Proposal for a

MultiPhotoVideo Specification

Metadata for Digital Photo-Video Content on Optical Storage Discs

Revision 0.14
Working Draft

December 3, 2001

POINTS OF CONTACT

OSTA	*Technical Content*
Ray Freeman OSTA Facilitator	Felix Nemirovsky Chairman, MultiRead Subcommittee
311 East Carrillo Street Santa Barbara, CA 93101	Oak Technology, Inc. 139 Kifer Court Sunnyvale, CA 94086
Tel: +1 805 963 3853 Fax: +1 805 962 1541 Email: ray@osta.org	Tel: +1 415 643 0944 Fax: +1 520 299 4319 Email: felixn@oaktech.com
http://www.osta.org	

ABSTRACT

This specification defines a format for organizing digital photo, video, and related audio content on an optical disc.

LICENSING
IMPORTANT NOTICES

DIG 35 Specification Copyright Notice

Contents

Chapter 1: Introduction

1.1 Motivation

Stamped optical storage discs (CDs, recently DVDs) are an established success story with 20 years of consumer familiarity. Recordable optical storage discs are a recent technology success story – the writer devices and blank discs are broadly available worldwide at ever decreasing cost and boast a projected 300M users by 2004. They are robust and broadly compatible with a wide variety of installed base devices. Because of their capacity, low cost, and robustness, recordable optical storage discs are a natural storage medium for personal digital photos and video.

Digital still cameras and digital camcorders are exploding in popularity and rapidly adding support for capture of still images with and without audio, video clips with and without audio, and even just audio clips. A rich diversity of capture devices and content formats exists.

Yet a lack of suitable standards hinders broad consumer use and enjoyment of the content captured by these devices. While many popular consumer electronics devices, computers, and services can produce optical discs containing photo-video content, few standards or even conventions exist for how to organize and structure that content. In fact, there is no format that exists today which can support a broad range of disc layouts, disc creation tools, and uses of photo-video content while also allowing for high usability, interoperability and compatibility.

This document specifies the MultiPhotoVideo (MPV) format, which provides means to organize and access a diversity of photo, video, and related audio content on optical storage discs, which may be used to provide users with a broad array of useful capabilities while achieving high usability, interoperability, and compatibility with current and future products.

DVD-Video and VideoCD

DVD player devices have achieved the highest rate of consumer adoption in history. They provide an ideal viewing platform for photo-video content contained on optical storage discs in the DVD-Video and VideoCD formats. Yet these formats are highly focused on playback of precisely formatted presentations on consumer electronics devices and assume production processes that are highly involved, require significant computing resources, and impose constraints on the disc content and capabilities.

These formats and assumptions are not appropriate to many practical and desirable uses of photo-video content on optical discs. They don't support many desirable tasks, such as printing of photo-video content, exchanging content with other devices and internet services, low cost/resource disc authoring, multisession disc writing, and so on. DVD player makers are acknowledge this, as shown by the emergence of new firmware applications specifically for interacting with photo-video content on discs not formatted as DVD-Video or VideoCD.

MultiPhotoVideo is complementary to the DVD-Video and VideoCD formats and does not diminish nor constrain the use and value of the DVD-Video and VideoCD formats on optical discs; indeed, it tacitly acknowledges and leverages their ubiquity.

MPV enables and embraces the use of optical discs containing photo-video content in a broader array of devices, software, and services that deliver more capabilities than DVD-Video and VideoCD can support. MPV also enables users of photo-video content on optical discs to receive similar usability and interoperability to that which DVD-Video and VideoCD have achieved and which has so benefited the consumer, content publishers, and the electronics industry alike.

1.2 Vision and Goals

The vision of the MultiPhotoVideo specification is:

> *A format, placed on optical storage discs created by many means and containing digital photo-video content files, that enables a user to use the content in many ways and by many means that are better than can be achieved using discs with photo-video content files alone.*

Expanding the vision to include specific examples brings it to life and reveals by example many of the goals of the specification.

A format,
- *consisting of metadata and not the photo-video content itself*
- *independent of the filesystem or storage medium*

placed on optical storage discs
- *CD or DVD*
- *write-once, rewritable, or stamped*

created by many means
- *consumer devices (CD and DVD recorders, digital cameras, …)*
- *computers (any software application, any OS, any form-factor)*
- *service providers (retail, wholesale, internet, commercial)*

and containing digital photo-video content files
- *personal or commercial content*
- *still images with zero to many associated audio clips*
- *video clips with zero to many associated audio clips*
- *audio clips*

that enables a user to use the content in many ways
- *long-term content archive*
- *view and interact with photo-video content*
- *print photo-video content*
- *incrementally modify and augment photo-video content*
- *index, search, retrieve, and manage large quantities of photo-video content*
- *access a diversity of services that act on the photo-video content*

and by many means
- *consumer devices (DVD players or recorders, game stations, …)*
- *computers (any OS, any form-factor)*
- *retail points-of-service (kiosks, service providers, …)*
- *internet points-of-service (photo websites, photo printing services, …)*

that are better
- *more usable – in responsiveness, presentation, interaction, …*
- *more useful – in compatibility, capabilities, flexibility, …*
- *more rewarding – in meeting the user's needs, wants and objectives*

than can be achieved using discs with photo-video content files alone.

1.3 Scope

The specification includes the following elements:
- informative discussion and description of requirements, considerations, approach, and intent
- normative definition of format structure and content
- examples
- considerations and recommendations for use and implementation

Focus on Metadata

The MPV format extends an existing metadata standard for digital images, DIG35 [DIG35-2001], which is maintained by the International Imaging Industry Association (I3A). DIG35 is the product of an extended collaborative effort by an organization with more than 70 member companies to define a set of public metadata for digital still images, both single images and collections of images. It has been adopted as the metadata format for JPEG2000.

MPV extends DIG35 to define a set of metadata for digital photo-video content, both individual items and collections. The metadata may be produced and consumed by a diverse set of current and future devices, software, and services. In addition to the metadata definition, a recommended implementation is defined to enable rapid location and identification of the metadata content on the storage medium.

MPV does not concern itself with vendor-specific proprietary metadata or particular application domains. These may be defined outside the scope of this specification, yet still be integrated seamlessly into a given MPV implementation. Devices, software, and services will ignore metadata in MPV that it cannot identify. This preserves broad interoperability of MPV-based solutions.

MPV emphasizes declarative metadata describing the properties and associations of the photo-video content while largely avoiding procedural or proscriptive metadata that mandates visual appearance or behaviour. This enables a diversity of implementations and uses of the metadata by devices, software, and services.

The MPV format is not a filesystem replacement. MPV structures typically are stored as files in a filesystem supported by the storage medium. MPV also does not place any constraints on content placement in a filesystem. Consequently, support for MPV can be added into most existing devices, software, and services that produce discs of photo-video content without requiring major changes to existing software and firmware. Furthermore, devices, software, and services that consume discs with photo-video content and which are not MPV-aware can continue to interact with the filesystem content.

Photo-Video Content

The MPV format does NOT specify the format of nor contain any digital photo-video content itself. However, it is designed to represent and organize photo-video content of the following forms:

- individual content items or collections of content (playlists)
- personal or commercial content
- still images with zero to many associated audio clips
- video clips with zero to many associated audio clips
- audio clips
- documents

Focus on Optical Storage Discs

Defining and establishing the MPV format for use on optical storage discs is the primary focus of this specification and the reason for its sponsorship by OSTA, the Optical Storage Technology Association. Because recordable optical discs are robust, broadly compatible with a wide variety of existing devices, high capacity and low cost, they are a natural and important storage medium for personal digital photos and video.

OSTA envisions that future devices, software, and services that create optical storage discs containing photo-video content will additionally place the MPV formatted metadata on these discs. Complementing these disc producers, a broad array of devices, software, and services will utilize MPV formatted metadata when present on discs to enhance the user experience and capabilities offered.

Opportunity for Use Beyond Optical Storage Discs

The characteristics of the MPV format are in fact entirely general purpose, although development and implementation of the specification under sponsorship of the OSTA necessarily has focused it on optical storage.

Without any changes, the format can be used with other storage media, such as memory cards, magnetic disks, or even for exchanging information between software applications or services. Broader use of the MPV format would facilitate use and exchange of photo-video content without regard to physical storage media while retaining the user value provided by the MPV format.

A particular opportunity exists for digital camera makers. Most digital cameras today store their photo-video content on a memory card according to the guidelines of the "Design Rule for Camera File System" specification [DCF-1999]. These guidelines are fairly loose and each vendor has their own implementation practices. By adding the MPV format to a memory card while retaining its existing file system structure, compatibility with existing processes, devices, software, and services is maintained while allowing MPV-aware application to offer the user a better experience for enjoying and using the photo-video content.

The broadest context for adoption of the MPV format specification would be to extend the format upon which it is based, namely the International Imaging Industry Association's (I3A) DIG35 standard for metadata for digital images.

Chapter 2: Design Requirements

2.1 Design Goals

As an extension to DIG35, MPV strives for the same design goals. These goals were excerpted directly from the DIG35 specification and adjusted to encompass the MPV extensions to the standard.

The design goals of the MPV initiative are to define a metadata set that is:

- **INTERCHANGEABLE**: MPV is based on a sound conceptual model that is both generally applicable to many applications and assured to be consistent over time. MPV will create a better environment for locating and (re)using specific photo-video content.
- **EXTENSIBLE AND SCALEABLE**: MPV enables application developers and hardware manufacturers to add additional metadata fields. This allows future needs for metadata to be fulfilled with limited disruption of current solutions.
- **FILE FORMAT INDEPENDENT**: MPV does not rely on any specific file format and can therefore be supported by many current and future file formats and compression mechanisms.
- **CONSISTENT**: MPV can work well with existing standards today allowing the metadata standard to be widely acceptable and usable in a variety of application domains and user situations.
- **INTERNET-READY**: MPV provides seamless integration with the Internet by utilizing XML (the recommended implementation method) and by incorporating standards for connectivity with internet services.

In addition to the DIG35-derived goals, MPV has several additional design goals:

- **LOW-COST DEVICE IMPLEMENTATION**: MPV can be implemented in low-cost devices, meaning an implementation can be achieved that does not require large amounts of firmware, memory, CPU, disc performance, or connectivity. This is achieved by utilizing dual and equivalent representations: a recommended XML format suitable for use in computer software and service provider settings and an equivalent binary format suitable for processing by low-cost devices.
- **OPERATING SYSTEM, FILE SYSTEM AND STORAGE MEDIA INDEPENDENT**: MPV does not rely on any specific operating system, filesystem and storage media and can therefore be supported by many current and future operating systems, filesystems and storage media.
- **ENABLE VALUE-ADDED DELIVERY**: MPV provides sufficient and appropriately structured metadata to enable enhanced user experiences and capabilities to be created, delivered and accessed via devices, software, and services from a diversity of vendors. This enables devices, software, and services that **both** produce and consume MPV structures to add value to their implementations while being loosely coupled.

2.2 Design Principles

MPV adheres to the same design principles specified for DIG35, as excerpted here. These principles state that the MPV initiative shall:

- use existing standards and output of other organizations as much as possible while creating a future-looking metadata standard.
- focus on mid and long-term perspectives, not only on what current digital imaging technology may be able to offer today
- be simple for developers to utilize but sophisticated enough to cover a wide spectrum of features
- support information preservation, not data preservation Many devices, such is digital cameras, may store metadata in a format that end users are not familiar with which will discourage use. Thus, applications may need to apply appropriate conversions to transform these values into user-understandable formats.

- allow metadata redundancy. While values exist that can be calculated from other fields, at the definition level, redundant metadata do exist and need to be managed appropriately.

In addition to the DIG35-derived principles, MPV has several additional design principles.

- allow for variation and innovation among MPV metadata producers and consumers. Carefully distinguish between required content, optional content, and recommended implementation.

2.3 Use Cases/Scenarios

2.3.1 Basic User Scenario

A basic use case that motivates the specification and use of the MPV format is as follows.

An individual:

- acquires personal photo-video content in digital form using a digital camera or scanner
- writes the photo-video content to a write-once or rewritable optical storage disc using a computer and software application or a consumer electronics device
- adds to or modifies the content of the disc over an extended period of time
- uses the disc with a consumer device or computer software application, and utilizing only information already on the disc, performs a variety of read-only tasks, including:
    - choose among multiple collections or organizations of the content on the disc
    - for a given collection
        - play one or more multimedia "slideshows" of the content
        - interactively browse the individual content items
        - view a list of and access related collections or content items
        - while playing the show or browsing, mark one or more content items to use in some task
        - access an internet service for the collection or marked items, such as photo site upload or emailing
    - for a given content item in a collection
        - mark/unmark a content item
        - print a content item
        - view related content items
        - access an internet service for the item, such as photo site upload or emailing
    - view a history of how and when the content got onto the disc
- uses the disc with a consumer device or computer software application, and utilizing only information already on the disc, performs a variety of read/write tasks, including:
    - create or modify collections
        - create, modify, or delete collections on the disc
        - select, add, remove and sequence content items
        - add, remove, or edit information about content items or collections
    - modify content items
        - add, remove, or edit an item
        - add, remove, or edit information about the item
        - augment the item with additional content
        - cross-reference items
- gives the disc or a copy of the disc to someone else. That person wants to be able to use the disc in the same ways as the first person.

2.3.2 MPV Application Domain Uses

- Playback control standards for photo-video playback applications
    - CE photo-video applications, e.g. DVD players as music jukeboxes and photo viewers

  - PC software applications
- Control and access standards for other photo-video "playback"/"use" applications
  - printing
  - e-services
  - proprietary capabilities
  - content management
- Manifest for associating and accessing content on write-once and rewritable media
  - all flavors: optical discs (CD, DVD), memory cards (SM/SD/CF/MS/…), harddisks, even internet
  - file system and low-level formatting neutral
- Manifest for photo-video capture devices and applications
  - Digital cameras, CD/DVD burning software

2.3.3 Content/Disc Creation Needs

General Needs

- Simple, non-invasive to existing disc formats and content: can be added on
- Supports incremental update/editing
- Viable in low-cost CE devices

Support Digital Cameras

- media types: stills, stills + audio, video
- camera produces metadata file
- multiple write sessions

Support Computers/Software Users

- produce fancy slideshow/video experiences
- capture metadata from user

Support Business Needs

- brand presence carried through to presentation, e.g. logo, "skin"
- value-added capabilities available in many use/playback environments

Support Editing

- edit content and properties
- create associations between files
- store results
- support asset management

2.3.4 Content/Disc Use/Playback

A given storage medium and MPV structure needs to support the following uses:

General Playback Needs

- fast startup, responsive
- allow for possible playback experiences: application-driven, content-driven
- allow for diverse use/playback environments: DVD players, PCs, internet, …

- detect & apply rotation metadata
- display titles, other info
- no interaction with file system
- honor content/producer branding, connectivity to business

Device Playback

- resource-constrained processing environment

Picture Lists/Albums

- plays pictures in given sequence
- can play associated content, e.g. stills + audio
- can have multiple picture lists/albums
- can cross-referencing to pictures in other lists, list to list

Interactive Browsing

- responsive
- display more metadata on demand

Slideshow

- play background audio
- on-the fly slideshow
- pre-rendered slideshow

Printing

- high resolution images
- print-formatted documents

Internet Services

- access to diversity of services
- link to web services, use content

Asset Management

- create and access rich metadata
- robustly refer to any kind of content

2.4 Derived Requirements

The use cases and scenarios lead to the following requirements on MPV:

Metadata Requirements:

- look for existing metadata standard that:
    - is or will be adopted by the digital camera industry
    - can accept proprietary or nextgen extensions without damaging compatibility

- video and audio integrated in right from the start => MultiPhoto and MultiVideo should not be separate: instead: MultiPhotoVideo
- provide for cross-referencing
    - one item or list can reference other items/lists
- provide for content "variants" – for a given content item or list (album):
    - multiple representations: file lists, video slideshows, print docs
    - multiple resolutions: high res, mid res, thumbnails
- connectivity to internet services
    - conventions for metadata tranfer enable access to diversity of services
- support presentation metadata & content for branding
- Unique IDs can go on everything

Implementation Requirements:

- single fixed entrypoint, from which all other files can be reached
- no “executable” code involved – everything is data driven
- supports single and multisession implementations
- support incremental content update
- no tie to physical layer implementations: any storage media
- supported data formats fully extensible, but baseline formats are:
    - images: JPEG / EXIF 2.1 (1998)
    - video: MPEG1 Whitebook (VCD)
    - audio: MPA, WAV, MP3
- supports multiple data representations
    - images: multiple resolutions
    - video: still frame, video
- support burning CDs/DVDs without requiring special applications
    - primary issue: variable handling by CD file system of file names

Chapter 3: Examples

3.1 MPV Basic Example

This example shows a MPV document containing a list of the following:

- still image, described in DIG35 format
- still image with two associated audio files, described in MPV format
- video, described in MPV format

MPV defines extensions to DIG35. As such, it retains a very similar structure and utilizes many existing DIG35 structures.

3.1.1 XML Representation

This is an XML representation, as it would exist on a disc.

MPV example of a lit of content items	Notes
`<?XML version="1.0" >`	XML document
`<MPV:MPV xmlns=" <http://www.digitalimaging.org/dig35/1.1/xml>"`	MPV Wrapper
`  xmlns:MPV="http://www.osta.org/MPV/1.0/xml#" >`	
`  <MPV:LIST" MPV:ID="0001">`	A list of metadata
`    <MPV:BASIC LIST PARAM>`	MPV Metadata for this list
`      <MPV:BASIC LIST INFO>`	
`        <FILE_FORMAT>`	
`          <FILE NAME>/MPV/TOC.MPV</FILE NAME>`	Filename of this list
`          <FORMAT TYPE>MPV</FORMAT TYPE>`	(refers to it's own file)
`        </FILE_FORMAT>`	
`        <MPV:UNIQUE ID>`	
`          <UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf6</UID>`	UUID of the list
`        </MPV:UNIQUE_ID>`	
`      </MPV:BASIC LIST INFO>`	
`    </MPV:BASIC LIST PARAM>`	
`    <MPV:LIST_CREATION>`	
`      <MPV:GENERAL CREATION INFO>`	
`        <CREATION TIME>2001-05-14T21:43:23</CREATION TIME>`	List creation time
`        <MPV:CREATOR>`	
`          <PERSON NAME>`	
`            <NAME COMP Type="Given">John</NAME COMP>`	List creator
`            <NAME_COMP Type="Family">Smith</NAME_COMP>`	
`          </PERSON NAME>`	
`        </MPV:CREATOR>`	
`      </MPV:GENERAL_CREATION_INFO>`	
`    </MPV:LIST CREATION`	
`    <CONTENT DESCRIPTION>`	
`      <CAPTION>Trip to Waldo Lake Sept. 14, 2001 Smith & Jones`	List caption (3 lines)
`       Familes</CAPTION>`	
`    </CONTENT DESCRIPTION>`	
`    <METADATA Type="Single" MPV:ID="0002"> <!-- DIG35 Metadata of Img -->`	Image 1 - DIG35 metadata
`      <BASIC IMAGE PARAM>`	
`        <BASIC IMAGE INFO>`	
`          <FILE_FORMAT>`	
`            <FILE NAME>/my pictures/image.jpg</FILE NAME>`	File information
`            <FORMAT_TYPE>EXIF</FORMAT_TYPE>`	Format

`</FILE FORMAT>`	
`<IMAGE_ID>`	UUID
`<UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf6</UID>`	
`<ID TYPE>http://www.digitalimaging.org/dig35/UUID</ID TYPE>`	
`</IMAGE_ID>`	
`<IMAGE SIZE>`	Width & height
`<WIDTH>1600</WIDTH>`	
`<HEIGHT>1200</HEIGHT>`	
`</IMAGE SIZE>`	Compression
`<COMPRESSION>JPEG</COMPRESSION>`	
`</BASIC_IMAGE_INFO>`	
`<COMPONENT INFO>`	Color depth
`<COMPONENTS>RGB</COMPONENTS>`	
`<NUM_COMPONENT>3</NUM_COMPONENT>`	
`<COMP SIZE>8</COMP SIZE>`	
`<COMP SIZE>8</COMP SIZE>`	
`<COMP_SIZE>8</COMP_SIZE>`	
`</COMPONENT INFO>`	
`</BASIC IMAGE PARAM>`	
`<IMAGE_CREATION>`	
`<GENERAL CREATION INFO>`	Date taken
`<CREATION TIME>2001-05-14T21:43:23</CREATION TIME>`	
`<IMAGE_CREATOR>`	
`<PERSON NAME>`	Photographer
`<NAME COMP Type="Given">John</NAME COMP>`	
`<NAME_COMP Type="Family">Smith</NAME_COMP>`	
`</PERSON NAME>`	
`</IMAGE CREATOR>`	
`</GENERAL_CREATION_INFO >`	
`<CAMERA CAPTURE>`	
`<CAMERA SETTINGS>`	
`<CAMERA_LOCATION>`	Location
`<COMMENT>At home</COMMENT>`	
`</CAMERA LOCATION>`	
`<ORIENTATION>`	Orientation
`<ROLL>90</ROLL>`	
`</ORIENTATION>`	
`</CAMERA_SETTINGS>`	
`</IMAGE CREATION>`	
`<CONTENT DESCRIPTION>`	Group title
`<GROUP_CAPTION>Trip to Waldo Lake Sept. 14, 2001 Smith &`	
`Jones Familes</GROUP CAPTION>`	Image title
`<CAPTION>Mary and the kids</CAPTION>`	Info
`<COMMENT>kids=Marty, John, Mike</COMMENT>`	
`</CONTENT DESCRIPTION>`	
`</METADATA>`	Image 2 - MPV metadata
`<MPV:IMAGE MPV:ID="0003">  <!-- MPV metadata of image with audio -->`	
`<MPV:BASIC IMAGE PARAM>`	
`<MPV:BASIC IMAGE INFO>`	
`<FILE_FORMAT>`	File information
`<FILE NAME>/my pictures/image2.jpg</FILE NAME>`	
`<FORMAT TYPE>EXIF</FORMAT TYPE>`	
`</FILE_FORMAT>`	
`<MPV:UNIQUE ID>`	UUID
`<UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf6</UID>`	
`</MPV:UNIQUE_ID>`	
`<IMAGE SIZE>`	Width & height
`<WIDTH>1600</WIDTH>`	

`<HEIGHT>1200</HEIGHT>`	
`</IMAGE_SIZE>`	Media type
`<MPV:MEDIA TYPE>image/EXIF</MPV:MEDIA TYPE>`	
`</MPV:BASIC IMAGE INFO>`	
`</MPV:BASIC_IMAGE_PARAM>`	Creation metadata
`<MPV:IMAGE CREATION>`	
`<MPV:GENERAL CREATION INFO>`	Date taken
`<CREATION_TIME>2001-05-15T09:18:35</CREATION_TIME>`	
`<MPV:CREATOR>`	
`<PERSON NAME>`	Photographer
`<NAME_COMP Type=”Given”>John</NAME_COMP>`	
`<NAME COMP Type=”Family”>Smith</NAME COMP>`	
`</PERSON NAME>`	
`</MPV:CREATOR>`	
`</MPV:GENERAL CREATION INFO >`	
`<CAMERA CAPTURE>`	
`<CAMERA_SETTINGS>`	
`<ORIENTATION>`	Orientation
`<ROLL>0</ROLL>`	
`</ORIENTATION>`	
`</CAMERA SETTINGS>`	
`</CAMERA CAPTURE>`	
`<MPV:IMAGE_CREATION>`	Content metadata
`<MPV:CONTENT DESCRIPTION>`	List caption
`<GROUP CAPTION>Trip to Waldo Lake Sept. 14, 2001 Smith &`	
`Jones Familes</GROUP_CAPTION>`	Associated content
`<MPV:ASSOCIATED MPV:ID="#0004">`	Audio content
`<MPV:AUDIO MPV:ID="#0005">`	
`<MPV:BASIC_AUDIO_PARAM>`	
`<MPV:BASIC AUDIO INFO>`	
`<FILE FORMAT>`	Filename
`<FILE_NAME>/my pictures/image2.wav</FILE_NAME>`	Format
`<FORMAT TYPE>WAV</FORMAT TYPE>`	
`</FILE FORMAT>`	
`<MPV:UNIQUE_ID>`	UUID
`<UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf7</UID>`	
`</MPV:UNIQUE ID>`	Media type
`<MPV:MEDIA_TYPE>stream/WAVE</MPV:MEDIA_TYPE>`	
`</MPV:BASIC AUDIO INFO>`	
`</MPV:BASIC AUDIO PARAM>`	
`<MPV:CONTENT_DESCRIPTION>`	Comment
`<COMMENT>Ambient sound</COMMENT>`	
`</MPV:CONTENT DESCRIPTION>`	
`</MPV:AUDIO>`	
`</MPV:ASSOCIATED>`	Associated content
`<MPV:ASSOCIATED>`	Audio content
`<MPV:AUDIO MPV:ID="#0006">`	
`<MPV:BASIC AUDIO PARAM>`	
`<MPV:BASIC AUDIO INFO>`	
`<FILE_FORMAT>`	Filename
`<FILE NAME>/my pictures/annotation1.mp3</FILE NAME>`	Format
`<FORMAT TYPE>MP3</FORMAT TYPE>`	
`</FILE_FORMAT>`	
`<MPV:UNIQUE ID>`	UUID
`<UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf8</UID>`	
`</MPV:UNIQUE_ID>`	Media type
`<MPV:MEDIA TYPE>audio/MPEG1Payload</MPV:MEDIA TYPE>`	
`</MPV:BASIC AUDIO INFO>`	

`</MPV:BASIC AUDIO PARAM>`	
`<MPV:CONTENT_DESCRIPTION>`	
`<PERSON>`	Person who is talking
`<NAME COMP Type="Given">Marty</NAME COMP>`	
`<NAME_COMP Type="Family">Smith</NAME_COMP>`	
`</PERSON>`	Comment
`<COMMENT>Marty talking about the moment.</COMMENT>`	
`</MPV:CONTENT_DESCRIPTION>`	
`</MPV:AUDIO>`	
`</MPV:ASSOCIATED>`	
`</MPV:CONTENT_DESCRIPTION>`	
`</MPV:IMAGE>`	Metadata of a video
`<MPV:VIDEO MPV:ID="0007">  <!-- MPV metadata of VIDEO -->`	
`<MPV:BASIC_VIDEO_PARAM>`	
`<MPV:BASIC VIDEO INFO>`	
`<FILE FORMAT>`	Filename
`<FILE_NAME>/my pictures/VIDEO1.mpg</FILE_NAME>`	Format type
`<FORMAT TYPE>MPG</FORMAT TYPE>`	
`</FILE FORMAT>`	
`<MPV:UNIQUE_ID>`	UUID
`<UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf6</UID>`	
`</MPV:UNIQUE ID>`	
`<IMAGE_SIZE>`	Video image size
`<WIDTH>352</WIDTH>`	
`<HEIGHT>240</HEIGHT>`	
`</IMAGE_SIZE>`	Media type
`<MPV:MEDIA TYPE>stream/MPEG1VideoCD</MPV:MEDIA TYPE>`	
`</MPV:BASIC VIDEO INFO>`	
`</MPV:BASIC_VIDEO_PARAM>`	
`<MPV:VIDEO CREATION>`	
`<MPV:GENERAL CREATION INFO>`	Date taken
`<CREATION_TIME>2001-05-15T10:13:29</CREATION_TIME>`	How captured
`<MPV:CREATION SOURCE>Digital Camera</VIDEO SOURCE>`	Of what
`<MPV:CREATION TYPE>Original Scene</SCENE TYPE>`	
`<MPV:CREATOR>`	
`<PERSON NAME>`	Photographer
`<NAME COMP Type="Given">John</NAME COMP>`	
`<NAME_COMP Type="Family">Smith</NAME_COMP>`	
`</PERSON NAME>`	
`</MPV:CREATOR>`	
`</MPV:GENERAL_CREATION_INFO >`	
`<CAMERA CAPTURE>`	
`<CAMERA INFO>`	
`<MANUFACTURER>`	Camera maker
`<ORG NAME>Acme</ORG NAME>`	
`</MANUFACTURER>`	Camera model
`<MODEL>MODEL 1000</MODEL>`	
`</CAMERA INFO>`	
`</CAMERA CAPTURE>`	
`</MPV:VIDEO_CREATION>`	
`<MPV:CONTENT DESCRIPTION>`	List caption
`<GROUP CAPTION>Trip to Waldo Lake Sept. 14, 2001 Smith & Jones Familes</GROUP_CAPTION>`	
`</MPV:CONTENT DESCRIPTION>`	
`</MPV:VIDEO>`	
`</MPV:LIST>`	MPV Table of Contents
`<MPV:TOC xmlns:MPV="mpv-nsval" MPV:ID="0008">`	Namespaces
`<MPV:NS ENTRIES>`	DIG35 namespace

`<MPV:NS ENTRY NS-PREFIX=""` `NS-URI="<http://www.digitalimaging.org/dig35/1.1/xml>"/>`	MPV namespace
`<MPV:NS ENTRY NS-PREFIX="MPV"` `NS-URI=" <http://www.osta.org/MPV/1.0/xml#>" />`	
`</MPV:NS_ENTRIES>`	Offsets for random access
`<MPV:METADATA OFFSETS>`	List
`<MPV:OFFSET type="MPV:LIST" byteref="545">`	Image 1
`<MPV:OFFSET type="DIG35:METADATA" byteref="705"/>`	Image 2
`<MPV:OFFSET type="MPV:IMAGE" byteref="1643">`	List
`<MPV:OFFSET type="MPV:LIST" byteref="1970">`	Image 2 Audio 1
`<MPV:OFFSET type="MPV:AUDIO" byteref="2010">`	Image 2 Audio 2
`<MPV:OFFSET type="MPV:AUDIO" byteref="2320">`	
`</MPV:OFFSET>`	Video
`<MPV:OFFSET type="MPV:VIDEO" byteref="2690"/>`	
`</MPV:OFFSET>`	
`</MPV:METADATA OFFSETS>`	
`</MPV:TOC>`	
`</MPV:MPV>`	Processing Instruction
`<?MPV TOC-OFFSET="2853" DOC-SIZE="3333" MPV?>`	

3.1.2 Data Pathnames

The example contains the following data items, as listed by element path. This view on the example makes it easy to see the structure of the file and what data items can be retrieved.

Info about this MPV list

```
/mpv:MPV/List/Basic_List_Params/Basic_List_Info/dig35:File_Format/File_Name
/mpv:MPV/List/Basic List Params/Basic List Info/dig35:File Format/Format Type
/mpv:MPV/List/Basic_List_Params/Basic_List_Info/Unique_Id/Uid
/mpv:MPV/List/List_Creation/General_Creation_Info/dig35:Creation_Time
/mpv:MPV/List/List Creation/General Creation Info/Creator/dig35:Person Name/Name Comp
/mpv:MPV/List/List_Creation/General_Creation_Info/Creator/dig35:Person_Name/Name_Comp
/mpv:MPV/List/List_Creation/dig35:Software_Creation/Software_Info/Manufacturer/Org_Name
/mpv:MPV/List/List Creation/dig35:Software Creation/Software Info/Model
/mpv:MPV/List/List_Creation/dig35:Software_Creation/Software_Info/Version
/mpv:MPV/List/Content_Description/Caption
```

First item – still image using pure DIG35 metadata

```
/mpv:MPV/List/dig35:Metadata/Basic Image Param/Basic Image Info/File Format/File Name
/mpv:MPV/List/dig35:Metadata/Basic_Image_Param/Basic_Image_Info/File_Format/Format_Type
/mpv:MPV/List/dig35:Metadata/Basic_Image_Param/Basic_Image_Info/Image_Id/Uid
/mpv:MPV/List/dig35:Metadata/Basic Image Param/Basic Image Info/Image Id/Id Type
/mpv:MPV/List/dig35:Metadata/Basic_Image_Param/Basic_Image_Info/Image_Size/Width
/mpv:MPV/List/dig35:Metadata/Basic_Image_Param/Basic_Image_Info/Image_Size/Height
/mpv:MPV/List/dig35:Metadata/Basic Image Param/Basic Image Info/Compression
/mpv:MPV/List/dig35:Metadata/Basic_Image_Param/Component_Info/Components
/mpv:MPV/List/dig35:Metadata/Basic_Image_Param/Component_Info/Num_Components
/mpv:MPV/List/dig35:Metadata/Basic Image Param/Component Info/Comp Size
/mpv:MPV/List/dig35:Metadata/Basic_Image_Param/Component_Info/Comp_Size
/mpv:MPV/List/dig35:Metadata/Basic_Image_Param/Component_Info/Comp_Size
/mpv:MPV/List/dig35:Metadata/Image Creation/General Creation Info/Creation Time
/mpv:MPV/List/dig35:Metadata/Image_Creation/General_Creation_Info/Image_Creator/Person_Name
/mpv:MPV/List/dig35:Metadata/Image_Creation/General_Creation_Info/Image_Creator/Person_Name
/mpv:MPV/List/dig35:Metadata/Image Creation/Camera Capture/Camera Settings/Camera Location/Comment
/mpv:MPV/List/dig35:Metadata/Image_Creation/Camera_Capture/Camera_Settings/Orientation/Roll
/mpv:MPV/List/dig35:Metadata/Content_Description/Group_Caption
/mpv:MPV/List/dig35:Metadata/Content_Description/Caption
```



```
/mpv:MPV/List/dig35:Metadata/Content_Description/Comment
```

Second item – image with two associated audio files, using MPV metadata for images

```
/mpv:MPV/List/Image/Basic Image Param/Basic Image Info/dig35:File Format/File Name
/mpv:MPV/List/Image/Basic_Image_Param/Basic_Image_Info/dig35:File_Format/Format_Type
/mpv:MPV/List/Image/Basic Image Param/Basic Image Info/Unique Id/dig35:Uid
/mpv:MPV/List/Image/Basic Image Param/Basic Image Info/dig35:Image Size/Width
/mpv:MPV/List/Image/Basic_Image_Param/Basic_Image_Info/dig35:Image_Size/Height
/mpv:MPV/List/Image/Basic Image Param/Basic Image Info/Media Type
/mpv:MPV/List/Image/Basic Image Param/Component Info/Components
/mpv:MPV/List/Image/Basic_Image_Param/Component_Info/Num_Components
/mpv:MPV/List/Image/Basic Image Param/Component Info/Comp Size
/mpv:MPV/List/Image/Basic Image Param/Component Info/Comp Size
/mpv:MPV/List/Image/Basic_Image_Param/Component_Info/Comp_Size
/mpv:MPV/List/Image/Image Creation/General Creation Info/dig35:Creation Time
/mpv:MPV/List/Image/Image Creation/General Creation Info/Creator/dig35:Person Name
/mpv:MPV/List/Image/Image_Creation/General_Creation_Info/Creator/dig35:Person_Name
/mpv:MPV/List/Image/Image Creation/dig35:Camera Capture/Camera Settings/Orientation/Roll
/mpv:MPV/List/Image/Content Description/Group Caption
/mpv:MPV/List/Image/Content_Description/Associated/Audio/Basic_Audio_Param/Basic_Audio_Info/dig35:File_Format/File_Name
/mpv:MPV/List/Image/Content Description/Associated/Audio/Basic Audio Param/Basic Audio Info/dig35:File Format/Format Type
/mpv:MPV/List/Image/Content Description/Associated/Audio/Basic Audio Param/Basic Audio Info/UniqueId/dig35:UID
/mpv:MPV/List/Image/Content_Description/Associated/Audio/Basic_Audio_Param/Basic_Audio_Info/Media_Type
/mpv:MPV/List/Image/Content Description/Associated/Audio/Content Description/Comment
/mpv:MPV/List/Image/Content Description/Associated/Audio/Basic Audio Param/Basic Audio Info/File Format/File Name
/mpv:MPV/List/Image/Content_Description/Associated/Audio/Basic_Audio_Param/Basic_Audio_Info/File_Format/Format_Type
/mpv:MPV/List/Image/Content Description/Associated/Audio/Basic Audio Param/Basic Audio Info/UniqueId/dig35:UID
/mpv:MPV/List/Image/Content Description/Associated/Audio/Basic Audio Param/Basic Audio Info/Media Type
/mpv:MPV/List/Image/Content_Description/Associated/Audio/Content_Description/dig35:Person/Name_Comp
/mpv:MPV/List/Image/Content Description/Associated/Audio/Content Description/dig35:Person/Name Comp
/mpv:MPV/List/Image/Content_Description/Associated/Audio/Content_Description/Comment
```

Third item – video, using MPV metadata for video

```
/mpv:MPV/List/Video/Basic_Video_Param/Basic_Video_Info/dig35:File_Format/File_Name
/mpv:MPV/List/Video/Basic Video Param/Basic Video Info/dig35:File Format/Format Type
/mpv:MPV/List/Video/Basic Video Param/Basic Video Info/Unique Id/dig35:Uid
/mpv:MPV/List/Video/Basic_Video_Param/Basic_Video_Info/dig35:Image_Size/Width
/mpv:MPV/List/Video/Basic Video Param/Basic Video Info/dig35:Image Size/Height
/mpv:MPV/List/Video/Basic Video Param/Basic Video Info/Media Type
/mpv:MPV/List/Video/Video_Creation/General_Creation_Info/dig35:Creation_Time
/mpv:MPV/List/Video/Video Creation/General Creation Info/dig35:Creation Time
/mpv:MPV/List/Video/Video Creation/General Creation Info/Creator/dig35:Person Name
/mpv:MPV/List/Video/Video_Creation/General_Creation_Info/Creator/dig35:Person_Name
/mpv:MPV/List/Video/Video Creation/Camera Capture/Camera Info/Manufacturer/Org Name
/mpv:MPV/List/Video/Video_Creation/Camera_Capture/Camera_Info/Model
/mpv:MPV/List/Video/Content_Description/Group_Caption
```

MPV Table of Contents

```
/mpv:MPV/TOC/NS ENTRIES/NS ENTRY
/mpv:MPV/TOC/NS ENTRIES/NS ENTRY
/mpv:MPV/TOC/METADATA_OFFSETS/OFFSET
/mpv:MPV/TOC/METADATA OFFSETS/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA OFFSETS/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA_OFFSETS/OFFSET/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA OFFSETS/OFFSET/OFFSET/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA_OFFSETS/OFFSET/OFFSET/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA_OFFSETS/OFFSET
```


3.2 MPV Features Example

This example shows a number of MPV features:

- an image, with mid-res and thumbnail resolutions
- a video, with a still thumbnail
- a pre-rendered slideshow for a playlist
- branding elements for use MPV playback

3.2.1 XML Representation

This is an XML representation, as it would exist on a disc.

MPV example of a list of content items with some MPV features	Notes
`<?XML version="1.0" >`	XML document
`<MPV:MPV xmlns=" <http://www.digitalimaging.org/dig35/1.1/xml>"`	MPV Wrapper
`  xmlns:MPV="http://www.osta.org/MPV/1.0/xml#" >`	
`  <MPV:LIST" MPV:ID="0001">`	A list of metadata
`    <MPV:BASIC_LIST_PARAM>`	MPV Metadata for this list
`      <MPV:BASIC LIST INFO>`	
`        <FILE_FORMAT>`	
`          <FILE_NAME>/MPV/TOC.MPV</FILE_NAME>`	Filename of this list
`          <FORMAT TYPE>MPV</FORMAT TYPE>`	(refers to it's own file)
`        </FILE_FORMAT>`	
`        <MPV:UNIQUE_ID>`	
`          <UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf6</UID>`	UUID of the list
`        </MPV:UNIQUE_ID>`	
`      </MPV:BASIC_LIST_INFO>`	
`    </MPV:BASIC LIST PARAM>`	
`    <MPV:LIST_CREATION>`	
`      <MPV:GENERAL_CREATION_INFO>`	
`        <CREATION TIME>2001-05-14T21:43:23</CREATION TIME>`	List creation time
`      </MPV:GENERAL_CREATION_INFO>`	
`    </MPV:LIST_CREATION`	
`    <CONTENT DESCRIPTION>`	
`      <CAPTION>Trip to Waldo Lake Sept. 14, 2001 Smith & Jones`	List caption (3 lines)
`       Familes</CAPTION>`	
`    </CONTENT DESCRIPTION>`	
`    <MPV:BRAND_IDENTITY>`	
`      <MPV:BRAND_ELEMENTS>`	
`        <CAPTION>Acme Products</CAPTION>`	Brand caption to use
`        <MPV:LOGO>`	
`          <MPV:IMAGE>`	
`            <MPV:BASIC IMAGE PARAM>`	
`              <MPV:BASIC_IMAGE_INFO>`	
`                <FILE_FORMAT>`	
`                  <FILE NAME>/acme/logo.jpg</FILENAME>`	Brand logo imae to use
`                  <FORMAT_TYPE>JPG</FORMAT_TYPE>`	
`                </FILE_FORMAT>`	
`                <IMAGE SIZE>`	
`                  <WIDTH>160</WIDTH>`	Brand Logo size
`                  <HEIGHT>160</HEIGHT>`	
`                </IMAGE SIZE>`	
`              </MPV:BASIC_IMAGE_INFO>`	

`</MPV:BASIC IMAGE PARAM>`	
`</MPV:IMAGE>`	
`</MPV:LOGO>`	
`</BRAND IDENTITY>`	
`</BRAND_PRESENTATION>`	
`<MPV:IMAGE MPV:ID="0003">  <!-- MPV metadata of an image -->`	Image 2 - MPV metadata
`<MPV:BASIC IMAGE PARAM>`	
`<MPV:BASIC_IMAGE_INFO>`	
`<FILE FORMAT>`	
`<FILE NAME>/my pictures/image2.jpg</FILE NAME>`	File information
`<FORMAT_TYPE>EXIF</FORMAT_TYPE>`	
`</FILE FORMAT>`	
`<MPV:UNIQUE ID>`	
`<UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf6</UID>`	UUID
`</MPV:UNIQUE ID>`	
`<IMAGE SIZE>`	
`<WIDTH>1600</WIDTH>`	Width & height
`<HEIGHT>1200</HEIGHT>`	
`</IMAGE SIZE>`	
`<MPV:MEDIA_TYPE>image/EXIF</MPV:MEDIA_TYPE>`	Media type
`</MPV:BASIC IMAGE INFO>`	
`</MPV:BASIC IMAGE PARAM>`	
`<MPV:IMAGE_CREATION>`	Creation metadata
`<MPV:GENERAL CREATION INFO>`	
`<CREATION TIME>2001-05-15T09:18:35</CREATION TIME>`	Date taken
`</MPV:GENERAL_CREATION_INFO>`	
`</MPV:IMAGE CREATION>`	
`<MPV:VARIANTS>`	
`<MPV:MID_RES>`	Variants
`<MPV:IMAGE MPV:ID="#0005">`	Mid-res image
`<MPV:BASIC IMAGE PARAM>`	
`<MPV:BASIC_IMAGE_INFO>`	
`<FILE FORMAT>`	
`<FILE NAME>/my`	
`pictures/screenres/image2.jpg</FILE_NAME>`	File name
`<FORMAT TYPE>JPG</FORMAT TYPE>`	Format type
`</FILE FORMAT>`	
`<MPV:UNIQUE_ID>`	
`<UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf7</UID>`	UUID
`</MPV:UNIQUE ID>`	
`<IMAGE_SIZE>`	
`<WIDTH>640</WIDTH>`	Image size
`<HEIGHT>480</HEIGHT>`	
`</IMAGE_SIZE>`	
`</MPV:BASIC IMAGE INFO>`	
`</MPV:BASIC IMAGE PARAM>`	
`</MPV:IMAGE>`	UUID
`</MPV:MID RES>`	
`<MPV:THUMB RES>`	
`<MPV:IMAGE MPV:ID="#0005">`	Thumbnail-res image
`<MPV:BASIC IMAGE PARAM>`	
`<MPV:BASIC IMAGE INFO>`	
`<FILE_FORMAT>`	
`<FILE NAME>/my`	
`pictures/screenres/image2.jpg</FILE NAME>`	File name
`<FORMAT_TYPE>JPG</FORMAT_TYPE>`	Format type
`</FILE FORMAT>`	
`<MPV:UNIQUE ID>`	

<UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf7</UID>	UUID
</MPV:UNIQUE_ID>	
<IMAGE SIZE>	
<WIDTH>160</WIDTH>	Image size
<HEIGHT>120</HEIGHT>	
</IMAGE SIZE>	
</MPV:BASIC IMAGE INFO>	
</MPV:BASIC_IMAGE_PARAM>	
</MPV:IMAGE>	
</MPV:THUMB RES>	
</MPV:VARIANTS>	
</MPV:IMAGE>	
<MPV:VIDEO MPV:ID="0007"> <!-- MPV metadata of VIDEO -->	Metadata of a video
<MPV:BASIC_VIDEO_PARAM>	
<MPV:BASIC VIDEO INFO>	
<FILE FORMAT>	
<FILE_NAME>/my pictures/VIDEO1.mpg</FILE_NAME>	Filename
<FORMAT TYPE>MPG</FORMAT TYPE>	Format type
</FILE FORMAT>	
<MPV:UNIQUE_ID>	
<UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf6</UID>	UUID
</MPV:UNIQUE ID>	
<IMAGE_SIZE>	
<WIDTH>352</WIDTH>	Video image size
<HEIGHT>240</HEIGHT>	
</IMAGE_SIZE>	
<MPV:MEDIA TYPE>stream/MPEG1VideoCD</MPV:MEDIA TYPE>	Media type
</MPV:BASIC VIDEO INFO>	
</MPV:BASIC_VIDEO_PARAM>	
<MPV:VIDEO CREATION>	
<MPV:GENERAL CREATION INFO>	
<CREATION_TIME>2001-05-15T10:13:29</CREATION_TIME>	Date taken
</MPV:GENERAL CREATION INFO >	
</MPV:VIDEO CREATION>	
<MPV:CONTENT_DESCRIPTION>	
<GROUP CAPTION>Trip to Waldo Lake Sept. 14, 2001 Smith & Jones Familes</GROUP CAPTION>	List caption
</MPV:CONTENT_DESCRIPTION>	
<MPV:VARIANTS>	
<MPV:THUMB RES>	Variants of video
<MPV:IMAGE MPV:ID="#0005">	Thumbnail image
<MPV:BASIC IMAGE PARAM>	
<MPV:BASIC IMAGE INFO>	
<FILE_FORMAT>	
<FILE NAME>/my pictures/screenres/video1.jpg</FILE NAME>	Filename
<FORMAT_TYPE>JPG</FORMAT_TYPE>	Format
</FILE FORMAT>	
<MPV:UNIQUE ID>	
<UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf7</UID>	UUID
</MPV:UNIQUE ID>	
<IMAGE SIZE>	
<WIDTH>160</WIDTH>	Image size
<HEIGHT>120</HEIGHT>	
</IMAGE SIZE>	
</MPV:BASIC_IMAGE_INFO>	
</MPV:BASIC IMAGE PARAM>	
</MPV:IMAGE>	

`</MPV:THUMB RES>`	
`</MPV:VARIANTS>`	
`</MPV:VIDEO>`	
`<MPV:VARIANTS>`	Variants of List
`<MPV:SLIDESHOW>`	Slideshow
`<MPV:VIDEO MPV:ID="#0005">`	Video
`<MPV:BASIC VIDEO PARAM>`	
`<MPV:BASIC_VIDEO_INFO>`	
`<FILE FORMAT>`	
`<FILE NAME>/my pictures/show/album.mpg</FILE NAME>`	File name
`<FORMAT_TYPE>MPG</FORMAT_TYPE>`	format type
`</FILE FORMAT>`	
`<MPV:UNIQUE ID>`	
`<UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf7</UID>`	UUID
`</MPV:UNIQUE ID>`	
`<IMAGE SIZE>`	
`<WIDTH>352</WIDTH>`	Video imag size
`<HEIGHT>240</HEIGHT>`	
`</IMAGE SIZE>`	
`<MPV:MEDIA_TYPE>stream/MPEG1VideoCD</MPV:MEDIA_TYPE>`	Media type
`</MPV:BASIC VIDEO INFO>`	
`</MPV:BASIC VIDEO PARAM>`	
`</MPV:VIDEO>`	
`</MPV:SLIDESHOW>`	
`</MPV:VARIANTS>`	
`</MPV:LIST>`	
`<MPV:TOC xmlns:MPV=”mpv-nsval” MPV:ID="0008">`	MPV Table of Contents
`<MPV:NS ENTRIES>`	Namespaces
`<MPV:NS_ENTRY NS-PREFIX=”” NS-URI=”<http://www.digitalimaging.org/dig35/1.1/xml>”/>`	DIG35 namespace
`<MPV:NS ENTRY NS-PREFIX=”MPV” NS-URI=” <http://www.osta.org/MPV/1.0/xml#>” />`	MPV namespace
`</MPV:NS ENTRIES>`	
`<MPV:METADATA OFFSETS>`	Offsets for random access
`[TODO]`	[TODO]
`</MPV:METADATA OFFSETS>`	
`</MPV:TOC>`	
`</MPV:MPV>`	
`<?MPV TOC-OFFSET=”2853” DOC-SIZE=”3333” MPV?>`	Processing Instruction

3.2.2 Data Pathnames

The example contains the following data items, as listed by element path. This view on the example makes it easy to see the structure of the file and what data items can be retrieved.

[TODO[

3.3 MPV Playlist Example

This example extends Example 1 by adding playlists. To do so, the list of pictures at the top-level is moved into a list of its own, and other playlists are added as peers. All three lists are children of the top-level list. The first playlist simply references content items in the other list; the second playlist references a list in another file.

3.3.1 XML Representation

This is as it would be written to a MPV file.

MPV playlists example	Notes
`<?XML version="1.0" >`	XML document
`<MPV:MPV xmlns=" <http://www.digitalimaging.org/dig35/1.1/xml>"`	MPV Wrapper
`xmlns:MPV="http://www.osta.org/MPV/1.0/xml#" >`	
`<MPV:LIST" MPV:ID="0001">`	A list of metadata
`... list metadata ...`	MPV metadata for this list
`<FILE NAME>trip-to-the-lake-Sept2001.mpv</CAPTION>`	Filename of this list
`... list metadata ...`	
`<MPV:LIST MPV:ID="0002">  <!-- MPV list -->`	Sub List 1: MPV Metadata
`... list metadata ...`	
`<CAPTION>Trip to Waldo Lake Sept. 14, 2001 Smith &`	List caption
`Jones Familes</CAPTION>`	
`<COMMENT>OrderedByDate</COMMENT>`	Comment
`... list metadata ...`	
`<METADATA MPV:ID="0003">  <!-- DIG35 metadata of an image -->`	Image 1 - DIG35 metadata
`... image metadata ...`	
`</METADATA>`	
`<MPV:IMAGE MPV:ID="0004">  <!-- MPV metadata of an image -->`	Image 2 - MPV metadata
`... image metadata ...`	
`<MPV:ASSOCIATED MPV:ID="0005">  <!-- MPV metadata assoc -->`	Associated item
`<MPV:AUDIO MPV:ID="0006">  <!-- MPV metadata of an audio -->`	MPV metadata of an audio
`... audio metadata ...`	
`</MPV:AUDIO>`	
`</MPV:ASSOCIATED>`	
`<MPV:ASSOCIATED MPV:ID="0007">`	Associated item
`<MPV:AUDIO MPV:ID="0008">  <!-- MPV metadata of an audio -->`	MPV metadata of an audio
`... audio metadata ...`	
`</MPV:AUDIO>`	
`</MPV:ASSOCIATED>`	
`</MPV:IMAGE>`	
`</MPV:IMAGE>`	
`<MPV:VIDEO MPV:ID="0009">  <!-- MPV metadata of a video -->`	MPV metadata of a video
`... video metadata ...`	
`</MPV:VIDEO>`	
`</MPV:LIST>`	
`<MPV:LIST MPV:ID="0010">  <!-- MPV list-->`	Sub List 2 : MPV metadata
`... (play)list metadata ...`	
`<CAPTION>Best Waldo Lake Trip Pictures</CAPTION>`	List caption
`<COMMENT>OrderedByCreator</COMMENT>`	Comment
`... (play)list metadata ...`	
`<MPV:ITEM_REF MPV:ID="0011" href="#0008"/>`	Ref to item #0008 – video
`<MPV:ITEM_REF MPV:ID="0012" href="#0003"/>`	Ref to item #0003 – image 1
`</MPV:LIST>`	
`<MPV:LIST MPV:ID="0013">  <!-- MPV list-->`	Sub List 3: MPV metadata
`... (play)list metadata ...`	
`<CAPTION>Aug. 2000 trip to the lake</CAPTION>`	List caption
`... (play)list metadata ...`	
`<MPV:ITEM_REF MPV:ID="0014" href="trip-to-lake-Aug2000.mpv"/>`	Ref to another MPV file
`</MPV:LIST>`	
`</MPV:LIST>`	
`<MPV:TOC xmlns:MPV="mpv-nsval">`	MPV Table of Contents
`<MPV:NS-ENTRIES>`	Namespaces
`<MPV:NS-ENTRY NS-PREFIX=""`	
`NS-URI="<http://www.digitalimaging.org/dig35/1.1/xml>"/>`	DIG35 namespace

`<MPV:NS-ENTRY NS-PREFIX="MPV"`	
`NS-URI=" <http://www.osta.org/MPV/1.0/xml#>" />`	MPV namespace
`</MPV:NS-ENTRIES>`	
`<MPV:METADATA-OFFSETS>`	Offsets for random access
`<MPV:OFFSET type="MPV:LIST" byteref="45">`	Top-level list
`<MPV:OFFSET type="MPV:LIST" byteref="545">`	Sub List 1
`<MPV:OFFSET type="DIG35:METADATA" byteref="705"/>`	Image 1
`<MPV:OFFSET type="MPV:IMAGE" byteref="1643"/>`	Image 2
`<MPV:OFFSET type="MPV:ASSOCIATED" byteref="1970">`	Associated
`<MPV:OFFSET type="MPV:AUDIO" byteref="2010"/>`	Image 2 Audio 1
`</MPV:OFFSET>`	
`<MPV:OFFSET type="MPV:ASSOCIATED" byteref="2020">`	Associated
`<MPV:OFFSET type="MPV:AUDIO" byteref="2320"/>`	Image 2 Audio 2
`</MPV:OFFSET>`	
`<MPV:OFFSET type="MPV:VIDEO" byteref="2500"/>`	Video
`</MPV:OFFSET>`	
`<MPV:OFFSET type="MPV:LIST" byteref="2945">`	Sub List 2
`<MPV:OFFSET type="MPV:ITEM-REF" byteref="3023"/>`	Item ref
`<MPV:OFFSET type="MPV:ITEM-REF" byteref="3054"/>`	Item ref
`</MPV:OFFSET>`	
`<MPV:OFFSET type="MPV:LIST" byteref="3085">`	Sub List 3
`<MPV:OFFSET type="MPV:ITEM-REF" byteref="3204"/>`	Item ref
`</MPV:OFFSET>`	
`</MPV:OFFSET>`	
`</MPV:METADATA-OFFSETS>`	
`</MPV:TOC>`	
`</MPV:MPV>`	
`<?MPV TOC-OFFSET="3227" DOC-SIZE="3536" MPV?>`	Processing Instruction

3.3.2 Data Pathnames

The example contains the following data items, as listed by element path. This view on the example makes it easy to see the structure of the file and what data items can be retrieved.

Info about this MPV list

```
/mpv:MPV/List/Basic List Params/Basic List Info/dig35:File Format/File Name
...
```

First Sub List

```
...
/mpv:MPV/List/List/Content Description/Caption
/mpv:MPV/List/List/Content_Description/Comment
...
```

First item – still image using pure DIG35 metadata

```
/mpv:MPV/List/List/dig35:Metadata/Basic_Image_Param/Basic_Image_Info/File_Format/File_Name
...
```

Second item – image with two associated audio files, using MPV metadata for images

```
/mpv:MPV/List/List/Image/Basic_Image_Param/Basic_Image_Info/dig35:File_Format/File_Name
...
```

Third item – video, using MPV metadata for video

```
/mpv:MPV/List/Video/Basic_Video_Param/Basic_Video_Info/dig35:File_Format/File_Name
...
```

Second Sub List

```
...
/mpv:MPV/List/List/Content_Description/Caption
/mpv:MPV/List/List/Content_Description/Comment
...
```

First item – item reference

```
/mpv:MPV/List/List/Item_Ref
```

Second item – item reference

```
/mpv:MPV/List/List/Item_Ref
```

Third Sub List

```
...
/mpv:MPV/List/List/Content_Description/Caption
/mpv:MPV/List/List/Content_Description/Comment
...
```

First item – item reference

```
/mpv:MPV/List/List/Item_Ref
```

MPV Table of Contents

```
/mpv:MPV/TOC/NS ENTRIES/NS ENTRY
/mpv:MPV/TOC/NS_ENTRIES/NS_ENTRY
/mpv:MPV/TOC/METADATA_OFFSETS/OFFSET
/mpv:MPV/TOC/METADATA OFFSETS/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA_OFFSETS/OFFSET/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA_OFFSETS/OFFSET/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA OFFSETS/OFFSET/OFFSET/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA_OFFSETS/OFFSET/OFFSET/OFFSET/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA_OFFSETS/OFFSET/OFFSET/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA OFFSETS/OFFSET/OFFSET/OFFSET/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA_OFFSETS/OFFSET/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA_OFFSETS/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA OFFSETS/OFFSET/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA_OFFSETS/OFFSET/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA_OFFSETS/OFFSET/OFFSET
/mpv:MPV/TOC/METADATA_OFFSETS/OFFSET/OFFSET/OFFSET
```

3.4 DIG35 Metadata Example

MPV uses DIG35's existing metadata to represent information about single still images. Here is an example of encoding typical image metadata using the DIG35 specification.

DIG35 metadata of a still image	Notes
`<?XML version="1.0" >`	Metadata of a single image
`<METADATA Type="Single">`	DIG35 metadata of an image
`<BASIC_IMAGE_PARAM>`	
`<BASIC IMAGE INFO>`	
`<FILE_FORMAT>`	
`<FILE_NAME>/my pictures/image.jpg</FILE_NAME>`	UUID
`<FORMAT TYPE>EXIF</FORMAT TYPE>`	

`</FILE FORMAT>`	
`<IMAGE_ID>`	
`<UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf6</UID>`	Width & height
`<ID TYPE>http://www.digitalimaging.org/dig35/UUID</ID TYPE>`	
`</IMAGE_ID>`	
`<IMAGE SIZE>`	
`<WIDTH>1600</WIDTH>`	File information
`<HEIGHT>1200</HEIGHT>`	
`</IMAGE SIZE>`	
`</BASIC IMAGE INFO>`	
`<COMPONENT_INFO>`	
`<COMPONENTS>RGB</COMPONENTS>`	Color depth
`<NUM COMPONENT>3</NUM COMPONENT>`	
`<COMP_SIZE>8</COMP_SIZE>`	
`<COMP SIZE>8</COMP SIZE>`	
`<COMP SIZE>8</COMP SIZE>`	
`</COMPONENT_INFO>`	
`</BASIC IMAGE PARAM>`	
`<IMAGE CREATION>`	
`<GENERAL_CREATION_INFO>`	
`<CREATION TIME>2001-05-14T21:43:23</CREATION TIME>`	Location
`<IMAGE CREATOR>`	
`<PERSON_NAME>`	
`<NAME COMP Type="Given">John</NAME COMP>`	Orientation
`<NAME COMP Type="Family">Smith</NAME COMP>`	
`</PERSON_NAME>`	
`</IMAGE CREATOR>`	
`</GENERAL CREATION INFO >`	
`<CAMERA_CAPTURE>`	Date taken
`<CAMERA SETTINGS>`	
`<CAMERA LOCATION>`	
`<COMMENT>At home</COMMENT>`	Image creator
`</CAMERA LOCATION>`	
`<ORIENTATION>`	
`<ROLL>90</ROLL>`	
`</ORIENTATION>`	
`</CAMERA SETTINGS>`	
`</IMAGE_CREATION>`	
`<CONTENT DESCRIPTION>`	Category
`<GROUP CAPTION>Trip to Waldo Lake</GROUP CAPTION>`	Title
`<CAPTION>Mary and the kids</CAPTION>`	Info
`<COMMENT>kids=Marty, John, Mike</COMMENT>`	
`</CONTENT_DESCRIPTION>`	
`</METADATA>`	

Chapter 4: Implementation Profiles

DIG35 and MPV are rich metadata standards. In this chapter we identify the key metadata expected to be used during playback and other uses, and the metadata that creators of the metadata should also populate.

The profiles are given as a collection of metadata paths to the data. These paths are used during parsing of the MPV metadata files to identify the desired data.

Each path component represents the name of a metadata element. Names like "mpv:LIST" mean the LIST element in the MPV specification, while "dig35:BASIC_IMAGE_PARAM" means the BASIC_IMAGE_PARAM element of the DIG35 specification. Lowercase is used for the specification name to enhance readability.

Understanding Metadata Paths

Pathnames are used to identify paths through a hierarchical metadata structure to specific data of interest.

Metadata path	Usage
`/mpv:MPV/mpv:LIST`	**Absolute paths** begin with a "/" and are rooted at the outside context of the XML document.
`dig35:FILE_FORMAT/dig35:FILE_NAME`	**Relative paths** apply relative to a containing context. They do NOT begin with a "/". They can be inserted anywhere the first segment names of the path matches the last segment name of another path.
`dig35:BASIC_IMAGE_PARAM/dig35:BASIC_IMAGE_INFO/<dig35:FILE_FORMAT>`	**Path substitution** is applied to a path when the element name is given in angle brackets "<xxx>". Any relative path beginning with that name can be substituted.
`namespace:ElementName/Element2Name`	**Namespaces** and **element names** are separated by a colon. The same element name can occur in different name spaces and thus refer to different things. Namespaces are typically written in lowercase for readability. Element names farther down a path that do not have a namespace are part of the same namespace as their parent element.

4.1 Device Playback (e.g. DVD Players)

[TODO]

The following paths are targeted at playback applications in a device context.

Metadata path	**Occurs**	**Usage**
[TODO] – lots still to do here		

4.2 Computer Software Playback (e.g. PCs)

[TODO] Touch on:

- Original Files
- Printing
- Internet Service Connectivity

4.3 Device Creation (e.g. Cameras)

[TODO] Touch on:

- Picturelists & Playlists

4.4 Software Creation (e.g. PCs)

[TODO] Touch on:

- Picturelists & Playlists

Chapter 5: DIG35 Image Metadata Overview

This chapter contains wholesale excerpts from the DIG35 specification and are used without further specific identification. Minor modifications have been made to clarify or provide additional information.

5.1 Organizational Overview

[[TODO: this section probably should be removed from the specification – leave for now to provide useful background.]]

The DIG35 specification originated under the auspices of the Digital Imaging Group (DIG), a not-for-profit imaging industry consortium of manufacturers, developers, and users of digital imaging solutions. Originally founded in 1997 by Adobe, Canon, Eastman Kodak, Fuji, Hewlett-Packard, IBM, Intel, Live Picture and Microsoft, the organization grew to over 70 members at three different levels of membership by early 2000. The primary goal of the Digital Imaging Group was to provide an open industry forum for the exploration, implementation and stewardship of technologies and methods for expanding the Digital Imaging market.

The DIG and PIMA (the Photographic and Imaging Manufacturers Association) merged in 2001 to form the International Imaging Industry Association (I3A), together representing almost 60 years of combined imaging industry leadership. The I3A is the largest imaging industry group worldwide. The approximately 90 member companies of this not-for-profit association are dedicated to developing and promoting the adoption of open industry standards, addressing environmental issues and providing a voice for the industry that will ultimately benefit all users.

As the central forum for the development and advancement of open standards within the imaging industry, I3A takes a broad view of the entire imaging market value chain. In doing so, I3A seeks to identify strategic market growth barriers, where a strong industry consortium has the greatest potential for impact, then targets specific initiatives or standards projects to reduce or eliminate those barriers. I3A uses a combination of technical and marketing activities to reach its goals of solving interoperability challenges, resolving infrastructure issues and minimizing overall market confusion.

http://www.i3a.org/

The members of the **DIG35** Initiative Group are (in alphabetical order):

Adobe Systems Inc., AGFA-GEVAERT N.V., Canon Inc., Digitella Technology Inc., Eastman Kodak Company, Fuji Photo Film Co., Ltd., Hewlett Packard Company, Microsoft Corporation, NETIMAGE, PhotoChannel Networks Inc., PictureIQ Corporation, Polaroid Corporation, Seattle FilmWorks, Inc.

Kodak have contributed an open-source toolkit for reading and writing multiple picture metadata formats, including DIG35. More information is available at
http://www.kodak.com/US/en/developers/productsTechnologies/prodTechMeta.shtml and
http://sourceforge.net/projects/picturemetadata

5.2 DIG35 Overview

The DIG35 specification includes the following major pieces:

- basic image parameter metadata
- image creation metadata
- content description metadata

- history metadata
- intellectual property rights metadata
- fundamental metadata types and fields
- DIG35 XML document definition, schema, and DTD
- metadata association
- embedding DIG35 metadata in file formats
- conformance

Image metadata is a "building block" for digital imaging that may be used within the wide spectrum of the imaging workflow. This specification defines a standard set of image metadata based on a generic concept that may be further divided into conceptual "sub-blocks." Each of these sub-blocks describes a unique aspect of the image. By partitioning metadata into discrete sub-blocks, developers may extend a particular sub-block without affecting the entire architecture thereby ensuring semantic interoperability while allowing vendors to add value to the metadata and image data itself. The DIG35 metadata sub-blocks are discussed in detail in the next section.

Note that procedural metadata are out of scope of this specification. Examples of procedural metadata include;

- Parameters for image processing that change the visual appearance by arbitrary cropping, rotation, or other transformations.
- Order information to specify product and quantity.
- E-commerce information such as billing or payment data or delivery addresses

The entire content of the DIG35 standard can be applied to both a single picture or a collection of pictures.

5.3 DIG35 Metadata Blocks

DIG35's metadata definition consists of five logical blocks with a separate common definition that is referred to by other blocks. While each block is logically partitioned, they may be linked to each other to form additional semantics.

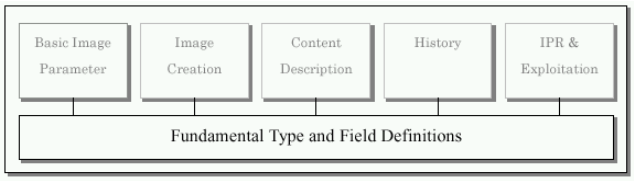

Figure F-1: High-level structure of the Fundamental Type and Field metadata

5.3.1 Basic Image Parameter Metadata

Since this specification is a general-purpose metadata standard, it must be applicable to the broadest possible class of file formats. Since each file format makes distinct decisions regarding what elements are important as header information, it is impossible to delegate the specification of header metadata to file format developers. In fact, this specification takes the opposite approach and assumes the existence of a file format that contains no header information. This assumption ensures that any format may be transcoded into another file-format.

In order to do this, a block of metadata is defined that contains information similar to, and identical in use to, file header metadata. There should never be conflicts between this sub-block and the file header metadata as this sub-block is intended to be used, as stated above, *only* when there is no file header metadata. However, if there is a conflict between the file-format header information and the Basic Image Parameter Metadata, the file header should always take precedence.

Note: Each file format defines its own header information that is very dependent on the features the format supports. Those features are specific on the image data and thus out of scope of this specification. This metadata should be considered informational and not be used to decode the image data stored in the associated file.

5.3.2 Image Creation Metadata

The Image Creation Metadata defines the "how" metadata that specifies the source of which the image was created. For example, the camera and lens information and capture condition are useful technical information for professional and serious amateur photographers as well as advanced imaging applications.

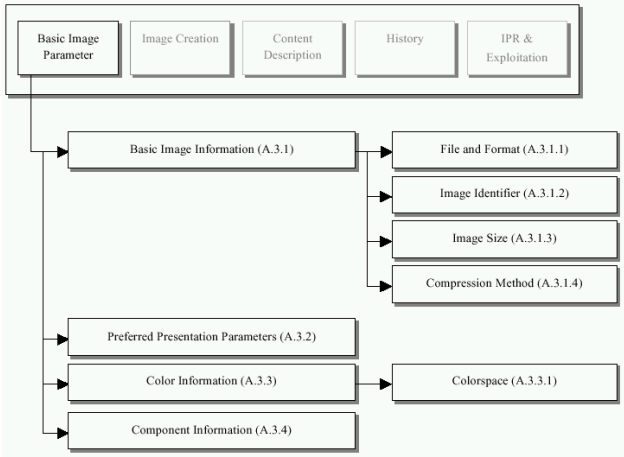

Figure A-1: Basic Image Parameter metadata structure

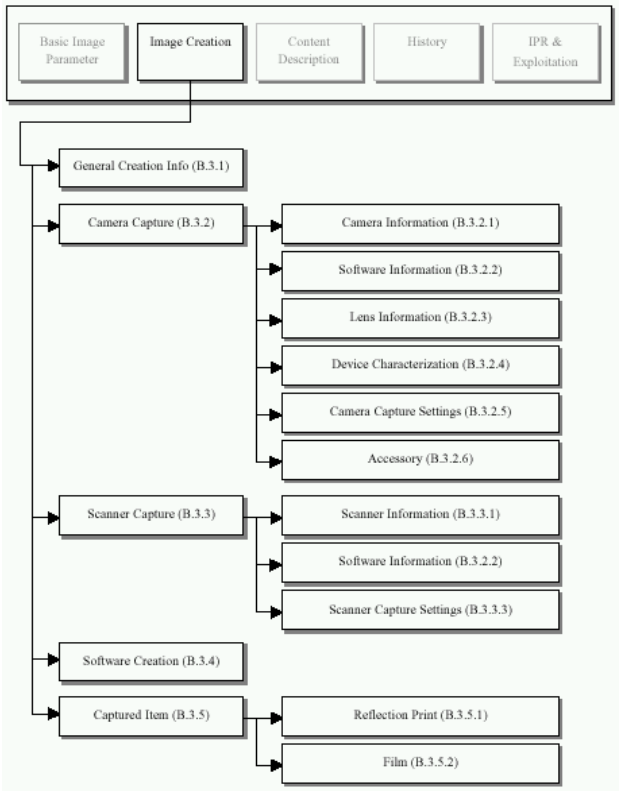

Figure B-1: Image Creation metadata structure

5.3.3 Content Description Metadata

The Content Description Metadata defines the descriptive information of "who", "what", "when" and "where" aspect of the image. Often this metadata takes the form of extensive words, phrases, or sentences to describe a particular event or location that the image illustrates. Typically, this metadata consists of text that the user enters, either when the images are taken or scanned or later in the process during manipulation or use of the images.

5.3.4 History Metadata

The History Metadata is used to provide *partial information* about *how* the image got to the present state. For example, history may include certain processing steps that have been applied to an image. Another example of a history would be the image creation events including digital capture, exposure of negative or reversal films, creation of prints, transmissive scans of negatives or positive film, or reflective scans of prints. All of this metadata is important for some applications. To permit flexibility in construction of the image history metadata, two alternate representations of the history are permitted. In the first, the history metadata is embedded in the image metadata. In the second, the previous versions of the image, represented as a URL/URI, are included in the history metadata as pointers to the location of the actual history. The history metadata for a composite image (i.e., created from two or more previous images) may also be represented through a hierarchical metadata structure. While this specification does not define the "how" or "how much" part of the processing aspect, it does enable logging of certain processing steps applied to an image as hints for future use.

Note: Neither the processing nor the history metadata specify the actual processing steps or the compositing steps.

Such metadata is reserved for further study.

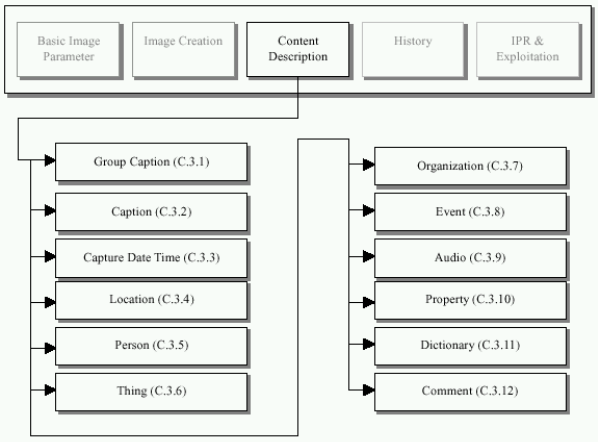

Figure C-1: Content Description metadata structure

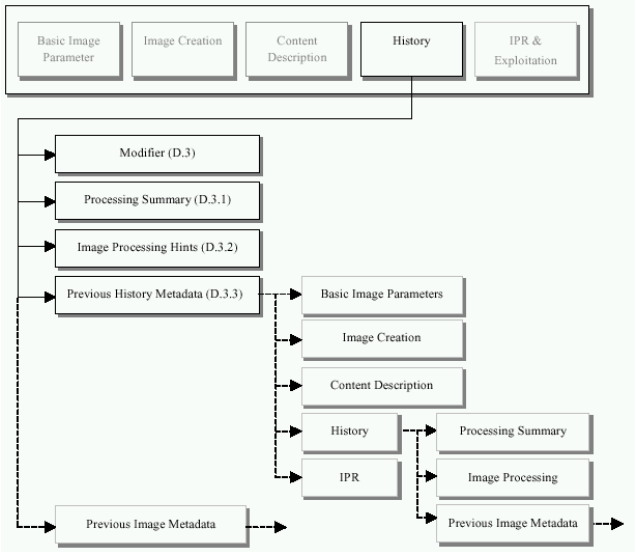

Figure D-1: History metadata structure

5.3.5 Intellectual Property Rights (IPR) Metadata

The Intellectual Property Rights Metadata (IPR) defines metadata to either protect the rights of the owner of the image or provide further information to request permission to use it. It is important for developers and users to understand the implications of intellectual property and copyright information on digital images to properly protect the rights of the owner of the image data.

Note: Several international organizations are looking at standards in the area of intellectual property rights.

5.3.6 Fundamental Metadata Types and Fields

The Fundamental Metadata Types define the format field defined in all metadata sub-blocks. Those may include a collection of metadata elements such as an address or a representation of an attribute of other elements such as the language. The Fundamental Metadata Fields define fields used in all metadata sub-blocks. These include a definition for language specification and timestamps.

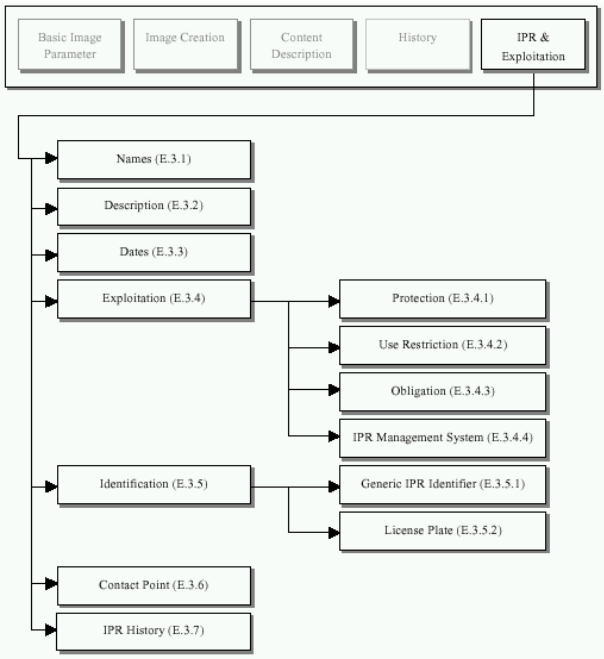

Figure E-1: IPR metadata structure

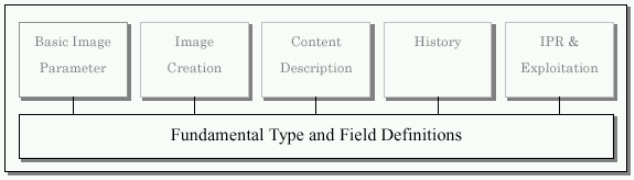

Figure F-1: High-level structure of the Fundamental Type and Field metadata

Chapter 6: MultiPhotoVideo Metadata Overview

To Do

Annex A: MultiPhotoVideo Document and List

DIG35 defines metadata for still images and has a rudimentary ability to associated audio files with the still image.

MPV extends DIG35 by defining content types for:

- images
- video
- audio
- documents
- improved ability to associate content items of any type to a content item or collection.

A.1 DIG35 Spec Review

DIG35 is a metadata specification for still images. All the metadata for still images is contained within the <METADATA> element.

There is no explicit statement of content type; it is a statement of quantity, i.e. Type="Single", not Type="Image". Content type is implicit by virtue of the element names, e.g. BASIC_IMAGE_PARAM, IMAGE_CREATION. This is a limitation of the DIG35 design that impacts the MPV approach to defining additional content types.

```
Figure: Example of DIG35 still image metadata

<METADATA TYPE="Single">
   <BASIC IMAGE PARAM>
      ...
   </BASIC_IMAGE_PARAM>
   <IMAGE CREATION>
      ...
   </IMAGE_CREATION>
   <CONTENT DESCRIPTION>
      ...
   </CONTENT_DESCRIPTION>
   <HISTORY>
      ...
   </HISTORY>
   <IPR>
      ...
   </IPR>
</METADATA>
```

```
Figure: DIG35 still image metadata schema

  <xsd:element name="METADATA">
    <xsd:annotation>
      <xsd:documentation>
          See section Annex G: DIG35 XML Document Definition
      </xsd:documentation>
    </xsd:annotation>
```

```
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:BASIC IMAGE PARAM" minOccurs="0"/>
        <xsd:element ref="dig35:IMAGE CREATION" minOccurs="0"/>
        <xsd:element ref="dig35:CONTENT DESCRIPTION" minOccurs="0"/>
        <xsd:element ref="dig35:HISTORY" minOccurs="0"/>
        <xsd:element ref="dig35:IPR" minOccurs="0"/>
        <xsd:element ref="dig35:METADATA" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute name="TYPE" use="optional" default="Single">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="Single"/>
            <xsd:enumeration value="Collection"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang" use="optional" default="en"/>
    </xsd:complexType>
  </xsd:element>
```

A.2 Design Approach

MPV supports DIG35 metadata for still images in it's entirety. But DIG35 itself is too constrained to directly insert support for additional media types.

Instead, MPV defines a new top-level structure for the metadata that is closely modeled on DIG35 and can include DIG35 metadata within it when describing still images. This means that it is straightforward to support metadata arriving in DIG35 format. It also means that MPV is a natural extension to DIG35 to support additional content types and could be considered for adoption as an extension to the DIG35 standard by the I3A.

A MPV document consists of a list of metadata about content items. The list may contain any number of metadata about specific content items with given media types and it may contain additional lists.

Each content type has it's own set of metadata. These are structured similarly and share many common elements.

An MPV document has an optional table of contents. This is defined elsewhere in the specification.

A.2.1 MPV Specification

A.2.2 MPV:MPV – the Wrapper

```
MPV Schema definition

  <xsd:element name="mpv:MPV">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="mpv:LIST" minOccurs="0" maxOccurs="1"/>
        <xsd:element ref="mpv:TOC" minOccurs="0" maxOccurs="1"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
```



```
<MPV:MPV xmlns=" <http://www.digitalimaging.org/dig35/1.1/xml>"
         xmlns:MPV="http://www.osta.org/MPV/1.0/xml#" >
  ...
</MPV:MPV>
```

A.2.3 MPV:LIST – the Top Level of MPV Metadata

The MPV list contains all the metadata about content items and can also contain additional lists. Note how the DIG35 metadata element is a peer and an alternative to the MPV:IMAGE element.

```
MPV Schema definition

  <xsd:element name="mpv:LIST">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="mpv:BASIC_LIST_PARAM" minOccurs="0"/>
        <xsd:element ref="mpv:LIST_CREATION" minOccurs="0"/>
        <xsd:element ref="mpv:CONTENT_DESCRIPTION" minOccurs="0"/>
        <xsd:element ref="mpv:HISTORY" minOccurs="0"/>
        <xsd:element ref="dig35:IPR" minOccurs="0"/>
        <xsd:element ref="mpv:BRAND_IDENTITY" minOccurs="0"/>
        <xsd:element ref="mpv:VARIANTS" minOccurs="0"/>
        <xsd:group ref="ITEM" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute ref="mpv:ID"/>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang" use="optional" default="en"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:group name="ITEM">
    <xsd:complexType>
       <xsd:choice>
         <xsd:element ref="dig35:METADATA"/>
         <xsd:element ref="mpv:IMAGE"/>
         <xsd:element ref="mpv:VIDEO"/>
         <xsd:element ref="mpv:AUDIO"/>
         <xsd:element ref="mpv:DOCUMENT"/>
         <xsd:element ref="mpv:ITEM-REF"/>
         <xsd:element ref="mpv:LIST"/>
       </xsd:choice>
    </xsd:complexType>
  </xsd:element>
```

In this example, the MPV list metadata provides the following information:

- unique id for the list itself
- name of the file holding the list
- time list was created
- creator of the list
- caption of the list

```
<MPV:MPV xmlns=" <http://www.digitalimaging.org/dig35/1.1/xml>"
         xmlns:MPV="http://www.osta.org/MPV/1.0/xml#" >
  <MPV:LIST" MPV:ID="0001">
    <MPV:BASIC_LIST_PARAM>
      <MPV:BASIC_LIST_INFO>
```



```
        <MPV:UNIQUE ID>
          <UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf6</UID>
        </MPV:UNIQUE_ID>
        <FILE FORMAT>
          <FILE NAME>Trip-to-the-Lake-Sept2001.MPV</FILE NAME>
          <FORMAT_TYPE>MPV</FORMAT_TYPE>
        </FILE FORMAT>
      </MPV:BASIC LIST INFO>
    </MPV:BASIC_LIST_PARAM>
    <MPV:LIST CREATION>
      <MPV:GENERAL CREATION INFO>
        <CREATION_TIME>2001-05-14T21:43:23</CREATION_TIME>
        <MPV:CREATOR>
          <PERSON NAME>
            <NAME_COMP Type="Given">John</NAME_COMP>
            <NAME COMP Type="Family">Smith</NAME COMP>
          </PERSON NAME>
        </MPV:CREATOR>
      </MPV:GENERAL CREATION INFO>
    </MPV:LIST CREATION
    <CONTENT_DESCRIPTION>
      <CAPTION>Trip to Waldo Lake</CAPTION>
      <CAPTION>Sept. 14, 2001</CAPTION>
      <CAPTION>Smith & Jones Families</CAPTION>
    </CONTENT DESCRIPTION>
  </MPV:LIST>
</MPV:MPV>
```

In this example, the MPV metadata list contains a DIG35 image, a MPV image, and a MPV video clip.

```
<?XML version="1.0" >
<MPV:MPV xmlns=" <http://www.digitalimaging.org/dig35/1.1/xml>"
         xmlns:MPV="http://www.osta.org/MPV/1.0/xml#" >
  <MPV:LIST" MPV:ID="0001">
       ... list metadata ...
           <FILE_NAME>trip-to-the-lake-Sept2001.mpv</CAPTION>
       ... list metadata ...
      <METADATA MPV:ID="0002">  <!-- DIG35 metadata of an image -->
         ... image metadata ...
      </METADATA>
      <MPV:IMAGE MPV:ID="0003">  <!-- MPV metadata of an image -->
         ... image metadata ...
      </MPV:IMAGE>
      <MPV:VIDEO MPV:ID="0004">  <!-- MPV metadata of a video -->
         ... video metadata ...
      </MPV:VIDEO>
    </MPV:LIST>
</MPV:MPV>
```

A.2.4 MPV:BASIC_LIST_PARAM

Changed dig35:BASIC_IMAGE_PARAM to remove image-related parameters and use mpv:BASIC_LIST_INFO.

```
  <xsd:element name="mpv:BASIC_LIST_PARAM">
    <xsd:annotation>
      <xsd:documentation>
      </xsd:documentation>
```

```
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="mpv:BASIC_LIST_INFO" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
```

A.2.5 MPV:BASIC_LIST_INFO

Changed dig35:BASIC_IMAGE_INFO to use mpv:UNIQUE_ID and mpv:MEDIA_TYPE and remove image-related elements.

```
  <xsd:element name="mpv:BASIC_LIST_INFO">
    <xsd:annotation>
      <xsd:documentation>
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:FILE_FORMAT" minOccurs="0"/>
        <xsd:element ref="mpv:UNIQUE_ID" minOccurs="0"/>
        <xsd:element ref="mpv:MEDIA_TYPE" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
```

A.2.6 MPV:LIST_CREATION

Changed dig35:IMAGE_CREATION to use mpv:GENERATION_CREATION_INFO and deleted dig35:CAPTURED_ITEM.

```
  <xsd:element name="mpv:LIST_CREATION">
    <xsd:annotation>
      <xsd:documentation>
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="mpv:GENERAL_CREATION_INFO" minOccurs="0"/>
        <xsd:element ref="dig35:CAMERA_CAPTURE" minOccurs="0"/>
        <xsd:element ref="dig35:SCANNER_CAPTURE" minOccurs="0"/>
        <xsd:element ref="dig35:SOFTWARE_CREATION" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
```


Annex B: MPV Common Types

B.1 MPV Specification

B.1.1 MPV:ID Attribute

```
<xsd:attribute name="mpv:ID" type="xsd:id">
  <xsd:annotation>
    <xsd:documentation>
              An ID to allow cross-references.
    </xsd:documentation>
  </xsd:annotation>
</xsd:attribute>
```

B.1.2 MPV:UNIQUE_ID

Unique Identifier specifies an identifier that must uniquely identify the item(s) which bear them. The format may be globally unique (e.g. UUID), vendor or application dependent. This field may contain the sub-fields listed below.

This element has identical structure to DIG35:IMAGE_ID. However, to add generality, MPV renames it to UNIQUE_ID and defines the default ID_TYPE.

```
Schema Definition:

<xsd:element name="mpv:UNIQUE_ID">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="mpv:UID"              type="xsd:string" minOccurs="0"/>
      <xsd:element name="mpv:ID_TYPE"          type="xsd:anyURI" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

Unique Identifier: This field specifies the unique identifier of an image. The Unique Identifier Type specifies the format of the field.

Unique Identifier Type: This field specifies the type of the Unique Identifier.

The DIG35 specification defined the following value for this field.

"http://www.digitalimaging.org/dig35/UUID"

This value specifies that the contents of this field in a UUID as defined by ISO/IEC 11578. Other vendors may define other values.

The default value of ID_TYPE is "http://www.digitalimaging.org/dig35/UUID".

```
Example:

<UNIQUE_ID>
  <UID>EC340B04-74C5-11D4-A729-879EA3548F0E</UID>
</UNIQUE_ID>
```


B.1.3 MPV:MEDIA_TYPE

MPV defines a rich set of media types that can be used describe in some detail the type of media stream. Note that media type is quite distinct from file format; it describes the format of the media stream within the file. Many file formats are container formats that can contain a variety of media types.

Each content type (IMAGE, VIDEO, AUDIO, DOCUMENT) defined by MPV suggest a standard set of strings for the MEDIA_TYPE values.

```
MPV Schema Definition:

<xsd:element name="mpv:MEDIA_TYPE" type="xsd:string">
    <xsd:annotation>
      <xsd:documentation>
      </xsd:documentation>
    </xsd:annotation>
  </xsd:element>
```

Media Type: This field specifies the major and minor types of the media contained in the file. The values are case insensitive and separated by a slash "/", as in "image/EXIF". Note that the media type values are NOT the same as MIME type values. Suggested values are defined by the MPV specification.

B.1.4 MPV Suggested Values for DIG35 Elements

B.1.4.1 dig35:FILE_FORMAT

The DIG35 definition of FILE_FORMAT is reused in MPV, but the set of suggested element values is substantially increased.

```
DIG35 Schema definition

  <xsd:element name="FILE FORMAT">
    <xsd:annotation>
      <xsd:documentation>
                See section A.3.1.1  File and Format
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="FILE NAME" type="xsd:anyURI" minOccurs="0"/>
        <xsd:element name="FORMAT TYPE" type="xsd:string" minOccurs="0"/>
        <xsd:element name="MIME_TYPE" type="xsd:string" minOccurs="0"/>
        <xsd:element name="VERSION" type="xsd:string" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
```


Annex C: Image Metadata

C.1 DIG35 Spec Review

DIG35 is entirely defined around metadata for still images.

```
Figure: DIG35 still image metadata schema

  <xsd:element name="METADATA">
    <xsd:annotation>
      <xsd:documentation>
          See section Annex G: DIG35 XML Document Definition
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:BASIC IMAGE PARAM" minOccurs="0"/>
        <xsd:element ref="dig35:IMAGE CREATION" minOccurs="0"/>
        <xsd:element ref="dig35:CONTENT DESCRIPTION" minOccurs="0"/>
        <xsd:element ref="dig35:HISTORY" minOccurs="0"/>
        <xsd:element ref="dig35:IPR" minOccurs="0"/>
        <xsd:element ref="dig35:METADATA" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute name="TYPE" use="optional" default="Single">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="Single"/>
            <xsd:enumeration value="Collection"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang" use="optional" default="en"/>
    </xsd:complexType>
  </xsd:element>
```

```
Figure: DIG35 still image metadata schema

  <xsd:element name="BASIC IMAGE PARAM">
    <xsd:annotation>
      <xsd:documentation>
                See section Annex A:  Basic Image Parameter Metadata
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:BASIC IMAGE INFO" minOccurs="0"/>
        <xsd:element ref="dig35:PREF PRESENTATION PARAM" minOccurs="0"/>
        <xsd:element ref="dig35:COLOR INFO" minOccurs="0"/>
        <xsd:element ref="dig35:COMPONENT INFO" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
```



```
    </xsd:complexType>
  </xsd:element>
```

```
Figure: DIG35 still image metadata schema

  <xsd:element name="IMAGE CREATION">
    <xsd:annotation>
      <xsd:documentation>
                See section Annex B:  Image Creation Metadata
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:GENERAL CREATION INFO" minOccurs="0"/>
        <xsd:element ref="dig35:CAMERA CAPTURE" minOccurs="0"/>
        <xsd:element ref="dig35:SCANNER CAPTURE" minOccurs="0"/>
        <xsd:element ref="dig35:SOFTWARE CREATION" minOccurs="0"/>
        <xsd:element ref="dig35:CAPTURED ITEM" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
```

The only alternate media type referenced by the specification is the metadata for audio content associated with a still image. This occurs by using one or more <AUDIO> elements in the Content Description metadata block <CONTENT_DESCRIPTION> … </CONTENT_DESCRIPTION>. The <AUDIO> element refers to the audio content using a URI (<AUDIO_STREAM>) that is typed according to <AUDIO_FORMAT>.

```
Figure: DIG35 content description audio metadata schema

  <xsd:element name="AUDIO">
    <xsd:annotation>
      <xsd:documentation>
                See section C.3.9  Audio
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="AUDIO STREAM" type="xsd:anyURI"/>
        <xsd:element name="AUDIO FORMAT" type="xsd:string" minOccurs="0"/>
        <xsd:element name="MIME TYPE" type="xsd:string" minOccurs="0"/>
        <xsd:element name="DESCRIPTION" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element ref="dig35:COMMENT" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
```

C.2 Design Approach

The MPV:LIST item accepts DIG35 METADATA elements directly without change. This allows a MPV structure to insert DIG35 content directly without further processing.

For newly generated content, the desired approach is to use the MPV:IMAGE structure. MPV schema is a variations on the DIG35 schema that uses MPV common types. The AUDIO element is removed in

mpv:CONTENT_DESCRIPTION. Instead, associated content items are specified using the mpv:LIST element of mpv:CONTENT_DESCRIPTION.

C.3 MPV Specification

C.3.1 MPV:IMAGE

The top-level MPV metadata element for images. Very similar to dig35:METADATA.

```
<xsd:element name="mpv:IMAGE">
  <xsd:complexType>
    <xsd:sequence>
       <xsd:element ref="mpv:BASIC_IMAGE_PARAM" minOccurs="0"/>
       <xsd:element ref="mpv:IMAGE_CREATION" minOccurs="0"/>
       <xsd:element ref="mpv:CONTENT_DESCRIPTION" minOccurs="0"/>
       <xsd:element ref="mpv:HISTORY" minOccurs="0"/>
       <xsd:element ref="dig35:IPR" minOccurs="0"/>
      <xsd:element ref="mpv:VARIANTS" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

C.3.2 MPV:BASIC_IMAGE_PARAM

Use the MPV Basic Image Info, not the DIG35 one.

```
MPV Schema definition

  <xsd:element name="mpv:BASIC_IMAGE_PARAM">
    <xsd:annotation>
      <xsd:documentation>
                See section Annex A:  Basic Image Parameter Metadata
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="mpv:BASIC_IMAGE_INFO" minOccurs="0"/>
        <xsd:element ref="dig35:PREF_PRESENTATION_PARAM" minOccurs="0"/>
        <xsd:element ref="dig35:COLOR_INFO" minOccurs="0"/>
        <xsd:element ref="dig35:COMPONENT_INFO" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
```

C.3.3 MPV:BASIC_IMAGE_INFO

Replace dig35:IMAGE_ID with mpv:UNIQUE_ID and dig35:COMPRESION with mpv:MEDIA_TYPE.

```
<xsd:element name="mpv:BASIC_IMAGE_INFO">
  <xsd:annotation>
    <xsd:documentation>
    </xsd:documentation>
```



```
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:FILE_FORMAT" minOccurs="0"/>
        <xsd:element ref="mpv:UNIQUE_ID" minOccurs="0"/>
        <xsd:element ref="dig35:IMAGE_SIZE" minOccurs="0"/>
        <xsd:element ref="mpv:MEDIA_TYPE" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
```

File and Format: This field specifies the file name and format of the video data.

Unique Identifier: This field specifies a unique identifier of the item.

Image Size: This field specifies the width and height of the video.

Media Type: This field specifies the media type of the video data.

C.3.4 Suggested Values for FILE_FORMAT for Images

File format specifies the format and name of the video file associated with the metadata and uses the same structure as DIG35. Additional suggested format types are defined that are relevant to video formats.

File Format Type: This field specifies the file format of the video. This is frequently a container format, in which media conforming to MEDIA_TYPE and MEDIA_SUBTYPE are found. The values are case insensitive.

Table: Suggested File Format Type values

Format Types	Description

MIME Type: This field specifies the Internet media type of the video file. The following are some common values.

MIME Types	Description

C.3.5 Suggested Values for MPV:MEDIA_TYPE for Images

MPV defines a rich set of media types that can be used describe in some detail the type of media stream. Note that media type is quite distinct from file format; it describes the format of the media stream within the file. Many file formats are container formats that can contain a variety of media types.

Major Media Type: This field specifis the major type of the media contained in the file.

Table: Suggested Major Media Type values.

Major Media Types	**Description**
Image	

Minor Media Type: This field specifies the minor type of the media contained in the file. The values are case insensitive.

Note that some of the minor media types contain color component information. This should not be relied upon for classifying according to color information. The preferred mechanism to record and use color information when it is available with the COLOR_INFO and COMPONENT_INFO elements.

Table: Suggested Minor Media Type values when Major Media Type is "Image".

Minor Media Types for Major Type = "Image"	**Description**

C.3.6 MPV:IMAGE_CREATION

Uses the MPV General Creation Info instead of dig35:GENERAL_CREATION_INFO.

```
MPV Schema definition

  <xsd:element name="mpv:IMAGE_CREATION">
    <xsd:annotation>
      <xsd:documentation>
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="mpv:GENERAL_CREATION_INFO" minOccurs="0"/>
        <xsd:element ref="dig35:CAMERA_CAPTURE" minOccurs="0"/>
        <xsd:element ref="dig35:SCANNER_CAPTURE" minOccurs="0"/>
        <xsd:element ref="dig35:SOFTWARE_CREATION" minOccurs="0"/>
        <xsd:element ref="dig35:CAPTURED_ITEM" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
```

C.3.7 Example: MPV Image With Associated Audio

MPV example of a lit of content items	Notes
`<?XML version="1.0" >`	XML document
`<MPV:MPV xmlns=" <http://www.digitalimaging.org/dig35/1.1/xml>"`	MPV Wrapper
`xmlns:MPV="http://www.osta.org/MPV/1.0/xml#" >`	
`<MPV:LIST" MPV:ID="0001">`	A list of metadata
`<MPV:IMAGE MPV:ID="0003">  <!-- MPV metadata of an image -->`	MPV metadata of an image
`<MPV:BASIC_IMAGE_PARAM>`	
`<MPV:BASIC IMAGE INFO>`	
`<FILE_FORMAT>`	

`<FILE NAME>/my pictures/image2.jpg</FILE NAME>`
`<FORMAT_TYPE>EXIF</FORMAT_TYPE>`
`</FILE FORMAT>`
	UUID

`<MPV:UNIQUE ID>`
`<UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf6</UID>`
`</MPV:UNIQUE ID>`
	Width & height

`<IMAGE SIZE>`
`<WIDTH>1600</WIDTH>`
`<HEIGHT>1200</HEIGHT>`
`</IMAGE SIZE>`
`<MPV:MEDIA_TYPE>image/EXIF</MPV:MEDIA_TYPE>`
	File information

`</MPV:BASIC IMAGE INFO>` `</MPV:BASIC IMAGE PARAM>`	Media type

`<MPV:IMAGE_CREATION>`
`...`
`<MPV:IMAGE CREATION>`	

`<MPV:CONTENT_DESCRIPTION>`
`<MPV:LIST>`
`<MPV:AUDIO>`
`<MPV:BASIC_AUDIO_PARAM>`
	Creation metadata

`<MPV:BASIC AUDIO INFO>`	Content metadata

`<FILE FORMAT>`
`<FILE_NAME>/my pictures/image2.wav</FILE_NAME>`
`<FORMAT TYPE>WAV</FORMAT TYPE>`
`</FILE FORMAT>`
`</MPV:BASIC_AUDIO_INFO>`
`</MPV:BASIC AUDIO PARAM>`
`<MPV:CONTENT DESCRIPTION>`
`<COMMENT>Ambient sound</COMMENT>`
`</MPV:CONTENT DESCRIPTION>`
`</MPV:AUDIO>`
`</MPV:LIST>`
`</MPV:CONTENT DESCRIPTION>`
`</MPV:IMAGE>`
`</MPV:LIST>`
	Associated audio files

Annex D: Video Metadata

D.1 MPV Specification

D.1.1 MPV:VIDEO

```
<xsd:element name="mpv:VIDEO">
  <xsd:complexType>
    <xsd:sequence>
        <xsd:element ref="mpv:BASIC_VIDEO_PARAM" minOccurs="0"/>
        <xsd:element ref="mpv:VIDEO_CREATION" minOccurs="0"/>
        <xsd:element ref="mpv:CONTENT_DESCRIPTION" minOccurs="0"/>
        <xsd:element ref="mpv:HISTORY" minOccurs="0"/>
        <xsd:element ref="dig35:IPR" minOccurs="0"/>
        <xsd:element ref="mpv:VARIANTS" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang" use="optional" default="en"/>
  </xsd:complexType>
</xsd:element>
```

D.1.2 MPV:BASIC_VIDEO_PARAM

This section defines basic video parameter metadata that contains generic information about the video, such as the video compression, size, length, and framerate. While this metadata that are commonly referred to as the "Header Data" of a video file format, the scope of this section is to define metadata that is file format independent. Thus this metadata should be considered informational and not to be used to decode the video data stored in the associated file. The metadata fields would be considered secondary to the video data.

The metadata defined in this section is useful where the metadata is stored externally to the video file. For example, the length of the video may be determined without interrogating the video file.

Basic Video Parameter Metadata may contain the sub-fields listed below.

```
MPV Schema Definition:

<xsd:element name="mpv:BASIC_VIDEO_PARAM">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="mpv:BASIC_VIDEO_INFO"  minOccurs="0"/>
      <xsd:element ref="dig35:PREF_PRESENTATION_PARAM"          minOccurs="0"/>
      <xsd:element ref="dig35:COLOR_INFO"                       minOccurs="0"/>
      <xsd:element ref="dig35:COMPONENT_INFO"  minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang"/>
  </xsd:complexType>
</xsd:element>
```

Basic Video Information: This field specifies generic information about the video.

Preferred Presentation Parameters: This field specifies the preferred height and width, respectively for presentation purposes.

Color Information: This field specifies the colorspace of the decompressed video data.

Component information: This field contains information about the image data components.

D.1.3 MPV:BASIC_VIDEO_INFO

Basic Video Information specifies generic information about the video, such as the image size and number of components. This field may contain the sub-fields listed below.

```
MPV Schema Definition:

<xsd:element name="mpv:BASIC_VIDEO_INFO">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="dig35:FILE_FORMAT"                    minOccurs="0"/>
      <xsd:element ref="mpv:UNIQUE_ID"                        minOccurs="0"/>
      <xsd:element ref="dig35:IMAGE_SIZE"                     minOccurs="0"/>
      <xsd:element ref="mpv:MEDIA_TYPE"                       minOccurs="0"/>
    </xsd:sequence>

    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang"/>
  </xsd:complexType>
</xsd:element>
```

File and Format: This field specifies the file name and format of the video data.

Unique Identifier: This field specifies a unique identifier of the item.

Image Size: This field specifies the width and height of the video.

Media Type: This field specifies the media type of the video data.

D.1.4 Suggested Values for FILE_FORMAT for Video

File format specifies the format and name of the video file associated with the metadata and uses the same structure as DIG35. Additional suggested format types are defined that are relevant to video formats.

```
DIG35 Schema Definition:

<xsd:element name="FILE FORMAT">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="FILE NAME"          type="xsd:anyURI"        minOccurs="0"/>
      <xsd:element name="FORMAT TYPE"        type="xsd:string"        minOccurs="0"/>
      <xsd:element name="MIME_TYPE"          type="xsd:string"        minOccurs="0"/>
      <xsd:element name="VERSION"            type="xsd:string"        minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

File Name: This field specifies the name of video file. It can be any URI, including a name relative the current directory, a root-relative pathname, or a URL.

File Format Type: This field specifies the file format of the video. This is frequently a container format, in which media conforming to MEDIA_TYPE and MEDIA_SUBTYPE are found. The values are case insensitive.

Table: Suggested File Format Type values

Format Types	Description
ASF	Microsoft Advanced Streaming Format

AVI	Microsoft Audio-Video Interleaved Format
MOV	QuickTime Format
MPG	MPEG Format
WMA	Windows Media Audio
WMV	Microsoft Windows Media Video

MIME Type: This field specifies the Internet media type of the video file. The following are some common values.

MIME Types	**Description**
video/mpeg	MPEG format [RFC2045], [RFC2046]
video/quicktime	QuickTime Format

MIME Types	**Description**
audio/basic	WAV format [RFC2045, RFC2046]
audio/mpeg	MPEG-1 format [RFC3003]
audio/mpa-robust	MPEG-1 Layer 2 format [RFC3119]

Version: This field specifies the version of the file format.

```
 Example:

<FILE FORMAT>
  <FILE NAME>video.mpg</FILE NAME>
  <FORMAT_TYPE>MPG</FORMAT_TYPE>
  <MIME TYPE>video/mpeg</MIME TYPE>
</FILE_FORMAT>
```

D.1.5 Suggested Values for MPV:MEDIA_TYPE for Video

MPV defines a rich set of media types that can be used describe in some detail the type of media stream. Note that media type is quite distinct from file format; it describes the format of the media stream within the file. Many file formats are container formats that can contain a variety of media types.

NOTE: The detailed information for major and minor media types is based directly on extensive information available in Microsoft DirectShow 8.0 SDK and is Copyright 1995-2001 Microsoft Corp, All Rights Reserved. Inclusion in this working draft of the MPV specification is preliminary; permission to utilize this information must be provided by Microsoft before it can be adopted into this specification.

Media Type: This field specifis the major and minor type of the media contained in the file, separated by a "/", such as "stream/MPEG1VideoCD". The values are case insensitive.

Note that some of the minor media types contain color component information. This should not be relied upon for classifying according to color information. The preferred mechanism to record and use color information when it is available with the COLOR_INFO and COMPONENT_INFO elements.

Table: Suggested Major Media Type values.

Major Media Types	**Description**
Video	Video.
Stream	Byte stream with no time stamps.

MPEG2_PES	MPEG-2. Used by DVD.
Interleaved	Interleaved. Used by Digital Video (DV).
AUXLine21Data	Line 21 data. Used by closed captions.
File	File. Used by closed captions.
ScriptCommand	Data is a script command, used by closed captions.
Text	Text.
Timecode	Timecode data.

Table: Suggested Minor Media Type values when Major Media Type is "Video".

Minor Media Types for Major Type = "Video"	**Description**
ARGB32	ARGB, 32 bits per pixel. Uncompressed RGB samples with valid alpha bits.
CFCC	MJPG format produced by some cards.
CLJR	Cirrus Logic Jr YUV 411 format with less than 8 bits per Y, U, and V sample. Cinepak can produce it and Cirrus 5440 can produce an overlay with it. A Y sample at every pixel, a U and V sample at every fourth pixel horizontally on each line; every vertical line sampled.
CPLA	Cinepak UYVY format.
dvhd	High Definition DV format.
dvsd	Standard DV format.
dvsl	Long Play DV format.
IF09	Indeo produced YVU9 format with additional information about differences from the last frame. 9.5 bits per pixel but reported as 9.
IJPG	Intergraph JPEG format.
MJPG	Motion JPEG (MJPG) compressed video.
MPEG1Packet	MPEG1 Video Packet.
MPEG1Payload	MPEG1 Video Payload.
Overlay	Video delivered using hardware overlay.
Plum	Plum MJPG format.
QTJpeg	QuickTime JPEG compressed data.
QTMovie	Apple® QuickTime® compression.
QTRle	QuickTime RLE compressed data.
QTRpza	QuickTime RPZA compressed data.
QTSmc	QuickTime SMC compressed data.
RGB1	RGB, 1 bit per pixel. Palettized.
RGB24	RGB, 24 bits per pixel. Uncompressed RGB samples.
RGB32	RGB, 32 bits per pixel. Uncompressed RGB samples. Do not use the alpha bits with this media type. (Compare MEDIASUBTYPE_ARGB32.)
RGB4	RGB, 4 bits per pixel. Palettized.
RGB555	555 format of RGB, 16 bits per pixel. Uncompressed RGB samples.
RGB565	565 format of RGB, 16 bits per pixel. Uncompressed RGB samples.
RGB8	RGB, 8 bits per pixel. Palettized.
TVMJ	TrueVision MJPG format.
UYVY	UYVY format data. A packed YUV format. A Y sample at every pixel, a U and V sample at every second pixel horizontally on each line; every vertical line sampled. Probably the most popular of the various YUV 4:2:2

	formats. Byte ordering (lowest first) is U0, Y0, V0, Y1, U2, Y2, V2, Y3, U4, Y4, V4, Y5, where the suffix 0 is the leftmost pixel and increasing numbers are pixels increasing left to right. Each 4-byte block is 2 image pixels.
VideoPort	Video port data, used with DVD.
VPVBI	Video port vertical blanking interval (VBI) data.
VPVideo	Video port video data.
WAKE	MJPG format produced by some cards.
Y211	YUV 211 format data. A packed YUV format. A Y sample at every second pixel, a U and V sample at every fourth pixel horizontally on each line; every vertical line sampled. Byte ordering (lowest first) is Y0, U0, Y2, V0, Y4, U4, Y6, V4, Y8, U8, Y10, V8, where the suffix 0 is the leftmost pixel and increasing numbers are pixels increasing left to right. Each 4-byte block is 4 image pixels.
Y411	YUV 411 format data. Same as Y41P.
Y41P	Y41P format data. A packed YUV format. A Y sample at every pixel, a U and V sample at every fourth pixel horizontally on each line; every vertical line sampled. Byte ordering (lowest first) is U0, Y0, V0, Y1, U4, Y2, V4, Y3, Y4, Y5, Y6, Y7, where the suffix 0 is the leftmost pixel and increasing numbers are pixels increasing left to right. Each 12-byte block is 8 image pixels.
YUY2	YUY2 format data. Same as UYVY but with different pixel ordering. Byte ordering (lowest first) is Y0, U0, Y1, V0, Y2, U2, Y3, V2, Y4, U4, Y5, V4, where the suffix 0 is the leftmost pixel and increasing numbers are pixels increasing left to right. Each 4-byte block is 2 image pixels.
YVU9	Standard YVU9 format uncompressed data. A planar YUV format. A Y sample at every pixel, a U and V sample at every fourth pixel horizontally on each line; a Y sample on every vertical line, a U and V sample at every fourth vertical line. 9 bits per pixel.
YVYU	YVYU format data. A packed YUV format. Same as UYVY but with different pixel ordering. Byte ordering (lowest first) is Y0, V0, Y1, U0, Y2, V2, Y3, U2, Y4, V4, Y5, U4, where the suffix 0 is the leftmost pixel and increasing numbers are pixels increasing left to right. Each 4-byte block is 2 image pixels.

Table: Suggested Minor Media Type values when Major Media Type is "Stream".

Minor Media Types for Major Type = "Stream"	**Description**
Asf	Advanced Streaming Format (ASF)
Avi	Data from AVI file
DssVideo	Dss Video
MPEG1System	MPEG system
MPEG1Video	MPEG video
MPEG1VideoCD	MPEG video CD

Table: Suggested Minor Media Type values when Major Media Type is "MPEG2_PES".

Minor Media Types for Major Type = "MPEG2_PES"	**Description**
DVD_SUBPICTURE	Subpicture data
MPEG2_TRANSPORT	MPEG-2 transport stream
MPEG2_TRANSPORT_STRIDE	MPEG-2 multiplexed packets (a.k.a. stride packets)
MPEG2_PROGRAM	MPEG-2 program stream

The suggested media types vary significantly according to the actual format of MPEG-1 data. The following information summarizes the media types for MPEG-1 data.

Table: Major and minor types corresponding to various MPEG-1 data.

Major Type	Minor Type	Description	Sample contents
Stream	MPEG1System	MPEG-1 System Stream	BYTE stream, no alignment
Stream	MPEG1VideoCD	MPEG-1 System Stream for Video CD	BYTE stream, no alignment
Stream	MPEG1Video	MPEG-1 Native Video Stream	Array of video stream bytes (no system layer)
Video	MPEG1Packet	MPEG-1 Video Packet	Single MPEG-1 packet including packet header
Video	MPEG1Payload	MPEG-1 Video payload	Byte-aligned MPEG-1 video data

D.1.6 MPV:VIDEO_CREATION

The video creation metadata for MPV uses some of the same elements found in the image creation metadata.

```
MPV Schmema

  <xsd:annotation>
    <xsd:documentation> Video Creation Metadata </xsd:documentation>
  </xsd:annotation>

  <xsd:element name="mpv:VIDEO_CREATION">
    <xsd:annotation>
      <xsd:documentation>
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="mpv:GENERAL_CREATION_INFO" minOccurs="0"/>
        <xsd:element ref="dig35:CAMERA_CAPTURE" minOccurs="0"/>
        <xsd:element ref="dig35:SOFTWARE_CREATION" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
```

D.1.7 Example

```
Example of Video Creation metadata
```



```
    <MPV:VIDEO MPV:ID="0004">  <!-- MPV metadata of VIDEO -->
      <MPV:BASIC_VIDEO_PARAM>
        <MPV:BASIC VIDEO INFO>
          <FILE FORMAT>
            <FILE_NAME>/my pictures/VIDEO1.mpg</FILE_NAME>
            <FORMAT TYPE>MPG</FORMAT TYPE>
          </FILE FORMAT>
          <MPV:UNIQUE_ID>
            <UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf6</UID>
          </MPV:UNIQUE ID>
          <IMAGE_SIZE>
            <WIDTH>352</WIDTH>
            <HEIGHT>240</HEIGHT>
          </IMAGE_SIZE>
          <MPV:MEDIA TYPE>stream/MPEG1VideoCD</MPV:MEDIA TYPE>
        </MPV:BASIC VIDEO INFO>
      </MPV:BASIC_VIDEO_PARAM>
      <MPV:VIDEO CREATION>
        <MPV:GENERAL CREATION INFO>
          <CREATION_TIME>2001-05-15T10:13:29</CREATION_TIME>
          <MPV:CREATION SOURCE>Digital Camera</VIDEO SOURCE>
          <MPV:CREATION TYPE>Original Scene</SCENE TYPE>
          <MPV:CREATOR>
            <PERSON NAME>
              <NAME COMP Type="Given">John</NAME COMP>
              <NAME_COMP Type="Family">Smith</NAME_COMP>
           </PERSON NAME>
          </MPV:CREATOR>
        </MPV:GENERAL_CREATION_INFO >
       <CAMERA CAPTURE>
          <CAMERA INFO>
             <MANUFACTURER>
                <ORG NAME>Acme</ORG NAME>
             </MANUFACTURER>
             <MODEL>MODEL 1000</MODEL>
          </CAMERA INFO>
       </CAMERA CAPTURE>
      </MPV:VIDEO_CREATION>
      <MPV:CONTENT DESCRIPTION>
        <GROUP CAPTION>Trip to Waldo Lake<br>Sept. 14, 2001<br>Smith & Jones
 Familes</GROUP_CAPTION>
      </MPV:CONTENT DESCRIPTION>
    </MPV:VIDEO>
```

D.1.8 Supported DIG35 Elements

The BASIC_VIDEO_PARAM element supports many DIG35 elements. For details on their content, refer to the DIG35 specification.

DIG35 elements: IMAGE_SIZE, PREF_PRESENTATION_PARAM, COLOR_INFO, COLOR_SPACE, and COMPONENT_INFO.

```
Example:

<IMAGE SIZE>
  <WIDTH>1600</WIDTH>
  <HEIGHT>1200</HEIGHT>
```

```
</IMAGE SIZE>

Example:

<PREF PRESENTATION PARAM>
  <WIDTH>0.1524</WIDTH>
  <HEIGHT>0.1016</HEIGHT>
</PREF PRESENTATION PARAM>

<COLOR INFO>
  <COLORSPACE>
    <PROFILE_NAME>sRGB</PROFILE_NAME>
  </COLORSPACE>
</COLOR INFO>

<!-- Example 1: 4 components, pre-multiplied, RGBA, all 8-bits -->
<COMPONENT INFO>
  <NUM_COMPONENT>4</NUM_COMPONENT>
  <PREMULTIPLIED>true</PREMULTIPLIED>
  <COMPONENTS>RGBA</COMPONENTS>
  <COMP_SIZE>8</COMP_SIZE>
</COMPONENT INFO>

<!-- Example 2: 3 components, YCbCr, 4:2:2 -->
<COMPONENT INFO>
  <NUM COMPONENT>3</NUM COMPONENT>
  <COMPONENTS>YCbCr</COMPONENTS>
  <SAMPLE RATIO>4:2:2</SAMPLE RATIO>
  <PIXEL COMP>Centered</PIXEL COMP>
  <COMP_SIZE>8</COMP_SIZE>
</COMPONENT_INFO>
```

D.1.9 Video Example

Following example shows the Basic Video Parameters of a MPEG video file in MPV.

```
Example of BASIC VIDEO PARAM metadata block

<?xml version="1.0" encoding="UTF-8"?>
<--
  - File name: image.jpg; File format: JPEG/JFIF; Image size: 1600 x 1200;
  -->
<METADATA TYPE="Single">
  <VIDEO>
   <BASIC VIDEO PARAM>
      <BASIC VIDEO INFO>
         <FILE_FORMAT>
            <FILE NAME>video.mpg</FILE NAME>
            <FORMAT TYPE>MPG</FORMAT TYPE>
            <MIME_TYPE>video/mpeg</MIME_TYPE>
         </FILE FORMAT>
         <IMAGE SIZE>
           <WIDTH>352</WIDTH>
           <HEIGHT>240</HEIGHT>

         <MEDIA_TYPE>stream/MPEG1VideoCD</MEDIA_TYPE>
      </BASIC VIDEO INFO>
   </BASIC_VIDEO_PARAM>
```

```
  </VIDEO>
</METADATA>
```

Annex E: Audio Metadata

E.1 MPV Specification

E.1.1 MPV:AUDIO

```
<xsd:element name="mpv:AUDIO">
  <xsd:complexType>
    <xsd:sequence>
        <xsd:element ref="mpv:BASIC_AUDIO_PARAM" minOccurs="0"/>
        <xsd:element ref="mpv:AUDIO_CREATION" minOccurs="0"/>
        <xsd:element ref="mpv:CONTENT_DESCRIPTION" minOccurs="0"/>
        <xsd:element ref="mpv:HISTORY" minOccurs="0"/>
        <xsd:element ref="dig35:IPR" minOccurs="0"/>
        <xsd:element ref="mpv:VARIANTS" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang" use="optional" default="en"/>
  </xsd:complexType>
</xsd:element>
```

E.1.2 MPV:BASIC_AUDIO_PARAM

Basic Audio Parameter Metadata may contain the sub-fields listed below.

```
MPV Schema Definition:

<xsd:element name="mpv:BASIC_AUDIO_PARAM">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="mpv:BASIC_AUDIO_INFO"  minOccurs="0"/>
    </xsd:sequence>

    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang"/>  </xsd:complexType>
</xsd:element>
```

Basic Video Information: This field specifies generic information about the video.

E.1.3 MPV:BASIC_AUDIO_INFO

Basic Audio Information specifies generic information about the audio. This field may contain the sub-fields listed below.

```
MPV Schema Definition:

<xsd:element name="mpv:BASIC_AUDIO_INFO">
  <xsd:complexType>
    <xsd:sequence>
```



```
      <xsd:element ref="mpv:UNIQUE_ID"                          minOccurs="0"/>
      <xsd:element ref="dig35:FILE_FORMAT"                      minOccurs="0"/>
      <xsd:element ref="mpv:AUDIO_SIZE"                         minOccurs="0"/>
      <xsd:element ref="mpv:MEDIA_TYPE"                         minOccurs="0"/>
    </xsd:sequence>

    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang"/>
  </xsd:complexType>
</xsd:element>
```

File and Format: This field specifies the file name and format of the audio data.

Unique Identifier: This field specifies a unique identifier of the item.

Audio Size: This field specifies the info about of the audio.

Media Type: This field specifies the media type of the audo data.

E.1.4 MPV:AUDIO_SIZE

Audio size defines some basic audio parameters.
[TODO]

E.1.5 Suggested Values for FILE_FORMAT for Audio

File format specifies the format and name of the audio file associated with the metadata and uses the same structure as DIG35. Additional suggested format types are defined that are relevant to audio formats.

```
DIG35 Schema Definition:

<xsd:element name="FILE_FORMAT">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="FILE_NAME"          type="xsd:anyURI"        minOccurs="0"/>
      <xsd:element name="FORMAT_TYPE"        type="xsd:string"        minOccurs="0"/>
      <xsd:element name="MIME_TYPE"          type="xsd:string"        minOccurs="0"/>
      <xsd:element name="VERSION"            type="xsd:string"        minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

File Name: This field specifies the name of audio file. It can be any URI, including a name relative the current directory, a root-relative pathname, or a URL.

File Format Type: This field specifies the file format of the audio. This is frequently a container format, in which media format is defined by the MEDIA_TYPE element. The values are case insensitive.

Table: Suggested File Format Type values

Format Types	Description
AIFF	AIFF format
MPA	MPEG 1 Layer 2 Format
MP3	MPEG 1 Layer 3 Format
WAV	WAV format
WMA	Windows Media Audio

MIME Type: This field specifies the Internet media type of the video file. The following are some common values.

Table: Suggested File Format Type values

MIME Types	Description
`audio/basic`	WAV format [RFC2045, RFC2046]
`audio/mpeg`	MPEG-1 format [RFC3003]
`audio/mpa-robust`	MPEG-1 Layer 2 format [RFC3119]

Version: This field specifies the version of the file format.

```
Example:

<FILE FORMAT>
  <FILE_NAME>audio.wav</FILE_NAME>
  <FORMAT TYPE>WAV</FORMAT TYPE>
  <MIME_TYPE>audio/basic</MIME_TYPE>
</FILE_FORMAT>
```

E.1.6 Suggested Values for MEDIA_TYPE for Audio

MPV defines a rich set of media types that can be used describe in some detail the type of media stream. Note that media type is quite distinct from file format; it describes the format of the media stream within the file. Many file formats are container formats that can contain a variety of media types.

NOTE: The detailed information for major and minor media types is based directly on extensive information available in Microsoft DirectShow 8.0 SDK and is Copyright 1995-2001 Microsoft Corp, All Rights Reserved. Inclusion in this working draft of the MPV specification is preliminary; permission to utilize this information must be provided by Microsoft before it can be adopted into this specification.

The MEDIA_TYPE schema is the same as is used for video items, but the suggested values are different.

Media Type is composed of major and minor types, separated by a "/", as in "stream/WAVE". MPV suggests a broad set of major and minor type values.

Major Media Type: This field specifis the major type of the media contained in the file.

Table: Suggested Major Media Type values.

Major Media Types	Description
Audio	Audio.
Midi	MIDI format.
Stream	Byte stream with no time stamps.

Minor Media Type: This field specifies the minor type of the media contained in the file. The values are case insensitive.

Table: Suggested Minor Media Type values when Major Media Type is "Stream".

Minor Media Types for Major Type = "Stream"	Description
AIFF	Data from AIFF file
AU	Data from AU file
DssAudio	Dss Audio
DssVideo	Dss Video
MPEG1Audio	MPEG audio
WAVE	Data from WAV file

Table: Suggested Minor Media Type values when Major Media Type is "MPEG2_PES".

Minor Media Types for Major Type = "MPEG2_PES"	Description

DOLBY_AC3	Dolby data
MPEG2_AUDIO	MPEG-2 audio data
DVD_LPCM_AUDIO	DVD audio data

Table: Suggested Minor Media Type values when Major Media Type is "Audio".

Minor Media Types for Major Type = "Audio"	**Description**
PCM	PCM audio.
MPEG1Packet	MPEG1 Audio packet.
MPEG1Payload	MPEG1 Audio Payload.

The suggested media types vary significantly according to the actual format of MPEG-1 data. The following information summarizes the media types for MPEG-1 data.

Table: Major and minor types for various types of MPEG-1 data.

Major Type	Minor Type	Description	Sample contents
Stream	MPEG1Audio	MPEG-1 Native Audio Stream	Array of audio stream bytes (no system layer)
Audio	MPEG1Packet	MPEG-1 Audio Packet	Single MPEG-1 packet including packet header
Audio	MPEG1Payload	MPEG-1 Audio payload	Byte-aligned MPEG-1 audio data

E.1.7 MPV:AUDIO_CREATION

E.1.8 Audio Example

Following example shows the Basic Audio Parameters of a WAVE audio file in MPV.

```
Example of BASIC_AUDIO_PARAM metadata block

<?xml version="1.0" encoding="UTF-8"?>
<METADATA TYPE="Single">
  <VIDEO>
   <BASIC VIDEO PARAM>
      <BASIC_VIDEO_INFO>
         <FILE FORMAT>
            <FILE NAME>audio.wav</FILE NAME>
            <FORMAT_TYPE>WAV</FORMAT_TYPE>
            <MIME TYPE>audio/basic</MIME TYPE>
         </FILE FORMAT>
         <MEDIA_TYPE>stream/WAVE</MEDIA_TYPE>
      </BASIC AUDIO INFO>
   </BASIC AUDIO PARAM>
  </VIDEO>
</METADATA>
```


Annex F: Document Metadata

[MUCH TODO]

F.1 MPV Specification

F.1.1 MPV:DOCUMENT

```
<xsd:element name="mpv:DOCUMENT">
  <xsd:complexType>
    <xsd:sequence>
        <xsd:element ref="mpv:BASIC_DOCUMENT_PARAM" minOccurs="0"/>
        <xsd:element ref="mpv:DOCUMENT_CREATION" minOccurs="0"/>
        <xsd:element ref="mpv:CONTENT_DESCRIPTION" minOccurs="0"/>
        <xsd:element ref="mpv:HISTORY" minOccurs="0"/>
        <xsd:element ref="dig35:IPR" minOccurs="0"/>
        <xsd:element ref="mpv:VARIANTS" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang" use="optional" default="en"/>
  </xsd:complexType>
</xsd:element>
```

F.1.2 MPV:BASIC_DOCUMENT_PARAM

F.1.3 Suggested Values for FILE_FORMAT for Documents

File format specifies the format and name of the video file associated with the metadata and uses the same structure as DIG35. Additional suggested format types are defined that are relevant to video formats.

File Format Type: This field specifies the file format of the video. This is frequently a container format, in which media conforming to MEDIA_TYPE and MEDIA_SUBTYPE are found. The values are case insensitive.

Table: Suggested File Format Type values

Format Types	Description

MIME Type: This field specifies the Internet media type of the video file. The following are some common values.

MIME Types	**Description**

F.1.4 Suggested Values for MPV:MEDIA_TYPE for Documents

MPV defines a rich set of media types that can be used describe in some detail the type of media stream. Note that media type is quite distinct from file format; it describes the format of the media stream within the file. Many file formats are container formats that can contain a variety of media types.

Major Media Type: This field specifis the major type of the media contained in the file.

Table: Suggested Major Media Type values.

Major Media Types	**Description**
Image	

Minor Media Type: This field specifies the minor type of the media contained in the file. The values are case insensitive.

Note that some of the minor media types contain color component information. This should not be relied upon for classifying according to color information. The preferred mechanism to record and use color information when it is available with the COLOR_INFO and COMPONENT_INFO elements.

Table: Suggested Minor Media Type values when Major Media Type is "Image".

Minor Media Types for Major Type = "Image"	**Description**

F.1.5 MPV:DOCUMENT_CREATION

Annex G: Content Description

DIG35 defines metadata for describing content. One notable shortcoming is the limited ability to associate a rich set of content with a content item; DIG35 limits associations to audio files for still images. MPV resolves this problem.

G.1 DIG35 Spec Review

DIG35 is a metadata specification for still images, and the only alternate media type referenced by the specification is the metadata for audio content to be associated with a still image. This occurs by using one or more <AUDIO> elements in the Content Description metadata block <CONTENT_DESCRIPTION> … </CONTENT_DESCRIPTION>. The <AUDIO> element refers to the audio content using a URI (<AUDIO_STREAM>) that is typed according to <AUDIO_FORMAT>.

```
<xsd:element name="CONTENT DESCRIPTION">
  <xsd:annotation>
    <xsd:documentation>
              See section Annex C:  Content Description Metadata
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="GROUP CAPTION" type="dig35:tLangString" minOccurs="0"/>
      <xsd:element name="CAPTION" type="dig35:tLangString" minOccurs="0"/>
      <xsd:element name="CAPTURE TIME" type="dig35:tDateTime" minOccurs="0"/>
      <xsd:element name="LOCATION" type="dig35:tLocation" minOccurs="0"/>
      <xsd:element ref="dig35:PERSON" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element ref="dig35:THING" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element ref="dig35:ORGANIZATION" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element ref="dig35:EVENT" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element ref="dig35:AUDIO" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element ref="dig35:PROPERTY" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element ref="dig35:DICTIONARY" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element ref="dig35:COMMENT" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang"/>
  </xsd:complexType>
</xsd:element>

<xsd:element name="AUDIO">
  <xsd:annotation>
    <xsd:documentation>
              See section C.3.9  Audio
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="AUDIO STREAM" type="xsd:anyURI"/>
      <xsd:element name="AUDIO FORMAT" type="xsd:string" minOccurs="0"/>
      <xsd:element name="MIME TYPE" type="xsd:string" minOccurs="0"/>
      <xsd:element name="DESCRIPTION" type="dig35:tLangString" minOccurs="0"/>
      <xsd:element ref="dig35:COMMENT" minOccurs="0"/>
```



```
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
```

G.2 Design Approach

The current DIG35 approach gives audio content a distinctly second class status with a highly constrained set of metadata. On the other hand, it is very simple and the processing application can immediately identify the presence of an audio stream to play along with a given image.

Being able to have an arbitrary collection of content items that are associated with or describe another item is an important requirement for MPV. Each of these content items may themselves have all the richness of any content item, including being single items or collections.

MPV does not embrace DIG35's approach that makes an arbitrary decision about what type of content items may be associated with a given content item. Instead, MPV deprecates the use of the DIG35 mechanism for specifying audio content associated with a still image.

MPV specifies instead that any content item can be described by one or more other content items. Thus, as the richness of content data types increase, so too can the richness of the description. Furthermore, the description can apply to collections as well as to singletons. To achieve parity with the current DIG35 specification, the audio content type metadata must be defined. This occurs in this chapter, along with defining the video content type metadata.

G.3 MPV Specification

G.3.1 MPV:CONTENT_DESCRIPTION

By adding the MPV List to the content description, any number of additional content items of any type can be specified to any level of desired hierarchy. The dig35:AUDIO item in dig35:CONTENT_DESCRIPTION has been deleted, as the new mpv:LIST can contain an audio item.

```
  <xsd:element name="MPV:CONTENT_DESCRIPTION">
    <xsd:annotation>
      <xsd:documentation>
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="GROUP_CAPTION" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="CAPTION" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="CAPTURE_TIME" type="dig35:tDateTime" minOccurs="0"/>
        <xsd:element name="LOCATION" type="dig35:tLocation" minOccurs="0"/>
        <xsd:element ref="dig35:PERSON" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dig35:THING" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dig35:ORGANIZATION" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dig35:EVENT" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dig35:PROPERTY" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dig35:DICTIONARY" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dig35:COMMENT" minOccurs="0"/ maxOccurs="unbounded">
        <xsd:element ref="mpv:ASSOCIATED" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
```

```
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="mpv:ASSOCIATED">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:group ref="ITEM" minOccurs="0" maxOccurs="1"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang" use="optional" default="en"/>
    </xsd:complexType>
  </xsd:element>
```

G.3.2 MPV:GENERAL_CREATION_INFO, MPV:CREATOR, MPV:CREATION_SOURCE, MPV:CREATION_TYPE

MPV:General Creation Info is based on DIG35 General Creation Info, but contained renamed elements to make it more general purpose. Renamed dig35:IMAGE_CREATOR to mpv:CREATOR for generality. Renamed dig35:IMAGE_SOURCE to mpv:CREATION_SOURCE for generality. Renamed dig35:SCENE_TYPE to mpv:CREATION_TYPE for generality.

```
  <xsd:element name="mpv:GENERAL_CREATION_INFO">
    <xsd:annotation>
      <xsd:documentation>
                See section B.3.1  General Creation Information
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="dig35:CREATION_TIME" type="xsd:dateTime" minOccurs="0"/>
        <xsd:element name="mpv:CREATION_SOURCE" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="mpv:CREATION_TYPE" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="mpv:CREATOR" type="dig35:tPerson" minOccurs="0"/>
        <xsd:element name="dig35:OPERATOR_ORG" type="dig35:tOrganization" minOccurs="0"/>
        <xsd:element name="dig35:OPERATOR_ID" type="dig35:tLangString" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
```

Annex H: History

H.1 MPV Specification

H.1.1 MPV:HISTORY

Moving this into the MPV namespaces allows replacement of dig35:METADATA with mpv:LIST and enables addition of hists for processing other media types.

```
<xsd:element name="mpv:HISTORY">
  <xsd:annotation>
    <xsd:documentation>
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="mpv:PROCESSING_SUMMARY" minOccurs="0"/>
      <xsd:element ref="dig35:IMAGE_PROCESSING_HINTS" minOccurs="0"/>
      <xsd:element ref="mpv:LIST" minOccurs="0" maxOccurs="unbounded"/>
    </xsd:sequence>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang"/>
  </xsd:complexType>
</xsd:element>
```

H.1.2 MPV:PROCESSING_SUMMARY, MPV:PROCESSING_TYPES

Moving this into the MPV namespace enables addition of processing summares of other media types. Additional processing types can be added to the MPV:PROCESSING_TYPES group. Also, the processing summary has been improved to allow a sequence of processing steps of any type to be recorded in the order they occurred.

```
<xsd:element name="mpv:PROCESSING_SUMMARY">
  <xsd:annotation>
    <xsd:documentation>
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexType>
    <xsd:sequence>
       <xsd:group ref="mpv:PROCESSING_TYPES" minOccurs="0" maxOccurs="unbounded">
    </xsd:sequence>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
  </xsd:complexType>
</xsd:element>

<xsd:group name="mpv:PROCESSING_TYPES">
  <xsd:annotation>
    <xsd:documentation>
    </xsd:documentation>
  </xsd:annotation>
  <xsd:complexType>
    <xsd:choice>
      <xsd:element name="dig35:IMG_CREATED" minOccurs="0">
```



```
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="dig35:IMG_CROPPED" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="dig35:IMG_TRANSFORMED" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="dig35:IMG_GTC_ADJ" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="dig35:IMG_STC_ADJ" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="dig35:IMG_SPATIAL_ADJ" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="dig35:IMG_EXT_EDITED" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="dig35:IMG_RETOUCHED" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="dig35:IMG_COMPOSITED" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="dig35:IMG_METADATA" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <!-- TODO:  Add additional processing types -->
      </xsd:choice>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
    </xsd:complexType>
  </xsd:element>
```

Annex I: Variants -- Multi-Resolution, Multi-Format Content

[MUCH TODO]

Basic needs:
- multiple resolutions for any given asset or collection
- multiple formats for any given asset or collection
- local or remote references

- <METADATA> needs to add a <VARIANT> sub-element. Like <HISTORY>, <VARIANT> can contain a metadata block in it, giving the proxy the full ability to describe single images or a collection. A typical use of <VARIANT> would be to define a collection of images, each image of different size. The use of <CONTENT_DESCRIPTION><CATEGORY>

Typical variant situations are as follows:
- for a single image: screen res and thumbnail res variants are available
- for a video: screen res and thumbnail res still image variants are available
- for a collection of images/videos: a slideshow, an index page, …

```
MPV Schema definition

  <xsd:element name="mpv:VARIANTS">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="mpv:FULL_RES" minOccurs="0"/>
        <xsd:element ref="mpv:MID_RES" minOccurs="0"/>
        <xsd:element ref="mpv:THUMB_RES" minOccurs="0"/>
        <xsd:element ref="mpv:SLIDESHOW" minOccurs="0"/>
        <xsd:element ref="mpv:PRINT" minOccurs="0"/>
        <xsd:element ref="mpv:CONTENT_DESCRIPTION" minOccurs="0"/>
        <xsd:element ref="mpv:HISTORY" minOccurs="0"/>
        <xsd:element ref="dig35:IPR" minOccurs="0"/>
        <xsd:group ref="ITEM" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute ref="mpv:ID"/>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang" use="optional" default="en"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="mpv:FULL_RES">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:group ref="ITEM" minOccurs="0" maxOccurs="1"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang" use="optional" default="en"/>
    </xsd:complexType>
  </xsd:element>
```



```
<xsd:element name="mpv:MID_RES">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:group ref="ITEM" minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang" use="optional" default="en"/>
  </xsd:complexType>
</xsd:element>

<xsd:element name="mpv:THUMB_RES">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:group ref="ITEM" minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang" use="optional" default="en"/>
  </xsd:complexType>
</xsd:element>

<xsd:element name="mpv:SLIDESHOW">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:group ref="ITEM" minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang" use="optional" default="en"/>
  </xsd:complexType>
</xsd:element>

<xsd:element name="mpv:PRINT">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:group ref="ITEM" minOccurs="0" maxOccurs="1"/>
    </xsd:sequence>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang" use="optional" default="en"/>
  </xsd:complexType>
</xsd:element>

<xsd:group name="ITEM">
  <xsd:complexType>
     <xsd:choice>
       <xsd:element ref="dig35:METADATA"/>
       <xsd:element ref="mpv:IMAGE"/>
       <xsd:element ref="mpv:VIDEO"/>
       <xsd:element ref="mpv:AUDIO"/>
       <xsd:element ref="mpv:DOCUMENT"/>
       <xsd:element ref="mpv:ITEM-REF"/>
       <xsd:element ref="mpv:LIST"/>
     </xsd:choice>
  </xsd:complexType>
</xsd:element>
```


Annex J: Brand Identity

[MUCH TODO]

MPV allows the entities that created a given content item to carry aspects of their brand identity through to the end user. This does require a playback application that will honor the provided metadata; this is optional.

```
MPV Schema definition

  <xsd:element name="mpv:BRAND_IDENTITY">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="mpv:BRAND_ELEMENTS" minOccurs="0"/>
        <xsd:element ref="mpv:CONTENT_DESCRIPTION" minOccurs="0"/>
        <xsd:element ref="mpv:HISTORY" minOccurs="0"/>
        <xsd:element ref="dig35:IPR" minOccurs="0"/>
        <xsd:group ref="ITEM" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute ref="mpv:ID"/>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang" use="optional" default="en"/>
    </xsd:complexType>
  </xsd:element>


  <xsd:element name="mpv:BRAND_ELEMENTS">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="CAPTION" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element ref="mpv:LOGO" minOccurs="0"/>
        <xsd:element ref="mpv:THEME" minOccurs="0"/>
        <xsd:element ref="dig35:COLOR_INFO" minOccurs="0"/>
        <xsd:element ref="dig35:WEB" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="mpv:ID"/>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang" use="optional" default="en"/>
    </xsd:complexType>
  </xsd:element>


  <xsd:element name="mpv:LOGO">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:group ref="ITEM" minOccurs="0" maxOccurs="1"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang" use="optional" default="en"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="mpv:THEME">
    <xsd:complexType>
      <xsd:sequence>
        <xsd:group ref="ITEM" minOccurs="0" maxOccurs="1"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang" use="optional" default="en"/>
```



```
    </xsd:complexType>
  </xsd:element>

  <xsd:group name="ITEM">
    <xsd:complexType>
      <xsd:choice>
        <xsd:element ref="dig35:METADATA"/>
        <xsd:element ref="mpv:IMAGE"/>
        <xsd:element ref="mpv:VIDEO"/>
        <xsd:element ref="mpv:AUDIO"/>
        <xsd:element ref="mpv:DOCUMENT"/>
        <xsd:element ref="mpv:ITEM-REF"/>
        <xsd:element ref="mpv:LIST"/>
      </xsd:choice>
    </xsd:complexType>
  </xsd:element>
```


Annex K: Cross-Referencing

K.1 Requirements

The support for resource-constrained environments in the MPV representation is specifically targeted at low-memory, low-performance hardware/firmware environments, such as found on lower-cost consumer electronics devices. The principle design requirements for the MPV are to enable:

- linked files – segment content into multiple files to avoid unnecessary memory usage
- a file a MPV file

K.2 Design Approach

The attribute "MPV:ID" is added onto the <METADATA> element. This enables cross-reference to other images and collections within the metadata. For example, if a collection of pictures is given, it is possible to create a playlist that simply references the metadata about the pictures by name, rather than repeating it. MPV:ID is an attribute of type ID in XML-Schema.

MPV provides support for referencing elements that have been identified via an ID attribute in several ways. A METADATA element that has been identified can be referenced by a METADATA-REF element. This element has a single attribute named href which is of type xsd:anyURI. This allows references to METADATA elements that either in the same document (using a bare fragment identifier) or in other documents using a URI. The type of content in cross-document references is undefined until the reference is followed.

```
Example of a collection of two of the same image and a collection or an image (unknown
 until the reference is followed) defined in another file.

  <METADATA Type="Collection">
     <METADATA Type="Image" MPV::ID="Image1">
        … content …
     </METADATA>
     <METADATA-REF  href="#Image1" />
     <METADATA-REF  href="/MPV/collection2.mpv#Item2" />
  </METADATA>
```

The element <METADATA MPV:ID="Image1"> defines a collection of metadata under the name "Image1". This metadata can then be referenced as a whole via <MPV:METADATA-REF href="#Image1">.

Note that since the reference is via a URI both local and remote references can be specified. Local references use bare fragment identifiers and remote references use full URI (plus fragment identifier to locate the METADATA element within the target document).

K.3 MPV Specification

```
  <xsd:attribute name="mpv:ID" type="xsd:id">
    <xsd:annotation>
```



```
      <xsd:documentation>
                An ID to allow cross-references.
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>
```


Annex L: Internet Services

[MUCH TODO]

A broad range of internet services exist and can be envisioned that can provide valuable capabilities relating to photo-video content. MPV provides mechanisms to connect photo-video content in one place with internet services in another.

Basic needs:
- manifest upload
- references to local content
- service provider references

A number of places need to add the ability to have web references:

TODO: We need to add references to web services. There are a number of services that are under consideration, and they can be applied to collections or singles. We need to do some more research before committing to a specific format. HP's format is no standards-based, but it does work. We need to do more research on this. Other parts of HP are engaged with the I3A to standardize web service references.

Annex M: Support for Resource-Constrained Environments

MPV must be useful and used not only in resource-rich environments like modern computers, with lots of available memory, CPU performance, harddisk backing store, and good LAN performance. It must be well suited for resource-constrained environments, such as found on many consumer electronics devices.

A significant concern of those considering use of MPV for resource-constrained environments may be that it is an XML format. XML is the dominant standard for metadata content today, and it is entirely well suited for processing in PC and internet server computing environments. MultiPhotoVideo fully embraces XML by defining the standard to include an XML schema.

In addition, MPV introduces enhancements to DIG35 that make MPV well suited for both reading and writing in resource-constrained environments while still maintaining XML compliance.

M.1 Solution Design

M.1.1 Design Requirements

The support for resource-constrained environments in the MPV representation is specifically targeted at low-memory, low-performance hardware/firmware environments, such as found on lower-cost consumer electronics devices. The principle design requirements for the MPV are to enable:

- little parsing: read directly from offsets, seeking directly to wherever is needed in the file
- little parsing: read directly into data structures
- low-overhead: data structures in memory should only contain useful information; avoid “empty” content in data structures – this wastes memory.
- linked files – segment content into multiple files to avoid unnecessary memory usage
- single-pass writing of MPV structures

M.1.2 Design Approach

There have been significant engineering activities and some research into use of XML in resource-constrained environments with a diversity of results.

One thing is clear – there are many open source and commercial XML parsers available that require less than 100K of firmware and operate in low-memory environments. Fundamentally, the basic tools are available to process XML in resource-constrained environments.

But more barriers remain.

Barriers

The key barriers to be overcome in meeting the design requirements are:

- Traditional XML parsing reads the entire documen and build an in-memory representation. This is unacceptable for resource-constrained environments because the memory may not be available to store a complete in-memory representation.
- Discarding some of the parsed XML document contents to reduce memory usage is possible, but recovering it means parsing through the whole document again
- To avoid reparsing the whole document, the application could build an in-memory TOC of the document, allowing random access to the appropriate location to then parse an XML fragment. However, this still requires a complete pass through the XML document to create the TOC.
- To parse a DIG35 XML fragment, the active namespaces must be known at the start of the fragment.
- DIG35 does not provide for referencing DIG35 structures in other storage containers, thus no linking behaviour is defined.
- XML files suffer from lower information density and processing performance can suffer compared to than binary files because of they are in text format and the required syntax is verbose.

Solution Elements

The MPV approach to support resource-constrained environments focuses on some key enablers:

1. random access to XML content
2. pre-generated TableOfContents (TOC)
3. incremental just-in-time parsing
4. incremental update
5. single-pass writing of MPV files
6. a single file format that meets all purposes
7. inter-file references

Random access is a key enabler – it allows for "read-on-demand" approaches that allow only what data is needed to be kept in memory. Parsing can be performed, This means that XML namespace prefixes must either use well-known values or all the namespace declarations have to be available on the top-most element of each fragment.

Pre-generated Table of Contents with offsets for random access into the file allows the processor not to have to read the entire XML file before random access is possible.

Just-in-time parsing of XML fragments accessed via random access enables implementations to avoid a full parse of the document to access particular information. Delaying parsing of content until needed saves time by not parsing and reparsing a file.

Incremental update allow content to be edited in-place without rewriting the entire MPV structure. This is especially important for situations in which the rate of change or performance requirements are high, such as when taking pictures.

Single-pass writing reduces memory requirements by avoiding the need to buffer a file for a second pass algorithm, and it means that MPV content can be streamed to a write-once storage medium.

A **single file format** that meets all needs is clearly superior to two parallel files, simplifying generation, update, and processing.

Interfile references allows content to be segmented into various files, limiting the amount of data to be dealt with in a given file.

These solution elements completely address all the barriers to achieving the design requirements, except for one. This is the barrier of information density and processing performance due to the verbose and textual format of the

content. As these problems are inherent to the definition of an XML document, they cannot be eliminated altogether, but they can be mitigated. The approaches taken are:

> **Sequence the XML content** in each block, such that the most important comes first. This enables an application to abandon parsing once the needed information is found, and the most important information will be found first. DIG35 already does this. When coupled with random access, it is possible to reduce the amount of unnecessary parsing substantially.
>
> **Limit the size of the XML content** being processed. This can be achieved by segmenting files into smaller collections of content.

Enabling Random Access

The support for random access is provided by wrapping the core DIG35 XML representation in another level of XML which provides offset information for the internal XML representation. This offset information is itself part of the overall XML document so it remains a valid XML document.

When accessing MPV content, the application has to perform a minimal amount of XML processing before it is able to switch to using direct binary offsets to access the various content item descriptions in the document.

An MPV structure that supports random access is required to have a valid MPV:TOC element and is also required to emit the MPV processing instruction as the last content in the document.

[TODO] No/Yes to Binary XML

The current working draft of the MPV specification does not define the use of binary XML file. Binary XML representations tokenize the XML file and create a binary representation that includes a symbol table.

There are many dimensions to resouce-constraints: low-bandwidth, low-memory, low-CPU performance, low throughput. Much of the research into low-resource use of XML has centered on binary representations of XML content for mobile applications, and are driven by low-bandwidth and limited devices. WBXML, http://www.w3.org/TR/wbxml/, in one popular specification and implementation. However, the research is inconclusive about results.

The desired goals of using binary XML would be:

- Increase processing performance by reducing file parsing time and data file size.
- Reduce amount of storage used on disc or in memory.

Complications using binary XML would introduce are:

- research has not suggested substantial, or even any, efficiency and performance gains in parsing or avoiding overhead, as compared to operating on the textual XML representations directly.
- For MPV, if a binary XML format were selected, the textual XML format would still need to live alongside so that applications that handle MPV in resource-rich environments could utilize the wide variety of available XML software and tools. Having to create/maintain two metadata files would increase the effort of creating and maintaining up-to-date MPV content, especially for dynamic content situations, such as during photo-video content capture and editing.

Consequently, the current draf of the MPV specification does not utilize a binary XML format. The benefits offered by binary XML – reduced storage size both on-disc and in-memory – are made less relevant by random access to content in the XML file without storing the entire file in memory.

M.1.3 Implementation Requirements

For the design, we make the following assumptions and requirements on implementations for the contents of a MPV document that can be used in resource-constrained environments.

- The XML content is in Canonical Normal Form [C14N-2001], I.e. UTF-8 and normalized to the W3C canonical XML recommendation. This assures that a given MPV structure can be serialized by many means and result in byte-for-byte identical streams.
- It contains a METADATA-OFFSETS element that specifies the absolute offsets of the METADATA elements in the document
- It contains a NS-ENTRIES element that specifies all the namespace bindings that are used in the document. Note that a namespace binding can only be used once in a document.
- It contains a trailing MPV processing instruction that specifies the length of the document and the location of the TOC element that wraps the METADATA-OFFSETS and NS-ENTRIES element.
- MPV files can be written to a write-once medium

A conforming MPV document can then be processed as both an XML document using usual XML processing tools and as a MPV random access document using XML processing tools that have been enhanced for support of XML fragment processing.

The application has to perform a minimal amount of XML processing before it is able to switch to using direct binary offsets to access the various content item descriptions in the document. The fragments that are accessed using the direct offset can themselves be processed using normal XML tools with some slight pre-processing to provide namespace fixup.

An MPV that supports random access is required to have a valid MPV:TOC element and is also required to emit the MPV processing instruction as the last content in the document.

M.2 Example

Complete MPV example	Notes
`<?XML version=”1.0” >`	XML document
`<MPV:MPV xmlns=” http://www.digitalimaging.org/dig35/1.1/xml”`	MPV Wrapper
`xmlns:MPV=”http://www.osta.org/MPV/1.0/xml#” >`	
`<METADATA Type="Collection" MPV:ID="foo">`	Metadata for a collection
`<METADATA Type=”Single” MPV:ID=”bar”>`	Metadata of a single image
`<IMAGE>`	
`<BASIC IMAGE PARAM>`	
`<BASIC_IMAGE_INFO>`	
`<UNIQUE ID>`	UUID
`<UID>f81d4fae-7dec-11d0-a765-00a0c91e6bf6</UID>`	
`<ID_TYPE>http://www.digitalimaging.org/dig35/UUID</ID_TYPE>`	
`</UNIQUE ID>`	
`<IMAGE SIZE>`	Width & height
`<WIDTH>1600</WIDTH>`	
`<HEIGHT>1200</HEIGHT>`	
`</IMAGE SIZE>`	File information
`<FILE_FORMAT>`	
`<FILE NAME>/my pictures/image.jpg</FILE NAME>`	
`<FORMAT TYPE>EXIF</FORMAT TYPE>`	
`</FILE_FORMAT>`	Media type
`</BASIC IMAGE INFO>`	
`<MEDIA TYPE>`	
`<MAJOR_MEDIA_TYPE>image</MAJOR_MEDIA_TYPE>`	Color depth
`<MINOR MEDIA TYPE>EXIF</MINOR MEDIA TYPE>`	
`</MEDIA TYPE>`	
`<COMPONENT_INFO>`	

```
            <COMPONENTS>RGB</COMPONENTS>
            <NUM_COMPONENT>3</NUM_COMPONENT>
            <COMP SIZE>8</COMP SIZE>
            <COMP SIZE>8</COMP SIZE>
            <COMP_SIZE>8</COMP_SIZE>
          </COMPONENT INFO>
        </BASIC IMAGE PARAM>
        <CAMERA_SETTINGS>
          <CAMERA LOCATION>
            <COMMENT>At home</COMMENT>
          </CAMERA_LOCATION>
          <ORIENTATION>
            <ROLL>90</ROLL>
          </ORIENTATION>
        </CAMERA SETTINGS>
        <IMAGE CREATION>
          <GENERAL_CREATION_INFO>
            <CREATION TIME>2001-05-14T21:43:23</CREATION TIME>
            <IMAGE CREATOR>
              <PERSON_NAME>
                <NAME COMP Type="Given">Piet</NAME COMP>
                <NAME COMP Type="Family">van Zee</NAME COMP>
             </PERSON_NAME>
            </IMAGE CREATOR>
          </GENERAL CREATION INFO >
        </IMAGE_CREATION>
      </IMAGE>
      <CONTENT DESCRIPTION>
        <GROUP_CAPTION>Mary's visit</GROUP_CAPTION>
        <CAPTION>Mary and the kids bake pies</CAPTION>
        <COMMENT>Info about the image</COMMENT>
      </CONTENT_DESCRIPTION>
    </METADATA>
    <METADATA Type="Single">
      <IMAGE>
      ... content ...
      </IMAGE>
    </METADATA>
  </METADATA>
  <MPV:TOC xmlns:MPV="mpv-nsval">
    <MPV:NS-ENTRIES>
      <MPV:NS-ENTRY NS-PREFIX=""
               NS-URI="http://www.digitalimaging.org/dig35/1.1/xml"/>
      <MPV:NS-ENTRY NS-PREFIX="MPV"
               NS-URI=" http://www.osta.org/MPV/1.0/xml#" />
   </MPV:NS-ENTRIES>
   <MPV:METADATA-OFFSETS>
     <MPV:OFFSET type="Collection" byteref="2445">
       <MPV:OFFSET type="Image" byteref="2456" />
       <MPV:OFFSET type="Image" byteref="2675" />
     </MPV:OFFSET>
   </MPV:METADATA-OFFSETS>
  </MPV:TOC>
</MPV:MPV>
<?MPV TOC-OFFSET="2853" DOC-SIZE="3333" MPV?>
```

Location

Orientation

Date taken

Image creator

Category
Title
Info

Metadata of a single image

MPV Table of Contents
Namespaces

Offsets for random access

Processing Instruction

M.3 Reader Processing Algorithm

When a MPV document is first encountered and will be read, the following algorithm should be applied.

- The processor will seek to the end of the file and compare the last five bytes to the value:
  'M', 'P', 'V', '?', '>'
- If the compare succeeds, it will process from that location backwards until it matches the following five bytes:
  '<', '?', 'M', 'P', 'V'
- At this point, the processor can extract the TOC-OFFSET and DOC-SIZE name-values pairs from this MPV processing instruction.
- The processor should check the DOC-SIZE value against the actual size of the document. If the sizes don't match, the document must be processed as an XML document, and optionally a TOC may be created on the fly. If the sizes match, the TOC-OFFSET can be used to seek to the start of the MPV:TOC element.
- The TOC element can be processed as a well formed XML fragment and the resulting information can be used to perform further byte offset processing.

M.4 One-Pass Writer Processing Algorithm

When a MPV document is being serialized and written, the following algorithm should be applied.
The writer of a MPV document must conform to the following requirements in order to generate a compliant MPV document:

- Generate a canonical xml document.
- Only bind a namespace prefix to a single namespace.
- Keep track of all namespace bindings in a document.
- Generate a document unique ID for each dig35:METADATA element and note the byte offset of the beginning of that element.
- Once the entire top-level METADATA element has been generated, the writer will generate the MPV:TOC and note its byte offset.
- It will populate the MPV:TOC with its METADATA-OFFSETS element and NS-ENTRIES element with the offsets and namespaces that it has been accumulating while generating the document.
- The writer will then generate the MPV processing instruction as the last content in the document. See the Reader Processing Algorithm for a description of the MPV processing instruction.

M.5 MPV Specification

M.5.1 MPV Wrapper Element

An MPV document is identified by a top-level MPV:MPV element. The MPV element contains a single dig35:METADATA or MPV:METADATA-REF element and a MPV TOC element.

```
MPV document schema

<xsd:element name="MPV">
    <xsd:annotation>
      <xsd:documentation>
      </xsd:documentation>
    </xsd:annotation>

    <xsd:complexType>
      <xsd:sequence>
```

```
        <xsd:choice>
            <xsd:element ref="dig35:METADATA" minOccurs="1" maxOccurs="1"/>
            <xsd:element ref="MPV:METADATA-REF" minOccurs="1" maxOccurs="1"/>
        </xsd:choice>
        <xsd:element ref="MPV:TOC" minOccurs="0" maxOccurs="1"/>
      </xsd:sequence>
    </xsd:complexType>
</xsd:element>
```

```
Complete MPV example

<?XML version="1.0" >
<MPV:MPV>
  <METADATA Type="Single" MPV:ID="foobar">
     ...
  </METADATA>
  <MPV:TOC xmlns:MPV="mpv-nsval">
     ...
  </MPV:TOC>
</MPV:MPV>
<?MPV TOC-OFFSET="2653" DOC-SIZE="3333" MPV?>
```

M.5.2 MPV:TOC Elements

The MPV TOC element follows the dig35:METADATA or METADATA-REF element in the MPV:MPV root element.

It contains an MPV:METADATA-OFFSETS element. MPV:METADATA-OFFSETS is a simple container for a hierarchical collection of MPV:OFFSET Elements that provide binary offset addresses to each element. While currently a singleton element, METADATA-OFFSETS anticipates other uses of the TOC and groups together the offset values.

It has the following schema:

```
MPV Table of Contents (TOC) schema

<xsd:element name="MPV:TOC">
    <xsd:annotation>
      <xsd:documentation>
                Table of Contents of the overall XML document.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
         <xsd:element ref="MPV:NS-ENTRIES" minOccurs="1" maxOccurs="1" />
         <xsd:element ref="MPV:METADATA-OFFSETS" minOccurs="1" maxOccurs="1"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

<xsd:element name="MPV:METADATA-OFFSETS">
    <xsd:annotation>
      <xsd:documentation>
                Set of offsets to metadata in the overall XML document.
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
```

```
          <xsd:element ref="MPV:OFFSET" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
```

```
Complete MPV example

<?XML version="1.0" >
<MPV:MPV>
  <METADATA Type="Collection" MPV:ID="foo">
     <METADATA Type="Single" MPV:ID="bar">
        ...
     </METADATA>
     <METADATA Type="Single" MPV:ID="zow">
        ...
     </METADATA>
  </METADATA>
  <MPV:TOC xmlns:MPV="mpv-nsval">
     <MPV:METADATA-OFFSETS>
        <MPV:OFFSET type="Collection" byteref="2445" >
           <MPV:OFFSET type="Image" byteref="2456" />
           <MPV:OFFSET type="Image" byteref="2575" />
        </MPV:OFFSET>
     </MPV:METADATA-OFFSETS>
  </MPV:TOC>
</MPV:MPV>
<?MPV TOC-OFFSET="2653" DOC-SIZE="3333" MPV?>
```

M.5.3 MPV:NS-ENTRIES element

The MPV:NS-ENTRIES element is a container element for one or more NS-ENTRY elements. In practice, there will always be at least two entries, one for the default dig35 namespace and one for the explicit MPV namespace.

M.5.4 MPV:NS-ENTRY element

The MPV:NS-ENTRY element contains the information that allows a random access reader to generate the context needed to process the well-formed fragments that are referenced in the METADATA:OFFSETS array. Each NS-ENTRY specifies the namespace prefix and namespace of single namespace used in the document. Note that MPV documents are required to only bind a namespace prefix to a single namespace in a document.

```
<xsd:complexType name="MPV:NS-ENTRY">

  <xsd:attribute name="NS-PREFIX" type=xsd:string />
  <xsd:attribute name="NS-URI" type="xsd:string" />

</xsd:complexType>
```

M.5.5 MPV:METADATA-OFFSETS element

The MPV:METADATA-OFFSETS element contains one or more OFFSET elements

M.5.6 MPV:OFFSET Element

The MPV OFFSET element provides the information that allows an application to perform random access on the METADATA elements in the document. Each OFFSET element has two attributes, type and byteref which allow the application to determine where the METADATA element is located and what type it is.

The use of the 'type' attribute is an important hint for efficient processing because it allows the processor to rapidly traverse the content items in the MPV document according to type without parsing any of the information. This information is redundant to the definitive statement of type declared with the element itself.

The types of "Single" and "Image" have identical semantics. In MPV, the preferred term is "Image", not "Single".

```
MPV Structure Byte offsets schema

<xsd:complexType name="OFFSET">
   <xsd:sequence>
       <xsd:element name="OFFSET" type="MPV:OFFSET" minOccurs="0", maxOccurs="*"/>
   </xsd:sequence>

   <xsd:attribute name="Type" use="optional" default="Single">
       <xsd:simpleType>
           <xsd:restriction base="xsd:string">
               <xsd:enumeration value="Single"/>
               <xsd:enumeration value="Image"/>
               <xsd:enumeration value="Audio"/>
               <xsd:enumeration value="Video"/>
               <xsd:enumeration value="Document"/>
               <xsd:enumeration value="Collection"/>
           </xsd:restriction>
       </xsd:simpleType>
   </xsd:attribute>
   <xsd:attribute name="byteref" type="xsd:number" />
</xsd:complexType>
```

M.5.7 The MPV Processing Instruction

The MPV processing instruction is the last content in an MPV document. It contains the minimal information that allows an MPV reader to determine that the document is a valid MPV document (based on a file size check) and to locate the absolute offset in the document where the Table of Contents can be found.

The MPV processing instruction contains two properties named TOC-OFFSET and DOC-SIZE

Property Name	Property Value	Description
TOC-OFFSET	Xsd:Integer	Byte offset of the MPV:TOC
DOC-SIZE	Xsd:Integer	Total size in bytes of the document

Annex N: DIG35 XML Schema

The DIG35 XML Schema as specified in this annex is a normative and integral part of this specification to support the DIG35- recommended implementation. This schema shall be used by XML Schema-based tools and parsers to validate a DIG35 XML document. Either the DIG35 XML Schema or the DIG35 DTD may be used to validate a document.

NOTE: The DIG35 XML Schema conforms to W3C's XML Schema specification that has been recently promoted to a W3C Recommendation. While the DIG35 XML Schema follows currently known guidelines, it may not work correctly with XML Schema tools in the future. As such, appropriate modifications may be necessary. Note that this schema has been validated by the XML Schema DTD (http://www.w3.org/2001/XMLSchema.dtd) and XSV service provided by W3C (http://www.w3.org/2001/03/webdata/xsv).

Refer to the website http://www.digitalimaging.org for the latest information on the DIG35 XML schema.

N.1 DIG35 Complex Types

```
<?xml version="1.0" encoding="utf-8"?>
<!DOCTYPE xsd:schema PUBLIC "-//W3C//DTD XMLSchema 200102//EN"
"http://www.w3.org/2001/XMLSchema.dtd" [
  <!ENTITY % p 'xsd:'>
  <!ENTITY % s ':xsd'>
]>
<xsd:schema xmlns:xsd="http://www.w3.org/2001/XMLSchema"
                        xmlns="http://www.digitalimaging.org/dig35/1.1/xml"
                        xmlns:dig35="http://www.digitalimaging.org/dig35/1.1/xml"
                        xmlns:xml="http://www.w3.org/XML/1998/namespace"
                        targetNamespace="http://www.digitalimaging.org/dig35/1.1/xml"
                        elementFormDefault="qualified">

  <!-- Import the xml:lang attribute definition defined by W3C -->
  <xsd:import namespace="http://www.w3.org/XML/1998/namespace"
                                  schemaLocation="http://www.w3.org/2001/xml.xsd">
    <xsd:annotation>
      <xsd:documentation>Import to get access to xml:lang</xsd:documentation>
    </xsd:annotation>
  </xsd:import>

  <xsd:annotation>
    <xsd:documentation>
           This is the XML Schema for DIG35 Metadata Version 1.1 (20010618).
           The specification for DIG35 that corresponds to this XML Schema
           can be found on the Digital Imaging Group website
(http://www.digitalimaging.org).

           Copyright (c) 2000-2001 Digital Imaging Group, All Rights Reserved.
       </xsd:documentation>
  </xsd:annotation>
```

```
  <xsd:element name="METADATA">
    <xsd:annotation>
      <xsd:documentation>
          See section Annex G: DIG35 XML Document Definition
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:BASIC IMAGE PARAM" minOccurs="0"/>
        <xsd:element ref="dig35:IMAGE CREATION" minOccurs="0"/>
        <xsd:element ref="dig35:CONTENT DESCRIPTION" minOccurs="0"/>
        <xsd:element ref="dig35:HISTORY" minOccurs="0"/>
        <xsd:element ref="dig35:IPR" minOccurs="0"/>
        <xsd:element ref="dig35:METADATA" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute name="TYPE" use="optional" default="Single">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="Single"/>
            <xsd:enumeration value="Collection"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang" use="optional" default="en"/>
    </xsd:complexType>
  </xsd:element>


  <xsd:annotation>
    <xsd:documentation> Annex F:  Fundamental Metadata Types and Fields
</xsd:documentation>
  </xsd:annotation>

  <xsd:simpleType name="tNonNegativeDouble">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.1  Non-negative Double Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:double">
      <xsd:minInclusive value="0"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="tRational">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.2  Rational Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:string">
      <xsd:pattern value="(\-|\+)?[0-9]+/[0-9]+"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:complexType name="tLangString">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.3  String Including Language Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:simpleContent>
      <xsd:extension base="xsd:string">
```

```
        <xsd:attribute ref="xml:lang"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>

  <xsd:simpleType name="tDegree">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.4  Degree Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:double">
      <xsd:minExclusive value="-180"/>
      <xsd:maxInclusive value="180"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:simpleType name="tHalfDegree">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.5  Half Degree Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:double">
      <xsd:minExclusive value="-90"/>
      <xsd:maxInclusive value="90"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:complexType name="tDoubleSize">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.6  Double Size Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="WIDTH" type="dig35:tNonNegativeDouble"/>
      <xsd:element name="HEIGHT" type="dig35:tNonNegativeDouble"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="tIntSize">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.7  Integer Size Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="WIDTH" type="xsd:positiveInteger"/>
      <xsd:element name="HEIGHT" type="xsd:positiveInteger"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="tDateTime">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.8  DateTime Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:choice minOccurs="0">
        <xsd:element name="EXACT" type="xsd:dateTime"/>
        <xsd:element name="DATE" type="xsd:date"/>
        <xsd:sequence>
```

```
          <xsd:element name="MONTH" type="dig35:tRecurringMonth" minOccurs="0"/>
          <xsd:element name="YEAR" type="xsd:gYear" minOccurs="0"/>
          <xsd:element name="CENTURY" type="dig35:tCentury" minOccurs="0"/>
        </xsd:sequence>
      </xsd:choice>
      <xsd:element name="WEEK DAY" type="dig35:tLangString" minOccurs="0"/>
      <xsd:element name="SEASON" type="dig35:tLangString" minOccurs="0"/>
      <xsd:element ref="dig35:COMMENT" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang"/>
  </xsd:complexType>
  <xsd:simpleType name="tCentury">
    <xsd:annotation>
      <xsd:documentation>
           A type that contains a century; see section F.2.8 for details
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:integer"/>
  </xsd:simpleType>
  <xsd:simpleType name="tRecurringMonth">
    <xsd:annotation>
      <xsd:documentation>
           A type that contains a month of the year; see section F.2.8 for details
      </xsd:documentation>
    </xsd:annotation>
    <xsd:restriction base="xsd:positiveInteger">
      <xsd:minInclusive value="1"/>
      <xsd:maxInclusive value="12"/>
    </xsd:restriction>
  </xsd:simpleType>

  <xsd:complexType name="tAddress">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.9  Address Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="ADDR NAME" type="dig35:tLangString" minOccurs="0"/>
      <xsd:element ref="dig35:ADDR COMP" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:choice minOccurs="0">
        <xsd:element name="ZIPCODE" type="xsd:string"/>
        <xsd:element name="POSTCODE" type="xsd:string"/>
      </xsd:choice>
      <xsd:element name="COUNTRY" type="dig35:tLangString" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="TYPE" type="xsd:string"/>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang"/>
  </xsd:complexType>
  <xsd:element name="ADDR COMP">
    <xsd:annotation>
      <xsd:documentation>
                A name component of an address; not intended for public use
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:simpleContent>
        <xsd:extension base="dig35:tLangString">
          <xsd:attribute name="TYPE" type="xsd:string"/>
        </xsd:extension>
      </xsd:simpleContent>
    </xsd:complexType>
```

```
  </xsd:element>

  <xsd:complexType name="tPhone">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.10  Phone Number Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="COUNTRY CODE" type="xsd:string" minOccurs="0"/>
      <xsd:element name="AREA" type="xsd:string" minOccurs="0"/>
      <xsd:element name="LOCAL" type="xsd:string" minOccurs="0"/>
      <xsd:element name="EXTENSION" type="xsd:string" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="TYPE" type="xsd:string"/>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
  </xsd:complexType>

  <xsd:complexType name="tEmail">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.11  Email Address Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:simpleContent>
      <xsd:extension base="dig35:tLangString">
        <xsd:attribute name="TYPE" type="xsd:string"/>
        <xsd:attribute ref="dig35:TIMESTAMP"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>

  <xsd:complexType name="tWeb">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.12  Web Address Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:simpleContent>
      <xsd:extension base="dig35:tLangString">
        <xsd:attribute name="TYPE" type="xsd:string"/>
        <xsd:attribute ref="dig35:TIMESTAMP"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>

  <xsd:complexType name="tPerson">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.13  Person Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="NAME TITLE" type="dig35:tLangString" minOccurs="0"/>
      <xsd:element ref="dig35:PERSON NAME" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="NICKNAME" type="dig35:tLangString" minOccurs="0"
maxOccurs="unbounded"/>
      <xsd:element name="JOB TITLE" type="dig35:tLangString" minOccurs="0"/>
      <xsd:choice minOccurs="0">
        <xsd:element name="PERSON ORG" type="dig35:tOrganization"/>
        <xsd:element name="ORG REF" type="xsd:string"/>
      </xsd:choice>
      <xsd:element name="ADDRESS" type="dig35:tAddress" minOccurs="0"
maxOccurs="unbounded"/>
```



```
      <xsd:element name="PHONE" type="dig35:tPhone" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="EMAIL" type="dig35:tEmail" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="WEB" type="dig35:tWeb" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="BIRTH DATE" type="xsd:date" minOccurs="0"/>
      <xsd:element name="AGE" type="xsd:duration" minOccurs="0"/>
      <xsd:element ref="dig35:COMMENT" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="ID" type="xsd:string"/>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang"/>
  </xsd:complexType>
  <xsd:element name="PERSON NAME">
    <xsd:annotation>
      <xsd:documentation>
                Defined in F.2.13  Person Type to specify a person's name
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:NAME COMP" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="NAME COMP">
    <xsd:annotation>
      <xsd:documentation>
                A name component of person's name; not intended for public use
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:simpleContent>
        <xsd:extension base="xsd:string">
          <xsd:attribute name="TYPE" use="optional" default="Given">
            <xsd:simpleType>
              <xsd:restriction base="xsd:string">
                <xsd:enumeration value="Prefix"/>
                <xsd:enumeration value="Given"/>
                <xsd:enumeration value="Family"/>
                <xsd:enumeration value="Suffix"/>
                <xsd:enumeration value="Maiden"/>
              </xsd:restriction>
            </xsd:simpleType>
          </xsd:attribute>
        </xsd:extension>
      </xsd:simpleContent>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="tOrganization">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.14  Organization Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="ORG NAME" type="dig35:tLangString" minOccurs="0"/>
      <xsd:element name="ADDRESS" type="dig35:tAddress" minOccurs="0"
maxOccurs="unbounded"/>
      <xsd:element name="PHONE" type="dig35:tPhone" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="EMAIL" type="dig35:tEmail" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="WEB" type="dig35:tWeb" minOccurs="0" maxOccurs="unbounded"/>
      <xsd:element name="LOGO_FILE" type="xsd:anyURI" minOccurs="0"/>
```

```
      <xsd:element name="LOGO FORMAT" type="xsd:string" minOccurs="0"/>
      <xsd:element name="MIME TYPE" type="xsd:string" minOccurs="0"/>
      <xsd:element ref="dig35:COMMENT" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute name="ID" type="xsd:string"/>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang"/>
  </xsd:complexType>

  <xsd:complexType name="tLocation">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.15  Location Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element ref="dig35:COORD LOC" minOccurs="0"/>
      <xsd:element name="ADDRESS" type="dig35:tAddress" minOccurs="0"/>
      <xsd:element ref="dig35:GPS" minOccurs="0"/>
      <xsd:element ref="dig35:COMMENT" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang"/>
  </xsd:complexType>

  <xsd:element name="COORD LOC">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.15.1  Coordinate Location
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="LONGITUDE" type="dig35:tDegree" minOccurs="0"/>
        <xsd:element name="LATITUDE" type="dig35:tHalfDegree" minOccurs="0"/>
        <xsd:element name="ALTITUDE" type="xsd:double" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GPS">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.15.2  Raw GPS Information
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="GPS LAT REF" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:enumeration value="N"/>
              <xsd:enumeration value="S"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="GPS LATITUDE" type="dig35:tDms" minOccurs="0"/>
        <xsd:element name="GPS LONG REF" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:enumeration value="E"/>
              <xsd:enumeration value="W"/>
            </xsd:restriction>
```

```
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="GPS LONGITUDE" type="dig35:tDms" minOccurs="0"/>
        <xsd:element name="GPS ALTITUDE" type="dig35:tNonNegativeDouble" minOccurs="0"/>
        <xsd:element name="GPS TIME" type="xsd:dateTime" minOccurs="0"/>
        <xsd:element name="GPS SATELLITES" type="xsd:string" minOccurs="0"/>
        <xsd:element name="GPS STATUS" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:enumeration value="A"/>
              <xsd:enumeration value="V"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="GPS MEASURE MODE" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:positiveInteger">
              <xsd:minExclusive value="2"/>
              <xsd:maxInclusive value="3"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="GPS DOP" type="dig35:tNonNegativeDouble" minOccurs="0"/>
        <xsd:element name="GPS SPEED REF" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:enumeration value="K"/>
              <xsd:enumeration value="N"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="GPS SPEED" type="dig35:tNonNegativeDouble" minOccurs="0"/>
        <xsd:element name="GPS TRACK REF" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:enumeration value="T"/>
              <xsd:enumeration value="M"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="GPS TRACK" type="dig35:tNonNegativeDouble" minOccurs="0"/>
        <xsd:element name="GPS IMAGE DIR REF" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:enumeration value="T"/>
              <xsd:enumeration value="M"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="GPS IMAGE DIR" type="dig35:tNonNegativeDouble" minOccurs="0"/>
        <xsd:element name="GPS MAP DATUM" type="xsd:string" minOccurs="0"/>
        <xsd:element name="GPS DEST LAT REF" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:enumeration value="N"/>
              <xsd:enumeration value="S"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="GPS DEST LATITUDE" type="dig35:tDms" minOccurs="0"/>
        <xsd:element name="GPS DEST LONG REF" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:enumeration value="E"/>
```

```
              <xsd:enumeration value="W"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="GPS DEST LONGITUDE" type="dig35:tDms" minOccurs="0"/>
        <xsd:element name="GPS DEST BEARING REF" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:enumeration value="T"/>
              <xsd:enumeration value="M"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="GPS DEST BEARING" type="dig35:tNonNegativeDouble"
minOccurs="0"/>
        <xsd:element name="GPS DEST DISTANCE REF" minOccurs="0">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:enumeration value="K"/>
              <xsd:enumeration value="N"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
        <xsd:element name="GPS DEST DISTANCE" type="dig35:tNonNegativeDouble"
minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:complexType name="tDms">
    <xsd:annotation>
      <xsd:documentation>
                A utility type defined for GPS only; not intended for public use
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="D" type="xsd:nonNegativeInteger"/>
      <xsd:element name="M" type="xsd:nonNegativeInteger"/>
      <xsd:element name="S" type="dig35:tNonNegativeDouble" minOccurs="0"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="tDirection">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.16  Direction Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="YAW" type="dig35:tDegree" minOccurs="0"/>
      <xsd:element name="PITCH" type="dig35:tHalfDegree" minOccurs="0"/>
      <xsd:element name="ROLL" type="dig35:tDegree" minOccurs="0"/>
      <xsd:element ref="dig35:COMMENT" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang"/>
  </xsd:complexType>

  <xsd:complexType name="tPosition">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.17  Position Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
```

```
      <xsd:choice minOccurs="0">
        <xsd:element name="POINT" type="dig35:tPoint"/>
        <xsd:element name="RECT" type="dig35:tRect"/>
        <xsd:sequence>
          <xsd:element name="RECT" type="dig35:tRect"/>
          <xsd:element name="REGION" type="dig35:tRegion"/>
        </xsd:sequence>
      </xsd:choice>
      <xsd:element ref="dig35:COMMENT" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang"/>
  </xsd:complexType>

  <xsd:complexType name="tPoint">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.17.1  Point Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="X" type="dig35:tNonNegativeDouble"/>
      <xsd:element name="Y" type="dig35:tNonNegativeDouble"/>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:complexType name="tRect">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.17.2  Rectangle Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexContent>
      <xsd:extension base="dig35:tPoint">
        <xsd:sequence>
          <xsd:element name="WIDTH" type="dig35:tNonNegativeDouble"/>
          <xsd:element name="HEIGHT" type="dig35:tNonNegativeDouble"/>
        </xsd:sequence>
      </xsd:extension>
    </xsd:complexContent>
  </xsd:complexType>

  <xsd:complexType name="tRegion">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.17.3  Region Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="POINT" type="dig35:tPoint"/>
      <xsd:choice minOccurs="0" maxOccurs="unbounded">
        <xsd:element name="POINT" type="dig35:tPoint"/>
        <xsd:element ref="dig35:SPLINE"/>
      </xsd:choice>
    </xsd:sequence>
  </xsd:complexType>

  <xsd:element name="SPLINE">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.17.3  Region Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
```

```
      <xsd:sequence>
        <xsd:element name="X1" type="dig35:tNonNegativeDouble"/>
        <xsd:element name="Y1" type="dig35:tNonNegativeDouble"/>
        <xsd:element name="X2" type="dig35:tNonNegativeDouble"/>
        <xsd:element name="Y2" type="dig35:tNonNegativeDouble"/>
        <xsd:element name="X" type="dig35:tNonNegativeDouble"/>
        <xsd:element name="Y" type="dig35:tNonNegativeDouble"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:complexType name="tProductDetails">
    <xsd:annotation>
      <xsd:documentation>
                See section F.2.18  Product Details Type
      </xsd:documentation>
    </xsd:annotation>
    <xsd:sequence>
      <xsd:element name="MANUFACTURER" type="dig35:tOrganization" minOccurs="0"/>
      <xsd:element name="MODEL" type="xsd:string" minOccurs="0"/>
      <xsd:element name="SERIAL" type="xsd:string" minOccurs="0"/>
      <xsd:element name="VERSION" type="xsd:string" minOccurs="0"/>
    </xsd:sequence>
    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang"/>
  </xsd:complexType>

  <xsd:attribute name="TIMESTAMP" type="xsd:dateTime">
    <xsd:annotation>
      <xsd:documentation>
                See section F.3.2  Timestamp Attribute
      </xsd:documentation>
    </xsd:annotation>
  </xsd:attribute>

  <xsd:element name="COMMENT">
    <xsd:annotation>
      <xsd:documentation>
                See section F.4.1  Comment
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:simpleContent>
        <xsd:extension base="dig35:tLangString">
          <xsd:attribute ref="dig35:TIMESTAMP"/>
        </xsd:extension>
      </xsd:simpleContent>
    </xsd:complexType>
  </xsd:element>


  <xsd:annotation>
    <xsd:documentation>Annex A:  Basic Image Parameter Metadata</xsd:documentation>
  </xsd:annotation>

  <xsd:element name="BASIC IMAGE PARAM">
    <xsd:annotation>
      <xsd:documentation>
                See section Annex A:  Basic Image Parameter Metadata
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:BASIC_IMAGE_INFO" minOccurs="0"/>
```

```
        <xsd:element ref="dig35:PREF PRESENTATION PARAM" minOccurs="0"/>
        <xsd:element ref="dig35:COLOR INFO" minOccurs="0"/>
        <xsd:element ref="dig35:COMPONENT INFO" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="BASIC IMAGE INFO">
    <xsd:annotation>
      <xsd:documentation>
                See section A.3.1  Basic Image Information
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:FILE FORMAT" minOccurs="0"/>
        <xsd:element ref="dig35:IMAGE ID" minOccurs="0"/>
        <xsd:element ref="dig35:IMAGE SIZE" minOccurs="0"/>
        <xsd:element ref="dig35:COMPRESSION" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="FILE FORMAT">
    <xsd:annotation>
      <xsd:documentation>
                See section A.3.1.1  File and Format
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="FILE NAME" type="xsd:anyURI" minOccurs="0"/>
        <xsd:element name="FORMAT TYPE" type="xsd:string" minOccurs="0"/>
        <xsd:element name="MIME TYPE" type="xsd:string" minOccurs="0"/>
        <xsd:element name="VERSION" type="xsd:string" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IMAGE ID">
    <xsd:annotation>
      <xsd:documentation>
                See section A.3.1.2  Image Identifier
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="UID" type="xsd:string" minOccurs="0"/>
        <xsd:element name="ID TYPE" type="xsd:anyURI" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IMAGE SIZE" type="dig35:tIntSize">
    <xsd:annotation>
      <xsd:documentation>
                See section A.3.1.3  Image Size
      </xsd:documentation>
    </xsd:annotation>
  </xsd:element>
```

```
  <xsd:element name="COMPRESSION" type="xsd:string">
    <xsd:annotation>
      <xsd:documentation>
                See section A.3.1.4  Compression Method
      </xsd:documentation>
    </xsd:annotation>
  </xsd:element>

  <xsd:element name="PREF PRESENTATION PARAM" type="dig35:tDoubleSize">
    <xsd:annotation>
      <xsd:documentation>
                See section A.3.2  Preferred Presentation Parameters
      </xsd:documentation>
    </xsd:annotation>
  </xsd:element>

  <xsd:element name="COLOR INFO">
    <xsd:annotation>
      <xsd:documentation>
                See section A.3.3  Color Information
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:COLORSPACE" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="COLORSPACE">
    <xsd:annotation>
      <xsd:documentation>
                See section A.3.3.1  Colorspace
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="PROFILE NAME" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="PROFILE REF" type="xsd:anyURI" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="COMPONENT INFO">
    <xsd:annotation>
      <xsd:documentation>
                See section A.3.4  Component Information
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="NUM COMPONENT" type="xsd:positiveInteger" minOccurs="0"/>
        <xsd:element name="PREMULTIPLIED" type="xsd:boolean" minOccurs="0"/>
        <xsd:element name="COMPONENTS" type="xsd:string" minOccurs="0"/>
        <xsd:element name="SAMPLE RATIO" type="xsd:string" minOccurs="0"/>
        <xsd:element name="PIXEL COMP" type="xsd:string" minOccurs="0"/>
        <xsd:element name="COMP SIZE" type="xsd:positiveInteger" minOccurs="0"
maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
```



```
  </xsd:element>


  <xsd:annotation>
    <xsd:documentation> Annex B:  Image Creation Metadata </xsd:documentation>
  </xsd:annotation>


  <xsd:element name="IMAGE CREATION">
    <xsd:annotation>
      <xsd:documentation>
                See section Annex B:  Image Creation Metadata
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:GENERAL CREATION INFO" minOccurs="0"/>
        <xsd:element ref="dig35:CAMERA CAPTURE" minOccurs="0"/>
        <xsd:element ref="dig35:SCANNER CAPTURE" minOccurs="0"/>
        <xsd:element ref="dig35:SOFTWARE CREATION" minOccurs="0"/>
        <xsd:element ref="dig35:CAPTURED ITEM" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="GENERAL CREATION INFO">
    <xsd:annotation>
      <xsd:documentation>
                See section B.3.1  General Creation Information
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="CREATION TIME" type="xsd:dateTime" minOccurs="0"/>
        <xsd:element name="IMAGE SOURCE" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="SCENE TYPE" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="IMAGE CREATOR" type="dig35:tPerson" minOccurs="0"/>
        <xsd:element name="OPERATOR ORG" type="dig35:tOrganization" minOccurs="0"/>
        <xsd:element name="OPERATOR ID" type="dig35:tLangString" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="CAMERA CAPTURE">
    <xsd:annotation>
      <xsd:documentation>
                See section B.3.2  Camera Capture
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="CAMERA INFO" type="dig35:tProductDetails" minOccurs="0"/>
        <xsd:element name="SOFTWARE INFO" type="dig35:tProductDetails" minOccurs="0"/>
        <xsd:element name="LENS INFO" type="dig35:tProductDetails" minOccurs="0"/>
        <xsd:element ref="dig35:DEVICE CHARACTER" minOccurs="0"/>
        <xsd:element ref="dig35:CAMERA SETTINGS" minOccurs="0"/>
        <xsd:element name="ACCESSORY" type="dig35:tProductDetails" minOccurs="0"
maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
```



```
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="DEVICE CHARACTER">
    <xsd:annotation>
      <xsd:documentation>
                See section B.3.2.4  Device Characterization
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:SENSOR TECHNOLOGY" minOccurs="0"/>
        <xsd:element name="FOCAL PLANE RES" type="dig35:tDoubleSize" minOccurs="0"/>
        <xsd:element name="SPECTRAL SENSITIVITY" type="xsd:string" minOccurs="0"/>
        <xsd:element name="ISO SATURATION" type="dig35:tNonNegativeDouble" minOccurs="0"/>
        <xsd:element name="ISO NOISE" type="dig35:tNonNegativeDouble" minOccurs="0"/>
        <xsd:element ref="dig35:SPATIAL FREQ RESPONSE" minOccurs="0"/>
        <xsd:element ref="dig35:CFA PATTERN" minOccurs="0"/>
        <xsd:element ref="dig35:OECF" minOccurs="0"/>
        <xsd:element name="MIN F NUMBER" type="dig35:tNonNegativeDouble" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="SENSOR TECHNOLOGY">
    <xsd:annotation>
      <xsd:documentation>
                Intended to be used with DEVICE CHARACTER; see section B.3.2.4  Device
Characterization
      </xsd:documentation>
    </xsd:annotation>
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="One-Chip Color Area"/>
        <xsd:enumeration value="Two-Chip Color Area"/>
        <xsd:enumeration value="Three-Chip Color Area"/>
        <xsd:enumeration value="Color Sequential Area"/>
        <xsd:enumeration value="Trilinear"/>
        <xsd:enumeration value="Color Sequential Linear Sensor"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:element>
  <xsd:element name="SPATIAL FREQ RESPONSE">
    <xsd:annotation>
      <xsd:documentation>
                Intended to be used with DEVICE CHARACTER; see section B.3.2.4  Device
Characterization
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:SPATIAL FREQ VAL" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="SPATIAL FREQ VAL">
    <xsd:annotation>
      <xsd:documentation>
                Intended to be used with SPATIAL FREQ RESPONSE; see section B.3.2.4  Device
Characterization
      </xsd:documentation>
    </xsd:annotation>
```

```
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="SPATIAL FREQ" type="dig35:tNonNegativeDouble"/>
        <xsd:element name="HORIZ SFR" type="dig35:tNonNegativeDouble"/>
        <xsd:element name="VERT SFR" type="dig35:tNonNegativeDouble"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="CFA PATTERN">
    <xsd:annotation>
      <xsd:documentation>
                Intended to be used with DEVICE CHARACTER; see section B.3.2.4  Device
Characterization
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:COLOR ROW" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="COLOR ROW">
    <xsd:annotation>
      <xsd:documentation>
                Intended to be used with CFA PATTERN; see section B.3.2.4  Device
Characterization
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="COLOR" maxOccurs="unbounded">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:enumeration value="Red"/>
              <xsd:enumeration value="Green"/>
              <xsd:enumeration value="Blue"/>
              <xsd:enumeration value="Cyan"/>
              <xsd:enumeration value="Magenta"/>
              <xsd:enumeration value="Yellow"/>
              <xsd:enumeration value="White"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:element>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="OECF">
    <xsd:annotation>
      <xsd:documentation>
                Intended to be used with DEVICE CHARACTER; see section B.3.2.4  Device
Characterization
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:LOG VAL" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="LOG VAL">
    <xsd:annotation>
      <xsd:documentation>
                Intended to be used with OECF; see section B.3.2.4  Device Characterization
      </xsd:documentation>
```

```
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="LOG EXPOSURE" type="xsd:double"/>
        <xsd:element name="OUTPUT LEVEL" type="dig35:tNonNegativeDouble"
maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="CAMERA SETTINGS">
    <xsd:annotation>
      <xsd:documentation>
                See section B.3.2.5  Camera Capture Settings
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:choice minOccurs="0">
          <xsd:element name="EXP TIME" type="dig35:tNonNegativeDouble"/>
          <xsd:element name="R EXP TIME" type="dig35:tRational"/>
        </xsd:choice>
        <xsd:element name="F NUMBER" type="dig35:tNonNegativeDouble" minOccurs="0"/>
        <xsd:element name="EXP PROGRAM" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="BRIGHTNESS" type="xsd:double" minOccurs="0"/>
        <xsd:element name="EXPOSURE BIAS" type="xsd:double" minOccurs="0"/>
        <xsd:element name="SUBJECT DISTANCE" type="dig35:tNonNegativeDouble"
minOccurs="0"/>
        <xsd:element name="METERING MODE" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="SCENE ILLUMINANT" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="COLOR TEMP" type="dig35:tNonNegativeDouble" minOccurs="0"/>
        <xsd:element name="FOCAL LENGTH" type="dig35:tNonNegativeDouble" minOccurs="0"/>
        <xsd:element name="FLASH" type="xsd:boolean" minOccurs="0"/>
        <xsd:element name="FLASH ENERGY" type="dig35:tNonNegativeDouble" minOccurs="0"/>
        <xsd:element name="FLASH RETURN" type="xsd:boolean" minOccurs="0"/>
        <xsd:element ref="dig35:BACK LIGHT" minOccurs="0"/>
        <xsd:element name="SUBJECT POSITION" type="dig35:tPosition" minOccurs="0"/>
        <xsd:element name="EXPOSURE INDEX" type="xsd:double" minOccurs="0"/>
        <xsd:element ref="dig35:AUTO FOCUS" minOccurs="0"/>
        <xsd:element ref="dig35:SPECIAL EFFECT" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element name="CAMERA LOCATION" type="dig35:tLocation" minOccurs="0"/>
        <xsd:element name="ORIENTATION" type="dig35:tDirection" minOccurs="0"/>
        <xsd:element name="PAR" type="dig35:tRational" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="BACK LIGHT">
    <xsd:annotation>
      <xsd:documentation>
                Intended to be used with CAMERA SETTINGS; see section B.3.2.5  Camera
Capture Settings
      </xsd:documentation>
    </xsd:annotation>
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="Front Light"/>
        <xsd:enumeration value="Back Light 1"/>
        <xsd:enumeration value="Back Light 2"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:element>
  <xsd:element name="AUTO_FOCUS">
```

```
    <xsd:annotation>
      <xsd:documentation>
                Intended to be used with CAMERA SETTINGS; see section B.3.2.5  Camera
Capture Settings
      </xsd:documentation>
    </xsd:annotation>
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="Auto Focus Used"/>
        <xsd:enumeration value="Auto Focus Interrupted"/>
        <xsd:enumeration value="Near Focused"/>
        <xsd:enumeration value="Soft Focused"/>
        <xsd:enumeration value="Manual"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:element>
  <xsd:element name="SPECIAL EFFECT">
    <xsd:annotation>
      <xsd:documentation>
                Intended to be used with CAMERA SETTINGS; see section B.3.2.5  Camera
Capture Settings
      </xsd:documentation>
    </xsd:annotation>
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="Colored"/>
        <xsd:enumeration value="Diffusion"/>
        <xsd:enumeration value="Multi-Image"/>
        <xsd:enumeration value="Polarizing"/>
        <xsd:enumeration value="Split-Field"/>
        <xsd:enumeration value="Star"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:element>

  <xsd:element name="SCANNER CAPTURE">
    <xsd:annotation>
      <xsd:documentation>
                See section B.3.3  Scanner Capture
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="SCANNER INFO" type="dig35:tProductDetails" minOccurs="0"/>
        <xsd:element name="SOFTWARE INFO" type="dig35:tProductDetails" minOccurs="0"/>
        <xsd:element ref="dig35:SCANNER SETTINGS" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SCANNER SETTINGS">
    <xsd:annotation>
      <xsd:documentation>
                See section B.3.3.3  Scanner Capture Settings
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="PIXEL SIZE" type="dig35:tNonNegativeDouble" minOccurs="0"/>
        <xsd:element name="PHYSICAL SCAN RES" type="dig35:tDoubleSize" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
```



```
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="SOFTWARE CREATION">
    <xsd:annotation>
      <xsd:documentation>
                See section B.3.4  Software Creation
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="SOFTWARE INFO" type="dig35:tProductDetails"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="CAPTURED ITEM">
    <xsd:annotation>
      <xsd:documentation>
                See section B.3.5  Captured Item
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:choice>
        <xsd:element ref="dig35:REFLECTION PRINT"/>
        <xsd:element ref="dig35:FILM"/>
      </xsd:choice>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="REFLECTION PRINT">
    <xsd:annotation>
      <xsd:documentation>
                See section B.3.5.1  Reflection Print
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="DOCUMENT SIZE" type="dig35:tDoubleSize" minOccurs="0"/>
        <xsd:element ref="dig35:MEDIUM" minOccurs="0"/>
        <xsd:element ref="dig35:RP TYPE" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="MEDIUM">
    <xsd:annotation>
      <xsd:documentation>
                Intended to be used with REFLECTION PRINT; see section B.3.5.1  Reflection
Print
      </xsd:documentation>
    </xsd:annotation>
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="Continuous Tone Image"/>
        <xsd:enumeration value="Halftone Image"/>
        <xsd:enumeration value="Line Art"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:element>
  <xsd:element name="RP TYPE">
    <xsd:annotation>
      <xsd:documentation>
```

```
                Intended to be used with REFLECTION PRINT; see section B.3.5.1  Reflection
Print
      </xsd:documentation>
    </xsd:annotation>
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="B/W Print"/>
        <xsd:enumeration value="Color Print"/>
        <xsd:enumeration value="B/W Document"/>
        <xsd:enumeration value="Color Document"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:element>

  <xsd:element name="FILM">
    <xsd:annotation>
      <xsd:documentation>
                See section B.3.5.2  Film
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="BRAND" type="dig35:tProductDetails" minOccurs="0"/>
        <xsd:element ref="dig35:CATEGORY" minOccurs="0"/>
        <xsd:element name="FILM SIZE" type="dig35:tDoubleSize" minOccurs="0"/>
        <xsd:element name="ROLL ID" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="FRAME ID" type="xsd:positiveInteger" minOccurs="0"/>
        <xsd:element name="FILM SPEED" type="xsd:positiveInteger" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="CATEGORY">
    <xsd:annotation>
      <xsd:documentation>
                Intended to be used with FILM; see section B.3.5.2  Film
      </xsd:documentation>
    </xsd:annotation>
    <xsd:simpleType>
      <xsd:restriction base="xsd:string">
        <xsd:enumeration value="Negative B/W"/>
        <xsd:enumeration value="Negative Color"/>
        <xsd:enumeration value="Reversal B/W"/>
        <xsd:enumeration value="Reversal Color"/>
        <xsd:enumeration value="Chromagenic"/>
        <xsd:enumeration value="Internegative B/W"/>
        <xsd:enumeration value="Internegative Color"/>
      </xsd:restriction>
    </xsd:simpleType>
  </xsd:element>


  <xsd:annotation>
    <xsd:documentation> Annex C:  Content Description Metadata </xsd:documentation>
  </xsd:annotation>


  <xsd:element name="CONTENT DESCRIPTION">
    <xsd:annotation>
      <xsd:documentation>
                See section Annex C:  Content Description Metadata
      </xsd:documentation>
    </xsd:annotation>
```

```
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="GROUP CAPTION" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="CAPTION" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="CAPTURE TIME" type="dig35:tDateTime" minOccurs="0"/>
        <xsd:element name="LOCATION" type="dig35:tLocation" minOccurs="0"/>
        <xsd:element ref="dig35:PERSON" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dig35:THING" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dig35:ORGANIZATION" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dig35:EVENT" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dig35:AUDIO" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dig35:PROPERTY" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dig35:DICTIONARY" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dig35:COMMENT" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="PERSON">
    <xsd:annotation>
      <xsd:documentation>
                See section C.3.5  Person Description
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:complexContent>
        <xsd:extension base="dig35:tPerson">
          <xsd:sequence>
            <xsd:element name="POSITION" type="dig35:tPosition" minOccurs="0"/>
            <xsd:element name="LOCATION" type="dig35:tLocation" minOccurs="0"/>
            <xsd:element ref="dig35:PROPERTY" minOccurs="0" maxOccurs="unbounded"/>
          </xsd:sequence>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="THING">
    <xsd:annotation>
      <xsd:documentation>
                See section C.3.6  Thing Description
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="NAME" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element ref="dig35:COMMENT" minOccurs="0"/>
        <xsd:element name="POSITION" type="dig35:tPosition" minOccurs="0"/>
        <xsd:element name="LOCATION" type="dig35:tLocation" minOccurs="0"/>
        <xsd:element ref="dig35:PROPERTY" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dig35:THING" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute name="ID" type="xsd:string"/>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="ORGANIZATION">
    <xsd:annotation>
      <xsd:documentation>
                See section C.3.7  Organization Description
```

```
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:complexContent>
        <xsd:extension base="dig35:tOrganization">
          <xsd:sequence>
            <xsd:element name="POSITION" type="dig35:tPosition" minOccurs="0"/>
            <xsd:element name="LOCATION" type="dig35:tLocation" minOccurs="0"/>
            <xsd:element ref="dig35:PROPERTY" minOccurs="0" maxOccurs="unbounded"/>
          </xsd:sequence>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="EVENT">
    <xsd:annotation>
      <xsd:documentation>
                See section C.3.8  Event Description
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="EVENT TYPE" type="dig35:tLangString"/>
        <xsd:element name="DESCRIPTION" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="LOCATION" type="dig35:tLocation" minOccurs="0"/>
        <xsd:element name="EVENT TIME" type="dig35:tDateTime" minOccurs="0"/>
        <xsd:element name="DURATION" type="xsd:duration" minOccurs="0"/>
        <xsd:element ref="dig35:COMMENT" minOccurs="0"/>
        <xsd:element ref="dig35:PARTICIPANT" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:element ref="dig35:EVENT RELATION" minOccurs="0" maxOccurs="unbounded"/>
        <xsd:choice minOccurs="0" maxOccurs="unbounded">
          <xsd:element ref="dig35:EVENT"/>
          <xsd:element name="EVENT REF" type="xsd:string"/>
        </xsd:choice>
      </xsd:sequence>
      <xsd:attribute name="ID" type="xsd:string"/>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="PARTICIPANT">
    <xsd:annotation>
      <xsd:documentation>
                Intended to be used with EVENT; see section C.3.8  Event Description
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="ROLE" type="dig35:tLangString" minOccurs="0"
maxOccurs="unbounded"/>
        <xsd:choice>
          <xsd:element name="OBJECT REF" type="xsd:string"/>
          <xsd:element ref="PERSON"/>
          <xsd:element ref="THING"/>
          <xsd:element ref="ORGANIZATION"/>
        </xsd:choice>
      </xsd:sequence>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="EVENT RELATION">
    <xsd:annotation>
      <xsd:documentation>
```

```
                Intended to be used with EVENT; see section C.3.8  Event Description
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="RELATION" type="dig35:tLangString" minOccurs="0"
maxOccurs="unbounded"/>
        <xsd:element name="EVENT REF" type="xsd:string" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="AUDIO">
    <xsd:annotation>
      <xsd:documentation>
                See section C.3.9  Audio
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="AUDIO STREAM" type="xsd:anyURI"/>
        <xsd:element name="AUDIO FORMAT" type="xsd:string" minOccurs="0"/>
        <xsd:element name="MIME TYPE" type="xsd:string" minOccurs="0"/>
        <xsd:element name="DESCRIPTION" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element ref="dig35:COMMENT" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="PROPERTY">
    <xsd:annotation>
      <xsd:documentation>
                See section C.3.10  Property
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="NAME" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="VALUE" type="dig35:tLangString" minOccurs="0"
maxOccurs="unbounded"/>
        <xsd:element ref="dig35:COMMENT" minOccurs="0"/>
        <xsd:element ref="dig35:PROPERTY" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute name="DICT REF" type="xsd:string"/>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="DICTIONARY">
    <xsd:annotation>
      <xsd:documentation>
                See section C.3.11  Dictionary Definition
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="DICT NAME" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element ref="dig35:COMMENT" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute name="ID" type="xsd:string"/>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
```



```
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>


  <xsd:annotation>
    <xsd:documentation> Annex H:  History Metadata </xsd:documentation>
  </xsd:annotation>


  <xsd:element name="HISTORY">
    <xsd:annotation>
      <xsd:documentation>
                See section Annex D:  History Metadata
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:PROCESSING SUMMARY" minOccurs="0"/>
        <xsd:element ref="dig35:IMAGE PROCESSING HINTS" minOccurs="0"/>
        <xsd:element ref="dig35:METADATA" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="PROCESSING SUMMARY">
    <xsd:annotation>
      <xsd:documentation>
                See section D.3.1  Processing Summary
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="IMG CREATED" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="IMG CROPPED" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="IMG TRANSFORMED" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="IMG GTC ADJ" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="IMG STC ADJ" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="IMG SPATIAL ADJ" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="IMG EXT EDITED" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="IMG RETOUCHED" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="IMG COMPOSITED" minOccurs="0">
          <xsd:complexType/>
        </xsd:element>
        <xsd:element name="IMG METADATA" minOccurs="0">
          <xsd:complexType/>
```



```
        </xsd:element>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IMAGE PROCESSING HINTS">
    <xsd:annotation>
      <xsd:documentation>
                See section D.3.2  Image Processing Hints
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence minOccurs="0" maxOccurs="unbounded">
        <xsd:element name="MODIFIER" type="dig35:tProductDetails" minOccurs="0"/>
        <xsd:choice minOccurs="0" maxOccurs="unbounded">
          <xsd:element name="IMG CREATED" type="dig35:tLangString"/>
          <xsd:element name="IMG CROPPED" type="dig35:tLangString"/>
          <xsd:element name="IMG TRANSFORMED" type="dig35:tLangString"/>
          <xsd:element name="IMG GTC ADJ" type="dig35:tLangString"/>
          <xsd:element name="IMG STC ADJ" type="dig35:tLangString"/>
          <xsd:element name="IMG SPATIAL ADJ" type="dig35:tLangString"/>
          <xsd:element name="IMG EXT EDITED" type="dig35:tLangString"/>
          <xsd:element name="IMG RETOUCHED" type="dig35:tLangString"/>
          <xsd:element name="IMG COMPOSITED" type="dig35:tLangString"/>
          <xsd:element name="IMG METADATA" type="dig35:tLangString"/>
        </xsd:choice>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>


  <xsd:annotation>
    <xsd:documentation> Annex E:  Intellectual Property Rights Metadata
</xsd:documentation>
  </xsd:annotation>


  <xsd:element name="IPR">
    <xsd:annotation>
      <xsd:documentation>
                See section Annex E:  Intellectual Property Rights Metadata
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:IPR NAMES" minOccurs="0"/>
        <xsd:element ref="dig35:IPR DESCRIPTION" minOccurs="0"/>
        <xsd:element ref="dig35:IPR DATES" minOccurs="0"/>
        <xsd:element ref="dig35:IPR EXPLOITATION" minOccurs="0"/>
        <xsd:element ref="dig35:IPR IDENTIFICATION" minOccurs="0"/>
        <xsd:element ref="dig35:IPR CONTACT POINT" minOccurs="0"/>
        <xsd:element ref="dig35:IPR HISTORY" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IPR NAMES">
    <xsd:annotation>
      <xsd:documentation>
```

```
                See section E.3.1  Names
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:choice maxOccurs="unbounded">
        <xsd:element ref="dig35:IPR PERSON"/>
        <xsd:element ref="dig35:IPR ORG"/>
        <xsd:element ref="dig35:IPR NAME REF"/>
      </xsd:choice>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IPR PERSON">
    <xsd:annotation>
      <xsd:documentation>
                See section E.3.1  Names
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:complexContent>
        <xsd:extension base="dig35:tPerson">
          <xsd:attribute name="DESCRIPTION" type="xsd:string"/>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IPR ORG">
    <xsd:annotation>
      <xsd:documentation>
                See section E.3.1  Names
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:complexContent>
        <xsd:extension base="dig35:tOrganization">
          <xsd:attribute name="DESCRIPTION" type="xsd:string"/>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IPR NAME REF">
    <xsd:annotation>
      <xsd:documentation>
                See section E.3.1  Names
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:simpleContent>
        <xsd:extension base="xsd:string">
          <xsd:attribute name="DESCRIPTION" type="xsd:string"/>
        </xsd:extension>
      </xsd:simpleContent>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IPR DESCRIPTION">
    <xsd:annotation>
      <xsd:documentation>
                See section E.3.2  Description
      </xsd:documentation>
```

```
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="IPR TITLE" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="IPR LEGEND" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="IPR CAPTION" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="COPYRIGHT" type="dig35:tLangString" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IPR DATES">
    <xsd:annotation>
      <xsd:documentation>
                See section E.3.3  Dates
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:IPR DATE" maxOccurs="unbounded"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>
  <xsd:element name="IPR DATE">
    <xsd:annotation>
      <xsd:documentation>
            See section E.3.3  Dates
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:complexContent>
        <xsd:extension base="dig35:tDateTime">
          <xsd:attribute name="DESCRIPTION" type="xsd:string"/>
          <xsd:attribute ref="dig35:TIMESTAMP"/>
          <xsd:attribute ref="xml:lang"/>
        </xsd:extension>
      </xsd:complexContent>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IPR EXPLOITATION">
    <xsd:annotation>
      <xsd:documentation>
            See section E.3.4  Exploitation
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="IPR PROTECTION" type="xsd:nonNegativeInteger" minOccurs="0"/>
        <xsd:element name="IPR USE RESTRICTION" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="IPR OBLIGATION" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element ref="dig35:IPR MGMT SYS" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IPR MGMT SYS">
    <xsd:annotation>
      <xsd:documentation>
            See section E.3.4.4  IPR Management System
```



```
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="IPR MGMT TYPE" type="xsd:string" minOccurs="0"/>
        <xsd:element name="IPR MGMT SYS ID" type="xsd:string" minOccurs="0"/>
        <xsd:element name="IPR MGMT SYS LOCATION" type="xsd:anyURI" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IPR IDENTIFICATION">
    <xsd:annotation>
      <xsd:documentation>
            See section E.3.5  Identification
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:IPR IDENTIFIER" minOccurs="0"/>
        <xsd:element ref="dig35:LICENCE PLATE" minOccurs="0"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IPR IDENTIFIER">
    <xsd:annotation>
      <xsd:documentation>
            See section E.3.5.1  Generic IPR Identifier
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="IPR ID MODE" type="dig35:tLangString" minOccurs="0"/>
        <xsd:element name="IPR ID" type="dig35:tLangString" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="LICENCE PLATE">
    <xsd:annotation>
      <xsd:documentation>
            See section E.3.5.2  License Plate
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element name="LP COUNTRY" type="xsd:string" minOccurs="0"/>
        <xsd:element name="LP REG AUT" type="xsd:string" minOccurs="0"/>
        <xsd:element name="LP REG NUM" type="xsd:string" minOccurs="0"/>
        <xsd:element name="LP DELIVERY DATE" type="xsd:dateTime" minOccurs="0"/>
      </xsd:sequence>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IPR CONTACT POINT">
    <xsd:annotation>
      <xsd:documentation>
            See section E.3.6  Contact Point
      </xsd:documentation>
```

```
    </xsd:annotation>
    <xsd:complexType>
      <xsd:choice>
        <xsd:element ref="dig35:IPR PERSON"/>
        <xsd:element ref="dig35:IPR ORG"/>
        <xsd:element ref="dig35:IPR NAME REF"/>
      </xsd:choice>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

  <xsd:element name="IPR HISTORY">
    <xsd:annotation>
      <xsd:documentation>
            See section E.3.7  IPR History
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:element ref="dig35:IPR" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang"/>
    </xsd:complexType>
  </xsd:element>

</xsd:schema>
```


N.2 DIG35 Basic Types and Fields

```
Schema Definition:

<xsd:simpleType name="tNonNegativeDouble">
  <xsd:restriction base="xsd:double">
    <xsd:minInclusive value="0"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="tRational">
  <xsd:restriction base="xsd:string">
    <xsd:pattern value="(\-|\+)?[0-9]+/[0-9]+"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="tLangString">
  <xsd:simpleContent>
    <xsd:extension base="xsd:string">
      <xsd:attribute ref="xml:lang"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>

<xsd:simpleType name="tDegree">
  <xsd:restriction base="xsd:double">
    <xsd:minExclusive value="-180"/>
    <xsd:maxInclusive value="180"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:simpleType name="tHalfDegree">
  <xsd:restriction base="xsd:double">
    <xsd:minExclusive value="-90"/>
    <xsd:maxInclusive value="90"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="tDoubleSize">
  <xsd:sequence>
    <xsd:element name="WIDTH"         type="dig35:tNonNegativeDouble"/>
    <xsd:element name="HEIGHT"        type="dig35:tNonNegativeDouble"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="tIntSize">
  <xsd:sequence>
    <xsd:element name="WIDTH"         type="xsd:positiveInteger"/>
    <xsd:element name="HEIGHT"        type="xsd:positiveInteger"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="tDateTime">
  <xsd:sequence>
    <xsd:choice
minOccurs="0">
      <xsd:element name="EXACT"       type="xsd:dateTime"/>
      <xsd:element name="DATE"        type="xsd:date"/>
      <xsd:sequence>
        <xsd:element name="MONTH"     type="dig35:tRecurringMonth" minOccurs="0"/>
        <xsd:element name="YEAR"      type="xsd:gYear"               minOccurs="0"/>
```

```
        <xsd:element name="CENTURY"   type="dig35:tCentury"          minOccurs="0"/>
    </xsd:sequence>
    </xsd:choice>

    <xsd:element name="WEEK DAY"      type="dig35:tLangString"       minOccurs="0"/>
    <xsd:element name="SEASON"        type="dig35:tLangString"       minOccurs="0"/>

    <xsd:element ref="dig35:COMMENT"                                 minOccurs="0"/>
  </xsd:sequence>

  <xsd:attribute ref="dig35:TIMESTAMP"/>
  <xsd:attribute ref="xml:lang"/>
</xsd:complexType>

<xsd:simpleType name="tCentury">
  <xsd:restriction base="xsd:integer"/>
</xsd:simpleType>

<xsd:simpleType name="tRecurringMonth">
  <xsd:restriction base="xsd:positiveInteger">
    <xsd:minInclusive value="1"/>
    <xsd:maxInclusive value="12"/>
  </xsd:restriction>
</xsd:simpleType>

<xsd:complexType name="tAddress">
  <xsd:sequence>
    <xsd:element name="ADDR NAME"     type="dig35:tLangString"   minOccurs="0"/>
    <xsd:element ref="ADDR COMP"               minOccurs="0"     maxOccurs="unbounded"/>
    <xsd:choice                                                          minOccurs="0">
      <xsd:element name="ZIPCODE"     type="xsd:string"/>
      <xsd:element name="POSTCODE"    type="xsd:string"/>
    </xsd:choice>
    <xsd:element name="COUNTRY"       type="dig35:tLangString"   minOccurs="0"/>
  </xsd:sequence>

  <xsd:attribute name="TYPE"          type="xsd:string"/>
  <xsd:attribute ref="dig35:TIMESTAMP"/>
  <xsd:attribute ref="xml:lang"/>
</xsd:complexType>

<xsd:element name="ADDR COMP">
  <xsd:complexType>
    <xsd:simpleContent>
      <xsd:extension base="dig35:tLangString">
        <xsd:attribute name="TYPE"    type="xsd:string"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>
</xsd:element>

<xsd:complexType name="tPhone">
  <xsd:sequence>
    <xsd:element name="COUNTRY CODE"  type="xsd:string"          minOccurs="0"/>
    <xsd:element name="AREA"          type="xsd:string"          minOccurs="0"/>
    <xsd:element name="LOCAL"         type="xsd:string"          minOccurs="0"/>
    <xsd:element name="EXTENSION"     type="xsd:string"          minOccurs="0"/>
  </xsd:sequence>

  <xsd:attribute name="TYPE"          type="xsd:string"/>
  <xsd:attribute ref="dig35:TIMESTAMP"/>
</xsd:complexType>

<xsd:complexType name="tEmail">
```



```
  <xsd:simpleContent>
    <xsd:extension base="dig35:tLangString">
      <xsd:attribute name="TYPE"      type="xsd:string"/>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="tWeb">
  <xsd:simpleContent>
    <xsd:extension base="dig35:tLangString">
      <xsd:attribute name="TYPE"      type="xsd:string"/>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
    </xsd:extension>
  </xsd:simpleContent>
</xsd:complexType>

<xsd:complexType name="tPerson">
  <xsd:sequence>
    <xsd:element name="NAME TITLE"    type="dig35:tLangString"   minOccurs="0"/>
    <xsd:element ref="PERSON NAME"                               minOccurs="0"
            maxOccurs="unbounded"/>
    <xsd:element name="NICKNAME"      type="dig35:tLangString"   minOccurs="0"
            maxOccurs="unbounded"/>
    <xsd:element name="JOB TITLE"     type="dig35:tLangString"   minOccurs="0"/>
    <xsd:choice
            minOccurs="0">
      <xsd:element name="PERSON ORG"  type="dig35:tOrganization"/>
      <xsd:element name="ORG REF"     type="xsd:string"/>
    </xsd:choice>
    <xsd:element name="ADDRESS"       type="dig35:tAddress"      minOccurs="0"
            maxOccurs="unbounded"/>
    <xsd:element name="PHONE"         type="dig35:tPhone"        minOccurs="0"
            maxOccurs="unbounded"/>
    <xsd:element name="EMAIL"         type="dig35:tEmail"        minOccurs="0"
            maxOccurs="unbounded"/>
    <xsd:element name="WEB"                    type="dig35:tWeb" minOccurs="0"
            maxOccurs="unbounded"/>
    <xsd:element name="BIRTH DATE"    type="xsd:date"            minOccurs="0"/>
    <xsd:element name="AGE"                    type="xsd:duration"       minOccurs="0"/>
    <xsd:element ref="dig35:COMMENT"                             minOccurs="0"/>
  </xsd:sequence>

  <xsd:attribute name="ID"                     type="xsd:string"/>
  <xsd:attribute ref="dig35:TIMESTAMP"/>
  <xsd:attribute ref="xml:lang"/>
</xsd:complexType>

<xsd:element name="PERSON NAME">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element ref="NAME COMP"
            maxOccurs="unbounded"/>
    </xsd:sequence>

    <xsd:attribute ref="dig35:TIMESTAMP"/>
    <xsd:attribute ref="xml:lang"/>
  </xsd:complexType>
</xsd:element>

<xsd:element name="NAME COMP">
  <xsd:complexType>
    <xsd:simpleContent>
      <xsd:extension base="xsd:string">
```

```
        <xsd:attribute name="TYPE" use="optional" default="Given">
          <xsd:simpleType>
            <xsd:restriction base="xsd:string">
              <xsd:enumeration value="Prefix"/>
              <xsd:enumeration value="Given"/>
              <xsd:enumeration value="Family"/>
              <xsd:enumeration value="Suffix"/>
              <xsd:enumeration value="Maiden"/>
            </xsd:restriction>
          </xsd:simpleType>
        </xsd:attribute>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>
</xsd:element>

<xsd:complexType name="tOrganization">
  <xsd:sequence>
    <xsd:element name="ORG NAME"      type="dig35:tLangString"   minOccurs="0"/>
    <xsd:element name="ADDRESS"       type="dig35:tAddress"      minOccurs="0"
            maxOccurs="unbounded"/>
    <xsd:element name="PHONE"         type="dig35:tPhone"        minOccurs="0"
            maxOccurs="unbounded"/>
    <xsd:element name="EMAIL"         type="dig35:tEmail"        minOccurs="0"
            maxOccurs="unbounded"/>
    <xsd:element name="WEB"                    type="dig35:tWeb" minOccurs="0"
            maxOccurs="unbounded"/>
    <xsd:element name="LOGO FILE"     type="xsd:anyURI"          minOccurs="0"/>
    <xsd:element name="LOGO FORMAT"   type="xsd:string"          minOccurs="0"/>
    <xsd:element name="MIME TYPE"     type="xsd:string"          minOccurs="0"/>
    <xsd:element ref="dig35:COMMENT"                             minOccurs="0"/>
  </xsd:sequence>

  <xsd:attribute name="ID"                     type="xsd:string"/>
  <xsd:attribute ref="dig35:TIMESTAMP"/>
  <xsd:attribute ref="xml:lang"/>
</xsd:complexType>

<xsd:complexType name="tLocation">
  <xsd:sequence>
    <xsd:element ref="dig35:COORD LOC"                                 minOccurs="0"  />
    <xsd:element name="ADDRESS"                type="dig35:tAddress"     minOccurs="0"/>
    <xsd:element ref="dig35:GPS"                                         minOccurs="0"/>
    <xsd:element ref="dig35:COMMENT"                                     minOccurs="0"/>
  </xsd:sequence>

  <xsd:attribute ref="dig35:TIMESTAMP"/>
  <xsd:attribute ref="xml:lang"/>
</xsd:complexType>

<xsd:element name="COORD LOC">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="LONGITUDE"   type="dig35:tDegree"       minOccurs="0"/>
      <xsd:element name="LATITUDE"    type="dig35:tHalfDegree"   minOccurs="0"/>
      <xsd:element name="ALTITUDE"    type="xsd:double"          minOccurs="0"/>
    <xsd:sequence>

    <xsd:attribute ref="dig35:TIMESTAMP"/>
  </xsd:complexType>
</xsd:element>

<xsd:element name="GPS">
  <xsd:complexType>
```

```
    <xsd:sequence>
      <xsd:element name="GPS LAT REF"                          minOccurs="0">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="N"/>
            <xsd:enumeration value="S"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="GPS LATITUDE"         type="tDms"               minOccurs="0"/>

      <xsd:element name="GPS LONG REF"                                   minOccurs="0">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="E"/>
            <xsd:enumeration value="W"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="GPS LONGITUDE"        type="tDms"               minOccurs="0"/>

      <xsd:element name="GPS ALTITUDE"         type="dig35:tNonNegativeDouble"
            minOccurs="0"/>

      <xsd:element name="GPS TIME"    type="xsd:dateTime"                minOccurs="0"/>

      <xsd:element name="GPS SATELLITES"       type="xsd:string"         minOccurs="0"/>
      <xsd:element name="GPS STATUS"                           minOccurs="0">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="A"/>
            <xsd:enumeration value="V"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="GPS MEASURE MODE"                     minOccurs="0">
        <xsd:simpleType>
          <xsd:restriction base="xsd:positiveInteger">
            <xsd:minExclusive value="2"/>
            <xsd:maxInclusive value="3"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="GPS DOP"     type="dig35:tNonNegativeDouble"
            minOccurs="0"/>

      <xsd:element name="GPS SPEED REF"                                  minOccurs="0">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="K"/>
            <xsd:enumeration value="N"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="GPS SPEED"   type="dig35:tNonNegativeDouble"
            minOccurs="0"/>

      <xsd:element name="GPS TRACK REF"                                  minOccurs="0">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="T"/>
            <xsd:enumeration value="M"/>
          </xsd:restriction>
        </xsd:simpleType>
```

```
      </xsd:element>
      <xsd:element name="GPS TRACK"   type="dig35:tNonNegativeDouble"
            minOccurs="0"/>

      <xsd:element name="GPS IMAGE DIR REF"                      minOccurs="0">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="T"/>
            <xsd:enumeration value="M"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="GPS IMAGE DIR"        type="dig35:tNonNegativeDouble"
            minOccurs="0"/>

      <xsd:element name="GPS MAP DATUM"        type="xsd:string"         minOccurs="0"/>

      <xsd:element name="GPS DEST LAT REF"                       minOccurs="0"  >
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="N"/>
            <xsd:enumeration value="S"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="GPS DEST LATITUDE" type="tDms"                 minOccurs="0"/>

      <xsd:element name="GPS DEST LONG REF"                      minOccurs="0">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="E"/>
            <xsd:enumeration value="W"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="GPS DEST LONGITUDE" type="tDms"                minOccurs="0"/>

      <xsd:element name="GPS DEST BEARING REF"                   minOccurs="0">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="T"/>
            <xsd:enumeration value="M"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="GPS DEST BEARING"  type="dig35:tNonNegativeDouble"
            minOccurs="0"/>

      <xsd:element name="GPS DEST DISTANCE REF"                         minOccurs="0">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="K"/>
            <xsd:enumeration value="N"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:element>
      <xsd:element name="GPS DEST DISTANCE"  type="dig35:tNonNegativeDouble"
            minOccurs="0"/>
    </xsd:sequence>

  </xsd:complexType>
</xsd:element>

<xsd:complexType name="tDms">
```

```
  <xsd:sequence>
    <xsd:element name="D"                     type="xsd:nonNegativeInteger"/>
    <xsd:element name="M"                     type="xsd:nonNegativeInteger"/>
    <xsd:element name="S"                     type="dig35:tNonNegativeDouble"
            minOccurs="0"/>
  </xsd:sequence>
</xsd:complexType>

<xsd:complexType name="tDirection">
  <xsd:sequence>
    <xsd:element name="YAW"                   type="dig35:tDegree"      minOccurs="0"/>
    <xsd:element name="PITCH"         type="dig35:tHalfDegree"   minOccurs="0"/>
    <xsd:element name="ROLL"          type="dig35:tDegree"       minOccurs="0"/>
    <xsd:element ref="dig35:COMMENT"                             minOccurs="0"/>
  </xsd:sequence>

  <xsd:attribute ref="dig35:TIMESTAMP"/>
  <xsd:attribute ref="xml:lang"/>
</xsd:complexType>

<xsd:complexType name="tPosition">
  <xsd:sequence>
    <xsd:choice
            minOccurs="0">
      <xsd:element name="POINT"       type="dig35:tPoint"/>
      <xsd:element name="RECT"        type="dig35:tRect"/>
      <xsd:sequence>
        <xsd:element name="RECT"      type="dig35:tRect"/>
        <xsd:element name="REGION"    type="dig35:tRegion"/>
      </xsd:sequence>
    </xsd:choice>
    <xsd:element ref="dig35:COMMENT"                             minOccurs="0"/>
  </xsd:sequence>

  <xsd:attribute ref="dig35:TIMESTAMP"/>
</xsd:complexType>
<xsd:complexType name="tPoint">
  <xsd:sequence>
    <xsd:element name="X"                     type="dig35:tNonNegativeDouble"/>
    <xsd:element name="Y"                     type="dig35:tNonNegativeDouble"/>
  </xsd:sequence>
</xsd:complexType>
<xsd:complexType name="tRegion">
  <xsd:sequence>
    <xsd:element name="POINT"         type="dig35:tPoint"/>
    <xsd:choice
            minOccurs="0"             maxOccurs="unbounded">
      <xsd:element name="POINT"       type="dig35:tPoint"/>
      <xsd:element ref="SPLINE"/>
    </xsd:choice>
  </xsd:sequence>
</xsd:complexType>

<xsd:element name="SPLINE">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element name="X1"          type="dig35:tNonNegativeDouble"/>
      <xsd:element name="Y1"          type="dig35:tNonNegativeDouble"/>
      <xsd:element name="X2"          type="dig35:tNonNegativeDouble"/>
      <xsd:element name="Y2"          type="dig35:tNonNegativeDouble"/>
      <xsd:element name="X"                   type="dig35:tNonNegativeDouble"/>
      <xsd:element name="Y"                   type="dig35:tNonNegativeDouble"/>
    </xsd:sequence>
  </xsd:complexType>
```

```
</xsd:element>

<xsd:complexType name="tProductDetails">
  <xsd:sequence>
    <xsd:element name="MANUFACTURER"  type="dig35:tOrganization" minOccurs="0"  />
    <xsd:element name="MODEL"         type="xsd:string"                 minOccurs="0"  />
    <xsd:element name="SERIAL"        type="xsd:string"                 minOccurs="0"  />
    <xsd:element name="VERSION"       type="xsd:string"                 minOccurs="0"  />
  </xsd:sequence>

  <xsd:attribute ref="dig35:TIMESTAMP"/>
  <xsd:attribute ref="xml:lang"/>
</xsd:complexType>

<xsd:attribute name="lang" type="xsd:language"/>

<!-- To be used in XML Schema documents, W3C's definition must be imported.  -->
<xsd:import namespace=http://www.w3.org/XML/1998/namespace
            schemaLocation="http://www.w3.org/2001/xml.xsd"/>

<xsd:attribute name="TIMESTAMP" type="xsd:dateTime"/>

<xsd:element name="COMMENT">
  <xsd:complexType>
    <xsd:simpleContent>
      <xsd:extension base="dig35:tLangString">
        <xsd:attribute ref="dig35:TIMESTAMP"/>
      </xsd:extension>
    </xsd:simpleContent>
  </xsd:complexType>
</xsd:element>
```


Annex O: What Is XML?

[MUCH TODO]

O.1 What is XML?

Extensible Markup Language (XML) is a subset of SGML (ISO standard). Its goal is to enable generic SGML to be processed on the Web in the same way that is now possible with HTML. XML has been designed for ease of implementation compared to SGML.

XML defines document structure and embeds it directly within the document through the use of markups. A markup is composed of two kinds of tags which encapsulate data : open tags and closed tags. XML is similar to HTML but the tags can be defined by the user. The definition of valid document structure is expressed in a language called the DTD (Document Type Definition) or a Schema Language (e.g., XML Schema, RDF Schema).

The XML 1.0 specification defines the concepts of well formedness and validity. XML well-formedness requires that document tags are correctly nested. XML validity requires that a document follow the constraints expressed in its associated document type definition (DTD) or schema.

Annex P: Typographic Conventions

Schema are in yellow boxes in Courier font.

```
Plain text is DIG35 schema.
Blue bold text is an MPV schema addition.
~~Red bold struckthrough text is DIG35 schema deprecated (but still supported) by MPV.~~
```

For example:

```
Metadata schema descriptions

  <xsd:element name="METADATA">
    <xsd:annotation>
      <xsd:documentation>
          See section Annex G: DIG35 XML Document Definition
      </xsd:documentation>
    </xsd:annotation>
    <xsd:complexType>
      <xsd:sequence>
        <xsd:choice>
           <xsd:element ref="mpv:VIDEO" minOccurs="0"/>
           <xsd:element ref="mpv:AUDIO" minOccurs="0"/>
           <xsd:element ref="mpv:IMAGE" minOccurs="0"/>
           <xsd:element ref="mpv:DOCUMENT" minOccurs="0"/>
           <xsd:sequence>
             <!-- equivalent to using the IMAGE element -->
             <xsd:element ref="dig35:BASIC_IMAGE_PARAM" minOccurs="0"/>
             <xsd:element ref="dig35:IMAGE_CREATION" minOccurs="0"/>
           </xsd:sequence>
        </xsd:choice>
        <xsd:element ref="dig35:CONTENT DESCRIPTION" minOccurs="0"/>
        <xsd:element ref="dig35:HISTORY" minOccurs="0"/>
        <xsd:element ref="dig35:IPR" minOccurs="0"/>
        <xsd:element ref="dig35:METADATA" minOccurs="0" maxOccurs="unbounded"/>
      </xsd:sequence>
      <xsd:attribute name="TYPE" use="optional" default="Single">
        <xsd:simpleType>
          <xsd:restriction base="xsd:string">
            <xsd:enumeration value="Single"/>
            <xsd:enumeration value="Collection"/>
          </xsd:restriction>
        </xsd:simpleType>
      </xsd:attribute>
      <xsd:attribute ref="dig35:TIMESTAMP"/>
      <xsd:attribute ref="xml:lang" use="optional" default="en"/>
    </xsd:complexType>
  </xsd:element>
```

Examples of DIG35 and MPV metadata structures are in blue boxes in Courier font.

```
Example.
```



```
<METADATA TYPE="Single">
   <BASIC IMAGE PARAM>
      ...
   </BASIC IMAGE PARAM>
</METADATA>
```


Annex Q: References

[DCF-1999] "Design rule for Camera File system, Version 1.0", JEIDA standard, English Version 1999.1.7, Japanese Electronic Industry Development Association (JEIDA).

[DIG35-2001] "DIG35 Specification – Metadata for Digital Images, Version 1.1 Working Draft", April 16, 2001, International Imaging Industry Association (I3A) [recently formed by combining the Digital Imaging Group and PIMA]. http://www.i3a.org

[C14N-2001] "Canonical XML Version 1.0", March 15th, 2001, W3C, http://www.w3.org/TR/xml-c14n