

Mindshare HTTP API

Version 0.3 - November, 17 2005

This document reflects the HTTP API to controlling the Mindshare system as of the initial release of Mindshare 3.1. This API is currently not supported as a product. The main shortcoming of this API, as it stands, is the lack of a good feedback mechanism. HTTP calls return HTML, not XML. Users of this API that wish to determine the state of the Mindshare system must extract that information from the returned HTML. Also, on occasion parameters values are required even when it's not clear what function they serve.

This API is not carefully tested nor is it supported. The information below is thought to be complete and correct, but no warranty is made as to its correctness or functionality.

Table of Contents

[API Calls](#)
[Typical Usage and Example](#)
[Start Mindshare System](#)
[Stop Mindshare System](#)
[Create New Presentation and Define Parameter](#)
[Start Encoding](#)
[Add Chapter During Live Encoding](#)
[Stop Encoding](#)
[Add Chapter After Live Encoding is Stopped](#)
[Publish MP4 File](#)
[Informational Requests](#)

[TOP](#)

API Calls

In the description below, actual text sent in the HTTP call is written in *typewriter font*, where as meta information that is supposed to be filled in is written in *italicized font*.

All the API calls are HTTP requests made to the Mindshare machine. If the machine is set up to use HTTP authentication using the Mindshare GUI (set in the settings->Network tab), then the same authentication must be used in the HTTP call. Parameters passed to the Mindshare system are set using the HTTP GET or POST method. Most requests use the GET method, described briefly below. Some requests, such as those that include file upload, use the POST method.

The HTTP GET method has the form:

```
http://Mindshare_IP_Address/cgi-bin/do.exe?action=ACTION&param1=value1&param2=value2&...&paramN=valueN
```

Note that spaces in the GET method must be represented as '+'.

In the documentation below, the *ACTION* and parameters are described for each desired functionality. Case is relevant.

[TOP](#)

Typical Usage and Example

A typical usage scenario consists of the following steps:

1. Start the Mindshare system
2. Define the presentation parameters
3. Start the encoding
4. Create indices during the encoding
5. Stop the presentation
6. Add yet more indices
7. Publish the MP4 presentation
8. Shut down the Mindshare system

Below are sample HTTP calls achieving these steps

1. Start the Mindshare system:
`http://Mindshare_IP_Address/cgi-bin/do.exe?action=start`
2. Define the presentation parameters:
`http://Mindshare_IP_Address/cgi-bin/do.exe?action=savePresentation&liveDate=August+5&liveSpeaker=Jimmy-Bob&livePresentation=A+Demo&template=blue800x600/blue800x600.ebp&transSpeed=800x600_medium`
3. Start the encoding:
`http://Mindshare_IP_Address/cgi-bin/do.exe?action=dashBoard&startEncoding.x=1`
4. Create indices during the encoding:
`http://Mindshare_IP_Address/cgi-bin/do.exe?action=dashBoard&indexText=Chapter+1&indexDesc=Chapter+1+Description&indexTimeH=&indexTimeM=&indexTimeS=&addIndex.x=1`

5. Stop the presentation:
`http://Mindshare_IP_Address/cgi-bin/do.exe?action=dashBoard&stopEncoding.x=1`
6. Add yet more indices:
`http://Mindshare_IP_Address/cgi-bin/do.exe?action=dashBoard&indexText=Chapter+3&indexDesc=Chapter+3+Description&indexTimeH=&indexTimeM=&indexTimeS=20&addIndex.x=1`
7. Publish the MP4 presentation:
`http://Mindshare_IP_Address/cgi-bin/do.exe?action=publish&fileName=outputFileName&publishmp4=on&eventDescription=event+description&ublish.x=1`
8. Shut down the Mindshare system:
`http://Mindshare_IP_Address/cgi-bin/do.exe?action=stop`

[TOP](#)

Start Mindshare System

| Parameter Name | Parameter Value | Required/Optional |
|----------------|-----------------|-------------------|
| action | start | Required |

This call starts the two 4Caster applications and gets the Mindshare system ready to be used.

[TOP](#)

Stop Mindshare System

| Parameter Name | Parameter Value | Required/Optional |
|----------------|-----------------|-------------------|
| action | stop | Required |

This call stops the two 4Caster processes. This call should not be called while the encoders are encoding. It should be only called when all usage of the Mindshare system is complete, as a final terminating call.

[TOP](#)

Create New Presentation and Define Parameter

| Parameter Name | Parameter Value | Required/Optional |
|----------------|------------------|-------------------|
| action | savePresentation | Required |
| eventName | The event name. | Optional |

| | | |
|-------------------|---|----------|
| liveDate | The date of the event. | Optional |
| liveSpeaker | The date of the event. | Optional |
| livePresentation | The topic. | Optional |
| presentationImage | Path to local image to be included in the presentation. This only works when using a POST method which includes the image using the HTTP file upload mechanism. | Optional |
| template | The name of the 4Mation to be used. The path must be relative to the Mindshare directory - e.g. blue1024x768/blue1024x768.ebp | Required |
| transSpeed | The 4Caster preset to use. Only the file name should be given, without any path or extension - e.g., 1024x768_low_bitrate | Required |

[TOP](#)

Start Encoding

| Parameter Name | Parameter Value | Required/Optional |
|-----------------|--|-------------------|
| action | dashBoard | Required |
| startEncoding.x | A value of 1 must be passed to start the encoding. | Required |

[TOP](#)

Add Chapter During Live Encoding

When this call is made, the index is inserted at the time the HTTP request is made in the live presentation.

| Parameter Name | Parameter Value | Required/Optional |
|----------------|--|-------------------|
| action | dashBoard | Required |
| indexTimeH | Must be blank. | Required |
| indexTimeM | Must be blank. | Required |
| indexTimeS | Must be blank. | Required |
| addIndex.x | Must have value 1 | Required |
| indexText | The chapter title. | Optional |
| indexDesc | The chapter description. | Optional |
| indexInvisible | Use valued "checked" to make chapter invisible. | Optional |
| indexFile | Use path of local image file to be included in the image. This will only when when using the HTTP POST method with HTTP file upload. | Optional |

[TOP](#)

Stop Encoding

| Parameter Name | Parameter Value | Required/Optional |
|----------------|---|-------------------|
| action | dashBoard | Required |
| stopEncoding.x | A value of 1 must be passed to stop the encoding. | Required |

[TOP](#)

Add Chapter After Live Encoding is Stopped

When this call is made, the time of the added chapter is included in the parameters. Adding a time longer than the duration of the recorded presentation will result in an error. Not specifying the time will also result in an error.

| Parameter Name | Parameter Value | Required/Optional |
|----------------|--|-------------------|
| action | dashBoard | Required |
| indexTimeH | The hours in the chapter time. | Required |
| indexTimeM | The minutes in the chapter time. | Required |
| indexTimeS | The seconds in the chapter time. | Required |
| addIndex.x | Must have value 1 | Required |
| indexText | The chapter title. | Optional |
| indexDesc | The chapter description. | Optional |
| indexInvisible | Use valued "checked" to make chapter invisible. | Optional |
| indexFile | Use path of local image file to be included in the image. This will only when when using the HTTP POST method with HTTP file upload. | Optional |

[TOP](#)

Publish MP4 File

| Parameter Name | Parameter Value | Required/Optional |
|----------------|-----------------|-------------------|
| action | publish | Required |

| | | |
|------------------|---|----------|
| fileName | The name of the mp4 file to be published; no "mp4" extension is needed. | Required |
| publishmp4 | Must have value "on". | Required |
| archive | Should have value "on" to create an MP4 archive for later editing. | Optional |
| eventDescription | An optional description that is included in the library page listing. | Optional |
| publish.x | Must have value 1. | Required |

[TOP](#)

Informational Requests

There are a variety of informational requests that can be made to the Mindshare system. These include `action=library`, `action=settings`, and others. These are currently not documented.