

# Getting Started Guide for Using Sprint WTK and Netbeans for Creating Java Apps for Samsung Instinct



# INSTINCT™

ONLY ON THE NOW NETWORK™



INSTINCT™

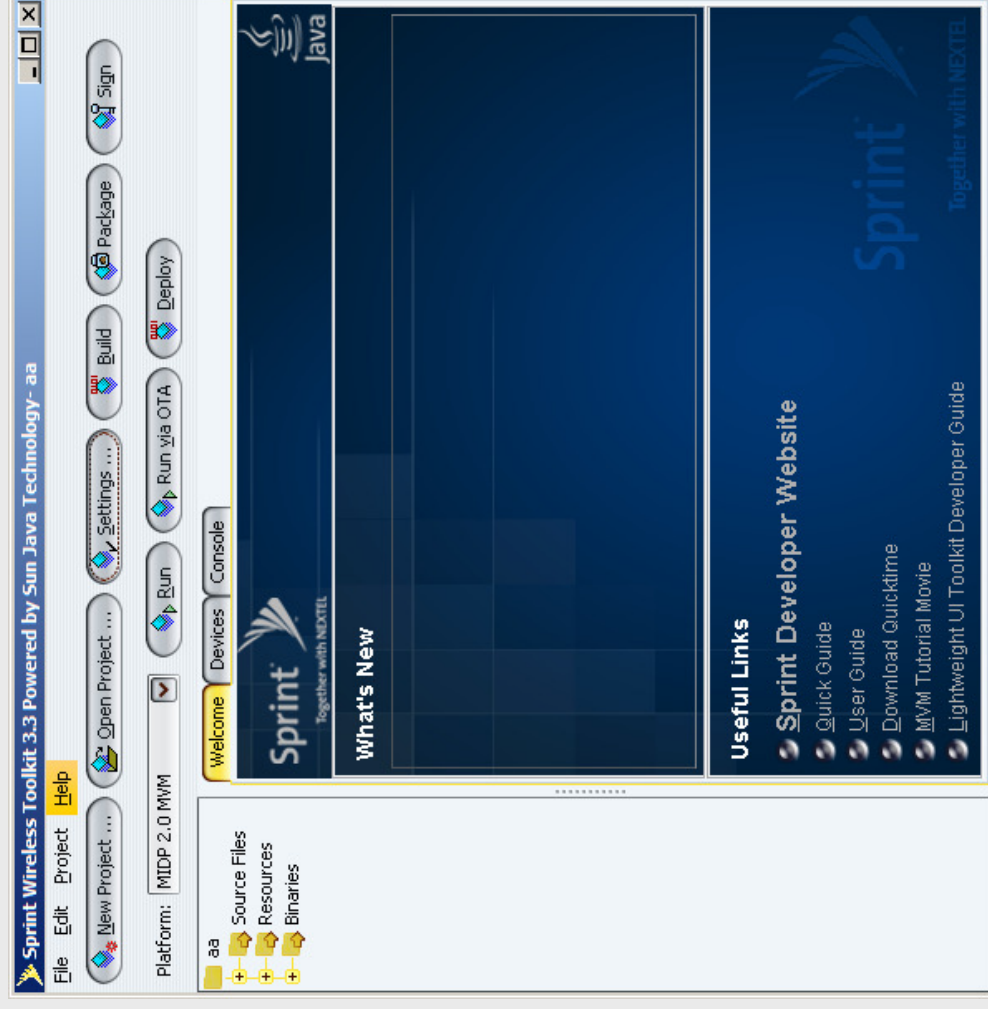
# Overview

---

- This guide is a pictorial based tutorial on how to create a Samsung Instinct project
  - 1: Installing Sprint WTK 3.3
  - 2: Adding Sprint WTK 3.3 into Netbeans
  - 3: Creating Samsung Instinct Sample Project in Sprint WTK 3.3
  - 4: Importing Samsung Instinct SWIL JAR to your Netbeans library (1x only)
    - Linking in Samsung Instinct Java Docs from the Sprint WTK 3.3
  - 5: Importing Sprint WTK project into Netbeans
    - How to assign Samsung emulator
  - 6: Creating configurations for Samsung Instinct Project
    - Package up to run on Samsung Instinct Handset (w/o SWIL JAR file included in your project)
  - 7: Testing
- Appendix
  - A: Creating Instinct projects in Sprint WTK from sample JAR/JAD
  - B: Creating Instinct project in Netbeans

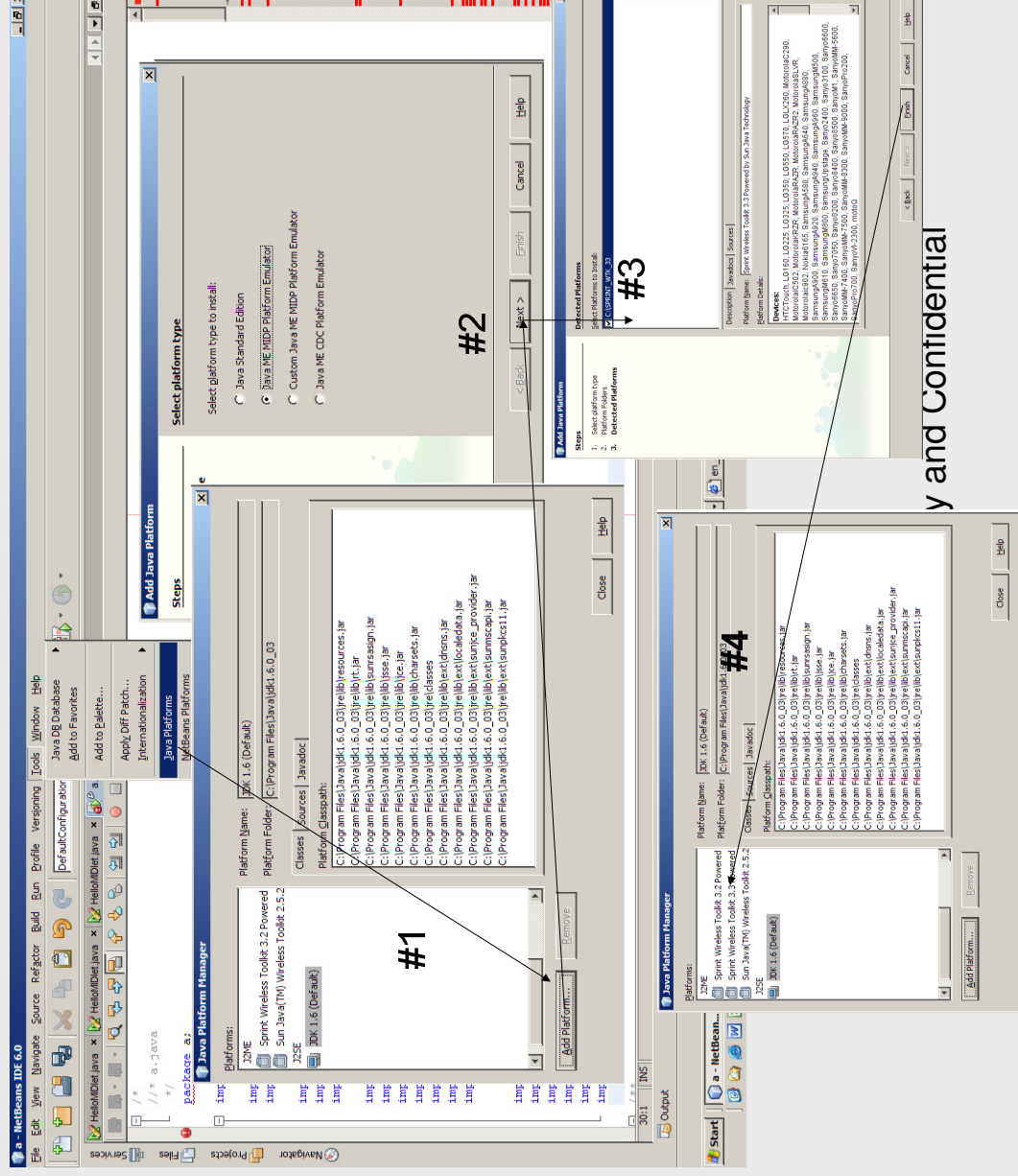
# 1: Installing Sprint WTK 3.3

- Pre-Conditions
  - Download and install JDK 1.5+ from Sun website
  - Download and install QuickTime to support multi-media codecs and RTSP
- Download and Install Sprint WTK 3.3
  - WTK available at <http://developer.sprint.com>
  - Will automatically detect JDK and QuickTime on installation
- Start up WTK by selecting the K-Toolbar from you quick start menu



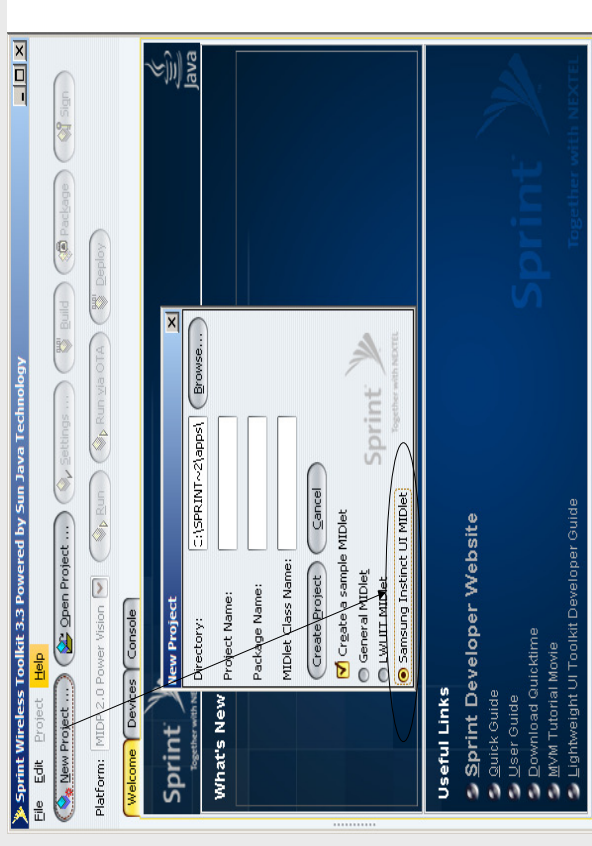
# 2: Adding Sprint WTK Into Netbeans

- Start up Netbeans → Click on Tools → Java Platforms → Add Platforms → Add Java Platforms → Java ME Platform Emulator → Sprint WTK 3.3



# 3: Create New Instinct Project in Sprint WTK 3.3

- Recommend creating project in Sprint WTK first (vs in Netbeans) as it will create a sample midlet, that declares the proper declarations to use the SWIL (Samsung Widget Instinct Library) – then import the project into Netbeans for your development
  - Open Sprint WTK 3.3
  - Select ‘Open Project’
    - Select Samsung Instinct Project
    - Make sure generates sample midlet

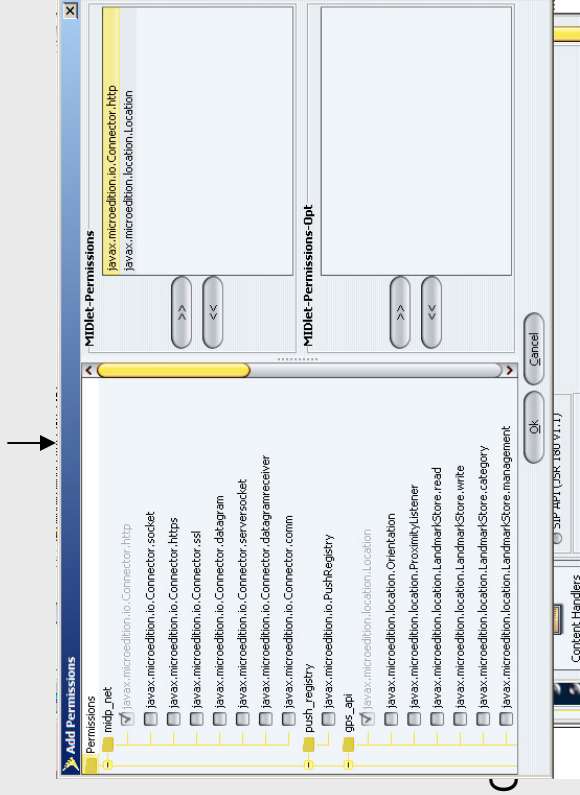


# 3: Create New Instinct Project in Sprint WTK 3.3 (setting JSRs/Permissions)

- Once you have created your project, define which JSRs / Sprint Extensions you wish to use in your handset
  - List of supported JSRs are:
    - JSR 75 PIM
    - JSR 82 Bluetooth (note only JSR by default, not selected – please make sure to check this if you wish to use bluetooth in your app)
      - Also refer to the bluetooth sample app provided on the ADP website
    - JSR 120 Messaging
    - JSR 135 Multi-Media
    - JSR 177 SATSA
    - JSR 179 LBS
    - JSR 184 3D
    - Sprint Extensions 2.0 are supported



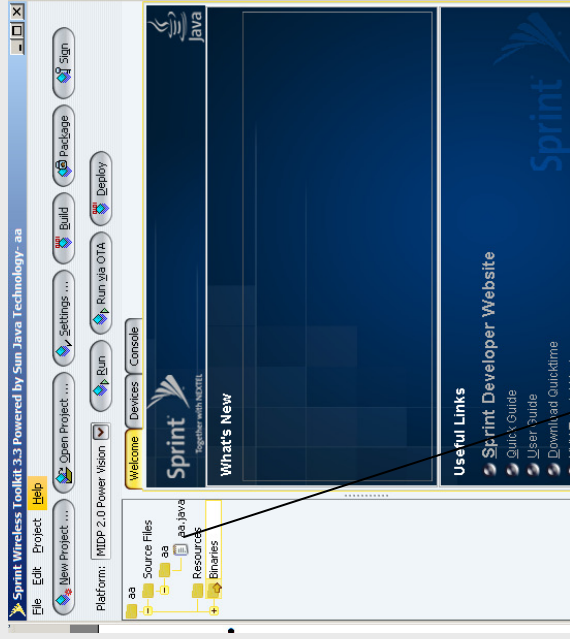
- Once you have selected which JSRs you wish to utilize in your application, the Sprint WTK 3.3 includes a wizard for you to add the required permissions associated with those capabilities
  - Please refer to the Sprint MIDP 2.0 CDMA code signing document to understand which JSRs are restricted and how to enable the development mode on the Instinct
  - If you use restricted APIs and don't declare them properly, you will see a 909 download error on your handset when trying to install the compiled app



Sprint Proprietary and Confidential

# 3: Create New Instinct Project in Sprint WTK 3.3 (looking at generated code)

- When you have completed creating your project you will see the project and associated folders
- Click in the source file folder and you can see the sample app generated code



```
aa.java - Notepad
File Edit Format View Help
/*
 * aa.java
 */
package aa;

import java.io.IOException;
import javax.microedition.lcdui.Command;
import javax.microedition.lcdui.Display;

import javax.microedition.lcdui.Image;
import javax.microedition.midlet.MIDlet;
import javax.microedition.midlet.MIDletStateChangeException;

import com.sprintpcs.lcdui.widget.BasicFrame;
import com.sprintpcs.lcdui.widget.ListFrame;
import com.sprintpcs.lcdui.widget.Tabbar;
import com.sprintpcs.lcdui.widget.Tabitem;
import com.sprintpcs.lcdui.widget.WidgetEventListener;
import com.sprintpcs.lcdui.widget.WidgetImage;
import com.sprintpcs.lcdui.widget.Widget;

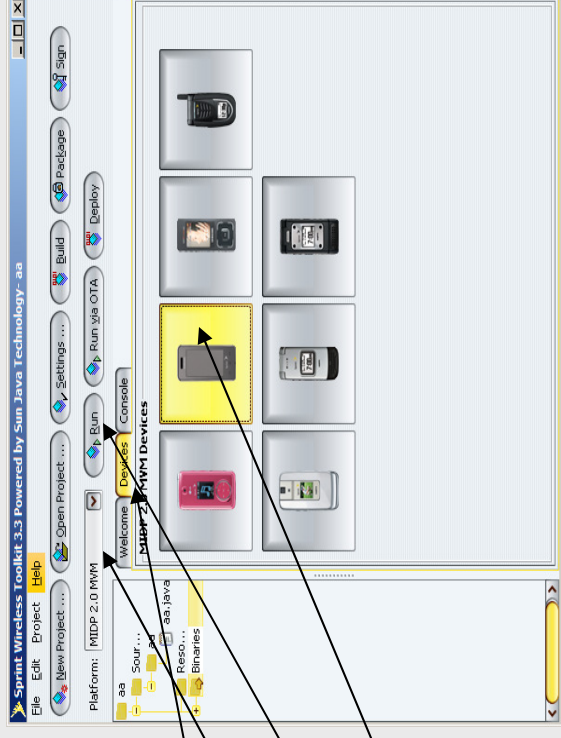
import com.sprintpcs.lcdui.widget.Frame;
import javax.microedition.lcdui.Graphics;
import javax.microedition.lcdui.CommandListener;
import com.sprintpcs.lcdui.widget.PushButton;
import com.sprintpcs.lcdui.widget.Titlebar;
import com.sprintpcs.lcdui.widget.Textitem;

/**
 * The application must extend this class to allow the applicati
```

Sprint Proprietary and Confidential

# 3: Create New Instinct Project in Sprint WTK 3.3 (Starting up in Emulator)

- To run your sample project in the emulator
  - 1: Select the Devices Tab
  - 2: Select the MIDP 2.0 MVM drop down tab
  - 3: Select the device labeled Samsung M800 (Instinct)
  - 4: Select Run
  - 5: The Instinct Emulator window will pop up with your sample application running

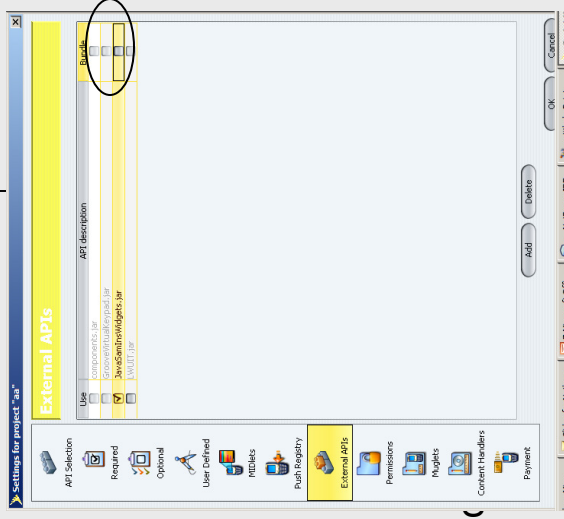
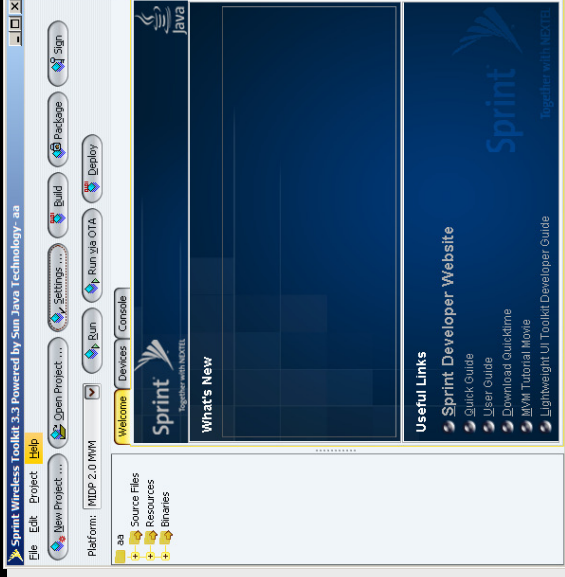


Sprint Proprietary and Confidential



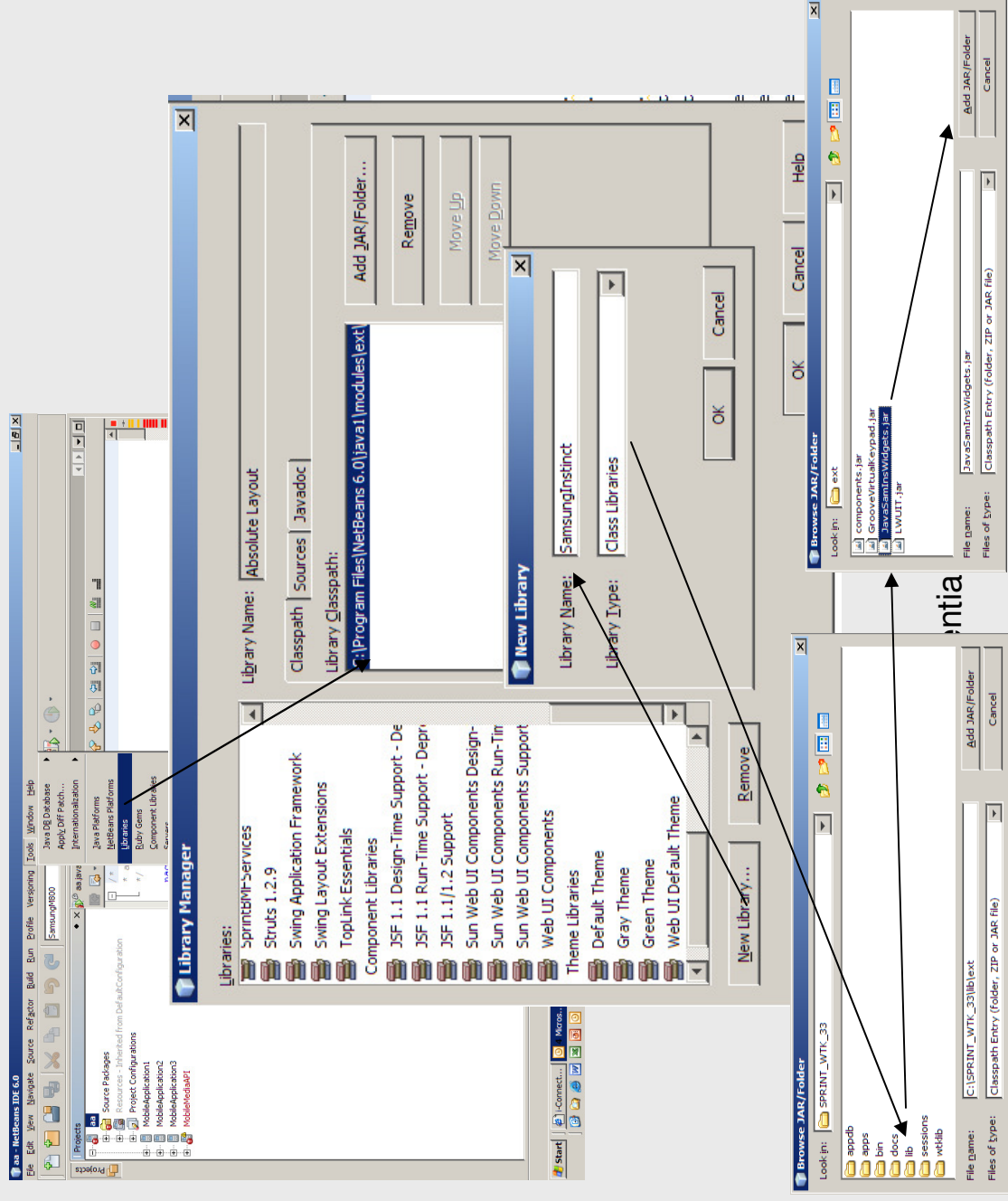
# 3: Create New Instinct Project in Sprint WTK 3.3 (Packaging Up)

- If you wish to package up app (not required to import your project into Netbeans) Click on 'Settings' in main menu and then select 'External APIs'
- Please note the default is to include the Samsung Instinct Widget Library into your JAR file. **Make sure you uncheck this because:**
  - If you deploy to a real handset and your app has this included in your app, your app will error out as the Samsung instinct handset already has the SWIL capabilities burned into the handset
  - If you wish to take your project and import into another dev/package environment (Netbeans, Eclipse, your own ant scripts, etc) we provide the SWIL JAR file as a separate binary downloadable that you utilize as your project **needs to** **Sprint Proprietary and Config**



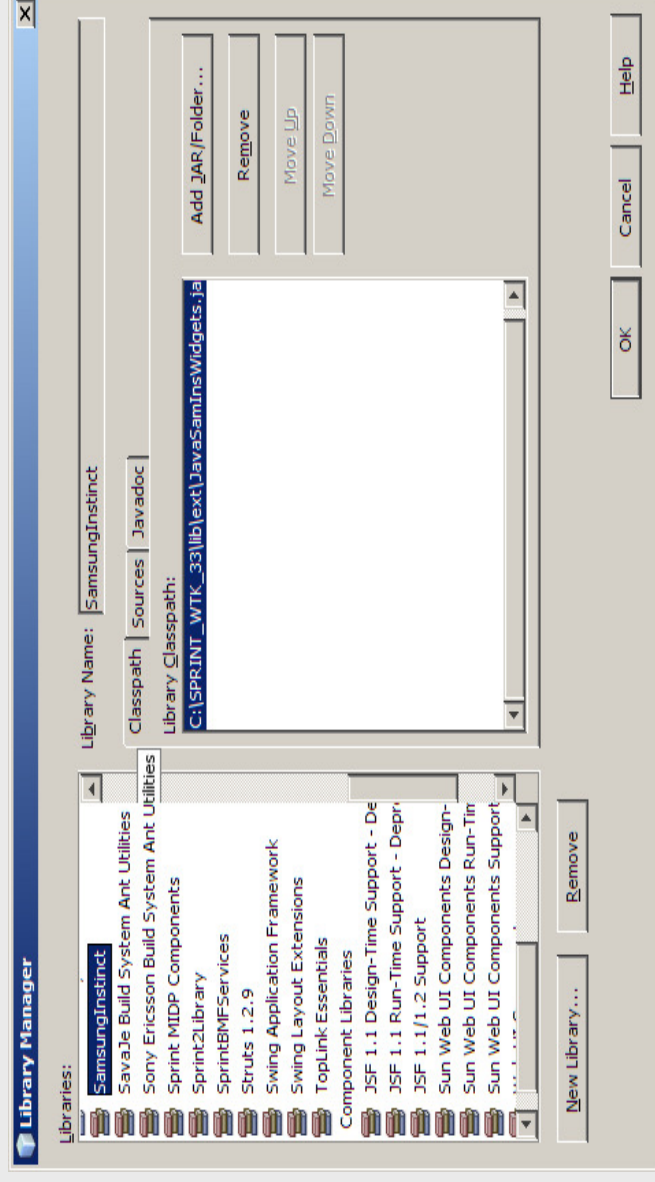
# 4: Opening Instinct Project Created with Sprint WTK 3.3 into Netbeans (add in Samsung Widget Instinct Library in Netbeans)

- You only need to do this step 1 time for your first Instinct project only
  - You need to create a new library which contains the Samsung Instinct Widget Library Jar file



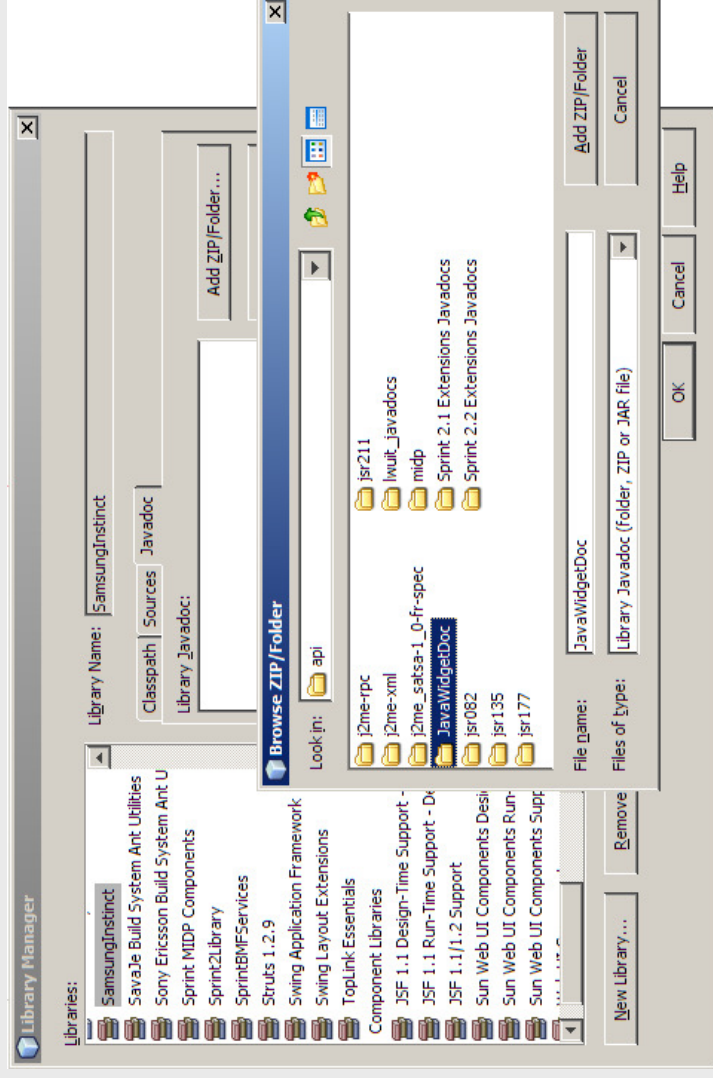
## 4: Opening Instinct Project Created with Sprint WTK 3.3 into Netbeans (Linking to Samsung Instinct Java Docs)

- Open the Library Manager from the Tools -> Libraries menu.
- Select the Samsung Instinct library
- Click the Javadoc tab and click on the Add JAR/Folder button



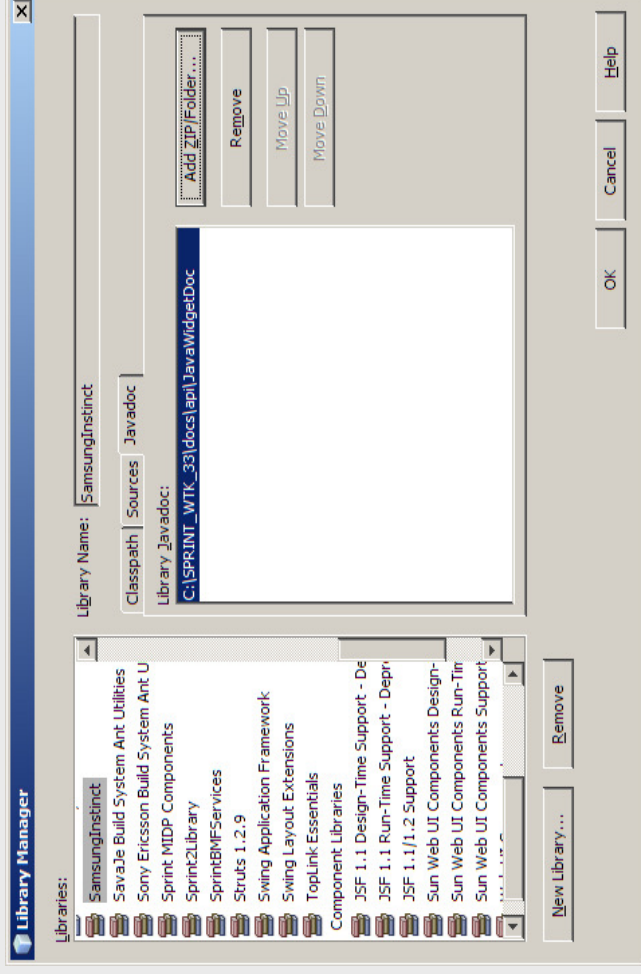
## 4: Opening Instinct Project Created with Sprint WTK 3.3 into Netbeans (Linking to Samsung Instinct Java Docs)

- Navigate to the docs folder in the WTK install directory.
- For the Instinct Javadocs choose the docs\api\JavaWidgetdoc folder



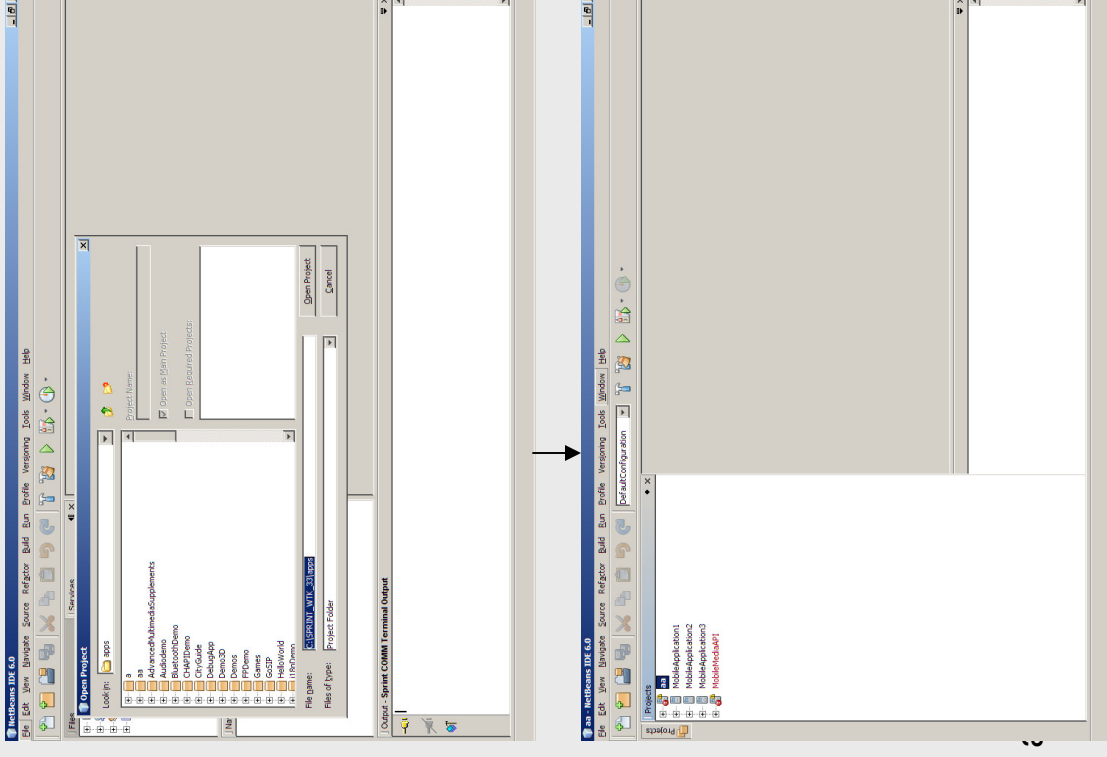
# 4: Opening Instinct Project Created with Sprint WTK 3.3 into Netbeans (Linking to ~~Samsung Instinct Java Docs~~)

- Click OK to complete the Javadoc library addition.



# 5: Opening Instinct Project Created with Sprint WTK 3.3 into Netbeans

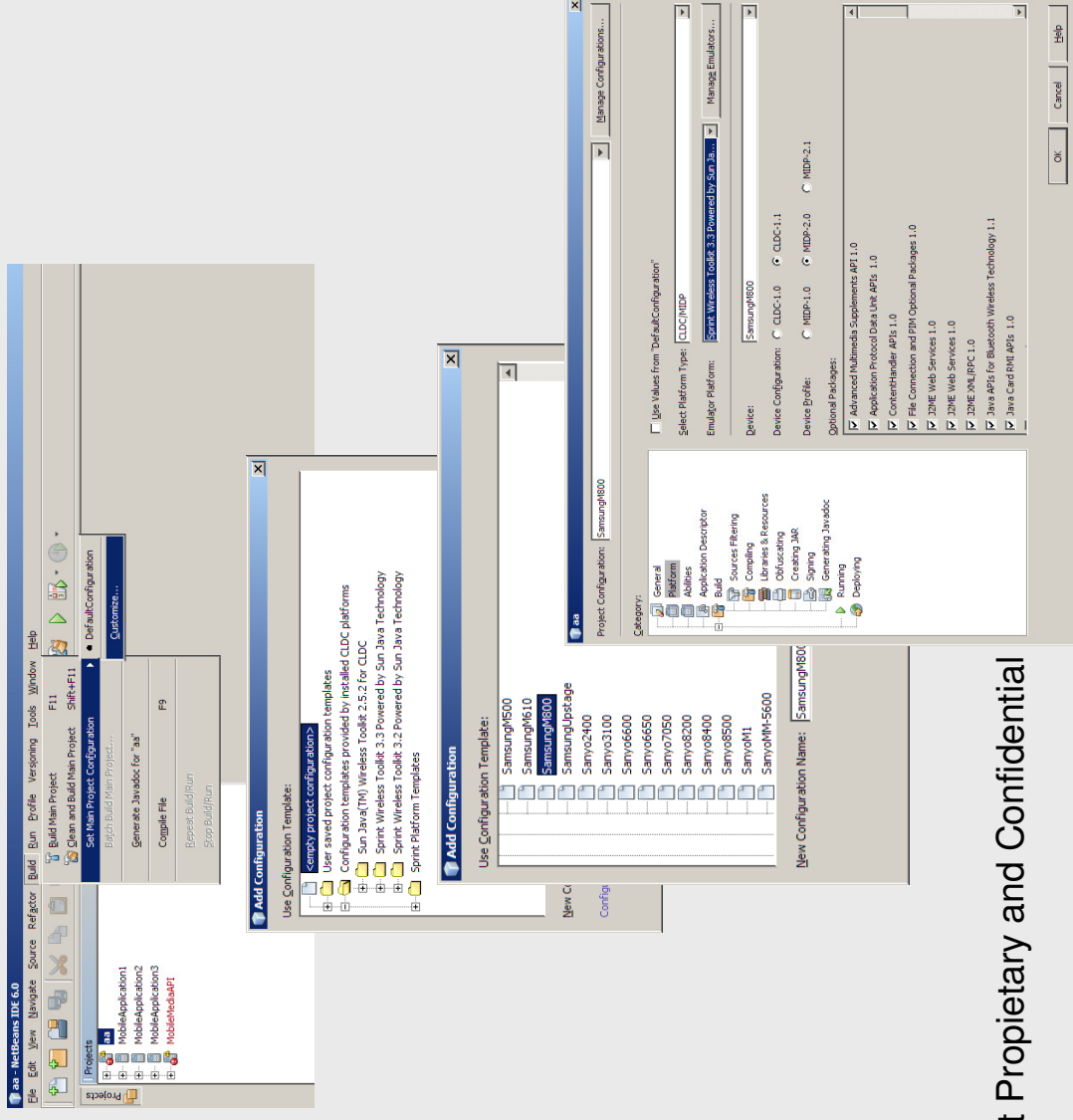
- Open Netbeans
- Go to File→Open Project→Your Project Name
- Your Project will now be Opened in Netbeans



Sprint Proprietary ©

# 5: Opening Instinct Project Created with Sprint WTK 3.3 into Netbeans (Assigning Instinct Emulator from Sprint WTK 3.3)

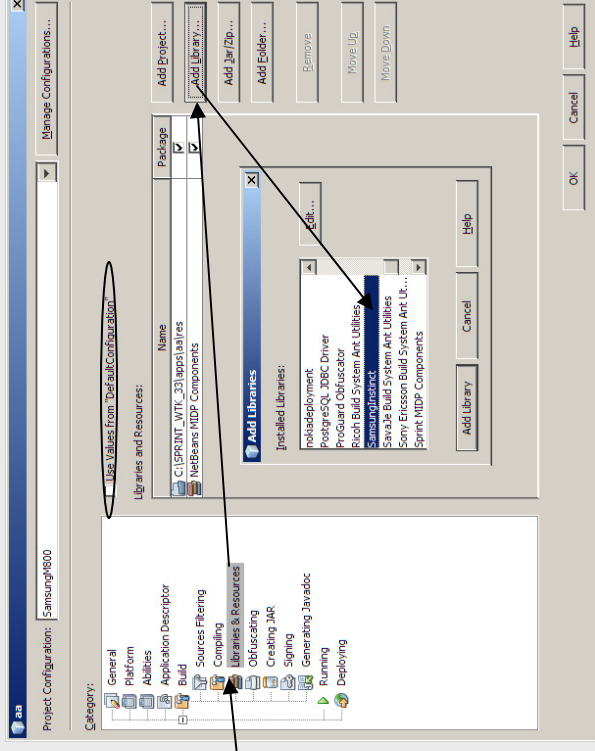
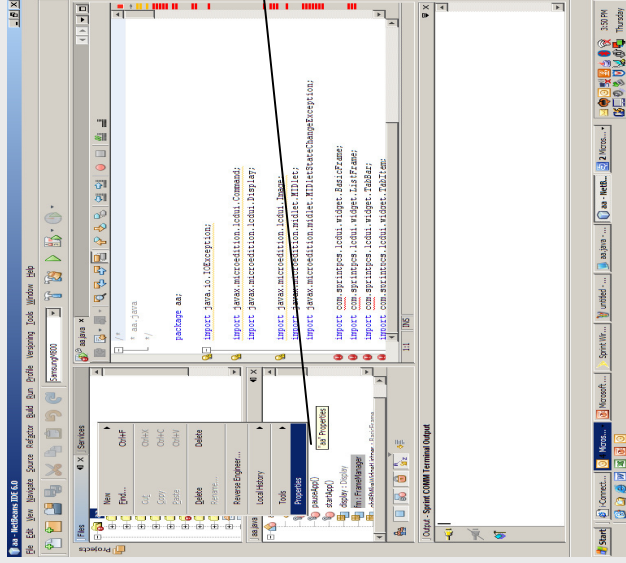
- To assign Sprint WTK 3.3 Instinct Emulator to your project within netbeans
  - Build→Set Main Project Configuration
  - Select Sprint WTK 3.3
  - Select Samsung M800(Instinct)



Sprint Proprietary and Confidential

# 5: Opening Instinct Project Created with Sprint WTK 3.3 into Netbeans (Adding Samsung Instinct Widget library to your project)

- Right click on your project, select 'properties'
- Select 'Libraries and Resources'
  - Select Add Library (make sure the 'Use Values from 'Default Config' is UNCHECKED)

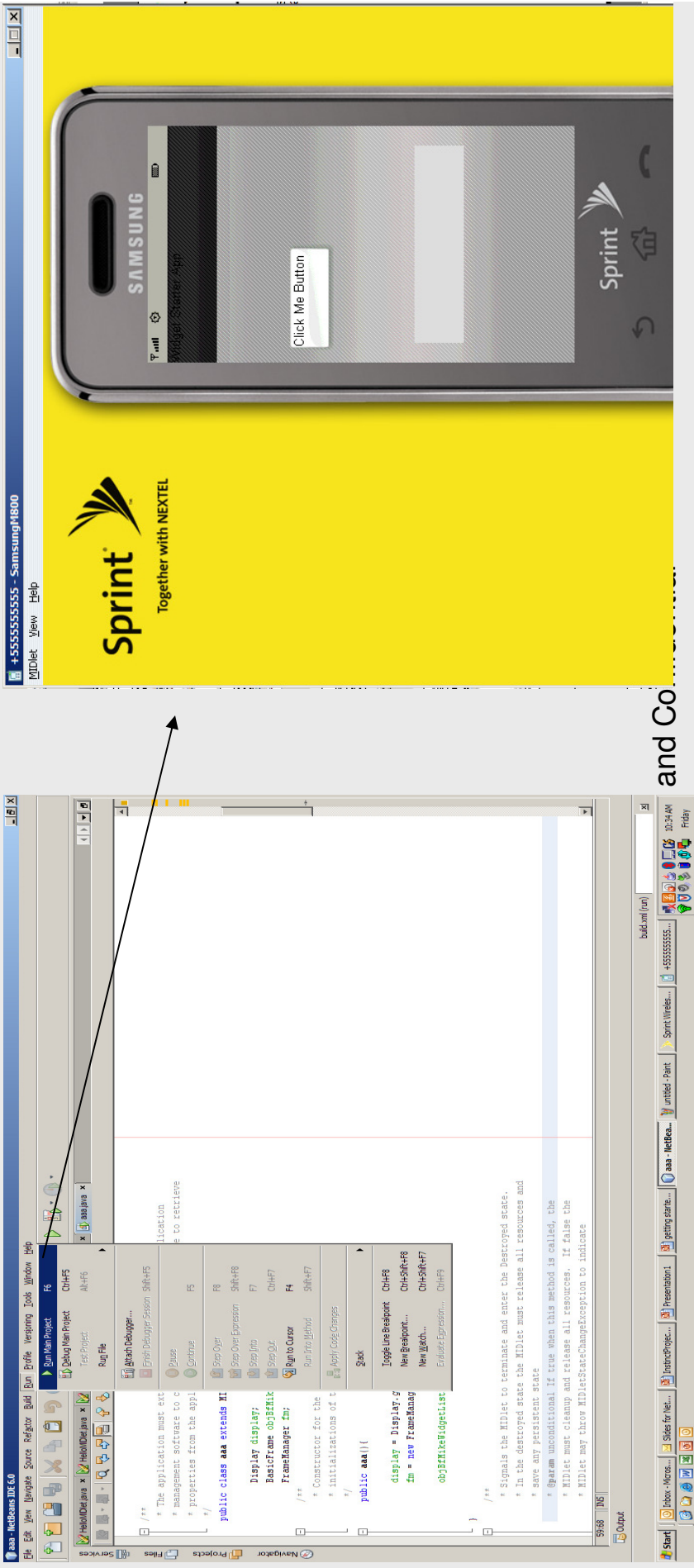


Sprint Proprietary and Confidential



# 5: Opening Instinct Project Created with Sprint WTK 3.3 into Netbeans (Running your application)

- Select Run in your menu options
  - If you have added in the Instinct JAR file and linked to the Samsung Instinct handset (M800) then your project should run in the emulator image



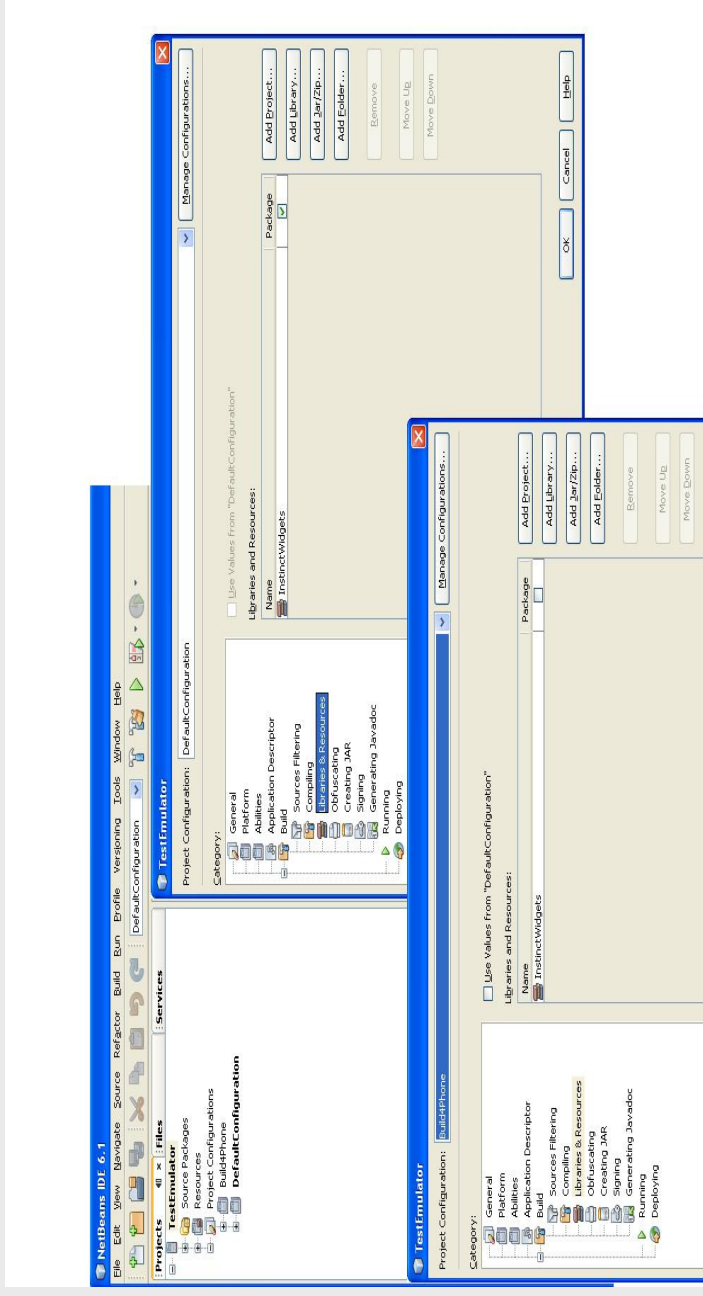
and Co

# 6: Deploying your project with two different configurations (Emulator / Handset)

- To avoid select/remove the Instinct Widget Library bundle each time you build the app to run in the emulator or to deploy to the real phone, create 2 configurations in your NetBeans project.

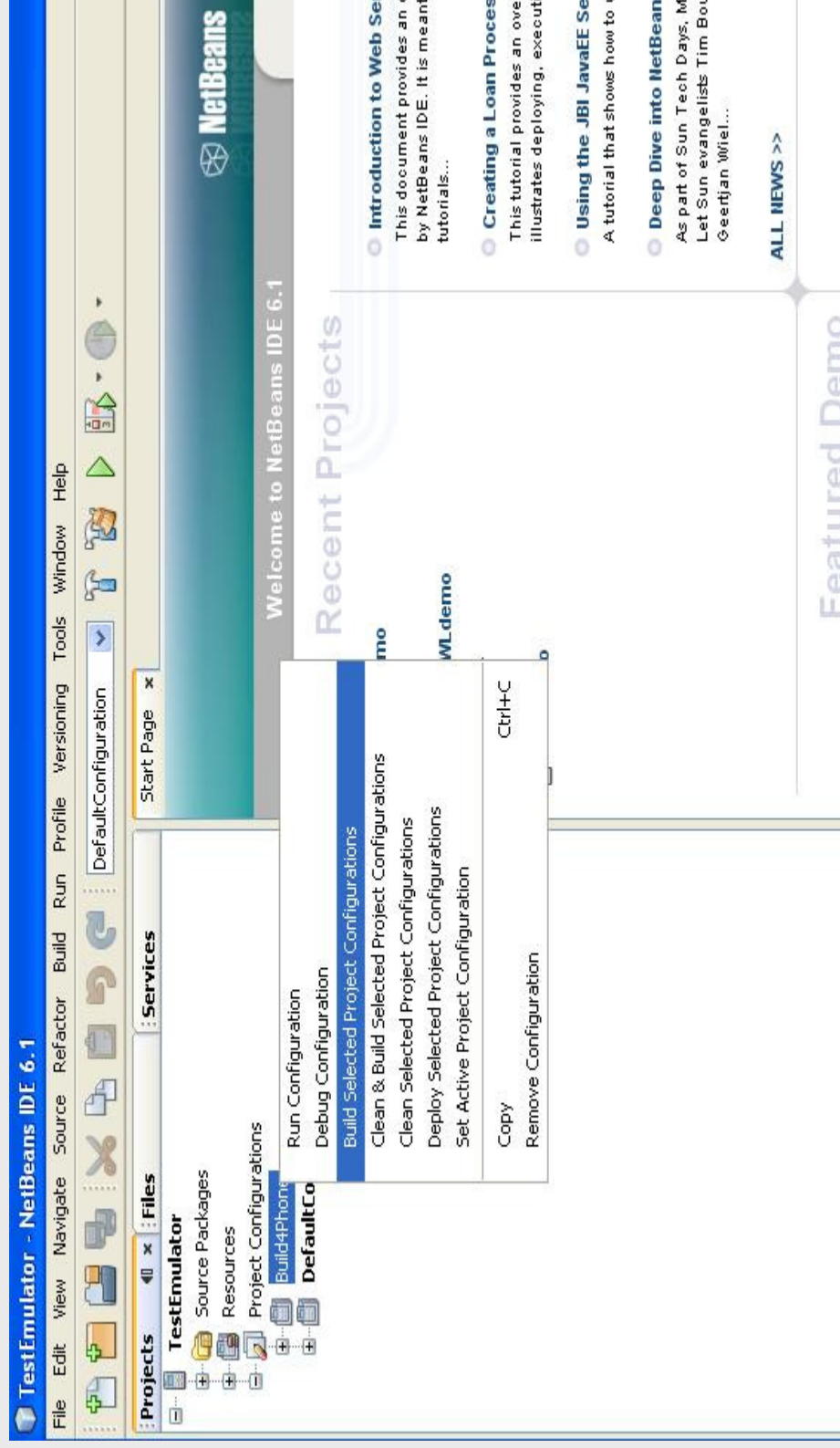
- 1: Instinct emulator
- 2: Instinct phone.

Note that the Build4Phone configuration has the InstinctWidgets library unchecked.



## 6: Deploying your project with two different configurations (Emulator / Handset)

- Use the Build4Phone configuration to build your app to deploy on the Instinct phone.



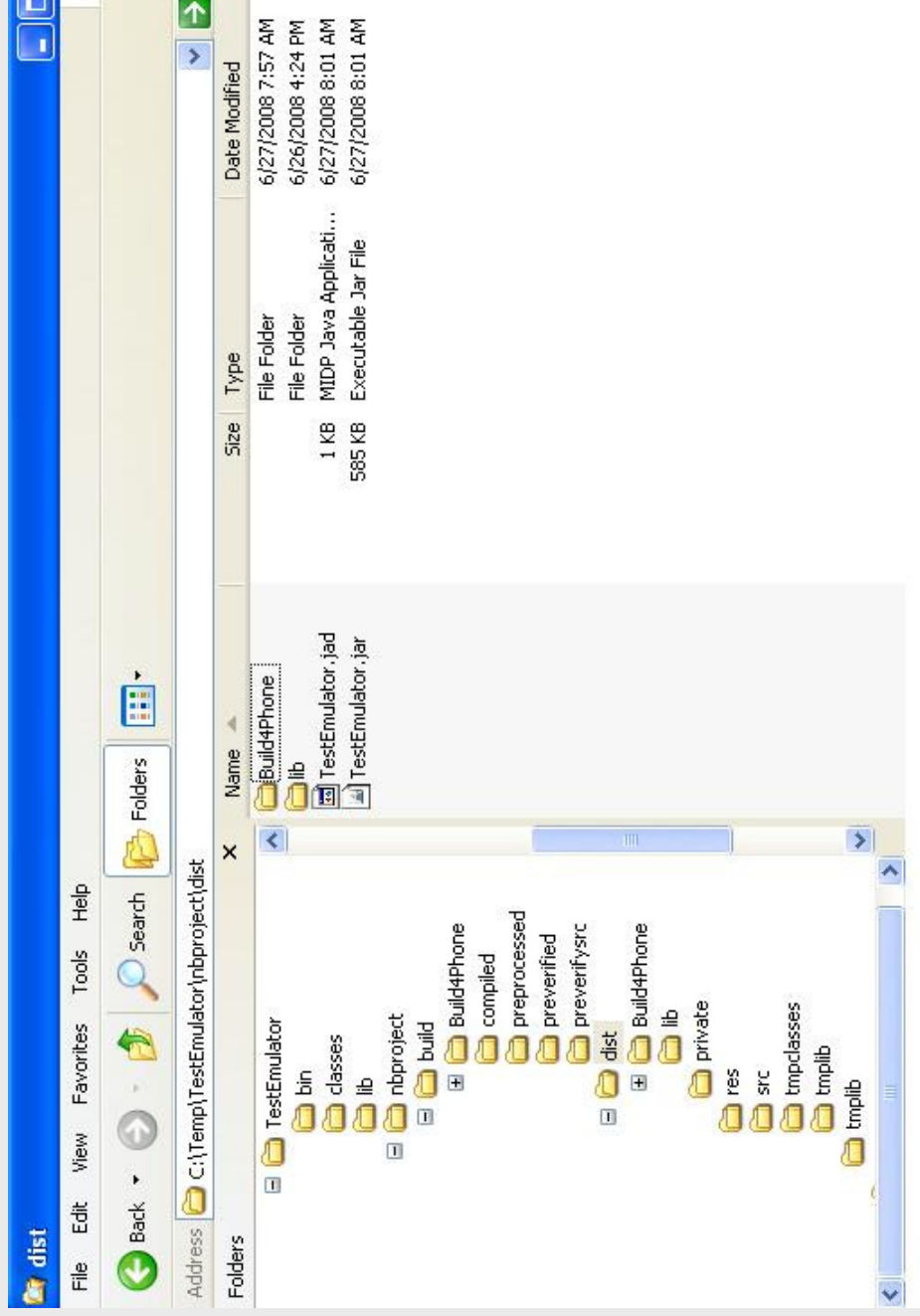
## 6: Deploying your project with two different configurations (Emulator / Handset)

- NetBeans will create a separate folder for the Build4Phone configuration. Here is where to find your JAD/JAR file to deploy to the Instinct phone.



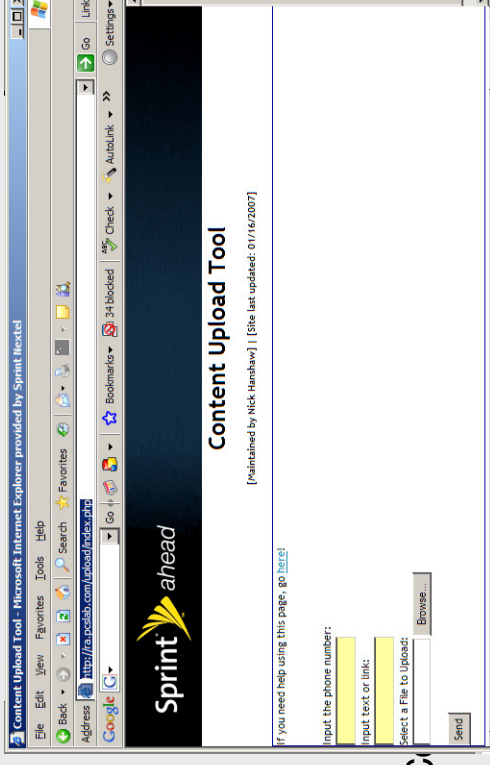
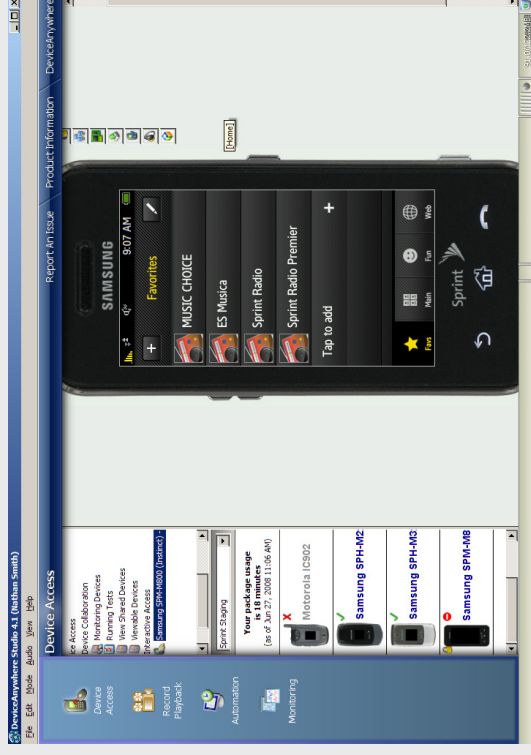
## 6: Deploying your project with two different configurations (Emulator / Handset)

- Notice the JAR file in the build for emulator configuration is significantly larger, as it bundles the Instinct Widgets library



# 7: Testing

- There are three options available to help you test your Instinct application:
  - Instinct emulator available through WTK 3.3
  - Access real Instinct handset via the Sprint Virtual Developer Lab  
<http://www.deviceanywhere.com/sprint/welcome.htm>
  - Deploy your JAD/JAR to a Samsung Instinct Handset in your physical possession  
<http://ra.pcslab.com/upload/in dex.php>



Sprint Proprietary and Confidential

# Appendix

Sprint Proprietary and Confidential

# A: Importing Samsung Instinct Sample Apps into Sprint WTK

- Sprint provides sample apps at <http://developer.sprint.com/instinct>
- You can import the JAR/JAD files into the Sprint WTK 3.3 to create a project that can be imported into Netbeans





# B: Creating Instinct Project in Netbeans (cont)

- Create Mobile Project using Sprint MIDP template
- See Section 4 to add in the instinct jar to your library (if you have not done this)
- See Section 5 to add the SWIL library to your project
- Reference Java Doc or Samsung Instinct Handset Developer guide for proper code declarations to utilize SWIL library
- See Section 5 to link to the Samsung M800 Emulator
- See Section 6 to build two different configuration (emulator and handset w/o SWIL library)

