

# Article Context

---

17 DECEMBER 2018

## Article Context

Article context currently supports three metrics - author name, published date, and category. These metrics are used in Big Data Analytics for the Author Metrics, Published Date, and Category reports under Content. Listed below are the ways we gather the data from the webpage in order to help publishers/pubteam understand how to utilize this information on their sites. Each section will have the different html tags we use and order is important! For example, if the author name is in script tag for application/ld+json format, then we will gather that data and not search for any of the other html tags on the page. JSON-LD is the first script tag pulled from the html - and generally has author and published date information.

## JSON-LD

### JSON-LD

JSON-LD is an industry standard and more information can be found here: <https://json-ld.org/>. A very good resource is <https://jsonld.com/> which has examples for the different json-ld formats for Articles, Blog Posts, etc.

This is an example of Article ld+json script tag you can use on your site.

```
<script type="application/ld+json">
{ "@context": "http://schema.org",
  "@type": "Article",
  "headline": "Extra! Extra! Read all about it",
  "alternativeHeadline": "This article is also about robots and stuff",
  "image": "http://example.com/image.jpg",
  "author": "Patrick Coombe",
  "award": "Best article ever written",
  "editor": "Craig Mount",
  "genre": "search engine optimization",
  "keywords": "seo sales b2b",
  "wordcount": "1120",
  "publisher": "Book Publisher Inc",
  "url": "http://www.example.com",
  "datePublished": "2015-09-20",
  "dateCreated": "2015-09-20",
  "dateModified": "2015-09-20",
  "description": "We love to do stuff to help people and stuff",
  "articleBody": "You can paste your entire post in here, and yes it can get really
really long."
}
</script>
```

This is an example of Blog Post ld+json script tag you can use on your site.

```
<script type="application/ld+json">
{ "@context": "http://schema.org",
  "@type": "BlogPosting",
  "headline": "14 Ways Json Can Improve Your SEO",
  "alternativeHeadline": "and the women who love them",
  "image": "http://example.com/image.jpg",
  "award": "Best article ever written",
  "editor": "John Doe",
  "genre": "search engine optimization",
  "keywords": "seo sales b2b",
  "wordcount": "1120",
  "publisher": "Book Publisher Inc",
  "url": "http://www.example.com",
  "datePublished": "2015-09-20",
  "dateCreated": "2015-09-20",
  "dateModified": "2015-09-20",
  "description": "We love to do stuff to help people and stuff",
  "articleBody": "You can paste your entire post in here, and yes it can get really
really long.",
  "author": {
    "@type": "Person",
    "name": "Steve"
  }
}
</script>
```

## Author & Published Date & Category

From the JSON-LD, we will extract the `"author"` and `"datePublished"`. Please note the different `"author"` structures between the Blog Post and Article examples above - we can either gather the information if the author is listed as a string (`"author": "Patrick Coombe"`) or we find the `"name"` element of the author object:

```
"author": {
  "@type": "Person",
  "name": "Steve"
}
```

If there is also an `"articleSection"` in the json it will be used for Article Category information.

## AUTHOR

If there is not json-ld we search other meta tags and html to get the author information. Listed in order are:

- The next standard used is the Open Graph protocol <http://ogp.me/> to add metadata.

This is an example of using Open Graph for profile metadata for the author:

```
<meta property="profile:first_name" content="John">
<meta property="profile:last_name" content="Doe">
```

- Author meta tag

```
<meta name="author" content="John Doe">
```

- Author Sailthru metatags

<https://getstarted.sailthru.com/site/personalization-engine/meta-tags/>

```
<meta name="sailthru.author" content="John Doe">
```

- Author hyperlink using rel="author" to indicate that a link is pointing to information about the author of a page or article. Whatever link text will be used as the author name. In the example below, **John Doe** is recorded as the author.

```
<a href="/author-page.html" rel="author">John Doe</a>
```

- Author from article Open Graph  
<https://developers.facebook.com/docs/reference/opengraph/object-type/article/>

```
<meta property="article:author" content="John Doe">
```

- Profile username from Open Graph  
<https://developers.facebook.com/docs/reference/opengraph/object-type/profile/>

```
<meta property="profile:username" content="johndoe">
```

Note: username is usually a single name (first and last all lowercase like above example). Unfortunately, this can also come back as a full url like <http://www.facebook.com/john.doe> which will be stored as the author. This metatag is the last data we check for a site in hopes to get some information rather than none.

## PUBLISHED DATE

If there is not json-ld we search other meta tags and html to get the published date information. Listed in order are:

- The next standard used is the Open Graph protocol <http://ogp.me/> to add metadata for article publish\_time  
<https://developers.facebook.com/docs/reference/opengraph/object-type/article/>

This is an example of using Open Graph for article publish\_time:

```
<meta property="article:published_time" content="1972-06-18T01:23:45Z">
```

- Published date using meta tag date

```
<meta name="date" content="1972-06-18T01:23:45Z">
```

OR

```
<meta name="date" content="1972-06-18">
```

- Published Date Sailthru metatags

<https://getstarted.sailthru.com/site/personalization-engine/meta-tags/>

```
<meta name="sailthru.date" content="2012-06-01 10:05:00">
```

## CATEGORY

If there is not json-ld we search other meta tags and html to get the category information. Listed in order are:

- The next standard used is the Open Graph protocol <http://ogp.me/> to add metadata for article section  
<https://developers.facebook.com/docs/reference/opengraph/object-type/article/>

This is an example of using Open Graph for article section:

```
<meta property="article:section" content="Politics">
```

- Category using meta tag category

```
<meta name="category" content="Politics">
```