

Forum Nokia employee located in Tampere, Finland.

Working as the technical lead for the Professional Support team my focus is in making sure that developers are getting the best possible service when calling on our help for solving a technical problem. I am also working on channeling developer feedback and requirements to the S60 platform.

## My Blog

### Lucian Tomuta's Forum Nokia Blog

- Load your application in a running emulator

Everybody loves to hate the Symbian SDK emulator. The first reaction to it usually is: "It is slow, really slooooooow, it takes ages to start it" ;) Yes, it is true, the emulator is slow, but considering the task it has to perform and the complexity of the system it emulates such slowness is to be expected. However, the problem is not as bad as one may think. Yes, the emulator takes time to load, but then why don't you start it in advance and let it load while you browse your recent emails, so that you have it up-and-running at the moment you really need it. Many developers already with some Symbian experience will tell you that you cannot do that. But yes, you can! The following video will show a basic Qt application being created, built and run into an already running emulator. Coming up next: Rebuild without restarting the emulator.

- Build without an SDK, test without a device

How many SDKs do I need? Which SDK is best for me? For a newcomer to the Symbian development world those are among the first questions asked. There are many S60 SDKs on offer, some with feature packs and plugins, some with compiler specific binaries and on top of that many plug-ins. Quite a lot to handle and figure out and nobody likes having all possible SDKs installed. So what will happen now that Qt is added to the picture? I can design my app using Qt for Windows (or Linux, or Mac) and, according to the "code once, compile anywhere" promise I should be able now to compile this application for Symbian and Maemo devices, with minimal changes. But that means that now I need to install at least one Symbian SDK for building, or a Maemo one, and I should have at least one smartphone for testing. Or do I? Yes, probably one will have an SDK installed and use its emulator for testing some really complicated solutions, with Symbian components part of the design. And yes, nothing can really replace hand-on-testing. But for simple use cases and for getting started trials one possible solution could be building and testing in the cloud. What you need is an IDE plugin, web hosted SDK and build tools and Forum Nokia's Remote Device Access solution, all mixed together in a rather simple build, deploy and test solution. Here's a live demo (boring parts speeded-up a bit) of such a solution: So, what do you think? Do you like the idea? Would you use it for your projects?

- The ultimate\* software installer debug guide

There are many things that can go wrong with a SIS installation package and unfortunately the software installer component provides no logging mechanism for developers to be able to precisely pinpoint the point of failure. Even the errors returned by the installer engine are filtered out so that the UI layer gets the minimal set of rather generic error messages to display for the user. But while "Unable to install" tells all the user needs to know about the fact that the application will not show up in his/her Applications menu, that message offers no hint as to what exactly the application developer should be fixing in order to have a successful install. Over time developers have created a comprehensive list of error messages and possible resolutions for the problem. In this blog post I will provide an up-to-date and hopefully complete list of errors as well as a nice little hack that will allow a bit more detailed resolution of the root cause of the error. The hack, available on S60 3rd Edition

Feature Pack 2 and later Symbian based devices, consists in creating one "flag" file, c:\temp\swierrors.txt. If this file exists the software installer's UI will augment the error message with the actual error code which triggered the error message currently displayed. See below screen captures showing the same install time error without and then with the flag file created. Here are the error messages and the extra bit of information that can be squeezed out of them: "Update error"

4 Cannot overwrite file 43 An upgrade failed because the package being installed is not a valid upgrade of the package on the device. 45 The package contains an invalid eclipsing file which is already in the device other than ROM. "Cannot install! This component is built-in!" 5 The file is already in ROM and cannot be eclipsed "File is corrupted!" 17 File to install is not a recognized SISX file. 14 A file needed to proceed cannot be found. 13 A filename given inside the SISX package is not valid. 9 SISX file corrupt, checksums stored in file do not match computed value. "Unable to install. Disk not present." 10 A removable media drive had no media while trying to access it. "Cannot read file!" 11 We cannot read a file which is needed for us to proceed. "Cannot delete file!" 12 We are forbidden to delete a certain file. "Unable to install. Insufficient memory." 19 There is not enough free memory for installer's UI components (e.g. drive selection dialog) to be constructed and displayed. 15 There was insufficient space left on the drive to perform an operation. "Cannot grant capabilities requested by the application" 16 The applications requires system capability it is not signed for. "Certificate check failed. Contact application supplier." 20 Cannot find the certificate. 21 An error has occurred during signature validation. 22 There is no signature to verify. 23 The signature could not be validated. 24 The certificate is not trusted for code signing usage. 25 Cannot find the certificate policies. 26 A signature resolving to one of the mandatory certificate cannot be found. "Only signed applications can be installed. Check your Installation settings in Application Manager." 42 The signature validates correctly, but chains back to a self-signed root "Certificate expired" 36 Certificate expired" Certificate validity dates do not match with the phone's date. Check your date settings." 37 Certificate validity dates do not match with the phone's date "Online certificate check could not be conducted. Check your settings." 30 The OCSP server reply is invalid. 27 The OCSP server URL provided is invalid. 29 It was not possible to validate the OCSP server response signature. "Online certificate check failed." 28 It is not possible to obtain the certificate status. 31 The certificate status information is invalid. 32 The certificate status is unknown. 35 The certificate status is unknown, but this is expected for self-signed. "Certificate revoked" 33 The certificate has been revoked. "Cannot install. Access denied." 34 Attempt to perform illegal operation "Unable to install. Install the base package first." 18 A base package was missing when installing an augmentation or a partial upgrade "Cannot uninstall partial upgrades" 41 Partial update package cannot be uninstalled. "Cannot install a protected application from untrusted supplier" 36 The package contains a file with a protected SID which is not signed for. 40 The package contains a file with a protected VID which is not signed for. 44 The package uid is protected but not signed for. "Unable to install. Constrained by the certificate." 38 Constraints imposed by a developer mode certificate have been exceeded. "Unable to install" or "Uninstallation failed" 6 Missing a dependency. 7 A dependency exists on a specific version of an application which is different from the one installed. 8 A dependency requires a version of an application which is greater than the one installed. 3 Anything and everything else. Of course, obeying Murphy's law, most of the error messages we're likely to see will be still "Unable to install (3)" so don't forget to consult and update the developer contributed list of common PKG scripting mistakes. Update: The attached sis file can be used to easily create (and remove) the flag file on/from your device. My only request would be not to deploy such hack (be it this or the errrd one) with your application. It might make some limited sense on beta deployments but there's no need to "improve" end-user's experience with it. \*Read "ultimate" as the best we have so far. Hopefully at some point the installer will have a logging feature (currently

enabled in debug builds only) which would let us read directly and exactly what the problem was and where.

- Nokia N97 SDK v1.0 released

Those engaged in developing applications for Nokia N97 will be interested to know that the final version of the Nokia N97 SDK has been released today. The new SDK version provides: Improved support of home screen features. Updated SDK documentation. Support for the Chinese version of the full keyboard. Updated example applications. Updated default emulator theme. Fixes for all other important bugs. Read more about features and known issues in SDK's release notes. It is probably worth making clear a few points about this SDK: Nokia N97 is and S60 5th Edition (Symbian^1) device and it is fully compatible that that platform release. Any application developed for S60 5th Edition devices using the S60 5th Edition (Symbian^1) SDK will be compatible with smartphone as well. The Nokia N97 SDK is only provided as an alternative, mostly beneficial to those developing applications targeting N97's specific features, features not supported by the platform release and thus not supported by other devices based on it, as for example the home screen widgets. While the APIs in the S60 5th Edition SDK are covered by platform's binary compatibility promise, all the additional device specific APIs provided by the Nokia N97 SDKs are provided "as is", without binary compatibility promise. That's it. Enjoy the new release and don't forget to let us know what you think about it.

- Good news on the Horizon

The Symbian Foundation announced a new application-publishing platform, called Horizon, which is intended to help developers, big and small, to publish their Symbian applications on the software market. Under this scheme the Symbian Foundation will take care of certifying the application through Symbian Signed and will promote it to the partner application stores. All that for free while the author maintains full ownership of the application and will receive 100% of its store generated income. Among the stores likely to participate in this program is of course Nokia's Ovi Store. To learn more about this platform visit Horizon's main page and the Horizon FAQ as well. If you are a software developer currently "releasing" unsigned applications you should definitely have a look at this offering.

- Qt for S60 "Tower" demo running on Nokia N97

- Qt for S60 "Tower" released

You will most probably be glad to hear fresh news from the Qt for S60 port team. They have a new release\* out, it is called Tower and it is based on Qt 4.5.2. There are many new things and many improvements on the existing features, you can see them all listed in the Detailed list of changes for "Tower". To name just a few: "Experimental" support for QtWebkit (must be enabled at configuration time) Phonon support but without the actual playback capabilities. QSql with sqlite3 backend Improved exceptions safety for Qt Core library (this comes with a patch for the Nokia C/C++ compiler and some really promising documentation, judging by the title ) Input methods All these and a lot more in an 126 MB binary installer which is smart enough to detect all compatible SDKs in your system and install "Tower" support in all of them. (From what I see the installer needs a bit of optimization or I need a new computer ) You can find all the binaries, documentation and support information on the Qt for S60 "Tower" pre-release site. Coming soon support for Samsung's S60 devices, an updated version of the Qt for S60 Developer Library here on Forum Nokia and a new release of the Mobility Extensions APIs. \*Please note that this is still a technology preview release and it is not yet ready for production grade application development.

- Symbian Foundation promo videos

Want to learn more about the Symbian Foundation? See More Symbian Foundation beta invites

- Happy Birthday Symbian Signed!

The Symbian Foundation is brand new but one of their services is celebrating it's 5th birthday today. You can find on the Symbian.org blog a brief summary of what the service is an some rather interesting usage stats. But more importantly you will find there a glimpse of what they are preparing

for the near future: " ... Lower cost Publisher IDs available to anyone Lower cost signing and lower barrier to signing A revamp and rethink of the Symbian Signed Test Criteria ..." Truly promising plans, I hope that we will hear more details about these soon. Until then: Happy Birthday Symbian Signed!

- More Symbian Foundation beta invites

You may have seen in Mark's recent post an invitation to become a Symbian Foundation site beta tester user. Don't worry about missing the deadline and no, you don't have to become a Twitter user just to get an invitation, the invitations are here, one message away for you. So, are you a software developer curious about the new Symbian Foundation web site? Want to be the first one posting a technical question on that forum? Want to be the first one who answers one? Want to see open sourced code? Let's do it then, let's get you in! Who can get an invitation? Any Symbian C++ developer active in the Forum Nokia developer community. How many invitations are available? There are enough for everyone. However, the distribution process is manual and time consuming so the most active contributors on the Forum Nokia Discussions Board & Wiki will have priority. How can I apply? To get the username and password that will give you access to the web site all you have to do is send me a private message through the Forum Nokia Discussions Board. Easy!