



J-Spring 2007

XUL & Swing,
Een natuurlijke combinatie



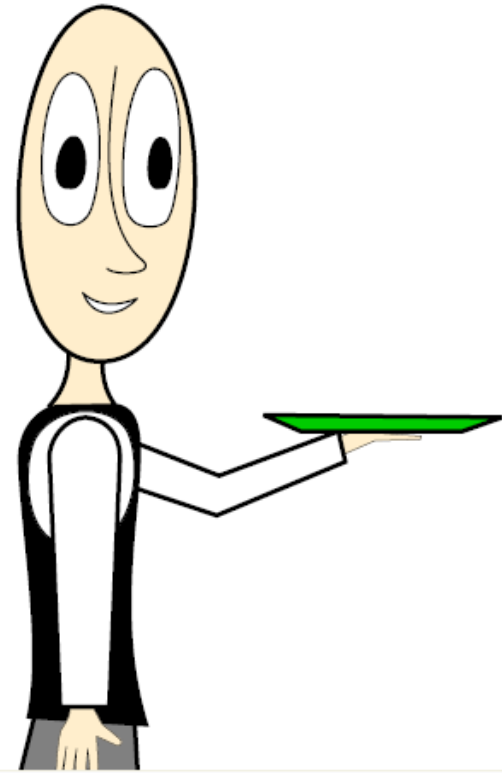
Even voorstellen

- 👤 Jan Tuitman
- 👤 Developer bij wisdom
- 👤 Hobby: stiptekenen
- 👤 <http://www.tuitman.org/>



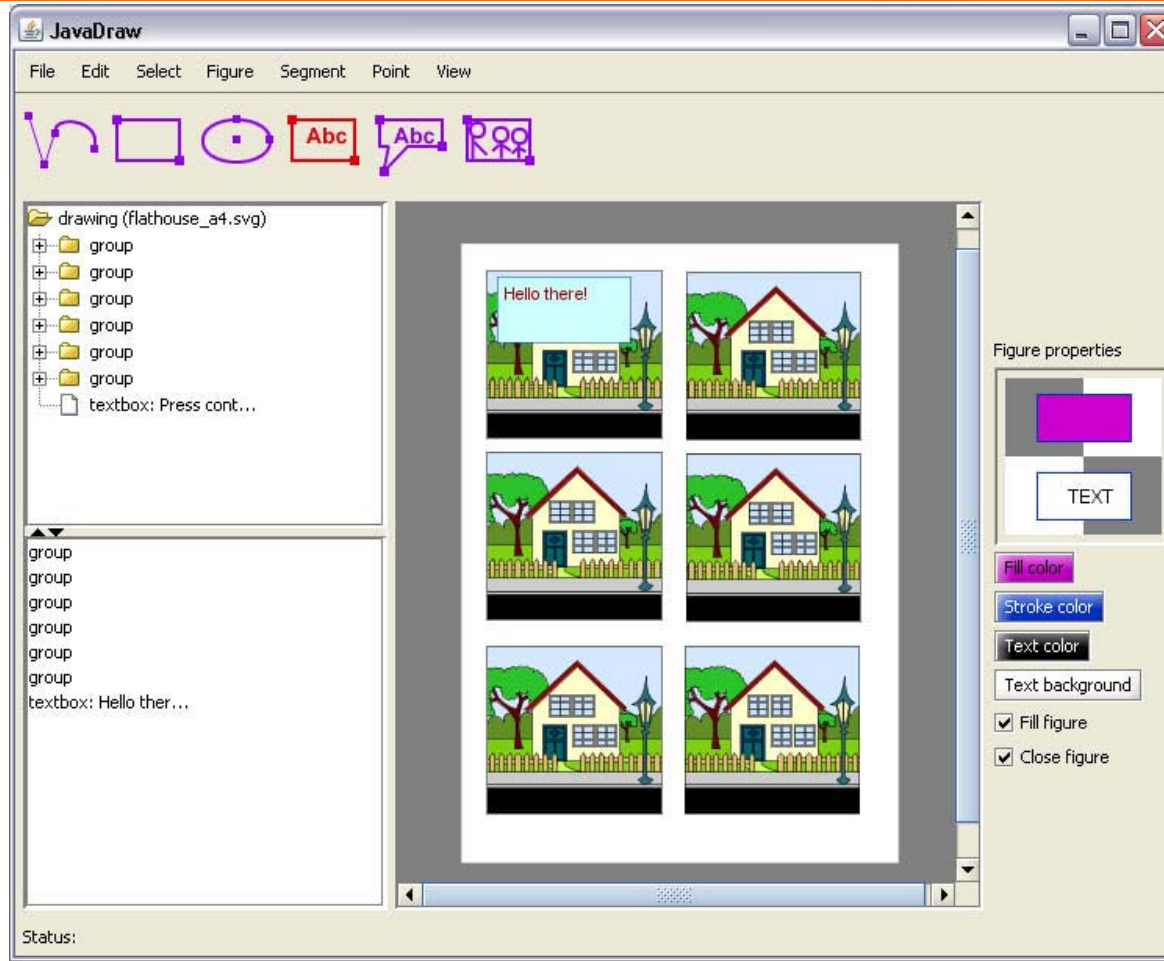
Even voorstellen

- 🦋 Jan Tuitman
- 🦋 Developer bij wisdom
- 🦋 Hobby: stiptekenen
- 🦋 <http://www.tuitman.org/>





Even voorstellen





Wat gaan we doen?

- 🚀 (Wat is Swing?) wat is XUL?
- 🚀 SwiX^{ml} :Hello world
- 🚀 Model, view, controller
- 🚀 Conclusies, vragen



(wat is swing?)

Prachtige library, maar...

```
//Create a formatted text field.
```

```
JFormattedTextField ftf = new JFormattedTextField(  
    java.util.Calendar.getInstance().getTime());  
ftf.setActionCommand(textFieldString);  
ftf.addActionListener(this);
```

```
//Create some labels for the fields.
```

```
JLabel textFieldLabel = new JLabel(textFieldString + ": ");  
textFieldLabel.setLabelFor(textField);  
JLabel passwordFieldLabel = new JLabel(passwordFieldString + ": ");  
passwordFieldLabel.setLabelFor(passwordField);  
JLabel ftfLabel = new JLabel(ftfString + ": ");  
ftfLabel.setLabelFor(ftf);
```

```
//Create a label to put messages during an action event.
```

```
actionLabel = new JLabel("Type text in a field and press Enter.");  
actionLabel.setBorder(BorderFactory.createEmptyBorder(10,0,0,0));
```



(wat is swing?)

```
private void addLabelTextRows(JLabel[] labels,
    JTextField[] textFields,
    GridBagLayout gridbag,
    Container container) {
    GridBagConstraints c = new GridBagConstraints();
    c.anchor = GridBagConstraints.EAST;
    int numLabels = labels.length;
    for (int i = 0; i < numLabels; i++) {
        c.gridwidth = GridBagConstraints.RELATIVE; //next-to-last
        c.fill = GridBagConstraints.NONE; //reset to default
        c.weightx = 0.0; //reset to default
        container.add(labels[i], c);

        c.gridwidth = GridBagConstraints.REMAINDER; //end row
        c.fill = GridBagConstraints.HORIZONTAL;
        c.weightx = 1.0;
        container.add(textFields[i], c);
    }
}

public void actionPerformed(ActionEvent e) {
    String prefix = "You typed \";
    if (textFieldString.equals(e.getActionCommand())) {
        JTextField source = (JTextField)e.getSource();
        actionLabel.setText(prefix + source.getText() + "\");
    } else if (passwordFieldString.equals(e.getActionCommand())) {
        JPasswordField source = (JPasswordField)e.getSource();
    }
}
```

...



Wat is XUL?

- ✈ Xml User interface Language
- ✈ Mozilla
- ✈ Meer dan html geschikt voor applicaties
- ✈ Layout door xml-nesting: hbox,vbox
- ✈ Dit laatste sluit aan bij java.



Java implementaties

🐱‍🚀 java XUL implementaties:

☕ thinlet

☕ luxor

☕ xulux

☕ XSWT

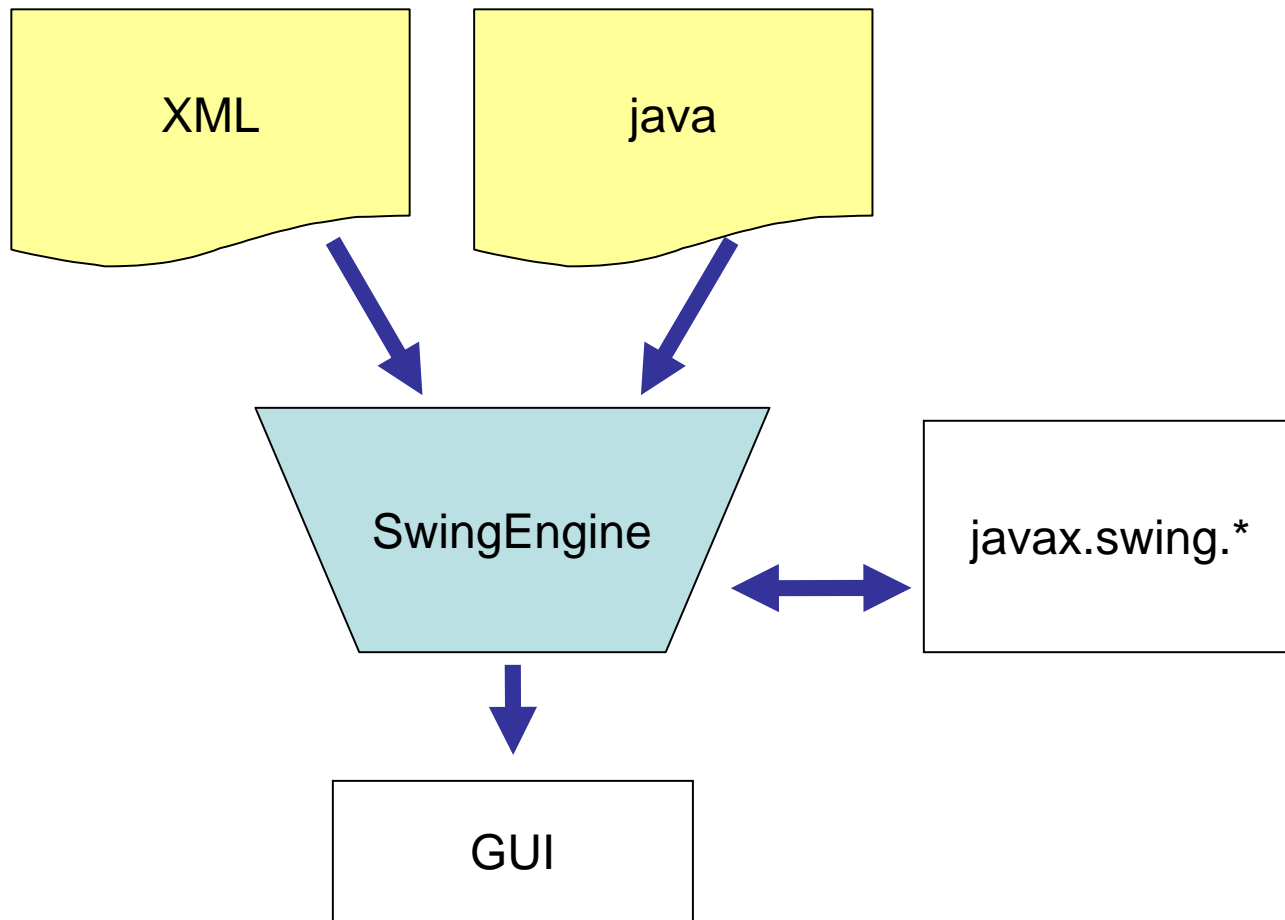
☕ SwiX^{ml}



- 🚀 SwiX^{ml} vindt het wiel niet opnieuw uit
- 🚀 Hierdoor: eenvoudig en klein!
- 🚀 open source en gratis
- 🚀 Auteur: Wolf Paulus
- 🚀 Huidige versie 1.5, plannen voor 2.0
- 🚀 <http://www.swixml.org/>



SwiX^{ml} Hello World





SwiX^{ml} Hello World

🚀 Java code:

```
new SwingEngine( this )  
    .render( "xml/helloworld.xml" )  
    .setVisible( true );
```

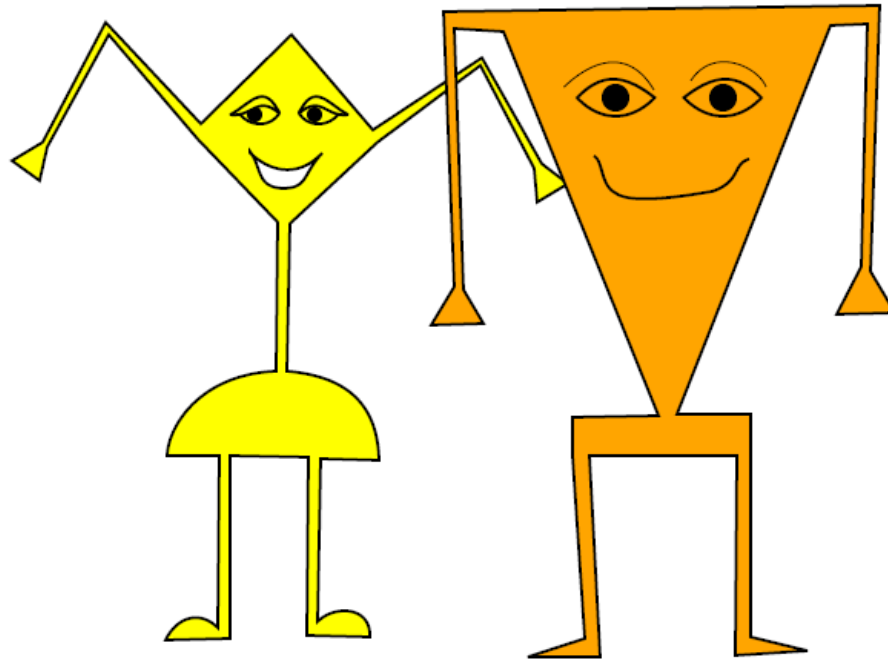


SwiX^{ml} Hello World (xml)

```
<?xml version="1.0"?>
<frame size="640,280" title="Hello SWIXML World"
  defaultCloseOperation="JFrame.EXIT_ON_CLOSE">
  <panel constraints="BorderLayout.CENTER">
    <label labelfor="tf" font="Georgia-BOLD-12" foreground="blue"
      text="Hello World!"/>
    <textfield id="tf" columns="20" Text="Swixml"/>
    <button text="Click Here" action="submit"/>
  </panel>
  <panel constraints="BorderLayout.SOUTH">
    <label font="Georgia-BOLD-36" text="Clicks:"/>
    <label font="Georgia-BOLD-36" id="cnt"/>
  </panel>
</frame>
```



Demo: hello world





Xml elementen

- ✈ Elementen corresponderen met java-classes
 - ☛ frame = javax.swing.JFrame
 - ☛ panel = javax.swing.JPanel
 - ☛ etc



Xml elementen

🚀 Custom elementen SwiXml
☛ hbox, vbox, glue : BoxLayout

🚀 Zelf component toevoegen.
`e.getTaglib().registerTag("GraphicsPanel",GraphicsPanel.class);`



Xml attributen

- ✎ Corresponderen met “set/get” methods.
- ✎ Type v.h. attribuut bepaalt formaat
- ✎ Constanten kunnen worden gebruikt.
- ✎ `PrimitiveConverter.addConstantProvider(JSplitPane.class);`

`constraints=“BorderLayout.SOUTH”`

`font=“Georgia-BOLD-36”`

`size=“200,100”`



Xml attributen

- 🐱 custom attributen
 - ☛ id (alle componenten)
 - ☛ initClass (combobox)
 - ☛ mac-specifieke attributen



Rapid prototypen

- ✈ WYSIWYG tool:
 - ☛ SwixEditor van Invivo software (niet gratis)
- ✈ Meegeleverde (of eigen!) teksttool
- ✈ Schema validatie
- ✈ Slechts beperkte javakennis nodig

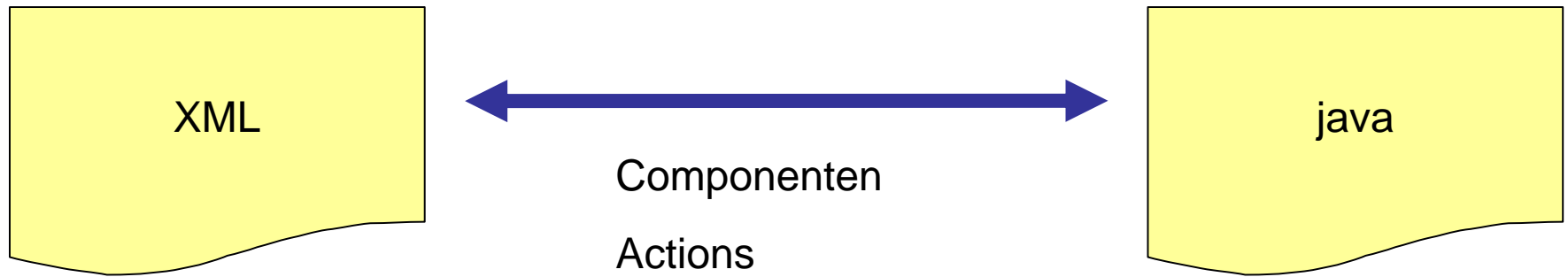


Demo 2





Java





Actions

```
<button text="Click Here" action="submit"/>
```

Klassieke swing acties:

```
Action submit = new AbstractAction () {
```

```
....
```

```
}
```

Of:

```
public void submit() { .... }
```



Componenten opvragen

```
<label font="Georgia-BOLD-36" id="cnt"/>
```

Op runtime:

```
JLabel x = (JLabel) e.getIdMap().get("cnt");
```

Of declaratief:

```
JLabel cnt ;
```



Werkende demo

```
int clicks=0;
public JTextField tf;
public JLabel cnt;

public void submit() {
    tf.setText( tf.getText() + '#' );
    cnt.setText(String.valueOf( ++clicks ));
}
```

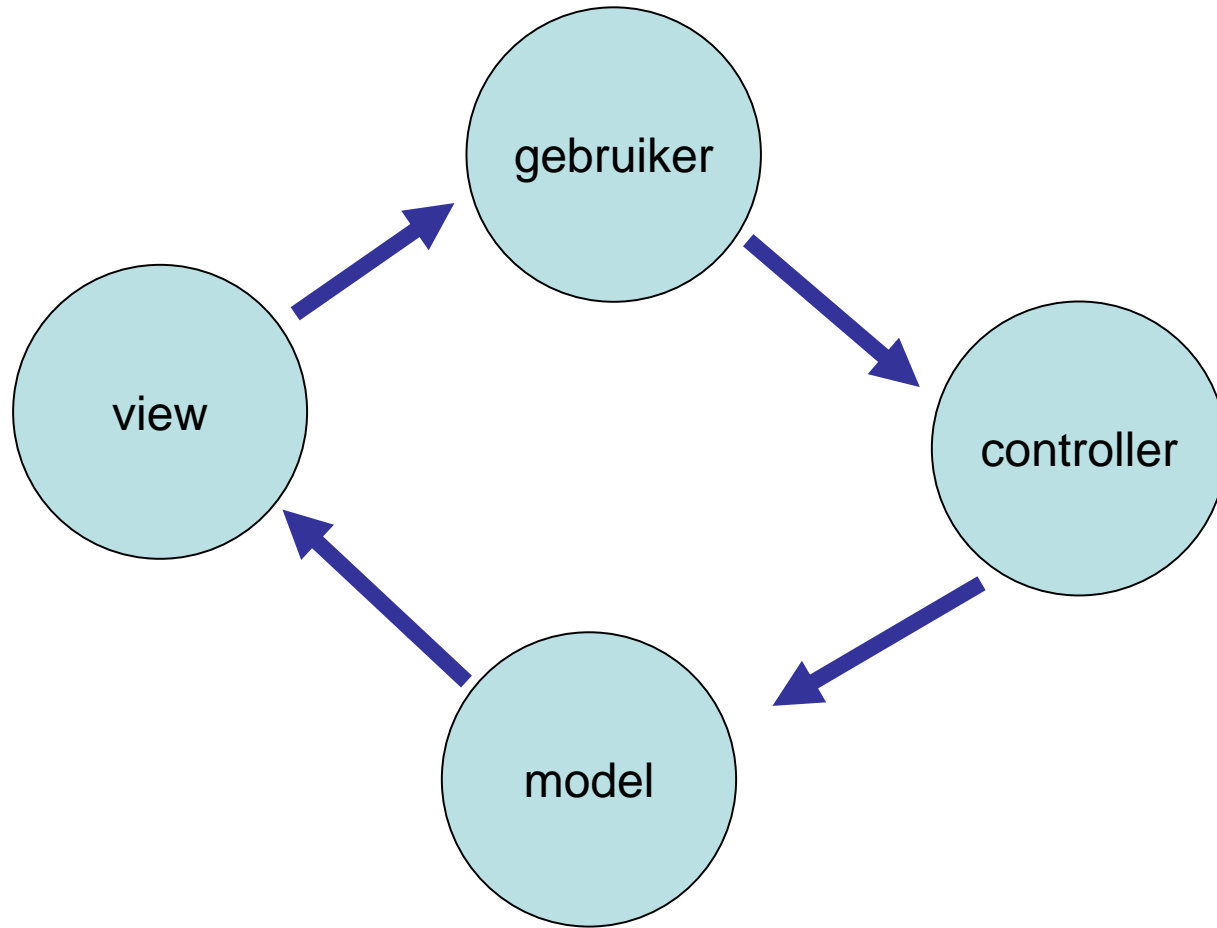


Werkende demo



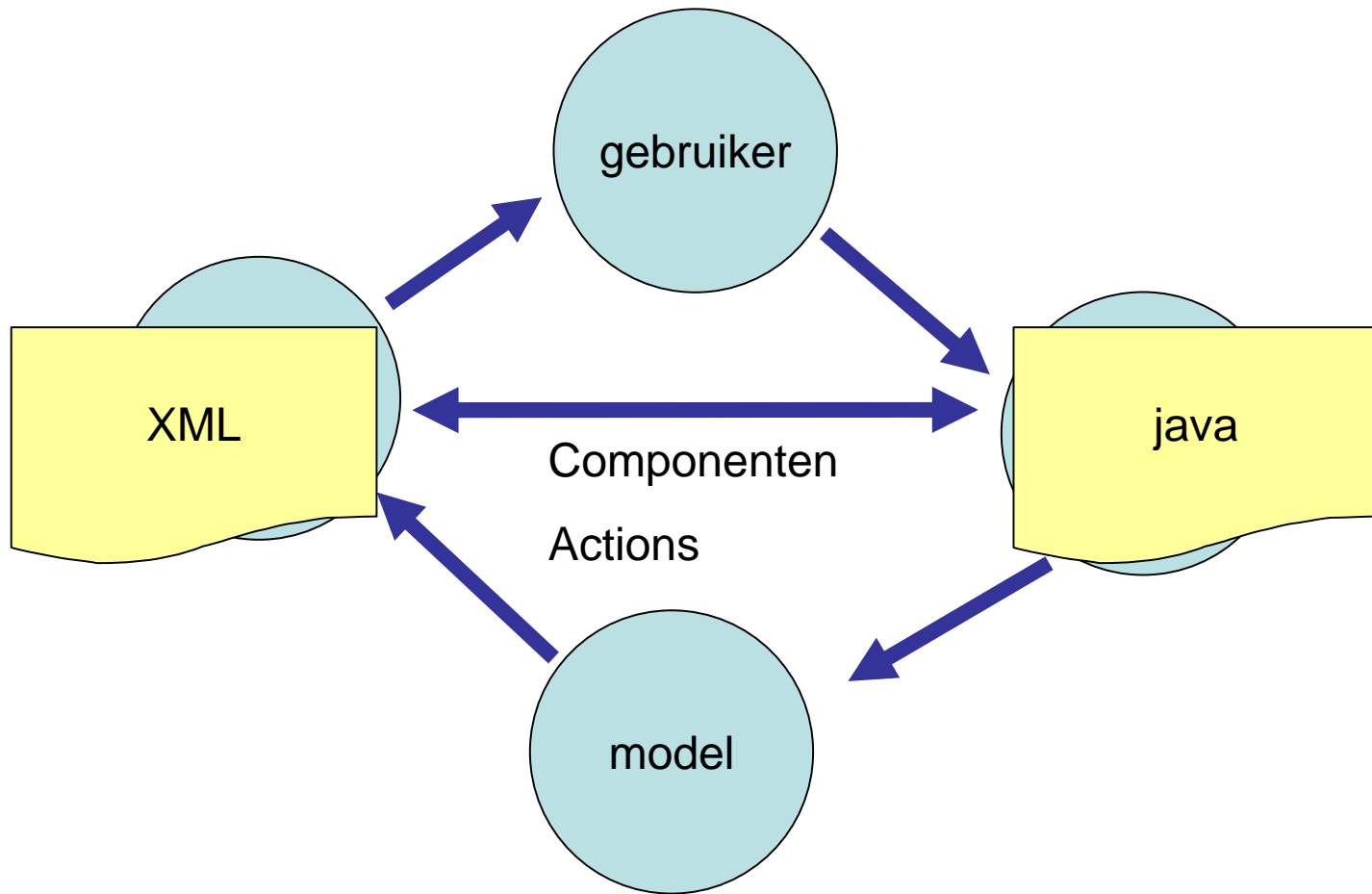


MVC





MVC





Los gekoppeld

- ✦ Losse view, losse controller
 - ☛ Xml-file: kan gevalideerd worden
 - ☛ Java: kan gecompileerd worden.
- ✦ Toch een koppeling: door namen
 - ☛ Makkelijk een unittest voor te bouwen



Meer mogelijkheden...

- 🚀 internationalisatie
- 🚀 xml:include
- 🚀 comboboxen populieren



... en wat ook zou kunnen

- 🚀 Zelf unittests schrijven voor swixml
- 🚀 Uitbreiden swixml
- 🚀 Het principe van koppelen-per-naam ergens anders gebruiken.



Vragen?

