

11.5.0.2 Fehlerbehandlung 3

Oft besteht die Notwendigkeit, für spezielle Aufgaben in Gambas Konsolen-Programme einzusetzen. Für diesen Zweck gibt es in Gambas die Klasse Process mit den Instruktionen EXEC sowie SHELL. Ausführliche Beschreibungen – auch zum Prozess-Fehler-Management – finden Sie im Online-Gambas-Buch auf <https://www.gambas-buch.de/doku.php?id=k21:start> im Kapitel `21.0 Prozess-Management`. Wenn Sie mit den Anweisungen SHELL oder EXEC einen eigenen Prozess erzeugt haben, dann lesen Sie Fehlerdaten nicht direkt von der Standard-Fehlerausgabe des Prozesses, sondern bekommen die Fehlerdaten bereits im Parameter des Event-Handlers

```
Process_Error(sErrorMessage As String)
```

wenn Fehlerdaten an der Standard-Fehlerausgabe des Prozesses vorliegen. Ob Sie die Fehlermeldung ungefiltert ausgeben oder noch eine Fehlerbehandlungsroutine vorsehen, um in Abhängigkeit vom angezeigten Fehler den Programmablauf zu beeinflussen, wird sicher durch die zu lösende Aufgabe bestimmt.

An einem einfachen GUI-Projekt für das Konsolen-Programm `ping` wird demonstriert, wie Sie mögliche Fehler mit dem Event-Handler ProcessEventName_Error(sError as String) abfangen und anzeigen können.

Der Quelltext-Auszug (Ping-Projekt) zeigt die relevanten Variablen und Prozeduren:

```
Private $hPing As Process
Private sProgramName As String = "ping" ' ping6 - für IPv6-Adressen
...
Public Sub btnPingAsProcessStart_Click()
    PingAsProcess()
End

Public Sub PingProcess_Error(sFehler As String)
    txaOutput.Insert("Process error!" & gb.NewLine & gb.NewLine)
    txaOutput.Insert(sFehler)
End

Public Sub PingProcess_Read()

    Dim sPuffer As String

    Read #hPing, sPuffer, Lof(hPing)
    txaOutput.Insert(sPuffer)

End

Private Sub PingAsProcess()

    Dim aCommand As New String[]

    If hPing Then hPing.Kill()
    aCommand = [sProgramName, "-c", Cint(txbPingCount.Text), txbURL.Text]
    $hPing = Exec aCommand For Read As "PingProcess"
    FMain.Text = "Program ping: PID = " & hPing.Id

End
```

Jetzt können Sie mit den Event-Handletern PingProcess_Read() und PingProcess_Error(...) die relevanten Daten dieses Prozesses auslesen. Das setzt voraus, dass Sie dem Prozess einen eigenen Namen gegeben haben. Der Name "PingProcess" ist eine gute Wahl. Fehlt dieser Name, dann werden durch den Prozess keine Ereignisse ausgelöst!

Exemplarisch werden Ihnen zum Ping-Projekt 4 Versuche vorgestellt, die mit unterschiedlichen Adressen und verschiedenen Anzahlen der Pings verschiedenartige Ergebnisse erzielten. Der erste und der letzte Versuch waren fehlerfrei – auch wenn der Mail-Server mx.freenet nicht antwortete. Mit einem fatalen Fehler endeten dagegen der zweite und dritte Versuch.

V1: Adresse: www.gambas-buch.de, Ping: 2 – fehlerfrei

```
PING www.gambas-buch.de (109.237.140.40) 56(84) bytes of data.
64 bytes from alfa3035.alfahosting-server.de (109.237.140.40): icmp_seq=1 ttl=59 time=39.9 ms
```

Kapitel 11.5.0.2 - Fehlerbehandlung 3

```
64 bytes from alfa3035.alfahosting-server.de (109.237.140.40): icmp_seq=2 ttl=59 time=39.4 ms
--- www.gambas-buch.de ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 39.420/39.680/39.940/0.260 ms
```

V2: Adresse: www.gambas+buch.de, Ping: 2

```
Process error!
ping: unknown host www.gambas+buch.de
```

V3: Adresse: www.gambas-buch.de, Ping: 0

```
Process error!
ping: bad number of packets to transmit.
```

V4: Adresse: mx.freenet.de, Ping: 2 – fehlerfrei

```
PING mx.freenet.de (195.4.92.212) 56(84) bytes of data.
--- mx.freenet.de ping statistics ---
2 packets transmitted, 0 received, 100% packet loss, time 1011ms
```