第 10 章 Symbian 开发平台

Symbian 是一个移动操作系统的核心,它定义了一些接口,在此基础上可以创建用户接口。 常用的用户接口是 Nokia 的 S60 和 UIQ,其中 S60 的市场占有率较高,因为 Nokia 的重点在 S60。本章主要讲解 Symbian OS 开发环境的搭建,只有把环境搭建好了,才能继续往下学习, 这是本章的重点。阅读本章几乎不需要任何相关的知识,从未接触过移动设备程序开发的读 者也能快速入门。

10.1 S60 开发环境的搭建

目前,S60是比较热门的话题。诺基亚的产品很多都搭载了S60用户界面,从市场占有率来看,它比Symbian用户界面有很大的优势,一方面是自身的优点所致,另一方面,由于诺基亚的主要对象也选择了S60,也让这个优秀的系统一跃成为市场占有率最高的移动操作系统。而目前世界消费电子产业正在逐渐迈进移动化,很多事务都可以通过手机以及功能比手机更强的PDA设备来完成。对于经常出差的商业人士来说,这给他们提供了很大的方便,现在有笔记本电脑可以随身携带,但其体积还是比较大。在这方面PDA具有很大的优势,它可以很方便地随时处理文档、收发邮件、即时聊天等。S60作为目前移动平台上市场占有率最大的手机系统,开发S60应用程序也会成为将来人生的一项技能。接下来从S60最基本的讲起,如何搭建好S60的开发环境,安装过程中应该注意些什么,并逐步进入程序开发讲解。

在安装 SDK 之前,应该检查自己的硬件和软件是否达到了要求,这些要求只是针对运行 SDK 而言的,如果需要一个集成开发环境,还得检查是否满足集成开发环境的最小要求。

硬件要求如下:

- 1GHz 的处理器 (最小的要求是 500MHz)。
- 512MB RAM (最少的要求是 128M 内存)。
- 至少支持 16 位颜色的显示器。
- 1024*768 的显示分辨率。
- 1GB 的空闲硬盘空间,如果项目比较多,可能需要更多硬盘空间。
- 如果要开发多媒体程序,一个 Windows 兼容声卡是必需的。

相对于硬件来说,软件要求比较容易满足,只要在安装 SDK 之前下载相应的软件,并按 照正确的步骤安装相应的软件,软件要求就算满足了。在操作系统方面,获得支持的系统有:

- Microsoft Windows XP Professional SP2.
- Microsoft Windows 2000 SP4.

安装 S60 SDK 的软件需求有:

- ActivePerl.
- Java Runtime Environment.

其他需要注意的是,安装过程中应该使用管理员权限,否则,一些环境变量不能被正确 设置,并且,SDK 必须被安装到一个不包含空格的目录中去。这对程序员来说应该比较清楚, 在某些情况下一个路径中包含空格会出现问题,建议尽量不要这样使用。

开发程序的过程中也应该注意, SDK 项目文件应当放到跟编译工具相同的一个逻辑分区中。所谓逻辑分区就是把扩展分区细化而划分出来的块,在 Windows 中就是除系统分区以外的分区。

如果需要把 SDK 和集成开发环境一起来开发应用程序,则集成开发环境应该和 SDK 处 在同一个逻辑磁盘,并且也需要系统管理员权限。被支持的集成开发环境有:

- Carbide.C++。
- Carbide.vs。
- CodeWarrior Development Studio for Symbian OS 3.1.

当硬件要求满足以后就可以进行软件环境的搭建了,下面以 Symbian C++开发环境为例,介绍开发环境的搭建过程,其中要用到的工具如下:

- ActivePerl-5.6.1.631。
- JDK-1.5.0.17。
- S60 C++ SDK-3.1。
- Carbide C++-2.0.

其中 ActivePerl 的安装尽量选 5.6.1.x 这个版本,在这一点上 S60 SDK 比较挑剔。在 http://downloads.activestate.com/ActivePerl/Windows/中能找到想要的版本,下载连接如下: http://downloads.activestate.com/ActivePerl/Windows/5.6/ActivePerl-5.6.1.631-MSWin32-x86.zip。

安装了 ActivePerl 后还需要安装 Java 的运行环境。最低的版本要求是 1.4.2_02, 这里选择 的版本是 1.5.0.17。学习过 Java 的程序员对 JDK 并不陌生, 但是, 非 Java 的程序员可能没有 接触过。JDK 是 Java Development Kit 首字母的缩写, 它是 Java 的 SDK, 其中包含了 Java 的 编译器、调试器、打包工具和文档工具, 另外还包含了完整的 JRE (Java Rumtime Environment) 和用于产品环境的各种库类, 以及给开发人员使用的补充库。JDK 还包含了很多程序例子用 来展示 Java API 中的各个部分。JDK 可以从 SUN Microsystem 的网站上下载, 它是免费提供 给程序员使用的。以下详细讲解安装 SDK 之前的软件配置过程。

10.2 安装 ActivePerl

双击"ActivePerl-5.6.1.631.exe"启动安装程序,如图 10-1 所示。

单击"Next"进入版权许可页面,如图 10-2 所示。

选择接受用户端使用协议,否则不能继续安装,接受协议以后单击"Next"将进入到目录和安装选项的设置,这也是最关键的部分。如图 10-3 所示。

把 ActivePerl 安装到 E:\Perl\目录下,再单击"Next"进入 PPM 设置页面,如图 10-4 所示。 对于这一项不用过多的关注,直接单击"Next"按钮,进行下一步就可以了。其中可以看 到 ActivePerl 的隐私策略。单击"Next"按钮进入环境设置页面,如图 10-5 所示。



图 10-1 ActivePerl 安装界面示意图

🙀 ActivePerl 5.6.1 Build 631 License Agreement 🛛 🔀
End-User License Agreement Please read the following license agreement carefult
ActiveState Community License
Preamble:
The intent of this document is to state the conditions under which the Package (ActivePerl) may be copied and distributed, such that ActiveState maintains control over the development and distribution of the Package, while allowing the users of the Package to use the Package in a variety of ways. The Package may contain software covered by the Artistic License. The Interview of the terms in the License Agreement
< <u>B</u> ack <u>N</u> ext > Cancel

图 10-2 ActivePerl 许可协议示意图



图 10-3 安装选项示意图



图 10-4 设置 PPM 示意图

Choos Cho	e Setup Options ose optional setup action	ns.		
	Add Peri to the	PATH environ	ment variable	
	Create IIS sori	ipt mapping fo	r Perl	
	🗖 Create IIS scri	pt mapping fo	r Peri ISAPI	
		< <u>B</u>	ack Next >	Cancel

图 10-5 环境设置示意图

选择把 Perl 目录加到 PATH 环境变量中,这也将是 SDK 找到 Perl 工具的方式,如果不选择加到 PATH 环境变量,也可以在安装完成以后自己添加到 PATH 中去。单击"Next"按钮进入最后的安装页面,如图 10-6 所示。

这是最后一次更改安装选项的机会,如果没有更改,可以直接单击"Install"按钮进入安装过程,如图 10-7 所示。

几分钟以后,如果安装过程没有出现任何差错,ActivePerl 安装完成,如图 10-8 所示。



图 10-6 准备安装示意图

ActivePerl 5.6.1 Build 631 Setup			
Installing ActivePerl 5.6.1 Build 6	31		
Please wait while the Setup Wiza This may take several minutes.	ard installs A	ctivePerl 5.6.1	Build 631.
Status:		1	
	< <u>B</u> ack	<u>N</u> ext >	Cancel

图 10-7 开始安装示意图



图 10-8 安装完成示意图

10.3 安装 JDK

接下来安装 JDK,双击"JDK"安装程序,启动安装过程,如图 10-9 所示。



图 10-9 安装界面示意图

接下来, JDK 安装程序进入 JDK 许可证协议界面, 在界面中用户必须接受 JDK 的使用授权, 否则不能继续安装。如图 10-10 所示。

d J2SE Development Kit 5.0 Update 17 - 许可证	×
许可证协议 请仔细阅读下面的许可证协议。	un.
Sun Microsystems, Inc. Binary Code License Agreement	-
for the JAVA 2 PLATFORM STANDARD EDITION DEVELOPMENT KIT 5.0	
SUN MICROSYSTEMS, INC. ("SUN") IS WILLING TO LICENSE THE SOFTWARE IDENTIFIED BELOW TO YOU ONLY UPON THE CONDITION THAT YOU ACCEPT ALL OF THE TERMS CONTAINED IN THIS BINARY CODE LICENSE AGREEMENT AND SUPPLEMENTAL LICENSE TERMS (COLLECTIVELY "AGREEMENT"). PLEASE READ THE AGREEMENT CAREFULLY. BY DOWNLOADING OR INSTALLING THIS SOFTWARE, YOU ACCEPT THE TERMS OF THE AGREEMENT. INDICATE ACCEPTANCE BY SELECTING THE "ACCEPT" BUTTON AT THE BOTTOM OF THE AGREEMENT. IF YOU ARE NOT WILLING TO BE BOUND BY ALL THE TERMS, SELECT THE "DECLINE" BUTTON AT THE BOTTOM OF THE AGREEMENT AND THE	Ŧ
 ● 我接受该许可证协议中的条款(A) ● 我不接受该许可证协议中的条款(D) 	
instal Shield 下一步(N) >取消	

图 10-10 JDK 许可证协议示意图

选择"接受协议",单击"下一步"按钮,进入目录设置和其他安装选项设置页面,如图 10-11 所示。

🕵 J2SE Development Kit 5.0 Update 17 - 🗎	定义安装
自定义安装 选择要安装的程序功能。	Sun.
诸从下面的列表中选择要安装的可选功能。安 册除程序"实用程序来更改您选择的功能。 开发工具 通▼ 演示程序 通▼ 次共 IRE	装完成后,您可以使用"控制面板"中的"添加/ 功能说明 独立 JRE 5.0。任何应用程序均 可以使用此 JRE。此 JRE 向浏 览器和系统注册 Java 插件和 Java Web Start。可以单独格其 从 JDK 中卸载。它需要 161 MB 的硬盘驱动器空间。
」 安装到: C:\Program Files\Java\jdk1.5.0_17\ InstallShield <上	

图 10-11 选项设置示意图

这里将 JDK 安装到默认的位置,并且安装公共 JRE。单击"下一步"按钮进入正式安装 过程,安装程序开始复制文件,并进行一些设置,如图 10-12 所示。

<mark>伊</mark> J2SE Dev 正在安装 正在安装	elopment Kit 5.0 Update 17 - 进度
1 .	安装向导正在安装 125E Development Kit 5.0 Update 17,请稍候。这可能需要持续几分钟时间。 状态:
InstallShield –	< 上一步(B) 下一步(B) >

图 10-12 开始安装示意图

接下来,安装 JRE,如图 10-13 所示。安装过程比较简单。 选择默认的目录安装 JRE。单击"下一步"按钮,进入插件安装界面,如图 10-14 所示。 安装插件可以让浏览器获得对 Java 应用程序的支持,根据需要选择相应的插件安装。再 单击"下一步"按钮,进入安装过程,如图 10-15 所示。

152



图 10-13 安装 JRE 示意图

뤻 J2SE Runtime Environment 5.0 Update 17 - 浏览器注册	×
浏览器往册 请选择要注册 Java(TM) 插件的浏览器。	Sun.
Microsoft Internet Explorer	
Mozilla 和 Netscape	
您可以以后在 Java(TM) 控制面板中更改设置。	
InstaliShield <u> く上一步(B)</u> 「 <u>下一步(N)</u> >	取消

图 10-14 插件安装示意图

👘 J2SE Runt	time Environment 5.0 Update 17 - 进度	
正在安装 正在安装	5您选择的程序功能。	& <u>Sun</u> .
19	安裝向导正在安裝 125E Runtime Environment 5.0 Update 17,请稍候。 这可能需要持续几分钟时间。 状态:]
InstallShield —	<上一步(1) 下一步(1) >	

图 10-15 开始安装示意图



等待一段时间后, JDK 安装完成。如图 10-16 所示。

图 10-16 安装完成示意图

10.4 安装 Carbide C++

至此为止,安装 SDK 前的最基本准备工作就算完成了。但由于选择了以 C++作为主要语言,安装一个 C++的集成开发环境有助于提高开发效率,这里选择的是 Carbide C++。它是由 一系列 C 语言和 C++组成的,用于开发 Symbian OS 程序的工具集合。它基于 Eclipse 集成开发环境和 C/C++开发工具,也就是常说的 CDT。它包含了 Eclipse 开发环境的优点,并且免费提供给开发者使用。这个工具可以在 Nokia Forum 下载。

双击"carbide.c++-2.0.exe"启动安装向导,如图 10-17 所示。



图 10-17 安装向导示意图

单击"Next"按钮进入用户端软件协议页面,如果不接受协议,不能继续安装,如图 10-18 所示。

Carbide. c++ v2.0 - InstallShield Wizard	×
End-user software agreement Please read the agreement carefully	
NOKIA CORPORATION END-USER SOFTWARE AGREEMENT FOR CARBIDE.C++ AND ASSOCIATED ON-DEVICE APPLICATIONS	
This Software Agreement ("Agreement") is between You (either an individual or an entity), the End User, and Nokia Corporation ("Nokia"). The Agreement authorizes You to use the Software specified in Clause 1 below, which may be stored on a CD-ROM, sent to You by electronic mail, or downloaded from Nokia's Web pages or Servers or from other sources under the terms and conditions set forth below. This is an agreement on end-user rights	
• I accept the terms in the License Agreement	
O I <u>d</u> o not accept the terms in the License Agreement InstallShield	
< <u>B</u> ack <u>N</u> ext > Cancel	

图 10-18 用户端软件许可协议示意图

选择"接受协议"并单击"Next"按钮进入另一个 Nokia 的开源软件许可协议。同样, 不接受这个协议也不能继续安装,如图 10-19 所示。

pen Source Software License Agreem	ent 📁 📢 Ca	rhid
Please read the agreement carefully		Carbide.c
This Nokia Product includes Open Source S	oftware.	
Certain software included in this Nokia Prod General Public License (GPL) and the GNU the GNU Library General Public License) an obligation or permission to provide the sourc executable form delivery of the said software delivered to you in accordance with the refe installation package.	ict is licensed and distributed under GN Lesser General Public License (LGPL, J d/or other copyright licenses containing e code of such software along the bina . The source code of such software is rred license terms and conditions in this	IU a.k.a. j ry/ :same
The exact license terms of GPL and LGPL a the required copyright and other disclaimers,	nd certain other copyright licenses, as unotices, permissions and acknowledge	well as ments, 🖵
The exact license terms of GPL and LGPL a the required copyright and other disclaimers, I accept the terms in the License	nd certain other copyright licenses, as unotices, permissions and acknowledge	well as ements, 🖵 Print
The exact license terms of GPL and LGPL a the required copyright and other disclaimers, I accept the terms in the License I ado not accept the terms in the L	nd certain other copyright licenses, as unotices, permissions and acknowledge Agreement cense Agreement	well as ements, 🖵 Print
The exact license terms of GPL and LGPL a the required copyright and other disclaimers, I accept the terms in the License I do not accept the terms in the L IShield	nd certain other copyright licenses, as un notices, permissions and acknowledge Agreement	well as ements, 💌 Print

图 10-19 Nokia 开源软件协议示意图

选择"接受使用协议"以后单击"Next"按钮进入 Carbide.C++的注意页面,接着介绍了 一些 Carbide C++版本问题,如图 10-20 所示。

bide.c++ v2.0 - InstallShield Wizard	
Information Please read the following text.	Carbide Carbide.c++
Welcome to Carbide.c++ v2.0	
Thank you for installing Carbide.c++.	<u></u>
Carbide.c++ is a free product to all developers. The dev Carbide Edition to install. The free product will come with your selection.	veloper will need to choose which n all the features associated with
Please visit Forum Nokia (www.forum.nokia.com/carbid updates on Carbide.c++.	e_cpp) for the latest news and
	Þ
(anomero -	ick Nevt > Cancel
<u>< B</u> a	ck <u>N</u> ext> Cancel

图 10-20 阅读文字

这个页面可以仔细读一下,为自己选择正确的版本。然后单击"Next"按钮进入版本选择页面。如图 10-21 所示。

Dide. CTT V2: 0 - InstallShield #12ard	
Setup Type Select the setup type that best suits your needs.	Carbide Carbide.c+
Click the type of setup you prefer.	
1) OEM Edition	Description
2) Professional Edition	Enabled features include:
3) Developer Edition	All Professional Features plus:
allShield	Back Next> Cancel

图 10-21 选择版本示意图

在这一页面中应该仔细选择自己想要的版本。这里选择 Professional Edition。单击"Next" 按钮进入安装目录设置页面。如图 10-22 所示。

应该注意, Symbian OS 的目录是从 "/"开始的,也就是 UNIX 中的根目录,它不能用 Windows 下的字母前缀作为路径。在集成开发环境中, S60 SDK 和项目文件需要放到同一个 逻辑磁盘上。因此,这里的目录应当设置为 SDK 将来目录所在的盘,单击 "Next"按钮进入 开始菜单选项设置,如图 10-23 所示。





图 10-22 选择安装路径示意图

图 10-23 创建开始菜单示意图

可以进行一些适当的设定,最后进行安装。如图 10-24 所示。

arbide.c++ v2.0 - InstallShield Wizard	2
Setup Status	Carbide Carbide.c++
The InstallShield Wizard is installing Carbide.c++ v2.0	
Installing	
E:\Carbide.c++\plugins\com.nokia.carbide.oss_1.4.0.014.jar	
	Cancel
	Lancel

图 10-24 开始安装示意图

经过一段时间后这个集成开发环境安装完成。如图 10-25 所示。



图 10-25 安装完成示意图

10.5 安装 S60 SDK

当以上三个软件安装完成以后,现在可以进行 S60 SDK 的安装了。S60 的 SDK 可以从 Nokia 的论坛中下载,其中有很多版本,这里选择了 S60 SDK 第三版第一个功能包的安装程 序,因为它可以兼容过去的 S60 系统。

把下载的安装包解压到一个目录,执行其中的 setup.exe 启动安装向导,如图 10-26 所示。



图 10-26 S60 SDK 安装向导示意图

单击"Next"按钮进入版权声明页面,选择"接受协议",否则不能继续安装。接下来出现安装目录设置页面,这里将 SDK 安装到 E:\Symbian\9.2 中。如图 10-27 所示。

Choose Destination Location Select folder where setup will install files. Setup will install S60 3rd Edition FP1 SDK for Symbian OS in the following folder.
Setup will install S60 3rd Edition FP1 SDK for Symbian OS in the following folder.
To install to this folder, click Next. To install to a different folder, click Browse and select another folder.
Note! Setup will add 'S60_3rd_FP1' after Selected Path.
Destination Folder
E:\Symbian\9.2
tallôhield
< <u>B</u> ack <u>Next</u> > Cancel

图 10-27 选择目录示意图

单击"Next"按钮进行安装,如图 10-28 所示。

) 3rd Edition FP1 SDK for Symbian OS - InstallShield Wizard Setup Status	
S60 3rd Edition FP1 SDK for Symbian OS is configuring your new software install	ation.
Installing core SDK	
E:\Symbian\9.2\S60_3rd_FP1\Epoc32\tools\rommask.exe	
taliShield	
	Cancel

图 10-28 开始安装示意图

在安装过程中有个默认设备的选择,由于这里只安装了 S60 第三版第一个功能包,所以选择第一项,如图 10-29 所示。

选择设备以后,会提示安装工具包,可以按照提示安装,比较简单,如图 10-30 所示。 在工具包安装完成以后,S60 SDK 安装完成,如图 10-31 所示。

移动软件开发技术



图 10-29 选择默认设备示意图



图 10-30 安装 CSL ARM 工具包示意图

S60 3rd Edition FP1 SDK for	Symbian OS - InstallShield Wizard
	InstallShield Wizard Complete The InstallShield Wizard has successfully installed S60 3rd Edition FP1 SDK for Symbian DS. Click Finish to exit the wizard. Miew Release Notes. Click Finish to complete S60 3rd Edition FP1 SDK for Symbian OS Setup.
	K Back Finish Cancel

图 10-31 安装完成示意图

到此为止, S60的开发环境搭建好了。接下来验证一下安装的环境是否可以正常工作。

10.6 S60 程序开发

Symbian 中的应用程序分为两个部分,分别是引擎和用户界面。这样划分有个好处是:用户的核心代码和界面可以分开,方便维护和管理。这里引擎相当于核心,主要进行大量的运算来处理数据,而用户界面则用于处理数据的显示。

Symbian 程序的外观有三种体系结构,跟 MFC 比较起来有相似的地方,这三种体系结构 分别是:

- 传统的 Symbian OS 控制体系结构。
- 基于对话框的体系结构。
- 视图体系结构。

看到这应该能明白为什么说 Symbian 程序的结构跟 MFC 程序结构有点相似了,原因是视 图、对话框等知识点是 MFC 程序员经常接触到的。

S60程序开发的一般步骤为:

- 设计应用程序的主体框架。这一步的主要工作是定义程序的主要功能,也就是首先设计。如果要写打印一个"Hello world!"在屏幕上应该做哪些工作,对于刚进入 S60 开发领域的程序员来说,设计一下应用程序的结构还是很有意义的,虽然这个设计并不一定要写在纸上。
- 在应用程序设计工作完成以后就可以创建一个 S60 的项目,然后开始编码。创建项目 有两种方式,一种方法是利用集成开发环境来创建,另一种方法是利用命令行接口来 创建。利用集成开发环境来创建项目比较简单,在安装了 Carbide C++以后可以很方 便地创建一个 Symbian OS 软件项目。第二种方法对于刚进入 Symbian OS 开发领域的 程序员来说有点繁琐,所以推荐选用第一种方法。
- 程序编码。在 Symbian OS 程序开发中编码是持续时间最长的阶段,当然对于其他项目也是如此。在这一阶段要反复进行编译、调试、运行等工作。
- 测试程序。好的软件必然要经过严格的测试。有句话叫"百密一疏",即使经过周密的思考,也会有想不到的地方。软件开发也因计算机工程的庞大而变得非常复杂,对于小程序的开发也许不存在这种问题,但是,当软件项目规模变大以后,软件的难度会以指数级增长。软件工程的目的就是以工程学原理来管理软件项目,尽量开发出优秀的软件。
- 项目开发的最后一步是发布软件。对于 Symbian 软件来说,应该申请获得一个数字签 名,然后打包,发布软件。

下面用一个简单的例子介绍一下 Symbian OS 程序开发的过程,并且为了让 Symbian OS 开发初学者能够顺利编译运行,尽量对新建的项目少作修改。在现在的集成开发环境中,新建的项目是一个能够运行的程序,并且规模很小,从中可以看出应用程序的主体框架。也就是说通过阅读并且记住集成开发环境,再通过模板产生程序,可以学会简单的程序开发,通过它能够对即将学习的应用程序的结构有所了解。接下来通过 Carbide C++演示一下这个过程。

首先打开 Carbide C++开发环境。通常可以通过开始菜单->所有程序->Carbide.c++

v2.0->Carbide.c++ v2.0 打开 Carbide C++开发环境,也可以进入到安装目录执行 Carbide.c++.2.0.exe 启动开发环境。程序打开后会提示选择一个工作目录,前面讲过这个目录 不能随便选择,应该放到 SDK 等工具安装目录相同的逻辑磁盘下面,如图 10-32 所示。

C Workspace Launcher		×
Select a workspace		
Carbide.c++ stores your projects in a folder called a workspace. Choose a workspace folder to use for this session.		
<u>₩</u> orkspace: <mark>E:\workspace</mark>		<u>B</u> rowse
Use this as the default and do not ask again		
	OK	Cancel

图 10-32 选择工作目录示意图

Carbide C++打开以后,如果是第一次进入一个新的工作目录,会出现一个欢迎界面,可以在其中选择一些帮助文档查看,也可以将其直接关闭。

创建一个新的项目。在 File 菜单下面选择 New,接着就可以选择一个合适的项目模板了,如图 10-33 所示。

New	Alt+Shift+N	🖌 💒 Symbian OS C++ Progec	st
Open Source/Header Open File <u>.</u>	Ctrl+'	Qt Project 😽	
Close Close All Save Save <u>As</u> Sav <u>e</u> All	Ctrl+W Ctrl+Shift+W Ctrl+S Ctrl+S Ctrl+Shift+S	 Source Folder Folder Source File M Header File File from Template 	
Moyer Rename Refresh Conyert Line Delimiter:	F2 F5 s To	C Class Symbian OS C++ Class Symbian OS MMP File 'UQ UI Design S60 UI Design	
Print	Ctrl+P	📑 <u>O</u> ther	Ctrl+
Switch <u>W</u> orkspace Res <u>t</u> art		•	
<u>] I</u> mport Exp <u>o</u> rt			
	12002200000		

图 10-33 新建项目示意图

这里选择 Symbian OS C++ Project,接下来会打开一个对话框,从中可以选择具体的项目 类型,选择第一项 Basic console application (EXE)模板,单击 "Next"进入另一个对话框,在 其中输入项目名字。如图 10-34 所示。

输入项目名称以后单击"Next"进入项目生成配置页面。如图 10-35 所示。

New Basic console	application (EX	E)		
New Symbian OS C Create a new Symbia	++ Project m OS C++ project	L.		
<u>P</u> roject name: ▼ Use <u>d</u> efault loc	ation			
Location: E:\works	pace			Browse
0	< <u>B</u> ack	Next >	Finish	Cancel

图 10-34 输入项目名称示意图

New Basic co	nsole application (EXE)	
Symbian OS S Select the bui	DKS ld configurations for this project.	
SDKs and Build	Configurations	
 	<u>[F]</u> based on selected template	
0	<u> </u>	<u>F</u> inish Cancel

图 10-35 生成配置选项示意图

单击"Next"按钮进入版权和作者设置页面,其中根据相应情况填写,再单击"Next"按钮进入目录设置页面,如图 10-36 所示。

🕜 New Bas	sic console application (EXE)	
Project I Generate the proj	Directories d files will be copied to the specified directories under ect root directory	
Include:	ine	
Source:	src	
Group:	[gr oup	
Data:	data	
Install:	sis	
0	Eack Ment > Finish	Cancel

图 10-36 目录设置示意图

最后单击"Finish"按钮完成项目的创建工作。现在可以看到项目浏览器中有了一个新的项目,通过鼠标来查看各个目录下的文件,了解项目的目录结构,这对于 Symbian OS 软件开发者经验的增长也是有好处的。下面简单介绍一下各个目录包含的文件以及它们的作用。

- data 包含应用程序的资源文件,比如.rls和.rss。资源文件包括本地化字符串等内容。
- gfx 包含可扩展的矢量图形文件,扩展名为.svg。这些图形文件是应用程序需要的。
- group 包含应用程序的组件和项目定义文件,比如 (.inf、.mmp 和.mk),其中.mmp 文件指定了项目的一些特性,这个文件比较重要。
- inc 包含应用程序的头文件,比如.h 和.hrh 文件。
- sis 包含应用程序的软件包定义文件(.pkg),软件包定义文件是一种包含了一些安装 文件工具 makesis 需要的信息。
- src 包含应用程序的源代码文件。

该项目在创建以后本身就是一个可以运行的程序,拥有很短的代码,而且扩展性较强, 在此基础上进一步修改,就可以添加一些其他的功能。其实一般的软件项目也是从新建一个 最基本的项目开始的,在这个基础上添加各种扩展功能,就形成了可以发布的软件。

为生成可执行版本的软件,需要在项目浏览器的根文件夹上右击选择"Build Project",或 者使用快捷键"Ctrl+B"。如果右下方的 Problems 栏中没有出现错误,则项目生成成功。此时 就可以调用 S60 的模拟器运行程序,这可以通过单击在 Run 菜单下的 Run 子菜单项或者单击 工具栏中的"运行"按钮来启动程序。集成开发环境将自动调用模拟器并且把应用程序加载 到模拟器中运行。上述刚刚建立成功的项目自带的程序运行结果如图 10-37 所示。

一般的 SDK 开发包中都提供了很多的源代码示例,这对于刚步入这个开发领域的程序员 来说算是无比的财富。其实也有很多专业的程序员会在其中寻找自己想要功能的代码。这里 就讲一下对 SDK 开发包中程序的利用,如何进行编译、运行、查看效果。只需找到要处理的 实例程序,用 Carbide C++编译就行。在 X:\Symbian\9.2\S60_3rd_FP1\S60Ex 下有一个 Editor 例子,这里选择它来编译。

用 Carbide C++新建一个空的 Symbian 项目,项目名字可以自己命名,接着拷贝 Editor 目录下的所有文件夹到项目文件夹下面,并且接受覆盖提示,这时候就可以直接编译并在模拟器中运行程序。如图 10-38 所示。在观看了效果以后就可以自己查看代码的实现了。

6	Editor	
ዋ 🛶	No.	abc D
Hello, I'm her	e!	
0		F . 14
Options		EXIT
图 10-38	运行结果示意	意图b

[press any key] =

Hellia, worldt

图 10-37 运行结果示意图 a

本书不是一本专门讲解 Symbian OS 开发的书,并且篇幅有限,不能深入讲解 Symbian OS 的开发,笔者仅提供一个入门的指导,如果需要继续学习,可以查找相关资料。接下来讲一下在 S60 上运行的 Python。

10.7 Python for S60 简介

Python 是一种面向对象的程序设计语言,也是一种脚本编程语言,和 Perl 一样直接通过 程序解释执行,而不用像 C 语言一样得用编译器编译连接生成一个可执行程序。这种语言有 一个非常好的特点是语法简捷,结构非常清晰。

Python 的创始人是 Guido Van Rossum。他在 1989 年根据 ABC 语言改编成 Python。Python 在英语中有大蟒的意思,因为 Guido Van Rossum 本人是一个 Monty Python 马戏团的爱好者。 Python 与 Perl 在设计理念迥然不同。Perl 的设计理念是:做一件事不只有一种解决方法。而 Python 的设计理念是:做一件事仅一种解决方法,但是要做好。Perl 代码的缺点是阅读难度大, 但优点是代码量小,即很短的代码能够完成很多问题。有人说 Perl 的难是因为正则表达式的 原因,有过正则表达式编写经验的程序员肯定深有同感。Perl 跟 Python 一样,也是脚本语言, 只需要一个解释器来装载执行就够了。Python 的 Hello world 程序如下:

#!/usr/bin/env python

print "Hello, world!"

Python 的 Hello world 程序和 Perl 的 Hello world 程序很相似,它们都是 UNIX shell 风格的,即注释以#开头。

下面看一个 Python 的例子,它的功能是生成 MD5 串, MD5 是一种加密的算法。

```
# md5.py
import md5
import hashlib
str = 'test string.'
ml = md5.new()
ml.update(str)
dest1 = ml.hexdigest()
m2 = hashlib.md5()
m2.update(str)
dest2 = m2.hexdigest()
print 'source string: ', str
print 'destination string1: ', dest1
print 'destination string2: ', dest2
```

Python 作为一种面向对象的语言,可以用少量的代码完成很多功能,而 C 语言是面向过程的。这里不是指面向对象高级一点。面向对象的语言简化了很多细节的东西,可以用很精简的代码表达更多的意思,使用面向对象的语言时,更多的是关注于项目整体的设计,如何抽象现实中的事物。所以在类创建好以后,可以用比较简单的过程得到结果。但是面向对象也给了程序员很大的束缚,相比之下,面向过程的 C 语言却有更大的自由。

Python for S60 是一个运行在 S60 平台上的 Python 解释程序,于是程序员可以使用 Python 在 S60 的机器上很方便地做更多的事情。要在 S60 平台上使用 Python,只需要安装 一个 sis 安装包就够了,然后就可以编写程序,接着在 S60 手机上运行。这样可以快速构 建 S60 应用程序。程序员只需要知道一些脚本语言的编程知识便能够理解 Python 程序。脚本语言有个好处是不需要编译,直接解释运行就够了,像 DOS 下的批处理程序一样。

10.8 S60 学习方法简介

学习一门新的语言(当然 S60 开发不涉及新的语言),从一无所知到入门,首要的是找一份最简单的代码详细阅读,弄懂其中的每一个标点符号,这也许有点咬文嚼字的味道,但在程序设计这方面的咬文嚼字和文学方面的不同。写文章的时候可以错一些标点符号,因为读者对文章的检查没有编译器对代码的检查严格,这也是众所周知的。对于第一份简单的代码可以背下来,当作以后的一个模板,这个要求可以用来单纯地学习一门语言,然而要学习建立在语言基础之上形成的一些程序开发库,还需要掌握更多的内容。如果要学习使用一个新的程序开发库,比如一些跟图形界面相关的用 C++封装的类库,可能一个简单的例子也需要很多的代码,使用这种库的前提是需要一个基本的程序框架,这个框架什么事情也不干,只是把一个最简单的程序,在这个框架上添加一点代码就可以实现一些功能,这个框架也可以理解为是所有的基于这个库的程序的公共代码,即所有的程序都可以在这个框架上添加代码而形成所得的程序。

国内的大部分程序员都是通过学习 MFC"出身"的,但对于很多人来说,他们眼中所谓 的 MFC 不是真正意义上的 MFC。曾听一些程序员说:"MFC 太好用了,以前学习的 C 语言 根本没法比,以前的 Turbo C 太不好用了,现在用 Visual C++可以拖控件,改几个属性就可以 完成很多事情。而且,只需要在对象名后打一个点号就能把所有可用的方法和属性调出来, 直接选择而不用不断的在手册和编辑器之间来回切换。"也许这也算是 MFC 吸引大多数人的 原因。但是,这些都不是 MFC,MFC 仅是一堆代码(在 Visual C++的安装目录中能找到), MFC 不是所谓的拖控件,MFC 也不提供拖控件的功能。如果有兴趣,可以只安装一个 SDK, 然后在命令行编译调试程序。其实广受称赞的只是 Visual C++的集成开发环境,却被强加给了 MFC。至于在输入"对象名"和"."后弹出来的一些提示,那也是集成开发环境在工作,跟 MFC 一点关系也没有。

下面谈一下 S60 开发的学习方法的问题,就是学习要抓住本质,当然关系得弄清楚,正 如上面所讲到的 MFC 与集成开发环境的关系,它们的异同。学习要抓住本质指的是不能只看 表面的东西,这个解释有些抽象,还是以上面的 MFC 的例子来说明。很多人学习 MFC 的时 候,一般都是用 Visual C++新建一个 MFC 工程,然后在其中进行修改。初学者(包括一些已 经入门很久的程序员)可能很少打开那些源代码文件去看看,集成开发环境生成的代码的功 能是什么,大部分人只是往其中添加代码和所谓的拖控件。当然专业程序员对此很了解,就 完全没有必要去看那些代码。不过对于初学者来说完全有必要去研究这段集成开发环境生成 的代码,这些代码应该可以算是一个很小的应用程序框架,当然还不算最小的。这段代码的 集成开发环境的提供商编写的代码,在很大程度上,如果不修改这段代码,那它就是大部分 应用程序的框架。对于 S60 的开发也是一样,用 Carbide C++新建的项目最开始也会生成一段 代码,初学者就研究这个,把这个程序研究透时,也就抓住了 S60 开发的本质,至少框架是记下来了。

10.9 本章小结

本章详细介绍了 Symbian 开发平台的相关知识,主要包括 S60 开发环境的搭建,安装 ActivePerl 的主要步骤,安装 JDK 的主要步骤,安装 Carbide C++的主要步骤,安装 S60 SDK 的主要步骤,S60 程序开发过程,Python for S60 简介以及 S60 学习方法简介等。

10.10 习题

1. 搭建 S60 开发环境的软、硬件要求有哪些? 需要安装的集成开发环境有哪些?

- 2. 分别简述安装 ActivePerl、JDK、Carbide C++和 S60 SDK 的主要步骤有哪些?
- 3. S60 程序开发的一般步骤有哪些? 简述其开发过程。

4. 简述 Symbian OS 程序开发过程中,当一个新的项目创建完成后,新项目中每个目录 包含哪些文件以及它们的作用有哪些?

5. Python 与 Perl 相比较各有哪些优缺点?