

Distributed Programming in Android

Elisa Gonzalez Boix
egonzale@vub.ac.be



32

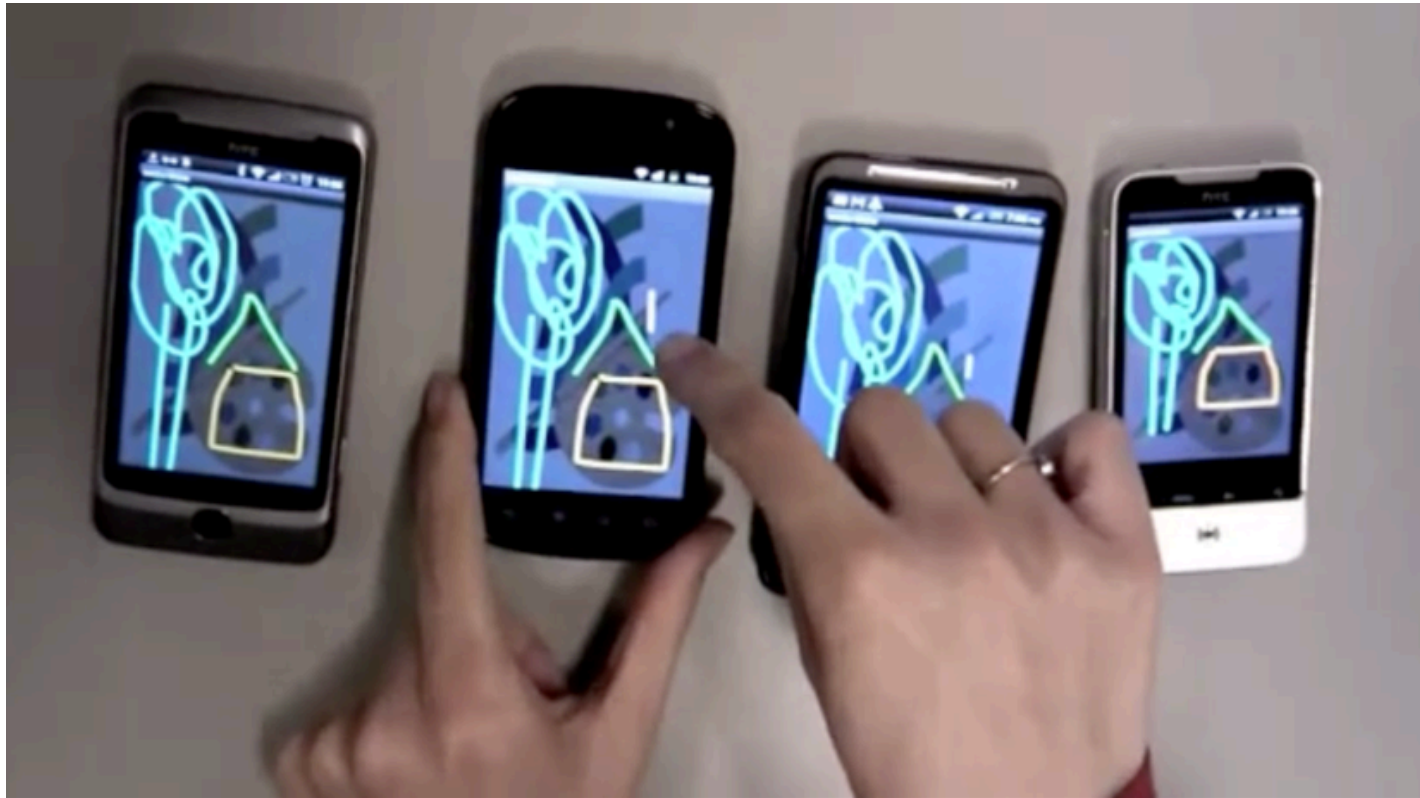


Vrije
Universiteit
Brussel



weScribble

Make drawings with your friends!

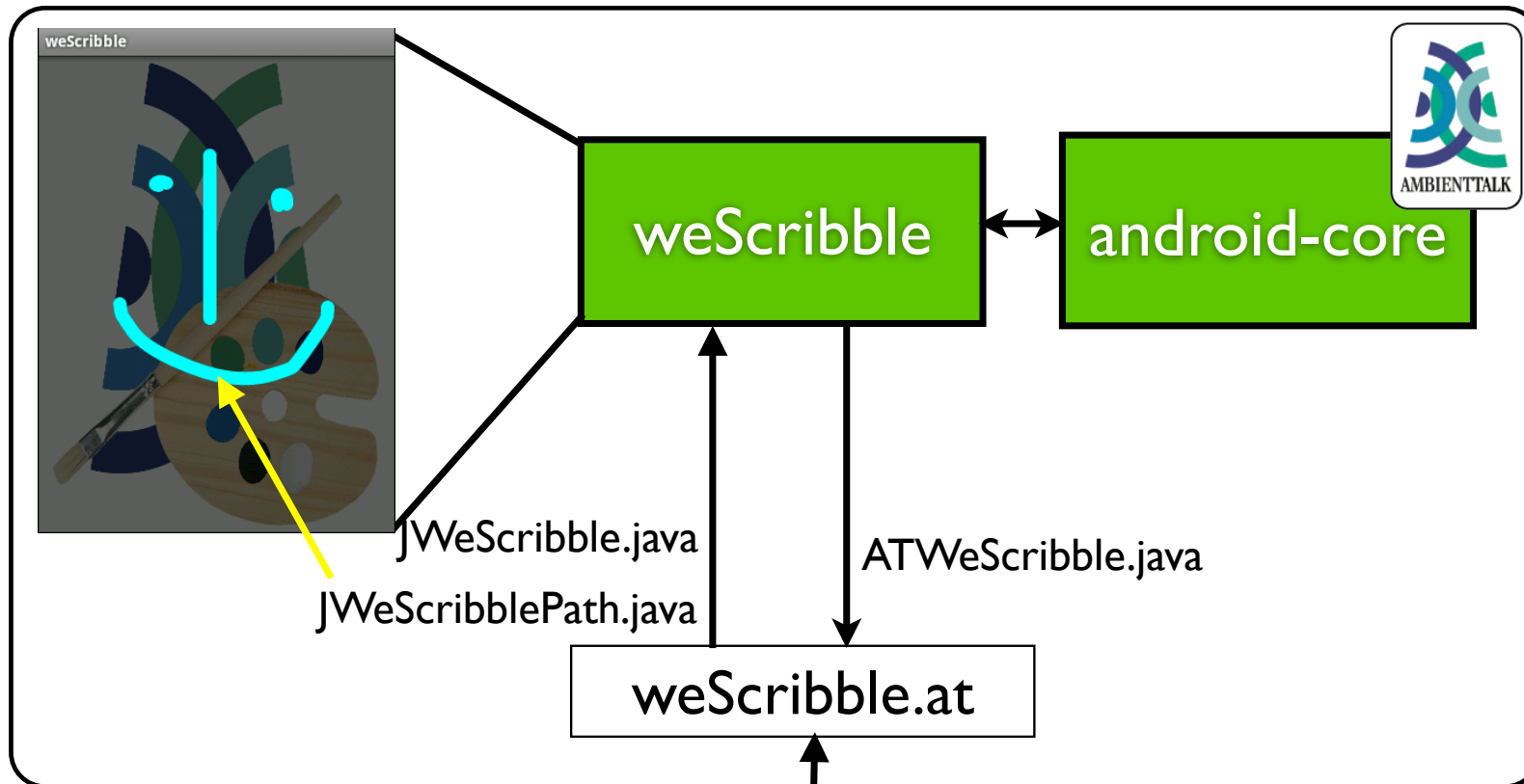


<http://youtu.be/k0HYqRCxtHc>

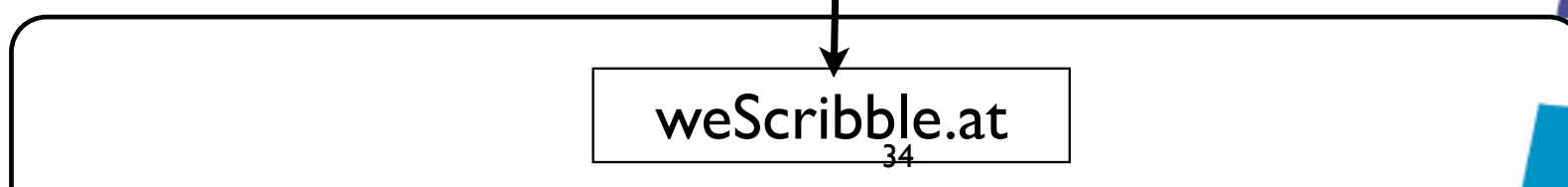


WeScribble

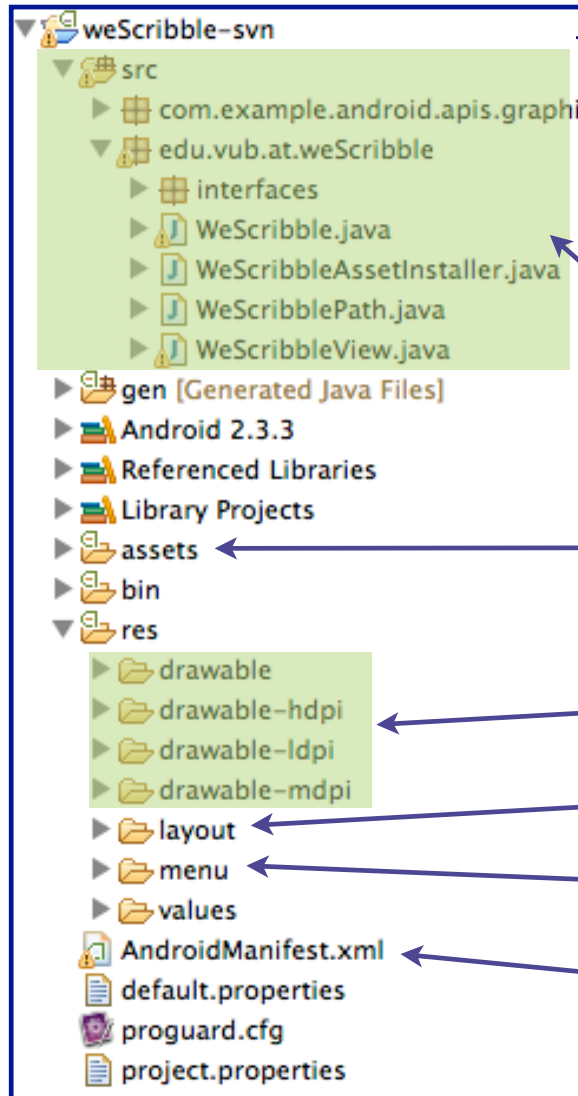
Phone



Phone



Android Project Layout



code + data → .apk file

Source files

Assets (copied to device)

Icons

Screen layouts

Menu definitions

Manifest file

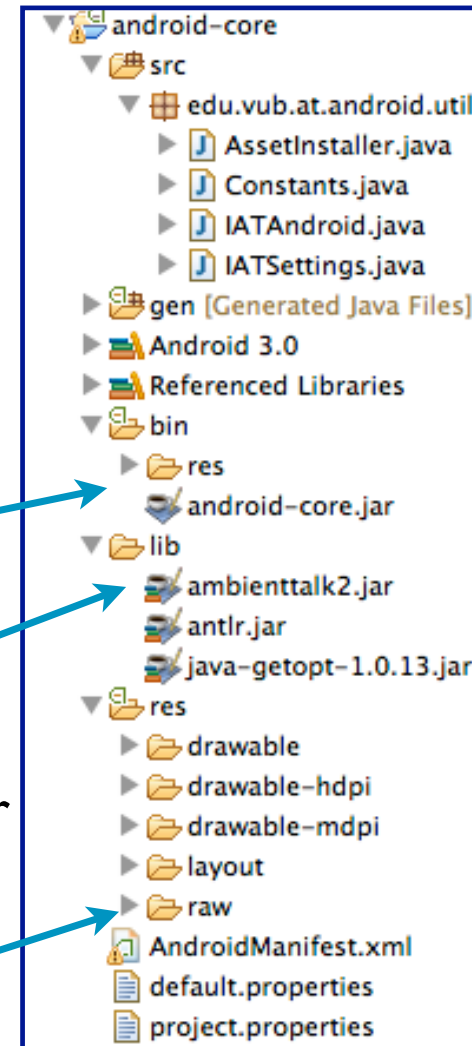
Android Library Project

- Shared code or resources are organized in a library project.
- Referenced from other Android project

does not compile
as an .apk

can include jar file,
but it cannot be exported to a .jar

cannot include its
own raw assets



WeScribble

installed directly on
a real device (if any connected)

- Install the application:
 - "Run As ->Android Application"
- Testing on virtual devices:

[http://code.google.com/p/ambienttalk/wiki/
AmbientTalkAndroidApplications](http://code.google.com/p/ambienttalk/wiki/AmbientTalkAndroidApplications)





AmbientTalk .net

Network Access

- Discovery works locally!
- By default, network access is disabled.

```
network.online()
```

```
network.offline()
```

- Useful iat command line arguments:

```
-n <network_name>
```

```
-a <ip_address>
```



Simulating Network Failures


- Disconnect far references to objects running on the same/different AmbientTalk VM:

```
def pub := disconnect: object;  
pub.reconnect;
```


Triggers `when:disconnected:` listeners!

Disconnects client-side of a reference
if object is already a far reference!



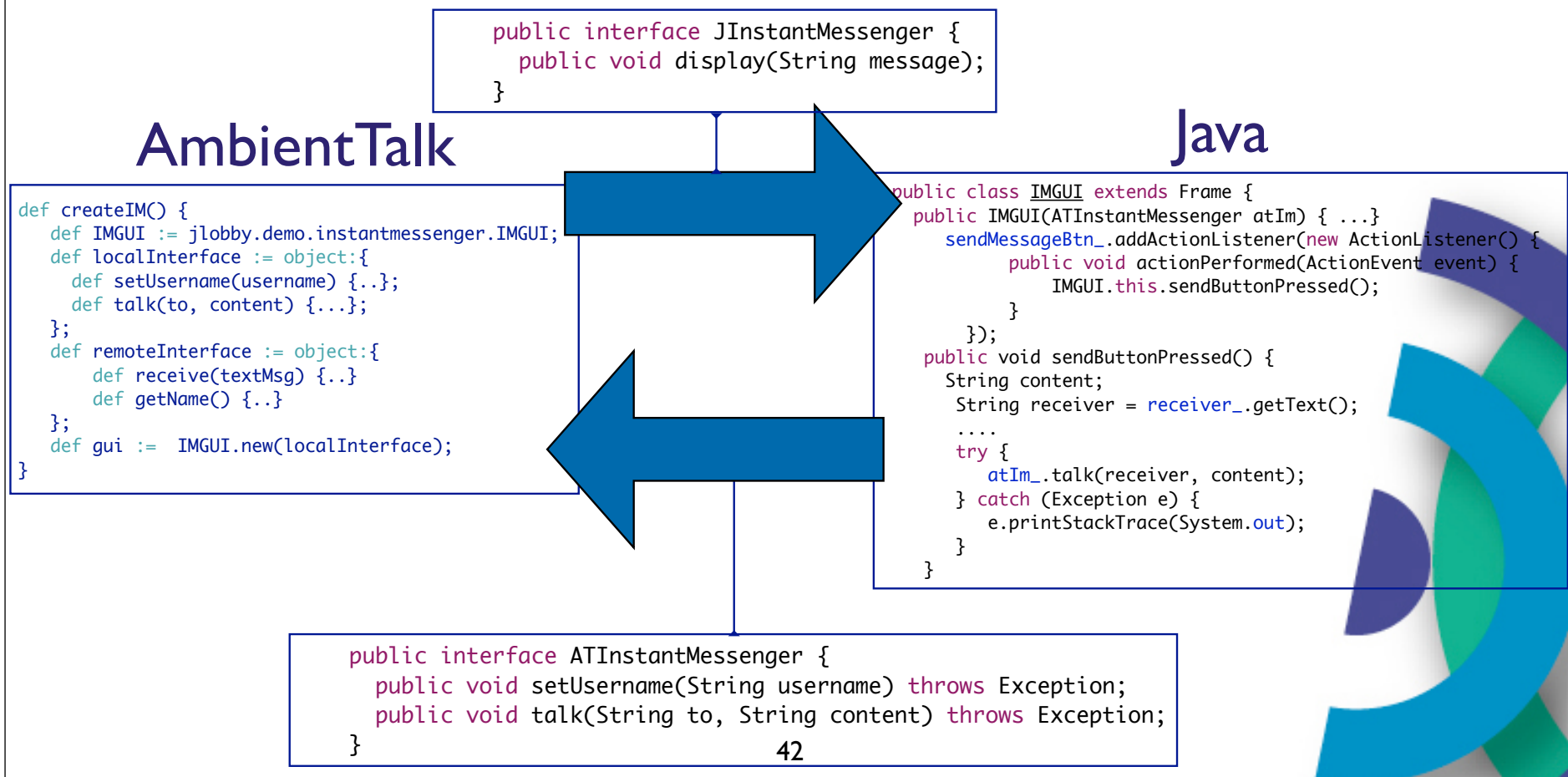


Symbiosis with Java and Android



Symbiosis with Java

- AmbientTalk applications launch the Java GUI by creating Java UI classes.



Symbiosis with Java

- AmbientTalk applications launch the Java GUI by creating Java UI classes.

AmbientTalk

```
def createIM() {  
  def IMGUI := jlobby.demo.instantmessenger.IMGUI;  
  def localInterface := object:{  
    def setUsername(username) {...};  
    def talk(to, content) {...};  
  };  
  def remoteInterface := object:{  
    def receive(textMsg) {...}  
    def getName() {...}  
  };  
  def gui := IMGUI.new(localInterface);  
}
```

```
public interface ATInstantMessenger {  
  public void setUsername(String username) throws Exception;  
  public void talk(String to, String content) throws Exception;  
}
```

Java

```
public class IMGUI extends Frame {  
  public IMGUI(ATInstantMessenger atIm) { ...}  
  sendMessageBtn_.addActionListener(new ActionListener() {  
    public void actionPerformed(ActionEvent event) {  
      IMGUI.this.sendButtonPressed();  
    }  
  });  
  public void sendButtonPressed() {  
    String content;  
    String receiver = receiver_.getText();  
    ....  
    try {  
      atIm_.talk(receiver, content);  
    } catch (Exception e) {  
      e.printStackTrace(System.out);  
    }  
  }  
}
```

Symbiosis with Android

- Android applications no `main()` function!
- AmbientTalk applications are launched within an Android project.



Launching AmbientTalk Apps

```
public class WeScribble extends Activity implements
    ColorPickerDialog.OnColorChangedListener, JWeScribble {

    public class StartIATTask extends AsyncTask<Void, String, Void> {
        @Override
        protected Void doInBackground(Void... arg0) {
            try {
                iat = IATAndroid.create(WeScribble.this);
                this.publishProgress("Loading weScribble code");
                iat.evalAndPrint(
                    "import /.demo.weScribble.weScribble.makeWeScribble()",
                    System.err
                );
            } catch (Exception e) {
                Log.e("AmbientTalk", "Could not start IAT", e);
            }
            return null;
        }
    }
}
```

....



Invoking Android code

- How do we invoke Android UI code within AmbientTalk?

```
def Android := object: {  
  def parent := jlobby.edu.vub.at.android.util.IATAndroid.getInstance().getParentActivity();  
  def Log := jlobby.android.util.Log;  
};
```

Android object is
defined in the root lexical scope.



Symbiosis with Android

```
def wifiManager := Android.parent.getSystemService(  
jlobby.android.content.Context.WIFI_SERVICE);  
def ipAddress := wifiManager.getConnectionInfo().getIpAddress();
```

- Access to the Log facilities:

```
Android.Log.v("myAmbientTalkApp", "helloWorld");
```

