NewsBreak Feed Specification

Purpose of This Document

This specification documents the feed formats that NewsBreak currently accepts and considers as the best practices.

Industry Specifications

NewsBreak accepts RSS 2.0 (https://www.rssboard.org/rss-profile) and Atom 1.0 (https://tools.ietf.org/html/rfc5023).

XML Namespaces

Please declare the following namespaces in the root of the feed. You can check the sample RSS for reference.

```
xmlns:nb="https://www.newsbreak.com/"
xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:content="http://purl.org/rss/1.0/modules/content/"
```

Required Fields for Every News Item

Required Field	Description
<title>	The title of the article
<link>	The URL of the article. Please use the canonical URL without appending any NewsBreak specific parameters.
<pubDate>	The publication date and time
<dc:creator>	The authors of the article. There might be more than one <dc:creator> tag per article.
<description>	The summary of the article
<content:encoded>	The full content of the article in the html format. This is what NewsBreak uses to display the content of the article. It should include all videos, audios, images, image captions and hyperlinks. *(Tips: Save the text of this field into a .html file and open it in a web browser. The displayed content will be similar to what users will see on NewsBreak. The only differences are the CSS styles.)*

GUID

GUID is used as the URL of the article if requested. **The default behavior is ignoring this field unless NewsBreak agreed to use it**. The GUID should be in the format of URL (aka. isPermalink=true). If there exist multiple feeds, they should share the same GUID generation logic.

Restricted Field	Description
<guid>	The permalink to the item.

Thumbnail Image

The first image (if available) in <content:encoded> is used as an article's thumbnail image when this field is missing.

Optional Field	Description
<media:thumbnail>	The URL of the thumbnail image

Images and Image Captions

Please include the images and their captions in the html content of <content:encoded> using HTML <figure> Tag along with <img> Tag and <figcaption> Tag. Special styles are applied to figcaption to distinguish it from normal text.

Images and image captions sample

```
<content:encoded><![CDATA[
        <figure>
                <img src="pic_trulli.jpg" alt="Trulli">
                 <figcaption>Fig.1 - Trulli, Puglia, Italy.</figcaption>
        </figure>
]]></content:encoded>
```

Video and Audio

Please include the video and audio in the html content of <content:encoded>. NewsBreak supports video and audio in HTML Video, HTML Audio and HTML Iframes format. When iframes are used, please add class="nb-video" or class="nb-audio" to the tags to denote the purposes. Refer to the *Tips* of <content:encoded> (on page 1 of this document) for testing.

Embedded Social Media

NewsBreak currently supports embedded posts and videos from Twitter, Facebook, Instagram and TikTok. Please include the embedding codes in <content:encoded>.

Disclosure Info

For sponsored, branded, promotional, and affiliated content, the relationship needs to be clearly disclosed. Read the "Commercial and Promotional Content" section in Community Standards for details.

Please include the plain text of the disclosure in this field. It will be displayed at the beginning of the article.

Optional Field	Description
<nb:disclosure>	Plain text discloses the relationship of commercial and promotional content. The description has to state who will benefit from the affiliate links present in the body of the content. For example: {media name} is supported by its audience. When you purchase through links on this article, {media name}, may earn an affiliate commission.

Analytical Scripts

NewsBreak supports third party scripts and tracking images for analytical purposes. The scripts will run in an iframe for security considerations. An optional <nb:scripts> field for each news item is used for this purpose.

Optional Field	Description

<nb:scripts>	All of the script, noscript and tracking images are used for analytical purposes. All of the text in this field will be wrapped in an iframe and render along with the article content. (Please add 'nb' namespace in your xml frame.)

ComScore Scripts Example

To run ComScore script in NewsBreak, publishers need to pass c4 and c9 explicitly. The tag source is different, as well. If a publisher has a ComScore c2/client ID of 12345678, the code should make a request to JavaScript file https://sb.scorecardresearch.com/cs/12345678/x-beacon.js

Please refer to the following example:

```
<nb:scripts><![CDATA[<script type="text/javascript">

        var _comscore = _comscore || [];
        _comscore.push({ c1: "2", c2: "12345678", c4: "https:\/\/example.com\/sample\/sample-news\/", c9:
"newsbreak.com", comscorekw: "newsbreak" });
        (function() {
                var s = document.createElement("script"), el = document.getElementsByTagName("script")[0];
                s.async = true;
                s.src = "https://sb.scorecardresearch.com/cs/12345678/x-beacon.js";
                el.parentNode.insertBefore(s, el);
        })();
</script>]]></nb:scripts>
```

Google Analytics Scripts Example

```
<nb:scripts><![CDATA[<script>
        (function(i,s,o,g,r,a,m){i['GoogleAnalyticsObject']=r;i[r]=i[r]||function(){
                (i[r].q=i[r].q||[]).push(arguments)},i[r].l=1*new Date();a=s.createElement(o),
                m=s.getElementsByTagName(o)[0];a.async=1;a.src=g;m.parentNode.insertBefore(a,m)
        })(
                window,document,'script','https://www.google-analytics.com/analytics.js','ga');
                ga('create', decodeURIComponent( 'UA-000000-00' ), 'auto');
                ga('require', 'displayfeatures');
                ga('set', 'checkProtocolTask', null);
                ga('set', 'checkStorageTask', null);
                ga('set', 'referrer', 'https://www.newsbreak.com/');
                ga('set', 'campaignName', 'News Break');
                ga('set', 'campaignSource', 'News Break');
                ga('set', 'campaignMedium', 'Article');
                ga('set', 'page', "https:\/\/sample.com\/sample-news");
                ga('set', 'title', 'sample title' );
                ga('send', 'pageview' );
</script>]]></nb:scripts>
```

Sample RSS

```
<rss version="2.0" xmlns:nb="https://www.newsbreak.com/" xmlns:dc="http://purl.org/dc/elements/1.1/"
xmlns:content="http://purl.org/rss/1.0/modules/content/">
   <channel>
     <item>
        <title>Sample Title</title>
        <link>https://www.newsbreak.com/mission</link>
        <pubDate>2021-02-05T18:36:33.652Z</pubDate>
        <dc:creator>Test Author1</dc:creator>
        <dc:creator>Test Author2</dc:creator>
        <description>Summary of the news.</description>
        <content:encoded>
          <![CDATA[

          <figure>
              <img src="https://static.newsbreak.com/_next/static/assets/icon_full_logo.svg"/>
              <figcaption>News Break is the Nation's #1 Intelligent Local News Platform</figcaption>
           </figure>
          <iframe src="https://www.youtube.com/embed/ZhfHXcfyIyg" class="nb-video" ></iframe>
          <p><a href="https://www.newsbreak.com/">News break</a></p>
          <p>Some text</p>
        ]]>
```

```xml
        </content:encoded>
        <nb:scripts>
          <![CDATA[
            <script type="text/javascript" src="https://www.googletagmanager.com/gtag/js?id=UA-XXXXXX-
1"></script>
            <script type="text/javascript">
              window.dataLayer = window.dataLayer || [];
              function gtag() { dataLayer.push(arguments); } gtag('js', new Date()); gtag('config', 'UA-XXXXXX-1');
            </script>
          ]]>
        </nb:scripts>
      </item>
    </channel>
</rss>
```

VIDEO – MRSS FEEDs

Required:
1. Video File (i.e. mp4)
2. Title
3. Time/created at (Publish Time)
4. Author/brand (if we want to split into multiple Newsbreak profiles/accounts)

Optional (good to have):
1. location (geo-political entity, for example: Palo Alto, California; Santa Clara County, California; Bay Area; California)
2. hashtags
3. description
4. category info
5. thumbnail image