Java remote debugging on mobile device

L0cke

10.11.2011

Table of Contents

1. Problem description 1
2. Prerequisities 1
3. Setting up the mobile device 2
    3.1. Debug vs runtime link file 2
    3.2. Preparing mobile device for advanced network features 2
4. Setting up eclipse IDE 4
5. Start to debug 5

For my wife Agnieszka

1. Problem description

The problem is to debug the running Java application on the mobile device with Windows Mobile (6.1, or 6.5). This is useful when you want to see, what is happening on the device during the application run. So we have to set up our device to enable debug features and our eclipse IDE explicit to do so.

2. Prerequisities

First of all we need couple of software for this one. Those are :

1. Device with Windows Mobile (at least 6.1)

   You need a device with Windows Mobile, appropriate cradle for it, and USB cable to connect the cradle with your desktop PC.

2. Microsoft Active Sync (4.5 is the actuall)

   You can get it from here : **http://www.microsoft.com/download/en/details.aspx?id=15**

   Of course you have to install this one on your desktop windows.

3. Eclipse IDE

   Download one of your favourites distributions from eclipse page. It's easy to find via google if you don't know it yet.

4. Java application.

   Here we need an application which we will deploy on the device in jar file form[1].

5. Java SDK for desktop

   You know where to get this, from Oracle of course. For applications for mobile devices you have to download and set JDK 1.4. Most of the java virtual machines for mobile devices wan't run applications compiled with higher JDK. So remember this one.

[1] It is very important to remember that on mobile devices virtual machines there are no features like in JDK 1.5 or 1.6 or 1.7. So you have to compile your java application with at most Java JDK 1.4.

6. Java Virtual Machine for the device mobile

   Here are couple of choices. For example CrEme, Sun JDK ME, IBM JVM and so on. In the test the IBM JVM was used, but the parameters for the JVM are the same for Sun JVM. So the parameters for it are for this machine specific. You have to read your mobile java virtual machine documentation to set your java virtual machine for debuging mode.

3. Setting up the mobile device

As I mentioned earlier I am using here IBM JVM for mobile devices. You have to read your JVM documentation for the appropriate infos about which parameters are needed for the machine to debug your application on the mobile device[2].

3.1. Debug vs runtime link file

To run (it doesn't matter in wich mode) Java application on the mobile device you have to create appropriate "batch" link with extension .lnk for it. The trick is, that for debugging you have to use different parameters in the link.

Let's call the files **Jtest01.lnk** - for runtime, and **Jtest01_debug.lnk** - for debug mode. Differences are as follows :

Table 1. Differences for the link for different modes

Runtime mode	Debug mode
Starts with #17	Starts with #225
Uses standard JVM parameters like -cp (for classpath)	uses standard JVM parameters like -cp (for classpath)
	uses additional parameters for debugging (this is JVM - dependend)

And below is the table with content of the files.

Table 2. Sample content of the link-files for debugging java application on mobile device

Jtest01.lnk	Jtest01_debug.lnk
17#"\vm\bin\j9.exe" -jcl:foun11 -cp "\vm\lib;\jtest01\JTest01.jar" JTest01	225#"\vm\bin\j9.exe" "-jcl:foun11" "-Xdebug" "-Xrunjdwp:transport=dt_socket,server=y,address=8000" -cp "\vm\lib;\jtest01\JTest01.jar" JTest01

So, for JVM you have to add in the debug link following additional parameters :

- -Xdebug

  this only means for the JVM that you start the application in debug mode

- -Xrunjdwp:transport=dt_socket,server=y,address=8000

  Here you say, that you will use TCP/IP with socket and port address 8000 for debuging mode

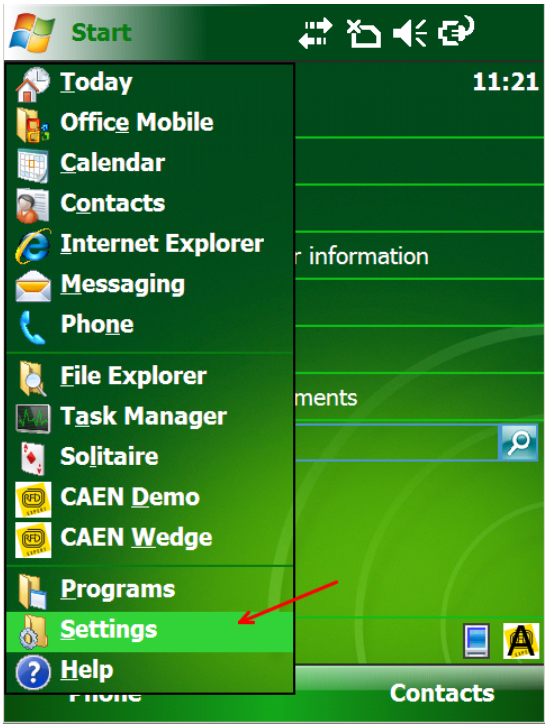
3.2. Preparing mobile device for advanced network features

So now you have to change some configuration in Windows Mobile.

1. Select **Start → Settings** on the mobile device

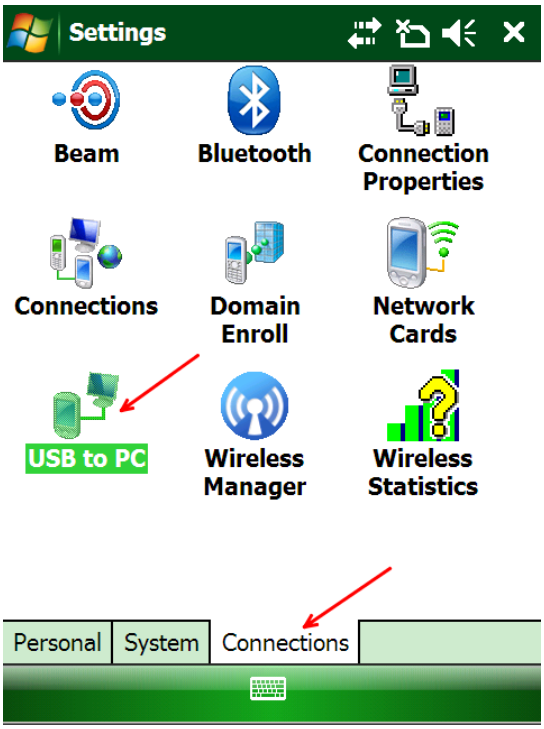
[2] IBM JVM is compatible with Sun's JVM for mobile devices. Sun's parameters for JVM for debuging are the same. See following link : http://download.oracle.com/javame/dev-tools/jme-sdk-3.0-mac/UserGuide-html/z400007746960.html, where under the section Debugging and Tracing Options is the description for the Sun's JVM.

Figure 1. Selecting settings on the mobile device



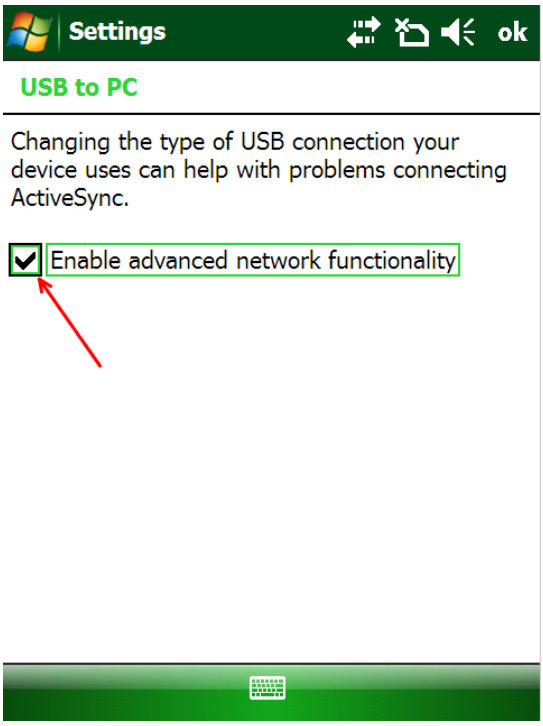
2. Select **Connections tab** and choose **USB to PC** item from the tab

Figure 2. Selecting item USB to PC on the Connections tab



3. Check the "**Enable advanced network functionality**"

Figure 3. Enabling advanced network functionality



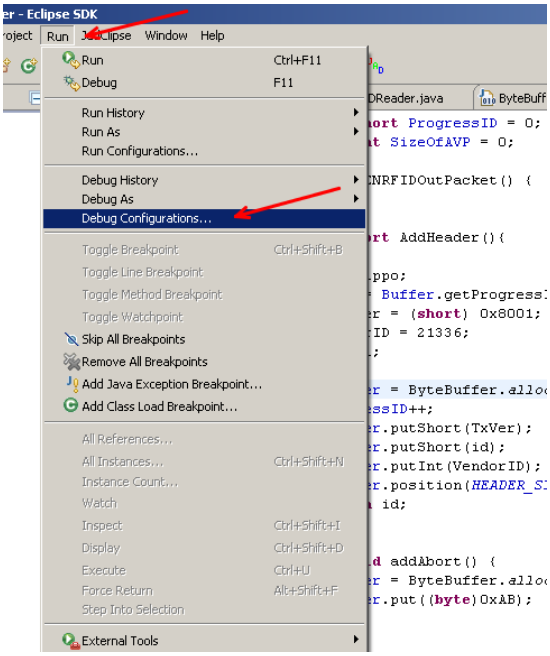
4. Click **OK** on the top right side of this window on mobile device.

4. Setting up eclipse IDE

After you have already installed IDE, JDK and written your Java application, you have to export it as jar-file. After that you have to configure your IDE for remote debugging this application on the mobile device. Let's get started :

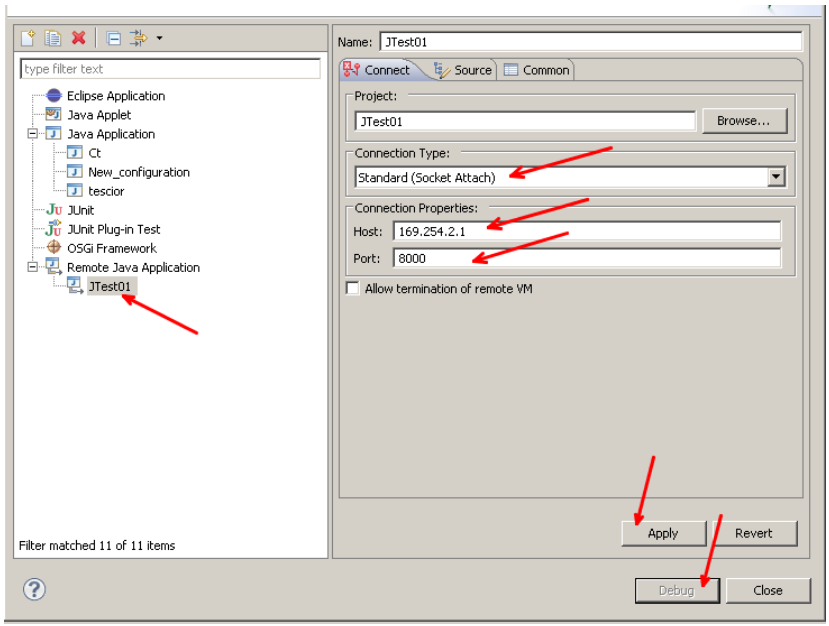
1. Select **Run → Debug Configurations...** from the main menu

Figure 4. Selecting Debug Configurations from the main menu



2. Now on the dialog box configure new Remote Debug Configuration :

Figure 5. Setting up remote debug configuration for JTest01 project



Here you have to set up :

- Connection Type : **Standard (Socket Attach)**
- Host : **169.254.2.1**

  This is standard. If you have already enabled advanced network features on the mobile device this is automatically set with this values (for the mobile device your desktop PC has IP Address 168.254.2.2)[3]
- Port : **8000**

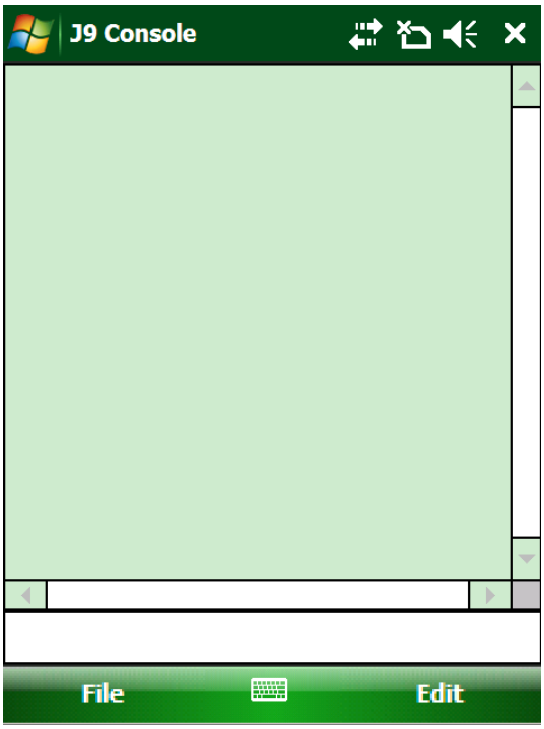
5. Start to debug

Now all you have to do is to copy your JTest01.jar to the corresponding directory (in my case it was \jtest01). Additionally you have to copy there your link-files. Now click the file **Jtest01_debug.lnk** on the device.

Application will start, but will not run.

[3]Be adviced, that if you want to use MyMobiler application, you have to shut down the advanced network features. For working with ActiveSync Explorer of the device, the advanced network features should be also shut down. Don't care about that, that when advanced network features are on, ActiveSync want see your device in some cases. Your device is connected with the desktop. Just follow on. If you want to explore your device, you have to shut down advance network features, then ActiveSync will connect to it and you can explore the file directories of the device.

Figure 6. Application on mobile device in waiting mode (debug mode before IDE is connected)



Application waits for connection with IDE. So in eclipse IDE choose **Run → Debug Configurations...** Here choose your remote debug configuration and click **Debug** button. Debug will start right now and the application on mobile device will run till the next breakpoint in your application in eclipse IDE.

Happy debugging!