

Let's start with a list of the new appconfigs:

UseImageResize	DefaultWidth_icon
CategoryImg_icon	DefaultHeight_icon
CategoryImg_medium	DefaultWidth_medium
CategoryImg_large	DefaultHeight_medium
DistributorImg_icon	DefaultWidth_large
DistributorImg_medium	DefaultHeight_large
DistributorImg_large	DefaultQuality
ManufacturerImg_icon	DefaultStretch
ManufacturerImg_medium	DefaultCrop
ManufacturerImg_large	DefaultCropHorizontal
ProductImg_icon	DefaultCropVertical
ProductImg_medium	DefaultFillColor
ProductImg_large	LargeCreatesOthers
ProductImg_swatch	LargeOverwritesOthers
SectionImg_icon	MultiColorMakesSwatchAndMap
SectionImg_medium	SwatchStyleAuto
SectionImg_large	MultiMakesMicros
VariantImg_icon	MicroStyle
VariantImg_medium	UseImagesForMultiNav
VariantImg_large	UseRolloverForMultiNav

The default appconfigs

- ❖ DefaultWidth_icon – default icon width (i.e. 150)
- ❖ DefaultHeight_icon – default icon height (i.e. 150)
- ❖ DefaultWidth_medium – default medium width (i.e. 250)
- ❖ DefaultHeight_medium – default medium height (i.e. 250)
- ❖ DefaultWidth_large – default large width (i.e. 500)
- ❖ DefaultHeight_large – default large height (i.e.500)
- ❖ DefaultQuality – default quality (i.e. 100)
- ❖ DefaultStretch – default stretch (i.e. true)
- ❖ DefaultCrop – default crop (i.e. true)
- ❖ DefaultCropHorizontal – default crop horizontal (i.e. center)
- ❖ DefaultCropVertical – default crop vertical (i.e. middle)
- ❖ DefaultFillColor – default fill color (i.e. #FFFFFF)


These are the defaults for the entire resizing functionality. If one of these is not specifically set for any of the other appconfigs, then these are the values that are used. These should never be left blank...but should be changed if some or most of the appconfigs will use the same values that you specify here. That way you don't have to add them over and over again.


The sizing appconfigs (icon and medium)

- ❖ CategoryImg_icon
- ❖ DistributorImg_icon
- ❖ ManufacturerImg_icon
- ❖ ProductImg_icon
- ❖ SectionImg_icon
- ❖ VariantImg_icon
- ❖ CategoryImg_medium
- ❖ DistributorImg_medium
- ❖ ManufacturerImg_medium
- ❖ ProductImg_medium
- ❖ SectionImg_medium
- ❖ VariantImg_medium

All of the sizing ‘_icon’ and ‘_medium’ appconfigs can control all, none, or some of the following properties of the icon and medium images:

- width – a number representing the width (width:250;)
- height – a number representing the height (height: 250;)
- quality – a number from 1 to 100 representing the percentage of the original quality (quality:80;)
- stretch – determines whether an image will stretch to fill the resized width and height specified (stretch:true;)

	<p>This is an icon image set with resize width and height set to 250, but with stretch set to ‘false’. The original image uploaded was 16 x 16. Note that the image is still 250 x 250 but the original picture stays the same size.</p>
---	--

	<p>This is the same 16 x 16 image uploaded as an icon 250 x 250 image but with stretch set to 'true'. The original image "stretches" to fill the 250 x 250 space.</p>
---	---

- `resize` – true or false...this value will override whatever you have in the appconfig **UseImageResize**. If you have one specific image that you don't want resizing add this to its appconfig to disable resizing for that specific image (`resize:false;`)
- `crop` – true or false...this value will override whatever you have in the appconfig **DefaultCrop** (`crop:false;`)
- `cropv` – top, bottom, or middle...this will override whatever value is in the appconfig **DefaultCropVertical** (`cropv:bottom;`)
- `croph` – left, right, or center...this will override whatever value is in the appconfig **DefaultCropHorizontal** (`croph:left;`)
- `fill` – any 6 digit hex color value with a preceding '#' ...this will override the **DefaultFill** appconfig (`fill:#00FF00;`)

Example appconfig values:

CategoryImg_icon – `width:150;height:150;crop:false;quality:100;`

CategoryImg_medium – `crop:true;width:250;croph:left;height:250;quality:80;`

ProductImg_icon – `resize:false;`

ProductImg_medium – `quality:100;height:250;stretch:true;width:250;`

The sizing appconfigs (large)

- ❖ CategoryImg_large
- ❖ DistributorImg_large
- ❖ ManufacturerImg_large
- ❖ ProductImg_large
- ❖ SectionImg_large
- ❖ VariantImg_large
- ❖ ProductImg_swatch

The ‘_large’ appconfigs are exactly the same as the ‘_icon’ and ‘_medium’ with two exceptions...they can take two additional parameters related to the two new appconfigs

LargeCreatesOthers and **LargeOverwritesOthers** (more on those in a minute).

- largecreates – true or false...will override whatever value is in appconfig **LargeCreatesOthers** and will allow icon and medium images to be created from the uploaded large image (largecreates:true;)
- largeoverwrites – true or false...will override whatever value is in appconfig **LargeOverwritesOthers** and decides whether icon and medium images will be created and overwritten if they already exist when **LargeCreatesOthers** is ‘true’ or when the attribute ‘largecreates:true’ is specified for the ‘_large’ appconfig (largeoverwrites:true;)

Example appconfig values:

CategoryImg_large – width:500;height:100;largecreates:true;largeoverwrites:false;

ProductImg_large – crop:true;croph:center;quality:100;width:500;height:100;

Manufacturer_large – resize:false;

AutoOthers appconfigs

- ❖ LargeCreatesOthers
- ❖ LargeOverwritesOthers


These two appconfigs control the creating of icon and medium images when a large image is uploaded. If **LargeCreatesOthers** is ‘true’ then icon and medium images will be created according to the respective ‘_icon’ and ‘_medium’ appconfigs. If **LargeOverwritesOthers** is ‘true’ or the attribute largeoverwrites:true; is specified then even if an icon or medium image already exist, they will be over-written with the icon and medium images that are created by uploading the large image. If this is ‘true’ and you are uploading in the multi image manager, the medium multi images and the icon multi images will also be created. If **LargeOverwritesOthers** is ‘false’ or the attribute largeoverwrites:false; is specified then medium and icon images will only be created if a medium and icon image do not already exist.

Additional Resizing appconfigs

- ❖ MultiColorMakesSwatchAndMap
- ❖ SwatchStyleAuto
- ❖ MultiMakesMicros
- ❖ MicroStyle
- ❖ UseImagesForMultiNav
- ❖ UseRolloverForMultiNav

These additional appconfigs will add some time-saving functionality and some user-friendly features to the store.

- **MultiColorMakesSwatchAndMap** – true or false...if set to ‘true’ and a variant has multiple colors, then when uploading images in the medium multi-image manager a swatch will be automatically created from the images uploaded into the first column based on the style width and height specified in appconfig **SwatchStyleAuto**. (i.e. if there are 5 colors specified and 5 images are uploaded in the first column, those 5 images will be resized to the width and height specified (width:25;height:25;) in **SwatchStyleAuto** and they will be combined into one image that is 125 pixels wide (5 * 25) and 25 pixels high. The html map code will also be generated for you).
- **SwatchStyleAuto** – takes width and height parameters (width:25; height:25;) that define the sizes of each small swatch image before they are combined into a single image.
- **MultiMakesMicros** – true or false...if set to ‘true’ the micro images are created for images uploaded into the first row (no-color) or the medium multi-image manager. These micros are created according the width and height specified in appconfig **MicroStyle** (width:40;height:40;) and are stored in the newly created directory ‘images/product/micros’. These micro images can then be used instead of the number icons when viewing a product through use of the **UseImagesForMultiNav** appconfig. Be careful though...even if **MultiMakesMicros** is set to false, deleting a medium multi-image will result in the micro image related to it being deleted as well.
- **MicroStyle** – width and height parameters to use for creating the micro images if **MultiMakesMicros** is set to ‘true’. **MicroStyle** can also have the cols, colspacing, and rowspacing attributes that specify the number of micros in a row, the spacing between each image in the row, and spacing between each row, (width:40; height:40;cols:4;colspacing:10;rowspacing:10;)

	<p>This is a screen shot of a product page with 10 images set in the medium multi-image manager, MultiMakesMicros set to 'true', and MicroStyle set to width:40;height:40;cols:5;colspacing:15;rowspacing:15;</p>
---	---

- **UseImagesForMultiNav** – true or false...if set to true then micro images will replace the number icons on the page for products that have multiple images. If a micro image does not exist then the 'no picture' image will be shown sized on the fly to match the width and height specified for the micros in **MicroStyle**.
- **UseRolloverForMultiNav** – true or false...if set to 'true' then on a product page with multiple images, the images will change on rollover rather than on click. If set to false then they will function normally, only changing when they are clicked.