Engineer

URL

adressen som anger var något finns

Vad det är

En URL, förkortning av Uniform Resource Locator och i vardagligt tal kallad webbadress eller bara länk, är en textrad som både namnger en resurs och talar om för programvara hur den hämtas: vilket protokoll som ska talas, vilken maskin som ska tillfrågas och vad maskinen ska tillfrågas om. Klistra in en i en webbläsare, en terminal eller en konfigurationsfil, så kan var och en av dem agera på den utan ytterligare hjälp.

Uppbyggnaden bestäms av den generiska syntax som varje schema följer:

```
https://shop.example.com:8443/orders/42?view=full#items

scheme     https
host       shop.example.com
port       8443
path       /orders/42
query      view=full
fragment   items
```

Schemat säger hur resursen nås, här HTTP över TLS. Värden och den valfria porten säger var, sökvägen säger vilken resurs på den värden det gäller, och frågesträngen bär parametrar, oftast som par av formen `name=value` som fogas ihop med `&`. Fragmentet efter `#` pekar in i resultatet, mot en rubrik eller en position. Det tillhör klienten: webbläsaren utelämnar det från HTTP-förfrågans request-target. Det gör det inte hemligt för servern, eftersom klientkod kan läsa det och uttryckligen skicka det i en annan förfrågan.

Eftersom tecken som `/`, `?`, `#`, `&` och `=` har uppgifter i den strukturen måste data som skulle krocka med en avgränsare i den komponent som byggs procentkodas: skrivas som `%` följt av två hexadecimala siffror. Vilka tecken som behöver det beror på komponenten och dess syntax. Ett mellanslag blir `%20`, ett et-tecken inuti värdet på en frågeparameter `%26` och själva procenttecknet `%25`. Bokstäver utanför ASCII färdas på samma sätt, som sina byte i UTF-8, och det är så å, ä och ö eller ukrainska ord i en sökväg tar sig fram.

Är en URL samma sak som en URI? Nästan. En URI, en Uniform Resource Identifier, är vilken sträng som helst i den syntaxen som identifierar något. En URL är den sorten som dessutom anger hur det går att hitta, medan `urn:isbn:9780000000002` namnger en bok utan att ge något sätt att hämta den. RFC 3986 ber nya specifikationer att säga URI. WHATWG:s URL Standard, som webbläsarna följer, gick åt andra hållet: den bestämde sig för URL som det enda ordet och tog på sig att beskriva vad verkliga implementationer gör, med målet att ersätta RFC:n. De två är överens om vanliga adresser och skiljer sig åt i gränsfallen, och det är värt att minnas när en webbläsare och ett serverbibliotek tolkar samma udda sträng.

Använd begreppet i ett beslut

Använd ”Vad är en URL?” i ett konkret beslut. Vi kan knyta begreppet till systemet ni bygger eller köper. En webbplats som är snabb, lätt att hitta och går att redigera av din egen personal, utan en faktura för varje ändrad formulering.

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. Utvecklade 100 webbapplikationer med HTML/HTML5 — 100 byggda webbapplikationer

Utvecklade 100 webbapplikationer med HTML/HTML5 Hör av er →

Varför det spelar roll

En publicerad URL är ett löfte. Den hamnar i bokmärken, sökindex, mejl, tryckta broschyrer, chattloggar och andras sidor, och ingen av de kopiorna uppdaterar sig själv. Att ändra adressen senare bryter varenda en av dem, om inte den gamla adressen fortsätter att svara med en omdirigering till den nya.

Därför är det användbara beslutet ett tidigt beslut: vad adressen byggs av. Allt som sannolikt ändras är en dålig grund, vare sig det är en sidrubrik, ett kategoriträd, ett produktnamn eller en filändelse som avslöjar dagens ramverk.

En kort identifierare, eller en slug som lagras som data och ändras med avsikt, håller. Det gör också en plan för vad som händer när en adress ändå måste flytta: den gamla blir en permanent omdirigering i stället för en 404.

Det andra beslutet gäller vad som hålls utanför. RFC 9110 säger det rakt ut: adresser är gjorda för att delas, inte för att skyddas. De skrivs till loggar i servrar och proxyservrar, sparas i webbläsarens historik, kopieras in i webbanalys och skickas, beroende på policy, med i huvudet `Referer` till det som sidan laddar härnäst. Sessionsidentifierare, token för återställning av lösenord, API-nycklar och personuppgifter hör hemma i huvuden eller i förfrågans body.

Längden är en mindre fråga. HTTP-specifikationen rekommenderar att programvara godtar adresser på minst 8 000 oktetter, vilket är generöst, men en frågesträng som bär ett helt formulär eller ett serialiserat filter brukar slå i en kortare gräns någonstans längs kedjan innan den når dit.

Så ser det ut i praktiken

Arbetet på den här webbplatsen behandlar adresser som data som lever längre än texten på sidan. När 495 resultatsidor på tre språk genererades från en skrivskyddad SQLite-export vid det dokumenterade implementeringsskedet i augusti 2026 gällde det mest betydelsefulla beslutet adressen. Webbplatsen hade härlett en sidas URL från de inledande orden i dess engelska påstående, så en rättad mening flyttade sidan i tysthet och bröt varje länk till den. Adressen skrevs därefter som data, en rad per adress och resultat: den första raden var den aktuella, och varje senare rad var en avvecklad adress som webbplatsen publicerade som en omdirigering. Det dokumenterade resultatet var att varje adress som hade publicerats fram till det skedet fortfarande svarade.

En anteckning om mobilplånböcker visar vad som händer när en URL färdas inuti en annan. Phantom, Solflare och Backpack öppnar en dApp via universella länkar som bär måladressen i sin egen sökväg och ursprunget som gör begäran i parametern `ref`, båda skickade genom `encodeURIComponent`. Ett okodat `?` eller `&` i målet förstör tolkningen, och plånboken öppnas på sin startskärm i stället för i webbläsarfliken. Det går ännu sämre för ett gissat eget schema som `backpack://`: appen registrerar ingen rutt som matchar det, och användaren hamnar på installationssidan.

Adresser fungerar också som identitet. När 963 block av strukturerad data ersattes av en enda länkad graf skapades identifierarna för organisationen, webbplatsen och personen utifrån webbplatsens ursprung, och listsidor publicerar sina poster enbart som URL:er. Vid det dokumenterade implementeringsskedet i juli 2026 mättes det valet: att namnge posterna på portföljens indexsida skulle ha inneburit 107 fullständiga påståenden och en komprimerad sida som var 52 % större. Kontrollen som vaktar grafen säkerställer att varje URL som den nämner faktiskt byggdes.

Var det går fel

Fel i en URL brukar vara tysta. Adressen ser fortfarande rätt ut. Den pekar bara någon annanstans, eller avslöjar mer än det var tänkt.

Kodningen sker noll gånger eller två. En returadress som läggs rå i en frågesträng, som i `?next=/cart?item=4&size=m`, delas vid det okodade `&`, och servern ser `size=m` som en egen parameter. Att koda en redan kodad sträng gör `%20` till `%2520`, och sidan tar emot ett bokstavligt `%20`. Koda varje komponent exakt en gång, i samma ögonblick som den placeras i adressen, med den funktion som är avsedd för komponenter. I JavaScript är det `encodeURIComponent`. `encodeURI` lämnar `?`, `&` och `#` orörda eftersom den förväntar sig en hel adress.

En hemlighet åker med i frågesträngen. En token för återställning av lösenord eller en API-nyckel i adressen kopieras in i åtkomstloggar, webbläsarhistorik och webbanalys, och under en tillåtande policy för referrer skickas den i `Referer` till varje tredje part som sidan laddar något från. Lägg inloggningsuppgifter i ett huvud. När en token måste färdas i en länk, som i ett mejl för återställning, gör den engångs och kortlivad och sätt en `Referrer-Policy` som håller den hemma.

En sida svarar på fem adresser. `http` och `https`, med och utan `www`, med och utan avslutande snedstreck, `/About` och `/about`. Schema och värd är per definition skiftlägesokänsliga, medan sökvägen är skiftlägeskänslig om inte servern bestämmer något annat. Sökmotorer delar upp sidans ställning mellan kopiorna, cachar lagrar var och en för sig och webbanalysen räknar dem separat. Välj en form, omdirigera permanent från alla andra former till den och ange den som kanonisk URL.

En relativ länk tolkas relativt till fel basadress. En referens som `img/logo.png` ersätter allt efter det sista snedstrecket i den aktuella sökvägen. På `/docs/guide` blir den `/docs/img/logo.png`, och på `/docs/guide/` blir den `/docs/guide/img/logo.png`. Ett enda avslutande snedstreck flyttar varje relativ länk på sidan. Länkar som utgår från roten och börjar med `/` slipper problemet.

Logiken hänger på fragmentet. Omdirigeringar på serversidan, åtkomstloggar och webbanalys som läser HTTP-förfrågans request-target får varken `#` eller dess fragment. En klient kan skicka det separat när applikationen behöver det. Allt som måste nå servern som en del av den första förfrågan hör hemma i sökvägen eller i frågesträngen.

Läs mer

- RFC 3986: Uniform Resource Identifier (URI): Generic Syntax — grammatiken, procentkodningen, upplösningen av relativa referenser och normaliseringen.
- WHATWG URL Standard — tolkningsalgoritmen som webbläsarna implementerar, steg för steg.
- RFC 3987: Internationalized Resource Identifiers — hur adresser skrivna i vilket skriftsystem som helst översätts till procentkodade byte.
- RFC 8820: URI Design and Ownership — varför den som äger en värd, och ingen annan, bestämmer hur dess sökvägar ser ut.
- RFC 9110, section 17.9 — säkerhetsanmärkningen om känslig information i adresser.

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 en URL?” 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/url/