# C++ Symbian Series 60 Application Development with Borland C++ Builder

Sasu Tarkoma

## Contents

- Requirements
- Installation
- Integrated Development Environment
- Creating projects and building applications
- Debugging
- Deployment

## Installation

- Borland C++ Builder with Series 60 SDK
- You may download the Borland C++ Mobile Edition from the Nokia Forum
  - requires registration (free)
- The package includes the Series 60 SDK 1.2 for Symbian OS (~300MB)

## Nokia Series 60 SDK

- C++ development for Series 60 phones
- documentation, API reference, add-on tools, emulator, target compiler
- No IDE, compatible with Borland C++ Builder, Visual Studio C++ .NET and 6.0
- Needs Java JDK v. 1.3.1 and ActivePerl 518
- Version 1.2 is for OS 6.1 and 2.0 for OS 7.0

## Borland C++ Builder IDE

- ARMIB udeb
  - a debugging build for the Nokia Series 60 SDK
- ARMIB urel
  - a final build for the mobile device
- WINSB udeb
  - A debugging build for the WINS Emulator
- WINSB urel
  - A final build for the WINSB Emulator

## Borland Files

- .sis
  - Installation files
- .pkg
  - Package files for creating SIS installation files
- .mmp
  - Project files
- .inf
  - Component description file

# MMP files

- Defines the properties of a project
  - platform and compiler dependent
  - each statement on a separate line
  - C++ style comments
  - May be several MMP files for different modules or DLLs

# Bld.inf file

- The component definition file
- Lists all components in the project
- Used by bldmake to define the abld.bat and makefiles to be created when you build the mobile application

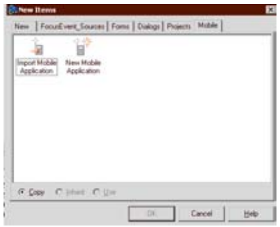
# PKG file

- The package description file
- A text file that contains installation information
- It is used to build the sis file which calls the deployment tool
- Custom features: text files to be displayed during installation, additional languages, etc.
- After building for ARMIB, you can choose Tools|Mobile Build Tools|Create Default Package File.

# Build Environment

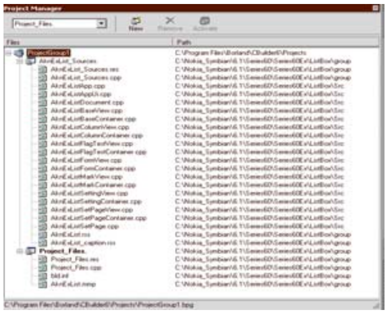
- Nokia C++ toolkit for Symbian 6.1
  - emulator for Series 60 phones (WINSB)
  - tools for building, and creating installation packages (.sis)
  - command-line build environment
    - Uses Perl and GCC
    - bldmake creates abld.bat
  - Borland C++ Builder for Symbian
    - IDE for creating Series 60 programs

# Create Application



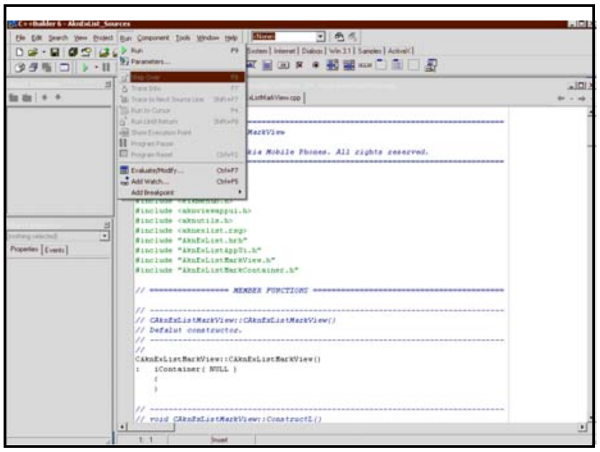
● Copy ○ Inherit ○ Use

# Project Manager



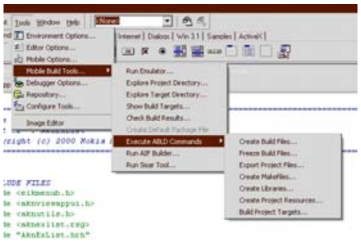
| Files | Path |
|---|---|
| ProjectGroup1 | C:\Program Files\Borland\CBuilder6\Projects |
| AknExList_Sources | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\group |
| AknExList_Sources.res | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\group |
| AknExList_Sources.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\group |
| AknExListApp.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListAppUi.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListDocument.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListBaseView.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListBaseContainer.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListColumnView.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListColumnContainer.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListFlagTextView.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListFlagTextContainer.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListFormView.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListFormContainer.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListMarkView.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListMarkContainer.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListSettingView.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListSettingContainer.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListSetPageView.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListSetPageContainer.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExListSetPage.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\Src |
| AknExList.rss | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\group |
| AknExList_caption.rss | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\group |
| Project_Files | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\group |
| Project_Files.res | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\group |
| Project_Files.cpp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\group |
| bld.inf | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\group |
| AknExList.mmp | C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\group |

C:\Program Files\Borland\CBuilder6\Projects\ProjectGroup1.bpg


C++Builder 6 - AknExList_Sources
File Edit Search View Project Run Component Tools Window Help
<None>
Run F9
Parameters...
Step Over F8
Trace Into F7
Trace to Next Source Line Shift+F7
Run to Cursor F4
Run Until Return Shift+F8
Show Execution Point
Program Pause
Program Reset Ctrl+F2
Evaluate/Modify... Ctrl+F7
Add Watch... Ctrl+F5
Add Breakpoint
System | Internet | Dialogs | Win 3.1 | Samples | ActiveX |
ListMarkView.cpp
(nothing selected)
Properties | Events
1: 1
Insert

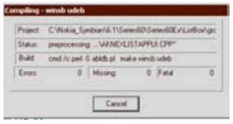
```
MarkView
kia Mobile Phones. All rights reserved.
#include <aknviewappui.h>
#include <aknutils.h>
#include <aknexlist.rsg>
#include "AknExList.hrh"
#include "AknExListAppUi.h"
#include "AknExListMarkView.h"
#include "AknExListMarkContainer.h"

// ================= MEMBER FUNCTIONS =======================

// ----------------------------------------------------------
// CAknExListMarkView::CAknExListMarkView()
// Defalut constructor.
// ----------------------------------------------------------
//
CAknExListMarkView::CAknExListMarkView()
:   iContainer( NULL )
    {
    }

// ----------------------------------------------------------
// void CAknExListMarkView::ConstructL()
```

# IDE



```
right (c) 2000 Nokia
UDE FILES
de <eikmenub.h>
de <aknviewappui.h>
de <aknutils.h>
de <aknexlist.rsg>
de "AknExList.hrh"
```

# Compilation

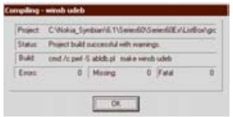


Project: C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\grc

Status: preprocessing ..\AKNEXLISTAPPUI.CPP"

Build: cmd /c perl -S abld.pl make winsb udeb

Errors: 0 Missing: 0 Fatal: 0



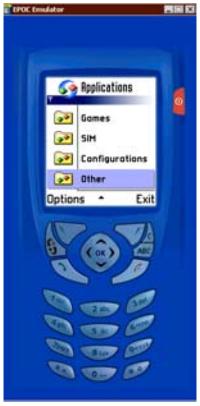
Project: C:\Nokia_Symbian\6.1\Series60\Series60Ex\ListBox\grc

Status: Project build successful with warnings

Build: cmd /c perl -S abld.pl make winsb udeb

Errors: 0 Missing: 0 Fatal: 0

# Emulator



# File system on the Emulator and on the Devices

- EPOC always has two disks present, and may have more:
  - c: RAM disk, full read/write file system
  - z: ROM disk, read-only file system. Contents are built by the EPOC OEM.
  - d:/e: MMC card, removable read/write media.

# Building Applications

1. Open project
2. Select Project|Build All Projects
3. Save project if not saved already (will be prompted)
4. Build process begins
5. Compiler progress dialog
6. Run in the emulator (WINSB) or deploy to the mobile device (ARMIB targets)

## Build errors

- Missing
  - One of the targets that was supposed to be built was not built. For example: unresolved references in the project.
- Errors
  - The build failed with an error because of compiler or linker errors, resource file errors, missing import libraries, etc. Usually something is missing.
- Fatal
  - The build failed because of problems with the build tool chain (out of disk space, could not find the compiler, etc.) usually related to configuration.

## Locating Files

- You can use the command Tools|Mobile Build Tools|Explore Target Directory to locate the files resulting from the build.
- ARM/DEBUG files are found here:
  - \Nokia_Symbian\6.1\Series60\Epoc32\Release\armi\udeb
- ARM/RELEASE files are found here:
  - Nokia_Symbian\6.1\Series60\Epoc32\Release\armi\urel
- WINSB/DEBUG files are found here:
  - \Nokia_Symbian\6.1\Series60\Epoc32\Release\winsb\udeb\z\system\apps\
- WINSB/RELEASE files are found here:
  - \Nokia_Symbian\6.1\Series60\Epoc32\Release\winsb\urel\z\system\apps\

## Building from the command line

- You may also build applications from the command line (SDK documentation)
- Debug build for Symbian OS emulator:
  - bldmake bldfiles
  - abld build winsb udeb
  - Target device build
  - bldmake bldfiles
  - abld build armib urel

## Deployment

1. Create a package (automatically created for “Build Symbian Installation System” in Borland IDE)
2. Build and create .sis file
3. Deployment to actual hardware (ARM target)
   - You need to have a connectivity software
   - Connect using Bluetooth, infrared, cable,..
   - Transfer the installation (.sis) package
     - Manually or using the Project|Deploy to Device
   - Run the installation package

## Debugging Mobile Applications

- A debug build contains additional debugging information that is omitted from the final build
- You can debug Symbian applications using the Borland C++ integrated debugger
- Breakpoints, use watches, view low-level information in the CPU or FPU windows, display function calls in the Call Stack watch window, and step through your program

## Debugging on the Emulator 1/2

- Build application with at least one breakpoint set, which will be hit when the program is run (click in the left margin to set the breakpoint)
- Choose Run|Parameters
- After Host Application, point to the path of the Emulator (for example, Nokia_Symbian\6.1\Series60\Epoc32\Release\winsb\udeb\epoc.exe) and click Load.
- The CPU window is displayed.

# Debugging on the Emulator 2/2

- Within the CPU window, press F9 to launch the Emulator. It may take several minutes to launch.
- When the Emulator is displayed, click the Other folder to locate and run your application. If you have breakpoints set, the application will stop at those places and you can use the debugging features in C++ Builder.
- Make sure to use debug function keys (F9, F8, F7, F4, etc.) within the CPU window.