



How-To Guide

SigPlus Java Simple Demo

Copyright © Topaz Systems Inc. All rights reserved.

For Topaz Systems, Inc. trademarks and patents, visit www.topazsystems.com/legal.

Table of Contents

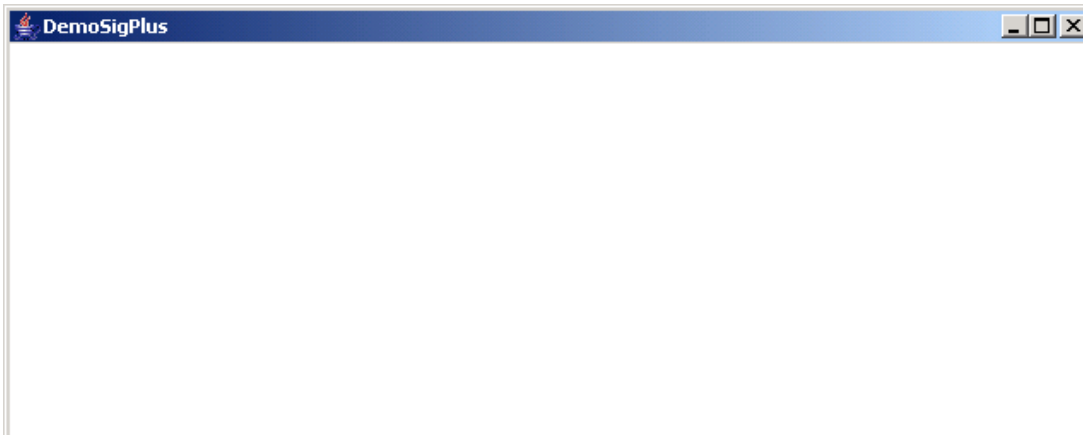
Overview.....	3
The Code	4
Dependencies and Files Necessary to Run This Demo	6

Overview

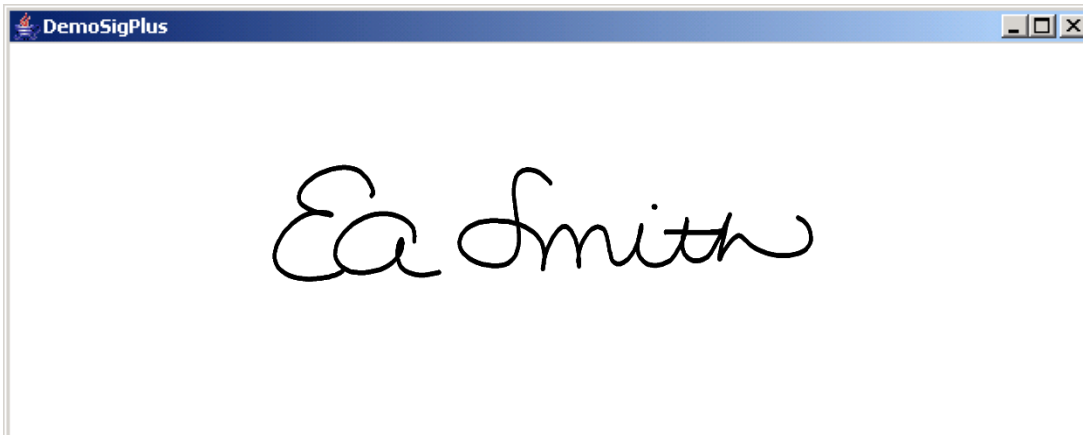
Welcome to the SigPlus Simple Demo Guide. This will walk you through our demo “SigPlusSimpleDemo.java”, and the same demo for HSB.

Download at: www.topazsystems.com/sigplusprojava.html

This application will show Java developers how to integrate SigPlus technology into their own applications. Begin by opening “SigPlusSimpleDemo.java” in a Java runtime environment. The window below will appear. There are no interface options, and all the user can do is sign the tablet.



Once signed, you will see the user signature on the screen.



Because there are no user options when creating this kind of application, it is very important to set all tablet properties correctly in your code.

The Code

We will now look over the Java code used to control this application. The code below creates a form for the “SigPlusSimpleDemo”.

```
import java.awt.*;
import java.awt.event.*;
import java.beans.*;
import com.topaz.sigplus.*;

import javax.comm.*;
import java.io.*;

public class SigPlusSimpleDemo extends Frame
{
    SigPlus    sigObj = null;

    public static void main( String Args[] )
    {
        SigPlusSimpleDemo demo = new SigPlusSimpleDemo();
    }

    public SigPlusSimpleDemo()
    {
        String drivername = "com.sun.comm.Win32Driver";
        try
        {
```

We instantiate the Comm Driver below. This is important for windows Java apps only.

```
CommDriver driver = (CommDriver) Class.forName(drivername).newInstance();
    driver.initialize();
}
catch (Throwable th)
{
    /* Discard it */
}
```

We instantiate SigPlus below using the beans.intantiate function.

```
try
{
    ClassLoader cl = (com.topaz.sigplus.SigPlus.class).getClassLoader();
    sigObj = (SigPlus)Beans.instantiate( cl, "com.topaz.sigplus.SigPlus" );
}
```

Add SigPlus to the layout here.

```
setLayout( new GridLayout( 1, 1 ) );  
add( sigObj );  
pack();  
setTitle( "DemoSigPlus" );
```

When the window closes, we turn off the tablet state.

```
addWindowListener( new WindowAdapter()  
{  
public void windowClosing( WindowEvent we )  
{  
sigObj.setTabletState( 0 ); System.exit( 0 );  
}  
  
public void windowClosed( WindowEvent we )  
{  
System.exit( 0 );  
}  
});
```

These three event handlers for SigPlus are not currently implemented.

```
sigObj.addSigPlusListener( new SigPlusListener()  
{  
public void handleTabletTimerEvent( SigPlusEvent0 evt )  
{  
}  
public void handleNewTabletData( SigPlusEvent0 evt )  
{  
}  
public void handleKeyPadData( SigPlusEvent0 evt )  
{  
}  
}  
});  
  
setSize( 640, 256 );  
show();
```

These three lines of code are the bare minimum to capture a signature. We set up the tablet model and port so Sigplus knows which to use, then we turn the tablet on.

```
        sigObj.setTabletModel( "SignatureGem4X5" );
        sigObj.setTabletComPort( "COM1" );

        sigObj.setTabletState( 1 );
    }
    catch ( Exception e )
    {
        return;
    }
}
```

Dependencies and Files Necessary to Run This Demo

Jar files must be included in the CLASSPATH

1. SigPlus2_xx.jar (at the creation of this document the current version was 2_45)
2. Comm.jar (This file will vary depending on OS, you will need the correct java com API for your OS.)

Additional Files

1. (Win) – Win32com.dll must reside in the System32 folder
2. (HSB in Win) – SigUsb.dll must reside in the System32 folder