



Visio anpassen und programmieren

Das Handbuch

Für Visio 2013, 2016, 2019 und Visio in Microsoft 365

René Martin

Inhaltsverzeichnis

1. Vorwort

- 1.1. Warum Microsoft Office Visio?
- 1.2. Warum kein anderes Programm?
- 1.3. Eine kurze Geschichte von Visio
- 1.4. Die beiden Visio-Versionen
- 1.5. Über dieses Buch
- 1.6. Kontakt zum Autor

2. Visio anpassen

- 2.1. Eigene Füllmuster und Linienmuster erstellen
- 2.2. Eigene Shapes erstellen
 - 2.2.1. Die Standardelemente
 - 2.2.2. 1D- und 2D-Shapes
 - 2.2.3. Offene und geschlossene Shapes
 - 2.2.4. Gruppieren
 - 2.2.5. Mehrere Texte durch Gruppieren zu einem Shape zusammenfassen
 - 2.2.6. Vorgänge
 - 2.2.7. Verbindungspunkte einfügen
 - 2.2.8. Shape-Daten
 - 2.2.9. Shape-Datensätze
 - 2.2.10. Layer (Ebenen)
 - 2.2.11. Funktionen der Layer: Layereigenschaften
 - 2.2.12. Auswahl aller Shapes auf einem Layer
 - 2.2.13. Verhalten

- 2.2.14.QuickInfo und Hyperlink
- 2.2.15.Maßstab
- 2.2.16.Schutz vor Veränderungen
- 2.2.17.Copyright eintragen
- 2.3. Eigene Schablonen erstellen
 - 2.3.1. Die Quick-Shapes
 - 2.3.2. Die Schablone speichern
 - 2.3.3. Eigene Schablonen weitergeben
 - 2.3.4. Eigene Schablonen modifizieren
- 2.4. Eigene Vorlagen erstellen
 - 2.4.1. Seite einrichten
 - 2.4.2. Mehrere Seiten einrichten
 - 2.4.3. Schablonen
 - 2.4.4. Formatvorlagen
 - 2.4.5. Muster
 - 2.4.6. Design
 - 2.4.7. Layer
 - 2.4.8. Farbpalette
 - 2.4.9. Lineale und Gitter
 - 2.4.10.Shape-Layout
 - 2.4.11.Neue Registerkarten
 - 2.4.12.VBA-Code
 - 2.4.13.Eigene Vorlagen weitergeben
 - 2.4.14.Speichern, Schließen und Öffnen
 - 2.4.15.Dokumentschablone
 - 2.4.16.Dateieigenschaften
 - 2.4.17.Ein Beispiel

2.4.18. Ein weiteres Beispiel

2.4.19. Zusammenfassung

3. ShapeSheet-Programmierung

3.1. Das ShapeSheet-Fenster

3.2. Aufbau des ShapeSheets

3.3. Shape Transform

3.3.1. Punkte festsetzen

3.3.2. User-defined Cells (benutzerdefinierte Zellen)

3.3.3. Ein Beispiel für User-defined Cells

3.4. Scratch (Entwurf)

3.5. Wissenswertes über Funktionen

3.6. Geometry (Geometrie)

3.7. Actions (Aktionen)

3.8. Events (Ereignisse)

3.9. Fill Format und Line Format (Füll- und Linienformate)

3.10. Die Abschnitte für die Effekte

3.11. Character, Paragraph, Tabs und Text Block Format

3.12. Text Transform (Texttransformation)

3.13. TxtWidth, TxtHeight, TextWidth, TextHeight und TheText

3.14. Textfields (Textfelder)

3.15. EVALTEXT

3.16. SHAPETEXT

3.17. Die Zelle HideText

3.18. Layer Membership und Layers

- 3.19. Controls (Steuerpunkte, Steuerelemente, Kontrollpunkte)
- 3.20. Connection Points (Verbindungspunkte)
- 3.21. Zugriff auf andere Shapes
- 3.22. Group Properties (Gruppeneigenschaften)
- 3.23. Protection (Schutz)
- 3.24. Weitere Informationen der Shapes
 - 3.24.1. Miscellaneous (Sonstiges)
 - 3.24.2. Shape-Layout
 - 3.24.3. Alignment (Ausrichtung)
 - 3.24.4. Hyperlinks
 - 3.24.5. Annotation (Kommentar)
 - 3.24.6. Foreign Image Info und Image Properties
- 3.25. Shape Data (Shape-Daten)
- 3.26. ODBC-Informationen
- 3.27. 1D-Endpoint (1D-Endpunkte)
- 3.28. Glue Info (Informationen zum Kleben)
- 3.29. Das Zeichenblatt
 - 3.29.1. Page Properties
(Zeichenblatteigenschaften)
 - 3.29.2. Page Layout (Seitenlayout)
 - 3.29.3. Ruler & Grid (Lineale und Gitter)
 - 3.29.4. Print Properties (Druckeinrichtung)
 - 3.29.5. Theme Properties (Designs)
- 3.30. Einige Beispiele
 - 3.30.1. Beispiel 1: Gewinkeltes Förderband
 - 3.30.2. Beispiel 2: Einzelader

- 3.30.3. Beispiel 3: Schraube und Tisch
- 3.30.4. Beispiel 4: Elemente ausblenden
- 3.30.5. Beispiel 5: Mehrsprachige Zeichnungen
- 3.30.6. Beispiel 6: Verknüpfung zwischen Vordergrund und Hintergrund
- 3.30.7. Beispiel 7: Klebende Verbinder hervorheben
- 3.31. Die Datei: Document Properties (Dokumenteigenschaften) und Reviewer (Informationen)
- 3.32. Formatvorlagen
- 3.33. Smart Tags
- 3.34. Was fehlt
- 3.35. Zusammenfassung: Bezüge auf andere Zellen
- 3.36. Übersicht über die Abschnitte und Zellnamen im ShapeSheet
- 3.37. Übersicht der ShapeSheet-Funktionen
- 3.38. Zusammenfassung

4. Zugriff auf die Visio-Objekte per VBA

- 4.1. Der Makrorekorder
 - 4.1.1. Die Arbeitsweise des Makrorekorders
 - 4.1.2. Der zweifelhafte Code des Makrorekorders
 - 4.1.3. Fazit zum Makrorekorder
- 4.2. Die oberste Ebene: *Application*
- 4.3. Die Fenster (*Windows*)
- 4.4. Die Dateien (*Documents*)
 - 4.4.1. Die wichtigsten Methoden und Eigenschaften der Auflistung *Documents*

- 4.4.2. Die wichtigsten Methoden des *Document-Objekts*
- 4.4.3. Die wichtigsten Eigenschaften des *Document-Objekts*
- 4.4.4. Die Dialogfelder
- 4.5. Schablonen
- 4.6. Das *Window-Objekt*
- 4.7. Seiten (die *Pages*-Auflistung)
- 4.8. Eine Seite (das *Page-Objekt*)
- 4.9. Die Layer
- 4.10. Layer und das ShapeSheet
- 4.11. Zugriff auf Shapes
 - 4.11.1. Master-Shapes in einer Schablone
 - 4.11.2. Shapes auf dem Zeichenblatt
 - 4.11.3. OLE Objekte
 - 4.11.4. Shapes markieren
 - 4.11.5. Neue Shapes zeichnen
 - 4.11.6. Shapes beschriften
- 4.12. Zellen (das *Cell-Objekt*)
- 4.13. Unterordner visualisieren
- 4.14. Shapes verbinden
- 4.15. Einige wichtige und interessante Befehle
- 4.16. Makros starten
 - 4.16.1. Prozeduraufruf über Entwicklertools/Code/Makros
 - 4.16.2. Makros per Doppelklick auf das Shape starten
 - 4.16.3. Der Abschnitt *Events* im ShapeSheet

- 4.16.4. Die *Action-Zelle* im ShapeSheet
- 4.16.5. Steuerelemente
- 4.16.6. Symbole in der Registerkarte
- 4.16.7. Visio-Ereignisse
- 4.16.8. Weitere Ereignisse

5. Beispiele

- 5.1. London Underground
- 5.2. Ein Beispiel zum Export von Visio-Daten
- 5.3. Ein Nummerierungsassistent
- 5.4. Export von Access-Daten nach Visio
- 5.5. Ein Ablaufdiagramm zur Fertigungssteuerung
 - 5.5.1. Die Shapes
 - 5.5.2. Die Excel-Liste
 - 5.5.3. Die Datei an die Zeichnung binden
- 5.6. Ein Assistent für Organigramme
 - 5.6.1. Shapes und ShapeSheet
 - 5.6.2. Die Vorlage
 - 5.6.3. Der VBA-Code
 - 5.6.4. Die Dialoge
- 5.7. Ein Reportingtool in Visio
 - 5.7.1. Die Eingabemaske
 - 5.7.2. Der Bericht wird erzeugt
- 5.8. Rackdaten einsammeln
- 5.9. Ein Assistent zum Verschieben

6. Bewegung in Visio

- 6.1. Autorennen
- 6.2. Der Viertaktmotor

- 6.3. Planetenumlaufbahn
- 6.4. Papierherstellung
- 6.5. Fazit

7. Von VBA zu VSTO

- 7.1. Vorüberlegung
- 7.2. Das Projekt
- 7.3. Von VBA nach VSTO
- 7.4. VSTO
- 7.5. Das Menüband
- 7.6. Zugriff auf Visio
- 7.7. Klassenprogrammierung
- 7.8. Interne Konstanten
- 7.9. Das Kontextmenü
- 7.10. Die Ereignisse
- 7.11. Beenden
- 7.12. Die Dialog
- 7.13. Das Zip-Programm
- 7.14. Arrays und Dictionary
- 7.15. Versionierung und Assembly
- 7.16. Download aus dem Internet
- 7.17. Der Installer
- 7.18. Zusammenfassung:

8. Index

9. Ein Wort zu mir

1 Vorwort

1.1. Warum Microsoft Office Visio?

Vor vielen Jahren habe ich Visio das erste Mal gesehen - es war die Version 3.0. Damals habe ich wenig Unterschiede zu anderen Grafik- oder Präsentationsprogrammen feststellen können. Aber je länger ich Visio unterrichtete und je länger ich mich damit beschäftigt habe, desto begeisterter bin ich. Die Anwendungsmöglichkeiten sind schier unbegrenzt.

Es gibt zwei Gründe, warum ich dieses Programm so sehr liebe. Zum einen ist es einfach zu bedienen. Schon früh - lange bevor es von Microsoft gekauft wurde - wurde seine Benutzeroberfläche der von anderen Microsoft-Produkten der Office-Palette angepasst. So findet sich jemand, der schon mit den anderen Bestandteilen von Microsoft Office (Word, Excel oder PowerPoint) gearbeitet hat, leicht hinein und kann schnell Geschäftsdiagramme erstellen.

Stellen Sie sich einen gezeichneten Raumplan vor. Stellen Sie sich vor, Sie müssten alle Linien einzeln per Hand ziehen. Die Arbeit wäre immens groß. Oder stellen Sie sich einen LAN-Schrank vor. Um alle Patch-Stecker, alle Module, Bretter und sonstigen Elemente per Hand zu zeichnen, würde man Tage benötigen. Visio bietet für solche Probleme eine Reihe vorgefertigter Lösungen, deren Elemente lediglich auf die Seite gezogen und dort richtig verbunden werden. Schon ist das physikalische oder das logische Objekt fertig. Ebenso stellen viele Firmen Shapes kostenlos im Download zur Verfügung.

Der andere Vorteil liegt in der Tiefe und in der Möglichkeit, Visio für eigene Bedürfnisse anzupassen. Vielleicht etwas versteckt und für den Benutzer nicht sofort sichtbar, liegen eine Reihe an Funktionen verborgen. Nicht nur, dass sich jeder Anwender eigene Shapes (Zeichenobjekte) erstellen

und abspeichern kann, jeder kann diese Shapes mit einer »Logik« versehen. Was heißt das?

Stellen Sie sich ein Quadrat vor. Zieht der Benutzer an einer der vier Seiten, dann wird das Quadrat zu einem Rechteck verzerrt. Nicht so bei Visio. Mit zwei Klicks können Sie die Eigenschaften des Quadrats so verändern, dass es immer ein Quadrat bleibt. Oder dass seine Größe gar nicht mehr verändert werden kann. Oder seine Lage. Das ist eine der offensichtlichen Stärken. Viel wichtiger dagegen ist jedoch, dass mit Symbolen auf einer Zeichnung bestimmte Informationen abgespeichert werden können.

Stellen Sie sich als Zeichnung den Grundriss eines Bürogebäudes vor. In ihm sind nicht nur die elektrischen Installationen verzeichnet, sondern auch das Mobiliar und die dort aufgestellten Computer. An jedes dieser Objekte auf dem Zeichenblatt werden Informationen gebunden, wie beispielsweise Preis, Artikelnummer, Bezeichnung, Benutzer und so weiter. Nun kann mit einem Mausklick eine Inventarliste erstellt werden, in der nicht nur die Informationen angezeigt werden, sondern auch die Summe der Anschaffungskosten berechnet wird. Diese Liste kann ebenso direkt in Excel erstellt werden, sodass die Daten sofort weitergegeben werden können.

Noch ein Beispiel: Stellen Sie sich eine Datenbank vor, in der die Mitarbeiter einer großen Firma aufgelistet sind. Per Knopfdruck soll in Visio ein Organisationsdiagramm erstellt werden, in dem jeder Mitarbeiter ein eigenes Kästchen besitzt, Hierarchien durch Linien gekennzeichnet werden und für jede Abteilung ein eigenes Zeichenblatt angelegt wird. Zwar stellt Visio für solche Standardaufgaben Assistenten zur Verfügung; wollen Sie jedoch eigene Funktionen hinterlegen, so können Sie Lösungen für solche Aufgaben selbst programmieren. Nicht nur, dass Sie »intelligente« Shapes erstellen und abspeichern können, Sie haben die Möglichkeit Informationen einer Zeichnung in

Tabellen oder Datenbanken zu speichern. Oder Sie können umgekehrt Visio von außen steuern, indem Sie gespeicherte Informationen nach Visio übertragen, wo daraus ein Diagramm erzeugt wird. Dies sind die Themen des vorliegenden Buchs.

1.2. Warum kein anderes Programm?

Um es deutlich zu sagen: Visio ist kein Zeichenprogramm, Visio ist kein Grafikprogramm und Visio ist kein Präsentationsprogramm! Vergleicht man Visio mit anderen Softwareprodukten, dann wird man schnell enttäuscht sein. Visio will dagegen etwas Anderes: Es geht um das schnelle Erzeugen eines Geschäftsdiagramms mittels vorgegebener Symbole. Diese werden auf das Zeichenblatt gezogen, dort angeordnet, formatiert, beschriftet und möglicherweise mit Linien verbunden.

Visio ist übrigens auch kein CAD-Programm. Viele technische Zeichner, die seit Jahren mit einem CAD-Programm arbeiten, vermissen einige Funktionen in Visio. Zwar besitzt Visio einige Werkzeuge, die sicherlich aus dem CAD-Bereich übernommen sind. Dennoch: Betrachten Sie den Preis! Visio kostet etwas nur 10% von dem Preis eines CAD-Programms. Und: Visio kann leicht angepasst werden.

1.3. Eine kurze Geschichte von Visio

Axon Corp., wie Visio Corp. zuerst genannt wurde, wurde 1990 von zwei Mitbegründern der Firma Aldus Corp. ins Leben gerufen. Aldus ist vor allem durch seinen PageMaker bekannt geworden; später fusionierte die Firma mit Adobe. Als Visio 1.0 1992 vorgestellt wurde, gewann es schnell wegen der einfachen Bedienbarkeit an Beliebtheit, die Firma benannt sich kurz vorher in ShapeWare Corp. um.

1993 stellte ShapeWare optionale Schablonen mit Shapes zur Verfügung, die »Visio Shapes« genannt wurden. Im

August 1995 wurde mit Visio 4.0 eines der ersten Programme für das neue Windows 95 vorgestellt.

Nach den Versionen 1.0, 2.0, 3.0 und 4.0 begann Visio verschiedene Module zu entwickeln. Visio Technical wurde als CAD-begleitende Software 1994 am Markt angeboten – damals noch als Visio 4.1. 1995 wurde ShapeWare Corp. in Visio Corp. umbenannt und ging unter diesem Namen an die Börse. Als die Schablonen 1996 in »Visio Solutions Library« umbenannt wurden, wurden sogar Add-Ons zur Verfügung gestellt.

Visio Professional wurde 1996 entwickelt – damals als Version 4.5. 1998 folgte Intelli-CAD, ein zu AutoCAD kompatibles CAD-Programm, das allerdings andere Supports benötigte als Visio und deshalb bald wieder aus dem Paket genommen wurde.

1999 wurde Visio von der Firma Microsoft Corp. gekauft. Damals gab es Visio als Version 2000 (eigentlich: Visio 6.0) mit den vier Paketen: Visio Standard, Professional, Technical und Enterprise erhältlich. Visio, SmartShapes und Visio Solutions Library sind Warenzeichen oder eingetragene Warenzeichen der Visio Corp. in den USA und/oder anderen Ländern. Während Microsoft an dieser Version noch wenig änderte, so zeigte sich die Gestaltungselemente von Microsoft deutlich in der Version 2002 oder Visio 10.0.

Weitere Informationen finden Sie auf der Microsoft-Website unter: <http://www.windows-marketplace.com/>. Dort wählen Sie Deutschland und anschließend Office/Office Desktop Programme. Nun gelangen Sie zu Office Visio Standard und Office Visio Professional.

1.4. Die beiden Visio-Versionen

Visio lag in der Version 5.0 und 2000 in drei verschiedenen Editionen vor: Standard, Technical und Professional. Im Jahr 2002 wurde es nun auf zwei Pakete minimiert: Standard und

Professional. Ebenso ist heute Visio in den beiden Paketen Standard und Professional erhältlich. Das Programm ist in beiden Versionen das gleiche - die Unterschiede liegen in den Shapes, Schablonen, Vorlagen und Assistenten.

Der gesamte Umfang von Visio Standard findet sich ebenso in Visio Professional. Umgekehrt enthält Visio Professional mehr Funktionalitäten als Visio Standard: Die Registerkarte Daten fehlt in Visio Standard. Und mit ihr natürlich die beiden Assistenten *Datengrafiken* und *Shapes mit Daten verknüpfen*. Die folgende Tabelle listet die Unterschiede auf und versucht zu verdeutlichen, für welchen Benutzer welches Paket geeignet ist:

Standard	Professional
Benutzer	Benutzer
Mitarbeiter von Firmen:	Datenbank- und Netzwerk-Administration
Leitung	Datenbank-Programmierer
Finanzen	Netzwerk-Spezialisten
Planung, Projektmanagement	Software-Programmierer
Controlling	Elektrotechniker, Ingenieure
Verkauf und Marketing	Architekten, Innenarchitekten
	Maschinenbauer (Hydraulik, Pneumatik)
	Verfahrenstechniker
Zweck	Zweck
Geschäftsdiagramme erstellen	Informationssysteme und -prozesse darstellen
	Zweidimensionale technische Zeichnungen erstellen
Daten	Daten
Verwenden der Shape-Daten	Verwenden der Shape-Daten
Erstellen neuer Felder für Shape-Daten	Erstellen neuer Felder für Shape-Daten
Erstellen von Berichten	Erstellen von Berichten
	Integrieren von Daten aus Excel, Access oder dem Microsoft SQL-Server mit einem Diagramm
	Erstellen von Datengrafiken
	Pivotdiagramme

Vorlagen	Vorlagen
Allgemein	Allgemein
Flussdiagramm	Flussdiagramm
Geschäft (Organigramm, Brainstorming, Pivot)	Geschäft (Organigramm, Brainstorming, Pivot)
Netzwerk	Netzwerk – jedoch mehr Vorlagen als in Visio Standard
Pläne und Grundrisse	Pläne und Grundrisse – jedoch mehr Vorlagen als in Visio Standard
Terminplan (Zeitachse, Gantt, PERT, Kalender)	Terminplan (Zeitachse, Gantt, PERT, Kalender)
	Konstruktion (Elektrotechnik, Pneumatik/Hydraulik, Maschinenbau, Verfahrenstechnik, P&ID)
	Software und Datenbank

1.5. Über dieses Buch

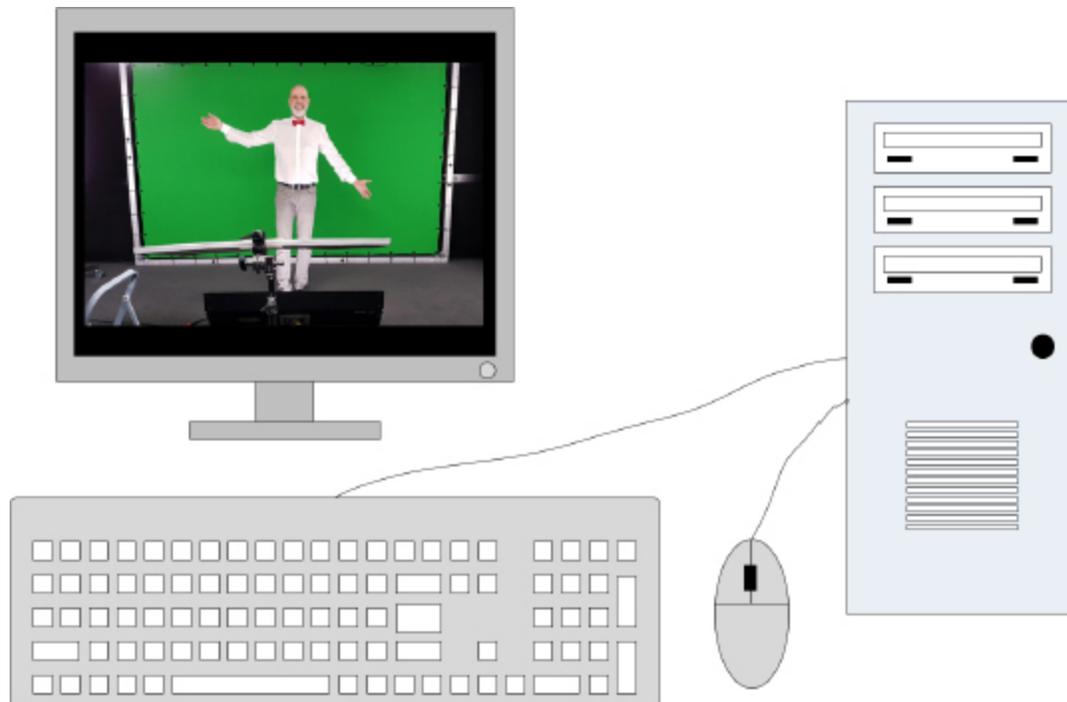
Ich habe eine Weile überlegt, ob ich eine bestimmte Versionsnummer in den Titel aufnehmen soll. Ich habe mich schließlich dagegen entschieden, 2019 oder Office 365 im Titel zu nennen. Der Grund ist denkbar einfach. Die Unterschiede zwischen Visio 2019 und den beiden Vorgängerversionen 2013 und 2016 liegen sich im Wesentlichen in der Optimierung fürs Tablett und in Visio online, das hier nicht behandelt werden soll. Weitere Unterschiede zwischen den einzelnen Versionen sind für den Anwender marginal. Für den Entwickler gibt es keine Unterschiede – nicht im Erstellen eigener Shapes, Schablonen und Vorlagen, auch nicht im ShapeSheet oder in VBA. Deshalb nur trägt das vorliegende Buch Visio im Titel – es ist für alle Versionen – 2013, 2016, 2019 und Office 365 geeignet.

Die folgenden Schwerpunkte bilden die zentralen Themen des vorliegenden Buchs.

[Kapitel 2](#) beschreibt wie man eigene Shapes erstellt. Der Umgang mit Linien, Ellipsen, Rechtecken und Freihandformen wird erläutert. Aber auch, wie man komplexe Figuren mit Hilfe der Vorgänge erstellen kann. Welche Besonderheiten Visio beim Gruppieren aufweist. Welche Verhaltenseigenschaften einem Shape zugewiesen werden können, welche Schutzmechanismen zur Verfügung stehen, was Layer in Visio bedeutet und wie man Datenfelder für das Shape definieren kann.

In [Kapitel 3](#) wird das ShapeSheet beschrieben. Einerseits spiegeln sich dort sämtliche Einstellungen wider, die in Visio vorgenommen werden. Andererseits können dort Einstellungen vorgenommen werden, die man nicht in Visio aktivieren kann: Steuerelemente, komplexe Schutzmechanismen, das Kontextmenü des Shapes, dynamische Shapes, ...

Microsoft liefert Visio, wie fast alle Anwendungsprogramme, mit der Programmiersprache VBA aus. Da der Sprachkern und die Dialoge von VBA die gleichen sind wie in Excel, Word, Access, ... soll es hier nicht wiederholt werden. Im Zentrum des Kapitels stehen die Objekte von Visio - also konkret: wie kann man Shapes per Programmierung erzeugen und verändern oder wie kann man auf eine bestehende Zeichnung zugreifen und Informationen auslesen, um sie dann beispielsweise in eine Datenbank zu schreiben. Dies finden Sie ausführlich und an vielen kleinen nachvollziehbaren Beispielen erläutert in [Kapitel 4](#) bis [6](#) beschrieben.



1.6. Kontakt zum Autor

Da ich Visio seit einigen Jahren unterrichte und auch Lösungen in Visio erstelle, bin ich sehr an Anregungen, Kritik und Meinungen interessiert. Wenn Sie Kontakt zu mir aufnehmen möchten oder mehr Informationen über mich haben möchten, dann finden Sie mich unter folgenden Internetadressen:

<http://www.compurem.de> <http://www.visio-schulungen.de>

<http://www.visio-training.de>

Dort finden Sie auch alle Beispiele dieses Buches zum Download unter

Beispiele (Download) / Visio.

Besuchen Sie mich auch auf

<http://www.facebook.com/pages/Der-Visio-Guru/319542344732487>

Nun bleibt mir nur noch, Ihnen viel Spaß beim Lesen des Buchs zu wünschen.

René Martin im April 2021

2 Visio anpassen

Visio liefert mehrere Tausend Shapes für sehr unterschiedliche Anwendungsbereiche. Dennoch kann es vorkommen, dass unter den vorhandenen Shapes sich nicht das richtige für Ihre Bedürfnisse befindet. Beispielsweise, weil in Ihrer Firma eigene Symbole für bestimmte abstrakte Vorgänge oder für konkrete, technische Elemente festgelegt wurden. Dann müssen Sie neue Shapes anlegen, in Schablonen speichern und möglicherweise Vorlagen dafür definieren. Wie dies funktioniert, wird in diesem Kapitel gezeigt.

2.1. Eigene Füllmuster und Linienmuster erstellen

Zwar stellt Visio 24 verschiedene Füllmuster, 23 Linienarten und 45 Pfeilspitzen zur Verfügung. Aber hausinterne Normen verlangen manchmal nach eigenen Mustern oder Linienarten. Oder Sie möchten bestimmte Pfeilspitzen verwenden, die nicht in Visio vorhanden sind.

So erstellen Sie ein eigenes Füllmuster:

- Öffnen Sie die Registerkarte *Entwicklertools*, indem Sie in *Datei/Optionen/Menüband anpassen* das Kontrollkästchen *Entwicklertools* einschalten.
- Öffnen Sie den Zeichnungsexplorer über die Registerkarte *Entwicklertools/Einblenden/Ausblenden/Zeichnungsexplorer*. Sollte die Registerkarte *Entwicklertools* nicht vorhanden sein, dann öffnen Sie sie über den Befehl *Datei/Optionen/ Erweitert/Im Entwicklermodus ausführen*.
- Wählen Sie im Kontextmenü des Ordners Füllmuster des Zeichnungsexplorers den Befehl *Neues Muster*, wie Sie in [Abbildung 2.1](#) sehen. Damit können Füll- und Linienmuster oder neue Linienenden erzeugt werden.

Wird ein neues Füllmuster angelegt, so wird der Name des Musters eingegeben und festgelegt, ob das Muster gekachelt, zentriert oder gezoomt wird. Bei den ersten beiden Einstellungen steht darüber hinaus die Option *Skaliert* zur Verfügung. Das Dialogfeld sehen Sie in [Abbildung 2.1](#).

Abbildung 2.1: Ein neues (Füll-)Muster wird erzeugt.

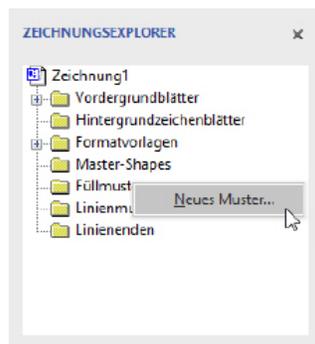
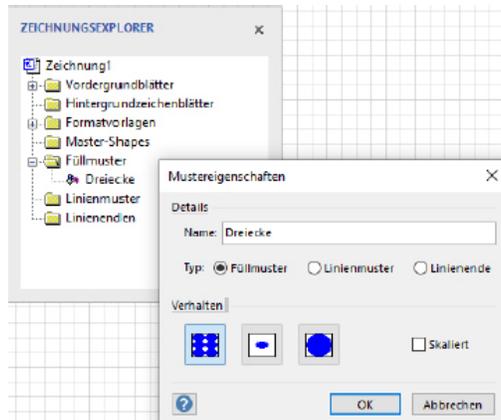


Abbildung 2.2: Die Eigenschaften des neuen Musters



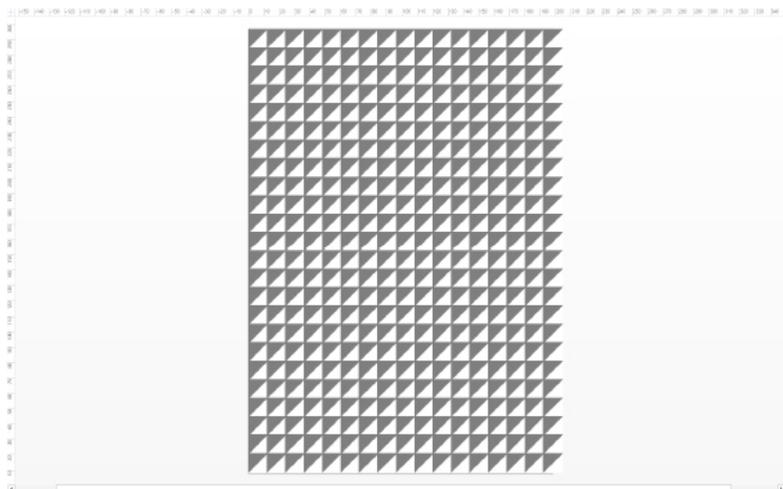
- Damit hat ein neues Muster einen neuen Namen. Nun muss das Muster noch erstellt werden. Öffnen Sie dazu den entsprechenden Knoten im Zeichnungsexplorer und doppelklicken Sie auf den Namen des Musters oder klicken Sie das Muster mit der rechten Maustaste an und wählen Sie im Kontextmenüs den Befehl *Muster-Shape bearbeiten*. Hierdurch wird das Zeichenblatt geöffnet, auf dem Sie das neue Muster erstellen und bearbeiten.

Hinweis

Die einzelnen Elemente, aus denen das Muster besteht, müssen dabei als Gruppe vorliegen.

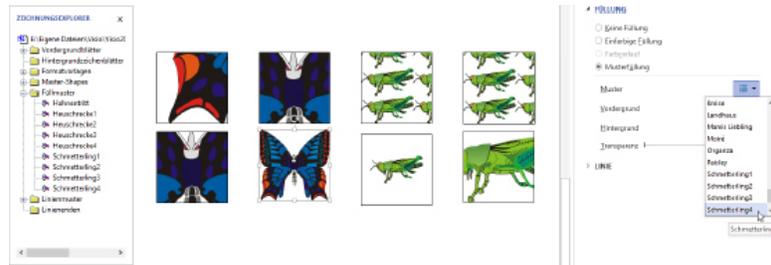
- Schließen Sie das Musterfenster (Sie sehen eines in **Abbildung 2.3**) und bestätigen Sie die Frage nach der Aktualisierung mit *Ja*.

Abbildung 2.3: Das neue Muster wird erstellt.



Das Verhalten des Musters wirkt sich bei der Größenänderung des Shapes aus. Normalerweise entspricht die Größe des Shapes nicht der Zeichenblattgröße. Die drei Verhaltensoptionen gekachelt, zentriert oder gezoomt legen dabei fest, wie sich das Muster innerhalb des Shapes verhält. Die folgende Beispielzeichnung **Abbildung 2.4** zeigt die Verwendung zweier Muster, die sich jeweils unterschiedlich verhalten.

Abbildung 2.4: Gekachelt (skaliert und nicht skaliert), zentriert und gezoomt



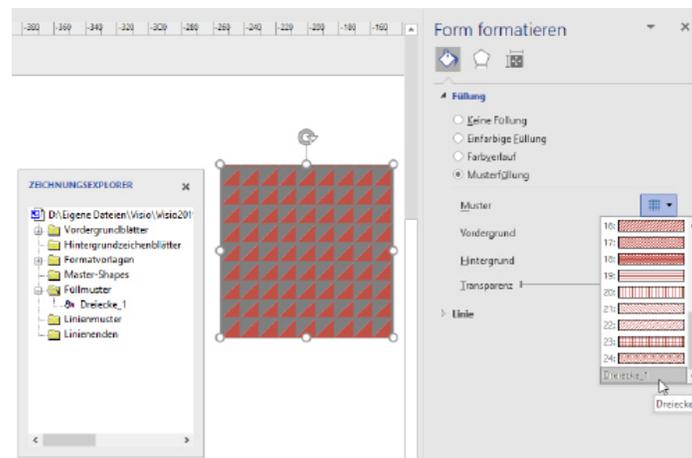
Es existieren insgesamt drei verschiedene Mustertypen: Füllmuster, Linienmuster und Linienenden.

Hinweis

Beachten Sie, dass Linienmuster und Linienenden immer horizontal zu zeichnen sind.

- Nun erscheinen alle selbst erstellten Muster mit dem von Ihnen gewählten Namen in der Liste der Muster im Dialogfeld *Füllbereich* beziehungsweise *Linie* (Registerkarte *Start*). Sie können nun verwendet werden (**Abbildung 2.5**).

Abbildung 2.5: Das neue Muster wird benutzt.

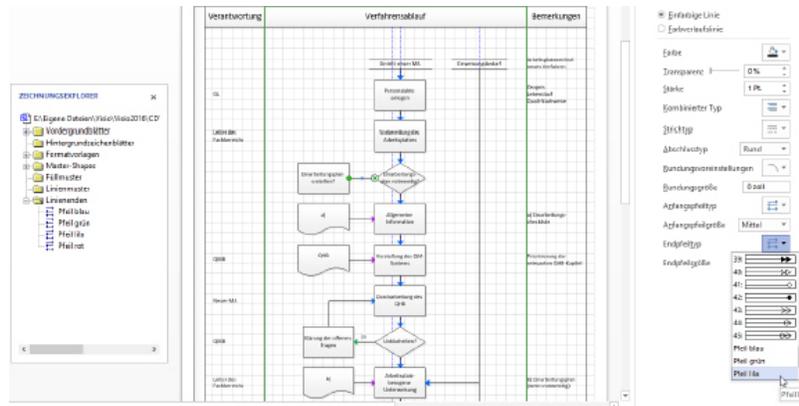


Hinweis

Die neuen Muster stehen stets am unteren Ende der Liste. Wird ein Muster verwendet, das im Fenster *Zeichnungsexplorer* nachträglich geändert wird, so wird das Muster bei den Shapes, die es verwenden, aktualisiert.

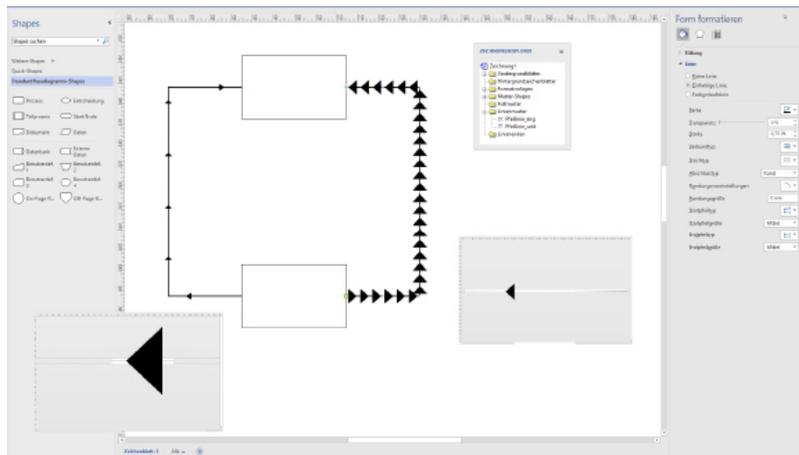
Analog gilt dies für das Linienmuster (**Abbildung 2.6**) und die Linienenden. Textmuster können nicht erstellt werden, da über die Gruppe *Start/Schriftart* keine benutzerdefinierten Schriften aufgerufen werden können.

Abbildung 2.6: Benutzerdefinierte Pfeilspitzen



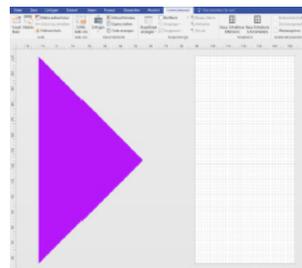
Wichtig ist hierbei das Verhältnis der Shapes in der Muster-Shape-Zeichnung zum Zeichenblatt. Möglicherweise müssen Sie das Zeichenblatt vergrößern. Verwenden Sie die **[Strg]**-Taste, um das Zeichenblatt zu vergrößern. In **Abbildung 2.7** sehen Sie zwei unterschiedliche Zeichenblattgrößen mit den gleichen Dreiecken in den Muster-Shapes-Vorlagen und ihre daraus resultierenden Linien.

Abbildung 2.7: Zwei Linienmuster mit unterschiedlichen Zeichenblattgrößen der Muster-Shapes



Mit dieser Technik können Sie bunte Pfeilspitzen erzeugen. Oder auch Pfeilspitzen, die nicht ganz am Ende der Linie sitzen, sondern etwas weiter davor.

Abbildung 2.8: Eigene Pfeilspitzen



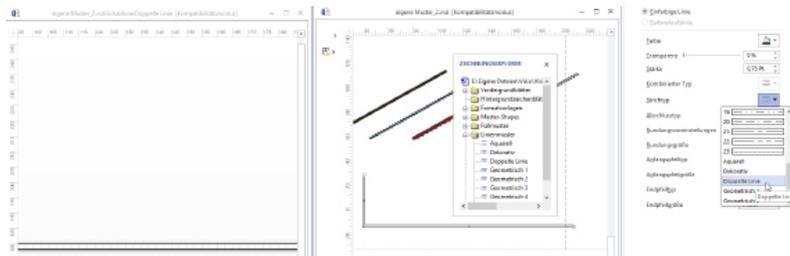
Hinweis

Alle selbst erzeugten Formate, die sich nun im Zeichnungsexplorer befinden, werden mit der Datei gespeichert. Dann stehen in neuen Zeichnungen im Dialogfeld

Füllbereich die selbst definierten Füllmuster und im Dialogfeld *Linie* die selbst erzeugten Linienenden und Linienmuster zur Verfügung.

Nun können alle Shapes, die sich in dieser Datei befinden, mit den Mustern formatiert werden. Selbstverständlich sind selbst definierte Muster nicht in einer Schablone sichtbar. Sie können lediglich sichtbar gemacht werden, indem der Zeichnungsexplorer geöffnet wird. Auch beim Formatieren können Sie die selbstdefinierten Muster erst dann sehen, wenn Sie am unteren Ende der Liste nachschauen.

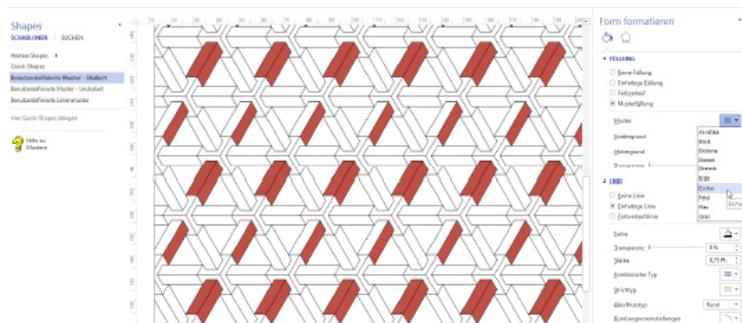
Abbildung 2.9: Mit diesem Wissen können Sie doppelte Linien erzeugen.



Übrigens können benutzerdefinierte Muster auch als Formatvorlagen abgespeichert werden. Umgekehrt können Formatvorlagen aber nur die Gestaltungselemente verwenden, die von Visio oder über andere Muster vorgegeben sind. Das bedeutet, dass *Muster* völlig neue Muster generiert, während *Formatvorlagen* nur auf vorhandene Muster zurückgreifen. Erstaunlicherweise können selbstdefinierte Muster nicht an Designs gebunden werden. Aber Sie könnten Designs und Muster mischen.

Wie dies funktioniert kann in den drei Schablonen *Benutzerdefinierte Muster - unskaliert*, *Benutzerdefinierte Muster - skaliert* und *Benutzerdefinierte Linienmuster* in der Kategorie *Visio-Extras* eingesehen werden. Wird die Schablone geöffnet, stehen Ihnen auf der Zeichnung weitere Füllmuster oder Linienmuster zur Verfügung, mit denen gearbeitet werden kann. Welches Muster auf der Zeichnung verwendet wurde, lässt sich im Zeichnungsexplorer einsehen, wie **Abbildung 2.10** zeigt.

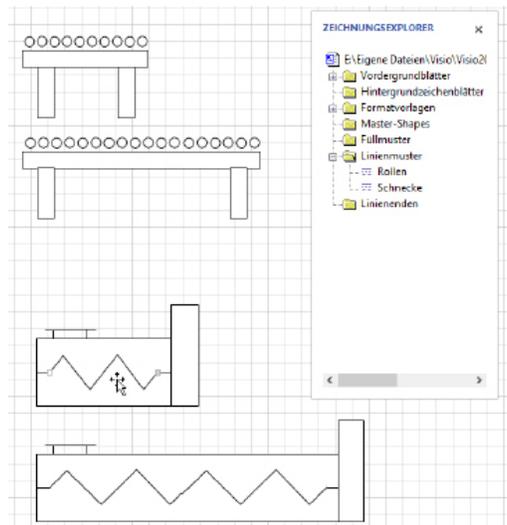
Abbildung 2.10: Die Visio-Schablonen mit den Mustern werden verwendet



Hinweis

Beachten Sie, dass die von Visio zur Verfügung gestellten Muster nur dann sichtbar werden, wenn die Hintergrundfarbe auf weiß gestellt wird.

Abbildung 2.11: Rolle und Schnecke



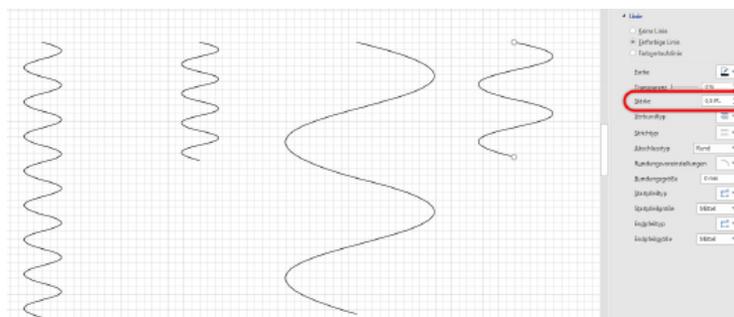
Nun ist es nicht mehr schwierig eigene Linienmuster für technische Zeichnungen zu erstellen. Beispielsweise ein Förderband, bei dem mehr Rollen erscheinen, wenn es in die

Breite gezogen wird. Der »Trick« liegt darin ein skaliertes Linienmuster zu erzeugen, das aus mehreren horizontal nebeneinander liegenden Kreisen besteht. Ebenso kann eine Zickzacklinie erstellt werden mit der Mustereigenschaft skaliert, die beim Verlängern mehr Zacken anzeigt. Sie sehen das Ergebnis in **Abbildung 2.11**: Rolle und Schnecke.

Hinweis

Beachten Sie, dass Sie keine eigenen Schriften definieren können – lediglich Linien- und Füllmuster können selbst erstellt werden. Um eigene Fonts oder Zeichen zu erstellen, verwenden Sie das Windows-Programm Eudcedit.

Abbildung 2.12: Der Ausschlag der Kurve hängt von der Linienstärke ab.



2.2. Eigene Shapes erstellen

Es gibt zwei Varianten neue Shapes zu erstellen: Sie können entweder vorhandene Shapes modifizieren oder ganz neue Shapes anlegen. In diesem Kapitel wird der Fokus auf die zweite Technik gelegt: auf das Generieren von neuen Shapes.

2.2.1. Die Standardelemente

- Öffnen Sie ein leeres Zeichenblatt.

- Öffnen Sie die Registerkarte *Start*. Dort finden Sie in der Gruppe *Tools* die sechs Werkzeuge *Rechteck/Quadrat*, *Ellipse/Kreis*, *Linie*, *Freihandform*, *Bogen* und *Bleistift*. Wählen Sie eines der Werkzeuge aus, um ein neues Objekt zu erzeugen.
- Lassen den Mauszeiger los und ziehen danach mit gedrückter linker Maustaste ein Rechteck auf dem Zeichenblatt auf. Wird ein weiteres Objekt benötigt, dann kann das nächste Objekt durch Ziehen erzeugt werden. Sie müssen nicht erneut den Befehl aktivieren – es bleibt aktiv.

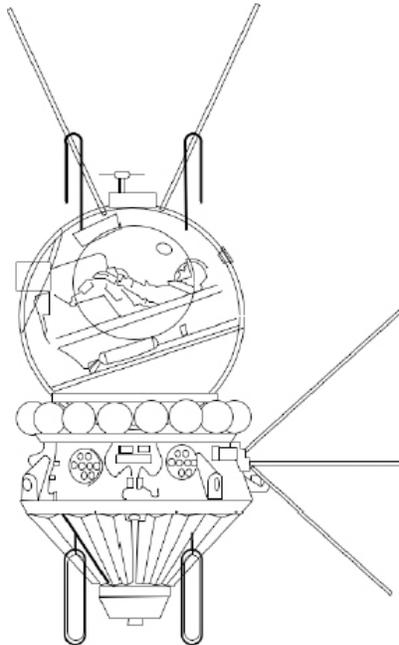
Hinweis

Wenn Sie beispielsweise ein kleines Rechteck auf einem großen zeichnen möchten, kann es passieren, dass Visio davon ausgeht, dass Sie das zuerst gezeichnete Objekt, das noch markiert ist, verschieben möchten. Abhilfe schafft ein Auflösen der Markierung oder ein Zeichnen des zweiten Objekts neben dem ersten. Anschließend kann man es über das erste verschieben.

Wird beim Aufziehen die **[Umschalt]**-Taste gedrückt, erzeugt Visio Quadrate und Kreise. Wird bei der einfachen Linie die **[Umschalt]**-Taste gedrückt, wird sie nur waagrecht, senkrecht oder im Winkel von 45 Grad gezeichnet.

Möchten Sie dagegen das Objekt verschieben, vergrößern oder verkleinern, sollten Sie das Zeigertool (Standardzeiger, weißer Pfeil) aus der Registerkarte *Start* oder **[Strg] + [1]** verwenden.

Abbildung 2.13: Ein neues Shape wird aus Linien, Ellipsen und Rechtecken erstellt.



Der Umgang mit Bogen und Freihandzeichnen erfordert ein wenig Geschick, die Sie sicherlich schnell erwerben, wenn Sie einige Objekte gezeichnet haben. Der Unterschied zwischen *Bogen* und *Freihandzeichnen* besteht darin, dass mit dem Werkzeug *Bogen* nur eine Vierteilellipse gezeichnet werden kann. Mit der Freihandlinie können Sie komplexe Figuren erstellen. Jedoch können Sie mehrere Bögen hintereinander anfügen. Zeichnen Sie einfach den zweiten Bogen vom Endpunkt des

ersten ausgehend bis zum Ende, lassen die Maus los und zeichnen anschließend den dritten Teilbereich.

In **Abbildung 2.13** sehen Sie eine Zeichnung, die aus Linien, Rechtecken und Ellipsen erstellt wurde. Diese geometrischen Objekte können als Shape gespeichert werden.

Wie werden Shapes erstellt, die aus geraden und gebogenen Linien bestehen? Hierzu könnten Sie mit dem Bleistiftwerkzeug arbeiten, wie Sie in **Abbildung 2.14** sehen können. Zeichnen Sie Linie für Linie des neuen Objekts, wobei Sie nach jeder Linie die Maustaste loslassen. Erzeugen Sie zuerst nur gerade Linien. Jede neue Linie wird an den Endpunkt der zuletzt gezeichneten angesetzt. Wenn Sie weitere Knoten benötigen, müssen Sie mit gedrückter **[Strg]**-Taste auf die Linie klicken, um einen weiteren Punkt zu erzeugen, den Sie dann verschieben können. Soll das Objekt geschlossen sein, ziehen Sie die letzte Linie bis zum Beginn der ersten. Das Objekt wird automatisch geschlossen. Zwischen zwei Linienendpunkten befindet sich nun jeweils ein Knoten. Wird er mit dem aktivierten Bleistiftwerkzeug verschoben, so entsteht eine Rundung. Im Befehl *Datei/Optionen/Erweitert* stehen Ihnen einige Einstellungen zur Freihandform-Zeichnung zur Verfügung

Abbildung 2.14: Ein neues Shape wird erstellt.

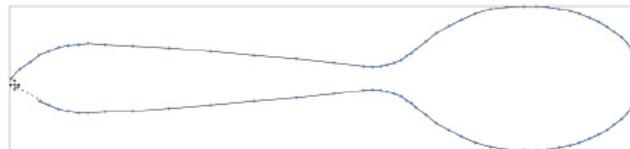


Abbildung 2.15: Die Optionen der Freihandform-Zeichnung



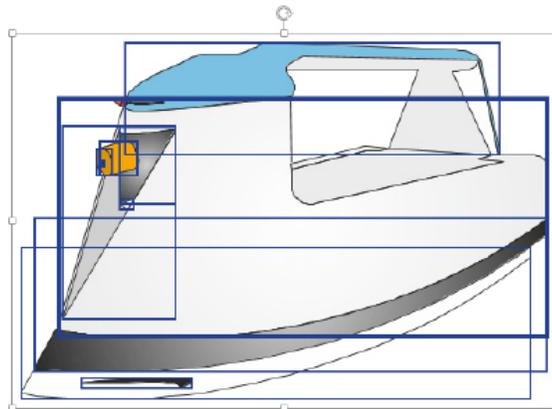
- Wird ein Shape mehrmals benötigt, kann es kopiert und eingefügt werden. Oder Sie duplizieren es mittels **[Strg]+[D]** oder *Start/Zwischenablage/Einfügen/Duplizieren*. Oder Sie verschieben es mit gedrückter **[Strg]**-Taste.
- Zum Formatieren von Shapes stehen Ihnen die Effekte, Linien- und Füllformate zur Verfügung, wie Sie in **Abbildung 2.16** sehen können. Ebenso können die Textattribute voreingestellt werden.

Abbildung 2.16: Die Linienstärke und Füllfarbe wurde geändert



- Wenn Sie die Reihenfolge der Shapes verändern möchten, können Sie ein markiertes Shape an eine andere Position über die Befehle *Start/Anordnen/Ebene nach hinten/In den Hintergrund* oder *Start/Anordnen/Ebene nach vorne/In den Vordergrund* legen.
- Soll im neuen Shape ein Vorgabetext stehen, kann dieser hineingeschrieben werden. Der Benutzer kann ihn jederzeit problemlos überschreiben.

Abbildung 2.17: Eine Zeichnung, die aus mehreren Einzelteilen besteht

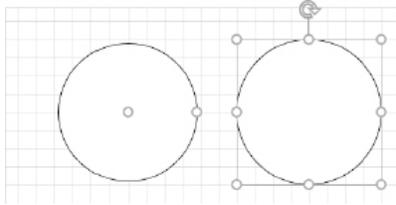


2.2.2. 1D- und 2D-Shapes

Wenn Sie mit den Werkzeugen *Linien*, *Bogen*, *Bleistift* oder *Freihandzeichnen* eine einzelne Linie oder Kurve erzeugen, ist das Shape immer eindimensional. Fügen Sie jedoch im Anschluss eine weitere Linie an, so wird das Objekt, das nun aus zwei Teilstrecken oder Teilkurven besteht, zu einem zweidimensionalen Objekt. Erkennbar ist dies an den Markierungspunkten die Sie erhalten, wenn Sie das Werkzeug *Zeigertool* aktivieren: Eindimensionale Shapes sind ausgerichtet – besitzen also einen Anfangs- und Endpunkt, während zweidimensionale Shapes Markierungspunkte und ein Rotationswerkzeug aufweisen. Das heißt: durch eine Box umschrieben sind. Dies ist deutlich in **Abbildung 2.18** zu erkennen. Sie können diese Eigenschaft explizit im Dialogfeld *Verhalten* in *Entwicklertools/Shape-Design* einsehen, wo eindimensionale Shapes in Rechtecke und Rechtecke in eindimensionale Shapes konvertieren werden können.

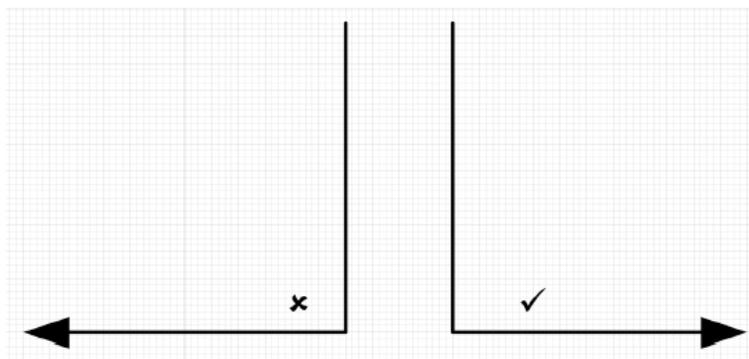
Beachten Sie, dass beim Konvertieren in ein eindimensionales Shapes die Anfangs- und Endpunkte immer links und rechts liegen. Das heißt: Sie sollten das zu konvertierende Shape waagrecht konstruieren.

Abbildung 2.18: Ein Kreis (Am Mittelpunkt ziehbarer Kreis) als Linie und ein Kreis als Rechteck



Übrigens: So wie Sie eigene 2-dimensionale Shapes erstellen können (und in einer Schablone speichern können), können Sie auch 1-dimensionale Shapes, beispielsweise Verbinder erstellen. Das kann interessant sein, wenn die Shapes bereits vorformatiert sein sollen (dicke und dünne Linien, rot für Strom, blau für Wasser, grau für Gas, ...) bestimmte Pfeilspitzen besitzen sollen, auf bestimmten Layern liegen sollen (Stromkabel, Netzwirkkabel, ...) oder mit bestimmten Texten versehen sind, beispielsweise die Ja002DVerbinder mit einem grünen Haken, die Nein-Verbinder mit einem roten x. Die Symbole kann man der Schriftart Wingdings entnehmen.

Abbildung 2.19: Selbst erstellte Verbinder

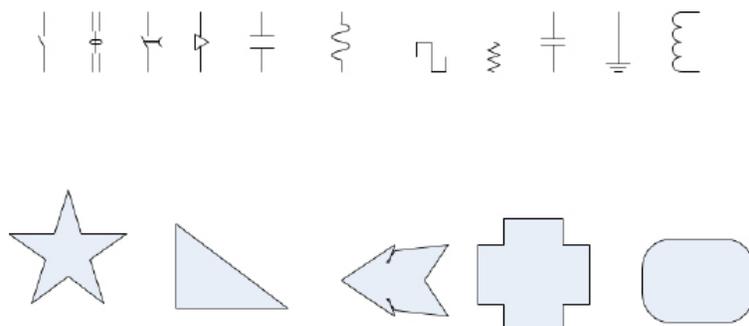


2.2.3. Offene und geschlossene Shapes

Während die Frage nach Ein- und Zweidimensionalität nur eine Frage des Verhaltens beim Verbinden ist, betrifft die Frage nach offenen und geschlossenen Objekten die Formatierung. Nur ein geschlossenes Objekt kann eine Füllfarbe besitzen, nur ein offenes Objekt kann Pfeilspitzen an den Linienenden haben (siehe **Abbildung 2.20**).

Wenn Sie ein offenes Objekt schließen möchten, markieren Sie es, wählen eines der Linien-Werkzeuge aus und ziehen eine Linie vom Anfangspunkt zum Endpunkt. Das Verwenden der Füllfarbe zeigt Ihnen, dass es nun geschlossen ist.

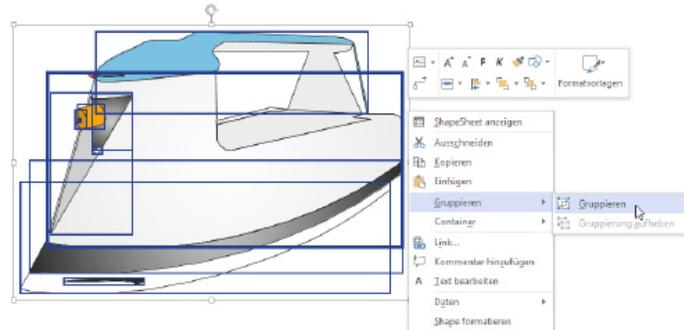
Abbildung 2.20: Mehrere offene und geschlossene Shapes



2.2.4. Gruppieren

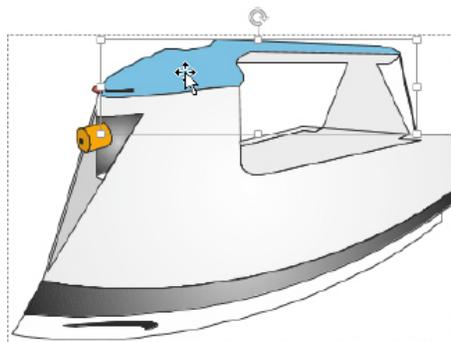
Zwei oder mehrere Shapes können zu einer Gruppe zusammengefasst werden. Dies hat den Vorteil, dass sie wie ein Objekt behandelt werden. Um zwei oder mehrere Shapes zu gruppieren, werden sie markiert und mit dem Befehl *Start/Anordnen/Gruppieren/Gruppieren*, der Tastenkombination **[Strg]+[G]** oder **[Umschalt] + [Strg] + [G]** oder dem Kontextmenü (*Gruppieren/Gruppieren*) gruppiert. Wird die Gruppe erneut markiert, erscheinen die Größenänderungs-Kontrollpunkte um alle Elemente dieser Gruppe. Visio erstellt nun ein neues Shape. Wurden drei Shapes auf einem Zeichenblatt zu einer Gruppe zusammengefasst, so befinden sich nun vier Objekte auf dem Blatt.

Abbildung 2.21: Mehrere Shapes werden zu einer Gruppe zusammengefasst.



Soll nun ein Element der Gruppe bearbeitet werden, beispielsweise formatiert, verschoben, verändert oder gelöscht, genügt ein zweiter Klick auf dieses Gruppenmitglied, um ein Element der Gruppe zu markieren - nun kann es verändert werden, so, als wäre es kein Teil der Gruppe (siehe **Abbildung 2.22**).

Abbildung 2.22: Ein Mitglied der Gruppe wurde markiert.



In manchen Grafikprogrammen muss man die Gruppe aufzuheben (über den Befehl *Start/Anordnen/Gruppieren /Gruppierung Aufheben*, **[Umschalt]+[Strg]+[U]** oder das Kontextmenü) oder eine bestimmte Tastenkombination zu drücken, um in die Gruppe zu gelangen. In Visio können Sie leicht einen Teil der Gruppe verändern, ohne den Verbund der einzelnen Shapes zu zerstören.

Um ein Element aus einer vorhandenen Gruppe herauszuholen, kann dieses markiert werden und über den Befehl *Start/Anordnen/Gruppieren/Aus Gruppe entfernen* extrahiert werden.

Soll ein weiteres Element zur Gruppe hinzugefügt werden, darf keine neue Gruppe erstellt werden, sondern Sie sollten dieses Element und die Gruppe markieren und über den Befehl *Start/Anordnen/Gruppieren/Zur Gruppe hinzufügen* zusammenführen. Würden Sie eine neue Gruppe erstellen, also zwei Gruppen ineinander schachteln,