



10年口碑积累，成功培养50000多名研发工程师，铸就专业品牌形象

华清远见的企业理念是不仅要良心教育、做专业教育，更要做受人尊敬的职业教育。

《Symbian 手机应用开发标准教程》

作者：华清远见

专业始于专注 卓识源于远见

第 3 章 Symbian 手机应用开发过程

本章目标

本章主要介绍基于 Symbian OS 的手机应用开发的基本过程，使读者掌握整体的开发过程以及工程的组织、建立、编译、发布等，通过本章的学习，读者将会掌握以下内容。

- 掌握 Symbian 程序的基本类型
- 掌握如何建立一个新的 Symbian 工程
- 掌握 Symbian 工程的结构
- 掌握 Symbian 工程的编译
- 掌握如何打包发布一个 Symbian 程序

专业始于专注 卓识源于远见

3.1 Symbian OS 上的程序类型

基于 Symbian 的智能手机应用程序主要有 3 种类型：控制台程序、有界面的 GUI 应用程序和 DLL 程序，具体开发哪一种类型的应用程序，开发之初根据要实现的功能进行选择。

3.1.1 控制台应用程序

在 Symbian 中，控制台应用程序是没有图形界面的，这与在 Windows 平台上的 DOS 程序只有文本提示是相同的，控制台应用程序可以实现对数据的处理，或者实现为运行在后台完成特定功能的程序。

控制台应用程序还可以再简单地分为两类：没有控制台输出界面、有控制台输出界面。

没有输出界面的控制台应用程序，程序启动后运行在后台，主动执行任务，或者为其他程序提供服务。这种程序没有界面、不和用户进行交互，可以用这种类型的控制台应用程序实现一个简单服务程序，为手机上的其他程序提供服务。

有控制台输出界面的控制台应用程序，用户可以把一些信息打印到控制台上，如图 3.1 所示。

这种类型的控制台应用直接在控制台输出程序，如图 3.1 中的“Hello, world!”就是在应用中输出的。

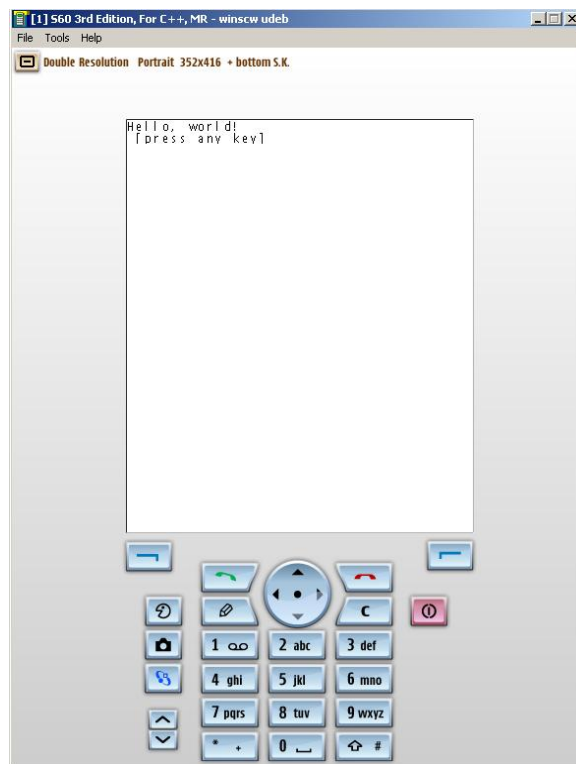


图 3.1 控制台应用的输出

控制台应用程序直接在模拟器或设备上运行，如果要在模拟器中运行这类程序，打开一个 DOS 窗口，在命令行输入下面的命令：

```
eshell.exe
```

这个命令会以控制台的形式打开 S60 模拟器，在 S60 模拟器的控制台窗口中输入一个控制台应用的名称，如 Exam0301Console.exe，程序运行效果如图 3.1 所示。

例子代码 Exam0301Console 可以通过 Carbide.c++1.3 进行创建，创建时程序类型选择“Generic Symbian OS”下的“Basic console application (EXE)”，然后单击“Next”按钮，以后所有选项都选择默认，直到完成，最后编译，按照上面的方法运行程序。

3.1.2 GUI 程序

在 Symbian 中的 GUI 程序是有丰富图形界面的应用程序，在用户手机的程序应用界面中，通过单击应用图标启动的应用都是 GUI 的应用程序，在这样的程序中，用户可以通过手机键盘控制程序中的视图切换，在编辑框中输入文本、浏览信息等。

Symbian 有两个 UI 实现 S60 和 UIQ，开发过程也是在这两个 UI 之上开发图形化、使用组件技术的 GUI 程序。下面使用 Symbian 中最有名的 UI 实现—S60 来介绍 GUI 程序的组成部分，如图 3.2 所示。

1. 按键定义

首先介绍 S60 智能手机对按键的定义，S60 手机都有如下的按键。

- 程序键：在手机待机状态，按下程序键后，手机界面会切换到功能选择界面，在这里，手机用户可以选择执行手机提供的功能和用户安装的程序。

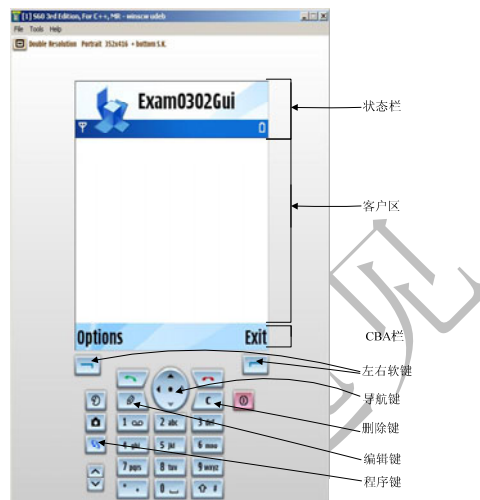


图 3.2 GUI 程序界面结构图

- 左右软键：这两个键分别对应手机应用界面左下角和右下角提示的功能，一般由应用定义这两个键的功能，通常左软键定义为肯定，右软键定义为否定。
- 导航键：由上、下、左、右 4 个方向键和一个确认键组成，用于控制程序中光标的移动或移动程序的被选中区域，确认键用于确认或打开用户选中的程序。
- 删除键：在编辑模式下，按该键可以删除输入的字符，在手机的选择程序界面，也就是按程序键后进入的所有程序列表界面，按该键可以卸载程序。
- 编辑键：在编辑模式下，按该键可以调出输入法选择框，让用户选择合适的输入法。早期版本的手机上都有这个键，现在有些手机已经不再使用这个编辑键了，而使用“#”号键切换输入法。

2. GUI 程序组成

在 S60 上，一个标准的 GUI 程序由从上到下的 3 部分组成：状态栏、客户区和 CBA 栏。

在状态栏上，主要显示程序的图标、程序的标题等，在指定的区域显示有系统全局的一些信息，如手机的信号指示、电池电量指示等，如图 3.3 所示。



图 3.3 状态栏

在程序界面的中间位置，占据大部分区域的是程序的客户区，用户程序可以在这个客户区输出文字、绘图等，也可以建立 view，然后在 view 中创建 UI 组件。

在程序界面的最下部分是 CBA 栏，在 CBA 栏主要显示应用定义的命令。CBA 栏提示的左按钮命令和右按钮命令分别对应左软键和右软键功能，左软键对应的 CBA 栏左按钮命令主要用来弹出程序的菜单。

3.1.3 DLL 程序

Symbian 作为一个操作系统，同样也支持 DLL，和其他的操作系统平台（如 Windows）一样，在使用 DLL 时，首先在程序中包含 DLL 的头文件，在编译和链接时，包含与 DLL 一致的 LIB 库文件，在程序运行时，把 DLL 复制到使用 DLL 的 EXE 所在的目录中或复制到系统的搜索路径中。

DLL 一般用来封装一些功能相对独立的功能函数，以利于其他程序重复使用 DLL 封装的功能。例如，可以把 MD5 摘要的计算封装在一个 md5encode.dll 的动态链接库中，这样我们就可以直接在任何的 Symbian 程序中使用这个 md5encode.dll 提供的方法计算指定字符串的 md5 摘要，而不用在程序中再实现 md5 算法。

DLL 是通过二进制的方式实现代码的重复利用的，在运行时，EXE 与所使用的 DLL 运行在同一个进程的地址空间中，系统会为 EXE 自动加载所使用的 DLL，系统只会加载一份 DLL 的副本到系统中，当有其他程序 B 也需要使用 DLL 时，系统发现 DLL 已经被另外的程序加载，这时系统就会给程序 B 一个 DLL 的映射，并且给 DLL 使用引用加 1，当程序退出时，系统发现 DLL 的使用引用减为 0，则把 DLL 从系统中卸载。

Carbide.c++1.3 中提供了生成 DLL 的向导，可以通过这个向导生成工程，填入自己要实现的功能代码就可以了。

3.2 建立新工程

3.1 节主要介绍了 Symbian 中的 3 种程序类型，相比 EXE 程序和 DLL 程序的工程组织及实现，GUI 程序实现及工程组织等涉及的内容比较多。在以后的开发过程中，基本上都是开发基于图形化的 GUI 应用，即使在一个项目中需要开发一个 EXE 的程序，基本上也是实现 GUI 应用的一个辅助功能。

下面讲解使用 Carbide.c++1.3 建立一个 GUI 应用程序的过程。

(1) 打开 Carbide.c++1.3，选择“File”→“New”，在弹出的子菜单中选择“Symbian OS C++ Project”，会打开新建 Symbian 工程向导，如图 3.4 所示。在向导的初始界面选择“S60 3rd Edition”下的“3rd-Future Ed. GUI Application”，创建一个 S60 3rd 下的 GUI 应用程序，单击“Next”按钮。

如果想建立 S60 2nd 下的 GUI 应用程序，可以使用 S60 2nd SDK 自带的应用程序创建向导或者使用 Carbide.c++1.2 的创建向导。

(2) 在弹出的对话框中输入新建工程的名字，如“Exam0303NewGui”，如图 3.5 所示，单击“Next”按钮。

(3) 在弹出的对话框中选择要使用的 S60 3rd 的 SDK，如图 3.6 所示。

选择 SDK 后，单击“Next”按钮进行后面的向导设置，一直单击“Next”按钮，直到单击“Finish”按钮完成新工程创建向导，新工程创建完成。

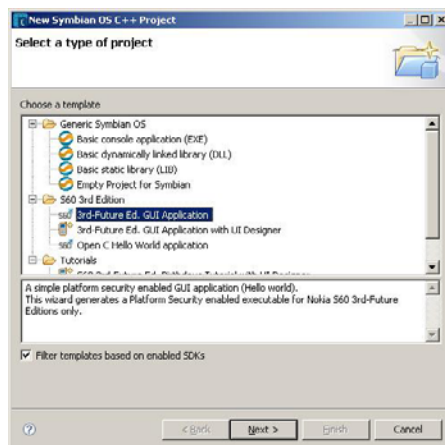


图 3.4 新建 Symbian 工程向导

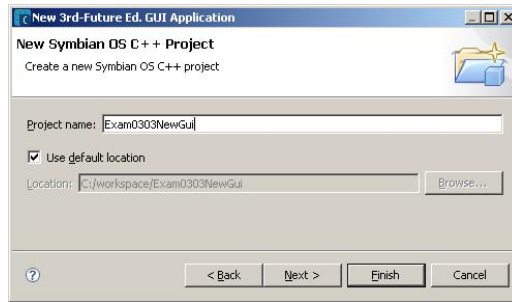


图 3.5 添加项目名称向导

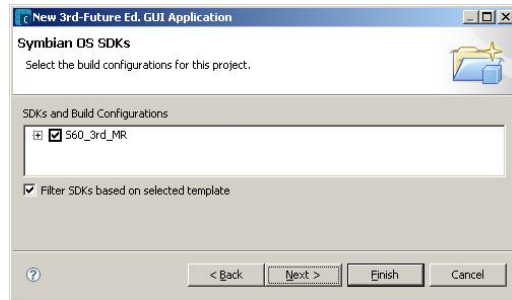


图 3.6 选择 SDK 向导

在 Carbide.c++中的项目管理器中，选择上面创建的项目“Exam0303NewGui”，在项目名称上单击右键，在弹出的快捷菜单中选择“Build Project”，开始编译工程，默认是在模拟器上编译。

3.3 工程结构

3.3.1 工程目录组织

在一个新建的目录中，通常包含如图 3.7 中所示的目录结构。

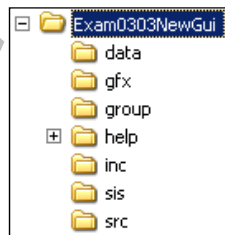


图 3.7 工程目录结构

这个工程目录结构的子目录的名称是 Carbide.c++向导给推荐的，在使用向导生成工程时可以根据需要进行修改，下面是对项目中所使用的目录的详细介绍。

- **group** 目录：该目录主要存储工程的组织文件 `bld.inf` 和 `*.mmp` 文件，在 S60 3rd 的 Symbian 开发环境中，程序的图标及程序使用图片的编译文件 `*.mk` 文件也存储在该目录中。
- **inc** 目录：该目录下主要存储的是程序代码所使用的头文件。
- **src** 目录：该目录下主要存储的是程序代码的源程序文件，在 Symbian OS C++中，也就是 `*.cpp` 文件。
- **data** 目录：该目录下存储的是程序的资源文件。
- **gfx** 目录：该目录下存储的是程序的图标文件，在 S60 3rd 中图标文件是 SVG 格式的，在 S60 3rd 的程序中可以使用位图或 SVG 图进行程序界面的绘制，这些在程序中使用的图片可以直接存储在这个目录，或者可以再建立一个专门用来存放程序中所使用图片的目录，如建立一个 `lookandfeel` 目录，专门用来存储跟程序界面有关的图片。

- help 目录：该目录主要存储程序的帮助文件。
- sis 目录：该目录主要存储制作安装程序时使用的*.pkg 文件，将来生成的安装文件*.sis 也会存储在该目录。

3.3.2 程序文件的组织

一个 Symbian 工程是由很多个文件组成的，包括程序代码文件、资源文件、图标文件、图片文件等，这些文件之间的关系主要是由 group 目录下的编译信息文件 bld.inf 和工程组织文件*.mmp 组织到一起的。在 bld.inf 文件中，主要记录了该工程可以在什么平台下编译和该工程包含哪些工程组织文件，在 S60 3rd 开发中常见平台主要有 winscw、gcce、armv5 等。Symbian 工程的编译是从这个文件 bld.inf 开始的，这个文件名是固定的。

在*.mmp 文件中，主要存储有下面的这些程序信息。

- 可执行程序名称、程序的类型、程序的 UID 等。
- 程序的代码源程序文件*.cpp。
- 需要编译资源文件。
- 编译连接程序需要使用的*.lib 库文件。
- 程序中使用的用户头文件目录和系统头文件目录。
- 程序使用的 API 的能力及必要的安全信息。

3.4 程序编译

当程序建立好后，就可以进行编译、测试刚刚建立的程序框架是否是我们想要的，只有编译好的程序才可以在模拟器上安装到设备上运行，才能看到程序执行的结果。

Symbian 的开发环境，不管是基于 S60 的，还是基于 UIQ 的，都能够支持两种编译方式：基于命令行的编译和基于 IDE 环境的编译。

3.4.1 命令行编译

在使用命令行进行编译时，首先要保证所使用的 Symbian 开发环境平台是要编译的 Symbian 的目标平台，也就是说，如果想把一个程序编译成 S60 3rd 的手机上运行的程序，那么就要切换到 S60 3rd 的开发环境平台。

1. 切换 Symbian 开发环境

能进行 Symbian 程序开发的环境很多，如 S60 2nd、S60 3rd、UIQ 2.0、UIQ3.0 等，而且这些 SDK 可以同时安装在开发人员的计算机上。如图 3.8 所示，笔者用来开发的计算机上就安装了两个 SDK：S60_2nd_FP2_CW 和 S60_3rd_MR。

```

c:\WINDOWS\system32\cmd.exe
Microsoft Windows XP [版本 5.1.2600]
(C) 版权所有 1985-2001 Microsoft Corp.

C:\Documents and Settings\Administrator>devices
S60_2nd_FP2_CW:com.nokia.series60
S60_3rd_MR:com.nokia.s60 - default

C:\Documents and Settings\Administrator>
    
```

图 3.8 列出 SDK 列表

如果想查看计算机上都安装了哪些 Symbian 的 SDK，可在 DOS 窗中输入以下命令：

```
devices
```

这个命令的执行结果就是列出已经安装的 SDK 列表, 在 SDK 列表中, 后面带 default 参数指示的那个 SDK, 就是系统当前缺省使用的 Symbian 的开发环境, 图 3.8 列出的列表中显示 S60_3rd_MR 是缺省使用的开发 Symbian 开发环境。

如果想切换到另外一个开发环境, 在 DOS 窗口中输入下面的命令:

```
devices -setdefault @S60_2nd_FP2_CW:com.nokia.series60
```

这个命令会把 Symbian 开发环境切换到 S60_2nd_FP2_CW, 命令参数 “@” 后为指定要切换到的目标环境平台。

2. 命令行编译

要在命令行下编译一个 Symbian 项目, 首先选择程序对应的正确的 Symbian 开发环境, 如果开发用的计算机只安装有一个 Symbian 开发环境, 就不涉及环境切换的问题了。在 DOS 窗口中进入程序的 Group 目录, 如图 3.9 所示。

```
C:\workspace\Exam0303NewGui\group 的目录
2008-07-26 11:29 <DIR>      .
2008-07-26 11:29 <DIR>      ..
2008-07-26 11:29             1,379 Exam0303NewGui.nmp
2008-07-26 11:29             494 bld.inf
2008-07-26 11:29             1,474 Icons_scalable_dc.mk
                3 个文件          3,347 字节
                2 个目录 14,085,488,640 可用字节
C:\workspace\Exam0303NewGui\group>
```

图 3.9 group 目录内容

在图 3.9 中还有 group 目录列表, 在这个目录中共有 3 个文件, 都是由向导默认生成的, 这里先记下。在使用命令行进行编译时, 需要输入的的第一个命令如下:

```
bldmake bldfiles
```

执行这个命令后, group 目录的列表如图 3.10 所示。

```
C:\workspace\Exam0303NewGui\group 的目录
2008-07-26 11:29 <DIR>      .
2008-07-26 11:29 <DIR>      ..
2008-07-26 11:29             1,379 Exam0303NewGui.nmp
2008-07-26 11:29             494 bld.inf
2008-07-26 11:29             1,474 Icons_scalable_dc.mk
2008-07-26 17:12             322 ABLD.BAT
                4 个文件          3,669 字节
                2 个目录 14,108,246,016 可用字节
C:\workspace\Exam0303NewGui\group>
```

图 3.10 bldmake 后 group 目录内容

图 3.10 所示的 group 目录列表中多了一个批处理文件 abld.bat, 这个文件是编译命令 bldmake bldfiles 根据当前的环境生成的, Symbian 项目的工程文件是与具体的目标平台无关的, 且与具体的 IDE 环境无关, 项目编译成什么目标平台, 由项目开发者在编译时决定, 在后续的编译中, 都要使用这个新生成的批处理命令 abld.bat 来完成。

从 Symbian OS 9.1 开始, 常用的目标平台主要有 3 种: WINSW、GCCE 和 ARMV5。WINSW 主要用于在模拟器上运行和调试程序。通过 GCCE 编译的程序可以运行在目标手机上, 但其编译出的可执行程序比较大且执行效率不是很高。ARMV5 编译器是 ARM 公司的新版编译器, 在 S60 3rd 的 SDK 中没有集成, 需要付费才能获得, 编译出的程序在目标手机上的执行性能非常高。

如果想要编译程序在模拟器上运行, 在 DOS 窗口的 group 下输入下面的命令后按回车键:

```
abld build winsw udeb
```

这个命令中的参数 `build` 表示编译程序所有，包括资源、代码等，`WINSW` 表示编译成 `WINSW` 平台的目标代码，主要运行在模拟器上，`udeb` 表示编译成 Unicode 编码的 debug 模式。

使用这个命令编译完成后，就可以在模拟器上运行程序了，在命令行启动模拟器，输入下面的命令后按回车键：

```
epoc
```

这个命令会启动模拟器，在模拟器的功能选择窗口找到 `Installation` 文件夹，按模拟器的确认键进入文件夹，会看到刚才编译成功的程序 `Exam0303NewGui` 的图标，执行它，第一个程序就可以运行起来了。

如果要使程序在目标手机上运行，在 `DOS` 窗口的 `group` 下输入下面的命令后按回车键：

```
abld build gcce urel
```

这个命令中的参数“`gcce`”表示使用 `GCCE` 编译器把程序编译成可在手机上执行的编译，如果在计算机上安装有 `ARMV5` 的编译器，只要把“`gcce`”替换成“`armv5`”就可以使用 `ARMV5` 编译器进行编译了，参数“`urel`”表示编译成 Unicode 编码的 `release` 模式。

这样编译完成后，使用 `makesis` 打包程序，然后就可以安装到手机上运行了。打包的相关内容会在本章后面进行介绍。

3.4.2 IDE 环境编译

现在有了 `Carbide.c++` 这个专门开发 `Symbian` 程序的 IDE 环境，使用 IDE 环境创建和编译就相对变得简单了。

1. 为模拟器编译

使用 `Carbide.c++1.3` 的新建工程向导创建程序后，在 `Carbide.c++1.3` 的“`Project Explorer`”窗口中选择想要编译的项目名称，右键单击选择项目，在弹出的快捷菜单中选择“`Build Project`”命令，进行项目的编译。对于刚刚建立的 `Symbian` 程序，默认的配置是编译 `WINSW` 平台的 `udeb` 版本，也就是编译完成后运行在模拟器上。

如果想使用 `Carbide.c++1.3` 打开一个已经存在的 `Symbian` 项目，打开 `Carbide.c++1.3`，选择“`File`”菜单中的“`Import`”菜单命令，在打开的“`Import`”项目向导中选择导入 `Symbian OS` 下的“`Symbian OS Bld.inf file`”，后面的过程，根据向导提示完成就可以把一个程序导入到 `Carbide.c++1.3` 中进行编辑和编译了。

2. 为目标手机编译

如想使用 `Carbide.c++1.3` 把程序编译成目标手机可运行的程序，首先用右键单击选择项目，在弹出的快捷菜单中选择“`Build Configurations`” → “`Set Active`” → “`Phone Release(GCCE) [S60_3RD_MR]`”，然后像上面一样执行“`Build Project`”，对程序进行编译。

3.5 编辑代码

到本节，程序的框架已经搭建好了，并且已经编译通过，能够在模拟器上运行，也可以在目标手机上运行，接下来的工作就是在建立好的程序框架中添加实现项目的特定功能的程序代码。

实现一个 `Symbian` 的程序需要编写的代码可能很多，如设计程序的主界面 `view`、定义菜单、创建对话框、收发信息等，具体怎么实现相关的代码，将在后续的章节介绍。

这里给前面建立的应用 `Exam0303NewGui` 添加一些代码，显示当前时间。

如果要在 view 中显示手机的当前时间,需在函数 CExam0303NewGuiAppView::Draw()的尾部加入如下代码:

```
//绘制当前时间
TTime tCur;
tCur.HomeTime();

TBuf<256> timeAsText;
_LIT ( KTimeFormat, "%D%M%Y%/0%1%/1%2%/2%3%/3 %H:%T:%S " );
tCur.FormatL( timeAsText, KTimeFormat );

gc.UseFont( CEikonEnv::Static()->DenseFont());
gc.DrawText(timeAsText,TPoint(10,40));
gc.DiscardFont();
```

加入这行代码后,在模拟器上执行代码,程序的界面中即能显示时间,但时间不会变化,为了能让时间变化起来,还需要给程序中加入一个定时器,让这个定时器每 1s 产生一个超时事件。在对超时事件进行处理时,刷新一下程序界面,这样时间就会有变化了。

在 Carbide.c++ 中加入一个定时器类,右键单击左侧“Project Explorer”中项目名称,在弹出的快捷菜单中选择“New”,在“New”的子菜单中选择“Symbian OS C++ Class”,在弹出的对话框中选择要添加类的项目,单击“Next”按钮,在新弹出的“Choose a template”对话框中选择“Active Object Class”,单击“Next”按钮后,输入要添加的定时器类的名称“UiRefreshTimer”,最后单击“Finish”按钮,添加定时器类完成。添加该类完成后,首先给在这个定时器类的头文件中定义侦听器类 Observer,代码如下:

```
class MUiRefreshTimerObserver
{
public:
    virtual void NotifyRefresh()=0;
};
```

然后,给定时器类添加一个下面的方法:

```
void CUiRefreshTimer::SetObserver(MUiRefreshTimerObserver* pObserver)
{
    m_pObserver=pObserver;
}
```

使用这个定时器类 CUiRefreshTimer 的其他类需要实现定时器类的接口类 MUiRefreshTimerObserver,这样,当定时器定时的时间到后,定时器会回调 Observer 接口类的 NotifyRefresh()方法通知调用者有定时事件产生,调用者可以在这个处理中做跟自己的业务有关的处理。

在 Exam0303NewGui 中,定时器事件主要用来通知视图窗口刷新界面上的时间,因此,CExam0303NewGuiAppView 需要实现 MuiRefreshTimerObserver 接口类,代码如下:

```
class CExam0303NewGuiAppView : public CCoeControl,
                               public MuiRefreshTimerObserver
{
public:
private:
    void NotifyRefresh();

private:
    CUiRefreshTimer* m_pRefreshTimer;
}
```

实现的接口类的方法,代码如下:

```
void CExam0303NewGuiAppView::NotifyRefresh()
{
    DrawDeferred();
}
```

每当有定时事件产生的时候，就通过 View 的 DrawDeferred()方法发送一个刷新界面请求，系统框架就会调用 View 的 Draw()方法重绘界面。

添加完上面的这些代码，一个显示当前时间的应用就完成了，利用 3.4 节介绍的方法，编译、运行它。

3.6 打包发布

当建立好 Symbian 项目，并且实现了需要的程序功能后，在模拟器上运行测试也没有问题了，这时候，就需要编译成能在目标手机平台运行的目标代码，如使用 GCCE 或 ARMV5 编译器进行编译，然后制作程序的安装包，安装到手机上运行。

3.6.1 打包文件.pkg

一个 Symbian 程序开发完成后，是要安装到手机上运行的，这时候就需要制作一个安装程序 sis 文件，通过 sis 文件才能将程序安装到手机上运行。

制作在 Symbian 手机上运行的安装程序，首先需要有一个制作安装程序描述文件*.pkg，在这个*.pkg 文件中，主要存储着有哪些文件需要被打包到安装程序 sis 文件中，而且还要指明将来在安装时，这些文件安装到手机的什么目录下。

通过 Carbide.c++1.3 的向导生成的 Symbian 工程，向导默认会产生一个 pkg 文件，这个 pkg 文件就存储在 sis 目录中。

3.6.2 制作 Symbian 安装程序 sis 文件

有了上面的 pkg 文件后，制作安装就比较轻松了。制作 sis 文件有两种方式：命令行和 IDE 环境中创建。

1. 通过命令制作安装程序

使用命令行的方式制作安装程序，比较简洁明了，需要做什么工作就在 DOS 命令窗口输入什么命令就可以了。首先要打开一个 DOS 窗口，进入到要制作安装程序的 Symbian 项目的 sis 目录中，如图 3.11 所示。

```
C:\WINDOWS\system32\cmd.exe
C:\>\cd C:\workspace\Exam0303NewGui\sis
C:\workspace\Exam0303NewGui\sis>makesis Exam0303NewGui_S60_3_X_v1_0_0.pkg
Processing Exam0303NewGui_S60_3_X_v1_0_0.pkg...
Created Exam0303NewGui_S60_3_X_v1_0_0.SIS
C:\workspace\Exam0303NewGui\sis>
```

图 3.11 制作安装程序

在开始执行 makesis 命令之前，先检查一下 Exam0303NewGui_S60_3_X_v1_0_0.pkg 这个 pkg 文件，打开这个文件，发现这个文件中有 3 个以 \$ 号开头的环境变量，这是 Carbide.c++1.3 的新建向导中生成的，在 IDE 环境中编译时可以使用，由于我们使用的 DOS 环境中没有设置这样的环境变量，因此需要把这几个环境变量修改为实际的值，如下：

```
$(EPOCROOT)修改为\Symbian\9.1\S60_3rd_MR\，以 SDK 的安装目录为准；
$(PLATFORM)修改为 gcce，如果使用 ARMV5 编译器，就修改为 armv5；
$(TARGET)修改为 urel。
```

对 pkg 文件中的环境变量修改完之后，在 DOS 的命令行中输入如下命令并按回车：

```
makesis Exam0303NewGui_S60_3_X_v_1_0_0.pkg
```

makesis 是 SDK 中自带的制作安装程序的命令，命令所使用的参数就是打包描述文件 pkg，执行后生成程序的安装文件 Exam0303NewGui_S60_3_X_v_1_0_0.SIS 存储在 sis 目录中。

使用以上命令制作的 S60 2nd 版手机安装程序 sis 文件，可以直接在手机上安装运行；使用以上命令制作的 S60 3rd 版手机的安装程序 sis 文件，还需要对 sis 文件进行签名才可以在手机上安装运行，sis 签名相关的内容将在 3.6.3 节介绍。

2. 通过 IDE 环境制作安装程序

使用 Carbide.c++1.3 开发 Symbian 项目，可以比较容易地在 IDE 环境下制作安装程序。在新建或导入一个 Symbian 项目后，使用右键单击项目的名称，在弹出快捷菜单中选择最下面的“Properties”，在打开的“Properties for Exam0303NewGui”对话框中，选择左侧树视图中的“Carbide.c++”，展开后再选择下面的“Carbide Build Configuration”，然后在右侧的“Active Configuration”中的下拉式组合列表框中选择“Phone Release(GCCE)[S60_3rd_MR]”，再选择下面的标签页“SIS Builder”，如果下面的列表是空的，就单击右侧的“Add”按钮，添加一个 SIS Builder，这时会打开“SIS Properties”的对话框，如图 3.12 所示。

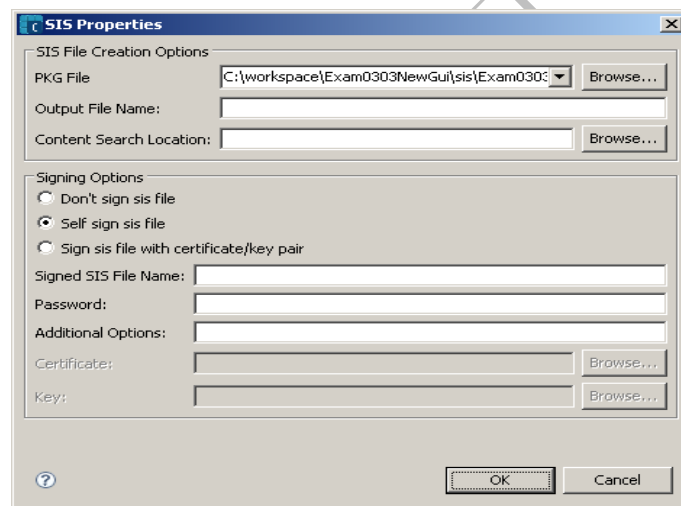


图 3.12 设置项目的 SIS 属性

这个对话框主要用于选择要使用的 pkg 文件，签名选项处根据需要进行设置。如果是 S60 2nd 版的程序或不想对 S60 3rd 版的程序进行签名，那么可以选择“Don't sign sis file”；下面的两个选项都是为 S60 3rd 版的程序进行设置的，如果选择了“Self sign sis file”，在编译时，Carbide.c++会在生成 sis 后，自动地生成一个证书对程序签名；如果选择了“Sign sis file with certificate/key pair”，就需要用户自己指定签名证书。通过上面的对话框设置，以后在使用 Carbide.c++为目标手机进行编译时，编译完成后，就可以直接在 sis 文件中找到安装程序了。

3.6.3 S60 3rd 版安装程序签名

从 Symbian OS 9.1 开始，在 OS 中引入了新的安全模型，更改了工程组织、目录权限、API 能力限制等，增强了程序的安全性。

Symbian 从以下几个方面增强了程序的安全性。

- (1) TCB 平台安全性的核心组件。
- (2) TCE 保护平台的敏感 API。
- (3) 能力访问敏感 API 的权限。

- (4) 数据锁定安全的本地存储。
- (5) SID 与 VID 为软件和厂商提供的安全 ID。
- (6) 边界与策略安全的 IPC。
- (7) 软件安装程序设备上仅允许受信任的软件运行。

从 Symbian OS 9.1 开始，对应的 S60 SDK 是第三版，引入了对 API 的能力限制，能力是授予访问敏感 API 权限的令牌。如果在程序中需要 Symbian OS 中的特定功能的 API，则需要在程序的工程组织文件中声明相应的能力，在制作安装程序时也要使用具有同样能力的证书进行签名。

Symbian 将 API 分为 4 类，如图 3.13 所示。60% 无限制的 API 和用户在安装期间可以授权的 API 可以满足大多数的应用，这类程序使用自签名证书就可以对安装程序进行签名，签名后的安装程序就可以在手机上安装、测试和发布应用。自签名的证书可以使用 SDK 提供的 makekeys 工具生成自签名证书。

无限制	用户可以授权 (在安装期间)	Symbian Signed	许可获得者/平台 批准
(60%的API)	ReadUserData WriteUserData NetworkServices LocalServices UserEnvironment	用户可以授权的 能力+ 声明: Location ReadDeviceData WriteDeviceData PowerMgmt SurroundingsDD ProtServ TrustedUI SwEvent	Symbian Signed的 能力+ DRM TCB AllFiles CommDD DiskAdmin MultiMediaDD NetworkControl

图 3.13 Symbian API 能力分类

如果程序使用了 Symbian Signed 能力授权的 API，开发过程中，需要从 www.symbiansigned.com 的网站上申请开发者证书进行签名，然后才可以安装到手机上进行测试。申请的开发者证书是和手机绑定的，在开发完成后，要在 [symbiansigned](http://symbiansigned.com) 的网站进行签名才可以发布并安装到任何目标手机上。

对生成的 sis 文件进行签名，因为还没有自签名证书，可以先使用下面的命令生成一个自签名证书：

```
makekeys -cert -password 123456 -len 2048 -dname "CN=myname OU=Development OR=symbian CO=GBEM=noone@nowhere.com" mykey.key mycert.cer
```

生成证书后，要给已经生成的 sis 文件进行签名，在命令行输入下面的命令后按回车键：

```
signsis Exam0303NewGui_S60_3_X_v_1_0_0.SIS Exam0303NewGu.SISX mycert.cer mykey.key 123456
```

或者使用类似下面的命令格式：

```
signsis Exam0303NewGui_S60_3_X_v_1_0_0.SIS Exam0303NewGu.SIS mycert.cer mykey.key 123456
```

该命令执行后，就会生成可以在 S60 3rd 版手机上安装的程序 Exam0303NewGu.SISX，只要是签了名的 sis，扩展名是 sis 或 sisx 都可以，在安装时，手机的安装程序能识别出来，使用 3.6.4 介绍的方法安装到手机上就可以运行了。

3.6.4 在设备上运行程序

把已经签了名的安装程序包 sisx 或 sis 安装到手机上有很多种方式，可以通过数据线、蓝牙、红外、读卡器等多种方式。

1. 通过 Nokia pc 套件软件安装

利用这种方式，开发者的计算机要安装 Nokia 的 PC 套件软件，PC 套件软件在 Nokia S60 手机的随机光盘上有，或者可以直接从 www.nokia.com 网站下载该软件，然后，把该软件安装到计算机上。

在安装程序之前，首先让手机和计算机上 PC 套件连接上，可以使用数据线、蓝牙、红外进行连接，手机和 PC 套件连接好之后，在计算机上的 Exam0303NewGui 工程的 sis 文件中，直接用鼠标双击签好名的 Exam0303NewGu.SISX 文件，按照提示，就可以把程序安装到手机上了。

2. 在手机上执行安装

通过这种方式在手机上安装程序，不用在开发者的计算机上安装 PC 套件，要做的就是通过某种途径把 sisx 文件复制或发送到手机上，就可以完成安装了。

通过蓝牙或红外的文件传送功能，把 sisx 文件发送到手机上，然后在手机的消息收件箱中，打开刚接到的蓝牙或红外消息，根据提示进行安装。

或者，使用数据线，把手机和计算机连接上，在手机上选择数据传送模式后，手机进入离线模式，这时计算机会把手机映射成一个磁盘，把 sisx 文件复制到该磁盘内的任何一个目录内，复制完成后，拔下数据线使手机进入正常模式，然后打开手机上的文件管理器，找到手机磁盘上的刚复制进来的 sisx，选中这个文件，按手机的确认键，开始安装程序。

程序安装后，就可以在手机上运行了，这样，第一个程序就开发完成了，虽说是个简单的文字时钟程序，但其功能完整。

联系方式

集团官网: www.hqyj.com

嵌入式学院: www.embedu.org

移动互联网学院: www.3g-edu.org

企业学院: www.farsight.com.cn

物联网学院: www.topsight.cn

研发中心: dev.hqyj.com

集团总部地址: 北京市海淀区西三旗悦秀路北京明园大学校内 华清远见教育集团

北京地址: 北京市海淀区西三旗悦秀路北京明园大学校区, 电话: 010-82600386/5

上海地址: 上海市徐汇区漕溪路 250 号银海大厦 11 层 B 区, 电话: 021-54485127

深圳地址: 深圳市龙华新区人民北路美丽 AAA 大厦 15 层, 电话: 0755-25590506

成都地址: 成都市武侯区科华北路 99 号科华大厦 6 层, 电话: 028-85405115

南京地址: 南京市白下区汉中路 185 号鸿运大厦 10 层, 电话: 025-86551900

武汉地址: 武汉市工程大学卓刀泉校区科技孵化器大楼 8 层, 电话: 027-87804688

西安地址: 西安市高新区高新一路 12 号创业大厦 D3 楼 5 层, 电话: 029-68785218

广州地址: 广州市天河区中山大道 268 号天河广场 3 层, 电话: 020-28916067