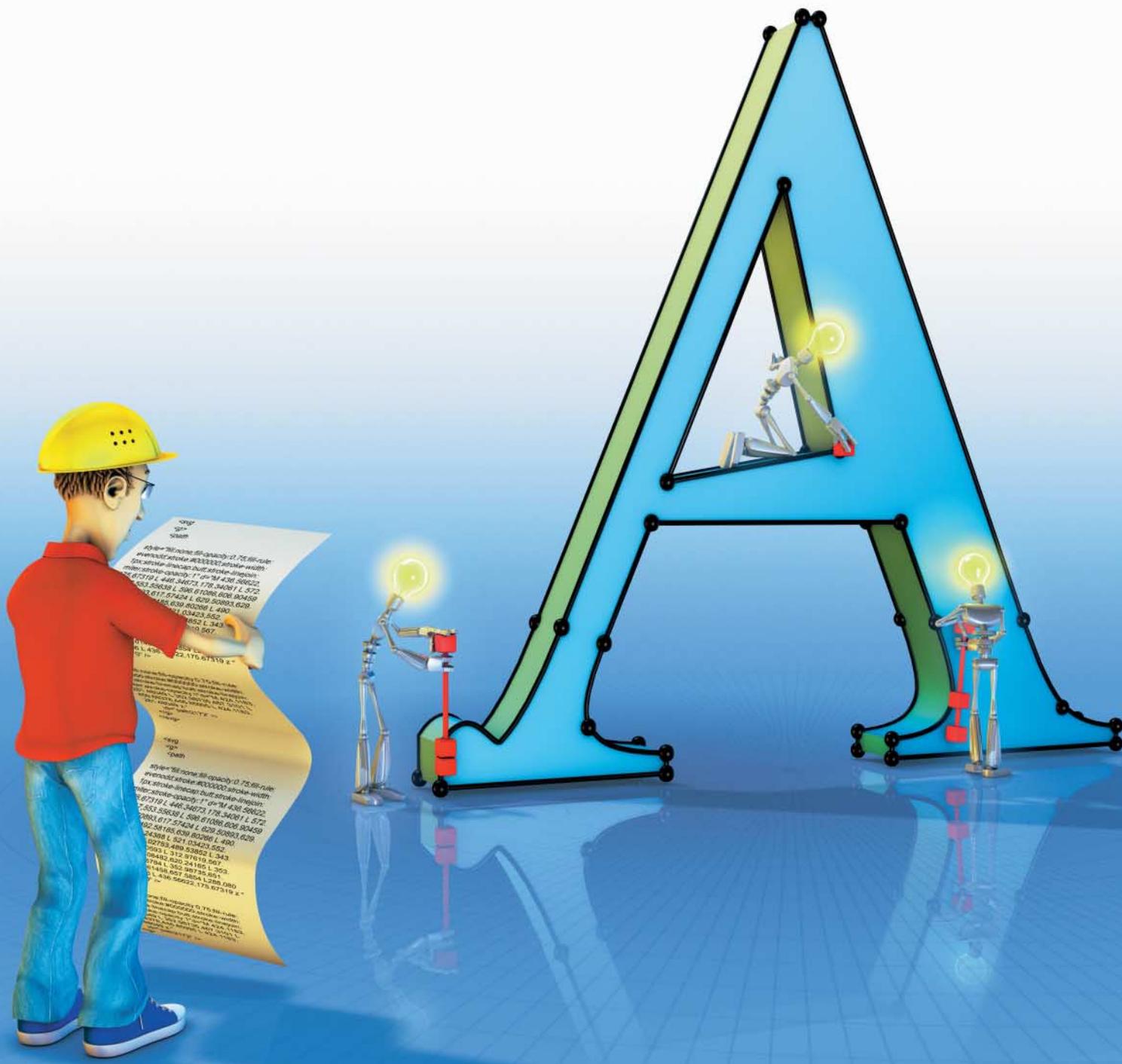


Jacqueline Preuß

Konstruktion nach Plan

Wie Vektorformate Grafiken beschreiben

Vektorgrafiken zeigen auch bei starker Vergrößerung kein Pixelraster und finden doch in schlanken Dateien Platz. Deshalb bilden sie auch die Grundlage für moderne Dateiformate wie PDF, XPS und Flash.



Mathematisch beschriebene Grafiken eignen sich für Konstruktionspläne ebenso gut wie für Dokumenten-Layouts. Und setzt man ein Logo mit Hilfe von Vektoren aus grafischen Primitiven wie Linien, Kurven, Kreisen und Polygonen zusammen, kann man es anschließend sowohl in Daumnagelgröße auf einen Briefkopf wie auch mannshoch auf ein Transparent zur Zierde des Firmengebäudes drucken lassen, ohne die Grafikdatei anpassen zu müssen.

Das liegt daran, dass eine Vektorgrafik-Datei kein fertiges Bild enthält, sondern lediglich mathematische Bauanleitungen für die einzelnen Objekte in denkbar knapper Beschreibung. Einen Kreis definiert sie durch sechs Parameter: die x- und y-Position des Mittelpunktes, seinen Radius, die Stärke der Randlinie sowie Farbangaben für Kontur und Füllung (siehe Bild unten). Anhand dieser berechnet der Computer, welche Bildschirmpixel die Rand- oder Füllfarbe annehmen müssen, um das Bild des Kreises in der gewünschten Vergrößerung zu zeigen. Wählt der Anwender einen anderen Bildausschnitt oder wünscht er eine stärkere Vergrößerung, beginnt die Berechnung von vorn. Dadurch kann man Vektorgrafiken beliebig vergrößern, ohne dass die Qualität leidet. Allerdings beanspruchen aus sehr vielen Einzelobjekten oder komplizierten Formen zusammengesetzte Bilder den Rechner spürbar.

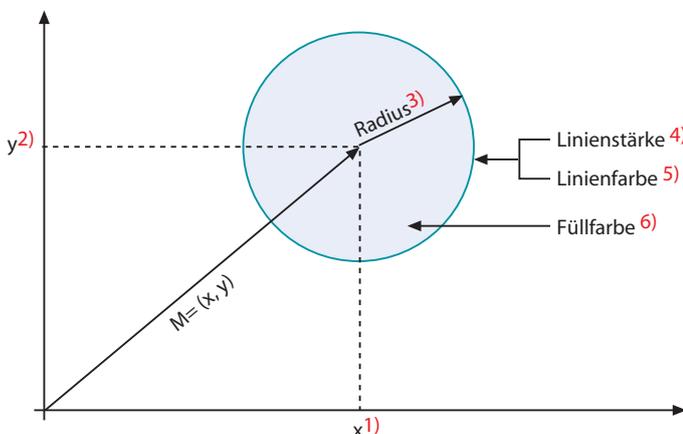
Speichert man eine Grafik in einem Vektor-Datenformat, bleibt die Aufteilung in einzelne

Objekte erhalten. Dadurch kann man diese später gezielt weiterbearbeiten und verdeckte Bildelemente wieder zum Vorschein bringen. Dafür muss nicht zwingend das ursprünglich zum Zeichnen der Vektorgrafik verwendete Programm eingesetzt werden, da sich für den Austausch und die Druckvorbereitung mittlerweile einige Formate etabliert haben (dazu später mehr).

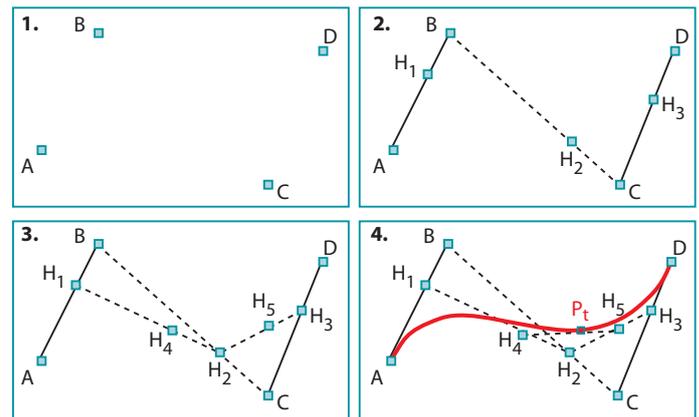
Vektorgrafiken spielen ihre Vorteile bei Motiven aus, die sich leicht aus einzelnen Formen mit klaren Konturen zusammensetzen lassen. Zwar puzzeln Könner mit viel Geschick und aus zahlreichen einzelnen Vektorobjekten auch nahezu fotorealistische Bilder zusammen [1], die typische Vektorgrafik sieht aber eher nach einer sauberen Zeichnung als nach einem Foto aus. Für viele praktische Anwendungen, etwa beim Layout von Zeitschriften, möchte man die Vorteile der reinen Vektorgrafik mit der Fähigkeit kombinieren, auch Bitmap-Bilder einzubinden. Die verbreiteten Vektorformate setzen diese Anforderung auf unterschiedliche Weise um: Während manche die Pixelbilder in die eigentliche Vektorgrafik-Datei einbetten und selbst komprimieren können, verweisen andere nur über einen Link darauf.

Runde Sachen

Alle Elemente einer Vektorgrafik bestehen aus Linien, Kurven oder Flächen. Zu ihrer exakten mathematischen Beschreibung dienen Vektoren. Jeden Vektor $V=(x, y)$ kann man sich als Pfeil in



Vektorgrafiken enthalten lediglich mathematische Beschreibungen der Objekte im Bild. Ein Kreis ist durch sechs Parameter eindeutig definiert.



Eine kubische Bézierkurve zwischen zwei Punkten A und D wird mit Hilfe von zwei Kontrollpunkten B und C konstruiert (1). Die Teilung der drei Strecken nach einem festen Verhältnis definiert drei Hilfspunkte (2). Weitere Punkte entstehen durch weitere Streckenteilungen (3), der letzte Punkt liegt auf der Kurve (4).

einer Ebene vorstellen, dessen Richtung und Länge die x- und y-Komponente gemeinsam festlegen (etwa beim Vektor M auf dem Bild unten). An welchem Punkt im Koordinatensystem der Pfeil genau startet, legt allerdings nicht die Definition des Vektors fest, sondern seine Funktion. Soll er als Ortsvektor die Lage eines Punktes bestimmen (im Bild den Mittelpunkt des Kreises), startet der Pfeil beim Ursprung des Koordinatensystems. Verbindet der Pfeil hingegen einen gegebenen Start- mit einem Endpunkt, so erhält man die beiden Komponenten des Verbindungsvektors, indem man die x- und y-Koordinaten des Startpunkts von jenen des Endpunkts abzieht.

Das Vektorformat SVG etwa speichert für ein Rechteck nur den Ortsvektor einer Ecke und konstruiert anschließend den Umriss aus Angaben zur Länge und Breite. Diese stellen faktisch waagerechte und senkrechte Verbindungsvektoren dar, deren eine Komponente jeweils null beträgt. Soll ein solches Rechteck anschließend auf der Zeichenfläche verschoben werden, definiert man Richtung und Entfernung zwischen alter und neuer Position als Verschiebevektor und addiert dessen Koordinaten zu denen des Ortsvektors. Die Größe des Rechtecks hingegen verändert man durch Multiplikation aller Verbindungsvektor-Koordinaten (und damit von Länge und Breite) mit dem gleichen Faktor. Ein unregelmäßiges Polygon beschreibt SVG

auf andere Weise als ein Rechteck: Hier verfügt jeder Eckpunkt über einen eigenen Ortsvektor, zu dem bei einem Ortswechsel der Verschiebevektor addiert werden muss.

Für weich geschwungene Formen im Bild kommen in der Regel sogenannte kubische Bézierkurven zum Einsatz, über deren Verlauf Kurvenpunkte und an diese angekoppelte Kontrollpunkte entscheiden. Die Positionen dieser Punkte legen wiederum Vektoren fest. Die Kurve selbst verläuft nur durch die Kurvenpunkte, die Kontrollpunkte bestimmen die Krümmung. Sie werden in Vektorzeichenprogrammen oft auch „Anfasser“ genannt, da sie auf die Kurve wie Gummibänder wirken: Je weiter man den Anfasser vom zugehörigen Kurvenpunkt wegzieht, desto stärker lenkt er die Kurve in die gewählte Richtung ab. Dabei wirkt die Verbindungslinie zwischen Kurven- und Kontrollpunkt wie eine Tangente, an die sich die Kurve anschmiegt.

Bei kubischen Bézierkurven verfügen sowohl Start- als auch Endpunkt über jeweils einen eigenen Kontrollpunkt. In der Bildfolge oben gehört B als Kontrollpunkt zum Kurvenpunkt A, C ist an den Kurvenpunkt D gekoppelt. Um aus der Position dieser vier Punkte näherungsweise eine vollständige Kurve zu berechnen, verwendet man das Prinzip der fortgesetzten Streckenteilung. Hierbei verbindet man die Kurven- und Kontrollpunkte zu den drei Strecken AB, BC und CD. Die Punkte H_1 bis H_5



Nahaufnahme des Scheitelpunkts dreier extrem gedehnter Bézierkurven: Während bei Xara Xtreme (links) und Adobe Illustrator (Mitte) keine Stufen zu sehen sind, zeigt Inkscape (rechts) deutliche Knicke an den einzelnen berechneten Kurvenpunkten.



Ältere Vektorgrafik-Formate simulieren Farbverläufe, indem sie das Objekt in einzelne Basiselemente aufteilen und die Farbe stufenweise verändern (oben). Für stufenlose Übergänge verwenden neuere Standards eine mathematische Farbinterpolation – das sogenannte Smooth Shading (unten).

teilen die Länge der Strecken in einem einheitlichen Verhältnis t , beispielsweise mit $t = 0,6$. Im nächsten Schritt verbindet man H_1 und H_2 sowie H_2 und H_3 zu neuen Strecken und teilt diese wiederum im Verhältnis t , wodurch neue Punkte H_4 und H_5 entstehen. Ein dritter und letzter Schritt teilt die Strecke H_4H_5 wiederum im Verhältnis t . Das Ergebnis ist der Punkt P_t , der auf der Kurve selbst liegt. Um die komplette Kurve zu erhalten, lässt man t von 0 bis 1 laufen und führt in jedem Schritt das Verfahren erneut durch.

Eine Vektorgrafik-Datei legt lediglich die Orte der zwei Kurven- und Kontrollpunkte fest, die Berechnung des Kurvenverlaufs nach dem beschriebenen Verfahren erfolgt erst unmittelbar bei der Darstellung auf dem Bildschirm oder bei der Druckausgabe. Das Zeichenprogramm oder der Dateibetrachter legt idealerweise die Anzahl der Berechnungsschnitte für t abhängig von der Krümmung und der Größe der Kurve im Verhältnis zur Auflösung des Rasters fest – andernfalls sieht man bei starker Vergrößerung Knicke (siehe Bild oben links). Verändert man die Lage der Kurvenpunkte oder zieht an einem Anfassern, berechnet die Software die Kurve komplett neu.

Kurven mit vielen Ausbuchtungen und Dellen lassen sich sowohl durch eine Kurve mit vielen Punkten und Anfassern darstellen, aber auch durch Kombination mehrerer einfacher Kurven, wobei der Endpunkt einer Kurve und der Startpunkt einer weiteren aufeinander liegen. Beide gehen weich ineinander

über, wenn die Verbindungslinien zwischen dem gemeinsamen Kurvenpunkt und den jeweiligen Kontrollpunkten auf einer gedachten Geraden liegen (siehe Bild unten).

Hülle und Fülle

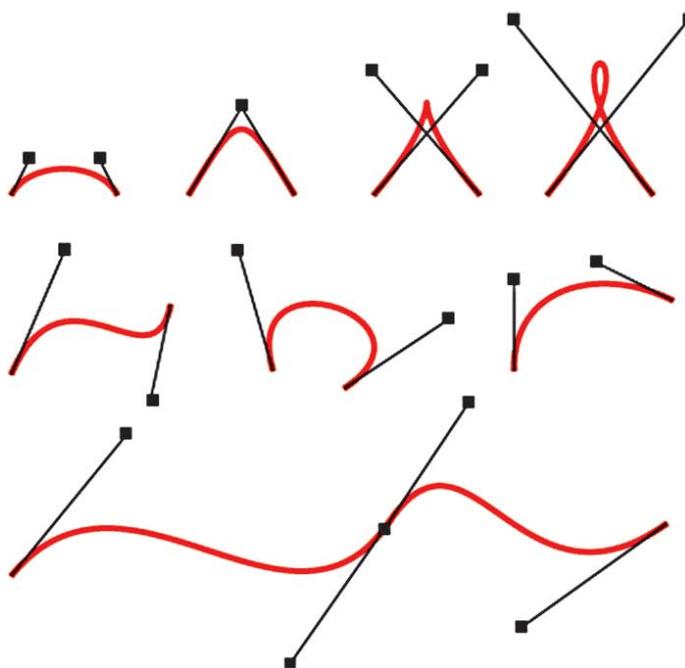
Der von der Kurve umrissenen Fläche weisen Vektorformate wahlweise eine gleichmäßige Farbe oder einen Verlauf als Füllung zu. Auf einfache Weise wird dieser simuliert, indem die Form in einzelne Basiselemente aufgeteilt wird, etwa in Streifen,

die abgestufte Farbtöne zeigen (siehe Bild oben rechts). Das Problem dabei: Wie errechnet man die passende Anzahl an Streifen für unterschiedliche Auflösungen und Farbfähigkeiten von Ausgabegeräten? Bei zu wenig Stufen sieht man die Streifen deutlich, zu viele verzögern die Ausgabe und benötigen außerdem viel Speicher. Der klassische Ausweg aus diesem Dilemma besteht darin, den Farbverlauf in einer festen Auflösung zu rastern und als Pixelgrafik einzubinden. Eine solche Füllung lässt sich allerdings

nachträglich nicht mehr ändern, und bei starker Vergrößerung stören sichtbare Pixel.

Eine höhere Ausgabequalität liefert das sogenannte Smooth Shading. Hierbei beschreiben mathematische Funktionen die Verläufe. Die Farbe einzelner Punkte ergibt sich aus der Interpolation zwischen den Farbwerten der Eck- und Randpunkte des Objektes; zur Auswahl stehen in der Regel lineare, radiale und teilweise komplexere, zweidimensionale Verläufe bis hin zu Farbverlaufsgittern. Je nachdem, welche Funktion den Verlauf bestimmt, verändern sich die Farbwerte zwischen einem Start- und Endwert linear oder exponentiell. Im ersten Fall verändert sich der Farbton kontinuierlich über die gesamte Strecke, im zweiten kann er sich zum Beispiel in Nähe des Start- und Endpunkts stärker verändern als im mittleren Bereich des Verlaufs. Zur Beschreibung sind deutlich weniger Daten notwendig als bei der mittlerweile überholten Streifenmethode. Von den Standard-Vektorformaten können etwa PostScript 3 und PDF ab Version 1.3 Smooth-Shading-Anweisungen speichern.

Vektorzeichenprogramme bereiten die Transparenz von Vektorgrafik-Objekten für den Bildschirm oder beim Druck unterschiedlich auf. Auf dem Monitor bestimmt der in der Vektordatei gespeicherte Alphakanal, wie stark durchsichtig Kontur und Füllung einer Form erscheinen soll. Den Rest erledigt die Render-Engine des jeweiligen Anwendungsprogramms. Der Export in ein Standard-Vektorfor-



Die Kontrollpunkte oder Anfassern ziehen wie Gummibänder an den Bézierkurven – je größer ihr Abstand ist, umso stärker lenken sie die Kurve in ihre Richtung ab. Weiche Übergänge zwischen mehreren Kurven entstehen, wenn der Verbindungspunkt und seine Kontrollpunkte auf einer gedachten Geraden liegen (unten).

mat und die Druckausgabe hingegen benutzen oft sogenanntes Flattening. Dabei werden halb transparente, sich überlappende Formen in viele einzelne Teilobjekte zerlegt, die wie Puzzleteile ineinander greifen und ohne Überschneidungen in eine einzige Bildebene passen, wobei allerdings die ursprüngliche Objektstruktur verloren geht (siehe Bild unten). Verkomplizieren Farbverläufe, Texte, Sonderfarben oder Effekte wie Schlagschatten die Sache zusätzlich, rastern viele Programme die problematischen Bereiche kurzerhand als Pixelbild. Unter Umständen kommt es zu leichten Farbverschiebungen, und ein nachträgliches Bearbeiten des Bereichs in der exportierten Datei ist kaum mehr möglich.

Bildsprache

Es gibt unterschiedlichste Vektorgrafik-Formate für so verschiedene Zwecke wie Animationen, statische Webgrafiken, technische Zeichnungen bis hin zu Druckvorlagen für Zeitschriften oder Bücher. Selbst die Buchstaben in reinen Textdokumenten müssen gezeichnet werden – auch das geschieht mit Hilfe von Bézierkurven, ihrerseits definiert durch Vektoren.

Beim internen Aufbau unterscheiden sich die Vektorformate deutlich (siehe Bild oben). So entpuppt sich das von Adobe seit 1984 entwickelte Format PostScript (PS) beim genaueren Hinsehen als vollwertige Programmiersprache. Sie arbeitet stapelorientiert, nach dem Prinzip der umgekehrten polnischen Notation (UPN). Dabei werden erst die Operanden eingegeben und danach der anzuwendende Operator (Postfix-

```

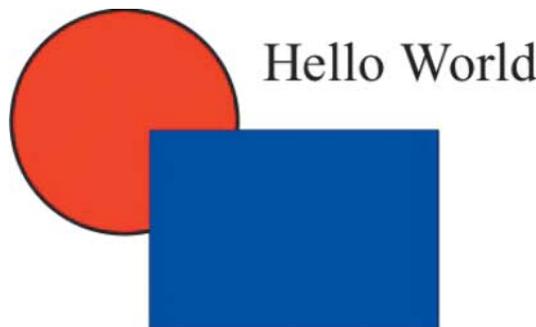
%!PS-Adobe-3.1
%%BoundingBox: 0 0 640 480

%Schrift
181 425 moveto           % Position setzen
/Times-Roman findfont 40 scalefont setfont
(Hello World) show      % Text anzeigen

%Kreis
82 398 moveto           % Position setzen
82 398 81 0 360 arc     % Kreis mit Radius 81
gsave                  % Pfad speichern
3 setlinewidth stroke  % Kontur zeichnen
grestore               % Pfad zurückholen
1.0 0.0 0.0 setrgbcolor % neue Farbe setzen
fill                   % Füllen

%Rechteck
newpath                % neuen Pfad beginnen
100 389 moveto         % neue Position setzen
207 0 rlineto          % Unterkante zeichnen
0 -144 rlineto         % rechte Kante zeichnen
-207 0 rlineto         % Oberkante zeichnen
closepath              % Pfad schliessen
gsave                  % Pfad speichern
0.0 0.0 0.0 setrgbcolor % neue Farbe setzen
1 setlinewidth stroke % Kontur zeichnen
grestore               % Pfad zurückholen
0.0 0.0 1.0 setrgbcolor % neue Farbe setzen
fill                   % Füllen

showpage                % Seite anzeigen
    
```



```

<?xml version="1.0" encoding="iso-8859-1"?>
<svg xmlns="http://www.w3.org/2000/svg"
width="640" height="480">

<!-- Text: Hello World, Position: 181, 55,
Schriftart: Times New Roman, Größe: 40 pt -->
<text x="181" y="55" font-family="Times New Roman"
font-size="40pt">Hello World</text>

<!-- Kreis, Position: 82, 82, Radius: 81 px,
Füllfarbe: rot, Kontur: schwarz, 3px stark -->
<circle fill="red" stroke="black" stroke-width="3"
cx="82" cy="82" r="81" />

<!-- Rechteck, Position: 100, 91, Maße: 207x144 px,
Füllfarbe: blau, Kontur: schwarz -->
<rect x="100" y="91" fill="blue" stroke="black"
width="207" height="144" />

</svg>
    
```

Die Bauanleitung für die gleiche Grafik, abgefasst in zwei unterschiedlichen Vektorformaten: PostScript (links) zeigt sich als Programmiersprache in umgekehrter polnischer Notation, SVG (rechts) folgt der aus XML bekannten Struktur ineinander geschachtelter Tags.

Notation), was den Code für Computer schneller lesbar macht. „3 4 +“ ergibt 7, „50 moveto“ setzt die Schreibposition auf die übergebene Koordinate. Ein PS-Interpreter auf dem Rechner oder direkt im Drucker, der sogenannte Raster Image Processor (RIP), arbeitet das Programm Stück für Stück ab und setzt es in eine Rastergrafik um. PostScript gehört zwar mittlerweile zu den Veteranen unter den Vektorgrafik-Standards, wird wegen seiner hohen Ausgabequalität aber immer noch häufig als Austauschformat zwischen DTP-Programmen und Druckerei verwendet.

Als in sich abgeschlossene Programme können PS-Dateien

nicht ohne weiteres in andere PS-Dateien eingebettet werden. Für diesen Zweck bietet Encapsulated PostScript (EPS) eine geeignete Teilmenge von PS. EPS beschreibt nur eine einzige Seite, die allerdings nicht einzeln gedruckt werden kann – der entsprechende PS-Befehl „showpage“ wie auch andere Anweisungen direkt an den Drucker fehlen im Umfang von EPS. Das Format ist vielmehr zum Einbinden und Transportieren von Grafiken gedacht. Deshalb enthält eine EPS-Datei optional ein Pixel-Vorschau-Bild, was manche Grafikprogramme beim Öffnen einer EPS-Datei ärgerlicherweise statt der eigentlichen Vektorgrafik importieren.

triebssysteme unverändert. Eine PDF-Datei enthält einen Header, eine Seitenbeschreibung und Angaben zu verwendeten Schriften, Farben und anderen eingebetteten Ressourcen. Jedes einzelne Objekt (etwa ein auf jeder Seite erscheinendes Kopfzeilen-Logo) ist dabei in der Datei nur einmal vorhanden, es kann über eine Referenztafel allerdings ohne weiteren Speicherbedarf mehrmals an verschiedenen Stellen ins Dokument eingebunden werden. Während eine vierfarbige Zeitungsseite als PS-Datei mehr als vier MByte benötigt, kommt die PDF-Datei mit Hilfe der integrierten, verlustfreien ZIP-Komprimierung mit nur 900 KByte aus. Die in PDF ebenfalls eingebaute JPEG-Kompression hält die Dateigröße auch bei eingebetteten Rasterbildern noch im erträglichen Rahmen.

Portabel

PDF (Portable Document Format) ist das zweite bekannte Vektorgrafik-Format aus dem Hause Adobe. Es baut auf das gleiche Grafikmodell wie PS auf, ist aber keine universelle Programmiersprache, sondern ein schlichtes Datenformat, das für die Anzeige auf einen Viewer oder ein Grafikprogramm angewiesen ist. PDF fasst Texte, Fonts und Bildinformationen in einem einheitlichen Dokument zusammen; dessen Layout bleibt bei der Weitergabe auf andere Be-

Immer mehr Anwendungsprogramme schreiben Daten selbst in PDF, alternativ überlässt man Werkzeugen wie Adobe Acrobat, Ghostscript, dem kostenlosen PDFCreator oder Online-Konvertierungsdiensten diesen Part. Alle Informationen über das Ausgangsformat gehen beim Speichern verloren. Eine weitere Bearbeitung von PDF ermöglichen nur wenige Anwendungen [2].



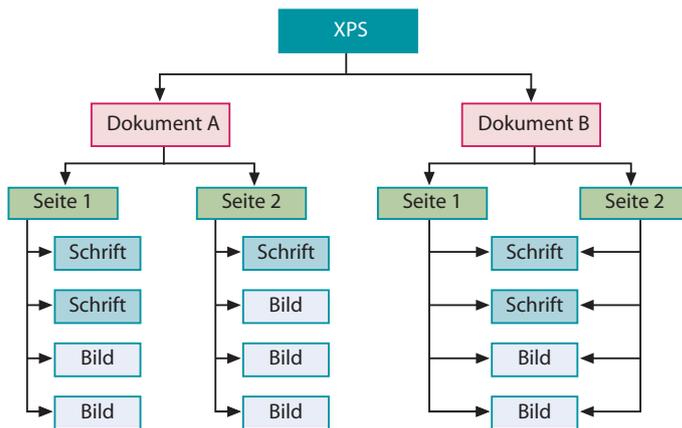
Sogenanntes Flattening reduziert Grafiken auf eine Ebene, transparente, sich überlappende Formen werden dabei in einzelne Objekte aufgeteilt.

Bereits beim Erzeugen eines PDF sollte die Entscheidung fallen, ob die Datei für den Ausdruck oder für die Ansicht am Bildschirm optimiert wird – im zweiten Fall reicht etwa für eingebundene Bitmaps in der Regel eine deutlich kleinere Auflösung aus, was die Exportdatei kleiner macht. Die PDF-Familie umfasst noch einige spezialisierte Mitglieder: So spiegelt der strengere Standard PDF/X die Anforderungen der Druckindustrie wider, indem er Inhalte untersagt, die das Druckergebnis beeinträchtigen können, und für den Drucker notwendige Angaben etwa zu Anschnitt oder verwendeten Farben verpflichtend macht [3]. PDF/A hingegen ist für die Langzeitarchivierung gedacht. Hier liegt besonderes Augenmerk auf der Vollständigkeit der Daten: Untersagt sind Referenzen auf Ressourcen wie Schriften oder Bilder, die nicht im PDF selbst eingebettet sind und damit irgendwann nicht mehr zugänglich sein könnten.

Papierformat

Als Alternative zu PDF präsentierte Microsoft zusammen mit Windows Vista und .NET-Framework 3.0 seine XML Paper Specification, kurz XPS [4]. Sie baut auf XAML (Extensible Application Markup Language) auf, einer ebenfalls von Microsoft konzipierten Beschreibungssprache auf XML-Basis für Vektorgrafiken, Animationen und Interaktionen [5]. Eine XPS-Datei ist eigentlich ein ZIP-Archiv, das Texte und Grafiken, eingebettete Ressourcen sowie Informationen zum Seitenlayout oder zur digitalen Rechteverwaltung als separate Dateien enthält (siehe Bild oben). Über letztere kann der Verfasser zum Beispiel erlauben, das Dokument anzuzeigen und weiterzuleiten, aber den Ausdruck verhindern.

XPS erlaubt Transparenzen, Verläufe und verschiedene Farbräume wie CMYK oder Sonderfarben, was gerade für die Druckvorstufe und Grafikanwendungen interessant ist. Dank XAML als gemeinsamer Basis können Anwendungen, die unter Vista oder .NET zur Bildschirmanzeige die Interface-Bibliothek Windows Presentation Foundation (WPF) benutzen, ohne lästige Farbkonvertierung oder ähnliche Trans-



Eine XPS-Datei kann mehrere Dokumente mit jeweils mehreren Seiten enthalten. Ressourcen wie Schrift und Bild werden außerhalb dieser Hierarchie gespeichert, sodass sie von mehreren Seiten verwendet werden können.

formationen direkt XPS-Dateien erzeugen. Nutzer der Programme von Microsoft Office 2007 müssen allerdings noch ein Plugin für den XPS- und PDF-Export nachinstallieren. Das hat keine technischen Gründe, sondern ist das Ergebnis einer Forderung von Adobe. Als kostenlosen XPS-Betrachter stellt Microsoft den Internet Explorer zur Verfügung, der steht aber nur für Windows-Rechner zur Verfügung. XPS ist ein reines Export-Format, eine nachträgliche Bearbeitung ist nicht möglich.

Animierend

Auch wenn auf den ersten Blick Welten zwischen XPS- oder PDF-Dokumenten und Flash-Filmchen zu liegen scheinen – im Kern handelt es sich in beiden Fällen um Vektorgrafiken. Flash ist ein kommerzielles Format für zweidimensionale Grafiken und Vektoranimationen. Beim Entwurf einer Flash-Grafik oder Animation benutzt man eine Projektdatei (Endung .FLA), vor der Wiedergabe exportiert man sie als komprimierte SWF-Datei.

Darstellung oder Wiedergabe einer SWF-Datei besorgt der kostenlose Flash Player, der für diverse Betriebssysteme implementiert ist und als Plug-in standardmäßig mit den Browsern von Mozilla und Microsoft ausgeliefert wird. SWF-Dateien sind sehr klein, lassen sich nachträglich aber kaum mehr bearbeiten. Sie weisen ein sogenanntes Tag-basiertes Binärformat auf: Einzelne Datenblöcke (Tags) erzeugen Objekte (Definitionstags) oder

manipulieren sie (Kontrolltags). Jedes Kontrolltag bezieht sich auf das unmittelbar vorher definierte Objekt, was SWF Streaming-fähig macht: Die Datei muss nicht komplett vorliegen, um mit ihrer Darstellung beginnen zu können. Kontrolltags für neuere Flash-Versionen hängt man an ältere an, ältere Ausgaben des Flash-Players ignorieren diese Tags einfach.

Skalierbar und offen

Mit Scalable Vector Graphics (SVG) entstand ein offener W3C-Standard für Vektorgrafikdateien in XML-Syntax, der von vielen Programmen wie CorelDraw, Adobe Illustrator oder dem kostenlosen Inkscape unterstützt wird. Anders als SWF lässt sich SVG direkt in den Code einer Webseite integrieren. Browser wie Firefox, Opera oder Safari unterstützen SVG bereits ohne zusätzliches Plug-in, lediglich der Internet Explorer benötigt noch eines.

SVG sieht neben den üblichen Vektorformen und Text auch Füllungen mit Farbverläufen, in den Render-Prozess eingeschaltete Effektfiler wie Weichzeichnung und Animationen vor. Pixelgrafiken und Schriften fügt SVG ähnlich wie HTML lediglich als Verweise ein, weshalb es sich nur bedingt als Austauschformat eignet: Wer Schriften oder Bitmaps benutzt, die auf dem Zielsystem möglicherweise nicht vorhanden sind, muss damit rechnen, dass Texte dort anders dargestellt werden und Bilder völlig fehlen. Dennoch ist SVG

als Austauschformat keineswegs unbrauchbar, da es dank seiner XML-Struktur stets lesbar und unabhängig von Rechnerplattformen oder speziellen Editoren bleibt. Im Web steht der Inhalt von Textblöcken einer SVG-Datei auch Suchmaschinen offen.

Auch sonst fügt sich SVG gut in die Web-Welt ein: CSS (Cascading Style Sheets) beeinflussen das Aussehen der Grafiken, CGI-Scripts, HTML-Parameter oder Javascript erzeugen sie dynamisch oder modifizieren sie auf bequeme Weise, weshalb etwa der Internet-Kartendienst Google Maps für die Darstellung von Polylinien und anderen Vektorgeometrien bevorzugt SVG einsetzt (wer den Internet Explorer benutzt, bekommt die Grafiken im VML-Format geliefert; ihre Darstellung dauert deutlich länger). Die Zukunft von SVG ist dennoch offen – Microsoft führt XAML als Herausforderer ins Feld, Adobe setzt auf Flash, seit es dessen Hersteller Macromedia übernommen hat und kündigte an, die Arbeit an seinem kostenlosen SVG-Plug-in zum Jahresende einzustellen.

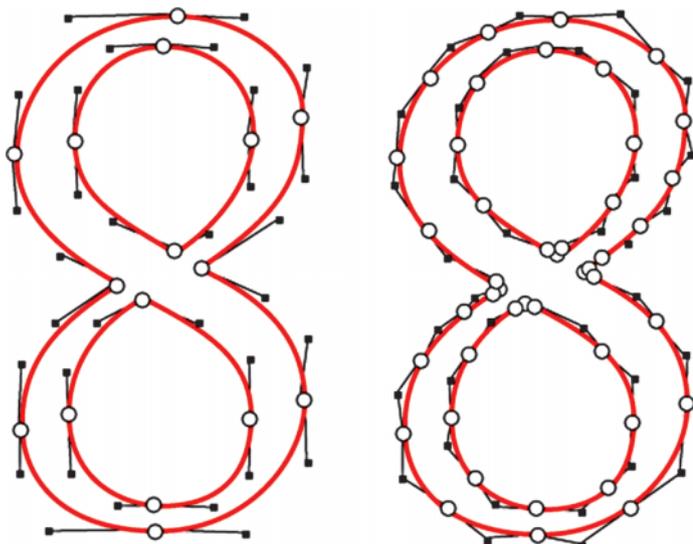
Wandlungsfähig

Obwohl die Vektorgrafik-Formate unterschiedliche Nischen besetzen und für spezielle Aufgaben besonders ausgestattet sind, lassen sich vor allem Standardelemente wie Kurven, Linienzüge, Textblöcke und einfarbige Formen mit Hilfe von Grafikanwendungen oder Konvertern ohne große Verluste in andere Formate übertragen [6]. Allerdings bietet kaum ein handelsübliches Vektorzeichenprogramm Importfilter für sämtliche Standardformate. Möchte man eine EPS- oder PS-Datei in Inkscape importieren, muss man zunächst PostScript in SVG übersetzen – mit einem separaten Werkzeug, da dem SVG-Editor Inkscape selbst der passende Importfilter fehlt. Eingebettete Ressourcen der Ausgangsdatei wie Bitmaps werden bei der Umwandlung als eigenständige Dateien ausgelagert. Transparente Bereiche liegen in Folge des Flattenings in PS als Rasterbilder vor; die Grafik lässt sich dann nicht mehr so einfach bearbeiten oder verlustfrei skalieren. Kopiert man sie auf einen anderen Rechner, muss man alle benutzten Bitmaps und Schriften von Hand mit dazupa-

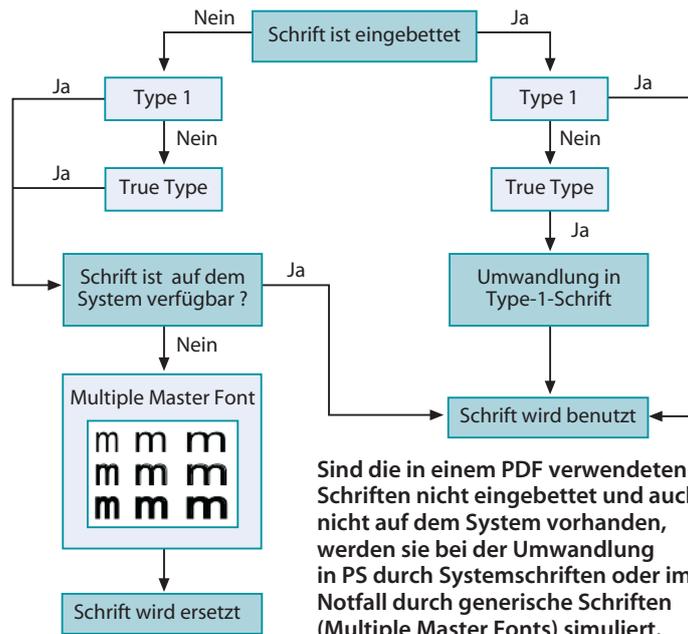
cken und auf dem neuen System im SVG die Pfadangaben zu diesen externen Ressourcen anpassen, um das originale Aussehen der Grafik wiederherzustellen.

Eine Wandlung von PS nach PDF gelingt meist ohne Schwierigkeiten, da beide Formate das gleiche Grafikkmodell verwenden. Umgekehrt ist das anders: So müssen die in PS per Flattening simulierten Transparenzen etwa für die Ausgabe auf einem PostScript-Drucker erst aufwendig gepixelt werden – und das kann dauern.

Externe Ressourcen wie Schriften integrieren die einzelnen Formate auf unterschiedliche Weise. So verwendet PS ausschließlich sogenannte Type-1-Schriften, wohingegen insbesondere auf Windows-Rechnern oft TrueType-Schriften zum Einsatz kommen. Der Unterschied liegt in der Vektorbeschreibung der Buchstaben: TrueType benutzt quadratische Bézierkurven, bei denen sich je zwei Kurvenpunkte einen Kontrollpunkt teilen müssen (siehe Bild unten), während Type 1 die zuvor im Detail beschriebenen kubische Bézierkurven benutzt. Wandelt man ein PDF, das eine TrueType-Schrift enthält, in PS um, muss auch die Schrift konvertiert werden. Bei der direkten Umrechnung in Type-1-Buchstaben können sich allerdings Formverfälschungen einschleichen. Alternativ kann der Druckertreiber die TrueType-Schrift mit einer äquivalenten



Type-1-Schriften verwenden kubische Bézierkurven (links) und benötigen daher meist weniger Kurvenpunkte zur Definition einer Form als TrueType-Schriften, die quadratische Bézierkurven benutzen (rechts).



PS-Schrift ersetzen, sofern diese auf dem System vorhanden ist. Ansonsten wird sie einfach durch eine Standardschrift ersetzt. PDF und PS führen dabei sogenannte Fontdeskriptoren ins Feld. Das sind Datensätze, die etwa Schnitte, Laufweite und Strichstärken der gesuchten Schrift beschreiben. Der verwendete Konverter sucht dann aus einem universellen Schriftsatz, den Multiple Master Fonts, eine Schrift mit ähnlicher Beschreibung heraus oder passt einen vorhandenen Font an die Sollwerte an und ersetzt damit den fehlenden Font.

Das stellt zumindest sicher, dass der Text immer angezeigt wird und sich der Textumbruch möglichst wenig verändert (Bild oben).

Die Ersetzung der Schrift durch einen PS-Font hat allerdings einen Haken: Type 1 unterstützt nur 256 verschiedene Zeichen. Einen Ausweg könnten OpenType-Schriften bieten, die Type-1- und TrueType-Schriften vereinen und etwa von XPS unterstützt werden. Eine direkte Umwandlung von PS nach XPS ist trotzdem nicht möglich, da der XPS-Druckpfad die Verarbeitung von PS-Code nicht vorsieht. Hier muss der Umweg über das PDF genommen werden.

Die Qualität der Umwandlung von einem Vektorformat in ein anderes hängt größtenteils vom verwendeten Programm ab; allerdings bleiben Spezialitäten einzelner Formate dabei in der Regel auf der Strecke – seien es die Animationstaleute von Flash und SVG, die Fähigkeit zur Interaktion bei XAML oder DTP-Eigenheiten wie Anweisungen zum Überdrucken oder Überfüllen bei PS und PDF. Fast jeder Hersteller von Vektorzeichenprogrammen pflegt darüber hinaus sein eigenes Dateiformat, das er optimal auf die benutzte Render-Engine abstimmen kann. So bietet Xara Xtreme von der Auflösung unabhängige Schlag Schatten oder einen Facetteneffekt für Vektorformen an, Inkscape bringt einen gaußschen Weichzeichnenfilter mit. Spätestens beim Export in eines

der Standard-Austauschformate müssen solche Effekte aber wieder auf dessen Bordmittel heruntergebrochen werden – oft bleibt da nur übrig, komplizierte Bildteile mit fester Auflösung zu rastern und als Bitmap einzubinden, was viele Vorteile des Vektorformats zunichte macht.

Keinesfalls kann man sich darauf verlassen, dass notwendige Konverter für die Umwandlung eines beliebigen Vektorformats in jedes andere erhältlich sind, schon gar nicht kostenlos – oft besteht die einzige Lösung darin, die Ausgangsdatei in einem teuren Profi-Zeichenprogramm zu öffnen und in einem anderen Format zu speichern. Ganz ohne Verluste geht selbst das nicht immer über die Bühne, daher sollte man vor Beginn der Arbeit an einer Grafik oder einem Dokument genau überlegen, welches Vektorgrafik-Format den Zweck am besten erfüllt und welches Zeichen- oder Satzprogramm Dateien des gewünschten Typs von sich aus erzeugt. Entsprechende Informationen liefern die Tabellen bei Software-Tests in c't [7, 8]. (pek)

Literatur

- [1] Fotorealistische Vektorgrafiken des japanischen Illustrators Takashi Morisaki: www.real-trace.com/index.html
- [2] Andrea Trinkwalder, Jäger des verlorenen Inhalts, PDF bearbeiten und Texte, Bilder, Grafiken und Formularextrahieren, c't 11/06, S. 152
- [3] Thomas Merz, Sorglos-Drucksache, PDF/X soll eine reibungslose Produktion garantieren, c't 1/03, S. 138
- [4] Hans Bärffuss, Andrea Trinkwalder, Microsofts PDF-Rivale, Das universelle Dokumentenformat XPS, c't 2/07, S. 148
- [5] Hajo Schulz, Bunte Tüte, Microsofts WPF soll den Desktop und das Web erobern, c't 16/07, S. 104
- [6] Dieter Brors, Peter König, Harald Vogel, Dr. Volker Zota, Auf Schirm und Papier, Betrachter und Konverter für Windows, Linux und Mac OS X, c't 10/07, S. 118
- [7] Andrea Trinkwalder, Peter König, Zeichnen nach Zahlen, Vektorgrafikprogramme für Design und technische Illustration, c't 25/05, S. 126
- [8] Peter König, Hausmannskunst, Günstige Vektorzeichenprogramme für Amateure, c't 12/06, S. 168

