An Ontology Design Pattern for Digital Video

Panagiotis Mitzias[1], Marina Riga[1], Simon Waddington[2], Efstratios Kontopoulos[1], Georgios Meditskos[1], Pip Laurenson[3], Ioannis Kompatsiaris[1]

[1] Information Technologies Institute, CERTH, GR-57001 Thessaloniki, Greece
{ pmitzias, mriga, skontopo, gmeditsk, ikom}@iti.gr
[2] King's College London, UK
simon.waddington@kcl.ac.uk
[3] Tate, London, UK
Pip.Laurenson@tate.org.uk

Abstract. In this paper we introduce an ODP for representing digital video resources. The aim is to model digital video files, their components and other associated entities, such as codecs and containers. The proposed design pattern facilitates the creation of relevant domain ontologies that will be deployed in the fields of media archiving and digital preservation of videos and video artworks. This ODP has been developed within the PERICLES FP7 project.

Keywords: Digital Video, Codec, Stream, ODP.

1 Introduction

This paper presents an Ontology Design Pattern (ODP) for modelling digital video resources. This work was motivated by the problem of consistent presentation of digital video files in the context of digital preservation within the PERICLES FP7 project[1]. Over the past five years, this challenge has emerged as a significant one within the conservation of video art [6] and was taken as a focus within Presto4U[2]. As a result of this initial work, Dave Rice was commissioned to produce a technical report [9] and it is this report which underpins the analysis of this challenge presented in this paper. Although those who are responsible for the conservation of video art have been particularly concerned with ensuring consistent playback, the problem is pertinent to any application domain requiring video playback. Presenting digital video consistently is dependent on the design, coordination and quality of all aspects of both the video file and the video player [9]. In particular, the ongoing development of media players can impact the capability to view video files as they were originally intended.

Playback of compressed video is reliant on a correct interpretation of the parameters associated to the file, with colour and aspect ratio being two of the most vulnerable properties. We focus our efforts here on the relationship between the video file

[1] www.pericles-project.eu/
[2] Presto4U FP7 project (ICT Call 9): www.tate.org.uk/about/projects/presto4u

itself, the codec used to compress the video and the wrapper. By wrapper, we mean a multimedia container format, which can identify and interleave different data types, including video and audio streams, subtitles, as well as synchronisation metadata to enable the streams to be played concurrently. A particular source of conflict is that the video file and the wrapper can potentially contain values for the same parameter, which can lead to inconsistency of playback. For example, aspect ratio information can be carried in both the video file and the wrapper, and is often handled differently by different players [9].

Many standards and specifications for video and multimedia containers exist, with similar definitions of key parameters. When considering the playback of video and audio using players supporting multiple video formats and multiple versions of those formats, there is a clear need for a consistent set of definitions of key parameters across different formats. To the best of our knowledge, `ontologydesignpatterns.org` currently features no such ODP for describing video. A literature review reveals several relevant ontologies and vocabularies that deal with the modelling of multimedia objects and their processes. The well-established multimedia standard *MPEG-7* [10], as well as several MPEG-7 based ontologies (*Hunter* [5], *Rhizomik* [3], *COMM* [1], *SWIntO* [7], *Boemie* [2], *DS-MIRF* [8], *M-OWL* [4]) can be used for creating metadata descriptions of multimedia content corresponding to low-level visual and audio features, or semantic objects (e.g. places, actors, events, objects). Furthermore, *OMR*[3] is a core vocabulary aimed at bridging the different descriptions of media resources and at providing an interoperable set of metadata. OMR includes, among others, technical metadata about media objects; nevertheless the represented properties do not cover the domain in sufficient detail. Similarly, the *audioMD* and *videoMD* schemas[4] define significant technical audio and video metadata, but do not contain all the partial components which constitute a digital video.

It is evident that the aforementioned ontologies do not focus specifically on the representation of digital videos but on various multimedia resources as a whole, by modelling information regarding the creator, the conceptual aspect (idea, content) behind the digital resource, its legal/intellectual properties, etc. Our proposed ODP deals with the structural and technical representation of digital videos in detail, carrying significant information for modelling characteristics and interrelationships (dependencies) that impact the ability to preserve a digital video over time.

2 Pattern Description and Formalization

This section presents the proposed ODP, focusing on the core classes, properties, and axioms. Fig. 1 features a diagrammatic overview of the pattern, which is available at: `http://ontologydesignpatterns.org/wiki/Submissions:DigitalVideo`. Although the ODP contains 31 classes and 27 object properties, in terms of expressivity it is deliberately lightweight, containing only subclass and subproperty axioms,

[3] www.w3.org/TR/mediaont-10/
[4] www.loc.gov/standards/amdvmd/index.html

property restrictions and class disjointness axioms, in order to be easily applicable to a wide range of use cases and scenarios.

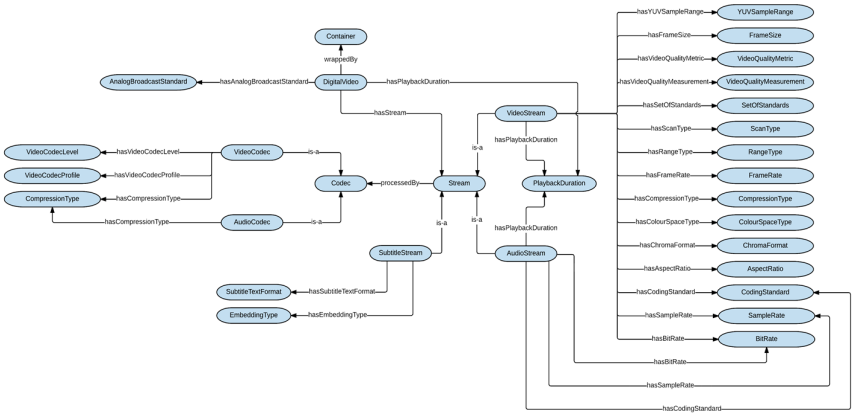

Fig. 1. ODP schematic view.

Our starting assumption for the design pattern is that the video entity itself comprises a video stream alongside optional associated audio and subtitle streams. The pattern also covers the case of having multiple video and audio streams. The following is a list of the core classes found in the proposed pattern:

DigitalVideo. The `DigitalVideo` class represents a single digital video file. Such a file typically consists of one or more streams, which are compressed using codecs and wrapped into a specific type of container.

Container. A `Container` (or wrapper) is typically 1-to-1 associated with the video file format. It acts as a discrete "black box" that contains the various components of a video and defines how different elements of data and metadata coexist in the video file. Sample container formats are AVI, Matroska, MP4, etc.

Codec. A `Codec` (coder-decoder) is a computer software capable of encoding or decoding a digital data stream or signal[5]. Video codecs convert raw video streams to a compressed format and vice-versa, while audio codecs process audio streams. Some well-known codecs are x264, DivX Pro and mp3HD.

Stream. A (data) stream is a sequence of digitally encoded coherent signals (packets of data or data packets) used to transmit or receive information[6]. Class `Stream` represents raw, uncompressed content (video, audio or subtitles) prior to being encoded into a wrapper or after being decoded from a wrapper. A digital video file includes at least one video stream and may also have any number and any kind of other streams:

$$\texttt{DigitalVideo} \sqsubseteq \exists \texttt{hasVideoStream}.\texttt{VideoStream} \quad (1)$$

[5] https://en.wikipedia.org/wiki/Codec
[6] https://en.wikipedia.org/wiki/Data_stream

$$\texttt{DigitalVideo} \sqsubseteq \forall \texttt{hasAudioStream.AudioStream} \quad (2)$$

$$\texttt{DigitalVideo} \sqsubseteq \forall \texttt{hasSubtitleStream.SubtitleStream} \quad (3)$$

Each type of stream (`VideoStream`, `AudioStream` and `SubtitleStream`) is associated with disparate types of properties and elements (see Fig. 1), though some of them apply to both video and audio streams, such as `BitRate` and `SampleRate`.

All in all, the proposed ODP is deliberately generic and can be extended with appropriate restrictions, depending on specific application requirements. Also, for reasons of flexibility, no data properties were included for numerical or text attributes.

3 Conclusions

This paper presented an ODP for representing digital video resources that can serve as the building block for domain-specific ontologies. Its main focus is on digital preservation and has been successfully deployed within the PERICLES FP7 project.

4 Acknowledgements

This work was supported by the European Commission Seventh Framework Programme under Grant Agreement Number FP7-601138 PERICLES.

References

1. Arndt, R. et al. (2007). *COMM: designing a well-founded multimedia ontology for the web*. LNCS 4825, pp. 30-43, Springer Berlin Heidelberg.
2. Dasiopoulou, S. et al. (2007). Capturing MPEG-7 Semantics. *2nd Int. Conf. on Metadata and Semantics (MTSR)*, Corfu, Greece.
3. Garcia, R. & Celma, O. (2005). Semantic Integration and Retrieval of Multimedia Metadata. *Int. Semantic Web Conference (ISWC'05)*, Galway, Ireland.
4. Harit, G. et al. (2006). Using Multimedia Ontology for Generating Conceptual Annotations and Hyperlinks in Video Collections. *IEEE/WIC/ACM Int. Conf. on Web Intelligence*, pp. 211-217, IEEE Computer Society.
5. Hunter, J. (2001). Adding Multimedia to the Semantic Web: Building an MPEG-7 Ontology. *SWWS'01*, Stanford University, California, USA.
6. Jarczyk, A & Phillips, J. (2014). Life after tape: Collecting Digital Video Art. Electronic Media Group. *42nd American Institute for Conservation Annual Meeting* (*to appear*).
7. Oberle, D. et al. (2007). DOLCE ergo SUMO: On Foundational and Domain Models in SWIntO (Smart Web Integrated Ontology). *Journal of Web Semantics*, 5(3), pp.156-174.
8. Polydoros, P. et al. (2006). GraphOnto: OWL-based ontology management and multimedia annotation in the DS-MIRF framework. *J. of Digital Inf. Management*, *4*(4), 214.
9. Rice, D. (2015). Sustaining Consistent Video Presentation. Available online: http://goo.gl/CHDfKB, last accessed: June'15.
10. Sikora, T. (2001). The MPEG-7 visual standard for content description – an overview. *IEEE Trans. on Circuits and Systems for Video Technology*, 11(6), pp. 696-702, IEEE.