Engineer

Leaflet

невелика бібліотека для карт у браузері

Що це таке

Leaflet — це бібліотека JavaScript із відкритим кодом, яка перетворює один елемент вебсторінки на карту, яку можна панорамувати, масштабувати й натискати. Проєкт оцінює її розмір приблизно в 42 KB JavaScript і зазначає, що вона обходиться без зовнішніх залежностей. Її створив Володимир Агафонкін, тепер її розвиває спільнота учасників, а поширюється вона за ліцензією BSD 2-Clause.

Власний огляд проєкту каже, що Leaflet не намагається робити все для всіх і зосереджується на тому, щоб базові речі працювали добре. `L.map` приймає елемент і набір параметрів і стає картою. Шар плиток завантажує квадратні зображення за шаблоном URL із `{z}`, `{x}` і `{y}` і розкладає їх сіткою елементів зображень, а варіант WMS робить те саме з Web Map Service. Ламані, полігони, кола й прямокутники за замовчуванням малюються як SVG або на одному canvas, якщо карту створено з `preferCanvas`. Маркери — це іконки, зображення або звичайний `div`, до яких прив’язуються спливні вікна й підказки. Шар GeoJSON перетворює кожен об’єкт на один із цих шарів, а точку — на маркер, якщо не вказано інше. Кнопки масштабу, атрибуція, перемикач шарів і масштабна лінійка становлять елементи керування. Система координат за замовчуванням — EPSG:3857, яку довідник називає найпоширенішою для онлайн-карт.

Усе, що виходить за межі цього списку, дають плагіни. База плагінів проєкту описує Leaflet як задуманий бути якомога легшим і вказує на сотні сторонніх плагінів, розкладених за групами: векторні плитки, кластеризація, теплові карти, редагування геометрії, маршрутизація, геокодування.

На момент написання існують дві лінії випусків. Стабільна — 1.9.4, випущена 18 травня 2023 року, і саме її відображає довідник API на сайті проєкту. Попередній випуск 2.0.0-alpha.1 від 16 серпня 2025 року прибирає підтримку Internet Explorer і постачається як ES-модуль. Глобальний `L` у ньому лишається тільки в зібраному файлі `leaflet-global.js`, фабричні функції замінено конструкторами, тож `L.marker(latlng)` стає `new Marker(latlng)`, а введення обробляється через pointer events.

Застосуйте поняття до рішення

Застосуйте «Що таке Leaflet?» до конкретного рішення. Ми можемо пов'язати поняття із системою, яку ви створюєте або купуєте. Усе, що відбувається в певному місці, — судна, транспорт, ділянки, датчики — на карті, де можна шукати, вимірювати й ділитися.

Переглянути наші послуги Зв'язатися з нами →

Перевірте припущення за терміном

Використайте пояснення та пов'язані роботи, щоб сформулювати запитання для проєкту. Ми можемо перевірити припущення разом до вибору інструмента. Прискорила пайплайн географічних даних у 50 разів — 50x швидші пайплайни географічних даних

Прискорила пайплайн географічних даних у 50 разів Зв'язатися з нами →

Чому це важливо

Додати карту на сторінку — це два рішення, які легко сплутати з одним: який код малює карту в браузері і звідки беруться зображення карти. Leaflet тримає їх окремо, і вибір Leaflet здебільшого означає вибір тримати їх окремо.

Leaflet малює. Плитки надходять звідти, куди вказує шаблон URL, тож базова карта — окреме рішення з власною ціною, умовами й власником. Це може бути сервер плиток OpenStreetMap, поки функцію розробляють, комерційний постачальник у продакшні, сервер плиток, який проєкт тримає сам, або WMS міської адміністрації через `TileLayer.WMS`. Перехід між ними змінює один рядок у коді й текст атрибуції, а код застосунку навколо карти лишається як є.

Розмір і ліцензія визначають, де Leaflet почувається природно. Близько 42 KB і відсутність залежностей пасують сторінці, де карта — одна частина серед багатьох: формі, де людина ставить шпильку, чи адмінпанелі, що показує, де розташовані записи. Ліцензія BSD 2-Clause не ставить між бібліотекою і сторінкою ні облікового запису, ні ключа, ні лічильника.

Ціна малого ядра в тому, що важчі функції приходять окремими пакетами, кожен зі своїм супровідником і темпом випусків, і карта, що спирається на шість плагінів, успадковує шість шляхів оновлення. Альфа 2.0 показує, що це означає: її нотатки про перехід на нову версію радять користувачам старіших плагінів поліфіл V1, бо ці плагіни очікують глобальний `L` і фабричні функції. Карті, якій потрібні векторні плитки зі стилями в браузері на кожному рівні масштабу, краще підходить бібліотека, побудована довкола них, і це рішення варто ухвалити раніше, ніж шість плагінів триматимуть карту.

Як це виглядає на практиці

Leaflet з'являється на двох сторінках цього сайту, обидві — з ролі керівниці команди розробки GIS-програмного забезпечення у 2015–2022 роках, і на обох він ділить фронтенд із Mapbox.

Роль охоплювала постачання понад тридцяти GIS-проєктів від розробки й розгортання до підтримки. На фронті цих проєктів стояли вебкарти на Mapbox GL і Leaflet, іноді з використанням API ArcGIS чи HERE, а рівень застосунку писали на JavaScript, Python, PHP і SQL. Під ними лежав PostgreSQL із PostGIS, а GDAL/OGR переносив дані між shapefile, GeoJSON, GeoTIFF, KML і векторними плитками. Відповідальність сягала далі за запуск, аж до підтримки в продакшні.

Та сама роль брала участь і в проєктуванні інтерфейсів цих проєктів. Це були фронтенди на React, Angular і Vue поверх карт Mapbox і Leaflet, і сторінка робить зауваження, яке варто тримати поруч із будь-якою картографічною бібліотекою: у вебкартографічному продукті карта — найпростіша частина, а губляться люди в елементах керування, фільтрації та послідовності дій навколо неї. Зручність залежала від деталей, як-от того, як фільтрувався набір даних і як людина переходила між поверхами на карті приміщення, і більшу частину різниці сторінка пояснює тим, що ці питання ставили під час дизайну, а не після релізу.

Де виникають проблеми

Leaflet досить малий, щоб більшість його збоїв походила зі сторінки навколо нього або від сервісів, на які він спрямований.

Карта показується як сірий прямокутник або смужка розкиданих плиток. Leaflet бере розмір від свого контейнера. Посібник Quick Start називає дві речі, які треба зробити до запуску будь-якого коду карти: підключити CSS-файл Leaflet перед JavaScript і задати контейнеру визначену висоту. `div` без висоти дає карту без висоти. Підступніший випадок — карта, створена у вкладці чи діалозі, які в той момент приховані: вона вимірює себе як нуль і зберігає цей розмір, коли з'являється. Leaflet 1.x стежить за зміною розміру вікна браузера через `trackResize`, тож контейнер, що змінює розмір сам, лишається непоміченим, доки код не викличе `invalidateSize()`, який довідник радить викликати після зміни розміру карти. Альфа 2.0 використовує `ResizeObserver` замість події resize вікна.

Точки опиняються за тисячі кілометрів. GeoJSON, як його визначає RFC 7946, записує позицію з довготою першою. `LatLng` у Leaflet і масиви, які бібліотека приймає замість нього, мають широту першою. `L.geoJSON` сам перетворює одне в інше, тож помилка з'являється, коли координати витягають із GeoJSON вручну й передають прямо в маркер чи ламану. Точка в Копенгагені з переставленими координатами потрапляє в Аравійське море. Розпакуйте пару в іменовані змінні на межі:

```
const res = await fetch("/api/sites/42");
const site = await res.json();
const [lon, lat] = site.geometry.coordinates; // GeoJSON: lon, lat
L.marker([lat, lon]).addTo(map); // Leaflet: lat, lng
```

Панорамування смикається, коли даних стає більше. Із рендерером SVG кожна лінія чи полігон — окремий елемент документа, і кожен стандартний маркер — окреме зображення. Шар із багатьма тисячами об'єктів — це багато тисяч елементів, які браузер має пересувати під час кожного панорамування й масштабування. `preferCanvas: true` малює контури на одному canvas, але стосується лише контурів, тож точки треба перетворити на `L.circleMarker`, коло з фіксованим радіусом у пікселях, яке саме є контуром, через параметр `pointToLayer` шару GeoJSON. Далі відповідь — надсилати менше: плагін кластеризації або API, що повертає лише те, що потрапляє в поточну область перегляду на заданому масштабі.

Плитки перестають завантажуватися після запуску. Посібник Quick Start будує першу карту на сервері плиток OpenStreetMap і прямо каже, що ці плитки годяться для програмування, але перед продакшеном слід прочитати Tile Usage Policy. Ці правила вимагають видимої атрибуції, User-Agent, що

ідентифікує застосунок, і коректного Referer із вебсторінок, кешування відповідно до HTTP-заголовків, а масове й попереднє завантаження плиток забороняють. Вони не дають угоди про рівень обслуговування й дозволяють заблокувати доступ без попередження. Навантаженій продакшен-карті місце в постачальника, чиї умови пасують до її трафіку, або на власному сервері плиток, і з Leaflet такий перехід зводиться до шаблону URL і тексту атрибуції.

Що почитати далі

- Leaflet — огляд проєкту, перелік можливостей і перша карта в кілька рядків.
- Leaflet API reference — кожен клас, параметр і метод у 1.9.4, зокрема `preferCanvas`, `invalidateSize` і параметри GeoJSON.
- Leaflet Quick Start Guide — CSS, висота контейнера й перший шар плиток, крок за кроком.
- Leaflet 2.0 Alpha released — що змінюється у 2.0 і як мігрувати код і плагіни.
- Tile Usage Policy — умови користування серверами плиток openstreetmap.org.
- RFC 7946: The GeoJSON Format — стандарт GeoJSON, зокрема порядок позицій із довготою першою.

Словник

Обирайте за фактичними умовами

Опишіть робочий процес, наявні інтерфейси та користувачів, яким потрібна зміна. Ми можемо визначити перевірюваний результат і спосіб передачі, який дасть вашій команді змогу підтримувати його.

Дивитися, скільки це коштує Зв'язатися з нами →

Зробіть термінологію практичною

Запитайте, як «Що таке Leaflet?» стосується вашої організації. Конкретного прикладу з поточної системи достатньо для початку розмови.

Переглянути наші послуги

welcome@engineer.company +45 50 10 61 84 Копенгаген, Данія

Візьміть цю сторінку із собою

Візитівка

Дбайливо зроблено з любов'ю — D.

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

Авторське право © 2025 – дотепер · Engineer ApS · Усі права захищено.

Надруковано з https://engineer.company/uk/glossary/leaflet/