

oEmbed

sharing content made easy

by B. Podlewski
for Liip Techday 2010

twitter: podlebar
blog: blog.robocode.ch

„oEmbed is an open format designed to allow embedding content from a website into another page.“

Benefits & Visions

- ▶ Embed content from any site
- ▶ Standard API for embeds
- ▶ Make embedding more easy

Request

News Feed Top News · Most Recent


HTTP GET Request
`http://vimeo.com/api/oembed.xml?url=http://vimeo.com/757219`



oEmbed API Endpoint

Response

Link



Meet the office
<http://vimeo.com/757219>
Got in early and setup a time lapse of everyone getting to work in the morning. Thought it might be interesting for everybody to see where we all sit while making the Vimeo.

Music: End by Lo-Fi Fnk

◀ ▶ 1 of 57 Choose a Thumbnail

No Thumbnail

Share

The Request

`http://vimeo.com/api/oembed.xml?url=http://vimeo.com/757219`

API Endpoint

encoded url

oEmbed speaks JSON or XML

The Response (XML)

```
<?xml version="1.0" encoding="UTF-8"?>
<oembed>
  <type>video</type>
  <version>1.0</version>
  <provider_name>Vimeo</provider_name>
  <provider_url>http://www.vimeo.com/</provider_url>
  <title>Meet the office</title>
  <author_name>Blake Whitman</author_name>
  <author_url>http://www.vimeo.com/blakewhitman</author_url>
  <is_plus>0</is_plus>
  <html>&lt;object type="application/x-shockwave-flash";...&lt;/object&gt;</html>
  <width>504</width>
  <height>380</height>
  <thumbnail_url>
    http://90.media.vimeo.com/d1/5/38/21/85/thumbnail-38218529.jpg
  </thumbnail_url>
  <thumbnail_width>160</thumbnail_width>
  <thumbnail_height>120</thumbnail_height>
  <video_id>757219</video_id>
</oembed>
```

The Response (JSON)

```
{  
  "type": "video",  
  "version": "1.0",  
  "provider_name": "Vimeo",  
  "provider_url": "http://www.vimeo.com/",  
  "title": "Meet the office",  
  "author_name": "Blake Whitman",  
  "author_url": "http://www.vimeo.com/blakewhitman",  
  "is_plus": "0",  
  "html": "&lt;object type=&quot;application...&lt;/object&gt;",  
  "width": "504",  
  "height": "380",  
  "thumbnail_url": "http://90.media.vimeo.com/d1...85/thumbnail-38218529.jpg",  
  "thumbnail_width": "160",  
  "thumbnail_height": "120",  
  "video_id": "757219"  
}
```


What informations are shared?

- ▶ Provider Informations
- ▶ Author Informations
- ▶ Content type
- ▶ Dimensions
- ▶ Thumbnail Paths
- ▶ HTML Code for embedding

Response parameters

valid for every type

type (required) - photo, video, rich or link

version (required) - 1.0 at the moment

title (optional) - Sweet kittens

author_name (optional) - Bart

author_url (optional) - <http://www.robocode.ch>

provider_name (optional) - Vimeo

provider_url (optional) - <http://www.vimeo.com>

cache_age (optional) - 3600

thumbnail_url (optional) - <http://...../.../kittens.jpg>

thumbnail_width (optional) - 200

thumbnail_height (optional) - 75

Response parameters (Photos)

url - The source URL of the Image (Only http or https)

width - In Pixels

height - In Pixels too.. what else..

Response parameters (Videos & Rich)

html - The HTML required to embed the video player

width - In Pixels

height - In Pixels

Response parameters (Link)

Responses of this type allow a provider to return any generic embed data (such as **title** and **author_name**), without providing either the **url** or **html** parameters. The consumer may then link to the resource, using the URL specified in the original request.

Consumers

Facebook * WordPress * Friendfeed * StudiVZ * NFB

Providers

5min * Amazon * Flickr * Google Video * Hulu * Imdb *
Metacafe * Myspace Videos * Qik * Revision3 * Screenr *
Slideshare * Twitpic * Viddler * Vimeo * Wikipedia *
WordPress * YouTube * Animoto * Mobypictures

YouTube (<http://www.youtube.com/>)

- ▶ API endpoint: <http://www.youtube.com/oembed>
- ▶ Supports discovery via <link> tags
- ▶ Example: <http://www.youtube.com/oembed?url=http%3A//www.youtube.com/watch?v%3D-UUx10KOWIE&format=xml>

Flickr (<http://www.flickr.com/>)

- ▶ URL scheme: http://*.flickr.com/
- ▶ API endpoint: <http://www.flickr.com/services/oembed/>
- ▶ Example: <http://flickr.com/services/oembed?url=http%3A//flickr.com/photos/bees/2362225867/>

Viddler (<http://www.viddler.com/>)

- ▶ URL scheme: http://*.viddler.com/
- ▶ API endpoint: <http://lab.viddler.com/services/oembed/>
- ▶ Documentation: <http://wiki.developers.viddler.com/index.php/OEMBED>
- ▶ Example: <http://lab.viddler.com/services/oembed/?url=http%3A//www.viddler.com/explore/cdevroe/videos/424/>

Revision3 (<http://revision3.com/>)

- ▶ URL scheme: http://*.revision3.com/
- ▶ API endpoint: <http://revision3.com/api/oembed/>
- ▶ Example: <http://revision3.com/api/oembed?url=http%3A//revision3.com/diggation/2008-04-17xsanned/&format=xml>

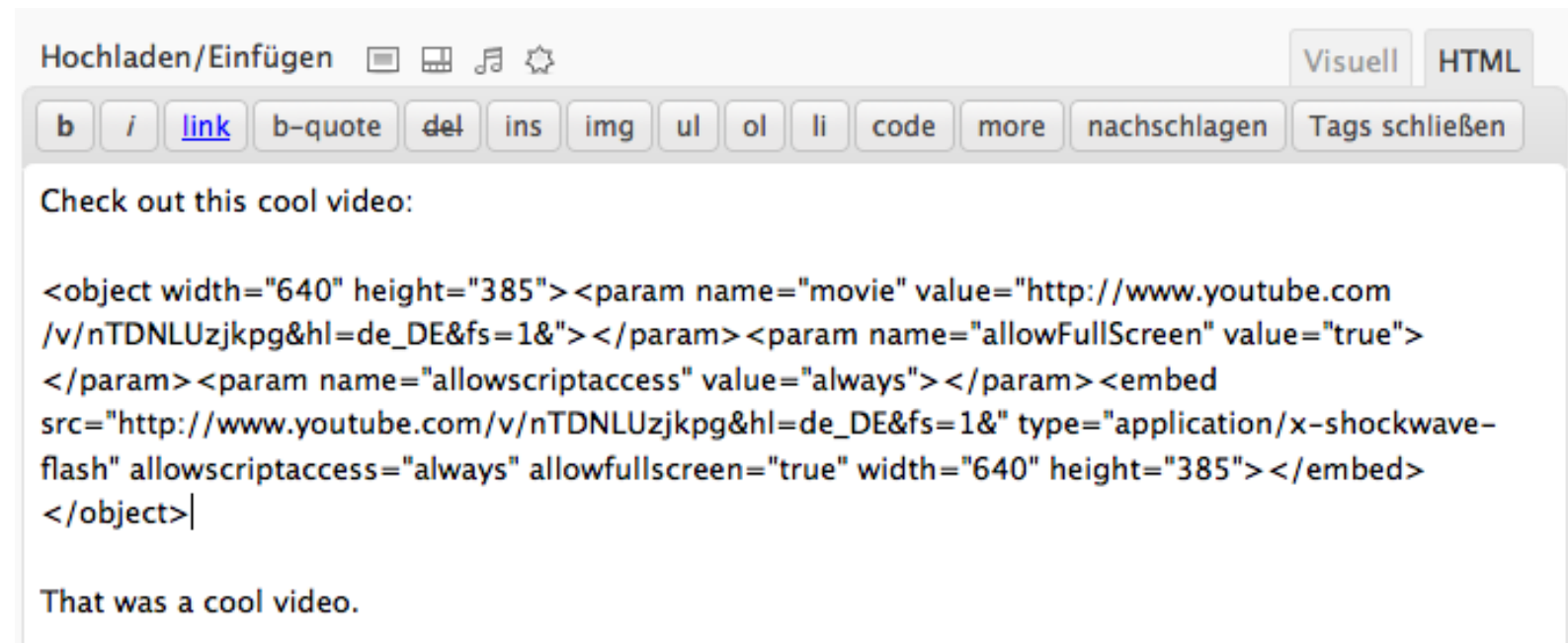
Vimeo (<http://vimeo.com/>)





- ▶ Docs: <http://vimeo.com/api/docs/oembed>
- ▶ URL scheme (video): <http://www.vimeo.com/>
- ▶ URL scheme (group video): http://www.vimeo.com/groups/*/
- ▶ Endpoint: <http://www.vimeo.com/api/oembed.{format}>
- ▶ Example: <http://www.vimeo.com/api/oembed.xml?url=http%3A//www.vimeo.com/757219>

Why should i use it?

Ugly

95% of all bloggers
don't know what
this is



Hochladen/Einfügen     Visuell HTML

b *i* [link](#) b-quote ~~del~~ ins img ul ol li code more nachschlagen Tags schließen

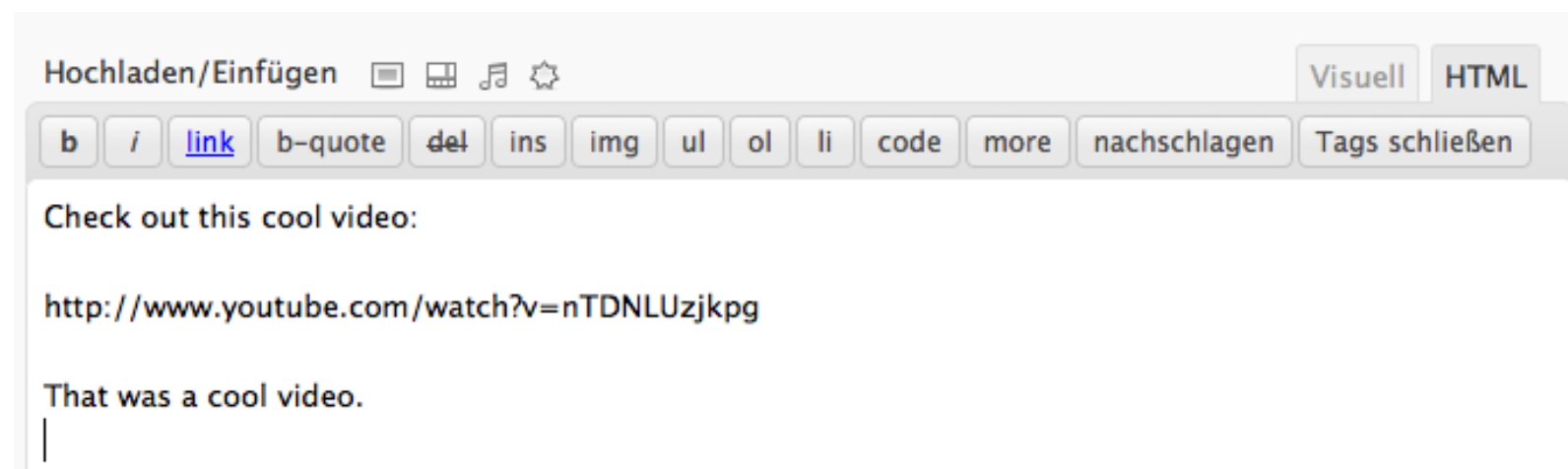
Check out this cool video:





```
<object width="640" height="385"><param name="movie" value="http://www.youtube.com/v/nTDNLUzjkpg&hl=de_DE&fs=1"></param><param name="allowFullScreen" value="true"></param><param name="allowsriptaccess" value="always"></param><embed src="http://www.youtube.com/v/nTDNLUzjkpg&hl=de_DE&fs=1" type="application/x-shockwave-flash" allowsriptaccess="always" allowfullscreen="true" width="640" height="385"></embed></object>
```

That was a cool video.

Awesome

100% of all bloggers know
what this is..
because it's
human readable



Hochladen/Einfügen     Visuell HTML

b *i* [link](#) b-quote ~~del~~ ins img ul ol li code more nachschlagen Tags schließen

Check out this cool video:

<http://www.youtube.com/watch?v=nTDNLUzjkpg>

That was a cool video.

|

What the Provider has to do

- ▶ Define the XML and JSON Endpoint
`http://vimeo.com/api/oembed.xml`
`http://vimeo.com/api/oembed.json`
- ▶ Configure URL Scheme
- ▶ add Discovery Link to the <head>

URL Scheme

`http://www.flickr.com/photos/*` OK
`http://www.flickr.com/photos/*/foo/` OK
`http://*.flickr.com/photos/*` OK
`http://*.com/photos/*` NOT OK
`*://www.flickr.com/photos/*` NOT OK

URL scheme: `http://www.mobypicture.com/user/*/view/*`

API endpoint: `http://api.mobypicture.com/oEmbed`

Supports discovery via tags

Example: `http://api.mobypicture.com/oEmbed?url=http://www.mobypicture.com/user/marjolijn/view/90053&format=xml`



What the Consumer has to do

- ▶ Handel the User-Input
- ▶ Know the Endpoint of the Provider
- ▶ Get the Discovery link from the provider

The Discovery Link

put this into your <head> if you are a provider

```
<link rel="alternate" type="application/json+oembed"  
  href="http://flickr.com/services/oembed?url=http%3A//flickr.com/photos/bees/2362225867/&format=json"  
  title="Bacon Lollys oEmbed Profile" />
```

and / or

```
<link rel="alternate" type="text/xml+oembed"  
  href="http://flickr.com/services/oembed?url=http%3A//flickr.com/photos/bees/2362225867/&format=xml"  
  title="Bacon Lollys oEmbed Profile" />
```

The URLs contained within the **href** attribute should be the full oEmbed endpoint plus URL and any needed format parameter.

The **type** attribute must contain either **application/json+oembed** for JSON responses, or **text/xml+oembed** for XML.

The bad Parts

- ▶ oEmbed is NOT or NOT YET Restfull
- ▶ It's 2010.. we have HTML5. Why provide Embedcode instead of Videosource-Url?
- ▶ Kids can embed Videos that are marked for Adults (on YouTube)
- ▶ What about Music?
- ▶ No Check for <meta> Tags like Facebok does
- ▶ Getting the Endpoint-Url of each Link is a lot of work.. but unnecessary... because..

..there is ..



„oohEmbed is an oEmbed compatible provider of HTML embed codes for various web sites.“

How to work with oohEmbed

and save time

instead of:

<http://vimeo.com/api/oembed.xml?url=http%3A//vimeo.com/757219>

or

<http://www.youtube.com/oembed?url=http%3A//www.youtube.com/watch%3Fv%3DbDOYN-6gdRE&format=json>

use

<http://oohembed.com/oohembed/?url=http%3A//vimeo.com/757219>

and get some nice JSON (XML not supported)

Alternative Share <meta> Tags

for Facebook

```
<link rel="image_src" href="http://www.foo.ch/.../preview.jpg"/>  
<meta name="medium" content="video" />  
<meta name="video_type" content="application/x-shockwave-flash"/>  
<meta name="video_height" content="300"/>  
<meta name="video_width" content="400"/>  
<link rel="video_src" href="http://www.youtube.com/v/uENEaiY6JEY"/>  
<link rel="target_url" href="http://www.foo.ch"/>
```

Thank you..

Resources

Infos:

<http://www.oembed.com/>

<http://oohembed.com/>

<http://embed.ly/>

http://www.webmonkey.com/2010/02/get_started_with_oembed/

http://developer.yahoo.net/blog/archives/2008/05/oembed_embeddin.html

http://wiki.developers.facebook.com/index.php/Facebook_Share/Specifying_Meta_Tags

<http://help.animoto.com/entries/109992-oembed-api>

Images:

<http://www.flickr.com/photos/11543883@N06/2370375218/>