

# Content Partner provider Api Guide

## Prerequisites and Expectations

1. Expected Request rate is 1 TPS.
2. this signature can have minor changes during the course of development.
3. UAT and production API url's will be shared after the development process.
4. API is sorting based upon `startDate` param, so if new content is added there is no guarantee that it will come at the start because `startDate` can be anything in the newly added content as well.

## Content Platform Endpoints

The following are the base URLs for different environments

	Environment	Base URL
1	Dev	<a href="https://qa-catalog-api.hotstar.com/">https://qa-catalog-api.hotstar.com/</a>
2	QA	<a href="https://qa-catalog-api.hotstar.com/">https://qa-catalog-api.hotstar.com/</a>
3	Pre Prod	<a href="https://pp-catalog-api.hotstar.com/">https://pp-catalog-api.hotstar.com/</a>
4	Prod	<a href="https://catalog-api.hotstar.com/">https://catalog-api.hotstar.com/</a>

## API Contract

### Request Header Parameters

Request Header Parameters: ( generic parameter should be appended to all API calls )

	Name	Description
1	<code>partner</code>	A unique value for partner identification, Hotstar will provide on the time of integration. This value should be added in the query parameter <code>partner</code> and header <code>x-partner-name</code> for every request in this document
2	<code>hdnea</code>	<p>The API is protected with a secure hMAC key to call this. To use this API, you will have to pass the token in the header ( <code>hdnea</code> ).</p> <p>To generate the token, request for the token generation library and secret from the Content Platform team</p> <p>Please use any of the following libraries for <code>hdnea</code> token generation:</p> <p><a href="https://github.com/orgs/akamai/repositories?q=token&amp;type=all&amp;language=&amp;sort=">https://github.com/orgs/akamai/repositories?q=token&amp;type=all&amp;language=&amp;sort=</a></p> <p>Below command will generate a token for java program:</p> <pre>1 java AkamaiToken -k &lt;secret&gt; -a /* -s now -w 2000</pre>

where `-w` is time window in seconds for which token should be valid.  
The token starts after `hdnts=`  
i.e If the output is `hdnts=st=1567324pp~exp=1567524057~acl=/*~hmac=e3ed3d`  
Then token is `st=1567324pp~exp=1567524057~acl=/*~hmac=e3ed3d`

Similarly for python:

```
1 python akamai_token_v2.py -k <secret> -a /* -s now -w 2000
```

Note:

1. Maximum size allowed is `1000`
2. Offset + Size should not exceed `10,000` limit, so consider using `fromStartDate` and `toStartDate` filter in request (values for starDate is in epoch format in seconds)
3. `fromStartDate` and `toStartDate` filter should be only used for getting older data. So this activity shouldn't run more than twice in a week
4. `fromUpdateDate` and `toUpdateDate` filter is also there to fetch all records that got updated in the time window. (values for the fields are also is in epoch format in seconds)
5. Use `fromUpdateDate` and `toUpdateDate` filter if you want to fetch incremental update changes done in some point of time. Some partners can find it pretty useful to fetch changes that has happened in last 24 hours. So this filter can be vital for such use cases. It is suggested to be used with the frequency of 4 hours window for polling. For matches, the frequency can be further increased since updates on matches happen more frequently.
6. In the case of 5XX from server or increase in response time is more than 5 sec, clients should add circuit break logic of retrying after 1 minute only.
7. The `premium` flag is optional.
8. The rate limit for API is 1 request per second, exceeding which will result in `429` HTTP status code in response.
9. The `playUri` can be used to fetch the playable URL of the content. Note: Only in case of non-encrypted content `playUri` can be shared due to limitations, in case of encrypted content `deepLinkUrl` will be used

## API Signatures

*The headers marked in bold are mandatory*

### Shows

*The following API is used to fetch shows sorted by start time*

#### Request Parameters

Item	Value
HTTP Method	GET
Request Path	/show/search

Query Parameters	
<code>orderBy</code>	The parameter to use for ordering the response, possible value (startDate / contentId). Recommended is contentId as startDate gives duplicate result
<code>order</code>	The sort order to be used
<code>offset</code>	Pagination offset
<code>size</code>	Total number of items per page
<code>premium</code>	Whether or not premium content is required
Headers	
<code>x-country-code</code>	Country code
<code>x-platform-code</code>	Code of the platform
<code>x-partner-name</code>	Name of the partner calling this API
<code>x-region-code</code>	
<code>x-client-code</code>	

## Sample Request

```

1  Curl 'http://{base_url}/show/search?orderBy=contentId&order=desc&offset=1&size=10&premium=false' -i \
2    -H 'x-country-code: in' \
3    -H 'x-platform-code: PCTV' \
4    -H 'x-partner-name: <partner name>' \
5    -H 'x-region-code: DL' \
6    -H 'x-client-code: pt'
```

## Sample Response

**Note:** All the response fields are mandatory except below

1. Images with Type vertical
2. searchKeywords
3. trailers
4. actors
5. anchors
6. directors
7. producers
8. channelObject

## Response Schema

```

1  {
2      "body": {
3          "results": {
```

```
4     "items": [
5         {
6             "id": 3257,
7             "title": "Marvel Studios ASSEMBLED",
8             "description": "ASSEMBLED is an immersive series of documentary-style specials examining th
9             "contentType": "SHOW",
10            "categoryId": 1260056004,
11            "contentId": 1260056004,
12            "thumbnail": "https://img.hotstar.com/image/upload/f_auto,t_hcdl/sources/r1/cms/prod/6336/1
13            "startDate": 1615536000,
14            "endDate": 4102424940,
15            "genre": [
16                "Documentary",
17                "Superhero"
18            ],
19            "lang": [
20                "English"
21            ],
22            "premium": true,
23            "images": [
24                {
25                    "url": "https://img.hotstar.com/image/upload/f_auto/sources/r1/cms/prod/6336/103633
26                    "transformation": "hcdl",
27                    "type": "HORIZONTAL"
28                },
29                {
30                    "url": "https://img.hotstar.com/image/upload/f_auto,t_vl/sources/r1/cms/prod/6337/1
31                    "transformation": "vl",
32                    "type": "VERTICAL"
33                }
34            ],
35            "sourceImages": [
36                {
37                    "url": "https://img1.hotstarext.com/image/upload/f_auto/sources/r1/cms/prod/6337/10
38                    "type": "VERTICAL"
39                },
40                {
41                    "url": "https://img1.hotstarext.com/image/upload/f_auto/sources/r1/cms/prod/6336/10
42                    "type": "HORIZONTAL"
43                }
44            ],
45            "vip": true,
46            "searchKeywords": [
47                "marvel studios",
48                " as assembled",
49                " as sembeled",
50                " marvel shows",
51                " marvel documentary",
52                " marvel making",
53                " mcu making",
54                " mcu documentaries",
55                " marvel docus",
56                "Marvel Studios ASSEMBLED",
57                "Marvel Studios ASSEMBLED",
58                "English",
59                "Documentary",
60                "Superhero",
61                "Marvel"
```

```
62     ],
63     "langObjs": [
64       {
65         "hide": false,
66         "id": 3,
67         "name": "English",
68         "iso3code": "eng",
69         "displayName": "English"
70       }
71     ],
72     "parentalRating": 2,
73     "parentalRatingName": "U/A 13+",
74     "locators": [
75       {
76         "url": "https://hotstar.com/1260056004",
77         "platform": "web"
78       },
79       {
80         "url": "hotstar://1260056004",
81         "platform": "ios"
82       },
83       {
84         "url": "hotstar://1260056004",
85         "platform": "android"
86       },
87       {
88         "url": "hotstar://1260056004",
89         "platform": "appletv"
90       },
91       {
92         "url": "hotstar://1260056004",
93         "platform": "androidtv"
94       },
95       {
96         "url": "https://livingroom.hotstar.com/1260056004",
97         "platform": "weblr"
98       }
99     ],
100    "channelObject": {
101      "id": 54,
102      "name": "Hotstar Specials",
103      "contentId": 1260003991
104    }
105  },
106],
107 "totalResults": 2260,
108 "offset": 0,
109 "size": 1,
110 "nextOffsetURL": "https://cmscontentpartner.hotstar-labs.com/show/search?offset=1&size=1&orderBy=st
111 "totalPageResults": 1,
112 "totalPages": 2260
113 },
114 },
115 "statusCode": "OK",
116 "statusCodeValue": 200
117 }
118 }
```

## Seasons

The following API is used to fetch seasons, filter by showID and sorted by Display No

### Request Parameters

Item	Value
HTTP Method	GET
Request Path	/season/search
Query Parameters	
showId	<p>The parameter is used to filter out the season of a respective show. Make sure it is the showId and not show ContentId. In season response Api, "id" field represents the showId.</p>
order	The sort order to be used
offset	Pagination offset
size	Total number of items per page
partner	Name of the partner
Headers	
x-country-code	Country code
x-platform-code	Code of the platform
x-partner-name	Name of the partner calling this API
x-region-code	
x-client-code	

### Sample Request

```
1 curl -X GET \
2 'http://{base_url}/season/search?showId=30&offset=0&size=100&order=desc&partner=<partner name>' \
3   -H 'Postman-Token: eb3a0e23-dbaf-4c8d-a65e-e28ea162a3d0' \
4   -H 'x-country-code: in' \
5   -H 'x-partner-name: <partner name>' \
6   -H 'x-platform-code: pctv'
```

### Sample Response

**Note:** All the response fields are mandatory except below

1. Images with Type vertical

### Response Schema

```
1 {
2   "body": {
```

```

3     "results": {
4         "items": [
5             {
6                 "displaySeasonNo": 28,
7                 "episodeCnt": 162,
8                 "title": "Chapter 28",
9                 "channelId": 3,
10                "showId": 1,
11                "id": 1793,
12                "categoryId": 3222,
13                "contentId": 3282,
14                "images": [
15                    {
16                        "url": "https://secure-media.hotstar.com/r1/thumbs/PCTV/82/3282/PCTV-3282-hcd1.jpg",
17                        "transformation": "hcd1",
18                        "type": "HORIZONTAL"
19                    },
20                    {
21                        "url": "https://secure-media.hotstar.com/r1/thumbs/PCTV/82/3282/PCTV-3282-v1.jpg",
22                        "transformation": "v1",
23                        "type": "VERTICAL"
24                    }
25                ],
26                "thumbnail": "https://secure-media.hotstar.com/r1/thumbs/PCTV/82/3282/PCTV-3282-hm.jpg"
27            }
28
29        ],
30        "totalResults" : 34,
31        "nextOffsetURL": "http://localhost:8089/season/search?showId=30&offset=7&size=5&order=desc",
32
33    }
34 },
35 "statusCode": "OK",
36 "statusCodeValue": 200
37 }
```

## Episodes

The following API is used to fetch episodes, ordered by time

Note: `showId` is optional

### Request Parameters

Item	Value
HTTP Method	GET
Request Path	/episode/search

  

Query Parameters	
<code>fromStartDate</code>	Date to start filtering from
<code>toStartDate</code>	Date till when the filtering should stop
<code>fromUpdateDate</code>	Date to start filtering updatedRecord date
<code>toUpdateDate</code>	Date till when filtering updatedRecord date stop

<code>showId</code>	The parameter to use for ordering the response
<code>orderBy</code>	The parameter to use for ordering the response, possible value (startDate / contentId)
<code>order</code>	The sort order to be used
<code>offset</code>	Pagination offset
<code>size</code>	Total number of items per page
<code>premium</code>	Whether or not premium content should be fetched
<code>partner</code>	Name of the partner
Headers	
<code>x-country-code</code>	Country code
<code>x-platform-code</code>	Code of the platform
<code>x-partner-name</code>	Name of the partner calling this API
<code>x-region-code</code>	
<code>x-client-code</code>	

## Sample Request

```

1 curl 'http://{base_url}/episode/search?fromStartDate=0&toStartDate=1528991761&showId=6&orderBy=startDate&order=de
2   -H 'x-country-code: in' \
3   -H 'x-platform-code: PCTV' \
4   -H 'x-partner-name: <partner name>' \
5   -H 'x-region-code: DL' \
6   -H 'x-client-code: pt'
```

## Sample Response

**Note:** All the response fields are mandatory except below

1. Images with Type vertical
2. searchKeywords
3. trailers
4. actors
5. anchors
6. directors
7. producers
8. playUri (This is functional only for JIO partner)

## Response Schema

```

1 {
2   "body": {
3     "results": {
4       "items": [
5         {
6           "id": 1069835,
7           "contentId": "1700008109",
```

```
8     "title": "Mangal Helps Gauri",
9     "description": "Mangal helps Gauri as she tries to exit the well. Following that, Laxmi que
10    "contentType": "EPISODE",
11    "duration": 1250,
12    "genre": [
13        "Drama"
14    ],
15    "lang": [
16        "Marathi"
17    ],
18    "broadCastDate": 1682352000,
19    "episodeNo": 756,
20    "seasonNo": 1,
21    "seasonId": 8157,
22    "showId": 2966,
23    "thumbnail": "https://img.hotstar.com/image/upload/f_auto,t_hcdl/sources/r1/cms/prod/9608/1
24    "startDate": 1682296200,
25    "endDate": 4102424940,
26    "premium": false,
27    "images": [
28        {
29            "url": "https://img.hotstar.com/image/upload/f_auto/sources/r1/cms/prod/9608/150960
30            "transformation": "hcdl",
31            "type": "HORIZONTAL"
32        },
33        {
34            "url": "https://secure-media.hotstar.com/r1/thumbs/PCTV/09/1700008109/PCTV-17000081
35            "transformation": "v1",
36            "type": "VERTICAL"
37        }
38    ],
39    "sourceImages": [
40        {
41            "url": "https://img1.hotstarext.com/image/upload/f_auto/sources/r1/cms/prod/9608/15
42            "type": "HORIZONTAL"
43        }
44    ],
45    "playUri": "https://api.hotstar.com/ph/v1/play?partnerId=airtel&contentId=1700008109",
46    "deepLinkUrl": "hotstar://1700008109",
47    "deepLinkUrlForLivingRoom": "hotstar://content/1700008109",
48    "vip": false,
49    "searchKeywords": [
50        "Unknown"
51    ],
52    "langObjs": [
53        {
54            "hide": false,
55            "id": 4,
56            "name": "Marathi",
57            "iso3code": "mar",
58            "displayName": "मराठी"
59        }
60    ],
61    "assetStatus": "PUBLISHED",
62    "updateDate": 1682382014,
63    "paid": false,
64    "parentalRating": 2,
65    "parentalRatingName": "U/A 13+",
```

```

66         "locators": [
67             {
68                 "url": "https://hotstar.com/1700008109/watch",
69                 "platform": "web"
70             },
71             {
72                 "url": "hotstar://1700008109/watch",
73                 "platform": "ios"
74             },
75             {
76                 "url": "hotstar://1700008109/watch",
77                 "platform": "android"
78             },
79             {
80                 "url": "hotstar://1700008109/watch",
81                 "platform": "appletv"
82             },
83             {
84                 "url": "hotstar://1700008109/watch",
85                 "platform": "androidtv"
86             },
87             {
88                 "url": "https://livingroom.hotstar.com/1700008109/watch",
89                 "platform": "weblr"
90             }
91         ]
92     }
93 ],
94 "totalResults": 354945,
95 "offset": 5000,
96 "size": 1,
97 "nextOffsetURL": "https://cmscontentpartner.hotstar-labs.com/episode/search?offset=5001&size=1&order=desc",
98 "totalPageResults": 1,
99 "totalPages": 354945
100 }
101 },
102 "statusCode": "OK",
103 "statusCodeValue": 200
104 }

```

## Movies

The following API is used to fetch movies

### Request Parameters

Item	Value
HTTP Method	GET
Request Path	/movie/search
Query Parameters	
fromStartDate	Date to start filtering from
toStartDate	Date till when the filtering should stop
fromUpdateDate	Date to start filtering updatedRecord date

<code>toUpdateDate</code>	Date till when filtering updatedRecord date stop
<code>orderBy</code>	The parameter to use for ordering the response, possible value (startDate / contentId)
<code>order</code>	The sort order to be used
<code>offset</code>	Pagination offset
<code>size</code>	Total number of items per page
<code>premium</code>	Whether or not premium content should be fetched
Headers	
<code>x-country-code</code>	Country code
<code>x-platform-code</code>	Code of the platform
<code>x-partner-name</code>	Name of the partner calling this API
<code>x-region-code</code>	
<code>x-client-code</code>	

### Sample Request

```

1  curl 'http://{base_url}/movie/search?fromStartDate=0&toStartDate=1528991761&orderBy=startDate&order=desc&offset=1
2    -H 'x-country-code: in'
3    -H 'x-platform-code: PCTV' \
4    -H 'x-partner-name: <partner name>' \
5    -H 'x-region-code: DL' \
6    -H 'x-client-code: pt'
```

### Sample Response

**Note:** All the response fields are mandatory except below

1. Images with Type vertical
2. searchKeywords
3. trailers
4. actors
5. anchors
6. directors
7. producers
8. playUri (This is functional only for JIO partner)

### Response Schema

```

1  {
2      "body": {
3          "results": {
4              "items": [
5                  {
6                      "id": 35331,
7                      "contentId": "1260128510",
8                      "title": "Dug Days: Carl's Date",
9                      "description": "When Carl agrees to go on a date, Dug offers tried-and-true tips for making
```

```
10     "genre": [
11         "Animation",
12         "Family",
13         "Romance",
14         "Comedy",
15         "Buddy"
16     ],
17     "lang": [
18         "English",
19         "Arabic (Modern Standard)",
20         "French (Parisian)",
21         "Spanish Castilian (European)",
22         "German",
23         "Italian",
24         "Portuguese (European)",
25         "Arabic (Classical Egyptian)"
26     ],
27     "contentType": "MOVIE",
28     "year": 2023,
29     "thumbnail": "https://img.hotstar.com/image/upload/f_auto,t_hcdl/sources/r1/cms/prod/4584/1",
30     "portraitThumbnail": "https://img.hotstar.com/image/upload/f_auto,t_vl/sources/r1/cms/prod/",
31     "startDate": 32535072000,
32     "endDate": 32535100799,
33     "duration": 574,
34     "premium": true,
35     "images": [
36         {
37             "url": "https://img.hotstar.com/image/upload/f_auto/sources/r1/cms/prod/4584/145458",
38             "transformation": "hcdl",
39             "type": "HORIZONTAL"
40         },
41         {
42             "url": "https://img.hotstar.com/image/upload/f_auto,t_vl/sources/r1/cms/prod/4592/1",
43             "transformation": "vl",
44             "type": "VERTICAL"
45         }
46     ],
47     "sourceImages": [
48         {
49             "url": "https://img1.hotstarext.com/image/upload/f_auto/sources/r1/cms/prod/4592/14",
50             "type": "VERTICAL"
51         },
52         {
53             "url": "https://img1.hotstarext.com/image/upload/f_auto/sources/r1/cms/prod/4584/14",
54             "type": "HORIZONTAL"
55         }
56     ],
57     "deepLinkUrl": "hotstar://1260128510",
58     "deepLinkUrlForLivingRoom": "hotstar://content/1260128510",
59     "vip": true,
60     "actors": [
61         "Ed Asner",
62         "Bob Peterson"
63     ],
64     "directors": [
65         "Bob Peterson"
66     ],
67     "searchKeywords": [
```

```
68     "Dug Days: Carl's Date",
69     "English",
70     "Arabic (Modern Standard)",
71     "Hebrew",
72     "Arabic (Classical Egyptian)",
73     "French (Parisian)",
74     "German",
75     "Italian",
76     "Spanish Castilian (European)",
77     "Portuguese (European)",
78     "Animation",
79     "Family",
80     "Romance",
81     "Comedy",
82     "Buddy",
83     "Ed Asner",
84     "Bob Peterson",
85     "Bob Peterson",
86     "Disney+",
87     "Disney",
88     "1260128510"
89 ],
90 "langObjs": [
91 {
92     "hide": false,
93     "id": 3,
94     "name": "English",
95     "iso3code": "eng",
96     "displayName": "English"
97 },
98 {
99     "hide": false,
100    "id": 968,
101    "name": "Arabic (Modern Standard)",
102    "iso3code": "arb",
103    "displayName": "Arabic (Modern Standard)"
104 },
105 {
106     "hide": false,
107     "id": 971,
108     "name": "French (Parisian)",
109     "iso3code": "fra-FR",
110     "displayName": "French (Parisian)"
111 },
112 {
113     "hide": false,
114     "id": 982,
115     "name": "Spanish Castilian (European)",
116     "iso3code": "spa-ES",
117     "displayName": "Spanish Castilian (European)"
118 },
119 {
120     "hide": false,
121     "id": 973,
122     "name": "German",
123     "iso3code": "deu",
124     "displayName": "German"
125 },
```

```
126      {
127          "hide": false,
128          "id": 974,
129          "name": "Italian",
130          "iso3code": "ita",
131          "displayName": "Italian"
132      },
133      {
134          "hide": false,
135          "id": 983,
136          "name": "Portuguese (European)",
137          "iso3code": "por-PT",
138          "displayName": "Portuguese (European)"
139      },
140      {
141          "hide": false,
142          "id": 970,
143          "name": "Arabic (Classical Egyptian)",
144          "iso3code": "arz",
145          "displayName": "Arabic (Classical Egyptian)"
146      }
147 ],
148     "assetStatus": "UNPUBLISHED",
149     "updateDate": 1676006130,
150     "paid": false,
151     "parentalRating": 2,
152     "parentalRatingName": "PG",
153     "locators": [
154         {
155             "url": "https://apps.disneyplus.com/1260128510/watch",
156             "platform": "web"
157         },
158         {
159             "url": "disneyplus://1260128510/watch",
160             "platform": "ios"
161         },
162         {
163             "url": "disneyplus://1260128510/watch",
164             "platform": "android"
165         },
166         {
167             "url": "disneyplus://1260128510/watch",
168             "platform": "appletv"
169         },
170         {
171             "url": "disneyplus://1260128510/watch",
172             "platform": "androidtv"
173         },
174         {
175             "url": "https://livingroom.apps.disneyplus.com/1260128510/watch",
176             "platform": "weblr"
177         }
178     ]
179 },
180 ],
181     "totalResults": 2270,
182     "offset": 0,
183     "size": 1,
```

```

184         "nextOffsetURL": "https://cmscontentpartner.hotstar-labs.com/movie/search?offset=1&size=1&orderBy=s
185         "totalPageResults": 1,
186         "totalPages": 2270
187     }
188 },
189 "statusCode": "OK",
190 "statusCodeValue": 200
191 }

```

## Sports VoD and Live Match

The following API is used to fetch sports VoD and live-match content

### Request Parameters

Item	Value
HTTP Method	GET
Request Path	/match/search
Query Parameters	
fromStartDate	Date to start filtering from
toStartDate	Date till when the filtering should stop
fromUpdateDate	Date to start filtering updatedRecord date
toUpdateDate	Datetime till when filtering updatedRecord date stop
orderBy	The parameter to use for ordering the response, possible value (startDate / contentId)
order	The sort order to be used
offset	Pagination offset
size	Total number of items per page
live	Whether live sports content should be fetched or VoD  live=false will return all sports VoD  live=true will return all live match clips
premium	Whether or not premium content should be fetched
Headers	
x-country-code	Country code
x-platform-code	Code of the platform
x-partner-name	Name of the partner calling this API
x-region-code	
x-client-code	

## Sample Request

```
1  Curl 'http://{base_url}/match/search?fromStartDate=0&toStartDate=1528991761&orderBy=startDate&order=desc&offset=1
2    -H 'x-country-code: in' \
3    -H 'x-platform-code: PCTV' \
4    -H 'x-partner-name: <partner name>' \
5    -H 'x-region-code: DL' \
6    -H 'x-client-code: pt'
```

## Sample Response

**Note:** All the response fields are mandatory except below

1. searchKeywords
2. playUri (This is functional only for JIO partner)

## Response Schema

```
1  {
2      "body": {
3          "results": {
4              "items": [
5                  {
6                      "id": 1070765,
7                      "contentId": "1540023334",
8                      "title": "Replay | Bigg Boss Malayalam S5 - Simulcast - Day 98",
9                      "description": "Replay | Bigg Boss Malayalam S5 - Simulcast - Day 98",
10                     "lang": [
11                         "Malayalam"
12                     ],
13                     "genre": [
14                         "Football"
15                     ],
16                     "contentType": "MATCH",
17                     "duration": 0,
18                     "startDate": 1688310600,
19                     "endDate": 4102444799,
20                     "sportsSeasonId": 12,
21                     "tournamentId": 9,
22                     "thumbnail": "https://secure-media.hotstar.com/r1/thumbs/PCTV/34/1540023334/PCTV-1540023334-
23                     "siMatchId": "708211",
24                     "gameName": "FOOTBALL",
25                     "premium": true,
26                     "deepLinkUrl": "hotstar://1540023334",
27                     "vip": false,
28                     "live": false,
29                     "assetStatus": "UNPUBLISHED",
30                     "updateDate": 1682666079,
31                     "paid": false,
32                     "searchKeywords": [
33                         "Replay | Bigg Boss Malayalam S5 - Simulcast - Day 98",
34                         "FOOTBALL",
35                         "1540023334",
36                         "Serie A 2014",
37                         "replays",
38                         "Live",
39                         "Live Match",
40                         "FOOTBALL",
41                         "Live Bigg Boss Malayalam S5 Simulcast",
```

```

42             "Bigg Boss Malayalam S5 Simulcast Live",
43             "Unknown"
44         ],
45         "sourceImages": [
46             {
47                 "url": "https://secure-media1.hotstarext.com/r1/thumbs/ANDROID/34/1540023334/ANDROID",
48                 "transformation": "hl_tab_xxhdpi",
49                 "type": "HORIZONTAL"
50             }
51         ],
52         "locators": [
53             {
54                 "url": "https://hotstar.com/1540023334/watch",
55                 "platform": "web"
56             },
57             {
58                 "url": "hotstar://1540023334/watch",
59                 "platform": "ios"
60             },
61             {
62                 "url": "hotstar://1540023334/watch",
63                 "platform": "android"
64             },
65             {
66                 "url": "hotstar://1540023334/watch",
67                 "platform": "appletv"
68             },
69             {
70                 "url": "hotstar://1540023334/watch",
71                 "platform": "androidtv"
72             },
73             {
74                 "url": "https://livingroom.hotstar.com/1540023334/watch",
75                 "platform": "weblr"
76             }
77         ]
78     }
79 ],
80     "totalResults": 120655,
81     "offset": 0,
82     "size": 1,
83     "nextOffsetURL": "https://cmscontentpartner.hotstar-labs.com/match/search?offset=1&size=1&orderBy=st",
84     "totalPageResults": 1,
85     "totalPages": 120655
86   }
87 },
88   "statusCode": "OK",
89   "statusCodeValue": 200
90 }

```

## Channel List API

In the context of Hotstar, "channels" refers to the classification of content according to their providers, eg. Star Plus, Hulu. Each channel represents a specific entity under which relevant content can be grouped.

The following API is used to fetch the channel list. API response contains the channel's `id`, `name`, `contentId`, and `enabledPlatforms`.

`contentId` is unique across countries and `enabledPlatforms` contains the list of platforms on which the channel is displayed.

## Request Parameters

Item	Value
HTTP Method	GET
Request Path	/channel/list
Query Parameters	
<code>fromUpdateDate</code>	If provided, only includes results that were updated after the provided timestamp (epoch in ms)
<code>toUpdateDate</code>	If provided, only includes results that were updated before the provided timestamp (epoch in ms)
<code>orderBy</code>	The parameter to use for ordering the response, possible value (esUpdateDate)
<code>order</code>	The sort order to be used
<code>offset</code>	Pagination offset
<code>size</code>	Total number of items per page
Headers	
<code>x-country-code</code>	Country code
<code>x-platform-code</code>	Code of the platform
<code>x-partner-name</code>	Name of the partner calling this API
<code>x-region-code</code>	
<code>x-client-code</code>	

## Sample Request

```
1 curl --location --request GET 'https://[base_url]/channel/list?toUpdateDate=1685702098&fromUpdateDate=1685702080&
2 --header 'x-country-code: in' \
3 --header 'x-partner-name: 91585742429' \
4 --header 'x-client-code: pt' \
5 --header 'x-region-code: DL' \
6 --header 'x-platform-code: PCTV'
```

## Sample Response

### Note:

1. All the fields will be present in the API response.
2. Channels which are not searchable or have empty `enabledPlatforms` list will not be present in the API response.

## Response Schema

```
1 {
2     "body": {
3         "results": {
4             "items": [
5                 {
6                     "id": 3,
7                     "contentId": 821,
8                     "name": "StarPlus",
```

```

9         "enabledPlatforms": [
10             "FREEPREMIUM",
11             "ROKU",
12             "OTTSTB",
13             "PLAYSTATION",
14             "FREEPREMIUM_DAI",
15             "FIRETV",
16             "HLS_INTERNATIONAL",
17             "DOWNLOAD",
18             "IPAD",
19             "IOS",
20             "TABLET",
21             "CHROMECAST",
22             "SCOPETV",
23             "PCTV",
24             "JIO",
25             "ASHA",
26             "XBOX",
27             "CONNECTED",
28             "PLAYREADY",
29             "ANDROID",
30             "APPLETV"
31         ]
32     }
33 ],
34     "totalResults": 40,
35     "offset": 0,
36     "size": 1000,
37     "totalPageResults": 40,
38     "totalPages": 1
39   }
40 },
41   "statusCodeValue": 200,
42   "statusCode": "OK"
43 }
```

## Changelog

New changes: (23rd August 2023)

Added a new field `trailerDeeplinks` which will contain the deep links for just trailers (not bonuses) associated with a content. Done for Episodes and Movies.

New changes: (1st September 2021)

Post the launch of the new subscription plans(SMP), all paid content will have both "vip" and "premium" flag marked as true and all free content will have both "vip" and "premium" flag marked as false.

New Changes: (16th August 2020)

Updated the document to include the non Mandatory response fields

New Changes: (7th August 2020)

Updated the Request for EPISODE/MOVIE/MATCH Search to take below parameters and returned the details of contents that got updated in this interval.

fromUpdateDate	Date to start filtering updatedRecord date
toUpdateDate	Date till when filtering updatedRecord date stop

New changes: (28 July 2020)

1. Added an enum assetStatus which corresponds to the status of the movie/match/episode and can correspond to (PUBLISHED or UNPUBLISHED)

```
1 "assetStatus": "UNPUBLISHED",
```

New changes : (1st September 2019)

1. Added playUri and deepLinkUrl to playable assets response.

```
1 "playUri": "https://api.hotstar.com/ph/v1/play?partnerId=<partner name>&contentId=2001450460",
2 "deepLinkUrl": "hotstar://2001450460"
```

New changes : (3rd Jan 2019)

1. Add season search API using show ID

New changes: (20th Nov 2018)

1. Added list of images in the search response of show, movie, and episode which contain the image URL corresponding to the transformation and image type ( horizontal and vertical). A horizontal image URL of dimension 640 \* 360 is added with transformation `hcd1` and a vertical image URL of dimension 390 \* 518 is added with transformation `v1` in the list. i.e.,

```
1 "images": [
2   {
3     "url": "https://img.hotstar.com/image/upload/f_auto,t_hcd1/sources/r1/cms/prod/4617/124617-h",
4     "transformation": "hcd1",
5     "type": "HORIZONTAL"
6   },
7   {
8     "url": "https://secure-media.hotstar.com/r1/thumbs/PCTV/78/1100012278/PCTV-1100012278-v1.jpg",
9     "transformation": "v1",
10    "type": "VERTICAL"
11  }
12]
```

New changes : (14th Nov 2018)

1. Added rate limit of 1 request per second to APIs based on the partner, exceeding which will result in 429 HTTP status code in response.

New changes will be in effect by (19 Oct 2018)

1. Premium filter added and also returned in response

2. Guideline added for the premium flag: The premium flag can be used for filtering. But for some partner, we have some special handling where premium content is marked as non-premium based on some conditions. This flag is not suitable for their use

3. startDate and endDate added in the response to all APIs

New changes: (27 Sept 2018)

1. Request rate is reduced to 1 TPS and max result limit increased to 1000.
2. Offset + Size should not exceed 10000 limit check added
3. fromStartDate & toStartDate filters non-mandatory filters added in all API's
4. Circuit break logic added

New changes: (2 March 2023)

1. added an additional option (contentId) in orderBy params.
2. for show/search API in the orderBy field startDate is not recommended instead of startDate please use contentId.

New changes: (3rd May 2023)

1. Added a new field - locators - to specify platform-specific deeplink urls. (Spec: [Deeplink format changes for different Platforms](#))
2. This field is available in movie, episode, show and match entities.

```
1 "locators": [  
2     {  
3         "url": "https://hotstar.com/1540023334/watch",  
4         "platform": "web"  
5     },  
6     {  
7         "url": "hotstar://1540023334/watch",  
8         "platform": "ios"  
9     },  
10    {  
11        "url": "hotstar://1540023334/watch",  
12        "platform": "android"  
13    },  
14    {  
15        "url": "hotstar://1540023334/watch",  
16        "platform": "appletv"  
17    },  
18    {  
19        "url": "hotstar://1540023334/watch",  
20        "platform": "androidtv"  
21    },  
22    {  
23        "url": "https://livingroom.hotstar.com/1540023334/watch",  
24        "platform": "weblr"  
25    },  
26]  
27}
```

New changes: (25th July 2023)

1. Added a new field - channelObj . It has channel id, name, and its contentId. contentId is unique across countries.
2. This field is available in the show entity.

```
1 "channelObject": {  
2     "id": 22,  
3     "name": "Select Picks",  
4     "contentId": 10403
```

New changes: (25th July 2023)

1. Added a channel list API - `/channel/list`. It returns the list of channels. It contains channel `id`, `name`, `contentId`, and `enabledPlatforms`. `contentId` is unique across countries. `enabledPlatforms` contains the list of platforms on which channel is enabled.

```
1  {
2      "id": 3,
3      "contentId": 821,
4      "name": "StarPlus",
5      "enabledPlatforms": [
6          "FREEMIUM",
7          "ROKU",
8          "OTTSTB",
9          "PLAYSTATION",
10         "FREEMIUM_DAI",
11         "FIRETV",
12         "HLS_INTERNATIONAL",
13         "DOWNLOAD",
14         "IPAD",
15         "IOS",
16         "TABLET",
17         "CHROMECAST",
18         "SCOPETV",
19         "PCTV",
20         "JIO",
21         "ASHA",
22         "XBOX",
23         "CONNECTED",
24         "PLAYREADY",
25         "ANDROID",
26         "APPLETV"
27     ]
28 }
```