



Swing Pluggable Look & Feel

Dipl.-Ing. Michael Johann

Unternehmensberatung j.talk()

mjohann@javatalk.net



Agenda

- **Warum Swing?**
- **Was ist das MVC-Pattern?**
- **GUI-Erstellung mit Swing**
- **Gründe für die Erstellung eines eigenen Look & Feels für Swing und wie es gemacht wird**
- **Design Grundlagen für das Look & Feel von Swing**
- **Hardcore Live Coding**



Warum Swing?

■ Das AWT

- **Peer Klassen :-()**
 - **Vorteil: Höhere Performanz**
- **Geringe Flexibilität**
- **Plattformabhängiges Aussehen**
 - **LayoutManager etc.**
- **Arthur van Hoff: We built the AWT in 30 days (uugghh)**



Warum Swing?

- **Swing ist peerless**
 - **Komponenten zeichnen sich selbst**
 - **Unterschiedliches Aussehen auf verschiedenen Plattformen realisierbar**
 - **Keine Probleme mit dem Layout aber geringere Performanz**
- **Viele neue Komponenten**
 - **JTree, JTable etc.**
- **Umfangreiche Klassenbibliothek**
 - **Aber: Beware of the bugs**

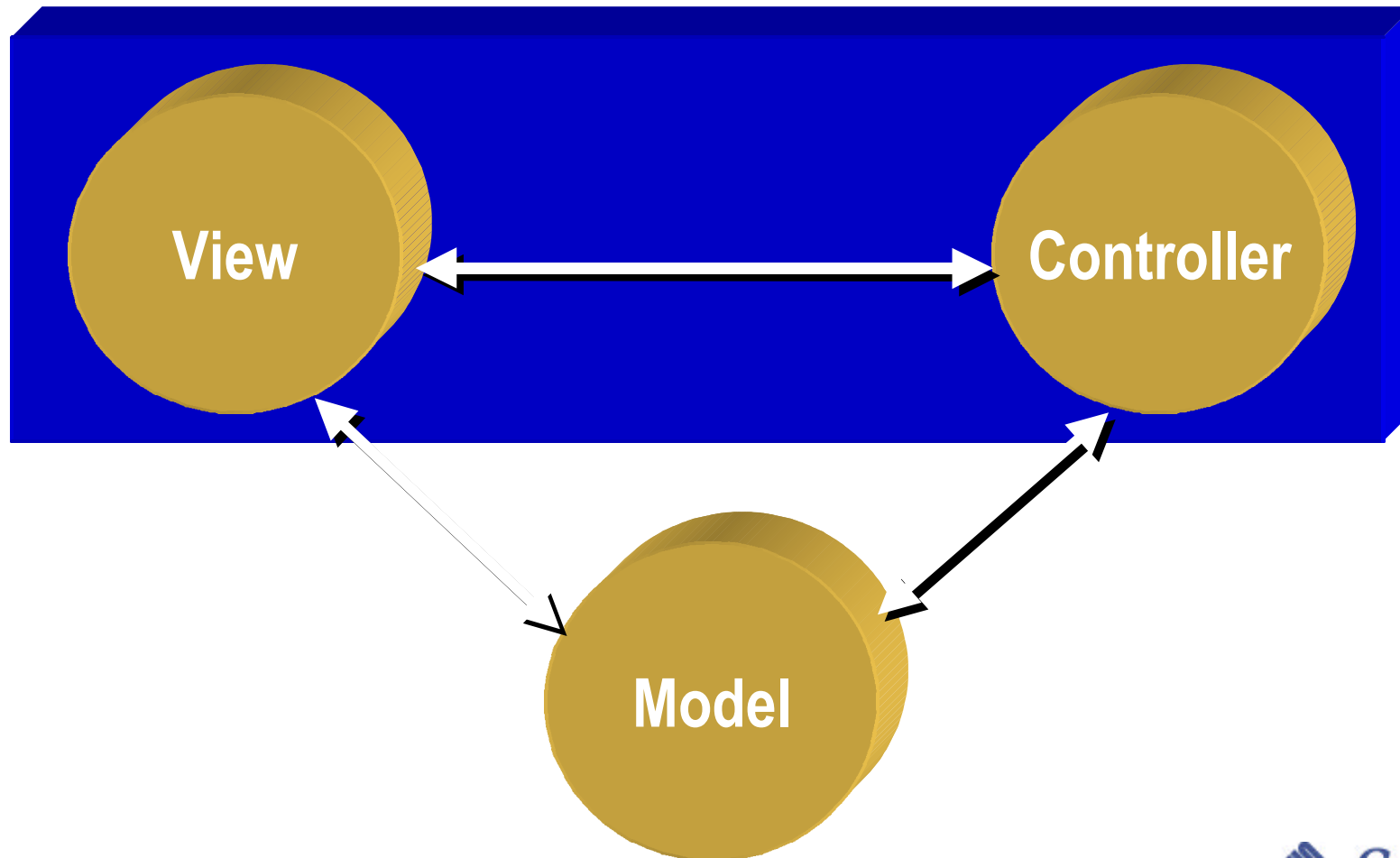


Was ist das MVC-Pattern?

- **MVC == Model View Controller**
 - **Bekannt aus Smalltalk**
 - **Separation von grafischer Anzeige und Applikationslogik auf Komponentenebene**
 - **Mehrere Ansichten für dieselbe Logik sind möglich**
 - **In Smalltalk Separate Klassen für Model, View und Controller**
 - **In Java Separate Model Klassen und View/Controller sind in UI-Klassen untergebracht**



Was ist das MVC-Pattern?





Was ist das MVC-Pattern?

- **Fast jede Komponente ist von JComponent abgeleitet**
 - z.B. JButton, JTree, JTable
 - Ausnahmen: JFrame, JDialog, JWindow (benutzen Peers)
- **Jede Subklasse hat eine gehört zu einem UI und einem Model**
- **Jede Subklasse ist gleichzeitig Controller (Nimmt Benutzereingaben entgegen)**
 - (z.B. addActionListener)
- **Mehr Flexibilität mit Border-Klassen**
 - Eigene Border-Arten sind einfach realisierbar



GUI-Erstellung mit Swing

- **Nicht alle Komponenten werden optimal unterstützt**
 - z.B. JSplitpane, JdbTable, JScrollPane with JTable (Abhilfe kann eine neue Swingversion sein)
- **Model-Klassen sind nicht als Komponenten verfügbar und müssen manuell verwendet werden**
- **Für die UI-Klassen gilt dasselbe**
- **Fazit: Es gibt keine visuelle Entwicklungsumgebung, die das Erstellen von Look & Feels für Swing mit Werkzeugen unterstützt.**



Demo: GUI-Erstellung mit Swing

Gründe für die Erstellung eines eigenen Look & Feels für Swing und wie es gemacht wird

JAVA DAYS '98



■ Warum...

- Sie mögen Windows nicht (oder Motif or Metal)
- Sie haben ein ansprechendes CI und wollen dieses in Ihre Applikationen integrieren ohne sie verändern zu müssen

■ Wann...

- Nur wenn das Budget und andere Ressourcen verfügbar sind
- Einbindung von Grafik-Designern ist zu empfehlen

Gründe für die Erstellung eines eigenen Look & Feels für Swing und wie es gemacht wird

JAVA DAYS '98



■ Wie...

- Just do it ;-)
- Ein (viele) Blick(e) in die Quellen der Swing-API helfen
- Schauen Sie sich konkret das `com.sun.java.swing.plaf.basic` package

Design Grundlagen für das Look & Feel von Swing



- **Jede Komponente hat eine UI-Klasse**
 - z.B. JComponent hat ComponentUI
 - z.B. JTree hat TreeUI
 - etc.
- **Verwenden Sie die Border-Schnittstelle**
 - Wichtig: paintBorder() Methode
- **Studieren Sie den UIManager und LookAndFeel**
 - Der Schlüssel zum Erfolg

Design Grundlagen für das Look & Feel von Swing



- **Entwickeln Sie IconFactories und BorderFactories**
 - Wenn Sie Icons zeichnen wollen (z.B. JTree)
- **Benutzen Sie GraphicsUtils von anderen PLAFs**
- **createUI, installUI and updateUI sind wichtig**
 - Demo BasicPasswordUI und ProfiPasswordUI
 - Demo UIManager
 - Demo BasicLookAndFeel
 - Demo ProfiLookAndFeel



Hardcore Live Coding

- **Wie wird das neue Look & Feel in Applikationen integriert?**
- **Wie eigene Border-Klassen erstellt werden**
- **Demo**
 - **Erstellen eines Borders**
 - **Erstellen einer UI-Klasse für JButton**
 - **Einbinden in Applikationen**



Call For Action

- **Stellen Sie klar, ob Sie ein eigenes Look & Feel benötigen**
- **Desingen Sie ein schönes Look & Feel mit Designern**
- **Schauen Sie sich vorhandene PLAFs an**
- **Vorsicht bei Swing 1.1**
 - **Vieles wird sich ändern**
- **Registrieren Sie sich bei developer.javasoft.com und lesen Sie regelmäßig die Swing-Connection**



Fragen & Antworten