Sons of Thunder Software
SOFTWARE DEVELOPMENT AND CONSULTING SERVICES

livecode

Setting Document Associations in Windows

(Windows Only)

> How do I go about setting up basic file association via
> the registry and how do I get the Rev standalone to recognize that it was
> launched with incoming parameters and access & use those parameters?

For the purposes of this example, I will assume a fictitious app called "TestApp", with a file extension of ".tst" and an installed location of "C:\Program Files\TestApp\TestApp.exe".

1) Create a key in `HKEY_CLASSES_ROOT` for the extension, and use the default value to point to the name of the application:

```
get setRegistry("HKEY_CLASSES_ROOT\.tst\","TestApp")
```

2) Create a key in `HKCR` for the application itself, using the default value to point to a descriptor of the kind of document used by the app - this will be used in list views to show the kind of file a document of TestApp is:

```
get setRegistry("HKEY_CLASSES_ROOT\TestApp\","TestApp document")
```

3) Create a subkey of `HKCR\TestApp` to hold the default icon for the application. The value used is the path to the application followed by a comma, followed by the index of the icon resource inside the application. MetaCard/Rev document icons are in the first position and I'll assume for this example that I have used an icon editor to change the icon in the first position of the TestApp executable:

```
get setRegistry("HKEY_CLASSES_ROOT\TestApp\DefaultIcon\","C:\Program Files\TestApp\TestApp.exe,1")
```

4) Create a subkey three layers deep in `HKCR\TestApp` to hold the command to open the application when the document with the ".tst" extension is launched. The path to the document is defined in the registry as %1. Any other info you want to pass to the application you can add to this line. The nice thing about MC/Rev is that you can create all these keys at once in a single command:

```
get setRegistry("HKEY_CLASSES_ROOT\TestApp\shell\open\command\","C:\Program Files\TestApp\TestApp.exe %1")
```

OK, now you have all the registry settings. The next thing to do is add some code to your app. When you launch an MC/Rev app, any command line information is sent to the application and is retrievable via *environment variables* numbered `$0` on up. The `$0` variable will always contain the path to the application you just launched, and the `$1` will contain the first command-line parameter passed (`$2` will be the second, etc.).

So some simple code for this is:

```
on openStack
  put $0 into theAppPath
  put $1 into theDocToOpen
  answer "The doc to open is: " & theDocToOpen
end openStack
```

(Note that you'll need to bundle in the answer dialog box to your standalone or you'll never see the message. ;-)

That's it! The next time you create a document with a ".tst" extension, it will take on the icon in the first position in the TestApp executable (the document icon), and if you double-click it, it will launch TestApp and its path will be passed to the standalone as `$1`.

Hope this clears things up...

Posted 7/24/2002 by Ken Ray to the Use Revolution List (*See the complete post/thread*)

Print this tip