



EMC® Watch4net®

Version 6.4u2

Event Mapping Administration Guide

P/N: pending

Rev : 45429

EMC²

Copyright ©2011 - 2014 EMC Corporation. All rights reserved. Published in the USA.

Published October, 2014

EMC believes the information in this publication is accurate as of its publication date. The information is subject to change without notice.

The information in this publication is provided as is. EMC Corporation makes no representations or warranties of any kind with respect to the information in this publication, and specifically disclaims implied warranties of merchantability or fitness for a particular purpose. Use, copying, and distribution of any EMC software described in this publication requires an applicable software license.

EMC², EMC, and the EMC logo are registered trademarks or trademarks of EMC Corporation in the United States and other countries. All other trademarks used herein are the property of their respective owners.

For the most up-to-date regulatory document for your product line, go to EMC Online Support (<https://support.emc.com>).

Contents

Preface	v
Purpose	v
Intended Audience	v
Prerequisites	v
Document Organization	vi
Where to get help	vii
1 Introduction	1
1 Classifying Fields	2
2 Configuration	5
1 Age	5
1.1 Description	5
1.2 Parameters	5
1.3 Example	5
2 Code Value	6
2.1 Description	6
2.2 Parameters	6
2.3 Example	7
3 Current Timestamp	7
3.1 Description	7
3.2 Parameters	7
3.3 Example	7
4 Date Seconds	7
4.1 Description	7
4.2 Parameters	8
4.3 Example	8
5 Formatted Timestamp	8
5.1 Description	8
5.2 Parameters	8

5.3	Example	9
6	Formatted Timestamp Milliseconds	9
6.1	Description	9
6.2	Parameters	9
6.3	Example	10
7	Graphable Value	10
7.1	Description	10
7.2	Parameters	10
7.3	Example	10
8	Greater Than Property	11
8.1	Description	11
8.2	Parameters	11
8.3	Parameters restriction	11
8.4	Example	11
9	Integer IPv4	12
9.1	Description	12
9.2	Parameters	12
9.3	Example	12
10	Integer IPv4 Netmask	12
10.1	Description	12
10.2	Parameters	13
10.3	Example	13
11	Integer IPv4 Network	13
11.1	Description	13
11.2	Parameters	13
11.3	Example	13
12	Integer IPv6	14
12.1	Description	14
12.2	Parameters	14
12.3	Example	14
13	Linked Property	15
13.1	Description	15
13.2	Parameters	15
13.3	Example	15
14	Min Property	16
14.1	Description	16
14.2	Parameters	16

14.3 Example	16
15 Multi Property	16
15.1 Description	16
15.2 Parameters	17
15.3 Example	17
16 Multi Timestamp	17
16.1 Description	17
16.2 Parameters	17
16.3 Example	18
17 Multi Timestamp Millis	18
17.1 Description	18
17.2 Parameters	18
17.3 Example	19
18 Network Application	19
18.1 Description	19
18.2 Parameters	19
18.3 Example	20
19 Numeric Value	20
19.1 Description	20
19.2 Parameters	20
19.3 Example	21
20 Resolved Integer IPv4	21
20.1 Description	21
20.2 Parameters	21
20.3 Example	21
21 Static Property	22
21.1 Description	22
21.2 Parameters	22
21.3 Example	22
22 Strict Boolean	22
22.1 Description	22
22.2 Parameters	22
22.3 Example	23
23 String Code Property	23
23.1 Description	23
23.2 Parameters	23
23.3 Example	23

24	String Value	24
24.1	Description	24
24.2	Parameters	24
24.3	Example	24
25	Timestamp Milliseconds	25
25.1	Description	25
25.2	Parameters	25
25.3	Example	25
26	Timestamp Seconds	25
26.1	Description	25
26.2	Parameters	25
26.3	Example	26
27	Timestamp to Numeric adapter	26
27.1	Description	26
27.2	Parameters	26
27.3	Example	26
28	TopN Property	27
28.1	Description	27
28.2	Parameters	27
28.3	Example	27
29	Unit	27
29.1	Description	27
29.2	Parameters	28
29.3	Example	28
30	Value Selection	28
30.1	Description	28
30.2	Parameters	28
30.3	Example	28
3	Annexe	29
1	Constants for date	30

Preface

Purpose

The purpose of this guide is to provide information about configuration of the *Event Mapping*.

Intended Audience

This guide is intended to be read by any administrator who will need to configure the *Event Mapping*.

Prerequisites

Watch4net's *APG* Frontend, version 5.1 or up, must be installed.

Document Organization

1. Introduction	Introduces how and why the <i>Event Mapping</i> takes place, and discusses how to classify properties
2. Configuration	Covers the configuration of the <i>Event Mapping</i>

Where to get help

EMC support, product, and licensing information can be obtained as follows:

PRODUCT INFORMATION - For documentation, release notes, software updates, or information about EMC products, go to EMC Online Support at:

<https://support.emc.com>

TECHNICAL SUPPORT - Go to EMC Online Support and click Service Center. You will see several options for contacting EMC Technical Support. Note that to open a service request, you must have a valid support agreement. Contact your EMC sales representative for details about obtaining a valid support agreement or with questions about your account.

1

Introduction

The *Event Mapping* takes place within one or more property mapping files that relate field names in the database to the property names they will have in the Frontend, which might be different.

Since events are unpredictable and can have different sorts of naming conventions for their field content, often depending on their source, and how they are interpreted and manipulated by processing elements in *APG*, the *Event Mapping* allows you to rationalize and clarify how fields will be named in the Frontend, removing duplicate names coming from different tables, providing more meaning to names, and generally organizing and adjusting the behavior of event properties for the Frontend end-user.

Property mapping files should be located in the: `<APG>/Custom/WebApps-Resources/<instance name>/property-mapping` directory. Typically, they are referenced in the **APG.xml** file where they relate directly to a table in the events database.

For example, in the following entry from the **APG.xml**, which gives access to a table containing basic text output from a *JMS Listener*:

```
<!-- Gives access to jms_text. -->
```

```
<ResourceLink name="jdbc/FLOW-JMS-TEXT:{_mapping=JMS-TEXT.xml,_table=jms_text_}"
  global="jdbc/FLOW-RPE2" type="javax.sql.DataSource" />
```

The property mapping file JMS-TEXT.xml maps the field names, as processed by the *JMS Listener*, and written to the database table `jms-text` by the *Datastore-Writer*, to their Frontend equivalents. The property mapping file is entered as a value for `mapping`, and the table in the events database that the property mapping relates to is entered as a value for `table`.

The properties in the property mapping file indicate that the table they relate to is determined outside of the file, as mentioned in the **APG.xml**, by including the tags and values as follows: `<table class="string">${table}</table>`. This allows, for example, for two flows written to different tables to use the same mapping file, since separate tables can be indicated in the **APG.xml** file.

Note: Although not typically done, you can hardcode a specific event table within the `<table class="string"></table>` tags to relate a particular property to a particular table.

In general, you would have a unique mapping file for each data flow that terminates at its own *database writer*, written to a different database table with a unique schema, and field composition. In practice this often means a unique table for each listener, though occasionally they can overlap in a single file if their fields are consistent, from perhaps two different listeners, or multiple instances of the same listener, or multiple flows that have otherwise been branched.

It is also mandatory to classify fields in the field mapping file to determine certain behaviors. This is covered in the next section.

1 Classifying Fields

There are three field types to use for properties in a property mapping file:

- *timestamp*: This indicates what column of the table to use for the event's timestamp values. It is mandatory and can only be used once per mapping file. It is included as an attribute in property tags with the following syntax: `timestamp="true"`.
- *key*: Key properties identify events as unique in the Frontend. Having at least one key property is mandatory, while there is no limit to the number of keys that could be used. Keys should be chosen that uniquely identify the event so it is usually good practice to combine DEFINITION fields, for example identifying a particular router, with OCCURANCE fields, for example identifying a

particular type of event, as an event's key properties.

Event fields are either of type `DEFINITION` or `OCCURENCE`, which are assigned at their source, and can be determined, in some circumstances, if you are configuring the source, if the listener is part of a customized *APG* package where mapping files are prepared beforehand by Watch4net, or by examining logs generated by an EventSpy, for example.

Key properties can only include the following characters: they can be alphanumeric, and include "." or ":" characters. A key property can also be used as a timestamp value, however some mappings, for example `FormattedTimestamp` (see [8](#)), will not work because this mapping includes characters not allowed in the key, such as spaces.

A key property is indicated by its inclusion as an attribute in property tags with the following syntax: `key="true"`.

- *default*: The default attribute indicates the field in the event that will be plotted in graphs. Unlike in time series values, where multiple metrics can be plotted that relate to a single property, for events only one value can be plotted at a time. This default value is used if the user themselves do not specify another value from the event to use.

A default type is mandatory and can only be used once in the mapping file. This property must be a `GraphableValue`. See [Graphable Value](#) on page [10](#). It is included as an attribute in property tags with the following syntax: `default="true"`.

Configuration

1 Age

1.1 Description

Age of a database value.

1.2 Parameters

relative-to the internal timestamp property

table the table to use

display-name the name that will be displayed in APG

1.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.Age">
  <relative-to name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.
TimestampMillis" timestamp="true">
    <name class="string">UNIX_MS</name>
    <table class="string">${table}</table>
    <display-name class="string">timestamp</display-name>
  </relative-to>
  <table class="string">${table}</table>
  <display-name class="string">age</display-name>
</property>
```

Creates a value from the column UNIX_MS in the given table named age in the Frontend. The value is the CURRENT TIME - (minus) the timestamp.

2 Code Value

2.1 Description

Property from a mapping file.

2.2 Parameters

name the internal name of the property

table the table to use

display-name the name that will be displayed in APG

mapping-file the name of the mapping file. Can be overwritten if placed in the custom directory of Generic-Flow

case (Optional) the option in order to handle the case of the resulting property. By default, the properties will be set to lowercase. Available options are:

lowercase : All the values will be set to lowercase. Option accept lowercase or lower.

uppercase : All the values will be set to uppercase. Option accept uppercase or upper.

file_defined : All the values will take as is from the file. Option accept file_defined, untouched or keep.

2.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.CodeValue" key="true">
  <name class="string">PROT</name>
  <table class="string">\${table}</table>
  <display-name class="string">protocol</display-name>
  <mapping-file class="string">protocols</mapping-file>
</property>
```

Creates a key value named `protocol` in lowercase in the Frontend, from the column `PROT` matched in the `protocols` mapping-file in the given table.

3 Current Timestamp

3.1 Description

Property that can be a timestamp based on the current time defined by the report preferences.

3.2 Parameters

display-name the name that will be displayed on APG

3.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.CurrentTimestamp"
timestamp="true">
  <display-name class="string">currentTime</display-name>
</property>
```

Create a value with name `currentTime`, defined by the current time from the report in the frontend.

4 Date Seconds

4.1 Description

An in-database formatted date selection.

4.2 Parameters

name the internal name of the property

table the table to use

display-name the name that will be displayed in APG

to-date-function the timestamp to date conversion code where ? will be replaced by the actual timestamp (optional, defaults to "DATE(FROM_UNIXTIME(?))")

date-format the in-database date format, to be able to parse returned values (optional, defaults to "yyyy-MM-dd")

calendar-field the calendar field, which represents the precision of the selected date (optional, defaults to Calendar.DATE (5))

4.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.DateSec">
  <name class="string">DATE</name>
  <table class="string">\${table}</table>
  <display-name class="string">date</display-name>
  <to-date-function class="string">DATE(FROM_UNIXTIME(?))</to-date-function>
  <date-format class="string">yyyy-MM-dd</date-format>
  <calendar-field class="int">5</calendar-field>
</property>
```

Creates a value from the column DATE in the given table named, which will be named date in the Frontend.

5 Formatted Timestamp

5.1 Description

A FormattedTimestamp is a well formatted timestamp property.

5.2 Parameters

name the internal name of the property

table the table to use

display-name the name that will be displayed in APG

format (optional, defaults to "yyyy/MM/dd HH:mm:ss z")

truncation-field the field to truncate date with (optional, defaults to Calendar.SECOND (13)) See the [Annexe](#) on page 29 for the codes used to indicate where the truncation will take place.

5.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.FormattedTimestamp"
">
  <name class="string">UNIX_SECS</name>
  <table class="string">\${table}</table>
  <display-name class="string">date</display-name>
  <truncation-field class="int">13</truncation-field>
</property>
```

Creates a value from the column UNIX_SECS in the given table, that will be named date in the Frontend, and will be truncated at seconds.

You can use this value as the timestamp of the event by including timestamp="true" as an attribute in the property tags.

6 Formatted Timestamp Milliseconds

6.1 Description

A FormattedTimestamp is a well formatted timestamp property.

6.2 Parameters

name the internal name of the property

table the table to use

display-name the name that will be displayed in APG

format (optional, defaults to "yyyy/MM/dd HH:mm:ss z")

truncation-field the field to truncate date with (optional, defaults to Calendar.SECOND). See the [Annexe](#) on page 29 for the codes used to indicate where the truncation will take place.

6.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.
FormattedTimestampMillis">
  <name class="string">TIME</name>
  <table class="string">\${table}</table>
  <display-name class="string">time</display-name>
  <truncation-field class="int">13</truncation-field>
</property>
```

Creates a value from the column TIME in the given table, which will be named time in the Frontend, and be truncated at seconds.

You can use this value as the timestamp of the event by including timestamp="true" as an attribute in the property tags.

7 Graphable Value

7.1 Description

A property which is a numeric value on which graphs can be generated.

7.2 Parameters

name the internal name of the property

table the table to use

display-name the name that will be displayed in APG

unit the unit (optional)

7.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.GraphableValue"
key="true" default="true">
  <name class="string">DOCTETS</name>
  <table class="string">\${table}</table>
  <display-name class="string">octets</display-name>
  <unit class="string">octets</unit>
</property>
```

Creates a default key graphable value from the column DOCTETS in the given table, which will be named octets in the Frontend, with octets as it's unit.

8 Greater Than Property

8.1 Description

Implements a comparison between two properties.

8.2 Parameters

property1 The first property to compare

property2 The second property to compare

display-name the name that will be displayed in APG

8.3 Parameters restriction

property1 and property2 are limited to these: NumericValue, StringCodeProperty, MinProperty, IntegerIPv4Netmask, GreaterThanProperty, GraphableValue, StrictBoolean, CodeValue

8.4 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.
GreaterThanProperty">
  <property1 class="property" name="com.watch4net.apg.v2.modules.database.GenericFlow.
properties.NumericValue">
    <name class="string">SRCPORT</name>
    <table class="string">\${table}</table>
    <display-name class="string">srcport</display-name>
  </property1>
  <property2 class="property" name="com.watch4net.apg.v2.modules.database.GenericFlow.
properties.NumericValue">
    <name class="string">DSTPORT</name>
    <table class="string">\${table}</table>
    <display-name class="string">dstport</display-name>
  </property2>
  <display-name class="string">greater</display-name>
```

```
</property>
```

Creates a value based on the columns SRCPORT and DSTPORT in the given table, which will be named greater in the Frontend. The value will be 1 (true) if SRCPORT > DSTPORT. 0 (false) if not.

9 Integer IPv4

9.1 Description

A property that is the IP address of a network, in the IPv4 format.

9.2 Parameters

name the internal name of the property

table the table to use

display-name the name that will be displayed in APG

9.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.IntegerIPv4">
  <name class="string">EXADDR</name>
  <table class="string">\${table}</table>
  <display-name class="string">device</display-name>
</property>
```

Creates a value from the column EXADDR in the given table, which will be named device in the Frontend. The value is converted from int to a readable string.

10 Integer IPv4 Netmask

10.1 Description

A property that describes an IP netmask in the IPv4 format.

10.2 Parameters

name the internal name of the property

table the table to use

display-name the name that will be displayed in APG

10.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.IntegerIPv4Netmask"
">
  <name class="string">SRC_MASK</name>
  <table class="string">\${table}</table>
  <display-name class="string">srcmask</display-name>
</property>
```

Creates a value from the column SRC_MASK in the given table, which will be named `srcmask` in the Frontend.

11 Integer IPv4 Network

11.1 Description

A property that is an IP address describing a network.

11.2 Parameters

name the internal name of the property

table the table to use

display-name the name that will be displayed in APG

mask the mask property

11.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.IntegerIPv4Network">
  <name class="string">SRCADDR</name>
  <table class="string">\${table}</table>
  <display-name class="string">srcnet</display-name>
  <mask class="property" name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.IntegerIPv4Netmask">
    <name class="string">SRC_MASK</name>
    <table class="string">\${table}</table>
    <display-name class="string">srcmask</display-name>
  </mask>
</property>
```

Creates a value from the columns SRCADDR and SRC_MASK in the given table, which will be named srcnet in the Frontend, and where the result is the network IP (computed from the given IP and mask).

12 Integer IPv6

12.1 Description

A duo of properties that are the combination of the IP address of a network, in the IPv6 format.

12.2 Parameters

upper-part the name of the property where the upper half (left part) of the address is

lower-part the name of the property where the lower half (right part) of the address is

table the table to use

display-name the name that will be displayed in APG

12.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.IntegerIPv6">
  <upper-part class="string">SRCADDR6_UP</upper-part>
  <lower-part class="string">SRCADDR6_DN</lower-part>
  <table class="string">\${table}</table>
  <display-name class="string">srcaddr</display-name>
</property>
```

Combines the values from the columns SRCADDR6_UP and SRCADDR6_DN in the given table, which will be named `srcaddr` in the Frontend. The values are converted from `int` to a readable string in hexadecimal format.

13 Linked Property

13.1 Description

A `LinkedProperty` is a property that relies on two different database values.

13.2 Parameters

properties the properties used in the link

display-name the name that will be displayed in APG

delimiter an optional delimiter (defaults to "-" when omitted)

13.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.LinkedProperty">
  <properties class="list">
    <property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.IntegerIPv4">
      <name class="string">SRCADDR</name>
      <table class="string">\${table}</table>
      <display-name class="string">srcaddr</display-name>
    </property>

    <property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.IntegerIPv4">
      <name class="string">DSTADDR</name>
      <table class="string">\${table}</table>
      <display-name class="string">dstaddr</display-name>
    </property>
  </properties>
  <display-name class="string">addr</display-name>
  <delimiter class="string">/</delimiter>
</property>
```

Creates value from the columns SRCADDR and DSTADDR in the given table, which will be named `addr` in the Frontend, and where the value is the concatenation of values from the columns separated by `"/"`.

14 Min Property

14.1 Description

A min property chooses the minimum value among all given properties.

14.2 Parameters

display-name the name that will be displayed in APG

properties the list of properties

14.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.LinkedProperty">
  <properties class="list">
    <property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.IntegerIPv4">
      <name class="string">SRCADDR</name>
      <table class="string">\${table}</table>
      <display-name class="string">srcaddr</display-name>
    </property>

    <property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.IntegerIPv4">
      <name class="string">DSTADDR</name>
      <table class="string">\${table}</table>
      <display-name class="string">dstaddr</display-name>
    </property>
  </properties>
  <display-name class="string">addr</display-name>
</property>
```

Creates a value from the smallest value from columns SRCADDR and DSTADDR in the given table, which will be named addr in the Frontend.

15 Multi Property

15.1 Description

Will return all the distinct values, or a list of flow properties, in a single APG property.

15.2 Parameters

properties The list of Flow Properties to aggregate

display-name The name of the property in APG

15.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.MultiProperty">
  <properties class="list">
    <property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.IntegerIPv4">
      <name class="string">SRCADDR</name>
      <table class="string">\${table}</table>
      <display-name class="string">srcaddr</display-name>
    </property>

    <property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.IntegerIPv4">
      <name class="string">DSTADDR</name>
      <table class="string">\${table}</table>
      <display-name class="string">dstaddr</display-name>
    </property>
  </properties>
  <display-name class="string">addr</display-name>
</property>
```

Creates a property which will be named `addr` in the Frontend, that will contain all the distinct values between the columns `SRCADDR` and `DSTADDR` in the desired table.

16 Multi Timestamp

16.1 Description

Create a property that will select all the values that have a duration overlapping the report time range. Start and end times must be within the report time range. The timestamp in the database are considered to be in seconds.

16.2 Parameters

opened-at The column containing the start timestamp.

closed-at The column containing the end timestamp.

missing-value (Optional) If the timestamp columns can't contains null values (which is the case, for example in an Oracle database), this value will be added in where an event is missing a timestamp value (Example: event not closed yet). Default is 0.

table The table to query

display-name The name of the property in APG

16.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.MultiTimestamp">
  <opened-at class="string">EVENT_START</opened-at>
  <closed-at class="string">EVENT_END</closed-at>
  <missing-value class="string">-1</missing-value>
  <table class="string">\${table}</table>
  <display-name class="string">eventPeriod</display-name>
</property>
```

Creates a property which will be named `eventPeriod` in the Frontend, and that will be returned when the content of the columns `EVENT_START` and `EVENT_END` are overlapping and within the report time range or when the property value will be -1.

17 Multi Timestamp Millis

17.1 Description

Create a property that will select all the values that have a duration overlapping the report time range. Start and end times must be within the report time range. The timestamp in the database are considered to be in milliseconds.

17.2 Parameters

opened-at The column containing the start timestamp.

closed-at The column containing the end timestamp

missing-value (Optional) If the timestamp columns can't contain a null value (which is the case, for example in an Oracle database), this value will be added in where an event is missing a timestamp value (Ex: event not closed yet). Default is 0

table The table to query

display-name The name of the property in APG

17.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.MultiTimestamp">
  <opened-at class="string">EVENT_START</opened-at>
  <closed-at class="string">EVENT_END</closed-at>
  <missing-value class="string">-1</missing-value>
  <table class="string">\${table}</table>
  <display-name class="string">eventPeriod</display-name>
</property>
```

Creates a numeric property which will be named eventPeriod in the Frontend, and that will be returned when the content of the columns EVENT_START and EVENT_END are overlapping and within the time range of the report or when the property value will be -1.

18 Network Application

18.1 Description

Create a property which maps a protocol/port to an application using a services list. A default services list is provided in the frontend-core library but a custom one can be defined in the APG frontend custom files directory under generic-flow and must be named applications without a file extension. Furthermore, the format of the file must be shortApplicationName—port/protocol—longApplicationName and use a single line for each entry.

18.2 Parameters

port The Flow Property that will retrieve the port (Typically a NumericValue property).

protocol The Flow Property that will retrieve the protocol (Typically a CodeValue property).

display-name The name of the property in APG

18.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.NetworkApplication">
  <port class="property" name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.NumericValue">
    <name class="string">SRCPORT</name>
    <table class="string">\${table}</table>
    <display-name class="string">srcport</display-name>
  </port>
  <protocol class="property" name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.CodeValue">
    <name class="string">PROT</name>
    <table class="string">\${table}</table>
    <display-name class="string">protocol</display-name>
    <mapping-file class="string">protocols</mapping-file>
  </protocol>
  <display-name class="string">srcappli</display-name>
</property>
```

Creates a property, which will be named `srcappli` in the Frontend, that will contain the application name found within the content of the columns `SRCPORT` and `PROT` in the desired table.

19 Numeric Value

19.1 Description

Create a property that will be evaluated as a numeric value allowing the usage of mathematical operators on it in the APG filters.

19.2 Parameters

name The name for the column in the database

table The table to query

display-name The name of the property in APG

19.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.NumericValue">
  <name class="string">SRCPORT</name>
  <table class="string">\${table}</table>
  <display-name class="string">srcport</display-name>
</property>
```

Creates a numeric property, which will be named `srcport` in the Frontend, with the content of the column `SRCPORT` in the desired table.

20 Resolved Integer IPv4

20.1 Description

Create a resolved IPv4 address property from an integer address with reverse DNS.

20.2 Parameters

name The name for the column in the database

table The table to query

display-name The name of the property in APG

20.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.
ResolvedIntegerIPv4">
  <name class="string">EXADDR</name>
  <table class="string">\${table}</table>
  <display-name class="string">devicehost</display-name>
</property>
```

Creates a property, which will be named `devicehost` in the Frontend, with the resolved host name coming from the column `EXADDR` in the desired table.

21 Static Property

21.1 Description

Create a property with a constant value.

21.2 Parameters

value The constant value of the property

display-name The name of the property in APG

21.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.StaticProperty">  
  <display-name class="string">devtype</display-name>  
  <value class="string">Router</value>  
</property>
```

Creates a constant property, which will be named devtype in the Frontend, with the value Router.

22 Strict Boolean

22.1 Description

Creates a numeric graphable value property for a true/false column in the database that can't implicitly convert true > 0 (this is the case, for example, in an Oracle database).

22.2 Parameters

name The name for the column in the database

table The table to query

display-name The name of the property in APG

22.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.StrictBoolean">
  <name class="string">STATUS</name>
  <table class="string">\${table}</table>
  <display-name class="string">online</display-name>
</property>
```

Creates a property, which will be named `online` in the Frontend, with the value true or false coming from the column `STATUS` in the desired table.

23 String Code Property

23.1 Description

Create a numeric graphable property containing a defined set of values that will be translated as human readable properties when displayed.

23.2 Parameters

name The name for the column in the database

table The table to query

display-name The name of the property in APG

mapping A map of key/value with the key being the database value and the value being the desired displayed property

23.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.StringCodeProperty"
">
  <name class="string">Active</name>
  <table class="string">\${table}</table>
  <display-name class="string">eventStateAsString</display-name>
  <mapping class="map">
    <entry>
      <string>0</string>
```

```
<string>INACTIVE</string>
</entry>
<entry>
  <string>1</string>
  <string>ACTIVE</string>
</entry>
</mapping>
</property>
```

Creates a property, which will be named `eventStateAsString` in the Frontend, with the value 0 or 1 coming from the column `Active` in the desired table, and which will be translated correspondingly in the Frontend as `INACTIVE` or `ACTIVE`.

24 String Value

24.1 Description

Create a string property from a column in the database.

24.2 Parameters

name The name for the column in the database

table The table to query

display-name The name of the property in APG

24.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.StringValue">
  <name class="string">IFINPUT</name>
  <table class="string">\${table}</table>
  <display-name class="string">ifinput</display-name>
</property>
```

Creates a string property named `ifinput` in the Frontend, containing the content of the `IFINPUT` column in the desired table.

25 Timestamp Milliseconds

25.1 Description

Create a property that can be a timestamp from a column containing the time in milliseconds.

25.2 Parameters

name The name for the column in the database

table The table to query

display-name The name of the property in APG

25.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.TimestampMillis"
timestamp="true">
  <name class="string">UNIX_MS</name>
  <table class="string">\${table}</table>
  <display-name class="string">timestamp</display-name>
</property>
```

Creates a timestamp property named `timestamp` in the Frontend, from the `UNIX_MS` column in the desired table.

26 Timestamp Seconds

26.1 Description

Create a property that can be a timestamp from a column containing the time in seconds.

26.2 Parameters

name The name for the column in the database

table The table to query

display-name The name of the property in APG

26.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.TimestampSec"
timestamp="true">
  <name class="string">UNIX_SECS</name>
  <table class="string">\${table}</table>
  <display-name class="string">timestamp</display-name>
</property>
```

Creates a timestamp property named `timestamp` from the `UNIX_SECS` column in the desired table.

27 Timestamp to Numeric adapter

27.1 Description

Create a property that will take a timestamp but returns it as if it was a numeric value. Useful if you want to use it with the `GreaterThan` property.

27.2 Parameters

timestamp The timestamp property to convert to a numeric value

27.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.
TimestampToNumericAdapter">
  <timestamp class="property" name="com.watch4net.apg.v2.modules.database.GenericFlow.
properties.CurrentTimestamp">
    <display-name class="string"></display-name>
  </timestamp>
  <display-name class="string">timestamp</display-name>
</property>
```

Create a numeric property named `timestamp` based on the `CurrentTimestamp` property.

28 TopN Property

28.1 Description

Make available a property that will allow the creation of different types of TopN reports in APG. The TopN reports will be made on the selected variable of the report with the default or a given aggregation. The format of the TopN property must be nValue[of customAggr]. The custom aggregation is optional and the possible values are average, min, max, sum or nbvalues.

28.2 Parameters

default-aggr The default aggregation to use

display-name The name to use to trigger a topN report

28.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.TopNProperty">  
  <display-name class="string">topN</display-name>  
  <default-aggr class="string">sum</default-aggr>  
</property>
```

Makes available the property `topN` that will create topN reports for the selected value with the default aggregation `sum`.

Once the TopN property is declared, it is available in the Advanced Parameters box, from the Filtering & Expansion tab, of a report's edit settings, applicable to the events using the corresponding event mapping file. When using multiple event mapping files, it should be declared in each one, and use the same display name, if you want to use the TopN property universally.

29 Unit

29.1 Description

Specify which APG property will be used to return the unit parameters in Graphable Values.

29.2 Parameters

display-name The name of the property in APG

29.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.Unit">  
  <display-name class="string">unit</display-name>  
</property>
```

Specifies that the `<unit>` tag in the Graphable Values can be retrieved with the `unit` APG property.

30 Value Selection

30.1 Description

Specify which APG property will be used to select a specific Graphable Value.

30.2 Parameters

display-name The name of the property in APG

30.3 Example

```
<property name="com.watch4net.apg.v2.modules.database.GenericFlow.properties.ValueSelection">  
  <display-name class="string">name</display-name>  
</property>
```

Specifies that the `<display-name>` tag in the Graphable Values can be retrieved with the `name` APG property.

3

Annexe

1 Constants for date

ERA	0
YEAR	1
MONTH	2
WEEK_OF_YEAR	3
WEEK_OF_MONTH	4
DATE or DAY_OF_MONTH	5
DAY_OF_YEAR	6
DAY_OF_WEEK	7
DAY_OF_WEEK_IN_MONTH	8
AM_PM	9
HOUR	10
HOUR_OF_DAY	11
MINUTE	12
SECOND	13
MILLISECOND	14
ZONE_OFFSET	15
DST_OFFSET	16
FIELD_COUNT	17