

Scope of this review

First things first: I'll try hard not to tell you things which you can watch & read about on the DMXZONE website, and this review is certainly not meant to provide a comprehensive introduction of each & every feature the current Feed Genie version (1.02) comes with.

My review is based on evaluating the "Picasa – User Album Photos" feed type, will highlight my (sometimes developer-centric) observations and occasionally provide some technical background information which may be of interest to those who'd like to look behind the curtains.

Enormous time-saving even for developers

Until recently I somehow managed to forego the whole "display feeds on your website" subject area. How comes? Although I'm adept at coding PHP stuff until it fits, integrating an RSS feed which grabs data from one of my Picasa Web Albums would have meant to...

- study the pretty complex Web Albums Data API and
- either develop a complex PHP class myself or make use of the official PHP client library provided by ZEND.

Well, all of the above is certainly doable, but there are times when even a guy like me prefers to keep the casual desire to just be bone-lazy unharmed by *not* having to wade through code – and the DMXZONE Feed Genie extension for Adobe Dreamweaver appears to be a perfect match, because it does the heavy lifting behind the scenes and allows you to integrate a variety of popular feeds without breaking into sweat.

Parameterized Feeds

The Feed Genie developers must have spent a huge amount of time studying the various feed Data APIs. As a result the Feed Genie server behavior allows for a very convenient way to define the most important Search (plus, depending on the Feed Type, even display related...) options which will later be passed to the respective Feed provider as parameter/value pairs embedded in a query string. As displayed next, the available parameters will of course vary across different Feed Types:

Feed Genie Options	Feed Genie Options
Name: <input type="text" value="YouTubeSearch"/>	Name: <input type="text" value="PicasaAlbum"/>
Feed Type: <input type="text" value="YouTube - Video Search"/> ▼ <input type="button" value="Test Feed"/>	Feed Type: <input type="text" value="Picasa - User Album Photos"/> ▼ <input type="button" value="Test Feed"/>
Search Options	Search Options
Search: <input type="text"/> ⚡*	User ID: <input type="text"/> ⚡*
Author: <input type="text"/> ⚡	Album ID: <input type="text"/> ⚡*
Time: <input type="text" value="All Time"/> ▼	Thumb Size: <input type="text" value="104 px, Cropped"/> ▼
Order By: <input type="text" value="Relevance"/> ▼	Image Size: <input type="text" value="200 px, Uncropped"/> ▼

As for me, this particular feature represents an indispensable highlight, because – well, even if you don't have the slightest clue about what parameter/value pairs the respective Feed provider expects in order to generate a valid feed cache file, it just works fine and doesn't allow for making too many mistakes. Guess what, it's even a perfect starting point for a feed noob like me ;-)

Using Dynamic Data as parameter values: benefits

While some Dreamweaver developers will settle for creating just one document and provide a single pair of Picasa User ID/Album ID credentials as static strings of digits, others might want to take the extra step and make the functionality provided by Feed Genie available to, say, all registered users of their custom Content

Management System. In this case it's needless to mention that this data need to be stored in a MySQL database table and derived from there by using a standard Dreamweaver recordset. Very fortunately the Feed Genie developers must have been anticipating such advanced use cases, because that's exactly what the lightning bolt icon on the right is meant for:

Search Options

User ID: ⚡*

Album ID: ⚡*

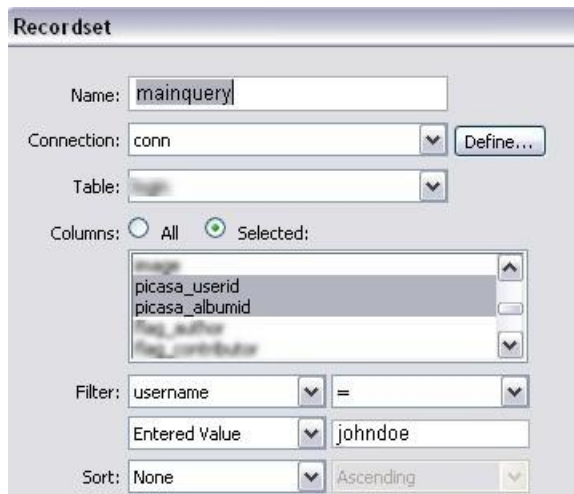
Using Dynamic Data as parameter values: a proof of concept

The basic requirements for realizing a very simple "per user" definition of a Picasa User ID respectively Album ID are: a functioning Dreamweaver Connection and a MySQL table with at least the following columns:

userid	type: int attribute: unsigned null: no extra: auto-increment	not needed for this tutorial, but such a unique column is a must have when identifying the logged in visitor via the Dreamweaver "MM_user_id" Session Variable
username	type: varchar	
picasa_userid	type: int or varchar, just make it long enough to hold the complete string of digits	
picasa_albumid	see above	

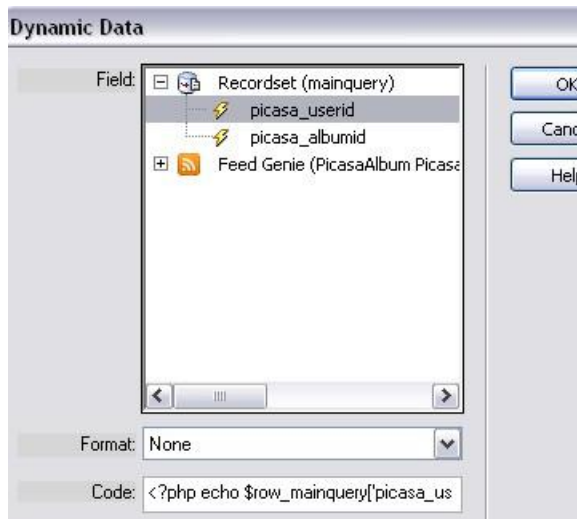
Assuming that a fictitious chap going by the username of "johndoe" has his Picasa User ID/Album ID values already stored by now, here's how to derive these two information units from the aforementioned table and pass them to the corresponding Feed Genie input fields:

1. create a Dreamweaver Recordset using the indicated Filter criteria:

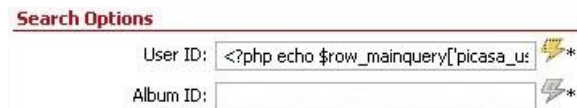


Confirm with **OK**.

2. open the Feed Genie server behavior, click the "User ID" lightning bolt icon and select the "picasa_userid" column from the Dynamic Data server behavior:



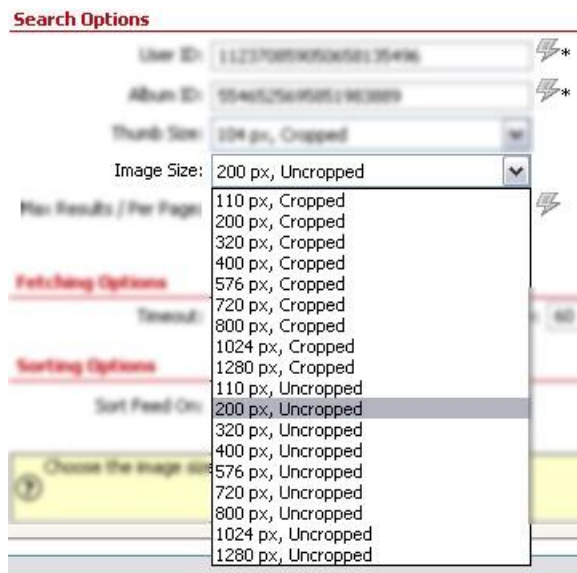
After confirming with **OK**, the updated "User ID" input field should now look like this:



3. repeat step 2 for the "Album ID" input field, but select the "picasa_albumid" column instead.

The available Image Size options vs. customizing the Feed's query string

While comparing the currently defined default values of some of Feed Genie's Search Options with the official [Picasa Web Albums query parameters reference](#), I noticed that – for instance – the list of "Image Size" options happens to leave out a few more values which are supported as well.



Here's what the Query Parameters Reference says:

"The following values are valid for the thumbsize and imgmax query parameters and are embeddable on a webpage...94, 110, 128, 200, 220, 288, 320, 400, 512, 576, 640, 720, 800, 912, 1024, 1152, 1280, 1440, 1600".

Does this mean I'm criticizing the Feed Genie developers? Not at all, because trying to cover each & every possible image size option within the bounds of such dropdown menus would clearly make the whole thing far too complex to a regular user who might prefer a concise list displaying the most used options.

However the more advanced user might look for a way to override the default Image Size options, and here's some good news for those who are not prone to panic attacks when having to switch Dreamweaver to Code View for amending little things: whatever value you define or select in the "Search Options" section of the Feed Genie server behavior, will become part of an easily editable Query String which gets inserted somewhere on top of the current document:

```
41 <?php
42 // *** Feed Genie PHP 1.0.0
43 $PicasaAlbum = new FeedGenie();
44 $PicasaAlbum->FeedType = "Picasa User Album Photos";
45 $PicasaAlbum->FeedUrl =
  "http://picasaweb.google.com/data/feed/api/user/ /albumid/ /?kind=photos&
  thumbsize=104c&imgmax=200u&max-results=20";
46 $PicasaAlbum->Timeout = 20;
47 $PicasaAlbum->CacheTime = 60;
48 $PicasaAlbum->FeedName = "PicasaAlbum";
49 $PicasaAlbum->ReadFeed();
50 $PicasaAlbum->SortOn("", FG_SORT_ASCENDING);
51 ?>
```

What would you have to do when changing one of Feed Genie's default "Image Size" values (say, 200px, Uncropped) to one of the currently unavailable values (say, 1440px, Cropped)?

Just locate the "imgmax" parameter, consider the difference between the flags "u" (uncropped) and "c" (cropped):

```
41 <?php
42 // *** Feed Genie PHP 1.0.0
43 $PicasaAlbum = new FeedGenie();
44 $PicasaAlbum->FeedType = "Picasa User Album Photos";
45 $PicasaAlbum->FeedUrl =
  "http://picasaweb.google.com/data/feed/api/user/ /?kind=photos&
  thumbsize=104c&imgmax=200u&max-results=20";
46 $PicasaAlbum->Timeout = 20;
47 $PicasaAlbum->CacheTime = 60;
48 $PicasaAlbum->FeedName = "PicasaAlbum";
49 $PicasaAlbum->ReadFeed();
50 $PicasaAlbum->SortOn("", FG_SORT_ASCENDING);
51 ?>
```

... and change the parameter value accordingly:

```
41 <?php
42 // *** Feed Genie PHP 1.0.0
43 $PicasaAlbum = new FeedGenie();
44 $PicasaAlbum->FeedType = "Picasa User Album Photos";
45 $PicasaAlbum->FeedUrl =
  "http://picasaweb.google.com/data/feed/api/user/ /?kind=photos&
  thumbsize=104c&imgmax=1440c&max-results=20";
46 $PicasaAlbum->Timeout = 20;
47 $PicasaAlbum->CacheTime = 60;
48 $PicasaAlbum->FeedName = "PicasaAlbum";
49 $PicasaAlbum->ReadFeed();
50 $PicasaAlbum->SortOn("", FG_SORT_ASCENDING);
51 ?>
```

There's just one minor downside which might confuse some: the dropdown menus of the Feed Genie server behavior are by nature limited to those values they know and will reject those they don't -- in other words: after manually changing a parameter value to something that's currently unsupported by Feed Genie, you'll note that later on the corresponding menu will trip over your custom value and display an arbitrary option as pre-selected.

PHP related observations

Whenever using PHP scripts from 3rd party vendors, the first thing I do is to scan them for a variety of criteria such as:

- a) which PHP versions are supported
- b) usage of features or functions which are deprecated in PHP 5.3
- c) possible issues which might require a certain php.ini configuration

1. Supported PHP versions

As I couldn't detect any usage of functions, classes and methods which are specific to PHP 5, and as any native PHP functions used by the Feed Genie libraries are compatible with PHP 4, the scripts should work equally fine with both PHP 4 and 5.

2. Usage of deprecated features/functions

The deprecated functions `ereg()` and `eregi()` are used by an auxiliary script, what will throw errors when used with PHP 5.3

3. Possible issues

To read data from a feed cache file stored in the server's file system, Feed Genie relies on the PHP function `fopen()`. Some web hosts might disable this function for security reasons, and it's obvious that Feed Genie will not work under this condition.