



Digital Music, Photo, and Video Collections



MPV Broadcast Television Profile

Revision 1.0

October 2004

© 2002-2004 Optical Storage Technology Association

**LICENSING
IMPORTANT NOTICES
(Revised 5-5-2004)**

- (a) THIS DOCUMENT IS AN AUTHORIZED AND APPROVED PUBLICATION OF THE OPTICAL STORAGE TECHNOLOGY ASSOCIATION (OSTA). THE SPECIFICATIONS CONTAINED HEREIN ARE THE EXCLUSIVE PROPERTY OF OSTA BUT MAY BE REFERRED TO AND UTILIZED BY THE GENERAL PUBLIC FOR ANY LEGITIMATE PURPOSE, PARTICULARLY IN THE DESIGN AND DEVELOPMENT OF WRITABLE OPTICAL SYSTEMS AND SUBSYSTEMS. THIS DOCUMENT MAY BE COPIED IN WHOLE OR IN PART PROVIDED THAT NO REVISIONS, ALTERATIONS, OR CHANGES OF ANY KIND ARE MADE TO THE MATERIALS CONTAINED HEREIN.
- (b) COMPLIANCE WITH THIS DOCUMENT MAY REQUIRE USE OF ONE OR MORE FEATURES COVERED BY THE PATENT RIGHTS OF AN OSTA MEMBER, ASSOCIATE OR THIRD PARTY. NO POSITION IS TAKEN BY OSTA WITH RESPECT TO THE VALIDITY OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT, WHETHER OWNED BY A MEMBER OR ASSOCIATE OF OSTA OR OTHERWISE. OSTA HEREBY EXPRESSLY DISCLAIMS ANY LIABILITY FOR INFRINGEMENT OF INTELLECTUAL PROPERTY RIGHTS OF OTHERS BY VIRTUE OF THIS OSTA DOCUMENT, NOR DOES OSTA UNDERTAKE A DUTY TO ADVISE USERS OR POTENTIAL USERS OF OSTA DOCUMENTS OF SUCH NOTICES OR ALLEGATIONS. OSTA HEREBY EXPRESSLY ADVISES ALL USERS OR POTENTIAL USERS OF THIS DOCUMENT AND SPECIFICATIONS TO INVESTIGATE AND ANALYZE ANY POTENTIAL INFRINGEMENT SITUATION, SEEK THE ADVICE OF INTELLECTUAL PROPERTY COUNSEL AND, IF INDICATED, OBTAIN A LICENSE UNDER ANY APPLICABLE INTELLECTUAL PROPERTY RIGHT OR TAKE THE NECESSARY STEPS TO AVOID INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHT. OSTA EXPRESSLY DISCLAIMS ANY INTENT TO PROMOTE INFRINGEMENT OF ANY INTELLECTUAL PROPERTY RIGHT BY VIRTUE OF THE EVOLUTION, ADOPTION, OR PUBLICATION OF THIS OSTA DOCUMENT.
- (c) OSTA MAKES NO REPRESENTATION OR WARRANTY REGARDING ANY SPECIFICATION, AND ANY COMPANY USING A SPECIFICATION SHALL DO SO AT ITS SOLE RISK, INCLUDING SPECIFICALLY THE RISKS THAT A PRODUCT DEVELOPED WILL NOT BE COMPATIBLE WITH ANY OTHER PRODUCT OR THAT ANY PARTICULAR PERFORMANCE WILL NOT BE ACHIEVED. OSTA SHALL NOT BE LIABLE FOR ANY EXEMPLARY, INCIDENTAL, PROXIMATE OR CONSEQUENTIAL DAMAGES OR EXPENSES ARISING FROM THE USE OR IMPLEMENTATION OF THIS DOCUMENT. THIS DOCUMENT DEFINES ONLY ONE APPROACH TO COMPATIBILITY, AND OTHER APPROACHES MAY BE AVAILABLE IN THE INDUSTRY.
- (d) THIS DOCUMENT IS A SPECIFICATION ADOPTED BY OSTA. THIS DOCUMENT MAY BE REVISED BY OSTA AT ANY TIME AND WITHOUT NOTICE AND USERS ARE ADVISED TO OBTAIN THE LATEST VERSION. IT IS INTENDED SOLELY AS A GUIDE FOR ORGANIZATIONS INTERESTED IN DEVELOPING PRODUCTS WHICH CAN BE COMPATIBLE WITH OTHER PRODUCTS DEVELOPED USING THIS DOCUMENT. THIS DOCUMENT AND THE SPECIFICATIONS ARE PROVIDED "AS IS".
- (e) MPV, Music/Photo/Video AND THE MPV LOGO ARE TRADEMARKS OF OSTA. ALL OTHER TRADEMARKS ARE THE PROPERTY OF THEIR RESPECTIVE OWNERS. THE TRADEMARKS MPV, Music/Photo/Video AND THE MPV LOGO MAY NOT BE USED EXCEPT FOR JOURNALISTIC PURPOSES WITHOUT AN EXPLICIT LICENSE FROM OSTA. INFORMATION REGARDING THE MPV TRADEMARK LICENSE IS AVAILABLE AT <http://www.osta.org/mpv>. OBTAINING A LICENSE TO THE OSTA TRADEMARKS MPV, MusicPhotoVideo, AND/OR THE MPV LOGO FROM OSTA DOES NOT PROVIDE A LICENSE TO ANY PATENT REFERENCED HEREIN.
- (f) USE OF OSTA'S MPV TRADEMARKS OR ANY STATEMENT TO THE EFFECT THAT A PRODUCT IS "COMPATIBLE" WITH THE MPV SPECIFICATIONS REQUIRES THE LICENSE DESCRIBED ABOVE. USE OF THE MPV TRADEMARKS WITHOUT THE LICENSE IS PROHIBITED, EXCEPT THAT ATTRIBUTION FOR USE OF THE MPV SPECIFICATIONS MAY BE STATED AS FOLLOWS:

“BASED ON THE MPV™ (MusicPhotoVideo™) SPECIFICATIONS DEVELOPED BY THE OPTICAL STORAGE TECHNOLOGY ASSOCIATION (OSTA). MPV IS A TRADEMARK OF OSTA.”

(g) THE MPV SPECIFICATIONS MAY BE USED ONLY FOR PRODUCTS USED LAWFULLY. OSTA DOES NOT PROMOTE OR CONDONE THE USE OF THE SPECIFICATIONS IN ANY PRODUCT WHICH IS USED FOR AN UNLAWFUL PURPOSE.

POINTS OF CONTACT

<p><u>OSTA</u> David Bunzel OSTA President</p> <p>Tel: +1 (408) 253-3695 Email: dbunzel@osta.org</p> <p>http://www.osta.org</p> <p><u>MPV Website</u> http://www.osta.org/mpv/</p>	<p><u>Technical Content</u></p> <p>Comments and Feedback: Email: mpv-comments@list.osta.org</p> <p>Pieter van Zee Editor, MPV Specification MPV Working Group Chairman Tel: +1 541-715-8658 Email: Pieter.van.Zee@hp.com</p> <p>Eric Shalkey MPV Broadcast Television Initiative Lead Tel: +1 (781) 276-8766 Email: eric.shalkey@tvguide.com</p> <p>Martin Serrano MPV Broadcast Television Initiative Co-Author Email: martin.serrano@tvguide.com</p> <p>Felix Nemirovsky Chairman, MultiRead Committee Tel: +1 415 643 0944 Email: felix@chubaconsulting.com</p>
--	--

ABSTRACT

The Broadcast Television Profile defines metadata and practices for processing of collections of recorded television shows stored on a variety of storage media such as hard drives or optical disks, or exchanged via internet protocols.

COPYRIGHT NOTICE

Copyright 2002-2004 Optical Storage Technology Association, Inc.

RELEASE HISTORY

<i>Version</i>	<i>Date</i>	<i>Comments</i>
1.0	September 30, 2004	First Public Release

Contents

Contents	6
Chapter 1: Introduction	8
1.1 Executive Summary.....	8
Chapter 2: MPV Broadcast Television Profile 1.0	9
2.1 Formalities For Use of the MPV Broadcast Television Profile.....	9
Chapter 3: MPV Broadcast Television Schema Introduction	11
3.1 Introduction	11
3.2 Examples.....	11
3.2.1 Namespaces and Profiles	11
3.2.2 Simple Example.....	12
3.2.3 Richer Example	13
3.3 MPV Broadcast Television Profile Metadata Introduction	18
Chapter 4: MPV Broadcast Television Video Asset Metadata	24
4.1 Locating a TVShow Asset : mpv:LastURL.....	24
4.2 MPV / Dublin Core TVShow Metadata	24
4.3 MPV / Dublin Core Qualifier TVShow Metadata.....	26
4.4 Broadcast Specification TVShow Metadata.....	27
4.4.1 Basic Broadcast Profile TVShow Data.....	27
4.4.2 Compound Broadcast Profile TVShow Data	29
4.4.3 Container-based Broadcast Profile TVShow Data.....	31
4.5 MPV Core Metadata Extended by Broadcast TV Profile	38
4.5.1 Audio	38
4.5.2 Renditions of a TVShow Asset.....	40
4.5.3 Related References to a TVShow Asset	41
4.6 OSTA MD5 Digital Signature Identification	42
Chapter 5: MPV Broadcast Television Playlists	44
5.1 MPV Broadcast Television Playlists.....	44

5.1.1	MPV / Dublin Core Element Usage.....	45
5.1.2	Broadcast Specification Metadata.....	45
5.1.3	PlaylistAsset Metadata.....	45
5.1.4	Summary Example.....	45
Chapter 6:	Broadcast Television Profile Manifests.....	48
6.1	<i>Best Practices for Generating Manifest Data.....</i>	48
6.1.1	Top-Level Playlist	48
Chapter 7:	MPV Broadcast Television Profile Extensions to MPV Core Specification	49
7.1	<i>Broadcast Television Manifest File Types & Extensions.....</i>	49
7.2	<i>TVShow Manifest MIME Media Type.....</i>	50
7.3	<i>Choosing Which File Type and MIME Media Type to Use</i>	50
7.4	<i>Finding a TVShow Manifest File</i>	50
7.5	<i>Media Types and File Formats</i>	52
Appendix I:	References	53

Chapter 1: Introduction

1.1 Executive Summary

MPV is an open specification that makes the representation, exchange, processing and playback of collections of digital media content (including music, still images, stills with audio, still sequences, video clips, and audio clips) easier for a consumer.

The development and promotion of MPV is sponsored by the Optical Storage Technology Association (OSTA). The specification development and promotion process is open to all members; all organizations and individuals are welcomed as members. The association includes over 50 member companies from all over the world that produce products that collectively represent a majority marketshare in mainstream recordable optical storage categories.

MPV uses a text-based format that is easily understood and also easy to produce and consume programmatically in firmware or computer software. MPV does *not* tackle a large number of problems at once – instead, it focuses on a few key problems that it solves with simple but robust approaches. Where possible and practical, it supports use of established specifications and standards.

Applications, devices and consumers that use MPV benefit even when they only interact with music and audio in basic ways such as personal music collections that can be burned on CDs by many software applications.

MPV technology has three central components: Collections, Metadata, and Identification. Each of these make reference in various ways to data files containing the music, photo, or video content. This information may be augmented by information from various profiles. For example, the Presentation profile provides information that may be used by player applications and devices to provide an attractive playback user experience.

The MPV Core Specification sets out specific formalities to follow -- an MPV file must declare which specifications it implements along with their namespaces. This allows a processing application to quickly determine whether a given MPV file meets its expectations for processing.

MPV is not only a specification. It also includes a compliance test suite and processes, compliance testing materials, a logo program for compliant products, and a website. These materials and procedures are made available and administered by OSTA at a modest cost. OSTA charges no royalty for use of the specification or logo. In addition, sample open-source code implementations of key steps in processing MPV content are being contributed by interested parties.

Chapter 2: MPV Broadcast Television Profile 1.0

With the advent of Digital Video Recorders (DVRs), users can easily record many television shows to a variety of digital media types. The MPV Broadcast Television Profile allows users, via applications and devices, to create and manipulate collections of broadcast television content organized into playlists. The MPV Broadcast Television Profile extends the existing MPV Core Specification by augmenting this framework with additional metadata and practices specific to television content.

A user may start with a collection of recordings made to a hard drive, then organize and burn a subset on a DVD. When video collections are represented on the disc using MPV files that implement the MPV Broadcast Television Profile, a playback application or device can quickly start playback when the disc is inserted and allow the user to easily navigate and play video content. In addition to basic video playback, an application or device can display television show information like titles and descriptions. Additional content may also be part of the collection and is available to be shown by the playback application or device, such as production years, actors, directors, or other kinds of pertinent data.

2.1 Formalities For Use of the MPV Broadcast Television Profile

COMPATIBILITY

The MPV Broadcast Television Profile 1.0 is an extension of the MPV Core Specification 1.0 and is fully compatible with the MPV framework it establishes. Thus MPV files that implement the MPV Broadcast Television Profile should be usable in basic ways by MPV-aware applications and devices not focused on video playback. This means, for example, that an MPV playback application or device can read and playback Broadcast Television Profile collections even if it doesn't understand the MPV Broadcast Television Profile; however, the TV-specific information will be ignored and the playback experience will be less full-featured than in a MPV Broadcast Television Profile compliant player.

SCHEMA NAMESPACE

To use the MPV Broadcast Television Profile, this information must be present in the namespace declarations of the MPV manifest:

Schema	Namespace Identifier	Schema Location	Conventional Namespace Prefix
Broadcast Television Profile	http://ns.osta.org/mpv/tv/1.0/	lax/profiles/tv/profile.xsd	mpvtv:

The schema location may be specified optionally. Multiple schema variations may exist depending on the degree of validation desired by the developer. Typical variations include “lax”, “strict”, and “fixed”. These schema will all implement the grammar of the MPV Broadcast Television Profile but will vary in the degree of flexibility and conformance requirements that they embody.

EXAMPLE

```
<?xml version="1.0" encoding="UTF-8"?>
<file:Manifest
  xmlns:file="http://ns.osta.org/manifest/1.0/"
  xmlns:file2="http://ns.osta.org/manifest/2.0/"
  xmlns:mpv="http://ns.osta.org/mpv/1.0/"
  xmlns:mpvtv="http://ns.osta.org/mpv/tv/1.0/"
  xmlns:dc="http://ns.osta.org/nmf/1.0/dc/"

  <nmf:Metadata>
    <file:ManifestProperties>
      <file2>AboutManifestMPVDocumentID>DOCID001</file2>AboutManifestMPVDocumentID>
      <file2:WrittenBy>http://www.mycompany.com/myapp/1.05/</file2:WrittenBy>
      <ProfileBag>
        <Profile>http://ns.osta.org/mpv/basic/1.0/</Profile>
        <Profile>http://ns.osta.org/mpv/dc/1.0/</Profile>
        <Profile>http://ns.osta.org/mpv/tv/1.0/</Profile>
      </ProfileBag>
    </file:ManifestProperties>
  </nmf:Metadata>

  <mpv:Document mpv:id="DOCID001">
    <mpv:InstanceID>urn.osta-org.mpv.uid.2342945BAEFF98AE8848848723</mpv:InstanceID>
    <nmf:Metadata>
      <dc:Properties>
        <dc:creator>My Company, Inc.</dc:creator>
        <dc:rights>(c) My Company. All rights reserved.</dc:rights>
        <dc:language>en-US</dc:language>
      </dc:Properties>
    </nmf:Metadata>
  </mpv:Document>
  ...
</file:Manifest>
```

Chapter 3: MPV Broadcast Television Schema Introduction

3.1 Introduction

The MPV Broadcast Television Profile makes use of the existing **[MPVCore]** Specification for creating collections of video content and organizes them into AssetLists. For metadata, it uses the MPV Dublin Core NMF specification wherever possible.

The TV metadata that may be represented using the MPV Broadcast Television Profile includes the following:

TVShow/Program: Filename, Show Title, Episode Title, Description, Ratings, Cast Members, Station Information, Start Time, Duration, Genre, and others.

Playlist: Title, Description, Playlist Assets, and others.

Unlike other MPV Profiles, the Broadcast Television Profile does not depend on the MPV Presentation Profile. The Broadcast Television Profile concerns itself with describing the metadata associated with a show, rather than defining playback experiences. Devices that support the MPV Broadcast Television Profile are free to pick and choose the relevant metadata that generates an acceptable user experience for that device.

This is not, however, intended to preclude the use of a MPV Broadcast Television Profile defined AssetList by devices that support the MPV Presentation Profile.

3.2 Examples

MPV Broadcast Television AssetLists can range from simple to sophisticated, depending on the amount of available information and the ability of the creating application or device. Playback applications and devices determine the extent to which they use available information and the presentation of that information.

3.2.1 Namespaces and Profiles

All MPV files begin with a preamble that declares the XML namespaces and profiles used by the file. The `xmlns:xyz="namespace identifier"` sequence assigns a shortcut prefix (xyz) to represent the unique namespace identifier within the file. Use of namespaces allows the same element name to be used from different schema without ambiguity. For example, `<foo:Element>` and `<bar:Element>` are different if the namespace identifiers for each prefix are different and are the same if the namespace identifiers are the same.

A typical preamble:

```
<?xml version="1.0" encoding="UTF-8"?>
<file:Manifest
  xmlns:file="http://ns.osta.org/manifest/1.0/"
  xmlns:file2="http://ns.osta.org/manifest/2.0/"
  xmlns:mpv="http://ns.osta.org/mpv/1.0/"
  xmlns:mpvtv="http://ns.osta.org/mpv/tv/1.0/"
  xmlns:dc="http://ns.osta.org/nmf/1.0/dc/"
  xmlns:nmf="http://ns.osta.org/nmf/1.0/"

  <nmf:Metadata>
    <file:ManifestProperties>
      <file2>AboutManifestMPVDocumentID>DOCID001</file2>AboutManifestMPVDocumentID>
      <file2:WrittenBy>http://www.mycompany.com/myapp/1.05/</file2:WrittenBy>
      <ProfileBag>
        <Profile>http://ns.osta.org/mpv/basic/1.0/</Profile>
        <Profile>http://ns.osta.org/nmf/1.0/dc/</Profile>
        <Profile>http://ns.osta.org/mpv/tv/1.0/</Profile>
      </ProfileBag>
    </file:ManifestProperties>
  </nmf:Metadata>

  <mpv:Document mpv:id="DOCID001">
    <mpv:InstanceID>urn.osta-org.mpv.uid.2342945BAEFF98AE8848848723</mpv:InstanceID>
    <nmf:Metadata>
      <dc:Properties>
        <dc:creator>My Company, Inc.</dc:creator>
        <dc:rights>(c) My Company. All rights reserved.</dc:rights>
        <dc:language>en-US</dc:language>
      </dc:Properties>
    </nmf:Metadata>
  </mpv:Document>

  ...
```

As a best practice, the Basic and Dublin Core profiles SHOULD be listed in the manifest. Only if a writer is adding terms from the Presentation profile should the Presentation profile be listed in the manifest. Adding additional profile entries allow players that only understand those profiles to provide simple playback of a MPV file implementing the Broadcast Television Profile.

3.2.2 Simple Example

This example MPV Broadcast Television Profile file has three programs with only file location information for each program. There is no <mpvtv:Playlist> playlist, so the presentation sequence is the order of appearance in the <mpvtv:AssetList>.

Even with this very simple usage, the MPV AssetList adds value to the user's playback experience because the order of video playback is specified explicitly and is different from the sort order of the video content by filename or file date.

```
<?xml version="1.0" encoding="UTF-8"?>
<file:Manifest
  xmlns:file="http://ns.osta.org/manifest/1.0/"
  xmlns:file2="http://ns.osta.org/manifest/2.0/"
  xmlns:mpv="http://ns.osta.org/mpv/1.0/"
```

```

xmlns:mpvtv="http://ns.osta.org/mpv/tv/1.0/"
xmlns:dc="http://ns.osta.org/nmf/1.0/dc/"
xmlns:nmf="http://ns.osta.org/nmf/1.0/"

<nmf:Metadata>
  <file:ManifestProperties>
    <file2>AboutManifestMPVDocumentID>DOCID001</file2>AboutManifestMPVDocumentID>
    <file2:WrittenBy>http://www.mycompany.com/myapp/1.05/</file2:WrittenBy>
    <ProfileBag>
      <Profile>http://ns.osta.org/mpv/basic/1.0/</Profile>
      <Profile>http://ns.osta.org/nmf/1.0/dc/</Profile>
      <Profile>http://ns.osta.org/mpv/tv/1.0/</Profile>
    </ProfileBag>
  </file:ManifestProperties>
</nmf:Metadata>

<mpv:Document mpv:id="DOCID001">
  <mpv:InstanceID>urn.osta-org.mpv.uuid.2342945BAEFF98AE8848848723</mpv:InstanceID>
  <nmf:Metadata>
    <dc:Properties>
      <dc:creator>My Company, Inc.</dc:creator>
      <dc:rights>(c) My Company. All rights reserved.</dc:rights>
      <dc:language>en-US</dc:language>
    </dc:Properties>
  </nmf:Metadata>
</mpv:Document>

<mpv:AssetList>
  <mpv:Video mpv:id="SEINFELD070999 ">
    <mpv:LastURL mpv:filesystem="UDF">
      Seinfeld_070999.MPG
    </mpv:LastURL>
  </mpv:Video>
  <mpv:Video mpv:id="FRIENDS071099 ">
    <mpv:LastURL mpv:filesystem="UDF">
      Friends_071099.MPG
    </mpv:LastURL>
  </mpv:Video>
  <mpv:Video mpv:id="FRIENDS071199 ">
    <mpv:LastURL mpv:filesystem="UDF">
      Friends_071199.MPG
    </mpv:LastURL>
  </mpv:Video>
</mpv:AssetList>
</file:Manifest>

```

3.2.3 Richer Example

In contrast to the previous example, the following example has much more information. In this case, a single video is specified along with a lot of information about it including titles, descriptions, record date, source channel, recorded duration, etc.

Note there is also a Playlist (the *mpvtv:Playlist* element) provided that specifies playlist-level information, such as the current resume point of the underlying asset, when it can be deleted, etc.

Careful reading of the *mpv:AssetList* in this example illustrates a more complete description of video content as well as embedded audio, associated external audio, alternate still image thumbnails and preview video assets. Not all these assets are considered primary: the user doesn't want to interact with all assets equally. Primary assets are the ones that match the user's idea of what the primary content is, such as a collection of videos. The *mpvtv:PlaylistAsset* element is used to identify the primary assets and other presentation information.

While this example does not illustrate this, the MPV Broadcast Television Profile can be applied to a "hybrid" storage medium, such as a hard drive that contains video, MP3 music, or still photos. When played in a compliant player, the user may enjoy listening to the MP3 music performances. In this case, no MPV Broadcast Television information is used, just audio content and navigation. Conversely, a Broadcast Television Profile device can ignore the MP3 music and concentrate on relevant video content.

Filenames of assets are specified using the *mpv:LastURL* element. Pathnames can be relative or absolute; relative names begin relative to the location of the MPV file. Pathnames are to be specified using URL-compliant syntax. This includes translation of special characters like the space (" ") into equivalent representations like "%20" and use of prefixes like *file:///* to introduce absolute pathnames to local files. Multiple pathnames may be specified for any given asset; they are interpreted as alternate paths to the same set of bits. A processing application tries them sequentially to try and locate the asset.

The term "LastURL" is used to emphasize that its value is a URL to the last-known location of the file; because media files may be moved or renamed independently of the MPV file, it is possible that the media file has moved and must be searched for. The *mpv:InstanceID* and *mpv:ContentID* elements, if provided, are identifiers that can be used to find files that cannot be located by any of the *LastURL* entries.

```
<?xml version="1.0" encoding="UTF-8"?>
<file:Manifest
  xmlns:file="http://ns.osta.org/manifest/1.0/"
  xmlns:file2="http://ns.osta.org/manifest/2.0/"
  xmlns:mpv="http://ns.osta.org/mpv/1.0/"
  xmlns:mpvtv="http://ns.osta.org/mpv/tv/1.0/"
  xmlns:dc="http://ns.osta.org/nmf/1.0/dc/"
  xmlns:nmf="http://ns.osta.org/nmf/1.0/"

  <nmf:Metadata>
    <file:ManifestProperties>
      <file2>AboutManifestMPVDocumentID>DOCID001</file2>AboutManifestMPVDocumentID>
      <file2:WrittenBy>http://www.mycompany.com/myapp/1.05/</file2:WrittenBy>
      <ProfileBag>
        <Profile>http://ns.osta.org/mpv/basic/1.0/</Profile>
        <Profile>http://ns.osta.org/nmf/1.0/dc/</Profile>
        <Profile>http://ns.osta.org/mpv/tv/1.0/</Profile>
      </ProfileBag>
    </file:ManifestProperties>
  </nmf:Metadata>

  <mpv:Document mpv:id="DOCID001">
    <mpv:InstanceID>urn.osta-org.mpv.uid.2342945BAEFF98AE8848848723</mpv:InstanceID>
    <nmf:Metadata>
      <dc:Properties>
        <dc:creator>My Company, Inc.</dc:creator>
        <dc:rights>(c) My Company. All rights reserved.</dc:rights>
        <dc:language>en-US</dc:language>
      </dc:Properties>
    </nmf:Metadata>
  </mpv:Document>
```

```

<mpv:AssetList>
  <mpv:Video mpv:id="VID0001">
    <mpv:lastURL mpv:filesystem="UDF"/>/Seinfeld/Seinfeld84.mpg</mpv:lastURL>
  <nmf:Metadata>

    <dc:Properties>
      <dc:title>Seinfeld</dc:title>
      <dc:description>Elaine fails a drug test---twice---and loses her job; Jerry talks about
his Uncle Leo on "The Tonight Show."</dc:description>
      <dc:creator>TV Guide</dc:creator>
      <dc:language>en-US</dc:language>
      <dc:format>video/mpeg</dc:format>
    </dc:Properties>

    <dct:Properties>
      <dct:Abstract>Four neurotic New Yorkers in a show about nothing.</dct:Abstract>
    </dct:Properties>

    <mpvtv:ShowProperties>

      <mpvtv:episodeTitle>The Showerhead</mpvtv:episodeTitle>
      <mpvtv:startTime>2004-03-24T19:01</mpvtv:startTime>
      <mpvtv:duration>28.0</mpvtv:duration>
      <mpvtv:channel>25-2</mpvtv:channel>
      <mpvtv:expectedStartTime>2004-03-24T19:00</mpvtv:expectedStartTime>
      <mpvtv:expectedDuration>30.0</mpvtv:expectedDuration>
      <mpvtv:productionYear>1998</mpvtv:productionYear>
      <mpvtv:firstAirDate>1998-02-15</mpvtv:firstAirDate>

      <!-- Here is a genreSeq that defines a primary genre of Comedy, secondary of Sitcom -->
      <mpvtv:genreSeq mpvtv:vocabulary="urn:osta-org:mpv:tv:genre:tvguide">
        <mpvtv:genre>Comedy</mpvtv:genre>
        <mpvtv:genre>Sitcom</mpvtv:genre>
      </mpvtv:genreSeq>

      <!-- As for ratings, let's have one US, two ratings reasons -->
      <mpvtv:rating mpvtv:ratingAuthority="urn:osta-org:mpv:tv:rating:en-US">
        <mpvtv:ratingValue>TV-PG</mpvtv:ratingValue>
        <mpvtv:ratingReason>Language</mpvtv:ratingReason>
        <mpvtv:ratingReason>Suggestive Dialog</mpvtv:ratingReason>
      </mpvtv:rating>

      <!-- Cast members are next -->
      <mpvtv:castMemberBag>
        <mpvtv:castMember mpvtv:castType="urn:osta-org:mpv:tv:cast:actor">
          <mpvtv:castMemberName>
            <mpvtv:castMemberSurname>Seinfeld</mpvtv:castMemberSurname>
            <mpvtv:castMemberGivenName>Jerry</mpvtv:castMemberGivenName>
          </mpvtv:castMemberName>
          <mpvtv:castMemberRole>
            <mpvtv:castMemberGivenName>Jerry</mpvtv:castMemberGivenName>
          </mpvtv:castMemberRole>
        </mpvtv:castMember>

        <mpvtv:castMember mpvtv:castType="urn:osta-org:mpv:tv:cast:actor">
          <mpvtv:castMemberName>

```

```

        <mpvtv:castMemberSurname>Louis-Dreyfus</mpvtv:castMemberSurname>
        <mpvtv:castMemberGivenName>Julia</mpvtv:castMemberGivenName>
    </mpvtv:castMemberName>
    <mpvtv:castMemberRole>
        <mpvtv:castMemberGivenName>Elaine</mpvtv:castMemberGivenName>
    </mpvtv:castMemberRole>
</mpvtv:castMember>

<mpvtv:castMember mpvtv:castType="urn:osta-org:mpv:tv:cast:actor">
    <mpvtv:castMemberName>
        <mpvtv:castMemberSurname>Richards </mpvtv:castMemberSurname>
        <mpvtv:castMemberGivenName>Michael</mpvtv:castMemberGivenName>
    </mpvtv:castMemberName>
    <mpvtv:castMemberRole>
        <mpvtv:castMemberGivenName>Kramer</mpvtv:castMemberGivenName>
    </mpvtv:castMemberRole>
</mpvtv:castMember>

<mpvtv:castMember mpvtv:castType="urn:osta-org:mpv:tv:cast:actor">
    <mpvtv:castMemberName>
        <mpvtv:castMemberSurname>Alexander</mpvtv:castMemberSurname>
        <mpvtv:castMemberGivenName>Jason</mpvtv:castMemberGivenName>
    </mpvtv:castMemberName>
    <mpvtv:castMemberRole>
        <mpvtv:castMemberGivenName>George</mpvtv:castMemberGivenName>
    </mpvtv:castMemberRole>
</mpvtv:castMember>

</mpvtv:castMemberBag>

<!-- A single showID -->
<mpvtv:showID mpvtv:idAuthority="urn:osta-org:mpv:tv:id:tvguide">
84990999090229</mpvtv:showID>

<!-- embedded audio -->
<mpvtv:Audio>
    <nmf:Metadata>
        <dc:properties>
            <dc:description>Main Audio</dc:description>
            <dc:language>en-US</dc:language>
            <dc:format>audio/mpeg</dc:format>
        </dc:properties>
        <mpvtv:ShowProperties>
            <mpvtv:audioType>normal</mpvtv:audioType>
            <mpvtv:audioEncoding>stereo</mpvtv:audioEncoding>
        </mpvtv:ShowProperties>
    </nmf:Metadata>
</mpvtv:Audio>

<!-- Reference to external audio stream(s) -->
<mpv:AudioRef mpv:refId="AUD00000001"/>

</mpvtv:ShowProperties>

</nmf:Metadata>

<!-- Related noCommercial video version of this asset -->

```



```

    <mpv:Related mpv:relationship="urn:osta-org:mpv:tv:relation:noCommercial">
      <mpv:VideoRef mpv:idref = "VID00010001"/>
    </mpv:Related>

  </mpv:Video>

  <!-- Audio associated with video, referenced by an AudioRef -->
  <mpv:Audio mpv:id="AUD00000001">
    <mpv:lastURL mpv:filesystem="UDF">ShowAudio.mpg</mpv:lastURL>
    <mpv:ContentID>urn:osta-org:dsig:md5:all:8735353445934AC</mpv:ContentID>
    <nmf:Metadata>
      <dc:properties>
        <dc:description>Spanish SAP Audio</dc:description>
        <dc:language>es-US</dc:language>
        <dc:format>audio/mpeg</dc:format>
      </dc:properties>
      <mpvtv:ShowProperties>
        <mpvtv:audioType>normal</mpvtv:audioType>
        <mpvtv:audioEncoding>stereo</mpvtv:audioEncoding>
        <mpvtv:audioSize>146</mpvtv:audioSize>
      </mpvtv:ShowProperties>
    </nmf:Metadata>
  </mpv:Audio>

  <!-- Related video, defines a commercial-free index point set of base video -->
  <!-- Note that there is no new instance of the underlying show in this case -->
  <mpv:Video mpv:id="VID00010001">
    <nmf:Metadata>
      <dc:Properties>
        <dc:title>No Commercial Version</dc:title>
      </dc:Properties>

      <mpvtv:ShowProperties>
        <!-- IndexPointSeq providing an ordered sequence of index points -->
        <mpvtv:IndexPointSeq>
          <mpvtv:IndexPoint>00:00:00:00</mpvtv:IndexPoint>
          <mpvtv:IndexPoint>00:10:00:00</mpvtv:IndexPoint>
          <mpvtv:IndexPoint>00:20:00:00</mpvtv:IndexPoint>
        </mpvtv:IndexPointSeq>

        <!-- CutPointSeq defines an ordered list of start-stop pairs -->
        <!-- This particular one cuts out minutes 13-16 of the original -->
        <mpvtv:CutPointSeq>
          <mpvtv:IndexPoint>00:00:00:00</mpvtv:IndexPoint>
          <mpvtv:IndexPoint>00:12:12:30</mpvtv:IndexPoint>
          <mpvtv:IndexPoint>00:17:00:00</mpvtv:IndexPoint>
          <mpvtv:IndexPoint>00:30:00:00</mpvtv:IndexPoint>
        </mpvtv:CutPointSeq>

      </mpvtv:ShowProperties>
    </nmf:Metadata>

    <!-- This video asset IS A commercial free version of the underlying video -->
    <mpv:Related mpv:relationship="urn:osta-org:mpv:tv:relation:noCommercialOf">
      <mpv:VideoRef mpv:idRef="VID0001"/>
    </mpv:Related>
  </mpv:Video>

```

```

        </mpv:Related>

        </mpv:Video>

    </mpv:AssetList>

    <mpvtv:Playlist mpvtv:id="PLAYLIST0001">
    <mpv:InstanceID>urn:osta-org:dsig:md5:all:043449CA9577688</mpv:InstanceID>
    <nmf:Metadata>
        <!-- General playlist specific properties -->
        <dc:Properties>
            <dc:title>Seinfeld Episodes</dc:title>
            <dc:description>My Favorites</dc:description>
            <dc:language>en-US</dc:language>
        </dc:Properties>
    </nmf:Metadata>

    <!-- Playlist assets refer to video assets in asset list -->
    <!-- First Asset -->
    <mpvtv:PlaylistAsset>
    <nmf:Metadata>
        <mpvtv:ShowProperties>
            <mpvtv:resumePoint>00:00:00:00</mpvtv:resumePoint>
            <mpvtv:lastAccess>1999-10-07T19:00 </mpvtv:lastAccess>
            <mpvtv:keepUntil>2030-12-31</mpvtv:keepUntil>
        </mpvtv:ShowProperties>
    </nmf:Metadata>

    <!-- Reference to video asset itself (Original video)-->
    <mpv:VideoRef mpv:idRef="VID0001"/>
    </mpvtv:PlaylistAsset>

    </mpvtv:Playlist>
</file:Manifest>

```

3.3 MPV Broadcast Television Profile Metadata Introduction

The [MPVCore] specification supports the concept of Assets organized into an AssetList. To this framework, the MPV Broadcast Television Profile adds a new nmf:Metadata type (mpvtv:ShowProperties) to the base mpv:Video asset. ShowProperties add metadata that specifically describes television content; the existing framework continues to be used in a manner fully consistent with existing specifications.

<i>Metadata</i>	<i>MPV Broadcast Television Profile</i>	<i>Discussion</i>
Mpv:Video Asset	Subelements of the mpv:Video asset	
Pathname	mpv:LastURL	one or more pathnames that should resolve to the file. Each pathname is specific to a filesystem, so there can be different names

		for each filesystem on a storage device.
Title	nmf:Metadata dc:Properties dc:title	Full Show Title
Short Title	nmf:Metadata dct:Properties dct:alternative	Short Show Title – may be more suitable for display when the Full Show Title is too long.
Episode Title	nmf:Metadata mpvtv:ShowProperties mpvtv:episodeTitle	Episode Title
Description	nmf:Metadata dc:Properties dc:description	Epsiode specific description
Show Description	nmf:Metadata dct:Properties dct:Abstract	Generic program description covering all episodes
Start Time	nmf:Metadata mpvtv:ShowProperties mpvtv:startTime	Actual Start Date/Time, ISO 8601 (local + timezone offset)
Duration	nmf:Metadata mpvtv:ShowProperties mpvtv:duration	Actual Duration, minutes, fractional seconds
Expected Start Time	nmf:Metadata mpvtv:ShowProperties mpvtv:expectedStartTime	Scheduled Start Date/Time, ISO 8601 (local + timezone offset)
Expected Duration	nmf:Metadata mpvtv:ShowProperties mpvtv:expectedDuration	Scheduled minutes, fractional seconds
Genre	nmf:Metadata mpvtv:ShowProperties mpvtv:genre	May be encoded as an ordered sequence of genres using mpvtv:genreSeq
Creator	nmf:Metadata dc:Properties dc:creator	Creator of the metadata content
Language	nmf:Metadata dc:Properties dc:language	RFC1766, optional territory, e.g., “en-US”
Rating(Bag)	nmf:Metadata mpvtv:ShowProperties mpvtv:Rating(Bag)	Container for ratings information. ratingAuthority URN types to describe rating type (MPAA, USTV, CATVEN, CATVFR, Star)
RatingValue	nmf:Metadata mpvtv:ShowProperties mpvtv:Rating(Bag) mpvtv:RatingValue	Actual Rating: MPAA: (G, PG, PG-13, R, NC-17, NR, X) USTV: : (TV-Y, TV-Y7, TV-G, TV-PG, TV-14, TV-MA) Star: Fractional, up to 5 by .5
RatingReason	nmf:Metadata mpvtv:ShowProperties mpvtv:Rating(Bag) mpvtv:RatingReason	Rating Reason: MPAA: : (Adult Situations, Graphic Language, Language, Brief Nudity, Strong Sexual Content,

		Graphic Violence, Mild Violence, Violence, Rape) USTV: (FV, V, S, L, D)
Production Year	nmf:Metadata mpvtv:ShowProperties mpvtv:productionYear	4 digit year
First Air Date	nmf:Metadata mpvtv:ShowProperties mpvtv:firstAirDate	ISO 8601 Date / Time (local + timezone offset)
Alternate Description Container	nmf:Metadata mpvtv:ShowProperties mpvtv:Description(Bag)	Container for one or more optional descriptions, description has mpvtv:creator URN type
Alternate Description Language	nmf:Metadata mpvtv:ShowProperties mpvtv:Description(Bag) dc:language	Language of alternate description, RFC1766, optional territory, e.g., "en-US".
Alternate Description	nmf:Metadata mpvtv:ShowProperties mpvtv:Description(Bag) dc:description	Alternate Description
Cast Member	nmf:Metadata mpvtv:ShowProperties mpvtv:CastMember(Bag)	Container for cast member data
Cast Member Type	nmf:Metadata mpvtv:ShowProperties mpvtv:CastMember(Bag) mpvtv:castMemberType	Cast Member Type (actor, director, writer, producer). See text.
Cast Member Name	nmf:Metadata mpvtv:ShowProperties mpvtv:castMember(Bag) mpvtv:castMemberName	Cast Member Name container
Cast Member Role	nmf:Metadata mpvtv:ShowProperties mpvtv:castMember(Bag) mpvtv:castMemberRole	Cast Member Role container, actors only
Cast Member Surname	(mpvtv:castMemberName mpvtv:castMemberRole) mpvtv:castMemberSurname	See text
Cast Member Given Name	(mpvtv:castMemberName mpvtv:castMemberRole) mpvtv:castMemberGivenName	See text
Station Info	nmf:Metadata mpvtv:ShowProperties mpvtv:StationInfo	Station Info container
Station Display Name	nmf:Metadata mpvtv:ShowProperties mpvtv:StationInfo mpvtv:stationName	Alpha, up to 8 characters
Station Full Name	nmf:Metadata mpvtv:ShowProperties mpvtv:StationInfo mpvtv:stationFullName	Alpha string
Station Logo	nmf:Metadata mpvtv:ShowProperties mpvtv:StationInfo mpvtv:stationLogo	Container for mpvtv:Still or StillRef
Channel	nmf:Metadata mpvtv:ShowProperties mpvtv:StationInfo mpvtv:channel	Receiver Channel, one part or two part
Affiliation	nmf:Metadata mpvtv:ShowProperties mpvtv:StationInfo mpvtv:stationAffiliation	Broadcast station network affiliation
Show Flags	nmf:Metadata mpvtv:ShowProperties mpvtv:ShowFlags	Container for descriptive flags
Closed Caption	nmf:Metadata mpvtv:ShowProperties mpvtv:ShowFlags mpvtv:cc	Closed caption
Letterbox	nmf:Metadata mpvtv:ShowProperties mpvtv:ShowFlags mpvtv:letterbox	Letterbox format

Live	nmf:Metadata mpvtv:ShowProperties mpvtv:ShowFlags mpvtv:live	Live broadcast
New	nmf:Metadata mpvtv:ShowProperties mpvtv:ShowFlags mpvtv:new	First airing
Repeat	nmf:Metadata mpvtv:ShowProperties mpvtv:ShowFlags mpvtv:repeat	Repeat airing
CopyInhibit	nmf:Metadata mpvtv:ShowProperties mpvtv:ShowFlags mpvtv:copyInhibit	Copy Inhibited
Encrypted	nmf:Metadata mpvtv:ShowProperties mpvtv:ShowFlags mpvtv:encrypted	Encrypted
Video Format	nmf:Metadata dc:Properties dc:format	Valid MIME type, e.g., video/mpeg
Video Encoding	nmf:Metadata mpvtv:ShowProperties mpvtv:VideoEncoding	Container for video encoding metadata
Method	nmf:Metadata mpvtv:ShowProperties mpvtv:VideoEncoding mpvtv:encodedMethod	(mpeg-cbr, mpeg-vbr) or other
Bitrate	nmf:Metadata mpvtv:ShowProperties mpvtv:VideoEncoding mpvtv:encodedBitrate	Integer kbps (max on DVD = 9800 kbps)
Size	nmf:Metadata mpvtv:ShowProperties mpvtv:VideoEncoding mpvtv:encodedSize	Size (in kB) of video stream
Source	nmf:Metadata mpvtv:ShowProperties mpvtv:VideoEncoding mpvtv:encodedSource	Source Encoding: (standardDef, highDef (720p or greater), enhancedDef (480p))
Standard	nmf:Metadata mpvtv:ShowProperties mpvtv:VideoEncoding mpvtv:encodedStandard	Video Standard (NTSC, SECAM, PAL, ATSC, DVB and variants, see text).
Audio	mpv:Audio nmf:Metadata	Container for audio encoding metadata.
Audio Type	mpv:Audio nmf:Metadata mpvtv:ShowProperties mpvtv:audioType	(normal, hearing impaired)
Audio Description	mpv:Audio nmf:Metadata dc:Properties dc:description	Descriptive presentation text for stream
Audio Description Language	mpv:Audio nmf:Metadata dc:Properties dc:language	Description language for descriptive text, RFC1766, optional territory
Audio Format	mpv:Audio nmf:Metadata dc:Properties dc:format	MIME, i.e., audio/mpeg
Audio Encoding	mpv:Audio nmf:Metadata mpvtv:ShowProperties mpvtv:audioEncoding	(mono, stereo, dolby surround, dolby digital 5.1)
Audio Filesize	mpv:Audio nmf:Metadata mpvtv:ShowProperties mpvtv:audioSize	Size (in kb) of audio stream
Audio Language	mpv:Audio nmf:Metadata mpvtv:ShowProperties mpvtv:audioLanguage	Spoken language of

	guage	stream, RFC1766, optional territory
ShowID(Bag)	nmf:Metadata mpvtv:ShowProperties mpvtv:ShowID(Bag)	Grouping of variety of unique categorization IDs, idAuthority type holds URN of registering authority (tvguide, v-isan, etc)
Index Point Sequence	nmf:Metadata mpvtv:ShowProperties mpvtv:IndexPointSeq	Ordered container for Index Points, defines a list of index points.
Index Point	nmf:Metadata mpvtv:ShowProperties mpvtv:IndexPointSeq mpvtv:IndexPoint	SMPTE-style index point.
Cut Point Sequence	nmf:Metadata mpvtv:ShowProperties mpvtv:CutPointSeq	Ordered container for Index Points, defines a start – stop list of index points.
Index Point	nmf:Metadata mpvtv:ShowProperties mpvtv:CutPointSeq mpvtv:IndexPoint	SMPTE-style index point.
Preview	mpv:Rendition, mpv:renditionUsage	Thumbnails, playback video clips, stills, etc.
	mpv:Metadata	Tunnel for well-formed XML, per [MPVCore]

Playlist of TVShow Assets	mpvtv:Playlist(Bag)	Container for playlist assets, references to AssetList of TVShow
Title	nmf:Metadata dc:Properties dc:title	Playlist title (e.g., “Dad’s Videos”).
Description	nmf:Metadata dc:Properties dc:description	Playlist description (further descriptive text for title)
Language	nmf:Metadata dc:Properties dc:language	Title/Description Language, RFC1766, optional territory
PlaylistRef	mpvtv:Playlist mpvtv:PlaylistRef	Reference to another playlist
PlaylistAsset	mpvtv:Playlist mpvtv:PlaylistAsset	Container for a playlist entry
ResumePoint	mpvtv:Playlist mpvtv:PlaylistAsset nmf:Metadata mpvtv:ShowProperties mpvtv:resumePoint	Resume Point, SMPTE style HH:MM:SS:FF, HH:MM:SS;FF, HH:MM:SS;FF.
KeepUntil	mpvtv:Playlist mpvtv:PlaylistAsset nmf:Metadata mpvtv:ShowProperties mpvtv:keepUntil	Date on which this asset may be deleted,

		NULL date if keep forever
LastAccess	mpvtv:Playlist mpvtv:PlaylistAsset nmf:Metadata mpvtv:ShowProperties mpvtv:lastAccess	Date on which this asset was last accessed, NULL date if never accessed
	mpv:Metadata	Tunnel for well-formed XML, per [MPVCore]

Chapter 4: MPV Broadcast Television Video Asset Metadata

Wherever possible, the Broadcast Television Profile utilizes metadata defined by [MPVCore] and [DC]. Where the Broadcast Television Profile can enhance the definition of a TVShow asset, it does so by adding new metadata defined by an **mpvtv:** tag. Thus, a TVShow asset is defined as an **mpv:Video** asset with additional properties. What follows is a description of those properties and their utilization within the scope of this profile.

4.1 Locating a TVShow Asset : *mpv>LastURL*

The base [MPVCore] specification defines **mpv>LastURL** as the last known file system location of a named asset. A particular asset **MUST** define at least one LastURL tag, but **MAY** define more than one if the asset in question has a different file name in different file systems.

EXAMPLE:

```
<mpv:Video mpv:id="ID00000001">
<mpv>LastURL>file:///Local/my%20Video%20Assets/ID00000001.mpg</mpv>LastURL>
<mpv>LastURL mpv:filesystem="NTFS">
  file:///c:/Documents%20and%20Settings/user/Video/my%20Video%20Assets/ID00000001.mpg
</mpv>LastURL>
...
</mpv:Video>
```

4.2 MPV / Dublin Core TVShow Metadata

One of the great benefits of MPV is its extensible structure. As new profiles are added, applications that process the older metadata are still useful, even if they weren't specifically designed to work with the new profile metadata.

In keeping with this, the Broadcast Television Profile makes full use of core metadata as applicable.

These include:

MPVCore Element	Description
Title	Full show title
Description	Episode specific description
Creator	Creator/Owner of metadata
Language	Language used by MPVCore title / description, in [RFC1766] format
Format	Applicable MIME format, typically video/mpeg
writtenBy	Manifest tag containing last writer of manifest
documentIDRef	Manifest reference to document metadata

EXAMPLE:

```
<?xml version="1.0" encoding="UTF-8"?>
<file:Manifest
  xmlns:file="http://ns.osta.org/manifest/1.0/"
  xmlns:file2="http://ns.osta.org/manifest/2.0/"
  xmlns:mpv="http://ns.osta.org/mpv/1.0/"
  xmlns:mpvtv="http://ns.osta.org/mpv/tv/1.0/"
  xmlns:dc="http://ns.osta.org/nmf/1.0/dc/"
  xmlns:nmf="http://ns.osta.org/nmf/1.0/"

  <nmf:Metadata>
    <file:ManifestProperties>
      <file2>AboutManifestMPVDocumentID>DOCID001</file2>AboutManifestMPVDocumentID>
      <file2:WrittenBy>http://www.mycompany.com/myapp/1.05/</file2:WrittenBy>
      <ProfileBag>
        <Profile>http://ns.osta.org/mpv/basic/1.0/</Profile>
        <Profile>http://ns.osta.org/nmf/1.0/dc/</Profile>
        <Profile>http://ns.osta.org/mpv/tv/1.0/</Profile>
      </ProfileBag>
    </file:ManifestProperties>
  </nmf:Metadata>

  ...

  <mpv:AssetList>
    <mpv:Video mpv:id="ID00000001">
      ...
      <nmf:Metadata>
        <dc:Properties>
          <dc>Title>Seinfeld</dc:title>
          <dc:description>Elaine fails a drug test---twice---and loses her job; Jerry talks about his Uncle Leo on "The Tonight Show."</dc:description>
          <dc:creator>TV Guide</dc:creator>
          <dc:language>en-US</dc:language>
          <dc:format>video/mpeg</dc:format>
        </dc:Properties>
        ...
      </nmf:Metadata>
    </mpv:Video>
```

```

...

</mpv:AssetList>
</file:Manifest>

```

4.3 MPV / Dublin Core Qualifier TVShow Metadata

In addition to the Dublin Core metadata defined in [MPVCore], the Dublin Core Metadata Initiative group has defined qualifiers to extend or refine the base set (see [DC-QUAL]) . The Broadcast Television Profile makes use of some of these qualifiers to give further information to a processing application.

These include:

Dublin Core Qualifier	Description
Alternative	Alternative Title, used for a short title if one exists
Abstract	Generic show description

EXAMPLE:

```

<?xml version="1.0" encoding="UTF-8"?>
<file:Manifest
  xmlns:file="http://ns.osta.org/manifest/1.0/"
  xmlns:file2="http://ns.osta.org/manifest/2.0/"
  xmlns:mpv="http://ns.osta.org/mpv/1.0/"
  xmlns:mpvtv="http://ns.osta.org/mpv/tv/1.0/"
  xmlns:dc="http://ns.osta.org/nmf/1.0/dc/"
  xmlns:nmf="http://ns.osta.org/nmf/1.0/"

  <nmf:Metadata>
    <file:ManifestProperties>
      <file2:AboutManifestMPVDocumentID>DOCID001</file2:AboutManifestMPVDocumentID>
      <file2:WrittenBy>http://www.mycompany.com/myapp/1.05/</file2:WrittenBy>
      <ProfileBag>
        <Profile>http://ns.osta.org/mpv/basic/1.0/</Profile>
        <Profile>http://ns.osta.org/nmf/1.0/dc/</Profile>
        <Profile>http://ns.osta.org/mpv/tv/1.0/</Profile>
      </ProfileBag>
    </file:ManifestProperties>
  </nmf:Metadata>

  ...

  <mpv:AssetList>
    <mpv:Video mpv:id="ID00000001">
      ...
      <nmf:Metadata>
        <dc:Properties>
          <dc:title>The Tonight Show With Jay Leno</dc:title>

```

```

    <dc:description>Actors Jennifer Aniston, Courteney Cox Arquette, Lisa Kudrow, Matt
    LeBlanc, Matthew Perry and David Schwimmer reminisce about the past 10 years on the set of
    ``Friends``; the Black Eyed Peas perform.</dc:description>
    <dc:creator>TV Guide</dc:creator>
    <dc:language>en-US</dc:language>
    <dc:format>video/mpeg</dc:format>
  </dc:Properties>

  <!-- Note the additional properties here -->
  <dct:Properties>
    <dct:Alternative>Tonight Show</dct:Alternative>
    <dct:Abstract>Late night talk program.</dct:Abstract>
  </dct:Properties>
  ...
  </nmf:Metadata>
</mpv:Video>

...

</mpv:AssetList>
</file:Manifest>

```

4.4 Broadcast Specification TVShow Metadata

The vast majority of metadata that the Broadcast Television Profile provides cannot be represented by the MPV Core or Dublin Core Qualifiers. These items are defined under a new <mpvtv:ShowProperties> tag.

4.4.1 Basic Broadcast Profile TVShow Data

The very simplest type of show and scheduling information are the ones that are described with a single element.

These include:

Simple Show Properties	Description
episodeTitle	Episode Title, for episodic shows
productionYear	Four digit production year
firstAirDate	Date show first aired, in [ISO8601] Date format
channel	One or two part (major,minor) receiver channel, with non-numeric separator
startTime	Actual start time, in [ISO8601] Date / Time format
duration	Actual duration, in minutes, fractional seconds
expectedStartTime	Scheduled start time, in [ISO8601] Date / Time format
expectedDuration	Scheduled duration, in minutes, fractional seconds

EXAMPLE:

```
<?xml version="1.0" encoding="UTF-8"?>
```

```

<file:Manifest
  xmlns:file="http://ns.osta.org/manifest/1.0/"
  xmlns:file2="http://ns.osta.org/manifest/2.0/"
  xmlns:mpv="http://ns.osta.org/mpv/1.0/"
  xmlns:mpvtv="http://ns.osta.org/mpv/tv/1.0/"
  xmlns:dc="http://ns.osta.org/nmf/1.0/dc/"
  xmlns:nmf="http://ns.osta.org/nmf/1.0/"

  <nmf:Metadata>
    <file:ManifestProperties>
      <file2:AboutManifestMPVDocumentID>DOCID001</file2:AboutManifestMPVDocumentID>
      <file2:WrittenBy>http://www.mycompany.com/myapp/1.05/</file2:WrittenBy>
      <ProfileBag>
        <Profile>http://ns.osta.org/mpv/basic/1.0/</Profile>
        <Profile>http://ns.osta.org/nmf/1.0/dc/</Profile>
        <Profile>http://ns.osta.org/mpv/tv/1.0/</Profile>
      </ProfileBag>
    </file:ManifestProperties>
  </nmf:Metadata>

  ...

  <mpv:AssetList>
    <mpv:Video mpv:id="ID00000001">
      ...
      <nmf:Metadata>
        <dc:Properties>
          <dc:title>Seinfeld</dc:title>
          <dc:description>Elaine fails a drug test---twice---and loses her job; Jerry talks about his Uncle Leo on "The Tonight Show."</dc:description>
          <dc:creator>TV Guide</dc:creator>
          <dc:language>en-US</dc:language>
          <dc:format>video/mpeg</dc:format>
        </dc:Properties>
        <dct:Properties>
          <dct:Abstract>Four neurotic New Yorkers in a show about nothing.</dct:Abstract>
        </dct:Properties>
        <mpvtv:ShowProperties>

          <mpvtv:episodeTitle>The Showerhead</mpvtv:episodeTitle>
          <mpvtv:startTime>2004-03-24T19:01</mpvtv:startTime>
          <mpvtv:duration>28.0 </mpvtv:duration>
          <mpvtv:channel>5-2</mpvtv:channel>
          <mpvtv:expectedStartTime>2004-03-24T19:00</mpvtv:expectedStartTime>
          <mpvtv:expectedDuration>30.0</mpvtv:expectedDuration>
          <mpvtv:productionYear>1998</mpvtv:productionYear>
          <mpvtv:firstAirDate>1998-02-15</mpvtv:firstAirDate>

        </mpvtv:ShowProperties>

        ...
      </nmf:Metadata>
    </mpv:Video>
    ...

    <mpv:Still mpv:id = "TNT-LOGO-ID">
      ...

```

```

</mpv:Still>

</mpv:AssetList>
</file:Manifest>
    
```

Note the differences in the two separate start time / duration fields: though the broadcast schedule represented by the expected values shows the show starting at 7:00pm local time for a duration of 30 minutes, the actual underlying asset was recorded starting one minute late (7:01pm), and ending one minute early (28 minute duration).

4.4.2 Compound Broadcast Profile TVShow Data

The next level of information is only slightly more complicated: the data that is described with a compound element.

4.4.2.1 StationInfo

The Station Info describes the channel the show appears on. The basic encapsulating tag is <mpvtv:StationInfo>, which contains the following items:

mpvtv:stationInfo tags	Description
stationName	Station display name, up to 8 characters
stationFullName	If different from station name
stationLogo	Encapsulates an mpv:StillRef
stationAffiliation	Broadcast station affiliation, e.g., "NBC"

EXAMPLE

```

<mpv:Video>
...
  <mpvtv:stationInfo>
    <mpvtv:stationName>TNT</mpvtv:stationName>
    <mpvtv:stationFullName>Turner Network Television</mpvtv:stationFullName>
    <mpvtv:stationLogo>
      <mpv:StillRef mpv:idref="TNT-LOGO-ID" />
    </mpvtv:stationLogo>
  </mpvtv:stationInfo>
...
</mpv:Video>

<mpv:Still mpv:id = "TNT-LOGO-ID">
...
</mpv:Still>

...
    
```

4.4.2.2 ShowFlags

The <mpvtv:showFlags> tag encapsulates a number of items usually treated as bit flags. They are:

mpvtv:showFlags	Description
cc	Closed Caption
live	Live Broadcast
new	New Airing
repeat	Repeat Airing
letterbox	Letterbox Format
copyInhibit	Copy Protect
encrypted	Content Encrypted

EXAMPLE:

```

<mpv:Video>
...
  <mpvtv:stationInfo>
    <mpvtv:stationName>TNT</mpvtv:stationName>
    <mpvtv:stationFullName>Turner Network Television</mpvtv:stationFullName>
    <mpvtv:stationLogo>
      <mpv:StillRef mpv:idref="TNT-LOGO-ID" />
    </mpvtv:stationLogo>
  </mpvtv:stationInfo>

  <mpvtv:showFlags>
    <mpvtv:cc/>
    <mpvtv:letterbox/>
    <mpvtv:new/>
  </mpvtv:showFlags>

...
</mpv:Video>

<mpv:Still mpv:id = "TNT-LOGO-ID">
...
</mpv:Still>

...

```

4.4.2.3 VideoEncoding

A Broadcast Television Profile device can sometimes make use of knowledge about the physical nature of the recording. These values are:

mpvtv:videoEncoding tag	Description
encodedSource	One of: (standardDef, enhancedDef, highDef)
encodedStandard	Video Standard, see below table for values
encodedMethod	One of: (mpeg-vbr, mpeg-cbr)
encodedBitrate	Integer, in kbps
encodedSize	File size in kB

encodedStandard types	encodedStandard values
NTSC	One of: (NTSC, NTSC-EIA, NTSC-JEIA)
PAL	One of: (PAL, PAL-B, PAL-G, PAL-H, PAL-D, PAL-I,

SECAM	PAL-K, PAL-N, PAL-M) One of: (SECAM, SECAM-B, SECAM-D, SECAM-G, SECAM-H, SECAM-K, SECAM-I, SECAM-N, SECAM-M, SECAM-K1, SECAM-L)
ATSC	One of: (ATSC, ATSC-8VSB, ATSC-QAM64, ATSC-QAM256)
DVB	One of: (DVB, DVB-T, DVB-C, DVB-S, DVB-S2, DVB-H)
DSS	DSS
ISDB	One of: (ISDB, ISDB-T, ISDB-C, ISDB-S)

EXAMPLE:

```

<mpv:Video>
...
  <mpvtv:stationInfo>
    <mpvtv:stationName>TNT</mpvtv:stationName>
    <mpvtv:stationFullName>Turner Network Television</mpvtv:stationFullName>
    <mpvtv:stationLogo>
      <mpv:StillRef mpv:idref="TNT-LOGO-ID" />
    </mpvtv:stationLogo>
  </mpvtv:stationInfo>

  <mpvtv:showFlags>
    <mpvtv:cc/>
    <mpvtv:letterbox/>
    <mpvtv:new/>
  </mpvtv:showFlags>

  <mpvtv:videoEncoding>
    <mpvtv:encodedSource>highDef</mpvtv:encodedSource>
    <mpvtv:encodedStandard>ATSC-8VSB</mpvtv:encodedStandard>
    <mpvtv:encodedMethod>mpeg-vbr</mpvtv:encodedMethod>
    <mpvtv:encodedBitrate>9800</mpvtv:encodedBitrate>
    <mpvtv:encodedSize>10240</mpvtv:encodedSize>
  </mpvtv:videoEncoding>
  ...
</mpv:Video>

<mpv:Still mpv:id = "TNT-LOGO-ID">
...
</mpv:Still>

...

```

4.4.3 Container-based Broadcast Profile TVShow Data

Next are a set of basic TVShow metadata defined within a container tag. Containers are used by XML and the [MPVCore] specification to encapsulate one or more related elements. The [MPVCore] specification also defines containers with an explicit ordering type (see <mpv:Seq>, <mpv:Par>). The Broadcast Television Profile extends these concepts in two ways. The first is the addition of a Bag ordering type, denoting an unordered collection. The

second is the appending of an ordering type to a base type to form a container of that base type. Thus, an unordered collection of <mpvtv:foo> types is defined as <mpvtv:fooBag>. An ordered sequence of <mpv:bar> types is defined by an <mpvtv:barSeq>.

The Broadcast Television Profile defines these new container types:

Container	Description
genreSeq	An ordered list of <mpvtv:genre> types
ratingBag	An unordered list of show ratings
descriptionBag	An unordered list of descriptions
castMemberBag/Seq	An unordered/ordered list of cast members
showIDBag	An unordered list of unique show identifiers

EXAMPLE:

```
<?xml version="1.0" encoding="UTF-8"?>
<file:Manifest
  xmlns:file="http://ns.osta.org/manifest/1.0/"
  xmlns:file2="http://ns.osta.org/manifest/2.0/"
  xmlns:mpv="http://ns.osta.org/mpv/1.0/"
  xmlns:mpvtv="http://ns.osta.org/mpv/tv/1.0/"
  xmlns:dc="http://ns.osta.org/nmf/1.0/dc/"
  xmlns:nmf="http://ns.osta.org/nmf/1.0/"

  <nmf:Metadata>
    <file:ManifestProperties>
      <file2>AboutManifestMPVDocumentID>DOCID001</file2>AboutManifestMPVDocumentID>
      <file2:WrittenBy>http://www.mycompany.com/myapp/1.05/</file2:WrittenBy>
      <ProfileBag>
        <Profile>http://ns.osta.org/mpv/basic/1.0/</Profile>
        <Profile>http://ns.osta.org/nmf/1.0/dc/</Profile>
        <Profile>http://ns.osta.org/mpv/tv/1.0/</Profile>
      </ProfileBag>
    </file:ManifestProperties>
  </nmf:Metadata>
  ...

  <mpv:AssetList>
    <mpv:Video mpv:id="ID00000001">
      ...
      <nmf:Metadata>
        <dc:Properties>
          <dc>Title>Seinfeld</dc:title>
          <dc:description>Elaine fails a drug test---twice---and loses her job; Jerry talks about
his Uncle Leo on "The Tonight Show."</dc:description>
          <dc:creator>TV Guide</dc:creator>
          <dc:language>en-US</dc:language>
          <dc:format>video/mpeg</dc:format>
        </dc:Properties>
        <dct:Properties>
          <dct:Abstract>Four neurotic New Yorkers in a show about nothing.</dct:Abstract>
        </dct:Properties>
        <mpvtv:ShowProperties>
          ...

          <mpvtv:genreSeq mpvtv:vocabulary="urn:osta-org:mpv:tv:genre:tvguide">
```



```

    <mpvtv:genre>Comedy</mpvtv:genre>
    <mpvtv:genre>Sitcom</mpvtv:genre>
  </mpvtv:genreSeq>

  ...

  </mpvtv:ShowProperties>

  ...
  </nmf:Metadata>
</mpv:Video>
...

</mpv:AssetList>
</file:Manifest>

```

In this example, note that the show has two genres contained within an <mpvtv:genreSeq> element. The Comedy genre should be considered the primary one, followed by the Sitcom genre.

4.4.3.1 Genres

A genre is a classification or categorization of a show based on its content, be it comedy, news, sports or some other classification.

A genre may appear alone, or within a genreSeq element as outlined above. GenreSeqs SHOULD be specified with a vocabulary attribute (in URN form) so that a processing application knows the source of categorization.

Genre vocabularies defined by this specification follow:

Genre Vocabulary	mpvtv:ratingAuthority
TV Guide, Inc.	urn:osta-org:mpv:tv:genre:tvguide

4.4.3.2 Ratings

Ratings are assigned by a variety of authorities and are meant to help a viewer select suitable content. Because each individual authority may assign a different rating to the same show, ratings elements may appear alone or within a Bag container.

Ratings authorities are specified as attribute types and defined using a URN syntax. The ratings authorities defined by this specification follow:

Rating Authority	mpvtv:ratingAuthority
MPAA (Motion Picture Association of America)	urn:osta-org:mpv:tv:rating:mpaa
US TV (V-Chip)	urn:osta-org:mpv:tv:rating:en-US
Canadian TV English (V-Chip)	urn:osta-org:mpv:tv:rating:en-CA
Canadian TV French (V-Chip)	urn:osta-org:mpv:tv:rating:fr-CA
Star Rating	urn:osta-org:mpv:tv:rating:5star

Defined within an <mpv:rating> tag are two element types: a rating value followed by one or more ratings reasons.

Rating Element	Description
ratingValue	Actual rating
ratingReason	Rating modifier

Ratings values for defined elements:

Rating Authority	mpvtv:ratingValue
MPAA (Motion Picture Association of America) US TV (V-Chip)	One of: (G, PG, PG-13, R, NC-17, NR, X) One of: (TV-Y, TV-Y7, TV-G, TV-PG, TV-14, TV-MA)
Canadian TV English (V-Chip)	One of: (E, C, C8+, G, PG, 14+, 18+)
Canadian TV French (V-Chip)	One of: (E, G, 8 ans+, 13 ans+, 16 ans+, 18 ans+)
Star Rating	Fractional value, by .5, up to 5 (0, 0.5, 1.0 ... 4.5, 5.0)

And the matching ratingReasons:

Rating Authority	mpvtv:ratingReason
MPAA (Motion Picture Association of America)	One of: (Adult Situations, Graphic Language, Language, Brief Nudity, Strong Sexual Content, Graphic Violence, Mild Violence, Violence, Rape)
US TV (V-Chip)	One of: (Fantasy Violence, Violence, Sexual Situations, Language, Suggestive Dialogue)
Canadian TV English (V-Chip)	None defined.
Canadian TV French (V-Chip)	None defined.
Star Rating	None defined.

4.4.3.3 Descriptions

The Broadcast Television Profile provides for a base show description defined by the [MPVCore] dc:description tag. While this is suitable for most applications, more advanced applications may wish to process more than one description type for a variety of reasons:

- A shorter or longer description than the base definition is available
- An alternate language encoding is available
- A description from another creator source is available

For each of these cases, an alternate description is provided via the <mpvtv:description> tag. This tag may appear alone or within an <mpvtv:descriptionBag> container.

A creator attribute defines the content origin and is specified by URN syntax.

Description Content Creator	mpvtv:creator type attribute
Cinebook	urn:osta-org:mpv:tv:desc:cinebook
TVGuide Editorial Description	urn:osta-org:mpv:tv:desc:tvguide_editorial
TVGuide Short Description	urn:osta-org:mpv:tv:desc:tvguide_short

An alternate description has two [MPVCore] properties:

MPVCore Element	Description
Description	Alternate description
Language	Language used by MPVCore description, in [RFC1766] format

4.4.3.4 Cast Members

The usual television show has many cast members, the Broadcast Television Profile has the capability to describe them all.

Cast members are defined using the <mpvtv:castMember> tag. As with descriptions above, a cast member may appear singly or within an <mpvtv:castMemberBag>.

With the <mpvtv:castMember> tag is an attribute that defines the cast member type:

Cast Member Type	mpvtv:castType
Actor	Actor
Director	Director
Writer	Writer
Guest Star	guestStar

Within an <mpvtv:castMember> tag are two subordinate container tags:

Cast Member Element	Description
castMemberName	Cast Member Name
castMemberRole	For castType Actor only

Within a cast member element are two tags that define name. Surname includes last name and optional suffix (e.g., Jr., Sr. III). A cast member with only one name (Madonna) will have surname only. Given Name includes first name and optional middle name or middle initial.

Cast Member Name Element	Description
castMemberSurname	Surname (last name)
castMemberGivenName	Given name (first name)

4.4.3.5 ShowIDs

Independent of the various computed ID fields provided by [MPVCore], a number of different organizations have expressed interest in providing a unique showID for television content. Rather than try and overload an existing ID, this profile defines a showID tag. The tag has a defined attribute (mpvtv:idAuthority) that a processing application can use to identify the registrar of the value.

ShowIDs may appear in a showIDBag if there is more than one defined.

Defined attribute values include:

Registering Authority	mpvtv:idAuthority
TVGuide	urn:osta-org:mpv:tv:id:tvguide
V-ISAN	urn:osta-org:mpv:tv:id:v-isan

4.4.3.6 Summary Example

In order to tie the definitions of this section together, here is an annotated example of their use.

EXAMPLE:

```

<?xml version="1.0" encoding="UTF-8"?>
<file:Manifest
  xmlns:file="http://ns.osta.org/manifest/1.0/"
  xmlns:file2="http://ns.osta.org/manifest/2.0/"
  xmlns:mpv="http://ns.osta.org/mpv/1.0/"
  xmlns:mpvtv="http://ns.osta.org/mpv/tv/1.0/"
  xmlns:dc="http://ns.osta.org/nmf/1.0/dc/"
  xmlns:nmf="http://ns.osta.org/nmf/1.0/"

  <nmf:Metadata>
    <file:ManifestProperties>
      <file2>AboutManifestMPVDocumentID>DOCID001</file2>AboutManifestMPVDocumentID>
      <file2:WrittenBy>http://www.mycompany.com/myapp/1.05/</file2:WrittenBy>
      <ProfileBag>
        <Profile>http://ns.osta.org/mpv/basic/1.0/</Profile>
        <Profile>http://ns.osta.org/nmf/1.0/dc/</Profile>
        <Profile>http://ns.osta.org/mpv/tv/1.0/</Profile>
      </ProfileBag>
    </file:ManifestProperties>
  </nmf:Metadata>

  ...

  <mpv:AssetList>
    <mpv:Video mpv:id="ID00000001">
      ...
      <nmf:Metadata>
        <dc:Properties>
          <dc>Title>Seinfeld</dc:title>
          <dc:description>Elaine fails a drug test---twice---and loses her job; Jerry talks about
his Uncle Leo on "The Tonight Show."</dc:description>
          <dc:creator>TV Guide</dc:creator>
          <dc:language>en-US</dc:language>
          <dc:format>video/mpeg</dc:format>
        </dc:Properties>
        <dct:Properties>
          <dct:Abstract>Four neurotic New Yorkers in a show about nothing.</dct:Abstract>
        </dct:Properties>
        <mpvtv:ShowProperties>

          <!-- Here is a genreSeq that defines a primary genre of Comedy, secondary of Sitcom -->
          <mpvtv:genreSeq>
            <mpvtv:genre>Comedy</mpvtv:genre>
            <mpvtv:genre>Sitcom</mpvtv:genre>
          </mpvtv:genreSeq>

          <!-- As for ratings, let's have one US, one Canadian -->
          <mpvtv:ratingBag>
            <mpvtv:rating mpvtv:ratingAuthority="urn:osta-org:mpv:tv:rating:en-US">
              <mpvtv:ratingValue>PG-13</mpvtv:ratingValue>
              <mpvtv:ratingReason>Language</mpvtv:ratingReason>
            </mpvtv:rating>

            <mpvtv:rating mpvtv:ratingAuthority="urn:osta-org:mpv:tv:rating:en-CA">
              <mpvtv:ratingValue>14+</mpvtv:ratingValue>

```

```

    </mpvtv:rating>
  </mpvtv:ratingBag>

  <!-- An alternate short description -->
  <mpvtv:description mpvtv:creator="urn:osta-org:mpv:tv:desc:tvguide_short">
    <nmf:Metadata>
      <dc:Properties>
        <dc:description>Elaine fails drug test, Jerry insults Uncle Leo</dc:description>
        <dc:language>en-US</dc:language>
      </dc:Properties>
    </nmf:Metadata>
  </mpvtv:description>

  <!-- Cast members are next -->
  <mpvtv:castMemberBag>
    <mpvtv:castMember mpvtv:castType="urn:osta-org:mpv:tv:cast:actor">
      <mpvtv:castMemberName>
        <mpvtv:castMemberSurname>Seinfeld</mpvtv:castMemberSurname>
        <mpvtv:castMemberGivenName>Jerry</mpvtv:castMemberGivenName>
      </mpvtv:castMemberName>
      <mpvtv:castMemberRole>
        <mpvtv:castMemberGivenName>Jerry</mpvtv:castMemberGivenName>
      </mpvtv:castMemberRole>
    </mpvtv:castMember>

    <mpvtv:castMember mpvtv:castType="urn:osta-org:mpv:tv:cast:actor">
      <mpvtv:castMemberName>
        <mpvtv:castMemberSurname>Louis-Dreyfus</mpvtv:castMemberSurname>
        <mpvtv:castMemberGivenName>Julia</mpvtv:castMemberGivenName>
      </mpvtv:castMemberName>
      <mpvtv:castMemberRole>
        <mpvtv:castMemberGivenName>Elaine</mpvtv:castMemberGivenName>
      </mpvtv:castMemberRole>
    </mpvtv:castMember>

    <mpvtv:castMember mpvtv:castType="urn:osta-org:mpv:tv:cast:actor">
      <mpvtv:castMemberName>
        <mpvtv:castMemberSurname>Richards </mpvtv:castMemberSurname>
        <mpvtv:castMemberGivenName>Michael</mpvtv:castMemberGivenName>
      </mpvtv:castMemberName>
      <mpvtv:castMemberRole>
        <mpvtv:castMemberGivenName>Kramer</mpvtv:castMemberGivenName>
      </mpvtv:castMemberRole>
    </mpvtv:castMember>

    <mpvtv:castMember mpvtv:castType="urn:osta-org:mpv:tv:cast:actor">
      <mpvtv:castMemberName>
        <mpvtv:castMemberSurname>Alexander</mpvtv:castMemberSurname>
        <mpvtv:castMemberGivenName>Jason</mpvtv:castMemberGivenName>
      </mpvtv:castMemberName>
      <mpvtv:castMemberRole>
        <mpvtv:castMemberGivenName>George</mpvtv:castMemberGivenName>
      </mpvtv:castMemberRole>
    </mpvtv:castMember>
  </mpvtv:castMemberBag>

```

```

<!-- A single showID -->
<mpvtv:showID mpvtv:idAuthority="urn:osta-org:mpv:tv:id:tvguide">
84990999090229</mpvtv:showID>

...

</mpvtv:ShowProperties>

...
</nmf:Metadata>
</mpv:Video>
...

</mpv:AssetList>
</file:Manifest>

```

4.5 MPV Core Metadata Extended by Broadcast TV Profile

4.5.1 Audio

In the usual case for a recorded television program, the audio is embedded with the video content. However, there may be additional audio streams associated with a show: a device may choose to record the SAP (Secondary Audio Program) stream, or a DVD may have multiple language streams available.

The Broadcast Television Profile solves both problems by extending the concepts presented in [MPVCore].

4.5.1.1 Embedded Audio

Since embedded audio does not reference a separate asset, the Broadcast Television Profile extends the base <mpv:Audio> asset to provide a new <mpvtv:Audio> tag.

Within the <mpvtv:Audio> tag, some Dublin Core metadata is used to describe the audio asset:

Dublin Core <mpvtv:Audio> tags	Description
description	Descriptive presentation text for stream
language	Audio language used, in [RFC1766] format
format	Applicable MIME format, typically audio/mpeg

Items that cannot be described by Dublin Core are specified using Broadcast Television Profile specific tags.

Broadcast Television Profile <mpvtv:Audio> tags	Description
audioType	One of: normal, hearing impaired
audioEncoding	One of: mono, stereo, dolby surround, dolby digital 5.1

4.5.1.2 External Audio

An audio stream external to the video stream is a concept well handled by [MPVCore]. The Broadcast Television Profile allows the inclusion of all specific embedded audio types outlined above, plus the addition of the file size of the described audio stream. Note that external audio uses the core <mpv:Audio> tag for greater compatibility with existing profiles.

Broadcast Television Profile <mpv:Audio> tags	Description
audioSize	File size, in KB, of audio file.

4.5.1.3 Summary Example

In order to tie the definitions of this section together, here is an annotated example of their use.

EXAMPLE:

```

<mpv:AssetList>

  <!-- definition of a base video asset -->
  <mpv:Video mpv:id="ID00000001">
    ...
    <nmf:Metadata>
      <dc:Properties>
        ...
      </dc:Properties>
      <mpvtv:ShowProperties>
        ...

      <!-- embedded audio -->
      <mpvtv:Audio>
        <nmf:Metadata>
          <dc:properties>
            <dc:description>Main Audio</dc:description>
            <dc:language>en-US</dc:language>
            <dc:format>audio/mpeg</dc:format>
          </dc:properties>
          <mpvtv:ShowProperties>
            <mpvtv:audioType>normal</mpvtv:audioType>
            <mpvtv:audioEncoding>stereo</mpvtv:audioEncoding>
          </mpvtv:ShowProperties>
        </nmf:Metadata>
      </mpvtv:Audio>

      <!-- Reference to external audio stream(s) -->
      <mpv:AudioRef mpv:refId="AUD00000001 " />

    </mpvtv:ShowProperties>
  </nmf:Metadata>
</mpv:Video>

<!-- Audio associated with video, referenced by an AudioRef -->
<mpv:Audio mpv:id="AUD00000001">
  <mpv:lastURL mpv:filesystem="UDF">ShowAudio.mpg</mpv:lastURL>
  <mpv:ContentID>urn:osta-org:dsig:md5:all:8735353445934AC</mpv:ContentID>
  <nmf:Metadata>
    <dc:properties>
      <dc:description>Spanish SAP Audio</dc:description>
      <dc:language>es-US</dc:language>
      <dc:format>audio/mpeg</dc:format>
    </dc:properties>
  </mpvtv:ShowProperties>

```

```

        <mpvtv:audioType>normal</mpvtv:audioType>
        <mpvtv:audioEncoding>stereo</mpvtv:audioEncoding>
        <mpvtv:audioSize>146</mpvtv:audioSize>
    </mpvtv:ShowProperties>
</nmf:Metadata>
</mpv:Audio>

</file:Manifest>
    
```

4.5.2 Renditions of a TVShow Asset

The file(s) specified in the primary TVShow asset should generally be the complete program. There are a number of cases where an alternate rendition may be specified to enhance either the user navigation or customer viewing experience.

Such cases might include:

- 1) A thumbnail still image displayed during playlist navigation
- 2) A subsampled version viewable on a specific device

Renditions of a TVShow asset are specified using the <mpv:Rendition> tag and appropriate mpv:renditionUsage attribute values. The [MPVCore] specification provides the normative reference for acceptable values, alternatives may be specified via URN syntax.

Acceptable renditions include:

RenditionUsage	Description
thumbnail	Thumbnail preview of video
subsampled	Subsampled resolution of base, new mpvtv:VideoEncoding tag expected

EXAMPLE

```

<mpv:Video>
...
  <mpv:Rendition mpv:renditionUsage="thumbnail">
    <mpv:StillRef mpv:idref="THUMBNAIL-STILL-ID"/>
  </mpv:Rendition>
...
</mpv:Video>

<mpv:Still mpv:id = "THUMBNAIL-STILL-ID">
...
</mpv:Still>

...
    
```


4.5.3 Related References to a TVShow Asset

As noted above, the file(s) specified in the primary TVShow asset should generally be the complete program. Just as a rendition, there are cases where a related asset might provide an alternate viewing experience, such as a commercial-free view of a program.

These relations are specified using the <mpv:Related> tag and appropriate mpv:relationship attribute value. In the Broadcast Television Profile, these relationships should be specified with URN syntax.

Relationships understood by this profile include:

Relationship	Description
urn:osta-org:mpv:tv:relation:noCommercial	References a noCommercial subset of the containing base Video asset.
urn:osta-org:mpv:tv:relation:noCommercialOf	Refers back to the base video the containing Video asset is a noCommercial subset of.
urn:osta-org:mpv:tv:relation:preview	References a preview subset of the containing base Video asset.
urn:osta-org:mpv:tv:relation:previewOf	Refers back to the base video the containing Video asset is a preview subset of.
urn:osta-org:mpv:tv:relation:userEditOf	Refers back to the base video the containing Video asset is a user edited subset of.

4.5.3.1 IndexPoints

There are many situations in which it is useful to describe subsets of video. For years, VCRs have had the capability to mark “index points” in video so that the user can perform a seek to an index point. In the Related references above, it is easy to see that a “noCommercialOf” version of a particular video can be described as a series of cut points from the original.

The Broadcast Television Profile supports the use of an IndexPoint tag within either an IndexPointSeq or CutPointSeq container:

Metadata	Description
IndexPoint	SMPTE-style (HH:MM:SS:FF) description of a particular video frame
CutPointSeq	Ordered container of start – stop timestamp pairs denoting included video
IndexPointSeq	Ordered container of IndexPoints.

A Related video containing a CutPointSeq may or may not have a physical file associated with it – it may simply be a description of what to cut from the original video file, or it can be a physical manifestation of the result of the cut process. By convention, a Video object with a LastURL tag refers to a physical file.

The following example illustrates an encoding of a noCommercialOf version of a video using CutPointSeqs and IndexPoints.

EXAMPLE:

```
<!-- Here is an original video asset, which has a related noCommercial version -->
<mpv:Video mpv:id="VID0001">
...

```

```

    <mpv:Related mpv:relationship="urn:osta-org:mpv:tv:relation:noCommercial">
      <mpv:VideoRef mpv:idref = "VID00010001"/>
    </mpv:Related>
    ...
  </mpv:Video>

  <!-- Related video, defines a commercial-free index point set of base video -->
  <!-- Note that there is no new instance of the underlying show in this case -->
  <mpv:Video mpv:id="VID00010001">
    <nmf:Metadata>
      <dc:Properties>
        <dc:title>My No Commercial Version</dc:title>
      </dc:Properties>

      <mpvtv:ShowProperties>
        <!-- CutPointSeq providing an ordered sequence of start-end pairs -->
        <!-- This particular one cuts out minutes 13-16 of the original -->
        <mpvtv:CutPointSeq>
          <mpvtv:IndexPoint>00:00:00:00</mpvtv:IndexPoint>
          <mpvtv:IndexPoint>00:12:12:30</mpvtv:IndexPoint>
          <mpvtv:IndexPoint>00:17:00:00</mpvtv:IndexPoint>
          <mpvtv:IndexPoint>00:30:00:00</mpvtv:IndexPoint>
        </mpvtv:CutPointSeq>
      </mpvtv:ShowProperties>
    </nmf:Metadata>

    <!-- This video asset IS A commercial free version of the underlying video, that
           HAS A preview loop of its own -->
    <mpv:Related mpv:relationship="urn:osta-org:mpv:tv:relation:noCommercialOf">
      <mpv:VideoRef mpv:idRef="VID0001"/>
    </mpv:Related>

    <mpv:Related mpv:relationship="urn:osta-org:mpv:tv:relation:preview">
      <mpv:VideoRef mpv:idRef="VID0001002"/>
    </mpv:Related>

  </mpv:Video>

```

4.6 OSTA MD5 Digital Signature Identification

One of the three core concepts defined by MPV is content identification.

There two ways that television shows are identified in the Broadcast Television Profile. One is the method discussed above, where unique showIDs are defined by known registered authorities. The other is the method inherited from [MPVCore], which defines IDs algorithmically computed at some point in their life (the mpv:ContentID / mpv:DocumentID / mpv:InstanceID).

Every asset can have zero or more mpv:ContentID / mpv:DocumentID / mpv:InstanceID elements which contain strongly-typed identifiers. [MPVCore] defines a basic identification scheme based on the MD5 algorithm to provide a statistically unique identifier. Other identification schemes may be defined by OSTA or any other organization.

The following informative discussion is derived from [MPVCore], which provides the normative reference. OSTA-defined digital signatures take the form

urn:osta-org:mpv:dsig:<algorithm>:<params>:<value>

Example: "urn:osta-org:mpv:dsig:md5:all:EF886AEFA3B340da971BAF09B17DBC122"

Three MD5-hash-based digital signatures that are also useful are:

urn:osta-org:mpv:dsig:md5:all:<value>

Every byte in the entire file is processed.

urn:osta-org:mpv:dsig:md5:head:<byte count>:<value>

Only the <byte count> integer number of bytes from the start of the file is processed. This is attractive to robustly refer to very large files or to files that are frequently edited or appended and for which the head can generate an approximately unique signature. If unspecified, the default byte count is 8192. Example:

"urn:osta-org:mpv:dsig:md5:head:30000:EF886AEFA3B340da971BAF09B17DBC122"

urn:osta-org:mpv:dsig:md5:tail:<byte count>:<value>

Only the <byte count> integer number of bytes from the end of the file is processed. This is attractive to quickly detect changes in files that are frequently edited or appended. If unspecified, the default byte count is 8192. Example: "urn:osta-

org:mpv:dsig:md5:tail:30000:EF886AEFA3B340da971BAF09B17DBC122"

In addition, a MD5 signature for the body of specific file types may be defined.

Chapter 5: MPV Broadcast Television Playlists

5.1 MPV Broadcast Television Playlists

The MPV Broadcast Television Profile Playlist is tailored to the presentation and navigation of underlying audio/video assets in a video recording system.

Assets representing TV Shows may be spread across one or more recording media in a system in a variety of ways, but a system may want to present a specific ordering to a user, independent of where the item is stored. For example, a system may wish to present available shows, sometimes sorted by name or by date recorded.

This means that it isn't possible to specify content location and playback order simply by requiring specific directory and filenames. Instead, MPV provides an approach that allows the content to be located anywhere and with any name.

A Playlist (<mpvtv:Playlist>) contains its own metadata such as title, description, and language and MAY contain zero or more PlaylistAssets (<mpvtv:PlaylistAsset>). A PlaylistAsset MUST contain a reference to an underlying media asset, and SHOULD contain metadata describing the playback state of that asset (such as current resume point, date last accessed, etc.).

Playlists MAY reference other Playlists, and those referenced playlists may be stored either locally (in the same filesystem) or remotely (in a different filesystem). A given MPV file MAY contain more than one Playlist, and if so, all Playlists are contained within a PlaylistBag. By convention, if multiple Playlists are encountered, then the first Playlist in the Bag is considered the default Playlist. A Playlist reference may be used by a rendering device as a form of display hierarchy akin to a container folder in a filesystem, if it so chooses. Otherwise, the device is under no obligation to display PlaylistAssets in the ordering provided, it MAY be treated as one suggested ordering.

The basic structure of a playlist uses the <mpvtv:Playlist> element, which may contain metadata, playlist assets, and references to other playlists.

```

...
<mpvtv:PlaylistBag>
  <mpvtv:Playlist>
    <nmf:Metadata> ... </nmf:Metadata>
    <mpv:Metadata> ... </mpv:Metadata>
    <mpvtv:PlaylistRef> ... </mpvtv:PlaylistRef>
    <mpvtv:PlaylistRef> ... </mpvtv:PlaylistRef>
    <mpvtv:PlaylistAsset> ... </mpvtv:PlaylistAsset>
    <mpvtv:PlaylistAsset> ... </mpvtv:PlaylistAsset>

```

```

    <mpvtv:PlaylistAsset> ... </mpvtv:PlaylistAsset>
  </mpvtv:Playlist>
<mpvtv:Playlist>
  <nmf:Metadata> ... </nmf:Metadata>
  <mpv:Metadata> ... </mpv:Metadata>
  <mpvtv:PlaylistRef> ... </mpvtv:PlaylistRef>
  <mpvtv:PlaylistAsset> ... </mpvtv:PlaylistAsset>
</mpvtv:Playlist>
</mpvtv:PlaylistBag>
...

```

5.1.1 MPV / Dublin Core Element Usage

Metadata placed in the mpvtv:Playlist element has the entire Playlist as a scope. These tags include:

[MPVCore] Element	Description
title	Playlist title
description	Optional playlist description
language	Language used by MPVCore title / description, in [RFC1766] format

5.1.2 Broadcast Specification Metadata

Within a Playlist element, the following tags are available:

Broadcast Specification Element	Description
PlaylistAsset	Playlist Asset: references an underlying video asset.
PlaylistRef	Reference a playlist from another source

5.1.3 PlaylistAsset Metadata

PlaylistAssets have the following properties:

PlaylistAsset Element	Description
resumePoint	Timestamp, in SMPTE-type format, denoting where playback will resume from
lastAccess	Last access time, in [ISO8601] Date / Time format
keepUntil	Keep until time, in [ISO8601] Date / Time format, after which the underlying video asset MAY be deleted.

5.1.4 Summary Example

In order to tie the definitions of this section together, here is an annotated example of their use.

EXAMPLE:

```

...
<!-- Playlist Bags contain one or more distinct Playlists, which may link other playlists -->
<mpvtv:PlaylistBag>

```

```

<mpvtv:Playlist mpvtv:id="PLAYLIST0001">
<mpv:InstanceID>urn:osta-org:dsig:md5:all:043449CA9577688</mpv:InstanceID>
<nmf:Metadata>
  <!-- General playlist specific properties -->
  <dc:Properties>
    <dc:title>Seinfeld Episodes</dc:title>
    <dc:description>Recorded September - October 1999</dc:description>
    <dc:language>en-US</dc:language>
  </dc:Properties>
</nmf:Metadata>

<!-- Playlist assets refer to video assets in asset list -->
<!-- First Asset -->
<mpvtv:PlaylistAsset>
<nmf:Metadata>
  <mpvtv:ShowProperties>
    <mpvtv:resumePoint>00:00:00:00</mpvtv:resumePoint>
    <mpvtv:lastAccess>1999-10-07T19:00 </mpvtv:lastAccess>
    <mpvtv:keepUntil>2030-12-31</mpvtv:keepUntil>
  </mpvtv:ShowProperties>
</nmf:Metadata>

  <!-- Reference to video asset itself (Original video)-->
  <mpv:VideoRef mpv:idRef="SEINFELD100799"/>
</mpvtv:PlaylistAsset>

<!-- Second Asset -->
<mpvtv:PlaylistAsset>
<nmf:Metadata>
  <mpvtv:ShowProperties>
    <mpvtv:resumePoint>00:15:00:00 </mpvtv:resumePoint>
    <mpvtv:lastAccess>1999-10-14T19:00 </mpvtv:lastAccess>
    <mpvtv:keepUntil>2030-12-31</mpvtv:keepUntil>
  </mpvtv:ShowProperties>
</nmf:Metadata>

  <!-- Reference to video asset itself -->
  <!-- Follow the reference to find out if there is a preview or thumbnail available -->
  <mpv:VideoRef mpv:idRef="SEINFELD101499"/>
</mpvtv:PlaylistAsset>

  <!-- Reference to a completely different playlist -->
  <mpvtv:PlaylistRef mpvtv:idRef="PLAYLIST0002"/>
</mpvtv:Playlist>

<!-- And here's a second Playlist, which may reference the same or different assets -->
<mpvtv:Playlist mpvtv:id="PLAYLIST0002">
<mpv:InstanceID> urn:osta-org:dsig:md5:all:324204345649CA95776</mpv:InstanceID>
<nmf:Metadata>
  <!-- General playlist specific properties -->
  <dc:Properties>
    <dc:title>Seinfeld Episodes </dc:title>
    <dc:description>Recorded November - December 1999</dc:description>
    <dc:language>en-US</dc:language>
  </dc:Properties>
</nmf:Metadata>

```

```
<!-- Playlist assets refer to video assets in asset list -->
<!-- First Asset -->
<mpvtv:PlaylistAsset>
  <nmf:Metadata>
    <mpvtv:ShowProperties>
      <mpvtv:resumePoint>00:00:00:00 </mpvtv:resumePoint>
      <mpvtv:lastAccess>1999-11-14T19:00 </mpvtv:lastAccess>
      <mpvtv:keepUntil>2030-12-31</mpvtv:keepUntil>
    </mpvtv:ShowProperties>
  </nmf:Metadata>

  <!-- Reference to video asset itself (commercial free version)-->
  <mpv:VideoRef mpv:idRef="SEINFELD111499" />
</mpvtv:PlaylistAsset>

</mpvtv:Playlist>
</mpvtv:PlaylistBag>
```

...

Chapter 6: Broadcast Television Profile Manifests

The [MPVCore] specification establishes the structure and nomenclature of MPV files and assets. The MPV file is an XML document called an MPV Manifest: thus, the outer-most element of a MPV file is <file:Manifest>.

One very useful capability that [MPVCore] provides is to link manifests to one another. The <mpv:ManifestLink> element creates a link to another MPV file. In this manner, just as with the World Wide Web, an endless chain of linked MPV files can be created.

Typically, when applied to a removable storage media like a CD, DVD, or memory card, all the links will be self-contained within the media. In this case, typically a file using the distinguished filename “index.tvvm” will contain a top-level list of playlists.

6.1 Best Practices for Generating Manifest Data

There are any number of ways for applications and devices to produce and update AssetLists and Playlists. They may be the result of extensive manual user interaction or by an automatic process that occurs when the underlying asset is recorded, played back, or discarded.

6.1.1 Top-Level Playlist

The simplest best practice is to provide a top-level playlist in the same Manifest file that defines the underlying assets, such as an index.tvvm. This gives the playback application or device the best ability to find and present contents intended by the creator application. All references should be self contained within the media containing the Manifest, especially in the case of removable media. In the case of a hybrid removable media system, such as a combo HD/DVD PVR device, each device SHOULD have their own Manifest file(s) listing their own AssetList / Playlist data. Manifest reference links are supported, but should reference manifests on the same physical device.

An application or device MAY keep AssetLists and Playlists in separate files. This approach is efficient and scalable because it separates the metadata updated by recording from the metadata updated by playback.

Chapter 7: MPV Broadcast Television Profile Extensions to MPV Core Specification

7.1 Broadcast Television Manifest File Types & Extensions

For systems in which file type is carried by the file name extension, such as Microsoft Windows and Unix, the TV Show Manifest file will utilize an extension. The MPV Broadcast Television Profile defines two extensions a manifest may carry.

.tvm

This extension identifies a file to be a **TV**show **M**anifest (TVM).

Usage is case insensitive. This extension may be registered by an application to provide default and alternate processors of TV Show Manifests.

.xml

This extension identifies a file as containing XML content. Usage is case insensitive. A TVShow Manifest should only use this extension if it expects to be processed by a general-purpose XML processor such as Microsoft Internet Explorer. It is recommended that the manifest include an XML processing instruction specifying a stylesheet to use for presentation.

This extension may be registered by an application to provide general purpose XML content processing. An application should register this extension with care, as many types of content may carry the .xml extension and an application should do its best to handle this content in a general fashion.

For example, Microsoft Internet Explorer 5.5 and above registers this extension; when it processes the file, it looks for a stylesheet processing instruction. IE renders the results of applying the stylesheet to the XML content. This separation of content and presentation allows IE to be a general purpose XML processing engine and suitable for handling the .xml extension.

Some applications examine leading characters of a file in an attempt to determine its file type. No byte sequences can be counted on to always be present, but generally all XML documents in the UTF-8 charsets begin with hexadecimal 3C 3F 78 6D 6C, ("<?xml"). While this will identify the document as an XML document, it does NOT identify it as an TVShow Manifest. This requires parsing the document to locate the outer element defined by the manifest schema.

7.2 TVShow Manifest MIME Media Type

MIME media types are widely used in internet applications to indicate the type of a file or content in a manner external of the file and independent of the name of the file or any information embedded in the file [MIME-2]. IANA maintains a registry of MIME media types and the set of MIME media types IANA thinks is registered at any time can be found at [MIMETYPES-REG].

The MIME media types are RECOMMENDED for use in a Broadcast TV Manifest are:

video/mpeg

This MIME type identifies the content as being mpeg-based video.

application/vnd.osta-org.tvm+xml

This MIME media type identifies the content to be a TVShow Manifest. Usage is case sensitive. This media type may be registered with internet browsers by an application to provide the default processor of a TVShow Manifest.

application/xml

This MIME media type identifies the content as containing XML content. Usage is case sensitive. A TVShow Manifest should only use this MIME type if it expects to be processed by a general-purpose XML processor such as Microsoft Internet Explorer. It is recommended that the manifest include an XML processing instruction specifying a stylesheet to use for presentation.

This MIME media type may be registered by an application to provide general purpose XML content processing. An application should register this media type with care, as many types of content may carry the application/xml media type and an application should do its best to handle this content in a general fashion.

Other MIME media types may be used as applicable to the content.

7.3 Choosing Which File Type and MIME Media Type to Use

For products authoring TVShow Manifests, the choice of file extension and MIME media type is important. The product should consider the contexts in which it expects the manifest to be used. The primary decision factor is whether the product expects the manifest to be used in an environment that is explicitly MPV-aware or one that is not.

A MPV-aware environment will have the **.tvm** file extension and **application/vnd.osta-org.tvm+xml** media type registered to an application. A MPV-unaware environment will not.

Generally speaking, it is preferable to use a TVShow Manifest in an MPV-aware environment because the MPV-aware application is better able to utilize fully the MPV capabilities. In particular, an MPV-aware environment will likely handle better the situation in which the default lastURL reference is invalid; it should use other available lastURL values or the identifiers available on an asset to fixup the lastURL value.

7.4 Finding a TVShow Manifest File

The TVShow Manifest is the essential document to be managed and manipulated for collections of television content. MPV collections define a structured association of assets and provide access to metadata about those assets.

When searching a file system for a TVShow Manifest, they can be located by name or by extension. When requested by name, the manifest is either found or not found. If not found, the algorithm defined elsewhere for lastURL fixup should be applied.

The MPV Broadcast Television Profile defines the following algorithm that describes how to locate an MPV manifest when the name is not known.

```

If dealing with a removable storage unit, e.g. an optical disc inserted, the starting
current working directory is the root directory.

If dealing with a user's personal computer "login" account, there may be a set of
directories to be considered in sequence that will lead to the "root" TVShow Manifest for
the account. Best Practices for which directories to consider are defined elsewhere.

If browsing a filesystem, the current working directory is decided by the application
conducting the search.

The scan algorithm to find a TVShow Manifest from a given current working directory is:

    In the current working directory, look for a file with one of the following
    case-insensitive names according to the order given.
        INDEX.TVM
        INDEXTVM.XML
        PLAY.TVM
        PLAYTVM.XML
        <any name>.TVM, in an undefined order when more than one is present
        INDEX.PVM
        INDEXPVM.XML
        <any name>.PVM, in an undefined order when more than one is present

    If no matching file is found, the child directories of the current directory are
    scanned in an alphabetical breadth-first traversal to a depth of one subdirectory.

    If no matching file is found, the parent and parent sibling directories of the
    current directory are scanned in an alphabetical breadth-first traversal to a
    height of one parent directory.

Files matching the pattern are processed in the order encountered. When a TVShow Manifest
is encountered, it is opened and scanned for an MPVTV PlaylistBag/Playlist or AssetList.
Without external guidance, the first MPVTV Playlist encountered is used for presentation;
if none is found, the AssetList is used.

```

The rationale behind this search algorithm is to first locate any top-level manifest containing MPV information, with a fallback of then finding named TVShow Manifests. It is allowed for the MPV document to be located up to one directory down.

N.B. By allowing the TVShow Manifest to carry the .XML extension or type, general purpose XML processors can operate on the MPV document and apply XML processing capabilities. For example, with Microsoft Internet Explorer 5.5 and above, an XML processing instruction in the INDEXTVM.XML file can invoke a style sheet that can transform the MPV document into an attractive browser-based presentation.

The search algorithm covers all of the following directories, where CWD is the current working directory. Naturally, when the path cannot be reached, it stops.

```

/R1/P1/CWD
/R1/P1/CWD/C1
/R1/P1/CWD/C2
/R1/P1/CWD/C3

```

/R1/P2
/R1/P3

But not these:

/R2
/R2/P2
/R1/P2/D1
/R1/P1/CWD/C1/E1

In each of the directories scanned, the application shall search for all of the possible TVShow Manifest file names.

7.5 Media Types and File Formats

MPV is an open asset management / playlist format that can support an expandable set of defined media file formats. Formats are identified using MIME media types, as is well-established practice for internet-era standards.

Appendix I: References

[CSS2]

"Cascading Style Sheets, level 2", Bert Bos, Håkon Wium Lie, Chris Lilley, Ian Jacobs. W3C Recommendation 12 May 1998.

Available at <http://www.w3.org/TR/REC-CSS2>

[DATETIME]

"Date and Time Formats", M. Wolf, C. Wicksteed. W3C Note 27 August 1998,

Available at: <http://www.w3.org/TR/NOTE-datetime>

[DC]

"Dublin Core Metadata Initiative", a Simple Content Description Model for Electronic Resources.

Available at <http://purl.org/DC/>

[DC-QUAL]

"Using Dublin Core – Dublin Core Qualifiers". Available at

<http://dublincore.org/documents/usageguide/qualifiers.shtml>

[DC-NMF]

"Dublin Core Normalized Metadata Format Profile Specification 1.0"; OSTA, 2002.

Available at <http://www.osta.org/mpv/>

[FreeDB]

"DISCID Howto", at <http://freedb.org/modules.php?name=Sections&sop=viewarticle&artid=6>

[ID3]

"The MPEG audio meta data format ID3", Martin Nielson, Dec. 1, 2000, Expired Internet Draft.

<http://www.watersprings.org/pub/id/draft-nilsson-id3-00.txt>

[ISO639-1]

'ISO/FDIS 639-1. Codes for the representation of names of languages, Part 1: Alpha-2 code.' Technical committee / subcommittee: TC 37 / SC 2

[ISO3166-1]

'ISO 3166-1 Codes for the representation of names of countries and their subdivisions - Part 1: Country codes.'

<http://www.iso.org/iso/en/prods-services/iso3166ma/index.html>

[ISO8601]

"Data elements and interchange formats - Information interchange - Representation of dates and times", International Organization for Standardization, 1998.

[ISO10646]

"Information Technology -- Universal Multiple-Octet Coded Character Set (UCS) -- Part 1: Architecture and Basic Multilingual Plane", ISO/IEC 10646-1:1993. This reference refers to a set of codepoints that may evolve as new characters are assigned to them. This reference therefore includes future amendments as long as they do not change character assignments up to and including the first five amendments to ISO/IEC 10646-1:1993. Also, this reference assumes that the character sets defined by ISO 10646 and Unicode remain character-by-character equivalent. This reference also includes future publications of other parts of 10646 (i.e., other than Part 1) that define characters in planes 1-16. "

[JFIF]

"JPEG File Interchange Format, Version 1.02"; Eric Hamilton, September 1992.
Available at <http://www.w3.org/Graphics/JPEG/jfif.txt>

[MANIFEST]

"XML Manifest Specification 1.0"; OSTA, 2002.,
Available at <http://www.osta.org/mpv/>

[MD5]

"The MD5 Message-Digest Algorithm", RFC 1321, April 1992.
Available at <http://www.ietf.org/rfc/rfc1321.txt>. Further information and source code available at <http://userpages.umbc.edu/~mabzug1/cs/md5/md5.html>

[MIME-2]

"RFC 2046: Multipurpose Internet Mail Extensions (MIME) Part Two: Media Types"; N. Freed, N. Borenstein, November 1996.
Available at <ftp://ftp.isi.edu/in-notes/rfc2046.txt>

[MIMEMPA]

"RTP Profile for Audio and Video Conferences with Minimal Control", H. Schulzrinne and others, anuary 1996, RFC 1890. Available at <http://www.ietf.org/rfc/rfc3003.txt>. Relevant discussion in [MIMEMPEG].

[MIMEMPEG]

"The audio/mpeg Media Type", Martin Nilsson, November 2000, RFC 3003. Available at <http://www.ietf.org/rfc/rfc3003.txt>

[MIMEMPEG4]

"RTP payload format for MPEG-4 Audio/Visual streams", Y. Kikuchi, T. Nomura, S. Fukunaga, Y. Matsui, H. Kimata, Nov. 2000, RFC 3016, Internet Engineering Task Force
Available at <ftp://ftp.isi.edu/in-notes/rfc3016.txt>

[MIMETYPES-REG]

IANA official registry of MIME media types
Available at <http://www.isi.edu/in-notes/iana/assignments/media-types/media-types>

[MPV-Basic]

"MPV – Basic Profile Specification", OSTA, 2002,
Available at <http://www.osta.org/mpv/>

[MPVCore]

"MPV Core Specification 1.0"; OSTA, 2002.,
Available at <http://www.osta.org/mpv/>

[MPV-Pres]

"MPV Presentation Profile Specification 1.0"; OSTA, 2002.,
Available at <http://www.osta.org/mpv/>

[MultiAudio]

"MultiAudio 1.0"; OSTA, 2001.,
Available at <http://www.osta.org/>

[NMF]

"Normalized Metadata Format Specification 1.0"; OSTA, 2002.,
Available at <http://www.osta.org/mpv/>

[PNG-MIME]

"Registration of new Media Type image/png"; Glenn Randers-Pehrson, Thomas Boutell, 27 July 1996.
Available at <ftp://ftp.isi.edu/in-notes/iana/assignments/media-types/image/png>

[PNG-REC]

"PNG (Portable Network Graphics) Specification Version 1.0"; Thomas Boutell (Ed.).
Available at <http://www.w3.org/TR/REC-png>

[QT]

"QuickTime Movie File Format Specification", May 1996.
Available at <http://developer.apple.com/techpubs/quicktime/qtdevdocs/REF/refFileFormat96.htm>

[QT-MIME]

"Registration of new MIME content-type/subtype"; Paul Lindner, 1993.
Available at <http://www.isi.edu/in-notes/iana/assignments/media-types/video/quicktime>

[RDFsyntax]

"Resource Description Framework (RDF) Model and Syntax Specification", Ora Lassila and Ralph R. Swick.
W3C Recommendation 22 February 1999,
Available at <http://www.w3.org/TR/REC-rdf-syntax/>

[RDFschema]

"Resource Description Framework (RDF) Schema Specification", Dan Brickley and R.V. Guha. W3C Proposed
Recommendation 03 March 1999,
Available at <http://www.w3.org/TR/PR-rdf-schema/>

[RFC1766]

"Tags for the Identification of Languages", H. Alvestrand, March 1995.
Available at <ftp://ftp.isi.edu/in-notes/rfc1766.txt>

[SMIL10]

"Synchronized Multimedia Integration Language (SMIL) 1.0" P. Hoschka. W3C Recommendation 15 June
1998,
Available at <http://www.w3.org/TR/REC-smil>.

[SMIL20]

"Synchronized Multimedia Integration Language (SMIL 2.0) Specification". W3C Working Draft, work in
progress.
Available at <http://www.w3.org/TR/smil20/>

[SMIL-MOD]

"Synchronized Multimedia Modules based upon SMIL 1.0", Patrick Schmitz, Ted Wugofski and Warner ten
Kate. W3C Note 23 February 1999,
Available at <http://www.w3.org/TR/NOTE-SYMM-modules>

[URI]

"Uniform Resource Identifiers (URI): Generic Syntax", T. Berners-Lee, R. Fielding, L. Masinter, August 1998.
Note that RFC 2396 updates [RFC1738] and [RFC1808].

[UCS-2]

16-bit encoding of ISO 10646, commonly known as the Unicode character set.

[UTF-8]

Yergeau, F., "UTF-8, a transformation format of ISO 10646", RFC 2279, January 1998.

[W3C-NSURI]

"URIs for W3C namespaces". Policy and administrative issue for W3C, Oct. 1999.
Available at <http://www.w3.org/1999/10/nsuri>

[XML10]

"Extensible Markup Language (XML) 1.0" T. Bray, J. Paoli and C.M. Sperberg-McQueen. W3C Recommendation 10 February 1998 ,
Available at <http://www.w3.org/TR/REC-xml>

[XML-NS]

"Namespaces in XML", Tim Bray, Dave Hollander, Andrew Layman. W3C Recommendation 14 January 1999,
Available at <http://www.w3.org/TR/REC-xml-names>

[XMP-FW]

"XMP – Extensible Metadata Platform 14 Sept 01" , Copyright 2001 Adobe Inc,
Available at <http://xml.coverpages.org/XMP-MetadataFramework.pdf>. Also at
<http://partners.adobe.com/asn/developer/xmp/download/docs/MetadataFramework.pdf>

[XSHEMA]

"XML Schema, XML Schema Part 1: Structures". W3C Working Draft, work in progress.
Available at <http://www.w3.org/TR/xmlschema-1/>

[XSL]

"Extensible Stylesheet Language (XSL) Specification", Stephen Deach. W3C Working Draft, work in progress.
Available at <http://www.w3.org/TR/xsl/>