Engineer

SVG

ett XML-språk för grafik som kan skalas

Vad det är

SVG, kort för Scalable Vector Graphics, är ett språk som beskriver tvådimensionell grafik som former och inte som ett rutnät av pixlar. Filen säger vad som ska ritas: en cirkel här, en linje från en punkt till en annan, en kontur fylld med blått. Webbläsaren räknar själv ut pixlarna i den storlek som teckningen visas i, och det är därifrån skalningen kommer: det finns inga pixlar i filen som behöver dras ut.

Specifikationen ges ut av W3C. Den senaste versionen, SVG 2, är en Candidate Recommendation Snapshot daterad den 6 oktober 2026, kan tidigast bli en Recommendation den 1 december 2026 och bygger vidare på SVG 1.1 Second Edition. Som specifikationen själv sammanfattar det vilar SVG på XML, kan blanda vektorformer med rasterbilder och går att forma med stilmallar, styra med skript och animera. En teckning kan stå för sig själv, ligga inuti en HTML-sida eller bäddas in i ett annat XML-språk.

Ordförrådet är litet. Sex grundformer, `rect`, `circle`, `ellipse`, `line`, `polyline` och `polygon`, klarar de enkla fallen, och en bana (`path`) ritar allt annat utifrån en kort sträng av kommandon. `text` placerar riktig text som går att markera, en grupp (`g`) låter flera former dela en stil eller en transformation, och en gradient eller ett mönster kan måla en form i stället för en enda färg. Filtereffekter, som var ett kapitel i SVG 1.1, har nu en egen specifikation. Det inledande koordinatsystemet har origo uppe till vänster, där x växer åt höger och y växer nedåt.

Specifikationen definierar SVG utifrån en dokumentobjektmodell och inte utifrån ett visst filformat. En teckning kan vara en egen fil, serverad som `image/svg+xml` med filändelsen `.svg`, eller ett `svg`-element skrivet direkt i en HTML-sida. En fristående fil måste deklarera SVG:s namnrymd. I HTML lägger parsern själv elementen i rätt namnrymd, så det behövs inget `xmlns`-attribut.

SVG är inte ett rasterformat som PNG eller JPEG, som lagrar en färg för varje pixel och blir suddiga när de förstoras. Ett fotografi hör hemma i ett av dem. SVG passar för det som är ritat: ikoner, logotyper, diagram, kartor och ritade bokstäver. En SVG kan också rymma en rasterbild, och det är det som specifikationen kallar blandad grafik.

Använd begreppet i ett beslut

Använd ”Vad är SVG?” i ett konkret beslut. Vi kan knyta begreppet till systemet ni bygger eller köper. Gränssnitt som laddar snabbt, fungerar i mobilen och beter sig likadant i alla webbläsare dina kunder råkar använda.

Se våra tjänster Hör av er →

Granska antagandena bakom ordet

Använd förklaringen och det länkade arbetet för att formulera frågor till projektet. Vi kan undersöka antagandena tillsammans innan ett verktyg väljs. Styrde full stack-utvecklingen av GIS-kartor — kartutveckling i full stack, från databas till webbläsare

Styrde full stack-utvecklingen av GIS-kartor Hör av er →

Varför det spelar roll

En fil räcker för alla skärmar. En ikon som ritats en gång är skarp på en äldre skärm, på en telefon med flera enhetspixlar per CSS-pixel och på papper, utan separata exporter i varje storlek som någon måste hålla i takt. Enkel grafik ger dessutom små filer, eftersom den beskriver några få konturer i stället för att lagra tusentals pixlar, och specifikationen registrerar `.svgz` för en SVG-fil som komprimerats med gzip.

Eftersom märkningen är text passar den in i hur ett team redan arbetar. En designers export kan läsas i en kodgranskning och jämföras rad för rad i versionshanteringen, och många diagram genereras av ett skript direkt ur data. När teckningen står direkt i sidan är dess element en del av DOM:en: CSS styr deras utseende, ett skript kan ändra dem, och de följer hovring, fokus och läsarens färgtema som allt annat på sidan.

Den bär också betydelse som en bitmappsbild inte kan bära. Text i en SVG förblir text, och i ett bearbetningsläge som tillåter interaktion kräver specifikationen att webbläsaren låter läsaren söka i den och markera den. Ett `title` -element ger en teckning, eller en enskild del av den, ett tillgängligt namn, och SVG 2 ger `svg` -elementet den implicita ARIA-rollen `graphics-document` . Ett diagram ritat i SVG kan ha namngivna delar som en skärmläsare når en i taget. Samma diagram exporterat som PNG är en enda bild med ett enda textalternativ.

Så ser det ut i praktiken

Här är en liten pil i en ram med rundade hörn, skriven direkt i en HTML-sida:

```
<svg viewBox="0 0 120 40" width="240" height="80">
  <title>Nästa steg</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` ger teckningen egna koordinater: fyra tal för vänsterkanten, överkanten, bredden och höjden på det område som ska visas, här 120 enheter brett och 40 högt. `width` och `height` begär 240 gånger 80 CSS-pixlar, så varje enhet ritas två pixlar bred. Ändra de två talen till 60 och 20 så krymper hela teckningen med, linjetjocklekarna inräknade, eftersom ingenting i den är mätt i pixlar.

Gruppen sätter färgerna en gång. Fyllning, linje och linjetjocklek ärvs, så både ramen och pilen hämtar dem från `g`. `rect` ritar ramen, och `rx` rundar dess hörn. Banan läses som instruktioner till en penna: `M 20 20` flyttar pennan till en punkt utan att rita, `H 92` drar en vågrät linje fram till x-koordinaten 92, det andra `M` lyfter pennan och sätter ned den vid pilspetsens övre ände, och de två `L`-kommandona ritar spetsens två streck. Kommandon med versal tar absoluta koordinater och kommandon med gemen relativa, så samma bana kan skrivas som `m 20 20 h 72 m -12 -10 l 20 10 l -20 10`, där varje steg mäts från där pennan stannade.

`currentColor` ger linjen värdet av CSS-egenskapen `color`, som ett `svg`-element i sidan ärver från HTML-koden runt omkring. Lägg pilen i en länk så ritas den i länkens färg. Byt sidan till mörkt tema så följer den med, utan en extra fil. `title` ritas inte. Elementet ger teckningen dess tillgängliga namn, och SVG 2 konstaterar att webbläsare ofta visar det som ett verktygstips när pekaren vilar över teckningen.

Samma teckning som en egen fil behöver namnrymden på sitt rotelement, `<svg xmlns="http://www.w3.org/2000/svg" viewBox="0 0 120 40">`, och en sida visar den med `<img src="/images/next.svg" alt="Nästa steg">`. Då är den ett separat dokument. Arv i CSS går från förälder till barn, och rotelementet i det dokumentet har ingen

förälder, så `currentColor` inne i filen följer inte längre texten runt `img`. Textalternativet kommer nu från `alt`-attributet, och SVG 2 förväntar sig att webbläsaren visar filen i säkert animerat läge, utan skript och utan externa filer.

Webbplatsens tillgänglighetsredogörelse säger att sidorna är skrivna i HTML, CSS och SVG, med WAI-ARIA där HTML ensamt inte ger något ett namn, och att varje sida är komplett utan körbar JavaScript. Där är SVG, liksom i den här artikeln, ett av språken som en sida skrivs i, bredvid märkningen och stilmallarna.

Var det går fel

En linjeteckning blir en massiv figur. Startvärdet för `fill` är svart, och startvärdet för `stroke` är `none`, alltså ingen linje. En öppen bana som var tänkt som en linje, till exempel en pilspets eller en bock, fylls som om ändarna vore sammanfogade, och utan `stroke` ritas själva linjerna inte alls. Sätt `fill="none"` och en `stroke` på linjeteckningar, en gång, på gruppen som håller dem.

Ikonen försvinner i mörkt läge. Exporterade ikoner har ofta en fast färg, `fill="#1a1a1a"`, som förblir mörk mot en mörk bakgrund. Ett sådant attribut är också den svagaste stil som finns: SVG 2 ger presentationsattribut specificiteten noll, så en bortglömd CSS-regel helt annanstans skriver över det utan att någon märker det. Använd `currentColor` för ikoner som står i text, och sätt färger från CSS i stället för i märkningen.

Teckningen skalas inte, eller skalas fel. Utan `viewBox` är teckningens enheter webbläsarens pixlar, så en ny bredd beskär bara teckningen eller lägger luft runt den, och `preserveAspectRatio` ignoreras helt. En negativ bredd eller höjd i `viewBox` är ett fel, och attributet räknas då som saknat. En bredd eller höjd på noll stänger av renderingen. Ge varje teckning en `viewBox` och låt `width` och `height`, eller CSS, bestämma storleken. Standardvärdet, `xMidYMid meet`, behåller proportionerna och centrerar teckningen i sin ruta.

Att kunna skalas är inte samma sak som att vara läsbar. Ingenting i en vektorteckning hindrar att den visas i 16 pixlar, men detaljer som syns tydligt i 200 kan flyta ihop till en fläck där. En portföljsida om hur företagets visuella identitet byggdes beskriver ett kugghjul i vektor, ritat för webbläsarfliken med argumentet att en kraftig siluett håller vid 16 pixlar. I det arbetet, som gjordes i juli 2026, visades kugghjulet och företagets märke sida vid sida i 16, 32 och 48 pixlar mot ljusa och mörka flikrader: märket förblev igenkännbart, kugghjulet

blev otydligt, och flikikonen gick över till märket. Testa en teckning i den minsta storlek den ska visas i, mot båda typerna av bakgrund, och rita en enklare version för små storlekar när den inte håller.

En uppladdad SVG är ett dokument, inte en bild. En SVG-fil kan innehålla skript och hänvisningar till andra filer. Registreringen av dess medietyp säger rakt ut att skriptspråk är körbart innehåll, och varnar för att illasinnat konstruerade XML-entiteter som expanderas om och om igen kan lägga beslag på stora mängder minne. Hur mycket av det som faktiskt körs beror på hur filen visas. SVG 2 förväntar sig att ett `object` -element utan sandlåda kör filen i dynamiskt interaktivt läge, med skript, medan `img` och bakgrundsbilder i CSS använder säkert animerat läge, utan skript och utan externa referenser. Visa SVG från främmande källor bara via `img` , servera uppladdningar från en separat domän, eller rensa dem mot en tillåtelselista över element och attribut innan de sparas.

Läs mer

- Scalable Vector Graphics (SVG) 2 — W3C:s Candidate Recommendation Snapshot från den 6 oktober 2026, med status, sammanfattning och alla kapitel.
- Coordinate Systems, Transformations and Units — `viewBox` , `preserveAspectRatio` och det initiala koordinatsystemet.
- Paths — banornas kommandon, absoluta och relativa koordinater och vad som händer med bandata som innehåller ett fel.
- Painting: Filling, Stroking and Marker Symbols — `fill` och `stroke` , deras startvärden och värdet `currentColor` .
- Conformance Criteria — de fem bearbetningslägena och vilket av dem varje sätt att bädda in får.
- Document Structure — `title` och `desc` och SVG-elementens implicita ARIA-roller.

Ordlista

Välj utifrån era faktiska förutsättningar

Beskriv arbetsflödet, befintliga gränssnitt och användarna som behöver förändringen. Vi kan definiera en testbar leverans och en överlämning som gör att teamet kan underhålla lösningen.

Se vad det kostar Hör av er →

Gör begreppen praktiska

Fråga hur ”Vad är SVG?” gäller i er organisation. Ett konkret exempel från ert nuvarande system räcker för att börja samtalet.

Se våra tjänster

welcome@engineer.company +45 50 10 61 84 Köpenhamn, Danmark

Ta med dig den här sidan

Visitkort

Omsorgsfullt skapad med kärlek av D.

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

Upphovsrätt © 2025 – nu · Engineer ApS · Alla rättigheter förbehållna.

Utskrivet från https://engineer.company/sv/glossary/svg/