConfiguration>
</RollupsConfigurationList>
```

<EndHour> 非独占。在此示例中,意味着如果您将 6 指定为 EndHour,那么累加处理和基准计算将在 06:00 结束时开始,而不会在 07:00 开始。允许完成任何正在进行的计算。

重要说明! 对默认排定的任何修改都会大大延迟数据在与相应解决方案有关的报告中的显示。例如,如果每小时累加被延迟,则显示每小时解决方案数据的报告只有在执行每小时累加后才是最新的。

更改受监视设备上的主 IP 地址

作为管理员，即使设备上的 IP 地址变更，您也会希望受监视设备可以保持数据的连续性。如果设备上的 IP 地址变更，而您没有更新，那么随后的发现将会创建新的受监视设备。

在您更改主 IP 地址之前，请考虑以下信息：

- 如果设备带有 DNS 主机名，那么在 IP 地址变更和旧有 IP 地址无法访问时，监视设备的 **Data Collector** 实例将会执行反向主机名查询，以查找和设置具有新 IP 地址的设备项。
- 如果您将设备的主 IP 地址更改为其他设备使用的 IP 地址，将会显示错误消息。

注意：如果受监视设备上的主 IP 地址变更，则设备上将会生成事件。

如要更改受监视设备上的主 IP 地址，请打开 REST 客户端编辑器或 HTTP 工具，输入以下条件：

- URL: `http://hostname:port/rest/devices/deviceitemID`

hostname:port

指定 Data Aggregator 主机名和端口号。

默认端口：8581

deviceitemID

是您要更改主 IP 地址的受监视设备的设备项标识号。

注意：如果您将 IP 地址更改为现有受监视设备的 IP 地址，将会返回错误。

- HTTP 方法：PUT
- 在 HTTP 请求窗格的“正文”选项卡中，输入更改的主 IP 地址。

例如：

```
<Device version="1.0.0">
  <PrimaryIPAddress>IP</PrimaryIPAddress>
</Device>
```

IP

是更改的主 IP 地址。

示例：在此示例中，您要将受监视设备上的主 IP 地址变更为 1.2.3.4：

```
<Device version="1.0.0">  
    <PrimaryIPAddress>1.2.3.4</PrimaryIPAddress>  
</Device>
```

如何创建自定义度量标准系列

工厂度量标准系列定义要监视的最常见的度量标准属性。安装后便包含多个度量标准系列，这些度量标准系列可以满足大多数用户的需求。但是，当您要为新度量标准属性收集数据时，可以创建一个自定义度量标准系列。例如，如果不存在用于收集进程数据的度量标准系列，则可以创建一个。

自定义度量标准系列定义以下详细信息：

- 要收集哪些度量标准
- 如何计算值
- （可选）如何在视图中显示值

或者，您然后可以创建供应商认证（如果不存在）通过新的度量标准系列来监控进程的度量标准属性。

重要说明！ 请始终先在测试环境中创建和验证您的自定义度量标准系列。创建自定义度量标准系列需要您手工编辑度量标准系列 XML 文件。该 XML 文件中的语义错误可能会导致不可预测的结果。

注意：Data Aggregator 提供了用于创建自定义供应商认证和度量标准系列的基本方法和高级方法。基本方法比较简单，包括使用用户界面为现有支持技术（度量标准系列）添加供应商支持。此方法满足许多用户的要求。而高级方法是以工厂认证格式为基础，并且提供一套完整的功能。此指南对基本认证方法进行了说明。有关高级认证方法的信息，请参阅《Data Aggregator 超级用户认证指南》。

创建自定义度量标准系列是一个复杂的过程。要创建自定义度量标准系列，请仔细遵循以下步骤：

1. [查看现有度量标准系列，以便确定您是否需要自定义度量标准系列](#) (p. 36)。
2. 下载架构和示例度量标准系列文件。
3. [创建自定义度量标准系列 XML 文件](#) (p. 37)。

4. [确认您的自定义度量标准系列在测试环境中的结果](#) (p. 63)。
5. [导入自定义度量标准系列 xml 文件](#) (p. 64)。

重要说明！ 为避免可能发生的数据丢失，请一定要在每次创建或更新供应商认证、度量标准系列或组件时都备份您的 **deploy** 目录。

查看度量标准系列

检查度量标准系列的列表以查看 **Data Aggregator** 安装中支持的度量标准。查看度量标准系列将显示其与设备集合、供应商认证和监视配置文件之间的关联。了解度量标准系列、设备集合与设备类型之间的关联关系可帮助您控制如何监视设备。此外，您还可以确定是否需要其他度量标准系列来充分监视环境。

请执行以下步骤：

1. 从 **Data Aggregator** 数据源的“监视配置”菜单中单击“度量标准系列”。
此时将显示度量标准系列列表，其中包括工厂和自定义度量标准系列。预定义认证会显示锁定符号。
2. 从列表中选择度量标准系列。
3. 单击选项卡以获取更多信息：

“度量标准”选项卡

显示包含在选定度量标准系列中的度量标准和每个度量标准的各种属性。

供应商认证优先级选项卡

显示与选定度量标准系列关联的设备集合列表。通常，一个度量标准系列与单个设备集合关联。选择设备集合时，将显示 **MIB** 源（供应商认证）的已确定优先级的列表。此信息说明供应商认证应用于选定度量标准系列的该设备集合的优先级顺序。

“监控配置文件”选项卡

显示关联的监控配置文件及其轮询速度的列表。

详细信息：

[如何创建自定义度量标准系列](#) (p. 35)

下载度量标准系列架构和示例文件

在创建自定义度量标准系列 XML 文件之前，下载并查看度量标准系列架构和示例度量标准系列 XML 文件。您需要该架构来创建自己的 XML 文件。示例演示了如何对成功的度量标准系列进行编码。在创建自己的文件之前检查这些文件有助于确保 XML 内容的准确性。

找到以下文件并将其下载到同一文件夹中：

- <http://hostname:port/resource/xsd/MetricFamily.xsd>
- <http://hostname:port/resource/xsd/Component.xsd>
- <http://hostname:port/resource/examples/metricFamily/ProcessInfoMFWithComponent.xml>

注意：有关每个 XML 标记的详细信息，请参阅 `MetricFamily.xsd` 和 `Component.xsd` 文件中提供的行内注释式文档。有关代码示例，请参阅 `ProcessInfoMFWithComponent.xml` 文件。此示例文件定义用于收集进程度量标准的度量标准系列。

详细信息：

[如何创建自定义度量标准系列](#) (p. 35)

创建自定义度量标准系列 XML 文件

在收集所需文件之后，您可以创建自定义度量标准系列 XML 文件。

注意：有关每个 XML 标记的详细信息，请参阅 `MetricFamily.xsd` 和 `Component.xsd` 文件中提供的行内注释式文档。有关代码示例，请参阅 `ProcessInfoMFWithComponent.xml` 文件。此示例文件定义用于收集进程度量标准的度量标准系列。

请执行以下步骤：

1. 使用任何 XML 编辑器从 `MetricFamily.xsd` 创建一个 XML 文件。任何自定义度量标准系列以 `<TechnologyCertification>` 为根元素。

建议您使用 Web 服务通过复制示例度量标准系列 XML 文件或导出工厂 XML 文件 (<http://hostname:port/genericWS/metricfamilies>)。

重要说明！ 不要从 `deploy` 目录复制度量标准系列 XML 文件。这些文件使用仅限内部使用的格式，与自定义度量标准系列不兼容。

注意：不能修改工厂度量标准系列。但是，您可以复制然后修改工厂度量标准系列。

2. 编辑 XML 文件中的内容，如下所示：
 - a. [复查度量标准系列 xml 文件中的数据类型用法](#) (p. 38)。
 - b. [添加用于定义度量标准系列的基本信息](#) (p. 41)。
 - c. [添加 <AttributeList> 部分](#) (p. 43)。
 - d. [添加 <BaselineList> 部分](#) (p. 47)。
 - e. [添加 <ComponentList> 部分](#) (p. 49)。
 - f. [添加 <ComponentDefinitionList> 部分](#) (p. 50)。
 - g. (可选) [添加 <ComponentReconciliation> 部分](#) (p. 55)。
 - h. (可选) [添加 <ReconfigDetectionAttr> 部分](#) (p. 60)。
 - i. [添加 <ExpressionGroupList> 部分](#) (p. 60)。
3. 保存文件。

自定义度量标准系列已创建，并可导入到 Data Aggregator 中。

重要说明！ 请始终先在测试环境中创建和验证您的自定义度量标准系列。创建自定义度量标准系列需要您手工编辑度量标准系列 XML 文件。该 XML 文件中的语义错误可能会导致不可预测的结果。

详细信息：

[如何创建自定义度量标准系列](#) (p. 35)

度量标准系列 XML 文件中的数据类型用法

本部分提供了有关在度量标准系列和供应商认证 XML 文件中使用数据类型的指南。

ObjectID

要访问轮询 OID，而不是轮询的返回值，请使用 *ObjectID* 类型。

Integer/Long/Double

在供应商认证中，数值通常存储为 *整数类型* 或 *Long* 类型。度量标准系列通常使用 *Double* 类型。

BigInteger/Double

Data Aggregator 支持轮询 64 位计数器，从而支持为高速接口收集数据。随着网络介质速度的增加，32 位计数器完成计数所需的最短时间将会缩短。使用 64 位计数器会延长计数器完成计数所花的时间并能够以正常的速率轮询。MIB2 中的 ifxTable 提供了 64 位计数器，对于供应商认证来说，它通常存储为 *BigInteger* 类型。度量标准系列通常使用 *Double* 类型。

String/OctetString

在供应商认证中，字符串值存储为 *字符串* 类型。度量标准系列使用 *OctetString* 类型。

QName

在度量标准系列中，针对 *FacetTypes* 属性使用的特殊类型。

注意：CA 性能管理 提供的架构文件详细介绍了类型信息。在创建 XML 文件期间，了解架构的 XML 编辑器（如 XML 记事本）会使用此信息，并可为您提供帮助。

ProcessInfoVCForEmpireMIB.xml 部分文件的示例，其中具有定义此供应商认证中 MIB 对象属性的数据类型

```
<Attribute name="INDEX" type="ObjectID">
  <!--This variable serves as the index for the other variables in the same
MIB table.-->
  <IsKey>true</IsKey>
  <IsIndex>true</IsIndex>
  <Source>1.3.6.1.4.1.546.1.1.4.1.1</Source>
</Attribute>
<Attribute name="processID" type="Long">
  <!--The unique process ID (e.g. 0).-->
  <IsKey>true</IsKey>
  <NeedsDelta>>false</NeedsDelta>
  <Source>1.3.6.1.4.1.546.1.1.4.1.1</Source>
</Attribute>
<Attribute name="processName" type="String">
  <!--The name of the running process (e.g. syslogd).-->
  <IsKey>true</IsKey>
  <NeedsDelta>>false</NeedsDelta>
  <Source>1.3.6.1.4.1.546.1.1.4.1.2</Source>
</Attribute>
<Attribute name="processRSS" type="Long">
  <!--进程的实际内存（驻留集）大小（KB）。此值表示进程保留的字节数。-->
  <IsKey>>false</IsKey>
  <NeedsDelta>>false</NeedsDelta>
  <Source>1.3.6.1.4.1.546.1.1.4.1.11</Source>
</Attribute>
```

ProcessInfoMFWithComponent.xml 部分文件的示例，其中具有定义此度量标准系列中度量标准属性的数据类型

```
<Attribute>
  <Name>{http://im.ca.com/normalizer}ProcessInfo.Indexes</Name>
  <AttributeDisplayName>Indexes</AttributeDisplayName>
  <Description></Description>
  <Type>ObjectID</Type>
  <WriteOnPoll>false</WriteOnPoll>
  <Polled>false</Polled>
  <IsList>true</IsList>
  <IsDbColumn>false</IsDbColumn>
</Attribute>
<Attribute>
  <Name>{http://im.ca.com/normalizer}ProcessInfo.PID</Name>
  <AttributeDisplayName>PID</AttributeDisplayName>
  <Description>The process ID for the process in the OS.</Description>
  <Type>Integer</Type>
  <WriteOnPoll>false</WriteOnPoll>
  <Polled>false</Polled>
  <IsList>true</IsList>
  <IsDbColumn>false</IsDbColumn>
</Attribute>
<Attribute>
  <Name>{http://im.ca.com/normalizer}ProcessInfo.Names</Name>
  <AttributeDisplayName>Names</AttributeDisplayName>
  <Description></Description>
  <Type>String</Type>
  <WriteOnPoll>false</WriteOnPoll>
  <Polled>false</Polled>
  <IsList>true</IsList>
  <IsDbColumn>false</IsDbColumn>
</Attribute>
<Attribute>
  <Name>{http://im.ca.com/normalizer}ProcessInfo.Memory</Name>
  <AttributeDisplayName>Memory</AttributeDisplayName>
  <Description>The total amount of real system memory allocated to this
process, in kilobytes.</Description>
  <Type>Double</Type>
  <WriteOnPoll>false</WriteOnPoll>
  <Polled>true</Polled>
  <IsList>true</IsList>
  <IsDbColumn>true</IsDbColumn>
  <Baseline>true</Baseline>
  <Maximum>true</Maximum>
  <Minimum>true</Minimum>
  <Variance>true</Variance>
  <StandardDeviation>true</StandardDeviation>
  <Percentile>95</Percentile>
  <RollupStrategy>Avg</RollupStrategy>
</Attribute>
```


详细信息:

[创建自定义度量标准系列 XML 文件](#) (p. 37)

添加用于定义自定义度量标准系列的基本信息

自定义度量标准系列的基本属性可帮助您将其与您创建的其他自定义度量标准系列区分开来。此外，这些属性还指示已收集的度量标准数据的存储位置。



在 XML 文件的该部分中包括的标记如下所示:

名称

指定度量标准系列名称。每个度量标准系列必须具有在系统内部用于标识的唯一名称。

注意: 绝不能对外公开此名称。要在用户界面中显示度量标准系列名称，请使用 `DisplayName` 元素。

DisplayName

指定要在用户界面中显示的度量标准系列名称。

说明

指定属性的外部说明。

CertificationType

定义度量标准系列的类型。

注意: 用户定义的度量标准系列仅支持 `CUSTOM`。

TableName

指定存储度量标准系列收集的度量标准的数据库表名称。

注意: 数据表名称必须是唯一的（其他度量标准系列不能使用该值），由大写字母和下划线组成，例如 `PROCESS_STATS`。

注意: 有关每个 XML 标记的详细信息，请参阅 `MetricFamily.xsd` 和 `Component.xsd` 文件中提供的行内注释式文档。有关代码示例，请参阅 `ProcessInfoMFWithComponent.xml` 文件。此示例文件定义用于收集进程度量标准的度量标准系列。

详细信息:

[创建自定义度量标准系列 XML 文件](#) (p. 37)

基本属性

如下图中“TechnologyCertification”部分所示，提供命名空间和架构信息：



下列信息描述了“TechnologyCertification”部分中的元素：

名称

指定度量标准系列名称。每个度量标准系列必须具有在系统内部用于标识的唯一名称。

注意：绝不能对外公开此名称。要在用户界面中显示度量标准系列名称，请使用 `DisplayName` 元素。

DisplayName

指定要在用户界面中显示的度量标准系列名称。

说明

指定属性的外部说明。

CertificationType

定义度量标准系列的类型。

注意：用户定义的度量标准系列仅支持 `CUSTOM`。

TableName

指定存储度量标准系列收集的度量标准的数据库表名称。

注意：数据表名称必须是唯一的（其他度量标准系列不能使用该值），由大写字母和下划线组成，例如 `PROCESS_STATS`。

添加 <AttributeList> 部分

<AttributeList> 部分定义了您希望为度量标准系列收集的配置度量标准或性能度量标准。即使许多属性对于某一设备类型可用，您也要仔细规划哪些属性对于监视基础架构重要。属性数据的三种常见用途如下：

- 报告—这些属性值可以经常更改。通常，在每个周期中都会轮询这些值，从而提供在报告中使用的易变值，以说明随着时间的推移而进行的变更。示例包括传入/传出字节数、带宽和 CPU 内存使用率。
- 配置—这些属性提供用于标识项的数据。该数据很少发生更改。因此，通常仅在发现期间收集该数据（即，在每个周期中不轮询该数据）。示例包括名称、说明和 ID。
- 调整—这些属性的用途是检测环境中的配置更改。检测到的变更会触发发现，以帮助确保您的 Data Aggregator 数据准确。

每个属性可以包括以下标记：

<Name>

该属性的唯一内部名称（限定名称格式）。

<Description>

该文本作为工具提示显示在您的 CA Performance Center 视图中。

<AttributeDisplayName>

该标记用以确定您的属性在报告和 CA Performance Center 显示板中的标识方式。

<Type>

该属性的数据类型。最常用的数据类型是整数、Long、Double、字符串或对象 ID。

<Polled>

设置为 True 时，Data Aggregator 将在每个轮询周期内收集此度量标准数据。设置为 False 时，仅在发现一个项目时收集该数据。

<IsList>

该属性提供标量或表数据。请考虑以下定义：

- 标量属性—这些属性一次返回一个值，此值与设备数据一起存储。例如，“当前进程数”的属性返回单个值。
- 表属性-- 这些属性在为表中每行返回一个值。示例进程度量标准系列使用表属性。每行对应于单个进程项。

注意：如果此标记对属性为 True，必须为相同属性列表中的每个属性将其设置为 True。在这种情况下，必须在列表中定义索引（对象 ID 类型）和名称（字符串类型）属性。

<IsDbColumn>

该标记指示存储在数据库表中的属性值。对于配置度量标准，请将此标记值设置为 **False**。

<RollupStrategy>

该标记指定计算单个轮询值的总和或平均值。当 **<Polled>** 和 **<IsDbColumn>** 设置为 **True** 时，需要使用此标记。

<Baseline>

设置为 **True** 时，将为该属性计算基准值，您可以在 **<BaselineList>** 部分中添加相应的基准定义。

当包括 **<RollupStrategy>** 或将 **<Baseline>** 设置为 **True** 时，还需要使用以下标记：

基准

表明是否计算此属性的基准。如果将其设置为 **True**，则必须存在对应的 **BaselineList** 定义。

最大值

表明在累加期间是否计算此属性的最大值。在数据库表中创建“max_”列。

最小值

表明在累加期间是否计算此属性的最小值。在数据库表中创建“min_”列。

方差

表明在累加期间是否计算此属性的方差。在数据库表中创建“var_”列。

StandardDeviation

表明在累加期间是否计算此属性的标准偏差。在数据库表中创建“std_”列。

RollupStrategy

在单独累加轮询值期间，指定执行每个周期的运算符。该值可以是用于计算总和或平均数的“Sum”或“Avg”

<Filterable>

该标记指示可用于监视配置文件筛选的属性。在应用筛选时，具有此标记的属性将适用于筛选表达式。

注意：有关每个 XML 标记的详细信息，请参阅 `MetricFamily.xsd` 和 `Component.xsd` 文件中提供的行内注释式文档。有关代码示例，请参阅 `ProcessInfoMFWWithComponent.xml` 文件。此示例文件定义用于收集进程度量标准的度量标准系列。

详细信息：

[创建自定义度量标准系列 XML 文件](#) (p. 37)

AttributeList

“属性列表”列出收集并存储在数据库表中的属性。下列信息描述了您在“AttributeList”部分中使用的元素：

名称

指定用于命名数据库列的唯一内部名称。系统指定名称要采用 QName 格式：

`{URI}FamilyName.AttrName`

注意：绝不能对外公开此名称。要在用户界面中显示属性名称，请使用 `AttributeDisplayName` 元素。

AttributeDisplayName

指定运算符和管理员界面中显示的值。`AttributeDisplayName` 可以本地化为多种语言。

说明

在用户界面中显示属性说明。将鼠标指针悬停在属性名称时，说明还会显示在工具提示中。

类型

表明此属性的数据类型。最常用的数据类型是整数、*Long*、*Double*、字符串或 *ObjectID*。

已轮询

表明该属性是否已轮询。如果将其设置为 `False`，则仅可在发现期间访问它。

IsList

如果将其设置为 **True**，则从表中检索属性。还可以为表中的每行创建组件项。

注意：在您将任意属性的特性设置为 **True** 后，则需将存储在数据库中度量标准系统的每个属性的特性都设置为 **True**。

IsDbColumn

在数据库表中存储其值。IsDbColumn 通常用于每个周期都轮询且包含于报告中的属性。

注意：如果您尚未使用 ProcessInfoMFWWithComponent.xml 但从 <http://host:port/genericWS/metricfamilies/<name>> 下载了 XML 文件以启用度量标准系列，请删除 RollupExpression 和单位节点，因为它们已被弃用。

常规属性

在下图中，任何度量标准系列都存在 Indexes、Names 和 Descriptions 属性。

Attribute	
Name	{http://im.ca.com/normalizer}ProcessInfo.Indexes
AttributeDisplayName	Indexes
Description	
Type	ObjectID
Polled	false
IsList	true
IsDbColumn	false
Attribute	
Name	{http://im.ca.com/normalizer}ProcessInfo.Names
AttributeDisplayName	Names
Description	
Type	String
Polled	false
IsList	true
IsDbColumn	false
Attribute	
Name	{http://im.ca.com/normalizer}ProcessInfo.Descriptions
AttributeDisplayName	Descriptions
Description	
Type	String
Polled	false
IsList	true
IsDbColumn	false

任何支持供应商认证可以提供度量标准系列展现的 URI，如：

{http://im.ca.com/normalizer}Name.Indexes

{http://im.ca.com/normalizer}Name.Names

{http://im.ca.com/normalizer}Name.Descriptions

简单属性

在下图中，数据库只存储大多数属性的值。没有进行进一步的处理（例如基准评估）。

Attribute	
Name	{http://im.ca.com/normalizer}ProcessInfo.PID
AttributeDisplayName	PID
Description	The process ID for the process in the OS.
Type	Integer
Polled	false
IsList	true
IsDbColumn	false
Attribute	
Name	{http://im.ca.com/normalizer}ProcessInfo.PPID
AttributeDisplayName	Parent PID
Description	The parent process ID for the process in the OS.
Type	Integer
Polled	false
IsList	true
IsDbColumn	false
Attribute	
Name	{http://im.ca.com/normalizer}ProcessInfo.Params
AttributeDisplayName	Parameters
Description	Any parameters passed to the process at...
Type	String
Polled	false
IsList	true
IsDbColumn	false
Attribute	
Name	{http://im.ca.com/normalizer}ProcessInfo.Owner
AttributeDisplayName	Owner
Description	The owner of the process.
Type	String
Polled	false
IsList	true
IsDbColumn	false

支持的供应商认证必须采用下列格式提供一些度量标准系列公开的 URI:

```
{http://im.ca.com/normalizer}Name.AttributeName
```

添加 <BaselineList> 部分

在分析环境时，了解某个项何时超出正常范围操作是很有价值的信息。为了帮助您确定“正常”值，Data Aggregator 可以计算基准值。

注意：有关报告中的基准的详细信息，请参阅《Data Aggregator 管理员指南》。

对于自定义度量标准系列，您可以指示 Data Aggregator 计算属性的基准信息。自定义度量标准系列中的 <BaselineList> 部分指示如何计算给定属性的基准信息。

重要说明！ 将 <Baseline_> 标记设置 True 后，必须在此部分为 <AttributeList> 部分中的每个属性定义基准计算。

在此部分中，设置基准计算的选项如下所示：

名称

指定基准定义名称。

ID

指定基准定义的唯一标识符。在此度量标准系列的所有基准定义集合内，该值必须是唯一的。

PerformanceMetric

指定需要计算基准的度量标准的名称。指定 `AttributeList` 中的名称；必须将该属性的“已轮询”特性设置为 `True`。

StartDate

已弃用。如果存在，则将其设置为 0。

EndDate

已弃用。如果存在，则将其设置为 0。

DaysOfWeek

已弃用。如果存在，则将其设置为 0。

时间

指定计算每时基准还是每日基准。

窗口

已弃用。如果存在，则将其设置为 30 天。

注意：有关每个 XML 标记的详细信息，请参阅 `MetricFamily.xsd` 和 `Component.xsd` 文件中提供的行内注释式文档。有关代码示例，请参阅 `ProcessInfoMFWWithComponent.xml` 文件。此示例文件定义用于收集进程度量标准的度量标准系列。

下图提供了使用 `BaselineList` 元素的示例：

BaselineList	
Baseline	
Name	HourlyBaseline
ID	1
PerformanceMetric	Memory
StartDate	0
EndDate	0
Period	1 Hour
DaysOfWeek	0
Window	30 Days
Baseline	
Name	HourlyBaseline
ID	2
PerformanceMetric	CpuTime
StartDate	0
EndDate	0
DaysOfWeek	0
Period	1 Hour
Window	30 Days
Baseline	
Name	DailyBaseline
ID	3
PerformanceMetric	Memory
StartDate	0
EndDate	0
DaysOfWeek	0
Period	1 Day
Window	90 Days

详细信息：

[创建自定义度量标准系列 XML 文件](#) (p. 37)

添加 <ComponentList> 部分

组件是与设备关联的项（例如，设备可与 CPU、接口和进程组件关联）。使用组件类型可帮助您对与设备关联的项进行分类。在度量标准系列中，表属性需要关联组件的列表。对于 `Data Aggregator` 中当前不存在的组件，您的度量标准系列 XML 还必须定义这些新组件。

度量标准系列 XML 文件中的组件信息用于以下两种用途：

- 发现—当发现为设备创建项目时，组件值将标识该类别（方面）。
- 同步—在 Data Aggregator 与 CA Performance Center 之间进行同步时，组件定义会映射项并确定项在界面中的显示方式。

在 <ComponentList> 部分中，列出您想为与自定义度量标准系列相关联的任何组件项创建的方面。例如，收集进程度量标准的度量标准系列与运行进程的设备相关联。我们可以在此部分中定义一个“进程”方面，如下所示：

```
<Component>{namespace}Process</Component>
```

此示例指示，会将为进程度量标准系列创建的所有项分类为一个“进程”组件。

重要说明！ 必须在 Data Aggregator 中定义此部分中列出的每个组件。如果在 Data Aggregator 中未默认定义某个组件，请在 <ComponentDefinitionList> 部分中定义该组件。

注意： 有关每个 XML 标记的详细信息，请参阅 MetricFamily.xsd 和 Component.xsd 文件中提供的行内注释式文档。有关代码示例，请参阅 ProcessInfoMFWWithComponent.xml 文件。此示例文件定义用于收集进程度量标准的度量标准系列。

详细信息：

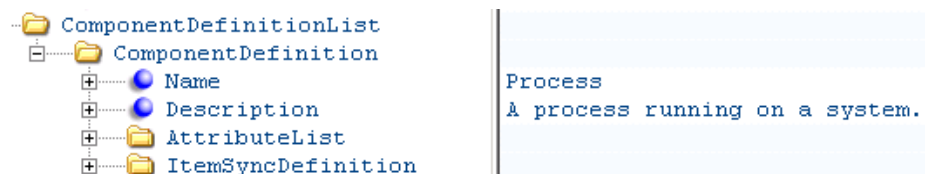
[创建自定义度量标准系列 XML 文件](#) (p. 37)

添加 <ComponentDefinitionList> 部分

<ComponentDefinitionList> 部分定义有关您的组件的详细信息。详细信息包括要存储在组件数据库中的更多属性的定义，以及组件及其属性与 CA Performance Center 同步的信息。

仅当满足以下条件时，才需要使用此部分：

- 您的自定义度量标准系列 XML 文件指定 <IsList_> 标记设置为 True 的属性。
- <ComponentList> 部分中的组件不是预定义的组件。



在自定义度量标准系列的此部分中包括以下标记：

<Name>

<Description>

<CertificationType>—始终将此标记设置为“自定义”。

AttributeList

（可选）定义了要在组件数据库中计算和存储的属性列表。通过将这些属性与 CA Performance Center 同步，使用组件属性使组件的更多信息（如进程组件的参数）在 CA Performance Center 中可用。本部分包括每个属性的以下标记：

名称

指定采用 QName 格式的唯一内部名称。

类型

指定此属性的数据类型。

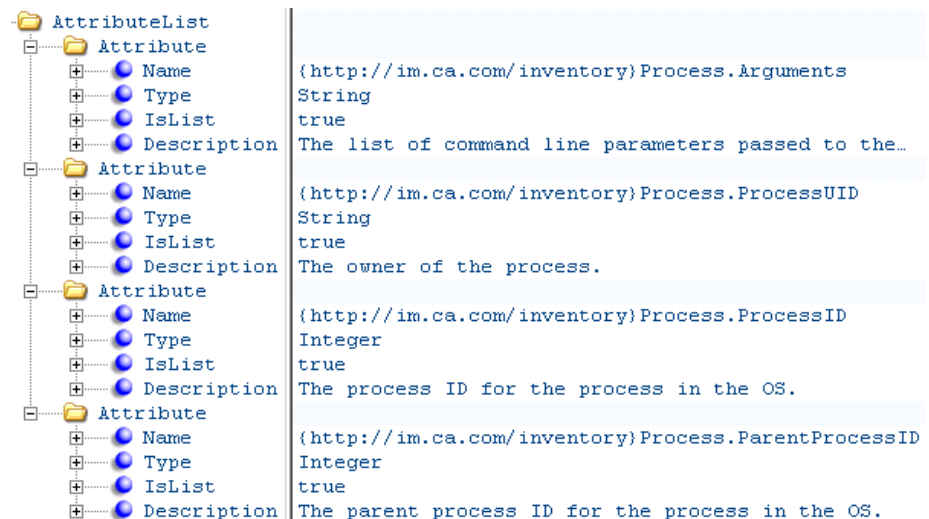
IsList

表明此属性是基于标量还是基于表。通常此值等于度量标准系列收集的相应属性的值。

说明

指定可在用户界面上显示的此属性的说明。

下图提供了使用 AttributeList 的示例：



ItemSyncDefinition

提供必要的详细信息，用于使组件与 CA Performance Center 保持同步。ItemSyncDefinition 也列出要在同步期间作为组件属性发送给 CA Performance Center 的附加属性，以便它们在 CA Performance Center 用户界面中可见。由于仅同步具有子类型的项，因此

<ItemSyncDefinition> 必须定义基本项类型和子类型。基本项类型可以用于创建逻辑组，并提供对所有子类型可用的默认信息和同步行为。在一些用例中，新的基本项类型可以改善性能。对于多数用例，它是“组件”项目类型。子类型项可提供更多特定于这些子类型的属性，必须对它们进行同步。例如，度量标准系列可以定义过程子类型的过程参数属性。此部分包括以下标记：

ItemTypeNames

定义项基本类型的名称，以便要在 CA Performance Center 中使用。

ItemSubTypeNames

定义项子类型的名称，以便在 CA Performance Center 中使用。

ItemTypeLabel

指定用户界面标签，以便在显示此类型的单个组件时使用。

ItemTypeLabelPlural

指定用户界面标签，以便在显示此类型的多个组件时使用。

IsDeviceComponent

指定此项是否为设备的组件（通常设置为 True）。在设为 True 时，此标记表示此组件列于“设备组件”视图中。

GroupBy

定义是否需要将 CA Performance Center 中此类型的组件分组。在设为 True 时，“清单”菜单下面的新菜单条目支持您将这些组件列在它们自己的组视图中。

Context

指定是否需要在 CA Performance Center 中为此类型的组件自动创建上下文类型（通常设置为 False）。

Categorize

指定是否需要在 CA Performance Center 中为此类型的组件创建动态组（通常设置为 False）。

Mapped

在多个 Data Aggregator 实例与 CA Performance Center 同步时，标识组件所属的 Data Aggregator 实例。

注意：由于该元素不受支持，因此要将其设置为 False。

ItemPropertyList

列出作为项类型的属性同步到 CA Performance Center 的所有属性。本部分包括每个属性的以下标记：

名称

定义 CA Performance Center 中此属性的内部名称。

注意：“名称”标记有 32 个字符的最大长度。请勿超出此限制。

Label

指定要在 CA Performance Center 用户界面中为此属性显示的标签。

AttributeName

引用在 CA Performance Center 同步期间填充此特性的组件属性。

Justification

指定显示此属性的 CA Performance Center 用户界面中的文本对齐。

默认：左对齐

DisplayWidth

定义列宽（以毫米为单位），以便在显示此属性的 CA Performance Center 用户界面中使用。

OrderBySQL

在多个 Data Aggregator 实例与 CA Performance Center 同步时，标识组件所属的 Data Aggregator 实例。

注意：由于该元素不受支持，因此要将其设置为 False。

DatabaseType/MaxLength

指定推断数据库类型的默认最大长度覆盖，以便用于此属性。

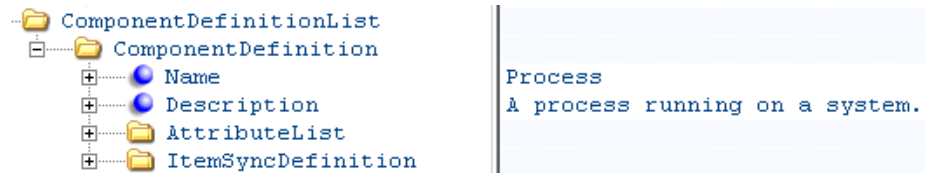
注意：有关每个 XML 标记的详细信息，请参阅 MetricFamily.xsd 和 Component.xsd 文件中提供的行内注释式文档。有关代码示例，请参阅 ProcessInfoMFWWithComponent.xml 文件。此示例文件定义用于收集进程度量标准的度量标准系列。

详细信息：

[创建自定义度量标准系列 XML 文件](#) (p. 37)

ComponentDefinitionList

如果您的自定义度量标准系列介绍了新组件项，例如下图所示的您在此部分中定义的进程。ComponentDefinition 为度量标准系列的组件的创建指定所有属性。特性包括要存储在组件数据库中的更多属性的定义，以及组件及其属性与 CA Performance Center 同步的信息。



与 CA Performance Center 同步

在 Data Aggregator 和 CA Performance Center 之间同步可确保组件项目显示在“清单”菜单中。而且，同步可在 CA Performance Center 中为这些项目分组，并针对它们的项目类型启用上下文页面。

Data Aggregator 将设备、接口和各种设备组件类型同步到 CA Performance Center。仅具有可以用于报告的数据的接口与 CA Performance Center 同步。这些包括正在积极监视的接口，以及有历史数据的退役接口。

Data Aggregator 还为具有一个或多个 IP 地址的任何接口同步 IP 地址。如果可用，Data Aggregator 同步给定接口 IP 地址的子网信息。

如果想使您的自定义度量标准系列的其他组件与 CA Performance Center 同步，您可以使用 <ItemSyncDefinition> 部分实现这一目的。本部分定义组件的类型和子类型，以及它们在 CA Performance Center 中的显示方式。

发现的设备和组件通常需要不超过 5 分钟的时间开始与 CA Performance Center 同步。当存在许多项时，完成同步所需的时间可能会超出 5 分钟。

当同步正在进行时，Data Aggregator 会进行等候，直到当前同步完成之后，才会同步新发现的设备和组件。

使“项属性”与超过 32 个字符的 <Name_> 标记值同步

ItemPropertyList 部分的 <Name> 标记值不应超过 32 个字符。然而，如果导致出现同步问题，您可以将项属性与超过 32 个字符的 <Name> 标记值同步。

请执行以下步骤：

1. 删除度量标准系列：
 - a. 找到 IMDataAggregator/apache-karaf-2.3.0/deploy 目录。
 - b. 删除您为所导入度量标准系列和供应商认证创建的 XML 文件。这些文件的名称如下：
 - im.ca.com-normalizer-<technology>.xml
 - im.ca.com-inventory-<technology>.xml
 - im.ca.com-certifications-snmp-<vendor>.xml
2. 使用以下命令重新启动 Data Aggregator：

```
/etc/init.d/dadaemon restart
```

重新启动 Data Aggregator 之后，请确认先前导入的度量标准系列或供应商认证不再显示在 CA Performance Center 中。此外，针对该自定义认证所有以前发现的组件都将删除。
3. 在 CA Performance Center 中单击“管理”，“数据源”。
4. 选择 Data Aggregator，然后单击“重新同步”按钮。

剩余度量标准系列的组成内容将在 Data Aggregator 和 CA Performance Center 之间同步。
5. 通过编辑您的自定义度量标准系列 XML 文件，使 <Name> 标记中的字符数少于 32 个字符。
6. 导入改正后的度量标准系列 xml 文件。

（可选）添加 <ComponentReconciliation> 部分

您可以在 <ComponentReconciliation> 部分中为您的自定义度量标准系列定义调整算法。调整算法用于支持受监视设备的配置更改。设备对给定度量标准系列的支持情况可视为设备的配置。例如，接口的数量以及接口的配置表示设备对接口度量标准系列的支持情况。在对设备的接口进行更改时，Data Aggregator 必须更新对这些接口的表示，以帮助确保监视是最新的。在此配置更新期间会应用调整算法。

为了支持某些度量标准系列，在 **Data Aggregator** 中创建了各个组件项。这些组件项表示支持给定度量标准系列的设备配置。以接口度量标准系列作为示例，**Data Aggregator** 创建端口组件项来表示设备的每个网络接口。

通过设备集合将监视配置文件应用于设备时，将进行组件项的初始发现和创建。为了支持受监视设备上的更改，会根据需要进行后续的组件发现。在进行后续组件发现之后，便会应用调整算法，以确定更新组件项的集合所需进行的更改。

当设备的配置发生更改时，有四种情况需要评估：

- 新组件—设备已更改，因此 **Data Aggregator** 中需要新的组件项。
- 未更改的组件—设备配置更改未改变某些现有组件，因此，这些组件项未更改。
- 更改的组件—设备配置改变了一些组件，但是它们仍然存在。**Data Aggregator** 的组件项必须更新，以反映新的配置。
- 删除的组件—设备发生更改，因此有一个或多个组件在设备中不再存在。这些组件项可以删除或报废。

调整算法定义了可在现有组件项与新发现结果之间比较的一组属性。通过对属性值进行比较来确定哪些新发现结果与现有的组件项匹配，哪些表示新组件或更改的组件。此比较将产生以下结果：

- 如果发现结果不与任何现有组件项匹配，会创建一个新组件项。
- 如果发现结果与现有组件项的所有属性都匹配，则组件项未更改。
- 如果发现结果发现匹配，但有些属性值不同，将用任何新属性值更新现有组件项。
- 有时，发现结果会确定有些现有组件项不以任何方式与新结果匹配。在这种情况下，可确定设备中不再存在这些组件，可将其删除或报废。

您可以在调整算法中定义两种匹配类型：**ExactMatch** 和 **BestofMatch**。

ExactMatch

指示发现结果必须与组件项的所有指定属性匹配。

示例：ExactMatch 调整算法

以下示例显示定义 ExactMatch 匹配类型的调整算法：

```
<ComponentReconciliation>
  <MatchAlgorithmList>
    <MatchAlgorithm>
      <AlgorithmType>Exact</AlgorithmType>
      <MatchAttributeList>
        <MatchAttribute>
          <Name>{http://im.ca.com/core}Item.Name</Name>
        </MatchAttribute>
        <MatchAttribute>
          <Name>{http://im.ca.com/inventory}Process.Path</Name>
        </MatchAttribute>
        <MatchAttribute>
          <Name>{http://im.ca.com/inventory}Process.Arguments</Name>
        </MatchAttribute>
      </MatchAttributeList>
    </MatchAlgorithm>
  </MatchAlgorithmList>
</ComponentReconciliation>
```

此算法表示发现结果必须匹配现有组件项的所有三个属性，以便完全匹配。如果找不到完全匹配，以下条件会生成下列结果：

- 如果三个属性值中的任一值不同，便会创建新组件项。
- 如果三个属性值都匹配，但其他属性值不同，便会用新属性值更新现有组件项。

BestofMatch

指定最少的属性数量，这些属性必须与现有组件项中的相同数量的属性匹配。每个属性都包括一个“必需”键。如果“必需”键设置为“true”，该属性必须是匹配的属性之一。

示例：BestofMatch 调整算法

以下示例显示定义 BestofMatch 匹配类型的调整算法：

```
<ComponentReconciliation>
  <MatchAlgorithmList>
    <MatchAlgorithm>
      <AlgorithmType>BestOf</AlgorithmType>
      <LeastMatchCount>2</LeastMatchCount>
      <MatchAttributeList>
        <MatchAttribute>
          <Required>true</Required>
          <Name>{http://im.ca.com/core}Item.Name</Name>
        </MatchAttribute>
        <MatchAttribute>
          <Name>{http://demo/custom}Process.Path</Name>
        </MatchAttribute>
        <MatchAttribute>
          <Name>{http://demo/custom}Process.Arguments</Name>
        </MatchAttribute>
      </MatchAttributeList>
    </MatchAlgorithm>
  </MatchAlgorithmList>
</ComponentReconciliation>
```

此算法指定以下要求：

- 发现结果必须与现有组件项至少匹配所列属性中的二个，才是匹配。如果找不到 BestofMatch 匹配，以下条件会生成下列结果：
 - 如果匹配的属性少于二个，便会创建新组件项。
 - 如果至少二个属性值匹配，但其他属性值不同，便会用新属性值更新现有组件项。

注意：匹配的属性值之一必须是必需的属性。

- 对于 Name 属性，“必需”键设置为“true”。这表示 Name 属性必须是匹配的三个属性之一。

在 BestOf 匹配算法包含在调整定义中时，会有以下结果：

- 如果发现结果仅与一个现有组件项匹配，则结果匹配。
- 如果发现结果与多个现有组件项匹配，会考虑匹配属性的数量。会将匹配属性最多的现有组件视为发现结果的匹配项。

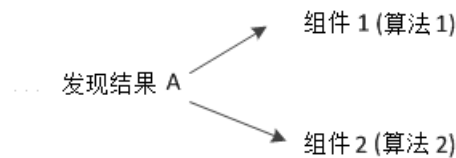
注意：如果多个组件的匹配属性数相同，会随机选择匹配组件项。因此，karaf 日志会记录一条关于此条件的警告。

可向调整算法中添加多个匹配算法。添加了多个调整算法时，它们将匹配优先级。MatchAlgorithmList 顶端的匹配算法有最高优先级。底部的匹配类型优先级最低。

当调整算法中包括多个匹配类型时，会有以下结果：

- 如果发现结果只与一个现有组件项匹配，则该项为匹配项。算法优先级无关紧要。
- 当发现结果与多个现有组件项匹配时，则优先采用具有最高优先级的算法。匹配将转向由优先级最高的算法生成的组件项。

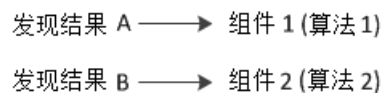
例如：



由于算法 1 的优先级较高，因此发现结果 A 与组件 1 匹配。

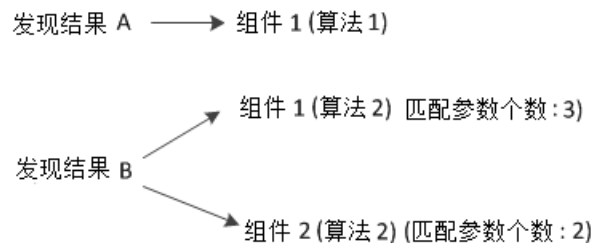
- 当多个发现结果与一个现有组件项匹配时，则优先采用具有最高优先级的算法。匹配将转向根据优先级最高的算法认定匹配的组件项。

例如：



由于算法 1 的优先级较高，因此发现结果 A 与组件 1 匹配。

例如：



在此示例中，发现结果 B 与组件 1 匹配的属性多于与组件 2 匹配的属性 2.发现结果 A 与组件 1 匹配，因为它使用优先级较高的算法来匹配。因此，组件 1 的匹配会转至发现结果 A，发现结果 B 匹配组件 2.2

注意：如果未定义调整算法，Data Aggregator 将使用 Item.Name 属性匹配组件。

详细信息：

[创建自定义度量标准系列 XML 文件](#) (p. 37)

（可选）添加 `<ReconfigDetectionAttr>` 部分

您可以使用度量标准系列中的专用数量属性来检测度量标准系列变化。与添加 `<componentReconciliation>` 部分相比，此方法更为有效；它执行效果更好，并且生成的网络流量要少得多。仅轮询标量属性，以确定是否存在更改。

如果度量标准系列有一个标量属性用于检测更改，则可在 `<ReconfigDetectionAttr>` 部分中指定该标量属性。

示例：在接口度量标准系列中配置更改检测

1. 指定更改检测标记 `<ReconfigDetectionAttr>`，以查看 `PortReconfig` 属性的更改：

```
<ReconfigDetectionAttr>
  {http://im.ca.com/normalizer}NormalizedPortInfo.PortReconfig
</ReconfigDetectionAttr>
```

详细信息：

[创建自定义度量标准系列 XML 文件](#) (p. 37)

添加 `<ExpressionGroupList>` 部分

表达式告诉 **Data Aggregator** 如何为您的度量标准系统中定义的或与您的度量标准系列相关的项目的属性计算值。自定义度量标准系列中包括的表达式将决定这些值在报告和显示板视图中的显示方式。表达式包含在 `<ExpressionGroupList>` 部分中。

如果您的度量标准系列的属性是表属性，您的度量标准系列 XML 必须包括 `<ExpressionGroupList>` 部分，并且至少对于以下两个相关项目（目标认证），必需一个表达式组：

- `{http://im.ca.com/core}Item`
- `{http://im.ca.com/inventory}DeviceComponent`

您还必须为自定义度量标准系列中定义的每个组件定义一个表达式组。
<ExpressionGroupList> 部分中的标记如下所示：

- <ExpressionGroup>—为项 DeviceComponent 和自定义度量标准系列监视的每个组件类型创建这些标记之一。此标记包括以下标记：
 - <DestCert>—表达式所作用的相关项目的名称。
 - <ExpressionList>—此标记包含为组中的属性定义的表达式。此标记包括以下标记：
 - <Expression>—在每个表达式列表中至少包括其中一个标记。这些标记定义如何计算给定属性的值。此标记包括以下标记：
 - <DestAttr>
 - <Expression>

注意：有关每个 XML 标记的详细信息，请参阅 MetricFamily.xsd 和 Component.xsd 文件中提供的行内注释式文档。有关代码示例，请参阅 ProcessInfoMFWithComponent.xml 文件。此示例文件定义用于收集进程度量标准的度量标准系列。

详细信息：

[创建自定义度量标准系列 XML 文件](#) (p. 37)

ExpressionGroupList

下列信息描述了如何计算属性列表中的元素。通常计算是琐碎的分配，如进程 ID。

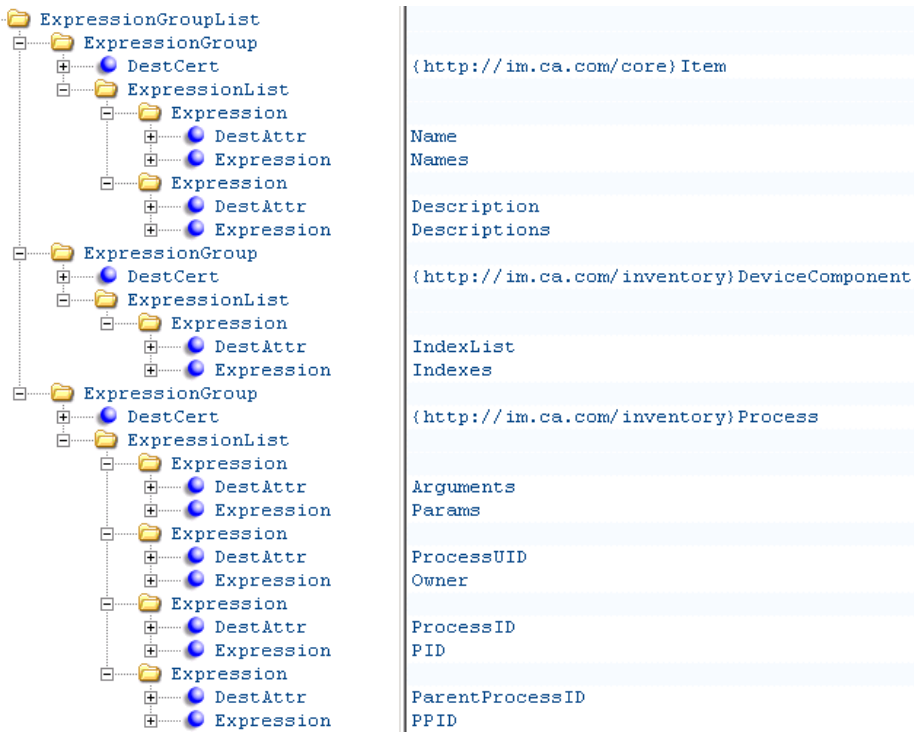
注意：请勿混淆度量标准系列和供应商认证 ExpressionGroup。度量标准系列采用下列格式公开了 URI 的属性：

```
{http://im.ca.com/normalizer}Name.AttributeName
```

供应商认证 ExpressionGroup 中涉及到了该属性。

度量标准系列 ExpressionGroup 为数据库填充了利用 DestCert 和 DestAttr URI 处理的不同项。计算表达式中的值，通常为属性列表中元素的琐碎分配。

例如，下图显示了如何将 {http: //im.ca.com/inventory}Process.ProcessID 从度量标准系列 AttributeList 映射到 PID 属性。



必须存在以下 DestCert URI:

DestCert URI	DestAttr
{http://im.ca.com/core}Item	名称
{http://im.ca.com/core}Item	说明

DestCert URI	DestAttr
{http://im.ca.com/inventory} DeviceComponent	IndexList
{http://im.ca.com/inventory} component	ComponentDefinitionList (p. 54) 中定义的属性名称。在示例中，组件进程提供了属性参数、ProcessUID、ProcessID 和 ParentProcessID。

在测试环境中确认您的自定义度量标准系列结果

在完成自定义度量标准系列 XML 并将其导入测试环境后，请验证结果。验证是一个重要过程，有助于确保自定义度量标准系列可在生产环境中生成正确结果。

重要说明！ 请始终先在测试环境中创建和验证您的自定义度量标准系列。创建自定义度量标准系列需要您手工编辑度量标准系列 XML 文件。该 XML 文件中的语义错误可能会导致不可预测的结果。

请执行以下步骤：

1. 使用自定义度量标准系列在测试系统中创建供应商认证。在创建供应商认证时验证以下结果：
 - a. 自定义度量标准系列 XML 文件出现在供应商认证向导中。
 - b. 选择自定义度量标准系列时，正确的度量标准系列属性便可见。
2. 验证供应商认证在发现期间是否创建了适当项。
3. 验证这些项是否轮询并收集您在度量标准系列 XML 文件中指定的度量标准数据。
4. 验证收集的数据是否正确。
5. 如果已配置项同步，则验证项数据是否与 CA Performance Center 正确同步。
6. 再次运行发现并验证是否已正确更新信息。

详细信息：

[如何创建自定义度量标准系列](#) (p. 35)

导入自定义度量标准系列 XML 文件

在您创建自定义度量标准系列 xml 文件并在测试环境中确认后，将其导入到您的 Data Aggregator 安装里。

请执行以下步骤：

1. 从 Data Aggregator 数据源的“监视配置”菜单中单击“度量标准系列”。
此时将显示度量标准系列列表，其中包括工厂和自定义度量标准系列。工厂度量标准系列显示锁定标志。
2. 单击“导入”。
3. 浏览以选择自定义度量标准系列 XML 文件，单击“打开”，然后单击“导入”。

将导入您的自定义度量标准系列。

重要说明！ 为避免可能发生的数据丢失，请一定要在每次创建或更新供应商认证、度量标准系列或组件时都备份您的 **deploy** 目录。

详细信息：

[如何创建自定义度量标准系列](#) (p. 35)

自动移除已报废的组件

作为管理员，您可以从网络中自动移除已报废的组件。在您编写脚本来自动化过程之前，了解如何使用包含在 Data Aggregator 中的脚本删除已报废的组件。例如，您可以每周设置定时任务，删除使用时间超过 1 个月的已报废组件。

包含在 Data Aggregator 中的 `remove_retired_items` 脚本由两个部分构成。脚本的第一部分将会根据您设置的筛选来标识和返回有关已报废组件的数据。脚本的第二部分将会发布已报废组件的删除列表。要使此过程自动化，请了解脚本的生成方式。

注意： 有关使用 `remove_retired_items` 脚本的信息，请参见《Data Aggregator 管理员指南》。

示例：按设备 IP 地址筛选已报废组件列表

在此示例中，您希望查找主 IP 地址是 10.252.1.1 的设备中所有已报废的组件。通过 IP 地址进行筛选是两步的过程，因为 IP 地址无法提供直接的组件筛选。如要筛选已报废的组件，请首先记录与组件关联的设备 IP 地址。凭借此 IP 地址信息，您可以为设备确定设备项 ID。之后通过设备项 ID，您可以确定已报废组件的用途。最后，您可以删除已报废的组件。

注意：此示例采用的是 curl 命令，但您可以采用任何熟悉的命令。

1. 创建 filterDeviceIP.xml 文件。您可以使用此文件返回主 IP 地址是 10.252.1.1 的设备项 ID。文件必须如下所示：

```
<FilterSelect xsi:noNamespaceSchemaLocation="filter.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Filter>
    <And>
      <Device.PrimaryIPAddress
type="EQUAL">10.252.1.1</Device.PrimaryIPAddress>
    </And>
  </Filter>
</FilterSelect>
```

2. 运行以下命令：

```
curl -X http://hostname:port/rest/devices/filtered -H "Content-Type:
application/xml" -T "filterDeviceIP.xml" > returnedDeviceID.xml
```

-X

创建您指示的筛选。

hostname:port

指定 Data Aggregator 主机名和端口号。

默认端口：8581

-H

指示您正在发布的文件内容类型。

-T

指示您正在发布的文件。

下列结果将作为 HTTP 响应返回：

```
<?xml version="1.0"?>
<DeviceList>
  <Device version="1.0.0">
    <ID>107881</ID>
    <PrimaryIPAddress>10.252.1.1</PrimaryIPAddress>
    <supportsOnDemandMFDIScovery>true</supportsOnDemandMFDIScovery>
    <SupportedProtocolsList>
      <SupportedProtocols>ICMP</SupportedProtocols>
    </SupportedProtocolsList>
    <DiscProfileID>107503</DiscProfileID>
    <HostName>rtp003723rts.ca.com</HostName>
    <RelatesTo>
      <MonitoredGroupIDList relatesURL="relatesto/monitoredgroups"
rootURL="monitoredgroups">
        <ID>509</ID>
      </MonitoredGroupIDList>
      <GroupIDList relatesURL="relatesto/groups" rootURL="groups">
        <ID>547</ID>
        <ID>530</ID>
        <ID>509</ID>
      </GroupIDList>
    </RelatesTo>
    <IsAlso>
      <IsA name="MetricFamilyDiscoveryHistory"
rootURL="devices/mfdiscoveryhistory"/>
      <IsA name="AccessibleDevice" rootURL="devices/accessible"/>
      <IsA name="Syncable" rootURL="syncable"/>
      <IsA name="IPDomainMember" rootURL="ipdomainmember"/>
    </IsAlso>
    <DataColectionMgrId version="1.0.0">
      <DcmID>dcname.ca.com:8f53bc55-f442-42fc-9bd5-a907d0261421</DcmID>
    </DataCollectionMgrId>
    <Syncable version="1.0.0">
      <SyncID>-1</SyncID>
    </Syncable>
    <Item version="1.0.0">
      <DisplayName>router.ca.com</DisplayName>
      <CreateTime>Wed Feb 05 10:20:26 EST 2014</CreateTime>
      <Name>router.ca.com</Name>
    </Item>
    <IPDomainMember version="1.0.0">
      <IPDomainID>2</IPDomainID>
    </IPDomainMember>
    <DeviceMonitoringProfile version="1.0.0">
      <ConsolidatedMonitoringProfile>2509</ConsolidatedMonitoringProfile>
    </DeviceMonitoringProfile>
  </Device>
</DeviceList>
```

将返回设备项 ID 107881。此结果还将显示有关设备的详细信息。

3. 创建 `filterRetired.xml` 文件。您可以使用此文件来返回与设备关联的已报废组件（设备项 ID 是 107881）。文件必须如下所示：

```
<FilterSelect xsi:noNamespaceSchemaLocation="filter.xsd"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <Filter>
    <And>
      <DeviceComponent.DeviceItemID
type="EQUAL">107881</DeviceComponent.DeviceItemID>
    </And>
  </Filter>
  <Select use="exclude">
    <Item use="exclude">
      <DisplayName use="include"/>
    </Item>
  </Select>
</FilterSelect>
```

4. 运行以下命令：

```
curl -X post http://hostname:port/rest/retired/filtered -H "Content-Type:
application/xml" -T "filterRetired.xml" > returnedRetireItems.xml
```

下列结果将作为 HTTP 响应返回：

```
<?xml version="1.0"?>
<RetiredList>
  <Retired version="1.0.0">
    <ID>128452</ID>
    <Item version="1.0.0">
      <DisplayName>GigabitEthernet0/239 -
GigabitEthernet0/239</DisplayName>
    </Item>
  </Retired>
  <Retired version="1.0.0">
    <ID>128451</ID>
    <Item version="1.0.0">
      <DisplayName>GigabitEthernet0/238 -
GigabitEthernet0/238</DisplayName>
    </Item>
  </Retired>
</RetiredList>
```

将返回两个符合筛选条件的已报废组件。组件的项 ID 是 128452 和 128451。

5. 创建 `deleteRetiredList.xml` 文件。您可以使用此文件来删除已报废组件的返回列表。文件必须如下所示：

```
<DeleteList>
  <ID>128452</ID>
  <ID>128451</ID>
</DeleteList>
```

6. 运行以下命令：

```
curl -X post http://hostname:port/rest/retired/deletelist -H
"Content-Type: application/xml" -T "deleteRetiredList.xml" >
deletelistreponse.xml
```

下列结果将作为 HTTP 响应返回：

```
<?xml version="1.0"?>
<DeleteListResult>
  <DeleteResult>
    <ID>128452</ID>
    <Error>SUCCESS</Error>
  </DeleteResult>
  <DeleteResult>
    <ID>128451</ID>
    <Error>SUCCESS</Error>
  </DeleteResult>
</DeleteListResult>
```

已成功删除报废的组件。

第 6 章：故障排除

此部分包含以下主题：

[查找有关错误消息的详细信息](#) (p. 69)

[故障排除：度量标准系列不完整](#) (p. 69)

[故障排除：度量标准系列不受支持](#) (p. 70)

查找有关错误消息的详细信息

症状：

Web 服务出现故障时，我收到了令人难解的错误消息。

解决方案：

要帮助调试该问题，请在信息的

IMDataAggregator/apache-karaf-2.3.0/data/log 目录中查找 karaf.log 文件。

在 Web 服务无法处理请求时，堆栈跟踪信息及更详细的错误消息将会打印在日志文件中。日志文件中通常包含导致问题的具体标记。

故障排除：度量标准系列不完整

症状：

我已成功导入自定义度量标准系列，但之后发现了有缺陷的度量标准定义。例如，<Name> 属性的最大长度为 32 个字符。如果超出此限制，则会导致同步问题。

解决方案：

删除自定义度量标准系列时务必谨慎，如下所示：

1. 找到 IMDataAggregator/apache-karaf-2.3.0/deploy 目录。
2. 删除为度量标准系列创建和部署的 XML 文件。这些文件的名称如下：
 - im.ca.com-normalizer-<technology>.xml
 - im.ca.com-inventory-<technology>.xml

如果适用，请同时删除为供应商认证创建的文件：

- im.ca.com-certifications-snmp-<vendor>.xml

3. 通过运行以下命令重新启动 Data Aggregator:

```
service dadaemon restart
```

重新启动 Data Aggregator 之后，请确认先前导入的度量标准系列或供应商认证不再显示在 CA Performance Center 中。此外，针对该自定义认证所有以前发现的组件都将删除。

4. 在 CA Performance Center 中单击“管理”，“数据源”。
5. 选择 Data Aggregator，然后单击“重新同步”按钮。
剩余度量标准系列的组成内容将在 Data Aggregator 和 CA Performance Center 之间同步。
6. 编辑并更正您的自定义度量标准系列 XML 文件。
7. 导入改正后的度量标准系列 xml 文件。

故障排除：度量标准系列不受支持

症状：

我创建了监视配置文件以在设备集合上轮询度量标准系列。但是，在“轮询的度量标准系列”表中，其中一个度量标准系列处于“不受支持”状态。

解决方案：

要更正问题，请执行以下步骤：

1. 确认轮询的设备对 SNMP 查询作出响应。
2. 单击不受支持的度量标准系列即可导航到该位置。
3. 确认供应商认证支持度量标准系列。如果未定义供应商认证，请创建一个自定义供应商认证。
4. 确认所有关键供应商认证属性在设备上受支持。如果支持所有关键供应商认证属性，请导航回该设备，并选择已为其添加自定义供应商认证的度量标准系列，然后单击“更新度量标准系列”。

将更新您的设备配置。