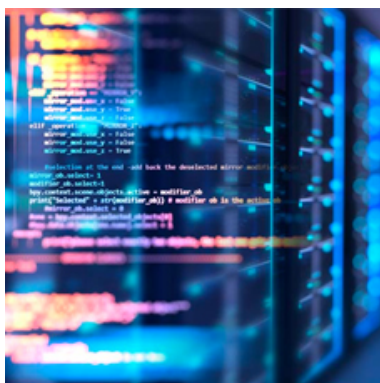


<https://www.patcatnats.fr/spip.php?article25>



Réduire toutes les images

- Inform@tic - SPIP -



Date de mise en ligne : mardi 7 novembre 2006

Copyright © PatCatNat's - Tous droits réservés

Contrib non testée.

La syntaxe est : `reduire_all_images{taille_max,direction}`

Par exemple, pour limiter la taille des images à 500 pixels de large on utilise :

```
<BOUCLE_art(ARTICLES){id_article}> [(#TEXTE|reduire_all_images{500,x})] </BOUCLE_art>
```

Avec ce filtre, une image de 450x600 ne serait pas redimensionnée, mais une image de 600x600 le serait.

Code à mettre dans le fichier *mes_fonctions.php3* :

```
<?php

/*Filter Name: Reduire_all_images
  Author: Renato Formato
  mail: renatoformato@virgilio.it
  Date: 6/6/2004
  License: GPL
  URL: www.spip-contrib.net?article.php3?id_article=550
*/

function reduire_all_images($texte, $taille = 120,$axes="both") {
    //a week in seconds
    $maint_delay = 3600*24*7;
    //retrieve last maintenance time
    $last_maint_date=@filemtime("IMG/last_maint.txt");
    //never done maintenance. Create last_maint file
    if (!$last_maint_date) {
        $handle=fopen("IMG/last_maint.txt","w");
        fclose($handle);
        $last_maint_date=@filemtime("IMG/last_maint.txt");
    }
    //create resized images
    //Process images with descriptions
    $texte=preg_replace("!(.*)?*src='[^>]*IMG/[^>]*>(?(1).*
)teisu","reduire_image_check('$0',$taille,$axes,$last_maint_date,$maint_delay)","$texte);
    //check last maintenance+delay 0) {
    if (substr($img_code,0,6)=="
]*src='[^>]*IMG/[^>]*>!ei","reduire_image_ex('$0',$taille,$axes,$last_maint_date,$maint_delay)","$img_code);
    //Change table cell width value
    if ($new_image_width!=0) {
        $img_code=preg_replace("!()!is","\\1width='$new_image_width'\\3",$img_code);
    }
    return $img_code;
} else return reduire_image_ex($img_code,$taille,$axes,$last_maint_date,$maint_delay);
}
```

```
}

function reduire_image_ex($img, $taille = 120,$axes="both",$last_maint_date,$maint_delay) {
    global $new_image_width;
    $new_image_width=0;
    if (strlen($img) > 0) {

        // recuperer le nom du fichier
        if (eregi("src=\"([^\"]+)\\"", $img, $regs)) $logo = $regs[1];
        if (eregi("align=\"([^\"]+)\\"", $img, $regs)) $align = $regs[1];
        if (eregi("name=\"([^\"]+)\\"", $img, $regs)) $name = $regs[1];
        if (eregi("hspace=\"([^\"]+)\\"", $img, $regs)) $espace = $regs[1];

        if (file_exists($logo)) {
            $logo = substr($logo, 4, strlen($logo));
            // recuperer nom de l'image et sa terminaison
            $path = dirname($logo);
            if ($path!="") $path=$path."/resized/";
            if (!is_dir("IMG/$path")) @mkdir("IMG/$path");
            $basename = basename($logo);
            $nom = substr($basename, 0, strpos($basename, "."));
            $format = substr($basename, strlen($basename)-3, strlen($basename));

            $axes=strtolower($axes);
            if ($axes!="both" && $axes!="x" && $axes!="y") $axes="both";
            // test de recalcul en fonction des dates des fichiers
            // pour verifier si mise a jour plus recente du logo
            $imagefile="IMG/$path$taille-$axes-$nom.$format";
            if (file_exists($imagefile)) {
                $imagetime=filemtime($imagefile);
                if ($imagetime > filemtime("IMG/$logo")) {
                    //check filetime and touch if filetime>last maiteinance time + half delay
                    if ($imagetime<($last_maint_date+$maint_delay/2) and time()>($last_maint_date+$maint_delay/2))
touch($imagefile);
                    //store image width
                    list($new_image_width)=getimagesize($imagefile);
                    return "['']";
                    $recalculer = false;
                }
            }
            else {
                $recalculer = true;
            }
        } else {
            $recalculer = true;
        }
    }

    $gd_formats = lire_meta("gd_formats");
    if ($recalculer AND ereg($format, $gd_formats)) {
        // Recuperer l'image d'origine
        if ($format == "jpg") {
            $srcImage = ImageCreateFromJPEG("IMG/$logo");
        }
    }
}
```

```
else if ($format == "gif"){
    $srcImage = ImageCreateFromGIF("IMG/$logo");
}
else if ($format == "png"){
    $srcImage = ImageCreateFromPNG("IMG/$logo");
}
if (!$srcImage) return;

// Calculer le ratio
$srcWidth = ImageSX($srcImage);
$srcHeight = ImageSY($srcImage);

if (($srcWidth > $taille && ($axes=="both" OR $axes=="x")) OR ($srcHeight > $taille && ($axes=="both" OR $axes=="y"))) {
    $ratioWidth = $srcWidth/$taille;
    $ratioHeight = $srcHeight/$taille;

    //Ridimensiono da altezza (solo se axes!=x)
    if ($ratioWidth < $ratioHeight && $axes!="x") {
        $destWidth = floor($srcWidth/$ratioHeight);
        $destHeight = $taille;
    }
    else {
        $destWidth = $taille;
        $destHeight = floor($srcHeight/$ratioWidth);
    }
    $new_image_width=$destWidth;
} else {
    return $img;
}

// Initialisation de l'image destination
if ($GLOBALS['flag_ImageCreateTrueColor'] AND $destFormat != "gif")
    $destImage = ImageCreateTrueColor($destWidth, $destHeight);
if (!$destImage)
    $destImage = ImageCreate($destWidth, $destHeight);
// Recopie de l'image d'origine avec adaptation de la taille
$ok = false;
if ($GLOBALS['flag_ImageCopyResampled'])
    $ok = ImageCopyResampled($destImage, $srcImage, 0, 0, 0, 0, $destWidth, $destHeight, $srcWidth, $srcHeight);
if (!$ok)
    $ok = ImageCopyResized($destImage, $srcImage, 0, 0, 0, 0, $destWidth, $destHeight, $srcWidth, $srcHeight);

// Sauvegarde de l'image destination
$destination = $imagefile;
if ($format == "jpg") {
    ImageJPEG($destImage, $destination, 70);
}
else if ($format == "gif") {
    ImageGIF($destImage, $destination);
}
```

```
}
else if ($format == "png") {
    ImagePNG($destImage, $destination);
}
ImageDestroy($srcImage);
ImageDestroy($destImage);

return "['']";
} else {
    $taille_origine = @getimagesize("IMG/$logo");
    if ($taille_origine) {
        // Calculer le ratio
        $srcWidth = $taille_origine[0];
        $srcHeight = $taille_origine[1];

        if (($srcWidth > $taille && ($axes=="both" OR $axes=="x")) OR ($srcHeight > $taille && ($axes=="both"
OR $axes=="y"))) {
            $ratioWidth = $srcWidth/$taille;
            $ratioHeight = $srcHeight/$taille;

            if ($ratioWidth < $ratioHeight && $axes!="x") {
                $destWidth = floor($srcWidth/$ratioHeight);
                $destHeight = $taille;
            }
            else {
                $destWidth = $taille;
                $destHeight = floor($srcHeight/$ratioWidth);
            }
            $new_image_width=$destWidth;
        } else {
            $destWidth = $srcWidth;
            $destHeight = $srcHeight;
        }
        return "['']";
    }
}
}
}
}
```

?>