Engineer

SVG

eine XML-Sprache für skalierbare Grafiken

Was es ist

SVG steht für Scalable Vector Graphics und ist eine Sprache, die zweidimensionale Grafiken als Formen statt als Pixelraster beschreibt. Die Datei sagt, was gezeichnet werden soll: hier ein Kreis, eine Linie von diesem Punkt zu jenem, dieser Umriss blau ausgefüllt. Der Browser berechnet die Pixel für die jeweilige Darstellungsgröße. Darauf beruht die Skalierung: In der Datei liegen keine Pixel, die gedehnt werden müssten.

Das W3C definiert die Sprache. Ihre neueste Fassung, SVG 2, ist ein Candidate Recommendation Snapshot vom 6. Oktober 2026. Eine Veröffentlichung als Recommendation wird frühestens für den 1. Dezember 2026 erwartet; die Fassung baut auf SVG 1.1 Second Edition auf. Die Spezifikation fasst SVG als XML-basierte Sprache zusammen, die Vektorformen mit Rasterbildern verbinden sowie Stile, Skripte und Animationen verwenden kann. Eine Zeichnung kann eigenständig sein, in einer HTML-Seite stehen oder in eine andere XML-Sprache eingebettet werden.

Das Vokabular ist klein. Sechs Grundformen, `rect`, `circle`, `ellipse`, `line`, `polyline` und `polygon`, decken die einfachen Fälle ab. Ein `path` zeichnet alles Weitere aus einer kurzen Befehlsfolge. `text` platziert echten, auswählbaren Text. In einer Gruppe (`g`) können mehrere Formen einen Stil oder eine Transformation gemeinsam nutzen; ein Verlauf oder Muster kann eine Form anstelle einer einzigen Farbe ausfüllen. Filtereffekte, einst ein Kapitel von SVG 1.1, haben inzwischen eine eigene Spezifikation. Das anfängliche Koordinatensystem hat seinen Ursprung links oben; x wächst nach rechts und y nach unten.

Die Spezifikation definiert SVG anhand eines Document Object Models, nicht anhand eines bestimmten Dateiformats. Eine Zeichnung kann als eigene Datei mit dem Inhaltstyp `image/svg+xml` und der Erweiterung `.svg` ausgeliefert werden oder als `svg` -Element direkt in einer HTML-Seite stehen. Eine eigenständige Datei muss den SVG-Namensraum deklarieren. In HTML ordnet der Parser die Elemente selbst dem richtigen Namensraum zu; ein `xmlns` -Attribut ist dort nicht nötig.

SVG ist kein Rasterformat wie PNG oder JPEG. Diese speichern für jedes Pixel eine Farbe und werden beim Vergrößern unscharf. Ein Foto gehört in eines dieser Formate. SVG eignet sich für Gezeichnetes: Icons, Logos, Diagramme, Schaubilder, Karten und Schriftzüge. Es kann auch ein Rasterbild enthalten; das meint die Spezifikation mit gemischten Grafiken.

Das Konzept für eine Entscheidung nutzen

Nutzen Sie „Was ist SVG?“ für eine konkrete Entscheidung. Wir können helfen, das Konzept mit dem System in Verbindung zu bringen, das Sie entwickeln oder einkaufen. Oberflächen, die schnell laden, auf dem Smartphone funktionieren und sich in allen Browsern Ihrer Kunden gleich verhalten.

Unsere Leistungen entdecken Kontakt aufnehmen →

Die Annahmen hinter dem Begriff prüfen

Nutzen Sie die Erklärung und die verlinkten Arbeiten, um Fragen für Ihr Projekt zu formulieren. Wir können die Annahmen gemeinsam mit Ihnen prüfen, bevor ein Werkzeug ausgewählt wird. Full-Stack-GIS-Kartenentwicklung geleitet — Full-Stack-Kartenentwicklung von der Datenbank bis zum Browser

Full-Stack-GIS-Kartenentwicklung geleitet Kontakt aufnehmen →

Warum es wichtig ist

Eine Datei genügt für jeden Bildschirm. Ein einmal gezeichnetes Icon bleibt auf einem alten Monitor, auf einem Telefon mit mehreren Gerätepixeln pro CSS-Pixel und auf Papier scharf. Separate Exporte für jede Größe müssen nicht auf demselben Stand gehalten werden. Einfache Grafiken ergeben zudem kleine Dateien: Sie beschreiben wenige Umrisse, statt Tausende Pixel zu speichern. Die Spezifikation registriert `.svgz` als Erweiterung für eine mit gzip komprimierte SVG-Datei.

Da das Markup Text ist, passt es in die gewohnte Arbeitsweise eines Teams. Der Export eines Designers lässt sich im Code-Review lesen und in der Versionsverwaltung Zeile für Zeile vergleichen. Ein Skript kann eine Zeichnung aus Daten erzeugen; so entstehen viele Diagramme. Steht die Zeichnung direkt

in einer Seite, gehören ihre Elemente zum DOM. CSS gestaltet sie, ein Skript kann sie verändern, und sie reagieren wie die übrige Seite auf Hover, Fokus und das Farbschema des Lesers.

SVG transportiert außerdem Bedeutung, die eine Bitmap nicht transportiert. Text in einem SVG bleibt Text. In einem Verarbeitungsmodus, der Interaktion erlaubt, verlangt die Spezifikation vom Browser, dass Leser ihn suchen und auswählen können. Ein `title`-Element gibt der Zeichnung oder einem ihrer Teile einen zugänglichen Namen. SVG 2 weist dem `svg`-Element die implizite ARIA-Rolle `graphics-document` zu. Ein als SVG gezeichnetes Diagramm kann beschriftete Teile haben, die ein Screenreader einzeln erreicht. Dasselbe Diagramm als PNG ist ein einzelnes Bild mit einer einzigen Textalternative.

Wie es in der Praxis aussieht

Hier steht ein kleiner Pfeil in einem abgerundeten Rahmen direkt im HTML einer Seite:

```
<svg viewBox="0 0 120 40" width="240" height="80">
  <title>Next step</title>
  <g fill="none" stroke="currentColor" stroke-width="3">
    <rect x="2" y="2" width="116" height="36" rx="8" />
    <path d="M 20 20 H 92 M 80 10 L 100 20 L 80 30" />
  </g>
</svg>
```

`viewBox` gibt der Zeichnung eigene Koordinaten: vier Zahlen für die linke Kante, die obere Kante, die Breite und die Höhe des sichtbaren Bereichs, hier 120 Einheiten breit und 40 hoch. `width` und `height` verlangen 240 mal 80 CSS-Pixel; jede Einheit wird somit zwei Pixel breit gezeichnet. Werden die beiden Zahlen auf 60 und 20 geändert, schrumpft die gesamte Zeichnung einschließlich ihrer Linienbreiten, weil darin nichts in Pixeln gemessen wird.

Die Gruppe legt die Darstellung einmal fest. Füllung, Kontur und Konturbreite werden vererbt; Rahmen und Pfeil übernehmen sie daher beide von `g`. `rect` zeichnet den Rahmen, dessen Ecken `rx` abrundet. Der Pfad liest sich wie eine Anleitung für einen Stift: `M 20 20` bewegt ihn ohne zu zeichnen an einen Punkt, `H 92` zieht eine waagerechte Linie bis x = 92, das zweite `M` hebt den Stift und setzt ihn oben an der Pfeilspitze wieder an, und die beiden `L`-Befehle zeichnen deren zwei Striche. Großgeschriebene Befehle verwenden absolute,

kleingeschriebene relative Koordinaten. Derselbe Pfad lässt sich deshalb als `m 20 20 h 72 m -12 -10 l 20 10 l -20 10` schreiben; jeder Schritt wird von der letzten Position des Stifts aus gemessen.

`currentColor` gibt der Kontur den Wert der CSS-Eigenschaft `color`, die ein direkt eingebettetes `svg` vom umgebenden HTML erbt. Wird der Pfeil in einen Link gesetzt, erscheint er in dessen Farbe. Beim Wechsel zu einem dunklen Thema passt sich seine Farbe ohne zweite Datei an. `title` wird nicht gezeichnet. Es gibt der Zeichnung ihren zugänglichen Namen; SVG 2 merkt an, dass Browser es gewöhnlich als Tooltip zeigen, wenn der Zeiger auf der Zeichnung ruht.

Als eigene Datei benötigt dieselbe Zeichnung den Namensraum am Wurzelelement, `<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 120 40">`. Eine Seite zeigt sie mit `<img src="/images/next.svg" alt="Next step">`. Sie ist dann ein separates Dokument. CSS-Vererbung verläuft vom Eltern- zum Kindelement, und das Wurzelelement dieses Dokuments hat kein Elternelement. `currentColor` innerhalb der Datei folgt deshalb nicht mehr dem Text um das `img`. Die Textalternative liefert nun das Attribut `alt`. SVG 2 erwartet, dass der Browser die Datei im sicheren animierten Modus zeigt, ohne Skripte und ohne externe Dateien.

Die Erklärung zur Barrierefreiheit dieser Website sagt, dass ihre Seiten in HTML, CSS und SVG geschrieben sind, ergänzt um WAI-ARIA, wo HTML allein etwas nicht benennt, und dass jede Seite ohne ausführbares JavaScript vollständig ist. Dort ist SVG ebenso wie in diesem Artikel eine der Sprachen, in denen eine Seite geschrieben wird, neben ihrem Markup und ihren Stylesheets.

Wo es schiefgeht

Eine Linienzeichnung erscheint als ausgefüllte Form. Der Anfangswert von `fill` ist schwarz und der von `stroke` none. Ein als Linie gedachter offener Pfad, etwa eine Pfeilspitze oder ein Häkchen, wird ausgefüllt, als wären seine Enden verbunden. Ohne Kontur werden die Linien selbst überhaupt nicht gezeichnet. Für Liniengrafiken einmal `fill="none"` und eine `stroke`-Kontur an der Gruppe festlegen, die sie enthält.

Das Icon verschwindet im dunklen Modus. Exportierte Icons enthalten häufig eine feste Farbe, `fill="#1a1a1a"`, die auch auf dunklem Hintergrund dunkel bleibt. Ein solches Attribut ist zudem die schwächste Form der Stilzuweisung:

SVG 2 gibt Präsentationsattributen die Spezifität null. Eine beiläufige CSS-Regel an anderer Stelle überschreibt es deshalb unbemerkt. Für Icons im Text `currentColor` verwenden und Farben über CSS statt im Markup festlegen.

Die Zeichnung skaliert nicht oder falsch. Ohne `viewBox` entsprechen ihre Einheiten den Pixeln des Browsers. Eine Änderung der Breite schneidet sie deshalb nur zu oder ergänzt Leerraum; `preserveAspectRatio` wird vollständig ignoriert. Eine negative Breite oder Höhe in `viewBox` ist ein Fehler, durch den das Attribut als fehlend gilt. Eine Breite oder Höhe von null unterbindet die Darstellung. Jeder Zeichnung eine `viewBox` geben und die Größe durch `width` und `height` oder CSS bestimmen lassen. Der Standardwert `xMidYMid meet` hält die Proportionen und zentriert die Zeichnung in ihrer Box.

Skalierbar bedeutet nicht automatisch lesbar. Nichts verhindert, eine Vektorzeichnung mit 16 Pixeln darzustellen. Details, die bei 200 gut zu erkennen sind, können dort jedoch zu einem Fleck verschmelzen. Ein Portfolioeintrag zum Aufbau der visuellen Identität dieses Unternehmens beschreibt ein Vektorzahnrad für den Browser-Tab mit der Überlegung, dass eine kräftige Silhouette 16 Pixel übersteht. Bei dieser Arbeit im Juli 2026 wurden Zahnrad und Unternehmenszeichen nebeneinander mit 16, 32 und 48 Pixeln auf hellen und dunklen Tab-Leisten gerendert. Das Zeichen blieb erkennbar, das Zahnrad verlor seine Konturen, und das Tab-Icon wurde durch das Zeichen ersetzt. Eine Zeichnung in ihrer kleinsten vorgesehenen Größe auf beiden Hintergrundarten testen und bei Bedarf eine einfachere Fassung für kleine Größen zeichnen.

Ein hochgeladenes SVG ist ein Dokument, kein bloßes Bild. Eine SVG-Datei kann Skripte und Verweise auf andere Dateien enthalten. Ihre Medientypregistrierung sagt ausdrücklich, dass Skriptsprachen ausführbaren Inhalt bilden. Sie warnt auch, dass bösartig konstruierte XML-Entitäten durch wiederholte Expansion große Mengen Speicher verbrauchen können. Wie viel davon ausgeführt wird, hängt von der Darstellung ab. SVG 2 erwartet, dass ein `object`-Element ohne Sandbox die Datei im dynamischen interaktiven Modus mit Skripten ausführt. `img` und CSS-Hintergrundbilder verwenden dagegen den sicheren animierten Modus ohne Skripte und externe Verweise. SVG aus fremden Quellen nur über `img` anzeigen, Uploads von einer separaten Domain ausliefern oder sie vor dem Speichern anhand einer Positivliste von Elementen und Attributen bereinigen.

Weiterführende Lektüre

- Scalable Vector Graphics (SVG) 2 — der Candidate Recommendation Snapshot des W3C vom 6. Oktober 2026 mit Status, Zusammenfassung und sämtlichen Kapiteln.
- Coordinate Systems, Transformations and Units — `viewBox`, `preserveAspectRatio` und das anfängliche Koordinatensystem.
- Paths — die Pfadbefehle, absolute und relative Koordinaten sowie der Umgang mit fehlerhaften Pfaddaten.
- Painting: Filling, Stroking and Marker Symbols — `fill`, `stroke`, ihre Anfangswerte und der Wert `currentColor`.
- Conformance Criteria — die fünf Verarbeitungsmodi und die Zuordnung der Einbettungsarten.
- Document Structure — `title` und `desc` sowie die impliziten ARIA-Rollen von SVG-Elementen.

Glossar

Für Ihre tatsächlichen Rahmenbedingungen auswählen

Beschreiben Sie den Arbeitsablauf, die vorhandenen Schnittstellen und die Nutzer, die die Veränderung benötigen. Wir können eine prüfbare Lieferung definieren und die Übergabe so gestalten, dass Ihr Team die gelieferten Bestandteile warten kann.

Preise ansehen Kontakt aufnehmen →

Die Begriffe praktisch nutzbar machen

Fragen Sie, wie „Was ist SVG?“ auf Ihre Organisation zutrifft. Ein konkretes Beispiel aus Ihrem bestehenden System reicht aus, um das Gespräch zu beginnen.

Unsere Leistungen entdecken

welcome@engineer.company +45 50 10 61 84 Kopenhagen, Dänemark

Nehmen Sie diese Seite mit

Kontaktkarte

Mit Sorgfalt und Liebe gestaltet von D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Copyright © 2025 – heute · Engineer ApS · Alle Rechte vorbehalten.

Gedruckt von https://engineer.company/de/glossary/svg/