CrossFPC on the move

Michaël Van Canneyt

October 4, 2005

Abstract

Some time ago, a new project was introduced: CrossFPC. CrossFPC is a plugin for Delphi, which allows to cross-compile CLX applications with FPC, to any target supported by FPC and FreeCLX. Meanwhile FPC and CrossFPC have reached a major milestone. A progress report.

1 Introduction

In one of the previous issues, CrossFPC was presented: CrossFPC is a replacement for CrossKylix. CrossKylix and CrossFPC have the same goal: Cross-compiling CLX applications for linux right from the Delphi IDE. CrossKylix was a plugin for Delphi that allowed to cross-compile CLX applications from the Delphi IDE using the Kylix compiler. Since support for Kylix has been officially discontinued, an alternative has been developed using the Free Pascal compiler: CrossFPC. More about this project can be found on it’s website:

http://www.crossfpc.org/

At the time of the first report, crossfpc was not yet capable of much: only the most basic console applications could be compiled. Since that time, a lot has changed, meanwhile a third beta has been released: time for an update.

2 The progress

The area where the biggest progress has been made is the FPC compiler: it is of course the fundament on which CrossFPC is built. There has been improvement in the following areas:

- The Libc unit distributed with FPC has been made more compatible with the one shipped with Kylix.
- There have been various patches to basic units.
- A special Kylix compatibility unit has been introduced, which is included by default when cross-compiling in CrossFPC. This unit masks differences in the basic RTL routines of Free Pascal and Kylix.
- Many compiler bugs and deficiencies have been fixed, allowing the compiler to compile most of the CLX.
- Last but not least, FPC now has support for embedded resources: a special tool was made to include resources in the application.
The CrossFPC installation itself was improved: a patched FreeCLX has been included instead of the original FreeCLX by Borland.

As a result, with CrossFPC the following should now be possible:

- Compile and run console applications.
- Compile and run Visual CLX applications. (Including resources)
- Compile DBExpress applications. Due to some bugs still under investigations, these applications crash when run.

The main missing part for complete support of the CLX is the use of TClientDataset datasets (DataSnap or Midas). Due to some compiler issues, the units for DataSnap do not yet compile.

In figure 1 on page 2 a screenshot of CLX applications compiled with CrossFPC is shown. Both applications run out of the box on a SuSE 9.3 distribution: No extra deployment libraries were needed.

3 Conclusion

6 months after the initial start of the project, a lot has been achieved. Not only the technical side of things (although this is most prominent) but the FreeCLX initiative has obtained a
license from Borland, allowing to perfect and enhance the FreeCLX: this ensures that any CLX code out there has a future.