

Einbau Mausereignisse

Einbau der Behandlung von
Mausereignissen
in den Raumplaner

Einbau Mausereignisse

- Basis sind
 - das Raumplanerprojekt, in das die Gui vollständig integriert ist.
 - die Datei *Leeres_Fenster_Mausereignisse.py*
- Erste Fragestellung:
Wohin gehört die Anwendung von Mausereignissen?
- Einfache Antwort:
 - In die Klasse Grafikfenster oder
 - in die Klasse Zeichenflaeche.

Einbau Mausereignisse

- Entscheidung für die Zeichenflaeche
- Kopieren der Bind-Anweisungen aus dem einfachen Mausprojekt in den Konstruktor von Zeichenflaeche

```
self.Bind(wx.EVT_LEFT_DOWN, self.OnLeftDown)
```

```
self.Bind(wx.EVT_LEFT_UP, self.OnLeftUp)
```

```
self.Bind(wx.EVT_LEFT_DCLICK, self.OnLeftDoubleClick)
```

```
self.Bind(wx.EVT_RIGHT_DOWN, self.OnRightDown)
```

```
self.Bind(wx.EVT_RIGHT_UP, self.OnRightUp)
```

```
self.Bind(wx.EVT_MOTION, self.OnMotion)
```

Einbau Mausereignisse

- Es werden nicht alle Ereignisse verwendet
- Zu den verwendeten werden jeweils einfache Methoden mit print-Anweisungen eingebaut:

```
def OnLeftDown(self, event):
```

```
    print( 'links angeklickt:', event.GetX(),event.GetY() )
```

```
def OnLeftUp(self, event):
```

```
    print( 'losgelassen:', event.GetX(),event.GetY() )
```

```
...
```

Einbau Mausereignisse

- Das Austesten zeigt eine große Zahl von Ausgaben, da bei „bewegt“ und „gezogen“ nicht nur Anfangs- und Endpunkt angezeigt werden, sondern auch Zwischenpunkte.

```
bewegt: 352 284  
bewegt: 350 283  
bewegt: 349 282  
bewegt: 348 281  
bewegt: 348 280  
bewegt: 348 278  
bewegt: 347 277  
links angeklickt: 347 277  
gezogen: 346 277  
gezogen: 345 278  
gezogen: 345 278  
gezogen: 345 279  
gezogen: 344 280  
gezogen: 343 281  
losgelassen: 343 281  
rechts angeklickt: 343 281  
rechts losgelassen: 343 281  
bewegt: 345 281  
bewegt: 355 281  
bewegt: 380 285
```

Einbau Mausereignisse

- Im zweiten Schritt geht es um die Frage, wo die Mausereignisse verarbeitet werden.
- Das Ziel ist das RaumplanerModell-Objekt.
- In einem ersten einfachen Schritt soll nur das Ereignis Linksklick zum Auswählen von Objekten verwendet werden.
- Alle anderen Ereignisbehandlungsmethoden werden deaktiviert.

Einbau Mausereignisse

- Wenn die Zeichenflaeche Nachrichten (*messages*) an das RaumplanerModell-Objekt schicken soll, muss es das Objekt kennen.
- Dazu bekommt es ein Attribut *self.__modell* und eine Set-Methode zum Setzen dieses Attributs, die von der RaumplanerApp beim Starten aufgerufen wird.

Einbau Mausereignisse

- Die Zeichenflaeche kann nun eine Nachricht an das RaumplanerModell-Objekt schicken:

```
def OnLeftDown(self, event):  
    x,y = event.GetX(),event.GetY()  
    #print( 'links angeklickt:', x, y )  
    self.__modell.Linksklick(x,y)
```


Einbau Mausereignisse

- RaumplanerModell muss die passende Methode bereitstellen, eine Triviallösung dazu wäre:

```
def Linksklick(self, x, y):  
    print( 'links angeklickt:', x, y )
```

Einbau Mausereignisse

- Eine Lösung mit einer Hilfsmethode:

```
def Linksklick(self, x, y):  
    """Linksklick ohne Treffer löscht die Auswahl"""  
    for moebel in self.__alleMoebel:  
        if self.ImMoebel(moebel,x,y):  
            self.Waehle(self.__alleMoebel.index(moebel))  
            return moebel  
  
    self.Waehle(-1)  
  
    return None
```

Einbau Mausereignisse

- Die Hilfsmethode verwendet die Methode *GetBox* aus der Klasse *GraphicsPath*, die ein *wx.Rect* - Tupel (*x,y,Breite,Tiefe*) zurück gibt:

```
def ImMoebel(self, moebel, x, y):  
    box=moebel.GibFigur().GetBox()  
    if x<box[0]: return False  
    if y<box[1]: return False  
    if x>box[0]+box[2]: return False  
    if y>box[1]+box[3]: return False  
    return True
```

Einbau Mausereignisse

- Eine einfachere Alternative zeigt die folgende Methode *Angeklickt* mit *GetBox* und *Contains* aus der Klasse *wx.Rect*:

```
def Angeklickt(self, moebel, x, y):
```

```
    pos=wx.Point2D(x,y)
```

```
    return moebel.GibFigur().GetBox().Contains(pos)
```

Einbau Mausereignisse

- Die Auswahl verwaltet die Klasse RaumplanerModell, teilt sie aber den Moebel-Objekten mit:

```
def Waehle(self, auswaehlen):  
    if auswaehlen: self.__ausgewaehlt=True  
    else: self.__ausgewaehlt=False  
    self.Update()
```

Einbau Mausereignisse

- Damit die Auswahl im Grafikfenster für den Anwender erkennbar ist, wird sie farblich gekennzeichnet: Ausgewählte Moebel-Objekte werden grau gekennzeichnet:

```
def GibFarbe(self):  
    """Get-Methode fuer die Farbe"""  
    if self.__ausgewaehlt: return "gray"  
    return self.__f
```