NSImage

Category Name: MiscExtensions
Declared In: NSImage_MiscExtensions.h

Category Description

This category adds methods which help loading images. Remember that in order to find an image faster you can ,or even should, assign it a name via **setName:**.

NOTE: The regular **imageNamed:** has a bug since loading a name named "MyImage.tiff" assigns it the name "MyImage"...but on the next request it will not try to find the "MyImage" image. This cases you code to leak all the images since tehy are still included into the internal table.

It remains to be seen if Apple fixes **imageNamed:**. In the mean time you can use our methods since they already include a workaround for that problem.

Method Types

Bundle aware image loading	+ imageNamed:inContext: + imageNamed:inContext:searchAllBundles:

Class Methods

imageNamed:inContext:
+ (id)**imageNamed:**(NSString *)*name* **inContext:**(id)*object*

Just calls **imageNamed:inContext:searchAllBundles:** with a search applied to all bundles and frameworks.

imageNamed:inContext:searchAllBundles:
+ (id)**imageNamed:**(NSString *)*name* **inContext:**(id)*object* **searchAllBundles:**(BOOL)*flag*

We first ask **imageNamed:** if it already knows about an image with the specified *name*. If there is no image the search continues in the context of *object*. In the case where *object* is a normal instance we frist check the bundle where *object* came from. In the case of *object* being an instance of NSBundle we first check that bundle. Only pass in the *name* of an image following the same rules as for the **imageNamed:** method. Never use a complete path since the purpose is to find a path for an image.

HACK: The following is not yet implemented since we need some more code for the NSBundle class.

If that still doesn't help we will go out and try to check with all bundles and all frameworks which have been registered up to now.