



Sync4j Quick Start Guide

30 October 2003

Change History

Date	Author	Description	Rev #
23 July 2003	Stefano Fornari	Initial version	1.0
30 October 2003	Stefano Fornari	Updated to 1.1	1.5

Table of Content

1. Introduction.....	4
1.1. Related Documents.....	4
1.2. Comments and Feedbacks.....	4
2. Server Quick Start.....	5
2.1. How to Get Sync4j.....	5
2.2. Installing Sync4j 1.1.x.....	5
2.3. The install.properties file.....	7
2.4. Post Installation Notes.....	8
3. Nokia 7650 Quick Start.....	9
3.1. Configuring Sync4j 1.1.x.....	9
3.2. Create a New Principal.....	9
3.3. Configuring Your Nokia 7650.....	10
4. Generic SyncML Client Quick Start.....	12
4.1. Configuring Sync4j 1.1.x.....	12
4.2. Configuring a Generic SyncML Client.....	13
4.3. Funambol SyncClient Command Line Edition for Sync4j.....	13
5. References and Resources.....	15
5.1. Resources.....	15

1. Introduction

This QuickStart Guide is intended to get you started in using Sync4j 1.1.x. Sync4j is delivered with a simple example PDI (personal data information) application that stores contacts and calendar information on the file system, so that you will be able to see it working in very few steps.

For people who own a Nokia 7650 phone, this document will guide you in configuring it to synchronize the phone's contacts and calendar with the Sync4j server.

For people who wants to see how Sync4j could work with a generic SyncML client, the document will guide you in testing Sync4j with a simplified SyncML command-line client.

1.1. Related Documents

The following documents are related to this quick start guide:

- Sync4j Administration Guide
- Sync4j Developer Guide
- Sync4j Architecture Document

1.2. Comments and Feedbacks

The Sync4j team wants to hear from you! Please submit your questions, comments, feedbacks or testimonials to sync4j-users@lists.sourceforge.net.

2. Server Quick Start

This section describes how to install and configure Sync4j 1.1.x so that it can handle PDI data represented by vCard and vCalendar objects.

2.1. How to Get Sync4j

Check the Sync4j homepage (<http://www.sync4j.org>) for information about the current version and for downloading instructions.

Sync4j is distributed as an archive file called sync4j-x.y.z.zip where x,y and z are the major, minor and build numbers.

2.2. Installing Sync4j 1.1.x

Sync4j is available in two forms: *bundled* with the JBoss application server^[2] and *unbundled*; in the latter case you must have an application server on which deploy the Sync4j server. Currently, Sync4j can be directly deployed on top of JBoss 3.0.x and Sun J2EE SDK 3.1^[3].

The unbundled Sync4j (sync4j-{major}.{minor}.{build}.jar) is the base package that can be deployed on supported application servers. The bundled Sync4j (sync4j-jboss-{major}.{minor}.{build}.jar) is a distribution that contains a bundled application server. Currently Sync4j is bundled with JBoss 3.0.8.

The requirements to install Sync4j 1.1.x bundled are:

1. JDK 1.4.x^[1]
2. Sync4j 1.1.x bundled archive (sync4j-jboss-1.1.x.zip)

The requirements to install Sync4j 1.1.x unbundled are:

1. JDK 1.4.x^[1]
2. Sun J2EESDK 1.3.1^[3]
or
JBoss 3.0.x^[2]
3. Sync4j 1.1.x (sync4j-1.1.x.zip)

The installation procedure is made up of a combination of shell and Ant^[4] scripts performing the following tasks:

- Updating configuration files accordingly to user's parameters
- Packaging for deployment on the chosen application server

- Database tables creation
- Deployment on the chosen application server

Installing Sync4j 1.1.x Bundled

To install Sync4j, follow the procedure below:

1. Install the JDK 1.4.x if not already present.
2. Unpack sync4j-jboss-1.1.x.jar in a directory of your choice. We will refer to that directory as the *installation directory*.
3. Under the installation directory you'll find the SYNC4J_HOME directory, which is called *sync4j-1.1*. Go into that directory and run:
`bin/start.sh (bin\start.cmd)`
 (Make sure that in your environment the JAVA_HOME variable is properly set).
4. Point the browser to `http://<server>:8080/sync4j` to check that Sync4j is properly installed (you should get the welcome page).

Installing Sync4j 1.1.x Unbundled on Sun J2EE SDK 3.1

To install Sync4j, follow the procedure below:

1. Install the JDK 1.4.x if not already present.
2. Install the J2EE SDK if not already present.
3. Unpack sync4j-1.1.x.jar in a directory of your choice. We will refer to that directory as SYNC4J_HOME.
4. Set up your database so that it can be accessed with a dedicated user (e.g. sync4j). This user needs to be granted permissions for connecting, creating, deleting and selecting tables.
5. Customize *install.properties* to reflect your system
6. Start the Sun J2EE RI server.
7. On unix systems, give execution permission to the executable scripts in bin and ant/bin. Use the command (from SYNC4J_HOME):
`chmod +x bin/*.sh ant/bin/*`
8. From SYNC4J_HOME, run:
`bin/install.sh sunri (bin\install.cmd sunri)`
 (Make sure that the environment variables JAVA_HOME and J2EE_HOME point respectively to your JDK home and to your J2EE SDK home)
 You will be asked if you want to create the database for Sync4j and some Sync4j modules. Respond yes ('y') to all questions.
9. Edit `{J2EE_HOME}/config/resource.properties` and add the following lines:

```
#
# Added for Sync4j
#
jdbcDataSource.5.name=jdbc/sync4j
jdbcDataSource.5.url={the jdbc url}
jdbcDriver.5.name={the jdbc driver}
```
10. Edit `{J2EE_HOME}/bin/userconfig.bat/sh` and append to the J2EE_CLASSPATH the complete classpath of your jdbc driver.
11. Stop the Sun J2EE RI server
12. Start Sync4j: from SYNC4J_HOME run:
`bin/start.sh (bin\start.cmd)`
13. Point the browser to `http://<server>:<port>/sync4j` to check that Sync4j is properly installed (you should get the welcome page).

Installing Sync4j 1.1.x Unbundled on JBoss 3.0.x

To install Sync4j, follow the procedure below:

1. Install the JDK 1.4.x if not already present.
2. Install JBoss 3.0.x if not already present.
3. Unpack sync4j-0.1.x.jar in a directory of your choice. We will refer to that directory as SYNC4J_HOME.
4. Set up your database so that it can be accessed with a dedicated user (e.g. sync4j). This user needs to be granted permissions for connection, creating, deleting and selecting tables.
5. Customize *install.properties* to reflect your system.
6. On unix systems, give execution permission to the executable scripts in bin and ant/bin. Use the command (from SYNC4J_HOME):

```
chmod +x bin/*.sh ant/bin/*
```
7. From SYNC4J_HOME, run:

```
bin/install.sh jboss (bin\install.cmd jboss)
```

 (Make sure that the environment variables JAVA_HOME and J2EE_HOME point respectively to your JDK/JRE home and to your JBoss home).
 You will be asked if you want to create the database for Sync4j and some Sync4j modules. Respond yes ('y') to all questions.
8. Start Sync4j: from SYNC4J_HOME run:

```
bin/start.sh (bin\start.cmd)
```
9. Point the browser to *http://<server>:<port>/sync4j* to check that Sync4j is properly installed (you should get the welcome page).

2.3. The install.properties file

This file is used by the installation procedure as the central repository of configuration information that are needed to properly set up a working Sync4j installation. It is a standard Java properties file containing the properties described in Table 1.

<i>Property</i>	<i>Description</i>	<i>Default Value</i>
context-path	The context path to be used to configure the web container for the Sync4j module. Sync4j will respond to URLs starting with this context path.	/sync4j
dbms	Name of the database where Sync4j tables will be created. One of: <ul style="list-style-type: none"> • ansisql99 • hypersonic • mysql • oracle • postgresql • sybase • sqlserver 	postgresql
jdbc.classpath	Classpath including the JDBC driver for the database if not included in the system classpath.	
jdbc.driver	JDBC driver class.	org.postgresql.Driver
jdbc.password	Database user password	sync4j
jdbc.url	JDBC connection URL	jdbc:postgresql://localhost/sync4j
modules-to-install	Comma separated list of Sync4j modules to install.	pdi-1.0

Property	Description	Default Value
server-name	The server URI that will be specified in the SyncML messages. The server will respond only to messages addressed to this URI.	http://localhost

Table 1 - install.properties properties

For a new Sync4j installation, you have usually to change only the database access configuration.

2.4. Post Installation Notes

The standard Sync4j installation configures Sync4j with two FileSystem SyncSources to store PDI (Personal Data Information) vCard and vCalendar items. Items are stored in the `<SYNC4J_HOME>/db` directory under, respectively, *contact* and *calendar* subdirectories: the directory name represents the name of the database and each file represents one contact or calendar card (the filename is the card id).

In a new Sync4j installation, those directories contain a few sample contacts/appointments (Figure 1).

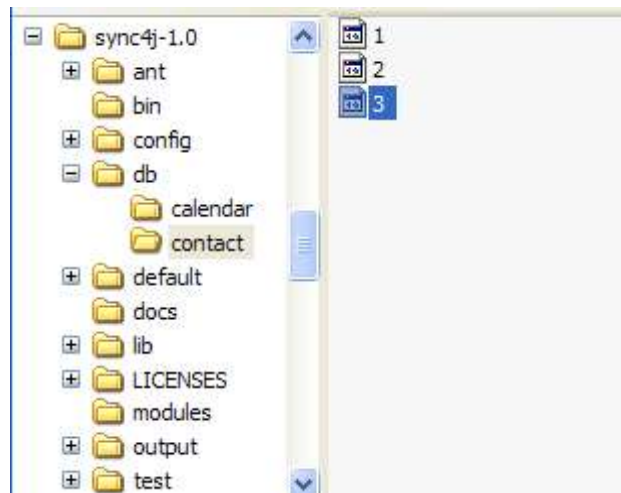


Figure 1 - FileSystem SyncSources for PDI data

3. Nokia 7650 Quick Start

This section describes how to set up Sync4j and your Nokia 7650 phone in order to synchronize contacts and calendar information.

3.1. Configuring Sync4j 1.1.x

The first thing to realize when you synchronize a mobile phone with a SyncML server is that the server must be reachable from Internet because the phones' SyncML clients use HTTP as transport protocol. This means that your server hostname or IP address must be visible on Internet.

To enable Sync4j to synchronize with the Nokia 7650, we need to apply two configuration changes:

1. Set a proper server URI
2. Create a new principal

Set the Server URI

Edit <SYNC4J_HOME>/config/Sync4j.properties and set

```
server-name=http://<hostname_or_ip>
```

3.2. Create a New Principal

Sync4j associates a user and a device into a *principal*. Therefore if you add a new user or a new device you need to create a new principal, otherwise Sync4j will not allow that user to synchronize.

When the Nokia 7650's remote sync software makes a SyncML request, regardless the user/password you specify in the configuration, it first tries without sending credentials. This is interpreted by Sync4j as a guest access, or, in other words as a SyncML request made by the 'guest' user.

The device id is specified by the phone in its "Source" field inside the SyncML message and is represented by the phone's IMEI (International Mobile Equipment Identity) code (e.g.: IMEI:351111103384988). This code can be found on the phone's box label, under the phone's battery or pressing *#06# on the phone keyboard.

When you have the serial number, you can add a new device and principal with sql statements like:

```
insert into sync4j_device (deviceid, description, type)
values ('IMEI:351211111184999', 'some description', 'N7650');
```

```
insert into sync4j_principal (username, device, principalid)
values ('guest', 'IMEI:351211111184999', 5);
```

3.3. Configuring Your Nokia 7650

In order to synchronize the Nokia 7650 you need to download and install the latest Remote Sync software.

To update to the latest software release, follow the steps below. There are two ways to install Remote sync on your phone: the first one is to download it wirelessly to your phone, the other way is to download it to your local PC and then transfer the installation file to your phone via Infrared or Bluetooth.

Installing Remote Sync from Internet

1. Create a Bookmark pointed to Remote Sync software
 - from the Main menu select Services.
 - Select **Options > Add bookmark**
 - Enter the following url:
http://nds1.nokia.com/phones/files/software/Nokia_7650_RemoteSync.SIS
 - Select your default Access Point.
 - Save bookmark.
2. Download Remote Sync software
Activate the bookmark by pressing the joystick middle button. Wait until you get the **'Install Sync?'** message. Select **Yes**.

Installing Remote Sync from PC or Laptop

1. Download the Remote SyncSoftware
Go to <http://www.nokia.com/cda1/0,1080,2019,00.html> to download the Remote Sync software.
When prompted, save the file to the hard disk on your computer.

NOTE: Subsequent steps describe how to transfer the file to your phone using an infrared connection. You may need to transfer the file to a laptop computer if your desktop computer does not have an infrared port. Alternatively, you can transfer the Remote Sync software to your phone using Bluetooth. Please see the manufacturer's instructions for information how to do this.

2. Activate the Phone Infrared Port
On your Nokia 7650 phone select the *Applications* menu. Open the *Communications* folder. Select *Infrared*, and *Open* it. The icon in the top right-hand corner will indicate that the infrared is activated.
3. Transfer the Application to the Phone
Position the phone infrared port facing your computer's infrared port. A tray icon on your desktop will indicate the phone has been detected. Open the tray icon for the infrared connection by double-clicking on it. The wireless link window will appear. Select the Remote Sync software you downloaded in Step 1 and click Send. You will see a progress dialog indicating the software is being transmitted to your phone. The phone will beep when the transfer is complete.
You can do the same with the *PC Suite for Nokia 7650* software: run it and select Tools/Install device Software. Choose the SIS file you have downloaded and click on Open. The software will be transferred to your phone.
4. Install the Remote Sync Client
If you just sent the file to the IR port without PC Suite, you'll find a new message titled *Infrared* in you Inbox. Open this message and select Yes when prompted to save. The phone will then install the application.

If you transferred the application with the PC Suite, the phone automatically prompts to install the application. Select Yes.

Configuring the SyncML Client

After installing the Remote Sync Client you will have a new Icon in the Applications menu. Follow the steps below:

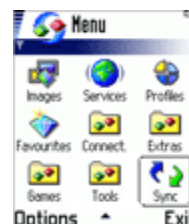
Move the selection to the *Sync* application and press the joystick.

Click on Options and create a new sync profile. Insert the following information:

Name: Sync4j
Transport: Internet
Access Point: choose your internet access point
Host Address: <yourhost>/sync4j/sync
Port: <yourport> (eg: 8080)
User: guest
Password: guest
Calendar: Yes
Remote Calendar: calendar
Address Book: Yes
Remote Address Book: contact
HTTP Authentication: No
HTTP User: <empty>
HTTP Password: <empty>

Press Back when finished.

Select the newly created profile and click the Select button, then select Synchronise. Depending on your phone settings, you will be prompted to make a connection. Press OK.



The first time the phone is synchronized, a slow synchronization is performed, so that all your contacts and appointments will be sent to the server, who replies with the contacts and appointments stored in the db/contact and db/calendar directories. If the synchronization successfully completed, you should find that in your phone book there are three new contact cards:

John Doe Marketing Director john_doe@somewhere.com +1 650 50504040	Mike Smith Business Developer mike.smith@yahoo.com +1 469 43220403	Vincent Brown CEO vbrown@mail.com +1 329 5864389
---	---	---

4. Generic SyncML Client Quick Start

In this section we are going to synchronize the Sync4j calendar and contact sources with a generic SyncML client. We will see a simple working example based on the use of the Funambol SyncClient Command Line Edition for Sync4j.

4.1. Configuring Sync4j 1.1.x

In this case, the only thing we have to do is to add a new device and principal based on the device and user id sent by the client. See your client documentation to know the exact device and user ids sent in a SyncML message. When you have that information, you can add a new device and principal with sql statements like:

```
insert into sync4j_user (username, email, first_name, last_name)
  values ('<username>', '<email>', '<first_name>', '<last_name>');
insert into sync4j_device (deviceid, description, type)
  values (<deviceid>, 'some description', 'some tipe');
insert into sync4j_principal (username, device, principal)
  values ('<username>', <deviceid>, 3);
```

For example, in the case of the Funambol SyncClient Command Line Edition for Sync4j, you would run the following commands:

```
insert into sync4j_user (username, email, firt_name, last_name)
  values ('jdoe', 'john\_doe@somewhere.com', 'John', 'Doe');
insert into sync4j_device (deviceid, description, type)
  values ('sc2', 'Funambol SyncClient Command Line Edition', 'J2SE');
insert into sync4j_principal (username, device, principalid)
  values ('jdoe', 'sc2', 3);
```

Note that some clients will not send credential unless requested by the server. In this case, Sync4j takes the user 'guest' as user id and the statements to execute become:

```
insert into sync4j_device (deviceid, description, type)
  values ('<deviceid>', 'some description', 'some tipe');
insert into sync4j_principal (username, device, principal)
  values ('guest', '<deviceid>', 3);
```

Regarding the server URI, not all clients allow you to configure which target URI they use in their SyncML messages. If they allow you to set the target URI (as the Funambol SyncClient does), set it on the client. Otherwise you will need to know which target URI the client sends and set it in `<SYNC4J_HOME>/config/Sync4j.properties`.

4.2. Configuring a Generic SyncML Client

The information you need to know at the client side to synchronize with Sync4j contact and calendar items are the following:

```
Url: http://<hostname>:<port>/sync4j/sync
User: <username>
Password: <password>
Remote Address Book: ./contact
Remote Calendar: ./calendar
```

If the client allows you to set the target URI, set it to the value of the *server.uri* property in `<SYNC4J_HOME>/config/Sync4j.properties`.

4.3. Funambol SyncClient Command Line Edition for Sync4j

The Funambol SyncClient Command Line Edition for Sync4j is a simple SyncML client for development purposes. It is highly configurable and stores/reads data items to/from the file system. In this section we will see how to download and install it and how to synchronize its data sources with Sync4j.

Downloading and Installing Funambol SyncClient CLE for Sync4j

1. Download Funambol SyncClient CLE for Sync4j from here:
<http://www.funambol.com/download/syncml/fscle-1.0.zip>
2. Install Funambol SyncClient CLE for Sync4j: unzip the package in a directory of your choice

After unpacking the zip file you will have the directory structure of Figure 2.

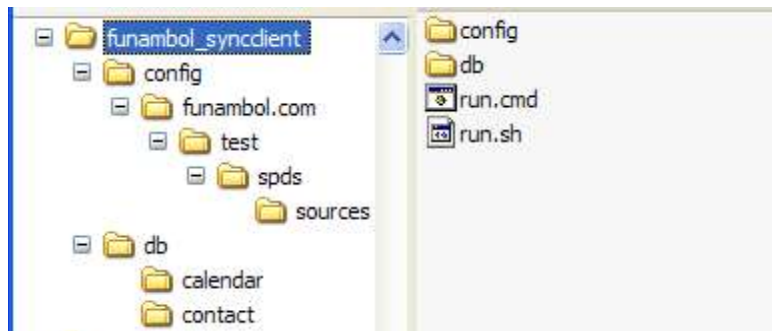


Figure 2 - Funambol SyncClient Command Line Edition for Sync4j installation directory structure

Configuring the Client

You can configure the client editing the properties file `config/spds/syncml.properties`. Here is an example of configuration file:

```
#The initial URL for the SyncML request
syncml-url=http://localhost:8080/sync4j/sync

# The target URI of the server being contacted
targetLocalUri=http://localhost:8080

# Username and password for authentication to the sync server
username=guest
password=guest
```

```
# The device identifier  
device-id=sc2
```

Running the Client

Make sure the variable `JAVA_HOME` is properly set in your shell environment. If so, just launch the client with the provided script:

```
run.cmd (./run.sh)
```

You can follow what is going on on the console and after the command exits, you can check the results in the *db/contact* and *db/calendar* directories. These directory are initially empty, but after the first sync, they should contain the same vCards and vCalendars stored on the server. Then you can try adding new items or modifying and deleting exiting ones either on the client side and on the server side; run the client again to see the changes applied.

5. References and Resources

5.1. Resources

- [1] <http://java.sun.com/j2se>
- [2] <http://www.jboss.org>
- [3] <http://java.sun.com/j2ee>
- [4] <http://ant.apache.org/>