Nokia Smart Installer for Symbian Developer's Guide

Version 1.0; June 17, 2010

Smart Installer

NOKIA

Contents

Change history

| February 15, 2010 | Version 1.0 beta | Initial document release |
| June 17, 2010 | Version 1.0 | Updated for Smart Installer v1.0 release |

1 Introduction

This document describes how developers can use Nokia Smart Installer for Symbian when packaging Qt-based applications for Symbian devices.

1.1 A brief overview

Nokia Smart Installer for Symbian (or Smart Installer, for short) is a tool that ensures that the installed 3rd party applications have always access to the required versions of their dependent libraries.

Smart Installer is split into two components: smartinstaller and smartinstaller ADM (for 'application dependency manager'). Smart Installer is a small component that will be embedded with the application to a wrapper package; it ensures that a recent version of ADM is installed on the device and installs/upgrades ADM, if necessary.

Because of the nature of Smart Installer functionality, network connection is always required during installation. See Section 1.2, 'Smart Installer network usage', for a detailed description about network usage.

Smart Installer is invoked during the application installation process on the Symbian device. It immediately fetches the latest ADM from the server and runs it. ADM checks whether the required components (dependencies) are already installed on the device and downloads and installs, or upgrades them, if needed. Information about the application's dependencies is extracted from the application sis file, where each dependency is defined using the standard dependency statements as defined in the pkg file format. For Qt, the dependencies are automatically generated during the build process, but developers can manually adjust them as well.

After the required dependencies have been successfully installed to the device, the application installation process is invoked. This follows the normal UI flow of application installation on a Symbian device.

1.2 Smart Installer internet access and network usage

Smart Installer tries to automatically select a proper internet access point. If the access point cannot be determined automatically, the user will be prompted for one. By default Smart Installer uses the browser access point settings to access the internet. If the installation started from the Ovi Store client, Smart Installer uses the access point used by the Ovi Store. Also, the user is always prompted the download size, if the installation is started from the Ovi Store. Otherwise no prompt is shown if WLAN is selected as the access point. For packet data connections, the user is always prompted about the download size.

Smart Installer will download the required packages from a server provided by Nokia at dl.qt.nokia.com. Because a data connection (WLAN, general packet radio service [GPRS], EDGE, 3G, and so on) is used, transmission fees may apply to the individual installing the Smart Installer-based application.

If the dependencies are met — that is, Qt and the necessary components are already on the device with the required versions — then the data downloaded from the server will be on the order of several kilobytes (the latest ADM, if one is available, and a few dependency definition files for the components involved). If all dependencies need to be installed, then the amount of data to download is currently about 12 MB. See Section 1.3 'Download and space requirements during installation', for a detailed description about the amount of data.

1.3 Installation

The following sections describe in detail, how Smart Installer installs the dependencies and the application.

1.3.1 Download and space requirements

Due to performance reasons the Qt libraries are always installed to C: drive of the device. Other packages are installed to other drives, if required. Therefore the minimum space required on C: drive before starting the installation is about 10 MB (up to 14 MB, if QtWebkit and/or QtMobility are also needed). The free space requirements are larger than explained in the table, because the S60 application installer requires temporarily some additional space during package installation. Smart Installer will show an error message, if the space requirements cannot be met. See Table 3 'Smart Installer messages' for details.

For other packages than Qt, Smart Installer chooses the next available drive(s), if drive C: would become full. The user cannot control, where the packages are going to be installed as the process of determining the installation drive is automatic to reduce the unnecessary user interaction.

Table 1 shows how much each package consumes space after installation from the device at the time of this document's publication.

The packages indicated with a light-gray background are the minimum set required for a Qt application to run in a Symbian device. For instance, even though the Open C libraries are preinstalled in some Symbian devices it still needs to be updated to v1.6, so these numbers represent the common scenario from the space-requirement point of view.

Table 1 also describes the dependencies that Smart Installer can provide at the present time. For updated list, please refer to http://wiki.forum.nokia.com/index.php/Nokia_Smart_Installer_for_Symbian#Packages_distributed_by_Smart_Installer.

Package name	UID	Versions	Download size	Space required	Description
Stdcpp	0x2000F866	1.6.0	0.2 MB	0.2 MB	Standard C++ library Common
Openc_ssl	0x200110CB	1.6.0	0.8 MB	0.8 MB	Open C LIBSSL Common
PIPS	0x20013851	1.6.0	1.2 MB	1.4 MB	PIPS Installer
Sqlite	0x2002AF5F	0.5.0	0.2 MB	0.1 MB	Sqlite library
Qt	0x2001E61C	4.6.3	5.8 MB	5.9 MB	Qt libraries
QtWebKit	0x200267C2	4.6.3	2.9 MB	3.0 MB	QtWebKit library
QtMobility	0x2002AC89	1.0.0	1.0 MB	1.0 MB	QtMobility library
Total for Qt excluding QtWebKit			**8.3 MB**	**8.5 MB**	
Total for Qt including QtWebKit			**12.1 MB**	**12.4 MB**	

Table 1: Supported dependencies and their space requirements

1.3.2　Application installation

When Smart Installer is invoked from the Ovi Store, the application will be installed to the drive selected in the Ovi Store as the installation target drive. It defaults to the drive with largest available space, usually the internal flash drive (eMMC) or a memory card. User can change this behavior in the Ovi Store settings.

Otherwise Smart Installer obeys the developer chosen installation drive defined in the wrapper package. See Section 2.1 'Wrapper package file' for details.

1.3.3　Resuming of an interrupted installation

Smart Installer supports resuming of cancelled or interrupted installation. To resume interrupted installation, choose the Smart Installer icon from the application menu grid. Smart Installer icon should be visible only, when there is a pending installation available.

	Smart Installer icon

Note: Only the latest interrupted installation can be resumed.

1.4　Smart Installer and Qt build tools

The Qt build chain supports Smart Installer from Qt 4.6.2 for Symbian onwards.

Note: It is recommended that Qt 4.6.3 or newer is used when creating the applications.

1.5　Incompatible devices

There are known issues with Open C upgradeability on some devices; this makes Smart Installer unsuitable for deploying Qt or any other package that require an Open C upgrade.

If a user tries to install to these devices, the error note 'Device not supported' is shown, and the installation is cancelled.

Table 2 lists devices that are known to have problems and are not supported by Smart Installer. For up-to-date list, please refer to http://wiki.forum.nokia.com/index.php/Nokia_Smart_Installer_for_Symbian#Supported_Devices.

Device	Incompatible firmware versions	Description of problem
Nokia 6650 AT&T variant	All	PIPS cannot be upgraded.

Table 2: Incompatible devices

1.6　Smart Installer messages

An informative message indicates if a problem has occurred during installation. Table 3 describes the messages and when they appear.

Message	Description
Verifying components	Smart Installer is checking to see which components the application requires to be installed, and which components are on the device.
Downloading X of Y	Required dependency X is being downloaded out of the total number of Y. Time required depends on network connection speed.
Installing X of Y	Required dependency X is being installed out of the total number of Y. Time required depends on phone speed.
Download XX MB?	The prompt the user gets when Smart Installer has determined which components to download. If user denies download, the installation is cancelled.
Phone is roaming. Download XX MB anyway?	User is warned that device is roaming, and additional data charges may apply. User needs to confirm this in order to continue the installation.
No pending installations	Smart Installer was opened from the application menu, but no pending installations were found.
Installation failed!	Unrecoverable installation failure has occurred. Installation can be resumed later by selecting the Smart Installer icon from the Application menu.
Installation cancelled!	User has cancelled the installation. Installation can be resumed later.
Device is not supported	Device is known to have problems with Smart Installer and installation is not allowed to continue. See Table 2 for list.
Download failed!	Unrecoverable error has occurred during downloading. Check network connectivity and coverage. Installation can be resumed later.
Unsupported Smart Installer version	Incompatible version of Smart Installer is in use. Software distributor must update the package. Installation cannot continue.
Cancelling installation	Installation was cancelled or interrupted due to an error. Smart Installer is removing any downloaded files. Installation can be resumed later.
Not enough space to install	Device does not have enough space to download and install the required components. Free space on the destination drive and resume the installation by selecting the Smart Installer icon from the Application menu. Space requirements for drive C: are described in Table 1.

Table 3: Smart Installer messages

1.7 About this version

Version 1.0 is the 2nd release of the documentation.

2   Smart Installer packaging

Smart Installer is wrapped with the application sis file to a wrapper sis package. This is needed in order to ensure successful installation of the dependencies before the application is installed.

An example of the wrapper package file is explained in Section 2.1, 'Wrapper package file'. Signing options for the wrapper package are explained in Section 2.2, 'Wrapper package signing', and adding the dependency information to the application is explained in Section 2.3, 'Application dependencies'.

The qmake build tool will automatically create the necessary files to the source directory. The make tool can be invoked to automatically create the Smart Installer wrapped package. Examples of the command flows are explained in Section 2.4, 'Building a self-signed Smart Installer package' and Section 2.5, 'Building a Symbian Signed Smart Installer package'.

The UIDs to be used with Smart Installer are explained in Section 2.6, 'Reserved UIDs for Smart Installer'.

2.1   Wrapper package file

Here is an example of the wrapper package file.

If a developer wants to deploy the wrapper package as a self-signed sis, then an appropriate, free UID from the proper UID range must be obtained. The normal restrictions apply about which UIDs are available for self-signed packages. Qmake uses a fixed UID (0xA000D7CE) when creating a self-signed wrapper package.

Nokia has reserved a UID to be used with the wrapper package (0x2002CCCF) when submitting the package to Symbian Signed (see Section 2.2, 'Wrapper package signing' and Section 2.5, 'Building a Symbian Signed Smart Installer package'). The wrapper package is used only temporarily to transmit the application with Smart Installer to the device, and it is deleted after installation, so a fixed UID can be used for the wrapper package.

Normal rules apply in the installation target drive selection: if the drive is "!", the user will be prompted for the drive, otherwise the specified target drive is used automatically without user interaction.

```
; The package header
&EN
#{"Application Installer"}, (0x2002CCCF), 1, 0, 0, TYPE=SA
%{"Vendor-EN"}
:"Vendor"

;Supports S60 v3.1
[0x102032BE], 0, 0, 0, {"SymbianProductID"}
;Supports S60 v3.2
[0x102752AE], 0, 0, 0, {"SymbianProductID"}
;Supports S60 v5.0
[0x1028315F], 0, 0, 0, {"SymbianProductID"}

; The actual application.sis file (required)
"application.sis"-"!:\private\2002CCCE\import\application.sis"

; Embedded Smart Installer (required)
@"smartinstaller.sis",(0x2002CCCD)
```

> **Note:** This file is created automatically by qmake, and it is overwritten each time qmake is run.

> **Note:** It is not recommended that developers change the qmake-created package file directly. If needed, developers should change only the **_bold-italic_** parts of the package file shown above.

2.2   Wrapper package signing

The wrapper package can be self signed, but a warning note will be displayed on the device stating that the source of the package is not trusted, as is always the case with self-signed applications on Symbian devices.

In order to assign trust to the wrapper package, the wrapper can be put through the Symbian Signed Express Signed service, which offers a low-cost self-certification service. Unfortunately, standard signing fees will apply to developers wanting to sign the wrapper package, but work is being done to remedy this situation.

> **Note:** Use the provided fixed UID (0x2002CCCF) when submitting a wrapper package to Symbian Signed.

2.3   Application dependencies

Qmake automatically generates package files (*.pkg) with correct dependencies for the version of Qt the application is built against. Developers can add additional dependencies using the deployment keyword in the `.pro` file (see Qt documentation for details).

> **Note:** All normal Qt for Symbian applications have their dependencies assigned automatically by the build tools. Smart Installer will take care of any dependencies that Qt libraries might have during installation. There is no need to specify any other dependencies unless your application is using the APIs provided by the package(s) in question.

> **Note:** Currently Smart Installer can provide only the dependencies described in Table 1.

2.4   Building a self-signed Smart Installer package

This section explains how developers can create a self-signed Smart Installer wrapped sis (it is assumed that developers are already familiar with Qt build tools and the build process).

`Application` is the target application name defined in the `.pro` file.

Invoke the qmake tool for creating the required files needed for the build process:

```
C:\Sources\Application> qmake
```

For building the application release version with the GCCE compiler, use the following command:

```
C:\Sources\Application> make release-gcce
```

The application has been compiled after successful completion of the previous command. The last thing to do is:

```
C:\Sources\Application> make installer_sis
```

Now the directory contains a self-signed `application_installer.sis`, which is the Smart Installer wrapped version of the `application.sis`.

2.5 Building a Symbian Signed Smart Installer package

Building a Symbian Signed Smart Installer package requires additional information to be passed for the tools in order to use the proper key to sign it.

`Application` is the target application name defined in the `.pro` file.

> **Note:** Commands need to be entered on a single line.

Invoke the qmake tool for creating the required files needed for the build process:

```
C:\Sources\Application> qmake
```

For building the application release version with the GCCE compiler, use the following command:

```
C:\Sources\Application> make release-gcce
```

Next, the sis file must be created and signed with the proper keys. For Symbian Signed certification submission these keys would have to be the Publisher ID. The example assumes that the Publisher ID key and certificate are in the current directory.

```
C:\Sources\Application> make sis QT_SIS_CERTIFICATE=publisherid.cer
QT_SIS_KEY=publisherid.key
```

> **Note:** Any valid path can be used to point to the key; it does not need to be in the current directory.

At this point, the developer needs to submit the created sis file (application.sis) to Symbian Signed fir certification. Depending on application's specific needs and properties (e.g. Platform Security capabilities used) you may use either the Express Signed or the Certified Signed path.

> **Note:** Please ensure that your application complies with the Symbian Signed Test Criteria before submitting the file for certification. Also, if the file is intended for Ovi Store publishing, verify that the application complies with Ovi Store's publishing requirements.

After receiving the sis from Symbian Signing, the developer must copy the signed package from Symbian Signed over the old `application.sis`. Then the signed wrapper package can be created:

```
C:\Sources\Application> make installer_sis
QT_SIS_CERTIFICATE=publisherid.cer QT_SIS_KEY=publisherid.key
```

The created wrapped sis file `application_installer.sis` must also be submitted to Symbian Signed. Express Signed is a suitable signing option for the wrapper package.

2.6 Reserved UIDs for Smart Installer

Nokia has reserved the following UIDs to be used with Smart Installer:

UID	Description
0x2002CCCD	Smart Installer UID. To be used in the wrapper package when including the smartinstaller.sis.
0x2002CCCF	Wrapper package UID when providing package to Symbian Signed.
0xA000D7CE	Reserved UID when creating a self-signed wrapper package using Qt tools.