

14.2.2 Maus und Ereignisse

Unter Maus-Ereignissen sind jene Ereignisse (Events) aufzufassen, die sich auf Ereignisse sichtbarer Steuerelemente (Komponenten) im Zusammenhang mit der Maus beziehen. Damit können Sie das Mouse-Objekt nur im Zusammenhang mit einem Event benutzen, das sich auf solche Maus-Ereignisse bezieht:

```
Public Sub Form_Open() ' Open() hat nichts mit der Maus zu tun...
    Print Mouse.X
End ' Form_Open()
```

Die o.a. Anweisung, den Wert von *Mouse.X* in einer Konsole anzuzeigen, bringt daher die Fehlermeldung: **No mouse event data**. Verständlich, denn die Prozedur *Form_Open()* hat nichts mit der Maus zu tun.

Der folgende Quelltext-Ausschnitt dagegen ist korrekt:

```
Public Sub Form_MouseDown() ' MouseDown() ist ein Maus-Event
    Print Mouse.X
End ' Form_Open()
```

Auf diese Weise wird intern in Gambas-Komponenten und komplexen Steuerelementen wie zum Beispiel einer *DrawingArea* und anderen Containern ermittelt, auf welches (logische) Sub-Element des Containers geklickt wurde. *Mouse.X* wird dabei mit der absoluten Position des Containers zuzüglich der relativen Position des Sub-Elements zu seinem Container verglichen.

Die Übersicht zu den Ereignissen in einer Klasse, erfasst alle Ereignisse, die im Zusammenhang mit der Maus stehen – auch wenn das bei einigen Ereignissen aus der Bezeichnung nicht unmittelbar abzulesen ist:

- *MouseDown*, *MouseUp*, *MouseMove*
- *Enter*, *Leave*
- *MouseWheel*, *MouseDown*, *MouseUp*, *MouseMove*

Die Ereignisse der letzten Zeile werden hier nur aufgezählt und in anderen Kapiteln (→ Kapitel 14.2.3 *MouseWheel*) ausführlich beschrieben.

14.2.2.1 Beispiele 1-3

Diese Prozeduren zeigen Ihnen den Einsatz ausgewählter Ereignisse – *MouseWheel*, *MouseDown*, *MouseUp* sowie *MouseMove* – für 3 unterschiedliche Komponenten:

```
Public Sub dwgKS1_MouseWheel()
If Mouse.Delta = +1 Then ' Mausewheel nach vorn → Bild vergrößern (freie Festlegung)
    If fZoom < 300 Then
        fZoom = fZoom + 5
        KS_RP_G_Zeichnen() ' Zeichnen von Koordinatensystem, Raster und Graph der Funktion
    Endif
Else
    If fZoom > 20 Then
        fZoom = fZoom - 5
        KS_RP_G_Zeichnen()
    Endif
Endif ' Mouse.Delta = +1 ?
End ' dwgKS1_MouseWheel()
```

Für ausgewählte Komponenten existiert das Ereignis '*Control_MouseWheel()*'. Dieses Ereignis wird ausgelöst, wenn das Mausewheel bewegt wird oder das Mausewheel gedrückt wird, solange sich der Maus-Cursor über der Komponente befindet. Das Ereignis gibt den Wert *Mouse.Delta* vom Typ *Float* zurück. Der Wert ist +1, wenn das Mausewheel vorwärts gedreht wurde.

Als Alternative für *IF Mouse.Delta = +1 THEN ...* können Sie auch *IF Mouse.Forward THEN ...* einsetzen.

```

PUBLIC SUB Form_MouseDown()
    IF Mouse.Left THEN
        iDeltaX = Mouse.X
        iDeltaY = Mouse.Y
    ENDIF ' Mouse.Left ?
END ' Form_MouseDown()

Public Sub btnGetInformation_MouseUp()
    Timer1.Stop
End ' btnGetInformation_MouseUp()

PUBLIC SUB Form_MouseMove()
    IF Mouse.Left THEN
        FMain.Move(FMain.X - iDeltaX + Mouse.X, FMain.Y - iDeltaY + Mouse.Y)
    ENDIF ' Bei der Maus-Bewegung zusätzlich linke Maustaste gedrückt ?
END ' Form_MouseMove()

```

14.2.2.2 Beispiel 4 – Einsatz von Maus und speziellen Tasten

Da die Komponente *WebView1* (gb.qt4.webkit) bereits das Mausevent nutzt, um durch die Webseite vertikal zu scrollen, musste die Zoom-Funktion mit einem Schalter zu- oder abgeschaltet werden, um mit dem Ereignis *MouseWheel()* arbeiten zu können. Als Schalter wird als eine Möglichkeit die Steuerungstaste (Strg, Ctrl) benutzt. Das Ereignis *Komponente_MouseWheel()* bezieht sich auf die o.a. *WebView*-Komponente. Der Zoom-Faktor wird nur dann geändert, wenn gleichzeitig beim Drehen des Mauseventes die Taste CTRL gedrückt gehalten wird, weil sonst die originale Scroll-Funktion für den Inhalt der Webseite aktiv wäre:

```

[1] Public Sub WebView1_MouseWheel()
[2]     If Mouse.Control Then ' CTRL-Taste (zusätzlich) gedrückt?
[3]         Select Case Mouse.Delta
[4]             Case -1
[5]                 If WebView1.Zoom < 2.26 Then
[6]                     WebView1.Zoom = Round(WebView1.Zoom - Mouse.Delta / 4, -2)
[7]                 Endif
[8]             Case +1
[9]                 If WebView1.Zoom > 0.61
[10]                    WebView1.Zoom = Round(WebView1.Zoom - Mouse.Delta / 10, -2)
[11]                Endif
[12]            End Select
[13]        Endif ' Mouse.Control = True?
[14] End ' WebView1_MouseWheel()

```

```

[1] Public Sub WebView1_MouseDown()
[2]     If Mouse.Middle Then WebView1.Zoom = 1
[3] End ' WebView1_MouseDown()

```

Über die Eigenschaft *Mouse.Middle* können Sie feststellen, ob die mittlere Maustaste – das Mausevent – über einer Komponente gedrückt worden war.