THE SIMPLE [A] TEAM PRESENTS

THE SCHEMA OF THINGS

How to lift search results using Schema.org and Sitecore CMS

Part 1:
This search result brought to you by Schema.org
The Marketer's Perspective

Part 2:
Implementing Schema.org with the Sitecore CMS
The Developer's Perspective

By
The [A] Team

The Schema Of Things

Presented in Two Parts

This paper is written to appeal to both business stakeholders, and technical team members.

Part 1:
What is Schema.org? What are the benefits?
Why should I consider marking up my content with it?
Schema.org SEO benefits and user experience examples.

Part 2:
How to implement Schema.org.
Requirements for applying Schema.org within the Sitecore CMS.
General insights about models and basic approaches.

Part 1: This search is brought to you by Schema.org

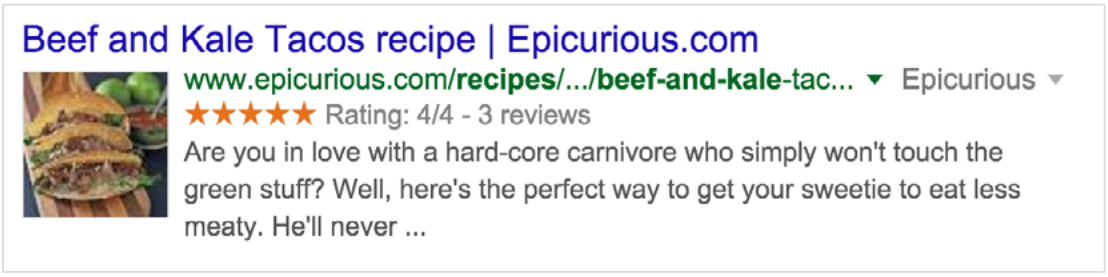
- Discover the why behind Schema.org.
- Learn to use Schema.org to drive search results and increase engagement.
- Read about one example that brought a 30% boost in traffic following changes to content presentation on SERP pages via Schema.org.
- Review Schema.org types and see how they enhance the appearance and relevance on SERPs.

Part 2: Implementing Schema.org with Sitecore content management system

- Peek into the technical side of using HTML5 microdata and the Sitecore CMS.
- Learn how Schema.org is used to free content editors to do their jobs, and seamlessly give search engines a solid understanding of your content.
- Get a step-by-step introduction to architecting content types, applying various properties and implementing presentation components in the Sitecore content management system.

PART 1

THIS SEARCH IS BROUGHT TO YOU BY SCHEMA.ORG



Do your organic results look like this in Google? Do you know why they should?

- To enhance listings and increase engagement within search engines.
- To help search engines better understand your content, potentially resulting in improved appearance in organic search results and increased clickability.
- To increase user interest and engagement on search engine results pages (SERPs).
- To free content editors to do their jobs, and search engines to index intelligently.
- To improve ease of content syndication among external content republishers.

SCHEMA.ORG: WHY YOU SHOULD PAY ATTENTION?

"Schema.org - Why You're Behind if You're Not Using It," "Why You Need to Know About Schema.org," and "Why You Can't Ignore Schema.org" are just a few of the Google results returned among the 333 million broad results on the search term "Schema.org."

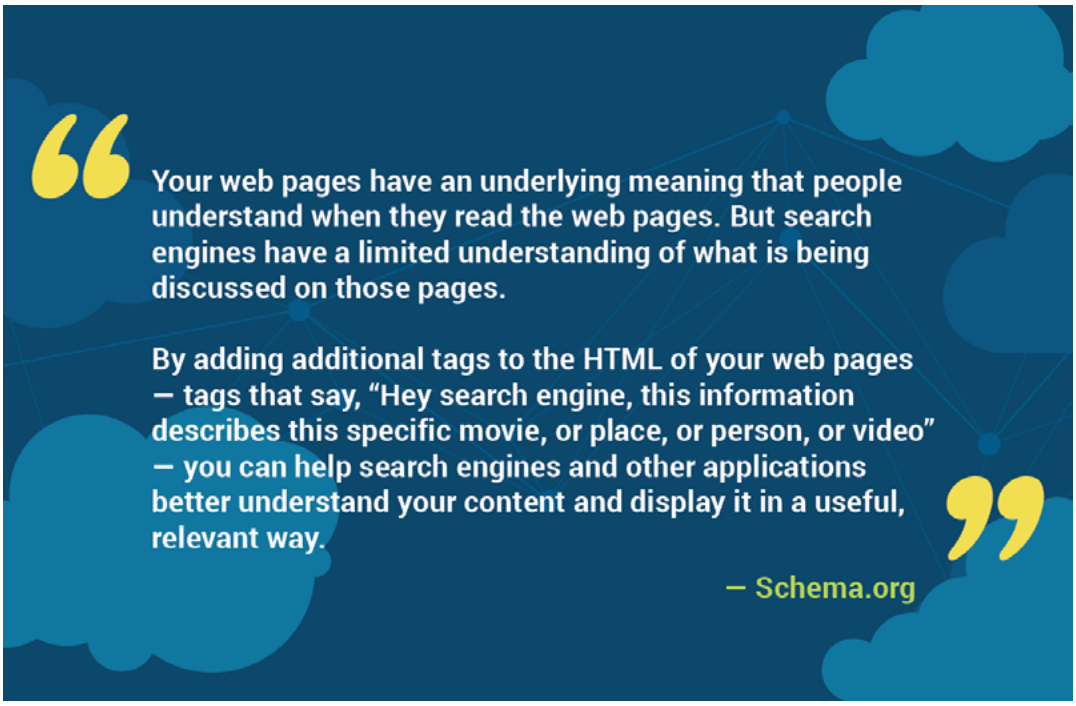
The 'semantic web' has been talked about for years, but not until Schema.org did enriching content with intelligent metadata markup really take off.

So, what's all the fuss? Why is it now so important to markup your content? In June 2011, Google, Bing and Yahoo collaborated to design a structured vocabulary geared to help search engines better understand the content on pages across the web. This resulted in Schema.org.

Since its launch, SEO gurus, bloggers, marketers and developers alike are applauding the value the approach brings to improving organic search results, both for the content consumer and publishers alike.

Marketers especially value Schema.org, as it guarantees search engines will understand organic content, and its use increases sites' organic visibility in search results.

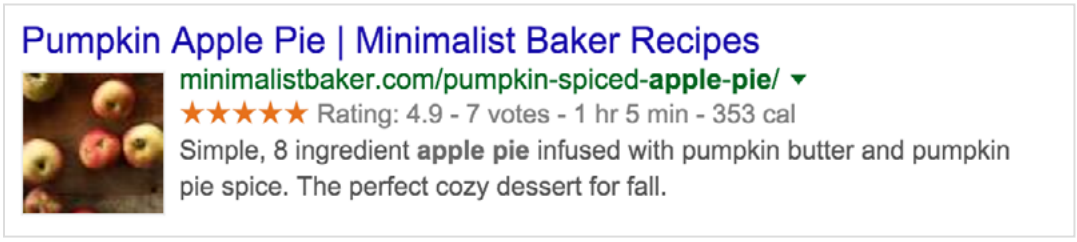
Here's how Schema.org explains it:



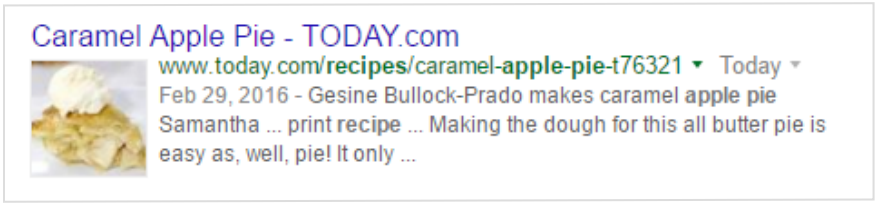
Let's take a look at a few examples of Schema.org in action:

This recipe highlights the benefits of Schema.org in action. In this organic search result, we can see the rating, number of reviews, cooking time, calories, comments by the creator and even the opportunity to find similar recipes.

Result 1: A recipe using Schema.org structured content:



Result 2: A recipe NOT using Schema.org structured content:



In these organic search results for apple pie recipes, we can see in the first result that using structured data returns a more attention-grabbing and informative result. While result one (enriched with Schema.org tags) is bubbling over with reviews, cooking time, caloric value, and comments by the creator, result two leaves us, excuse the pun, hungry for more. Which would you click on first?

This is only one example of the many schema types Schema.org offers to help search engines understand exactly what your content is about.

SCHEMA.ORG DRIVES ENGAGEMENT

One of the most beneficial reasons for adding Schema.org microdata to your site is the fact that on search engine results listings, your web pages can appear as enhanced listings or "rich snippets."

With rich snippets, users are more likely to visit if the search result has a little more "pop." According to Paul Breummer's article "How To Get A 30% Increase In CTR with Structured Markup," published on SearchEngineLand.com, case studies show retail firms can expect to get up to a 30 percent increase in organic traffic by using structured markup to add that "pop."

Supported by stats from seomoz.org's article "How to Get More Clicks With Low Rankings," Google click through rate (CTR) reports show that 36.4 percent of users are likely to click through an ad with a first position ranking, while a third position-ranking ad only sees a 9.5 percent click through rate. Drop to 6th position and the numbers drop to a grim CTR of 4.1 percent. Therefore the more "pop" you can achieve using structured data, the better.

COMMON SCHEMA.ORG TYPES

- Article
- Other creative works: CreativeWork, Book, Movie, MusicRecording, Recipe, TVSeries
- Embedded non-text objects: AudioObject, ImageObject, VideoObject
- Event
- Health and medical types: notes on the health and medical types under MedicalEntity
- Organization
- Person
- Place, LocalBusiness, Restaurant
- Product, Offer, AggregateOffer
- Review, AggregateRating

Notice how Google returns this search result for the movie Avatar from the IMDB website:

Avatar (2009) - IMDb
www.imdb.com/title/tt0499549/ ▾ Internet Movie Database ▾
★★★★☆ Rating: 7.9/10 - 860,078 votes
Videos. **Avatar** -- From Academy Award winning director James Cameron (Aliens, The Abyss, · **Avatar** -- Clip: Thanator chase ...
Full Cast & Crew - Trivia - Parents Guide - Movie connections

Users can learn about the movie's average rating, who directed it, cast details and even see they will get access to trivia and parental guide if they go ahead and click. This rich snippet is made possible by embedding Schema.org microdata to the page.

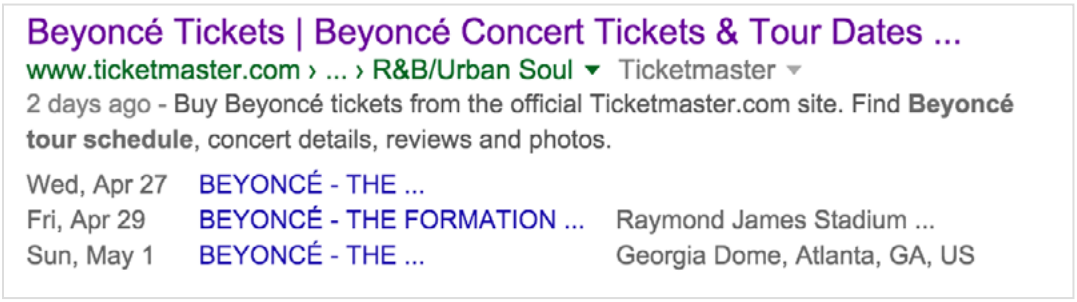
```
<div itemscope itemtype="http://schema.org/Movie">
    <h1 itemprop="name">
          Avatar
    </h1>
  <div itemprop="director" itemscope itemtype="http://
schema.org/Person">
    Director:
          <span itemprop="name">James Cameron</span>
    (born
          <time itemprop="birthDate" dateti-
me="1954-08-16">August 16, 1954</time>
    )
  </div>
  <span itemprop="ratingValue">7.9</span>
</div>
```

Just as the average rating of a movie can be displayed from the search result, it's just as easy to show contact information or music, product, and service reviews. A full list of types can be found at http://schema.org/docs/full.html

THIS SUPPORTING DATA IS ACTUALLY METADATA, OFFICIALLY REFERRED TO AS "MICRODATA," AND IS SIMPLY A SET OF TAGS INTRODUCED WITH HTML5.

- Itemscope – indicates the content in this container is an item.
- Itemtype – indicates the type of item being described.
- Itemprop – gives a value to each property of the item.

Here is another example of how Google returns search results for the terms "Beyoncé Concert Dates." Note how the data presented engages the reader by sharing content dates right away. In this way, a user need not click on the ad to already be engaged with the content advertised.

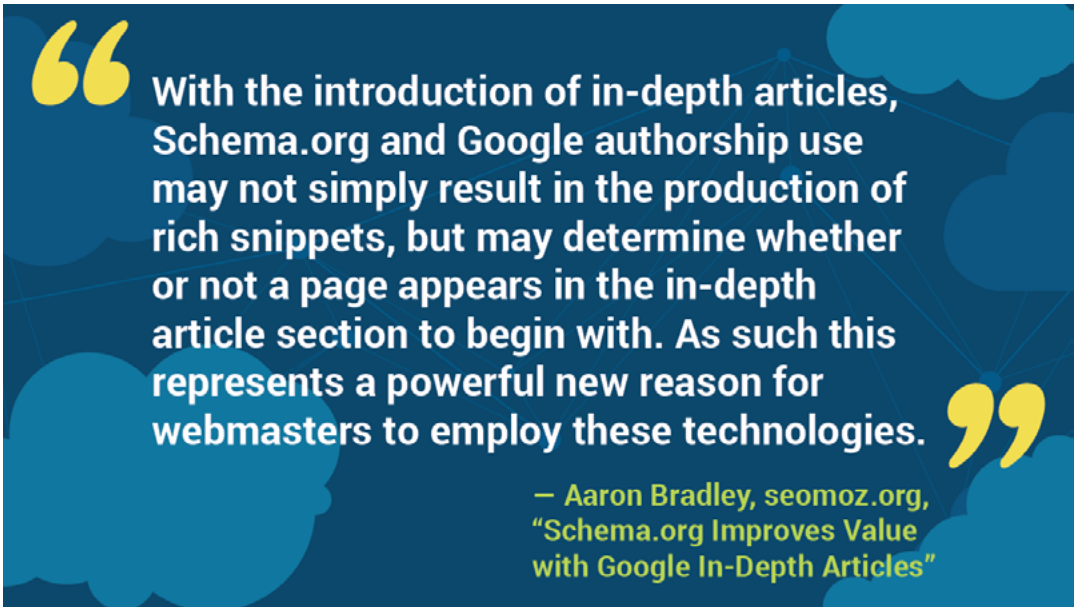


It should be noted, however, this markup will not necessarily improve your search rankings, but will likely increase the chances of having your link clicked first. Visibility matters in SERPs, and Schema.org-enriched content improves content visibility.

That said, Google makes specific recommendations for in-depth articles markup, which when applied can result in Google placing your content in the in-depth article section, which appears as a block in the center of Google's search result pages.

GOOGLE RECOMMENDS THE FOLLOWING FOR IN-DEPTH ARTICLE MARKUP:

1. Use Schema.org/article markup.
2. Use Google authorship markup.
3. Use pagination markup and avoid canonicalization errors.
4. Provide Google with information about your logo.
5. Use Schema.org/organization markup to facilitate logo identification.
6. Link to your Google+ organization profile.



In his article "Schema.org Improves Value with Google In-Depth Articles," Arron Bradley of seomoz.org notes that Google may use authorship to surface subject matter experts and authorities.

"Aside from the acknowledgment that Schema.org use may percolate content directly to the SERPs that otherwise may not have appeared there," he says, "the authorship recommendation is of particular interest because it more-or-less explicitly validates the long-standing assumption that Google would use authorship to surface, well, authorities in particular subject areas."

With a solid understanding of what Schema.org is and why you should use it, let's look into the technical side of what it takes to implement Schema.org within the Sitecore content management system. Simple [A] has models for how to accomplish the same results with Kentico or Ektron, but has only engineered full-cycle Schema.org solutions for Sitecore to date.

PART 2

IMPLEMENTING SCHEMA.ORG WITH THE SITECORE CMS – THE DEVELOPER'S PERSPECTIVE

Content creators, marketers and developers should work closely together when implementing Schema.org. It's best to plan ahead to utilize item types and properties on your web pages. This starts with working together to architect content types, define structured content types, build data templates and manage implementation.

DEFINING STRUCTURED CONTENT TYPES

Defining content types is an essential part of any CMS implementation project and should come early in the process. To integrate Schema.org tagging efficiently, think about requirements, what content types will be needed, and what Schema.org types are available and closest to the required content types.

Take, for example, a Location page. The Location page content type may contain the following fields.

- Name
- Street Address
- City
- State
- Zip
- Image
- Description

Digging into the Schema.org documentation, find the closest match to the desired content type. In this case, we'll use Place http://schema.org/Place.

A CLOSER LOOK AT THE PROPERTIES

Thing > Place

Entities that have somewhat fixed, physical extension.

Property	Expected Type	Description
Properties from Thing		
additionalType	URL	An additional type for the item, typically used for adding more specific types from external vocabularies in microdata syntax. This is a relationship between something and a class that the thing is in In RDFa syntax, it is better to use the native RDFa syntax —the 'typeof' attribute— for multiple types. Schema.org tools may have only weaker understanding of extra types, in particular those defined externally.
alternateName	Text	An alias for the item.
description	Text	A short description of the item.
image	URL	URL of an image of the item.
name	Text	The name of the item.
sameAs	URL	URL of a reference Web page that unambiguously indicates the item's identity. E.g. the URL of the item's Wikipedia page, Freebase page, or official website.
url	URL	URL of the item.
Properties from Place		
address	PostalAddress	Physical address of the item.
aggregateRating	AggregateRating	The overall rating, based on a collection of reviews or ratings, of the item.
containedIn	Place	The basic containment relation between places.
event	Event	Upcoming or past event associated with this place or organization.
events	Event	Upcoming or past events associated with this place or organization (legacy spelling.; see singular form, event).
faxNumber	Text	The fax number.
geo	GeoCoordinates or GeoShape	The geo coordinates of the place.

It's important to note not all content needs to be exposed to the user. If some information is more useful to the search engines than the site's visitors, hide the data using meta tags. Additional useful information in this example may be the geo-coordinates, so let's add two additional fields to our content type: latitude and longitude.

Occasionally, the use of nested types is needed. For instance, we are using the type "Place," but within the markup we will be using the "PostalAddress" type (http://schema.org/PostalAddress) to encapsulate the address fields.

Once all of the fields for content types are defined, be sure to document the mappings of each field to its related Schema.org.

Field	Property
Name	Name
Street Address	PostalAddress:streetAddress
City	PostalAddress:addressLocality
State	PostalAddress:addressRegion
Zip	PostalAddress:postalCode
Image	image
Description	description
Latitude	GeoCoordinates:latitude
Longitude	GeoCoordinates:longitude

Repeat this exercise for all defined content types.
A product may look like this:

Field	Property
Name	Name
Price	Offer:price
Description	description
Rating	AggregateRating:ratingValue
In Stock	Offer:availability

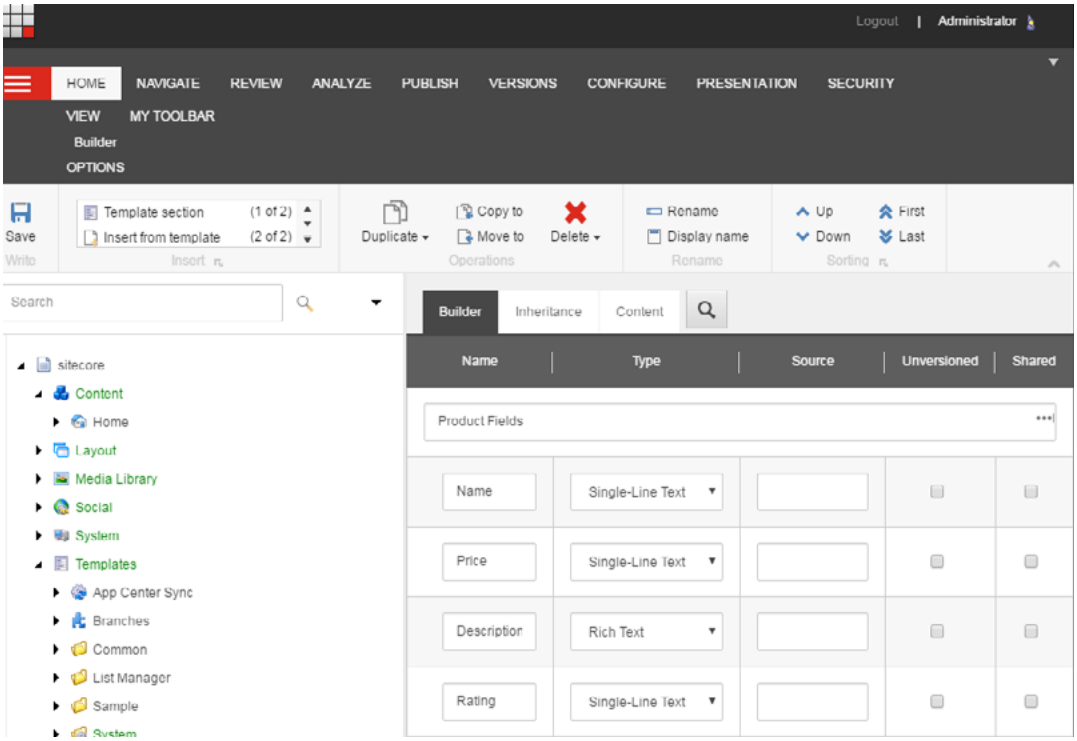
Once the content definitions are created and mapped to the appropriate Schema.org types and properties, this documentation can be handed to a developer who will build out the data templates within Sitecore.

IMPLEMENTATION AND DATA TEMPLATES

Next comes the implementation. The main point here is content editors should be saved the burden of Schema.org tagging. Editors should enter content as they normally would and create presentation components that handle inputting the appropriate items. As with any other Sitecore component development, this starts with creating data templates.

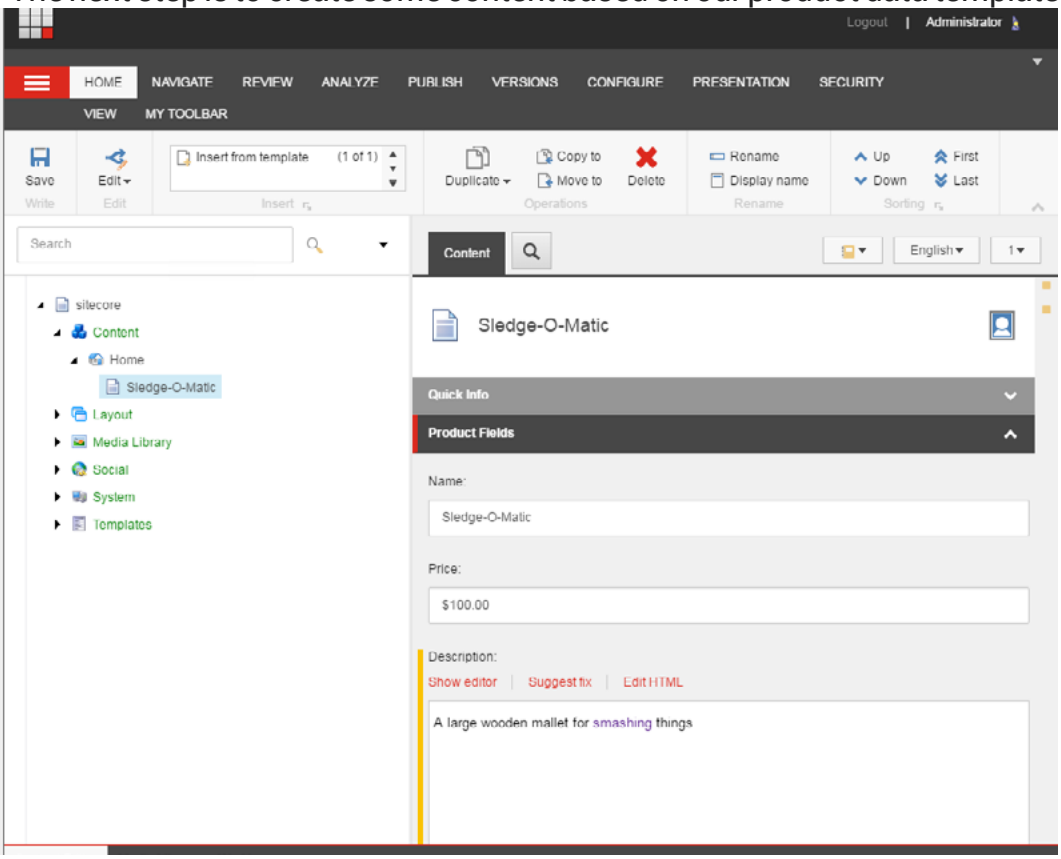
Take the last example of a product, remember the goal is to align fields to Schema.org properties wherever possible - http://schema.org/Product.

The data template for a product based on our definition will look like this:



Typically, additional steps exist when creating data templates in Sitecore such as setting an icon, defining standard values, etc. but these are outside the scope of this paper.

The next step is to create some content based on our product data template:



Now that content is in Sitecore (and assuming there is a layout to put components in), it's time to create the presentation. The presentation component, which renders the product, will come in the form of a Sublayout. Sublayouts are simply .NET user controls which get put into placeholders within the layout or other renderings.

Sitecore comes with several server controls that output data template field values. For this demonstration, we'll use the FieldRenderer control. FieldRenderer has a property called FieldName whose value corresponds to the field name on our data template. Whatever value is in that named field gets rendered out to the presentation, thus the name FieldRenderer. This control also gives the added benefit of allowing content administrators to use Page Editor to edit field values in context.

Start with wrapping the component markup with the itemscope — associated with a specific Schema.org URL, in this case http://schema.org/Product.

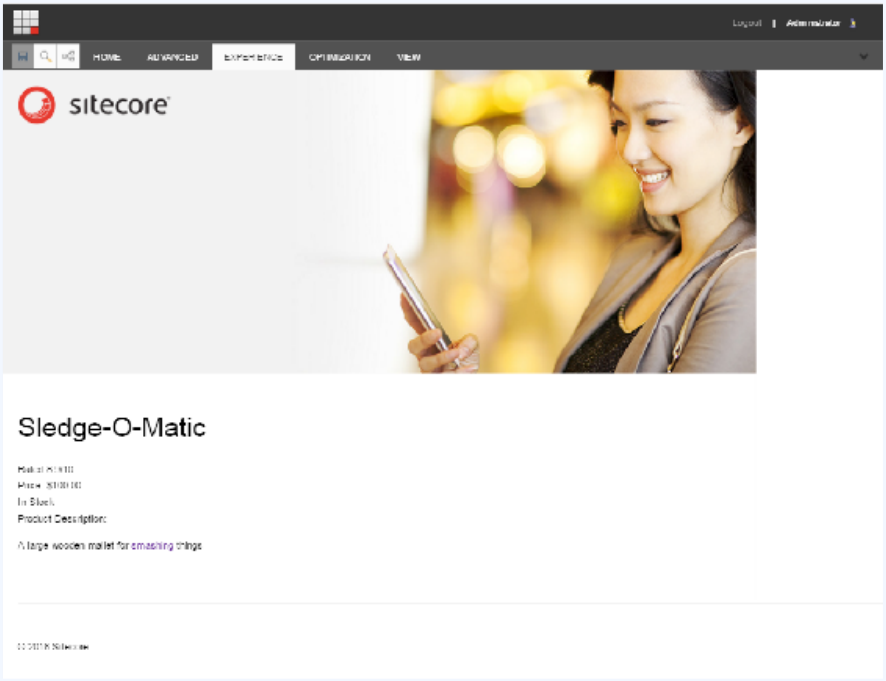
Next, wrap field renderers with applicable itemprop values. As mentioned previously, some fields will relate to another item type and be nested in the markup. For instance in this example, we can use "Aggregate Rating" and "Offer".

The resulting code looks like this:

```
<div itemscope itemtype="http://schema.org/Product">
  <span itemprop="name"><h3><sc:FieldRenderer runat="ser-
ver" FieldName="Name"/></h3></span>
  <div itemprop="aggregateRating" itemscope itemtype="ht-
tp://schema.org/AggregateRating">
   Rated <span itemprop="ratingValue"><sc:FieldRenderer
runat="server" FieldName="Rating"/></span>/10
  </div>
  <div itemprop="offers" itemscope itemtype="http://schema.
org/Offer">
    <p>Price:<span itemprop="price"><sc:FieldRenderer ru-
nat="server" FieldName="Price"/></span></p>
      <link itemprop="availability" href="http://schema.
org/InStock" /><asp:Literal runat="server" ID="inStockLit"
/>
  </div>
  <p>Product description:</p>
  <span itemprop="description"><sc:FieldRenderer  runat="-
server" FieldName="Description"/></span>
 </div>
```

A little extra processing to show the status of “in stock” inStockLit.Text = Sitecore.Context.Item.Fields[“In Stock”].Value == “1” ? “In Stock” : “Out of Stock”;

Here’s what it looks like, just an ordinary product detail page:



Viewing the underlying markup, see the Schema.org metadata:

```
<div itemscope itemtype=”http://schema.org/Product”>
  <span itemprop=”name”><h3>Sledge-O-Matic</h3></span>
  <div itemprop=”aggregateRating”
    itemscope itemtype=”http://schema.org/AggregateRa-
ting”>
   Rated <span itemprop=”ratingValue”>8.9</span>/10
  </div>

  <div itemprop=”offers” itemscope itemtype=”http://schema.
org/Offer”>
    <p>Price:<span itemprop=”price”>$100.00</span></p>
      <link itemprop=”availability” href=”http://schema.
org/InStock” />In Stock
  </div>

  <p>Product description:</p>
  <span itemprop=”description”><p>A large wooden mallet
used for smashing things!</p></span>

 </div>
</div>
</div>
```

Sitecore's completely customizable nature means Rich Text Field Renderers can be extended to automatically output itemprop tags. For example, to give every anchor tag in a rich text field value an item property of "url" – just hook into the RenderField pipeline like so:

```
public class AddMicroData
    {
        public void Process(Sitecore.Pipelines.RenderField.Ren-
derFieldArgs args)
        {
            if (args.FieldTypeKey != "rich text")
            {
                return;
            }
            args.Result.FirstPart = EnsureItemProp(args.Result.
FirstPart);
            args.Result.LastPart  = EnsureItemProp(args.Result.
LastPart);
        }
        protected string EnsureItemProp(string markup)
        {
            // optimization in case the markup does not contain
any links
            if (!markup.Contains("<"))
            {
                return markup;
            }
            // parse the markup into an object
            var doc = new HtmlAgilityPack.HtmlDocument();
            doc.LoadHtml(markup);
            // retrieve all <a> elements with an empty itemprop
attribute
            // or no itemprop attribute
            HtmlAgilityPack.HtmlNodeCollection nodes = doc.Docu-
mentNode.SelectNodes("//a[@itemprop = '' or not(@itemprop)]");
            // if there are no such nodes, return the original
markup
            if (nodes == null || nodes.Count < 1)
            {
                return markup;
            }
            // for each such anchor
            foreach(HtmlAgilityPack.HtmlNode node in nodes)
            {
                node.SetAttributeValue("itemprop", "url");
            }
            // convert from object form back to markup, and re-
turn that markup.
            var sb = new StringBuilder();
            var sw = new StringWriter(sb);
            doc.Save(sw);
            sw.Close();
            return sb.ToString();
        }
    }
```

This will wrap all of the links with an itemprop field of “url.”

Warning: Don’t forget to add a reference to the processor in a configuration file!

Note: The above code is a modified version of a pipeline processor example from John West’s book, Professional Sitecore Development.

Tip: Google provides a Structured Data Testing Tool to test the rich snippets. Just paste resulting HTML into the testing tool to get validation results: http://www.google.com/webmasters/tools/richsnippets

THE PROMISE OF STRUCTURED DATA

In the article, “How Retailers Can Improve Product Visibility Using Structured Markup,” Breummer describes the promise of structured markup, “search robots will better understand Web content that’s made machine-readable by the markup vocabulary; therefore, when search engines answer queries, they do so with a better understanding of user intent. Shoppers are happy campers because with less effort, they get relevant results they can trust.”

So that’s all there is to it! You now know what Schema.org is, the value it brings to search results, how to architect content types to align with item types and properties, and how to implement presentation components in Sitecore.

With these steps, you will help search engines better understand your content; and your content will stand out in search results, thus living a more exciting life beyond your website.

Below are some informative articles on this complex, but important topic.

- Future SEO – Understanding Entity Search - Read More
- 10 Reasons Why Search is In Vogue: Hot Trends in Semantic Search - Read More
- 2 Underutilized Opportunities for Schema on Your Website - Read More
- In-depth articles in search results - Read More
- Getting started with structured data - Read More
- Introducing the Structured Data Dashboard - Read More
- Measuring The Effect Of Semantic Markup On Your Search Traffic - Read More
- Google Knowledge Graph Carousel Sightings Becoming More Frequent Within A Wider Variety Of Searches - Read More
- From Microdata & Schema To Rich Snippets: Markup For The Advanced SEO - Read More
- Embedded Video Schema Markup: A 7-Step Checklist for Great Results - Read More
- Tim Berners-Lee: The next web (video) - Read More
- LinkedData.org: Publishing and connecting structured data - Read More

sitecore
Certified Solution Partner

About [A]

[A] is a distributed technology consulting and training company dedicated to humanizing the digital experience. We perform .NET, as well as Drupal CMS implementation and integration, and provide CMS and CEM platform selection consulting, including fully-managed selection cycles or simple advisory services.

[A] brings tens of thousands of hours successfully implementing and crafting intelligent, engaging customer experiences for large national associations, healthcare organizations, university systems and other multi-faceted organizations. We can manage the entire process for you, or augment your process and coach your team through the considerations, building your decision matrix, and engaging with various platform vendors.

We simplify complex technology implementations to deliver powerful web content management and content marketing platforms. Engage users with relevant content experiences and keep them loyal to your brand.

Partner with [A] to create intelligent content tailored to your audience needs, aligned to your organizational goals and delivered with personalized, targeted marketing tools and technologies. [A] delivers consulting, training, application development, architecture and strategy engagement. Bilingual services are also available.

Contact [A] for help applying Schema.org or assistance with Sitecore we are a Sitecore certified solution partner.

info@simplea.com

815A Brazos, Suite 115
Austin, TX 78701
512-646-2100
simplea.com

Find us online

simpleateam | +SimpleaTeam | company/simple-a