



# Symbian Signed Test Criteria

**Security Classification:** Public

**Status:** ISSUED

**Version:** 3.0.3

**Last Revised Date:** 27<sup>th</sup> November 2008

**Valid From:** 14<sup>th</sup> December 2008

- 1 Introduction .....3
- 2 Useful Links and Tools .....4
  - 2.1 Recommended Reading .....4
  - 2.2 Useful Tools .....4
- 3 What Tests do I Need to Run? .....5
  - 3.1 Applications for pre Symbian OS v9.x (S60 2nd Edition, UIQ 2.x) .....6
  - 3.2 Applications for Symbian OS v9.x and later (S60 3rd Edition, UIQ 3.x) .....6
  - 3.3 Further Testing requirements for System Capabilities; AllFiles, DRM, TCB.....6
  - 3.4 Exceptions.....7
  - 3.5 Other Non-Standard Submissions .....7
  - 3.6 Eligibility for *Express Signed*.....7
  - 3.7 Declarative Statements .....7
- 4 Submission Criteria .....8
- 5 Symbian Signed Test Cases .....13
  - 5.1 Universal Tests.....13
  - 5.2 Capability Related Tests .....32
- 6 Appendix 1.....41
  - 6.1 Waivers .....41
  - 6.2 Stub .SIS files.....41
  - 6.3 Shared DLLs and ECom plugins .....41
  - 6.4 Embedded .SIS files.....41
- Disclaimer .....42

# 1 Introduction

This document defines the test cases that an application must comply with to be

- *Express Signed* or
- *Certified Signed*.

This document should be used as best practice guidance for *all* developers and can be used as the final check list for tests to be passed for developers seeking to have their applications signed. It is recommended these test cases be included as part of any testing and/or quality assurance you do on an application irrespective of the signing path prior to release

The tests outlined in this document have evolved since the launch of Symbian Signed in 2004, they aim to provide an agreed baseline for application stability for the following stakeholders:

- Developers** – desire a uniform and consistent approach to testing and certification, and wish to test once and deploy an application globally.
- Mobile Network Operators** – who are concerned that applications do not harm the operation of the network(s) to which they may connect
- Phone Manufacturers** – who must ensure their device continues to operate normally, specifically as a fully specification compliant phone, messaging and alerting platform; before, during and after an application is installed or removed
- End Users** – who desire a positive, trust-worthy experience, especially with regard to notification of expected usage, provision for user data privacy and protection against data loss.

The *Symbian Signed Test Criteria* is designed to be effective yet light weight; hence it is **not** intended to cover the following areas:

- Universal device coverage; i.e. applications are tested on representative devices, not all possible device choices.
- Content quality; e.g. style, taste, color or content appropriateness.
- Censorship.
- Localization.
- User interface standards conformance and style guides.
- Business and/or commercial issues.

## 2 Useful Links and Tools

### 2.1 Recommended Reading

- *Web: Symbian Signed Web Site*, <[www.symbiansigned.com](http://www.symbiansigned.com)>
- *Web: Symbian Signed Wiki*; <<http://developer.symbian.com/sswiki>>
- *Forums: Symbian Signed Support Forum*; <<http://developer.symbian.com/forum/forum.jspa?forumID=54>>
- *Free Guide: Complete Guide to Symbian Signed* <<http://developer.symbian.com/ssguide>>
- *Book: Symbian OS Platform Security: Software Development Using the Symbian OS Security Architecture*, by Craig Heath <<http://developer.symbian.com/wiki/display/academy/Symbian+OS+Platform+Security>>

### 2.2 Useful Tools

It is recommended that developers use or create the utilities listed below,

Useful Utilities	
<Disk Usage Tool>	List files, total disk usage and allows a before/after comparison (e.g. via a DIFF on file lists)
<Task Profiler Tool>	Any kind of application profiler/task list
<LowMem Tool>	LowMem Tool for memory load/stress testing Not currently available for Symbian OS v9.x. - any tests for applications on Symbian OS v9.x requiring this tool are not required
<VerifySymbianSigned Tool>	Verifies that a SIS file is signed correctly with your <i>Publisher ID</i> .

Download links for some existing tools can be found at <http://developer.symbian.com/sstools>

In addition, the following additional hardware and software is required for to carry out all of the tests effectively:

Required additional Hardware/Software
A text editor on your development machine (e.g. Desktop PC).
Secondary phone with an ability to call and SMS/MMS the phone under test
Any external media which may be required by the application
Media for mass memory devices
PC connectivity suite and appropriate connection hardware, or any other mechanism to install the application



The phones your application will be tested on is based on the following:

- Choose a single phone during the submission process to Symbian Signed. This phone will be used for all the Symbian Signed Test Criteria tests.
- The UIDs specified in the .pkg file used to build the SIS file are used as a basis for test UNI-10 (Scalable UI). This means that, for example, if the S60 3<sup>rd</sup> edition FP1 Platform UID is specified (0x102032BE) the Scalable UI test will be done for all the lead devices of that platform. If the generic S60 3<sup>rd</sup> edition Platform UID is specified the Scalable UI test will be done for all S60 3<sup>rd</sup> edition lead devices.
- See the information here for details: <http://developer.symbian.com/ssscalableui>

### 3.1 Applications for pre Symbian OS v9.x (S60 2nd Edition, UIQ 2.x)

There is no technical requirement to sign applications for releases prior to Symbian OS v9.x (i.e. Symbian OS v6.x, v7.x, v8.x). However, signing via *Express Signed* or *Certified Signed* will remove the installer warning screen that is presented to the end user at installation.

### 3.2 Applications for Symbian OS v9.x and later (S60 3rd Edition, UIQ 3.x)

From Symbian OS v9.x onwards a Platform Security model was introduced to the operating system. Some OS functionality is protected by *Capabilities*.

- Successful signing of applications via *Express Signed* will grant the *Capabilities* as requested by the application other than *DiskAdmin*, *CommDD*, *MultimediaDD*, *NetworkControl*, *AllFiles*, *DRM*, *TCB*.
- Successful signing of applications via *Certified Signed* will grant the *Capabilities* as requested by the application other than *AllFiles*, *DRM*, *TCB*.
- Signing of applications via *Express Signed* and *Certified Signed* will remove the installer warning screen at install time.

### 3.3 Further Testing requirements for System Capabilities; AllFiles, DRM, TCB

If your application uses the most sensitive *Capabilities*; *AllFiles*, *DRM*, *TCB* you will be required to successfully comply with *Symbian Signed Test Criteria*.

Additional testing and authorization for these *Capabilities* is defined by each device manufacturer and is outside the scope of the *Symbian Signed Test Criteria*. For more information

- Nokia: <http://www.forum.nokia.com/testing>
- Sony Ericsson: <http://developer.sonyericsson.com/symbiansigned>
- Samsung: <http://innovator.samsungmobile.com/bbs/signing/view.do>
- LG: Contact directly.
- Motorola: Contact directly.

If an application uses any of the Phone Manufacturer Capabilities (*DRM*, *AllFiles* or *TCB*) the package file must contain a conditional block that allows this SIS file to install only to the phones of one manufacturer.

This is done using the following code in the .pkg file:

```
IF manufacturer = 2 ; (2 is Nokia)
```

```
;This part will then contain the installation information about the files of the application
```

```
ELSE
```

```
"badmanufacturer.txt"-", FILETEXT, TEXTABORT
```

ENDIF

See <http://www3.symbian.com/faq.nsf/0/CC206C924BE43771802570AF001D6AD4?OpenDocument> for a complete list of manufacturer UIDs.

The Test House will verify that a file using Phone Manufacturer Capabilities can only be installed on phones by a single manufacturer. This involves attempting to install on other manufacturer devices. The text in "badmanufacturer.txt" should be displayed in this case.

### 3.4 Exceptions

Some types of application may require functionality that falls outside the scope of the tests. In these circumstances, valid exceptions are listed at the end of each test case.

If submitting using *Express Signed* it is essential that you declare any Exceptions using the form during submission.

If submitting using *Certified Signed* you can include details of any Exceptions in the readme.txt Release Notes file. This will ensure that the Test House is aware of any Exceptions in advance of testing.

### 3.5 Other Non-Standard Submissions

There are a small number of situations which can not be treated in the same way as other submissions:

- Waivers
- Stub .SIS files
- Shared DLLs
- Embedded .SIS files

These are documented in Appendix 1 of this document.

### 3.6 Eligibility for *Express Signed*

If a submission is made to *Express Signed* the submission file must be testable by a Test House if chosen for an audit. This means that if your application requires, for example, a specific SIM card or specific phone network it must be possible for you to facilitate the testing of your application when requested by the Test House. If it is not possible for a Test House to test a submission that is chosen for audit then it will be failed. If you are unable to facilitate testing by a Test House at the Test House location you should use *Certified Signed* and make alternative arrangements for testing.

### 3.7 Declarative Statements

Declarative statements are part of the guarantee Symbian Signed gives to Network Operators and Phone Manufacturers.

Declarative statements are intended to:

- Provide traceability of your application's capabilities and why you need to use the capabilities
- Allow Symbian to review the statements so that API and capability mappings can be reviewed

Here are some examples showing the information that should be submitted to Symbian Signed for the Declarative Statements:

**Does your application use the following capability: ProtSrv**

The application requires ProtSrv because it uses a recognizer to enable the application to load .mov file types.

**If yes, please outline the APIs used in this capability**

The CApaDataRecognizerType class is used to implement the recognizer.

**Does your application use the following capability: Location**

The application requires Location in order to get the current location of the phone through the GPS functionality

**If yes, please outline the APIs used in this capability**

The Location Acquisition API

RPositionServer::Connect(), RPositioner::NotifyPositionUpdate()

## 4 Submission Criteria

The following self checklist is provided for developers and should be carried out prior to submission. Please ensure that your application meets the Submission Criteria. Non compliance will be treated as follows:-

- **WARNING:** Signing of the application is permitted, assuming all relevant test cases pass, although the developer is warned that this may cause further issues.
- **FAIL:** The application will either fail during submission to [www.symbiansigned.com](http://www.symbiansigned.com) or in some circumstances if a signed application is found to not comply at any stage after signing, the signature will be revoked.

Checklist		Requirement/Rationale	Impact of non-compliance
<input type="checkbox"/> CHECK-01	User Guide or user instructions, Marketing Material, Functional description	<ul style="list-style-type: none"> <li>▪ This information assists in defining “end user expectations” for your application; i.e. describing your application’s functionality and/or documentation which takes reasonable steps to inform the end user of the applications intended behavior.</li> <li>▪ This information will also be used to aid correct testing. The functionality tested in your application must generally correspond to the functionality described in your user guides, marketing material and declarative statements.</li> </ul>	FAIL
<input type="checkbox"/> CHECK-02	Embedded .SIS files are already signed	<ul style="list-style-type: none"> <li>▪ Embedded SIS files need to be signed BEFORE being contained</li> </ul>	FAIL

Checklist		Requirement/Rationale	Impact of non-compliance
		<p>in the outer SIS file.</p> <ul style="list-style-type: none"> <li>▪ This may cause installation warnings or failed installations.</li> </ul>	
<input type="checkbox"/> CHECK-03	<p>Can the application be tested via the user interface?</p>	<ul style="list-style-type: none"> <li>▪ If your submission does not have a user interface, ensure that there is a suitable test harness application that can also be submitted and used to carry out testing.</li> <li>▪ The test harness must be capable of stimulating the functionality of the application or DLL.</li> </ul>	<p>FAIL</p>
<input type="checkbox"/> CHECK-04	<p>The .SIS file being submitted must be correctly signed with a valid <i>Publisher ID</i></p>	<ul style="list-style-type: none"> <li>▪ The <i>Publisher ID</i> you used to sign the SIS file must match the <i>Publisher ID</i> in your Symbian Signed account.</li> </ul>	<p>FAIL</p>
<input type="checkbox"/> CHECK-05	<p>The .SIS file includes correct version information which corresponds to the application 'About' box and/or supporting documentation.</p>	<ul style="list-style-type: none"> <li>▪ If you do not adhere to a consistent naming convention you may have difficulties in delivering future upgrades and patches.</li> <li>▪ As a suggested guideline, always specify two digits for your minor version to avoid confusion. For example:             1,10,7= v1.10 (build/revision number 7)            1,1,7 = v1.01 (build/revision number 7) – i.e. this is not equivalent to the version above            1,01,7= v1.01 (build/revision number 7) – i.e. equivalent to the one above</li> </ul>	<p>WARNING</p>
<input type="checkbox"/> CHECK-06	<p>Declarative Statements for submission.</p>	<ul style="list-style-type: none"> <li>▪ If you use ANY <i>Capabilities</i> you will be required to provide an explanation of your API usage at time of submission the developer must declare their <i>Capability</i> usage with satisfactory rationale.</li> <li>▪ The <i>Capability</i> usage must disclose the API(s) being used for each of the declared <i>Capabilities</i>.</li> <li>▪ These statements shall be retained on file and must accurately describe the functionality of the application in accordance with user guides and</li> </ul>	<p>FAIL</p>

Checklist		Requirement/Rationale	Impact of non-compliance
		marketing information associated with the application	
<input type="checkbox"/> CHECK-07	Declare Backup or No Backup	<ul style="list-style-type: none"> <li>▪ At submission you are required to declare whether the application is intended to be backed up or not. You should declare this as in the “UNI-07” test result or the “readme.txt” file provided with your submission.</li> <li>▪ If intended to be backed up, the application should backup and restore successfully along with other applications on the device.</li> <li>▪ If the application is not intended to be backed up it must not interfere with backup and restore of other applications.</li> </ul>	FAIL
<input type="checkbox"/> CHECK-08	Correct <i>Platform</i> UID	<ul style="list-style-type: none"> <li>▪ The <i>Platform</i> UID matches the range of target devices for the application.</li> <li>▪ The application must be testable on all these devices.</li> </ul>	FAIL
<input type="checkbox"/> CHECK-09	Correct <i>Package (PKG)</i> file UID	<ul style="list-style-type: none"> <li>▪ The <i>Package (PKG)</i> UID is owned by the developer</li> <li>▪ The <i>Package (.PKG)</i> UID is NOT in the following range             <ul style="list-style-type: none"> <li>▪ 0x01000000 to 0x0FFFFFFF (pre-Symbian OS v9.x)</li> </ul> </li> <li>▪ The <i>Package (.PKG)</i> UID is in the following range if a Symbian OS v9.x application             <ul style="list-style-type: none"> <li>▪ 0x20000000 to 0x2FFFFFFF</li> </ul> <p style="text-align: center;">OR</p> <li>▪ The <i>Package (.PKG)</i> UID is 0x10202BE9 or 0x101F7989</li> </li></ul>	FAIL
<input type="checkbox"/> CHECK-10	<p>If the <i>Package (.PKG)</i> UID= 0x10202BE9, i.e. an update to the Central Repository.</p> <p>Please note that approval is required from a Device Manufacturer and Symbian</p>	<ul style="list-style-type: none"> <li>▪ It must NOT contain binaries</li> <li>▪ The vendor string must be “Symbian Software Ltd”</li> <li>▪ The package must contain .txt AND/OR .cre file(s) which are</li> </ul>	FAIL

Checklist		Requirement/Rationale	Impact of non-compliance
	before a company can submit using this UID.	named in the format '<NNNNNNNN>.cre' and/or '<NNNNNNNN>.txt' where <NNNNNNNN> is a protected range UID and owned by the submitter.	
<input type="checkbox"/> CHECK-11	<p>If the <i>Package (.PKG)</i> UID=0x101F7989 package is an update to the C32</p> <p>Please note that approval is required from a Device Manufacturer and Symbian before a company can submit using this UID.</p>	<ul style="list-style-type: none"> <li>▪ The <i>package</i> must contain only one .ESK file.</li> <li>▪ The <i>package</i> must also contain all .PRT DLLs referenced in the .ESK file.</li> <li>▪ The name of the .ESK file for an IP hook should be "ip.&lt;prt name&gt;.esk", where &lt;prt name&gt; is the same as the name of the .PRT DLL, otherwise it should be &lt;thread name&gt;.&lt;prt name&gt;.esk. The package may contain multiple .PRT DLLs, in which case the naming of the .ESK file may vary. If a .PRT DLL is uninstalled, the corresponding ESK file must also be removed.</li> </ul>	FAIL
<input type="checkbox"/> CHECK-12	<p>Correct <i>Package</i> file VID</p> <p><i>Symbian OS v9.x Only</i></p>	<ul style="list-style-type: none"> <li>▪ If a VID is specified, VID is owned by the developer.</li> <li>▪ If a VID is specified, the VID it is from the correct dedicated range;               <p style="text-align: center;">VID = 0x70000000 - 0x7FFFFFFF</p> </li> <li>▪ If a VID is NOT specified VID=0</li> </ul>	FAIL
<input type="checkbox"/> CHECK-13	<p>Correct application UID</p> <p><i>Symbian OS v9.x Only</i></p>	<ul style="list-style-type: none"> <li>▪ All UIDs are owned by the developer</li> <li>▪ UIDs are from the the correct dedicated range UID = 0x20000000 - 0x2FFFFFFF</li> </ul>	FAIL
<input type="checkbox"/> CHECK-14	<p>Binary file names contain a UID that is owned by the submitter and used within the application.</p>	<ul style="list-style-type: none"> <li>▪ The Recommended Best Practice here is to simply name your binaries in the format MyBinaryName_UID.dll (or MyApplicationName_UID.exe).</li> <li>▪ For example, your application may install the following two binaries: MySpecialEngine_0x2345678.dll and</li> </ul>	WARNING

Checklist		Requirement/Rationale	Impact of non-compliance
		MySpecialApplication_0x2345678.dll	
<input type="checkbox"/> CHECK-15	<p>Illegal recognizer usage for auto-start</p> <p><i>Only applicable for:-</i></p> <ul style="list-style-type: none"> <li>▪ <i>S60 3<sup>rd</sup> Edition applications</i></li> <li>▪ <i>Where auto-start functionality is present and the Prot.Serv Capability is required</i></li> </ul>	<ul style="list-style-type: none"> <li>▪ Check the <i>Package (PKG)</i> file indicates that there is a RSC file installed to the c:\private\101f875a\import-directory and is named &lt;package UID&gt;.RSC             <ul style="list-style-type: none"> <li>▪ If the file is present the developer must also declared upon submission that they do NOT use recognizers to achieve auto-start functionality, but rather that recognizers are included for other (legitimate) purposes.</li> <li>▪ If the file is not in place and the application starts at device boot then it is safe to assume that the application does not use a supported method to start at the device boot and this case must be failed.</li> </ul> </li> </ul>	FAIL
<input type="checkbox"/> CHECK-16	<p>Operator approval required for use of NetworkControl Capability if directly accessing the SIM card.</p>	<ul style="list-style-type: none"> <li>▪ An application must not access the SIM card for network access purposes, without explicit permission from the operator who provides the SIM cards.</li> <li>▪ The installation of the application must be limited to the operator via the IMSI.</li> <li>▪ Within the “Declarative Statements” the NetworkControl Capability must be declared as:             <ul style="list-style-type: none"> <li>▪ The application does not use this capability to directly access the SIM card for network purposes</li> </ul> <p>OR</p> <ul style="list-style-type: none"> <li>▪ Details of when/how approval was obtained from Network operator.</li> <li>▪ Email and phone contact</li> </ul> </li> </ul>	FAIL

Checklist		Requirement/Rationale	Impact of non-compliance
		<p>details of approver within the operator who will be contacted for verification of the above.</p>	
<input type="checkbox"/> CHECK-17	<p>Application using any of the capabilities AllFiles, DRM or TCB must only allow installation to approved manufacturer devices</p>	<p>If an application uses any of the Phone Manufacturer Capabilities (DRM, AllFiles or TCB) the package file must contain a conditional block that allows this SIS file to install only to the phones of one manufacturer.</p> <p>This is done using the following code in the .pkg file:</p> <pre> IF manufacturer = 2 ; (2 is Nokia, see <a href="http://www3.symbian.com/faq.nsf/0/CC206C924BE43771802570AF001D6AD4?OpenDocument">http://www3.symbian.com/faq.nsf/0/CC206C924BE43771802570AF001D6AD4?OpenDocument</a> for a complete list of manufacturer UIDs)  ;This part will then contain the installation information about the files of the application  ELSE  "badmanufacturer.txt"-"" , FILETEXT, TEXTABORT  ENDIF                     </pre>	<p>FAIL</p>

## 5 Symbian Signed Test Cases

### 5.1 Universal Tests

Test ID	Test Title	Estimated Test Time (minutes)
<b>UNI-01</b>	<b>Installation, Normal and Stressed Usage</b>	<b>60-120</b>
<p><i>Test Description</i></p> <p><b>The application .SIS file installs and successfully functions after start up.</b></p> <p><b>The application does not affect the use of the core system features or other applications.</b></p> <p><b>During extended usage, the application can handle exceptional, illegal, or erroneous user actions.</b></p> <p><b>At no time should the application cause the phone to crash or freeze, and it should exit gracefully from any application-specific exceptions.</b></p> <p><i>Not Required for:</i></p> <ul style="list-style-type: none"> <li><input type="radio"/> <b>Passive content</b></li> </ul> <p><i>Required for:</i></p>		

Test ID <b>UNI-01</b>	Test Title <b>Installation, Normal and Stressed Usage</b>	Estimated Test Time (minutes) <b>60-120</b>
<ul style="list-style-type: none"> <li>⊙ <b>Pre Symbian OS v9.x application</b></li> <li>⊙ <b>No Capabilities - Symbian OS v9.x</b></li> <li>⊙ <b>ALL Symbian OS v9.x Capabilities</b></li> </ul>		
<p><i>Testing Note</i></p> <p>Note1: Do not carry out emergency call test on the public telephone network as it will cause calls to actual emergency services.</p> <p>Note2: The practical purpose of this test is as a regression test prior to release. For wide scale deployment, it is expected that the application has been extensively and thoroughly system tested in many different scenarios prior to submission, which is anticipated to be essential to successfully pass this test case.</p> <p><i>Testing Steps</i></p> <p>STEP1: Install the .SIS file. <i>RESULT: The .SIS file installs successfully.</i></p> <p>STEP2: Start the application. <i>RESULT: The application starts successfully within a reasonable time period (5 seconds) or provides appropriate progress indication.</i></p> <p>STEP3: Exit the application and re-start the application (if possible). <i>RESULT: The application exits and restarts successfully.</i></p> <p>STEP4: Use the application as per normal operation as specified in the user guide and in accordance with Declarative Statements received via submission. <i>RESULT: The application works in accordance with the end user expectations for the applications.</i></p> <p>STEP5: Whilst using the application, switch away and use the main system applications and features (e.g. <b>Phone, Calendar/Agenda, Clock, Contacts</b>). <i>RESULT: The application does not inappropriately control thread priority or switch/steal focus from other applications, and it releases resources when not in use.</i> <i>RESULT: The application allows other applications access to sufficient OS resources to run, i.e. the application shuts down/releases resources if the system requests it to do so.</i></p> <p>STEP6: Whilst the application is running in the foreground proceed to activate the <b>System Lock</b>. Verify it is still possible to unlock the phone and return to the application. <i>RESULT: The application must not inadvertently override the system lock; i.e. unless it is a feature or characteristic of the application in line with the user's expectation.</i></p> <p>STEP7: Apply stressful scenarios to the device whilst using application as per its normal operation. For example:</p> <ul style="list-style-type: none"> <li>• Switching rapidly between applications.</li> <li>• Opening and using many other applications simultaneously.</li> <li>• Launch the camera application and take several pictures whilst the application is running.</li> <li>• Pressing keys rapidly and/or tapping the screen rapidly (if touch-screen driven) in the application to send it many events at once..</li> <li>• Enter inappropriate data into the application.</li> </ul>		

Test ID <b>UNI-01</b>	Test Title <b>Installation, Normal and Stressed Usage</b>	Estimated Test Time (minutes) <b>60-120</b>
<ul style="list-style-type: none"> <li>○ Special characters.</li> <li>○ Leaving input fields blank.</li> <li>○ Entering strings of maximum length (where possible)..</li> <li>● Removing memory cards whilst the application is under test.</li> <li>● If the application creates connections start a connection and close it from the connection manager before the connection is fully established..</li> <li>● If the application uses connections. Use the connection manager to close the connection suddenly.</li> </ul> <p><i>RESULT: The application must not invalidate basic Type Approval of the phone, i.e. it must always be able to switch away from the application and to make an Emergency Call.</i></p> <p><i>RESULT: The application must be able to handle exceptional, illegal, or erroneous actions. It will not cause the phone to crash, freeze or the phone to become unusable.</i></p> <p><i>RESULT: The application exits gracefully (i.e. with appropriate error message not system panics) from unrecoverable exceptions and can be restarted successfully following any exception.</i></p>		
<p><i>For Test Houses/Test Runs – Result of Test</i></p> <p><input type="checkbox"/> <b>PASS</b>                      <input type="checkbox"/> <b>FAIL</b></p> <p><b>EXCEPTION(S)</b></p> <p><input type="checkbox"/> <b>UNI-01.EX1: Infrequent or Non Repeatable Crash.</b></p> <p>If an application fails this test as a result of crashing, it may receive an exception if the crash is infrequent, non repeatable and the application, other applications and mobile phone remains usable within user expectations.</p> <p><input type="checkbox"/> <b>UNI-01.EX2: No User Interface</b></p> <p>If the SIS file is supplying only shared libraries (DLLs) and/or system/middleware components with no user interaction (i.e. no user interface) you must provide a “Test Harness” for the application. This should include documentation for setup and execution of the “Test Harness” and should also provide a test report for the “Test Harness” and expected results. The “Test Harness” should also pass all Symbian Signed Test Criteria.</p>		

Test ID <b>UNI-02</b>	Test Title <b>Service Interruption</b>	Estimated Test Time (minutes)
<p><i>Test Description</i></p> <p><b>The application should handle interruptions appropriately for the type of application and the type of interruption.</b></p>		
<p><i>Not Required for:</i></p> <p><input type="radio"/> <b>Passive content</b></p> <p><i>Required for:</i></p> <p><input checked="" type="radio"/> <b>Pre Symbian OS v9.x application</b></p> <p><input checked="" type="radio"/> <b>No Capabilities - Symbian OS v9.x</b></p> <p><input checked="" type="radio"/> <b>ALL Symbian OS v9.x Capabilities</b></p>		
<p><i>Testing Steps</i></p> <p>STEP1: Start and/or use the application as per normal use conditions.</p> <p>STEP2: Create and test the following scenarios.</p> <ul style="list-style-type: none"> <li>• Receiving an incoming voice call.</li> <li>• Receiving an incoming SMS.</li> <li>• Receiving an incoming MMS.</li> <li>• An alarm notification event.</li> <li>• Connecting external power charger.</li> <li>• Switching of UI Modes (e.g. flip, horizontal/vertical).</li> </ul> <p><i>RESULT: All interruptions are handled as a user would expect and the application continues to operate normally after the interruption. For example: a game should store its state and pause, whereas a timer application should continue even with the incoming interruption.</i></p> <p><i>RESULT: The interruptions are given user interface focus, e.g. audio, visual and user interface notifications occur as per normal phone operation.</i></p>		
<p><i>For Test Houses/Test Runs – Result of Test</i></p> <p><input type="checkbox"/> <b>PASS</b>      <input type="checkbox"/> <b>FAIL</b></p> <p><b>EXCEPTION(S)</b></p> <p><b>No exceptions permitted.</b></p>		

Test ID <b>UNI-03</b>	Test Title <b>Low Memory Startup</b>	Estimated Test Time (minutes)
<p><i>Test Description</i></p> <p><b>The application should gracefully handle low memory situations during startup as per the LowMem Tool User Guide. When exiting due to low memory, the application displays appropriate error messages.</b></p>		
<p><i>Not Required for:</i></p> <ul style="list-style-type: none"> <li><input type="radio"/> <b>Passive content</b></li> </ul> <p><i>Required for:</i></p> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> <b>Pre Symbian OS v9.x application</b></li> <li><input checked="" type="radio"/> <b>No Capabilities - Symbian OS v9.x</b></li> <li><input checked="" type="radio"/> <b>ALL Symbian OS v9.x Capabilities</b></li> </ul>		
<p><i>Testing Steps</i></p> <p>STEP1: Run &lt;LowMem Tool&gt; to simulate a low memory situation.</p> <p>STEP2: Start the application.</p> <p style="padding-left: 40px;"><i>RESULT: The application displays appropriate warning messages if it is unable to start during low memory conditions.</i></p> <p>STEP3: Exit &lt;LowMem Tool&gt;.</p> <p style="padding-left: 40px;"><i>RESULT: The &lt;LowMem Tool&gt; application reports a failure rate of no more than 10% errors..</i></p>		
<p><i>For Test Houses/Test Runs – Result of Test</i></p> <p><input type="checkbox"/> <b>PASS</b>                      <input type="checkbox"/> <b>FAIL</b></p> <p><b>EXCEPTION(S)</b></p> <p><input type="checkbox"/> <b>UNI-03.EX1: Non native application/s</b></p> <p style="padding-left: 40px;">If the application under test is running inside a Virtual Machine (e.g. Python, Flash Lite) then the application should not be responsible for memory failures caused by the Virtual Machine itself. As such, these applications are exempted from low memory testing.</p> <p><input type="checkbox"/> <b>UNI-03.EX2: &lt;LowMem Tool&gt; Not Available For Symbian OS v9.x</b></p> <p style="padding-left: 40px;">Until LowMem Tool is generally available, this test can be skipped for applications on Symbian OS v9.x (i.e. S60 3<sup>rd</sup> Edition and UIQ3.x).</p>		

Test ID <b>UNI-04</b>	Test Title <b>Low Storage Memory During Startup &amp; Execution</b>	Estimated Test Time (minutes)
<p><i>Test Description</i>  <b>The application should not fill all the available storage space of the device.                      If there is not enough storage space to run, the application indicates this to the user and exits gracefully.</b></p>		
<p><i>Not Required for:</i></p> <ul style="list-style-type: none"> <li><input type="radio"/> <b>Passive content</b></li> </ul> <p><i>Required for:</i></p> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> <b>Pre Symbian OS v9.x application</b></li> <li><input checked="" type="radio"/> <b>No Capabilities - Symbian OS v9.x</b></li> <li><input checked="" type="radio"/> <b>ALL Symbian OS v9.x Capabilities</b></li> </ul>		
<p><i>Testing Steps</i></p> <p>STEP1: Start application.</p> <p>STEP2: Simulate a low storage memory situation by filling the space with a number of large files (e.g. miscellaneous JPGs, MP4, MP3).</p> <p>STEP3: Use application as per normal expected use.</p> <p><i>RESULT: The application either runs successfully or provides appropriate warning messages to the user to explain why it cannot run, exiting gracefully.</i></p> <p>STEP4: Close the application (if not already closed).</p> <p>STEP5: Start and use the application as per normal expected use.</p> <p><i>RESULT: The application either runs successfully or provides appropriate warning messages to the user to explain why it cannot run, exiting gracefully.</i></p> <p>STEP6: If desired, delete files added in STEP2.</p>		
<p><i>For Test Houses/Test Runs – Result of Test</i></p> <p><input type="checkbox"/> <b>PASS</b>      <input type="checkbox"/> <b>FAIL</b></p> <p><b>EXCEPTION(S)</b></p> <p><input type="checkbox"/> <b>UNI-04.EX1: Executing within a Run Time Environment or Virtual Machines</b></p> <p>If the application under test is running inside a Virtual Machine (including, but not necessarily limited to, OPL, Python, Flash Lite) then the application should not be responsible for memory failures caused by the Virtual Machine itself. As such, these applications are exempted from low memory testing and Symbian will work with the Virtual Machine vendors to ensure these environments behave appropriately.</p>		

<i>Test ID</i> <b>UNI-04</b>	<i>Test Title</i> <b>Low Storage Memory During Startup &amp; Execution</b>	<i>Estimated Test Time (minutes)</i>
<p><input type="checkbox"/> <b>UNI-04.EX2: Warning the End User Degrades the User Experience</b></p> <p>For applications that do not have any user interaction, for example system and middleware services without a user interface, it is permissible to not provide a warning. However, the application should still handle low storage memory situations gracefully (i.e. the phone should remain usable and the application should be usable once sufficient memory is available).</p> <p>Acceptable application types for this exception are</p> <ul style="list-style-type: none"> <li>▪ Anti-virus applications,</li> <li>▪ Firewall and VPN clients,</li> <li>▪ Device Management software,</li> <li>▪ Front End Processor (FEP) plug-ins,</li> <li>▪ System and middleware components (without any user interaction)</li> <li>▪ Sis files that only contain DLLs</li> </ul>		

Test ID <b>UNI-05</b>	Test Title <b>System Events and Task List Compliance</b>	Estimated Test Time (minutes)
<p><i>Test Description</i> <b>The application can be closed by the Task List or Task Manager.</b></p>		
<p><i>Not Required for:</i></p> <ul style="list-style-type: none"> <li><input type="radio"/> <b>Passive content</b></li> </ul> <p><i>Required for:</i></p> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> <b>Pre Symbian OS v9.x application</b></li> <li><input checked="" type="radio"/> <b>No Capabilities - Symbian OS v9.x</b></li> <li><input checked="" type="radio"/> <b>ALL Symbian OS v9.x Capabilities</b></li> </ul>		
<p><i>Testing Steps</i></p> <p>STEP1: Ensure the application is running.</p> <p>STEP2: On S60 non touch screen device:</p> <p>Switch focus to <b>Task List</b> (by holding down the S60 Applications key)</p> <ol style="list-style-type: none"> <li>1. Close application from <b>Task List</b> using the 'c' or back key.</li> </ol> <p>On Series 60 Touch Screen device:</p> <ol style="list-style-type: none"> <li>1. Perform a long press on the application key</li> <li>2. To close an application through the task list, perform a long touch on the application icon in the task list</li> <li>3. The menu is opened using the "Open" option, select "Exit" to close the application.</li> </ol> <p>On Series 80 device:</p> <ol style="list-style-type: none"> <li>1. Press <b>Menu</b>.</li> <li>2. Select the <b>Task List</b> menu item.</li> <li>3. Close the application from the <b>Task List</b> by selecting <b>End Task</b>.</li> </ol> <p>On UIQ device (<b>Task Manager</b> present):</p> <ol style="list-style-type: none"> <li>1. Run the system <b>Task Manager</b>.</li> <li>2. Select the menu option to close the application from the <b>Task List</b>.</li> </ol> <p>On UIQ device (<b>Task Manager</b> not present):</p> <ol style="list-style-type: none"> <li>1. Run &lt;File Storage Tool&gt; to force a low memory message to the application.</li> <li>2. Use &lt;Task Profiler Tool&gt; to list active tasks and processes, or the built in task manager if applicable.</li> </ol> <p><i>RESULT: The application is visible in the task list and can be closed such that it is no longer running in memory.</i></p>		
<p><i>For Test Houses/Test Runs – Result of Test</i></p> <p><input type="checkbox"/> <b>PASS</b>      <input type="checkbox"/> <b>FAIL</b></p> <p><b>EXCEPTION(S)</b></p>		

Test ID <b>UNI-05</b>	Test Title <b>System Events and Task List Compliance</b>	Estimated Test Time (minutes)
<p><input type="checkbox"/> <b>UNI-05.EX1: Application Already Closed</b></p> <p>If the application closes down on change of focus in accordance with end user expectations, it is not required to be closable via the <b>Task List</b> as this can never occur. An example of this type of application may be a business card scanner that uses camera to capture input, upon loss of focus it may close as to release the camera for general use.</p> <p><input type="checkbox"/> <b>UNI-05.EX2: Application Should Not Close or Does Not Run</b></p> <p>If the SIS file contains only shared DLLs, it cannot be 'run' and therefore closed in this manner.</p> <p><input type="checkbox"/> <b>UNI-05.EX3: Application Meets End User Expectations</b></p> <p>With certain types of applications it is inherent in their functionality that they are not intended to close. Providing the application clearly functions within the end-user expectations of the application it can receive an exception from being closed via the task list. However in such case the user must be able to close the application from an Exit or Close option in the application menu.</p> <p>Acceptable application types for this exception are</p> <ul style="list-style-type: none"> <li>▪ Anti-virus applications,</li> <li>▪ Firewall and VPN clients,</li> <li>▪ Device Management software,</li> <li>▪ Front End Processor (FEP) plug-ins,</li> <li>▪ System and middleware components (without any user interaction)</li> <li>▪ Sis files that only contain DLLs</li> </ul>		

Test ID <b>UNI-06</b>	Test Title <b>Application Functionality In Between Device Reboots</b>	Estimated Test Time (minutes)
<p><i>Test Description</i></p> <p><b>The application and device should both run normally after a sudden loss of power (whilst the application was running) and subsequent reboot of the device.</b></p> <p><b>The application should not crash, panic or freeze the device at any time.</b></p>		
<p><i>Not Required for:</i></p> <ul style="list-style-type: none"> <li><input type="radio"/> <b>Passive content</b></li> </ul> <p><i>Required for:</i></p> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> <b>Pre Symbian OS v9.x application</b></li> <li><input checked="" type="radio"/> <b>No Capabilities – Symbian OS v9.x</b></li> <li><input checked="" type="radio"/> <b>ALL Symbian OS v9.x Capabilities</b></li> </ul>		
<p><i>Testing Steps</i></p> <p>STEP1: Ensure the application is running.</p> <p>STEP2: Remove the battery to force the phone to suddenly power down.</p> <p>STEP3: Re-insert battery, reboot the phone and restart the application.</p> <p>STEP4: Verify the application is running correctly (as before).</p> <p><i>RESULT: The application and device operate normally. The device reboots successfully.</i></p>		
<p><i>For Test Houses/Test Runs – Result of Test</i></p> <p><input type="checkbox"/> <b>PASS</b>      <input type="checkbox"/> <b>FAIL</b></p> <p><b>EXCEPTION(S)</b></p> <p><b>No exceptions permitted.</b></p>		

Test ID <b>UNI-07</b>	Test Title <b>Backup and Restore Compliance</b>	Estimated Test Time (minutes)
<p><i>Test Description</i></p> <p><b>The application should not interfere with a system <i>backup</i> and <i>restore</i>.</b>  <b>If the application is intended to be <i>backed-up</i>, i.e. it is registered for back-up with the system, it should also continue to operate successfully upon <i>restore</i>.</b></p>		
<p><i>Not Required for:</i></p> <ul style="list-style-type: none"> <li><input type="radio"/> <b>Passive content</b></li> </ul> <p><i>Required for:</i></p> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> <b>Pre Symbian OS v9.x application</b></li> <li><input checked="" type="radio"/> <b>No Capabilities - Symbian OS v9.x</b></li> <li><input checked="" type="radio"/> <b>ALL Symbian OS v9.x Capabilities</b></li> </ul>		
<p><i>Testing Notes</i></p> <p>Note1: Applications and data are <b>not</b> automatically backed up on Symbian OS v9.x phones. You must explicitly register with the backup server to do this.</p> <p>Note2: Backup and restore may not work during testing and/or when the application is signed via <i>Open Signed</i> in early some earlier versions of S60 3<sup>rd</sup> Edition devices. To complete tests in such devices the application will need to be <i>Express Signed</i> or <i>Certified Signed</i>. This is issue is fixed in S60 3<sup>rd</sup> Edition Feature Pack 1.</p> <p><i>Testing Steps</i></p> <p>STEP1: Start the application.</p> <p>STEP2: Switch away from the application and add two new contacts and two new items in the default calendar</p> <p>STEP3: Connect the phone to the PC Suite supplied by the manufacturer and commence a backup. If there is no appropriate PC Suite version or software for the phone (e.g. a prototype), attempt to commence a Backup to Memory Stick/Card from the appropriate utility on the phone itself.</p> <p><i>RESULT: The backup process completes with no errors – the application does not lock and fail to release any files, for example.</i></p> <p>STEP4: Once backup has completed, uninstall the application under test.</p> <p>STEP5: Commence a full restore.</p> <p>STEP6: Run the application once the restore process has completed.</p> <p><i>RESULT: The restore process completes with no errors. Once restore has completed, the application can be run and functions correctly.</i></p> <p>STEP7: Exit the application under test.</p> <p>STEP8: Run the main system applications and features (e.g. <b>Phone, Calendar/Agenda, Clock, Contacts</b>) and verify they still work. Additionally verify that the added contacts and calendar items added in step 2 are present.</p> <p><i>RESULT: The device continues to operate normally after the restore.</i></p> <p>STEP9: Take another compliant clean phone or format the phone you have, after which repeat STEP4 to STEP7 again with the new/formatted phone.</p>		

Test ID <b>UNI-07</b>	Test Title <b>Backup and Restore Compliance</b>	Estimated Test Time (minutes)
<p><i>RESULT: The restore process completes with no errors. Once restore has completed, the application can be run and functions correctly.</i></p> <p><i>RESULT: The device continues to operate normally after the restore.</i></p>		
<p><i>For Test Houses/Test Runs – Result of Test</i></p> <p><input type="checkbox"/> <b>PASS</b>      <input type="checkbox"/> <b>FAIL</b></p> <p><b>EXCEPTION(S)</b></p> <p><input type="checkbox"/> <b>UNI-07.EX1: Backup Not Possible</b></p> <p>An appropriate PC Suite or supplied a 'to memory card' backup option which functions is not available (e.g. prototype devices).</p> <p><input type="checkbox"/> <b>UNI-07.EX2: Testing Not Possible (S60 3<sup>rd</sup> Edition on Symbian OS v9.1, i.e. prior to FP1)</b></p> <p>This functionality is not testable on S60 3<sup>rd</sup> Edition on Symbian OS v9.1 devices hence this test case is not required for applications if the device ID (in the PKG file) constrains the application to only these device types.</p>		

Test ID <b>UNI-08</b>	Test Title <b>Uninstall</b>	Estimated Test Time (minutes)
<p><i>Test Description</i></p> <p><b>The application should uninstall itself cleanly.</b></p>		
<p><i>Required for:</i></p> <ul style="list-style-type: none"> <li>⊙ <b>Passive content</b></li> <li>⊙ <b>Pre Symbian OS v9.x application</b></li> <li>⊙ <b>No Capabilities - Symbian OS v9.x</b></li> <li>⊙ <b>ALL Symbian OS v9.x Capabilities</b></li> </ul>		
<p><i>Testing Notes</i></p> <p><b>Note 1. Pre-Symbian OS v9.x (i.e. Symbian OS v6.x, v7.x, v8.x)</b> Do not forget to remove .INI files which are automatically created in !:\System\Apps\AppName\ by adding the following to your PKG file:</p> <pre style="margin-left: 40px;">; Files to remove on uninstallation "- " !:\System\Apps\AppName\AppDataName.ini" , FN</pre> <p><b>Note 2. Symbian OS v9.x onwards</b> In Symbian OS v9.x, all data in your private folder will be automatically removed. If you create data in public areas you should follow the advice in Note 1, to ensure that your data in these areas is also removed.</p> <p><b>Note 3. In use Themes cannot be uninstalled</b> A Theme which is set as the current theme cannot be uninstalled. Change to a different theme before testing uninstall.</p> <p><i>Testing Steps</i></p> <p><b>STEP1:</b> Ensure that the application is already installed and then uninstall the application <i>RESULT: Any icon which was present in the system screen(s) disappears.</i></p> <p><b>STEP2:</b> Use &lt;Disk Usage Tool&gt; to check that all files are removed <i>RESULT: Application uninstalls removing all application files with the exception of only leaving small configuration files to save user preferences.</i></p> <p><b>STEP3:</b> Excluding user generated files (e.g. as a result of creating documents, bookmarks MP3s etc) use Use &lt;Disk Usage Tool&gt; to check the amount of free storage memory is the same as when testing started. <i>RESULT: Amount of free storage memory, before installation vs after uninstall, is within a 100Kb tolerance.</i></p>		
<p><i>For Test Houses/Test Runs – Result of Test</i></p> <p><input type="checkbox"/> <b>PASS</b>                      <input type="checkbox"/> <b>FAIL</b></p>		

Test ID <b>UNI-08</b>	Test Title <b>Uninstall</b>	Estimated Test Time (minutes)
<p><b>EXCEPTION(S)</b></p> <p><input type="checkbox"/> <b>UNI-08.EX1: Font File Installation on Symbian OS v6.1 and v7.0</b></p> <p>If your application installs additional fonts there is a known defect on earlier versions of Symbian OS (v6.1, v7.0) causing issues with font file removal. Hence, if your application would otherwise pass this test case, you are granted an exception.</p> <p>See FAQ-0860 in the Symbian OS FAQ database at <a href="http://developer.symbian.com">developer.symbian.com</a> for more details.</p> <p><input type="checkbox"/> <b>UNI-08.EX2: Known Defects</b></p> <p>Where a known defect in the installer prevents compliance with a test case, applications are considered exempt until such time as approved workarounds are published</p> <p>Developers should clearly state in their Release Notes when a known defect is preventing them from passing this test case. The defect must be recognized by the licensee on their web site such as <a href="http://wiki.forum.nokia.com/index.php/Category:Known_Issue">http://wiki.forum.nokia.com/index.php/Category:Known_Issue</a></p> <p><input type="checkbox"/> <b>UNI-08.EX3: Intended Behavior To Not Allow Uninstall</b></p> <p>In certain circumstances, if it is intended to prevent uninstall, as per the following scenarios.</p> <ul style="list-style-type: none"><li>▪ upgrading a ROM application on Symbian OS v9.x with a Partial Upgrade (PU) type SIS file, the user is given no option to uninstall the update/new version.</li><li>▪ The end user expectation of the application is such that it should not be installed and it is clearly documented in user guides.</li></ul>		

Test ID <b>UNI-09</b>	Test Title <b>Reinstallation and mass memory storage</b>	Estimated Test Time (minutes)
<p><i>Test Description</i>  <b>After uninstall, the application can be reinstalled successfully, including to other media locations (e.g. memory card).</b></p>		
<p><i>Required for:</i></p> <ul style="list-style-type: none"> <li><input type="radio"/> <b>Passive content</b></li> <li><input type="radio"/> <b>Pre Symbian OS v9.x application</b></li> <li><input type="radio"/> <b>No Capabilities - Symbian OS v9.x</b></li> <li><input type="radio"/> <b>ALL Symbian OS v9.x Capabilities</b></li> </ul>		
<p><i>Testing Steps</i></p> <p>STEP1: Having uninstalled the application, reinstall it.</p> <p>STEP2: Start the application.  <i>RESULT: Application should successfully re-install and function after initial uninstall.</i></p> <p>STEP3: Uninstall it again.</p> <p>STEP4: Reinstall it to an alternative memory location (i.e. if you installed to the Internal drive initially, try to install to the Memory card and Mass Memory if appropriate and vice versa).</p> <p>STEP5: Verify it works correctly again on the new memory location.  <i>RESULT: Application should install and run from any user-selected memory location correctly.</i></p>		
<p><i>For Test Houses/Test Runs – Result of Test</i></p> <p><input type="checkbox"/> <b>PASS</b>      <input type="checkbox"/> <b>FAIL</b></p> <p><b>EXCEPTION(S)</b></p> <p><input type="checkbox"/> <b>UNI-09.EX1: Installation To All Storage Devices Not Intended</b></p> <p>This exception can apply where it is not appropriate for an application to be installed on all mass memory devices. The application is expected to provide notification, warning or a mechanism to prevent installation on inappropriate mass storage devices; where no such mechanism is in place, it will be assumed that the application can be installed (and function correctly) on all mass memory devices.</p> <p>The application must state the memory device and/or the application is also prevented from being installed onto other memory devices (e.g. by hard-coding locations in the .PKG file to this effect).</p> <p>Valid reasons may include installer defects in complex situations, especially when attempting to upgrade ROM applications with a Partial Update (PU) type SIS file, or an unusually large application which may not be suitable for installation on the internal drive which may not be big enough on some phones.</p>		

Test ID <b>UNI-10</b>	Test Title <b>Scalable UI Compliance (S60 3<sup>rd</sup> Edition or UIQ3.x)</b>	Estimated Test Time (minutes)
<p><i>Test Description</i></p> <p><b>The application should function and render its display as specified, regardless of the device screen resolution and format.</b></p>		
<p><i>Not Required for:</i></p> <ul style="list-style-type: none"> <li><input type="radio"/> <b>Passive content</b></li> <li><input type="radio"/> <b>Pre Symbian OS v9.x application</b></li> </ul> <p><i>Required for:</i></p> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> <b>No Capabilities - Symbian OS v9.x</b></li> <li><input checked="" type="radio"/> <b>ALL Symbian OS v9.x Capabilities</b></li> </ul>		
<p><i>Testing Notes</i></p> <p>Note1: It is recommended that you read the following section on how to adhere to the Scalable UI test <a href="http://developer.symbian.com/ssscalableui">http://developer.symbian.com/ssscalableui</a></p> <p>Note2: To aid testing and access to different phones there are remote device testing solutions available for developers:-</p> <ul style="list-style-type: none"> <li>▪ Nokia Remote Device Access: <a href="https://www.forum.nokia.com/rda">https://www.forum.nokia.com/rda</a></li> <li>▪ Samsung Lab.dev <a href="http://innovator.samsungmobile.com/bbs/lab/view.do">http://innovator.samsungmobile.com/bbs/lab/view.do</a></li> </ul> <p><i>Testing Steps</i></p> <p>If one or more Product IDs or Machine UIDs are identified in the PKG file the test should be carried out on all lead device/s for these device families.</p> <p style="text-align: center;">OR</p> <p>If no Product ID is specified in the PKG file, then the application should be tested on the lead device for ALL different screen resolutions for the version of S60 or UIQ user interface.</p> <p>A list of lead devices and screen resolutions is available at <a href="http://developer.symbian.com/ssscalableui">http://developer.symbian.com/ssscalableui</a></p> <p>STEP1: Install to lead device for screen resolution type as identified in the PKG file.</p> <p>STEP2: Start the application.</p> <p>STEP3: Use the application normally through as many screens and scenarios as possible.</p> <p><i>RESULT: The application responds to orientation switch events appropriately, adjusting its display accordingly and continues to operate normally.</i></p> <ul style="list-style-type: none"> <li>○ <i>Provides its full functionality on all screen resolutions/orientations that are defined as supported within the application PKG file.</i></li> <li>○ <i>Uses the display area to its full extent.</i></li> <li>○ <i>Responds to orientation switch events appropriately, adjusting its display accordingly and continues to operate correctly, redrawing the screen properly.</i></li> </ul> <p><i>RESULT: If a flip or slider is present, upon moving the flip/slider:</i></p>		

Test ID <b>UNI-10</b>	Test Title <b>Scalable UI Compliance (S60 3<sup>rd</sup> Edition or UIQ3.x)</b>	Estimated Test Time (minutes)
<ul style="list-style-type: none"> <li>○ [S60 applications only] the application adjusts accordingly between portrait and landscape orientation.</li> <li>○ [UIQ applications only] the application adjusts its display accordingly and continues to operate normally. On closing the flip/slider, the application can return to the system default view.</li> </ul> <p>STEP4: Exit the application.</p>		
<p><i>For Test Houses/Test Runs – Result of Test</i></p> <p><input type="checkbox"/> <b>PASS</b>                      <input type="checkbox"/> <b>FAIL</b></p> <p><b>EXCEPTION(S)</b></p> <p><input type="checkbox"/> <b>UNI-10.EX1 Not Testable – No User Interface</b></p> <p>    If the SIS file is supplying only shared libraries (DLLs) and/or system/middleware components with no user interaction (i.e. no user interface) then this test case is not required.</p> <p><input type="checkbox"/> <b>UNI-10.EX2 Not a Generic Application</b></p> <p>    Application is intended for a specific device and User interface size only. Test Houses will check the devices specified through the results of the “.SIS file scan” and test based on these results.</p> <p><input type="checkbox"/> <b>UNI-10.EX3 Application Provides warning to user upon first startup</b></p> <p>    If an application fails this test, it may be granted an exception if it provides appropriate error message, e.g. “Screen Resolution Not Supported”, to the end user to indicate that the device’s screen resolution is not supported</p> <p><input type="checkbox"/> <b>UNI-10.EX4 Application does not switch screen mode but meets user expectations</b></p> <p>    If a main feature of the application should be used in a particular screen orientation, for example game play, then it is acceptable for the application not to respond to screend mode changes when using this feature.</p>		

Test ID <b>UNI-11</b>	Test Title <b>Correct Auto-start Behavior</b>	Estimated Test Time (minutes)
<p><i>Test Description</i>  <b>Application must provide end user option to enable/disable automatic application start up at the device boot.</b></p>		
<p><i>Not Required for:</i></p> <ul style="list-style-type: none"> <li><input type="radio"/> <b>Passive content</b></li> </ul> <p><i>Required for:</i></p> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> <b>Pre Symbian OS v9.x application</b></li> <li><input checked="" type="radio"/> <b>No Capabilities - Symbian OS v9.x</b></li> <li><input checked="" type="radio"/> <b>ALL Symbian OS v9.x Capabilities</b></li> </ul>		
<p><i>Testing Steps</i></p> <p>Note1: Automatic startup of the application can only be done if the user has enabled this functionality. It must be possible for the user to disable automatic startup of the application at any time during the running of the application, this option can never be hidden and may not be protected by a security code or otherwise.</p> <p>Note2: On installation the user can be asked if automatic startup can be allowed when installation is complete. The dialog must clearly state that the application is requesting automatic startup every time the phone is started and must give the user the option to reject this request. The user should be able to disable automatic startup following this at any time during the running of the application through an easily accessible option</p> <p>STEP1: Start the application  STEP2: Close the application  STEP3: Turn off the test device and then turn back on</p> <p style="padding-left: 40px;"><i>RESULT: Upon restart the application does not start automatically by default.</i></p> <p>STEP4: Start the application again  STEP5: Check the settings within the application that auto-start can be set to on/off.</p> <p style="padding-left: 40px;"><i>RESULT: There must be an ON/OFF setting available.</i></p> <p>STEP6: Set auto-start to ON  STEP7: Turn off the test device and then turn back on</p> <p style="padding-left: 40px;"><i>RESULT: The test device starts up successfully.</i></p> <p>STEP8: Start the application again  STEP9: Set auto-start to OFF</p> <p style="padding-left: 40px;"><i>RESULT: The user can disable auto-start successfully having enabled it previously.</i></p>		

Test ID <b>UNI-11</b>	Test Title <b>Correct Auto-start Behavior</b>	Estimated Test Time (minutes)
<p>STEP10: Turn off the test device and then turn back on</p> <p>STEP11: Start the application again</p> <p><i>RESULT: The test device starts up successfully.</i></p>		
<p><i>For Test Houses/Test Runs – Result of Test</i></p> <p><input type="checkbox"/> <b>PASS</b>      <input type="checkbox"/> <b>FAIL</b></p> <p><b>EXCEPTION(S)</b></p> <p><input type="checkbox"/> <b>UNI-11.EX1 Functionality Not Present</b></p> <p>An exception to this test case is granted to applications that do not provide auto-start functionality</p>		

## 5.2 Capability Related Tests

Test ID	Test Title	Estimated Test Time (minutes)
<b>CAP-01</b>	<b>Applications Do Not Interfere With Voice Calls</b>	<b>TBD</b>
<i>Test Description</i> <b>Applications do not interfere with voice calls</b>		
<i>Not Required for:</i> <ul style="list-style-type: none"> <li><input type="radio"/> <b>Passive content</b></li> <li><input type="radio"/> <b>No Capabilities - Symbian OS v9.x</b></li> </ul> <i>Required for:</i> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> <b>Pre Symbian OS v9.x application</b></li> <li><input checked="" type="radio"/> <b>Symbian OS v9.x Capabilities (as follows): MultimediaDD, NetworkControl</b></li> </ul>		
<i>Testing Notes</i>  <p>Note1: In many countries it is illegal to record and/or monitor audio streams without the consent of one or both of the calling participants and/or providing audible warning tones during the conversation. You are responsible to ensure that your application complies with all legal and privacy requirements. More information is available at <a href="http://en.wikipedia.org/wiki/Telephone_recording_laws">http://en.wikipedia.org/wiki/Telephone_recording_laws</a></p> <i>Testing Steps</i>  <p>STEP1: Start the application.</p> <p>STEP2: While using the audio resources/streaming features of the application (which require MultimediaDD and/or NetworkControl), make a phone call using another device to the device being tested.</p> <p><i>RESULT: When an incoming phone call is received, your applications audio is paused and the user is able to answer the call.</i></p> <p>STEP3: Check that the only allowed audio mixing is to the uplinked call, and that the volume of the background stream is set low by the third-party application.</p> <p><i>RESULT: Where applicable/essential to the application (e.g. adding sound effects), the only allowed mixing is to uplinked call, and the volume of the stream is set low by the third-party application.</i></p>		
<i>For Test Houses/Test Runs – Result of Test</i>  <input type="checkbox"/> <b>PASS</b> <input type="checkbox"/> <b>FAIL</b>   <b>EXCEPTION(S)</b>		

Test ID <b>CAP-01</b>	Test Title <b>Applications Do Not Interfere With Voice Calls</b>	Estimated Test Time (minutes) <b>TBD</b>
<i>Test Description</i> <b>Applications do not interfere with voice calls</b>		
<i>Not Required for:</i> <ul style="list-style-type: none"> <li><input type="radio"/> <b>Passive content</b></li> <li><input type="radio"/> <b>No Capabilities - Symbian OS v9.x</b></li> </ul> <i>Required for:</i> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> <b>Pre Symbian OS v9.x application</b></li> <li><input checked="" type="radio"/> <b>Symbian OS v9.x Capabilities (as follows): MultimediaDD, NetworkControl</b></li> </ul>		
<input type="checkbox"/> <b>CAP-01.EX1: Not Testable</b>  The application uses MultimediaDD and/or NetworkControl <i>Capability</i> but does not have this type of functionality and accurate declaration is made upon submission.		

Test ID <b>CAP-02</b>	Test Title <b>Telephony UI Application Control</b>	Estimated Test Time (minutes) <b>TBD</b>
<p><i>Test Description</i>  <b>Any application that is used to route calls must have a UI.</b></p>		
<p><i>Not Required for:</i></p> <ul style="list-style-type: none"> <li><input type="radio"/> <b>Passive content</b></li> <li><input type="radio"/> <b>No Capabilities - Symbian OS v9.x</b></li> </ul> <p><i>Required for:</i></p> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> <b>Pre Symbian OS v9.x application</b></li> <li><input checked="" type="radio"/> <b>Symbian OS v9.x Capabilities (as follows):</b>  <b>NetworkControl, MultimediaDD</b></li> </ul>		
<p><i>Testing Steps</i></p> <p>STEP1: Make a phone call using the application under test (i.e. one that is routing the call). Check that an application UI is displayed and that it can be interacted with.</p> <p><i>RESULT: When a call is routed via a third-party application instead of system Telephone application, the application UI is visible in the foreground. The Third-party application is <b>not</b> allowed to mimic or copy the system Telephone application user experience or UI (which may mislead the user).</i></p>		
<p><i>For Test Houses/Test Runs – Result of Test</i></p> <p><input type="checkbox"/> <b>PASS</b>      <input type="checkbox"/> <b>FAIL</b></p> <p><b>EXCEPTION(S)</b></p> <p><input type="checkbox"/> <b>CAP-02.EX1: Not Testable</b></p> <p>The application uses <code>NetworkControl</code> and/or <code>MultimediaDD</code> <i>Capabilities</i> but does not have telephony related functionality and accurate declaration is made upon submission.</p>		

Test ID <b>CAP-03</b>	Test Title <b>Manufacturer Disclaimer for VoIP Applications</b>	Estimated Test Time (minutes) <b>TBD</b>
<p><i>Test Description</i>  <b>A proper disclaimer is shown when a VoIP application is installed and launched for the first time.</b></p>		
<p><i>Not Required for:</i></p> <ul style="list-style-type: none"> <li><input type="radio"/> <b>Passive content</b></li> <li><input type="radio"/> <b>No Capabilities - Symbian OS v9.x</b></li> </ul> <p><i>Required for:</i></p> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> <b>Pre Symbian OS v9.x application</b></li> <li><input checked="" type="radio"/> <b>Symbian OS v9.x Capabilities (as follows):</b>  <b>NetworkControl, MultimediaDD</b></li> </ul>		
<p><i>Testing Steps</i></p> <p>STEP1: At install and also at launch of the VoIP application for the first time, check that the following disclaimer is shown to the user (or an equivalent local language version matching the locale setting of the phone).</p> <p style="text-align: center;">           You are about to use &lt;APPLICATION NAME&gt;, which is developed and owned by &lt;YOUR COMPANY NAME&gt;. The manufacturer of this device shall have no liability for any aspect of the application whatsoever, including its performance, call routing, intellectual property rights or support.         </p> <p>STEP2: The user can proceed with or cancel the install. If they proceed with the install, they should see the warning again on first usage in the form of a message or prompt – again giving the option to continue, or exit the application.</p> <p><i>RESULT: Disclaimer is displayed at installation and also at first launch of application..</i></p>		
<p><i>For Test Houses/Test Runs – Result of Test</i></p> <p><input type="checkbox"/> <b>PASS</b>            <input type="checkbox"/> <b>FAIL</b></p> <p><b>EXCEPTIONS:</b></p> <p><input type="checkbox"/> <b>CAP-03.EX1: Not Testable</b></p> <p>The application uses <code>NetworkControl</code> and/or <code>MultimediaDD</code> <i>Capabilities</i> but does not have telephony related functionality and accurate declaration is made upon submission.</p>		

Test ID <b>CAP-04</b>	Test Title <b>Active VoIP or CSD Call &amp; Notification of An Incoming Call</b>	Estimated Test Time (minutes)
<p><i>Test Description</i> <b>When a VoIP call is active, any aural notification of an incoming GSM/UMTS call is not too loud.</b></p>		
<p><i>Not Required for:</i></p> <ul style="list-style-type: none"> <li><input type="radio"/> <b>Passive content</b></li> <li><input type="radio"/> <b>No Capabilities - Symbian OS v9.x</b></li> </ul> <p><i>Required for:</i></p> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> <b>Pre Symbian OS v9.x application</b></li> <li><input checked="" type="radio"/> <b>Symbian OS v9.x Capabilities (as follows):</b> <b>NetworkControl, MultimediaDD</b></li> </ul>		
<p><i>Testing Steps</i></p> <p>Note1: Call Waiting tone is described at <a href="http://en.wikipedia.org/wiki/Call_waiting">http://en.wikipedia.org/wiki/Call_waiting</a></p> <p><b><u>Test Case A</u></b></p> <p>STEP.A1: Start the application.            STEP.A2: Plug in the headset.            STEP.A3: Select <b>Default</b> profile (if applicable).            STEP.A4: Initiate a VoIP call.            STEP.A5: Make a GSM/UMTS phone call using another device to the device being tested.            STEP.A6: When an incoming GSM/UMTS call is received, check that a call waiting tones provided.</p> <p><i>RESULT: When the device profile is other than <b>Silent</b> a call waiting tone is heard as a notification for an incoming GSM/UMTS call. This is to avoid the notification being played too loud so that it will not harm the user's hearing.</i></p> <p><b><u>Test Case B</u></b></p> <p>STEP.B1: Start the application.            STEP.B2: Use the device without a headset (but not with a loudspeaker).            STEP.B3: Select <b>Default</b> profile (if applicable).            STEP.B4: Initiate or receive a VoIP call.            STEP.B5: Make a GSM/UMTS phone call using another device to the device being tested.            STEP.B6: When an incoming GSM/UMTS call is received, check that only a call waiting tone is provided.</p> <p><i>RESULT: When the device profile is other than <b>Silent</b>, only a call waiting tone is heard as a notification for an incoming GSM/UMTS call. This is to avoid the notification being played too loud so that it will not harm the user's hearing.</i></p> <p><b><u>Test Case C</u></b></p> <p>STEP.C1: Ensure the VoIP application is in the foreground.            STEP.C2: Check the user is able to answer or reject the GSM/UMTS call.            STEP.C3: If the GSM/UMTS call is answered, end the call with the <b>End</b> key (if available).</p>		

Test ID <b>CAP-04</b>	Test Title <b>Active VoIP or CSD Call &amp; Notification of An Incoming Call</b>	Estimated Test Time (minutes)
<p><i>RESULT: GSM/UMTS incoming call can be answered or rejected.</i></p> <p><i>If GSM/UMTS call is answered:</i></p> <ul style="list-style-type: none"> <li>○ VoIP call goes on hold or ends.</li> <li>○ VoIP application comes to foreground (S60 only).</li> <li>○ After ending the GSM/UMTS call, VoIP call goes back to active if on hold.</li> </ul> <p><i>If GSM/UMTS call is rejected by user:</i></p> <ul style="list-style-type: none"> <li>○ VoIP application comes to foreground.</li> <li>○ VoIP call continues as before the incoming call.</li> </ul> <p><b><u>Tests Case D</u></b></p> <p>STEP.D1: Ensure the VoIP application is started and in the foreground.</p> <p>STEP.D2: Initiate or receive a VoIP call.</p> <p>STEP.D3: Make a GSM/UMTS phone call using another device to the device being tested.</p> <p>STEP.D4: Answer the call.</p> <p>STEP.D5: If the GSM/UMTS call is answered, end the call with the <b>End</b> key (if available)</p> <p><i>RESULT: When the device profile is other than <b>Silent</b>, only a call waiting tone is provided as a notification for an incoming GSM/UMTS call. This is to avoid the notification being played too loud so that it will not harm the user's hearing.</i></p> <p><i>RESULT: GSM/UMTS incoming call can be answered.</i></p> <p><i>If GSM/UMTS call is answered:</i></p> <ul style="list-style-type: none"> <li>○ VoIP call goes on hold or ends</li> <li>○ After ending the GSM/UMTS call, VoIP call goes back to active if on hold</li> </ul> <p><b><u>Tests Case E</u></b></p> <p>STEP.E1: Initiate or receive a VoIP call</p> <p>STEP.E2: Switch the VoIP application to the background using the <b>Task List</b></p> <p>STEP.E3: Make a GSM/UMTS phone call using another device to the device being tested</p> <p>STEP.E4: Answer the GSM/UMTS call</p> <p>STEP.E5: End the call with the <b>End</b> key</p> <p><i>RESULT: GSM/UMTS incoming call can be answered.</i></p> <p><i>When GSM/UMTS call is answered:</i></p> <ul style="list-style-type: none"> <li>○ VoIP call goes on hold or ends</li> <li>○ After ending the GSM/UMTS call, VoIP call goes back to active if on hold</li> <li>○ VoIP application remains in the background</li> </ul> <p><b><u>Tests Case F</u></b></p> <p>STEP.F1: Initiate or receive a VoIP call</p> <p>STEP.F2: Switch the VoIP application to the background using the <b>Task List</b></p>		

Test ID <b>CAP-04</b>	Test Title <b>Active VoIP or CSD Call &amp; Notification of An Incoming Call</b>	Estimated Test Time (minutes)
<p>STEP.F3: Make a GSM/UMTS phone call using another device to the device being tested</p> <p>STEP.F4: Reject the GSM/UMTS call with the <b>End</b> key</p> <p><i>RESULT: GSM/UMTS incoming call can be rejected.</i></p> <p><i>When GSM/UMTS call is rejected:</i></p> <ul style="list-style-type: none"> <li>o <i>VoIP call continues as before the incoming call</i></li> <li>o <i>VoIP application remains in the background</i></li> </ul>		
<p><i>For Test Houses/Test Runs – Result of Test</i></p> <p><input type="checkbox"/> <b>PASS</b>      <input type="checkbox"/> <b>FAIL</b></p> <p><b>EXCEPTIONS</b></p> <p><input type="checkbox"/> <b>CAP-04.EX1: Not Testable</b></p> <p>The application uses <code>NetworkControl</code> and/or <code>MultimediaDD Capabilities</code> but does not have telephony related functionality and accurate declaration is made upon submission.</p>		

Test ID <b>CAP-05</b>	Test Title <b>Emergency Call When VoIP or CSD Application is Open</b>	Estimated Test Time (minutes) <b>TBD</b>
<p><i>Test Description</i>  <b>The user must be able to make an emergency call (911, 999,112 etc) when VoIP application is open.</b></p>		
<p><i>Not Required for:</i></p> <ul style="list-style-type: none"> <li><input type="radio"/> <b>Passive content</b></li> <li><input type="radio"/> <b>No Capabilities - Symbian OS v9.x</b></li> </ul> <p><i>Required for:</i></p> <ul style="list-style-type: none"> <li><input checked="" type="radio"/> <b>Pre Symbian OS v9.x application - VoIP Applications only</b></li> <li><input checked="" type="radio"/> <b>Symbian OS v9.x Capabilities (as follows):</b>  <b>NetworkControl, MultimediaDD - VoIP Applications only</b></li> </ul>		
<p><i>Testing Notes</i></p> <p>Note1: Before carrying out this test, please ensure that you have adequate permissions from the appropriate authorities for making emergency calls for testing purposes.</p> <p><i>Testing Steps</i></p> <p><b><u>Test Case A – SIM card in phone, VoIP application in foreground and no VoIP call active.</u></b></p> <p>STEP.A1: Insert a SIM card into the device</p> <p>STEP.A2: Open VoIP application but do not make a call</p> <p>STEP.A3: Ensure the 'home' screen has focus (e.g. telephone, Idle screen) and the VoIP application is in the background</p> <p>STEP.A4: Dial 112 or emergency number for phone locale  <a href="http://en.wikipedia.org/wiki/Emergency_number">http://en.wikipedia.org/wiki/Emergency_number</a></p> <p>STEP.A5: The call is routed over GSM/UMTS</p> <p><i>RESULT: Emergency calls can be initiated from the home or 'Idle' screen.</i></p> <p><b><u>Test Case B – SIM card in phone, VoIP application in foreground and VoIP call active.</u></b></p> <p>STEP.B1: Insert a SIM card into the device</p> <p>STEP.B2: Open VoIP application and make a call</p> <p>STEP.B3: Ensure the 'home' screen has focus (e.g. telephone, Idle screen) and the VoIP application is in the background</p> <p>STEP.B4: Dial 112 or emergency number for phone locale  <a href="http://en.wikipedia.org/wiki/Emergency_number">http://en.wikipedia.org/wiki/Emergency_number</a></p> <p>STEP.B5: The call is routed over GSM/UMTS and the WLAN/VoIP call is ended</p> <p><i>RESULT: Emergency calls can be initiated from the home or 'Idle' screen.</i>  <i>RESULT: The VoIP call is ended or put on hold.</i></p>		

Test ID <b>CAP-05</b>	Test Title <b>Emergency Call When VoIP or CSD Application is Open</b>	Estimated Test Time (minutes) <b>TBD</b>
--------------------------	--	---

**Test Case C – SIM card in phone, VoIP application in the background and VoIP call active.**

STEP.C1: Insert a SIM card into the device

STEP.C2: Open VoIP application and make a call, put the application to the background

STEP.C3: Ensure the **Home** screen is active (e.g. Telephone, Idle screen)

STEP.C4: Dial 112 or emergency number for phone locale

[http://en.wikipedia.org/wiki/Emergency\\_number](http://en.wikipedia.org/wiki/Emergency_number)

STEP.C5: The call is routed over GSM and the WLAN/VoIP call is ended or put on hold

*RESULT: Emergency calls can be initiated from the home or 'Idle' screen.*

*RESULT: In Test Case C, the VoIP call is ended or put on hold (either can pass).*

**Test Case D – No SIM card in phone (where test device allows “Offline” or “Flight Mode”)**

STEP.D1: Repeat test cases A to C without a SIM card present (but where there is GSM/UMTS coverage in the area).

*RESULT: In all cases, emergency calls can be initiated from the **Home** or **Idle** screen.*

*For Test Houses/Test Runs – Result of Test*

**PASS**       **FAIL**

**EXCEPTION(S)**

**CAP-05.EX1: Not Testable**

The application uses `NetworkControl` or `MultimediaDD` *Capabilities* but does not have VoIP functionality

## 6 Appendix 1

### 6.1 Waivers

If you are unable to conform to a Test Case in the Symbian Signed Test Criteria and believe there is a valid reason why you are unable to conform you may be able to apply for a Waiver. Waivers are only granted in special circumstances and developers should make all attempts to conform to the Test Criteria.

**Important note:** Waivers are not acceptable for *Express Signed* submissions and any submissions made with Waivers will be failed. Submissions that require a waiver must be submitted through Certified Signed.

**Important note:** Waivers are a once-only exception to a particular test case. Your application can be signed with an appropriate waiver once. At future testing and signing instances, your application (even updates) is either expected to pass the test case or you will have to re-apply for the waiver.

The waiver document is located here:

<http://developer.symbian.com/sswaiver>

This document must be completed and sent to the Test House that you use for Certified Signed. The Test House will contact the Phone Manufacturer and/or Network Operator for approval.

### 6.2 Stub .SIS files

If your application is being included in a firmware build it may be desirable for the Phone Manufacturer to include your stub file in the firmware too. Stub SIS files do not require any capabilities and do not need to be Symbian Signed if they are to be included in the ROM. If, however, your application is being shipped on a memory card it may be required to be Symbian Signed. In this case the SIS file should be submitted in the standard way to be Symbian Signed as a “Passive Content” file type. The main application associated with the stub file should be Symbian Signed before the stub file is.

### 6.3 Shared Dlls and ECom plugins

If you are submitting a “Shared Dll” or “ECom plugin” you should also provide a test harness with the submission that tests all the functionality of the Dll and passes all the Symbian Signed Test Criteria. The test harness must test all functionality of the dll and pass all the Symbian Signed Test Criteria. It should also provide a test report for the “Test Harness” and expected results. In order to provide the Test House with as much information as possible you should provide information in your release notes or user guide explaining the nature of the submission and how to setup and run the test harness.

### 6.4 Embedded .SIS files

Embedded .SIS files should follow the process:

- Embedded SIS file is submitted through Symbian Signed ensuring that complete information is entered for “Declarative Statements”
- Release Notes include information on what application this SIS file will be embedded to
- Test House will go through the Submission Criteria for the submission
- Test House will sign the Embedded SIS file
- Company will submit the application as named in the Release Notes for the Embedded SIS file including the embedded SIS file

If an Embedded SIS file is signed and there is no subsequent submission for the named application included in the Embedded SIS file Release Notes, the company involved will be pursued and the Embedded SIS file revoked.

## Disclaimer

The information contained in this document is for general information purposes only and should not be used or relied on for any other purpose whatsoever. While Symbian has taken great care in the preparation of this document, Symbian makes no warranty or guarantee about the suitability or accuracy of the information contained in this document.