

Ereignisbehandlung in Swing

Dr. Andreas Schroeder



Was dieses Video behandelt

- Die wichtigsten High-level Events und Listener
 - ActionListener
 - ChangeListener
 - ComponentListener
- Input Events und Event Consumption
 - KeyListener
 - MouseListener
- EventListener vs. EventAdapter



Ablauf der Ereignisbehandlung

1. Benutzer interagiert mit Swing-Komponente
2. Swing erzeugt Event für die Interaktion und Komponente
3. Swing benachrichtigt Event Listener der Komponente
4. Event Listener-Implementierungen stoßen die eigene Programmlogik an



- ActionListener können registriert werden auf
 - JButton / JToggleButton (Click)
 - JTextField (Enter)
 - JComboBox (Selektion)
 - JCheckBox (Selektion)
 - JMenuItem (Click)
- ActionListener werden im Code sehr oft auftauchen



- Elaboriertere Komponenten verwenden ChangeListener
 - JSlider
 - JSpinner
 - JProgressBar
 - JCheckBox
 - JToggleButton
- ChangeListener werden benachrichtigt, wenn sich der Zustand ändert



- ComponentListener können auf jede Komponente registriert werden.
- Benachrichtigungen:
 - Resize, move
 - Shown, hidden
- Insbesondere resize kann interessant sein



- Input Events sind „low level“
 - Eingabeereignisse, die direkt vom Benutzer stammen
- Input Events können verbraucht (consumed) werden
 - Erlaubt das ursprüngliche Verhalten einer Komponente zu unterbinden
 - z.B. die kann für eine Chat-Eingabe die Enter-Taste in einem JTextArea abgefangen werden um eine Nachricht abzuschicken.



- KeyEvents sind Input Events
- Erlauben direktes Reagieren auf Tastaureingaben
- Benachrichtigungen:
 - Key pressed, key released (low level)
 - Key typed (high level)
- Achtung: nicht alle pressed/released events werden von einem key typed begleitet



- MouseEvents sind Input Events
- Für JPanels mit eigenen Zeichenroutinen interessant
- Benachrichtigungen:
 - Pressed, released
 - Clicked
 - Entered, exited
- Achtung: Click Events werden nur gefeuert, falls press und release gleiche Koordinaten haben
- Für Bewegungs-Events verwende MouseMotionListener



- Was tun wenn nur eine Methode (von vier) interessant ist?
- ... z.B. mouse released
- Hierzu gibt es Adapter
 - Abstrakte Klasse, die alle Methoden leer implementiert
 - Benötigte Methoden können überschrieben werden.



Was in diesem Video behandelt wurde

- Die wichtigsten High-level Events und Listener
 - ActionListener
 - ChangeListener
 - ComponentListener
- Input Events und Event Consumption
 - KeyListener
 - MouseListener
- EventListener vs. EventAdapter
 - Falls das Listener-Interface zu viele Methoden hat