Bs_TextType:
create title images on the fly

Andrej Arn
Sam Blume

BlueShoes

Bs_TextType creates images with text that look better than just plain text. No need to have a designer create all the new title images just because the boss wants another text or another color.

Featurs:
- Define the font face, size, and text color
- Use any font you want and have on your server
- Antialiasing
- Transparency
- Background colors or background images
- Create multiline text
- Specify width/height or have it auto-calculated
- Define the posX posY (x-axis, y-axis)
- Have text aligned to the right
- Draw the text with an angle, for example 90 degrees
- Automatically store (cache) the images on the server
- Use profiles and/or pass params in the URL

What you need:
- An existing installation of PHP/Apache (or IIS) and ***BlueShoes***.
- GD library. Windows users might have to uncomment the line:
  `extension=php_gd.dll`
  in the php.ini file. I recommend using gd2.
- A website with the option to use .htaccess files. Not all webhosts allow that.
- Some knowledge about PHP, understanding of how to use existing classes. Some knowledge of how .htaccess and a 404 works wont' hurt.

The 404 redirect

In your webroot, create a new directory called “`textType`”. In that directory create a file called `.htaccess` with the content:

```
ErrorDocument 404 /textType/404.php
```

Now create a file named `404.php` in the new `textType` directory, like this:

```
<?php
phpinfo();
?>
```

So… for every request that points somewhere into the /textType directory to a file that does not exist will cause a hit to /textType/404.php, our script, and will show the phpinfo screen. Let’s try it, say your site is www.yoursite.com then call http://www.yoursite.com/textType/foo.png

⚠ If you don’t see the phpinfo screen, I bet that your apache conf file is set to not allow such htaccess settings. The sites config has an `AllowOverride setting inside the` <directory> tags:

```
# This controls which options the .htaccess files in directories
# can override. Can also be "All", or any combination of "Options",
# "FileInfo", "AuthConfig", and "Limit"

    AllowOverride None
```

“None” is no good, “All” works. If you don’t have access to this configuration, tell your hosting company to set it to “All” for you.

The 404 script

Now that this worked, we can go on and handle the 404 error. Replace the `phpinfo();` line with the following snippet:

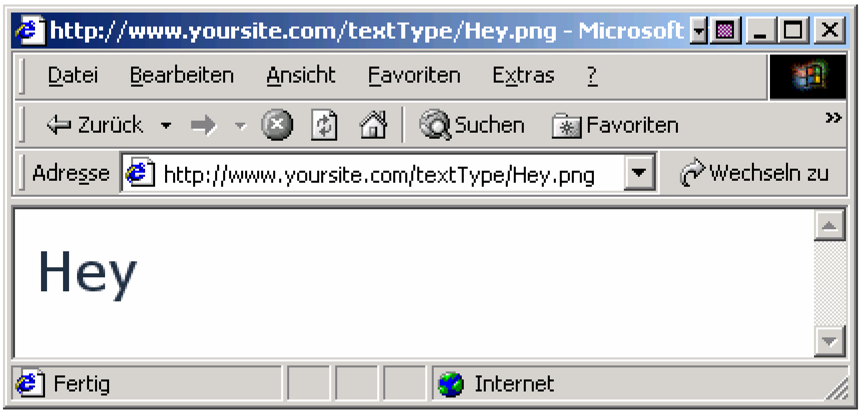
```
<?php
require_once($_SERVER['DOCUMENT_ROOT'] . '../global.conf.php');
require_once($APP['path']['core']      . 'gfx/text/Bs_TextType.class.php');

$defaults = array(
  'fontDir' => 'c:/winnt/fonts/',
  'store'   => FALSE,
);
$tt =& new Bs_TextType();
$tt->doItYourself($defaults);
?>
```

Customize the fontDir for your system.

The images

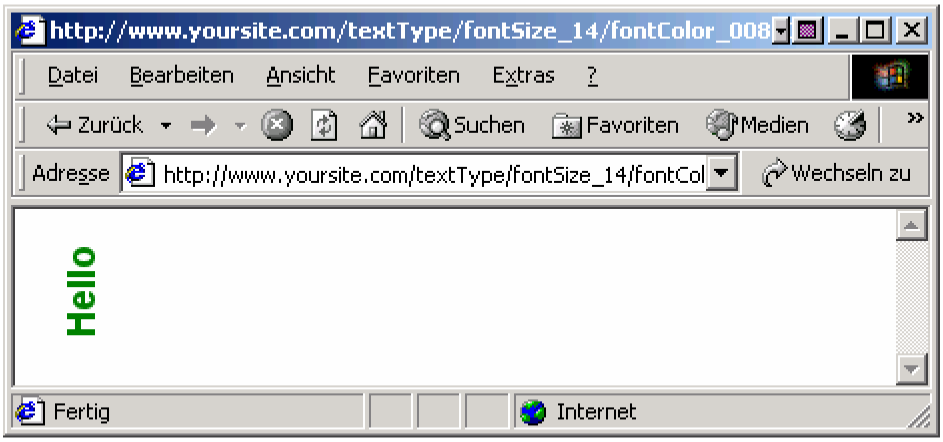
Now go ahead and type an url like: http://www.yoursite.com/textType/Hey.png



Adresse: http://www.yoursite.com/textType/Hey.png

Hey

We can also put settings directly into the url, like this:
/textType/fontSize_14/fontColor_008000/fontFace_arialbd/angle_90/posX_30/Hello_eWorld.png



Adresse: http://www.yoursite.com/textType/fontSize_14/fontCol

Hello

You write the property and value separated by an underscore, packed into a pseudo-directory. /propertyName_value/propertyName_value/… and at the end there is the filename.png, where filename will be the printed text.

List of the available properties:

fontSize	Font size, eg 11, default is 20.
fontFace	The font, eg arialbd, default is ‘VERDANA’. Don’t add the ‘.ttf’ ending.
imgWidth	The width of the image in pixels. Don’s specify or set to ‘auto’ to have it auto-calculated based on the text.
imgHeight	The height of the image in pixels. Don’s specify or set to ‘auto’ to have it auto-calculated based on the text.
fontColor	The foreground color, default is 213042 (near black, a little blue, good readable). Don’t use color names like “red” [yet]. Don’t use the # sign.
bgColor	The background color, default is FFFFFF (white). Don’t use color names like “red” [yet]. Don’t use the # sign.
fontDir	The directory where the fonts are. eg c:/winnt/fonts/
bgDir	The directory where the background images are, not yet supported.
bgImage	The background image to use. If you use one, and you specified a width and height that is larger than the width/height of this image, then the larger ones will be used. If your text is bigger, the final image will be the size of the text; bigger. An example follows later down.
store	If the image should be cached or not. 0 = false and 1 = true.
profileName	A profile to use that exists in the profile folder. An example follows later down.
posX	If you want to indent (or outdent) the starting point of the text on the x-axis. This is important when using an angle of 90 degrees for example.
posY	If you want to indent (or outdent) the starting point of the text on the y-axis.
xAling	Where to align the text. Default is ‘left’, can also be ‘right’.
yAlign	Not supported yet.
angle	To flip the text, for example 90, 180 or 270.

The settings can be defined on 4 levels:

1. At first use hard-coded class var defaults.
2. Overwrite them with defaults on the `doItYourself()` call in the 404.php script.
3. Overwrite them with the settings made in the profile, if one is used. (An example follows later down.)
4. Overwrite them with the settings made in the URL.

Profiles

If you want to create many similar title images, it makes sense to define the settings like font face, size, color ... in a profile file instead of the URL. This way, if you want to change something, you only have to change it once in the profile, and not in every URL.

Create a subdirectory for `textType` named `_profiles`. In that directory, create a new file and name it for example header1.ini. In this ini file you can define all the properties you want, for example:

```
fontDir    = c:/winnt/fonts/
fontSize   = 11
fontFace   = arialbd
```

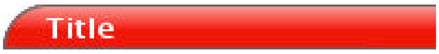
Then you can form your URL like this: /textType/profileName_header1/fontColor_800000/MyTitle.png

Background Images

bgImage is the property you want.

The same way we did it for the `_profiles` folder, create a `_backgrounds` folder as sibling.
Save a png image in that folder, something like this:

I called mine smRedRound.png, now i can call it like this:
/textType/bgImage_smRedRound.png/fontColor_FFFFFF/fontSize_9/fontFace_verdanab/posX_20/posY_15/Title.png

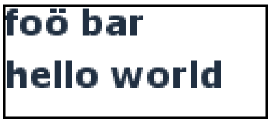


Title

Special Characters and Multiline Text

Escape special characters with an underscore (instead of a backslash) to make your URL valid.
Examples:

/textType/store_0/fontSize_12/fontFace_verdanab/fo_oe_ebar_nhello_eworld.png



foö bar
hello world

Character list:

_e	=	space (empty)
_n	=	new line (like \n)
_t	=	tab (like \t)
_ae	=	ä (german umlaut)
_oe	=	ö (german umlaut)
_ue	=	ü (german umlaut)

Creating Images by Code

There's also the code file: core/gfx/text/examples/createByCode.php

```
require_once($_SERVER['DOCUMENT_ROOT'] . '../global.conf.php');
require_once($APP['path']['core']      . 'gfx/text/Bs_TextType.class.php');

$tt =& new Bs_TextType();
$tt->set('fontSize', 30);
$tt->set('imgWidth', 500);
$tt->set('xAlign', 'right');
$tt->setText("Hello World");
$tt->create();
$tt->send();
$tt->destruct();
```

Save() stores the image in that directory so it can be fetched the next time. Send() writes the image with appropriate header lines to the client. Destruct() is the class destructor.

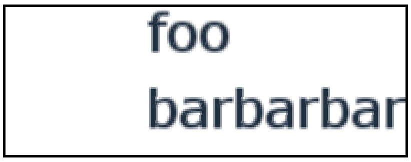
For a multiline text you can use an array, like this:

```
myText = array("line one", "line two", "line three");
$tt->setText($myText);
```

Known limits

A multiline text aligned to the right won't do what you might expect. This is how it looks:

/textType/imgWidth_200/xAlign_right/foo_nbarbarbar.png



foo
barbarbar

Caching

There's server side and client side caching.

Server side:
In the 404.php script we've set 'store' to FALSE. That was ok while playing around. I recommend you remove that line, so that the default value (which is TRUE) is used.
When you call another URL now to generate an image, there will be a new path under /textType/ going down to the image. The next user that hits this URL will receive the file from the servers' cache, it won't be re-generated.

And when the same user needs the same image again, depending on settings in the users browser, it will:

1. refetch the file from the server
2. ask the server if there is a newer version (if-modified-since request), if yes then refetch, if not then use the local cached one
3. use the local cached image

You can avoid most of the hits to the server by setting correct headers. Add something along the lines below into the new .htaccess file.

```
<IfModule mod_expires.c>
  ExpiresActive On
  ExpiresByType image/png  "access plus 12 hours"
  ExpiresByType image/gif  "access plus 12 hours"
  ExpiresByType image/jpeg "access plus 12 hours"
</IfModule>
```

You can change “12 hours” to “3 minutes” or so. Yahoo sets its images to expire in 10 years or something.
Additional reading: http://httpd.apache.org/docs/mod/mod_expires.html

You may need to activate mod_expires in your httpd.conf and restart apache.

Of course you can also use that for other things. A place to consider is your default image directory.

Make sure that your configuration worked. Use a http connection sniffer or proxy. Such a webtool can be found here: http://webtools.mozilla.org/web-sniffer/

Type in the url of the image, and select “verbose mode”.
The header output should look something like this (see the “Cache -Control:” and “Expires:” line):

```
HTTP/1.1 200 OKCRLF
Date: Sun, 22 Sep 2002 16:31:31 GMTCRLF
Server: Apache/1.3.26 (Win32) PHP/4.2.2CRLF
Cache-Control: max-age=43200CRLF
Expires: Mon, 23 Sep 2002 04:31:31 GMTCRLF
Last-Modified: Sun, 22 Sep 2002 02:27:17 GMTCRLF
ETag: "0-173-3d8d2a85"CRLF
Accept-Ranges: bytesCRLF
Content-Length: 371CRLF
Connection: closeCRLF
Content-Type: image/pngCRLF
```

Take care about this:

If you create a title image the old fashion way naming it “title.png” and putting the text “hello” on it, and then your boss wants it to read “welcome”, you have a problem. Browsers need to update. If they have cached it for 12 hours, users will see the old text for 12 hours. So you tell your boss you’ve changed the text, he goes and reloads the page, and then he’ll get angry with you. To avoid that, you can change your URL by adding another dummy param, like this:

/textType/**a_b**/imgWidth_200/foo.png

Png vs Gif

In the past the png used to be a problem because the browser support was bad. Even the png site (http://www.libpng.org/pub/png/) had gif’s on the page. But these days are gone. Gif is the past, png the future.

If you still want gif, you have these options:

1. create your files with your favorite graphic editor like photoshop
2. patch the gd library, see http://www.rime.com.au/gd/
   note: that may not be legal in country your server is located.