

Porting a Mobile Application Builder application to CodeWarrior

Author: Quentin Siraut

Date: Jul 2005

Special notices

The information in this publication is not intended as a substitution of the IBM DB2 Everyplace® product documentation provided by IBM. See the Library section of the IBM DB2 Everyplace Web Site for more information about what publications are considered to be product documentation. References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates.

Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service. Information in this publication was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

The information contained in this publication was derived under specific operating and environmental conditions. While IBM has reviewed the information for accuracy under the given conditions, the results obtained in your operating environments may vary significantly. Accordingly, IBM does not provide any representations, assurances, guarantees, or warranties regarding performance. Any information about non-IBM ("vendor") products, in this document, has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness.

Trademarks IBM

DB2 Everyplace, DB2 are trademarks or registered trademarks of IBM Corporation in the United States, other countries, or both.

Other company, product, and service names may be trademarks or service marks of others.

© Copyright IBM Corporation 2005. All rights reserved.

Introduction

The following procedure has been tested with Mobile Application Builder Version 8.1.4 and CodeWarrior Version 9.0

You can use CodeWarrior's stationery to migrate your Mobile Application Builder (MAB) applications to CodeWarrior or to create new DB2 Everyplace application programs.

Getting the source code from MAB

- Build your application using the debug mode in MAB. To build your application in the debug mode, open the "File -> Preferences" and select "Build". In that screen, ensure that the debug mode is selected.
- If your application is larger than 64KB, ensure that the "Compile with multiple segments" option is selected the "File -> Preferences" window.
- Extract from the MAB project directory the files that you will need to port your application to CodeWarrior:
 - All source code files: MAB_Main.c, MAB_Forms.c, MAB_Tables.c, MAB_FormHandler.c
 - All header files: Acallback.h, isyncore.h, MAB.h, MAB_Common.h, MAB_Events.h, MAB_Messages.h, MAB_Tables.h, sqlcli.h, sqlcli1.h, sqlext.h, sqlsystem.h.
 - If your application uses a barcode scanner, the project directory contains the following additional header files: MAB_Scanner.h, ScanMgr.h, ScanMgrDef.h, ScanMgrStruct.h
 - If your application uses a printer, the project contains the following additional header file: MAB_Print.h.
 - MAB.rcp
 - MAB_Icon.bmp: You might need this file if you kept the default application bitmap within MAB.
 - MAB.def: You need this file as a guide if you have to split the code.
- Ensure that your source code files are smaller than 64 KB. If your source code files are larger than 64 KB, ensure that you used the multiple segments option, and build the application again. If the source code files are still larger than 64 KB, complete the tasks in "Splitting the source code". If all of your source code files are smaller than 64 KB, go to the "Creating the CodeWarrior project" step.

Splitting the source code

If any source code file is larger than 64 KB, you must divide the source code files into smaller files that are smaller than 64 KB. MAB uses virtual segmentation. This method contains the steps for splitting the code into as many files as there are virtual segments.

For example, a MAB_Forms.c file might contain the following lines:

```
SQLRETURN  
SelectFrom_Form2Table7(SQLCHAR sqlStmt, Boolean fetch)  
__attribute__((section ("formfn2")));  
void Form3DrawForm() __attribute__((section ("formfn3")));
```

In this example, "formfn2" and "formfn3" are two different virtual segments. The binaries for "SelectFrom_Form2Table7" method are located in the "formfn2" virtual segment, and the binaries for "Form3DrawForm()" are located in the "formfn3" virtual segment.

The list of the virtual segments that MAB defined is located in a the "MAB.def" file in the MAB

project folder. For example, this file might contain the following lines:

```
application
{"APPNAME" LAB1 stack=0x8000 }
multiple code { userfns helpfns tablfn1 tablfn2 tablfn3 tablfn4
formfn1 formfn2 }
```

The "multiple code" section defines the virtual segments that are defined by MAB. These virtual segments correspond to the new source code files that you must create.

To split the code, complete the following steps:

- In the MAB_Forms.c and MAB_tables.c files, create one file per table and one file per form.
- In the MAB_Tables.c file, search for the method prototypes. Move these prototypes to the first source code file "MAB_Tables1.c". In the new source code file, remove the virtual segment part.
- In the MAB_Tables.c file, find the implementation of the methods, and move them to the new C file.
- Complete the above steps for each of the virtual segments.
- Repeat the above steps for MAB_Forms.c.
- In the file MAB_FormHandler.c, move the event handlers to the corresponding MAB_Forms_X.c file. If the corresponding event handler method prototypes are declared as `extern`, you can remove that declaration.
- In the file MAB_Main.c, remove the functions that will be located in the "helpfns" section, as well as those that will be located in the "userfns" section, then create the two new corresponding files "helpfns.c" and "userfns.c"

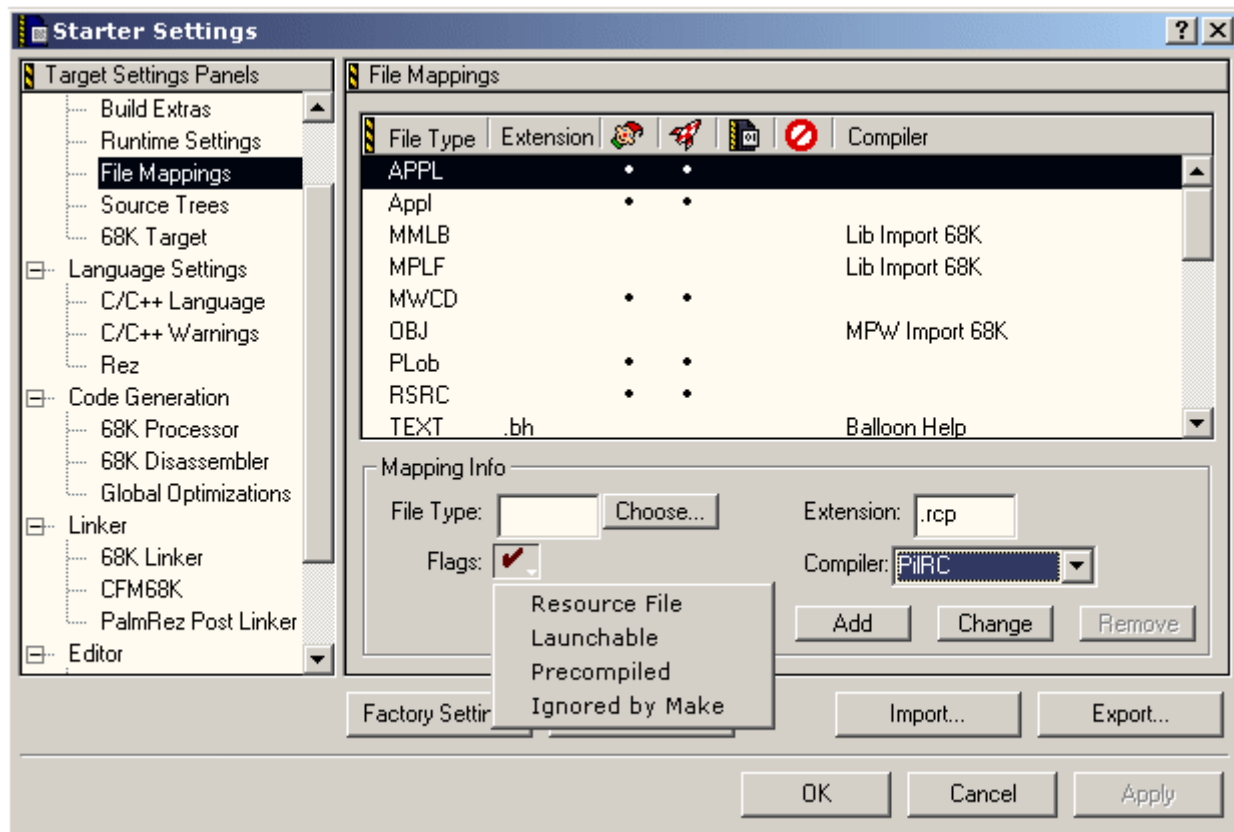
Important: While splitting your files, be careful when dealing with global variables that might have been declared. If you move them to a different file, you may need to declare them as `extern` so that you can use them in the different C files.

Porting the application to CodeWarrior

If your source code files are less than 64 KB in size, you can begin to port your application to CodeWarrior.

Creating a new project file

- Open CodeWarrior, and create a New project by selecting "File -> New" from the menu.
- Click on Palm OS Stationery, and enter an appropriate name for the project and a path where the project will be created. Click OK.
- Select the appropriate type of stationary.
 - If your application does not use multiple segments, select "Palm OS C App".
 - If your application does use multiple segments, select "Palm OS Multi-Segment App".
- From the project explorer, delete any C file that is in the "Source" folder
- From the project explorer, delete any C file that is in the "Resources" folder
- For each of the configurations (debug, release) you want to define the following:
 - If you are using CodeWarrior Version 9: click "Target Settings". Change the Linker to "Palm OS 68K".
 - If you are using CodeWarrior Version 8:
 - Click "File Mapping" Choose. Select the MAB.rcp file.
 - Check of all the flags and select PilRC as the compiler, as shown below



- Click Add, then click OK.

Adding the source code to the project

Complete the following steps to add the source code to the project:

- If you used the multiple segments option, manually create segments for each of the configurations (debug, release), and then add the source files (.h and .c) of your application in the different segments. You can add more than one file to a segment, but the total size of the segment must be smaller than 64 KB.
 - To create a segment, select the correct configurations from the project explorer, then select the "Segments" panel of the project explorer, then click on the "Project" global menu -> "Create Segment".
 - To add a file to a segment, in the project explorer, right-click on the segment folder, select "Add Files"
- If you did not use the multiple segments option, add all of the .h and .c files that were generated by Mobile Application Builder to your project.

Recommendation: YCopy the source files to the /Src folder of the CodeWarrior project before adding them to the project. This way, you will have a clean project file, which does not use any files from outside of the project directory.

Important: *MAB_Main.c and the Palm OS runtime library must be in the first segment.

Adding resources and libraries to the project

To add resources and libraries to the project, complete the following steps:

- Right-click on "Resources" in the project explorer and select "Add Files". Add MAB.rcp.

- Click Project -> Add Files, and add MAB_Icon.bmp
- Add the following libraries:
 - DB2e.lib: To add this file to the project, click Project -> Add files and select DB2e.lib. This file is in the Clients\PalmOS\database directory.
 - You might have to add PalmOSGlue.lib, which is located in Palm OS Support\CodeWarrior Libraries\Palm OS Glue in the CodeWarrior Version 9 install directory.
 - If your application involves synchronization, add isynconf.lib and isyncore.lib to the project. These files are located in the Clients\PalmOS\sync directory.
 - If you are using any special feature of MAB, such as integrating a printer or a scanner, add the corresponding libraries to the project, using the same previous procedure. For those devices, add the device library (driver) and the MAB's corresponding library (wrapper).

The device library for the scanner and printer capabilities are located under CW for Palm OS Support/(Other SDKs)/Symbol/Lib/ in the CodeWarrior 9 install directory (ScanMgr.lib for the scanner, ptstatic.lib for the printer).

Building MAB's scanner and printer wrappers for CodeWarrior

We are providing two CodeWarrior Version 9 project files, MABScanner.mcp and MABPrinter.mcp. These files allow you to build the MAB wrappers libraries. These wrappers libraries are a way to provide a unified API, which does depend on the mobile device that you use, for all MAB applications. These files are available on the DB2Everyplace website <http://www.ibm.com/software/data/db2/everyplace>, in the Library section.

After you download and extract the files, complete the following steps:

- Open the project file with CodeWarrior 9
- Depending on the language of your application, you might want to update the file that defines the output messagees (MAB_Print_Messages.h for the printer, MAB_ScannerMessages.h for the scanner):
 - Remove the default language (English) from the project (right-click on the file from the project explorer -> remove)
 - Add the new language (right-click on the Source folder from the project explorer -> add files) which is under the /Src/&LANG& directory of the project.
- Build the library (Make icon). After building, the library is the available in the Obj\ directory of the project.
- Add the library to your main project (click Project -> Add files and select the library you just built. You may want to copy this library to a subdirectory of your project before adding it).

Building the prc

Click the Make button to build one of the configurations. The prc is available in the Obj\ directory in the CodeWarrior project directory.