



```
_LIT(KYourMifFile, ?resource\\apps\\yourmif.mif?);
```

//yourmif.mif contains svgs

```
include <yourmif.mbg> in .cpp file
```

// In header file

```
CFbsBitmap* iFocusBitmap;
CFbsBitmap* iBitmapMask;
```

// anywhere in .cpp file

```
if(iFocusBitmap == NULL)
{
    // load the image and mask
    AknIconUtils::CreateIconL( iFocusBitmap,
                               iBitmapMask,
                               KYourMifFile,
                               EMbmYourMif_focus,
                               EMbmYourMif_mask );
}
TSize size(Rect().Width()*10/100,Rect().Height()*10/100);
AknIconUtils::SetSize(iFocusBitmap, size, EAspectRatioNotPreserved);
AknIconUtils::SetSize(iBitmapMask, size, EAspectRatioNotPreserved );
```

// In the destructor

```
delete iFocusBitmap;
delete iBitmapMask;
```

AknIconUtils::SetSize initializes the icon to the given size, if the parameter aBitmap is an instance of CAknBitmap, i.e.created with AknIconUtils methods. If it is not CAknBitmap, this method does nothing. Note that this call sets the sizes of both bitmap and mask (if it exists), regardless of which is given as the parameter. Note Without a call to SetSize Bitmap will not be drawn and seen on the screen

Although **CreateIconL** have overloaded method for creating only the single bitmap rather than its mask but this has been more efficient interms of size and loading