

---

# S60 平台：常见问题解答

版本 1.7

2007 年 1 月 31 日

S60 platform

## 法律提示

版权©（2005 年，2006 年）属于诺基亚公司，诺基亚公司保留全部权利。

“诺基亚”、“诺基亚论坛”、“诺基亚 6600”、“诺基亚 7650”、“诺基亚 7710”、及“诺基亚 9200”是诺基亚公司的商标或注册商标。Java 和所有基于 Java 的标志是 Sun 微系统有限公司的商标或注册商标。在此提到的其它产品和公司名称可能是其所有者的商标或商业名称。

### 声明

本文档中的信息基于其现有状况，不存在任何保证，包括销售保证、适用某一特殊用途的保证，或从任何建议、规范或范例中衍生出来的保证。此外，本文档所提供的并非最终信息，在其最终发布前会做较大改动。本文档仅用作信息通报。

诺基亚公司不承担所有因实施本文档中所表述的信息而产生的相关责任，包括侵犯任何知识产权的责任。诺基亚公司并不保证或认为使用这些信息不会构成对这些知识产权的侵犯。

诺基亚公司保留不预先通知而随时修改或撤销本规范的权力。

### 授权许可

本授权仅限于因个人应用而下载和打印本说明，除此之外，不存在对其它任何知识产权的授权许可。

## 目录

<b>1.</b>	<b>一些重要概念</b> .....	<b>7</b>
1.1	S60 平台是什么? .....	7
1.2	S60 平台提供了哪些通用功能? .....	7
1.3	S60 平台不同的“版本”分别代表什么意思? .....	7
1.4	什么是 Feature pack? .....	7
1.5	S60 平台对开发伙伴的意义? .....	7
1.6	如何部署 S60 平台? .....	8
1.7	S60 平台有哪些不同的 edition 和 feature pack? .....	8
1.8	S60 终端的目标市场? .....	8
1.9	S60 终端上有哪些用户界面? .....	8
<b>2.</b>	<b>S60 平台的架构</b> .....	<b>9</b>
2.1	S60 平台的架构? .....	9
2.2	Symbian OS 表示什么? .....	9
2.3	Symbian OS 扩展是什么? .....	9
2.4	S60 平台服务指什么? .....	10
2.5	S60 应用服务指什么? .....	10
2.6	S60 Java™ 技术服务指什么? .....	11
2.7	S60 应用代表什么? .....	11
2.8	S60 平台是否提供用户界面? .....	11
<b>3.</b>	<b>S60 平台的特性</b> .....	<b>12</b>
3.1	各个版本 和 feature pack 的底层操作系统分别是什么? .....	12
3.2	S60 1st Edition 的基础软件中包括了些什么? .....	12
3.3	S60 1st Edition, Feature Pack 1 有哪些主要特性? .....	12
3.4	S60 2nd Edition 中包括哪些基础软件? .....	12
3.5	S60 2nd Edition 的各个 Feature Pack 都有哪些主要特性? .....	13
3.6	S60 3rd Edition 中包括哪些基础软件? .....	14
3.7	S60 3 <sup>rd</sup> Edition, Feature Pack 1 有哪些主要特性? .....	15
3.7.1	Adobe 公司的 Macromedia Flash Lite 2.0 是否会配置到所有 S60 3rd Edition, Feature Pack 1 终端上? .....	16
3.7.2	S60 3rd Edition, Feature Pack 1 还向开发伙伴们提供了哪些增强? .....	16
<b>4.</b>	<b>S60 3rd Edition</b> .....	<b>17</b>
4.1	S60 3rd Edition 有哪些重要概念和事项? .....	17
4.2	这些新特性的开销和利益是什么? .....	17
4.3	S60 3rd Edition 是否与 S60 平台前几个版本保持二进制兼容? .....	18

4.4	针对ARM®架构的应用程序二进制接口（ABI）有哪些优点？	18
4.5	二进制兼容断点是否意味着更高的移植成本？	18
4.6	平台安全包含哪些特性？	18
4.7	什么是 Capability 组别？	19
4.8	我的应用需要哪些信任级别才能使用各个 capability？	20
4.9	S60 3rd Edition 应用有哪些签名要求？	20
4.10	所有应用都必须签名吗？	21
4.11	所有应用都必须通过 Symbian Signed 吗？	21
4.12	如果一个应用无需 Symbian Signed 也可运行，是否有其他理由让其通过 Symbian Signed？	22
4.13	测试期间能否使用受保护的 capabilities？	22
4.14	获得 Symbian Signed 资格需要哪些条件？	22
4.15	Symbian Signed 过程中有什么工具可供辅助？	22
4.16	获取 Symbian Signed 资格有开销，那么免费软件应用如何进入市场？	23
<b>5.</b>	<b>S60 平台支持的开发技术</b>	<b>24</b>
5.1	有哪些可用的开发技术？	24
5.2	C++ 应用开发	24
5.2.1	应用开发是否使用标准 C 或 C++？	24
5.2.2	用这些 C++ API 可以创建哪些类型的应用？	24
5.2.3	我的 C++应用是否适用于其他 S60 终端？	24
5.2.4	将现有应用移植到 S60 平台有何信息？	24
5.2.5	如何将应用安装到 S60 终端上？	24
5.3	Java™ 开发	24
5.3.1	S60 平台支持哪些 Java™ MIDP 版本？	24
5.3.2	S60 1st Edition 支持哪些 Java™ API？	25
5.3.3	S60 2nd Edition 支持哪些 Java™ API？	25
5.3.4	S60 3rd Edition 支持哪些 Java™ API？	26
5.3.5	平台安全是否影响到 Java™开发？	26
5.3.6	我的 Java™ 应用是否兼容其它 S60 终端？	26
5.3.7	我的 Java™应用是否兼容基于其它平台的终端？	26
5.3.8	我的 Java™ 应用是否兼容其它 MIDP 终端？	26
5.3.9	如何将应用下载并安装到 S60 终端上？	26
5.4	内容开发	26
5.4.1	S60 平台支持哪些浏览标准？	26
5.4.2	S60 平台支持哪些消息技术标准？	27
5.4.3	S60 3rd Edition 的浏览器应用是否能运行在基于早期 edition 的终端上？	27

5.4.4	S60 3rd Edition 的 MMS 内容和应用是否适用于该平台的早期 edition? .....	27
5.4.5	什么是 OMA DRM? .....	28
5.4.6	什么是 OMA 客户端配置? .....	28
5.4.7	S60 3rd Edition 对 DRM 做了哪些加强? .....	28
<b>6.</b>	<b>S60 平台开发工具 .....</b>	<b>29</b>
6.1	C++ 开发伙伴们可以使用哪些 IDE? .....	29
6.2	C++ 开发有哪些工具? .....	29
6.3	S60 平台有哪些 Java™ 开发工具? .....	30
6.4	内容开发伙伴们可使用哪些工具? .....	30
6.5	所有诺基亚工具都对全体开发伙伴开放吗? .....	30
<b>7.</b>	<b>为 S60 平台开发 .....</b>	<b>31</b>
7.1	为何 S60 平台可作为移动应用开发的初始目标? .....	31
7.2	在各个系列的 S60 终端上对应用进行优化有哪些相关问题? .....	31
7.3	向其他诺基亚平台迁移应用时会有哪些相关问题? .....	31
7.4	向其他基于 Symbian OS 的终端移植应用时会有哪些相关问题? .....	32
7.5	向移动市场上各非相关操作系统移植应用时会有哪些相关问题? .....	32
<b>8.</b>	<b>一些已知问题 .....</b>	<b>33</b>
8.1	S60 应用开发伙伴应注意哪些已知问题? .....	33
<b>9.</b>	<b>业务案例 .....</b>	<b>34</b>
9.1	使用 S60 平台的业务案例是什么? .....	34
<b>10.</b>	<b>将应用投入市场 .....</b>	<b>35</b>
10.1	诺基亚如何帮助我的应用进入市场? .....	35
<b>11.</b>	<b>请对本文进行评价 .....</b>	<b>36</b>

## 修订记录

2003 年 10 月 29 日	版本 1.0	文档首次发布
2004 年 6 月 28 日	版本 1.1	文档全面更新
2005 年 4 月 25 日	版本 1.2	针对 S60 3rd Edition 进行更新
2005 年 10 月 10 日	版本 1.3	针对 S60 3rd Edition, Feature Pack 1 进行更新
2005 年 11 月 9 日	版本 1.4	第四章更新，对签名（ <b>signing</b> ）要求作了细化。 第五章更新， <b>Carbide.c++</b> 工具信息
2005 年 12 月 16 日	版本 1.5	第五章更新， <b>DRM</b> 细化
2005 年 12 月 23 日	版本 1.6	第三章和第五章更新，关于 <b>JSR</b> 支持细节
2006 年 7 月 3 日	版本 1.7	第二章更新，关于平台架构细节，第三章更新， 新增了有关 S60 3rd Edition, Feature Pack 1 方面的内容
2007 年 1 月 31 日	中文版本 1.7	中文文档全面更新

## 1. 一些重要概念

### 1.1 S60 平台是什么？

S60 平台构建于 Symbian OS 之上，是市场领先的智能手机平台。它集中了日益成长的企业和消费用户所期望的所有重要的移动技术，也为移动市场中所有各方提供了获益机会。随着 S60 平台的发展及其开发和实施方面的不断创新，智能手机的功能配置标准也得到了提升。

### 1.2 S60 平台提供了哪些通用功能？

基于 S60 平台的终端所具有的功能为：

- 先进的通话功能。
- 最小分辨率为 176 x 208 像素的彩色显示屏。
- 创新的外观设计和键盘布局。
- 各种个人信息管理（PIM）应用（如名片夹和日历）。
- 消息。
- 互联网浏览。
- 数码相机。

S60 终端还具备以下功能：

- 音乐播放器。
- 多媒体资料。
- 视频摄录程序。
- 录音程序。
- FM 收音机。

### 1.3 S60 平台不同的“版本”分别代表什么意思？

S60 平台不断发展，不断增添新功能。必须确保业内各参与方买进的是最大最好的平台。S60 平台的新版本 代表一组重要的更新和新增功能，常常伴随其底层 Symbian OS 版本的修改。

### 1.4 什么是 Feature pack？

Feature pack 是新版本 发布之前一些新增功能的重点发布。例如：S60 2nd Edition 有三个 feature pack，每一个 feature pack 都为该版本 带来了一些增强功能。各种 feature pack 的不断发行使 S60 平台持续发展，同时也维持了各个版本的共同基准。

### 1.5 S60 平台对开发伙伴的意义？

行业参与者们总是希望以最低成本扩展其总体市场率。面向某个目标终端的新应用或新内容的开发总会引发利弊权衡上的考虑：将自己的产品转到新终端上的成本，及更大规模的目标用户群所产生的利益。这就要求跨越基于该平台某个特定版本的不同终端的软件实现。开发

伙伴们将得到保证：那些仅用平台版特性所创建的应用将被非常方便地移植到基于该版本的任何终端上。

## 1.6 如何部署 S60 平台？

S60 平台被授权给一批业界同行，如：联想、LG 电子、诺基亚移动电话，和三星，他们为市场开发各种终端。S60 终端的范围一直在变化，您可以从 S60 网站终端专区 ([www.s60.com/products](http://www.s60.com/products)) 上找到有关当前终端范围的详细信息。

## 1.7 S60 平台有哪些不同的 edition 和 feature pack？

S60 平台的主要版本 和 feature pack 如下：

- S60 1st Edition.
  - S60 1st Edition, Feature Pack 1.
- S60 2nd Edition.
  - S60 2nd Edition, Feature Pack 1.
  - S60 2nd Edition, Feature Pack 2.
  - S60 2nd Edition, Feature Pack 3.
- S60 3rd Edition.
  - S60 3rd Edition, Feature Pack 1.

## 1.8 S60 终端的目标市场？

S60 平台的目标是用面向大规模市场的终端服务于最广泛的消费和企业用户，同时也向企业、娱乐、个人增效，及游戏等专业的细分领域提供产品。

## 1.9 S60 终端上有哪些用户界面？

S60 平台并不在其规范中强制规定用户界面特性、屏幕解析度、或输入方式等。因此，它允许制造商们生产风格迥异的定制终端。平台规范中确实包括用户界面样式和一些 API，因此，希望在所有用户界面的实现中都遵循通用的用户界面样式。作为所有用户界面样式的基础，Avkon UI 库加强了这一特性。

## 2. S60 平台的架构

本章讲解 S60 平台的底层架构。

### 2.1 S60 平台的架构？

S60 平台的架构如图 1 所示。

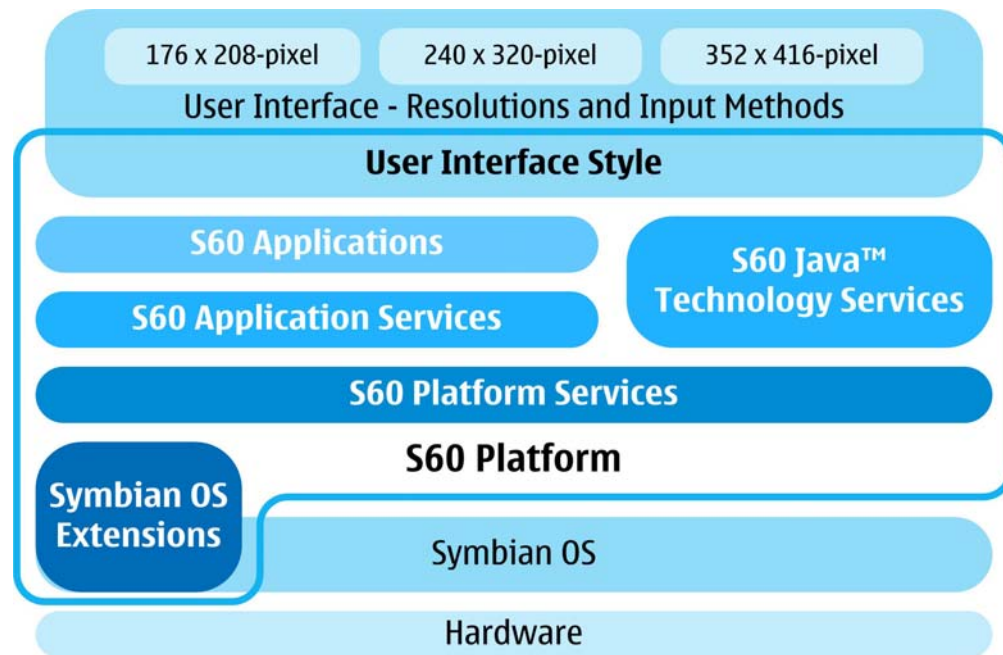


图 1：S60 平台架构示意图

S60 平台的主体部分为 S60 用户界面样式、S60 应用、S60 应用服务、S60 Java™ 技术服务、S60 平台服务，及构建于 Symbian OS 之上的 Symbian OS 扩展。以下各节将讨论这些术语。

### 2.2 Symbian OS 表示什么？

S60 平台的底层操作系统就是 Symbian OS。开发前，开发伙伴们应访问诺基亚论坛网站（[www.forum.nokia.com/documents](http://www.forum.nokia.com/documents)）了解 Symbian OS 版本各个版本或 feature pack 的功能细节。

### 2.3 Symbian OS 扩展是什么？

Symbian OS 扩展是一组（能力）capabilities，它们让 S60 平台与终端的各项硬件功能（如振动提示、终端灯光，及电池电量等）进行交互。

## 2.4 S60 平台服务指什么？

S60 平台服务是由 S60 平台提供的一些基本服务，包括：

- 应用框架服务（Application Framework Services）— 为启动各种应用和服务、管理持续性状态及用户界面组件等提供一些基本能力。
- 用户界面框架服务（UI Framework Services）— 为各种用户界面组件及各种用户界面事件的处理提供正确的外观。
- 图形服务（Graphics Services）— 为创建图形并将其绘制到显示屏上提供各种能力。
- 定位服务（Location Services）— 让 S60 平台能确定某台终端的位置。
- 基于 Web 的服务（Web-based Services）— 为各种基于 Web 的功能（包括浏览、文件下载，及消息等）建立连接并进行交互提供服务。
- 多媒体服务（Multimedia Services）— 提供音频/视频播放能力，支持流媒体和语音识别。
- 通讯服务（Communication Services）— 为本地及广域通讯提供支持，范围从蓝牙技术到语音呼叫等。

## 2.5 S60 应用服务指什么？

S60 应用服务是供各种 S60 应用使用的一组能力，第三方开发伙伴可用来为各种应用提供一些基本功能，包括：

- PIM 应用服务（PIM Application Services）— 提供个人信息管理（personal information manager，PIM）基本功能，包括：名片夹、日历、任务管理，及诸如记事本和时钟等相关功能。
- 消息应用服务（Messaging Application Services）— 支持各种消息类型，如：短信息服务（SMS）、多媒体消息服务（MMS）、电子邮件、BIO 消息（智能消息），及即时消息（IM）。
- 浏览器应用服务（Browser Application Services）— 提供浏览 Web 内容的能力，包括对 Adobe Macromedia Flash Lite 2.0 的支持、视频渲染、可缩放矢量图形— Tiny（SVG-T）渲染，及音频渲染等。

## 2.6 S60 Java™ 技术服务指什么？

S60 Java™ 技术服务支持 Java™ 2 Platform, Micro Edition (J2ME™) Java™ Technology for the Wireless Industry (JTWI) specification (JSR-185)。S60 平台支持的功能包括：有限连接设备 (CLDC) 配置 1.1 (JSR-139) 及对这个配置的移动信息设备描述 (MIDP) 2.0 (JSR-118) 扩展。

此外，S60 Java 技术服务支持一系列的附加 API，帮助实现对 S60 文件系统和个人信息管理 (PIM) 向关数据的访问、蓝牙技术的使用、消息、音频、视频、Web 服务、安全和信任服务、定位信息、会话发起协议 (SIP)，和三维图形等。

## 2.7 S60 应用代表什么？

S60 应用就是供终端用户使用的应用程序，包括个人信息管理 (PIM)、消息、媒体应用，及各种用户资料管理等。

## 2.8 S60 平台是否提供用户界面？

S60 平台定义了一个用户界面样式和一些 API，但并未强制规定屏幕尺寸及输入方式。获授权厂商可以自由实现自己的用户界面。开发伙伴们在编制用户界面应用时必须考虑到可缩放性，因为并不能确定某种特定的用户界面尺寸。

## 3. S60 平台的特性

这一章讨论 S60 平台各版本中的底层操作系统、基础软件、及导引软件。

### 3.1 各个版本和 feature pack 的底层操作系统分别是什么？

各个版本及各自的操作系统是：

- S60 1st Edition — Symbian OS v6.1.
- S60 2nd Edition — Symbian OS v7.0s.
  - S60 2nd Edition, Feature Pack 1 — Symbian OS v7.0s.
  - S60 2nd Edition, Feature Pack 2 — Symbian OS v8.0a.
  - S60 2nd Edition, Feature Pack 3 — Symbian OS v8.1a.
- S60 3rd Edition — Symbian OS v9.1.
  - S60 3rd Edition, Feature Pack 1 — Symbian OS v9.2.

### 3.2 S60 1st Edition 的基础软件中包括了些什么？

S60 1st Edition 中包括下列基础软件：

- Java™ Platform, Micro Edition (Java™ ME) APIs，包括：
  - 移动信息设备描述（Mobile Information Device Profile, MIDP） 1.0。
  - 有限连接设备配置（Connected Limited Device Configuration, CLDC） 1.0。
  - Wireless Messaging API (JSR-120).
  - Mobile Media API (JSR-135).
- XHTML/WML 浏览。
- 多媒体消息服务（MMS）。

### 3.3 S60 1st Edition, Feature Pack 1 有哪些主要特性？

值得一提的是：S60 1st Edition, Feature Pack 1 中引入了 Java™ APIs for Bluetooth (JSR-82)。

### 3.4 S60 2nd Edition 中包括哪些基础软件？

S60 2nd Edition 包括下列基础软件：

- 有关皮肤（主题）和数字版权管理的（DRM）C++ API
- Java™ Platform, Micro Edition (Java™ ME), Mobile Information Device Profile (MIDP) 2.0，包括因纳入了有限连接设备（CLDC）HotSpot 编译器而获得增强的性能。

- 通过 TCP/IP 的 XHTML 浏览。
- 支持 HTTP 传输的多媒体消息服务，及因支持增强的同步多媒体集成语言（SMIL）而获得的一些高级表现能力。
- 开放移动联盟（Open Mobile Alliance, OMA）客户端配置，允许针对如浏览、MMS，及空中下载（OTA）日历同步等服务进行终端设置。这项技术实现了方便的终端配置，并确保消费者能充分利用终端的各项能力。
- 通过 OMA 转发锁定实现的 DRM。

### 3.5 S60 2nd Edition 的各个 Feature Pack 都有哪些主要特性？

以下列出了 S60 2nd Edition 各 feature pack 所提供的主要特性。更多详情请参阅 [www.forum.nokia.com](http://www.forum.nokia.com)。

- Feature Pack 1
  - Presence Open 和 Bluetooth notifier C++ APIs.
  - Wireless Messaging API 1.1 (JSR-120) 和 Mobile Media API 1.1 (JSR-135) Java™ APIs.
  - 支持 4 倍变焦百万像素相机，及视频短篇的录制和回放。
- Feature Pack 2
  - 用于浏览器插件的各种 C++ API，连接监控服务器，数字版权管理（DRM）授权管理器，仿真文件，DRM 帮助程序，pictograph，DevASR，语音识别工具，多媒体框架（MMF）语音识别系统（SRS）定制命令，数据同步描述列表，内容访问框架，GIF 缩放程序，Huffman 编码解码，消息队列，发行及预订，位置获取（Location Acquisition），OpenGL ES，ES，EGL，和 Symbian XML 框架。
  - Mobile 3D Graphics API for J2ME™ (JSR-184), FileConnection API (JSR-75), 及 PIM API (JSR-75).
  - 支持六倍变焦 130 万像素相机、宽频 CDMA（WCDMA）和 EDGE 网络，以及 IPv6。
- Feature Pack 3
  - 各种 API，用于功能发现、可缩放图标、浏览器控件、下载管理器用户界面库、下载管理器引擎、图像转换库、图像转换插件、通用串行总线（USB）客户端驱动程序，及一个功能已升级的工具 API。
  - J2ME™ Web 服务规范 (JSR-172)和支持 OBEX 的 Java API for Bluetooth (JSR-82)。
  - 支持可缩放用户界面（176 x 208 像素、240 x 320 像素，和 352 x 416 像素显示屏）。

### 3.6 S60 3rd Edition 中包括哪些基础软件？

除 S60 2nd Edition 中已提供的功能外，下列 API 也被包括在 S60 3rd Edition 的基础软件中：

- 一些 C++ API，包括：
  - 各种定位 API（包括 Landmark API、Landmark UI Add、Edit，及 Select API、Landmark Search API，及 Basic Location Info Display [BLID] application API）。
  - 各种 Web Services API（包括 WS Connection API、WS Description API、WS Manager API，和 XML Extensions API）。
  - 各种会话发起协议（SIP）API。
  - 开放移动联盟（OMA）数字版权管理（DRM）v2 API。
  - OMA Datasynch 1.2 API.
  - Multimedia Framework (MMF) DRM API.
  - Light API.
  - Instant messaging (IM) API.
  - IM Application Launch API.
  - 支持 Bluetooth 1.2。
  - EXIF API.
  - Find Item API.
  - Accessory API.
  
- 各种 Java™ API，包括：
  - Security and Trust Services API (JSR-177)。
  - Java Location API (JSR-179)。
  - SIP API (JSR-180)。
  - Wireless Messaging API (WMA) 2.0 (JSR-205)。

### 3.7 S60 3<sup>rd</sup> Edition, Feature Pack 1 有哪些主要特性？

以下列出了 S60 3rd Edition, Feature Pack 1 所提供的一些主要特性。更多详情请参阅 [www.forum.nokia.com](http://www.forum.nokia.com)。

- 空中下载固件(Firmware over-the-air, FOTA)更新。
- 最新开源软件(Open Source Software, OSS)浏览器
- 增强的终端管理功能，包括应用程序交付和管理、强制设置、终端锁定和消除，及其他一些安全特性。
- 作为终端可选功能的诺基亚“一键通话”功能（Push to Talk over Cellular, PoC）。
- 针对获授权厂商和运营商的增强的用户界面定制，可通过空中下载更新。
- 增强的用户界面和本地应用，包括实现通用设置的“控制面板”。
- 可选的 Adobe Marcomedia Flash Lite 2.0 。
- 一些配置选项让 S60 获授权厂商能创建低成本终端，以进一步拓展 S60 终端的市场占有量。
- 一些最新 C++ API，包括
  - Gallery Content Listing API.
  - Optical Character Recognition API.
  - Central Repository Notification Handler API.
  - Profiles Engine API.
  - Screen Mode API.
  - OpenGL V1.1 API (已更新)
  - Telnet API.
  - App Framework Animation API.
  - OBEX API (已更新)
  - OBEX MTM API (已更新)
- 一些最新的 Java™ API，包括：
  - Advanced Multimedia Supplements (AMMS) API (JSR-234).
  - Scalable Vector Graphics 2D API (JSR-226).
  - Mobile 3D Graphics (M3G) API for J2ME™ v1.1 (JSR-184)（已更新）。

### 3.7.1 Adobe 公司的 Macromedia Flash Lite 2.0 是否会配置到所有 S60 3rd Edition, Feature Pack 1 终端上?

不会。Adobe 公司的 Macromedia Flash Lite 2.0 是 S60 平台的可选组件，不会被实现到所有终端上。但是，消费用户可以在并未预装该软件的终端上自行安装 Flash Lite 2.0。

Flash Lite Framework API 作为 S60 平台的组成部分出现，允许开发伙伴与 Flash Lite 2.0 播放器交互。但是，该 API 只能运行于已安装 Flash Lite 2.0 播放器的终端，不管该播放器是预装组件还是由用户自己安装的。

### 3.7.2 S60 3rd Edition, Feature Pack 1 还向开发伙伴们提供了哪些增强?

在 S60 3rd Edition 中，随 SDK 一起供应的 S60 文档有了显著改进。在文档结构和内容两方面都有不少修订，包括：

- 每个 API 的文档结构已得到简化，通过一个单一的文档来提供所有相关信息。
- 每个 API 都以相同一致的模板进行文档归档。因为相关材料其位置都可预测，所以查找信息更为方便。
- 每个 API 的文档都是完整的：不存在任何遗漏部分。
- 为每个 API 提供了清晰正确的代码范例。

总而言之，这些修订将使开发伙伴们更方便地找到自己所需要的信息，并确保文档信息的完整性。

## 4. S60 3rd Edition

本章讲解 S60 3rd Edition 的各种特征，如：二进制兼容断点、平台安全，及应用认证过程等。

### 4.1 S60 3rd Edition 有哪些重要概念和事项？

关于 S60 3rd Edition，相关各方必须注意的重要概念和事项有：

- 平台安全。
- 数据锁定（Data caging）。
- 受信任计算基。
- 开发伙伴对不递交 Symbian Signed 的应用进行自签名。
- Symbian Signed 认证。
- Capabilities.
- 二进制兼容断点。
- 实时内核。
- 新编译器和 build 工具。
- 日益增长的终端差异和细分。
- 市场拥有量。

### 4.2 这些新特性的开销和利益是什么？

开发伙伴们需要支付一些费用。例如：S60 3rd Edition 需要一些新开发工具，尽管有一些免费工具（请见 6.1 “C++ 开发伙伴们可使用哪些 IDE”）。另外，开发伙伴们还需要承担认证费用，其中包括：

- 获取用于 Symbian Signed 认证的 VeriSign Authenticated Content Signing (ACS) Publisher ID。目前一个 ACS Publisher ID 需要 295 欧元（合 350 美元），可签名 10 次。
- Symbian Signed 认证测试。在编写本文时的测试价格是：第一轮测试从 185 欧元（合 220 美元）到 560 欧元（合 665 美元）不等，而同一应用的后续测试费用则从 160 欧元（合 190 美元）到 280 欧元（合 330 美元）不等。对内容认证，小批量的每项 40 欧元（合 47 美元），但大批量的价格相应降低。目前的测试价格可以从 Symbian Signed 网站（[www.symbiansigned.com/app/page/testhouses](http://www.symbiansigned.com/app/page/testhouses)）上查得。

虽然认证费用对每个应用都会发生，但各种开发工具的成本却可以被分摊到多个项目中。请注意，如果创建的是无需 Symbian Signed（请见 4.10，“所有应用都必须签名吗”）的应用，您还是有可能避免这些认证开销。此外，Handango 公司提供了一种 Symbian Signed 服务，认证费用可通过销售版税支付，而无需预先支付。

S60 3rd Edition 的一些新特性所带来的利益是：

- 针对ARM® 架构的应用程序二进制新接口（Application Binary Interface, ABI）编译器将产生较少的ROM 内存占用，具有更为高效的代码，从而导致更小更快的应用程序。
- 增强的平台安全性降低了应用数据遭到破坏或被盗用的可能性。
- 市场信心大大增强，这是因为：S60 平台将对各种安全问题进行预处理，并提供一种应用验证机制。谨慎的消费者将可放心使用应用，而不会如以前那样担心使用了未知供应商的应用。
- 消费者如果对应用不放心，可以拒绝使用那些未经认证的应用。
- 通过认证，第三方开发伙伴将获得更多的 API，由此实现增强的客户化定制能力，从而提供更为广泛的平台功能。
- 增强的数字版权管理（DRM）将加强软件的防盗版能力。
- S60 终端已支持单芯片硬件解决方案，提高了硬件配置的灵活性，因而能提供更为廉价的终端。

#### 4.3 S60 3rd Edition 是否与 S60 平台前几个版本保持二进制兼容？

引入针对ARM® 架构的应用程序二进制接口（ABI）意味着：S60 3rd Edition 不再保持与S60 平台前几个版本间的二进制兼容。

#### 4.4 针对ARM® 架构的应用程序二进制接口（ABI）有哪些优点？

Symbian OS 引入了针对ARM® 架构的应用程序二进制接口（ABI），为基于S60 3rd Edition及其后续版本的终端提供了显著的性能优势。

#### 4.5 二进制兼容断点是否意味着更高的移植成本？

二进制兼容断点确实表示：需要用不同的工具组合对应用重新进行 build，使之能在该类断点的两边都能运行。但是，经过合理的方案设计花费将不会很高。大多数功能都不会受到影响，而且通过可缩放设计也能避免大部分的严重用户界面问题。

#### 4.6 平台安全包含哪些特性？

在 S60 3rd Edition 中引入平台安全有许多重要目的，即为了保护终端的完整性，限制对敏感数据和操作的访问。消费用户和企业用户将获得更强的抗病毒能力。同时，运营商、获授权厂商及第三方开发伙伴将能获得更强的品牌及数据保护能力。平台安全包含三个关键概念：

- **数据锁定（Data caging）** — 一种全新的目录系统，允许应用以限制访问的方式安全地保存数据。
- **信任计算基** — 包括软件认证和授权、安全储存和执行、启动保护，及各种信任硬件服务。换句话说，它提供了一种机制，能保证非信任应用是受控的，并且终端是受保护的。
- **Capability 模型** — Capability是Symbian OS内部的一组特性和功能，按照其使用对终端所产生的影响进行分组。每个capability都要求对试图使用它的应用的某种级别的信任。Capability操作越敏感，那么某个应用为使用它所需要的信任级别就越高。大多数API并不需要信任，但某些API 却要求应用通过Symbian Signed后才能使用。

## 4.7 什么是 Capability 组别？

共有四个组别的 capabilities:

- **完全开放** — 这些capability 大约占Symbian OS API 的 60%。属于这一组别的API足以创建那些访问用户界面及具备数据存储能力的独立应用。
- **安装时由用户授予** — 这些capability 覆盖大部分通讯功能、个人网络、互联网接入、消息，及电话呼叫等，也包括对敏感用户数据的访问功能，如名片夹和日历内容。它也覆盖对基本位置信息的访问。
- **通过Symbian Signed 授予** — 这些capability包括“安装时由用户授予”的capability加上诸如终端设置、电源管理、事件生成，及扩展的位置信息等更多的高级终端信息和控件功能。
- **由制造商授予** — 这些capability提供了对操作系统和受保护数据的基本访问能力。

#### 4.8 我的应用需要哪些信任级别才能使用各个 capability?

应用如要使用某些 capabilities 需要下列信任级别：

- *完全开放*— 这些capability 向全部应用开放，应用也无需Symbian Signed。
- *安装时由用户授予* — 这些capability 可以由用户在安装应用时授予。已通过Symbian Signed的应用则无需用户授予即可访问这些capability。
- *由Symbian Signed 授予* — 这些capability 包括全部“安装时由用户授予”的capability，加上一个扩展组。这些扩展的capability 要求应用必须通过Symbian Signed后才能使用。此外，在提交Symbian Signed时还需附送一份声明，解释为什么需要使用这些capabilities。
- *由制造商授予* — 这些capabilities 需要终端制造商的业务支持，以便在Symbian Signed 过程中授予这些capabilities。

#### 4.9 S60 3rd Edition 应用有哪些签名要求?

表一列出了 S60 3rd Edition 应用的一些签名要求。

实施阶段	需要 Symbian Signed	无需 Symbian Signed
终端上测试	用 Symbian 开发伙伴证书签名 (如果使用某些 capabilities)	用自生成证书签名—“自签名”
提交 Symbian Signed	用 Authenticated Content Signing (ACS) Publisher ID 签名	不需要
已获 Symbian Signed	用一个获 Symbian 根证书信任的一次性 ACS Publisher ID 签名 (由测试机构签名或用自认证证书签名)	不需要

表 1: S60 3rd Edition 应用的各种签名要求

这些签名要求分别是：

- **自签名** — 不被提交给Symbian Signed 的应用和\*.sis 包内容必须由开发伙伴进行自签名。开发伙伴使用S60 3rd Edition SDK所提供的工具创建一个密钥对（证书），并由此提供签名。自签名并不会向应用授予任何capability。消费用户在安装应用时能向该应用授予“*安装时由用户授予*”的各种capability。
- **Symbian 开发伙伴证书签名** — 需要使用由Symbian Signed 授予，或者本应“*安装时由用户授予*”但通过Symbian Signed后可避免的那些应用必须用Symbian 开发伙伴证书认证以便于开发期间在终端上进行测试。这些证书可以从Symbian Signed 网站（[www.symbiansigned.com](http://www.symbiansigned.com)）免费获取。
- **Authenticated Content Signing (ACS) Publisher ID签名** — 将提交Symbian Signed 的应用和\*.sis 包内容必须由开发伙伴使用ACC Publisher ID进行签名。这种签名将确保测试机构能验证应用和内容的来源。
- **Symbian Signed证书签名** — 这种签名由一个安装在Symbian OS v9 终端上获Symbian 根证书信任的一次性ACS Publisher ID 提供，表明了某个应用或\*.sis 内容包已通过了Symbian Signed，并解除了该应用所声明的对相关capability的锁定。应用或\*.sis 内容包可以以这种方式由Symbian Signed测试机构或已达到自认证资质的开发伙伴签名。

#### 4.10 所有应用都必须签名吗？

是的。S60 3rd Edition 上的应用安装器要求所有应用和\*.sis 包内容都必须经过签名。既可以使用一种自签名机制（用一个自生成证书）完成这种签名，同时也可以通过 Symbian Signed 签名。

#### 4.11 所有应用都必须通过 Symbian Signed 吗？

不是。一个应用是否需要Symbian Signed取决于它是否需要使用一些受信任的capabilities。某些应用，如一些游戏，可能无需证书即可安装和运行。而那些使用了“*安装时由用户授予*” capabilities的应用，可以由用户进行授权，也不需要这种证书。当然，拥有证书也是有利的，因为这样就无需消费用户在安装时为受保护的capabilities授权。所有需要使用“*由Symbian Signed授予*”和“*由制造商授予*” capabilities的应用则必须通过Symbian Signed。

#### 4.12 如果一个应用无需 Symbian Signed 也可运行，是否有其他理由让其通过 Symbian Signed?

为某个应用获得 Symbian Signed 意味着：当用户安装该应用时不会再收到一条警告消息，也不会再出现提示要求用户授予“安装时由用户授予”的 capabilities（如果该应用用到了这些 capabilities）。因此，安装过程将更直观，而用户会更乐意安装该应用，因为不会再有任何警告提示。

更重要的是，Symbian Signed 也是进入众多销售渠道的需求。例如，所有通过诺基亚销售渠道分销的 Symbian 本地应用都必须通过 Symbian Signed。

#### 4.13 测试期间能否使用受保护的 capabilities?

可以。S60 SDK 模拟器会关闭 capability 访问控制，所以即使并未通过 Symbian Signed 也可以对这些应用进行测试。至于在终端上测试，就需要一个 Symbian 开发伙伴证书。免费提供的这些证书允许该应用针对某个特定终端进行签名，这样就可以对还未通过 Symbian Signed 的应用进行终端上调试和接受性测试。

#### 4.14 获得 Symbian Signed 资格需要哪些条件?

Symbian Signed 流程如下：

1. 在 Symbian Signed 网站 (<http://www.symbiansigned.com>) 上在线注册，签署合同协议书。
2. 用申请的用户名和密码登录到 Symbian Signed 账户，下载一个工具，以便将 VeriSign Authenticated Content Signing (ACS) Publisher ID 导出为一种标准格式，用于 Makesis。
3. 签名该应用的 \*.sis 文件（用前面下载的工具和 Makesis），并将其与 \*.pkg 文件和用户指南一起提交给选定的测试机构。开发伙伴将收到一份回执。
4. 测试机构验证 ACS Publisher ID 后通过电子邮件发出测试报价。
5. 付款后开始测试。
6. 无论应用是否通过测试都会发出通知：
  - 如果该应用未能通过测试，需要对应用进行修改并回到第三步重新提交。
  - 如果该应用通过了测试，将用嵌入在 S60 终端上的 Symbian 根证书（Symbian B）所信任的一次性 ACS Publisher ID 进行再签名。然后这个应用将被上载到该开发伙伴的账户上供分发。

#### 4.15 Symbian Signed 过程中有什么工具可供辅助?

有的。Symbian 有限公司与 SysOpen Digia Plc 公司一起开发了一个测试工具，供提交认证前根据 Symbian Signed 测试标准对应用进行验证。该工具目前支持 S60 2nd Edition, Feature Pack 3 及之前版本的应用。更多详情请参阅 [www.symbiansigned.com](http://www.symbiansigned.com) 网站。

#### 4.16 获取 Symbian Signed 资格有开销，那么免费软件应用如何进入市场？

免费软件应用可参加“Symbian Signed 自由软件市场之路”计划。该计划为各种免费软件提供了免费的 Symbian Signed 机会。这个计划所涵盖的应用软件能让用户免费使用其全部功能。该计划不包括供演示的应用、共享软件，或以其它授权形式需要用户付费才能使用其全部功能的软件，也不包括开发伙伴可从中获取收益的应用，如广告软件等。Honorware 或 donationware 则被纳入到该计划，这些软件并不强制消费者付费但允许他们自愿付费。

该计划针对免费应用，对开发伙伴的业务模式并无任何限制。因此，商业开发者如创作自由软件，他们也可以与开源或业余开发爱好者一样使用该计划。

## 5. S60 平台支持的开发技术

### 5.1 有哪些可用的开发技术？

C++ 是 S60 平台的本地编程语言。开发伙伴们可以用 C++ 创建各种能访问一些应用引擎（包括照相和电话簿）的应用，及那些采用了多种技术（如蓝牙连接、红外[IR]、多媒体、消息、网络，和通话等）的应用。

也可以使用 Java™ MIDP 2.0 进行应用开发，并为 XHTML 浏览和多媒体消息服务开发内容。

### 5.2 C++ 应用开发

#### 5.2.1 应用开发是否使用标准 C 或 C++？

S60 平台上使用的 C++ 语言已为内存有限的小型终端做过优化，不支持标准模板库（Standard Template Library, STL），而且还对标准 C++ 作了其他方面的修改。本地 C++ 应用是用 Symbian 有限公司自己的 C++ 版本开发的，它承袭了 C++ 的大部分习语，但也具有自己独特的术语。虽然 ANSI C 获支持，但很少用到。

#### 5.2.2 用这些 C++ API 可以创建哪些类型的应用？

用 C++ 开发应用将让开发伙伴们充分发挥该平台的潜力，以创建范围广泛的应用。例如：一些公共 API 让开发伙伴们创建那些能操控内置相机、能编辑并发送多媒体消息（MMS）、能使用蓝牙技术与其它终端进行数据交换，以及能使用通话引擎等功能的应用。

#### 5.2.3 我的 C++ 应用是否适用于其他 S60 终端？

为 S60 平台某个特定版本编制的 C++ 应用可以运行于使用同一版本的其他终端上，条件是这些应用未用到 feature pack APIs 或导引软件 APIs。如前所述，在 S60 2nd Edition 和 S60 3rd Edition 之间存在着二进制兼容断点。诺基亚致力于减少平台不同版本间的兼容性问题，但在 S60 1st Edition 和 S60 2nd Edition 之间还是存在着一些已知问题。有关这些问题的更多信息请参阅第 8 章“一些已知问题”。

#### 5.2.4 将现有应用移植到 S60 平台有何信息？

有。诺基亚论坛网站移植专区（[www.forum.nokia.com/porting](http://www.forum.nokia.com/porting)）中有系列文档讲述从 UIQ、Palm OS，及 Pocket PC 等平台及从诺基亚 9200 通讯器系列终端移植时的一些要求及所需考虑的一些问题。

#### 5.2.5 如何将应用安装到 S60 终端上？

可以用多种方法下载应用。用蓝牙或红外(IR)连接实现本地 PC 传递是一种方法，而用移动浏览器进行空中下载（OTA）则是另一种方法。已下载应用通过应用安装器被安装到 S60 终端。

### 5.3 Java™ 开发

#### 5.3.1 S60 平台支持哪些 Java™ MIDP 版本？

S60 1st Edition 支持 Java™ Platform, Micro Edition (Java™ ME)，有限连接设备配置（CLDC）1.0，和移动信息设备描述（MIDP）1.0。

S60 2nd Edition 支持 MIDP 2.0 和具有 CLDC HotSpot 虚拟机实现的 CLDC 1.0。

S60 3rd Edition 增加了对 CLDC 1.1 的支持。

### 5.3.2 S60 1st Edition 支持哪些 Java™ API?

S60 1st Edition 所支持的 Java™ API 有：

- Wireless Messaging API (WMA) (JSR-120)，支持通过 GSM 和 CDMA 网络发送和接收短消息（SMS）。
- Mobile Media API (JSR-135)，应用能使用声音、视频，及动画回放，以及视频和音频录制。
- Nokia UI API，诺基亚专属 Java API，提供用于生成声音、底层图形，及访问整个屏幕的接口。

### 5.3.3 S60 2nd Edition 支持哪些 Java™ API ?

S60 2nd Edition 所支持的 Java™ API 有：

- S60 1st Edition 所支持的所有 Java API。
- Wireless Messaging API (WMA) (JSR-120)，增强了对短消息服务（SMS）推送的支持。
- Java APIs for Bluetooth (JSR-82)，用于蓝牙连接。

### 5.3.4 S60 3rd Edition 支持哪些 Java™ API ?

S60 3rd Edition 所支持的 Java™ API 有：

- S60 2nd Edition 所支持的所有 Java API。
- Security and Trust Services API for J2ME™ (JSR-177)。
- Location API for J2ME™ (JSR-179)。
- Session Initiation Protocol (SIP) API for J2ME™ (JSR-180)。
- Wireless Messaging API (WMA) 2.0 (JSR-205)。

### 5.3.5 平台安全是否影响到 Java™ 开发?

不会。Java™ 开发不会受平台安全的影响。只有本地 Symbian C++ 应用才受到影响，因为它们继承了某些底层 capabilities。

### 5.3.6 我的 Java™ 应用是否兼容其它 S60 终端?

各种 Java™ 应用应能兼容使用 S60 平台同一版本的终端，条件是这些应用未用到可能被用作导引软件实现的其他那些 Java™ Specification Requests (JSRs)。S60 平台的各个版本之间有一些变化（详见 5.3.1 节“S60 平台支持哪些 Java™ MIDP 版本？”），可能引起一些兼容性问题。而且，当开发伙伴在 S60 各款终端上进行应用迁移时，必须时刻注意与用户界面缩放及输入方式有关的问题。

### 5.3.7 我的 Java™ 应用是否兼容基于其它平台的终端?

Series 40 平台、Series 80 平台，及诺基亚 7710 宽屏智能手机都实现了 S60 平台上相同的移动信息描述 (MIDP) API 和 Java™ Specification Requests (JSR)，因此它们之间可以达到较高的兼容性。在用户界面变化和不同内存条件下会出现一些一般性问题。如，Series 40 终端的屏幕比 S60 终端的小，内存也更为有限。

### 5.3.8 我的 Java™ 应用是否兼容其它 MIDP 终端?

运行移动信息设备描述 (MIDP) 应用的其它终端的兼容范围取决于该目标终端上获支持的 MIDP 和 Java™ Specification Requests (JSR) 的版本。开发伙伴们若希望充分利用 S60 平台所提供的 Java 支持，可能需要更改自己的应用以兼容其它终端。开发伙伴们若希望其创建的应用能运行于范围最广的终端上，就应该考虑使用那些被最普遍实现的 API。

很显然，使用 Nokia UI API 将使该应用不能兼容基于其它平台的终端。

### 5.3.9 如何将应用下载并安装到 S60 终端上?

有多种方法可将 MIDlet 部署到一台终端上。各种方法的选择主要取决于硬件。主要的部署方式有：蓝牙连接、红外 (IR) 连接、空中下载 (OTA，从 WAP 和 Web 网站下载)、电子邮件、多媒体消息服务 (MMS)，或串行线缆。

## 5.4 内容开发

### 5.4.1 S60 平台支持哪些浏览标准?

S60 1st Edition 支持：

- WML 和 WMLScript。
- XHTML Mobile Profile (XHTML MP)。
- WAP 及相关服务，如 WAP 推送。
- WAP Cascading Style Sheets (WAP CSS)。

S60 2nd Edition 和 S60 3rd Edition 支持：

- HTTP/1.1 Protocol over TCP/IP。
- XHTML over TCP/IP。
- Secure Sockets Layer (SSL) v3.0。
- Transport Layer Security (TLS) 1.0。

#### 5.4.2 S60 平台支持哪些消息技术标准？

S60 平台支持多媒体消息服务（MMS）。MMS 基于最初由 WAP Forum（[www.wapforum.org](http://www.wapforum.org)）和 3GPP（[www.3gpp.org](http://www.3gpp.org)）所建立的一些规范。MMS 具有与短消息（SMS）类似的“储存转发”模型，但大大扩展了这项早期技术，能处理诸如音频片断和图片等内容。S60 2nd Edition（及后续版本）提供了增强的对同步多媒体集成语言（Multimedia Integration Language, SMIL）的支持。

#### 5.4.3 S60 3rd Edition 的浏览器应用是否能运行在基于早期 edition 的终端上？

是的。该浏览器的各个版本既支持 XHTML 移动描述（XHTML MP）也支持 WML。诺基亚建议用 XHTML MP 来创建各类服务。

#### 5.4.4 S60 3rd Edition 的 MMS 内容和应用是否适用于该平台的早期 edition？

多媒体消息服务（MMS）内容适用于全部 S60 终端。S60 3rd Edition 使用 MMS 数字版权管理（DRM）。这些特性不适用于那些早期 edition。而且，因为 S60 1st Edition 并不支持同步多媒体集成语言（SMIL），开发伙伴们应确信：要求发送到 S60 1st Edition 终端上的内容是正确的。

#### 5.4.5 什么是 OMA DRM?

S60 2nd Edition 支持开放移动联盟 (Open Mobile Alliance, OMA) 转发锁定方法用于内容保护。这意味着：某些类型的文件将得到保护，防止被从某台终端上转发（通过范围广泛的消息和通讯技术）出去。这种保护适用于多种媒体文件类型和 Java™ MIDlet。用 OMA 转发锁定技术保护的内容仅可通过 HTTP 下载或多媒体消息服务 (MMS) 接收。收到被保护的内容之后，转发锁定功能即被自动激活。

处理受保护对象的应用必须使用由数字版权管理 (DRM) 引擎所提供的加密服务以保存或临时保存这些对象。大部分预装应用，如语音录音程序和多媒体资料等，都被设计成能识别这些获转发锁定保护的對象。

#### 5.4.6 什么是 OMA 客户端配置?

可以通过开放移动联盟 (OMA) 的客户端规范以最少量的用户交互进行终端设定。例如，某台终端可以接收一条消息，其中含有针对 WAP 客户端的设置，并能对终端本身进行相应的配置。可以用这种方式实现的其他设置包括：多媒体消息服务 (MMS)、电子邮件、即时消息，及接入点等。

发送配置信息可以有多种机制，包括空中下载 (OTA)，或某些媒介如用户身份模块 (Subscriber Identity Module, SIM) 卡。

#### 5.4.7 S60 3rd Edition 对 DRM 做了哪些加强?

S60 3rd Edition 就数字版权管理 (DRM) 提供了下列 API 及特性：

- S60 开放移动联盟 (OMA) 数字版权管理 v2 API 迁移。
- Multimedia Framework (MMF) DRM API.
- OMA DRM 2.0 (其实现视具体终端而定)。
- 基于 OMA data synch 1.2 的电子邮件同步 (本地及空中下载[OTA]远程)，电子邮件过滤，及针对日历的时区支持。
- 丰富的推送电子邮件 (IMAP/POP、OMA 电子邮件提示、OMA 数据同步 [DS]，及 polling 电子邮件)。

## 6. S60 平台开发工具

### 6.1 C++ 开发伙伴们可以使用哪些 IDE?

可以使用多种集成开发环境（integrated development environments, IDE）来创建 S60 C++ 应用。如：Carbide.c++、CodeWarrior® Development Studio for Symbian OS、Borland C++ Mobile Edition、Microsoft Visual C++ 6.0，及 Microsoft Visual Studio .NET 2003。

2006 年，诺基亚用基于 Eclipse 技术的工具 Carbide.c++ 取代 CodeWarrior Development Studio for Symbian OS。这套工具有三个版本：

- Carbide.c++ Express. 这个版本具有用 S60 SDK, Series 80 SDK, 或 UIQ SDK 创建 C++ 应用所需的所有工具。可以到 [www.forum.nokia.com/carbide](http://www.forum.nokia.com/carbide) 免费下载 Carbide.c++ Express。
- Carbide.c++ Express. 这个版本包括一个图形化设计工具，提供了真机调试所需的功能。支持在 S60 平台、Series 80 平台，及 UIQ 平台进行开发。其价格估计为 299 欧元（355 美元）。
- Carbide.c++ Professional Edition. 这个版本将向手机制造商提供创建 Symbian OS 终端所需要的全部工具。

### 6.2 C++ 开发有哪些工具?

诺基亚论坛网站工具和 SDKs 专区（[www.forum.nokia.com/tools](http://www.forum.nokia.com/tools)）可下载用于 C++ 开发的、支持 Carbide.c++、CodeWarrior® Development Studio for Symbian OS、Borland C++ Mobile Edition、Microsoft Visual C++ 6.0，和 Microsoft Visual Studio .NET 2003 等开发环境的各种 SDK。各个 SDK 中包括：

- S60 APIs.
- 用于测试的 S60 模拟器。
- 各种范例应用。
- 文档。

目前提供了支持 S60 3rd Edition 及之前各个版本的 SDK。2006 年下半年有望发布针对 S60 3rd Edition, Feature Pack 1 的 SDK。

请注意，需要 Carbide.vs 才能使用 Microsoft Visual Studio .NET 2003。

### 6.3 S60 平台有哪些 Java™ 开发工具？

可以从诺基亚论坛网站工具和SDKs专区 ([www.forum.nokia.com/tools](http://www.forum.nokia.com/tools)) 下载S60 移动信息设备描述 (MIDP) SDK。该SDK 包括：

- S60 模拟器。
- Java™ 类库。
- API 集 (包括 Nokia UI API)。
- 文档。

这个工具包兼容 IBM 公司、Sun Microsystems 公司，及 Borland 软件公司各种专业的 Java 开发环境。最近版本的一些 SDK 还支持 NetBeans 和 Eclipse 集成开发环境 (IDE)。

Borland 和 Sun 公司的 IDE 可以与 Carbide.j 结合使用。Carbide.j 也是使用 Eclipse 的先决条件。这个工具能接受针对各款获支持终端的 SDK 插件，也包括 S60 MIDP SDK。

### 6.4 内容开发伙伴们可使用哪些工具？

诺基亚移动互联网工具包 (Nokia Mobile Internet Toolkit, NMIT) 提供了一些工具，用于让开发伙伴们创建及测试浏览和消息应用。在众多的NMIT工具中有一个浏览器参考实现。更多详情请访问诺基亚论坛网站的工具和SDKs专区 ([www.forum.nokia.com/tools](http://www.forum.nokia.com/tools))。

消息应用开发伙伴可使用诺基亚多媒体消息服务 (MMS) 开发伙伴套件，而开发 WAP/XHTML和各种推送应用则应使用NMIT。使用诺基亚多媒体消息服务中心 (Nokia Multimedia Messaging Service Center, MMSC) 及从事服务器实施项目的开发伙伴们还可以使用另一资源，这就是Nokia Mobile Server Services API 及库。所有这些工具都可从诺基亚论坛网站工具和SDKs专区 ([www.forum.nokia.com/tools](http://www.forum.nokia.com/tools)) 免费下载。

我们为多媒体项目提供了诺基亚音频套件，这是面向 SP-MIDI (可扩展和弦) 铃音开发伙伴的工具。

### 6.5 所有诺基亚工具都对全体开发伙伴开放吗？

诺基亚提供让开发伙伴加入诺基亚论坛PRO的机会。诺基亚论坛PRO计划的优点之一就是：PRO会员能率先获得各种新工具和已升级工具。一旦诺基亚论坛PRO的早期介入期结束，所有工具都可通过诺基亚论坛网站工具和SDKs专区 ([www.forum.nokia.com/tools](http://www.forum.nokia.com/tools)) 下载。

## 7. 为 S60 平台开发

### 7.1 为何 S60 平台可作为移动应用开发的初始目标？

自第一批智能手机出现以来，设计的多样化和新技术的采用都大大拓展了移动终端的市场。自 2002 年诺基亚 7650 推出起，S60 平台已经占据了市场领先地位，众多 S60 获授权厂商也随之推出了品种丰富的新终端及新发布。

基于两个重要原因，S60 平台成为移动应用开发的理想起点。第一，它是智能手机市场中的领先平台，保有大量的终端。因此，各种新应用所面对的是一个既有市场，拥有成百上千万习惯于经常使用这些销售渠道的消费者。

第二，为 S60 平台开发的软件高度可移植。例如，把专门为 S60 平台设计的应用迁移到其它诺基亚平台相对就比较容易。另外还有一些基于 Symbian OS 的平台，如 UIQ，只要对 S60 应用的用户界面进行修改就能将其移植到这类平台，无论是本地应用（C++ 编制）还是 MIDlet 莫不如此。诺基亚论坛网站移植专区（[www.forum.nokia.com/porting](http://www.forum.nokia.com/porting)）提供了一些文献，专门解释如何从 S60 平台向这些平台（也包括 Microsoft Pocket PC 和 Palm OS 等）进行移植。

### 7.2 在各个系列的 S60 终端上对应用进行优化有哪些相关问题？

S60 平台终端的多样性主要源于用户界面、平台 editions、feature packs，及导引软件等。

S60 平台并没有在其规范中强制规定某种特定的用户界面解析度、显示布局，或输入方式等（尽管用户界面样式是该平台的一部分）。因而开发伙伴们应考虑创建具备灵活性和可扩展性的应用，并尽可能使用如 feature discovery API 那样的特性，。

S60 平台的前两个 edition 之间存在着二进制兼容方面的一些已知问题。（详细描述请参阅第 8 章，“一些已知问题”。）显然，为 S60 3rd Edition 编制的应用可能用到 S60 2nd Edition 和 S60 1st Edition 中并不存在的一些特性。如果开发伙伴们能熟练掌握各个 edition 中的可用技术，且设计的应用能应对这些特性差异，那么就可以避免大部分的问题。

开发伙伴们可能希望使用某些导引软件提供的功能，它们可能并不适用于整个 S60 终端系列。使用这些导引软件时，开发伙伴们应该进行核查以确保部署其应用的终端已经实现了这些功能。这当然也会限制某个应用在整个目标市场上的覆盖范围。

S60 3rd Edition 的面世及其与之前 edition 之间的二进制兼容断点问题显然对那些想同时利用新款终端和现有终端的开发伙伴们形成了问题。然而，经过仔细设计的应用只需使用不同的工具链进行简单重编译就能应付这类问题。

### 7.3 向其他诺基亚平台迁移应用时会有哪些相关问题？

诺基亚还有另外两种平台，即 Series 40 平台和 Series 80 平台。同时，公司还有一款基于 Symbian OS 的诺基亚 7710 宽屏智能手机。它们都能运行基于 Java™ MIDP 的应用。除 Series 40 平台以外，它们都基于 Symbian OS，能够运行以本地 C++ 语言编制的应用。请注意，7710 智能手机的特性将于 2006 年被完全纳入到 S60 平台中。

希望在这些平台间移植应用的开发伙伴们将遇到的一些主要问题是：用户界面样式和底层 API 之间的差异、内存资源限制方面的不同（特别是 Series 40 终端），及各种实现技术的变化（请参阅 [www.forum.nokia.com/platforms](http://www.forum.nokia.com/platforms) 平台页面）。还需注意的是，在一些现有的终端实现中，Series 80 终端和诺基亚 7710 智能手机的横向显示屏布局和各种 Series 40 终端上的竖向显示屏布局是不同的。S60 平台的一些早期 edition 也使用竖向显示屏布局的用户界面，但是从 S60 3rd Edition 起，对横向显示屏布局的支持也被添加到某些解析度的屏幕上。这意味着，在 Series 80 终端和诺基亚 7710 智能手机上为该宽度所设计的用户界面可能需要

被彻底重设计，以便能成功地运行于 S60 终端和 Series 40 终端。如要求同一应用在不同的终端版本上保持高度的一致性，那么在用户界面设计的开始阶段就必须考虑到这些因素。

#### 7.4 向其他基于 Symbian OS 的终端移植应用时会有哪些相关问题？

基于 Symbian OS 的所有终端都能运行以本地 C++ 和 Java™ MIDP 编写的应用，因此可以期望一定程度的兼容性。各种差异的最常见源头是用户界面，但开发伙伴们也应核查功能方面的差异和技术差异。

#### 7.5 向移动市场上各非相关操作系统移植应用时会有哪些相关问题？

要确保相对便利的跨平台兼容性，比较自然的选择就是用 Java™ MIDP 开发应用，但是这取决于具体应用所需要的功能。

以 Symbian OS C++ 语言编制的应用很可能需要针对使用其它操作系统的平台而完全重写。然而，由于 Symbian OS 支持以 ANSI C 开发，所以有可能为 S60 平台编写一个独立于操作系统的应用引擎，然后将这个引擎用于为其它操作系统开发应用的基础。这种方法需要开发伙伴们理解所有目标平台的局限性，并创建一个能包容所有这些局限性的引擎。

应该注意的是，有一些工具能用标准 C++ 在诸如 Palm OS 这类平台上编写应用程序，但并没有获得基于 Symbian OS 的终端的支持。

## 8. 一些已知问题

### 8.1 S60 应用开发伙伴应注意哪些已知问题？

诺基亚致力于尽力维护 S60 平台不同 edition 之间及各种平台之间的兼容性。

如果无法满足兼容，诺基亚将维护并发行有关问题的记录，如API 实现方面的细微差别或技术实现方面的某些问题。这份记录，以及详细的技术解决方案（主要包括由诺基亚论坛专业支持团队解答的开发伙伴问题）都能在诺基亚论坛网站中的诺基亚论坛技术库

（[www.forum.nokia.com/library](http://www.forum.nokia.com/library)）中找到。诺基亚论坛技术库以三种格式发布，它们分别是：Microsoft HTML Help、WebHelp，及PDF格式。

## 9. 业务案例

### 9.1 使用 S60 平台的业务案例是什么？

S60 平台是智能手机市场无可争辩的领导者，从而使该平台成为各种第三方应用及内容开发伙伴的收益来源。S60 3rd Edition 显著地拓展了业务机会，并向行业各方提供了确保长期稳定性的增强性设计。以企业、游戏、音乐、视频、个人效率增强，及其他方面为目标的一系列功能和应用为各个细分市场都提供了巨大的机会。

S60 平台让开发伙伴共同体能使用各种行业标准技术，并让他们有机会进入拥有成百上千万用户的市场。这类机会将超越 S60 终端，因为该平台提供了一些标准技术，允许开发伙伴们构建针对 Series 40 终端、Series 80 终端、诺基亚 7710 宽屏智能手机，以及其它制造商所提供的终端的应用和内容。

S60 平台获得了诺基亚论坛网站 ([www.forum.nokia.com](http://www.forum.nokia.com)) 和许多工具公司所提供的开发工具和文档方面的完善支持。既有的销售渠道则帮助开发伙伴们尽可能快捷方便地收回其投资。

三个原因吸引开发伙伴们将 S60 平台用作移动应用及内容开发的基础。

第一，使用这些平台的终端用户也是这些应用和内容的重要消费者。

第二，仅诺基亚一家就已供应了 5000 多万台 S60 终端（截至 2006 年 2 月），而 2006 年的智能手机市场还有望显著增长。已经有 30 多款 S60 终端上市，且全世界大部分运营商都在销售 S60 终端。S60 平台拥有四家获授权厂商，代表着重要且不断成长的潜在客户市场。

第三，这种平台解决方案能让开发伙伴们为一种终端或一个系列的终端创建内容，然后通过最小的附加开发工作量将这些内容优化为适用于其他使用了这些平台的终端。由此使开发伙伴们只需在优化和移植方面增加少量的投资就能使其应用或内容获得成倍的潜在市场。

## 10. 将应用投入市场

### 10.1 诺基亚如何帮助我的应用进入市场？

诺基亚以多种方式将移动应用开发伙伴与市场连接起来。

**Nokia Content Discovery** 客户端是一种驻留于终端的内容门户，让消费者能方便地发现、下载，及购买精彩内容 and 应用。该客户端能与多种内容发布系统集成，帮助运营商最大化其移动应用和内容销售。

诺基亚还向开发伙伴们提供了通过诺基亚销售渠道（诺基亚软件市场和诺基亚应用目录）销售移动内容和 **Java Verified™** 应用的良好机会。另外，还有可能选择一些应用，将其预装到新款诺基亚终端中。

更详细信息请访问诺基亚网站开发伙伴专区（[www.nokia.com/developers](http://www.nokia.com/developers)）上的市场链接。

## 11. 请对本文进行评价

请花一点时间对[本文进行评价](#)，以帮助我们改善文档质量，也让我们了解您所发现的最有价值的资源。