

afterUploadVideo

RELEASE 2.2.0 OR ABOVE

This event occurs after a video is uploaded.

Adding Event

Using API

```
var editorId = 'synapEditor';
var editorConfig = {};
var html = '';
var editor = new SynapEditor(editorId, editorConfig, html);

editor.setEventListener('afterUploadVideo', function (e) {
});
```

Function

```
var editorId = 'synapEditor';
var editorConfig = {};
var html = '';

function SynapEditorAfterUploadVideo(e) {
}

new SynapEditor(editorId, editorConfig, html);
```

When the Editor is initialized

```
var editorId = 'synapEditor';
var editorConfig = {};
var html = '';
var eventListeners = {
  afterUploadVideo: function (e) {
  }
};

new SynapEditor(editorId, editorConfig, html, eventListeners);
```

Object Delivered through Functions

In the form of parameter e delivered through functions

e

```
// release 2.2.0
{
    editor: SynapEditor,
    eventType: 'afterUploadVideo',
    cancelable: false,
    returnValue: null,
    path: '/upload/path/filename.mp4', // The upload path.
    elementId: 'se-123456-abcdefg-7890' // HTML Element Id (wrapper)
}
// release 2.2.1 or above
{
    editor: SynapEditor,
    eventType: 'afterUploadVideo',
    cancelable: false,
    returnValue: null,
    fileType: 'video', // [2.2.1] File type
    path: '/upload/path/filename.mp4',
    elementId: 'se-123456-abcdefg-7890'
}
// release 2.10.0 or above
{
    editor: SynapEditor,
    eventType: 'afterUploadVideo',
    cancelable: false,
    returnValue: null,
    fileType: 'video',
    path: '/upload/path/filename.mp4',
    responseData: {}, // [2.10.0] complete response data
    elementId: 'se-123456-abcdefg-7890'
}
// release 2.15.0 or above
{
    editor: SynapEditor,
    eventType: 'afterUploadVideo',
    cancelable: false,
    returnValue: null,
    fileType: 'video',
    path: '/upload/path/filename.mp4',
    responseData: {},
    elementId: 'se-123456-abcdefg-7890',
    error: undefined // [2.15.0] Error information when upload fails
}
}
```

Video upload failed

RELEASE 2.15.0 OR ABOVE

Starting with the 2.15.0 release, the afterUploadVideo event is also raised when an upload fails.

An Error object is passed to the error property of the object that is passed to the afterUploadVideo event when upload fails.

Example of removing failed video when upload fails

```
editor.setEventListener('afterUploadVideo', function (e) {
    if (e.error && e.error.status === 500) {
        e.editor.execCommand('showDialog', 'alert', 'Video upload failed.');
```

```
        var video = e.editor.getAPIModelById(e.elementId);
        video.remove();
    }
});
```