

Overview

This code snippet demonstrates how to use MMAPi and eSWT together to take a picture using the phone's camera. This code displays the camera viewfinder on an eSWT Control. It also provides an eSWT Command which will take the picture and save it.

Source code

```
import java.io.IOException;
import java.io.OutputStream;
import javax.microedition.io.Connector;
import javax.microedition.io.file.FileConnection;
import javax.microedition.media.Manager;
import javax.microedition.media.MediaException;
import javax.microedition.media.Player;
import javax.microedition.media.control.VideoControl;
import javax.microedition.midlet.MIDlet;
import org.eclipse.ercp.swt.mobile.Command;
import org.eclipse.swt.events.SelectionEvent;
import org.eclipse.swt.events.SelectionListener;
import org.eclipse.swt.graphics.Rectangle;
import org.eclipse.swt.widgets.Control;
import org.eclipse.swt.widgets.Display;
import org.eclipse.swt.widgets.Shell;

public class MMAPiExample extends MIDlet implements Runnable, SelectionListener {

    Thread UIThread;
    Display display;
    Shell shell;
    private Command takePictureCommand;
    private VideoControl videoControl;

    public void startApp() {
        // Create the eSWT UI thread.
        if(UIThread == null) {
            UIThread = new Thread(this, "UIThread");
            UIThread.start();
        }
    }

    public void pauseApp() {
        // Here we could reduce the resources but we should keep the Display instance
        // and the eSWT UI Thread.
    }

    public void destroyApp(boolean unconditional) {
        // Dispose the remaining resources.
        display.dispose();
    }

    // The eSWT UI Thread
    public void run() {
        // Create the Display.
    }
}
```

Using_eSWT_and_MMAPI_to_take_a_picture_in_Java_ME

```

        = new Display();

        = new Shell(display);
        open();    shell.

                CommandExecutor(shell, Command.EXIT, 0);
        setText(Command);
        addSelectionListener(this);

                takePictureCommand(shell, Command.GENERAL, 10);
        takePictureCommand();
        addSelectionListener(this);
        addDisposeListener();

        = MediaPlayer player

try {
    = Manager.createMediaPlayer("capture://video");
    realize();    player.
        = (VideoControl) player.getControl("VideoControl");
    Control vControl = (Control) videoControl.initDisplayMode(VideoControl.USE_GUI_PRIMITIVE, Control
        setParent(shell.vControl.
    Rectangle r = vControl.getBounds();
        setDisplaySize(vControl.r.height);
        setVisible(vControl.
    start();    player.

} catch (IOException e) {

} catch (MediaException e) {

}

// Execute the eSWT event loop.
while(!shell.isDisposed()) {
    if(!display.readAndDispatch()) {
        sleep();    display.
    }
}
if (player != null){
    deallocate();    player.
    close();    player.
    = null;    player
}

// Clean up and destroy the MIDlet
    (true) destroyApp
        (notifyDestroyed

}

public void widgetDefaultSelected(SelectionEvent e) {
}

public void widgetSelected(SelectionEvent e) {
    if(e.widget == takePictureCommand){
        ();    takePicture
    }else{
// Exit command selected, exit the event loop.
        dispose();    shell.
    }
}
}

```

Using_eSWT_and_MMAPI_together_to_take_a_picture_in_Java_ME

```
private void takePicture() {  
  
    byte[] snapshot;  
    try {  
        snapshot = webViewControl.getSnapshot(null);  
        if (snapshot == null || snapshot.length == 0) return;  
        String filename = "file:///c:/data/images/I"+System.currentTimeMillis()+".png";  
        FileConnection fileConnection = (FileConnection) Connector.open(filename, Connector.RETRIEVE);  
        fileConnection.open();  
        OutputStream stream = fileConnection.openOutputStream();  
        stream.write(snapshot);  
        stream.flush();  
        stream.close();  
        fileConnection.close();  
  
    } catch (MediaException e) {  
  
    } catch (IOException e) {  
  
    }  
  
}
```

Postconditions

A PNG image containing the picture is saved under C:\Data\Images.

See also

- [eSWT](#)