Engineer

Metadata

data som beskriver andre data, så de kan finnes

Hva det er

Metadata er data som handler om andre data: tittelen, datoen, språket, forfatteren, formatet og størrelsen som følger med et dokument, slik at mennesker og programmer kan sortere det, stole på det og handle ut fra det før de har lest det. Eldre standarder skriver ordet med bindestrek, meta-data, og i det daglige går metadataene under smalere navn: headerfelt i HTTP, egenskaper i en kontorfil, tagger i et bilde og front matter øverst i en Markdown-side.

Det er vanlig å dele dem i tre slag. Beskrivende metadata forteller hva noe handler om: tittel, opphav, emne og sammendrag. Strukturelle metadata forteller hvordan delene henger sammen: kapitler, sider og leserekkefølge. Administrative metadata forteller hvor noe kommer fra og hva det er lov å gjøre med det: når det ble laget og endret, av hvem, under hvilken lisens og i hvilket format. Dublin Core, femten elementer som ble standarden ISO 15836 i 2003, er den best kjente felles listen over beskrivende felt, og tittel, opphavsperson, dato, språk og rettigheter derfra dukker opp igjen under andre navn i nesten hvert format som har kommet siden.

En nettside bærer metadata i flere lag. Først kommer HTTP-svaret. RFC 9110 sier at et headerfelt kan endre eller utvide betydningen av meldingen, beskrive avsenderen, definere innholdet eller gi mer kontekst, og `Content-Type`, `Content-Language` og `Last-Modified` beskriver siden før en eneste byte av den er kommet fram. Inne i HTML-en ligger `<title>`, `lang`-attributtet og `<meta>`-elementer med navn som HTML-standarden definerer, f.eks. `description`, `author` og `generator`. Over dem igjen ligger vokabularer som andre parter leser: Open Graph-egenskaper til lenkeforhåndsvisninger og strukturerte sidedata for søkemotorer og assistenter.

Hva skiller så metadata fra data? Mer leseren enn bytene. En katalogpost er metadata om en bok, men for katalogen er postene selve datagrunnlaget. Tidspunktet et bilde ble tatt, er en detalj for den som ser på bildet, og råmateriale for den som kartlegger hvor bilder blir tatt. Metadata mot strukturerte data er det smalere spørsmålet, og svaret er at strukturerte data er metadata skrevet i et felles vokabular, slik at et program kan lese et felt uten å gjette hva det betyr.

Bruk begrepet i en beslutning

Bruk «Hva er metadata?» i en konkret beslutning. Vi kan knytte begrepet til systemet dere bygger eller kjøper. Regler og kontroller som fanger opp dårlige data på vei inn, og et klart svar på hvem som eier hvilket tall og hvor det kommer fra.

Se tjenestene våre Ta kontakt →

Undersøk antakelsene bak ordet

Bruk forklaringen og det tilknyttede arbeidet til å formulere spørsmål for prosjektet. Vi kan undersøke antakelsene sammen før et verktøy velges. Én infrastrukturplan for 14 avdelinger — 14 forskningsavdelinger på én felles infrastrukturplan

Én infrastrukturplan for 14 avdelinger Ta kontakt →

Hvorfor det er viktig

Hvert felt er et lite løfte, og beslutningen som teller, er hvor hvert felt henter verdien sin fra: en person som skriver den inn, versjonshistorikken, en kolonne i en database eller det filsystemet tilfeldigvis melder. Programmer handler på disse løftene uten å kontrollere dem. Det er hele poenget med metadata, og samtidig risikoen.

Cacher og søkeroboter fordeler innsatsen sin etter dem. En nettleser som har en kopi av en side, sender tilbake `Last-Modified` -datoen den fikk, og når ingenting er endret, får den et kort `304 Not Modified` i stedet for hele siden på nytt. RFC 9110 ber en server sende den datoen for enhver representasjon der den siste endringen kan fastslås på en rimelig og konsekvent måte, og vilkåret skjærer begge veier: der det ikke finnes noen dato som kan fastslås pålitelig, ber standarden heller ikke om en. Endringsdatoen i et sitemap gjør den samme jobben for en søkerobot som skal velge hva den henter på nytt.

Mennesker og hjelpemidler er også avhengige av metadata. `lang` -attributtet forteller en skjermleser hvilke uttaleregler den skal bruke, og tittelen er det en fane, et bokmerke og et søkeresultat viser. Rettigheter reiser på samme måte: en pakkebehandler kjenner lisensen til en avhengighet bare fordi avhengigheten selv har deklarert den, så en lisensgjennomgang bygget på den listen er nøyaktig så god som deklarasjonene under.

Hvert felt må holde seg sant, så en side med førti felt har førti sjanser til å ta feil. Noen felt leser ingen: HTML-standarden påpeker at mange søkemotorer ser bort fra navnet `keywords`, fordi det ble misbrukt til å fylle søkeresultater med spam. Og metadata lekker, for posisjonen i et bilde eller revisjonshistorikken i et dokument kan si mer enn innholdet. Færre felt, hvert fylt fra en kilde som ikke kommer i utakt med innholdet, slår som regel mange.

Slik ser det ut i praksis

Metadata dukker opp i tre prosjekter på dette nettstedet, hver gang i en annen type fil: et sitemap, et sett PDF-er og en avhengighetsgraf.

Sitemapet til ett nettsted påsto en gang at 172 av de 176 sidene var endret i samme øyeblikk. Datoene kom fra filenes endringstidspunkt, et stykke metadata filsystemet fører selv, og innholdseksporten flytter det for alle filene samtidig, siden den skriver hver av dem på nytt ved hver kjøring. Rettelsen går gjennom en reservekjede, først et eksplisitt felt, så historikken i git, så filen, og den dokumenterer fellen i sin egen konfigurasjon: det bokstavelige feltnavnet må stå i listen, ellers leser generatoren aldri feltet. Postene bak sidene holder nå dagen hver oppføring ble skrevet og revidert, atskilt fra årene selve arbeidet foregikk, og en dato ingen kjenner, blir stående tom, ut fra prinsippet om at «et presist, feil tall er verre enn et fraværende». På det dokumenterte implementeringstrinnet berørte en ny synkronisering etter en måned med endringer 8 filer av 186.

På det dokumenterte valideringstrinnet besto ni genererte PDF-dokumenter målt mot standarden for universell utforming alle 106 konfigurerte regler i tilgjengelighetsprofilen. Arbeidet omfattet et deklarert dokumentspråk og en reell tittel i metadataene, sammen med tagget struktur, alternativ tekst og en eksplisitt leserekkefølge. I den dokumenterte kjøringen feilet de samme dokumentene på nøyaktig én regel av 146 i arkivprofilen. Siden beskriver hvor nær resultatet var å oppfylle kravene, og oppgir regelen som feilet, fordi 145 bestått regler ikke innebærer samsvar med standarden.

Og da lisensene til 644 Rust-crates ble kontrollert ved hvert bygg, var den oppgitte grensen for revisjonen en grense i metadataene. Kontrollen leser lisensen hver crate har deklarert, deklarasjoner kan være feil eller ufullstendige, og kontrollen beviser ingenting om en crate som deklarerer seg selv feil. Den ble provosert til å feile med vilje to ganger, og det viser at kontrollen virker. Ingenting en kontroll gjør, kan vise at deklarasjonene den leser, er ærlige.

Hvor det går galt

Feilene nedenfor har ett trekk felles: alle kontroller som ikke leter etter dem, består.

Hver side ser endret ut etter hver produksjonssetting. En webserver som sender endringstidspunktet til en statisk fil som `Last-Modified`, eller en generator som daterer et sitemap på samme måte, oppgir øyeblikket filen ble kopiert, og en checkout, en eksport eller et containerbygg kopierer alle sammen. Cacher validerer alt på nytt, og søkerobotene lærer ingenting. Hent datoer fra versjonshistorikken eller fra et felt en person har skrevet, og la en dato ingen kjenner, forbli tom. I Hugo er kildene en liste som leses i rekkefølge, og feltets eget navn må stå i den:

```
enableGitInfo = true

[frontmatter]
date = ["date", ":git", ":fileModTime"]
lastmod = ["lastmod", ":git", ":fileModTime"]
```

Tittelen eller lenkeforhåndsvisningen viser ødelagte tegn. Tegnkodingen er metadata om hver eneste annen byte, og en parser som møter tekst før deklarasjonen, må gjette. HTML-standarden krever at elementet som deklarerer kodingen, får plass i sin helhet innenfor de første 1 024 bytene, og legger du det først i `<head>`, blir det ingenting å gjette på. På dette nettstedet leste URL-inspeksjonen i Bing (URL Inspection) en gang tittelen og begge Open Graph-strengene som Latin-1, fordi en lang tankestrek i tittelen kom før `charset`. HTTP-headeren oppga også UTF-8, men et verktøy som leser bytene uten headeren, har bare siden å gå etter. `charset` er nå det første elementet, og en kontroll bekrefter rekkefølgen på hver bygde side.

Den samme opplysningen er forskjellig fra sted til sted. Tittelen på en side bor i `<title>`, i `og:title`, i de strukturerte sidedataene og, for en PDF, i dokumentegenskapene. Språket bor i `lang`, i `Content-Language` og i `og:locale`. Skrevet inn hver for seg glir de fra hverandre, og hver mottaker tror på sin versjon. Regn ut hver verdi én gang, i én mal eller funksjon, og la hvert sted som skriver den ut, lese akkurat den verdien.

Et dokument røper mer enn forfatteren mente å publisere. Bilder har med seg kameraet, tidspunktet og ofte posisjonen. Kontordokumenter har med seg forfatternavn, redigeringshistorikk og stier til maler, og en `generator` -tag navngir programvaren som bygde siden. Fjern ved eksport alt ingen har valgt å publisere, og behandle hvert felt som tekst en fremmed kommer til å lese.

Les mer

- RFC 9110, section 6.3: Header Fields — hvordan HTTP sender metadata foran innholdet, og hva et headerfelt har lov til å gjøre.
- RFC 9110, section 8.8.2: Last-Modified — når en server bør sende en endringsdato, og hvorfor en dato i fremtiden blir byttet ut.
- HTML Standard: the meta element — standardnavnene for metadata og reglene for å deklarere tegnkoding.
- DCMI Metadata Terms — de femten Dublin Core-elementene og vokabularet som er bygget rundt dem.
- The Open Graph protocol — de fire egenskapene protokollen krever på hver side, og de valgfrie rundt dem.

Ordliste

Velg ut fra deres faktiske rammer

Ta med et representativt datasett, spørsmålene det skal besvare og dagens flaskehals. Vi kan avtale målinger av kvalitet, gjennomstrømning og gjenoppretting.

Se hva det koster Ta kontakt →

Gjør begrepene praktiske

Spør hvordan «Hva er metadata?» gjelder i organisasjonen deres. Et konkret eksempel fra dagens system er nok til å starte samtalen.

Se tjenestene våre

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

Ta med deg denne siden

Visittkort

Laget med kjærlighet av D.

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

Opphavsrett © 2025 – nå · Engineer ApS · Alle rettigheter forbeholdt.

Skrevet ut fra https://engineer.company/nb/glossary/metadata/