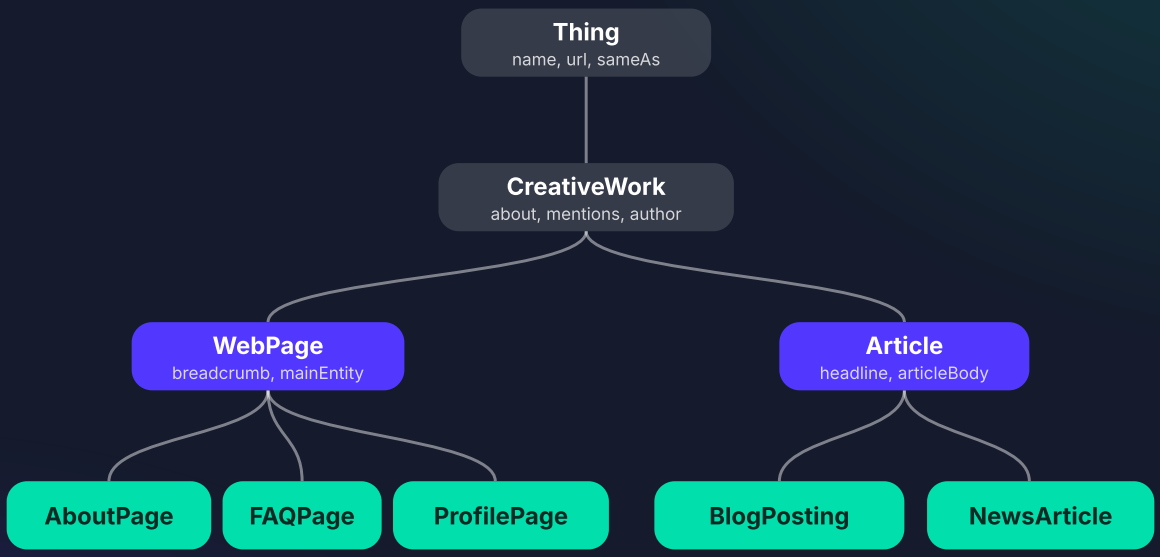
Schema Inheritance Guide

How schema.org types inherit properties, how every page on a site should inherit one shared graph, and page-type templates you can hand to a developer.

Comprehensive guide · v1.0 · built from a 2023 in-house schema guide

EVERY TYPE INHERITS THE PROPERTIES OF THE TYPES ABOVE IT



2	12	14
kinds of inheritance	page-type templates	2023 items corrected

Ameneh Saeednia
Senior SEO and Generative Search Strategist

About this guide

One graph, many page types

In 2023 I contributed to an in-house schema guide for a group of large publishing sites. It had the right idea: **define site-wide schema once, then extend it by page type**. This guide rebuilds that idea for 2026, corrects what has changed, and explains the inheritance underneath it.

1

Who it's for

SEO leads, developers and content managers who need **one consistent schema system** across many page types and templates.

2

What you'll get

The type tree, a **baseline site graph**, **12 page-type templates**, worked JSON-LD, a list of corrections and a QA checklist.

3

What it isn't

A promise of rich results. Markup makes pages **eligible** for some features and helps describe content. Google decides what to show. [1]

Section	Page
Part 1: Two kinds of inheritance	3
Part 2: The type tree you'll actually use	4
Part 3: What a subtype inherits	5
Part 4: The site-wide graph	6
Part 5: Page-type templates	7 to 8
Part 6: Worked examples	9 to 10
Part 7: Corrections to the 2023 guide	11
Part 8: Workflow and QA checklist	12 to 13

Part 1

Two kinds of inheritance

“Inheritance” in schema means two different things. Most implementation problems come from **mixing them up**.

1. Type inheritance (the vocabulary)

schema.org types form a **hierarchy**. Every type is a subtype of **Thing**, and inherits all properties of the types above it. [2]

AboutPage is a WebPage, which is a CreativeWork, which is a Thing. So AboutPage can use **breadcrumb** (from WebPage), **about** (from CreativeWork) and **sameAs** (from Thing).

Rule: **use the most specific type** that is true, and you keep every property of its parents.

2. Graph inheritance (your site)

Every page on a site shares the same **Organization** and **WebSite**. Each page **references** them by **@id** instead of repeating them.

A page's WebPage node points to the WebSite with **isPartOf**. The WebSite points to the Organization with **publisher**.

Rule: **define once, reference everywhere**. Change the logo once and every page inherits it.

Why this matters

Type inheritance decides **which properties are allowed**. Graph inheritance decides **whether your site describes one organization or a thousand slightly different ones**. You need both.

Part 2

The type tree you'll actually use

schema.org has hundreds of types. A publishing or travel site uses a small, predictable set. These are the branches that matter, with **parents on the left**. [2]

Branch	Path in the hierarchy	Use it for
Pages	Thing > CreativeWork > **WebPage** > AboutPage, ContactPage, FAQPage, ProfilePage, CollectionPage, ItemPage, QAPage, SearchResultsPage	The page itself, one per URL
Articles	Thing > CreativeWork > **Article** > NewsArticle, TechArticle, Report; Article > SocialMediaPosting > **BlogPosting**	Editorial content
Reviews	Thing > CreativeWork > **Review** > CriticReview, UserReview, ClaimReview, Recommendation	Assessments of a specific item
Organizations	Thing > **Organization** > LocalBusiness, Corporation, NewsMediaOrganization, SportsOrganization > SportsTeam	Publisher, brand, businesses
People	Thing > **Person**	Authors, reviewers, experts
Places	Thing > **Place** > LandmarksOrHistoricalBuildings, TouristAttraction, AdministrativeArea > City, State	Destinations and locations
Events	Thing > **Event** > SportsEvent, Festival, BusinessEvent	Dated events
Software	Thing > CreativeWork > **SoftwareApplication** > WebApplication, MobileApplication	Calculators, apps, tools

Reading the tree

A type can have **more than one parent**. TouristAttraction, for example, sits under Place. When in doubt, open the type on schema.org: the page lists **every inherited property, grouped by parent**.

Part 3

What a subtype inherits

AboutPage defines **no properties of its own**. Everything it can use is inherited. That's typical: subtypes mostly add **meaning**, not fields.

Inherited from	Useful properties	Typical use on an About page
Thing	name, url, description, image, sameAs, identifier, alternateName	Page name and URL
CreativeWork	about, mentions, author, publisher, datePublished, dateModified, isPartOf, audience, inLanguage	**about** the Organization, **isPartOf** the WebSite
WebPage	breadcrumb, mainEntity, primaryImageOfPage, lastReviewed, reviewedBy, relatedLink, specialty, speakable, mainContentOfPage	**mainEntity** the Organization, **lastReviewed** date
AboutPage	(none of its own)	The type itself tells machines what the page is for

A correction from 2023

The 2023 guide labelled **specialty** and **relatedLink** as “from AboutPage.” They're **WebPage** properties, inherited by every page subtype. The difference matters when a developer builds a shared template.

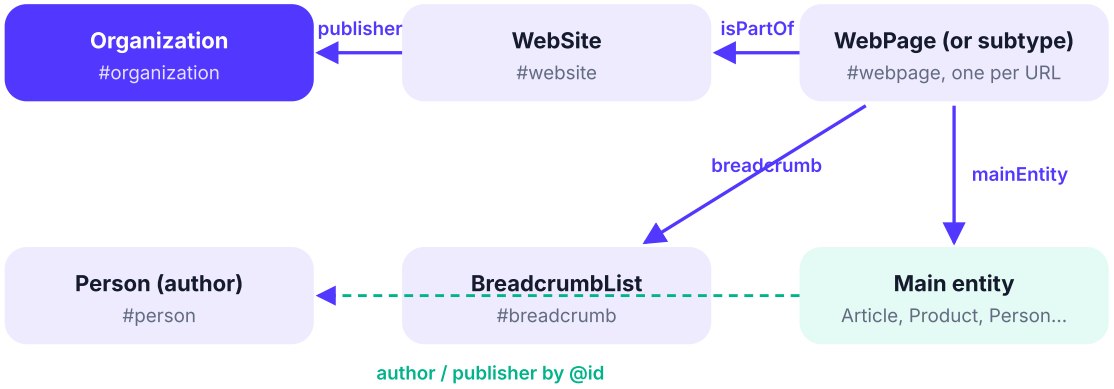
The practical rule

Build **one WebPage template** with the inherited fields. Page subtypes (AboutPage, ContactPage, FAQPage) swap only the **@type** and the **mainEntity**.

Part 4

The site-wide graph

Every page should output the same four connected nodes, plus its own main entity. **Stable @id values** are what tie them together.



Node	@id pattern	Defined	Referenced by
Organization	https://example.com/#organization	Once, site-wide	WebSite publisher, article publisher, provider, brand
WebSite	https://example.com/#website	Once, site-wide	Every WebPage isPartOf
WebPage	[page URL]#webpage	On each page	mainEntityOfPage, breadcrumb owner
Person	[profile URL]#person	On the profile page	author, reviewedBy on every article

Yoast and similar plugins

Plugins such as Yoast already output a connected graph of WebSite, Organization, WebPage and BreadcrumbList with @id values. **Extend that graph** with custom pieces; don't output a second, disconnected one.

Part 5

Page-type templates (1 of 2)

Each template inherits the baseline graph and changes only **the page type, the main entity and the extra properties**.

Page type	Page @type	mainEntity	Add	Google feature
Homepage	WebPage	(optional) Organization	WebSite **name** and **alternateName**	Site names [3]
About	AboutPage	Organization	about, logo, sameAs, foundingDate, contactPoint	Organization details [4]
Contact	ContactPage	Organization	contactPoint with telephone, email, areaServed	Organization details
Author / team member	ProfilePage	Person	jobTitle, worksFor, knowsAbout, sameAs	Profile page [5]
Article / blog post	WebPage	Article or BlogPosting	author by @id, datePublished, dateModified, about, mentions	Article [6]
News	WebPage	NewsArticle	Same as article, plus dateline where relevant	Article

Bold marks the properties that most often go missing in audits.

Part 5, continued

Page-type templates (2 of 2)

Page type	Page @type	mainEntity	Add	Google feature
Editorial review	WebPage	Review	itemReviewed (supported type), reviewRating, author, positiveNotes	Review snippet [7]
Category / hub / HTML sitemap	CollectionPage	(optional) ItemList	about, hasPart or ItemList of child pages	None specific
FAQ	FAQPage	Question (each)	acceptedAnswer per Question	Limited since 2023 [8]
Policy, terms, disclaimer	WebPage	CreativeWork	name, datePublished, dateModified, publisher	None
Event or match page	WebPage	Event or SportsEvent	startDate, location, homeTeam, awayTeam	Event, where eligible
Tool or calculator	WebPage	WebApplication	applicationCategory, operatingSystem, offers if free	Software app, where eligible

FAQ in 2026

Since August 2023, Google shows FAQ rich results mainly for **well-known government and health sites**. FAQPage markup is still valid; just don't promise the feature. [8]

Tools are software

The 2023 guide suggested **Product** for an odds calculator. A free calculator is a **WebApplication**; Product is for things you can buy.

Part 6

Worked example: the baseline graph

This block, or your plugin's equivalent, appears on **every page**. Other pages extend it.

Organization, WebSite, WebPage and BreadcrumbList, connected by @id

```
{
  "@context": "https://schema.org",
  "@graph": [
    {"@type": "Organization",
     "@id": "https://example.com/#organization",
     "name": "Example Travel",
     "alternateName": "ExTravel",
     "url": "https://example.com/",
     "logo": {"@type": "ImageObject",
              "url": "https://example.com/logo.png"},
     "sameAs":
["https://www.linkedin.com/company/example"]},
    {"@type": "WebSite",
     "@id": "https://example.com/#website",
     "url": "https://example.com/",
     "name": "Example Travel",
     "publisher": {"@id":
"https://example.com/#organization"}},
    {"@type": "WebPage",
     "@id": "https://example.com/rome/#webpage",
     "url": "https://example.com/rome/",
     "name": "Rome Tours",
     "isPartOf": {"@id":
"https://example.com/#website"},
     "breadcrumb": {"@id":
"https://example.com/rome/#breadcrumb"}},
    {"@type": "BreadcrumbList",
     "@id": "https://example.com/rome/#breadcrumb",
     "itemListElement": [
       {"@type": "ListItem", "position": 1, "name":
"Home",
        "item": "https://example.com/"},
       {"@type": "ListItem", "position": 2, "name":
"Rome Tours",
        "item": "https://example.com/rome/"}]}
  ]
}
```

What to notice

Organization and **WebSite** have site-wide @id values that never change.

alternateName (not “alternativeName”) gives Google a recognized short name for the site. [3]

publisher connects the WebSite to the Organization. The 2023 guide used isPartOf here, which describes the wrong relationship.

Each page gets its own **#webpage** and **#breadcrumb** @id based on its URL.

Nothing about the page's subject is repeated here; that belongs to **mainEntity**.

Worked examples: About page and blog post

Both inherit the baseline graph and add only what's specific to the page.

AboutPage: inherits WebPage, references the Organization

```
{
  "@context": "https://schema.org",
  "@type": "AboutPage",
  "@id": "https://example.com/about/#webpage",
  "url": "https://example.com/about/",
  "name": "About Example Travel",
  "isPartOf": {"@id":
"https://example.com/#website"},
  "mainEntity": {"@id":
"https://example.com/#organization"},
  "about": {"@id":
"https://example.com/#organization"},
  "lastReviewed": "2026-10-01",
  "reviewedBy": {"@id":
"https://example.com/team/ana/#person"}
}
```

Blog post: WebPage with a BlogPosting main entity

```
{
  "@context": "https://schema.org",
  "@graph": [
    {"@type": "WebPage",
     "@id": "https://example.com/blog/rome-with-
kids/#webpage",
     "url": "https://example.com/blog/rome-with-
kids/",
     "isPartOf": {"@id":
"https://example.com/#website"},
     "mainEntity": {"@id":
"https://example.com/blog/rome-with-kids/#article"}},
    {"@type": "BlogPosting",
     "@id": "https://example.com/blog/rome-with-
kids/#article",
     "headline": "Rome with Kids: A Three-Day Plan",
     "datePublished": "2026-10-20",
     "author": {"@id":
"https://example.com/team/ana/#person"},
     "publisher": {"@id":
"https://example.com/#organization"},
     "about": {"@type": "City", "name": "Rome",
               "sameAs":
"https://www.wikidata.org/wiki/Q220"}}
  ]
}
```

About page

AboutPage replaces WebPage. Its mainEntity and about both point to the **same Organization @id**, so there's still only one Organization on the whole site.

Blog post

The **BlogPosting** inherits from Article and CreativeWork. Author and publisher are **references**, not copies. **about** names the real subject with a verified sameAs.

Part 7

Corrections to the 2023 guide

Fourteen items from the original guide, checked against schema.org and current Google documentation.

2023 guide said	In 2026
“alternativeName” on WebSite	The property is **alternateName**.
specialty and relatedLink “from AboutPage”	They're **WebPage** properties, inherited by all page types.
WebSite isPartOf Organization	Use **publisher**. isPartOf is for parts of a larger work.
SearchAction for a sitelinks search box	Google **retired** the sitelinks search box in November 2024. [9]
FAQPage for rich results	Valid markup, but FAQ rich results are **limited** since 2023. [8]
Product for an odds calculator	Use **WebApplication**.
Game and Play for a poker site	Game describes a game; **Play is a theatre play**. Neither describes a gambling site.
ClaimReview on review pages	ClaimReview is for **fact-checking claims**, not product or site reviews.
termsOfService as a page type	It's a **property**, not a type. Mark up the policy page as WebPage with a CreativeWork mainEntity.
about: {“@type”: “Thing”, ...}	Use the **most specific type** that's true: Place, Organization, Game, Product.
“Knowledge panels prefer the About page's first paragraph”	**Not documented** by Google. Keep the About page clear, but don't promise this.
Recommendation and Offer on every review	Only where the page **visibly** recommends or sells something.
One template per site	Keep one template, but **one @id scheme across all sites** in a group makes audits possible.
Benefits: “trust and credibility”	Markup **describes** trust signals; it doesn't create them.

Part 8

Implementation workflow

The 2023 workflow was sound. This version adds the steps that prevent regressions.

Step	Owner	Output
1. Define the @id scheme and baseline graph	SEO + Dev	One documented pattern for Organization, WebSite, WebPage, Person
2. Map page types to templates	SEO	The table in Part 5, adapted to your site
3. Build template fields	Dev	Custom fields only for values that **vary by page**
4. Fill values	SEO / Content	Values that match **visible content**
5. Validate	QA	Schema Markup Validator and Rich Results Test on **one URL per template**
6. Monitor	SEO	Search Console enhancement reports; recheck after every **Google documentation change**

Where projects fail

Templates that output a **second Organization**, fields left with placeholder text, and markup that drifts from the page after a redesign.

Version the schema

Keep a **changelog** of template changes. When rich results change, you'll know whether it was you or Google.

Checklist

QA checklist for any page

1

Types

The page @type is the **most specific true** WebPage subtype. The main entity uses the most specific true type.

2

Properties

Every property **exists on schema.org** for that type or a parent. No invented keys.

3

Graph

One Organization and one WebSite **@id** across the site. Pages reference them; nothing duplicates them.

4

Visibility

Every marked-up value **appears on the page** as written.

5

Identity

sameAs links point to records of the **same entity**, verified, not topics or competitors.

6

Currency

No retired features (sitelinks search box). FAQ and HowTo expectations **updated**.

The one-line version

Use the most specific true type, define site-wide nodes once, reference them by @id, and only mark up what the page visibly says.

Sources

References

01 **Introduction to structured data markup. Google Search Central**
https://developers.google.com/search/docs/appearance/structured-data/intro-structured-data

02 **Full schema hierarchy. Schema.org**
The type tree and inherited properties.
https://schema.org/docs/full.html

03 **Site names in Google Search. Google Search Central**
WebSite name and alternateName.
https://developers.google.com/search/docs/appearance/site-names

04 **Organization structured data. Google Search Central**
https://developers.google.com/search/docs/appearance/structured-data/organization

05 **ProfilePage structured data. Google Search Central**
https://developers.google.com/search/docs/appearance/structured-data/profile-page

06 **Article structured data. Google Search Central**
https://developers.google.com/search/docs/appearance/structured-data/article

07 **Review snippet structured data. Google Search Central**
https://developers.google.com/search/docs/appearance/structured-data/review-snippet

08 **Changes to HowTo and FAQ rich results. Google Search Central Blog, August 2023**
https://developers.google.com/search/blog/2023/08/howto-faq-changes

09 **Farewell, sitelinks search box. Google Search Central Blog, October 2024**
https://developers.google.com/search/blog/2024/10/sitelinks-search-box

Guide by Ameneh Saeednia, prepared with AI assistance. Built from a 2023 in-house schema guide I contributed to; internal names, sites and people are removed. Example URLs are placeholders.