

Utilisation de l'API Carbon avec Ada

Carbon est un élément important de Mac OS X. Il propose une compatibilité avec l'interface de programmation (API) des versions antérieures de Mac OS X. À partir d'un minimum de modification, Carbon permet de faire fonctionner sous Mac OS X quasiment toutes les applications antérieures avec une simple recompilation et quelques modifications mineures. Carbon propose des API revues et corrigées : suppression des API peu utilisées ou obsolètes, modification des API peu ergonomiques ou mal conçues au départ et introduction de nouvelles API.

1) Utilisation de l'API Carbon en C :

Nous allons effectuer un simple essai d'utilisation de la fonction `GetDateTime` qui retourne un entier représentant le temps interne du Mac.

```
% more hello.c
#include <Carbon/Carbon.h>
main () {
    UInt32 DT;
    printf("Hello, with C\n");
    GetDateTime (&DT);
    printf("Date - Time : %d.\n", DT);
}
% gcc -framework Carbon hello.c
% ./a.out
Hello, with C
Date - Time : -1198226605.
% ./a.out
Hello, with C
Date - Time : -1198226598.
```

2) Utilisation de l'API Carbon en Ada :

Nous allons traduire simplement le programme C en Ada, puis lancer la compilation avec une édition des liens comportant la bibliothèque Carbon.

```
% more hello.adb
with Text_IO; use Text_IO;
procedure Hello is
procedure GetDateTime (Sec : out Integer);
  pragma Import (C, GetDateTime, "GetDateTime");
DT : Integer;
begin
put_line("Hello again, avec Ada et maintenant Carbon.");
GetDateTime(DT);
put_line("Date - Time : " & Integer'Image(DT));
end;
% gnatmake hello -larges /System/Library/Frameworks/Carbon.framework/Carbon
gcc -c hello.adb
gnatbind -x hello.ali
gnatlink /System/Library/Frameworks/Carbon.framework/Carbon hello.ali
% ./hello
Hello again, avec Ada et maintenant Carbon.
Date - Time : -1197966509
```

Noter l'ajout de l'option :

-larges /System/Library/Frameworks/Carbon.framework/Carbon
qui permet d'utiliser la bibliothèque Carbon.

3) Installation de la bibliothèque Carbon Ada :

Récupérer la bibliothèque Carbon (pour XCode 2.2) sur le site GNAT-OSX

"<http://www.macada.org/bindings.html>"

Ouvrir "Xcode_Bindings_2_2.dmg" sous le Finder et lancer l'installation de

"Carbon_Bindings_Xcode.pkg".

Des erreurs se produisent avec les compilateurs GNAT récents.

Télécharger le correctif "carbon-bindings-2.2-patch.txt" sur :

"<http://blady.pagesperso-orange.fr/alpha.html>"

Corriger les fichiers suivant le correctif.

Puis lancer le terminal pour effectuer les commandes suivantes :

```
$ cd /usr/local/Bindings/Frameworks
```

```
$ sudo ./Compile
```

Les scripts "build_ada", "build_ada_nomain" et les outils "reformat",

"makebody" s'installent dans le répertoire :

```
/usr/local/bin
```

La bibliothèque "Carbon Ada" s'installe dans le répertoire :

```
/usr/local/Bindings/Frameworks
```

La bibliothèque "ncurses" (fenêtrage avec le Terminal) s'installe en même temps dans le répertoire :

```
/usr/local/Bindings/ncurses
```

La bibliothèque "OpenGL" (Graphisme en 3D) s'installe en même temps dans le répertoire :

```
/usr/local/Bindings/AdaGL
```

Des exemples de programmes s'installent aussi en même temps dans les répertoires :

```
/Developer/Examples/Ada/Carbon
```

```
/Developer/Examples/Ada/Command_Line
```

```
/Developer/Examples/Ada/GtkAda
```

```
/Developer/Examples/Ada/OpenGL
```

```
/Developer/Examples/Ada/Tash_Bindings.pkg
```

```
/Developer/Examples/Ada/X11
```

Les liens des bibliothèques (Framework) pour la recherche d'API s'installent également en même temps dans le répertoire :

```
/Developer/Extras/Ada.framework
```

Les modèles de projets et les scripts pour XCode s'installent également en même temps dans le répertoire :

/Library/Application Support/Apple/Developer Tools

Modifier, si nécessaire, les fichiers de configuration des bibliothèques par défaut de GNAT (voir installation de GNAT sur Blady) :

Exemple 'ada_source_path' :

```
adainclude
```

```
/usr/local/Bindings/Frameworks
```

Exemple 'ada_object_path' :

```
adalib
```

```
/usr/local/Bindings/Frameworks
```

4) Utilisation de la bibliothèque Carbon Ada :

Le source du programme devient :

```
with Text_IO; use Text_IO;
with CoreServices.CarbonCore.MacTypes;
use CoreServices.CarbonCore.MacTypes;
with CoreServices.CarbonCore.DateTimeUtils;
use CoreServices.CarbonCore.DateTimeUtils;
procedure Hello is
  DT : aliased UInt32;
begin
  put_line("Hello again, avec Ada et maintenant Carbon.");
  GetDateTime(DT'access);
  put_line("Date - Time : " & UInt32'Image(DT));
end;
```

Noter la disparition du "pragma Import" et l'apparition des unités de Carbon "MacTypes" et "DateTimeUtils" permettant l'utilisation des interfaces de programmation Carbon en Ada. Les spécifications des unités se trouvent dans le répertoire "/usr/local/Bindings/Frameworks".

```
$ gnatmake hello -larges /System/Library/Frameworks/Carbon.framework/Carbon
gcc -c hello.adb
gnatbind -x hello.ali
gnatlink hello.ali /System/Library/Frameworks/Carbon.framework/Carbon
$ ./hello
Hello again, avec Ada et maintenant Carbon.
Date - Time : 3269413616
```

5) Installation de Ada dans XCode :

Récupérer la bibliothèque Carbon (pour XCode 2.2) sur le site GNAT-OSX
"<http://www.macada.org/bindings.html>"

La bibliothèque Carbon contient aussi les modèles pour XCode, installer les modèles si cela n'est pas déjà fait :

Ouvrir "Xcode_Bindings_2_2.dmg" sous le Finder et lancer l'installation de "Carbon_Bindings_Xcode.pkg". Les modèles pour Project Builder sont placés sous

/Library/Application Support/Apple/Developer Tools/Project Templates.

Cette installation fonctionne avec le compilateur GNAT (GCC 3.3). Pour utiliser des compilateurs GNAT plus récents (GCC 4.1 à 4.5), installer alors les nouveaux modèles de projets pour XCode présents dans "AdaCarbAppXCodeProj.zip" sur :
"<http://blady.pagesperso-orange.fr/alpha.html>"

Une fois l'archive téléchargée et ouverte, placer les projets dans le répertoire:

/Library/Application Support/Apple/Developer Tools/Project Templates

ou si vous n'avez pas les droits de l'administrateur :

~/Library/Application Support/Apple/Developer Tools/Project Templates

ou pour XCode 2.5 ou 3.0 dans

/Library/Application Support/Developer/Shared/Xcode

ou si vous n'avez pas les droits de l'administrateur :

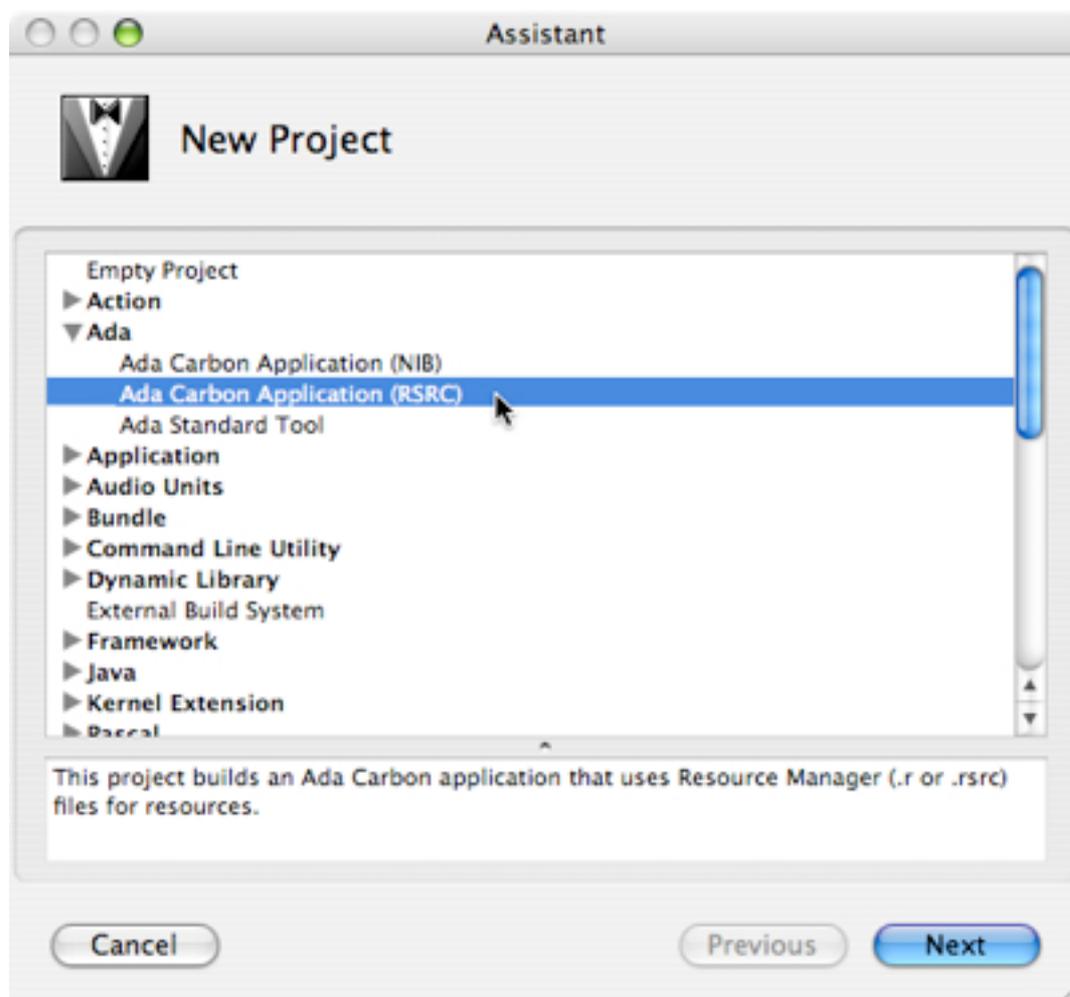
~/Library/Application Support/Developer/Shared/Xcode

6) Utilisation de Ada dans XCode :

Lancer XCode, activer le menu "New Project..." dans "File".

La fenêtre "New Project" s'affiche avec le choix entre trois projets Ada :

- Ada Carbon Application (Nib) : permet de construire des applications Carbon avec une interface utilisateur construite avec InterfaceBuilder.
- Ada Carbon Application (RSRC) : permet de construire des applications Carbon avec une interface utilisateur construite à base de ressources "rsrc",
- Ada Standard Tool : permet de construire des programmes utilisables sous Terminal.



Cliquer sur "Ada Carbon Application (RSRC)" ou "Ada Carbon Application (NIB)" suivant le type de ressources utilisées et enregistrer le projet.

Pour modifier les paramètres de compilation par défaut, activer le menu "Edit Active Target" dans "Project".

Activer l'onglet "Build", sélectionner "All Configurations", "All Settings" et rechercher les paramètres tout en bas de la page.

Modifier les champs :

GNAT_BINARIES = /usr/local/gnat/bin

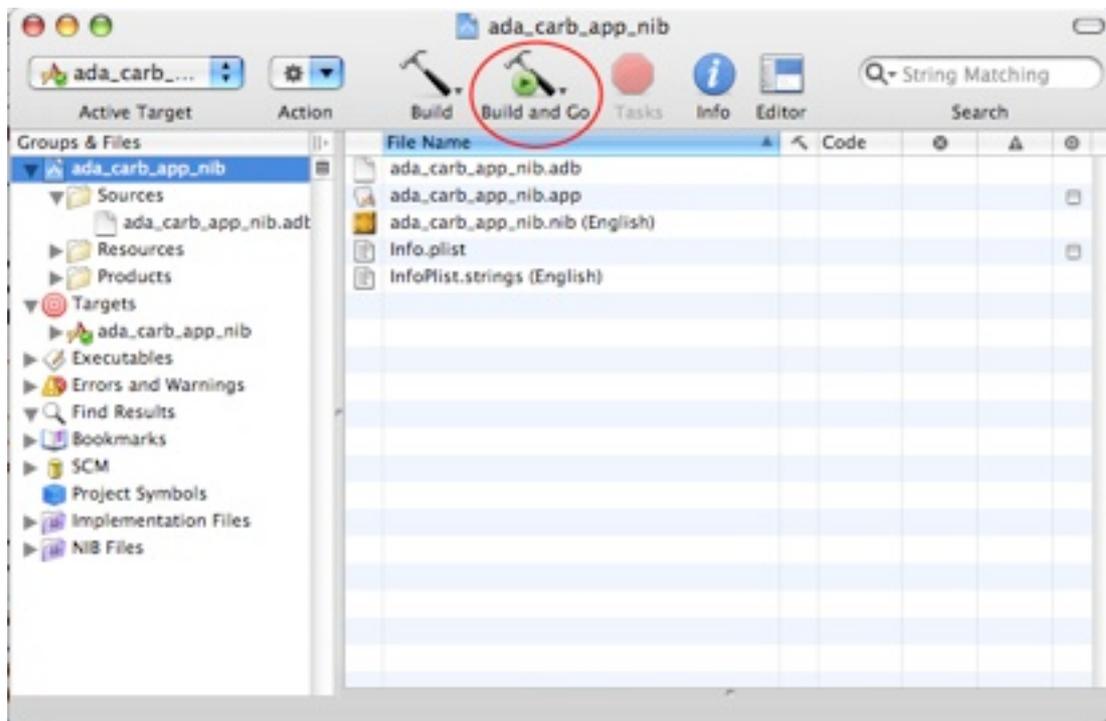
OTHER_ADAFLAGS = -gnatf -g -I/usr/local/Bindings/Frameworks

OTHER_ADALINKFLAGS = /System/Library/Frameworks/Carbon.framework/
Carbon

ADA_CARBON_FRAMEWORK_LIB = /usr/local/Bindings/Frameworks/
libAdaFrameworks.a

Vous pouvez soit les modifier suivant le compilateur utilisé ou mieux créer des liens symboliques sur le compilateur utilisé.

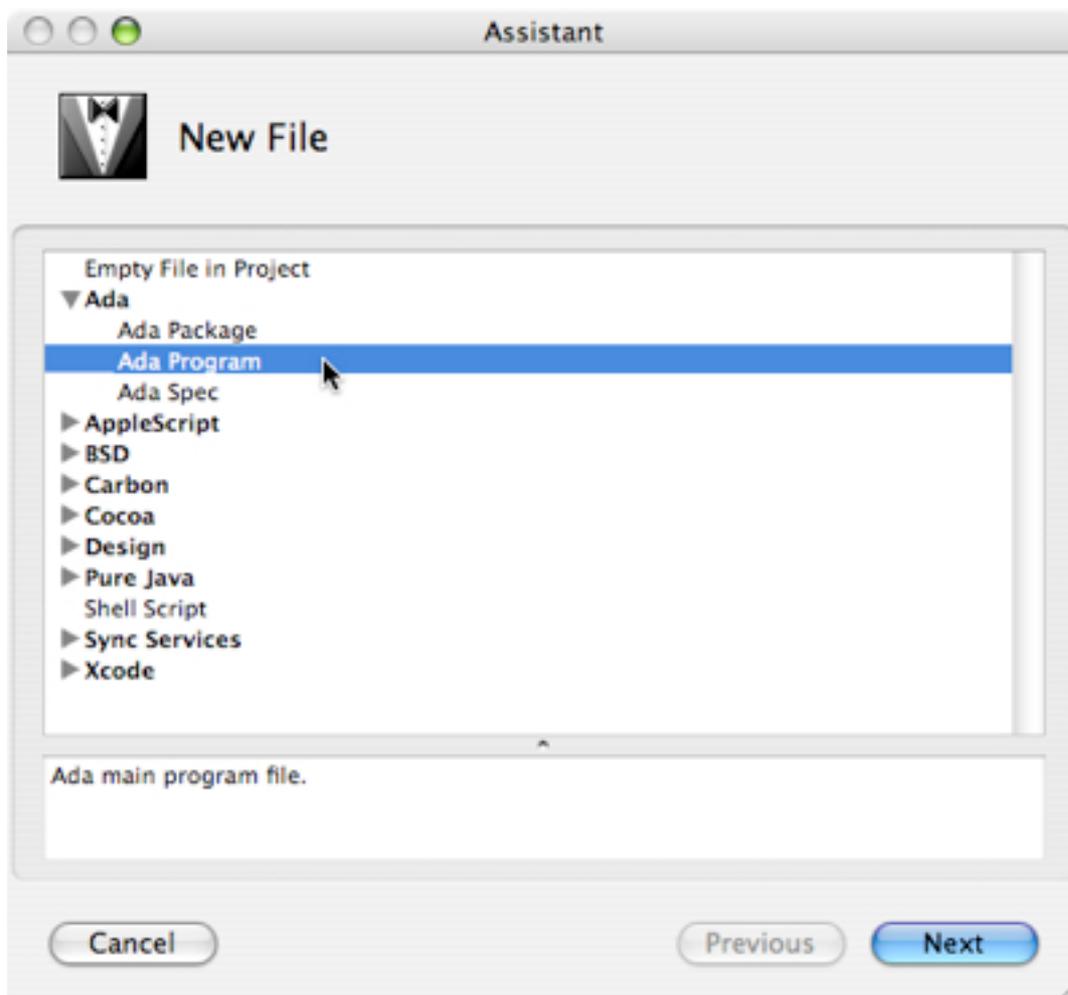
Compiler et exécuter avec le menu Build->Build and Run.



Les applications se construisent alors de manière classique avec XCode.

L'ajout de fichiers s'opère en choisissant "New File..." dans le menu "File". La fenêtre "New File" s'affiche avec le choix entre trois fichiers Ada :

- Ada Package : permet de construire le corps d'une unité Ada,
- Ada Program : permet de construire la procédure principale d'un programme Ada.
- Ada Spec : permet de construire la spécification d'une unité Ada,



7) Utilisation des scripts avec XCode

Les scripts suivants sont disponibles dans le menu User Scripts de XCode :

1. **Reformat Ada** - remplace le contenu d'un fichier par le code source formatté. Pour modifier les options de formatage, éditez le fichier /Library/Application Support/Apple/Developer Tools/Scripts/10-User Scripts/70-Ada/20-ada_reformat.sh.
Les options sont les listées dans ce fichier.
2. **Comment** - Commente le texte sélectionné.
3. **Uncomment** - Enlève les commentaires du texte sélectionné.
4. **Sweep CVS** - nettoie les traces CVS dans les répertoires du projet courant.
5. **Makebody** - génère le corps d'une spécification dans le presse papier.
Faire <cmd>-V (coller) pour récupérer le code généré.

8) Utilisation des bibliothèques avec XCode

Dans les deux modèles de projet Ada Carbon Application, l'accès aux bibliothèques Ada est pré-installé sous le nom de "Ada.Framework".

À ce jour, elles sont au nombre de cinq :

- bibliothèque du compilateur (Ada et GNAT)
- bibliothèque ApplicationServices
- bibliothèque Core
- bibliothèque Carbon
- bibliothèque ncurses

Ces bibliothèques sont visibles dans la fenêtre du projet dans le répertoire "External Frameworks and Libraries".

Pour rechercher une fonction, utiliser le remplissage automatique :

- après avoir saisi quelques caractères de la fonction, ceux-ci sont soulignés lorsque la fonction est reconnue
- pressez la touche ESC, une liste avec les différentes possibilités apparaît
- sélectionner alors celle que vous recherchez

Pour rechercher les paramètres Ada d'une fonction :

- sélectionner la fonction dans l'éditeur de source
- sélectionner le menu Find -> Use Selection for Find
- sélectionner le menu Find -> Find In Project...
- sélectionner In Frameworks et Textual dans la fenêtre de recherche
- cliquer sur Find, la liste des sources s'affiche
- sélectionner la fonction recherchée
- la source de la fonction s'affiche dans l'éditeur intégré

Pour rechercher la documentation d'une fonction :

- sélectionner la fonction dans l'éditeur de source
- sélectionner le menu contextuel Find Selected Text in API Reference
- la fenêtre de la documentation s'affiche (en C et en anglais)

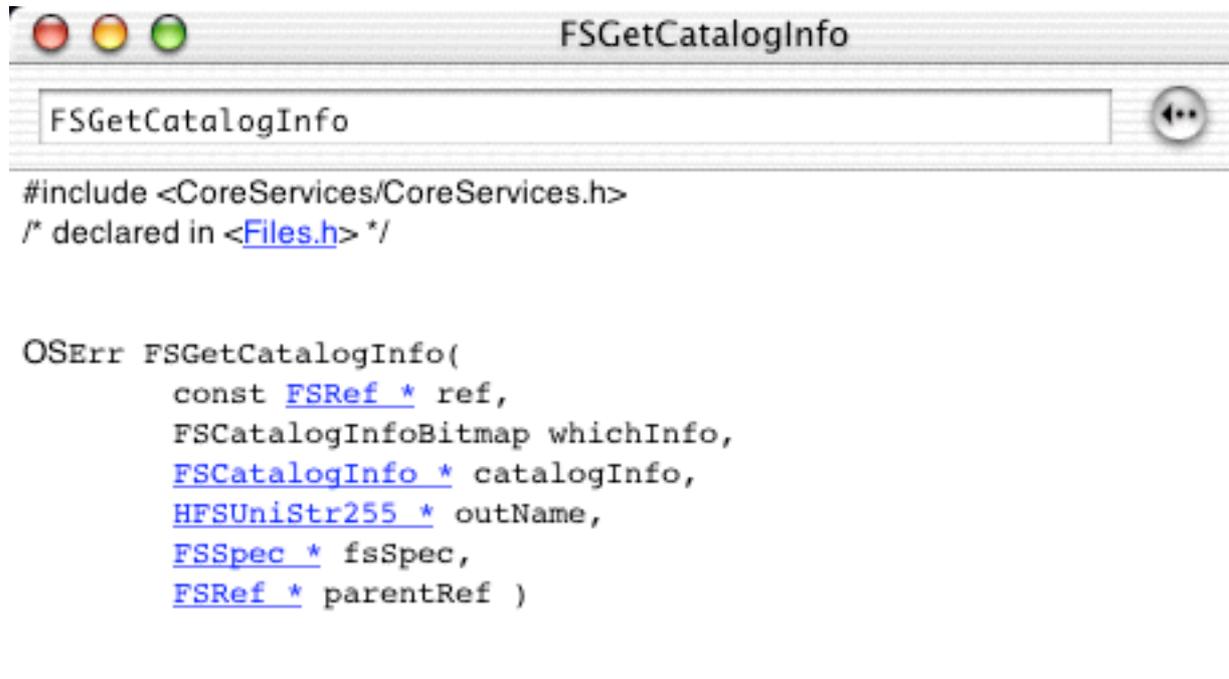
9) Outils de recherche des APIs :

Lookup est un outil précieux pour déterminer l'unité d'appartenance d'une routine de Carbon Ada :



Il ne reste plus qu'à insérer l'appel et son unité d'appartenance avec l'instruction 'with' et un simple copier / coller.

XDoc permet d'obtenir la documentation Apple des APIs, cela nettement plus rapidement qu'avec les outils natifs :



Parameters:

[ref](#)

A pointer to an FSRef specifying the file or directory for which to retrieve information. See [FSRef](#) for a description of the FSRef data type.

[whichInfo](#)

A bitmap specifying the catalog information fields to return. If you don't want any catalog information, set whichInfo to the constant `kFSCatInfoNone`. See "Catalog Information Bitmap Constants" for a description of the bits in this parameter.

10) Documentation Carbon fournie avec XCode :

Page d'accueil de la documentation Carbon :

file:///Developer/ADC%20Reference%20Library/referencelibrary/Carbon/index.html

Les trois bibliothèques constituant Carbon :

file:///Developer/ADC%20Reference%20Library/referencelibrary/API_Fundamentals/Carbon-api-date.html

Bibliothèque "Application Services" :

file:///Developer/ADC%20Reference%20Library/documentation/Carbon/Reference/ApplicationServicesFrameworkReference/index.html

Bibliothèque "Carbon" :

file:///Developer/ADC%20Reference%20Library/documentation/Carbon/Reference/CarbonFrameworkReference/index.html

Bibliothèque "Core Services" :

file:///Developer/ADC%20Reference%20Library/documentation/Carbon/Reference/CoreServicesReferenceCollection/index.html

Les codes d'erreur classés par nom et par valeur :

file:///Developer/ADC%20Reference%20Library/documentation/Carbon/indexes/ResultCodes/resultCodeNames.html

file:///Developer/ADC%20Reference%20Library/documentation/Carbon/indexes/ResultCodes/resultCodeValues.html

L'index de toutes les fonctions :

##file:///Developer/ADC%20Reference%20Library/documentation/Carbon/indexes/Functions/index.html

Traduction depuis Mac OS 9 :

file:///Developer/ADC%20Reference%20Library/documentation/Carbon/Conceptual/Upgrading_HIToolbox/index.html#//apple_ref/doc/uid/TP30001140
http://developer.apple.com/documentation/Carbon/Conceptual/carbon_porting_guide/index.html

Les APIs Carbon n'ont alors plus de secret.

11) Applications Carbon Ada avec le Terminal :

Télécharger le script "ada_carb_rsrc.tgz" sur :

"<http://blady.pagesperso-orange.fr/alpha.html>"

Une fois l'archive téléchargée et ouverte sur le bureau, un exemple d'application est fourni :

```
$ cd ~/Desktop/ada_carb_rsrc  
$ ./mk_ada_carb_rsrc.sh ada_carb_rsrc "-gnatf -g"
```

Le résultat est dans le répertoire build.

Le script mk_ada_carb_rsrc.sh est générique. Il accepte le nom du programme Ada comme un premier argument et les options GNAT comme second argument optionnel :

```
$ mk_ada_carb_rsrc.sh progamme_name [gnat_args]
```

Il est aussi personnalisable avec les variables au début du script :

```
XCODE_TOOLS : emplacement des utilitaires XCode  
GNAT_BIN :    emplacement des exécutables GNAT  
ADA_LIB :    emplacement des bibliothèques Ada  
LD_LIB :     emplacement des bibliothèques C  
GNAT_FLAGS : options de GNAT
```

Pascal Pignard, février, mars, mai 2002, janvier 2003, juillet-septembre 2007, juillet 2009.