Engineer

Mapbox

en kommersiell källa till kartplattor och kartstilar

Vad det är

Mapbox är både ett företag och den utvecklarplattform det säljer, och plattformen levererar delarna i en interaktiv karta som tjänster: grunddata, utseende, koden som ritar kartan och API:erna runt omkring. I kod dyker namnet ofta upp som Mapbox GL, efter ritbiblioteket. I sin egen kom-igång-guide beskriver företaget sig som en platsplattform som ger utvecklare byggstenar för kartor, navigering och sökning.

Kartdata ligger i tilesets, oftast uppdelade i vektorplattor (”vector tiles”) så att en klient bara hämtar det område och den zoomnivå som syns på skärmen. Mapbox underhåller egna globala grunddataset, och ett teams egna data kommer in i ett tileset via Mapbox Tiling Service (MTS) eller Uploads API. Utseendet bestäms av en stil. Den är ett JSON-dokument skrivet enligt Mapbox Style Specification, som namnger sina källor och listar lager, där varje lager plockar ut objekt ur en källa och anger hur de ska ritas. Stilar adresseras med en URL, t.ex. `mapbox://styles/mapbox/streets-v12`, och tas fram i kontots konsol med designverktyget Mapbox Studio.

Mapbox GL JS ritar resultatet i en webbläsare, och Maps SDK för iOS och Android gör samma sak i mobilappar. Enligt guiden till GL JS, som i skrivande stund gäller version 3.33.0, renderar biblioteket kartor i 2D och 3D med OpenGL genom att kombinera vektorplattor med stilregler i webbläsaren i stället för på en server. Det är därför en karta kan byta stil eller data medan någon använder den. Static Images API returnerar en bild av en karta, och Static Tiles API levererar rasterplattor gjorda av Studio-stilar till bibliotek som Leaflet. Varje karta skapas med en åtkomsttoken, och den token knyter kartan till ett Mapbox-konto.

Själva plattformatet, Mapbox Vector Tile-specifikationen, är öppet publicerat på GitHub och kodar varje platta som ett meddelande i Protocol Buffers.

Använd begreppet i ett beslut

Använd ”Vad är Mapbox?” i ett konkret beslut. Vi kan knyta begreppet till systemet ni bygger eller köper. Allt som händer på en plats — fartyg, fordon, tomter, sensorer — på en karta du kan söka i, mäta på och dela.

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. Gjorde pipelinen för geodata 50 gånger snabbare — 50x snabbare datapipelines för geodata

Gjorde pipelinen för geodata 50 gånger snabbare Hör av er →

Varför det spelar roll

En webbkarta behöver kod som ritar den och data att rita, från en leverantör eller från flera. Mapbox är svaret med en enda leverantör: data, stilar, rendering och drift under ett konto och på en faktura. Att välja Mapbox är därför ett beslut om hur stor del av kartstacken man vill hyra.

Vinsterna är verkliga. Någon annan ansvarar för att grunddata hålls aktuella. Kartans utseende ligger i ett stildokument med ett eget designverktyg, skilt från applikationens kod. Eftersom ritandet sker i webbläsaren kan ett tileset visas i flera stilar, ett lager kan filtreras eller färgas om vid ett klick och samma karta kan visa terräng i 3D. För en produkt där kartan är huvudskärmen lämnar det mycket arbete till leverantören.

Kostnaderna följer med samma konto. GL JS faktureras per kartladdning (”map load”), och prisguiden räknar en varje gång ett `Map` -objekt initieras på en sida. Panorering, zoomning, lager som slås av och på och stilbyten inne i kartan kostar inget extra, och en kartladdning varar högst 12 timmar. Den token som kartan debiteras via finns i varje besökares webbläsare. Och sedan version 2.0.0, som släpptes i december 2020, omfattas själva renderaren av kommersiella villkor: ändringsloggen för den versionen anger att GL JS inte längre ligger under 3-Clause BSD-licensen och att en uppgradering innebär att man godtar användarvillkoren hos Mapbox.

Flyttbarheten är därför en fråga för projektets början. Plattformatet är öppet. MapLibre GL JS började som en fork med öppen källkod av mapbox-gl-js, gjord före licensbytet. Dess 1.x-linje var tänkt som en direkt ersättare för Mapbox 1.x med öppen källkod, och enligt dess README har den utvecklats mycket sedan dess. Det går att lämna Mapbox, men det är ett flyttprojekt, och det är billigare att väga det innan en produkt byggs runt en leverantörs stilar och tilesets.

Så ser det ut i praktiken

Tre sidor i portfolion visar vad Mapbox gjorde i arbetet.

Som teamledare för utveckling av GIS-mjukvara 2022–2024 var uppgiften att leda fullstackutvecklingen av en GIS-karta som hade vuxit fram i bitar: spatiala data i PostgreSQL, ReactJS i frontend, NodeJS bakom och Mapbox som ritade kartan. Sidan sammanfattar Mapbox-delen på en rad. Mapbox stod för ritandet, och jobbet var att mata det med rätt data på rätt zoomnivåer i stället för allt på en gång. Resultatet blev en enda integrerad kartapplikation som blev en central del av plattformen och tog emot nya lager utan att ge vika.

Med samma titel, 2015–2022, fanns rollen med i designprocessen för projektens gränssnitt, med frontends i React, Angular och Vue ovanpå kartor i Mapbox och Leaflet. Sidans poäng är att kartan är den enkla delen. Det är i kontrollerna, filtreringen och flödet runt den som folk går vilse, i detaljer som att röra sig mellan våningsplan på en inomhuskarta.

Som dataingenjör 2022–2024 bestod jobbet i att åtgärda över 1 000 problem i geografiska data och tidsseriedata med GDAL, QGIS, ArcGIS, PostGIS och Bash. De rensade dataseten gick sedan in i interaktiva visualiseringar byggda med Mapbox, vilket gjorde data lättare att nå för slutanvändarna. Sidan tillskriver arbetet som helhet 35 % kortare körtider för spatiala databasfrågor.

Var det går fel

Det mesta som strular med Mapbox börjar när en karta möter verklig trafik, verkliga data eller en större uppgradering.

Fakturan växer med sidvisningarna i stället för med kartanvändningen. En kartladdning räknas när kartan initieras, så kostnaden följer hur ofta en sida skapar en karta. En ensidesapp som skapar `Map` på nytt vid varje ruttbyte betalar varje gång, och det gör även en karta långt ned på en lång sida som de flesta besökare aldrig når. Guiden från Mapbox om att hålla nere kostnaden för webbkartor föreslår att kartan skapas först när dess behållare rullas in i bild, att den läggs bakom en knapp av typen Load Map och att Static Images API används där ingen behöver panorera eller zooma. En enda kartinstans som lever vidare mellan rutterna löser fallet med ensidesappen.

En token som kopierats från sidan används någon annanstans. Tokenguiden slår fast att varje token på en publik webbsida eller i källkod syns för den som letar. Token med hemliga scopes, sorten `sk`, hör hemma på en server. En publik `pk`-token på en sida bör vara gjord för just den applikationen, med URL-begränsningar, och inte vara kontots publika standardtoken, som inte tar några. Guiden kallar URL-begränsningar ett skydd efter bästa förmåga som vilar på referer-huvudet som webbläsaren skickar. Efter en läcka roterar man: skapa en ny token, byt den i koden, driftsätt på nytt och radera sedan den gamla.

Kartan blir långsammare när datasetet växer. Klienten gör om en GeoJSON-källa till vektorplattor i farten, så en stor fil betyder att varje besökares webbläsare gör plattningen. Guiden om stora GeoJSON-källor listar åtgärderna: ladda filen från en URL i stället för från ett JavaScript-objekt, sänk källans `maxzoom` under standardvärdet 18 (12 passar de flesta punktdata), höj lagrets `minzoom` över 0, klustra täta punkter och skala bort oanvända egenskaper och decimaler efter den sjätte. Det sista steget krymper guidens exempelfil från 19,3 MB till 3,3 MB. Över 500 000 punkter föreslår guiden att källan delas upp eller att plattningen görs på servern med MTS eller Uploads API.

```
map.addSource("sites", {
    type: "geojson",
    data: "/data/sites.geojson", // en URL, inte ett objekt i minnet
    maxzoom: 12, // standardvärdet är 18
    cluster: true, // grupperar täta punkter vid låg zoom
});
map.addLayer({
    id: "sites",
    type: "circle",
    source: "sites",
    minzoom: 6, // dold under zoom 6
});
```

En större uppgradering ändrar både licensen och fakturan. När GL JS gick från 1.x till 2.0.0 flyttade biblioteket från BSD-licensen till användarvillkoren hos Mapbox, och faktureringsregeln ändrades. Under 1.x krävde en kartladdning en initiering plus en begäran om en platta som Mapbox själva hostar. Från och med 2.0.0 räcker initieringen, så en karta som bara ritade egna plattor börjar generera kartladdningar. Ändringsloggen ber läsaren titta på prisdokumentationen före uppgraderingen, och MapLibre GL JS är vägen för ett projekt som vill stanna på villkor med öppen källkod.

Läs mer

- Getting Started — företagets översikt över sina produkter för kartor, sökning och navigering.
- Mapbox GL JS pricing — vad som räknas som en kartladdning, sessionen på 12 timmar och de äldre reglerna.
- Token management — publika och hemliga token, scopes, URL-begränsningar och rotation.
- Working with large GeoJSON sources in Mapbox GL JS — zoomgränser, klustring, städning och plattning på servern.
- Mapbox Vector Tile specification — det öppna formatet bakom plattorna.
- MapLibre GL JS — renderaren med öppen källkod som började som en fork av mapbox-gl-js.

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 Mapbox?” 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/mapbox/