# Guide on how to use the sprites

Always keep the PSD’s in the same folder as the png’s.

Using less sprites is better but they all have their uses. Most sites can get away with only the tall sprite.

## The tall sprite

* This is the sprite.png/sprite@2x.png files
* This will be your most commonly used sprite

### Positives

* This sprite gives you the ability to use the “right” and “center” horizontal background positions (“right” is particularly useful).
* This sprite is manual so there wont ever be any issues with the sprite if SASS has to be abandoned
* It allows for easy fine tuning
* It’s much smaller than the wide sprite for the same amount of images

### Negatives

* It’s manual…
  + so you have to worry about co-ordinates and positioning when placing images inside the sprite.
  + co-ordinates need to be halved before applying them to the SASS files.
* If the element it’s going in is too tall, the next sprite down will begin to show up.
* Can’t use vertical “bottom” and “center” background positions

### Rules for using this sprite

* Never ever change the width of this sprite this sprite, 100px (shrinks to 50px) should be more than enough for most cases.
* Never place images horizontally inline (side by side) in this sprite.
* Never extend the top of this sprite, only ever extend the bottom of the sprite.
* Separate each image with plenty of space (use guides to help you space the images out, 100px per guide is normally enough)
* If an image needs to be right aligned but not hard up against the right edge of the element, add the extra space to the image in the psd by moving the image to the left by the amount of space you want. So, for 10px of extra space on the right, move the image to the left 10px in the PSD.

### Best uses for this sprite

* Inline text icons (like the pdf or word icons)
* Right aligned non-css arrows/icons on links
* Custom radio and checkbox state images for stylised forms
* Times when a background image needs to be horizontally centre aligned in an element that has a variable width
* Alternative to using the auto-sprite for image replacement images (see auto sprite)

### How to set up to use this sprite

1. Open the sprite in phototshop
2. Make a change to it so you can save it (maybe just create a new layer then delete it, but don’t save it yet!)
3. Open the actions menu (provide screenshot)
4. Create a new action and give it a hot key shortcut (screenshot)
5. Open the ‘save for web and devices’ dialog (ctrl + alt +shift + s) (screenshot)
6. Save the file as the @2x image type
7. Open the ‘save for web and devices’ dialog again (ctrl + alt +shift + s)
8. Set image size to 50% and “Quality” to Bicubic Sharper   
   (according to photoshop, Bicubic Sharper is in general the best at preserving quality while shrinking an image)
9. Save the file as the regular image type (no @2x)
10. Save the photoshop document (ctrl + s)
11. Stop the action from recording any more (screenshot)
12. Now, whenever you make a change to the sprite, just hit that hot key you set the action to earlier and it will instantly create both a retina and non-retina version of the sprite as well as saving the Photoshop document ☺

**Extra notes on tall sprite set up**

* When referencing an image in the sprite, remember that you are looking at the @2x version of the file. All co-ordinates need to be halved before putting them in the css (so an image 300px from the top and 0px to the left would have the co-ordinates “0 -150px”).
* I would not recommend adding a halving function to the SASS mixin. If you did, whenever you make adjustments to a sprites position in the browser tools, you would need to double the value before putting it in your SASS file.
* Doing math once for a rough positioning then putting the exact positioning straight into the SASS file is better than putting the rough positioning straight into the SASS then having to do math for all fine tuning.

## The wide sprite

* This is the wide-sprite.png/wide-sprite@2x.png files

### Positives

* This sprite gives you the ability to use the “bottom” and “center” vertical background positions.
* Gives the ability to give elements background images that have a variable height without the fear of having any of the other sprites showing up when they are not wanted.
* It’s manual so no issues if SASS has to be abandoned

### Negatives

* You don’t want to put too many images into this sprite because of how spread out they need to be.
* For the same number of images, this is the largest of the three Sprites.
* Same negatives as the tall sprite in relation to being a manual sprite.
* If the images aren’t spread out far enough, unwanted sprite images will also show up.
* Can’t use horizontal “right” and “center” background positions

### Rules for using this sprite

* Never ever change the height of this sprite this sprite.
* Never extend the left side of this sprite, only ever extend the right side of the sprite.
* Never place images on top of one another in this sprite
* Give every image a huge amount of space between them. Remember you’re looking at the @2x sized sprite when looking at the PSD so you need double the max-width of the element it’s going in for no other unwanted sprites to show up. (note where I put my guides in the PSD)

### Best uses for this sprite

* Vertically centre aligned and bottom aligned background images
* Custom dot points (this is it’s primary purpose)

### How to set up to use this sprite

* Pretty much the same set up as the tall sprite.
* Just create a separate action in the exact same way as you did for the tall sprite except this time point to the “wide-sprite” png files.

## The auto sprite

* These are the auto-generated sprites that SASS is able to build for us.

### Positives

* This gives easy access by name to your individual sprite images
* You don’t need to worry about positioning sprites in a PSD file or finding the exact co-ordinates of images when referencing them in CSS.
* It’s a little easier to make both vertically and horizontally aligned images in elements that have a set height and width with auto-sprites than it is with manual sprites.

### Negatives (or at least issues I have with auto sprites)

* This sprite gives you no control over an images position in the css… although you can tweak the positioning by changing the cropping of the source images.
* I feel like the time it takes to perfectly crop an image for this is about the same amount of time it takes to add the image to the tall sprite and find its co-ordinates, maybe even longer if you want to alter the positioning a bit.
* It can drastically slow down the time it takes for your SASS compiler to compile the SASS when saving.
* Anything that you can do with this sprite can be done with the tall sprite, however the tall sprite gives you more direct control.
* Auto sprites can be a pain to get working. If you’re new to the project and can’t get it working, it completely stops you from doing your work since you can’t do anything with the SASS files with your compiler constantly throwing errors.
* I’ve had an auto-sprite bug out on me and put images in the wrong positions which lead to almost having to abandon the SASS files.
* I’ve had the retina versions of the auto-sprite images be out of alignment
* Almost every project I’ve had to abandon the SASS files for has been because of the auto-sprite not working or stopping someone from compiling.
* If SASS has to be abandoned, auto sprite images are more difficult to alter without causing more issues to the site than manual sprites.
* For me, it’s just not worth using auto-sprites any more.

### Rules for using this sprite

* Never ever manually set a background position or background size on this sprite.
* It needs compass to work and a fairly up to date version of SASS compiler.

### Best uses for this sprite

* Image replacement images, the auto sprite is excellent for this ☺   
  (Image replacement images are elements that only show an image but has visually hidden text that only users with screen readers will notice is there).
* A set of icon-only buttons with a set height and width.
* If the sprite isn’t being compacted (fitted together to make the file as small as possible… don’t compact the auto sprite, it can cause painful bugs and will ruin your editibility if SASS ever has to be abandoned) then you can also use this for left aligned image icons, but the ability to alter the exact position of the image in the css may makes the tall sprite more preferable for this.

### How to set up to use this sprite

I haven’t managed to get this working yet but this is how it *should* end up working:

1. Place the regular version of the image (as cropped as possible) in the auto-sprite folder
2. Place the @2x version of the same image in the auto-sprite-2x folder
3. Reference the file name (not including the file extension) in the auto-sprite mixin, so for a file called example.png do this:  
   @include M-retina(example);
4. Save your SASS file and it will generate a new sprite including your new image and have the correct positioning for the image in the outputted css (assuming you want the image in the very top left corner of the element)