

Programmiermethodik

Übung 8

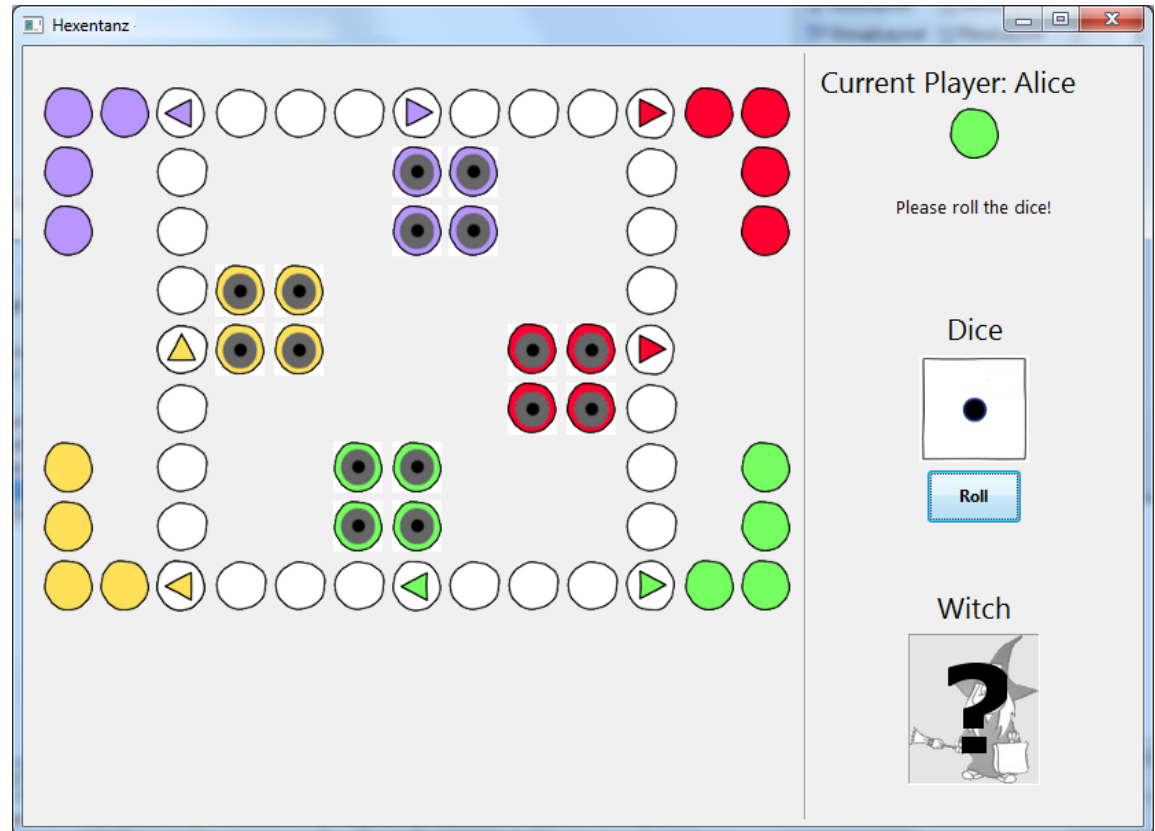
Wintersemester 2011/12
Fachgebiet Software Engineering

Tobias George
george@uni-kassel.de

Agenda

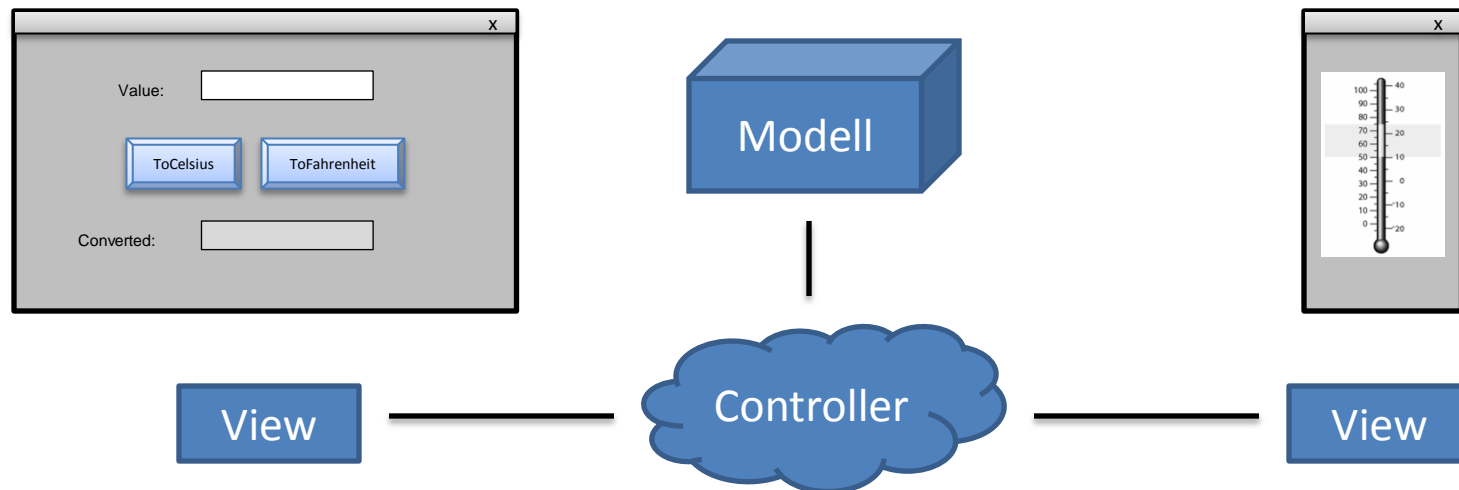
- **Besprechung HA 6**
- **Model – View – Controller**
- **Live-Demo Temperatur Converter**
- **Vorschau HA 7**
- **Praktische Übung: HA 7 anfangen**

Besprechung HA6

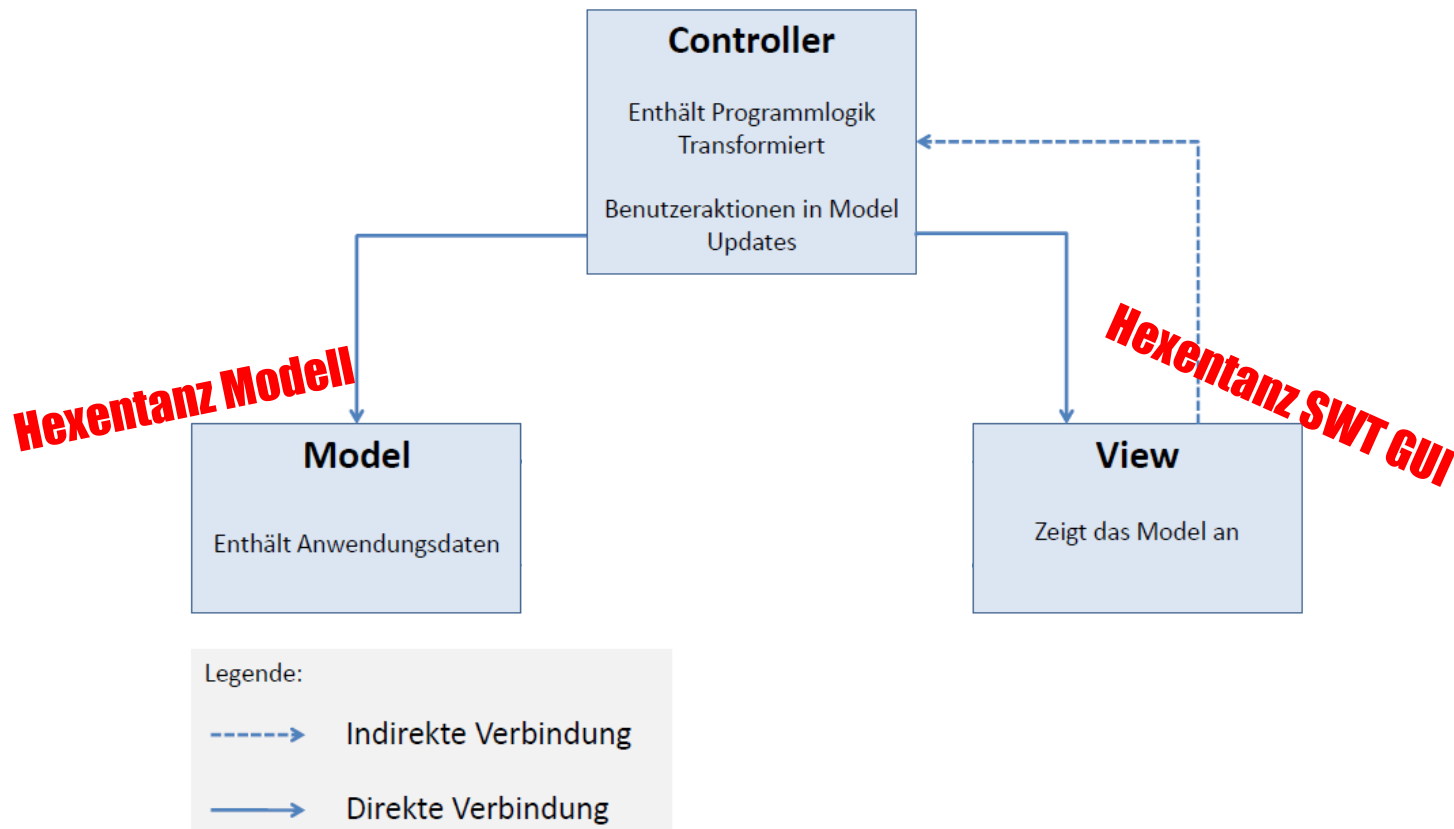


Model – View – Controller (MVC) I

- **MVC ist ein Entwurfsmuster**
 - Entwurfsmuster lösen immer wiederkehrende Softwaretechnische Probleme
- **Entkoppelt die Anwendungslogik von ihrer Darstellung**
- **Vorteile**
 - Modell und Logik müssen nur einmal programmiert werden
 - Verschiedene (gleichzeitige) Darstellungen für dasselbe Modell

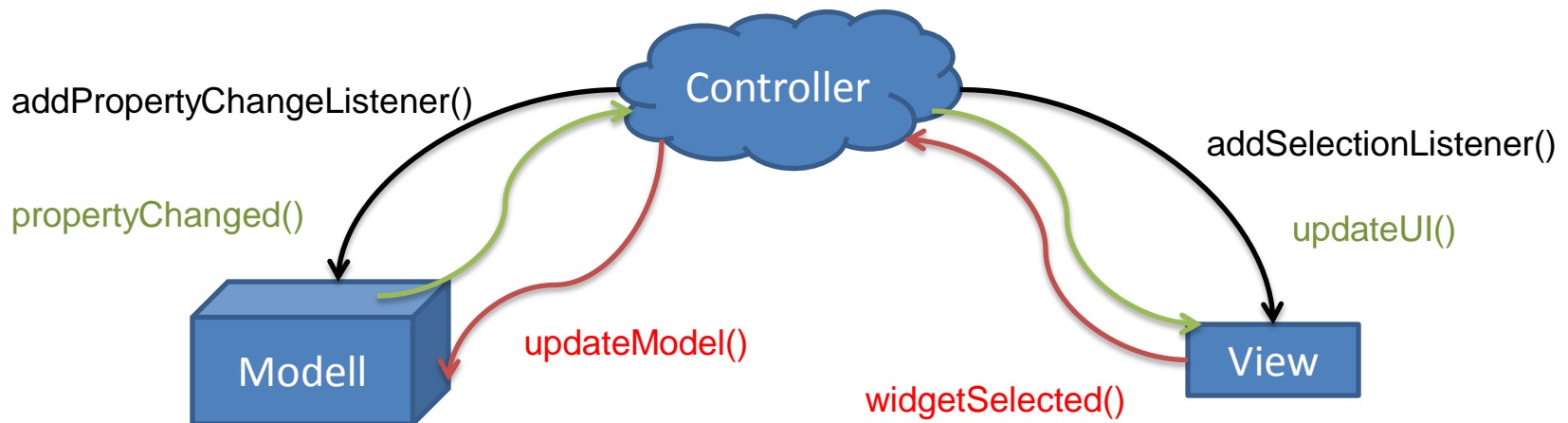


Model – View – Controller II



MVC III

- Controller meldet sich als Listener an Modell und View an
- Controller empfängt User-Event (z.B. Button „XYZ“ gedrückt)
- Controller verändert Modell
- Modell informiert seine PropertyChangeListener (u.a. den Controller)
- Controller updated UI



MVC IV

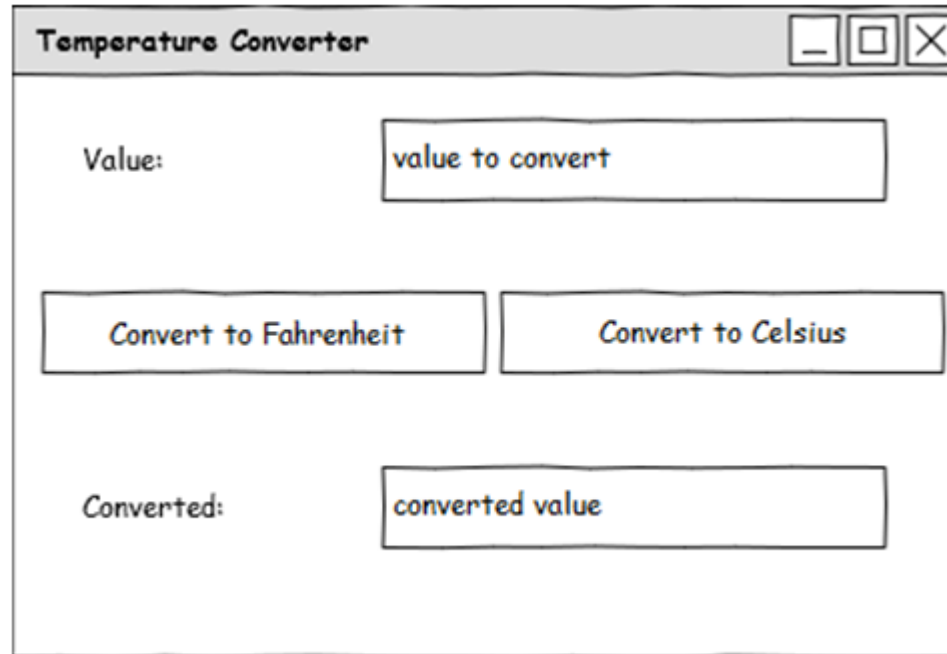
- **Problem: Modell und View müssen Mechanismen bereitstellen um Listener anzumelden**
- **Bei den meisten GUI Bibliotheken kein Problem!**
- **Modell: Bei Settern immer angemeldete Listener informieren**
 - Viel Aufwand
 - Immer das gleiche
 - Warum nicht einfach mitgenerieren lassen? → Fujaba JavaBean Stereotyp
- **<<JavaBean>> Stereotyp**
 - An jede Klasse an der man PropertyChangeListener anmelden will
 - `addPropertyChangeListener(PropertyChangeListener listener)`
 - `removePropertyChangeListener(PropertyChangeListener listener)`

Model – View – Controller V

- **Modellelemente (Klassen) in Fujaba müssen den JavaBean Stereotyp besitzen**
- **Implementierungstipps für Controller**
 - start() : Startet den Controller.
 - „Kind-Controller“ erzeugen und starten. (z.B. GameController -> PlayerController)
 - Listener registrieren (Modell und GUI)
 - refreshUI()
 - stop(): Stoppt den Controller
 - Listener abmelden
 - refreshUI(): Grafische Oberfläche updaten

Live-Demo: Controller mit Jface entwickeln

Entwicklung eines Controllers für:



- **Umrechnungsformeln:**
 - Celsius in Fahrenheit = $((T_{\text{Celsius}} \times 9) / 5) + 32$
 - Fahrenheit in Celsius = $(T_{\text{Fahrenheit}} - 32) \times 5 / 9$

Vorschau HA 7

- **Deadline: 12.01.2012, 23:59 Uhr**
- **Aufgabe 1.1: LoginController erstellen**
- **Aufgabe 1.2: GameController erstellen**
- **Aufgabe 1.3: PlayerController erstellen**
- **Aufgabe 1.4: WitchController erstellen**
- **Aufgabe 1.5: DiceController erstellen**
- **Aufgabe 1.6: Startklasse mit main-Methode, die den Login-Controller startet**

Praktische Übung: HA9 anfangen

- Schreibt den LoginController
 - Bekommt im Konstruktor das HexentanzGame Modell Objekt sowie die Loginoberfläche (HexentanzLoginScreen GUI) übergeben.
 - Drückt der Nutzer auf „Start Game“, soll zunächst geprüft werden, ob in mindestens zwei Textfelder etwas eingegeben wurde. Falls nicht, soll ein Popup erscheinen mit einer entsprechenden Meldung (Tip: Hierzu bietet sich die Klasse JOptionPane an).
 - Drückt der Nutzer auf „Start Game“ und es wurden mindestens zwei Namen eingegeben, soll der GameController instanziiert und gestartet werden. Der LoginController hat damit seinen Dienst erfüllt und kann gestoppt werden.
 - Drückt der Nutzer auf „Exit Game“ soll das Spiel beendet werden.

Ende

Jetzt: Betreutes Arbeiten

Ansonsten: Schönes WE!