

Technical Disclosure Commons

Defensive Publications Series

August 17, 2017

EMBEDDED CONTENT ANNOTATIONS FROM EMBEDDING SOURCE

Justin Lewis

Scott Davies

Follow this and additional works at: http://www.tdcommons.org/dpubs_series

Recommended Citation

Lewis, Justin and Davies, Scott, "EMBEDDED CONTENT ANNOTATIONS FROM EMBEDDING SOURCE", Technical Disclosure Commons, (August 17, 2017)
http://www.tdcommons.org/dpubs_series/635



This work is licensed under a [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/).

This Article is brought to you for free and open access by Technical Disclosure Commons. It has been accepted for inclusion in Defensive Publications Series by an authorized administrator of Technical Disclosure Commons.

EMBEDDED CONTENT ANNOTATIONS FROM EMBEDDING SOURCE

Publishers often embed third-party content in their webpages and applications. Example third-party content may include links to videos of advertisements or how-to content for the publishers. News publishers also embed videos related to news articles. In some instances, social content (e.g., a social media post) may be embedded in the webpage or applications of the publishers. However, the publishers may have little control over the embedded content. For example, the embedded content may be provided by a content provider (e.g., embedding source), such as a content sharing platform where content is uploaded, and the embedded content may include irrelevant content that may not be separable from the relevant content. Also, the publisher may desire to focus on a small amount of the embedded content, or perhaps deal with the embedded content sequentially in a discussion. For example, a publisher of a news story may desire to discuss different parts of an embedded video, which may be more effective if the reader reads certain paragraphs after watching certain segments of the embedded video. Additionally, a publisher of an article may desire to make reference to certain parts of a social media post in the article. Conventionally, the publishers may refer to the part of the embedded content in writing. That is, an article may specify that certain events occur in an embedded video at a certain timestamp. When discussing a social media post, the article may refer to certain words and possibly reproduce the entire social media post.

We present a mechanism that solves these issues by allowing publishers to embed their own annotations in embedded content that they do not own. For example, an annotation may be added, modified, or highlighted within embedded content using the mechanism. In an implementation, a webpage developer or mobile application developer may embed a video in a webpage or application and apply additional parameters to the embedded video to add or modify

annotations in the embedded video. It should be noted that, in addition to annotations in videos, the mechanism may enable modifying or adding annotations in other types of media such as music streams or social media posts.

Using the mechanism, a user can watch an embedded video, and annotations may appear at appropriate times that provide the user a link to a discussion about the events shown at that point in the embedded video. Certain words in social media posts may be highlighted or contain links to discussions. Embedded content containing a large number of comments can include highlights or links on some of the comments that the publisher discusses elsewhere in the webpage or application. For example, the mechanism may be used to highlight one of the comments so that when the media content (e.g., streaming audio track) is embedded, that comment is given priority over other comments associated with the media content in a user interface.

Further, the mechanism may be used to give different meaning to content than the content may have on its own. For example, a publisher's story may be about a very specific element in a piece of content, such as a singer in a band. The mechanism may be used to highlight the singer in an embedded music video and the highlighting annotation may be easily clickable for the reader/viewer. Upon clicking the highlighted singer, the webpage may redirect the user to another part of the webpage that includes certain information about the singer. Thus, the highlighting annotation may tie the music video together with the story more closely than just showing the music video without the annotation. Accordingly, the mechanism may enable publishers to use and discuss embedded content flexibly in ways that were not anticipated by the embedded content owner.

The mechanism discussed herein may also be useful for embedded content that advertises

the publisher's products. For example, the publisher may use an advertisement video hosted on a content sharing platform, but with this mechanism, the publisher may add annotations that take the user to different portions of the webpage including the embedded video or to different webpages within the publisher's domain. Additionally, the annotations may cause the embedded media player to jump to another portion of the video being displayed or to display another video within the embedded media player. Also, if the user selects a particular product displayed in the embedded video, for example, an annotation may redirect the user to a webpage with more information about the product. As discussed further below, the mechanism may include an API by which these annotations may be added to requested embedded content. An alternative implementation allows a referrer to include annotations in hosted content. For example, if an advertiser refers a user to a content sharing platform to view videos with more information about the advertiser's products, the advertiser may include special annotations in the video even though it is hosted on the content sharing platform. As such, the mechanism may be used to add or modify annotations in embedded content as well as referred content.

Embedded content may be embedded in a webpage or a mobile native application. In both cases, the developer requesting the embedded content with customized annotations may access the mechanism (e.g., application programming interface (API)) provided by the embedding source to embed the content. The API may include various annotation parameters for specifying the annotations that the embedder wants to add or modify. In some implementations, when embedding is not used, another mobile application may be invoked. In such instances, the content creators may add parameters to the invocation of the mobile application to specify the desired annotations. These parameters may be passed as extra intent parameters or as additional uniform resource locator (URL) parameters.

The various annotation parameters may include the text to be displayed by the annotation, the time at which to start displaying the annotation within the content, the time at which to stop displaying the annotation within the content, a display location (e.g., x/y coordinates) within the content to include the annotation, display properties (e.g., color, transparency, etc.), annotation type (e.g., intractable box, overlay recommendation, information card (referred to as “info card”), etc.), size (e.g., length and width) of the annotation, and/or a hyperlink parameter that causes a redirection to be performed when the hyperlink is selected. The link may enable a developer to have control over the navigation endpoints followed by selecting the annotation within the embedded content. Thus, the mechanism may enable the developer to not only specify the time, location, and visual rendering of the annotation within the embedded content, but also the navigation details when a user engages with the annotation.

In some implementations, the hyperlink parameter may cause the webpage including the embedded content to jump to a particular portion of the webpage when selected. The portion of the webpage jumped to may include information relevant to the portion of the video from which the webpage navigates away as a result of the annotation selection. In some implementations, the link may cause the embedded media player to jump to another portion of the content being played or may cause the media player to jump to a particular portion of another content within the embedded media player.

In some implementations, the location information may include a set of locations associated with different times to enable the annotation to move around within the content as the content is played. For example, the annotation may be moved around on a video so that it is consistently located in empty space (e.g., the sky) in the video and not covering a person that moves around within the video.

The mechanism may be implemented in various ways from an API perspective. For example, the raw variables (e.g., text, time, location, color, transparency, type, size, hyperlink, etc.) may be included as individual parameters in a request for annotated embedded content to the API from the client-side website or application of the developer. Further, the variables may be included in an encrypted, obfuscated or simply concatenated data object, such as a JavaScript Object Notation (JSON) object. The client-side website or application may pass the encrypted, obfuscated JSON object as a string to the API and the server hosting the API may decrypt the JSON object and modify or add annotations to the content as specified. In yet another implementation, the variables may be stored server-side and a unique token (e.g., unique identifier) is provided to the client-side website or application. The unique token may be provided as a parameter in a request to the API by the website or application. The API may use the unique token provided to fetch the annotation variables associated with the unique token and modify them as specified by the parameters in the request.

The mechanism enables rendering the annotations in various forms within the content in view of the type of annotation, color, transparency, size, location, and/or time of display provided in the request. For example, the type of annotation may include an intractable box annotation, overlay content recommendation annotation, info card annotation, or the like. Intractable box annotations may appear within the content (e.g., video) and include text or a click target. The text may be colored based on the color parameter and the transparency may be set based on the transparency parameter. The intractable box may appear on the content at the specified time and be located in the specified location of the content.

The overlay content recommendation annotations may be generated by the embedding source (e.g., content host) and may recommend either other content (e.g., video) or a channel

with which to subscribe. The overlay content recommendation annotation may be colored according to the color parameter and transparency set according to the transparency parameter. In some instances, the size and location of the overlay content recommendation annotations may be statically set. In other instances, the size and location may be configurable. The overlay content recommendation annotation may be visible within the content based on the time parameter.

The info cards may be included in a list of info cards in a bucket within the content. By default, the list of info cards may be hidden and just an icon representing the bucket may be displayed within the content. The icon may be visible in a corner of the content (e.g., top left, top right, bottom left, bottom right). The list of info cards within the bucket can be triggered to appear (e.g., upon selection). The bucket may highlight at a specific time in the video when a new annotation (e.g., info card) is added to the list. In some implementations, the info cards may be more mobile friendly for smaller display screens than the intractable box annotations or the overlay content recommendation annotations.

The info card annotations may include text and/or background colored based on the color parameter and transparency set according to the transparency parameter. The size of the info card annotations may be set based on the size parameter and the location of the info cards may be set based on the location parameter. In some implementations, the time at which info card annotations are added to the bucket, thereby causing the icon to highlight, may be specified by the time parameter.

Figure 1 depicts a flow diagram of a method for using the mechanism to customize annotations in embedded content provided by an embedding source, in accordance with some implementations. First, at step 102, a request is sent to the mechanism (e.g., API) for content to

embed including one or more parameters for annotations to include in the content. The request may be sent to an API of an embedding source (e.g., content sharing platform server) that uploads content provided by various entities (e.g., users, companies, etc.). In some implementations, the request may also include the unique token described above. The API may add or modify annotations in the requested content according to the text, visible time (e.g., start/stop times), location (e.g., x/y coordinates), color, transparency, type (e.g., info card), size (e.g., length/width), and/or hyperlink (e.g., navigation endpoint) parameters specified in the request. In instances where the variables are stored server-side, the unique token may be used to fetch the variables and modify them based on the parameters provided in the request. The server may return the requested content including the added or modified annotations.

Next, at step 104, the content to embed including the annotations that were added or modified in view of the parameters may be received from the server. At step 106, the website or application may embed the received content including the annotations at a desired location. For example, if the publisher is developing a webpage that tells a story about a singer in a band, the content may be embedded below a paragraph that introduces the singer.

Next, at step 108, the annotations may be rendered in the embedded content at the specified times. In one example, the annotations may be info cards as discussed above and an icon representing the info card bucket may highlight at certain times when new info card annotations are added. In another example, at a specified time when the content is playing, the singer in the band may be highlighted according to a particular added annotation. As another example, the annotations may be rendered and move around within the content based on the location and time parameters.

Next, at step 110, a selection of the annotation that is rendered on the content may be

received. The selection may be received via any suitable input peripheral, such as a mouse, keyboard, touch-screen, or the like. As discussed above, the annotations that are added or modified may specify a hyperlink parameter to enable the developer to customize the navigation endpoints.

Thus, in response to selection of the annotation, at step 112, the hyperlink may cause navigation to a specified endpoint. In the example where the annotation includes highlighting the singer in the band, the navigation may redirect the focus of the webpage from the embedded content to a portion of the webpage including paragraphs of the story related to details of the singer. In addition to redirecting focus to other portions of the webpage in which the content is embedded, the annotations can include hyperlinks that redirect to another webpage or application, that redirect the embedded media player to other portions of the same content being presented or to a specific portion of other content that may be displayed in the embedded media player, and so forth.

The mechanism described herein allows publishers of websites or applications to embed their own annotations in embedded content that the publishers do not own. The mechanism may include an API that enables a publisher to add, modify, or highlight an annotation with the embedded content by applying an additional set of annotation parameters to the embedded content. The annotations may include info cards, text, highlights, overlay recommendations, intractable boxes, or pictures. The annotations may include a hyperlink that specifies endpoint navigation upon selection. As such, the user may watch an embedded video in a website or application, and annotations added or modified using the mechanism may appear at appropriate times that provide the user a hyperlink to a specified endpoint.

ABSTRACT

A mechanism for adding and/or modifying annotations in embedded content from an embedding source is presented herein. The mechanism enables a developer of a website or application to add or modify annotations in embedded or referred content without owning the content. The mechanism may apply a set of annotation parameters including text, type (e.g., info cards), size (e.g., length/width), location (e.g., x/y coordinates), time (e.g., start/stop times), and/or a hyperlink (e.g., navigation endpoint) to add or modify the annotations within the content. Thus, the developer may use the mechanism to render and display the annotations at particular times and locations when the content is playing, as well as specifying what happens when the annotations are selected. The content may include video, audio, social media posts, or the like.

Keywords: annotations, embedded content, referred content, video, music stream, social media post, application programming interface (API)

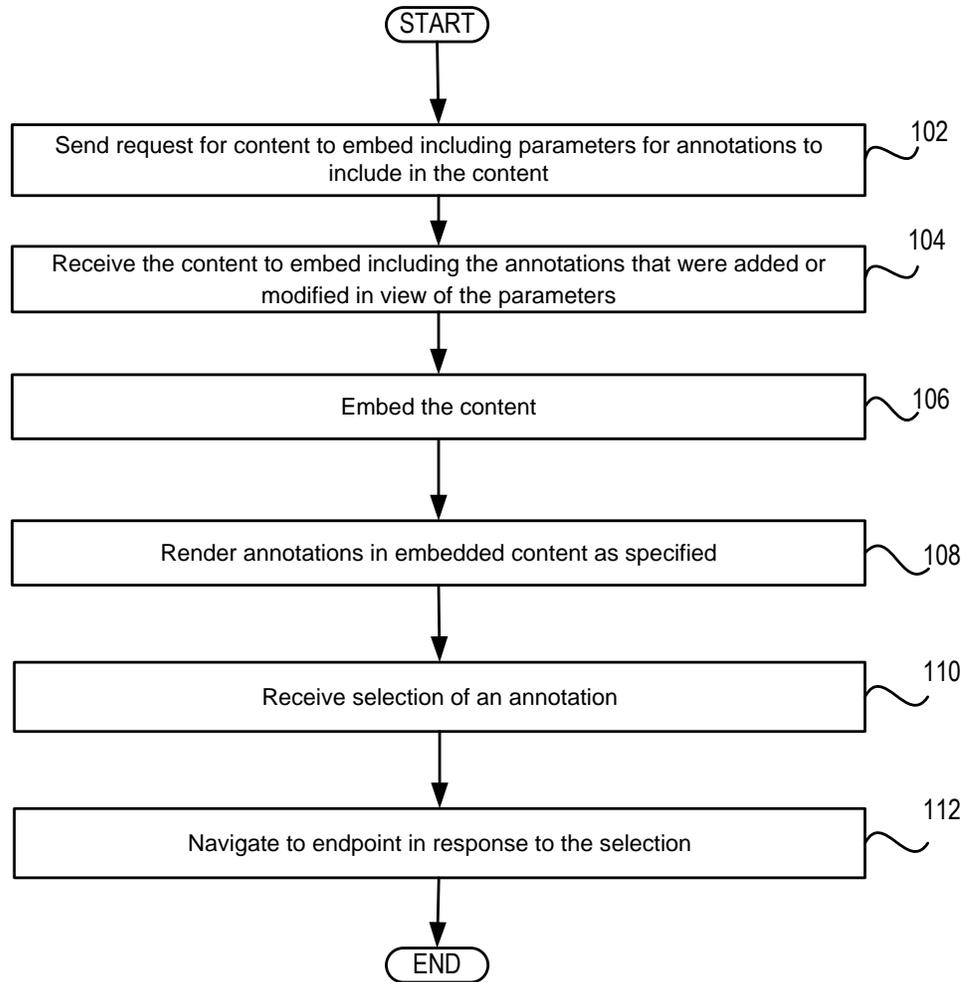


FIG. 1