

---

# **Mediathread Documentation**

***Release 2017.1***

**Columbia University CTL**

**Dec 22, 2018**



---

## Contents

---

<b>1</b>	<b>Supported Media</b>	<b>3</b>
1.1	Video . . . . .	3
1.2	Images . . . . .	4
1.3	Metadata . . . . .	4
1.4	Specific Websites . . . . .	6



Mediathread is Columbia University CTL's innovative, open source platform for exploration, analysis, and organization of web-based multimedia content. Mediathread connects to a variety of image and video collections (such as YouTube, Flickr, library databases, and course libraries), enabling users to lift items out of these collections and into an analysis environment. In Mediathread, items can then be clipped, annotated, organized, and embedded into essays and other written analysis.

Work in Mediathread can be shared with classmates or larger audiences, requiring students to formalize thinking, clarify interpretations, and improve arguments with evidence. A customized home page helps students track work being done by their classmates on shared items and projects. Instructors may also publish announcements, assignments, and model projects to the home page.

Mediathread is a Django application for multimedia annotations facilitating collaboration on video and image analysis, developed at the Columbia University Center for Teaching and Learning.

This site documents specific features and technical details of [Mediathread](#). There's additional info in our GitHub wiki: <https://github.com/ccnmtl/mediathread/wiki> which we are migrating over here.



---

## Supported Media

---

Supporting discovery of media on a web page is a daunting task. Reasons for this include:

- Many formats exist for video (and even images)
- The way images and video are embedded into web sites varies greatly
- Many web sites do not want you to embed their media into other sites.

Despite this, we're committed to expanding support for as many web sites as possible. If you're interested in including media at a certain site, and it's not working or faulty, please [submit an issue](#).

### 1.1 Video

Generally, MediaThread performs much better with streaming video, so users can seek deep into the video without loading the entire video.

- [Flowplayer v3](#).
  - Detection Requirements:
    - \* must be an OBJECT tag. If you embedded it with the flowplayer javascript, then it should be fine
    - \* If you have several videos ordered to play, currently the extension will only discover the first one or try to figure out which one is playing.
- YouTube
  - Strangely, we support YouTube embeds on other pages slightly better than the YouTube website itself.
  - The extension will bring in metadata such as author and description.
  - Detection Requirements:
    - \* must be an EMBED tag with `src="https://www.youtube.com/v/..."`
- HTML5 video tag
  - We have some experimental for the video tag.

- The main issue with using the video tag is that no single video format is supported in all browsers. Therefore, for cross-browser support, make sure to include h.264 and either WebM or ogg/theora.
- Quicktime
  - Quicktime streaming is generally reliable, however the plugin can exhibit buggy behavior in some browsers, so Flash or HTML5 is preferred.
    - \* Detection Requirements:
      - must be an OBJECT tag with classid="clsid:02BF25D5-8C17-4B23-BC80-D3488ABDDC6B"
      - the type must have 'quicktime' in it or the @src must end in ".mov"
- Vimeo
  - Detection Requirements
    - \* The extension can pull in any video on vimeo.com or embedded on an external site using the Moogaloop player syntax
    - \* The extension looks for an object with classid:D27CDB6E-AE6D-11cf-96B8-444553540000 and the fragment "moogaloop" in the object's value.
- Kaltura
  - Detection Requirements
    - \* The extension can pull in any video served in a Kaltura player assuming there is no access control/DRM issues.
    - \* The extension looks for an object with classid:D27CDB6E-AE6D-11cf-96B8-444553540000 and the fragment "kaltura" in the object's value.

## 1.2 Images

- IMG tags
  - IMGs that have either width or height greater than 400 pixels are assumed to be 'interesting'
  - Detection Requirements:
    - \* smaller images are not included to avoid logos, etc.
    - \* We also avoid images that have the words "logo" in their SRC or ID attributes and images that are marked as header or footer images.
- Embedded [Zoomify](#) tiled images

## 1.3 Metadata

We are interested in importing limited amounts of metadata to make available on the MediaThread asset page. The other advantage to these standards is that they are more reliable for detecting video urls (e.g. even if someone hasn't clicked an image to play the video).

- [unAPI](#) with format="pbcore":<http://pbcore.org/PBCore/PBCoreXMLSchema.html>
  - purely a metadata standard, but that's OK.
  - a good example was our motivated support of "WGBH OpenVault":<http://openvault.wgbh.org/>
  - Detection Requirements:



- \* must have LINK element with rel="unapi-server"
- \* currently, we only support the pages with a single ABBR[class="unapi-id"] element.
- \* current metadata items that we bring in:
  - title, description, contributor, coverage, rightsSummary, subject, and publisher

- **oEmbed.json**

- oEmbed is a great standard but does not quite have what we need for annotating, which requires more information about how to seek in a video and know what part of a video is being played. We support some extensions to oEmbed.json that we use on our own sites
  - \* **Tiling URL patterns** with xyztile:{url:XYZ\_URL, width:WIDTH, height: HEIGHT}
  - \* metadata:{Label1:[Value1, Value2, Value], Label2:[Value1]}
- Detection Requirements:
  - \* must have a LINK element on the page with type="application/json+oembed"
  - \* the oEmbed URL must reside within the same security domain as the current page (if the oEmbed link is on a different server, then the extension cannot make an AJAX request)

- **Microdata**

- Limited support uses two html attributes: @itemscope and @itemprop
- For adding some simple metadata, MediaThread supports limited microdata support on embedded images
  - \* One of the <img> parent elements should have an itemscope="itemscope" attribute
  - \* Any sub elements of that @itemscope element can have an @itemprop attribute where the text of the element is the value, or for <img>, <a> tags (and some others), the @src or @href attribute is the property. As an example, title and author are set by the following HTML on the image foo.jpg:

```
:: <div itemscope="itemscope"> <h2 itemprop="title">The Foo Image</h2> 
  <b>Author:</b> <span itemprop="author">Schuyler</span> </div>
```

- **Table Microformat for Images**

- We've added experimental metadata support for images through a <table> microformat so metadata can be added in contexts where authors do not have control of the underlying HTML (e.g. content management systems where HTML content or markdown is filtered).
- Requirements:
  - \* The <img> parent must be a <td> cell element.
  - \* The next table row must include the word "Metadata"
  - \* The subsequent table rows must then include two columns, where the first column's text is the metadata key name, and the second column is the metadata key value.
- An example:

```
| \2. !http://ccnmtl.columbia.edu/images/portfolio/thumbs/348.jpg (mediathread_
↪ logo) ! |
| \2. Metadata |
| author | Marc Raymond |
| title | MediaThread logo |
```

## 1.4 Specific Websites

- **ArtStor**
  - ArtStor subscribers, can import ArtStor images into MediaThread.
  - User Instructions:
    - \* You do not need to log in to ARTstor to bring images into Mediathread.
    - \* When you find an image you want to analyze in an ARTstor collection, click on its title (underneath its thumbnail).
    - \* A pop-up window will display the image’s metadata. At this point, click on the Analyze w/Mediathread extension in your main browser window. The image will then load in Mediathread.
- **Blake Archive**
  - example asset: <http://www.blakearchive.org/exist/blake/archive/object.xq?objectid=milton.a.illbk.33>
- **ClassPop**
  - example asset: <http://classpop.ccnmtl.columbia.edu/content/perspectives-freedom-speech>
- **Flickr**
  - MediaThread uses “Flickr’s javascript API”: [http://blog.jquery.com/2007/09/10/jquery-1-2-released/#Cross-Domain\\_getJSON\\_.28using\\_JSONP.29](http://blog.jquery.com/2007/09/10/jquery-1-2-released/#Cross-Domain_getJSON_.28using_JSONP.29) to get the largest image URL available and the metadata available to that image.
- **Moving Image Research Collection**
  - example asset: <http://mirc.sc.edu/islandora/object/usc%3A1974>
- **“Tibetan And Himalayan Library”: thlib.org**
  - We support the images at <http://www.thlib.org/places/monasteries/meru-nyingpa/murals/> which has a non-standard but consistent URL structure to retrieve image tiles for these high-resolution images.
- **Wikipedia**
- **YouTube**
  - It’s not always easy to figure out which video is playing (for example on channel pages), but we attempt to learn that and get the video thumbnail and title.
  - HTML5 video: <http://www.youtube.com/html5> is now supported, though the import may still work if you have flash installed, but are in youtube’s “HTML5 mode”
- **Vimeo**