CONTENTS

Figures 7

Tables 9

Part 1  Data Enrichment 11

Chapter 1  Data Enrichment Introduction 13
About data enrichment .................................................................14
User interfaces ..............................................................................14
Data enrichment uses and examples ..............................................15
How data enrichment works ......................................................18

Chapter 2  Groups Management for Data Enrichment 21
Summary of groups .........................................................................22
Access Groups Management UI .....................................................23
Flat and hierarchical groups .......................................................24
Overview of the Groups Management UI ......................................24
  Use the preview table .................................................................26
  Create a group ...........................................................................27
  Edit a group .............................................................................31
  View and reorder rules ..............................................................31
  Create new group by copying an existing group .......................33
  Get group membership rules as regex expression .....................33
  Manage groups by exporting and importing files .....................33
  Groups Management limitations .................................................34
Detailed group descriptions .........................................................35
  Service Level by LUNs ..............................................................35
  Service Levels by File Share ......................................................37
  Bucket Service Level ...............................................................39
  ECS Capacity Rates ..................................................................40
  Block Chargeback Grouping .....................................................42
  File Chargeback Grouping .......................................................44
  Platform ..................................................................................46
  Customer, location, and business unit .......................................47
Video: Use Groups Management to populate and use the business unit
  property ..................................................................................48

Chapter 3  Raw Data Enrichment 49
About raw data enrichment ..........................................................50
Guidelines for new property names ..............................................50
Supported accessors for getting data .............................................51
Using the Data Enrichment UI .....................................................51
  Register a Collector Manager .....................................................52
  Create new tagging structure .....................................................54
  Populate the tag set rows ..........................................................59
  (Optional) Run the import property task and restart Collectors ... 63
  Verify the new property .............................................................64
<table>
<thead>
<tr>
<th>FIGURES</th>
<th>DESCRIPTION</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Example of populated customer filter</td>
<td>15</td>
</tr>
<tr>
<td>2</td>
<td>Example service level report</td>
<td>17</td>
</tr>
<tr>
<td>3</td>
<td>Example Platform group filter</td>
<td>18</td>
</tr>
<tr>
<td>4</td>
<td>ViPR SRM collection process</td>
<td>19</td>
</tr>
<tr>
<td>5</td>
<td>Role of Property-Tagging-Filter</td>
<td>20</td>
</tr>
<tr>
<td>6</td>
<td>Groups Management UI</td>
<td>25</td>
</tr>
<tr>
<td>7</td>
<td>Groups Management UI—Rules</td>
<td>26</td>
</tr>
<tr>
<td>8</td>
<td>Predefined groups for Service Level by LUNs</td>
<td>36</td>
</tr>
<tr>
<td>9</td>
<td>Prepopulated rule example for Service Level by LUNs</td>
<td>36</td>
</tr>
<tr>
<td>10</td>
<td>Prepopulated group for Service Level by File Share</td>
<td>38</td>
</tr>
<tr>
<td>11</td>
<td>Prepopulated rule for Service Level by File Share</td>
<td>38</td>
</tr>
<tr>
<td>12</td>
<td>Predefined groups for Bucket Service Level</td>
<td>39</td>
</tr>
<tr>
<td>13</td>
<td>Predefined rule example for Bucket Service Level</td>
<td>39</td>
</tr>
<tr>
<td>14</td>
<td>Predefined group for ECS Capacity Rates</td>
<td>41</td>
</tr>
<tr>
<td>15</td>
<td>Predefined rule example for all S3 buckets</td>
<td>41</td>
</tr>
<tr>
<td>16</td>
<td>Predefined Block Chargeback group names</td>
<td>43</td>
</tr>
<tr>
<td>17</td>
<td>Predefined File Chargeback group names</td>
<td>45</td>
</tr>
<tr>
<td>18</td>
<td>Example custom block chargeback report</td>
<td>111</td>
</tr>
<tr>
<td></td>
<td>Table Description</td>
<td>Page</td>
</tr>
<tr>
<td>---</td>
<td>-----------------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>1</td>
<td>Summary of predefined group types</td>
<td>22</td>
</tr>
<tr>
<td>2</td>
<td>New properties for Service Level by LUNs</td>
<td>37</td>
</tr>
<tr>
<td>3</td>
<td>New properties for Service Level by File Share</td>
<td>38</td>
</tr>
<tr>
<td>4</td>
<td>New properties for Bucket Service Levels</td>
<td>40</td>
</tr>
<tr>
<td>5</td>
<td>New properties for ECS (S3) Capacity Rates groups</td>
<td>42</td>
</tr>
<tr>
<td>6</td>
<td>New property for Block Chargeback Grouping</td>
<td>43</td>
</tr>
<tr>
<td>7</td>
<td>New property for File Chargeback Grouping</td>
<td>45</td>
</tr>
<tr>
<td>8</td>
<td>New property for Platform Group</td>
<td>47</td>
</tr>
<tr>
<td>9</td>
<td>New properties for customer, location, and business unit</td>
<td>48</td>
</tr>
<tr>
<td>10</td>
<td>Location of Groups Management Configuration Files</td>
<td>88</td>
</tr>
<tr>
<td>11</td>
<td>Shared disk with no storage used</td>
<td>98</td>
</tr>
<tr>
<td>12</td>
<td>Shared disk with some storage used</td>
<td>98</td>
</tr>
</tbody>
</table>
PART 1

Data Enrichment

VIPR SRM provides several methods for enhancing (enriching) collected data.

Chapter 1, "Data Enrichment Introduction"
Chapter 2, "Groups Management for Data Enrichment"
Chapter 3, "Raw Data Enrichment"
Chapter 4, "Reporting Features to View Enrichment Results"
Chapter 5, "Advanced Topics"
Chapter 6, "Troubleshooting"
CHAPTER 1

Data Enrichment Introduction

The following topics introduce data enrichment features and provide a summary of how it works.

- About data enrichment ................................................................. 14
- User interfaces ............................................................................. 14
- Data enrichment uses and examples .......................................... 15
- How data enrichment works ....................................................... 18
About data enrichment

The ViPR SRM data enrichment features provide ways for end-users to expand, or enrich, collected data with additional properties. The new property values are stored in the database along with collected metrics and are available for use in reports and filters.

Enrichment rules define when to tag collected data with a new property by matching key properties in the collected data. The rules define the match values and the resulting new property values.

In most cases, the rules are user-defined. The user-defined nature of data enrichment means that enterprises can label (tag) collected data for their own internal uses, or insert information from other sources, including other databases, into the ViPR SRM database.

In addition to the user-defined use cases, ViPR SRM uses data enrichment features to implement its own features. Examples are for service level capacity reporting and chargeback reporting. For these uses, the associated Solution Packs provide predefined rules and values. There are custom opportunities available for each of these as well.

User interfaces

ViPR SRM includes multiple interfaces for implementing data enrichment.

Groups Management UI

The Groups Management interface provides predefined properties and a predefined structure for assigning values. The Groups Management interface is a quick way to enhance the database with customized data if the predefined properties (groups) fit your needs.

Users add the new property values directly on the UI, as group names and other editable fields. Buttons are available for exporting or importing the CSV file that contains the property values.

Collected records are filtered using group membership rules that you configure in the Groups Management UI. Matched records are enriched with the predefined new properties, using values as configured in the Groups Management UI.

Data Enrichment UI

The Data Enrichment interface is an open-ended interface that lets users create new properties and the structure for assigning values. This interface can enrich the APG database or the Events (Alerting) database.

Users create the match rules and add the new property values directly on the UI. Buttons are available for exporting or importing the CSV file that contains the property values.

Collected records are filtered using key fields and values that you configure in the Data Enrichment UI. Matched records are enriched with the new property names and values as configured in the Data Enrichment UI.

Customized Property-Tagging-Filters

A Property-Tagging-Filter (PTF) is a component inserted into the collection chain that defines new properties and the rules for assigning values. XML configuration
files define the new properties and the external source for obtaining the values. The external source can be a file, a database, a web service, and others.

Installing and configuring a customized PTF is an advanced development activity.

Note

The Groups Management and Data Enrichment UIs configure PTFs in the background. Both UIs configure the PTF to expect a CSV file as the source of the rules data.

Data enrichment uses and examples

Here are illustrations of a few data enrichment uses.

Populate existing groups

Suppose you want to filter capacity reports for a specific customer, location, or business unit, or maybe you need to create customized reports based on one or more of those attributes. Assuming that these attributes can be inferred from collected values, such as array names, you can achieve these goals by using either the Groups Management UI or the Data Enrichment UI to populate values into the customer, location, or business unit properties. Then use the predefined group filters at the top of many reports to filter the report for the desired property values. Another option is to display a column for the populated property.

For example, here is the Customer filter populated with a few values, with a filter set to show enterprise capacity for a single customer value.

Figure 1 Example of populated customer filter

Predefined columns for Customer, Location, and Business Unit are available, but hidden by default, in many reports. Users can easily unhide these columns.
Add new properties to the database and create rules to populate the values

Suppose you want to add contact information for all devices into reports. Assuming that the contact information is based on collected attributes, such as vendor and model, you can achieve this goal using the Data Enrichment UI. You can create the new property and a set of rules for populating that property based on vendor and model. You can then upload a CSV file that contains the match information. As the final step, you can quickly edit a report definition to add a column for the new property.

Here is a customized report showing contact information that was inserted into the database by a data enrichment configuration.

Populate new properties from external sources

Suppose you have an external database that maintains rack locations, serial numbers, and warranty information, per device. By manually installing a new Property Tagging Filter (PTF) process and configuring it with a JDBC accessor and tagging rules, the PTF can connect to the external database to obtain the desired information and add it to collected data in ViPR SRM. You can create a new report that combines collected information with the new properties.

ViPR SRM features that depend on data enrichment

ViPR SRM uses data enrichment to implement the following features.
Service level classifications

Service level assignments are based on rules that classify storage based on characteristics of the storage. For some types of storage, administrators can edit the rules to customize the service levels or add new service levels.

Here is the Service Level Capacity report, showing existing storage (collected metrics) classified into service levels by data enrichment processing.

Figure 2 Example service level report

You can drill down to the underlying details of each service level by clicking a row.

Chargeback reports and rates

The Chargeback reports are based on rules that classify storage assets into groups specifically for the purpose of chargeback reporting.
Chargeback costs are calculated per service level within a chargeback group. The rates per GB are user-defined for each service level.

Here is a Block Chargeback report for a custom group of hypervisors.

![Block Chargeback report](image)

**Group filters on reports**

Selected reports contain group filter buttons in the header that enable easy filtering by selected property values. Common out-of-the-box filters are for Customer, Location, Business Unit, and Platform.

Platform is prepopulated with values by your installed SolutionPacks.

Here is the All Alerts report with the Platform filter.

![All Alerts report with Platform filter](image)

**How data enrichment works**

The ViPR SRM collection process normalizes data coming from many different sources into a common set of properties. Data enrichment inserts additional properties into the database after the collection and normalization process.

**Basic collection process**

Collectors collect data from various sources. The collected data is processed and tagged with standard properties before getting forwarded to the database. The rules for tagging the collected data with standard properties ensure that the collected data
conforms to the ViPR SRM data model. These rules are specified in each Collector's configuration files. This basic collection process ensures conformity across all SolutionPacks and provides the basis for standardized reporting. The following figure summarizes the collection process.

**Figure 4** ViPR SRM collection process

Administrators and power-users can add more information to what is already collected by inserting additional tagging rules into the process. The additional information might be business data or extra information not present or available in the collected entity. The new information is added to the collected information and is available in the database. You can then leverage this information in the user interface by making slight customizations to existing reports or creating new reports, if needed.

**Property-Tagging-Filter**

An EMC M&R component called a Property-Tagging-Filter (PTF) applies the additional properties to collected records. A PTF is associated with one or more Collectors. The PTF role is to read possible property matches from an input source, try to detect them in the Collector's input stream, and tag the matched records with additional properties before the records are forwarded to the database.

The input source to a PTF can be a text file, another database, a web service, or other external source. The following figure shows the PTF inserted into a collector manager processing stream. In this case, the PTF is reading an input file that was derived from an external database.
Tag sets
A PTF configuration file contains one or more tag sets. A tag set defines:

- The collected data fields to match against (key properties)
- The new properties to add when a match is found (new properties)
- The source for obtaining the key values to match against and the associated new property values to add when a match occurs

For example, given a device serial number (a key), a PTF can tag collected records with a rack number or a site name (new properties). The serial numbers and rack location values are obtained from an external source, such as a .csv file or an external database.

User Interfaces for configuring PTFs
ViPR SRM includes two interfaces for configuring preinstalled PTFs. These are:

- **Administration > Centralized Management > Groups Management.** This interface is useful for simple tagging tasks where the properties and tag set rules are already defined. Users only need to supply the matching values for keys and resulting new property values.
- **Administration > Centralized Management > Data Enrichment.** This interface is for more advanced tasks where users can define tagging sets from scratch.

Each of the above interfaces uses a separate, preinstalled, PTF component. These user interfaces require a file as the input source.

Configuring a PTF for accessors other than file accessors
Developers can manually install and configure a new Property-Tagging-Filter component. The manually configured PTF is required to implement a data enrichment solution that uses a non-file source, such as another database or a web service.
CHAPTER 2
Groups Management for Data Enrichment

The following topics describe how to enrich data in ViPR SRM by using the features in the Groups Management UI.

- Summary of groups.................................................................22
- Access Groups Management UI .............................................23
- Flat and hierarchical groups...............................................24
- Overview of the Groups Management UI ..............................24
- Detailed group descriptions................................................35
- Video: Use Groups Management to populate and use the business unit property ............................................48
Summary of groups

The Groups Management UI supports a predefined set of group types, each with a predefined purpose. It is important to use and populate each group type according to its intended purpose.

The following table contains a high-level description and intended purpose for each of the group types supported by the Groups Management UI. Each group type tags collected data with one or more additional properties.

**Note**

Your installation might not contain all of these group types. They are installed by the SolutionPacks that use them.

<table>
<thead>
<tr>
<th>Group type</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Business Unit</strong></td>
<td>These three properties are used for filtering and reporting based on customer-defined attributes. Many reports contain predefined (but hidden) columns for these properties. Those same reports include predefined group filters for these properties, for easy filtering by selected customer, location, or business unit values. Installations can use these three properties for any desired purpose. For example, to track storage usage by applications, you might populate the Business Unit property with application names, and set up the rules to filter on storage assets used by specific applications.</td>
</tr>
<tr>
<td><strong>Location</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Customers</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Platform</strong></td>
<td>This property is used for filtering and reporting based on platform type. Each installed SolutionPack adds to the list of predefined Platform values. Many reports contain the Platform group filter, for easy reporting by selected platforms.</td>
</tr>
<tr>
<td><strong>Service Level by LUNs</strong></td>
<td>These properties are used for Service Level Agreement (SLA) classifications for global reporting by service levels. In addition, a cost per GB property, required for chargeback reporting, is associated with each service level. Predefined service levels are installed; users can create customized service levels as well, with restrictions.</td>
</tr>
<tr>
<td><strong>Service Level by File Share</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Object Service Level</strong></td>
<td></td>
</tr>
<tr>
<td><strong>Block Chargeback Grouping</strong></td>
<td>These properties define rules that group storage assets for the purpose of global chargeback reporting. The chargeback reports show costs by service levels within each grouping. Predefined groupings are installed; users can create customized groupings for reporting purposes.</td>
</tr>
<tr>
<td><strong>File Chargeback Grouping</strong></td>
<td></td>
</tr>
</tbody>
</table>
Table 1 Summary of predefined group types (continued)

<table>
<thead>
<tr>
<th>Group type</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>ECS Capacity Rates</td>
<td>This group defines several properties used for billing rates for AWS S3 object storage managed on ECS assets. The billing reports appear under Report Library &gt; Amazon AWS &gt; Operations.</td>
</tr>
</tbody>
</table>

**Access Groups Management UI**

You access the Groups Management UI under Centralized Management.

**Procedure**

1. Go to Administration > Centralized Management > Groups Management.
   A list of group types appears, showing the groups supported by SolutionPacks installed at your site. For example:

   ![Groups Management](image)

   - List of all existing groups.

   - Showing 1 to 9 of 9 entries

   - Group Name: Block Chargeback Grouping, Bucket Service Levels, Business Service Levels, Customers, File Chargeback Grouping, Location, Platform, Service Level by File Share, Service Level by LUNs.

2. Click the row of the group type you want to edit or view.
Flat and hierarchical groups

A group type is either flat or hierarchical.

Flat groups
In flat groups, collected data can match only one rule and the property being added can have only one value per matched database entry. The Service Levels by LUNs is an example of a flat group. The Service Level for a given LUN can be gold or platinum, but not both. The Service Level property (psvclvl) can store only one value.

In a flat group type, the group names appear in a simple list. The order of the group names is important. The matching is performed in the order of the groups listed, and the first match wins.

Hierarchical groups
In hierarchical groups, collected data can match one or more rules. As a consequence, the new property can store one or more values. These values are separated by pipes (|).

Platform is an example of a hierarchical group type. A given device can match the rules for All Hosts and also for Unix Hosts. Therefore, the stored property value could be "All Hosts|unix".

Note
A subgroup does not inherit the rules from its parent. The underlying rules must be constructed correctly to support the hierarchy.

The ordering of the rules does not matter in a hierarchical group.

Overview of the Groups Management UI

For each group type, you can create new group name values and edit the rules of membership in the groups.

The following image shows the Groups Management UI for the Block Chargeback group type.

Note
The interface is slightly different for hierarchical and flat groups but the concepts are the same.
The group type represents predefined properties in the database. In this case, Block Chargeback Grouping represents the `nodegrp` property already defined in the database.

The action buttons are used to manage the group of property values. For example:
- **Create** creates a new possible value for the `nodegrp` property.
- **Edit** edits the rules of membership associated with a property value.

The set of actions is different for flat and hierarchical group types.

The group names are the possible values of the data enrichment property. In this example, the `nodegrp` property can have the value All Hypervisors, All Hypervisors - Compliance, or any other value in the hierarchy.

The preview table lists a partial view of objects in the database that match the rules for a selected value. For example, with All Hypervisors selected, the preview table lists hypervisors but not physical hosts or VM guests.

These actions manage the underlying CSV file associated with this group type.

Select a group name and then click **Edit** to see the match rules for that group name. For example, select All Hypervisors and click **Edit** to see the rules that control when the value All Hypervisors is assigned to the `nodegrp` property in a collected entry.
Use the preview table

The preview table shows the members of a selected group and some additional information about the members.

Procedure

1. Go to Administration > Centralized Management > Groups Management > *group_type*.
   
The page opens with the preview table showing the first 10 members of the first group on the page.

2. To see more members, use the Show entries drop-down box to adjust the number of entries displayed.

3. In cases where there are potentially many members, search for specific members using the Search text box.

   Type an exact value for any of the columns in the table, or use the * wildcard for a partial value search. The search operates on any of the columns in the preview table.

   For example, the search in the following figure (PowerEdge*) finds the members of the All Hypervisors group that match a device model starting with PowerEdge.
The wildcard can appear anywhere in the search phrase. For example, the search in the following figure (**lss**) finds all entries with the character sequence of `lss` anywhere in a value.

4. To preview another group, click to select the desired group name.

**Create a group**

The **Create** action creates a new property value and specifies the match rules for when that value is tagged to collected data.

Following is a general procedure for creating a group. Some group types, because of their intended purpose, have logical restrictions on the membership rules. For these and other limitations, see the individual group descriptions.

**Procedure**

1. Go to **Administration > Centralized Management > Group Management > group_type**.
2. In a hierarchical group type, click the parent under which you want the new group to appear.

**Note**

This step does not apply to flat groups.
3. Click **Create**.

4. In the first text field, enter the new group property value.
   
   For example, to create a new Customer value, enter the Customer name. This is the value that will be added to tagged records.

5. If additional text boxes are displayed for additional properties, enter those values.
   
   For example, the service level groups display the Cost per GB field.

6. For flat or hierarchical groups, add dynamic rules to define the records that will be tagged with the new property values.

   **Note**
   
   For flat groups, dynamic rules are the only way to add members. For hierarchical groups, the UI shows tabs for **Dynamic Members** and **Static Members**. Click **Dynamic Members**.

   A dynamic member definition consists of one rule or multiple rules ANDed together to create a rule set. You can OR multiple rule sets to create a detailed filter.

   The following steps provide guidelines for creating dynamic rules.

   a. Click **Select a property**,

   ![Select a property](image)

   Do either of the following:

   - Select one of the suggested property descriptions. The actual property name is automatically provided in the next text box. For example, if you select **Device type**, the property name `devtype` appears in the text box.
   - Select **Advanced**, and then provide a database property name in the next text box.

   b. In the second drop-down list, select an operation. Select **Wildcard** to use wildcards in the value.
c. In the last text box, provide the property value.

Click in the box to see suggested values based on the property named in the first text box. The suggestions are obtained from existing discovered values in your infrastructure.

d. To AND another rule, click Add Rule inside the ruleset box.

e. To OR another ruleset, click Add new ruleset outside of the existing ruleset box.
f. To verify defined membership, click Show Members.

7. For hierarchical groups, optionally add static members.

Static members are those that you select from a list and manually add to the membership.

**Note**

Static and dynamic members are combined to create the total membership for a group.

a. Click the Static Members tab.

b. Make selections from the Type or Source Group drop-down lists to populate the preview table. This creates a subset of items to choose from.

c. To add a static member to the group, select a row in the preview table and click Add to group.

The selection moves to the static member list.
8. Click Save.

Edit a group

Edit a group to change the membership rules for the group or change the new property values associated with the group.

Procedure

1. Go to Administration > Centralized Management > Group Management > group_type.
2. In the right pane, select the group to edit by clicking it.
3. Click Edit.
4. Change the property values in the text boxes at the top of the page, if needed.
5. Change or add to the membership rules, as needed.
6. Click Save.

Results

If you change a group name, all new tagging after the change uses the new group name. Regarding existing records in the database:

- if you changed only the group name, the change is applied the next time the PTF process runs, and the old group name is overridden by the new group name.
- If you changed both the group name and the rule definition, you will see both the old and new names in reports for a while, until the old group name ages out of the reports.

View and reorder rules

For flat group types, the order of the group names affects the order that the rules are processed and the tagging outcome. You can review all rules for all groups, and reorder the groups to change the processing order of the rules.

The Reorder and View All Rules buttons only appear on the dialogs for flat groups.

The new properties in flat groups can contain only one value. The first group name whose rules match the collected data is assigned. Make sure to order the groups so the most restrictive groups are first.

For example, in the list of service level groups, System Resource and Pool Contributor identify specialized storage uses. After that, Platinum rules are most restrictive. At the
bottom of the list, Other is very general and acts as the catchall group when no other rules match.

Procedure

1. Go to **Administration > Centralized Management > Groups Management > group_type**.
   This procedure is relevant to flat groups only.
2. To view all rules for all groups, click **View All Rules**.
3. To exit the list of rules, click **OK**.
4. To reorder the rules, click **Reorder**.
5. Click the group whose placement you want to change.
6. Use the arrows to change that group’s place in the list.
   Make sure that the more restrictive groups appear before the less restrictive groups.
7. Click Save.

Create new group by copying an existing group

In hierarchical group types, you can quickly create a new group with almost the same rules as an existing group by copying and pasting the existing group.

The Copy and Paste buttons are only available on the dialogs for hierarchical group types.

Procedure
1. Go to Administration > Centralized Management > Groups Management > group_type.
2. Click to select the existing group that you want to copy, and then click Copy.
3. Click to select the group in the hierarchy that will be the parent of the new group, and then click Paste.
   The new group appears under the selected group. It has the same name as the copied group.
4. Click to select the new group, and then click Edit.
5. On the Edit dialog, change the group name and alter or add to the rules and static member selections, as needed.
6. Click Save.

Get group membership rules as regex expression

You can see the complete membership rules for a group as a single regex expression.

Procedure
1. Go to Administration > Centralized Management > Groups Management > group_type.
2. Click to select the group name whose membership rules you want to see.
3. Click Edit.
4. Click Click to show query at the bottom of the Edit page.

Manage groups by exporting and importing files

As an alternative to creating and editing group names in the UI, developers or administrators can manage groups by exporting and importing files.

The Groups Management UI creates and maintains the underlying configuration files that support your group definitions. You can export and save those files to a location outside of ViPR SRM for backup purposes or to view and edit the settings. You can import an edited version into ViPR SRM.

Note

Editing files offline and importing them is an advanced activity. End users are recommended to edit group settings using the Edit and Create buttons on the Groups Management UI.

Procedure
1. Go to Administration > Centralized Management > Groups Management > group_type.
For example, click the **Service Level by LUNs** group type.

2. On the group type page, click **Export** and follow your browser's prompts to save the zip file.

3. Extract the files with an unzip utility.

   Each group type is supported by two files:
   
   **XML configuration file**
   
   The XML file defines the Property Tagging Filter (PTF) configuration for the group type. It includes:
   
   - The `<files>` element defines the name of the associated data input file (the CSV file), and the characters used in the file for delimiters and wildcards.
   
   - The `<key-properties>` element defines the structure of the rules that are used to match existing data in the database.
   
   - The `<new-properties>` element defines the names of the new properties to add when a match occurs.

   Typically, you do not change this file. Use it to interpret the format of the CSV file.

   **CSV data input file**
   
   The CSV file contains the group names and definitions in the acceptable format required by the PTF. Each line represents a group name, with all of the rules that define members of that group and the new property values for that group.

   For example, for the **Service Level by LUNs** group type, each line contains the defining rules for membership in a group, and, at the end of each line, the chargeback cost basis, the minimum and maximum response rates, and the service level name.

4. To import files:

   a. Make sure you have preserved an exported version of the existing settings.

   **Note**
   
   This step is important. You are about to overwrite the existing settings.

   b. Archive your edited CSV file and the XML file into a zip file. Both are required.

   c. Click **Import**, and choose the zip file to upload.

   d. Click **OK**.

**Groups Management limitations**

Note the following limitations related to the Groups Management UI.

**Deleted or changed group name remains in reports**

When a group name is changed or deleted, there are situations when the old name still exists as property values in the database, and as a result, still appears in reports and group filters.
In the case of a deleted group name, this situation occurs because there is no new value to substitute for the old value. The old values eventually age out of the reports.

In the case of a changed group name, the situation is temporary until the new value propagates throughout the system.

**Hierarchical groups do not inherit rules**

In hierarchical groups, a child group does not inherit the rules of the parent group.

To implement the implied hierarchical relationship, the user must make sure that the rules of a parent group are included in the rules of the child groups.

**Limitation on custom Service Levels by LUN groups**

The tagging rules for a custom Service Level by LUN group cannot use properties that are added to the database with data enrichment. This limitation applies whether you are editing rules for an existing Service Level by LUN group or creating a new Service Level by LUN group.

The limitation occurs because of the sequencing of the involved processes. The tagging for service levels occurs in the load-balancer connector (LBC) before the data enrichment tagging occurs in the Collector-Managers.

**NOTE:** A rule that uses data enrichment properties can appear to work in the Groups Management interface. That is, the **Show Members** button and the preview table both apply the membership rules to data that already exists in the database, including the data enrichment properties. However, the same rule will not work during actual processing of newly collected data because of the sequencing problem.

### Detailed group descriptions

The following topics show specific usage and property values for each group type.

**Service Level by LUNs**

The Service Level by LUNs groups classify LUNs based on disk type and performance characteristics.

This group type is installed with LUN-related SolutionPacks. It is installed with a set of predefined service level names and related rules that support common industry best practices for service level agreements. The following figure shows the predefined group names.
Membership in a group is based entirely on dynamic rules analysis. (Static members are not permitted.) All groups contain the rule that parttype is LUN. Other rules classify LUNs based on characteristics such as disk type and speed. The following figure shows one of the rules for classifying a LUN as Platinum. In this example, if parttype is LUN and disktype is Flash Drive and isfast is 0, then the LUN is tagged as Platinum.

New properties
The following new properties are added to tagged records.
Table 2 New properties for Service Level by LUNs

<table>
<thead>
<tr>
<th>Field name in UI</th>
<th>Property name in database</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost per GB</td>
<td>costgb</td>
<td>The cost is used to calculate chargeback costs. Decimals are allowed.</td>
</tr>
<tr>
<td>Minimum response time</td>
<td>minrtime</td>
<td>The response times are used in the service level agreement (SLA) reports. If a LUN's response time does not fall within these configured values for the service level, the LUN is violating the SLA.</td>
</tr>
<tr>
<td>Maximum response time</td>
<td>maxrtime</td>
<td></td>
</tr>
<tr>
<td>Service Level</td>
<td>svclevel</td>
<td>The service level name. This is the group name in the UI.</td>
</tr>
</tbody>
</table>

Permitted user changes
The following changes are possible for Service Level by LUNs:

- Edit a service level's Cost per GB, minimum response time, and maximum response time.

  Note
  You can not edit the shaded groups. The new property values do not apply to these groups, and the rules are fixed. To view the rules behind the shaded groups, click View All Rules.

- Create new service levels with the Create action.
- Delete a service level.

Note
The global Block Chargeback reports and the host-specific Associated Storage reports use the out-of-the-box service level by LUNs names. If you add or delete service levels, you might want to adjust the chargeback reports accordingly. See the Block Chargeback section for more information.

Service Levels by File Share

The Service Level by File Share group type tags all file systems and file shares into the same group.

This group type is installed with SolutionPacks related to file shares and file systems. It is prepopulated with one service level name (NAS). All file systems are members of this group. Although you can create additional groups, the file share chargeback reports can calculate chargeback metrics only for the NAS group.
Figure 10 Prepopulated group for Service Level by File Share

Membership in the group is based on the following rule.

Figure 11 Prepopulated rule for Service Level by File Share

New properties
Two new properties are added to tagged records.

Table 3 New properties for Service Level by File Share

<table>
<thead>
<tr>
<th>Field name in UI</th>
<th>Property name in database</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Service Level</td>
<td>svclevel</td>
<td>The service level name. This is the group name.</td>
</tr>
<tr>
<td>Cost per GB</td>
<td>costgb</td>
<td>The cost is used to calculate chargeback costs. Decimals are allowed.</td>
</tr>
</tbody>
</table>

Permitted user changes
The following changes are permitted for Service Level by File Share:

- Edit the Cost per GB for the NAS group.
- Create new groups for reporting purposes. (Only the NAS group is used for chargeback calculations.)
Delete a group.

Bucket Service Level

The Bucket Service Level group type classifies the buckets in EMC ECS namespaces into service levels.

This group type is installed by the SolutionPack for EMC ECS. Two service levels are predefined that classify object storage in ECS namespaces by replication policy, as follows:

- Remote Protected is assigned to buckets that are associated to multiple sites, meaning that data is copied offsite.
- Local Protected is assigned to buckets that are associated to one site, meaning that data is local only.

**Figure 12** Predefined groups for Bucket Service Level

Membership in a group is based on dynamic rules analysis. Here is the rule for the Local Protected service level.

**Figure 13** Predefined rule example for Bucket Service Level

**New properties**

The following new properties are added to tagged records.
### Table 4 New properties for Bucket Service Levels

<table>
<thead>
<tr>
<th>Field name in UI</th>
<th>Property name in database</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cost per GB</td>
<td>costgb</td>
<td>Used to calculate chargeback costs. The chargeback reports calculate all three cost types. Decimals are allowed.</td>
</tr>
<tr>
<td>Download Cost per GB</td>
<td>costdwld</td>
<td></td>
</tr>
<tr>
<td>Upload Cost per GB</td>
<td>costupld</td>
<td></td>
</tr>
<tr>
<td>Service Level</td>
<td>svclevel</td>
<td>The service level name. This is the group name on the UI.</td>
</tr>
</tbody>
</table>

#### Permitted user changes

The following changes are permitted for Bucket Service Levels:

- Edit a group’s cost-related property values.
- Edit membership rules.
- Create new groups—New service levels, with customized rules, are permitted. Chargeback costs for customized service levels are calculated and reflected on the global Object Chargeback reports.
- Delete groups.

#### ECS Capacity Rates

The ECS capacity rates group type defines groupings of S3 buckets and configures billing rates for each group.

This group type is installed by the SolutionPack for Amazon S3. One grouping is predefined that defines all S3 buckets stored on ECS assets.
Membership in a group is based on dynamic rules analysis. Here is the predefined rule for all S3 buckets.

You can create new groupings that start with this definition and then restrict it further.
New properties
The following new properties are added to tagged records.

Table 5 New properties for ECS (S3) Capacity Rates groups

<table>
<thead>
<tr>
<th>Field name in UI</th>
<th>Property name in database</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Storage cost ($ / GB/Month)</td>
<td>ecostgb</td>
<td>Used to calculate S3 billing costs. The AWS billing reports show all three cost types. Decimals are allowed.</td>
</tr>
<tr>
<td>Data transfer (download) cost ($/GB)</td>
<td>ecostdld</td>
<td></td>
</tr>
<tr>
<td>Data transfer (upload) cost ($/GB)</td>
<td>ecostuld</td>
<td></td>
</tr>
<tr>
<td>Bucket Grouping</td>
<td>bucktgrp</td>
<td>The bucket grouping name. This is the group name on the UI.</td>
</tr>
</tbody>
</table>

Permitted user changes
The following changes are permitted for bucket groupings:

- Edit a group's cost-related property values.
- Edit membership rules.
- Create new groups—New groups, with customized rules, are permitted. Billing costs for customized groupings are calculated and reflected on the billing reports located in Report Library > Amazon S3 > Operations.
- Delete groups.

Block Chargeback Grouping
The Block Chargeback Grouping classifies groups of host devices for the purpose of chargeback reporting. With custom-defined groups, you can generate Block Chargeback reports for storage attached to a custom-defined group of hosts. Hosts include all discovered physical hosts, hypervisors, and VMs.

This group type is installed by the SolutionPack for Block Chargeback. Predefined groupings for the three categories of host are installed for general chargeback reporting.
This is a hierarchical group type. The order of the groups does not matter, and a device can belong to more than one group. You can define new groups at the top level (under All) or within existing groups.

**Note**

Groups do not inherit rules from their parent. The rules for each group must contain the complete set of filters for the group.

**New properties**

When a match occurs, one new property is added to tagged records.

**Table 6** New property for Block Chargeback Grouping

<table>
<thead>
<tr>
<th>Field name in UI</th>
<th>Property name in database</th>
<th>Description</th>
</tr>
</thead>
</table>
| Group Name       | nodegrp                   | Identifies the device group name. This name is used in the chargeback by group reports. If a record is tagged with multiple property names (group names), the values are separated with a pipe (|).

The chargeback reports show costs by service level within each block chargeback group. The costs per GB are associated with the service levels, and are configured in the service level groups.

For more information, see the Chargeback chapter.
Permitted user changes
You can edit a group's rules of membership, create new groups, and delete groups.

Membership rule limitations
Block chargeback processing operates on groupings of hosts, VMs, and hypervisors. This processing is very efficient and easily supports any number of custom groups that filter and tag records associated with hosts, VMs, and hypervisors.

Note
The interface is open-ended, and lets you create rules that tag non-host devices, such as, for example, specific arrays. In that case, the tagging in the database still occurs. (That is, the array's records are tagged with the nodegrp property with a value equal to the block chargeback group name). However, this association is not picked up by chargeback processing or the chargeback reports. Chargeback processing examines only the database records associated with hosts, VMs, and hypervisors. Complicated custom reports are required to show chargeback costs based on device types other than hosts, VMs, and hypervisors. Contact Customer Support to discuss your requirements.

Additional processing
Some amount of time must pass before a new block chargeback group is populated and becomes visible in the chargeback reports. In general, expect to wait a day. The actual timing depends on the frequency settings for the following processes:

- As with all data enrichment changes, all affected collectors must complete a few cycles before new properties and values are propagated though the system.
- The import-properties task must run to push collected data to the in-memory property store on the Frontend.
- Finally, the chargeback-processor task must run to calculate chargeback costs based on the data in the in-memory Frontend property store.

File Chargeback Grouping
The File Chargeback Grouping classifies file systems and file shares for the purpose of chargeback reporting. With custom-defined groups, you can generate File Chargeback reports for specific customers or applications.

This group type supports file systems and quota-based file shares. The predefined groups are for file system-based file shares. Custom groups must be created for the quota-based NFS exports and CIFS shares to appear in the file chargeback reports.
This is a hierarchical group type. The order of the groups does not matter, and a file system can belong to more than one group. You can define new groups within existing groups.

**Note**
Groups do not inherit rules from their parent. The rules for each group must contain the complete set of filters for the group.

**New properties**
One new property is added to tagged records. If a record is tagged with multiple groups, the values are separated with a pipe (|).

<table>
<thead>
<tr>
<th>Field name in UI</th>
<th>Property name in database</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group Name</td>
<td>compgrp</td>
<td>Identifies a file chargeback group name.</td>
</tr>
</tbody>
</table>

The chargeback reports show costs for each file chargeback group. The cost per GB is associated with the NAS service level, and is configured in the service level group. Only the NAS service level group is used for chargeback calculations.

For more information, see the Chargeback chapter.

**Permitted user changes**
You can edit any group, create new groups, and delete groups.

To create customer-specific or application-specific file chargeback reports, create custom groups that include the file systems and file shares used by the customer or application.
Platform

The Platform group type classifies configuration items by type. This group type provides easy report filtering on reports.

Platform is installed with predefined groups as shown in the figure below. The SolutionPacks contribute to group definitions, as shown in the expanded Storage Systems group in the figure.

This is a hierarchical group type. A configuration item can belong to more than one group. You can define new groups within existing groups for your own reporting and filtering purposes.

**Note**

Groups do not inherit rules from their parent. The rules for each group must contain the complete set of filters for the group.

A group filter for Platform is included on many of the ViPR SRM global reports. The filter displays all of the Platform group names that are configured and that are valid for the current report and its current data. In the following example on the Storage Systems report, the filter does not include choices for Hosts or Storage Connectivity because those items are not included in the current report.
By applying a filter, you can limit a report to display, for example, only metrics for Atmos systems.

Membership in a group is based on dynamic rules and static selections. The Create and Edit dialogs include the Dynamic and Static tabs to support the two ways to define membership in a group.

**New properties**

One new property is added to tagged records. If a record is tagged with multiple groups, the values are separated with a pipe (|).

**Table 8 New property for Platform Group**

<table>
<thead>
<tr>
<th>Field name in UI</th>
<th>Property name in database</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Group Name</td>
<td>pltfmgp</td>
<td>Identifies the platform group name. These names are used to construct the Platform filter on user reports.</td>
</tr>
</tbody>
</table>

**Permitted user changes**

It is not recommended to make any changes to the platform groups.

**Customer, location, and business unit**

These three group types are available for users to define and use as needed. By tagging collected metrics with values for customer, location, or business unit, you can create reports that are filtered by values in one or several of those groups.

These group types are flat. A collected metric can belong to only one customer, one location, and one business unit. As with all flat group types, the order of the groups within each group type is important. The first group rules to match is the tagged value.

**Flexible usage**

These group types and their associated predefined properties indicate a purpose (customer, business unit, location). However, your installation can define its own usage policies. For example:

- The customer or business unit properties are commonly used to tag applications.
- The location property can be used to indicate geographic locations, office buildings, or both.
Predefined group filters and report columns
The following features make it easy to use these group types for reporting purposes on the global reports:

- Group filters—Multiple filters can be applied, such as filters on both customer and location to show metrics for a specific customer at a specific location.

- Predefined columns—The columns for customer, location, and business unit are predefined but hidden by default.

New properties
For each group type, one new property is added to tagged records.

Table 9 New properties for customer, location, and business unit

<table>
<thead>
<tr>
<th>Field name in UI</th>
<th>Property name in database</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>customer</td>
<td>customer</td>
<td>Identifies a customer (or any other entity defined by your installation).</td>
</tr>
<tr>
<td>location</td>
<td>location</td>
<td>Identifies a location (as defined by your installation).</td>
</tr>
<tr>
<td>business unit</td>
<td>bunit</td>
<td>Identifies a business unit (or any other entity defined by your installation).</td>
</tr>
</tbody>
</table>

Video: Use Groups Management to populate and use the business unit property

Create rules to populate business unit values. Then use the business unit column and group filters to view the results in reports.

See How
Watch the video
CHAPTER 3

Raw Data Enrichment

The following topics describe how to add new properties to the APG database, how to populate the properties, and how to verify and troubleshoot your results.

- About raw data enrichment ................................................................. 50
- Guidelines for new property names .................................................. 50
- Supported accessors for getting data .................................................. 51
- Using the Data Enrichment UI ............................................................ 51
- Video: Use data enrichment to add new property and view it in reports........ 65
About raw data enrichment

Raw data enrichment provides a way to define and add new properties into the database. The new properties are available to filters, searches, expansions, and for display on reports.

Data enrichment is supported by the Property Tagging Filter (PTF) feature in EMC M&R. To create new properties, you define new PTFs, or add to existing PTFs.

A PTF defines the following:

- **Keys**—The property names of existing properties that will define the match to identify the data to tag with new properties.
- **New properties**—The property names of new properties to add when a key match is found.
- **The type of accessor to use to obtain the data to populate the filter. The default accessor is the file accessor.**
- **Information required by the accessor.** For example, the file accessor needs the file name of a csv file that contains the data to populate the new properties. The JDBC accessor needs the connection information.

The data obtained by the accessor consists of the key values to match against and the resulting new property values.

The PTF processor works by filtering collected data against the key values. When there is a match, the data is enhanced (tagged) with the new property and the associated values.

A variety of formats are accepted for specifying the key values to match upon, including string matches, wildcard patterns, ranges, and regular expressions.

Guidelines for new property names

New property names must be eight characters or less and unique.

The database limits property names to eight characters. If the configuration uses a longer name, the database truncates it to the first eight characters.

If the name already exists in the database, your data enrichment rules will populate the existing property.

Use the following procedure to ensure that your planned name is unique in the affected database.

**Procedure**

1. In the User interface, go to any report page and click the Advanced Search arrow.

   ![EDIT MODE](image)

2. In the Expansion field, click the icon for the Property selection helper.

3. Click the APG tab.

4. In the search field at the bottom of the dialog, type the planned name.

   If the name is found in the list, use a different name.
5. If you intend to use the property to enrich data in the Events database (to enhance alerts reporting), click the Events tab and ensure that the intended new property is not found there.

Supported accessors for getting data

Accessors obtain information from external sources to populate the new property values.

The following accessor types are supported by EMC M&R:

- JDBC accessor obtains property values from external databases.
- File accessor obtains property values from named files.
- Static accessor obtains property values directly from the XML configuration file.
- Web service accessor obtains property values from an external source using SOAP requests.
- Property accessor obtains values from the EMC M&R Web Service, which can obtain properties from multiple EMC M&R databases in the same request.
- Sparql accessor obtains values from a Sparql endpoint to retrieve data from a Resource Description Framework (RDF) repository, such as the EMC M&R Topology repository.

Note

The default accessor type is the file accessor. The file accessor is the only one supported by the data enrichment GUI. The UI configures the accessor and manages the file.

To use accessors other than the file accessor requires installation of a separate PTF and manual configuration of the XML files. For details, see the following:

- For new PTF installation procedures, see Install and configure a new Property Tagging Filter on page 78.
- For accessor configurations, see the Property Tagging Filter chapter in the EMC M&R Backend Configuration Guide.

Using the Data Enrichment UI

Use the data enrichment GUI to create the new properties and define the rules for tagging data with the new property values.

Here is the sequence of tasks required to add a property, populate it, verify it, and make it visible in a report.

1. Register a Collector Manager on page 52
2. Create new tagging structure on page 54
3. Populate the tag set rows on page 59
4. (Optional) Run the import property task and restart Collectors on page 63
5. Verify the new property on page 64
6. Add the new property to a report on page 64
Register a Collector Manager

To enable a PTF to run, you must associate it, or register it, to a Collector Manager.

To associate a data enrichment PTF to a Collector Manager, you register the Collector Manager.

Typically, a collector server hosts many Collector Managers. Typically, you want all collectors to use the same PTF to consistently tag data and populate your new properties. A best practice is to register a single module, the Collector-Manager :: Load-Balancer :: DataEnrichment module, to minimize overhead and ensure that the same set of tagging rules is used on all data processed on a collector server.

If there is a reason to use different sets of rules for different Collector Managers, you can register each Collector Manager separately, and associate different tagging rulesets to each one.

Procedure

1. Log onto the Console and go to Administration > Centralized Management > Data Enrichment.
2. In the right pane, click Register a new module.
3. In the Server column, select the server that hosts the data collecting module you want to register.
   
   Typically, select the Collector server. However, other server types also host a few collecting modules.
4. In the Categories column, select Collecting.
5. In the Modules column, select one or more modules that you want to participate in the property tagging.

   For typical applications, choose Collector-Manager :: Load-Balancer :: DataEnrichment.
Note
The list shows only running modules. If the desired module is not in the list, exit and check the status of the Collector Manager under Administration > Centralized Management > Physical Overview > server_name > collector_manager_name. Use the Start or Restart buttons on the page if needed.

6. Click Register.

The Data Enrichment page redisplay, and the newly registered module appears in the table. You can now associate property tagging rules to the module.
Create new tagging structure

Define the new tagging structure by defining the keys to match against and the new properties to add.

Procedure

1. Go to Administration > Centralized Management > Data Enrichment.

   The table shows the modules that are registered to participate in data enrichment.

2. Click a row for a registered module.

   The tagging page appears.

   This page contains a blue bar for each defined tag set.

   ![Image of tagging page](image)

   Click a row for a registered module.

   The tagging page appears.

   This page contains a blue bar for each defined tag set.

   **Note**

   VIPR SRM is installed with a predefined tag set named **global-enrichment**.

3. (Optional) Click the **global-enrichment** entry to expand it.

   Examine the structure of this predefined tagging structure as a preview to the next few steps. Notice the following:
   - This tagging structure contains one key, **device**.
   - It contains four possible properties to add to records when a collected device value matches a rule.
   - To populate the tagging structure, you would create rows in the table that specify the collected device values to match and the resulting four property values to add.
   - The bunit, customer, and location properties can also be managed using the Groups Management interface.

4. To proceed with defining a new ruleset structure, click **New tagging**.

   A new blue bar appears, with input fields for defining the new tag set.
5. For **Name**, create a unique name for your new tag set. This name becomes the name of the CSV input file containing the rules you define in the tag set.

6. To define a key for the structure, click **Add new key** and complete the dialog.

   a. For **New column**, start typing the APG database field name that you want to match on, and select the field from the list.
b. For **after**, select the position of this new key in the key set. (For the first key, the position must be **first**.)

c. For **Default value**, optionally enter a default value for this field.

The value must be in the format specified by the **Type** field in the next step. **Default value** is used as follows:

- A first record is created that uses all of the default values.
- If you add a new key into an existing structure that was already populated with records, the default value is used in all of the existing records.

d. For **Type**, select the type of match value to use for this field when populating the ruleset.

See **Type attribute for PTF keys** on page 57 for descriptions of each **Type** value.

e. For **Delete after use**, indicate whether to delete the matched property from the collected data after tagging is complete.

> **WARNING**

Use this option with extreme caution. Deleting fields can break your data model. Usually leave unchecked.

The **Delete after use** option is useful if your intention is to replace a collected property and value with a new property and value.

f. Click **Save**.

7. Optionally repeat the previous step to add additional keys to the structure.

8. To define a new property, click **Add new property** and complete the dialog.

```
Create a new property

* New column

after

Default value

Save
```

a. For **New column**, type a name for the new database property.

**Note**

APG database property names are limited to 8 characters. If you use more than 8, the property name is truncated to 8.
Note

If you reuse a name that already exists in the APG database, your rules will populate the existing property. It is generally recommended to create a new property.

b. For **after**, select the position of this new property in this tag set. (For the first property, the position is always **first**.)

c. For **Default value**, optionally enter a default value for this field.

   It is used as follows:

   - A first record is created that uses all of the default values.
   - If you add a new property into an existing structure that was already populated with records, the default value is used in all of the existing records.

d. Click **Save**.

9. Optionally repeat the previous step to add additional new properties to the tag set structure.

10. After creating all keys and all properties, click **Save**.

11. On the **Save Data Enrichment** dialog, select the modules that should use this tagging structure.

   If you have multiple registered modules, select all that apply.

12. Click **Update**.

   The tagging page reappears with a new blue bar for the new tag set.

---

**Type attribute for PTF keys**

PTF keys are defined using a Type attribute. The Type defines the format of the match values.

**View the defined Type for a key**

You can view the Type that was defined for a key by hovering your cursor over the column header until a tooltip appears for that column. The following image shows that the vendor key was defined with a Type of string. This means that the values you provide in that column must be strings, and cannot include wildcards.
Type attribute values
The following list defines each Type value.

String
An unquoted string, where the string must match exactly.

Wildcards
Wildcard characters are accepted in a match value. The following values are supported:

<table>
<thead>
<tr>
<th>Wildcard</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>* or %</td>
<td>Any multiple characters</td>
</tr>
<tr>
<td>?</td>
<td>Any one character</td>
</tr>
<tr>
<td>\</td>
<td>Escape character if needed</td>
</tr>
</tbody>
</table>

Range
A range of numerical values is accepted as the match value. Decimal values are permitted. The following rules apply:

- Specify the beginning and ending integer or decimal values
- Separate the two values with a semi-colon
- Enclose the phrase with brackets, as follows:
  - [ ] enclosing brackets indicate inclusive
  - ] [ non-enclosing brackets indicate exclusive
  - combinations are permitted

Range value examples:

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>[5;20]</td>
<td>Matches any value 5 through 20, including 5 and 20.</td>
</tr>
<tr>
<td>]5;21[</td>
<td>Matches any value greater than 5 and less than 21.</td>
</tr>
<tr>
<td>[5;20[</td>
<td>Matches any value 5 up to but less than 20. 19.5 would match.</td>
</tr>
<tr>
<td>[.5;1]</td>
<td>Matches any value .5 to 1, inclusive.</td>
</tr>
</tbody>
</table>

Regex
Any Java regular expression is supported as the match value.
@ values
The following special values are available for keys:

<table>
<thead>
<tr>
<th>Value in the GUI</th>
<th>Value in the XML/CSV files</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>@DEFAULT</td>
<td>&quot;**&quot;</td>
<td>Matches all values.</td>
</tr>
<tr>
<td>@EMPTY</td>
<td>&quot;&quot; (no space)</td>
<td>Matches a property that exists but is empty.</td>
</tr>
<tr>
<td>@NULL</td>
<td>&quot;@@&quot;</td>
<td>Matches when the property does not exist.</td>
</tr>
<tr>
<td>@MATCHALL</td>
<td>&quot;%%&quot;</td>
<td>Always matches, including when the property does not exist.</td>
</tr>
</tbody>
</table>

Populate the tag set rows

Populate the tag set with the key values to match and new property values to add.

There are several ways to populate the rows:

- Add rows manually using the Console interface. This method creates a CSV file using the name of the tagging set.
- Import a CSV or XSL file

  Note

  Importing a file overwrites all entries currently in the table.

- Add a few rows using the Console, and then download the file, edit it to add more rows, and upload it.
- Add a few rows using the Console, and then open the file in the Console, edit it, and save it.

Key values must match the Type attribute that was configured for the key or use one of the special @ values.

Procedure

1. Go to Administration > Centralized Management > Data Enrichment.
2. Click a registered module associated with the tagging set that you want to populate.
   A blue bar appears for each tagging set.
3. Click to expand the tagging set that you want to populate.
4. To enter the first row:
   a. Click in the first key field and type or select a match value.
      The value for a key field must be either:
      - A value that conforms to the key’s Type attribute (Range, String, Regex, or Wildcard).
      - One of the @value selections.
      To view the Type attribute for a key, hover the cursor over the column heading until the tooltip appears.
In the example above, a valid wildcard match value is `lgl*` to match all device names that start with LGL or lgl.

b. Click in each key field and provide a match value.

c. Click in each property (result) field and provide the desired value for the new properties.

d. Press Enter to commit a row.

5. To enter subsequent rows, right-click the list icon at the beginning of a row, and select either Insert before or Insert after to place the new row in the desired spot.

Rows are processed in the order in which they appear in this table (or the file). Row order can affect results, especially when you are using wildcards, regular expressions, or ranges. A subsequent row can overwrite a match by a previous row. Typically, you would start with the most general rule, such as a match on @DEFAULT or @MATCHALL, and end with more specific rules.

Here is a new tagging set named contact-email. The structure contains two keys and one new property. The four rows specify contact email values for the new property named cemail.
Expected results from these rules are:

- The contact for all vendors not specifically listed is supervisor@my.com.
- The contact for Brocade is joe@my.com.
- The contact for EMC is mary@my.com, except in the case of Isilon models, which has isilon_support@my.com as the contact.

6. Click **Save**.

   This command saves the table contents to a file. The filename is the tag set name (**email-contacts** in the example above), with an extension of **.csv**.

7. On the **Save Data Enrichment** dialog, select all of the registered modules that the file applies to, and click **Update**.

   The file is saved on each registered server, under **Collecting/Property-Tagging-Filter/module-name/conf/tagset_name**.

8. To export a tag set file, click **Export CSV** or **Export XSL**.

9. To import tagging set values from an external file:

   **Note**

   Importing a file overwrites all entries currently in the table.

   a. Click **Import CSV** or **Import XSL**.

   b. Follow your browser prompts to import the file.

   c. Click **Save** after importing the file.
d. Select all of the registered modules that the file applies to, and then click Update.

View and edit the PTF input file

After creating a few rows using the Console interface, you can view the file to determine the required format, and then manually add additional rows or import a file in the proper format.

There are two ways to view and edit a saved PTF input file.

- Export the file, edit it in a text editor, and then import it.

  **Note**

  Importing a file overwrites all entries currently in the table.

- Open the file in the Console's text editor.

Either way, you will notice that the @values and wildcards used in your Console entry are translated into standardized wildcard values.

Procedure

1. To export and import a PTF input file:
   a. Go to Administration > Centralized Management > Data Enrichment.
   b. Click a registered modules that you associated to the tag set.
   c. Expand the tag set name.
   d. Click Export or Import.

2. To open the file on the Console:
   a. Go to Administration > Centralized Management > Physical Overview > registered-server-name > Modules > Collecting > Property-Tagging-Filter :: registered-module-name.

   The registered-server-name is typically a Collector server. The registered-module-name is typically Load-Balancer.

   b. In the right pane, expand Configuration Files.
   c. Locate the conf/tag-set-name.csv file.
   d. Click Edit (pencil icon) next to the file.

   The file opens in the Console file editor.

3. Add additional rows using the format of the existing file.

   For example, here is the conf/email-contacts.csv file.

   Notice that @DEFAULT values entered on the Console were translated into "**".
   If you add rows, use the "**" notation, not the @DEFAULT notation.

4. After editing or importing an edited input file, apply the updates to all registered modules that use the tag set.
Note

If there is only one registered module that uses the tag set, this step is not needed.

a. Go to Administration > Centralized Management > Data Enrichment.
b. Select all modules that use the tag set associated with the edited file.
c. Click Update.

Edit a ruleset

You can change the structure of the ruleset by adding or deleting keys and properties. You can edit the values in the rows.

Procedure

1. Logon to the Console and go to Administration > Centralized Management > Data Enrichment.
2. Click a registered module associated with the tagging ruleset that you want to edit.
   A blue bar appears for each tagging ruleset.
3. Click to expand the tagging ruleset you want to populate.
4. To delete a key or a property from the structure, click Delete (trashcan icon) in the column header.
   This action removes the column and all values.
5. To add a row, right-click the list icon in a row and select Insert Before or Insert After.
   For entry instructions, see Populate the tag set rows on page 59.
6. To delete a row, right-click the list icon in the row and select Delete.
7. To edit a value in a row, click on the value, provide a new value, and press Enter to commit the change.
8. Click Save to save the table, and then Update to apply the changes to selected registered modules.

(Optional) Run the import property task and restart Collectors

Data enrichment processing depends on the configured intervals for the database property import task and the Collectors. For quicker data enrichment results, you can run these manually.

It might take a day for all involved processes to run. The appearance of new properties depends on the following:

- The database import-properties task must run. This task updates the in-memory database cache on the Frontend.
- The collecting process must execute the Property-Tagging-Filters. For accurate tagging results, sometimes two collection cycles are required.

If needed, you can run these processes manually.

Procedure

1. To run the database property import task manually:
a. Under Physical Overview, go to server_name - Front End > Tasks > Database > import-properties-Default.

b. In the right pane, click Run Now.

2. To restart a Collector-Manager:
   a. Click Administration > Centralized Management > Physical Overview > collector_server_name.
   b. With the desired collector server selected, click the Services tab.
   c. Select a Collector-Manager, and click Restart.

A Collector-Manager calls all of the individual collecting components in its configuration, such as the stream and XML collectors and the PTFs.

Verify the new property

After the new property has propagated through the system and the new PTF tag set is applied to collected data, you can see the property using Frontend tools.

Procedure

1. In the User interface, go to any report page and click the Advanced Search arrow.

2. In the Expansion field, click the icon for the Property selection helper.

3. Click the APG tab, and then use the search field at the bottom of the dialog to locate the new property.

   When searching, remember that a long property name was truncated to the first 8 characters.

   If the new property does not appear in the list, wait an hour and try again.

4. If the new property appears in the list, select it and exit the property selection helper.

   The new property name should appear in the Expansion field. Delete any other property names.

5. Click Apply.

   The search result lists the values for the property that have so far been applied to records in the database. It can take up to a day for all collectors to run, depending on settings at your site.

Add the new property to a report

To make the new property visible, create a new report or add the new property to an existing report.

Choose a report that includes the keys on which the tagging was based. The following procedure describes how to add the new property to an existing table report.

Procedure

1. Navigate to the report and click EDIT MODE.
2. Click the Report Details: Table tab.
3. Click + Property.
4. Scroll to the end of the property list (blue bars) and expand the Property: * bar.
5. Configure the new column:
   a. For Column Name, type the column header.
   b. (Optional) For Description, type the description that will appear in the tooltip if the user hovers the cursor over this column header.
   c. For Property, type the new property name.
   d. Click Save.
6. Click BROWSE MODE to return to the report.

**Video: Use data enrichment to add new property and view it in reports**

Create the tagging structure for a new property, add tagging rules to populate the property, and then add a column for the new property to an existing report.

This scenario adds a new property called cemail (for contact emails) to the APG database. The tagging structure uses existing vendor and model properties as keys. Several tagging rules define cemail values for a few vendor/model pairs. The tagging results are viewed in a table report.

See How
Watch the video!
The following topics describe the reporting features that help expose new properties added through data enrichment.

- Unhide columns for business unit, customer, and location ................................ 68
- Add a new column to a table report ............................................................... 69
- Filtering a report using group filters .............................................................. 70
- Add or remove group filters ......................................................................... 72
- Applying filters to table columns ................................................................. 72
Unhide columns for business unit, customer, and location

Many of the ViPR SRM global reports contain predefined columns for displaying business unit, customer, and location properties. The columns are hidden by default.

The columns for these popular data enrichment properties are predefined as a convenience. If your installation populates those properties, you can easily show the columns in reports.

Use the following procedure to determine whether the report includes the predefined columns and unhide one or all of them. The procedure describes two ways to unhide a column:

- Unhide for your login account only
- Unhide for all users

If an out-of-the-box report does not include the hidden columns, use the Console’s EDIT MODE to add a new column.

Procedure

1. On the Console, go to the report where you want to unhide the group columns.
2. To unhide the column for your login account:
   a. Hover the cursor over the upper right corner of the report until a small menu of icons appears, and click the Customize Table Columns icon.
   
   ![Customize Table Columns](image)

   b. On the Table Customization dialog, in the Displayed columns list, scroll to find the columns named Business Unit, Customer, or Location. These columns are typically at the end of the list.
c. Click the box by the desired column names to select them.
   The report shows the columns in the same order as they appear in this list.

d. Optionally, you can rearrange the column order by dragging the column name to the desired location in the list.

e. To save the changes, do one of the following:
   - Click **Save and Apply** to make the change persist for future login sessions.
   - Click **Apply** to implement the change for the current login session only.

3. To unhide the column for all users:
   a. Click **EDIT MODE**.
   b. Click the **Report Details: Table** tab.
   c. Locate the blue bar for the desired column name, and click the small arrow to expand it.
   d. Expand **Advanced Properties**.
   e. For **Display Customization**, choose one of the following:
      - **displayed, user customizable** to allow your users to hide/unhide the column using the method described above
      - **displayed, always**
   f. Optionally, rearrange column order on the **Report Details: Table** tab by dragging the column's blue bar to the desired location in the list.
   g. Click **Save**.
   h. Click **BROWSE MODE** to return to the report.

## Add a new column to a table report

To view values for a new data enrichment property, you can add a new column to an existing report.

The results of this procedure are visible by all users.

**Procedure**

1. Log on to the Console and navigate to the report you want to change.
2. Click **EDIT MODE** at the top of the page.
3. Click the **Report Details: Table** tab.
   
   This tab shows a blue band for each defined column in the report.
4. To add the new column to this report, click the **Property** button near the top of the page.
A new blue band is added at the bottom of the page. Notice that the last band is expanded, exposing fields for defining the new column.

5. Define the new column as follows:
   a. For **Column Name**, type the desired column header.
   b. For **Property**, type the new data enrichment property. You can also click the list icon, select a value, and click **Apply**.
   c. Click **Save** on the bottom of the page.

6. To move the new column into the desired position in the report:
   a. Hover the pointer over the blue band until the **move** icon appears.
   b. Drag and drop the blue band to its new position.
   c. Click **Save**.

7. To view the report with the new column, click **BROWSE MODE** at the top of the page.
   The system regenerates the report, populating values in the new column.

**Filtering a report using group filters**

Group filters appear across the top of the report. They present a dialog of checkboxes that let you select multiple values to filter on.

The dialog lists only the values that are relevant to the current report. The list reflects values found in your database and is further limited by the report filter in the report
definition. For example, the following filter on the Platform property lists only the platforms that are being monitored at your installation and are relevant to the report definition:

![Filter on Platform](image)

**Note**

If there are no values in a filter dialog box, your installation is not populating that property or no values apply to the current report.

**Note**

The list of values in a group filter is not limited by column filters. For example, a **Situations to Watch** report might contain a column filter that eliminates rows based on values in that column. In that way, a **Situations to Watch** report typically contains just a few rows. The group filter dialog, however, continues to list all values relevant to the entire report, not just to the displayed rows.

A group filter can be defined for any database property.

- If the filter is for one of the data enrichment properties maintained in the **Centralized Management > Groups Management** module, the dialog uses group names as defined under Groups Management. Some groups have predefined names. Other groups define a Default group, and expect each installation to create additional groups.

- If the filter is for a collected property, the dialog uses the property values from the database.

**Procedure**

1. Click the icon for a group filter.
   
   A dialog of values appears.

2. Select one or more values to include in the report, and click **Apply**.
   
   The funnel for the applied filter changes color to indicate that the filter is active.
   
   The report redisplayes, including only the data for the values you specified in the filter. For example:

   - In a table report, the filter eliminates rows from the table.
   - In a consolidated bar chart, the filter eliminates data from the metric calculations.

3. If multiple filters are available, optionally apply additional filters to further limit the displayed results.

4. To cancel a filter, click the filter icon and select **Clear**.
Add or remove group filters

A group filter appears in the header of a report. It provides a convenient way for users to filter a report on a predefined property or set of properties.

A group filter presents values for one or more properties in a checkbox format. Users can select multiple property values under a property, and the report is filtered to show data for the selected values. Any property that appears in the report can be used in a group filter, including the data enrichment properties.

Use this procedure to add or delete a group filter. For group filter syntax information, see Group Filter in the Help topic named Filtering and expansion parameters.

Procedure

1. In EDIT mode, click the Filtering & Expansion tab.
2. Expand the Group filter section.
3. To remove a filter:
   a. Click Switch to input field.
   b. Remove a property from the expression.
4. To add a new filter:
   a. Click Add Property.
   b. Click Simple at the top of the dialog.
   c. Select the property you want to create a filter for.
      To be meaningful, the property must be used in the report. Use the Report Details tab to research the property names used in the report.
   d. To combine properties in the same filter, select multiple properties and then click Selected properties must be grouped at the bottom right of the dialog.
   e. Click OK.
   f. Click Switch to list and edit the expression if needed.
5. Click Save.

Applying filters to table columns

Filtering on a table column redisplays the report, showing only the rows that match the filter condition.

The filter icon ( ) in a column header indicates that the column can be filtered. Columns with metrics, dates, and times cannot be filtered.

Procedure

1. Click the icon in the column header.
2. In the text box that appears, enter the filter value using any of these methods:
Note

The values are case-sensitive.

- Type a valid value for the column
- Type a space, wait for the system to show suggestions, and select from the list
- Start typing a value, wait for the system to show suggestions that start with your entry, and select from the list
- Type a value containing wildcards. Supported wildcards are:

<table>
<thead>
<tr>
<th>%</th>
<th>Matches any character any number of times.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>For example:</td>
</tr>
<tr>
<td></td>
<td>- <strong>VPLEX</strong> matches any value that starts with VPLEX</td>
</tr>
<tr>
<td></td>
<td>- %04 matches any value that ends with 04</td>
</tr>
<tr>
<td></td>
<td>- %Unified% matches any value that contains the characters Unified in the beginning, middle, or end</td>
</tr>
<tr>
<td>*</td>
<td>Same as %</td>
</tr>
<tr>
<td>_</td>
<td>Matches any one character.</td>
</tr>
<tr>
<td></td>
<td>For example:</td>
</tr>
<tr>
<td></td>
<td>- __<em>.%</em>.%_.% matches any IP address whose first component is 2 digits.</td>
</tr>
</tbody>
</table>

3. Press Enter.

The report redisplays, showing only those rows with values that match the filter.

In addition, the following visual cues remind you that a filter is in effect:

- The filter icon for the filtered column is blue.
- The phrase **cancel filtering** appears in the sentence above the table.

4. To revert to the original report, click **cancel filtering**.
Reporting Features to View Enrichment Results
CHAPTER 5

Advanced Topics

The following topics describe advanced customization and development opportunities for the Groups Management and Data Enrichment modules.

- Changing the Groups Management UI configuration ........................................... 76
- Install and configure a new Property Tagging Filter ........................................... 78
Changing the Groups Management UI configuration

You can modify the table header, the filters, the drop down choices, and even the preview tables for each of the groups in the Group Management UI.

For example, you can add new columns to the preview tables. You can create a custom Group Management UI. These actions are possible by editing the Group Management configuration files.

Directory structure

Each group type in the Group Management UI is defined by a separate set of configuration files.

The group management configuration files are located on the Frontend server here:

```
<APG>/Custom/WebApps-Resources/Default/centralizedmanagement/group-tagging
```

For example, here are some files that might appear in the group-tagging directory:

```
srm-fe-01:/opt/APG/Custom/WebApps-Resources/Default/centralized-management/group-tagging # ls
defaultBusinessUnit  defaultCustomers  defaultDevice
defaultLocation     defaultServiceLevel
defaultBusinessUnit.xml defaultCustomers.xml defaultDevice.xml
defaultLocation.xml   defaultServiceLevel.xml
```

Note

The default group-tagging files are packaged with certain SolutionPacks, such as the SolutionPack for EMC M&R Health, SolutionPack for EMC VMAX, SolutionPack for EMC VNX, and so on. Each ViPR SRM site sees only the group management files that are packaged with the SolutionPacks that are installed.

Each group type is supported by the following:

- A directory named `default<group-name>`
- An XML file named `default<group-name>.xml`

As an example, the Service Level by LUN group type is supported by a directory called `defaultServiceLevel` and an XML file called `defaultServiceLevel.xml`.

The directories contain resource files named `resource_en.properties` and `resource_fr.properties`, used to control the language settings of the group. The XML files control the look of the group and options for that group in the Group Management UI.

Group Management configuration file description

To change the UI for a group in Group Management, edit the configuration file for the group type.

This section uses the `defaultBusinessUnit.xml` file as an example. The other `default<group-name>.xml` files follow the same logic and have the same syntax.
The contents of the defaultBusinessUnit.xml file is shown below:

```xml
<group-tagging-properties
    xmlns="http://www.watch4net.com/APG/Management/
    Centralized-Management/Plugins/Tagging"
    name="group3" type="hierarchical" group-property="bunit">
    <tagger-instance-name>PTF-Business-Tagging</tagger-instance-name>
    <csv-name>business-enrichment</csv-name>
    <eval-type>sqlpattern</eval-type>
    <rule-properties>
        <property>device</property>
    </rule-properties>
    <display-properties>
        <property searchable="true" used-in-static-preview="true">devtype</property>
        <property searchable="true" used-in-static-preview="true">device</property>
        <property searchable="true">devdesc</property>
        <property searchable="true">parttype</property>
        <property searchable="true">part</property>
    </display-properties>
    <expand-on>
        <property>devtype</property>
        <property>device</property>
    </expand-on>
</group-tagging-properties>

<group-tagging-properties>
    Defines properties of this group type.

| xmlns | Names the plugin that processes the file. Do not change. |
| name | |
| type | Hierarchical or flat. |
| group-property | Property name to add to records as a result of tagging. This property is added into the database if not already defined in the database. Use this property name to add the property to reports. For the business unit group, the property name is bunit. |

<tagger-instance-name>
    Maps the group into a specific <filter> element in a Collector-Manager's collecting.xml file.
    Business Unit is mapped to a filter named PTFBusiness-Tagging.

<csv-name>
    Name of the csv file associated with this group. This csv file is updated when a user makes changes to the group using the Group Management UI. The path for the file is <APG>/Collecting/Property-Tagging-Filter/<instance-name>/conf.
    For Business unit, the businessenrichment.csv file is updated.

<eval-type>
    The type of evaluation to use when performing the match of key values against collected data. Valid values are:
• sqlpattern
• regex

<rule-properties>
The <rule-properties> defines the properties that will be defined in the drop down menus for the group in question. By default, the property device will be shown in the drop down menu.

<display-properties>
The properties to include in the preview table on the Group Management UI. Each property represents a column in that table.

The searchable attribute indicates whether the property is searchable from the Group Management UI using the search text field.

For Business Unit, the preview table contains five columns: devtype, device, devdesc, parttype, and part.

<expand-on>
Lists the properties that the preview table expands on.

For Business Unit, the expansion is on both devtype and device.

Install and configure a new Property Tagging Filter

Use these procedures to create and configure a new Property-Tagging-Filter (PTF).

This scenario installs an unconfigured PTF. It configures a file accessor and a tag set that adds three new properties based on the values of the device property. It adds the new PTF to the processing chain for the SNMP Collector-Manager.

For configuration examples of other types of accessors, see the available at .

This scenario includes the following:
1. Install a PTF
2. Configure the PTF
3. Add the PTF to a Collector-Manager
4. Verify the results
5. Enable the Backend connector

Install a new PTF

Install a new PTF from the server command line interface.

This procedure installs a new PTF called test. You can install multiple PTFs using this procedure, and then link them together in a processing chain associated with the Collector-Manager, if desired.

This procedure is required to perform data enrichment tagging with any accessor other than the file accessor. The procedure is also valid for a file accessor. For example, you can install a new PTF to isolate tag sets for a specific collector, rather than merging the tag sets into the out-of-the-box PTF managed on the Data Enrichment GUI.
Procedure

1. Log onto the ViPR SRM server.
2. Initiate the installation using the following command:

   ```
   /opt/APG/bin/manage-modules.sh install filter_location
   filter_name
   ```

   For example:

   ```
   /opt/APG/bin/manage-modules.sh install Property-Tagging-Filter test
   ```

   The above command installs a PTF with this path name:

   ```
   /opt/APG/Collecting/Property-Tagging-Filter/test
   ```

3. Verify the dependencies listed by the shell, and answer yes.
4. Wait while the installation installs the required components, listed in the dependencies.

   The following installation example installs two components.

   [root@glbf061 ~]# /opt/APG/bin/manage-modules.sh install property-tagging-filter test
   Required dependencies, in processing order:
   
   [1] java '8.8u31' v8.8u31
   [2] module-manager '1.7u2' v1.7u2
   [3] license-manager 'Default' v5.4
   [4] collector-manager 'Default' (none) => v5.6u1
   [5] jdbc-drivers 'Default' v2.3u1
   [6] property-tagging-filter 'test' (none) => v2.8u1
   > 4 not modified, 2 to install
   > 9.9 MB space required / 13.3 GB available
   > ? Enter the step to modify, 'yes' to accept them, or 'no' to cancel the operation [yes] >
   
   Starting installation of Collector-Manager v5.6u1 from collector-manager-5.6u1-linux-x64...
   * Gathering information...
   * 'Collector-Manager v5.6u1' will be registered with instance name 'Default'.
   * It will be installed in '/opt/APG/Collecting/Collector-Manager/Default'.
   * Unpacking files...
   * Installing files... 100%
   * 56 files have been installed.
   * Finalizing installation...
   * Installing service 'collector-manager Default'...  [ installed ]
   Installation complete.
   
   Starting installation of Property-Tagging-Filter v2.8u1 from property-tagging-filter-2.8u1-linux...
   * Gathering information...
   * 'Property-Tagging-Filter v2.8u1' will be registered with instance name 'test'.
   * It will be installed in '/opt/APG/Collecting/Property-Tagging-Filter/test'.
   * Unpacking files...
   * Installing files... 100%
   * 31 files have been installed.
   * Finalizing installation...
   Installation complete.
   [root@glbf061 ~]#

5. Verify that the new PTF was added into the named directory.
For example:

```
[root@lglbf051 Property-Tagging-Filter]# ls
brocade-fc-switch  emc-vnx  emc-watch4net-health  Generic-RSC  Generic-SNMP  Load-Balancer  test
[root@lglbf051 Property-Tagging-Filter]#
```

Configure the PTF

A configuration file specifies the type of accessor and other attributes for the PTF.

This scenario uses a file accessor. The file accessor is the default if no accessor is defined; therefore, the following procedure does not explicitly define an accessor type.

See the "Property Tagging Filter" chapter in the *EMC M&R Backend Administration Guide* for detailed configuration instructions for each type of accessor.

The following example configures two files:

- The PTF configuration file (*property-tagging-filter.xml*) — Defines the tag set structure (the key properties and resulting new property names) and the input file name.
- An input file (*enrich.txt*) — Contains the key values to match and the resulting property values to assign.

Procedure

1. Locate the PTF configuration file.
   It resides in the sub-directory for the installed PTF.
   For example:

   ```
   [root@lglbf051 test]# cd /opt/APG/Collecting/Property-Tagging-Filter/test/
   [root@lglbf051 test]# ls
   bin  conf  doc  lib
   [root@lglbf051 test]# cd conf/
   [root@lglbf051 conf]#
   custom-oid2Models  global-enrichment.csv  property-tagging-filter-config.xsd  property-tagging-filter.xml
   [root@lglbf051 conf]#
   ```

2. Edit the *property-tagging-filter.xml* file as follows:
   a. In the `<text-file>` element, identify the input file name.
      For example:
      ```
      <text-file path="conf/enrich.txt" encoding="UTF-8">
      ```
   b. In the `<field-separator>` element, identify the character that separates fields in the input file. The default value in the installed PTF is the colon (:).
      For example:
      ```
      <field-separator>:</field-separator>
      ```
c. In `<key-property>` elements, identify a collected property name to match on.

   For example:

   ```xml
   <key-property delete-after-use="false" string-type="string">device</key-property>
   ```

   Multiple `<key-property>` elements are permitted.

d. In `<new-property>` elements, define new property names.

   Multiple `<key-property>` elements are permitted.

   For example:

   ```xml
   <new-properties>
   <!-- Use a "new-property" tag for each new property to add to the raw values. -->
   <new-property>site</new-property>
   <new-property>contact</new-property>
   <new-property>code</new-property>
   </new-properties>
   ```

   **Note**

   The order of the `<key-property>` and `<new-property>` elements must match the order used on each line in the input file.

3. Verify and save the file.

4. Create the input file.

   For this example:

   - Name the file `enrich.txt` to match the value configured above.
   - Add a few lines of values in the format `device:site:contact:code` to match the tag set defined above

   For example:

   ```
   [root@lglbf051 conf]# cat enrich.txt
   london:oxford street:joe:ABXXCD12
   paris:champs elysees:marcel:PAYY123
   ```

   This input file augments the collected data as follows:

   - If `device = london`, tag the record with additional properties:

     ```
     site=oxford street
     contact=joe
     code=ABXXCD12
     ```

   - If `device = paris`, tag the record with additional properties:
Results

The following image shows the completed `property-tagging-filter.xml` file.

```
[root@gibfo5i conf]# cat property-tagging-filter.xml
<?xml version="1.0" encoding="UTF-8"?>
<property-tagging-filter-config xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.watchmeout.com/APO/Filter/PropertyTaggingFilter
property-tagging-filter-config.xsd"
xmns="http://www.watchmeout.com/APO/Filter/PropertyTaggingFilter">
<!-- Set refresh to check for reload every interval of time. -->
<!-- Unit can be "seconds", "minutes", "hours" or "days". -->
<refresh unit="minutes">5</refresh>
<files>
  <!-- Use a "text-file" tag for each input file -->
  <text-file path="conf\enrich.txt" encoding="UTF-8">
    <!-- The field separator for the input file -->
    <field-separator>:</field-separator>
    <!-- The quoting character. Double it to escape. -->
    <field-quoting>"</field-quoting>
    <!-- The default symbol for the input file. To use with key properties only. -->
    <default-symbol>**</default-symbol>
    <!-- The null symbol for the input file. Means property not there. -->
    <null-symbol>@@</null-symbol>
  </files>
  <!-- Match-all symbol. To use with key properties only. -->
  <match-all-symbol>%%</match-all-symbol>
  <!-- Use to insert property value for new properties. To use with new properties only. -->
  <property-insertion start="${ end}"/>
  <!-- List of the properties that make the key -->
  <key-properties>
    <!-- Use a "key-property" tag for each property to make the key -->
    <!-- string-type can be "string", "sqlpattern", "regex" or "range" -->
    <!-- the order of the "key-property" must match the order of the columns in the -->
    <!-- key-property delete-after-use="false" string-type="string">device=key-property
  </key-properties>
  <!-- List of the properties to add to the raw values -->
  <new-properties>
    <!-- Use a "new-property" tag for each new property to add to the raw values -->
    <new-property>site</new-property>
    <new-property>contact</new-property>
    <new-property>code</new-property>
  </new-properties>
</text-file>
</property-tagging-filter-config>
[root@gibfo5i conf]#  
```

Add the PTF to a Collector-Manager

Edit the Collector-Manager configuration file to add the new PTF into the Collector-
Manager's chain of processing tasks.

A Collector-Manager consists of listening collectors, filters, and sending connectors
which are connected in a processing chain. To add a PTF to the collection, you
introduce a new link into the middle of the processing chain.
This procedure shows how to edit the SNMP Collector-Manager to add a new PTF into the processing chain for the Generic-SNMP collector.

**Procedure**

1. Open the collector's configuration file for editing (`conf/collecting.xml`).

   ```bash
   [root@lglbf051 conf]# cd ../../../Collector-Manager/Generic-SNMP/conf/
   ```

   Here is an example before any configuration changes.

   ```xml
   <filter enabled="true" name="juniper-rpm-filter-tagrtttype" next="CalculateSize" config="Property-Tagging-Filter/Generic-SNMP/conf/juniper-rpm-tagrtttype.xml" />
   <filter enabled="true" name="juniper-rpm-microsectomilisec" next="juniper-rpm-filter-tagrtttype" config="Variable-Handling-Filter/Generic-SNMP/conf/juniper-rpm-microsectomilisec.xml" />
   <filter enabled="true" name="juniper-rpm-filter" next="juniper-rpm-microsectomilisec" config="Variable-Handling-Filter/Generic-SNMP/conf/juniper-rpm-filter.xml" />
   <!-- This is the filter used to calculate device Availability, based on its Uptime -->
   </filters>
   <collectors>
   <collector enabled="true" name="generic-snmp" next="AvailabilityBasedOnSysuptime" config="SNMP-Collector/Generic-SNMP/conf/snmpcollector.xml" />
   </collectors>
   </config>
   [root@lglbf051 conf]#
   ```

   Notice the value of the `next` attribute in the `<collector>` element.

2. Add a new `<filter>` element to define the new PTF.

   ```xml
   <filter enabled="true" name="test-PTF" next="AvailabilityBasedOnSysuptime" config="PropertyTagging-Filter/test/conf/property-tagging-filter.xml" />
   ```

   Use the value of the `next` attribute from the collector element (above) as the value of the `next` attribute.

3. In the `<collector>` element, change the value of the `next` attribute to point to the new PTF filter.

   ```xml
   <collector enabled="true" name="generic-snmp" next="test-PTF" config="SNMP-Collector/GenericSNMP/conf/snmpcollector.xml" />
   ```
4. Temporarily change the `<connector>` element to send output to a file, rather than to the Backend.

This step is recommended for initial verification and debugging. You can verify the tagging results and catch any errors in the configuration before updating the database. It is more complex to resolve errors when the data is in the database.

   a. Add a `<connector>` element for a file connector, if one does not exist. Set the `enabled` attribute to `true`.

   ```xml
   <connector enabled="true" name="File" type="File-Connector" config="conf/fileconnector.xml" />
   ```

   b. Disable the Backend connector.

   ```xml
   <connector enabled="false" name="Backend" type="Socket-Connector" config="conf/socketconnector.xml" />
   ```

Results

The following shows the end of the `collecting.xml` file as edited.

```xml
<filter enabled="true" name="AvailabilityBasedOnSysuptime" next="other-filter juniper-rpm-filter
juniper-qos-filter cisco-qos-filter cisco-ipsla-filter cisco-mds-nexus-filter emc-isis-filter
brocade-fc-switch-filter emc-data-domain-filter aruba-wifi-filter
cisco-wifi-filter cisco-cdpfilter"
config="Availability-Filter/Generic-SNMP/conf/availability-filter.xml" />
<filter enabled="true" name="test-PTF" next="AvailabilityBasedOnSysuptime" config="PropertyTagging-Filter/test/conf/property-tagging-filter.xml" />
<collectors>
  <collector enabled="true" name="generic-snmp" next="test-PTF"
config="SNMP-Collector/GenericSNMP/conf/snmpcollector.xml" />
</collectors>
<connectors>
  <connector enabled="false" name="Backend" type="Socket-Connector" config="conf/socketconnector.xml" />
  <connector enabled="true" name="File" type="File-Connector" config="conf/fileconnector.xml" />
  <connector enabled="true" name="TopoBackend" type="Socket-Connector" config="conf/topoconnector.xml" />
</connectors>
```

Verify results

You can view the results of tagging in the file connector output.

Before you begin

Procedure

1. Restart the associated Collector-Manager.
2. View the contents of the file connector output that we configured.

For example:

```bash
# tail -f /opt/APG/Collecting/Collector-Manager/Generic-SNMP/logs/file-connector.log
```

3. Identify lines that contain `device=london` or `device=paris` and verify that the correct new properties were added.

**Device=london**

```
1432695460: group::10.9.254.17.stdIfT2..1.19.RATEP.Pkts/s.-1419189655({ifindex=35, code=ABXXCD12,
mxhspeed=1000, ip=10.9.254.17, part=GigabitEthernet2/26,
maxspeed=1000000000, source=W4NSnmp-1,
parttype=Interface, pollgrp=PG_CISCO, devdesc=Cisco IOS Software. c7600s3223_rp Software (c7600s3223_rp-ADVENTERPRISEK9-M). Version 15.2(4)S. RELEASE SOFTWARE (fc1). Technical Support: http://www.cisco.com/techsupport..Copyright (c) 1986-2012 by Cisco Systems. Inc...Compiled Fri 20-Jul-12 18:18 by , devtype=Router, unit=Pkts/s, site=oxford street, ifalias=, ifname=Gi2/26, ifacemac=0023EB6804D9, contact=joe, iftype=ethernetCsmacd, name=ifOutDiscards, location=DevLab. row 5-rack 5. 1133 Westchester Ave.. 3rd floor, model=7604, device=london, datagrp=GENERICINTERFACES})=0.0
```

**Device=paris**

```
1432705967: group::10.9.254.16.stdIfXT4a..1.4.RATEP.Pkts/s.-1665029324({ifindex=77, code=PAYY123,
mxhspeed=0, ip=10.9.254.16, part=Tunnel727,
maxspeed=100000, source=W4NSnmp-1, parttype=Interface, pollgrp=PG_CISCO, devdesc=Cisco IOS Software. c7600s3223_rp Software (c7600s3223_rp-ADVENTERPRISEK9-M). Version 15.2(4)S. RELEASE SOFTWARE (fc1). Technical Support: http://www.cisco.com/techsupport..Copyright (c) 1986-2012 by Cisco Systems. Inc...Compiled Fri 20-Jul-12 18:18 by , devtype=Router, unit=Pkts/s, site=champs elysees, ifalias=, ifname=Tu727, ifacemac=, contact=marcel, iftype=mplsTunnel , name=ifOutMulticastPkts, location=DevLab. row 5-rack 5. 1133 Westchester Ave.. 3rd floor, model=7604, device=paris, datagrp=GENERIC-INTERFACES})=0.0
```

**Enable the Backend connector**

Enable the Backend connector to send collected and tagged data to the database. When you are satisfied that the tagging is correct, disable the file connector and enable the Backend connector. Otherwise, no data will display in reports.
Procedure

1. Open the collector's configuration file for editing (conf/collecting.xml).

```bash
[root@lglbf051 conf]# cd ../../../Collector-Manager/Generic-SNMP/conf/
```

2. In the `<connector>` elements, change the state for the Backend connector to "true" and the state for the file connector to "false".

```xml
<connector enabled="true" name="Backend" type="Socket-Connector" config="conf/socketconnector.xml" />
<connector enabled="false" name="File" type="File-Connector" config="conf/fileconnector.xml" />
```
CHAPTER 6

Troubleshooting

The following topics describe information helpful for troubleshooting data enrichment configurations.

- Groups Management configuration files ............................................................88
- Data enrichment configuration files ................................................................. 90
Groups Management configuration files

For troubleshooting and reference purposes, you can access the Property-Tagging-Filter configuration files.

Each group type is implemented with its own Property-Tagging-Filter. Each Property-Tagging-Filter is supported by a separate set of the following files:

- **XML configuration file**—Defines the structure of the data enrichment tag set used by the group type. Also contains the name of the CSV data file, in the `<text-file>` element.
- **CSV data file**—Contains the data for populating the properties, including the group names, the rules of membership for each group, and the new property value for each group.

The files are stored on the ViPR SRM servers.

- The files are located here:

  `<APG>/Collecting/Property-Tagging-Filter/process-instance-name/conf/`

- On the Console, the files can be accessed here:

  Administration > Centralized Management > Logical Overview > Collecting > Property-Tagging-Filter :: `process-instance-name`

In the previous paths, the `process-instance-name` is user-provided during installation. The following table shows default instance names.

**Table 10 Location of Groups Management Configuration Files**

<table>
<thead>
<tr>
<th>Group type</th>
<th>Default instance names/Configuration file names</th>
</tr>
</thead>
</table>
| Bucket Service Levels                   | Property-Tagging-Filter:: emc-ecs  
                                        | `ptf_bucket_svclevel_tagging.xml`                                                    |
| ECS Capacity Rates                      | Property-Tagging-Filter:: amazon-aws  
                                        | `conf/ptf_bucket_rates_tagging.xml`                                                  |
| Block Chargeback Grouping & Service Levels by LUNs | All Collector-Manager and Load-Balancer instances have a copy of the following configuration files in their `conf/` directory:  
                                        | PTF-Group-Tagging.xml  
                                        | `ptf-ServiceLevelsByLUN.xml`                                                        |
| All other group types                   | Property-Tagging-Filter::Load-Balancer  
                                        | NOTE: In an installation that uses multiple collector servers, each collector server has a PTF:: Load-Balancer. Each load balancer has a copy of the following PTF configuration files.  
                                        | Customer  
                                        | PTF-Customers.xml  
                                        | Location  
                                        | PTF-Location.xml  
                                        | Business unit  
                                        | PTF-Business-Tagging.xml  
                                        | Platform  
                                        | PTF-Platform-Tagging.xml  |
Table 10 Location of Groups Management Configuration Files (continued)

<table>
<thead>
<tr>
<th>Group type</th>
<th>Default instance names/Configuration file names</th>
</tr>
</thead>
<tbody>
<tr>
<td>File Chargeback Grouping</td>
<td>PTF-Group-Tagging.xml</td>
</tr>
<tr>
<td>Service Levels by File Share</td>
<td>PTF-ServiceLevelByFS.xml</td>
</tr>
<tr>
<td>Block Chargeback Grouping</td>
<td>PTF-Group-Tagging.xml</td>
</tr>
<tr>
<td>Service levels by LUNs</td>
<td>ptf-ServiceLevelsByLUN.xml</td>
</tr>
</tbody>
</table>

NOTE:

Procedure

1. Navigate to Administration > Centralized Management > Logical Overview > Collecting > Property-Tagging-Filter :: process-instance-name.
2. In the right pane, click to expand Configuration Files.
3. Use the Search text box to locate the configuration file you need.
4. Click Edit (pencil icon) next to the file of interest.
   The file opens in a text editor.

Note

There might be multiple copies of the same file throughout the installation. The Groups Management UI updates the files appropriately in all locations and keeps the files in sync with updates saved or imported through the Groups Management interface. Manually editing a file only updates one file. For this reason, it is best to use the Groups Management UI to edit the configuration or import file changes.

Example 1 Example configuration file

The XML file for the platform group type is shown below. Notice the following:

- The <text-file> path attribute shows the associated rules file.
- The <key-properties> define the structure of the match rules (that is, the structure of each line in the CSV file).
- The <new-properties> defines the new property names. In this case, there is only one new property.

```xml
<property-tagging-filter-config xmlns="http://www.watch4net.com/APG/Filter/PropertyTaggingFilter">
  <refresh unit="hours">1</refresh>
  <files>
    <text-file encoding="UTF-8" path="conf/platform-groups.csv">
```
Example 1 Example configuration file (continued)

```xml
<field-separator>,</field-separator>
<field-quoting>"</field-quoting>
<default-symbol>**</default-symbol>
<null-symbol>@</null-symbol>
<match-all-symbol>%%</match-all-symbol>
<property-insertion end="}" start=":${/">
<key-properties>
  <key-property delete-after-use="false" string-
type="regex">devtype</key-property>
  <key-property delete-after-use="false" string-
type="sqlpattern-cs">devtype</key-property>
  <key-property delete-after-use="false" string-
type="regex">device</key-property>
  <key-property delete-after-use="false" string-
type="sqlpattern-cs">device</key-property>
  <key-property delete-after-use="false" string-
type="regex">datagrp</key-property>
  <key-property delete-after-use="false" string-
type="sqlpattern-cs">datagrp</key-property>
  <key-property delete-after-use="false" string-
type="regex">devdesc</key-property>
  <key-property delete-after-use="false" string-
type="sqlpattern-cs">devdesc</key-property>
  <key-property delete-after-use="false" string-
type="regex">datatype</key-property>
  <key-property delete-after-use="false" string-
type="sqlpattern-cs">datatype</key-property>
  <key-property delete-after-use="false" string-
type="regex">source</key-property>
  <key-property delete-after-use="false" string-
type="sqlpattern-cs">source</key-property>
  <key-property delete-after-use="false" string-
type="regex">arraytyp</key-property>
  <key-property delete-after-use="false" string-
type="sqlpattern-cs">arraytyp</key-property>
</key-properties>
<new-properties>
  <new-property>pltfmgrp</new-property>
</new-properties>
</text-file>
</files>
</property-tagging-filter-config>

Data enrichment configuration files

One Property-Tagging-Filter is associated with the Data Enrichment UI. Multiple tag sets are defined in one configuration file.

The configuration file name for the Data Enrichment PTF is `conf/property-tagging-filter.xml`. You can access it here:

Administration > Centralized Management > Logical Overview > Collecting > Property-Tagging-Filter::Load-Balancer - server_instance

The `<text-file>` element defines a tag set. Each time a user defines a new tag set, a new `<text-file>` element is inserted into this configuration file. The path attribute of the `<text-file>` element names the associated input file name. By default, the text file name is the tag set name entered on the user interface.
Here is the `<text-file>` element for the predefined tag set named global-enrichment. Note the following:

- The path name for the match data is `conf/global-enrichment.csv`.
- The match structure consists of only one key (device).
- Up to four new properties can be assigned on device matches.

```xml
<text-file path="conf/global-enrichment.csv" encoding="UTF-8">
  <!-- Use a "text-file" tag for each input file -->
  <!-- The field separator for the input file -->
  <field-separator>,</field-separator>
  <!-- The quoting character. Double it to escape. -->
  <field-quoting>"</field-quoting>
  <!-- The default symbol for the input file. To use with key properties only. -->
  <default-symbol>**</default-symbol>
  <!-- The null symbol for the input file. Means property not there. -->
  <null-symbol>@</null-symbol>
  <!-- Match-all symbol. To use with key properties only. -->
  <match-all-symbol>%%</match-all-symbol>
  <!-- Use to insert property value for new properties. To use with new properties only. -->
  <property-insertion start="${" end="}" />
  <!-- List of the properties that make the key -->
  <key-properties>
    <!-- Use a "key-property" tag for each property to make the key. -->
    <key-property delete-after-use="false" string-type="string">device</key-property>
  </key-properties>
  <!-- List of the properties to add to the raw values -->
  <new-properties>
    <!-- Use a "new-property" tag for each new property to add to the raw values. -->
    <new-property>bunit</new-property>
    <new-property>customer</new-property>
    <new-property>appli</new-property>
    <new-property>owner</new-property>
  </new-properties>
</text-file>
```

Here is the `<text-file>` element for a user-entered tag set named contacts.

```xml
<text-file encoding="UTF-8" path="conf/contacts.csv">
  <!-- The field separator for the input file -->
  <field-separator>,</field-separator>
  <!-- The quoting character. Double it to escape. -->
  <field-quoting>"</field-quoting>
  <!-- The default symbol for the input file. To use with key properties only. -->
  <default-symbol>**</default-symbol>
  <!-- The null symbol for the input file. Means property not there. -->
  <null-symbol>@</null-symbol>
  <!-- Match-all symbol. To use with key properties only. -->
  <match-all-symbol>%%</match-all-symbol>
  <!-- Use to insert property value for new properties. To use with new properties only. -->
  <property-insertion start="${" end="}" />
  <!-- List of the properties that make the key -->
  <key-properties>
    <!-- Use a "key-property" tag for each property to make the key. -->
    <key-property delete-after-use="false" string-type="string">vendor</key-property>
  </key-properties>
  <!-- List of the properties to add to the raw values -->
  <new-properties>
    <!-- Use a "new-property" tag for each new property to add to the raw values. -->
    <new-property>bunit</new-property>
    <new-property>customer</new-property>
    <new-property>appli</new-property>
  </new-properties>
</text-file>
```
Troubleshooting
ViPR SRM uses data enrichment features to implement its chargeback / showback features.

Chapter 7, "Block Chargeback"
Chapter 8, "File Chargeback"
Chapter 9, "Object Chargeback"
Chapter 10, "Cloud Chargeback"
CHAPTER 7

Block Chargeback

The following topics describe the SolutionPack for Block Chargeback, the block chargeback reports, and associated concepts.

- Administration of block chargeback data collection ........................................... 96
- Block chargeback reporting ................................................................................97
- About service level by LUNs ............................................................................ 103
- About Block Chargeback Groupings ................................................................. 110
Administration of block chargeback data collection

The block chargeback reports depend on a combination of the usual ViPR SRM metrics collectors and an additional system task that runs on a configured schedule. Various SolutionPack options control which metrics are collected.

Install the SolutionPack for Block Chargeback

The block chargeback computations and reports require the SolutionPack for Block Chargeback. If you see a broken link error when navigating to the report locations, the SolutionPack is not installed.

For specific instructions about installing the SolutionPack for Block Chargeback, see the appropriate chapter in the *EMC ViPR SRM SolutionPack Guide* available here: https://community.emc.com/docs/DOC-52237.

Metrics options

The SolutionPack for Block Chargeback presents a set of options that control which metrics are collected.

- Choose between collection of Used or Presented metrics, or both
- Enable/disable collection of non-chargeable metrics
- Enable/disable collection of component-level chargeback metrics (components of the hosts and virtual machines, including host devices, RDMs and VMDK/log files.

You can reconfigure the solutionpack to change these options. These changes effectively increase or decrease the number of metrics collected and also affect the columns and data available for viewing on reports.

If the Solutionpack is reconfigured, expect to wait one day plus the frequency of the chargeback preprocessor task to see reconfigured chargeback data after such changes. To shorten these wait times, wait for data collection to occur and then manually run the chargeback preprocessor task.

Block chargeback preprocessor task

The SolutionPack for Block Chargeback creates a chargeback preprocessor task that collects the block chargeback metrics for all physical hosts and virtual machines. The task runs as scheduled during SolutionPack installation, or you can run the task manually.

The block chargeback reports are empty immediately after the SolutionPack is installed. They start displaying data only after:

- Collector data has had sufficient time to propagate through the environment, and
- The chargeback preprocessor task completes successfully. Depending on how the preprocessor task is configured, it could take up to 6 days for the task to run. See *Running the chargeback preprocessor task manually* on page 97.

Group changes affect collection

Chargeback data is affected by user configuration changes in *Centralized Management > Groups Management*, such as changes to service level tags and user-defined group definitions.

In general, expect to wait one day plus the frequency of the chargeback task to see reconfigured chargeback data after such changes. To shorten these wait times, wait for data collection to occur and then manually run the chargeback preprocessor task. For example, if the chargeback preprocessor task is scheduled to run every 5 days, wait one day for data to settle and then manually run the chargeback preprocessor task.
Running the chargeback preprocessor task manually

You can run the chargeback preprocessor task manually rather than waiting for the scheduled run.

Procedure

1. Go to Administration > Centralized Management > Scheduled Tasks.
2. Expand the Chargeback node.
3. Select the chargeback processor instance. The default instance name is chargeback-processor-generic-chargeback.
4. In the right pane, click Run Now.

Block chargeback reporting

The block chargeback reports provide visibility into block storage usage and associated costs by host or by defined groups of hosts in your environment.

Chargeback calculations are performed per service level agreement (SLA) of LUNs mapped to each host. Within a host or group of hosts, you can see the service level breakdown of associated storage and the corresponding chargeback costs.

The cost basis is user-configurable, per service level.

Block chargeback use cases

The block chargeback reports can help you understand the actual cost of storage in your environment. For chargeback purposes, you can choose among several use cases for charging for storage costs.

A set of four reports show chargeback metrics for different aspects of storage usage.

Primary Used Chargeable report

Primary storage is space for the original copy of data (and not for clones, snapshots, or replications). Used storage is the blocks that were allocated to arrays and are now unavailable for use by another device.

Primary Presented Chargeable report

Primary storage is for the original copy of data (and not for clones, snapshots, or replications). Presented storage is space that was made available to the host systems by the storage systems.

Total Used Chargeable report

Total storage is space for the original copy of data plus data clones, snapshots, and replications. Used storage is the blocks that were allocated to arrays and are now unavailable for use by another device.

Total Presented Chargeable report

Total storage is space for the original copy of data plus data clones, snapshots, and replications. Presented storage is space that was made available to the host systems by the storage systems.
**Chargeable and non-chargeable metrics**

This SolutionPack can generate both chargeable and non-chargeable metrics. Non-chargeable metrics are optional, and are disabled by default during SolutionPack installation.

If storage is not shared across hosts, there is no difference in chargeable and non-chargeable metric values.

The values are different when multiple hosts share a storage disk.

- Non-chargeable metrics are the values as seen directly by the host. The same values are duplicated for each of the hosts sharing that storage, without regard for how many hosts are sharing it.
- Chargeable metrics take host sharing into account, and divide the metrics by the number of hosts that are sharing it. The metrics are deduplicated, making them representative of chargeability.

**Examples**

If two hosts are sharing 500GB of block storage from the same disk, and none of that storage is used, the capacity metrics are as shown in the following table.

Table 11  Shared disk with no storage used

<table>
<thead>
<tr>
<th>Metric name</th>
<th>Host A - Disk1</th>
<th>Host B - Disk1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-Chargeable Presented (= total capacity of the disk)</td>
<td>500GB</td>
<td>500GB</td>
</tr>
<tr>
<td>Non-Chargeable Used (= used capacity of the disk)</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>Chargeable Presented (= total capacity of the disk ÷ # hosts sharing it)</td>
<td>250GB</td>
<td>250GB</td>
</tr>
<tr>
<td>Chargeable Used (= used capacity of the disk ÷ # hosts sharing it)</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Alternatively, if Host A uses 100GB of this storage, the capacity metrics are as follows.

Table 12  Shared disk with some storage used

<table>
<thead>
<tr>
<th>Metric name</th>
<th>Host A - Disk1</th>
<th>Host B - Disk1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non-Chargeable Presented (= total capacity of the disk)</td>
<td>500GB</td>
<td>500GB</td>
</tr>
<tr>
<td>Non-Chargeable Used (this has changed from 0 to 100 GB)</td>
<td>100GB</td>
<td>100GB</td>
</tr>
<tr>
<td>Chargeable Presented (= total capacity of the disk ÷ # hosts sharing it)</td>
<td>250GB</td>
<td>250GB</td>
</tr>
<tr>
<td>Chargeable Used</td>
<td>50GB</td>
<td>50GB</td>
</tr>
</tbody>
</table>
**Table 12** Shared disk with some storage used (continued)

<table>
<thead>
<tr>
<th>Metric name</th>
<th>Host A - Disk1</th>
<th>Host B - Disk1</th>
</tr>
</thead>
<tbody>
<tr>
<td>( \text{used capacity of the disk ÷} ) ( # \text{hosts sharing it} )</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**View chargeback reports**

The block chargeback reports provide the information needed to reclaim costs of storage from the users of the storage arrays, and to analyze storage usage and trends.

---

**Note**

Information about block chargeback reports applies only if the SolutionPack for Block Chargeback is installed.

The block chargeback reports are available under Reports Library > Chargeback and also under the Dashboard and Explore nodes, as described in the procedure below.

Two sets of reports are generated.

- The **Chargeback by Hosts** reports shows each host in the following categories:
  - Host/Hypervisor
  - Virtual machines — This category includes replica capacities for VM files on the datastore. Usages are prorated using the ratio of VM file capacity to datastore capacity.
  - Hypervisor cluster

- The **Chargeback by Group** reports show capacities and costs for groups of devices. Some installed device groups include All Physical Hosts, All Hypervisors, and All VMs.

You can define additional groups. If you do, the customized groups appear automatically as rows in these reports. For example, you can define a group of devices specific to a customer or business unit to see a chargeback row for that customer or business unit. When you click the row, a detailed report for that group is generated. See Create customized group chargeback report on page 111 for a demonstration that defines a group of devices and generates a customer-specific report.

All of the chargeback reports show capacities and costs by LUN service levels.

**Procedure**

1. To view the Chargeback by Host reports, go to Explore > Storage > More Reports > Chargeback by Host.
A summary report shows the device count for each host category.

2. Click a host category.
   A list of report types, one for each use case, appears.

3. Click the report type of interest.
   The detailed report appears below the report type list. You can easily switch between the use cases.

The report shows storage usage and costs by service levels within host.
- There are columns on the report for the most common service levels.
- The Other column aggregates service levels that do not have their own column, including your customized service levels. You can edit the report and add columns for additional service levels.
- The Chargeable Cost column is the calculated cost using the cost basis for each of the service levels. You can view and change cost basis by editing a
service level in Administration > Groups Management > Service level by LUNs.

- The Last Timestamp column indicates the last time the block chargeback preprocessor task ran. If you need more recent metrics, you can run the task manually.

4. To temporarily show or hide columns:
   a. Click the small table icon above the upper right corner of the report.
   b. In the displayed columns list, click to select or deselect columns that you want to appear or hide, and click Apply.

Here is a report for the Silver service level, generated by hiding all of the other service level columns.

5. To see more details about storage on one of the hosts, click the host name.

The detailed host report shows disks, arrays, and LUNs associated to a host.

6. To view the Chargeback by Group reports, go to Dashboards > Storage > Chargeback by Group.

7. Click the report type.

   The report appears below the summary list.

8. To see details about a group, click the group row.
View associated storage details per host

You can explore the underlying details of the calculated chargeback storage on each host's Associated Storage reports.

Procedure

1. On a chargeback report, click a host name in the Host column.
   The host names are links to the host home report. Be sure to click the underlined name that indicates a link.

   The home report for the host opens to the Associated Storage tab. The first report on the tab is a repeat of the numbers you saw above.

2. Scroll down to see the breakdown by arrays.
3. Click the Host Devices tab to see array details, including LUN names.

### About service level by LUNs

The block chargeback reports use service level tagging to categorize capacity allocations.

Service level agreements (SLAs) help to ensure compliance to service requirements and customer needs. The LUN service level tagging classifies block storage space according to storage characteristics in an SLA, such as the storage device type, disk type, speed, or response time. A cost per GB is assigned for each SLA.

To implement service level tagging that corresponds to your SLAs, use the Service Level by LUNs group type under Groups Management.

Service level names are predefined and consist of commonly used names, such as Platinum, Gold, and Silver. You can also create custom service levels. Nested service levels are not permitted.

Each service level is defined by underlying rules. With appropriate permissions, administrators can customize existing rules and add new rules to a service level. For new service levels, you can copy the rules from an existing service level or create new ones.
The rules match characteristics such as array type, disk technology, disk size, and RAID protection. Discovered LUNs that match the rules for a given service level are tagged with that service level.

Rules are applied to each LUN in the order that the rules appear in the list. The first matching service level is the one that is assigned to the LUN. If you created a new rule and you do not see your LUNs at the expected place, a rule higher up in the order of rules probably matched it.

Service level tagging currently supports the following technologies:
- EMC VMAX
- EMC VNX
- EMC XtremIO
- HDS
- HP 3PAR
- HP P9000 StorageWorks
- IBM XIV
- NetApp Block

Set cost basis and response times for block chargeback reports

Chargeback reports use a cost per GB configured on the service level. Some service levels assign minimum and maximum response times as additional attributes for reporting purposes.

Procedure
1. Click Administration > Centralized Management > Groups Management > Service Level by LUNs.
2. Click to select a service level name, and then click Edit.

Note
Names that are shaded are not configurable. To view the tagging definitions for those names, click View All Rules.

3. On the Edit dialog, in Cost per GB, type the cost basis for this service level.
   Cost is numerical with no assumption of any specific currency. Enter an integer or a decimal value. If you enter 3, the cost for 100GB is 300. If you enter .3, the cost for 100GB is 30.

4. Optionally set values for the Minimum Response Time and Maximum Response Time properties to associate with the service level.

5. Click Save.

Define a new LUN service level and add the column to reports

Use the Groups Management features to define a new service level agreement (SLA), and then edit the block chargeback reports to add a corresponding column for the service level name to the reports.

The capacities and costs for a new service level agreement are not displayed in the chargeback reports unless a corresponding column is added. This procedure shows how to create the new service level and then add a corresponding column to a report.
by copying and pasting an existing column definition, and then changing the filter on the new column.

**Procedure**

1. Go to Administration > Centralized Management > Groups Management > Service Level by LUNs.
2. Click Create.
3. For Service Level, type the name for the new service level.
4. For Cost per GB, type the cost basis for chargeback for this service level.
   Use a decimal or integer value.
5. For Minimum Response Time and Maximum Response Time, type the upper and lower bounds of response time that defines this service level.
6. Optionally create one or more rules to further define the characteristics of the new service level.
   a. Select a property.
      The property drop down menu lists the most common properties for LUN service level rules. You can create a rule using any property in the database by clicking Advanced and using the wizard.
   b. Select an operation.
   c. Type or select a value.
      For example:
   d. To add another rule to this rule set (Boolean AND), click Add rule.
   e. To create an alternate set of rules (Boolean OR), click Add New Rule Set.
7. Click Save.
   The new service level appears in the list, and a partial list of members appears in the table. The members are components that match the rules you just created.
8. To return to the User Interface:
   a. Click the browser tab with the tab name Modules - Administration.
b. Click **User Interface** in the banner.

9. Open the report where you want to add the new service level column in **EDIT MODE**:

Because the reports are linked to other nodes, some of the tree navigation must be performed in **EDIT MODE**. Use these steps:

a. Navigate to **Operations > Chargeback > chargeback category**.
   For example, go to **Operations > Chargeback > Chargeback by Group**.

b. Click **EDIT MODE**.

c. In **Linked To**, click the link.

d. In the left pane, expand the selected node to show the four chargeback reports.

![Chargeback by Group](image)

```
- Primary Used Chargeable
- Primary Presented Chargeable
- Total Used Chargeable
- Total Presented Chargeable
```

e. Select a report, such as **Total Used Chargeable**.

10. Click the **Report Details: Table** tab.

11. Select a column to copy by clicking the selection box next to the column title.
   For example, select **Gold** to copy that column's attributes.

12. Click **Copy** at the bottom of the page, and then click **Paste** at the top of the page.
   You will find the pasted column at the bottom of the list.

13. Click the white arrow next to the new column name to open the column configuration.

14. Configure the new column by changing the name, description, and filter to the new service level name.
a. To change the filter, right-click in the svclevel box and select **Edit expression**.

b. Type the new service level name exactly as you created it, and click **OK**.

15. Click **Save**.

16. To move the new column to another location, hover the cursor over the column's blue bar until the Move icon appears, and drag the column to the desired location.

17. Click **Save**.

18. Click **BROWSE MODE** at the top of the page.

You see the new column in the report.
Add new service level to Associated Storage reports

The host **Associated Storage** reports are based on a formula and require a simple edit to the formula to correctly report on the new service level.

The Associated Storage reports are based on templates. When you change a template, the changes are propagated to all of the reports that use the template. In the following procedure, we create the new metric for the new service level in a lower report node, and then use the metric in a new column in the Associated Storage report. The change is propagated to all Associated Storage reports.

**Procedure**

1. Navigate to an Associated Storage report. For example, go to **Explore > Hosts > All Hosts > host_name**.
2. Click the **Capacity** tab.
3. Click the **Associated Storage** report title, so that the report is at the top of the page.
4. Click **EDIT MODE**.
5. Click the **Associated Storage** template link in the **Linked to** field.
6. In the left pane, expand the Associated Storage node to show the use cases.
7. Select a use case.
8. In the right pane, click the **Formula** tab.
9. Select a **util.Nop** formula to copy.

   Do not select a **util.Nop4** formula. If there is no **util.Nop** formula, create a **util.Nop** formula from scratch, copying the filter and other information from the existing **util.Nop4**.

**Note**

These notations are used to create metrics that are pushed up the reporting hierarchy. **Nop** creates one result, and **Nop4** is a shortcut for creating four results. You do not have four results so you cannot use a **Nop4** formula.
10. Click **Copy**.
11. Click **Paste**.
   A new blue bar appears.
12. Expand the new bar.
13. Expand **Parameter to Enter**.
    a. Click the **Edit** icon next to the filter.
    Right-click in the svclevel box and choose **Edit expression**.
    b. In the Edit dialog, change the svclevel value to your custom service level.
    c. Click **Ok**.
14. Expand **Result Returned**.
    a. In **Name**, create a name for the new service level metric.
15. Click **Save**.
16. In the left pane, click to select the **Associated Storage** node.
17. In the right pane, click the **Report Details: Table** tab.
18. Add a new column:
a. Click to select one of the existing Value: blue bars.
b. Click Copy.
c. Click Paste.
d. Scroll down to find the copied blue bar. (It has an * after the name).
e. Change the Column Name.
f. Optionally change the Description.
g. In Selected Value, click the arrow and select the metric name you created previously from the drop-down list.
h. In Use time settings from, select this column.

19. Drag and drop the blue bar to place the column where you want it in the report.
20. Scroll down and click Save.
21. Click BROWSE MODE.
22. Navigate to any host Associated Storage report to verify the appearance of the new column.

![Image of Associated Storage report]

**About Block Chargeback Groupings**

Block chargeback reports are generated for each predefined and custom-defined block chargeback grouping.

A Block Chargeback Grouping is a collection of hosts. The hosts can be VMs, hypervisors, physical hosts, or passively discovered hosts.

Block chargeback processing occurs at the host level. The computed costs are based on all storage connected to the host devices in the grouping. When you create a custom Block Chargeback Grouping, the members of the group must be items with a devtype property equal to one of the following:

- devtype is "Host"
- devtype is "PassiveHost"
- or
devtype is "VirtualMachine"
or
devtype is "Hypervisor"

You do not need to specify devtype in the rule, but the result of the selection process must be items with a devtype equal to one of the above-listed values

---

**Note**

The Groups Management interface is open-ended, and lets you use any property to create group membership rules. However, chargeback processing will not work as expected for a grouping that includes non-host items. For example, a block chargeback grouping that selects items whose devtype is Array would not produce a row in the Block Chargeback report.

---

**Examples**

Here are some examples of membership rules for customized block chargeback groupings:

- a static collection of host, PassiveHost, VMs, and hypervisor device names
- a dynamic collection of host device names at a certain location or for a certain business unit, assuming that the location or bunit property tagging occurs on host devices
- a dynamic collection of VM device names starting with a specific string

Complicated custom reports are required to show chargeback costs based on device types other than hosts, VMs, and hypervisors. Contact Customer Support to discuss your requirements.

---

**Create customized group chargeback report**

Customized groups, along with customized service levels, provide a way to classify, analyze, and charge back for storage usage according to your business needs and SLAs.

To capture chargeback metrics for a specific customer, application, or business unit, define a block chargeback grouping that contains all of the storage hosts used by that customer, application, or business unit. ViPR SRM automatically generates chargeback reports for every group on the Chargeback by Group report.

*Figure 18 Example custom block chargeback report*
Block Chargeback
 CHAPTER 8

File Chargeback

The following topics describe file chargeback reports and associated concepts.

- Fileshare chargeback features ................................................................. 114
- Fileshare chargeback metrics ................................................................. 114
- About fileshare service levels and cost basis ......................................... 115
- Setting cost basis for file chargeback .................................................... 115
- About file chargeback groupings .......................................................... 116
- View fileshare chargeback reports ....................................................... 116
Fileshare chargeback features

Fileshare chargeback reports show capacities and costs for storage in file systems and file shares.

The term fileshare chargeback includes file shares with tree quotas and all file systems without quota. The supported platform technologies for fileshare chargeback are:

- Classic VNX—Unified VNX and File Only VNX. (These features do not apply to VNX block arrays.)
- Unity and VNXe2—File systems only. (These features do not apply to quotas or fileshares on these systems.)
- Isilon
- NetApp

The fileshare chargeback reports cover two chargeback use cases:

- Primary Presented capacities and costs - Primary storage is the space for the original copy of data (and not for snapshots or replications). Presented capacity is available storage space.
- Primary Used capacities and costs - Primary storage is the space for the original copy of data (and not for snapshots or replications). Used capacity is the storage space actually consumed.

The presented and used metrics are shown on separate reports, allowing you to analyze and charge costs based on either presented or used capacities. Within each use case, a summary-level chargeback report shows metrics by fileshare group. You can drill progressively into the details of a group.

Fileshare chargeback metrics

The fileshare chargeback reports provide chargeback capacities and costs on combined file system and file share storage usage within a fileshare group.

Capacities used for chargeback are derived as follows:

**File shares with tree quotas**

A file share quota provides a hard limit capacity on storage consumption by disk space and by the amount of files that can be created. A file share with tree quota is represented in the database using the following parttype property value:

```
parttype='Quota'
```

For the Quota parttypes:

- **Chargeback Presented Capacity** is the tree quota hard limit capacity metric.

  **Note**

  For Isilon, if a hard limit is not defined, the Chargeback Presented Capacity is set to be the same as the tree quota used capacity.

- **Chargeback Used Capacity** is the tree quota used capacity.
**File systems or paths without quotas**

A file system or file path without tree quotas is represented in the database using the following parttype property value:

```plaintext
parttype='FileSystem'
```

For the **FileSystem** parttypes:

- **Chargeback Presented Capacity** is the file system's presented capacity
- **Chargeback Used Capacity** is the file system's used capacity

**Total fileshare chargeback (sum of quota and filesystem metrics)**

Fileshare chargeback reports combine the metrics for file systems and quotas. An additional parttype property value, **FileShare**, provides the combined metrics. All of the summary-level Fileshare Chargeback by Group reports include the following filter component, so they show the combined file system and quota metrics.

```plaintext
parttype='FileShare'
```

---

**About fileshare service levels and cost basis**

All fileshares are tagged with a fileshare service level. Chargeback costs are set by the service level.

The Groups Management group type named **Service Level for Fileshare** is used for tagging file systems with a service level. One service level, named NAS, is installed by default, and all file systems are members of this service level.

Although you can create additional service levels under Service Level for Fileshare, the fileshare chargeback reports can calculate chargeback metrics only for the NAS service level.

The cost basis for calculating fileshare chargeback is associated with the NAS service level.

**Setting cost basis for file chargeback**

Chargeback reports use a cost per GB configured on the service level.

**Procedure**

1. Click **Administration > Centralized Management > Groups Management > Service Level by File Share**.
2. Click to select the NAS service level name, and then click **Edit**.
3. In **Cost per GB**, type the cost basis for this service level.
   
   Cost is numerical with no assumption of any specific currency. Enter an integer or a decimal value. If you enter 3, the cost for 100GB is 300. If you enter .3, the cost for 100GB is 30.
4. Click **Save**.
About file chargeback groupings

The fileshare chargeback reports show metrics by file chargeback groups. The Groups Management group type named File Chargeback Grouping defines groupings of file systems and file shares for reporting purposes. Each defined group is a row in the chargeback summary reports. Drilldown from the summary row shows a group-specific chargeback report.

The following default File Chargeback Grouping groups are provided at installation:

- All Isilon
- All NetApp
- All Unity and VNXe2
- All VNX

You can create customized groups to identify all of the fileshares for a specific customer, business unit, or other purpose. Define your customized groups in a hierarchy of groups under the default groups, or mix file system types together in a top level group.

To create a filter that includes file shares with tree quota and file systems without quotas, you can include the following phrase in the group filter:

```
parttype Is FileShare
```

Note

To avoid duplication of space in capacity metrics, do not include both a parent and a child file system in the same group.

View fileshare chargeback reports

The fileshare chargeback reports help you analyze the true cost of file system storage.

Procedure


2. Click the tab for the use case you want to view, either Used Chargeable or Presented Chargeable.

   The reports summarize the amount of file storage and chargeable costs for each file chargeback group.
3. Click a row to see details by array for a group.
   The details appear below the summary report.

   ![Image](image1.png)

4. Click on the **Array Name** to drill down into the array and view fileShare and file system names, and a detailed cost breakdown.

   ![Image](image2.png)

Here is another example that includes file systems and quotas. Notice that the number of elements in the Fileshare report equals the sum of the elements in the File System and Tree Quota reports. The Presented Capacity column in the File Share report shows the Presented Capacities for each file system and the Hard Limit Capacity for each tree quota.
### FileShare

<table>
<thead>
<tr>
<th>Device</th>
<th>File Share</th>
<th>Faid</th>
<th>Fname</th>
<th>File Share Type</th>
<th>Service Level</th>
<th>UsedCapacity</th>
<th>PresentedCapacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>dev</td>
<td>File Share</td>
<td>201</td>
<td>Thick</td>
<td>File Share</td>
<td>SAS</td>
<td>0.00</td>
<td>0.90</td>
</tr>
<tr>
<td>asa</td>
<td>File Share</td>
<td>201</td>
<td>Thick</td>
<td>File Share</td>
<td>SAS</td>
<td>0.00</td>
<td>0.34</td>
</tr>
<tr>
<td>thick</td>
<td>File Share</td>
<td>201</td>
<td>Thick</td>
<td>File Share</td>
<td>SAS</td>
<td>0.00</td>
<td>0.50</td>
</tr>
<tr>
<td>nightfuryvds_fs_1</td>
<td>File System</td>
<td>221</td>
<td>nightfuryvds_fs_1</td>
<td>SAS</td>
<td>SAS</td>
<td>98.41</td>
<td>110.11</td>
</tr>
<tr>
<td>ryadhms_fs_1</td>
<td>File System</td>
<td>232</td>
<td>ryadhms_fs_1</td>
<td>SAS</td>
<td>SAS</td>
<td>0.41</td>
<td>73.79</td>
</tr>
</tbody>
</table>

### File Systems

<table>
<thead>
<tr>
<th>Device</th>
<th>Part</th>
<th>Faid</th>
<th>Fname</th>
<th>UsedCapacity</th>
<th>PresentedCapacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>LJS_FS_QUOTA_USER_1AB</td>
<td>224</td>
<td>LJS_FS_QUOTA_USER_1AB</td>
<td>0.00</td>
<td>0.90</td>
<td></td>
</tr>
<tr>
<td>LJS_FS_QUOTA_GROUP_1A7</td>
<td>225</td>
<td>LJS_FS_QUOTA_GROUP_1A7</td>
<td>0.00</td>
<td>0.90</td>
<td></td>
</tr>
<tr>
<td>ryadhms_100D</td>
<td>229</td>
<td>ryadhms_100D</td>
<td>0.24</td>
<td>0.79</td>
<td></td>
</tr>
<tr>
<td>nightfuryvds_fs_1</td>
<td>231</td>
<td>nightfuryvds_fs_1</td>
<td>98.41</td>
<td>110.11</td>
<td></td>
</tr>
<tr>
<td>ryadhms_fs_1</td>
<td>232</td>
<td>ryadhms_fs_1</td>
<td>0.41</td>
<td>73.79</td>
<td></td>
</tr>
</tbody>
</table>

### Tree Quota

<table>
<thead>
<tr>
<th>Device</th>
<th>Part</th>
<th>Faid</th>
<th>Fname</th>
<th>UsedCapacity</th>
<th>HardLimitCapacity</th>
</tr>
</thead>
<tbody>
<tr>
<td>asa</td>
<td>201</td>
<td>Thick</td>
<td>0.00</td>
<td>0.34</td>
<td></td>
</tr>
<tr>
<td>dev</td>
<td>201</td>
<td>Thick</td>
<td>0.00</td>
<td>0.50</td>
<td></td>
</tr>
</tbody>
</table>
CHAPTER 9

Object Chargeback

The following topics describe object chargeback reports and associated concepts.

- View ECS chargeback reports ................................................................. 120
- Bucket service levels ............................................................................ 123
- Edit bucket service level cost basis ....................................................... 123
- Create custom bucket service level ....................................................... 124
View ECS chargeback reports

A set of object chargeback reports shows ECS storage usage and calculated costs for service levels within namespaces and buckets.

---

**Note**

Information about chargeback reports applies only when the SolutionPack for EMC ECS is licensed for Full ViPR SRM (VIPRSRM_ECS_TB license).

The ECS chargeback reports provide the information needed to reclaim costs of storage from the owners of ECS namespaces or buckets, and to analyze storage usage and trends. The reports show usage metrics by a variety of attributes, described below.

- **Chargeback by Namespace** shows:
  - Total size of objects (GB) in a namespace, by service level
  - Total object count in the namespace
  - Numbers of objects created and deleted during the reporting period
  - Data uploaded and downloaded during the reporting period
  - Computed chargeback costs

- Click a namespace in the table to see additional information, including:
  - Daily object counts and used capacity for the namespace
  - Network usage and trends for the namespace
  - Simple chargeback report for the namespace by service level
  - **Chargeback by Bucket** report for the namespace

- **Chargeback by Bucket** shows:
  - Virtual data center and replication group name associated to each bucket
  - Total object count in the bucket
  - Numbers of objects created and deleted in the bucket during the reporting period
  - Data uploaded and downloaded in the bucket during the reporting period
  - Computed chargeback costs for the bucket

- Click a bucket in the table to see additional information, including:
  - Daily object counts and used capacity for the bucket
  - Network usage and trends for the bucket
  - Simple chargeback report for the bucket by service level

**Procedure**

1. Log in to the Console and navigate to **Operations > Chargeback > Object Chargeback**.
The **Total Cost** column at the far right uses the cost per GB value assigned to the service level in **Administration > Centralized Management > Groups Management > Bucket Service Levels**.

2. Click a namespace name to explore it in more detail.

   A set of tabbed reports appears for the namespace.

3. Click **Chargeback by Bucket** to view the cost breakdown within buckets.
4. Click **Chargeback by Service Level** for a simple report of cost per service level within the bucket.
Bucket service levels

The SolutionPack for EMC ECS uses bucket service levels to classify object storage in ECS namespaces. You can create custom bucket service levels and edit existing ones. You can edit the service level cost basis for chargeback computations.

Note

Information about bucket service levels applies only when the SolutionPack for EMC ECS is licensed for Full ViPR SRM (VIPRSRM_ECS_TB license).

The bucket service levels are based on replication policy. A bucket that has the characteristics defined by the rules in the service level definition is assigned to that service level.

There are two out-of-the-box service levels:

- **Locally protected**—This service level is assigned to buckets that are associated to one site, meaning that data is local only.
- **Remotely protected**—This service level is assigned to buckets that are associated to multiple sites, meaning that data is copied offsite.

Edit bucket service level cost basis

Change the cost basis used for the ECS bucket chargeback computations.

Procedure

1. Log into the Console and navigate to Administration > Centralized Management > Groups Management > Bucket Service Levels.
2. In the list under All, click a service level to select it.
3. Click Edit.
4. To change the cost basis, in the Cost per GB field, type a new value, and then click Save.
Create custom bucket service level

Create additional bucket service levels with different or more specific criteria than the out-of-the-box levels.

Suppose you want to create a bucket service level for a locally protected specific namespace and chargeback a different rate for that namespace.

**Procedure**

1. Log into the Console and navigate to Administration > Centralized Management > Groups Management > Bucket Service Levels.
2. Click All to select it, and then click Create. By selecting All, the new service level will appear directly under All in the list.
3. In Service Level, type a name for the new service level.
4. In Cost per GB, type a decimal value, such as .02, to represent a monetary unit.
5. For the first rule, in Select a Property, select Component type, and fill in the next two fields with Is Bucket.
6. Click Add rule.
7. Create a second rule to define the replication as locally protected.
   numsites Is 1
8. Click Add rule.
9. Create a third rule to define the namespace.
   a. In Select a Property, select Advanced.
   b. Fill in the next two fields with ns Is.

   **Note**

   The database property name for namespace is ns.

   c. In the third field, select or type the namespace value.
      For example, you might have a rule ns Is apzatcity.
   d. Click Save.
10. If you are not sure of the database property name, try this method to research it:
    a. Find a report that contains a column for the property you want.
    b. Click EDIT MODE at the top of that report page.
    c. Click the Reports Detail:Table tab.
    d. Click the blue bar for the column name.
    e. In the Property field, you see the database property name used for that column. You can use that value in your rules.
11. Navigate to the Chargeback by Namespace report.
12. After sufficient time for new collections, the Other Service Levels column begins to show metrics for the new service level.
13. To create a column specifically for the new service level, edit the report definition to add a new column.
CHAPTER 10
Cloud Chargeback

The following topics describe cloud chargeback reports and associated concepts.

- Chargeback for S3 buckets on ECS resources .................................................128
- Set cost basis for S3 bucket chargeback .........................................................128
- Create custom S3 bucket grouping.................................................................128
Chargeback for S3 buckets on ECS resources

The SolutionPack for Amazon AWS can generate chargeback reports for S3 buckets that are managed on EMC ECS resources.

These reports have dependencies on AWS billing reports. See the installation instructions for the SolutionPack for Amazon AWS for required settings in the Amazon AWS account.

The chargeback reports for S3 buckets appear under Report Library > Amazon AWS > Operations in ViPR SRM.

The reports show ECS-managed S3 buckets, optionally grouped into customized groupings, with chargeable costs computed using the cost basis associated with the group names. The groups are managed using the ViPR SRM Groups Management feature, in the group type called ECS Capacity Rates.

The following chargeback costs are calculated for each grouping:

- Storage cost
- Data transfer (downloads) cost
- Data transfer (uploads) cost

One predefined grouping for all S3 buckets is installed by default. You can create customized groupings using dynamic rules or static selections.

Set cost basis for S3 bucket chargeback

S3 chargeback reports use costs per GB configured on the ECS Capacity Rate groupings.

Procedure

1. Click Administration > Centralized Management > Groups Management > ECS Capacity Rates.
2. To edit the costs per GB for the default S3 grouping, which applies to all S3 buckets:
   a. Click to select the S3 group.
   b. Click Edit.
   c. In the cost fields, type the cost basis.
      
      Cost is numerical. Enter an integer or a decimal value. If you enter 3, the cost for 100GB is 300. If you enter .3, the cost for 100GB is 30.
   
   d. Click Save.
3. To create a different cost basis for a subset of S3 buckets, create a new grouping.

Create custom S3 bucket grouping

You can create custom S3 bucket groupings for the purposes of chargeback reporting.

To identify S3 buckets in a dynamic ruleset, include the following phrases in the ruleset:

`prodcode is S3 AND parttype is Bucket`
Procedure

1. Go to Administration > Centralized Management > Groups Management > ECS Capacity Rates.

2. Select S3 or some other existing group to select the location of the new group in the hierarchy of groups. Then click Create.

3. Type a new ECS Capacity group name.
   This is the name of the new bucket grouping that will appear in reports.

4. To select buckets for the new group using rules:
   a. Select the tab Dynamic Members - select using rules.
   b. Create a rule.
      To identify S3 buckets in a dynamic ruleset, include the following phrases in the ruleset:
      prodcode is S3 AND parttype is Bucket
      Click Add rule to Boolean AND additional phrases to a rule set.
      Click Add new rule set to Boolean OR additional rule sets.
   c. Click Show Members to display the results of the rules.
   d. Click Save.

5. To select buckets for the new group manually from a list of devices:
   a. Select the tab Static Members - select from list.
   b. Use the Type and Source Group fields to define the list of objects that appear in the table.
   c. Select objects from the table and click Add to group. Repeat until the new group is complete.
   d. When the new group is complete, click Save.