

INSTALLATION GUIDE FOR THE NOKIA REMOTE SENSING PLATFORM

Authors: Elena Balandina, Dirk Trossen
Nokia Research Center / Helsinki

ABSTRACT

The guide aims at providing sufficient information for installing and start using Nokia Remote Sensing (NORS) platform.

For that, the first chapter gives a general overview of the NORS platform. The second chapter provides a list of development tools needed for setting up NORS development environment. In addition, instructions can be found regarding the required tools to be downloaded. The third chapter gives guidance as to compile the NORS platform. The following chapter specifies requirements for runtime environments for NORS platform. Information regarding copyright, disclaimer, distribution information, and so on can be found in the Annex of the document.

Version:	1.0
Authors:	Elena Balandina, Dirk Trossen

TABLE OF CONTENT

GLOSSARY	2
1. INTRODUCTION	3
2. REQUIREMENTS FOR DEVELOPMENT ENVIRONMENT FOR NORS PLATFORM	3
3. NORS INSTALLATION	4
3.1 PACKAGE CONTENTS.....	4
3.2 INSTALLATION OF THE APPLICATION SERVER	4
3.3 INSTALLATION OF THE INTERMEDIARY GATEWAY AND ECLIPSE CONFIGURATION.....	5
4. REQUIREMENTS FOR RUNTIME ENVIRONMENT FOR NORS PLATFORM	6
4.1 RUNTIME ENVIRONMENT FOR THE APPLICATION SERVER	6
4.2 RUNTIME ENVIRONMENT FOR THE INTERMEDIARY GATEWAY	7
5. INSTALLATION OF THE PHONE GATEWAY APPLICATION	8
6. REFERENCES	8
ANNEX A	9
A.1 COPYRIGHT INFORMATION AND DISCLAIMER.....	9
A.2 TRANSLATIONS	9
A.3 INFORMATION DISTRIBUTION	9

GLOSSARY

API	Application Program Interface
AS	Application Server
GPRS	General Packet Radio Service
IG	Intermediary Gateway
IMEI	International Mobile Equipment Identity number
M2M	Machine to Machine modem
MIDP	Mobile Information Device Profile
N-RSA	Nokia Remote Sensing Architecture
NORS	NOKia Remote Sensing platform
PG	Phone Gateway
SDK	Software Development Kit
WTK	Wireless ToolKit

1. INTRODUCTION

The document provides basic information required for installing and start using the Nokia Remote Sensing (NORS) platform. The NORS platform is a test platform according to the Remote Sensing Architecture (N-RSA), as defined in [NRSA].

One of the main characteristic of the N-RSA is that intelligence required for information processing is moved closer to the actual sensors, namely to the actual collecting point hereafter call Intermediary Gateway (IG). With this, it potentially reduces the overall amount of sensor data transmitted to the back end system by sending data only when necessary and possibly even in already preprocessed form (e.g., aggregated or fused). This is achieved through implementing a subscription & notification delivery of sensing data, which is locally obtained according to given queries in these subscriptions. With this, data is only sent when it is necessary through performing appropriate preprocessing of the data (e.g. data aggregation) in the intermediary gateway. It is assumed that the IG has the capability of performing certain reasoning, and reasoning requires proper sensor and sensor data abstractions as well as capabilities of combining simpler sensor data into more complex one. This also assumes the existence of a proper abstraction model for sensor data and aggregated sensor information. Such abstraction model will be the basis for the query language that is used for acquiring sensor data and aggregated data within the N-RSA.

2. REQUIREMENTS FOR DEVELOPMENT ENVIRONMENT FOR NORS PLATFORM

This chapter contains information about basic requirements for the system hardware and software.

The minimum hardware requirements are:

- 500 MB hard disk space
- 256 MB system RAM
- 800 MHz Pentium III CPU

Before installing NORS platform, the following software needs to be installed on your PC (for each package we specify the version with which the NORS platform was tested, but other versions, especially newer might also work fine):

1. *J2SE Runtime Environment 5.0 Update 4*. The package is available for free from <http://java.sun.com/j2se/downloads/index.html>. Be sure that you download the correct installation package for your operational system.
2. *J2ME Wireless Toolkit 2.2*. The package is available for free (using your Sun Online Account) from <http://java.sun.com/j2me/download.html>. If you don't have a Sun Online Account, you can create it for free by filling in a registration on Sun Developer Network web (the same link can be used). NOTE: it is important to install J2ME WTK into the proposed default directory, as otherwise you will have to do a lot of additional configuration to make it work properly with the system.
3. *Eclipse SDK 3.1.2*. The package can be downloaded from one of the mirror sites, a list of which can be found at <http://www.eclipse.org/downloads/>. The downloaded package can be extracted into any directory. The package does not require any special installation procedure, so it is enough to unpack it and after that it can be used.
4. *EclipseME 1.2.3*. The package is available for free from <http://eclipseme.org/docs/installation.html>. The web page also provides instructions on how to install and configure EclipseME. In nutshell, there are

three basic steps that have to be performed to successfully complete the installation:

- a) Installation of the wireless toolkit (you should have it already according to the previous bullet items)
- b) Installation of EclipseME has to be done according to the instructions on the web <http://eclipseme.org/docs/installEclipseME.html>. We recommend installing it using the downloaded archive site option. The web page contains a very extensive explanation of the installation procedure including screenshots of all major steps (some of the screens are out of date, but the right behavior is intuitive clear).
- c) Configuring of [EclipseME](#) and [Eclipse](#). It is very important to carefully follow instructions provided on <http://eclipseme.org/docs/configuring.html>. Note that you can skip this step if you are planning to use Eclipse only together with NORS, as you will need to make Eclipse configuration anyway after loading NORS environment.

3. NORS INSTALLATION

3.1 Package Contents

The NORS package consists of three parts:

- The back end server application component implements the desired functionality for the vertical application. For that, it accesses the middleware (API provided by this package) component in order to implement the required remote data gathering functionality within the vertical application. The package includes an example for such vertical application component under the GPL license. Other proprietary applications can be written on top of the middleware without using the given application component.
- The back end server middleware component implements the N-RSA functionality according to the N-RSA specifications. It is licensed under LGPL and linked to the vertical application component.
- The Intermediary Gateway component implements the gateway functionality according to the N-RSA specifications and is to be installed on MIDP2.0-compliant mobile devices.

Please see the `license_server.txt` and `license_middleware.txt` files in the package for the exact license terms of the different parts

3.2 Installation of the Application Server

The Application Server (AS) consists of 2 parts, which are covered by different license agreements:

- NORS Middleware library – implements N-RSA according to specification and provides an API to access the main functions. Please refer to the implementation documentation [NORSb] for the exact API description. The license for this part is described in the `license_middleware.txt` file of the package.
- Example implementation of an application Server – demonstrates the functionality of NORS through implementing simple server functionality, like initiating remote sensing subscriptions, displaying discovered sensor functionality and dynamically

generating web pages with the received sensor data. More detailed description of this functionality is given in [NORSa]. Modifications to this code are covered under the license agreement as described in license_server.txt of the package. The current package contains two versions for the application server:

- Application Server with support of the Sierra wireless GPRS card, which addresses a case when testbed has M2M devices;
- Application Server without GRPS support, to be used for cases where only mobile phones are used as intermediary gateways.

The packages contain project files for compiling the package under Microsoft Visual C++ 6.0 or higher and Dev-C++ version 4.9.9.1 (only for “without Sierra” part of the package) or higher. We selected Dev-C++ as a secondary compiling platform since it is a free of charge compiler package available from <http://www.bloodshed.net/dev/devcpp.html>.

You need to install the application server through unpacking the provided package in any directory that is most suitable for you and compile the package using the VC or Dev-C++ environment. If you downloaded NORS as a single package, you should unpack the entire NORS package in a directory. You can then find the server package part in the subdirectory /RSA_AS.

3.3 Installation of the Intermediary Gateway and Eclipse configuration

- 1) Unpack the Intermediary Gateway (IG) package. If you downloaded NORS as a one package, the IG package is located in the /Midlet subdirectory of the extracted package.
- 2) Move source files of the IG package to “C:/USERS/Code/” directory. NOTE: some of the directory files starts with ‘.’ symbol which might be incorrectly interpreted by some windows shell programs, so in order to prevent possible problems we recommend to copy the directory using Windows “My computer” shell.
- 3) Start Eclipse environment by executing eclipse.exe from your Eclipse directory. Depending on the version of Eclipse package and default configuration settings, the system might automatically request you to define the current workspace at the boot time or otherwise after booting program you have to set it manually by starting “File/Switch workspace” menu option. The workspace must be set to the IG package directory – “C:/USERS/Code/”.
- 4) As it was mentioned before, now you have to make setting of your EclipseME inline with the recommended EclipseME and Eclipse settings.
- 5) The next step is to compile NORS IG package in Eclipse environment. The provided package consist on two parts:
 - a. Imlet – is an Intermediary Gateway implementation for m2m devices, e.g. former Nokia product called N12, currently manufactured and distributed by Aplicom: http://www.aplicom.com/gsm_modules.html. Before compiling this module make sure that the J2ME Platform Definition setting is correct. For this place mouse onto “N-RSA Imlet” and on right click choose “Properties”. In properties page open J2ME section and set J2ME Platform Definition to “J2ME Wireless Toolkit 2.2 MIDP 1.0 Platform”. Also from the same list of Properties, you might check settings of “Java build path/Libraries” and, if needed, change (by using “Edit” button) location of classes.zip file, by default it should be in “C:/USERS/Code/” directory.
 - b. Midlet – is an Intermediary Gateway implementation for MIDP 2.0 devices, e.g. most of Nokia Series60 and Series40 phones. To compile this part, you have to verify configuration of the J2ME Platform Definition. For this, point mouse to “N-RSA Midlet” and on right click “Properties”. In properties page,

open J2ME section and set J2ME Platform Definition to “J2ME Wireless Toolkit 2.2 MIDP 2.0 Platform”. Also from the same list of Properties, you might want to check settings of “Java build path/Libraries” and, if needed, change (by using “Edit” button) location of mmapi.jar file.

- 6) Now to finish preparation of IG package, you need to generate new jar and jad files for IG parts. For that, select by mouse the first project, e.g. “N-RSA Imlet”, click on it the right button and select the “J2ME/Create package” option. If Imlet was compiled without errors, no error reports will be shown and the new N-RSA_Imlet.jar and N-RSA_Imlet.jad files will appear in directory C:/USERS/Code/N-RSA_Imlet/deployed/. Repeat the same procedure for the “N-RSA Midlet” package. Correspondingly for Midlet package, the N-RSA_Midlet.jar and N-RSA_Midlet.jad files will be generated in directory C:/USERS/Code/N-RSA_Midlet/deployed/. In case of errors we recommend first to check if the system configuration is correct (especially EclipseME and Eclipse settings) and if the error remains please inform about such problem authors of this document.

Remember that if you make any changes to the project code you need to save your project changes in Eclipse manually by using “File/Save” (Ctrl+S) or “File/Save As” or “File/Save All” (Ctrl+Alt+S) option. The Java compiler will recompile the code and display potential errors in the status section of Eclipse.

4. REQUIREMENTS FOR RUNTIME ENVIRONMENT FOR NORS PLATFORM

This section gives an initial overview how to start and use the application server and gateway software. More details about the specifics of the application server and gateway user interface are given in [NORSa].

4.1 Runtime environment for the Application Server

As it was mentioned before, the Application Server should be written in C++ using the NORS API provided as a part of the Middleware component package and described in [NORSb]. If you use your own implementation of the Application Server, you can skip this section, as it describes how to use example implementation of the Application Server provided as a part of the RSA_AS package.

Before starting the Application Server, please make sure that your PC is connected to a public IP network in order to be addressable and reachable for the intermediary gateways. Also you need to remove personal firewall or other protection system in order to allow opening ports 21, 9000 and 80. For starting the Application Server you need to execute N_RSA_AS_wo_Sierra.exe program that is located in Exec subdirectory inside your /RSA_AS directory. You also can use the executable scripts (.bat files), which simply create a loop to automatically restart the main executable after failure.

In addition we can say that the current NORS platform was tested with Sierra AirCard 710 card and all file examples for inserting Sierras support into NORS platform are done for this particular card. But Sierra Wireless is announcing the End of Life for the AirCard 710 for GSM/GPRS networks, so in case somebody has another card in the first place he/she will need to adapt the GPRS/SMS code. If you want to use AS with GPRS support of Sierra wireless card (which is used when having M2M devices in the testbed) you should install appropriate Sierra SDK on your PC.

In order to receive your free Developer's Central account and to obtain access to Sierra SDK packages you need to fill out the following application form at Sierra web page: <http://www.sierrawireless.com/services/developers/registration.aspx>. Then create Sierra subdirectory in /RSA_AS directory and copy in it all necessary .lib and header files. The file list is as follows: Swi32_AC710.lib, SwiApi.h, SwiApiCm.h, SwiDataTypes.h, SwiMpApiCm.h, SwiMpStructsCm.h, SwiRcodes.h, SwiSms.h, SwiStructs.h, SwiStructsCm.h, also copy Swi32_AC710.dll file to the /RSA_AS/Exec directory. After

these files are in their proper location, you can execute N_RSA_AS.exe located in the /RSA_AS/Exec subdirectory.

A set of gateways with their phone numbers is specified in the GWphones.txt file, located in the same directory where the executables are located. The default file name can be changed to any preferred name by using “Settings” button at the N-RSA Middleware Status window, which appears when starting the application server. The file format syntax is self-explanatory and the file header contains enough comments to allow you to make proper configuration of the gateway set. Basically definition of each phone-gateway consists of 4 lines, where the meaning of lines is as follows:

1. Phone number, starting with prefix “+”.
2. International Mobile Equipment Identity (IMEI) number. On the Symbian phone the IMEI number can be obtained by typing the following command – *#06#. Usually the IMEI number is a 15-digits number.
3. Text specification of the phone (gateway) location and/or ownership.
4. The wakeup SMS parameter, 0 = no wakeup SMS, 1 = wakeup SMS. Note that the phone number entry in the configuration file does not really matter if you are not using SMS wakeup for phones (important only for M2M devices).

Also in the same directory you can find file scripting.txt. This is an N-RSA acquisition scripting file, i.e. it is part of the N-RSA application server not the N-RSA middleware. This file includes acquisition requests to be executed in scripting mode, i.e. once a particular gateway connects to the application and publishes the sensor data acquisitions are created with the given queries. Note that one can create several acquisitions with each gateway. The current file format is very simple, you have to specify: gateway number, number of queries, query strings (one in each line).

4.2 Runtime environment for the Intermediary Gateway

The only requirement for Intermediary Gateway is the presence of MIDP 2.0 on your mobile device(s) or the presence of an emulator for MIDP on your development PC. For compiling and modifying code make sure that wireless toolkit on your PC contains the following jar files. If you use J2ME Wireless Toolkit 2.2, these files should be located in directory C:/WTK22/lib.

- For Imlet:
 - cldcapi10.jar
 - midpapi10.jar
- For Midlet
 - mmapi.jar
 - cldcapi11.jar
 - wma20.jar
 - midpapi20.jar
 - jsr082.jar

5. INSTALLATION OF THE PHONE GATEWAY APPLICATION

After installing and configuring NORS platform, you need to install the N-RSA phone gateway application on the mobile devices, which you are planning to use as NORS gateways. The installation procedure is very simple, you need to open your mobile device PC suite package and select the application installer option. The application installer shows you file tree of your PC. In the file tree you should go to Intermediary Gateway package "C:/USERS/Code/" directory, and enter to "N-RSA Midlet/deployed" subdirectory. There select the N-RSA_Midlet.jar package and click "Install" button. The PC suite package will transfer the installation package to your mobile device and tell you to continue installation using mobile device UI.

After that on your mobile device you will see a prompt "Install N-RSA Phone GW?" You have answer "Yes" and then select option "Continue". Then the mobile device might inform you that Application N-RSA Phone GW is untrusted and ask whether you want to continue, answer "Yes". Then you need to select which memory to use (usually the choice is phone memory or MMC card), we recommend to use phone memory. After selecting the memory type, the actual installation will be done in automatic mode. When the installation process is completed, the mobile phone will be automatically returned back to the mode in which it was before the installation. Now you can find the N-RSA Phone Gateway application in the bottom of your main menu.

6. REFERENCES

- [NRSA] Dirk Trossen & Dana Pavel, "Report: N-RSA High-Level System Architecture", Nokia Research Center / Helsinki, 2006
- [NORSa] Elena Balandina & Dirk Trossen, "Nokia Remote Sensing Platform Middleware and Demo Application Server: Features and User Interface", Nokia Research Center / Helsinki, 2006
- [NORSb] Elena Balandina & Dirk Trossen, "Implementation guidelines for NORS platform", Nokia Research Center / Helsinki, 2006

ANNEX A

A.1 Copyright information and Disclaimer

The NORS library is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. If you have any questions, please contact Elena Balandina (Elena.Balandina@nokia.com) or Dirk Trossen (dirk.trossen@nokia.com).

No liability for the contents of this document can be accepted. You can use the concepts, examples and other content at your own risk. As this is a first edition of this document, there may be errors and inaccuracies.

All copyrights are held by their respective owners, unless specifically noted otherwise.

A.2 Translations

If you are planning to translate this document, or if you have already translated this document into any other language, please inform us.

Please make sure that you have read, understood and have accepted the terms of the License given in the files License_Server.txt and License_Middleware.txt.

A.3 Information distribution

In addition to the document of the package itself, including this document, we offer you a number of documents in order to get more information about the N-RSA system architecture, its specifications and protocols used. Note that the documents are created and distributed by Nokia Oyj. They are freely downloadable.