

File Reference Resources

File reference ('FREF') resources perform two main functions. First, they associate icons you define with file types used by your application. Second, they allow users to drag document icons to your application icon in order to open them from your application.

Create an 'FREF' resource for your application file itself and create separate 'FREF' resources for each file type that your application can open. The following code shows, in Rez input format, the file reference resources for an application file, text documents, stationery pads, and editions, and for **TeachText** read-only documents.

Using file reference resources

```
resource 'FREF' (208, purgeable) { /* application */
    'APPL', 0, /* type 'APPL', maps to 'ICN#' w/ local ID 0 in 'BNDL' */
    "" /* leave empty string for name: not implemented */
};

resource 'FREF' (209, purgeable) { /* document */
    'TEXT', 1, /* type 'TEXT', maps to 'ICN#' w/ local ID 1 in 'BNDL' */
    ""
};

resource 'FREF' (210, purgeable) { /* stationery pad */
    'TEXT', 2, /* type 'TEXT', maps to 'ICN#' w/ local ID 2 in 'BNDL' */
    ""
};

resource 'FREF' (211, purgeable) { /* edition */
    'edtt', 3, /* type 'edtt', maps to 'ICN#' w/ local ID 3 in 'BNDL' */
    ""
};

resource 'FREF' (212, purgeable) { /* TeachText read-only files */
    'ttro', 4, /* These documents have TeachText as their creator. */
    "" /* Finder uses TeachText's 'ICN#' for these documents. */
    /* included here so users can drag these docs to */
    /* SurfWriter's app icon */
};
```

Each 'FREF' resource specifies a file type and a local ID. The file type can be defined for files created by your application only, for files created by other applications that your application supports, or for files of the existing general types, such as 'TEXT' or 'PICT'.

As described in **The Bundle Resource**, the local ID maps the file type to an 'ICN#' resource that is assigned the same local ID in the bundle. If you wanted two file types to share the same icon, for example, you could create two separate 'FREF' resources that share the same local ID, which the bundle would map to the same 'ICN#' resource. (Creating two file types that share the same icon is not recommended, however, because a shared icon would make it very difficult for the user to distinguish between the different

file types on the desktop.)

If you provide your own icon for the stationery pads that users create from your application's documents, create an 'FREF' resource for your stationery pads. Assign this 'FREF' resource a file type in the following manner: use the file type of the document upon which the stationery pad is based, but replace the first letter of the original document's file type with a lowercase s. As with other 'FREF' resources, you map this to an 'ICN#' resource in the bundle. (This convention necessitates that you make the names of your documents' file types unique in their last three letters.) For example, in this code, the 'TEXT' file type assigned within the 'FREF' resource is used for stationery pads created from documents of the 'TEXT' file type. In this case, when the `isStationery` bit (see **Finder Information in the Volume Catalog**) is set on a document of file type 'TEXT', the **Finder** looks in the application's 'BNDL' resource to determine what icon is mapped to documents of type 'TEXT'. The **Finder** then displays the document using the stationery pad icon shown in the figure above.

When the user drags a document icon to your application icon, the **Finder** checks a list of your 'FREF' resources. If the document's file type appears in the 'FREF' resource list, the **Finder** launches your application with a request to open that document.

If your application supports file types for which it does not provide icons, you can still define 'FREF' resources for them, and then users can launch your application by dragging these document icons to your application icon. For example, the 'FREF' resource with resource ID 212 in this program is created so that the **Finder** launches the application when users drag **TeachText** read-only documents to the application icon. Since these documents have **TeachText** as their creator, the **Finder** displays the icon that the **TeachText** application defines for them in its own bundle.

If your application supports the Open Documents event, you can also specify disks, folders, and a wildcard file type for all possible files in your 'FREF' resources so that users can launch your application by dragging their icons to your application icon. As explained in the **Apple Event Manager** text, the Open Documents event is one of the four required Apple events. After the **Finder** uses the **Process Manager** to launch an application that supports high-level events, the **Finder** sends the application an Open Documents event, which includes a list of alias records for desktop objects that the application should open.

Since alias records can specify volumes and directories as well as files, an Open Documents event gives you the opportunity to handle cases where users drag disk or folder icons to your application. Create an 'FREF' resource and specify 'disk' as the file type to allow users to drag hard disk and floppy disk icons to your application icon. Create an 'FREF' resource and specify 'fold' to allow users to drag folder icons to your application icon. Create an 'FREF' resource that specifies '*****' as the file type to allow users to drag all file types-including applications, system extensions, documents, and so on, but not including disks or folders-to your application icon. If you create three 'FREF' resources that specify 'disk', 'fold', and '*****' as their file types and if your application supports the Open Documents event, you effectively allow users to launch your application by dragging any desktop icon to your application icon. It is up to your application to open disks, folders, or all possible file types in a manner appropriate to the needs of the user.