



Alpha Networks' Generic VOD and EPG Driver for CONTAINER

September 2019

CONNECTING **CONTENT**
WITH **SUBSCRIBERS**

Table of Contents

1	Summary	3
2	Process	4
2.1	Ingestion modes	4
3	Metadata files.....	5
3.1	File naming convention	5
3.2	Media and trailer location	5
3.3	Media and trailer technology and DRM	6
3.3.1	<i>Technology.....</i>	6
3.3.2	<i>DRM.....</i>	7
3.4	assets.json	8
3.4.1	<i>Example file.....</i>	8
3.4.2	<i>JSON schema.....</i>	10
3.4.3	<i>Supported asset metadata.....</i>	16
3.4.4	<i>Supported media metadata</i>	17
3.4.5	<i>Notes.....</i>	17
3.5	categories.json	18
3.6	series.json	23
3.6.1	<i>Example file.....</i>	23
3.6.2	<i>JSON schema.....</i>	25
3.7	broadcasts.json	29
3.7.1	<i>Example file.....</i>	29
3.7.2	<i>JSON schema.....</i>	30



1 Summary

Today on the market there is multiple customers with custom demands for video ingest metadata and workflow.

For this purpose, we have a generic "product" driver whose format is strongly inspired from tucanoRED's canonical metadata format, with a few simplifications aiming at making implementation simpler for the providers.



2 Process

The "pseudo-canonical" data structures are received as JSON files. They are converted to a normal canonical and sent to the CMS.

The files are received in a single input folder **per ingest**, whose name must be the timestamp at which the process of delivery of the files started (something like `/input/1567497600/`). Such naming would allow the driver to be down for a while, and then ingest the folders in the right order. The structure of files inside this "per-ingest folder" depends on the ingestion mode - see below.

In both modes however, the files are moved to another one for archiving purpose after ingestion (e.g. to a folder like `/ingested/1567497600/`).

Also in both modes, images are provided in a subdirectory of the main ingest folder (`/input/1567497600/images/`), and removed after having been copied to the storage used by the CMS.

2.1 Ingestion modes

There are two ingestion modes: incremental and integral.

Incremental mode consists in receiving updates, "deltas", which are applied on top of the current data.

In incremental mode, metadata is put in two subdirectories: `add/` and `remove/`.

Integral mode consists in receiving the entirety of the data at every ingest. This is like asking the provider to export the contents of a relational database.

In integral mode, files are put directly in the main ingest folder.

Whether a provider uses integral or incremental mode is configured at the EPG provider or VOD source catalog level.



3 Metadata files

3.1 File naming convention

For **EPG** ingests, the following files are received:

- assets.json
- broadcasts.json

For **VOD** ingests, the following files are received:

- assets.json
- categories.json
- series.json (optional)

For VOD ingests, **videos** can optionally be provided too. If they are provided, they must be put in a separate subdirectory whose name is the asset's id (`/input/1567497600/asset1/`), and the video files themselves must start with the asset id, with an optional suffix, which can be (for instance) the bitrate, if multiple MBTS files are provided (eg `/input/1567497600/asset1/asset1_1600.ts`).

For VOD ingests, **trailers** can also be provided. The difference with "regular" videos is that trailers can be watched even by non-subscribed users. The rules for trailers video files are the same as for the main video content, except the subdirectory in which they must be uploaded is different: `/input/1567497600/asset1_trailers/`.

3.2 Media and trailer location

Both media and trailers have a "location" property which indicates "where" the corresponding video is.

As a reminder, an asset "media", in tucanoRED's nomenclature, is a single published multimedia file. A given asset can have multiple media, for instance because of the presence of burned-in subtitles in the video.

Note that if tucanoRED manages video content (transcoding, DRM protection, publication on a CDN), each input media will be duplicated, as there is a published media per streaming protocol. In other words: for a given MP4 file provided as input, the output will be (for instance) one "HLS media", one "DASH media", and one "HSS media". If tucanoRED does not manage video workflow, and instead the provider does, then the metadata must provide (at least) one media per supported streaming protocol, and in that case, the location is the URL of the corresponding manifest or playlist file.



The following table lists the possible use-cases of the driver, in terms of media management:

Use-case	Media location format	Example	Description
video content is not managed by tucanoRED	Absolute HTTP URL, generally of the manifest / playlist file	http://cdn.provider.com/asset.m3u8	The provider has its own video distribution infrastructure. The location provided is the origin URL, sent to the end-user as-is or processed (e.g. tokenized) by another driver at asset playback time.
MBTR files are provided for each media	An absolute or relative path on the storage used, with a wildcard to match all video files corresponding to a media	/media/VOD/asset1/asset1_*.ts	For each media, multiple video files are provided - one per bitrate. In this case, there must be a single entry in the list of media for all of the MBTR files; therefore, a wildcard must be used to match them all.
A single video file is provided for each media	An absolute or relative path on the storage used, pointing to the single video file	/media/VOD/asset1/asset1_vostfr.ts	For each media, a single video file is provided. In this case, there is simply a single path toward the video file. Remote URLs can be used (toward an FTP server, S3 bucket, etc.), with the exception of HTTP URLs , which are reserved to the first use-case.

Note that we do not support video content being cut in multiple parts (or "chapters"). No matter which use-case above is used, each video file must contain the entirety of the asset.

3.3 Media and trailer technology and DRM

Both media and trailers have two **optional** properties: "technology" and "drm".

3.3.1 Technology

The `technology` property indicates "what" the corresponding video is:

- If tucanoRED manages video content, the technology provided must be the multimedia container format provided ("MP4", "TS" for MPEG2-TS, etc.).



- If tucanoRED does not manage video content, then the media provided is a "published" origin URL for the content. In this case, the technology is the streaming protocol available at that URL (e.g. "HLS", "HSS", "DASH").

If this property is absent, the technology is inferred from the extension of the provided files:

Provided file's extension	Inferred technology
.mp4	mp4
.ts	ts (MPEG-2 TS)
.m3u8	HLS
.mpd	DASH
.ism	HSS
any other	not inferred yet; please use the technology field.

For convenience, **this property is case-insensitive.**

3.3.2 DRM

This property is **only** used in the case tucanoRED does not manage video content, which means the media provided is a published origin URL for the content.

In this case, the URL is that of a manifest, for an asset which might be DRM-protected. The DRM protocol used for this asset can then be provided in the `drm` field; the ID of the DRM resource must also be provided in this case, in the media metadata `drm_resource_id` (see section "Supported media metadata" below).

If no DRM resource ID is provided and no `drm` property either, the asset is considered not DRM-protected.

If no DRM resource ID is provided and a `drm` property is provided, the DRM resource id is considered to be the **asset ID**. **Relying on this behavior is not recommended.**

If a DRM resource ID is provided and no `drm` field is provided, the protocol is inferred from the technology, according to the following mapping:

Media technology	Inferred DRM
DASH	FairPlay



Media technology	Inferred DRM
HLS	Widevine
HSS	PlayReady

3.4 assets.json

This file contains the assets, in the tucanoRED sense of the term.

3.4.1 Example file

```
{
  "assets": [
    {
      "id": "e5dd7acd-c432-45d6-8d3c-6508bbb61a22", // mandatory
string (idAssetProvider)
      "title": { // mandatory
        "eng": "lorem ipsum ..."
      },
      "subtitle": { // optional (AD_SUBTITLE)
        "eng": "lorem ipsum ..."
      },
      "synopsis": { // mandatory
        "eng": "lorem ipsum ..."
      },
      "morality_level": 18, // mandatory
      "adult": false, // mandatory
      "duration": 3600000, // mandatory, in ms
      "series_id": "523c4c5c-68c5-4e00-a98b-754b9e9f4b71", //
Optional. Empty string is also OK, if not a series
      "season_id": "759bab6c-1168-4ed5-ab91-b4ec41501daa", //
mandatory if series_id provided. Empty string OK for non-series
content.
      "season_episode_number": 7, // mandatory if series_id
provided. Episode number in the season
      "genres": [ // mandatory (can be empty if no genres)
        "horror",
        "love",
        "college"
      ],
      "available_from": 1566047956123, // mandatory. UTC timestamp,
in ms
      "available_to": 1666047956456, // mandatory. UTC timestamp, in
ms
      "metadata": {}, // mandatory (but can be empty). Arbitrary
key/value pairs. Keys are strings. Values can be text, numbers,
booleans, or even structures such as lists of maps.
      "images": [ // mandatory (can be empty).
        {
          "id": "7b4ef709-51d8-438c-b2b1-4123dc5c6754",
```




```

        "type": "poster", // mandatory. Types to be agreed
beforehand (usual ones are "poster" and "background")
        "language": "eng", // optional, defaults to null. ISO 639-
3 language code
        // Mandatory
        // the image location. Valid values are:
        // - Absolute remote URLs:
        // -
http(s)://(user:password@)server.com(:80)/a/b/c/d.png
        // - ftp(s)://(user:password@)my-ftp-
server(:21)/vod/posters.123456.svg
        // - sftp://(user:password@)my-ftp-
server(:22)/vod/posters.123456.svg
        // - s3://keyId:keySecret@aws-region/bucket-
name/test.png
        // - Relative local URLs:
        // - images/123.jpeg
        "location":
"https://upload.wikimedia.org/wikipedia/commons/thumb/d/d9/Earth_by_th
e_EPIC_Team_on_21_April_2018.png/280px-
Earth_by_the_EPIC_Team_on_21_April_2018.png"
    }
],
    "medias": [ // Mandatory; list of video files.
    {
        "id": "37409ea7-8a21-4899-9b18-fce68cc22f71", // Mandatory
string
        "type": "TRAILER", // Mandatory. "TRAILER" for trailers,
"MEDIA" for the actual movie / episode
        "title": "lorem ipsum...", // Mandatory string. If you
don't have a title, you can repeat the id.

        // Mandatory
        // the media location. Valid values are:
        // - Absolute remote URLs:
        // -
http(s)://(user:password@)server.com(:80)/a/b/c/d.ts
        // - ftp(s)://(user:password@)my-ftp-
server(:21)/vod/trailers.123456.mp4
        // - sftp://(user:password@)my-ftp-
server(:22)/vod/movies.123456.mkv
        // - s3://keyId:keySecret@aws-region/bucket-name/test.ts
        // - Relative local URLs:
        // - medias/123.raw
        //
        // In case multiple files are to be considered (for
instance multiple resolutions), you must use globs such as:
/vod/123456_*.mp4 to match files /vod/123456_240.mp4,
/vod/123456_360.mp4, /vod/123456_480.mp4, and /vod/123456_720.mp4
        "location": "https://server.com/movie.mp4",
        "technology": "mp4",
        "audio_languages": [ "eng"], // Mandatory; list of audio
languages included (ISO-639-3 code)
        "subtitles_languages": [ "fra" ], // Mandatory; list of
caption languages included (ISO-639-3 code)
    }
    ]
}

```



```

        "metadata": {} // Mandatory, but can be empty. Arbitrary
        key/value pairs. Keys are strings. Values can be text, numbers,
        booleans, or even structures such as lists of maps
    }
}
]
}
]
}

```

3.4.2 JSON schema

```

{
  "definitions": {},
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "http://tucano.io/red/ingest/canonical/assets.json",
  "type": "object",
  "title": "assets.json file for tucano's standard ingest",
  "required": [
    "assets"
  ],
  "properties": {
    "assets": {
      "$id": "#/properties/assets",
      "type": "array",
      "items": {
        "$id": "#/properties/assets/items",
        "type": "object",
        "required": [
          "adult",
          "available_from",
          "available_to",
          "duration",
          "genres",
          "id",
          "images",
          "medias",
          "metadata",
          "morality_level",
          "synopsis",
          "title"
        ],
        "dependencies": {
          "series_id": {
            "required": [
              "season_id",
              "season_episode_number"
            ]
          }
        }
      },
      "properties": {
        "adult": {
          "$id": "#/properties/assets/items/properties/adult",
          "type": "boolean",
          "description": "Whether this asset is adult content",
          "examples": [
            false
          ]
        }
      }
    }
  },
}

```



```

    "available_from": {
      "$id": "#/properties/assets/items/properties/available_from",
      "type": "integer",
      "description": "UTC epoch timestamp at which the asset
becomes available to end-users, in milliseconds.",
      "examples": [
        1566047956123
      ]
    },
    "available_to": {
      "$id": "#/properties/assets/items/properties/available_to",
      "type": "integer",
      "description": "UTC epoch timestamp at which the asset stops
being available to end-users, in milliseconds.",
      "examples": [
        1666047956456
      ]
    },
    "duration": {
      "$id": "#/properties/assets/items/properties/duration",
      "type": "integer",
      "description": "The duration of the asset, in seconds",
      "examples": [
        3600000
      ]
    },
    "genres": {
      "$id": "#/properties/assets/items/properties/genres",
      "type": "array",
      "description": "List of genres of the asset",
      "items": {
        "$id": "#/properties/assets/items/properties/genres/items",
        "type": "string",
        "examples": [
          "horror",
          "love",
          "college"
        ]
      }
    },
    "id": {
      "$id": "#/properties/assets/items/properties/id",
      "type": "string",
      "description": "Unique id of the asset",
      "examples": [
        "e5dd7acd-c432-45d6-8d3c-6508bbb61a22"
      ]
    },
    "images": {
      "$id": "#/properties/assets/items/properties/images",
      "type": "array",
      "description": "List of images linked to this asset",
      "items": {
        "$id": "#/properties/assets/items/properties/images/items",
        "type": "object",
        "required": [
          "location",
          "type",
          "id"
        ]
      }
    },
  ],

```



```

        "properties": {
            "id": {
                "$id":
"#/properties/assets/items/properties/images/items/properties/id",
                "type": "string",
                "description": "ID of the image",
                "examples": [
                    "7b4ef709-51d8-438c-b2b1-4123dc5c6754"
                ]
            },
            "language": {
                "$id":
"#/properties/assets/items/properties/images/items/properties/language",
                "type": "string",
                "description": "Language of the image, as an ISO 639-3
code.",
                "default": null,
                "examples": [
                    "eng"
                ],
                "pattern": "^[a-z]{3}$"
            },
            "location": {
                "$id":
"#/properties/assets/items/properties/images/items/properties/location",
                "type": "string",
                "description": "Image location. Accepted values are
absolute remote URLs (http, ftp, sftp, Amazon S3...), or relative local
paths.",
                "examples": [

                    "https://upload.wikimedia.org/wikipedia/commons/thumb/d/d9/Earth_by_the_E
PIC_Team_on_21_April_2018.png/280px-
Earth_by_the_EPIC_Team_on_21_April_2018.png",
                    "images/123.jpeg",
                    "ftp(s)://(user:password@)my-ftp-
server(:21)/vod/posters.123456.svg",
                    "sftp://(user:password@)my-ftp-
server(:22)/vod/posters.123456.svg",
                    "s3://keyId:keySecret@aws-region/bucket-
name/test.png"
                ]
            },
            "type": {
                "$id":
"#/properties/assets/items/properties/images/items/properties/type",
                "type": "string",
                "description": "Image type; an arbitrary string agreed
to beforehand. Usually used ones are 'poster' and 'background',
respectively for a portrait and landscape format.",
                "default": "",
                "examples": [
                    "poster",
                    "background"
                ]
            }
        }
    },
    "medias": {

```



```

"$id": "#/properties/assets/items/properties/medias",
"type": "array",
"description": "List of medias (video files) of the asset",
"items": {
  "$id": "#/properties/assets/items/properties/medias/items",
  "type": "object",
  "required": [
    "audio_languages",
    "id",
    "location",
    "metadata",
    "subtitles_languages",
    "title",
    "type"
  ],
  "properties": {
    "audio_languages": {
      "$id":
"#/properties/assets/items/properties/medias/items/properties/audio_langu
ages",
      "type": "array",
      "description": "List of audio languages available in
this video, as a list of ISO 639-3 language codes.",
      "items": {
        "$id":
"#/properties/assets/items/properties/medias/items/properties/audio_langu
ages/items",
        "type": "string",
        "default": "",
        "examples": [
          "eng",
          "fra"
        ],
        "pattern": "^[a-z]{3}$"
      }
    },
    "id": {
      "$id":
"#/properties/assets/items/properties/medias/items/properties/id",
      "type": "string",
      "description": "Unique ID of the media; if you don't
have any, you can use the asset id with a suffix",
      "examples": [
        "37409ea7-8a21-4899-9b18-fce68cc22f71"
      ]
    },
    "location": {
      "$id":
"#/properties/assets/items/properties/medias/items/properties/location",
      "type": "string",
      "description": "Media location. Accepted values are
absolute remote URLs (http, ftp, sftp, Amazon S3...), or relative local
paths. Remote URLs are not downloaded, but instead considered to be
origin URLs for content ready to play (no workflow management by tucano).
Wildcards are supported, and mandatory for MTBR files.",
      "examples": [
        "https://server.com/movie.mp4",
        "http(s)://(user:password@)server.com(:80)/a/b/c/d.ts",

```



```

        "ftp(s)://(user:password@)my-ftp-
server(:21)/vod/trailers.123456.mp4",
        "sftp://(user:password@)my-ftp-
server(:22)/vod/movies.123456.mkv",
        "s3://keyId:keySecret@aws-region/bucket-
name/test.ts",
        "medias/asset123/123.raw",
        "medias/asset1/asset1_*.ts"
    ]
},
"technology": {
    "$id":
"#/properties/assets/items/properties/medias/items/properties/technology"
,
    "type": "string",
    "description": "Case-insensitive. 'Technology' of the
media, which can be an indication of the video container format ('mp4',
'ts', etc.) if video content is managed by tucanoRED, or the streaming
protocol ('hls', 'dash', or 'hss') otherwise.",
    "examples": [
        "mp4",
        "ts",
        "hls",
        "hss",
        "dash"
    ]
},
"drm": {
    "$id":
"#/properties/assets/items/properties/medias/items/properties/drm",
    "type": "string",
    "description": "Case-insensitive. DRM protocol used to
protect the media, if video content is not managed by tucanoRED.",
    "examples": [
        "fairplay",
        "widevine",
        "playready"
    ]
},
"metadata": {
    "$id":
"#/properties/assets/items/properties/medias/items/properties/metadata",
    "type": "object",
    "description": "Arbitrary key/value pairs. Keys are
strings; values can be text, numbers, booleans, or more complex data
structures like lists of maps. Although arbitrary keys are accepted, some
are listed in an accompanying document, in an effort to simplify and
standardize the most common use-cases."
},
"subtitles_languages": {
    "$id":
"#/properties/assets/items/properties/medias/items/properties/subtitles_1
anguages",
    "type": "array",
    "description": "The subtitles languages available in
that media, as a list of ISO 639-3 language codes",
    "items": {
        "$id":
"#/properties/assets/items/properties/medias/items/properties/subtitles_1
anguages/items",

```



```

        "type": "string",
        "examples": [
            "fra"
        ],
        "pattern": "^[a-z]{3}$"
    },
    {
        "title": {
            "$id":
"#/properties/assets/items/properties/medias/items/properties/title",
            "type": "string",
            "description": "The media's title. If you don't have a
title, you can repeat the ID.",
            "examples": [
                "lorem ipsum..."
            ]
        },
        "type": {
            "$id":
"#/properties/assets/items/properties/medias/items/properties/type",
            "type": "string",
            "description": "Type of media: TRAILER, for trailers,
or MEDIA, for the actual movie or show.",
            "examples": [
                "TRAILER",
                "MEDIA"
            ],
            "pattern": "^(MEDIA|TRAILER)$"
        }
    }
},
"metadata": {
    "$id": "#/properties/assets/items/properties/metadata",
    "type": "object",
    "description": "Arbitrary key/value pairs. Keys are strings;
values can be text, numbers, booleans, or more complex data structures
like lists of maps. Although arbitrary keys are accepted, some are listed
in an accompanying document, in an effort to simplify and standardize the
most common use-cases."
},
"morality_level": {
    "$id": "#/properties/assets/items/properties/morality_level",
    "type": "integer",
    "description": "Morality level of the asset (aka parental
rating), as an integer indicating the minimum age recommended for
viewers",
    "default": 0,
    "examples": [
        18,
        16,
        12
    ]
},
"season_episode_number": {
    "$id":
"#/properties/assets/items/properties/season_episode_number",
    "type": "integer",
    "description": "Mandatory if series_id provided. For series'
episodes, number of the episode within the season",

```



```

        "examples": [
          7
        ]
      },
      "season_id": {
        "$id": "#/properties/assets/items/properties/season_id",
        "type": "string",
        "description": "Mandatory if series_id provided. Unique id of
the season, as defined in the series.json file.",
        "examples": [
          "759bab6c-1168-4ed5-ab91-b4ec41501daa"
        ]
      },
      "series_id": {
        "$id": "#/properties/assets/items/properties/series_id",
        "type": "string",
        "description": "Unique id of the season; reference to
series.json.",
        "default": "",
        "examples": [
          "523c4c5c-68c5-4e00-a98b-754b9e9f4b71"
        ]
      },
      "subtitle": {
        "$id": "#/properties/assets/items/properties/subtitle",
        "type": "object",
        "description": "Secondary title of the asset. It is a map of
translations. Keys are ISO 639-3 language codes, values are the subtitle
in that language."
      },
      "synopsis": {
        "$id": "#/properties/assets/items/properties/synopsis",
        "type": "object",
        "description": "Synopsis of the asset. It is a map of
translations. Keys are ISO 639-3 language codes, values are the synopsis
of the asset in that language."
      },
      "title": {
        "$id": "#/properties/assets/items/properties/title",
        "type": "object",
        "description": "Title of the asset. It is a map of
translations. Keys are ISO 639-3 language codes, values are the title of
the asset in that language."
      }
    }
  }
}

```

3.4.3 Supported asset metadata

While arbitrary key/value pairs are accepted, some keys are pre-defined for common use-cases, making it easier to standardize most projects.

Keep in mind that only VOD has been specified for now, and that even for EPG, some metadata would be attached to the broadcast, not the asset.



Name in the ingest file	Type	Example	tucanoRED equivalent	Description
country	string	"Spain"	AD_CTRYNATI	
production_year	integer	2007	TS_PRODYEAR	
release_year	integer	2007	TS_RELEDATE	
morality_level_full	string	"R"	AD_MLRAWVAL	
morality_level_type	string	"MPAA"	AD_MLTYPENA	
summary	array of strings	["part 1", "part 2"]	AD_SUMMARY	
georestriction	string	"bel"	AD_GEOREST	Three-letter country code, as defined in ISO 3166 (alpha-3).
scores	dictionnary (set of key/value pairs) with values as numbers	{ "Rotten Tomatoes": 88, "Telerama": 3 }	many AD_SCORxxxx metadata	
budget	integer	2000000	AD_BUDGET	
tags	array of strings	["Blockbuster", "highlight"]	AD_LIKEWOR	
external_ids	dictionnary (set of key/value pairs) with values as strings	{ "IMDB": "tt1038988" }	AD_EXTERIDS	
sport_championship	string	"Ligue 1 Conforama"	AD_SPORTCHM	
sport_event	string	"Nantes / Montpellier"	AD_SPORTEVT	
sport_location	string	"La Beaujoire"	AD_SPORTLOC	
actors	array of strings	["Manuela Velasco"]	AD_LIACTORS	
directors	array of strings	["Jaume Balagueró"]	AD_LIREALIS	
tvod_price	integer	5	AD_TCMPRICE	
external_links	dictionnary (set of key/value pairs) with values as strings	{ "website": " http://alphanetworks.tv " }	AD_EXTLINKS	

3.4.4 Supported media metadata

While arbitrary key/value pairs are accepted, some keys are pre-defined for common use-cases, making it easier to standardize most projects.

Name in the ingest file	Type	Example	tucanoRED equivalent	Description
drm_resource_id	string	48f84734-9248-4009-90d4-ea70cd04bf7	io.tucano.drmResourceId	For assets that have already been provisioned in a DRM system

3.4.5 Notes

Translations

No distinction is made between translated titles (synopsis, subtitles...) and "technical" (or default) titles (synopsis, subtitles...). Instead, the default will be decided by the following priority list:

- platform's language



- first language in the provided list

In most cases, only one language will be provided.

3.5 categories.json

This file contains the categories, in the tucanoRED sense of the term, as a flat list. This allows the integrator to prepare the data to be exported as a relational database, and then simply "dump" the content of the tables.

Example:

```
{
  "categories": [
    {
      "id": "7b4ef709-51d8-438c-b2b1-4123dc5c6756", // Mandatory string
      "name": { // mandatory
        "eng": "lorem ipsum ...."
      },
      "description_translations": { // optional
        "eng": "lorem ipsum ..."
      },
      "hidden": false, // mandatory; whether the category is hidden to
end-users
      "morality_level": 18, // mandatory; morality level of the category.
Users under this age won't be able to see it
      "parental_control": false, // mandatory boolean; whether to require
parental control (pin code) to access this category
      "adult": false, // mandatory boolean; whether this category
contains adult content
      "downloadable": false, // Optional (default is null), meaning "use
the value on the asset".
      "offline": false, // Optional (default is null), meaning "use the
value on the asset".
      "metadata": {}, // mandatory (but can be empty). Arbitrary
key/value pairs. Keys are strings. Values can be text, numbers, booleans,
or even structures such as lists of maps.
      "images": [ // mandatory (can be empty)
        {
          "id": "7b4ef709-51d8-438c-b2b1-4123dc5c6754",
          "type": "poster", // mandatory. Types to be agreed beforehand
(usual ones are "poster" and "background")
          "language": "eng", // optional, defaults to null. ISO 639-3
language code
          // Mandatory
          // the image location. Valid values are:
          // - Absolute remote URLs:
          //   - http(s)://(user:password@)server.com(:80)/a/b/c/d.png
          //   - ftp(s)://(user:password@)my-ftp-
server(:21)/vod/posters.123456.svg
          //   - sftp://(user:password@)my-ftp-
server(:22)/vod/posters.123456.svg
          //   - s3://keyId:keySecret@aws-region/bucket-name/test.png
          // - Relative local URLs:
          //   - images/123.jpeg

```



```

        "location":
"https://upload.wikimedia.org/wikipedia/commons/thumb/d/d9/Earth_by_the_E
PIC_Team_on_21_April_2018.png/280px-
Earth_by_the_EPIC_Team_on_21_April_2018.png"
    }
    ],
    "series_id": "523c4c5c-68c5-4e00-a98b-754b9e9f4b71", // optional
(default: null). Reference to a series defined in series.json.
    "season_id": "759bab6c-1168-4ed5-ab91-b4ec41501daa", // optional
(default: null). Reference to a season defined in series.json. Mandatory
if series_id provided.
    "parent_id": "", // mandatory. Reference to the parent category, as
defined in categories.json. If there is no parent, null or empty.
    "assets": [
        "123c4c5c-68c5-4e00-a98b-754b9e9f4b71"
    ]
    }
    ]
}

```

JSON schema:

```

{
  "definitions": {},
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "http://tucano.io/red/ingest/canonical/categories.json",
  "type": "object",
  "title": "categories.json file for tucano's standard ingest",
  "required": [
    "categories"
  ],
  "properties": {
    "categories": {
      "$id": "#/properties/categories",
      "type": "array",
      "items": {
        "$id": "#/properties/categories/items",
        "type": "object",
        "required": [
          "id",
          "name",
          "hidden",
          "morality_level",
          "parental_control",
          "adult",
          "metadata",
          "images",
          "parent_id"
        ],
        "properties": {
          "id": {
            "$id": "#/properties/categories/items/properties/id",
            "type": "string",
            "description": "Unique ID of the category",
            "examples": [
              "7b4ef709-51d8-438c-b2b1-4123dc5c6756"
            ]
          },
          "name": {

```



```

        "$id": "#/properties/categories/items/properties/name",
        "type": "object",
        "description": "Category name. It is a map of translations.
Keys are ISO 639-3 language codes; values are the category's name in that
language.",
    },
    "description_translations": {
        "$id":
"#/properties/categories/items/properties/description_translations",
        "type": "object",
        "description": "Description of the category. It is a map of
translations. Keys are ISO 639-3 language codes; values are the
category's description in that language."
    },
    "hidden": {
        "$id": "#/properties/categories/items/properties/hidden",
        "type": "boolean",
        "description": "Whether the category is visible to end-
users.",
        "examples": [
            false,
            true
        ]
    },
    "morality_level": {
        "$id":
"#/properties/categories/items/properties/morality_level",
        "type": "integer",
        "description": "Morality level of the category. Users under
this age won't be allowed to see its content. Should be given the highest
value of the assets contained within it.",
        "examples": [
            18
        ]
    },
    "parental_control": {
        "$id":
"#/properties/categories/items/properties/parental_control",
        "type": "boolean",
        "description": "Whether the parental PIN code should always
be requested before opening this category, disregarding morality level
checks",
        "examples": [
            false,
            true
        ]
    },
    "adult": {
        "$id": "#/properties/categories/items/properties/adult",
        "type": "boolean",
        "description": "Whether this category contains adult
content",
        "examples": [
            false,
            true
        ]
    },
    "downloadable": {
        "$id":
"#/properties/categories/items/properties/downloadable",

```



```

        "type": "boolean",
        "title": "The Downloadable Schema",
        "default": null,
        "examples": [
            false
        ]
    },
    "offline": {
        "$id": "#/properties/categories/items/properties/offline",
        "type": "boolean",
        "title": "The Offline Schema",
        "default": null,
        "examples": [
            false
        ]
    },
    "metadata": {
        "$id": "#/properties/categories/items/properties/metadata",
        "type": "object",
        "description": "Arbitrary key/value pairs. Keys are strings;
values can be text, numbers, booleans, or more complex data structures
like lists of maps. Although arbitrary keys are accepted, some are listed
in an accompanying document, in an effort to simplify and standardize the
most common use-cases."
    },
    "images": {
        "$id": "#/properties/categories/items/properties/images",
        "type": "array",
        "description": "List of images linked to this category",
        "items": {
            "$id":
"#/properties/categories/items/properties/images/items",
            "type": "object",
            "title": "The Items Schema",
            "required": [
                "id",
                "type",
                "location"
            ],
            "properties": {
                "id": {
                    "$id":
"#/properties/categories/items/properties/images/items/properties/id",
                    "type": "string",
                    "description": "ID of the image",
                    "examples": [
                        "7b4ef709-51d8-438c-b2b1-4123dc5c6754"
                    ]
                },
                "type": {
                    "$id":
"#/properties/categories/items/properties/images/items/properties/type",
                    "type": "string",
                    "description": "Image type; an arbitrary string agreed
to beforehand. Usually used ones are 'poster' and 'background',
respectively for a portrait and landscape format.",
                    "examples": [
                        "poster",
                        "background"
                    ]
                }
            }
        }
    }

```



```

        },
        "language": {
            "$id":
"#/properties/categories/items/properties/images/items/properties/language",
            "type": "string",
            "description": "Language of the image, as an ISO 639-3
code.",
            "examples": [
                "eng",
                "fra"
            ],
            "pattern": "^[a-z]{3}$"
        },
        "location": {
            "$id":
"#/properties/categories/items/properties/images/items/properties/locatio
n",
            "type": "string",
            "description": "Image location. Accepted values are
absolute remote URLs (http, ftp, sftp, Amazon S3...), or relative local
paths.",
            "examples": [

"https://upload.wikimedia.org/wikipedia/commons/thumb/d/d9/Earth_by_the_E
PIC_Team_on_21_April_2018.png/280px-
Earth_by_the_EPIC_Team_on_21_April_2018.png",
                "images/123.jpeg",
                "ftp(s)://(user:password@)my-ftp-
server(:21)/vod/posters.123456.svg",
                "sftp://(user:password@)my-ftp-
server(:22)/vod/posters.123456.svg",
                "s3://keyId:keySecret@aws-region/bucket-
name/test.png"
            ]
        }
    }
},
    "series_id": {
        "$id": "#/properties/categories/items/properties/series_id",
        "type": "string",
        "description": "If this category is a series, this must refer
to a series defined in series.json.",
        "examples": [
            "523c4c5c-68c5-4e00-a98b-754b9e9f4b71"
        ]
    },
    "season_id": {
        "$id": "#/properties/categories/items/properties/season_id",
        "type": "string",
        "description": "If this category is a season of a series,
this must refer to a season defined in series.json.",
        "examples": [
            "759bab6c-1168-4ed5-ab91-b4ec41501daa"
        ]
    },
    "parent_id": {
        "$id": "#/properties/categories/items/properties/parent_id",
        "type": "string",

```



```
"description": "ID of the parent category. Must refer to another category defined in categories.json.",
  "examples": [
    "44ee242a-0af2-4352-af3c-6d9cdc2c8822"
  ],
  "assets": {
    "$id": "#/properties/categories/items/properties/assets",
    "type": "array",
    "description": "List of IDs of assets contained in this category. Must refer to assets defined in assets.json.",
    "items": {
      "$id": "#/properties/categories/items/properties/assets/items",
      "type": "string",
      "examples": [
        "123c4c5c-68c5-4e00-a98b-754b9e9f4b71"
      ]
    }
  }
}
```

3.6 series.json

3.6.1 Example file

```
{
  "series": [
    {
      "id": "523c4c5c-68c5-4e00-a98b-754b9e9f4b71", // mandatory string
      "external_ids": { // mandatory; can be empty
        "tmdb": 123,
        "imdb": "tt5834204",
        "anmdb": "b375bbfc-6a76-4046-a486-b444af96cdc4"
      },
      "title": { // mandatory
        "eng": "lorem ipsum ..."
      },
      "synopsis": { // mandatory
        "eng": "lorem ipsum ..."
      },
      "images": [ // mandatory
        {
          "id": "7b4ef709-51d8-438c-b2b1-4123dc5c6754",
          "type": "poster", // mandatory
          "language": "eng", // optional (default null)
          // Mandatory
          // the image location. Valid values are:
          // - Absolute remote URLs:
          //   - http(s)://(user:password@)server.com(:80)/a/b/c/d.png
          //   - ftp(s)://(user:password@)my-ftp-
server(:21)/vod/posters.123456.svg
          //   - sftp://(user:password@)my-ftp-
server(:22)/vod/posters.123456.svg
```



```

        // - s3://keyId:keySecret@aws-region/bucket-name/test.png
        // - Relative local URLs:
        // - images/123.jpeg
        "location":
"https://upload.wikimedia.org/wikipedia/commons/thumb/d/d9/Earth_by_the_E
PIC_Team_on_21_April_2018.png/280px-
Earth_by_the_EPIC_Team_on_21_April_2018.png"
    }
],
    "seasons": [ // mandatory
    {
        "id": "759bab6c-1168-4ed5-ab91-b4ec41501daa", // mandatory
        "external_ids": { // optional
            "tmdb": 123,
            "imdb": 456,
            "anmdb": "b375bbfc-6a76-4046-a486-b444af96cdc4"
        },
        "season_number":1,
        "title": { // mandatory
            "eng": "lorem ipsum ..."
        },
        "synopsis": { // mandatory
            "eng": "lorem ipsum ..."
        },
        "images": [ // mandatory
            {
                "id": "7b4ef709-51d8-438c-b2b1-4123dc5c6754",
                "type": "poster", // mandatory
                "language": "eng", // optional
                // Mandatory
                // the image location. Valid values are:
                // - Absolute remote URLs:
                // -
http(s)://(user:password@)server.com(:80)/a/b/c/d.png
                // - ftp(s)://(user:password@)my-ftp-
server(:21)/vod/posters.123456.svg
                // - sftp://(user:password@)my-ftp-
server(:22)/vod/posters.123456.svg
                // - s3://keyId:keySecret@aws-region/bucket-
name/test.png
                // - Relative local URLs:
                // - images/123.jpeg
                "location":
"https://upload.wikimedia.org/wikipedia/commons/thumb/d/d9/Earth_by_the_E
PIC_Team_on_21_April_2018.png/280px-
Earth_by_the_EPIC_Team_on_21_April_2018.png"
            }
        ]
    }
]
}
]
}
}
}

```



3.6.2 JSON schema

```
{
  "definitions": {},
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "http://tucano.io/red/ingest/canonical/series.json",
  "type": "object",
  "title": "series.json file for tucano's standard ingest",
  "required": [
    "series"
  ],
  "properties": {
    "series": {
      "$id": "#/properties/series",
      "type": "array",
      "title": "The Series Schema",
      "items": {
        "$id": "#/properties/series/items",
        "type": "object",
        "title": "The Items Schema",
        "required": [
          "id",
          "external_ids",
          "title",
          "synopsis",
          "images",
          "seasons"
        ],
        "properties": {
          "id": {
            "$id": "#/properties/series/items/properties/id",
            "type": "string",
            "description": "Unique ID for the series",
            "examples": [
              "523c4c5c-68c5-4e00-a98b-754b9e9f4b71"
            ]
          },
          "external_ids": {
            "$id": "#/properties/series/items/properties/external_ids",
            "type": "object",
            "title": "Map of external IDs for this series. Can be empty.
Keys are the external system (eg IMDB); values are the ID in that system
as strings or numbers.",
            "required": [],
            "properties": {}
          },
          "title": {
            "$id": "#/properties/series/items/properties/title",
            "type": "object",
            "description": "Title of the series. This is a map of
translations. Keys are ISO 639-3 language codes; values are the title of
the series in that language.",
            "required": [],
            "properties": {}
          },
          "synopsis": {
            "$id": "#/properties/series/items/properties/synopsis",
            "type": "object",
```



```

        "description": "Synopsis of the series. This is a map of
translations. Keys are ISO 639-3 language codes; values are the synopsis
of the series in that language.",
        "required": [],
        "properties": {}
    },
    "images": {
        "$id": "#/properties/series/items/properties/images",
        "type": "array",
        "description": "List of images for this series.",
        "items": {
            "$id": "#/properties/series/items/properties/images/items",
            "type": "object",
            "required": [
                "id",
                "type",
                "location"
            ],
            "properties": {
                "id": {
                    "$id":
"#/properties/series/items/properties/images/items/properties/id",
                    "type": "string",
                    "description": "ID of the image",
                    "examples": [
                        "7b4ef709-51d8-438c-b2b1-4123dc5c6754"
                    ]
                },
                "type": {
                    "$id":
"#/properties/series/items/properties/images/items/properties/type",
                    "type": "string",
                    "description": "Image type; an arbitrary string agreed
to beforehand. Usually used ones are 'poster' and 'background',
respectively for a portrait and landscape format.",
                    "examples": [
                        "poster",
                        "background"
                    ]
                },
                "language": {
                    "$id":
"#/properties/series/items/properties/images/items/properties/language",
                    "type": "string",
                    "description": "Language of the image, as an ISO 639-3
code.",
                    "examples": [
                        "eng"
                    ],
                    "pattern": "^[a-z]{3}$"
                },
                "location": {
                    "$id":
"#/properties/series/items/properties/images/items/properties/location",
                    "type": "string",
                    "description": "Image location. Accepted values are
absolute remote URLs (http, ftp, sftp, Amazon S3...), or relative local
paths.",
                    "examples": [

```



```

"https://upload.wikimedia.org/wikipedia/commons/thumb/d/d9/Earth_by_the_E
PIC_Team_on_21_April_2018.png/280px-
Earth_by_the_EPIC_Team_on_21_April_2018.png",
    "images/123.jpeg",
    "ftp(s)://(user:password@)my-ftp-
server(:21)/vod/posters.123456.svg",
    "sftp://(user:password@)my-ftp-
server(:22)/vod/posters.123456.svg",
    "s3://keyId:keySecret@aws-region/bucket-
name/test.png"
    ]
    }
  }
},
"seasons": {
  "$id": "#/properties/series/items/properties/seasons",
  "type": "array",
  "description": "List of seasons of this series",
  "items": {
    "$id":
"#/properties/series/items/properties/seasons/items",
    "type": "object",
    "required": [
      "id",
      "title",
      "synopsis",
      "season_number",
      "images"
    ],
    "properties": {
      "id": {
        "$id":
"#/properties/series/items/properties/seasons/items/properties/id",
        "type": "string",
        "description": "Unique id of the season. If you don't
have any, you can use the series id with the season number as suffix.",
        "examples": [
          "759bab6c-1168-4ed5-ab91-b4ec41501daa"
        ]
      },
      "external_ids": {
        "$id":
"#/properties/series/items/properties/seasons/items/properties/external_i
ds",
        "type": "object",
        "description": "Map of external IDs for this season.
Can be empty. Keys are the external system (eg IMDB); values are the ID
in that system as strings or numbers.",
      },
      "title": {
        "$id":
"#/properties/series/items/properties/seasons/items/properties/title",
        "type": "object",
        "description": "This season's title. This is a map of
translations. Keys are ISO 639-3 language codes; values are the season's
title in that language."
      },
      "synopsis": {

```



```

        "$id":
"#/properties/series/items/properties/seasons/items/properties/synopsis",
        "type": "object",
        "description": "This season's synopsis. This is a map
of translations. Keys are ISO 639-3 language codes; values are the
season's synopsis in that language."
    },
    "season_number": {
        "$id":
"#/properties/series/items/properties/seasons/items/properties/season_num
ber",
        "type": "integer",
        "description": "This season's number."
    },
    "images": {
        "$id":
"#/properties/series/items/properties/seasons/items/properties/images",
        "type": "array",
        "description": "List of images for this season.",
        "items": {
            "$id":
"#/properties/series/items/properties/seasons/items/properties/images/ite
ms",
            "type": "object",
            "required": [
                "id",
                "type",
                "location"
            ],
            "properties": {
                "id": {
                    "$id":
"#/properties/series/items/properties/images/items/properties/images/item
s/properties/id",
                    "type": "string",
                    "description": "ID of the image",
                    "examples": [
                        "7b4ef709-51d8-438c-b2b1-4123dc5c6754"
                    ]
                },
                "type": {
                    "$id":
"#/properties/series/items/properties/seasons/items/properties/images/ite
ms/properties/type",
                    "type": "string",
                    "description": "Image type; an arbitrary string
agreed to beforehand. Usually used ones are 'poster' and 'background',
respectively for a portrait and landscape format.",
                    "examples": [
                        "poster",
                        "background"
                    ]
                },
                "language": {
                    "$id":
"#/properties/series/items/properties/seasons/items/properties/images/ite
ms/properties/language",
                    "type": "string",
                    "description": "Language of the image, as an ISO
639-3 code.",

```



```

        "examples": [
            {
                "eng",
                "spa"
            },
            {
                "pattern": "[a-z]{3}"
            }
        ],
        "location": {
            "$id": "#/properties/series/items/properties/seasons/items/properties/images/items/location",
            "type": "string",
            "description": "Image location. Accepted values are absolute remote URLs (http, ftp, sftp, Amazon S3...), or relative local paths.",
            "examples": [
                "https://upload.wikimedia.org/wikipedia/commons/thumb/d/d9/Earth_by_the_EPIC_Team_on_21_April_2018.png/280px-Earth_by_the_EPIC_Team_on_21_April_2018.png",
                "images/123.jpeg",
                "ftp(s)://(user:password@)my-ftp-server(:21)/vod/posters.123456.svg",
                "sftp://(user:password@)my-ftp-server(:22)/vod/posters.123456.svg",
                "s3://keyId:keySecret@aws-region/bucket-name/test.png"
            ]
        }
    }
}

```

3.7 broadcasts.json

3.7.1 Example file

```
{
  "broadcasts": [
    {
      "idBroadcast": "5781b2f7-7f13-471c-9c8c-f7100f8491db",
      "channels": [
        "133733ef-676f-4442-a644-e403981bc3b2",
        "3b7ccfd0-c296-4671-a6f6-4f1ff9bf27ed"
      ],
      "idAsset": "e5dd7acd-c432-45d6-8d3c-6508bbb61a22",
      "start": 1568073600,
      "end": 1568073600,
      "catchupEnabled": true,
      "startoverEnabled": true,
      "npvrEnabled": false,
    }
  ]
}
```



```

        "metadata": {
            "AD_ASSTITLE": {
                "fra": "titre",
                "eng": "title"
            }
        }
    }
}
]
}

```

3.7.2 JSON schema

```

{
  "definitions": {},
  "$schema": "http://json-schema.org/draft-07/schema#",
  "$id": "http://tucano.io/red/ingest/canonical/broadcasts.json",
  "type": "object",
  "title": "broadcasts.json file for tucano's standard ingest",
  "required": [
    "broadcasts"
  ],
  "properties": {
    "broadcasts": {
      "$id": "#/properties/broadcasts",
      "type": "array",
      "items": {
        "$id": "#/properties/broadcasts/items",
        "type": "object",
        "required": [
          "idBroadcast",
          "channels",
          "idAsset",
          "start",
          "end",
          "catchupEnabled",
          "startoverEnabled",
          "npvrEnabled",
          "metadata"
        ],
        "properties": {
          "idBroadcast": {
            "$id": "#/properties/broadcasts/items/properties/idBroadcast",
            "type": "string",
            "description": "Unique ID for the broadcast",
            "examples": [
              "5781b2f7-7f13-471c-9c8c-f7100f8491db"
            ]
          },
          "channels": {
            "$id": "#/properties/broadcasts/items/properties/channels",
            "type": "array",
            "description": "Array of channel IDs. The mapping between channel IDs and channels configured in tucanoRED's CMS is configured in tucanoRED; there is no channels.json file.",
            "items": {
              "$id": "#/properties/broadcasts/items/properties/channels/items",
              "type": "string",

```



```

        "examples": [
            "133733ef-676f-4442-a644-e403981bc3b2",
            "3b7ccfd0-c296-4671-a6f6-4f1ff9bf27ed"
        ]
    },
    "idAsset": {
        "$id": "#/properties/broadcasts/items/properties/idAsset",
        "type": "string",
        "description": "ID of the asset being broadcasted. Reference
to assets.json.",
        "examples": [
            "e5dd7acd-c432-45d6-8d3c-6508bbb61a22"
        ]
    },
    "start": {
        "$id": "#/properties/broadcasts/items/properties/start",
        "type": "integer",
        "description": "UTC epoch timestamp of the time at which the
broadcast starts",
        "examples": [
            1568073600
        ]
    },
    "end": {
        "$id": "#/properties/broadcasts/items/properties/end",
        "type": "integer",
        "description": "UTC epoch timestamp of the time at which the
broadcast ends",
        "examples": [
            1568073600
        ]
    },
    "catchupEnabled": {
        "$id": "#/properties/broadcasts/items/properties/catchupEnabl
ed",
        "type": "boolean",
        "description": "Whether catch-up is enabled for this
broadcast.",
        "examples": [
            true,
            false
        ]
    },
    "startoverEnabled": {
        "$id": "#/properties/broadcasts/items/properties/startoverEna
bled",
        "type": "boolean",
        "description": "Whether start-over is enabled for this
broadcast.",
        "examples": [
            true,
            false
        ]
    },
    "npvrEnabled": {
        "$id": "#/properties/broadcasts/items/properties/npvrEnabled"
,
        "type": "boolean",
        "description": "Whether NPVR is enabled for this broadcast.",

```



```
    "examples": [
      false,
      true
    ]
  },
  "metadata": {
    "$id": "#/properties/broadcasts/items/properties/metadata",
    "type": "object",
    "description": "Arbitrary key/value pairs. Keys are strings; values can be text, numbers, booleans, or more complex data structures like lists of maps.",
    "properties": {}
  }
}
}
```

