

Workshop: Director FOLGE 6



Textdarsteller

Es gibt mehrere Verfahren, um Text in ein Director-7-Projekt einzubinden. In dieser Folge unseres Workshops zeigen wir Ihnen, welche davon wann am sinnvollsten ist.

■ Director 7 bietet zwei Möglichkeiten zur Speicherung von Text: Felder und Textdarsteller. Felder bieten deutlich weniger Formatierungsfunktionen und sind wegen des fehlenden Antialiasings nur zur Speicherung von Texten in kleinen Schriftgrößen zu empfehlen, bei denen eine Glättung zugunsten der Lesbarkeit nicht notwendig ist. Textdarsteller unterstützen im Gegensatz dazu Antialiasing, Kerning und Formatop-

Director 7 bietet mit der Möglichkeit, Text auch zur Laufzeit zu importieren, einen Weg zu sich selbst aktualisierenden Programmen

tionen wie zum Beispiel Blocksatz, Tabstops oder einstellbaren Zeilenabstand. Daher eignen sie sich auch für anspruchsvollen Satz und für Schrift in größeren Punktgrößen.

Man könnte den Feldeditor von Director mit SimpleText und den integrierten Texteditor mit Word vergleichen, jedoch mit der Einschränkung, daß dem Texteditor eine ganze Reihe wichtiger Funktionen wie zum Beispiel eine Rechtschreibprüfung oder die Fähigkeit zur Speicherung von Druckformaten fehlen.

Selbstverständlich können auch andere Medienformate wie etwa Bitmap Text enthalten. Hier soll es allerdings um die Speicherung von Text gehen, der sich auch zur Laufzeit der Anwendung editieren läßt, weshalb diese alternativen Optionen in dieser Folge des Workshops unberücksichtigt bleiben.

Die Text-Engine in Director 7 ist komplett neu. Die meisten Änderungen betreffen allerdings die Textdarsteller. Für Darsteller vom Typ Feld gibt es nur eine wesentliche Veränderung: Die

Datenmenge eines Felds ist jetzt nicht mehr auf 32 Kilobyte Text begrenzt. Vielmehr stellt der zur Verfügung stehende Speicher die – theoretische – Grenze dar. Zumindest die momentan aktuellen Versionen 7.0 und 7.0.1 behandeln Textmengen im Megabyte-Bereich allerdings derart träge, daß sich dieses neue Feature kaum sinnvoll einsetzen läßt.

Alle nachfolgend vorgestellten Features betreffen ausschließlich Textdarsteller. Die Text-Engine von Director 7 beseitigt viele frühere Kritikpunkte. Textdarsteller sind jetzt auch zur Laufzeit der Anwendung editierbar und besitzen grundsätzlich vier Repräsentationen: eine Text-, eine HTML-, eine RTF- und eine Bitmap-Repräsentation. Text läßt sich im HTML-Format importieren, wobei sowohl Formatierungen als auch enthaltene Hyperlinks erhalten bleiben. Obwohl es in den vorliegenden Versionen Probleme beim Import umfangreicher, verschachtelter Tabellen gibt, eröffnet dieses Feature die Möglichkeit, Online-Daten ohne großen Aufwand in Director-Filme zu integrieren. Der umgekehrte Weg, also der Export der HTML-Repräsentation in eine externe HTML-Datei wird sicherlich seltener benötigt, ist aber prinzipiell genauso möglich.

Der folgende Auszug aus dem Nachrichtenfenster illustriert das Handling der vier Text-Repräsentationen (text, rtf, html und picture):

```
-- neuen Textdarsteller erzeugen
rtfText = new(#text)

put rtfText
-- (member 1 of castLib 1)
-- enthaltenen Text ändern
member(rtfText).text = "
Hello world!"
-- und ausgeben
```

```
put member(rtfText).text
-- "Hello world!"
-- Anzeige der RTF-Komponente
put member(rtfText).rtf
-- "{\rtf1\mac\def3 \
{\fonttbl{\f3\fswiss Geneva;}}\
{\colortbl\red0\green0\blue0;}\
{\stylesheet{\s0\fs24 Normal \
Text;}}\pard\fs24 Hello world!\
\par}"

-- Anzeige der HTML-Komponente
put member(rtfText).html
-- "<html>
<head>
<title>Untitled</title>
</head>
<body bgcolor="#FFFFFF">
Hello world!</body>
</html>
"

-- Anzeige der Bitmap-Komponente
put member(rtfText).picture
-- <Picture:b4c1028>
-- neuen Bitmapdarsteller erzeugen
testbitmap = new(#bitmap)

put testbitmap
-- (member 9 of castLib 1)
-- Zuweisung der Bitmap an neuen \
Bitmapdarsteller
member(testbitmap).picture = \
member(rtfText).picture
```

Die Möglichkeit, innerhalb von Director die aus HTML bekannten Hyperlinks in Textdarstellern zu definieren, vereinfacht die Realisierung von Hypertextfunktionen erheblich, da sich außer der Verknüpfung mit URLs auch beliebige andere Befehlsaufrufe platzieren lassen (Abbildung 1).

Äußerst praktisch ist auch die Option, ganze Schriften oder einzelne Sets von Zeichen in den Film integrieren zu können (Abbildung 2).

Wie kommt der Text in den Film?

Wenn der Text noch nicht in digitaler Form vorliegt, spricht eigentlich wenig dagegen, ihn in Director direkt einzugeben. Zu beachten ist dabei lediglich, daß Director keine Rechtschreibprüfung besitzt und eventuell ein Mehraufwand entsteht, wenn die Texte nach der Eingabe doch noch einmal in einem externen Dokument benötigt werden (so zum Beispiel zur Abnahme durch den Auftraggeber).

Liegen die Texte bereits als Datei vor, können Sie die Daten wahlweise importieren (was entsprechend die Konvertierung der Textdatei ins HTML-, RTF- oder Ascii-Format erfordert) oder unmittelbar aus dem Textprogramm einfügen. Letzteres funktioniert entweder wie gewohnt über die Zwischenablage oder per Drag-and-drop (ziehen Sie die markierten Textpassagen aus dem Textprogramm einfach an die gewünschte Position in einem Textfenster). Selbstverständlich ist das manuelle Einsetzen lediglich bei kleineren Textmengen praktikabel. Bei größeren

Texten macht es Sinn, über automatisierte, skriptgesteuerte Importmechanismen nachzudenken.

Mögliche Ansätze sind etwa:

- die Aufbereitung eines externen Textdokuments für jeden zu erzeugenden Textdarsteller
- die Verwendung von eindeutigen Trennzeichen – beispielsweise Sonderzeichen wie „#“ – innerhalb der Datei, die den Beginn eines neuen Textdarstellers markieren

Wenn Texte als Dokument geliefert werden, sollten Sie schon frühzeitig ein verbindliches und möglichst einfaches Dateiformat festlegen. Bei der Verwendung vom Autor oder Redakteur einzugebender Trennzeichen wie beispielsweise „#“ bestehen jede Menge denkbare Fehlerquellen. Diese sollten Sie in keinem Fall unterschätzen. Besser geeignet sind Tabellen oder spezielle Eingabemasken, da hier die Gefahr von Fehleingaben deutlich geringer und eine Kontrolle vor dem Import leichter durchzuführen ist.

Die neue Text-Engine macht es ausgesprochen einfach, im Ascii-, HTML-, oder RTF-Format vorliegende Daten zu

konvertieren. So ist es beispielsweise möglich, den formatierten Inhalt von Textdarstellern mit Hilfe des Xtras „FileIO“ in einem externen Dokument zu speichern, gegebenenfalls mit einem externen Editor bearbeiten zu lassen und schließlich wieder per FileIO zu importieren. Der Befehl „Datei/Importieren“ bietet Ihnen Textdateien im HTML- und RTF-Format an. Ascii-Textdateien erscheinen ebensowenig in der Liste wie Dateien im Word-Format; die Verfügbarkeit der Option „Mit externer Datei verknüpfen“ ist in diesem Fall irreführend. Unabhängig von der Einstellung importiert Director den Text immer vollständig – auf diesem Weg sind Bezüge zu externen Textdateien möglich.

Zum Textimport aus Lingo heraus stehen unterschiedliche Optionen zur Verfügung. Eine davon ist das Setzen der Darstellerproperty „the filename of member“.

```
member(1).filename = ~
"Macintosh_HD:etc:hilfe.rtf"
member(12).filename = ~
http://www.ggmc.de/index.html →
```

Abbildung 1: Hyperlinks im Textdarsteller

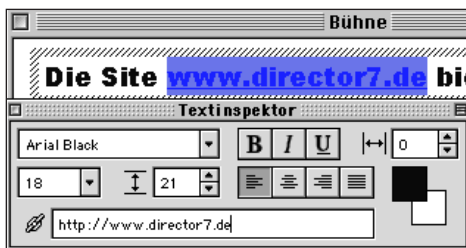
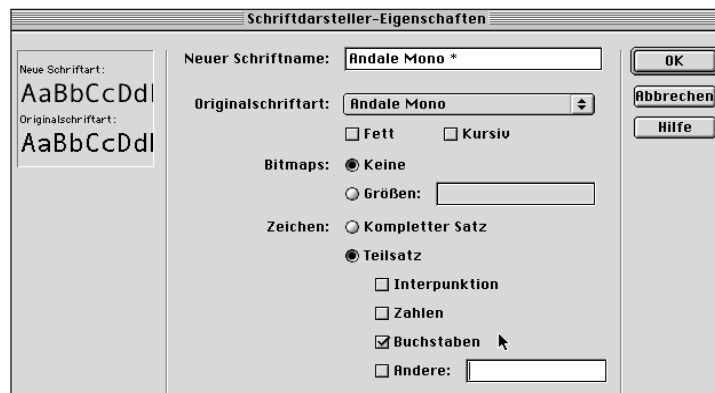


Abbildung 2: Eigenschaften von Schriftdarstellern



→ Auch der Befehl „importfileinto“ ist zur Laufzeit auf Textdokumente anwendbar.

```
importfileinto (member 1, ~
"C:\Eigene Dateien\swgetcd.htm")
importfileinto member 13, ~
"http://www.ggmc.de/index.html"
```

Der Netlingo-Befehl „getNetText“ funktioniert mit URLs und mit lokalen Pfadangaben, wie dieses Beispiel zeigt.

```
getNetText "Macintosh_HD:~
etc:hilfe.rtf"
newmem = new(#text)
put newmem
-- (member 5 of castLib 1)
member(newmem).rtf = ~
nettextresult()
```

Beachten Sie hier, daß die Funktion „nettextresult()“ die Datei als reinen Text behandelt und Sie bei der Zuweisung an den Darsteller angeben müssen, ob der Text als HTML- oder RTF-Codierung interpretiert werden soll.

```
getNetText "Macintosh_HD:~
etc:extevent.html"
newmem = new(#text)
put newmem
-- (member 7 of castLib 1)
member(newmem).html = ~
nettextresult()
```

Wenn Sie Textdaten aus dem Internet laden wollen, ist die zuletzt vorgestellte Variante zu empfehlen. Der große Vorteil gegenüber den anderen beiden Befehlen besteht hier darin, daß sich der Abschluß des Ladevorgangs mit der Funktion „netDone()“ überprüfen läßt.

Ein komplettes Behavior finden Sie in der Rubrik „Text“ der mit dem Director Studiobundle mitgelieferten Verhaltensbibliothek (Abbildung 3).

Beispiel: Die Rezeptkartei

Das folgende Beispiel realisiert eine einfache Kochrezeptkartei (Abbildungen 4a und 4b). Im oberen Fenster läßt sich jeweils ein Rezept auswählen, das

daraufhin in den Rubriken Zutaten, Zubereitung und Bemerkung erscheint. Zwischen diesen Rubriken können Sie per Mausklick auf den jeweiligen Rubriktitel wechseln.

Zunächst wollen wir uns dem grundlegenden Aufbau und der Auswahl der Rubriken widmen. Die Hintergrundgraphiken laden Sie entweder über den Dialog „Datei/Importieren“ oder ziehen sie direkt aus einem Finder-Fenster in die Besetzung. Da die Sprites in diesem Beispiel lediglich drei Bilder des Drehbuchs benötigen, wählen Sie zunächst „Einschlußdauer: 3 Bilder“ im Dialog des Befehls „Voreinstellungen/Sprite“ aus.

Plazieren Sie den Darsteller „Auswahl Hintergrund“ in Kanal 2 und den Darsteller „Rezept Hintergrund“ in Kanal 4 (Abbildung 5).

Markieren Sie nun Bild 1 des Kanals 3, und erzeugen Sie dann mit dem Textwerkzeug der Werkzeugpalette (Befehlstaste-7) einen Textdarsteller. Geben Sie den Namen des ersten Kochrezepts ein. Im Anschluß öffnen Sie den Dialog →

Abbildung 3: Text aus dem Web laden

Abbildung 4a und b: Ein Klick auf die Reiter zeigt den entsprechenden Darsteller

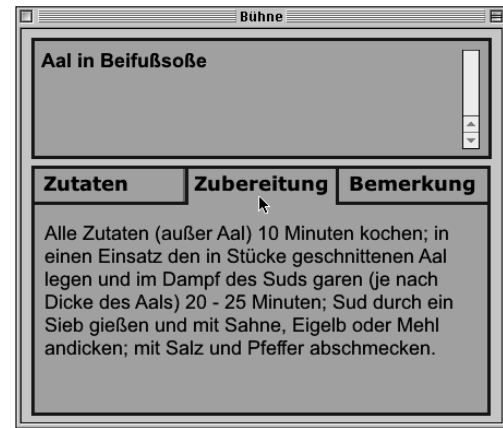
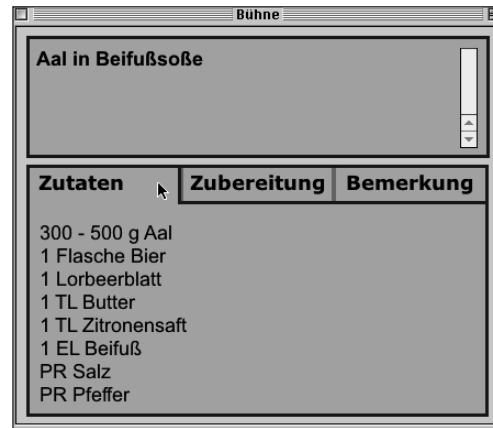
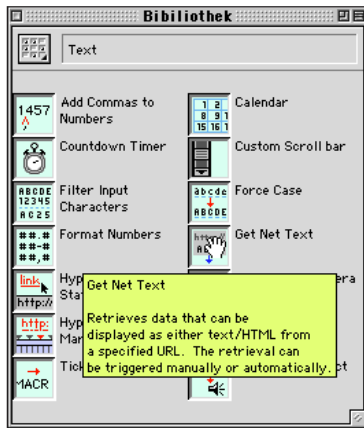


Abbildung 5: Darsteller plazieren

Abbildung 6: Eigenschaften des Textdarstellers festlegen

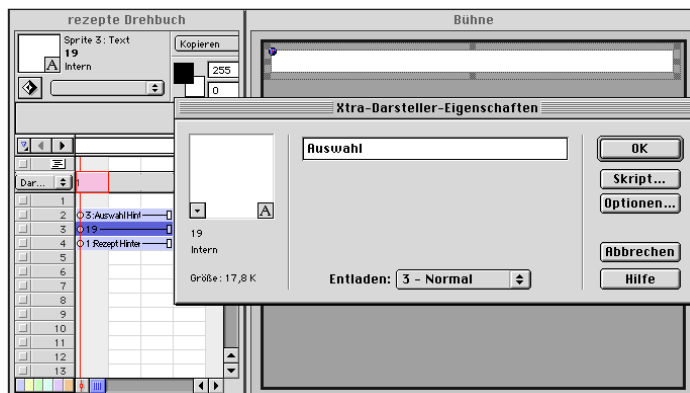
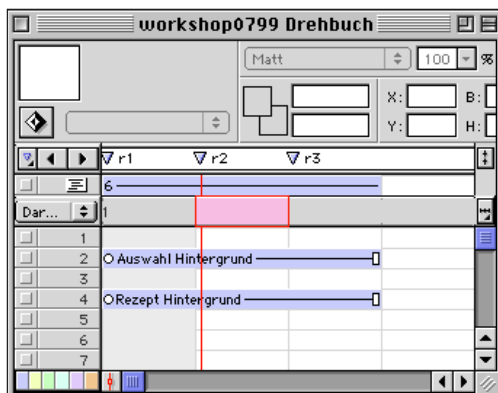


Abbildung 7: Rollbalken des Textdarstellers aktivieren



Abbildung 8: Keyframes des „Rezept-Tags“ anlegen

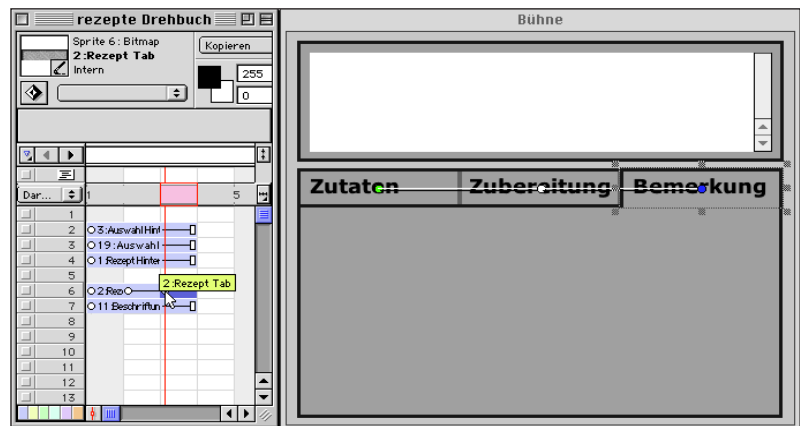


Abbildung 11: Loop-Skript eingeben

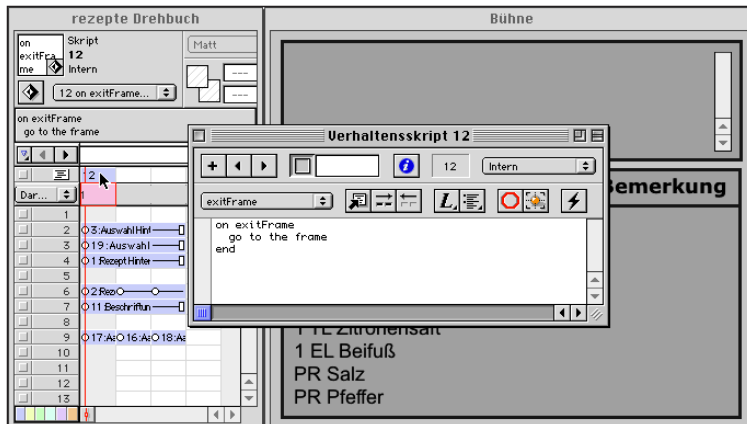


Abbildung 12: Unsichtbare QuickDraw-Sprites plazieren



► der Darstellereigenschaften (Befehlstaste-I) und nennen den Darsteller „Auswahl“ (Abbildung 6). Wechseln Sie nun in den Dialog „Optionen“, und wählen Sie hier den Rahmentyp „Rollen“ aus (Abbildung 7). Dies erzeugt am rechten Rand eine Textlaufleiste, mit der Sie später in weiteren Rezepten

Texte sind statt per Menübefehl auch per Drag-and-drop aus anderen Programmen in einen Textdarsteller importierbar

scrollen können. Nach dem Schließen des Darsteller-Info-Dialogs läßt sich der Rahmen des Textdarstellers mit der Maus skalieren und so an die Größe des Hintergrunds anpassen.

Plazieren Sie nun den Darsteller „Rezept Tab“ so im Kanal 6, daß er in Bild 1 links, in Bild 2 in der Mitte und in Bild 3 rechts liegt. Um dies zu erreichen, kön-

nen Sie den Sprite entweder bildweise bearbeiten (über den Befehl im Kontextmenü, per Controltaste-Mausklick) oder Keyframes einsetzen. Im nächsten Schritt wird die Beschriftung der Tabs benötigt. Erzeugen Sie einen weiteren Textdarsteller im Kanal 7 des Drehbuchs (Abbildung 8).

Zur Erzeugung der drei für das erste Rezept gebrauchten Textdarsteller können Sie das Textdokument „Aal in Beifußsoße“ vollständig in Director importieren (Befehlstaste-R) und dort in die Bestandteile „Zutaten“, „Zubereitung“ und „Bemerkung“ zerlegen. Alternativ dazu können Sie auch Word und ein neues Director-Textfenster nebeneinander positionieren und den Text per Drag-and-drop einfügen (Abbildung 9).

Geben Sie den neu erzeugten Textdarstellern eindeutige Bezeichnungen, zum Beispiel den des Rezepts mit dem Zusatz „zut“, „zub“ und „bem“, und plazieren Sie sie im Kanal 9. Markieren Sie jetzt sämtliche Text-Sprites (Befehls-

taste-Klick) im Drehbuch, und geben Sie ihnen den Farbeffekt „Matt“ (Abbildung 10). Zwischen den Bildern 1, 2 und 3 sollen während der Wiedergabe nur Wechsel stattfinden, wenn der Anwender auf die Rubriktitel klickt. Wir benötigen also ein Skript im Skriptkanal, das den Abspiekkopf im aktuellen Bild hält. Ein zweites Skript sorgt bei einem Mausklick auf die erste, zweite oder dritte Rubrik für den Wechsel in das Bild 1, 2 oder 3.

Doppelklicken Sie in die Zelle des Skriptkanals von Bild 1, und geben Sie den Befehl „go to the frame“ in das sich nun öffnende Skriptfenster ein (Abbildung 11). Schließen Sie das Skriptfenster, und kopieren Sie das Skript in die Bilder 2 und 3 des Skriptkanals, indem Sie die Zelle in Bild 1 bei gedrückter Wahltaste in das Bild 3 ziehen.

Drei unsichtbare QuickDraw-Sprites in den Kanälen 11 bis 13 übernehmen die Buttonfunktion. Markieren Sie zunächst Bild 1 in Kanal 11, und verwenden Sie anschließend das Recht-

Abbildung 9: Text aus Word per Drag-and-drop importieren

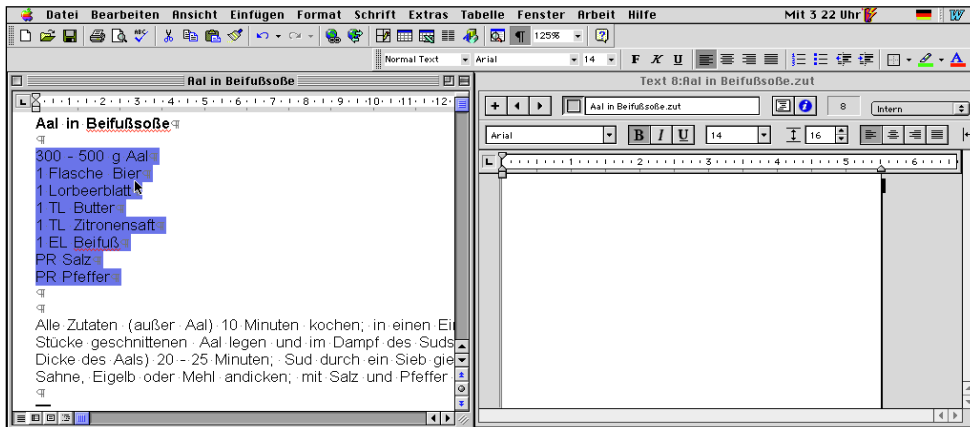


Abbildung 10: Textdarsteller mattieren

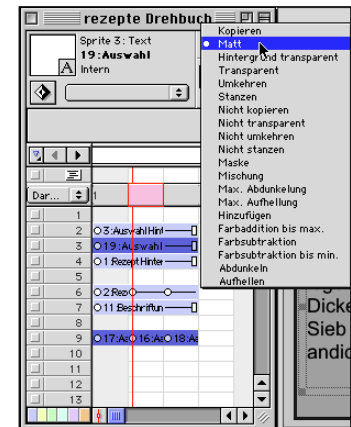


Abbildung 13: Verhalten

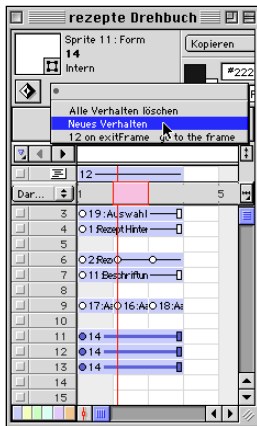
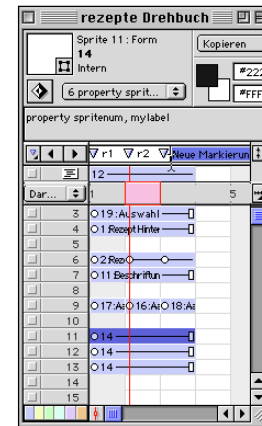


Abbildung 14: Verhaltensskript für Tab-Sprites



Abbildung 15: Sprungziel anlegen



eck-Werkzeug mit der Linienstärke 0, um einen unsichtbaren Rahmen über dem Tab „Zutaten“ aufzuziehen (Abbildung 12). Diesen Rahmen kopieren Sie im Drehbuch in die Kanäle 12 und 13 (Wahltaste-Ziehen) und verschieben ihn im Anschluß „Zubereitung“ beziehungsweise „Bemerkung“. Markieren Sie jetzt alle drei QuickDraw-Sprites, und weisen Sie ihnen mit dem Skript-Pop-up-Menü ein neues Verhalten zu (Abbildung 13).

Das abgebildete Skript (Abbildung 14) besteht aus zwei Prozeduren. Die Prozedur „beginSprite“ führt Director automatisch aus, sobald der Abspielkopf den Sprite erreicht. Hier wird die Propertyvariable „mylabel“ in Abhängigkeit von der aktuellen Kanalnummer („spritenum“) mit den Werten r1, r2 beziehungsweise r3 versehen. Der Inhalt der Property „mylabel“ wird von der durch das Loslassen der Maustaste ausgeführten Prozedur „mouseUp“ als Sprungziel verwendet. Beim Schließen

des Skriptfensters überprüft Director das Skript automatisch und macht den Anwender auf eventuelle Syntaxfehler aufmerksam.

Zum Schluß erhalten die Bilder 1, 2 und 3 die „r1“ bis „r3“ bezeichneten Sprungmarken im Drehbuch. Hierzu klicken Sie ganz einfach in die Labelzeile des Drehbuchs und geben den Sprungmarkentext ein (Abbildung 15). Die Verwendung der Sprungmarken hat gegenüber der Angabe fester Bildnummern den Vorteil, daß Sie bei eventuellen Verschiebungen der Sequenz (also Einfügen oder Löschen von Bildern) die Skripttexte nicht aktualisieren müssen.

Gerd Gillmaier ■

VORSCHAU

■ In der nächsten Folge erweitern wir dieses Beispiel unter anderem um die Auswahl weiterer Rezepte und um eine Online-Anbindung.