
S60 5th Edition: What's New for Developers

Version 1.0
October 2, 2008

S60 platform

Legal notice

Copyright © 2008 Nokia Corporation. All rights reserved.

Nokia and Nokia Connecting People are registered trademarks of Nokia Corporation. Java and all Java-based marks are trademarks or registered trademarks of Sun Microsystems, Inc. Other product and company names mentioned herein may be trademarks or trade names of their respective owners.

Disclaimer

The information in this document is provided “as is,” with no warranties whatsoever, including any warranty of merchantability, fitness for any particular purpose, or any warranty otherwise arising out of any proposal, specification, or sample. This document is provided for informational purposes only.

Nokia Corporation disclaims all liability, including liability for infringement of any proprietary rights, relating to implementation of information presented in this document. Nokia Corporation does not warrant or represent that such use will not infringe such rights.

Nokia Corporation retains the right to make changes to this document at any time, without notice.

License

A license is hereby granted to download and print a copy of this document for personal use only. No other license to any other intellectual property rights is granted herein.

Contents

1.	Introduction	5
2.	New platform features.....	5
3.	For Symbian C++ developers.....	5
4.	For Open C/C++ developers	7
5.	For Java™ developers	7
6.	For Web and widget developers	8
7.	For Flash Lite developers.....	9
8.	For Python developers	9
9.	For media and content developers.....	9
10.	Evaluate this resource	10

Change history

October 2, 2008	Version 1.0	Initial document release
-----------------	-------------	--------------------------

1. Introduction

This document is designed to provide developers familiar with the S60 platform up to and including S60 3rd Edition, Feature Pack 2 with a compact guide to the new features offered in S60 5th Edition.

Developers who are unfamiliar with the S60 platform are recommended to read the [S60 Platform: Developers' Introductory Guide](#) white paper. This document can be found on the [S60 documents](#) page of the Forum Nokia Web site.

2. New platform features

S60 5th Edition is built on Symbian OS v9.4. This upgrade of the underlying OS provides improvements to demand paging so that this feature is used for all items in a device's internal memory. This offers faster device boot-up and application startup times, and reduces the likelihood of out-of-memory situations.

The S60 UI has been updated to support 640 x 360-pixel nHD (ninth of 1080p High-Definition) screens in addition to the existing 240 x 320-pixel Quarter Video Graphics Adapter (QVGA) screen-resolution support.

With the nHD screen, S60 5th Edition delivers a touchscreen capability with tactile feedback through the device's vibrate function. This enables device users to interact with their devices in a more natural and expressive way. The touch capability uses a select-activate paradigm and offers gesture support for tasks such as panning within images or Web pages. To support touch interaction, S60 5th Edition delivers on-screen ITU-T and full keyboards, as well as handwriting-recognition input.

A new sensor framework allows for the inclusion of accelerometers, magnetometers, and tap sensors. These sensors enable the creation of applications that respond to a device's motion and orientation. The framework is also extensible, so that support for additional sensor technology can be added to the S60 platform or to devices by S60 licensees.

The UI enhancements in S60 5th Edition bring many new opportunities to developers. Additional screen real estate and direct-touch access to application elements enable developers to extend the usability of their applications. To enable the most expressive use of touch, S60 5th Edition delivers new UI components such as a toolbar that allows key application functions to be accessed directly by touch.

Device users also benefit from a number of enhancements to the platform's built-in applications. These benefits include new features in the camcorder application, such as a scene mode and a self-timer. The image and video editors have been rewritten to make editing easier and faster. A new search application has been added to help users locate information on their devices, a particularly useful feature now that devices can contain multiple internal-storage locations and memory cards, each with a capacity of up to 32 GB.

3. For Symbian C++ developers

The following new APIs are delivered to Symbian C++ developers in S60 5th Edition:

- *Accessory Monitoring API* — This API enables applications to determine the accessories, such as headsets, car kits, desk stands, loopsets, TTYs, and audio-visual devices, attached to an S60 device, including the accessory's

type, connection, and capabilities. The API also enables the monitoring of accessory connection and disconnection.

- *Choice List API* — This new UI component API enables the selection of choices from a vertical list.
- *Generic Button API* — This API allows applications to draw an arbitrary button with an icon or text label and set its characteristics, such as dimmed or pressed.
- *Hierarchical Lists API* — This API allows data to be presented in a hierarchical list using text and graphics. Three list types are provided: a general-purpose hierarchical tree list, a single-column style list that allows for node expansion without indentation, and a single-style tree to represent a folder and file structure. Features also are provided for applying a custom order to the list and a list-events observer. This API was published in the S60 3rd Edition SDK for Symbian OS, Feature Pack 2 but was not implemented on any devices until S60 5th Edition.
- *Hostlet Connection API* — This API enables the creation of and communication with an application that provides Web services on an S60 device, which is identified by a unique endpoint and/or a contact URI.
- *Incoming Call Monitor API* — This API enables an application to define itself as the handler application for incoming calls.
- *Messaging Integration API* — This API enables third-party developers to better integrate their Message Type Module (MTM) services into the S60 Messaging application and the Send UI API.
- *Sensor APIs* — A collection of new APIs has been provided to allow applications to connect to and receive data from various sensors on an S60 device. The SensorChannel API provides the mechanism by which an application connects to a sensor, while the Sensor-Definition API provides the definitions of the data structures used to pass sensor-specific information through the channels. Specialized classes then provide for listening to and receiving meaningful data from sensors designed to detect acceleration, orientation, magnetic fields, and tapping.
- *Stylus Pop-up Menu API* — This UI API provides an application with the ability to draw a pop-up menu containing a single list of options, at the point where a tap on a touchscreen is detected.
- *Tactile Feedback Client API* — This API provides a mechanism for applications and UI controls to register their tactile-feedback areas and receive notification of touch events.
- *Title Pane Touch Observer API* — This API allows an application to observe the user tapping on the title pane. An example use for this API would be to restore a default application view when the user taps on the title pane.
- *Toolbar API* — This API allows an application to construct a fixed or floating toolbar within an application view. The toolbar is constructed from AVKON buttons and can contain up to three buttons (although a floating toolbar displayed in the full-screen view can contain four buttons). Each button should provide an application-specific function. Toolbar buttons can be view-specific if required.
- *Touch UI Utilities API* — Currently a single utility is provided, to detect a long tap on the touchscreen.

- *Web Service Messaging APIs* — These APIs extend basic Simple Object Access Protocol (SOAP) messaging features to enable the setting of a security header and security token in any message. A new abstract interface that defines a set of Web services framework message classes is also provided.
- *WLAN SDK Info API* — This API enables a developer to read the MAC address of a device's WLAN.
- XML Engine Document Object Model (DOM) APIs — These APIs enable the manipulation of a DOM.
- XML Fragment API — This API implements Web-services-framework-specific adaptations of the standard XML APIs. These new fragment classes provide an efficient way to build an XML Engine-based DOM tree from Simple API for XML (SAX) parsing output. Such a fragment can be described as a specialized combination of both a SAX parser and content handler.

In addition, the Multiple Drive Support APIs, introduced in S60 3rd Edition, Feature Pack 2, are integrated into the S60 5th Edition SDK for Symbian OS.

More information on the S60 5th Edition features for Symbian C++ developers can be found in the [S60 5th Edition C++ Developer's Library](#), available from the [Symbian/C++ Documentation](#) section of the Forum Nokia Web site.

4. For Open C/C++ developers

S60 5th Edition delivers the Open C++ APIs (Standard Template Library [STL] and the IOStreams and Boost libraries) as part of the platform. This removes the requirement to include the Open C++ SIS file in the installation package of any application using these standard C++ APIs.

More information on the S60 5th Edition features for C/C++ developers can be found in the [S60 5th Edition C++ Developer's Library](#), available from the [Symbian/C++ Documentation](#) section of the Forum Nokia Web site.

5. For Java™ developers

Java™ application developers gain the features of the touchscreen that are inherently supported within the Java APIs. This includes full use of the on-screen keyboard and appropriate tactile feedback for controls. Generally, these features should be compatible with existing Java applications, but developers may also want to consider what additional design opportunities these new UI features offer their applications.

Where an application uses the standard high-level UI components, the Screens and Items pointer events (created when a user taps on the touchscreen) are handled automatically. Where a UI has been implemented using `Canvas`, the pointer events must be handled programmatically, using three standard MIDP LCDUI methods:

`pointerPressed()`, `pointerDragged()`, and `pointerReleased()`. These features are supported for the new nHD screen resolution, in both portrait and landscape modes.

Where an application cannot be redeveloped to take full advantage of the nHD screen resolution, three JAD attributes provide for scaling the UI to fit the nHD screen: `Nokia-MIDlet-Canvas-Scaling-Orientation-Switch`, `Nokia-MIDlet-Original-Display-Size`, and `Nokia-MIDlet-Target-Display-Size`. For more information, see the [Java ME Developer's Library](#).

In addition, the attribute `Nokia-MIDlet-On-Screen-Keypad` provides for defining an on-screen keypad.

The MIDlet on-screen keypad offers backwards-compatibility for MIDlets that use `Canvas` and that originally were not designed for touchscreen devices with a limited number of physical keys. The on-screen keypad takes part of the screen from `Canvas` and generates key events for the `Canvas` application. Possible values for `Nokia-MIDlet-On-Screen-Keypad` are `no` (no keypad), `gameactions` (providing up, down, left, right, fire, `game_a`, `game_b`, `game_c`, and `game_d` keys), and `navigationkeys` (providing up, down, left, right, and fire keys). Where a `Canvas` is in full-screen mode, left and right softkeys also are provided.

In addition, S60 5th Edition offers Java developers version 1.1 of Advanced Multimedia Supplements (JSR-234).

6. For Web and widget developers

Web and Web Runtime (WRT) widget developers benefit from the UI enhancements in S60 5th Edition, with more screen real estate and touch interaction enabling users to more easily and intuitively navigate Web pages or widgets. The on-screen keyboard also is available for text and data entry.

In addition, WRT has been integrated with S60 Platform Services in S60 5th Edition. This means that widgets can access device data and information through new JavaScript™ extensions. These extensions provide access to:

- The application manager.
- Calendar records.
- Contacts records.
- Log information.
- Short message service (SMS) and multimedia messaging service (MMS) messaging.
- The media gallery.
- Device location.
- Landmarks.
- System information.
- Sensors.

Using these new APIs, widgets are now able to access information from a device's Calendar and Contacts applications, enabling mashups with information from the Internet to create new, personalized service experiences. Using location estimates, WRT widget developers are also able to provide location-aware information to users.

Together, the UI and platform-access enhancements enable Web developers to create a completely new breed of context-aware applications and services that provide users with information unique to their own experiences.

More information on the S60 5th Edition features for Web developers can be found in the [Web Developer's Library](#), available from the [Web Technologies Documentation](#) section of the Forum Nokia Web site.

7. For Flash Lite developers

Developers of Flash Lite from Adobe applications benefit from the UI enhancements in S60 5th Edition, with more screen real estate and touch interaction enabling users to more easily and intuitively navigate Flash Lite content and applications. The on-screen keyboard also is available for text and data entry.

S60 5th Edition continues to support Flash Lite 3.0, with the addition of ActionScript extensions to take advantage of S60 Platform Services. These extensions provide access to:

- The application manager.
- Calendar records.
- Contacts records.
- Log information.
- SMS and MMS messaging.
- The media gallery.
- Device location.
- Landmarks.
- System information.
- Sensors.

These new APIs enable Flash Lite developers to access device-application data, such as that held in Calendar and Contacts, as well as information on a device's location.

Together, the UI- and platform-access enhancements enable Flash Lite developers to create a completely new breed of context-aware applications and services that provide users with information unique to their own experiences.

8. For Python developers

Python for S60 applications are able to make use of the UI enhancements in S60 5th Edition, with more screen real estate and touch interaction enabling users to more easily and intuitively navigate Python scripts and applications. (Note that Python does not support the processing of raw touch events.) The on-screen keyboard also is available for text and data entry.

9. For media and content developers

Media and content developers benefit from the UI enhancements in S60 5th Edition. The additional screen real estate offered by the nHD screen will allow a richer display of graphics, images, and video. The touch interaction will provide users with a more intuitive approach to navigation through media.

Selected devices may also provide support for Windows Media (WM) DRM 10, WM Audio 9, and WM Video 9 codecs.

10. Evaluate this resource

Please spare a moment to help us improve documentation quality and recognize the resources you find most valuable, by [rating this resource](#).