SCED   by Klaus Hofrichter,

May 1993, Version 1.2a

Welcome to Sced, a little application which is useful for people who are sometimes using DOS disks to exchange files. You may have a problem with the file-suffix, because DOS only allows three characters. The Workspace Manager uses this extension for mapping files and applications, so you can't doubleklick on a file like "foo.fm" to open FrameMaker (Frame expects foo.frame). Other folks may find Sced useful, at least

for the tasteful application icon or for the chance to define your own document icons for some existing application which comes without sourcecode (like FrameMaker).

Sced takes the filename (e.g. foo.fm) from the WM after doubleklick on a document icon and scans an internal map. If there is an entry, Sced will redirect the "openFile:ok" message from the Workspace Manager to the given application. Sced does not take every file, only those you have added to the code. With other words, you should have a bit knowledge about compiling and editing code to prepare Sced for your site.

Since Version 1.2 you can advise Sced to create a soft link with the appropriate file extension and ask the WM to find the application for this file. The link will be placed in the tmp folder. This feature was suggested by Bruce Gingery together with other ideas.

The Version 1.2a is a minor bugfix to 1.2. Instead of openFile:ok we use openTempFile:ok for the new created link. This suggests to delete the link after use by the target application. In addition, the error-messages are displayed by panels (instead of a secret line in the console window).

There are two things to edit. One is called Sced.iconheader, the other one is ScedApp.m (a subclass of Application [and the only custom class coming with Sced. I like Objective-C code with only one custom class...]).

Open PB.project and click Attributes to edit Sced.iconheader: At the bottom of main window you see some Icons and filename extensions. Here you can map custom document-icons (like ScedDoc.tiff, which comes with Sced) and extensions. Drag the Document icons you want to see and edit the "???".

The second file to edit is ScedApp.m. Open ScedApp.m with Edit.app. You see some lines like:

```
struct tab {
        char *suffix;
        char *portname;
        char *properSuffix;
} map[] = {
        { "fm" , "FrameMaker" , "frame" },
        { "EPS" , NULL , "eps" },
```

```
        { NULL, NULL, NULL}
};
```

Sced uses this static table to map the file extensions coming from the WM and the "real" target-application. The first entry of one line is the suffix, the second is the portname of the target application, e.g. fm and FrameMaker. If you enter NULL, the file will be opened by the WM. The third entry identifies the expected extension. If you specify something here, you allow Sced to create a link from /tmp to the document file with the proper extension. This link will be forwarded to WM or the application itself, in this case

openTempFile:ok will be used. If you specify NULL no link will be created and the original path will be used. It is not allowed to specify NULL for both the portname and the proper extension. This will definitly cause a deadlock. Take care and be consistent with Sced.iconheader.

Type "make install" in a shell and Sced will be installed in your ~/Apps folder. Logout/Login, reboot or type "cmd-u" in the Workspace and Sced will do it's job (hopefully).

You can test Sced by renaming a EncapsulatedPostscript-file called foo.eps to foo.EPS. The Document icon should change to something with the words "detour". Doubleclick and Sced comes up, hides itself and opens your local eps viewer for you (if you have one).

However, there are some restrictions. The target app must accept files with a "wrong" extension. Preview does not, so you can't use Preview for *.EPS files. In this case you are forced to use the proper extension field in the table.

Have fun, read the Info and good luck!

Klaus Hofrichter
kho@fokus.gmd.de

PS: There is no copyright. Use the code freely. If you have improvements please drop me a note. I like NeXTmail.