Engineer

構造化データ

ページの事実を機械が読める形で記述したデータ

どのようなものか

企業のお問い合わせページは、オフィスの場所を文章で伝えます。構造化データは、それを通り名、郵便番号、国を持つ `PostalAddress` としてもう一度記述し、ページを読むプログラムが、どの単語が町の名前かを推測せずに済むようにします。ページが文章で述べていることを、発行側と利用側が合意した語彙に従い、型を持つ記述として繰り返す手法です。検索エンジンなどのソフトウェアは、レイアウトを解釈せずに事実を読み取れます。

語彙には通常schema.orgを使います。Googleのドキュメントでも、検索向け構造化データの多くが使う語彙として挙げられています。Google、Microsoft、Yahoo、Yandexが設立し、2015年4月以降はW3Cのコミュニティグループが主な議論の場です。 `Organization` 、 `Person` 、 `Service` 、 `Review` 、 `DefinedTerm` などの型と、 `name` 、 `address` 、 `author` 、 `price` など、各型が持てるプロパティを定義しています。

記述には三つの構文が使えます。MicrodataとRDFaは、表示されるHTML要素そのものに属性を追加します。JSON-LDは記述を分離し、 `script` 要素の型を `application/ld+json` として、そこにJSONのブロックを置きます。バージョン1.1は、2020年7月16日に公開されたW3C勧告です。Googleは三つを同等に受け付け、大規模な運用で最も保守しやすいとしてJSON-LDを推奨しています。

```
<script type="application/ld+json">
    {
        "@context": "https://schema.org",
        "@type": "Organization",
        "@id": "https://example.com/#organization",
        "name": "Example Ltd",
        "url": "https://example.com/"
    }
</script>
```

このブロックで最も重要なのは `@id` の行です。JSON-LDはこれでノードを識別します。これを持たないノードは空白ノードであり、文書に依存しない識別子を持ちません。空白ノードの識別子は、一つの文書内の複数ブロックで共有できます。グローバルに使える識別子なら、ほかのページのどのブロックからでも `"publisher": {"@id": "https://example.com/#organization"}` と書けます。仕様ではこれをノード参照と呼びます。別の場所で説明されているものを、改めて説明する代わりに指し示す仕組みです。識別子が、散在するブロックを一つのグラフに変えます。

近い言葉にメタデータがあります。ページのhead内のタイトル、説明、Open Graphタグは、文書としてのページを記述します。一方、構造化データは、そのページが扱うもの、つまり運営する企業、執筆した人、提供するサービス、定義する言葉を記述します。

概念を判断に生かす

「構造化データとは？」を実際の判断に生かしましょう。その概念を、開発または購入するシステムに結び付けて考えるお手伝いをします。 お客様のAIアシスタントがサイトやアプリを見つけ、読み取り、操作できる状態を、最初の約束だけで済ませず、リリースごとに検証します。

サービスを見る お問い合わせ →

用語の背後にある前提を検証する

解説とリンク先の実績を参考に、プロジェクトについての質問を整理してください。ツールを選ぶ前に、前提を一緒に検討できます。 3言語の多言語Hugoサイトを構築 — 108 3言語・4形式を提供するテンプレートファイル

3言語の多言語Hugoサイトを構築 お問い合わせ →

なぜ重要なのか

判断すべきなのは、どこまで述べるかです。通常は、語彙が用意する項目をすべて使うのではなく、数を絞って完全に記述するのが誠実な答えです。構文上は、schema.orgが提供するすべてのプロパティを埋めてもかまいません。しかしGoogle自身の指針は逆です。不完全または誤ったデータで可能なすべてを埋めるより、推奨プロパティを少数に絞って完全かつ正確に示し、たとえ事実でも、読者がページ上で見られないものはマークアップしないよう求めています。

マークアップした事実はどれも、ページがすでに述べていることの二つ目のコピーであり、そのコピーは食い違う可能性があるためです。表示上の表だけ直してJSONに残した価格、機械だけに見せる評価、ページごとに少し違う住所表記。いずれも、人には一つのことを、プログラムには別のことを伝える原因になります。構造化データの負担は、書くことより、正しい状態を保つことにあります。役立つ問いは、ページがどの記述に責任を持てるか、それが表示する文章と同じ情報源から来ているかです。

得られるのは、読み取りやすさです。百回の言及を同じ企業のものとして結び付けられる検索エンジンやエージェントは、無関係な百の名前として読むものより多くを理解できます。指針に従うページは、追加情報を含む検索結果であるリッチリザルトの対象になる可能性がありますが、対象資格があっても表示は保証されません。変わるのは、機械がページの主題をどこまで理解できるかです。

実際の活用例

このサイトはクレジットで、ページ上で事実として確認でき、目に見えるものだけをマークアップすると約束しています。その一文を明記する必要があった理由は、構造化データの変更履歴に表れています。

2026年7月の改修前には、963個のブロックが同じ企業を1,671回記述し、安定した識別子は一つもありませんでした。そのため、あるページの組織を別のページの組織と同一だと判断する根拠が、利用側にありませんでした。現在のグラフは、サイトのオリジンから作る、組織、ウェブサイト、創業者の三つの識別子を基礎にしています。ほかのブロックは、再度説明する代わりに `@id` で参照します。識別子はすべての言語で共通です。テンプレートのコメントにも、デンマーク語用の別識別子を作ると、同じ企業登録番号を持つ別組織を宣言することになると書かれています。その段階では十六種類の型が記録されていました。リスト内の項目はURLだけで示しました。当時ポートフォリオ一覧にあった107件を詳しく記述すると、圧縮後のページが52パーセント重くなるためです。

いくつかの事実は、表示ページと同じ情報源から取得します。サービス料金は画面上の料金と同じデータファイルから取得するので、機械にも読者にも同じ料金を示します。仕事の対象期間は、その期間を説明する文章を出力するパーシャルから取得します。パンくずリストは目に見える経路をたどり、リストの位置は画面上の順序に従います。機械にだけ示す順位ではありません。ページ自身のノードはcanonicalリンクと同じアドレスを使い、マークアップとheadが同じことを述べるようにします。

記述しないものも、記述するものと同じくらい意識的に選びます。企業が自分の仕事について書いた文章をレビューとしてマークアップすると、自分の言葉に架空の著者を作ってしまうため、そうしていません。推薦文のテンプレートは、公開への同意が得られた項目に限り `Review` ノードを使い、評価点は追加しません。このページを含む各用語集ページは、一覧に表示する一行の短い定義を説明文にした `DefinedTerm` を持ちます。機械が読む定義と読者が見る定義は同じです。

ビルド後のサイトを検査し、すべてのブロックを解析できること、すべての識別子を解決できること、グラフが挙げるすべてのURLが実際に生成されていることを確認します。存在しないページを指すグラフは、品質ゲートを失敗させます。同じクレジットのページは、データを読むためのSchema Markup Validatorと、一つの検索エンジンがどう扱うかを調べるGoogleのRich Results Testも紹介しています。

問題が起きるところ

すべてのバリデーターを通っても、リッチリザルトにならない。 マークアップは有効なJSON-LDで、実在するschema.orgの型を使っているのに、通常の検索結果のままです。誰も付けていない評価、ページに表示されない質問、画面と違う価格など、読者に見えないものを記述しています。Googleは、指針に反するマークアップでは、構文が正しくても対象外になることがあると警告しています。ページに表示するものだけをマークアップし、両方を一つの情報源から生成して食い違いを防ぎます。

何百ものブロックが同じ企業を説明しながら、結び付いていない。 各ページが組織名、ロゴ、住所を繰り返し、ときには少しずつ異なるため、一つ直すにもすべてを探す必要があります。ノードが匿名だったり、同一の実体に対してページや言語ごとに別の識別子を作ったりしています。現実の各実体に、フラグメント付きの安定したURLから作る `@id` を一つ与え、一度だけ説明し、ほかの場所から参照します。

テンプレートの変更一つで、全ページのマークアップが壊れる。 余分な引用符、末尾のカンマ、空の値によって、すべてのブロックが無効になり、利用側は何も使えなくなります。Googleのガイドも、テンプレート処理によってデプロイ後にマークアップが壊れることがあると警告しています。波括弧の間に値を貼り込む形でJSONを文字列として組み立てると、フィールドの欠落が空文字列や壊れた構造を残します。このサイトのテンプレートがHugoの `jsonify` で行うように、記述をデータ構造として作ってシリアライズします。値のないプロパティは空で出力せず省き、ビルド後の出力にある全ブロックを解析します。

グラフが、存在しないページを挙げる。 マークアップのURLが404、ステージング用ホスト、ミラーに向かっています。アドレスを手入力したか、リンクやcanonicalとは別の規則で作ったことが原因です。ページのcanonicalアドレスを作る関数から、マークアップのすべてのURLも導出し、それぞれがビルドされているか確認します。

参考資料

- schema.org — 共通の型とプロパティの語彙で、各型の説明ページがあります。
- JSON-LD 1.1 — ノード識別子とノード参照を含む構文のW3C勧告です。
- schema.org: DefinedTerm — 各用語集ページが、説明する言葉を記述するために使う型です。
- Google Search Central: 構造化データの概要 — 三つの構文、可視性の規則、リッチリザルトのテストを説明しています。
- Schema Markup Validator — ページの構造化データをschema.orgの語彙に照らして読み取ります。

用語集

実際の制約に合わせて選ぶ

業務の流れ、既存のインターフェース、変更を必要とする利用者をお知らせください。検証可能な納品内容と、お客様のチームが保守を続けられる引き継ぎ内容を定められます。

料金を見る お問い合わせ →

用語を実務に結び付ける

「構造化データとは？」がお客様の組織にどう当てはまるかをご相談ください。現在のシステムの具体例が一つあれば、話を始められます。

サービスを見る

welcome@engineer.company +45 50 10 61 84 デンマーク、コペンハーゲン

このページを持ち歩く

連絡先カード

心を込めて丁寧に制作： D.

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

著作権 © 2025 – 現在 · Engineer ApS · すべての権利を留保します。

印刷元： https://engineer.company/ja/glossary/structured-data/