### Attribute Mapping v3.1

2015-01

This file documents the Advanced Commands you can use to tailor your feeds for the Cart Product Feed v3.1 series plugin / extension.

The specifications for Advanced Commands come in the form:

```
commandName parameter as/to parameter [Optional1] [Optional2]
```

Note: Double quotes are necessary if a parameter contains a space

The previous help documents apply to earlier versions of the plugin. Their contents may no longer be applicable. Some of the v2.0 switches will be retained throughout the v3.0 life cycle. Examples of switches are \$shipping, \$field\_delimiter, and \$timeout.

To view available product attributes (aka: the harmonized products), choose the ProductlistXML feed and click view source.

### **Table of Contents**

#### **Commands**

deleteAttribute	<u>3</u>
deleteLicenseKeys	<u>5</u>
limitOutput	<u>6</u> 7
mapAttribute	<u>7</u>
mapTaxonomy	8
rule	<u>9</u>
set	10
setAttribute	11
setAttributeDefault	12

#### Rules

concat	<u>14</u>
discount	<u>15</u>
pos	<u>17</u>
shipping	<u>18</u>
strlen	<u>19</u>
substr	<u>20</u>
tax	<u>21</u>

# **Advanced Commands**

### deleteAttribute

#### deleteAttribute attributename

Remove an attribute from a feed. For example, Google's g:price identifier normally comes from the regular price attribute. Deleting regular price will cause g:price to be suppressed.

#### **Before:**

```
<item>
  <g:id>23</g:id>
  <g:item group id>11</g:item group id>
  <title><![CDATA[Spots That Are Red]]></title>
  <description><![CDATA[Spots That Are Red]]></descri</pre>
  <g:google product category><![CDATA[Animals & Pet :
  <![CDATA[http://192.168.0.80/wordpress/?produ]</li>
  <g:image link><![CDATA[]]></g:image link>
  <g:google product category>Animals Samp; Pet Suppl:
  <g:condition>excellent</g:condition>
  <g:availability>out of stock</g:availability>
  <g:mpn>12346</g:mpn>
  <g:price>2.00 USD</g:price>
  <q:shipping>
    <g:service>Ground</g:service>
    <g:price>0.00 USD</g:price>
  </g:shipping>
</item>
```

### Add:

deleteAttribute regular price

#### After:

```
<item>
  <g:id>23</g:id>
  <g:item group id>11</g:item group id>
  <title><![CDATA[Spots That Are Red]]></title>
  <description><![CDATA[Spots That Are Red]]></descri</pre>
  <g:google product category><![CDATA[Animals & Pet S
  Link><! [CDATA[http://192.168.0.80/wordpress/?production]</li>
  <g:image link><![CDATA[]]></g:image link>
  <g:google product category>Animals &amp; Pet Suppli
  <g:condition>excellent</g:condition>
  <g:availability>out of stock</g:availability>
 <g:mpn>12346</g:mpn>
  <g:shipping>
    <g:service>Ground</g:service>
    <g:price>0.00 USD</g:price>
  </g:shipping>
</item>
```

### example:

deleteAttribute \*

Deletes all attributes. This is especially useful for ProductListXML and ProductListCSV.

# deleteLicenseKeys

### deleteLicenseKeys

Remove all license keys from the database. This affects CartProductFeed licenses, GTS Extension (Google Trusted Store) licenses, and Replace Featured Image licenses.

# **limitOutput**

### limitOutput FROM low TO high

Reduce total number of products considered in the feed.

### example:

limitOutput from 0 to 999

(outputs the first 1000 products)

limitOutput from 1000 to 1999

(outputs the next 1000 products)

limitOutput to 2000

(outputs two thousand products)

## mapAttribute

mapAttribute attribute-name as identifier-name [SpecialData true/false]

Create a custom attribute mapping. Note that the attribute must exist, so this command is best combined with setAttributeDefault.

### example:

setAttributeDefault misc as randomtext mapAttribute misc as g:misc

#### result:

```
<item>
  <g:id>23</g:id>
  <g:item group id>11</g:item group id>
  <title><![CDATA[Spots That Are Red]]></title>
  <description><![CDATA[Spots That Are Red]]></descri</pre>
  <g:google product category><![CDATA[Animals & Pet S
  <![CDATA[http://192.168.0.80/wordpress/?produ]</li>
  <g:image link><![CDATA[]]></g:image link>
  <g:google product category>Animals &amp; Pet Suppli
  <g:condition>excellent</g:condition>
  <g:availability>out of stock</g:availability>
  <g:mpn>12346</g:mpn>
  <g:price>2.00 USD</g:price>
  <g:misc>randomtext</g:misc>
  <g:shipping>
   <q:service>Ground</q:service>
    <g:price>0.00 USD</g:price>
  </g:shipping>
</item>
```

Note: The fourth parameter can be used to specify the CData format in XML which allows for a wide range of special characters. In the CSV format, setting SpecialData to true forces quotes.

### example:

mapAttribute title as example true

### result:

```
<example><![CDATA[A product Title]]></example>
```

# mapTaxonomy

mapTaxonomy source as attribute

Consults the Taxonomy table and fills in attribute information from it. Ideal for mapping data stored in 3<sup>rd</sup> party plugins. This is a command to use if even ProductListXML cannot see the field of interest and the entry in the taxonomy table (source) is known.

At the time of writing, this command only works in WordPress. Other CMS ports will ignore it.

### example:

mapTaxonomy ean13 as ean mapAttribute ean as g:upc

Creates a g:upc field containing the data found in ean13.

# rule

rule rulename (parameters)

Establishes a new rule. More information on rules is available here.

# set

set value as parameter

Configure a feed output setting. Valid set values are listed here.

## setAttribute

setAttribute attributename mapto new-name

edit an *existing* attribute mapping. This is not for adding new attribute mappings, but, rather, to reconfigure existing mappings that the feed provider already provides.

### example:

setAttribute regular price mapto US-Dollar-Price

#### result:

```
<item>
 <g:id>23</g:id>
 <g:item group id>11</g:item group id>
 <title><![CDATA[Spots That Are Red]]></title>
 <description><![CDATA[Spots That Are Red]]></descri
 <g:google product category><![CDATA[Animals & Pet S</pre>
 <![CDATA[http://192.168.0.80/wordpress/?produ]</li>
 <g:image link><![CDATA[]]></g:image link>
  <g:google product category>Animals &amp; Pet Suppli
  <g:condition>excellent</g:condition>
 <g:availability>out of stock</g:availability>
  <g:mpn>12346</g:mpn>
 <US-Dollar-Price>2.00 USD</US-Dollar-Price>
  <g:shipping>
   <g:service>Ground</g:service>
   <g:price>0.00 USD</g:price>
  </g:shipping>
</item>
```

# setAttributeDefault

setAttributeDefault attribute-name as defaultvalue

Configure the default value for an attribute. This is especially helpful when creating new attributes.

### example:

(See the mapAttribute command for an example of setAttributeDefault)

# Rules

## concat

rule concat(string1, string2, string3, ...) as name

Joins several strings into name.

### example:

```
rule concat(title, " ", sku, "456") as combo_title setAttributeDefault title as $combo_title
```

### result:

(sets a default value for title equal to the concatenation of all four parameters

### example:

```
rule pos(title, "on sale") as $sale_title_pos
rule concat("Title position", $sale_title_pos) as $sale_title_combined
setAttributeDefault combined title as $sale_title_combined
```

### result:

(nested rules resolve in order of appearance)

## description

rule description(parameter1, parameter2, ...)

Modifies a product description

Note: There is only ever one description rule. When two or more description rules co-exist, the most recent overwrites the others.

Note: description() accepts parameters in any order

### example:

rule description(maximum\_length=50)

#### result:

Trims the product description to 50 characters.

### example:

rule description(long, strict, replacement character = )

### result:

Forces the description to use the long format is available (as opposed to an excerpt). Enables replacement of special characters

Sets the replacement value to an underscore

### List of parameters

long Force long description short Force product excerpt

length Synonymous to maximum length

maximum length=x trim output to x characters

replacement character = x when using strict mode, use x as the replacement character

strict disallow special characters and control characters

allow empty title never substitute description into title

# discount

rule discount(parameter1, parameter2)

Creates a custom discount for the feed.

Note: discount() accepts parameters in any order

### example:

rule discount(5.00)

### result:

All products have a sale price \$5.00 less than the regular price.

### example:

```
rule discount(0.90, *, p)
```

### result:

All products have a sale price that is 10% less (90%) of the regular price.

### List of parameters

(numeric value)	The amount to discount
*	Treat the numeric value as a decimal percentage
p	Calculate discount based on price (Default value)
S	Calculate discount based on existing sale price

## pos

rule pos(haystack, needle) as name

Return the numeric location (0-based) of needle within haystack.

### example:

rule pos(description, "cars") as index rule substr(description, \$index, 80) as reduced\_description setAttributeDefault my\_custom\_attribute as reduced\_description mapAttribute my\_custom\_attribute as g:title

### result:

Title is equal to description from the word "car" and the next 80 characters

# shipping

rule shipping(parameter1, parameter2)

Assign shipping values for this feed

Note: shipping() accepts parameters in any order

### example:

rule shipping(5.00)

### result:

Apply a flat rate of \$5.00 to all products

### example:

rule shipping(0.10, \*, p)

### result:

Shipping fees equal to ten percent of the regular price

### List of parameters

(numeric value)	The dollar amount
*	Treat the numeric value as a decimal percentage
p	Calculate shipping based on price (Default value)
S	Calculate shipping based on existing sale price

## strlen

rule strlen(parameter) as name

Return the string length of the given parameter

### example:

rule strlen(description) as length\_of\_description setAttributeDefault dlen as \$length\_of\_description mapAttribute dlen as length

### result:

The length of the description is mapped into the feed

## substr

rule substr(source, start, length) as name

Return a portion of a source starting from start and having length characters.

### example:

rule substr(description, 0, 100) as portion setAttributeDefault excerpt as \$portion mapAttribute excerpt as excerpt

### result:

The first 100 characters of description are mapped into a new field called excerpt.

## tax

```
rule tax(value)
```

Adds tax parameters to the feed.

### example:

rule tax(3.10)

### result:

\$3.10 tax is added to the product.

Note: Google Feeds, when they spot the tax rule, will build a structured tax tag and fill in the tax\_country attribute as US. You may override this if needed.

Note: Most other feeds will not automatically map tax or tax\_counry.