IMPULSE

IMmersive digitisation: uPcycling cULtural heritage towards new reviving StratEgies

Deliverable D19:

Analysis of 2D, 3D, and Audiovisual File Formats for Virtual Environment Applications in Cultural Heritage Digitisation

Funded by the European Union

Funded by the European Union. Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or the European Education and Culture Executive Agency (EACEA). Neither the European Union nor EACEA can be held responsible for them.

1 Document Information

Document Identification			
Status	Final	**Due Date**	15 June 2025
Version	2.0	**Submission Date**	13 June 2025
Related WP	WP3	**Document Reference**	D19
Related Task(s)	T.3.2.2	**Document Type**	Report
Related Deliverable(s)	D2.1 (11), D2.2 (12) D.3.1 (17)	**Dissemination Level**	Public
Lead Participant	KU Leuven	**Lead Author**	Bruno Vandermeulen (KUL)
Contributors	Theodora Rontzova (KUL)	**Reviewers**	Jan Tretschok (K8)
		Reviewers	Charalampos Rizopoulos (NKUA)

Author(s)		
First Name	**Last Name**	**Partner**
Bruno	Vandermeulen	KU Leuven
Theodora	Rontzova	KU Leuven

Document History			
Version	**Date**	**Modified by**	**Modification reason**
0.1	08/05/2025	Bruno Vandermeulen and Theodora Rontzova	First draft
0.2	25/05/2025	Bruno Vandermeulen	Second draft
0.3	30/05/2025	Bruno Vandermeulen	Third draft
0.4	06/06/2025	Theodora Rontzova	Internal review
1.0	10/06/2025	Jan Tretschok and Charalompos Rizopoulos	Evaluation by expert reviewers
1.1	10/06/2025	Bruno Vandermeulen	Evaluation team's comments implementation
2.0	11-13/06/2025	Żaneta Żegleń, Łukasz Pieczonka	Quality Control

Quality Control		
Role	**Who (Partner short name)**	**Approval Date**
Deliverable leader	Bruno Vandermeulen (KUL)	10/06/2025
Quality manager	Łukasz Pieczonka (JU)	11/06/2025
Project Coordinator	Żaneta Żegleń (JU)	12-13/06/2025

2 Executive Summary

This report, "Analysis of 2D, 3D, and Audiovisual File Formats for Virtual Environment Applications in Cultural Heritage Digitisation," addresses a critical challenge in the digital cultural heritage sector: the disparity between file formats optimized for archival purposes and those suitable for real-time, interactive virtual environments (VE). Heritage institutions typically prioritize high-fidelity, preservation-focused formats like TIFF, RAW, PLY, and WAV to ensure long-term integrity. While essential for archives, these formats are often large and complex, rendering them inefficient for direct use in VEs that demand rapid loading and efficient GPU rendering.

The document provides a comprehensive technical analysis of a wide array of 2D, 3D, and audiovisual file formats. For each format, the evaluation considers its usability, engine integration, interoperability, metadata capabilities, and overall suitability for VE applications, aligning with the IMPULSE project's goals of enhancing accessibility, streamlining digitization, and fostering collaboration.

Key findings underscore that formats ideal for VEs—such as JPEG for 2D images, PNG for transparency, glTF/GLB for 3D models, and MP4 (with H.264/AAC codecs) for audiovisual content—are generally not the primary output of heritage institutions. Consequently, a crucial conversion and optimization pipeline is necessary to transform archival-quality assets into VE-ready formats. This involves processes like converting archival TIFFs to JPEG or PNG, processing high-resolution PLY or OBJ 3D scans into glTF/GLB, and compressing lossless WAV audio to formats like Ogg Vorbis, MP3 or AAC.

This process of adaptation asks for informed curatorial decisions to preserve essential details and while meeting technical performance targets. The analysis reveals a preservation-application gap, where vast collections of high-quality digitized assets remain underutilized in immersive contexts due to technical fragmentation and incompatible formats. This contributes to a "digital heritage paradox," where digitized content is technically available but functionally inaccessible for new applications.

Ultimately, the report calls for an integrated, proactive strategy where VE requirements are embedded into digitization workflows from the outset. This approach is essential to bridge the gap between preservation and application, ensuring that digital cultural heritage assets are findable, accessible, interoperable, and reusable in the immersive contexts central to the IMPULSE project's mission.

Key words:
2D; 3D; audio; audio-video; cultural heritage; digitization; file formats; interoperability; virtual worlds

3 Table of Contents

Contents

4 Abbreviations and Acronyms

Abbreviation / acronym	Description
2D	Two-Dimensional
3D	Three-Dimensional
3DS	File format from Autodesk 3D Studio software
3GP	Multimedia container format
4D	Four-Dimensional (used in context of audiovisual data)
AAC	Advanced Audio Coding
ABC	Alembic (file format for animated geometry)
AIFF	Audio Interchange File Format (Audio Format)
AP	Application Protocol(s) (in STEP format)
APNG	Animated Portable Network Graphics
API	Application Programming Interface
AR	Augmented Reality
ARCore	Google's Augmented Reality platform
ASCII	American Standard Code for Information Interchange
ASTC	Adaptive Scalable Texture Compression
AVC	Advanced Video Coding (also H.264, MPEG-4 Part 10)
AVCHD	Advanced Video Coding High Definition (Video Container/Format)
AVI	Audio Video Interleave (Video Container/Format)
BCn	Block Compression (texture compression formats)
BMP	Bitmap (Image file format)
B-rep	Boundary Representation
BWF	Broadcast Wave Format
CAD	Computer-Aided Design
CAE	Computer-Aided Engineering
CAM	Computer-Aided Manufacturing
CPU	Central Processing Unit
CR2	Canon RAW 2 (Canon RAW file extension)
CR3	Canon RAW 3 (Canon RAW file extension)
CSS	Cascading Style Sheets
DAE	Digital Asset Exchange (Collada file format)
DAW	Digital Audio Workstation(s)
DCC	Digital Content Creation
DCT	Discrete Cosine Transform
DNG	Digital Negative (Adobe's RAW image format)
DNA	Data structure definition (in Blender's .blend files)
DOM	Document Object Model
DOS	Disk Operating System

DV	Digital Video (Video Container/Format)
DWT	Discrete Wavelet Transform
EU	European Union (from "IMPULSE Project")
EXIF	Exchangeable Image File Format
FBX	Filmbox (3D file format)
FLAC	Free Lossless Audio Codec
FPS	Frames Per Second
GD&T	Geometric Dimensions and Tolerances
GIF	Graphics Interchange Format
GLB	GL Transmission Format Binary (Binary file for glTF)
glTF	GL Transmission Format
GPS	Global Positioning System
GPU	Graphics Processing Unit
H.264	Video Codec (also AVC, MPEG-4 Part 10)
HAnim	Humanoid Animation (X3D component)
HD	High-Definition
HEIC	High Efficiency Image Coding
HEIF	High Efficiency Image File Format
HEVC	High Efficiency Video Coding (also H.265)
HTML	HyperText Markup Language
ID	Identifier
ID3	Metadata tags for MP3 files
IEC	International Electrotechnical Commission
IIIF	International Image Interoperability Framework
IMPULSE	Name of the Project (IMPULSE Project)
IPTC	International Press Telecommunications Council
ISO	International Organization for Standardization
JFIF	JPEG File Interchange Format
JPEG	Joint Photographic Experts Group
JSON	JavaScript Object Notation
LOD	Level(s) of Detail
LZW	Lempel-Ziv-Welch (compression algorithm)
MAX	File format for 3ds Max software
MKV	Matroska Video (Video Container/Format)
MP3	MPEG Audio Layer III
MP4	MPEG-4 Part 14 (Video Container/Format)
MPEG	Moving Picture Experts Group
MTL	Material Template Library (used with OBJ files)
NEF	Nikon Electronic Format (Nikon RAW file extension)
NURBS	Non-Uniform Rational B-Splines
OBJ	Wavefront Object (3D model format)
OpenGL	Open Graphics Library
OpenGL ES	OpenGL for Embedded Systems

PBR	Physically Based Rendering
PCM	Pulse Code Modulation
PLY	Polygon File Format (also Stanford Triangle Format)
PMI	Product Manufacturing Information (in STEP format)
PNG	Portable Network Graphics
RAF	Fujifilm RAW (Fujifilm RAW file extension)
RAM	Random Access Memory (implied by "memory consumption")
RAW	Category for minimally processed image sensor data files
RDF	Resource Description Framework
RIFF	Resource Interchange File Format
RGB	Red, Green, Blue (color model)
SDK	Software Development Kit
SDF	Signed Distance Fields
STEP	Standard for the Exchange of Product model data (ISO 10303)
STL	Stereolithography / Standard Triangle Language
STP	Standard for the Exchange of Product model data (common file extension for STEP)
SVG	Scalable Vector Graphics
TIFF	Tagged Image File Format
TS	Transport Stream (MPEG container format)
UI	User Interface
USD	Universal Scene Description
USDZ	Zero-compression zip archive for USD (for AR delivery)
UV	Texture Coordinates (in 3D modelling)
VE	Virtual Environment
VEs	Virtual Environments (plural)
VFX	Visual Effects
VR	Virtual Reality
VRM	Format for VR Avatars (based on glTF)
VRML	Virtual Reality Modelling Language
WAV	Waveform Audio File Format
WebGL	Web Graphics Library
WebXR	Web Extended Reality (for VR/AR on the web)
WMV	Windows Media Video (Video Container/Format)
X3D	Extensible 3D (file format)
XML	Extensible Markup Language
XMP	Extensible Metadata Platform
XR	Extended Reality

5 Introduction

5.1 IMPULSE Project

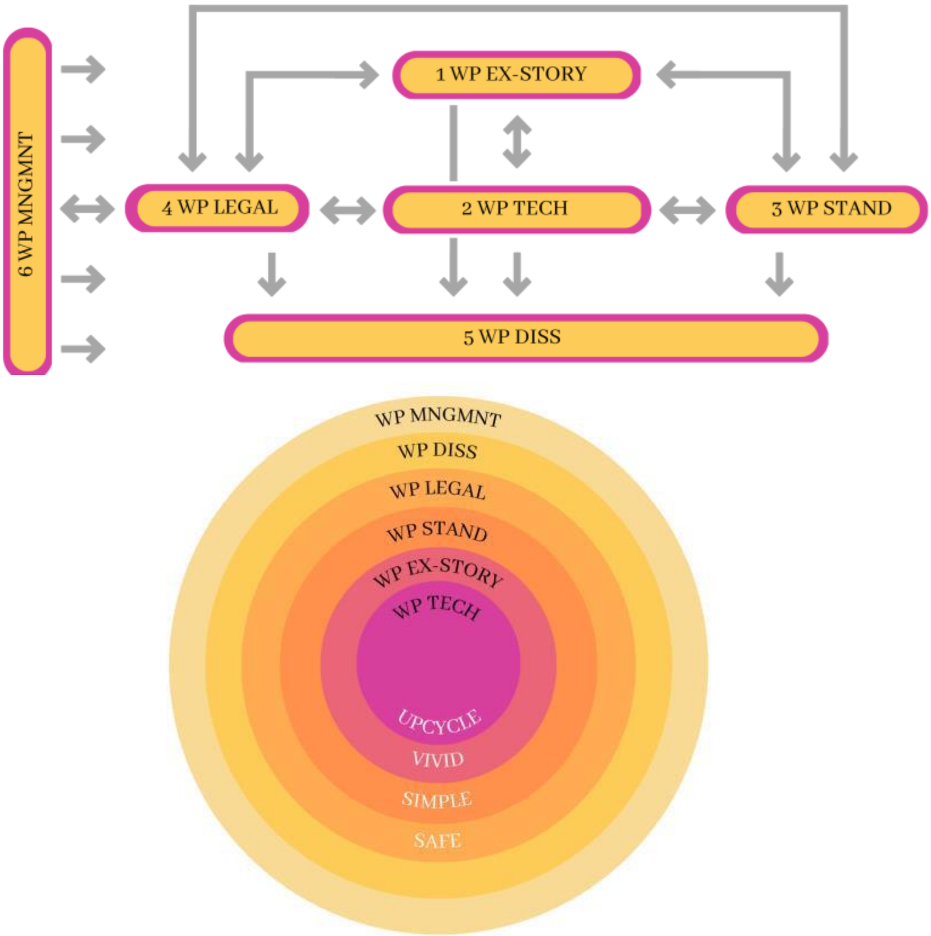
IMPULSE emerged out of the vision of a European immersive digitisation framework, driven by the forces of culture, creativity, storytelling, upcycled technology and safe, simplified standards. The project aims to synthesise innovative, multifaceted solutions and methodologies addressing the digitisation and accessibility processes of the collections that make up digital cultural heritage.

IMPULSE aims to address some of the most pressing gaps in the digitisation of European cultural heritage by building on existing knowledge, the capacity of its partners, and activities and networks. To achieve its stated intentions, the IMPULSE project has conceived a strategic plan which is divided into six distinct yet interconnected work packages (WPs). Each WP is indicative of the stated objectives and monitors the progress of the respective research activities and project implementation initiatives.

These initiatives aim to promote the innovative (re)use of digital cultural heritage, address challenges in platform interoperability and enhance the use of already digitized cultural heritage materials in novel contexts, such as the Metaverse and other immersive platforms. In those platforms, (whether they encompass MR, VR or AR technologies) the upcycling and appropriate reuse of digital assets remains a desideratum that our project will actively address. Additionally, IMPULSE seeks to develop pioneering standardisation protocols and revise the legal framework to better tackle contemporary challenges. Ultimately, the end goal of the project is to be achieved through a set of specific, measurable, achievable, realistic and timebound (SMART) objectives, which entail:

- Solutions that will augment the quantity and range of cultural heritage objects displayed through VR/XR technologies. The now easily accessible collections shall act as a powerhouse for a diverse set of demographic audiences and underrepresented communities, to empower them and help them engage with the topics and themes on display.
- Technological solutions developed within the project that will enable the efficient (re)use of digitised cultural heritage content in novel contexts and immersive environments, with a focus on educational / teaching, artistic and creative dimensions, on par with the three prototypes that are being developed within the project (see IMPULSE proposal).
- Innovative standardisation procedures and simplified strategies specifically targeted towards digitisation processes in emerging platforms, immersive and multi-user fictional environments to achieve easily comprehended formats by deploying existing (technical) standards and metadata / paradata simplification protocols, tailored for the utilisation of education, arts, and the CCSI.

- Legal and organisational frameworks with detailed evaluations of risks and barriers in fields such as the copyrights, database rights, ownership, provenance, personal data protection and other related rights in the field of digital cultural heritage in novel environments. The lack of proper legal frameworks is an identified gap that IMPULSE aims to address, all while working within multiple national jurisdictions (namely the ones of the selected partners, i.e., Poland, Greece, Belgium, Italy, Germany, and Malta) aiming to achieve harmonisation.
- Connections among researchers, artists, cultural heritage practitioners and other relevant stakeholders through initiatives such as the IMPULSE Community of Practice (referred to as IMCO), the Hackathon and the three thematic Workshops surrounding it, as well as the Acceleration & Mentoring Hub, all aiming to promote dialogue, co-creation, and capacity building in immersive digitisation.



To conclude, the overarching objective of IMPULSE is to create innovative and comprehensive solutions that foster the digitisation of cultural heritage in a standardised, findable, accessible, interoperable and reusable manner. The project is in equal parts founded on academic research and existing practice, and cons frequently on methodologies specifically fit for each Work Package.

5.2 Objectives of the Work Package 3

Within IMPULSE, Work Package 3 (WP3) emerged out of a growing need for efficient, easy to apply standardisation practices to facilitate data sharing across various platforms, and more specifically Multi-User Virtual Environments.

Without standardisation, institutions may face challenges in sharing data. Some institutions may be reluctant to adopt certain standards as this comes with sometimes significant long-term changes on systems, digital preservation, data exchange and aggregation. Some institutions may not have elaborate IT or digital resources and are dependent on software tools that have only base standards implemented, which, in some cases, the software vendor customizes without the possibility for adaptation.

Well-established file formats and framework standards are well-documented, widely adopted, and supported by a substantial user base. In cases where these standards do not adequately address the specific needs of professionals in the field and researchers, it may be necessary to develop supplementary or alternative standards. However, from a sustainability perspective, this approach is not always optimal. It is essential to first examine the foundational standards before considering modifications or alternatives.

When it comes to the digital transition and digitisation, the very basics should not be underestimated. Correct and efficient standardisation practices in digitisation workflows ensure that the digital representations of heritage objects are consistent, reliable, findable and comparable, both now and in the future. They enable access to and preservation of the data, facilitate sharing and collaboration among stakeholders from different EU countries, disciplines, and legal frameworks, and support informed decision-making. Without the appropriate standardisation practices data may become fragmented, inaccessible, and susceptible to loss or misinterpretation over time, compromising its value and significance. Most importantly, standardisation enables the creation of trustworthy data, ensuring it can be (re)used in the broadest manner.

More specifically, standardisation is important in the following areas:

- **Consistency**: Standardisation ensures that the data is captured, processed, and stored in a consistent manner, reducing the possibility of errors or unreliable datasets.
- **Interoperability**: Standardisation enables digital heritage data to be shared between different systems and platforms, reducing the risk of data loss, inaccessibility or incomprehension (fuzzy data). Describing the genesis of datasets (equipment, software, algorithms used) will help understanding and interpreting the data.
- **Long-term preservation**: Standardised data is more likely to be preserved and be accessible in the future, ensuring its long-term availability.
- **Improved accuracy**: Standardisation provides clear guidelines and protocols for capturing, processing, and storing data, reducing the possibility of human error and improving accuracy.

It is no exaggeration to claim that, today, the cultural heritage sector is at the forefront of standardising practices. Within heritage institutes digitisation practices have been revolutionised by the development of ISO standards such as the ISO19264 image quality standards that ensure high quality, consistent data at base level for 2D objects or, on the other end of the spectrum ISO 14721:2012 Reference Model for an Open Archival Information System (OAIS) to name a few.

Practical guidelines based on the ISO 19264 standard have been developed and implemented by national institutes both in Europe (such as the Metamorfoze Preservation Imaging Guidelines) and the US (Federal Agencies Digital Guidelines Initiative, FADGI Guidelines). Many cultural institutions have implemented these guidelines, and by building upon them have generated additional practical workflows for specific contexts, for example the Rijksmuseum manuals for photographing 2D objects material types (manuscripts) or data types (3D data, audiovisual material, compression standards for video or for image). For common data formats, standards and guidelines have been developed, but there is a lack of standards regarding advanced imaging techniques such as multi-spectral imaging, multi-light reflectance imaging as they create multimodal datasets.

In terms of metadata, several standards exist for various metadata types, e.g., technical metadata (Exif, IPTC); descriptive metadata (Dublin Core, Marc21, Cidoc/CRM, Premis, specific metadata schema for 3D objects) etc. depending on material types, and depending on type of collection: library, archive or museal. For data interoperability and sharing of data and to ensure the information is consistent, accurate and accessible across systems, several models have been developed: LIDO, Europeana Data Model (EDM), Schema.org or International Image Interoperability Framework (IIIF). Data preservation guidelines provide information for the long-term preservation and management of digital data, ensuring that information remains accessible and usable: OAIS (Open Archival Information System) Reference Model, ISO 14721:2012.

However, as datasets expand in size at a rapid pace, novel digitisation and imaging techniques emerge, and new digital preservation strategies are constantly developed. While standardisation itself provides clear frameworks, the abundance of different standards that coexist often simultaneously used poses several challenges: this standards “pluralism” hampers interoperability, consistency in classification and cooperation for support and development.

The adoption of standards by cultural institutions remains a challenge. Implementing and conforming to standards can be expensive for institutes with limited budgets, as it is a process that requires specialised technical knowledge and skills. A substantial number of institutes have implemented software tools based on proprietary technology, sometimes making it difficult to share and exchange data with other systems. There is also a resistance to change and a reluctance to share data for a variety of reasons, even more so towards novel immersive environments.

IMPULSE tries to answer these challenges by research on simplifying (meta)data sharing strategies toward multi-user virtual environments. This move aims to lift away many barriers, as it is easier for different systems and platforms to exchange and (re)use data, reduces the cost and effort of data transformation and data sharing, and opens opportunities for innovation and experimentation, fostering new and creative reuse of digitized heritage. Emerging data sharing platforms should play a key role in streamlining data ingestion into their systems, instead of outsourcing this task to content providers. This requires not only existing metadata and technical standards; but also, paradata and other documentation that describe the context and conditions of data creation and (future) data use.

6 Introduction: Analysis of 2D, 3D, and 4D File Formats for Virtual Environment Applications in Cultural Heritage Digitisation

6.1 The Challenge of File Format Selection for Virtual Environments

Integrating digitized cultural heritage into virtual environments presents a notable challenge due to the disparity between archival file formats and those suited for real-time interactive experiences. Heritage institutions prioritize formats for long-term preservation, high fidelity, and rich metadata, such as TIFF and JPEG 2000 for 2D images, and PLY or OBJ for 3D models. These formats, while excellent for detailed study and conservation, often result in large file sizes and complex decoding processes that are inefficient for real-time rendering in virtual environments. RAW files, capturing the highest quality sensor data, are also unsuitable for direct VE use due to their size and processing requirements.

Conversely, virtual environments demand formats optimized for rapid loading, efficient GPU rendering, and broad compatibility. This necessitates a conversion and optimization pipeline for digitized cultural heritage assets. For instance, archival TIFFs and RAW files are converted to formats like JPEG where some quality compromise is acceptable for file size reduction. Similarly, high-resolution 3D models undergo processing and decimation before conversion to optimized formats such as glTF/GLB or FBX for use in virtual environments. Audio formats like WAV or FLAC, preferred for lossless archival quality, are often compressed into smaller, lossy formats like MP3, AAC or Ogg Vorbis for background audio in VEs to enhance performance.

The file formats most suitable for virtual environments are generally not created de facto by heritage institutions. Heritage institutions typically generate formats optimized for long-term archival and high-fidelity source data, which are often large and complex. These formats, while crucial for preservation, are often unsuitable for direct use in real-time virtual environments due to their impact on loading times, memory consumption, and rendering performance.

Instead, the formats that excel in virtual environments—such as JPEG for 2D data, glTF/GLB for 3D models, MP3/AAC/Ogg Vorbis for audio and MP4 for audiovisual data—are designed with efficiency, real-time rendering, and broad platform compatibility in mind. For example, glTF/GLB is specifically developed for efficient transmission and loading of 3D scenes in real-time applications. Similarly, JPEG's high compression is advantageous for 2d data where file size is a primary concern. These formats are engineered to integrate seamlessly with game engines and web platforms, which are the foundational technologies for virtual environments. This fundamental difference in design philosophy underscores why a conversion pipeline is essential when migrating cultural heritage data from archival repositories to interactive virtual experiences.

6.2 Scope of Analysis

This report provides a structured technical analysis of a specified list of file formats across three domains:

- 2D data: TIFF, JPEG, JPEG 2000, GIF, PNG, BMP, HEIF/HEIC, RAW (image), pdf, txt, rtf, docx, odf, alto, hOCR (textual documents)
- 3D data: glTF/GLB, OBJ, USD, PLY, FBX, VRML, X3D, DAE, STL, Alembic STEP/STP, Blender (.blend), 3DS (.3ds), 3ds Max (.max)
- 4D data: MP3, WAV, FLAC, AAC, AIFF, Ogg Vorbis (Audio); MP4, AVI, MOV, MKV, WMV, AVCHD, DV (Video Containers/Formats); H.264, MPEG (Video Codecs)

6.3 Relevance to IMPULSE Project Objectives

The analysis directly informs the IMPULSE project's goals to:

- Enhance Accessibility and Presentation: Achieving broad accessibility requires formats that perform well across a range of XR devices (from high-end PCs to mobile VR) and are compatible with web standards for wider reach. High-fidelity representation requires formats capable of accurately capturing geometry, materials, textures, and audio, potentially leveraging Physically Based Rendering (PBR) and high-resolution, lossless data where appropriate.
- Streamline Digitization Processes: Efficient workflows benefit from formats that are easily created or exported from standard digitization tools (scanners, cameras, modeling software) and are readily integrated into VE development platforms (e.g., Unity, Unreal Engine) with the minimal friction or data loss possible.
- Foster Collaboration and Capacity Building: Collaboration is facilitated by the use of standardized, open formats that ensure interoperability between different tools and teams. Robust metadata support is crucial for tracking provenance, authorship, licensing, and semantic information, enabling better asset management and reuse within the IMPULSE Community of Practice.

6.4 Methodology

The analysis of 2D, 3D, and 4D file formats for virtual environment (VE) applications in cultural heritage digitization involves a systematic review for each listed format. The core objective is to assess the suitability for file formats regularly created by heritage institutions for use in virtual environments to enhance accessibility, streamline digitization, and foster collaboration.

The methodology for evaluating each file format is structured around several key criteria.

- Overview: Each format of analysis begins with a general description, including its primary purpose and key characteristics. This provides foundational context before delving into specific technical attributes
- Usability: assessment of using the file format in a VE based on
  - Creation/modification: evaluation of the ease of use of the file format
  - Performance/Resource Requirements: This examines the format's impact on real-time application performance, considering factors like file size, loading times, and resource demands.
  - Engine/Platform Integration: The complexity of integrating the format with common game engines (e.g., Unity, Unreal Engine) and VR/AR platforms is assessed.
- Interoperability: This evaluates compatibility and data fidelity when transferring the format between content creation software, virtual/augmented reality systems, and web platforms.
- Metadata Embedding Capabilities: This section investigates the types of metadata supported by the format, the existence of standardized schemas, and the ease of access and utilization by applications.
- Suitability for IMPULSE or Virtual Environments: This final aspect synthesizes the findings from the previous criteria to discuss the overall appropriateness of the format for virtual environment applications within the cultural heritage domain, specifically considering the IMPULSE project's aims.

7 Analysis of 2D File Formats for Virtual Environments

The scope of 2D formats relevant to cultural heritage VEs is broad, encompassing both image data and textual data, which presents documentary heritage, provides contextual information, and enables narrative exploration. Each image file offers different balances of fidelity, compression, and feature sets ranging from RAW image data that comes directly from the sensor over high quality TIFF raster image for archival purposes to more efficient file formats such as JPEG. Concurrently, textual file formats are vital for integrating written cultural heritage, varying from simple plain text (TXT) and richly formatted documents (PDF, DOCX) to structured OCR outputs (ALTO, hOCR) that make digitized text computationally accessible.

7.1 Image File Formats

7.1.1 TIFF (Tagged Image File Format)

- **Overview:** TIFF is a versatile, high-quality raster image format frequently employed in digitisation and digital archiving workflows. It is known for its ability to store image data lossless, although options for lossy compression also exist. TIFF supports multiple layers within a single file, various color depths (including high bit depths), and different color spaces.
- **Usability:**
    - *Creation/Modification:* TIFF is a standard format, making creation and modification straightforward in professional contexts.
    - *Performance/Resource Requirements:* The primary drawback for VE use is file size. Lossless compression or lack of compression results in large files compared to formats like JPEG or optimized texture formats. These large sizes negatively impact VE loading times, bandwidth usage (if streamed), and especially texture memory consumption on the GPU. While game engines might support TIFF import, they typically convert these files into more performant, GPU-friendly formats during the asset import process. Direct runtime loading of TIFF textures is generally inefficient and not recommended for real-time applications.
- *Engine/Platform Integration:* Major engines like Unity and Unreal Engine can import TIFF files, but this is an offline process. The engine converts the TIFF data into an internal or platform-specific format optimized for GPU texture sampling. Direct use at runtime is impractical due to performance penalties associated with large file sizes and potentially complex decompression schemes not optimized for GPUs.

- **Interoperability:**
    - *Content Creation Software:* Excellent support for professional image editing and desktop publishing software.
    - *VE/AR Systems & Web Platforms:* Poorly suited for direct use. Large file sizes make web delivery impractical, and native browser support is minimal. While VR/AR hardware *could* display textures originating from TIFFs after engine conversion, the TIFF format itself is not directly handled at runtime. Furthermore, the existence of various TIFF sub-formats and compression options can sometimes lead to compatibility issues between different software.
- **Metadata Embedding Capabilities:**
    - TIFF offers robust metadata support, leveraging its tag-based structure. It can embed standard metadata schemas including EXIF (camera information), IPTC (descriptive information), and XMP (extensible metadata). This allows for storing extensive information about the image's origin, content, copyright, and technical parameters, which is highly valuable for archival and asset management. Some systems can map custom metadata into XMP fields upon download. The format's support for layers can also potentially be used to store structural or semantic information, although this is not a standardized practice for metadata.
- **Suitability for IMPULSE and Virtual Environments:**
    - TIFF is highly suitable as an archival format for high-fidelity source images, such as master scans of documents, photographs, or artworks, due to its lossless quality preservation and comprehensive metadata support. This aligns with the preservation aspects of cultural heritage projects.
    - It is unsuitable for direct use as runtime textures or UI elements within VEs due to significant performance drawbacks related to file size and loading/memory overhead.
- A typical workflow involving TIFFs would see them used as high-quality source assets that are then processed and converted into optimized formats (e.g., PNG for UI or elements needing transparency, jpg for smaller file) for efficient delivery and rendering within the virtual environment. This need for conversion is a common pattern in VE pipelines where source asset quality differs from runtime asset requirements.

7.1.2 JPEG (Joint Photographic Experts Group)

- **Overview:** JPEG is the most prevalent format in digital photography, utilizing a lossy compression algorithm. Its primary goal is to achieve significant file size reduction (typically 10:1 compression or more) with acceptable perceptual quality loss, making it ideal for sharing and transmission. JFIF (JPEG File Interchange Format) defines the standard file structure for containing the JPEG-compressed data stream.

- **Usability:**
  - *Creation/Modification:* Universally supported by virtually all image digital cameras and editing software. The level of compression (and thus quality vs. file size) is typically adjustable during saving.
  - *Performance/Resource Requirements:* Small file sizes resulting from lossy compression are advantageous for faster loading times, reduced bandwidth consumption, and lower storage requirements compared to lossless formats. Decoding JPEG images is generally fast and well-optimized. However, lossy compression can introduce visible artifacts (like blocking, ringing or banding), especially at lower quality settings or on images with sharp edges or fine details. These artifacts can be more noticeable on high-resolution VR displays or when textures are viewed up close. While suitable for many photographic textures in VEs, careful quality control is needed. Engines typically convert JPEGs to GPU-optimized formats upon import.
  - *Engine/Platform Integration:* Natively supported for import by all major game engines, including Unity and Unreal Engine. It's a common format for textures, especially for environmental assets or where file size is a primary concern.
- **Interoperability:**
  - *Content Creation Software:* Universal compatibility with image editors, viewers, and digital asset management systems.
  - *VE/AR Systems & Web Platforms:* Native support across all web browsers and operating systems makes it ideal for web delivery. Widely usable on mobile devices and VR/AR platforms (after engine processing).
- **Metadata Embedding Capabilities:**
  - JPEG files robustly support standard metadata formats embedded within the file structure. This includes EXIF (Exchangeable Image File Format), commonly used by digital cameras to store technical shooting parameters, date/time, and GPS location. It also supports IPTC (International Press Telecommunications Council) metadata, often used for descriptive information like captions, keywords, copyright, and creator details. Additionally, XMP (Extensible Metadata Platform) data can be embedded, offering a flexible and extensible way to store various types of metadata. This comprehensive metadata support is valuable for tracking provenance and descriptive information.
- **Suitability for IMPULSE and Virtual Environments:**
  - JPEG is suitable for delivering images and photographic textures in VEs, particularly when file size and loading performance are critical considerations, such as web-based experiences or applications targeting mobile VR. Its metadata capabilities are beneficial for associating provenance and descriptive information with images.
  - However, its lossy compression makes it less ideal for assets requiring perfect fidelity, such as detailed scans of artifacts or graphics with sharp lines or text, where compression artifacts could misrepresent an object. The lack of transparency support also limits its use for UI elements.
  - The inherent trade-off between compression and quality in JPEG necessitates careful consideration within a cultural heritage context. While efficient,

the potential loss of detail might be unacceptable for certain artifacts, requiring the use of lossless formats like PNG for those specific assets, even if it means larger file sizes.

7.1.3 JPEG 2000 (.jp2,.jpx,.j2k,.j2c)

- **Overview:** Developed as a successor to the original JPEG format, JPEG 2000 employs a compression scheme based on discrete wavelet transforms (DWT). It offers several technical advantages, including higher compression efficiency (better quality at the same file size, or smaller size at the same quality compared to JPEG), support for both lossless and lossy compression within a single architecture, and progressive decoding capabilities. Progressive decoding allows an image to be reconstructed gradually, either by resolution (displaying a lower-resolution version first) or by quality, which is beneficial for large images or streaming scenarios. It also features better error resilience, support for high bit depths, transparency (alpha channels), and various colour spaces. The format is deployed in frameworks such as IIIF (International Image Interoperability Framework), used by many heritage institutions to share digitised objects in a 2D format.
- **Usability:**
  - *Creation/Modification:* Support for creating and editing JPEG 2000 files is significantly less common in mainstream consumer and prosumer image editing tools compared to JPEG or PNG. Specialized software, libraries (like OpenJPEG), or plugins are often required.
  - *Performance/Resource Requirements:* The wavelet transform and entropy coding used by JPEG 2000 can be more computationally complex to encode and decode than JPEG, potentially requiring more CPU resources and memory. While offering better compression ratios, this computational overhead can be a drawback for real-time applications on constrained hardware. The scalability features, however, could theoretically benefit VEs by allowing adaptive streaming of large textures based on viewpoint or bandwidth, although this requires specific implementation support.
  - *Engine/Platform Integration:* Native support for JPEG 2000 is generally absent in major game engines like Unity and Unreal Engine. Integrating JPEG 2000 assets would likely necessitate the use of third-party libraries or plugins for decoding, adding complexity and potential performance bottlenecks. A workflow would almost certainly involve converting JPEG 2000 assets to engine-supported formats prior to or during import.
- **Interoperability:**
  - *Content Creation Software:* Limited support outside specialized domains. Not a common interchange format for general graphics workflows.
  - *VE/AR Systems & Web Platforms:* Poor interoperability. Native web browser support is limited (only Safari was noted as supporting it). Use on the web generally requires server-side conversion or client-side JavaScript decoders (which would likely have significant performance costs). Not practical for broad VE/AR deployment across diverse hardware and platforms.
- **Metadata Embedding Capabilities:**

- The JP2 and JPX file formats, which act as containers for the JPEG 2000 codestream, provide mechanisms for embedding metadata. This can include color space information, resolution details, and potentially standardized schemas like EXIF or XMP (XMP support is mentioned for JP2). The JPX (JPEG 2000 Part 2) extension offers more advanced features, including richer metadata support.

- **Suitability for IMPULSE and Virtual Environments:**
  - JPEG 2000's strengths in lossless compression, high bit depth support, and scalability make it a potentially strong candidate for archival purposes, like TIFF, but with better compression efficiency.
  - However, its significant lack of widespread adoption, limited software and tool support, poor native browser compatibility, and potential performance overhead for decoding make it impractical for use as a delivery format for images, textures or UI elements in VEs intended for broad accessibility.
  - Relying solely on it for delivery within the IMPULSE project would introduce significant technical hurdles and compatibility risks.

7.1.4 GIF (Graphics Interchange Format)

- **Overview:** GIF is one of the earliest raster image formats widely used on the web, developed by CompuServe in 1987. It uses lossless LZW compression but is fundamentally limited to a palette of a maximum of 256 colours (8-bit indexed color) per frame. Its defining features are support for simple frame-based animation and the ability to designate one color in the palette as transparent (binary transparency, not alpha blending).
- **Usability:**
  - *Creation/Modification:* Supported by nearly all image editing software, although primarily for legacy purposes or simple animations. Creating simple animations is straightforward.
  - *Performance/Resource Requirements:* For images with very few colours (like simple logos or diagrams), the combination of indexed color and LZW compression can result in very small file sizes. However, for images with continuous tones or complex graphics, the 256-color limit leads to significant quality degradation (dithering or banding). Animated GIFs can become large and inefficient for complex or long sequences compared to modern video codecs. Engine support exists, but GIFs are rarely used for in-game textures due to color limitations. They might occasionally be used for simple UI animations via plugins, but performance can be suboptimal.
  - *Engine/Platform Integration:* While engines like Unity and Unreal might be able to import GIF files (often converting them), they are not a recommended format for textures or sophisticated UI elements due to the severe color palette restriction. For UI animation, sprite sheets or engine-specific animation systems are generally preferred for better quality and control.

- **Interoperability:**
  - *Content Creation Software:* Universally supported by image editing tools and web design software, though often superseded by PNG for static images needing lossless quality or transparency.
  - *VE/AR Systems & Web Platforms:* Universal support in all web browsers makes it highly interoperable for web content. Its persistence is largely due to its simplicity and ubiquity in online communication (memes, simple animations). It is not suitable for direct use in demanding VE/AR applications requiring high visual fidelity.
- **Metadata Embedding Capabilities:**
  - GIF supports embedding textual information through Comment Extension blocks within the file structure. These are intended for human-readable comments, credits, or descriptions and are not part of the image data itself. GIF does not have standardized support for rich metadata formats like EXIF, IPTC, or XMP. Its metadata capabilities are therefore very basic and unstructured. Application-specific data can also be embedded using Application Extension blocks, but this is not standardized for metadata purposes.
- **Suitability for IMPULSE and Virtual Environments:**
  - GIF is generally unsuitable for representing cultural heritage assets in VEs due to its severe 256-color limitation, which cannot accurately capture the nuances of photographs, artworks, or realistic textures. Its binary transparency is less flexible than the alpha channels offered by PNG or HEIF.
  - While animated GIFs could potentially be used for very simple UI indicators, modern alternatives like APNG, WebP animation, or video textures offer significantly better quality, color support, and often better performance.
  - The extremely limited metadata support is also a significant drawback for a project focused on documenting and sharing cultural heritage information. Its primary relevance is in legacy web contexts or informal digital communication, not high-fidelity virtual environments.

7.1.5 PNG (Portable Network Graphics)

- **Overview:** PNG is a raster graphics file format designed as a patent-free, improved replacement for GIF. It utilizes lossless data compression. PNG supports a wide range of colour types, including palette-based (like GIF), grayscale, and true colour (24-bit RGB or 48-bit RGB), significantly exceeding GIF's 256-color limit. A key feature is its support for an alpha channel, allowing for variable transparency (semi-transparency), not just the binary transparency of GIF. PNG is intended for single static images; animation is handled by the related APNG format.
- **Usability:**
  - *Creation/Modification:* Universally supported by all modern image editing software. Its lossless nature makes it ideal for intermediate storage during editing workflows, as no data is lost upon saving, unlike JPEG.
  - *Performance/Resource Requirements:* Lossless compression typically results in larger file sizes than lossy JPEG for photographic images but can be smaller

than JPEG or even GIF for images with large areas of flat colour, sharp lines, or text. The alpha channel support is crucial for UI elements, decals, and textures requiring transparency (e.g., foliage, fences). Decoding performance is generally good. In game engines, PNGs are widely used for UI assets and textures where high fidelity or transparency is needed. Like other formats, they are often converted to GPU-optimized texture formats upon importing into engines to improve runtime performance and reduce GPU memory usage.
  - *Engine/Platform Integration:* Excellent native support for Unity and Unreal Engine. It's a standard format for UI elements, sprites, and any texture requiring lossless quality or an alpha channel. Integration is straightforward.
- **Interoperability:**
  - *Content Creation Software:* Excellent compatibility across the board – editors, viewers, asset management systems.
  - *VE/AR Systems & Web Platforms:* Universal support in modern web browsers makes it a reliable choice for web graphics, especially those needing transparency. It's a popular format for web development alongside JPEG and SVG. Suitable for assets used in VR/AR applications developed with standard engines.
- **Metadata Embedding Capabilities:**
  - PNG supports embedding metadata through various "chunks" defined in the specification. Standard chunks exist for textual information, gamma correction (gamma), chromaticity (CHRM), embedded ICC color profiles (ICC), and physical pixel dimensions (pHYs). An EXIF chunk can store EXIF metadata, and the format can also accommodate embedded XMP metadata. While flexible, the consistency of metadata interpretation across all software might be less uniform than JPEG's established EXIF/IPTC standards. Some systems might not display embedded metadata easily.
- **Suitability for IMPULSE and Virtual Environments:**
  - PNG is an excellent and highly recommended format for numerous assets within the IMPULSE project. Its lossless compression ensures fidelity for representing textures of artifacts or detailed graphics. The robust alpha channel support is essential for creating clean UI elements, overlays, decals, or textures with transparency (e.g., vegetation, glass) in VEs.
  - It offers a good balance between quality, features (transparency), and widespread compatibility across software, engines, and web platforms. Its metadata capabilities are sufficient for embedding essential provenance and descriptive information. While file sizes can be larger than JPEG for photos, this is often an acceptable trade-off for the quality and transparency benefits, particularly UI and critical object textures. It serves as a reliable default choice for non-photographic raster assets or those needing transparency.

7.1.6 BMP (Bitmap)

- **Overview:** BMP is an early raster image file format developed by Microsoft, primarily associated with the Windows operating system. It is known for typically storing image data in an uncompressed format. This lack of effective compression results in very large file sizes but preserves the original pixel data perfectly. The format is device-independent and supports various color depths, from monochrome to 24-bit true color.
- **Usability:**
    - *Creation/Modification:* Supported by basic image editors like Microsoft Paint and many other graphics applications, particularly on Windows. However, it's rarely used in modern professional graphics workflows due to its inefficiency.
    - *Performance/Resource Requirements:* The extremely large file sizes associated with uncompressed BMPs are a major performance bottleneck. They lead to slow loading times, high memory consumption (especially for textures), and significant bandwidth usage if transferred over networks. BMP is fundamentally unsuitable for real-time performance in VEs.
    - *Engine/Platform Integration:* While game engines like Unity or Unreal might be able to import BMP files (likely converting them immediately), using BMP assets directly in a build is strongly discouraged due to severe performance implications. There is no practical reason to use BMP over formats like PNG or even TIFF in an engine pipeline.
- **Interoperability:**
    - *Content Creation Software:* Good compatibility with Windows-based software. Support on other platforms (macOS, Linux) exists but it's not a preferred format.
    - *VE/AR Systems & Web Platforms:* Completely unsuitable for web delivery due to excessive file sizes. No role in modern VE/AR asset delivery pipelines.
- **Metadata Embedding Capabilities:**
    - The BMP file format itself has very limited metadata capabilities. The file headers contain basic technical information such as image dimensions, colour depth, and the (usually uncompressed) compression scheme. It does not have standardized support for embedding rich descriptive or technical metadata like EXIF, IPTC, or XMP.
- **Suitability for IMPULSE and Virtual Environments:**
    - BMP is **entirely unsuitable** for use in the IMPULSE project for any purpose related to virtual environments (textures, UI, web delivery) due to its extremely poor compression and resulting performance issues. It offers no advantages over more modern, efficient formats like PNG (for lossless quality) or JPEG (for lossy compression). Its relevance is purely historical or for specific legacy Windows applications.

7.1.7 HEIF/HEIC (High Efficiency Image File Format / High Efficiency Image Coding)

- **Overview:** HEIF is a modern container file format designed to store individual images or image sequences, often utilizing the HEVC (H.265) video codec for compression – in which case the files typically use the .heic extension. Apple adopted HEIC as the default format for photos taken on recent iPhones and iPads, aiming to provide significantly better compression efficiency than JPEG (roughly halving the file size for similar visual quality). The HEIF container itself is flexible, capable of storing multiple images (e.g., bursts, live photos), auxiliary data like depth maps and alpha channels, thumbnails, and metadata, all within a single file.

- **Usability:**
    - *Creation/Modification:* Primarily created by default on modern Apple devices. Support for editing HEIF/HEIC files is improving in third-party software (e.g., Adobe Photoshop, Affinity Photo) but is not yet as universal as JPEG or PNG. Conversion to more widely compatible formats is often necessary for broader workflows.
    - *Performance/Resource Requirements:* The primary advantage is high compression efficiency, leading to smaller file sizes compared to JPEG for similar quality. This benefits storage space and transfer speeds. However, decoding HEVC-encoded images (HEIC) can be computationally more demanding than decoding JPEGs, although modern hardware often includes dedicated HEVC decoding capabilities, especially on mobile devices. Direct support in game engines is lacking.
    - *Engine/Platform Integration:* Native support for HEIF/HEIC is not standard in major game engines like Unity or Unreal Engine. Assets in this format would need to be converted to engine-supported formats (e.g., PNG, JPEG, or engine-specific compressed textures) during the development of pipeline.
- **Interoperability:**
    - *Content Creation Software:* Native support is strong within the Apple ecosystem (macOS, iOS). Windows requires the installation of specific HEIF Image Extensions and potentially HEVC Video Extensions from the Microsoft Store for native viewing and use in some applications. Android platform support has been growing on newer devices and OS versions. Overall software compatibility is significantly less widespread than JPEG or PNG. Potential patent licensing issues surrounding the HEVC codec may also hinder broader, free adoption.
    - *VE/AR Systems & Web Platforms:* Poor interoperability for direct use. Web browser support is very limited (primarily Safari). Not suitable for direct web delivery to a broad audience without server-side conversion or client-side fallbacks.
- **Metadata Embedding Capabilities:**
    - As a container format based on the ISO Base Media File Format (like MP4), HEIF is designed to encapsulate various data streams, including metadata. It supports standard EXIF metadata, like JPEG, capturing camera settings, GPS data, etc. The container structure also allows for storing auxiliary data streams like depth maps (useful for computational photography effects like portrait mode) or alpha channels for transparency. It can also store image sequences (bursts, animations) and non-destructive editing information.
- **Suitability for IMPULSE and Virtual Environments:**
    - The high compression efficiency and potential for storing auxiliary data like depth maps make HEIF/HEIC technically interesting, especially if source imagery for digitization originates from modern iPhones or involves computational photography techniques.

- However, the significant interoperability challenges, limited software support outside the Apple ecosystem, poor web compatibility, and lack of native game engine support make HEIF/HEIC currently unsuitable as a primary delivery or interchange format for a collaborative, cross-platform project like IMPULSE.
- If HEIC files are received as source material (e.g., photos taken during fieldwork with iPhones), a robust conversion step to more standard formats like JPEG will be essential early in the pipeline. The format's trajectory highlights the difficulties in displacing deeply entrenched standards like JPEG, even with technically superior alternatives, due to ecosystem inertia and compatibility hurdles.

7.1.8 RAW

- **Overview:** RAW is not a single file format but rather the general term for a family of file formats that store minimally processed image data captured directly by the sensor of a digital camera (or scanner). Each camera manufacturer typically uses its own proprietary RAW format (e.g., Canon's .CR2/.CR3, Nikon’s .NEF, Sony's .ARW, Fujifilm's .RAF), although Adobe's DNG (Digital Negative) format aims to be an open, standardized archival RAW format. RAW files contain the "raw" sensor data, often with higher bit depth (e.g., 12-bit, 14-bit, or 16-bit) than standard 8-bit formats like JPEG, preserving a wider dynamic range and more colour information. This allows for maximum flexibility during post-processing, enabling significant adjustments to exposure, white balance, colour grading, and noise reduction with minimal quality degradation. RAW files are typically large.
- **Usability:**
  - *Creation/Modification:* RAW files are generated directly by cameras set to capture in RAW mode. They cannot be edited directly like JPEGs; they must be processed using specialized RAW converter software (e.g., Adobe Camera Raw/Lightroom, Capture One, manufacturer-specific software or open-source software such as Rawtherapee or Darktable). This processing step involves demosaicing (interpreting the sensor data), applying colour profiles, adjusting parameters, and then exporting the image to a standard viewable format (like TIFF or JPEG).
  - *Performance/Resource Requirements:* RAW files are very large and require significant processing (demosaicing, colour mapping) before they can be displayed. They are completely unsuitable for direct use in real-time applications like virtual environments. Performance within game engines is not applicable, as they are never loaded directly.
  - *Engine/Platform Integration:* No direct support in game engines like Unity or Unreal. RAW files must always be processed and converted into standard image formats (TIFF, PNG, JPEG, etc.) before they can be imported into an engine or used in a VE pipeline.

- **Interoperability:**
    - *Content Creation Software:* Interoperability is a challenge due to the proliferation of proprietary formats. RAW processing software needs to be regularly updated to support files from new camera models. DNG offers a potential solution for better long-term compatibility and archival.
    - *VE/AR Systems & Web Platforms:* Not directly viewable or usable. Requires conversion to standard formats like JPEG or PNG for web display or VE integration.
- **Metadata Embedding Capabilities:**
    - RAW files typically contain extensive metadata. This includes standard EXIF data with detailed camera settings, lens information, date/time, etc. They often also contain manufacturer-specific metadata ("makernotes") with additional technical details. Edits and adjustments made in RAW processing software are often stored non-destructively, either within the RAW file itself (e.g., in DNG) or more commonly in associated "sidecar" files (e.g., XMP files) or in a catalogue database. This preserves the original RAW data while tracking processing steps and user-added metadata (keywords, ratings, etc.).
- **Suitability for IMPULSE and Virtual Environments:**
    - RAW files are highly suitable and recommended for **archiving the original captures** from digital photography efforts within the project (e.g., documenting artifacts, sites, or events). They represent the highest quality of source data, preserving maximum information and offering the greatest flexibility for future use, analysis, or reprocessing.
    - They are **completely unsuitable** for direct use within virtual environments. A mandatory workflow step involves processing these RAW files and exporting them into appropriate delivery formats (e.g., PNG for textures needing transparency, JPEG for photographic textures where some loss is acceptable, TIFF for intermediate high-quality masters) that can be imported and optimized by the chosen VE platform or game engine. This distinction between the archival "digital negative" (RAW) and the optimized delivery asset is fundamental to a professional digital asset pipeline.

7.2 Textual File Formats

7.2.1 PDF (Portable Document Format)

- **Short Overview:** PDF is a file format developed by Adobe to present documents, including text formatting and images, in a manner independent of application software, hardware, and operating systems. It has become an open standard (ISO 32000) and is widely used for document exchange, publishing, and archiving. PDFs encapsulate a complete description of a fixed-layout flat document, including text, fonts, vector graphics, raster images, and other information needed to display it.
- **Usability:**
  - Creation/Modification: PDFs are typically created from other document formats (e.g., word processors, desktop publishing software) via "Save As" or "Print to PDF" functionalities or images. Dedicated PDF editing software (like Adobe Acrobat Pro, Foxit PDF Editor, etc.) allows for modification, though extensive editing of content can be less flexible than in the source document format. Within the file, a text layer can be implemented to embed OCR textual data.
  - Performance/Resource Requirements: PDF rendering is generally efficient, and viewers are available on almost all platforms. File sizes can vary widely depending on the content (text-only PDFs are small; image-heavy or complex vector graphic PDFs can be large). Complex PDFs can sometimes be slow to load or render on less powerful devices.
  - Engine/Platform Integration (VE/AR Context): Displaying PDFs within a VE/AR environment usually involves rendering the PDF pages as 2D textures, which can then be applied to virtual screens or objects. This might require third-party libraries or plugins within game engines or VE platforms.
- **Interoperability:**
  - Content Creation Software: Excellent interoperability. Most document creation applications can export to PDF. PDF viewers are ubiquitous and needed.
  - VE/AR Systems & Web Platforms: PDFs are widely viewable on web platforms through browser-native support or JavaScript libraries (like PDF.js). For VE/AR systems, integration often means converting PDF pages to images or using specialized UI elements capable of displaying rendered PDF content. The fixed-layout nature can be a limitation for responsive display in varied VE contexts.
- **Metadata Embedding Capabilities:** PDFs have robust support for metadata. This includes standard document properties (Title, Author, Subject, Keywords, Creator, Producer, CreationDate, ModDate) stored in the Document Information Dictionary. More extensively, PDFs support XMP (Extensible Metadata Platform) metadata, allowing for rich, structured, and custom metadata to be embedded. PDF/A (PDF for Archiving) standards often mandate specific metadata requirements.
- **Suitability for IMPULSE and VE:** Highly suitable for distributing and displaying fixed-layout documents, reports, publications, and scanned materials within the IMPULSE

project and in VEs (e.g., as virtual books, information panels). Its strength lies in preserving document fidelity across platforms. For interactive text or adaptable content in VEs, extracting text or using other formats might be preferred. PDF/A is particularly suitable for long-term archival of project documentation.

7.2.2 TXT (Plain Text File)

- **Short Overview:** TXT is a file format that contains plain, unformatted text. It stores characters using standard character encodings such as ASCII, UTF-8, or UTF-16. It is one of the simplest and most universal file formats.
- **Usability:**
  - Creation/Modification: Can be created and modified by virtually any text editor (e.g., Notepad, Notepad++ TextEdit). Simplicity is its key feature.
  - Performance/Resource Requirements: Extremely lightweight with very small file sizes. Parsing and rendering are very fast and require minimal system resources.
  - Engine/Platform Integration (VE/AR Context): Text from .txt files can be easily loaded and displayed in VE/AR environments using standard UI text elements in game engines or platforms. It's suitable for displaying simple textual information, labels, or as a source for text-to-speech.
- **Interoperability:**
  - Content Creation Software: Universally supported by all text editors and most software that handles text.
  - VE/AR Systems & Web Platforms**:** Text content is easily integrated into web pages and any VE/AR application that needs to display text.
- **Metadata Embedding Capabilities:** TXT files inherently do not support embedded metadata within the file structure itself. Any metadata (e.g., author, creation date) must be stored separately or inferred from the file system.
- **Suitability for IMPULSE and VE:** Suitable for simple textual data, notes, configuration files, or as a basic format for textual content that needs to be displayed or processed in VEs. Its lack of formatting and metadata support makes it unsuitable for rich documents but excellent for raw text content. Ensuring consistent use of character encodings (preferably UTF-8 for broad language support) is important.

7.2.3 RTF (Rich Text Format)

- **Short Overview:** RTF is a proprietary document file format developed by Microsoft for cross-platform document interchange. It allows for text formatting (font, color, size), basic page layout, and embedding images. RTF files are human-readable to some extent, as they use control words and groups to represent formatting.

- **Usability:**
    - Creation/Modification: Supported by many word processors (including Microsoft Word, LibreOffice Writer, Apple TextEdit) for both import and export. Editing is straightforward in these applications.
    - Performance/Resource Requirements: File sizes are generally larger than plain TXT but can be smaller than DOCX for similar content, especially if no complex objects are embedded. Parsing can be more resource-intensive than plain text.
    - Engine/Platform Integration (VE/AR Context): Direct rendering of RTF within game engines or VE platforms is uncommon. Typically, RTF content would need to be converted to plain text (losing formatting) or rendered to an image/texture for display in a VE. Libraries exist to parse RTF, which could be used to extract and display formatted text in custom UI elements.
- **Interoperability:**
    - Content Creation Software: Good interoperability across a wide range of word processing applications on different operating systems.
    - VE/AR Systems & Web Platforms: Not directly supported for display in web browsers. For VE/AR, conversion is generally required.
- **Metadata Embedding Capabilities:** RTF can store some basic document properties (e.g., title, author, subject, keywords) within its structure using specific control words (e.g., \title, \author). However, its metadata capabilities are less extensive and standardized compared to formats like PDF or DOCX with XMP support.
- **Suitability for IMPULSE and VE:** Can be useful for exchanging formatted text documents where a degree of styling is needed and broader compatibility than DOCX is desired without relying on full PDF layout. For VE, the text would likely be extracted, or the document converted, for display. Its suitability is moderate, mainly as an interchange format that might then be processed for VE use.

7.2.4 DOCX (Office Open XML Document)

- **Short Overview:** DOCX is the default file format for Microsoft Word, introduced with Microsoft Office 2007. It is an XML-based format, part of the Office Open XML (OOXML) standards. A DOCX file is a ZIP archive containing various XML files and other resources that make up the document (content, styles, metadata, images, etc.).
- **Usability:**
    - Creation/Modification: Primarily created and edited using Microsoft Word. Other office suites like LibreOffice Writer, Google Docs, and Apple Pages offer varying degrees of compatibility for opening and saving DOCX files.
    - Performance/Resource Requirements: File sizes can vary significantly based on content complexity (embedded images, objects, formatting). Rendering complex DOCX documents requires a compatible word processing application.
    - Engine/Platform Integration (VE/AR Context): Direct rendering of DOCX files in game engines or VE platforms is not supported. To use content from DOCX

files in a VE, it would typically be converted to plain text, HTML, images, or PDF, or text content would be extracted for display in UI elements.

- **Interoperability:**
    - Content Creation Software: Excellent interoperability with Microsoft Word. Good (though sometimes not perfect) interoperability with other major office suites.
    - VE/AR Systems & Web Platforms: Not directly viewable in web browsers without conversion (e.g., using online viewers like Google Docs or Microsoft Office Online which render them to HTML or other formats). For VE/AR, conversion or content extraction is necessary.
- **Metadata Embedding Capabilities:** DOCX files have robust metadata support. This includes standard document properties (e.g., Title, Author, Subject, Keywords, Category, Status) accessible through the application's properties interface. Additionally, as an OOXML format, it can embed custom XML parts for extended metadata and supports Dublin Core metadata elements.
- **Suitability for IMPULSE and VE:** Excellent for creating, sharing, and archiving richly formatted source documents within the IMPULSE project (e.g., reports, research papers, detailed documentation). For use in VEs, content will need to be extracted, or the document converted to a displayable format (e.g., PDF pages shown as textures, or text content for UI elements). Its rich formatting capabilities are less directly applicable to real-time VE rendering but crucial for source material.

7.2.5 ODF (OpenDocument Format)

- **Short Overview:** ODF is an open standard, XML-based file format for office documents, including text documents (.odt), spreadsheets (.ods), presentations (.odp), and graphics (.odg). It was developed by OASIS and is the native format for office suites like LibreOffice and Apache OpenOffice. It is designed to provide an application-independent and vendor-neutral format for office documents.
- **Usability:**
    - Creation/Modification: Primarily created and edited using office suites that support ODF natively, such as LibreOffice and Apache OpenOffice. Microsoft Office also has support for ODF formats.
    - Performance/Resource Requirements: Like DOCX, file sizes and performance depend on content complexity. Rendering requires a compatible office application.
    - Engine/Platform Integration (VE/AR Context): Like DOCX, direct rendering of ODF files (e.g., odt) in game engines or VE platforms is not common. Content would need to be converted or extracted for display in VEs.
- **Interoperability:**
    - Content Creation Software: Excellent interoperability among office suites that prioritize open standards. Good support in major office suites, including Microsoft Office.

  - VE/AR Systems & Web Platforms: Not directly viewable in most web browsers without extensions or conversion. Some web-based office suites can view/edit ODF. For VE/AR, conversion or content extraction is required.
- **Metadata Embedding Capabilities:** ODF has a powerful and standardized metadata system. It uses a meta.xml file within the ODF package (which is a ZIP archive) to store predefined metadata elements (based on Dublin Core, such as Title, Creator, Subject, Description, Date) and user-defined metadata fields. This facilitates good metadata management.
- **Suitability for IMPULSE and VE:** Highly suitable as an open standard format for creating, sharing, and archiving richly formatted documents within the IMPULSE project, promoting long-term accessibility and interoperability. Its status as an ISO standard is beneficial. As with DOCX, for VE use, content needs to be extracted or converted. Its strong, standardized metadata support is an advantage.
-

7.2.6 ALTO (Analyzed Layout and Text Object) XML

- **Short Overview:** ALTO is an XML schema designed to store layout and content information for digitized written material (like pages of a book or newspaper) that has been processed by Optical Character Recognition (OCR) software. It describes the physical structure of a page (regions, text lines, words, glyphs) and the recognized text content, including coordinates and font information. It is often used in digital libraries.
- **Usability:**
  - Creation/Modification: ALTO XML is typically generated as output by OCR engines (e.g., Tesseract OCR via post-processing, ABBYY FineReader). It can be modified using XML editors, but direct manual creation is complex and rare.
  - Performance/Resource Requirements: XML parsing can be resource-intensive for very large and complex ALTO files. The verbosity of XML means files can be relatively large.
  - Engine/Platform Integration (VE/AR Context): ALTO itself is not directly renderable in VEs. Its value is as a source of structured text content and layout information. This data can be parsed and used to reconstruct a visual representation of the document page (e.g., by placing text elements according to coordinates in a VE) or to enable text search and analysis on digitized documents displayed as images.
- **Interoperability:**
  - Content Creation Software (OCR Software): Supported as an output format by several major OCR software packages and digital library systems.
  - VE/AR Systems & Web Platforms: The raw XML is not directly displayed. Data from ALTO can be used by web applications to provide features like text highlighting over page images, search, or text extraction. In VEs, this data could drive interactive elements associated with virtual documents.
- **Metadata Embedding Capabilities:** ALTO files describe the layout and text of a page. They can contain metadata about the OCR processing (e.g., OCR engine name,

processing date) and basic descriptive metadata about the source document page. For richer bibliographic metadata, ALTO files are often used in conjunction with other metadata standards like METS (Metadata Encoding and Transmission Standard).

- **Suitability for IMPULSE and VE:** Highly suitable for representing the output of OCR processing of digitized documents. It is a key format for making scanned textual heritage accessible and usable computationally. It is not a display format itself, but a data-rich source. This allows for full-text search, text extraction, and potentially recreating the textual layout for display alongside scanned images in a VE.

7.2.7 hOCR (HTML-based OCR format)

- **Short Overview:** hOCR is an open standard format for representing OCR output. It embeds layout information (bounding boxes for pages, paragraphs, lines, words), character confidences, and other OCR-related information directly into an HTML file. This makes the OCRed text immediately viewable in a web browser and accessible to web technologies.
- **Usability:**
    - Creation/Modification: Generated by various OCR engines, including Tesseract OCR. As it's HTML, it can be viewed directly in browsers and manipulated with HTML/XML tools, though direct editing of OCR metadata might be complex.
    - Performance/Resource Requirements: Being HTML, it's generally easy to parse and render in web browsers. File sizes depend on the amount of text and the detail of layout information.
    - Engine/Platform Integration (VE/AR Context): hOCR files can be displayed in web views embedded within VE/AR applications. The structured HTML can be parsed to extract text and layout information, like ALTO, for use in custom UI elements or for enabling text interaction with virtual documents.
- **Interoperability:**
    - Content Creation Software (OCR Software): Supported by several OCR engines, particularly open-source ones.
    - VE/AR Systems & Web Platforms: Highly interoperable with web technologies due to their HTML base. This makes it easier to integrate OCR results into web-based VEs or applications. For non-web VEs, parsing the HTML structure is necessary.
- **Metadata Embedding Capabilities:** hOCR uses HTML meta tags and custom attributes within HTML elements (e.g., class attributes like ocr_page, ocr_line, ocrx_word, and title attributes for bounding box and confidence information) to store metadata about the document structure and OCR process.
- **Suitability for IMPULSE and VE:** Suitable for representing OCR output, especially when web-based display or easy browser previewing of the OCR layer is desired. It provides a good balance of human readability (via browser) and machine readability (parsing HTML for layout and text). Like ALTO, it's a source of data for enhancing

interactions with digitized documents in VEs (e.g., text search, highlighting over page images)

8 Analysis of 3D File Formats for Virtual Environments

The selection of 3D file formats is arguably the most critical for virtual environments, as these formats define the geometry, appearance, structure, and behaviour of the virtual world itself. The landscape of 3D formats reflects an evolution driven by the increasing complexity of 3D assets and the demanding performance requirements of real-time rendering, especially in VR. Early formats like OBJ and STL focused on simple geometry descriptions. As needs grew, richer interchange formats emerged, capable of handling materials, animation, and scene structure, often tied to specific software ecosystems (e.g., FBX, native DCC formats like .blend, .max). However, the rise of web-based 3D and performance-constrained devices like mobile VR headsets spurred the development of modern transmission formats, specifically designed for efficient loading and rendering at runtime. glTF and USD are the leading file formats in this category, emphasizing performance, Physically Based Rendering (PBR) materials, and increasingly, open standards and extensibility. The choice for a project like IMPULSE involves balancing the need for high-fidelity representation of cultural heritage assets, interoperability across a potentially diverse toolchain, and the performance demands of immersive, real-time experiences.

8.1 glTF/GLB (GL Transmission Format)

- **Overview:** glTF (GL Transmission Format) is a royalty-free, open standard specification developed and maintained by the Khronos Group, explicitly designed for the efficient transmission and loading of 3D scenes and models in applications, particularly web-based and real-time environments. Often dubbed the "JPEG of 3D," it aims to minimize both the file size of 3D assets, and the runtime processing required to unpack and render them. glTF 2.0 is the current version, supporting PBR (Physically Based Rendering) materials, geometry, scene graph hierarchy, animation and skins. It defines an extensible architecture allowing for additional features via extensions. glTF files can be represented as JSON (.glTF) with separate binary data (.bin) and textures or bundled into a single binary file (.glb) for easier distribution.
- **Usability:**
    - *Creation/Modification:* glTF is primarily an export target format. Exporters are available for major DCC software including Blender (native import/export), Autodesk Maya, and 3ds Max. Various converters exist to translate from other formats like FBX or OBJ. While direct authoring in glTF is less common than in native DCC formats, the workflow involves preparing assets in standard tools and exporting to glTF for delivery. Tools for validation and preview are readily available.
    - *Performance/Resource Requirements:* Performance is a core design goal of glTF. The format structure is optimized for fast parsing and direct loading into GPU APIs like WebGL. Binary (.glb) format and mesh compression significantly reduce file sizes and loading times. PBR material definitions align well with

modern real-time rendering pipelines. It is well-suited for performance-sensitive platforms like mobile VR and web browsers (WebXR). While highly efficient, very complex models (high polygon counts, large textures) can still pose performance challenges, especially in mobile AR/VR contexts.
  - *Engine/Platform Integration:* Excellent support in web-based 3D engines. Strong and growing support in major game engines like Unity and Unreal Engine, often recommended as a preferred import format alongside or as an alternative to FBX. Godot Engine uses glTF as its primary 3D interchange format. Integration is generally straightforward for core features. Native WebXR support is a significant advantage for web-based VR/AR applications.
- **Interoperability:**
  - *Content Creation Software:* As an open standard, glTF promotes interoperability. Support is widespread and growing across DCC tools, converters, and platforms like Sketchfab. It serves as common ground for exchanging ready-to-render assets.
  - *VE/AR Systems & Web Platforms:* glTF is the leading standard for 3D asset delivery on the web (WebGL, WebXR). It's the format of choice for ARCore (Android's Scene Viewer) and is used as a basis for other standards like 3D Tiles (for streaming large geospatial datasets) and VRM (for VR avatars).
- **Metadata Embedding Capabilities:**
  - The core glTF specification supports basic metadata within the asset object (e.g., copyright, generator) and allows arbitrary JSON data in extras properties on most objects. More structured and powerful metadata embedding is enabled through extensions. Key extensions include EXT_mesh_features and EXT_structural_metadata, which allow associating rich, structured metadata (e.g., properties like material type, historical significance, database IDs) with specific parts of a mesh (features) or even individual vertices/texels. This allows for semantic queries and styling within applications. Integration with external schemas like Schema.org is also possible. Accessing and utilizing this metadata depends on application/engine support for the specific extensions.
- **Suitability for IMPULSE and Virtual Environments:**
  - glTF/GLB is highly suitable and strongly recommended as a primary delivery format for 3D cultural heritage assets within the IMPULSE project's virtual environments. Its focus on runtime efficiency, performance, and web compatibility aligns perfectly with the goals of accessibility across diverse platforms, including WebXR.
  - The open standard approach promotes interoperability and longevity, crucial for collaborative projects and archival considerations. PBR material support enables high-fidelity visual representation.
  - The extensible metadata system, particularly via extensions like EXT_structural_metadata, offers a powerful mechanism for embedding rich semantic information directly with the 3D models, enabling interactive exploration and contextual information display within the VE. This capability is particularly valuable for cultural heritage applications. Its position as the "JPEG of 3D" signals its importance as the modern standard for getting 3D assets efficiently into runtime applications.

8.2 OBJ (Wavefront Object)

- **Overview:** The OBJ file format, originating from Wavefront Technologies' Advanced Visualizer software, is a simple, text-based format primarily used to define the surface geometry of 3D models. It stores vertex positions, texture coordinates (UVs), vertex normals, and face definitions (lists of vertices forming polygons). While primarily geometric, it can reference external Material Template Library (.mtl) files to describe basic surface shading properties. OBJ is widely regarded as a vendor-neutral format for basic 3D mesh exchange.
- **Usability:**
  - *Creation/Modification:* OBJ files can be exported from nearly all 3D modelling, sculpting, and CAD applications, making it a ubiquitous interchange option. It's simple, human-readable text structure makes it relatively easy to parse or even manually edit for simple cases.
  - *Performance/Resource Requirements:* Being text-based, OBJ files can be larger and slower to parse than equivalent binary formats, especially complex models. Its major limitation is its feature set: it lacks native support for skeletal animation, morph targets, complex scene hierarchies, physically based materials (PBR), lights, or cameras. Performance in a VE context is therefore limited to static geometry. Rendering performance depends solely on the complexity (polygon count) of the imported mesh.
  - *Engine/Platform Integration:* Widely supported for importing static meshes in game engines like Unity and Unreal Engine. Integration is straightforward due to the format's simplicity, but it only brings in the basic geometry and potentially simple materials (if MTL is supported by the importer).
- **Interoperability:**
  - *Content Creation Software:* Excellent interoperability for exchanging basic, static 3D mesh data between different software packages. However, the interpretation and support for associated MTL files can be inconsistent across applications, sometimes leading to loss of material information.
  - *VE/AR Systems & Web Platforms:* OBJ files can be loaded and displayed in web environments using libraries like Three.js or Babylon.js, but they are generally less efficient for web delivery than formats like glTF. OBJ is also commonly used as an input format for 3D printing workflows.
- **Metadata Embedding Capabilities:**
  - The OBJ format itself has extremely limited metadata capabilities. It primarily stores geometric data and references to external MTL files. The MTL format allows defining basic material properties (colours, simple texture maps) but lacks a standardized way to embed rich metadata like authorship, licensing, or semantic information. Conventions exist for embedding vertex color data directly in the OBJ file, bypassing MTL, but this is not universally supported. Overall, OBJ is unsuitable for carrying complex metadata.
- **Suitability for IMPULSE and Virtual Environments:**
  - OBJ can serve as a basic interchange format for simple, static 3D models of cultural heritage objects, particularly when broad compatibility with a wide

range of modelling or processing tools is required early in a pipeline. Its simplicity ensures that the core geometry can usually be transferred reliably.
    - However, its significant limitations – lack of support for animation, PBR materials, scene hierarchy, and meaningful metadata – make it insufficient as a primary format for creating rich, interactive, and informative virtual environments. Assets would need to be significantly enhanced in other software or formats (like glTF, USD, or FBX) before being suitable for the final VE application. Its role is best suited for simple geometry transfer or as input to processes like 3D printing.

8.3 USD (Universal Scene Description)

- **Overview:** USD, or OpenUSD, is a powerful, open-source framework and ecosystem of file formats developed by Pixar Animation Studios. It goes beyond being just a file format; it's designed as a comprehensive system for describing, composing, simulating, and collaborating on complex 3D scenes. Key strengths include its non-destructive editing workflow based on layering and composition arcs (allowing multiple artists or data sources to contribute to a scene without overwriting each other's work), its scalability for handling extremely large datasets, and its focus on interoperability. USD supports a rich set of features including complex geometry (meshes, curves, points, subdivision surfaces), physically based materials and shading networks (UsdShade), lighting (UsdLux), skeletal and blend-shape animation, physics (UsdPhysics), and hierarchical scene structure. It utilizes several file formats: .usda (human-readable ASCII), .usdc (binary), .usd (can be either), or .usdz (a zero-compression zip archive designed for delivery, notably used by Apple for AR). The Alliance for OpenUSD (AOUSD), including Pixar, Adobe, Apple, Autodesk, and NVIDIA, promotes its standardization and development.
- **Usability:**
    - *Creation/Modification:* Support for USD is rapidly growing across the DCC industry. Native or plugin-based support exists in tools like Maya, Houdini, Blender, 3ds Max, Cinema 4D, and various CAD applications. NVIDIA's Omniverse platform is built foundationally on OpenUSD, providing extensive tools for USD-based workflows and collaboration. The layering system enables powerful collaborative and non-destructive workflows but also introduces a level of complexity that requires understanding of USD's composition principles.
    - *Performance/Resource Requirements:* USD is designed with performance and scalability in mind, particularly for handling the massive datasets encountered in film production and complex simulations. Its architecture allows for efficient data streaming and lazy loading. Real-time rendering performance is a key focus, especially within integrated platforms like Omniverse and through engine plugins. While both USD and glTF aim for real-time efficiency, USD's strength lies more in composing and managing complex scenes, whereas glTF is more focused on streamlined delivery of individual assets. Performance in specific VR/AR applications depends heavily on the complexity of the composed scene and the efficiency of the runtime implementation.

    - *Engine/Platform Integration:* Integration into major game engines is advancing rapidly. Unreal Engine offers robust USD import capabilities, allowing developers to work with USD stages, layers, and variants directly within the editor, leveraging it for complex pipelines. Unity also provides official USD packages for similar integration. This allows engines to benefit from USD's strengths in scene assembly and data management.
- **Interoperability:**
    - *Content Creation Software:* A primary goal of USD and the AOUSD is to establish USD as a standard for 3D interoperability across different tools and pipelines. Its adoption by major software vendors is driving this forward. It aims to provide a common language for describing and composing 3D data from various sources.
    - *VE/AR Systems & Web Platforms:* The .usdz variant is the standard format for AR Quick Look on Apple iOS devices, giving USD significant reach in mobile AR. Direct web platform support (like WebGL/WebXR) is less inherent than glTF's, as USD is more of a scene description framework. However, platforms like Omniverse enable streaming and web viewing of USD content, and conversion from USD to glTF is possible for web delivery pipelines.
- **Metadata Embedding Capabilities:**
    - USD possesses a highly sophisticated and extensible metadata system. Metadata can be attached to any primitive ('Prim' - the fundamental building block in USD, representing objects like meshes, lights, cameras) or property (attributes, relationships) within the scene hierarchy. Metadata is strongly typed, can be defined in custom schemas, and participates in the composition process (meaning metadata can also be layered and overridden). USD defines core schemas for common concepts (geometry, shading, physics, etc.) and allows users to define their own application-specific schemas. This makes USD exceptionally well-suited for encoding rich semantic information, provenance, technical parameters, and complex relationships within a 3D scene.
- **Suitability for IMPULSE and Virtual Environments:**
    - USD presents a very compelling option for the IMPULSE project, particularly for managing complex cultural heritage scenes, facilitating collaboration among partners using different software, and embedding rich, structured metadata. Its layering system is ideal for non-destructively integrating data from various sources (e.g., scanned geometry, reconstructed elements, annotations).
    - The open standard nature and strong industry backing ensure growing support and longevity. Improving game engine integration makes it viable for high-fidelity VE development pipelines. Its powerful metadata capabilities are a significant advantage for documenting cultural heritage assets within their context.
    - It could serve as both a primary interchange format within the production pipeline and potentially as the runtime scene description format if using compatible engines or platforms. The USDZ variant is essential if targeting AR experiences on Apple devices. While potentially more complex to implement

initially than simpler formats, its capabilities for handling complexity and collaboration align well with the ambitious goals of projects like IMPULSE.

8.4 PLY (Polygon File Format / Stanford Triangle Format)

- **Overview:** The PLY format, also known as the Stanford Triangle Format, was developed at Stanford University specifically for storing data originating from 3D scanners. It is designed to be simple yet flexible, capable of representing both polygonal meshes (defined by vertices and faces) and point clouds. A key feature is its ability to store arbitrary properties associated with elements (vertices, faces, edges), such as colour (RGB), surface normals, texture coordinates, transparency, and even custom data like sensor readings. PLY files have a header section that defines the elements and their properties, followed by the data section, which can be stored in either human-readable ASCII or a more compact binary format.
- **Usability:**
    - *Creation/Modification:* PLY is a common output format for 3D scanning hardware and software. It's also supported for import/export by 3D processing software like MeshLab and modelling tools like Blender. The format's simplicity facilitates parsing and generation.
    - *Performance/Resource Requirements:* The binary version of PLY is relatively compact and efficient for storing large point clouds or meshes resulting from scans. Performance in applications depends on the density of the geometry (number of points or polygons). It's well-suited for representing static scanned data. Direct support for game engines is limited.
    - *Engine/Platform Integration:* PLY is not typically a natively supported format for direct import into game engines like Unity or Unreal Engine in the same way as FBX, glTF, or OBJ. Importing PLY data, especially point clouds, often requires specialized plugins or asset store packages designed for handling point cloud data or involves converting the PLY file to a standard mesh format (like OBJ or FBX) using external tools before engine import.
- **Interoperability:**
    - *Content Creation Software:* Good interoperability within the 3D scanning and processing ecosystem. Tools like MeshLab heavily utilize PLY. Support in general-purpose modelling software is present (e.g., Blender) but might be less comprehensive than for OBJ or FBX. It's a common format in academic research involving 3D geometry.
    - *VE/AR Systems & Web Platforms:* Not a standard format for web delivery. Its use in VE/AR systems is typically indirect, after conversion or through specialized viewers designed for point cloud or scan data visualization.
- **Metadata Embedding Capabilities:**
    - PLY's metadata capability lies in its extensible header and the ability to define custom properties per element. The header contains metadata about the file structure (elements, properties, format type). Beyond standard properties like position (x, y, z), colour (r, g, b), and normals (nx, ny, nz), users can define additional properties in the header (and include corresponding data for each vertex or face). This allows embedding scan-specific information or other

relevant data directly with the geometry points. While flexible, it lacks standardized schemas and broader semantic capabilities of formats like USD or metadata extensions in glTF. Unity's internal mesh data structure supports common per-vertex attributes like position, normal, colour, and multiple UV sets, which PLY can represent.

- **Suitability for IMPULSE and Virtual Environments:**
    - PLY is highly relevant specifically at the data acquisition stage. It is an excellent format for storing the raw output of scanners, including point clouds or meshes with per-vertex colour information captured during the scan.
    - Its ability to store custom properties allows potentially embedding scanner settings or confidence values directly within the file.
    - However, due to limited direct support in game engines and web platforms, PLY files will typically serve as an intermediate format in the pipeline. The data would need to be processed (e.g., cleaned, decimated, retopologized) and then converted into a format suitable for the target VE platform (like glTF, FBX, or OBJ) before integration.

8.5 FBX (Filmbox)

- **Overview:** FBX is a proprietary 3D file format, originally developed by Kaydara (acquired by Autodesk in 2006), that has become a de facto standard for transferring 3D assets between various digital content creation (DCC) applications and real-time game engines. It supports a rich set of features, including complex polygonal geometry, scene hierarchy, materials and textures, lighting, cameras, skeletal animation (rigging and skinning), blend shapes (morph targets), and custom metadata. FBX files can be saved in either binary (more common and compact) or ASCII format.
- **Usability:**
    - *Creation/Modification:* FBX is widely supported for export from almost all major 3D modelling and animation software, including Autodesk's Maya and 3ds Max, as well as Blender, Cinema 4D, Houdini, and others. It serves as a primary pipeline format for moving complex, animated assets into game development workflows.
    - *Performance/Resource Requirements:* The binary FBX format is relatively efficient for storing complex data. Performance within a game engine largely depends on the complexity of the asset itself (polygon count, animation complexity, material complexity) and the efficiency of the engine's FBX importer. While widely used, it may not be as optimized for pure loading speed and runtime performance as formats specifically designed for transmission like glTF. It remains a workhorse for complex character animations and detailed assets in many high-fidelity VR/AR projects developed with Unity or Unreal.

  - *Engine/Platform Integration:* FBX enjoys excellent and mature support in both Unity and Unreal Engine, often considered the most robust pipeline for importing complex animated characters and scenes. Engine importers typically handle geometry, materials (often requiring adjustments), textures, skeletal rigs, animations, and morph targets. Advanced engine features like Unreal's Datasmith and Interchange Framework are often built to work with or improve upon FBX import workflows, especially for data coming from CAD or specific DCC tools.
- **Interoperability:**
  - *Content Creation Software:* High degree of interoperability between major DCC applications and game engines makes it a standard choice for studio pipelines. However, being a complex proprietary format, inconsistencies or limitations can arise when transferring data between different software versions or vendors. Certain features might not translate perfectly. Users sometimes encounter issues requiring specific export settings or workarounds.
  - *VE/AR Systems & Web Platforms:* FBX is not designed for direct web delivery; assets must be converted to web-friendly formats like glTF. It is heavily used in the *development* pipeline for VR/AR applications built on engines like Unity and Unreal.
- **Metadata Embedding Capabilities:**
  - FBX supports the embedding of custom user-defined properties, often referred to as metadata. In Autodesk applications like Maya and 3ds Max, users can add custom attributes or use the "User Defined" properties tab, and this data can be exported within the FBX file. Game engines like Unreal Engine can import this metadata (specifically FbxProperty values) and make it accessible through scripting interfaces, typically prefixed with FBX. Metadata can be associated with scene objects or even individual bones within a skeleton. Houdini also supports reading and writing fbx_custom_attributes. This capability allows for pipeline automation, asset tagging, and embedding application-specific information. Unity's direct support for importing custom FBX properties might be less explicit or require specific scripting.
- **Suitability for IMPULSE and Virtual Environments:**
  - FBX is a strong candidate as an interchange format within the IMPULSE production pipeline, especially for transferring complex, animated 3D assets (like animated characters, objects with moving parts, or pre-visualized sequences) from various DCC tools into the primary development engines (Unity/Unreal). Its robust support for animation and widespread adoption in industry tools are major advantages.
  - Its proprietary nature is a potential drawback compared to open standards like glTF or USD, particularly concerning long-term archival and potential licensing implications (though currently use is generally free via SDKs). Potential interoperability issues between software versions should also be considered.
  - It is not suitable for direct web delivery and would need conversion to glTF for that purpose. Its metadata capabilities are useful for pipeline integration but may be less standardized or powerful than USD's system. FBX often serves as the bridge between detailed authoring and real-time engine integration.

8.6 VRML (Virtual Reality Modelling Language)

- **Overview:** VRML was an early file format standard designed for representing interactive 3D vector graphics, particularly intended for use on the World Wide Web in the mid-to-late 1990s. It is a text-based format that defines geometry, appearances (materials, textures), lighting, viewpoints, and basic forms of animation and user interaction (through sensors and script nodes, often using Java or JavaScript). VRML 1.0 and VRML97 (VRML 2.0) were the major versions. It has largely been superseded by its successor, X3D.
- **Usability:**
  - *Creation/Modification:* Authoring tools for VRML are mostly legacy software. While some modern 3D applications might retain VRML export capabilities (often alongside X3D export), it's not a common format in current workflows. Files can be edited in text editors due to their structure.
  - *Performance/Resource Requirements:* VRML implementations often suffer from performance limitations, especially when handling large or complex scenes. The text-based nature could lead to slow parsing, and rendering performance was heavily dependent on the specific browser plugin and available hardware of the era. It lacks many optimizations found in modern real-time formats and rendering techniques.
  - *Engine/Platform Integration:* VRML is not natively supported by modern game engines like Unity or Unreal Engine. It is considered legacy technology in the context of current VE development platforms.
- **Interoperability:**
  - *Content Creation Software:* Support is very limited to modern DCC tools. Conversion utilities might exist to translate VRML to formats like X3D or OBJ.
  - *VE/AR Systems & Web Platforms:* Requires specific browser plugins (like Cortona3D Viewer) to view VRML content on the web. These plugins are largely obsolete and not supported by modern browsers, which rely on technologies like WebGL. VRML is not suitable for delivery to modern VR/AR hardware or web platforms.
- **Metadata Embedding Capabilities:**
  - VRML included basic mechanisms for scene information, such as the WorldInfo node which could contain title and info string fields (example shown for X3D, but concept originated in VRML). It allowed DEF names to identify nodes. However, it lacked standardized, extensible metadata frameworks like XMP or detailed schema capabilities found in later formats like X3D or USD. Representing complex metadata, such as detailed provenance or semantic information for cultural heritage, was not a primary design goal and is poorly supported.
- **Suitability for IMPULSE and Virtual Environments:**
  - VRML is entirely unsuitable for use in the IMPULSE project for developing or delivering modern virtual environments. Its performance limitations, lack of support in current engines and browsers, obsolete web integration technology, and limited features set compared to modern standards like glTF, X3D, or USD make it historical format. Its only potential relevance might

be if the project encounters legacy 3D cultural heritage data already stored in VRML format, which would then require conversion to a modern format.

8.7 X3D (Extensible 3D)

- **Overview:** X3D is an ISO/IEC standard, royalty-free file format and runtime architecture for representing interactive 3D graphics, designed as the successor to VRML. Managed by the Web3D Consortium, it aims to provide a robust and extensible framework for 3D content on the web and other platforms. X3D supports multiple file encodings: an XML-based syntax (.x3d), a syntax compatible with VRML97 called ClassicVRML (.x3dv), a Compressed Binary Encoding (.x3db), and a draft JSON encoding. It features a componentized architecture, allowing implementations to support different feature sets defined by profiles (e.g., Core, Interchange, Interactive, Immersive, Full, CADInterchange, Medical). X3D boasts a rich feature set, including advanced geometry (NURBS, subdivision surfaces), physically based rendering (PBR) materials (added in later versions to align with glTF 2.0 capabilities), complex animations, scripting, custom shaders, Humanoid Animation (HAnim), geospatial components, and strong metadata support.
- **Usability:**
    - *Creation/Modification:* X3D files can be exported from a variety of 3D modelling and visualization tools, including Blender, Rhino, Mathematica, and CAD software via Open Cascade. Conversion tools are available to translate from other formats like OBJ, glTF, STL, and SVG. The XML and ClassicVRML encodings are human-readable and can be edited with text editors. Specialized editors and viewers also exist within the Web3D ecosystem.
    - *Performance/Resource Requirements:* Performance varies depending on the chosen encoding (binary is more compact), the complexity of the scene, the features used (profile), and the efficiency of the rendering engine or browser implementation. Open-source JavaScript libraries like X3DOM and X_ITE enable plugin-free rendering of X3D scenes within standard web browsers using WebGL. While capable, performance for highly complex, game-like scenarios might not match native game engines or formats hyper-optimized for GPU pipelines like glTF, but it is suitable for many interactive web visualizations, simulations, and educational applications.
    - *Engine/Platform Integration:* X3D is not a primary format directly integrated into mainstream game engines like Unity or Unreal Engine. Integration would typically require using specific libraries, custom importers, or converting X3D assets to engine-native formats (like FBX or glTF). The Castle Game Engine is noted for using X3D. X3D's focus is stronger on web standards and standalone applications/viewers rather than deep integration with dominant commercial game engines.
- **Interoperability:**
    - *Content Creation Software:* Support exists for import/export in a range of tools, particularly those focused on visualization, simulation, CAD, and open standards. It may be less common in pipelines heavily reliant on Autodesk products compared to FBX, or in web pipelines compared to glTF.

- *VE/AR Systems & Web Platforms:* X3D has a strong emphasis on web integration through HTML5 and the DOM, facilitated by libraries like X3DOM and X_ITE. This makes it a viable option for creating complex, interactive 3D experiences directly within web browsers, including potential WebXR applications.

- **Metadata Embedding Capabilities:**
  - X3D provides robust and flexible mechanisms for embedding metadata. Metadata can be associated with the entire scene using <meta> tags within the <head> section (like HTML, but these are typically not persistent after loading). More importantly, X3D defines specific metadata node types (MetadataBoolean, MetadataDouble, MetadataFloat, MetadataInteger, MetadataString, MetadataSet) that can be attached as children to almost any node in the scene graph. These nodes store typed arrays of values, have name attributes for identification, and an optional reference attribute to link to external definitions or schemas (e.g., Dublin Core, FOAF, or custom ontologies). The WorldInfo node also provides basic title and info string fields. This structured, typed, and referenceable metadata system is excellent for embedding rich semantic information directly within the 3D scene structure.
- **Suitability for IMPULSE and Virtual Environments:**
  - X3D offers several advantages for the IMPULSE project, particularly if web-based delivery of interactive cultural heritage experiences is a primary goal. Its strengths include:
    - Open Standard (ISO/IEC): Ensures longevity and avoids vendor lock-in.
    - Rich Feature Set: Supports diverse geometry types, PBR, animation, scripting, and specialized components like Geospatial and HAnim that could be relevant to certain cultural heritage contexts.
    - Strong Metadata Support: The dedicated metadata nodes allow for deep semantic annotation of scene elements, ideal for educational and informational purposes.
    - Web Integration: Mature solutions (X3DOM, X_ITE) enable direct embedding and interaction within web pages without plugins.
  - However, its **weaker integration with mainstream game engines** (Unity/Unreal) compared to FBX, glTF, or USD makes it less suitable if the primary development platform is one of those engines. While technically capable, it faces strong competition from the simpler, widely adopted glTF format for many web delivery use cases. X3D is best suited for projects prioritizing web standards, deep metadata integration, and potentially leveraging its specialized components.

8.8 DAE (Collada - Collaborative Design Activity)

- **Overview:** COLLADA (Collaborative Design Activity) is an XML-based schema and file format (.dae) designed as an open standard interchange format for 3D digital assets. Managed by the Khronos Group (the same body overseeing glTF, OpenGL, Vulkan, etc.), its goal is to facilitate the transfer of assets between different DCC applications (like Maya, 3ds Max, Blender) while preserving information such as geometry, materials, textures, animations (including skinning and morphs), physics, and kinematics. It is intended primarily as an intermediate format within a production pipeline, rather than an optimized format for final delivery or runtime loading.

- **Usability:**
    - *Creation/Modification:* COLLADA (.dae) export is supported by many major 3D DCC packages, either natively or through plugins. Its XML structure means files can be inspected or potentially modified using text editors, which can be useful for debugging or custom processing.
    - *Performance/Resource Requirements:* Being an XML-based format, COLLADA files can be verbose and significantly larger than equivalent binary formats like FBX or GLB. Parsing XML is generally slower and more memory-intensive than parsing optimized binary data structures. Consequently, COLLADA is not well-suited for direct loading and rendering in performance-critical real-time applications or VEs. Its role is in the offline interchange of assets between tools. Engine import times for DAE files might be slower than for FBX or glTF. Unreal Engine developers historically favoured improving FBX reliability over adding native DAE support due to potential complexities and edge cases.
    - *Engine/Platform Integration:* Engine support for COLLADA varies. Unity provides some level of support for importing.dae files. Unreal Engine has historically lacked robust native support, with the recommendation often being to convert DAE files to FBX using external tools (like Autodesk's FBX Converter) before importing into Unreal. The focus of engine developers has generally been on optimizing pipelines for FBX and, more recently, glTF and USD.
- **Interoperability:**
    - *Content Creation Software:* COLLADA was designed specifically to improve interoperability between different DCC tools, aiming to provide a more reliable transfer of complex asset data than older formats like OBJ or 3DS. It can be effective for moving assets, including animations and materials, between software packages that might have poor native compatibility. However, the complexity of the format and potential differences in implementation across tools can still lead to interoperability issues or loss of data in some cases.
    - *VE/AR Systems & Web Platforms:* COLLADA is not suitable for direct use in web browsers or as an optimized runtime format for VE/AR systems. Assets must be converted to formats like glTF for web delivery or optimized engine formats.
- **Metadata Embedding Capabilities:**
    - The XML structure of COLLADA allows for the inclusion of metadata. The schema defines elements for asset information, such as contributor details (author, comments, creation dates), units, and coordinate system orientation (<asset>). It also allows for application-specific data to be embedded using <extra> tags within various elements throughout the file structure. This provides a mechanism for storing custom data or metadata relevant to specific tools or pipelines, although standardization relies on convention rather than predefined metadata schemas like XMP within the core specification.

- **Suitability for IMPULSE and Virtual Environments:**
    - COLLADA could potentially serve as an open interchange format within the IMPULSE project's pipeline, particularly if there's a need to transfer complex assets between different DCC tools where FBX might prove problematic, or if an open, text-based intermediate format is strongly preferred for archival or processing reasons.
    - However, it is not suitable as a final delivery format for the virtual environment itself due to performance limitations (large file size, slow parsing) and less robust game engine support compared to FBX, glTF, or USD.
    - Its role would be strictly as an intermediate step, requiring conversion to an optimized runtime format before use in the VE. Given the strong support for FBX and the rise of glTF/USD, the necessity for using COLLADA in a modern pipeline might be limited unless specific tool compatibility issues arise.

8.9 STL (Stereolithography / Standard Triangle Language)

- **Overview:** STL is one of the oldest 3D file formats, originally developed for stereolithography-based 3D printing systems. Its purpose is solely to describe the surface geometry of a 3D object as a collection of unordered, interconnected triangles (a "triangle soup" or tessellation). Each triangle is defined by the 3D coordinates of its three vertices and a surface normal vector indicating the outward direction. STL exists in two forms: ASCII (text-based, human-readable but verbose) and binary (more compact and much more common in practice).
- **Usability:**
    - *Creation/Modification:* STL files can be exported from virtually all CAD packages and 3D modelling software. The simplicity of the format makes exports straightforward. Direct modification usually involves mesh editing tools.
    - *Performance/Resource Requirements:* The format only contains raw triangle geometry. Performance in a rendering context depends entirely on the number of triangles in the mesh. File size is determined by the triangle count, and whether ASCII or binary encoding is used (binary is significantly smaller). STL lacks any concept of materials, textures, colours, levels of detail (LODs), instancing, or other optimization structures crucial for efficient rendering in complex VEs.
    - *Engine/Platform Integration:* Game engines like Unity and Unreal Engine can typically import STL files as static geometry. However, the lack of UV coordinates, material information, or animation support severely limits its utility within an engine. Imported STL models would require significant additional work within the engine (UV unwrapping, material assignment) to be visually useful.
- **Interoperability:**
    - *Content Creation Software:* Excellent interoperability as an export format from CAD and 3D modelling tools, primarily for outputting geometry for 3D printing.
    - *VE/AR Systems & Web Platforms:* STL is the de facto standard input format for most 3D printers and slicing software. It is not suitable for web delivery

or for creating visually rich VE/AR experiences due to its lack of appearance information.

- **Metadata Embedding Capabilities:**
    - The STL file format has **no capability** to store metadata. It does not support information about colour, texture, materials, authorship, copyright, or any other attributes beyond the raw surface geometry defined by triangles.
- **Suitability for IMPULSE and Virtual Environments:**
    - STL's primary and essentially only relevance to the IMPULSE project would be if 3D printing of physical replicas of cultural heritage objects is a requirement. It is the standard format for this purpose.
    - It is completely unsuitable for creating the digital assets (visually rich, potentially interactive 3D models) needed for the virtual environment itself, due to its fundamental lack of support for colour, textures, materials, animation, and metadata. Using STL data as a starting point for a VE asset would require extensive processing and data addition in other software. Its simplicity, while beneficial for additive manufacturing, makes it inadequate for the complexities of modern virtual environments.

8.10 Alembic (.abc)

- **Overview:** Alembic is an open-source computer graphics interchange file format, developed collaboratively by Sony Pictures Imageworks and Industrial Light & Magic. Its primary purpose is to efficiently store and exchange complex, animated geometry by "baking" the results of simulations or procedural animation into a sequence of sampled geometric data over time. It focuses on capturing the final vertex positions, topology changes, transforms, and other geometric attributes, rather than the underlying rigs or procedural setups that generated them. This makes it ideal for transferring data like fluid simulations, cloth dynamics, fracturing objects, or complex character deformations between different software packages or into rendering/game engines. Alembic supports polygon meshes, subdivision surfaces, NURBS curves/patches, particles, transform hierarchies, and cameras, with initial support for materials and lights added later.
- **Usability:**
    - *Creation/Modification:* Alembic files are typically exported from high-end DCC and VFX software where complex simulations or animations are generated, such as Maya, Houdini, 3ds Max, Blender, Cinema 4D, RealFlow, etc. It's an output format representing the computed results, not usually a format for direct interactive modelling.
    - *Performance/Resource Requirements:* Alembic is designed for efficient storage and streaming of potentially very large and dense animated geometry caches. Performance in a real-time engine depends on factors like the complexity of the geometry per frame, the length of the animation, the efficiency of the engine's Alembic streaming implementation, and available memory/disk bandwidth. While it enables the use of extremely complex animations not achievable with traditional methods, playing back dense, rapidly changing vertex caches can be computationally expensive compared to standard

skeletal animation. File sizes can become very large for long or high-resolution caches.
    - *Engine/Platform Integration:* Both Unity (via an official package) and Unreal Engine (native importer) support importing Alembic files. This allows developers to bring in complex, pre-baked vertex animations for things like detailed facial performances, cloth simulations, or environmental effects that go beyond the capabilities of standard skeletal animation systems. Integration involves managing the playback and streaming of the geometry cache asset within the engine. Careful performance profiling is often required.
- **Interoperability:**
    - *Content Creation Software:* Alembic has become widely adopted as a standard interchange format within the visual effects and animation industries, ensuring good compatibility for transferring baked geometry between major software packages.
    - *VE/AR Systems & Web Platforms:* Alembic is not designed for direct web delivery. Its use in VE/AR is primarily through import into game engines for high-fidelity applications requiring complex, non-skeletal animations.
- **Metadata Embedding Capabilities:**
    - Alembic supports the storage of arbitrary user-defined properties (metadata) associated with objects within the Alembic hierarchy. Tools like Houdini and Blender provide mechanisms to export and import these user properties, often represented as JSON strings stored in attributes. It can also store the hierarchical path of objects as an attribute. While this allows for embedding custom data, Alembic's primary focus is on the geometry cache itself, and it doesn't have the same level of sophisticated, structured metadata schemas and composition features found in formats like USD. The Alembic library mentioned in relates to database migrations using SQLAlchemy and is distinct from the graphics file format.
- **Suitability for IMPULSE and Virtual Environments:**
    - Alembic could be valuable for the IMPULSE project in specific scenarios requiring the import of highly complex, pre-computed animations that cannot be adequately represented using standard skeletal animation techniques. Examples relevant to cultural heritage might include:
        - Accurate cloth simulation for historical garments on animated characters.
        - Complex destruction or transformation sequences for visualizing historical events or processes.
        - High-fidelity facial animation captured from performance capture systems.
    - It serves as a specialized tool for these demanding cases, complementing standard animation workflows using FBX or glTF. It is **not** a general-purpose asset format for static models or standard character rigging. Performance implications and potentially large file sizes need careful consideration during production.

8.11 STEP/STP (Standard for the Exchange of Product model data - ISO 10303)

- **Overview:** STEP (Standard for the Exchange of Product model data), formally ISO 10303, is a comprehensive international standard designed for the representation and exchange of product data throughout its lifecycle, particularly within Computer-Aided Design (CAD), Computer-Aided Manufacturing (CAM), and Computer-Aided Engineering (CAE) systems. Unlike mesh-based formats (like STL, OBJ, glTF), STEP typically represents geometry using precise mathematical descriptions (Boundary Representation or B-rep), which define solids and surfaces exactly, rather than approximating them with polygons. The standard is extensive, composed of many parts, including the EXPRESS data modelling language used to define schemas, and various Application Protocols (APs) tailored to specific industries or lifecycle stages (e.g., AP203 for configuration-controlled design, AP214 for automotive, AP242 for managed model-based 3D engineering, which includes Product Manufacturing Information - PMI). Files commonly use the .step or .stp extension.
- **Usability:**
    - *Creation/Modification:* STEP is the primary vendor-neutral export format for nearly all professional mechanical CAD systems (e.g., CATIA, SolidWorks, Siemens NX, Creo, Inventor). It's designed for transferring high-fidelity design data between these systems. Direct modification typically occurs within CAD software.
    - *Performance/Resource Requirements:* STEP files represent precise B-rep geometry, which is not directly usable by the polygon-based rendering pipelines of real-time game engines. To use STEP data in a VE, the geometry must be tessellated – converted into a polygonal mesh. This tessellation process can be computationally intensive, and the quality (accuracy vs. polygon count) needs careful control to balance visual fidelity with real-time performance requirements. STEP files themselves can be large, and the resulting meshes can be extremely dense if not optimized. Direct runtime use is impossible; significant data preparation is required.
    - *Engine/Platform Integration:* Game engines like Unity and Unreal Engine do not natively support importing STEP files. Integrating STEP data requires specialized third-party plugins or middleware (e.g., PiXYZ plugin, Autodesk Platform Services (formerly Forge), CADfix VIZ, SimLab Composer) that handle the import, tessellation, and optimization of the CAD data into formats the engine can understand (typically converting to meshes like FBX or engine-native formats). This CAD-to-VE data preparation pipeline is a critical and often complex part of using engineering data in real-time applications.
- **Interoperability:**
    - *Content Creation Software:* STEP is the gold standard for high-fidelity, vendor-neutral exchange of 3D CAD data between different mechanical CAD, CAM, and CAE systems. It ensures that precise geometric definitions are maintained during transfer.

    - *VE/AR Systems & Web Platforms:* Not suitable for direct use. Requires conversion and optimization for real-time rendering. The focus is on engineering and manufacturing data exchange, not real-time delivery.
- **Metadata Embedding Capabilities:**
    - STEP has powerful capabilities for embedding rich product data beyond just geometry. Application Protocols like AP242 explicitly support Product Manufacturing Information (PMI), which includes geometric dimensions and tolerances (GD&T), 3D annotations, surface finish specifications, material information, and other data critical for manufacturing and inspection. This metadata is structured according to the EXPRESS schemas defined in the standard. Accessing and utilizing this embedded PMI within a VE typically requires specialized tools or plugins capable of parsing the STEP file and translating the PMI into a usable format (e.g., visual annotations, data overlays).
- **Suitability for IMPULSE and Virtual Environments:**
    - STEP is highly relevant if the IMPULSE project involves working with precise CAD models of cultural heritage objects. This could include:
        - Digitizing industrial heritage sites or machinery.
        - Creating accurate digital reconstructions based on engineering drawings.
        - Working with objects where precise dimensions and manufacturing details (potentially represented as PMI) are important.
    - Its strength lies in preserving geometric accuracy and potentially rich technical metadata (PMI).
    - However, using STEP data in a VE necessitates a dedicated data preparation pipeline involving tessellation and optimization tools to convert the precise B-rep geometry into performant polygonal meshes suitable for real-time rendering. This adds complexity to the workflow compared to working directly with mesh-based formats. It is unsuitable if assets are primarily derived from artistic modelling or 3D scanning (where formats like FBX, glTF, or PLY might be more direct).

8.12 Blender (.blend)

- **Overview:** The .blend file is the native project file format for Blender, the popular open-source 3D creation suite. It is a binary format designed to save the entire state of a Blender project, including not just 3D models (meshes, curves, surfaces) but also materials, textures, lighting setups, animation data (keyframes, rigs, constraints), physics simulations, particle systems, compositing node trees, scene settings, Python scripts, and even the user interface layout. Essentially, it's a direct dump of Blender's internal data structures to disk, designed for saving and resuming work within Blender itself. It includes a "Structure DNA" block that describes the data structures used, enabling a degree of forward and backward compatibility between different Blender versions.

- **Usability:**
  - *Creation/Modification:* .blend files are created, edited, and saved exclusively using the Blender application. They provide the most complete way to save and reopen a Blender project with all settings and data intact.
  - *Performance/Resource Requirements:* As a native project format containing potentially vast amounts of diverse data, .blend files can become very large. They are not designed or optimized for direct use as an interchange or runtime delivery format. Loading .blend files in external applications requires parsing Blender's complex and version-specific internal data structures, which can be slow and challenging to implement robustly. Performance in game engines relies entirely on the efficiency of the engine's .blend importer.
  - *Engine/Platform Integration:* Both Unity and Unreal Engine have developed direct .blend file importers. This offers a potentially streamlined workflow for Blender users, bypassing the need for manual export to formats like FBX or glTF. The importer runs Blender in the background to convert the .blend data into engine-compatible assets. However, this process can have limitations: it depends on having Blender installed correctly, might be slower than importing pre-exported files, and may not support all Blender features or data types perfectly. Issues with exporting complex animations or specific features to FBX from Blender are sometimes reported, which might influence the choice between direct import and exporting to FBX/glTF.
- **Interoperability:**
  - *Content Creation Software:* .blend files are primarily intended for use within Blender. While direct importers in Unity and Unreal enhance interoperability with those specific engines, .blend is not a general-purpose interchange format for exchanging data with other DCC tools (like Maya, 3ds Max, Houdini). For broader interoperability, Blender users rely on exporting to standard formats like FBX, glTF, OBJ, DAE, etc.
  - *VE/AR Systems & Web Platforms:* Not suitable for direct delivery. Assets must be exported from Blender (or imported via direct importer) into engine-specific or web-friendly formats (like glTF).
- **Metadata Embedding Capabilities:**
  - Blender provides a powerful and flexible system for embedding custom metadata through its "Custom Properties" feature. Custom properties can be added to almost any data block (objects, meshes, materials, bones, scenes, etc.) and can store various data types, including integers, floats, strings, booleans, and arrays (which can represent vectors or colours). These properties are stored within the .blend file and can be used extensively for rigging setups, driving parameters, storing application-specific data for Python scripts, or adding semantic information to assets. The extent to which this custom metadata is recognized and transferred by the direct .blend importers in Unity or Unreal depends on the specific implementation of those importers. Exporting to formats like glTF (with metadata extensions) or FBX might be necessary to transfer certain custom properties reliably.

- **Suitability for IMPULSE and Virtual Environments:**
    - If Blender is used as a primary authoring tool within the IMPULSE project, the .blend format is essential for saving and managing the source project files.
    - For integrating assets into game engines (Unity/Unreal), developers have the choice between using the direct .blend importer for convenience or exporting to standard interchange formats like FBX or glTF for potentially greater control, reliability, or compatibility, especially with complex assets or specific metadata requirements.
    - .blend files themselves are not suitable for final delivery or archival interchange outside the Blender ecosystem or direct engine integration workflows. The project should define clear export standards (e.g., glTF for delivery, potentially FBX or USD for interchange) regardless of whether direct .blend import is used during development. Blender's custom properties offer a good way to manage metadata during authoring.

8.13 3DS (.3ds)

- **Overview:** The.3ds format is a legacy binary file format originating from Autodesk's early 3D Studio DOS software (precursor to 3ds Max). It was one of the first widely used formats for 3D graphics and became a common, albeit limited, interchange format. It stores basic 3D information, including mesh geometry (triangles only), material colours and basic texture mapping information (using DOS 8.3 filenames), lighting, cameras, and basic object hierarchy and animation keyframes. It uses a chunk-based structure.
- **Usability:**
    - *Creation/Modification:* Export support for.3ds exists in many 3D modelling applications, including 3ds Max and Blender, largely for compatibility with older software or specific pipelines that still rely on it. It is not a format typically used for primary authoring in modern workflows.
    - *Performance/Resource Requirements:* The format suffers from significant limitations imposed by its DOS origins:
        - Meshes must be composed only of triangles.
        - Vertex and polygon count per mesh are limited (often cited as 65,536).
        - Texture filenames are restricted to the 8.3 DOS naming convention.
        - Object, light, and camera names have short character limits (e.g., 10 characters). Material names are also limited (e.g., 16 characters).
        - Accurate vertex normals are not stored; instead, it relies on "smoothing groups" (a bitmask per face) for the receiving application to recalculate normals, which can lead to inconsistencies.
        - Limited support for modern material properties (no PBR) or advanced lighting (e.g., no directional lights). These limitations restrict the complexity and visual fidelity of assets that can be represented. Performance in modern engines is generally not a primary concern as the format's limitations prevent highly complex assets, but the lack of features is the main issue.

  - *Engine/Platform Integration:* Support for importing.3ds files may exist in engines like Unity or Unreal, often as a legacy option. However, due to its severe limitations, it is rarely recommended or used in modern game development workflows compared to FBX, glTF, or even OBJ.
- **Interoperability:**
  - *Content Creation Software:* While export support exists in many tools, its limitations often make it a poor choice for reliable interchange compared to formats like FBX or OBJ. Smoothing group interpretation can vary, and the strict naming/count limits can cause data truncation or errors.
  - *VE/AR Systems & Web Platforms:* Not suitable for modern VE/AR or web delivery due to its limitations and lack of support for essential features.
- **Metadata Embedding Capabilities:**
  - The.3ds format has very rudimentary metadata capabilities. It primarily stores the scene elements themselves (geometry, materials, lights, hierarchy). There are no standardized mechanisms for embedding rich descriptive metadata like EXIF, IPTC, XMP, or custom properties in the way modern formats support. Information is limited to basic object/material names (with strict length constraints) and the inherent structure.
- **Suitability for IMPULSE and Virtual Environments:**
  - The.3ds format is entirely unsuitable for use in the IMPULSE project for creating or exchanging assets for modern virtual environments. Its severe limitations on geometry complexity, naming conventions, texture handling, materials, lighting, and lack of metadata support make it obsolete for representing cultural heritage assets with the required fidelity and richness.
  - Its only potential relevance would be in dealing with very old legacy 3D data that might only exist in this format, which would necessitate immediate conversion to a more capable format like FBX or glTF. It should not be considered for any new asset creation or pipeline development.

8.14 3ds Max (.max)

- **Overview:** The .max file format is the native, proprietary project file format for Autodesk 3ds Max, a widely used professional 3D modelling, animation, and rendering software package. Similar to Blender's .blend files, .max files are designed to save the complete state of a 3ds Max scene, including all geometry (meshes, patches, NURBS), modifiers, materials (including complex shader networks), textures, lighting setups, cameras, animation controllers and keyframes, particle systems, simulation data, scene settings, helper objects, and potentially plugin-specific data. It retains all information specific to 3ds Max, allowing users to save and reload their work within the application without loss of data or functionality.
- **Usability:**
  - *Creation/Modification:* .max files are created, edited, and saved exclusively within Autodesk 3ds Max. They are the standard way to work on projects within the 3ds Max environment.

- *Performance/Resource Requirements:* Like .blend files, .max files are not designed for interchange or direct runtime use. They contain a wealth of data specific to the 3ds Max application and can become very large. Loading .max files directly in external applications is generally not possible or practical due to their proprietary and complex nature. Performance in game engines relies on exporting assets from 3ds Max into a supported interchange format.
- *Engine/Platform Integration:* There is generally **no direct import support** for .max files in game engines like Unity or Unreal Engine. Unlike the situation with .blend files where direct importers have been developed, the standard workflow for getting assets from 3ds Max into engines involves **exporting** to formats like **FBX** (most common and recommended by Autodesk and engine developers), glTF, or potentially OBJ.

- **Interoperability:**
  - *Content Creation Software:* .max files are specific to 3ds Max and cannot be opened directly by other DCC applications like Maya, Blender, or Cinema 4D. Interoperability relies heavily on exporting from 3ds Max to standard interchange formats, primarily FBX. Autodesk provides tools like the FBX Converter to facilitate this.
  - *VE/AR Systems & Web Platforms:* Not suitable for direct delivery. Assets must be exported from 3ds Max to engine-compatible or web-friendly formats.
- **Metadata Embedding Capabilities:**
  - 3ds Max allows users to embed metadata within .max files. This includes standard file properties (title, subject, author, keywords, comments) accessible through the File Properties dialog. Additionally, 3ds Max supports Asset Metadata streams, which can be accessed and modified via MAXScript or external applications, allowing for more structured metadata management related to scene assets. Custom attributes can also be added to objects, which can function as metadata and may potentially be exported via FBX (as seen in).
- **Suitability for IMPULSE and Virtual Environments:**
  - If Autodesk 3ds Max is used as a primary authoring tool within the IMPULSE project, the .max format is essential for saving the source project files.
  - However, .max files cannot be used directly for interchange with other DCC tools or for import into game engines. A crucial part of the workflow will involve exporting assets from 3ds Max into a well-defined standard format, with FBX being the most common and robust choice for transferring complex assets (geometry, materials, animation) to Unity or Unreal. Alternatively, exporting to glTF might be preferred for web delivery or simpler assets.
  - The project needs clear guidelines for exporting assets from 3ds Max to ensure consistency and compatibility with the target VE platform. Metadata created within 3ds Max (File Properties, Asset Metadata, Custom Attributes) needs to be considered during the export process to ensure relevant information is transferred, potentially via FBX custom properties.

9 Analysis of Audiovisual File Formats for Virtual Environments

Audiovisual formats are crucial for incorporating sound effects, ambient audio, background music, narration, and potentially video content within virtual environments. The analysis involves understanding both audio/video codecs (algorithms for compressing/decompressing data, like H.264, AAC, MP3, FLAC) and container formats (file structures that hold the encoded audio/video streams along with metadata and synchronization information, like MP4, MOV, AVI, MKV, WAV). For VE applications, key considerations include compression efficiency (balancing quality with file size/bandwidth), playback performance (low latency decoding), platform compatibility (engines, devices, web), and support for features like spatial audio and metadata embedding.

9.1 4.1 Audio Formats

9.1.1 MP3 (MPEG Audio Layer III)

- **Overview:** MP3 is the most ubiquitous lossy audio compression format, developed by the Moving Picture Experts Group (MPEG). It achieves significant file size reduction compared to uncompressed audio (like WAV) by discarding audio information considered less perceptible to human hearing. The level of compression (and quality) is determined by the bitrate (e.g., 128 kbps, 192 kbps, 320 kbps), with higher bitrates generally yielding better quality but larger files.
- **Usability:**
  - *Creation/Modification:* Easily created from uncompressed sources (like WAV) using numerous widely available encoders (e.g., LAME is a popular open-source encoder). Editing MP3s directly can lead to further quality loss due to re-compression; editing is best done on the original uncompressed source.
  - *Performance/Resource Requirements:* Small file sizes are excellent for storage, streaming, and quick loading, making it suitable for background music or non-critical sound effects in VEs where file size is a concern. Decoding is computationally inexpensive and well-optimized on virtually all platforms. However, the lossy compression means audio fidelity is inherently lower than lossless formats, which might be noticeable for high-quality sound effects or critical audio cues in immersive VR. Looping MP3s seamlessly can sometimes be problematic due to encoder-introduced silence/padding.
  - *Engine/Platform Integration:* Historically, direct MP3 import support in game engines like Unreal Engine was limited due to patent/licensing concerns, although these have largely expired now. Unreal Engine 5 documentation lists MP3 as an importable format, but some users report issues with specific functionalities like the MediaPlayer. Plugins exist to ensure robust runtime MP3 import. Unity generally has better support for MP3 playback. Even when imported, engines often re-compress audio using formats like Ogg Vorbis or platform-specific codecs for the final build.

- **Interoperability:**
  - *Content Creation Software:* Universally supported by audio editing software, media players, and portable devices.
  - *VE/AR Systems & Web Platforms:* Excellent compatibility for web playback (HTML5 audio). Playable on almost all consumer hardware. Engine support can be inconsistent or require plugins for reliable import/playback.
- **Metadata Embedding Capabilities:**
  - MP3 files commonly use ID3 tags (versions ID3v1 and ID3v2) to embed metadata. ID3 tags can store information like title, artist, album, track number, genre, year, comments, and even embedded images (album art). ID3v2 is more flexible and extensible than ID3v1. While primarily used for music organization, these tags can store descriptive metadata relevant to VE assets. XMP metadata can also potentially be embedded within ID3 tags.
- **Suitability for IMPULSE and Virtual Environments:**
  - MP3 can be suitable for background music or non-critical ambient sounds in VEs where file size and loading speed are prioritized over absolute audio fidelity. Its universal compatibility makes it easy to source or distribute audio.
  - However, its lossy nature makes it less ideal for high-quality sound effects, critical audio cues (especially for spatialization), or archival purposes where preserving the original audio quality is important. Potential engine import issues should also be considered. For higher quality compressed audio, AAC or Ogg Vorbis are often preferred. For archival or production, lossless formats like WAV or FLAC are superior.

9.1.2 WAV (Waveform Audio File Format)

- **Overview:** WAV is a standard audio file format developed by Microsoft and IBM, based on the RIFF (Resource Interchange File Format) structure. It is most used to store uncompressed audio data using Pulse Code Modulation (PCM), resulting in high fidelity but large file sizes. Because it typically contains raw, uncompressed audio samples, it serves as a standard format for professional audio recording, editing, and mastering. It can support various bit depths and sample rates, as well as multiple channels (mono, stereo, surround sound).
- **Usability:**
  - *Creation/Modification:* WAV is the default output format for most Digital Audio Workstations (DAWs) and audio recording software. Editing WAV files is straightforward and lossless, as no compression is involved.
  - *Performance/Resource Requirements:* High audio quality is the main advantage. Decoding uncompressed PCM data is computationally trivial, requiring minimal CPU resources. However, the large file sizes lead to longer loading times, higher storage requirements, and significant memory usage (especially if loaded entirely into RAM rather than streamed) compared to compressed formats. This can be a major drawback for VEs, particularly on mobile platforms or web delivery. Streaming WAV files is generally inefficient due to bandwidth requirements.

- *Engine/Platform Integration:* WAV is universally supported for import by game engines like Unity and Unreal Engine. It's often the recommended format for importing high-quality source audio assets (sound effects, voiceovers) into the engine. However, engines almost always compress WAV files during the build process using more efficient codecs (like Ogg Vorbis, Mp3, AAC, or platform-specific formats like ADPCM) to reduce the final application size and improve runtime performance. Using uncompressed WAVs directly at runtime is generally reserved for very short sounds where decompression latency might be critical, but this is rare.

- **Interoperability:**
  - *Content Creation Software:* Excellent compatibility with virtually all audio editing software, DAWs, and media players across Windows, macOS, and Linux. It's a standard professional audio interchange format.
  - *VE/AR Systems & Web Platforms:* While playable in web browsers (HTML5 audio), its large size makes it unsuitable for web streaming. It serves as the high-quality source format in VE development pipelines, but the final delivered audio is typically compressed.
- **Metadata Embedding Capabilities:**
  - As a RIFF-based format, WAV files can contain metadata within specific chunks. The standard includes an optional INFO chunk for storing basic metadata like title, artist, copyright, creation date, etc., although its usage and interpretation can be inconsistent. More relevant for professional audio is the Broadcast Wave Format (BWF) extension, which adds a bext (Broadcast Audio Extension) chunk containing standardized metadata crucial for broadcast and post-production, including timecode references, loudness information, and originator details. WAV files can also embed other metadata chunks, including XMP and ID3 tags, although support depends on the reading application.
- **Suitability for IMPULSE and Virtual Environments:**
  - WAV is the ideal format for capturing, editing, and archiving high-fidelity audio content for the project, such as original recordings, voiceovers, or mastered sound effects, due to its lossless, uncompressed nature. It ensures that maximum quality is preserved during the production process. Using the BWF extension is recommended for professional workflows requiring timecode or detailed provenance metadata.
  - It is generally unsuitable for direct delivery within the final VE application due to large file sizes and performance implications.
  - The standard workflow involves using WAV for source assets and allowing the game engine or a separate tool to compress the audio into an efficient runtime format (e.g., Ogg Vorbis, AAC, MP3) for the final build, balancing quality and performance.

9.1.3 FLAC (Free Lossless Audio Codec)

- **Overview:** FLAC is an open-source audio format that provides lossless compression. Unlike lossy formats (MP3, AAC) that discard data, FLAC compresses audio data without any loss of quality, meaning the decompressed audio is identical to the original uncompressed source. It typically achieves file sizes that are 40-60% smaller than uncompressed WAV files while preserving the full audio fidelity. FLAC supports high-resolution audio, including high bit depths (up to 32-bit) and sample rates (up to 655 kHz), as well as multi-channel audio.
- **Usability:**
    - *Creation/Modification:* FLAC files can be created by encoding from uncompressed sources (like WAV) using various free and commercial tools. Editing is best performed on the uncompressed source or by decoding FLAC to WAV, editing, and re-encoding, although direct FLAC editing support exists in some software.
    - *Performance/Resource Requirements:* Offers identical audio quality to uncompressed formats like WAV but comes with significantly reduced file sizes. Decoding FLAC requires more CPU processing than playing uncompressed PCM (WAV), but it is generally considered efficient and suitable for playback on modern hardware, including mobile devices. Its file size advantage over WAV makes it better for storage and potentially for reducing loading times if decoded efficiently.
    - *Engine/Platform Integration:* Support for FLAC import is available in modern versions of game engines like Unreal Engine. Unity also supports FLAC imports. Like WAV, engines typically re-compress FLAC files into lossy formats for the final build to achieve even smaller sizes, unless uncompressed audio is specifically required. Direct runtime playback of FLAC might be possible via plugins but is less common than using engine-compressed formats.
- **Interoperability:**
    - *Content Creation Software:* Widely supported by audio players, media management software, and many DAWs, especially in the audiophile and open-source communities.
    - *VE/AR Systems & Web Platforms:* Compatibility is good on desktop and mobile platforms, often supported natively or through common media frameworks. Web browser support for direct FLAC playback exists in major browsers (Firefox, Chrome, Edge, Opera) but might be less universal than MP3 or AAC. Not ideal for streaming due to larger file sizes than lossy formats, though better than WAV.
- **Metadata Embedding Capabilities:**
    - FLAC uses Vorbis comments for metadata embedding, the same system used by Ogg Vorbis. Vorbis comments use simple FieldName=Data text tags and are very flexible, allowing for standard tags (like ARTIST, TITLE, ALBUM) as well as custom tags. Multiple tags with the same name are allowed (e.g., multiple ARTIST tags). FLAC also supports embedding cover art and cue sheets in dedicated metadata blocks separate from the Vorbis comments.

- **Suitability for IMPULSE and Virtual Environments:**
  - FLAC is an excellent choice for archiving high-quality audio source material when storage space is a consideration, offering perfect fidelity like WAV but with significantly smaller file sizes. Its open-source nature is also advantageous for long-term preservation and interoperability.
  - It can serve as a high-quality source format for import into game engines, providing the original fidelity for the engine's subsequent compression step.
  - Like WAV, it is generally not the preferred format for final delivery in VEs due to file size compared to lossy formats, unless absolute lossless quality is deemed essential at runtime and the performance impact is acceptable. The typical workflow would involve using FLAC for archival/source and compressing to Ogg/Opus/AAC for delivery.

9.1.4 AAC (Advanced Audio Coding)

- **Overview:** AAC is a lossy audio compression standard developed as a successor to MP3, generally providing better sound quality than MP3 at the same bitrate. It is part of the MPEG-4 standard and is widely used by Apple (iTunes, Apple Music), YouTube, streaming services, and digital radio broadcasters.
- **Usability:**
  - *Creation/Modification:* AAC encoders are widely available in audio software and hardware. Like MP3, editing is best done on the original source to avoid any re-compression losses.
  - *Performance/Resource Requirements:* Offers a good balance between compression efficiency (small file sizes) and audio quality, often perceived as superior to MP3 at comparable bitrates. Decoding is efficient and well-supported by hardware decoders on many devices, especially mobile platforms. Suitable for streaming and playback on devices with limited bandwidth or storage.
  - *Engine/Platform Integration:* AAC is a common format supported by mobile platforms (iOS, Android). Game engines like Unity may use AAC as a compression option, particularly when targeting iOS platforms. Unreal Engine's native support might be less direct, often favouring Ogg Vorbis, but AAC playback is possible through platform media frameworks or plugins. Users have reported using AAC successfully in Unreal Engine 5.4 where MP3 caused issues.
- **Interoperability:**
  - *Content Creation Software:* Good support for audio editing software and media players, particularly within the Apple ecosystem.
  - *VE/AR Systems & Web Platforms:* Widely compatible with modern web browsers (HTML5 audio) and streaming services. Excellent compatibility with iOS and Android devices makes it a strong choice for mobile VE/AR applications.

- **Metadata Embedding Capabilities:**
    - AAC audio data is typically stored within container formats like MP4 (.mp4,.m4a), 3GP, or MPEG Transport Streams (.ts). Metadata is handled in the container format. The MP4 container, commonly used for AAC (.m4a extension for audio-only files), supports embedding metadata using its own atom structure (based on QuickTime) and can also contain XMP metadata. This allows for storing standard tags like title, artist, album, as well as more extensive information.
- **Suitability for IMPULSE and Virtual Environments:**
    - AAC is a very suitable format for delivering compressed audio (background music, ambient sounds, voiceovers) in VEs, especially for applications targeting mobile platforms (iOS/Android) or web streaming, due to its excellent balance of quality, compression efficiency, and broad compatibility.

9.1.5 Ogg Vorbis (.ogg)

- **Overview:** Ogg Vorbis is an open-source, royalty-free, lossy audio compression format. It is part of the Xiph.org multimedia project, which also includes the Ogg container format (of which Vorbis is a component, but often the terms are used interchangeably when referring to audio files). Vorbis was designed as a technically superior alternative to MP3, offering better quality at equivalent bitrates, especially at lower bitrates, and supporting multi-channel audio. Its open nature makes it free from patent encumbrances, which has historically been a significant advantage over proprietary formats like MP3 and AAC.
- **Usability:**
    - *Creation/Modification:* Ogg Vorbis files can be created from uncompressed audio sources (like WAV or FLAC) using various free and open-source encoders (e.g., Audacity, FFmpeg). Like other lossy formats, direct editing can lead to quality degradation upon re-compression; it's best to edit the original uncompressed source.
    - *Performance/Resource Requirements:* Ogg Vorbis offers good compression efficiency, resulting in smaller file sizes than uncompressed formats, which benefits storage and loading times. Decoding is generally efficient and well-optimized for software playback. It can be more CPU-intensive to decode than MP3, but on modern hardware, this difference is usually negligible. Its variable bitrate (VBR) encoding allows for dynamic quality and file size adjustments. It's particularly well-suited for game audio due to efficient streaming capabilities and good loopability.
    - *Engine/Platform Integration:* Ogg Vorbis has excellent native support in major game engines like Unity and Unreal Engine. It is often the default or recommended lossy audio format for in-game audio due to its open nature, good performance characteristics, and the ability to seamlessly loop audio without introducing silence or gaps (a common issue with poorly encoded MP3s).

- **Interoperability**:
  - *Content Creation Software:* Supported by a wide range of audio players, media management software, and many DAWs, especially in the open-source community. It is a common format in game development pipelines.
  - *VE/AR Systems & Web Platforms:* While not as universally supported as MP3 or AAC for direct HTML5 audio playback across all browsers, support has significantly improved (e.g., Firefox and Chrome support it). It is widely used in desktop and mobile gaming due to its strong engine integration. For web-based VEs, it might require fallbacks or specific libraries if broad browser compatibility is needed, though its usage is strong in the game development space.
- **Metadata Embedding Capabilities:** Ogg Vorbis uses "Vorbis comments" for metadata embedding. This is a flexible, text-based key-value system that allows for standard tags (e.g., ARTIST, TITLE, ALBUM, GENRE) and custom, user-defined tags. Multiple tags with the same name are allowed. This system is similar to how FLAC embeds metadata. While simple, it is robust and extensible enough for embedding descriptive metadata relevant to cultural heritage assets.
- **Suitability for IMPULSE and Virtual Environments:** Ogg Vorbis is highly suitable and recommended for delivering compressed audio content (background music, ambient sounds, voiceovers, and most sound effects) within the IMPULSE project's virtual environments. Its key advantages for this project include:
    - Open Standard and Royalty-Free: Aligns with principles of long-term accessibility and avoids proprietary licensing concerns, crucial for a public cultural heritage project.
    - Good Compression Efficiency and Quality: Offers excellent quality at competitive file sizes, often outperforming MP3 at lower bitrates.
    - Excellent Engine Integration: Its robust native support in Unity and Unreal Engine streamlines the development workflow.
    - Seamless Looping: Critical for ambient soundscapes and background music in VEs.
    - Flexible Metadata: Vorbis comments provide a simple yet effective way to embed descriptive metadata.
  - While WAV or FLAC should be used for archival source audio to preserve lossless quality , Ogg Vorbis is an excellent choice for the optimized, performant audio assets used at runtime within the virtual environments, balancing quality and efficiency effectively.

9.2 Audio Visual Format

9.2.1 MP4 (MPEG-4 Part 14)

- **Short Overview:** MP4 is a digital multimedia container format most used to store video and audio, but it can also be used to store other data such as subtitles and still images. It is an international standard and one of the most universal and widely supported formats for video distribution and playback, especially on the web and mobile devices. It offers a good balance between file size and quality.

- **Usability:**
  - *Creation/Modification:* MP4 files can be created and edited by a vast array of video editing software, from professional suites to consumer-level applications. Many cameras and mobile devices can also record directly in MP4 format.
  - *Performance/Resource Requirements:* MP4 files are known for their good compression, leading to relatively small file sizes for good quality video, which is beneficial for streaming and storage. Playback is generally efficient and well-supported by hardware acceleration on most modern devices.
  - *Engine/Platform Integration:* Most game engines and development platforms (like Unity and Unreal Engine) support MP4 import and playback, often using native operating system capabilities or built-in libraries. It's a common choice for in-application video.
- **Interoperability:**
  - *Content Creation Software:* Excellent interoperability with virtually all video editing software, converters, and media players.
  - *VE/AR Systems & Web Platforms:* Highly interoperable. MP4 is a standard format for HTML5 video and is widely supported by web browsers, making it ideal for web-based VR/AR experiences and general web video. Most VR/AR systems and devices support MP4 playback.
- **Metadata Embedding Capabilities:** MP4 files can store metadata within their structure, often using a format based on Apple's QuickTime container format. This can include standard tags like title, artist, album (for audio), creation date, and copyright information. It can also accommodate XMP (Extensible Metadata Platform) data for more extensive and structured metadata. The MP4-AT extension allows for auxiliary tracks and specific metadata for editing operations.
- **Suitability for IMPULSE and Virtual Environments:** Highly suitable. Its universality, good compression, wide compatibility, and reasonable metadata support make MP4 an excellent choice for delivering video content within the IMPULSE project, especially for web-based applications, training materials, or in-VE displays.

9.2.2 AVI (Audio Video Interleave)

- **Short Overview:** AVI is a multimedia container format introduced by Microsoft in November 1992 as part of its Video for Windows technology. It can contain both audio and video data in a file container that allows synchronous audio-with-video playback. While an older format, it can support multiple codecs.
- **Usability:**
  - *Creation/Modification:* AVI files can be created and edited by many video editing tools, particularly on Windows. However, it's less common as a primary export format in modern workflows compared to MP4.
  - *Performance/Resource Requirements:* File sizes can vary greatly depending on the codec used within the AVI container. Uncompressed or less efficiently compressed AVIs can be very large. Playback performance also depends on the codec and system support.

  - *Engine/Platform Integration:* Support in game engines and modern platforms can be mixed. While many systems can play AVI files (often relying on OS-level codecs), it's not always the most efficient or recommended format for integration into VEs due to potential codec compatibility issues and larger file sizes compared to more modern alternatives. Microsoft specified how DV data can be stored in AVI files (Type-1 and Type-2).
- **Interoperability:**
  - *Content Creation Software:* Good support for older and Windows-based video editing software. Modern cross-platform tools still often support import, but export might be less emphasized.
  - *VE/AR Systems & Web Platforms:* Less suitable for web platforms due to potentially large file sizes and inconsistent browser support. For VE/AR systems, conversion to more optimized formats are often preferred.
- **Metadata Embedding Capabilities:** AVI files can store some basic metadata in the RIFF INFO chunk, including information like title, artist, copyright, and creation date. However, its metadata capabilities are generally less extensive and standardized compared to formats like MP4 or MKV.
- **Suitability for IMPULSE and Virtual Environments:** Generally, less suitable for final delivery in modern VE applications due to potential issues with file size, codec compatibility, and limited metadata features compared to MP4 or MKV. It might be encountered with legacy footage, in which case conversion to a more suitable format would be advisable.

9.2.3 MOV (QuickTime File Format)

- **Short Overview:** MOV is a multimedia container file format developed by Apple and is native to the QuickTime framework. It can contain video, audio, text, effects, and subtitles. It's widely used in professional video editing environments, especially within the Apple ecosystem.
- **Usability:**
  - *Creation/Modification:* Excellently supported by video editing software on macOS (e.g., Final Cut Pro, Adobe Premiere Pro). Many cameras, particularly those popular in professional videography, can record in MOV format.
  - *Performance/Resource Requirements:* Like MP4, MOV is a container, and the performance depends heavily on the codecs used (e.g., H.264, ProRes). Files can range from highly compressed for distribution to very large, high-quality files for professional editing. Playback is well-optimized on Apple devices.
  - *Engine/Platform Integration:* Support in game engines (Unity, Unreal Engine) is generally good, especially for common codecs like H.264. However, some professional codecs within MOV containers might require conversion for broader compatibility or optimal performance in VEs.
- **Interoperability:**
  - *Content Creation Software:* Very good interoperability, especially in professional video workflows and across Apple software. Windows support is generally good with QuickTime or compatible players/editors installed.

  - *VE/AR Systems & Web Platforms:* MOV files using common codecs like H.264 are often playable on web platforms and VE/AR systems. However, for maximum web compatibility, MP4 is sometimes preferred. Some proprietary Apple codecs (like ProRes) are not natively supported in web browsers.
- **Metadata Embedding Capabilities:** MOV files have robust metadata capabilities, derived from the QuickTime format structure. They can store a wide range of metadata, including descriptive information (title, director, copyright), technical details, timecode, and custom annotations. This makes it suitable for professional workflows where metadata is important.
- **Suitability for IMPULSE and Virtual Environments:** Suitable, particularly if source footage comes from Apple devices or professional video production workflows that use MOV. For broader distribution or web use within the project, conversion to MP4 might be considered to ensure maximum compatibility, unless specific high-quality Apple codecs are required for archival or specific presentation purposes.

9.2.4 MKV (Matroska Multimedia Container)

- **Short Overview:** MKV is an open standard, free container format that can hold an unlimited number of video, audio, picture, or subtitle tracks in one file. It's known for its flexibility and support for a wide range of codecs and features, such as multiple audio and subtitle tracks, chapter points, and rich metadata. It is based on EBML (Extensible Binary Meta Language).
- **Usability:**
  - *Creation/Modification:* MKV files can be created using various tools. Many video converters and editors support MKV. Its flexibility allows for packaging diverse multimedia elements.
  - *Performance/Resource Requirements:* Performance depends on the codecs used within the container. MKV itself adds little overhead. Its ability to hold multiple high-quality streams can lead to large file sizes if not managed.
  - *Engine/Platform Integration:* Direct support in game engines like Unity and Unreal Engine can be limited without third-party plugins or conversion. While many media players support MKV, it's not universally supported at the OS or hardware level as MP4 for direct application integration.
- **Interoperability:**
  - *Content Creation Software:* Good support in open-source tools and increasingly in commercial software for import/export. It's popular for storing and distributing video content, especially high-definition movies with multiple language tracks and subtitles.
  - *VE/AR Systems & Web Platforms:* Not natively supported by most web browsers for HTML5 video. For VE/AR systems, conversion to MP4 or another widely supported format is usually necessary for broad compatibility, though some specialized players might handle MKV.
- **Metadata Embedding Capabilities:** MKV has robust and flexible metadata capabilities. It can store extensive metadata, including tags for title, director, actors,

track languages, chapter information, and even attachments (like cover art or fonts). Its EBML structure allows for custom metadata elements.

- **Suitability for IMPULSE and Virtual Environments:** Excellent as an archival or master format due to its flexibility, support for multiple tracks, rich metadata, and open standard nature. However, for direct use in most VE applications or web platforms, conversion to MP4 would likely be required. Its strong metadata features could be valuable for detailed cataloguing of audiovisual assets within the project.

9.2.5 WMV (Windows Media Video)

- **Short Overview:** WMV is a video compression format developed by Microsoft. It was originally designed for internet streaming applications as a competitor to RealVideo. WMV files are often contained within the ASF (Advanced Systems Format) container.
- **Usability:**
    - **Creation/Modification:** Primarily created and edited using Windows-based software, such as Windows Movie Maker or professional editors with WMV export options.
    - **Performance/Resource Requirements:** WMV can offer good compression, especially at lower bitrates. Performance is generally good on Windows platforms.
    - **Engine/Platform Integration:** Support in game engines and cross-platform development environments can be limited compared to MP4. While Windows-based applications might handle WMV natively, integration into non-Windows VE/AR systems often requires conversion.
- **Interoperability:**
    - **Content Creation Software:** Good support within the Windows ecosystem. Cross-platform support can be less comprehensive.
    - **VE/AR Systems & Web Platforms:** Not widely supported for native web playback. For VE/AR systems, especially cross-platform ones, WMV is generally not a preferred format and would typically be converted.
- **Metadata Embedding Capabilities:** WMV files (within an ASF container) support metadata, including attributes like title, author, copyright, and rating.
- **Suitability for IMPULSE and Virtual Environments:** Generally, not recommended for primary use in the IMPULSE project due to its proprietary nature and limited cross-platform interoperability compared to standards like MP4. If legacy content is in WMV format, conversion to a more open and widely supported format would be advisable for broader accessibility and integration.

9.2.6 AVCHD (Advanced Video Coding High Definition)

- **Short Overview:** AVCHD is a file-based format for digital recording and playback of high-definition video. It was jointly developed by Sony and Panasonic and is primarily used in their camcorders. AVCHD uses MPEG-4 AVC/H.264 for video compression and Dolby Digital (AC-3) or linear PCM for audio. It's designed for recording HD signals using high-efficiency compression.

- **Usability:**
  - *Creation/Modification:* AVCHD is primarily a recording format for camcorders. Video editing software (e.g., Adobe Premiere Pro, Final Cut Pro) can import and edit AVCHD footage, often from the complex directory structure recorded by cameras.
  - *Performance/Resource Requirements:* Offers good quality high-definition video at relatively efficient file sizes due to H.264 compression. Editing can be resource-intensive due to the inter-frame compression.
  - *Engine/Platform Integration:* Direct import into game engines is unlikely. AVCHD footage typically needs to be transcoded to a more engine-friendly format (like MP4 with H.264) for use in VEs.
- **Interoperability:**
  - *Content Creation Software:* Well-supported by professional and consumer video editing software for import.
  - *VE/AR Systems & Web Platforms:* Not suitable for direct web playback or VE/AR system integration. The complex file structure and specific transport stream (.MTS or .M2TS) files require processing and conversion. Playback is possible on AVCHD-compatible devices like Blu-ray players.
- **Metadata Embedding Capabilities:** AVCHD files store metadata related to the recording, such as date, time, camera settings, and sometimes GPS information, within the file structure or associated clip information files (.CPI). However, standardized descriptive metadata embedding for broader use is less straightforward than in formats like MP4 or MKV.
- **Suitability for IMPULSE and Virtual Environments:** If high-definition footage is captured using AVCHD camcorders, it will serve as a source format. For use within the IMPULSE project's VEs or for web distribution, this footage will need to be transcoded into a more standard and accessible format like MP4. The original AVCHD files can be kept for archival purposes.

9.2.7 DV (Digital Video)

- **Short Overview:** DV is a format for recording and playing back digital video, launched in 1995. It was widely used in standard-definition consumer and professional camcorders. DV uses intraframe compression (compressing each frame individually), which makes editing less computationally intensive.
- **Usability:**
  - *Creation/Modification:* DV was a common recording format. Video editing software widely supports importing DV footage.
  - *Performance/Resource Requirements*: Standard-definition DV has a consistent data rate (around 3.6 MB/s), resulting in relatively large file sizes for its resolution compared to modern inter-frame codecs. Editing performance is generally good due to intraframe compression.
  - Engine/Platform Integration: Direct use in modern game engines or VEs is uncommon. DV footage would typically be deinterlaced (if applicable) and transcoded to a format like MP4 for use in such applications. Microsoft specified how DV data could be stored in AVI files.

- **Interoperability:**
    - *Content Creation Software:* Excellent support in older and current video editing software for capturing and editing.
    - *VE/AR Systems & Web Platforms:* Not suitable for direct web playback or modern VE/AR integration due to its resolution limitations, file size, and older codec technology.
- **Metadata Embedding Capabilities:** DV streams can carry metadata, including timecode, recording date/time, and camera settings. When wrapped in containers like AVI or MOV, these containers can also add their own metadata.
- **Suitability for IMPULSE and Virtual Environments:** Primarily relevant if dealing with legacy standard-definition footage. For any use in modern VEs, DV content would require upscaling (if desired, though often with quality loss), deinterlacing, and transcoding to a modern format. Its main value would be as an archival source for older material.

9.3 Analysis of Video Codecs

9.3.1 H.264 (AVC - Advanced Video Coding / MPEG-4 Part 10)

- **Short Overview:** H.264, also known as Advanced Video Coding (AVC) or MPEG-4 Part 10, is currently one of the most widely used video compression standards. It offers significantly better compression efficiency than previous standards, providing good video quality at lower bitrates. It's used in everything from Blu-ray Discs and streaming services to video conferencing and broadcast television.
- **Usability:**
    - *Creation/Modification (Encoding/Decoding):* H.264 encoding is available in virtually all video editing software, conversion tools, and many hardware devices (cameras, phones). Decoding is supported by a vast range of software and hardware. It has multiple profiles (Baseline, Main, High) catering to different application needs.
    - *Performance/Resource Requirements:* H.264 strikes a good balance between compression efficiency and computational requirements for encoding/decoding. Hardware acceleration for H.264 is common, making playback efficient on most devices.
    - *Engine/Platform Integration:* Widely supported by game engines, mobile operating systems, and web browsers. It's a common codec choice within MP4 containers for video in VEs.
- **Interoperability:**
    - *Content Creation Software:* Universal support across video creation and processing tools.
    - *VE/AR Systems & Web Platforms:* Excellent interoperability. H.264 is the dominant codec for web streaming (often in MP4) and is ubiquitously supported on playback devices and platforms.
- **Metadata Embedding Capabilities:** As a codec, H.264 itself doesn't define metadata storage. Metadata is handled by the container format (e.g., MP4, MOV, MKV) that

wraps the H.264-encoded video stream. The H.264 standard focuses on efficiently representing video content (Video Coding Layer - VCL) and formatting it for transport or storage (Network Abstraction Layer - NAL).

- **Suitability for IMPULSE and Virtual Environments:** Highly suitable and recommended as the primary video codec for delivering video content within the IMPULSE project. Its combination of good quality, efficient compression, and unparalleled compatibility makes it ideal for various applications, including web, mobile, and desktop VEs.

9.3.2 MPEG (Moving Picture Experts Group) Video Codecs

- **Short Overview:** MPEG refers to a family of digital video compression standards developed by the Moving Picture Experts Group. Key MPEG video codecs include:
    - MPEG-1: Developed in 1993, primarily for Video CDs (VCDs) and early digital video applications. Resolution: typically, 352x240 or 352x288.
    - MPEG-2 (H.262): Developed in 1995, used for DVDs, digital television (DVB), and early Blu-ray Discs. Offers better quality than MPEG-1 and supports higher resolutions and interlaced video.
    - MPEG-4 Part 2 (often just called "MPEG-4 Visual"): An older MPEG-4 standard, distinct from MPEG-4 Part 10 (H.264/AVC). It includes codecs like DivX and Xvid. Offers better compression than MPEG-2.
    - (H.264/AVC is MPEG-4 Part 10, already covered).
    - MPEG-H Part 2 (HEVC/H.265): A successor to H.264, offering roughly double the compression efficiency for the same quality. Used for 4K/UHD content, HDR video.
    - MPEG also develops standards for immersive media (MPEG-I), including immersive video and audio.
- **Usability:**
    - *Creation/Modification (Encoding/Decoding):*
        - MPEG-1 & MPEG-2: Encoding/decoding is widely supported by older software and hardware. Modern tools can still handle them, but they are less common for new content creation.
        - MPEG-4 Part 2: Popularized by DivX/Xvid, many tools support it.
        - HEVC/H.265: Increasingly supported in modern cameras, software, and hardware, but encoding can be more computationally intensive than H.264.
    - *Performance/Resource Requirements:*
        - MPEG-1/2: Relatively low computational requirements but less efficient compression than newer codecs.
        - MPEG-4 Part 2: Better compression than MPEG-2, moderate resource use.
        - HEVC/H.265: More demanding for encoding/decoding but offers superior compression. Hardware support is important for efficient playback of high-resolution content.
    - *Engine/Platform Integration:*

        - MPEG-1/2: Older formats; support in modern game engines might be limited or rely on OS capabilities. Generally, not ideal for VEs.
        - MPEG-4 Part 2: Like MPEG-1/2 regarding modern engine integration.
        - HEVC/H.265: Support is growing in modern engines and platforms, especially for 4K content. However, licensing and royalty considerations have sometimes impacted adoption speed compared to H.264.
- **Interoperability:**
    - *Content Creation Software:*
        - MPEG-1/2, MPEG-4 Part 2: Widely supported for legacy content.
        - HEVC/H.265: Good support in modern video editing and encoding tools.
    - *VE/AR Systems & Web Platforms:*
        - MPEG-1/2: Not suitable for modern web or VE/AR.
        - MPEG-4 Part 2: Limited web support.
        - HEVC/H.265: Growing support, but H.264 remains more universally compatible for web. Some browsers support HEVC if the OS does. Important for high-quality 4K/HDR delivery where supported.
- **Metadata Embedding Capabilities:** Like H.264, MPEG video codecs themselves don't typically store extensive metadata. This is handled by the container format (e.g., MP4, TS, MKV) that holds the MPEG-encoded video. MPEG standards do define various metadata aspects related to the stream itself (e.g., for synchronization, content description within the transport stream). For instance, MPEG-7 is a standard specifically for multimedia content description, and MPEG-21 defines a multimedia framework that includes metadata.
- **Suitability for IMPULSE and Virtual Environments:**
    - MPEG-1, MPEG-2, MPEG-4 Part 2: Generally, not suitable for new content creation for the IMPULSE project due to lower compression efficiency and limited modern platform support compared to H.264 or HEVC. Relevant mainly for dealing with legacy archives.
    - HEVC/H.265: Could be considered for very high-quality video delivery (e.g., 4K, HDR immersive experiences) if target platforms support it and the increased encoding/decoding complexity is manageable. However, for broad compatibility, H.264 is currently a safer choice. The project should weigh the quality benefits of HEVC against the broader reach of H.264. MPEG's work on immersive video (MPEG-I) might become relevant for future VR/AR applications.

10 Conclusion

This section synthesizes the analysis of 2D, 3D, and audiovisual file formats for virtual environment applications in cultural heritage digitization. It further integrates key outtakes from the other deliverables towards the joint effort to provide an integrated perspective on the challenges and pathways for the usage of digitized CH in immersive contexts.

10.1 File Format for Cultural Heritage in Virtual Environments

The selection of appropriate file formats for representing cultural heritage within virtual environments stands as a critical decision point, demanding a balance between several competing factors: archival fidelity, real-time performance capabilities, interoperability across diverse software and hardware platforms, the richness of embedded or associated metadata, and considerations for long-term accessibility and preservation.

The systematic evaluation has revealed that the file formats ideally suited for archival purposes—such as TIFF, RAW camera data, or high-polygon count 3D models in formats like PLY or OBJ, FLAC or MOV for audio data— frequently present substantial challenges when direct integration into VEs is attempted. This difficulty arises from their large file sizes and the computationally intensive decoding processes they require, rendering them inefficient for the demands of real-time interaction. Conversely, formats optimized for virtual environments, including JPEG for 2D assets, glTF/GLB for 3D models, and MP4 for audiovisual content, are engineered to prioritize efficient rendering and smaller data footprints, often achieving this at the expense of some data fidelity or a reduction in metadata capacity.

For instance, TIFF (Tagged Image File Format) is highly valued in archival settings for its lossless quality and robust support for various colour profiles and metadata, yet its large file sizes make it generally unsuitable for direct use in VEs without prior conversion and optimization. JPEG is inherently lossy and lacks support for transparency, which can be a limitation for certain CH assets. In the 3D domain, glTF/GLB (GL Transmission Format) has emerged as a suitable standard for VE delivery, specifically designed for efficient transmission and loading of 3D scenes with support for Physically Based Rendering (PBR) materials. In contrast, older formats like OBJ, while useful for basic static geometry exchange, lack the comprehensive features needed for rich, interactive VEs. For audiovisual content, MP4 containers with H.264 video and AAC/MP3/Ogg Vorbis audio codecs provide a common, well-supported solution for VEs.

The fundamental difference identified between archival-focused formats and those tailored for real-time virtual environments is not merely a technical distinction. Archival formats prioritize long-term preservation, aiming for the highest possible, resolution, quality, fidelity in representation and comprehensive metadata capture, often resulting in large, complex files (e.g., TIFF, RAW, high-poly PLY/OBJ). VE-optimized formats, conversely, are engineered for rapid loading, efficient GPU rendering, and broad compatibility, often accepting trade-offs in resolution or metadata capacity to achieve smaller file sizes and streamlined processing (e.g., JPEG, glTF/GLB). This distinction highlights a fundamental strategic divide in contemporary CH digitization practices.

This divide manifests as a preservation-application gap, where institutions create high-quality archival assets that are not immediately suitable for practical use in VEs. The divergence has profound implications for CH workflows, resource allocation, and the ultimate potential for asset reuse in immersive applications. Digitization workflows, traditionally focus towards preservation, often lack standardized, efficient conversion pipelines to bridge this gap, necessitating a "curatorial compression" process where decisions are made about acceptable levels of detail and data loss for VE suitability—a process often lacking clear guidelines. Resource allocation is impacted as initial investments focus on archival masters, potentially leading to significant future costs for VE adaptation, a burden particularly for institutions with limited budgets. Furthermore, the future reuse of these digital assets in immersive applications is severely hampered, this underutilization contributes to the "digital heritage paradox," where vast digitized collections remain functionally inaccessible due to technical fragmentation, incompatible formats, and inadequate metadata.

If heritage institutions continue to primarily generate archival masters without adequate consideration for future VE use cases, they inadvertently erect significant barriers to the adoption of their digital collections in these emerging platforms. This oversight can lead to assets becoming technologically orphaned as platforms evolve, necessitating either costly re-digitization efforts or the implementation of complex, and potentially lossy, conversion pipelines to bridge the gap. Such reactive measures are inefficient and risk compromising the integrity of the CH assets. Consequently, the analysis of the research underscores an urgent need for a more proactive and integrated strategy regarding file format selection and management in CH digitization, one that anticipates the requirements of immersive environments from the outset, potentially through metadata-first frameworks and XR-optimized asset production strategies.

A closer examination of key format considerations for VE applications in CH further illuminates the following needs:

- **2D Data**: For the display of 2D cultural heritage in VEs, optimized JPEGs are generally suitable for photographic content where file size is a critical constraint. PNGs are indispensable for elements that require transparency (e.g., virtual information overlays) or demand perfect lossless fidelity. Archival masters in TIFF or RAW formats invariably require conversion and optimization processes before they can be efficiently utilized in VEs. For integrating textual documents, such as virtual books or information panels, PDF (Portable Document Format) is valuable for preserving fixed-layout fidelity across platforms. However, for more interactive textual experiences, formats like ALTO (Analyzed Layout and Text Object) XML or hOCR (HTML-based OCR format), which provide structured OCR data, are crucial as they enable text searchability, highlighting, and dynamic presentation within the VE.

- **3D Data**: The glTF/GLB format is strongly advocated as the primary delivery standard for 3D CH assets in VEs. Its design prioritizes runtime efficiency, supports PBR materials for realistic rendering, and benefits from a rapidly growing ecosystem of creation tools and engine support. Universal Scene Description (USD) is also gaining significant traction, particularly for describing complex 3D scenes and fostering interoperability in professional production pipelines that might feed into VE development. While formats like OBJ and PLY (Polygon File Format) are commonly used for basic geometry interchange or as archival formats for 3D scans, they inherently lack the rich feature sets—such as support for animations, complex material definitions, or hierarchical scene structures—that are necessary for creating dynamic, interactive, and contextually rich VEs. Native authoring formats from Digital Content Creation (DCC) tools, such as .blend (Blender) or .max (3ds Max), are integral to the 3D modelling pipeline but are not suited for direct delivery to VEs ; assets created in these formats must be exported to VE-optimized formats like glTF.
- **Audiovisual Data**: For incorporating audiovisual CH content into VEs, MP4 containers encoded with H.264 for video and AAC for audio offer a widely supported and effective balance of visual/auditory quality, compression efficiency, and cross-platform compatibility. For ambient soundscapes or background audio where minimizing file size is paramount, MP3 remains a viable, albeit lossy, option. Ogg Vorbis is also an excellent choice for compressed audio due to its open standard nature, good compression efficiency, and excellent engine integration, particularly for seamless looping in VEs. Archival-quality audio formats such as WAV (Waveform Audio File Format) or FLAC (Free Lossless Audio Codec), which are prized for their uncompressed or losslessly compressed fidelity, must typically be converted to more efficient, often lossy, formats for use in VEs to avoid significant impacts on loading times and runtime performance.

The transition from archival master formats to VE-ready delivery formats is not merely a technical conversion; it represents a critical "curatorial compression". This process involves informed decisions about the level of detail or information that is essential to convey the cultural significance of an asset within an immersive experience, versus what aspects can be acceptably simplified, optimized, or even omitted to meet performance targets. These are fundamentally curatorial judgments, informed by technical possibilities.

However, the absence of standardized guidelines for this "curatorial compression" process, particularly in relation to specific file formats such as 3D asset types, identifies a significant gap. A lack of standardized approaches to these pipelines can lead to inconsistent asset quality and data fragmentation, even if standardized delivery formats are ultimately chosen. Future work within projects like IMPULSE could address this by developing best-practice recommendations for these essential transformation pipelines, ensuring that the integrity and interpretative potential of CH assets are maintained as effectively as possible when adapted for immersive environments. The choice of tools, parameters for optimization (e.g., reduction levels for 3D models, compression ratios for images), and quality control measures within this pipeline directly dictates the quality of the final VE experience and the fidelity of the CH representation.

Data Type	Recommended VE Formats	Key Strengths for VE	Archival Counterparts / Source Formats	Primary Challenges for VE Integration
2D Image	PNG, JPEG (optimized)	Transparency (PNG), Lossless (PNG), Good compression (JPEG)	TIFF, RAW, BMP	Optimization required from archival formats; balancing quality vs. file size; consistent metadata application
2D Text	PDF (for fixed display), ALTO XML, hOCR (for interactive display)	Preserves layout (PDF), Structured text & layout data (ALTO/hOCR), Enables searchability & interaction	Original physical documents, DOCX, ODF, RTF	Ensuring legibility in VE; making text interactive vs. static display; integrating with 3D environments; consistent OCR quality.
3D Model	glTF/GLB, USD	VE-optimized, PBR materials, Animation (glTF), Scene description (USD), Growing ecosystem support	PLY, OBJ, STL, DAE, FBX (as interchange), Native CAD/DCC formats (.blend, .max)	Extreme polygon counts from scans; complex topology unsuitable for real-time; unoptimized UVs; lack of LODs; inconsistent material definitions; missing or poor metadata; integration into custom MUVE platforms.
4D Audio	MP3, AAC, Ogg Vorbis	Small file size, Wide compatibility	WAV, FLAC, AIFF	Balancing quality with performance; integrating spatial audio; managing licensing for music/narration; lack of contextual metadata.

4D Video	MP4 (H.264/AVC video, AAC audio)	Good balance of quality, compression, and compatibility	AVI, MOV, MKV, DV, Professional codecs (ProRes, DNxHD)	Large file sizes impacting streaming/loading; codec compatibility across platforms; ensuring sufficient resolution for immersive displays; lack of interactive elements or rich metadata; variable quality from diverse sources.

10.2 File Format Challenges and Opportunities

The file format analysis in this document is further contextualized by insights from wider research within the IMPULSE project, including work on MUVE technologies, the use of digital heritage objects, and AI-driven solutions. These findings show that file format decisions are deeply linked to MUVE development challenges, the need for standardization, the digital heritage paradox, and the potential of AI.

The project's research into CH MUVEs indicates a common prioritization of visual fidelity in representation.
This impacts file format choices, pushing towards high-resolution and complex models which strain VE performance without optimization (e.g., using glTF/GLB). The prevalence of custom-made MUVE solutions often leads to proprietary format integrations, hindering interoperability and long-term asset accessibility. This aligns with our findings on significant gaps in managing and sharing "6D data" (integrated 2D, 3D, and/or 4D data in immersive environments) and institutional difficulties in implementing even established standards. The lack of 6D data or evolving standards makes careful selection of underlying file formats even more critical. This situation highlights a crucial priority for the IMPULSE project: to promote and exemplify, through its pilot initiatives, workflows that emphasize the use of open, well-documented, and interoperable file formats (such as glTF, along with standardized image, audio, and video formats discussed here) as the foundational layer for all CH assets intended for MUVEs, regardless of the potentially customized nature of the hosting platform.

Broader project research on content suitability brings to light the “digital heritage paradox” (WP2): vast amounts of digitized cultural heritage content remain underutilized due to a combination of lack of consistency/depth in metadata (metadata crisis), technical fragmentation, and a disconnect between preservation of files and applications. This document’s analysis of file formats offers a lens through which to better understand these challenges. Technical fragmentation often arises from inconsistent format choices and the absence of standardized conversion pipelines. The preservation-application gap is particularly evident in the mismatch between archival formats (e.g., high-resolution TIFFs, dense point clouds) and formats suitable for virtual environments. Effective file format management—including strategic format selection, reliable conversion workflows, and comprehensive metadata embedding—is essential to bridging this divide. Even with optimal file formats, the metadata crisis significantly undermines asset usability. This is demonstrated through our analysis of the metadata capabilities of key formats such as TIFF, JPEG, glTF. The assessment of digital object use reveals a substantial gap in metadata standards and their implementation—especially for immersive platforms and complex data types—thereby contributing to the persistence of orphaned digitization or data without necessary context. Addressing these issues requires integrated strategies that consider format choice, metadata creation and aggregation, and the intended application from the outset of any digitization initiative.

11 Selected References

Data type	File Format	Reference Title	Document Link
3D	3DS (.3ds)	FileFormat.com 3DS Overview	https://docs.fileformat.com/3d/3ds/
3D	3DS (.3ds)	.3ds - Wikipedia	https://en.wikipedia.org/wiki/.3ds
3D	3DS (.3ds)	Autodesk 3DS Max Legacy Documentation	https://help.autodesk.com/view/3DSMAX/2025/ENU/
3D	3DS (.3ds)	3DS files - Adobe Creative Cloud	https://www.adobe.com/creativecloud/file-types/image/vector/3ds-file.html
3D	3DS (.3ds)	3DS MAX Formats - CADinterop	https://www.cadinterop.com/en/formats/mesh/3ds-max.html
3D	3DS (.3ds)	Understanding.3ds File Format: What You Need to Know - Modelo.io	https://www.modelo.io/damf/article/2024/05/12/0031/understanding-.3ds-file-format--what-you-need-to-know
3D	3DS (.3ds)	How to Open 3ds Files – 3ds Viewer - Sibe.io	https://www.sibe.io/3d-viewer/3ds
3D	3ds Max (.max)	What is 3ds Max? - 3dviewermax.com	https://3dviewermax.com/guides/3ds-max-viewer/
3D	3ds Max (.max)	One of the key benefits of Autodesk 3ds Max is photorealism - CGIFurniture	https://cgifurniture.com/blog/3d-max-furniture-rendering/
3D	3ds Max (.max)	Convert 3D models online - free and secure - Convert3D	https://convert3d.org/
3D	3ds Max (.max)	Autodesk 3ds Max - Wikipedia	https://en.wikipedia.org/wiki/Autodesk_3ds_Max
3D	3ds Max (.max)	Blender MAX Import Add-on	https://extensions.blender.org/add-ons/io-scene-max/
3D	3ds Max (.max)	MAX Files - Autodesk	https://help.autodesk.com/view/3DSMAX/2024/ENU/?guid=GUID-88DE2443-6869-4820-973A-10E7695B6DE6
3D	3ds Max (.max)	Autodesk 3ds Max File Format Info	https://www.autodesk.com/support/technical/article/caas/sfdcarticles/sfdcarticles/What-file-formats-does-3ds-Max-import-and-export.html
3D	3ds Max (.max)	3ds Max Sample Files - Autodesk Support	https://www.autodesk.com/support/technical/article/caas/tsarticles/ts/3CM2c0t6Fvo2lSawUNRICT.html
4D	AAC (Advanced Audio Coding)	Advanced Audio Coding - Wikipedia	https://en.wikipedia.org/wiki/Advanced_Audio_Coding
4D	AAC (Advanced Audio Coding)	AAC Codec is a top choice for compressing audio files efficiently - Gumlet	https://www.gumlet.com/learn/aac-codec/
3D	AAC (Advanced Audio Coding)	Library of Congress AAC Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000036.shtml

4D	AAC (Advanced Audio Coding)	AAC (MPEG-4) Low Complexity Object - Library of Congress	https://www.loc.gov/preservation/digital/formats/fdd/fdd000233.shtml
4D	AIFF (Audio Interchange File Format)	Library of Congress AIFF Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000005.shtml
3D	Alembic (.abc)	Alembic File Importer in Unreal Engine - Unreal Engine Documentation	https://dev.epicgames.com/documentation/en-us/unreal-engine/alembic-file-importer-in-unreal-engine
3D	Alembic (.abc)	Alembic - Marmoset Toolbag Documentation	https://docs.marmoset.co/docs/alembic/
3D	Alembic (.abc)	Wikipedia Alembic Entry	https://en.wikipedia.org/wiki/Alembic_(computer_graphics)
3D	Alembic (.abc)	Alembic - Wikipedia	https://en.wikipedia.org/wiki/Alembic_computer_graphics
3D	Alembic (.abc)	Autodesk Alembic Docs	https://help.autodesk.com/cloudhelp/2022/ENU/3DSMax-Data-Exchange/files/GUID-D80A02B6-BC7B-4070-A959-94EC5FCA22F8.htm
3D	Alembic (.abc)	Alembic Caching - Autodesk	https://help.autodesk.com/view/MAYAUL/2026/ENU/?guid=GUID-9D272E39-9279-4146-8449-928DDA865C9D
3D	Alembic (.abc)	Alembic Official Site (Open-Source Project)	https://www.alembic.io/
3D	Alembic (.abc)	Autodesk Alembic - Library of Congress	https://www.loc.gov/preservation/digital/formats/fdd/fdd000560.shtml
2D text	ALTO XML	ALTO XML GitHub	https://altoxml.github.io/
2D text	ALTO XML	Analyzed Layout and Text Object - Wikipedia	https://en.wikipedia.org/wiki/Analyzed_Layout_and_Text_Object
2D text	ALTO XML	GitHub ALTO XML	https://github.com/altoxml
2D text	ALTO XML	Introduction to the Alto/Didl, TEI and Page format	https://kbnlresearch.github.io/xml-workshop/6_xmlformats_intro.html
2D text	ALTO XML	METAe—Automated Encoding of Digitized Texts	https://www.clarin.eu/sites/default/files/muehlberger_0.pdf
2D text	ALTO XML	ALTO XML Data Conversion	https://www.easydataworld.com/alto-xml/
2D text	ALTO XML	Library of Congress ALTO XML Format Description	https://www.loc.gov/standards/alto/
2D text	ALTO XML	ALTO: Technical Metadata for Optical Character Recognition	https://www.loc.gov/standards/alto/description.html
2D text	ALTO XML	METS: Standardized Encoding for Digital Library Objects	https://www.researchgate.net/publication/32954905_METS_Standardized_Encoding_for_Digital_Library_Objects

4D	AVCHD (Advanced Video Coding High Definition)	AVCHD - Wikipedia	https://en.wikipedia.org/wiki/AVCHD
4D	AVCHD (Advanced Video Coding High Definition)	What Is an AVCHD File? - Wondershare Recoverit	https://recoverit.wondershare.com/video-recovery/what-is-avchd-file.html
4D	AVCHD (Advanced Video Coding High Definition)	AVCHD (Advanced Video Code High Definition) is a standard that uses coding technology - Canon	https://support.usa.canon.com/kb/s/article/ART109154
4D	AVCHD (Advanced Video Coding High Definition)	AVCHD Information Web Site	https://www.avchd-info.org/format/
4D	AVCHD (Advanced Video Coding High Definition)	Library of Congress avchd description format	https://www.digitalpreservation.gov/formats/fdd/fdd000081.shtml
4D	AVCHD (Advanced Video Coding High Definition)	What is the AVCHD format? - Sony USA	https://www.sony.com/electronics/support/articles/00016537
4D	AVI (Audio Video Interleave)	AVI RIFF File Reference - Microsoft Learn	https://learn.microsoft.com/en-us/windows/win32/directshow/avi-riff-file-reference
4D	AVI (Audio Video Interleave)	AVI files - Adobe	https://www.adobe.com/creativecloud/file-types/video/container/avi.html
	AVI (Audio Video Interleave)	Library of Congress AVI description format	https://www.digitalpreservation.gov/formats/fdd/fdd000059.shtml
4D	AVI (Audio Video Interleave)	AVI (Audio Video Interleaved) File Format - Library of Congress	https://www.loc.gov/preservation/digital/formats/fdd/fdd000059.shtml
4D	AVI (Audio Video Interleave)	MP4 versus AVI comparison - Movavi	https://www.movavi.com/learning-portal/avi-vs-mp4.html
3D	Blender (.blend)	How does blender really save a blendfile? - Blender StackExchange	https://blender.stackexchange.com/questions/73913/how-does-blender-really-save-a-blendfile
3D	Blender (.blend)	BLEND Converter Online (Free) - Convert 3D	https://convert3d.org/convert/blend

3D	Blender (.blend)	Blender Manual	https://docs.blender.org/manual/en/3.6/files/import_export.html
3D	Blender (.blend)	Blender Foundation Documentation (official)	https://docs.blender.org/manual/en/latest/
3D	Blender (.blend)	.blend file format of Blender: Python parsing library - Kaitai Struct	https://formats.kaitai.io/blender_blend/python.html
3D	Blender (.blend)	Library of Congress Blender scene description format	https://www.loc.gov/preservation/digital/formats//fdd/fdd000559.shtml
3D	Blender (.blend)	Blendify -- Python rendering framework for Blender - ResearchGate	https://www.researchgate.net/publication/385177193_Blendify_--_Python_rendering_framework_for_Blender
2D image	BMP	The Evolution of BMP (Bitmap) Image Formats - LSoft Technologies	http://www.lsoft.net/posts/bmp-image-formats/
2D image	BMP	Should You Still Use BMP Format? - Cloudinary	https://cloudinary.com/guides/image-formats/should-you-still-use-bmp-format-use-cases-and-pros-cons
2D image	BMP	What is a BMP file? - docs.fileformat.com	https://docs.fileformat.com/image/bmp/
2D image	BMP	BMP file format - Wikipedia	https://en.wikipedia.org/wiki/BMP_file_format
2D image	BMP	FileInfo BMP	https://fileinfo.com/extension/bmp
2D image	BMP	Library of Congress BMP Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000189.shtml
3D	DAE (Collada - Collaborative Design Activity)	Collada (.dae) - Marmoset Toolbag Documentation	https://docs.marmoset.co/docs/collada/
3D	DAE (Collada - Collaborative Design Activity)	COLLADA - Wikipedia	https://en.wikipedia.org/wiki/COLLADA
3D	DAE (Collada - Collaborative Design Activity)	Adobe COLLADA Overview	https://www.adobe.com/creativecloud/file-types/image/vector/collada-file.html
3D	DAE (Collada - Collaborative Design Activity)	Khronos Group COLLADA Specification	https://www.khronos.org/collada/
3D	DAE (Collada - Collaborative Design Activity)	COLLADA: Sailing the gulf of 3D digital content creation - ResearchGate	https://www.researchgate.net/publication/328870627_COLLADA_Sailing_the_gulf_of_3D_digital_content_creation

3D	DAE (Collada - Collaborative Design Activity)	How to Open DAE Files – DAE Viewer - Sibe.io	https://www.sibe.io/3d-viewer/dae
2D text	DOCX	Analyze Malicious Microsoft Office Files - Intezer	https://intezer.com/blog/analyze-malicious-microsoft-office-files/
2D text	DOCX	Structure of a WordprocessingML document - Microsoft Learn	https://learn.microsoft.com/en-us/office/open-xml/word/structure-of-a-wordprocessingml-document
2D text	DOCX	Learn about file formats - Microsoft Support	https://support.microsoft.com/en-us/office/learn-about-file-formats-56dc3b55-7681-402e-a727-c59fa0884b30
2D text	DOCX	What.DOCX is and how to use it - Adobe Acrobat	https://www.adobe.com/acrobat/resources/document-files/docx.html
2D text	DOCX	ISO/IEC 29500:2008 Office Open XML File Formats	https://www.iso.org/standard/51463.html
2D text	DOCX	Library of Congress DOCX Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000400.shtml
2D text	DOCX	DOCX stands out as one of the most widely used formats - OnlyOffice	https://www.onlyoffice.com/blog/2024/03/docx
4D	DV (Digital Video)	DV Format - Planetoftunes	http://www.planetoftunes.com/digital-video/dv-format.php
4D	DV (Digital Video)	Digital video - Wikipedia	https://en.wikipedia.org/wiki/Digital_video
4D	DV (Digital Video)	DV - Wikipedia	https://en.wikipedia.org/wiki/DV_video_format
4D	DV (Digital Video)	DV Formats: Everything You Need To Know - Adam Wilt	https://www.adamwilt.com/DV-tech.html
4D	DV (Digital Video)	Library of Congress DV description format	https://www.digitalpreservation.gov/formats/fdd/fdd000183.shtml
4D	DV (Digital Video)	Libray of Congres DV file description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000183.shtml
4D	DV (Digital Video)	DV Formats: Everything You Need To Know - VideoUniversity	https://www.videouniversity.com/articles/dv-formats-everything-you-need-to-know/
3D	FBX (Filmbox)	FileFormat.com FBX Overview	https://docs.fileformat.com/3d/fbx/
3D	FBX (Filmbox)	Wikipedia FBX Entry	https://en.wikipedia.org/wiki/FBX
3D	FBX (Filmbox)	Viewing and Sharing FBX Files Online with Sibe - Sibe.io	https://sibe.io/3d-viewer/fbx
3D	FBX (Filmbox)	FBX - Taylor & Francis	https://taylorandfrancis.com/knowledge/Engineering_and_technology/Computer_science/FBX/
3D	FBX (Filmbox)	FBX - Autodesk	https://www.autodesk.com/products/fbx/overview
3D	FBX (Filmbox)	FBX (Filmbox) - Library of Congress	https://www.loc.gov/preservation/digital/formats/fdd/fdd000558.shtml

3D	FBX (Filmbox)	The 8 Best 3D File Formats in 2024 - The Pixel Lab	https://www.thepixellab.net/8-best-3d-file-formats
4D	FLAC (Free Lossless Audio Codec)	FLAC - Wikipedia	https://en.wikipedia.org/wiki/FLAC
4D	FLAC (Free Lossless Audio Codec)	Overview of lossless audio formats - erk.fe.uni-lj.si	https://erk.fe.uni-lj.si/2023/papers/zeleznikoverview_of.pdf
4D	FLAC (Free Lossless Audio Codec)	Lossless audio compression - pubs.aip.org	https://pubs.aip.org/aip/acp/article-pdf/doi/10.1063/1.5002024/13753864/020006_1_online.pdf
4D	FLAC (Free Lossless Audio Codec)	What is FLAC? - Lenovo	https://www.lenovo.com/us/en/glossary/what-is-flac/
4D	FLAC (Free Lossless Audio Codec)	FLAC - RFC Editor	https://www.rfc-editor.org/rfc/rfc9639.html
4D	FLAC (Free Lossless Audio Codec)	FLAC - Xiph.Org Foundation	https://xiph.org/flac/documentation_tools_flac.html
2D image	GIF	GIF Format: Past, Present, and Future - Cloudinary	https://cloudinary.com/guides/video-formats/gif-format-past-present-and-future
2D image	GIF	GIF - Wikipedia	https://en.wikipedia.org/wiki/GIF
2D image	GIF	GIF files: How to create, edit and open them - Adobe	https://www.adobe.com/creativecloud/file-types/image/raster/gif-file.html
2D image	GIF	GIF Graphics Interchange Format, Version 89a - Library of Congress	https://www.loc.gov/preservation/digital/formats/fdd/fdd000133.shtml
2D image	GIF	The Evolution of GIF Image Formats - LSoft Technologies	https://www.lsoft.net/posts/gif-evolution/
2D image	GIF	Graphics Interchange Format, Version 89a - W3C	https://www.w3.org/Graphics/GIF/spec-gif89a.txt
3D	glTF/GLB (GL Transmission Format)	glTF Mesh Features - Aspose.3D for.NET	https://docs.aspose.com/3d/net/developer-guide/meta-data/gltf-mesh_features/
3D	glTF/GLB (GL Transmission Format)	glTF Structural Metadata - Aspose.3D for.NET	https://docs.aspose.com/3d/net/developer-guide/meta-data/gltf-structural_metadata/
3D	glTF/GLB (GL Transmission Format)	Wikipedia glTF Entry	https://en.wikipedia.org/wiki/GlTF

3D	glTF/GLB (GL Transmission Format)	KhronosGroup/glTF - GitHub	https://github.com/KhronosGroup/glTF
3D	glTF/GLB (GL Transmission Format)	What is the GLB format? - Ikarus3D	https://ikarus3d.com/media/3d-blog/glb-and-gltf-files-purpose-difference-and-area-of-application-in-3d-modeling-services/
3D	glTF/GLB (GL Transmission Format)	Performance Analysis of GLTF/GLB to Improve 3D Content Rendering Performance - Journal of Platform Technology	https://koreascience.kr/article/JAKO202325443294224.view
3D	glTF/GLB (GL Transmission Format)	Khronos Group glTF(TM) 2.0 Specification - Metaverse Standards Register	https://register.metaverse-standards.org/spps/232
3D	glTF/GLB (GL Transmission Format)	glTF 2.0: A Runtime Asset Format for WebGL and Beyond	https://registry.khronos.org/glTF/specs/2.0/glTF-2.0.html
3D	glTF/GLB (GL Transmission Format)	glTF/GLB: GL transmission format is an open-source file format developed by the Khronos Group - Amazon	https://sellercentral.amazon.ca/help/hub/reference/external/G7RGSNQFZ2BAG7K3
3D	glTF/GLB (GL Transmission Format)	Your 3D Models on the Web - 8th Wall	https://www.8thwall.com/docs/legacy/guides/your-3d-models-on-the-web/
3D	glTF/GLB (GL Transmission Format)	Khronos Group glTF Overview	https://www.khronos.org/gltf/
3D	glTF/GLB (GL Transmission Format)	Library of Congress Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000498.shtml
4D	H.264 (AVC - Advanced Video Coding / MPEG-4 Part 10)	H.264 video encoding: How it works, benefits, and 9 best practices - Cloudinary	https://cloudinary.com/guides/video-formats/h-264-video-encoding-how-it-works-benefits-and-9-best-practices
4D	H.264 (AVC - Advanced Video Coding / MPEG-4 Part 10)	Advanced Video Coding - Wikipedia	https://en.wikipedia.org/wiki/Advanced_Video_Coding
4D	H.264 (AVC - Advanced Video Coding / MPEG-4 Part 10)	ITU H.264 - Vocal.com	https://vocal.com/video-codecs/itu-h-264/

4D	H.264 (AVC - Advanced Video Coding / MPEG-4 Part 10)	Libray of CongresH.264 file description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000081.shtml
4D	H.264 (AVC - Advanced Video Coding / MPEG-4 Part 10)	Video coding with H.264/AVC: tools, performance, and complexity - ResearchGate	https://www.researchgate.net/publication/3432389_Video_coding_with_H264AVC_tools_performance_and_complexity
2D image	HEIF (High Efficiency Image File Format)	RAW, JPEG and now HEIF - Canon	https://en.canon-cna.com/pro/infobank/image-file-types/
2D image	HEIF (High Efficiency Image File Format)	High Efficiency Image File Format - Wikipedia	https://en.wikipedia.org/wiki/High_Efficiency_Image_File_Format
2D image	HEIF (High Efficiency Image File Format)	Comprehensive Image Quality Assessment (IQA) of JPEG, WebP, HEIF and AVIF Formats - OSF	https://osf.io/ud7w4/download/?format=pdf
2D image	HEIF (High Efficiency Image File Format)	The format is an updated variant of the High Efficiency Image Format (HEIF) - Adobe	https://www.adobe.com/creativecloud/file-types/image/raster/heic-file.html
2D image	HEIF (High Efficiency Image File Format)	ISO/IEC 23008-12:2017 - Information technology -- High efficiency coding and media delivery in heterogeneous environments -- Part 12: Image File Format	https://www.iso.org/standard/83650.html
2D image	HEIF (High Efficiency Image File Format)	HEIF (HEIC) - LEADTOOLS	https://www.leadtools.com/help/sdk/v22/main/api/heif-heic.html
2D image	HEIF (High Efficiency Image File Format)	Here, we considered a modern HEIF coder applied to grayscale (component) images of different complexity corrupted by additive white Gaussian noise - MDPI	https://www.mdpi.com/2076-3417/15/6/2939
2D image	HEIF/HEIC	Library of Congress HEIF/HEIC Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000525.shtml
2D text	hOCR	OCR output format - Internet Archive Developers	https://archive.org/developers/ocr.html

2D text	hOCR	hOCR - Wikipedia	https://en.wikipedia.org/wiki/HOCR
	hOCR	GitHub hOCR Specification	https://github.com/kba/hocr-spec
2D text	hOCR	Public Specification for the hOCR Format - GitHub	https://github.com/kba/hocr-spec/blob/master/1.1/spec.md
2D text	hOCR	hOCR Export as HTML - IronSoftware	https://ironsoftware.com/csharp/ocr/how-to/html-hocr-export/
2D text	hOCR	hOCR tools library - PyPI	https://pypi.org/project/hocr-tools-lib/
2D text	hOCR	HOCR River Control - hocrvolunteer.org	https://rivercontrol.hocrvolunteer.org/roles-and-positions-overview/
	hOCR	Tesseract OCR	https://tesseract-ocr.github.io/
2D text	hOCR	The hOCR Microformat for OCR Workflow and Results - DFKI	https://www.dfki.de/fileadmin/user_upload/import/4373_The_hOCR_Microformat.pdf
2D text	hOCR	Create OCR-Processed PDFs In 2 Steps - Mindee Blog	https://www.mindee.com/blog/create-ocrized-pdfs-in-2-steps
2D text	hOCR	The hOCR Microformat for OCR Workflow and Results (PDF) - ResearchGate	https://www.researchgate.net/publication/232632963_The_hOCR_Microformat_for_OCR_Workflow_and_Results_PDF
2D image	JPEG	<img> - Mozilla	https://developer.mozilla.org/en-US/docs/Web/HTML/Reference/Elements/img
2D image	JPEG	JPEG.org – Official JPEG Committee	https://jpeg.org/jpeg/
2D image	JPEG	What is a JPEG file? - Shorthand	https://shorthand.com/the-craft/what-is-a-jpeg-file/index.html
2D image	JPEG	JPEG - Taylor & Francis Online	https://taylorandfrancis.com/knowledge/Engineering_and_technology/Computer_science/JPEG/
2D image	JPEG	Library of Congress JPEG Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000018.shtml
2D image	JPEG	JPEG JFIF - W3C	https://www.w3.org/Graphics/JPEG/
2D image	JPEG 2000	Is JPEG 2000 a Preservation Risk? - Library of Congress Blog	https://blogs.loc.gov/thesignal/2013/01/is-jpeg-2000-a-preservation-risk/
2D image	JPEG 2000	The JPEG2000 still image coding system: An overview - CiteSeerX	https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=3216b12f6c788e9a0289f797356abbc9f598f179
2D image	JPEG 2000	JPEG 2000 image compression - Analog Devices	https://www.analog.com/en/resources/analog-dialogue/articles/jpeg-2000-image-compression.html
2D image	JPEG 2000	Library of Congress JPEG 2000 Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000143.shtml

2D image	JPEG 2000	JPEG 2000 Image Compression Techniques: Advantages and Numeric Example - ResearchGate	https://www.researchgate.net/publication/387022628_JPEG_2000_Image_Compression_Techniques_Advantages_and_Numeric_Example
3D	MKV (Matroska Multimedia Container)	Matroska Media Container Format Specification - IETF	https://datatracker.ietf.org/doc/rfc9559/
3D	MKV (Matroska Multimedia Container)	Matroska Multimedia Container - Library of Congress	https://guides.lib.vt.edu/mkvformat/resources
3D	MKV (Matroska Multimedia Container)	MKV files - Adobe	https://www.adobe.com/creativecloud/file-types/video/container/mkv.html
3D	MKV (Matroska Multimedia Container)	Matroska - Library of Congress	https://www.loc.gov/preservation/digital/formats/fdd/fdd000342.shtml
4D	MOV (QuickTime File Format)	QuickTime Movie Files - Apple Developer	https://developer.apple.com/documentation/quicktime-file-format/quicktime_movie_files
4D	MOV (QuickTime File Format)	QuickTime File Format - Wikipedia	https://en.wikipedia.org/wiki/QuickTime_File_Format
4D	MOV (QuickTime File Format)	MOV (file format) - Restream.io	https://restream.io/learn/what-is/mov-file-format/
4D	MOV (QuickTime File Format)	QuickTime - Taylor & Francis Online	https://taylorandfrancis.com/knowledge/Engineering_and_technology/Computer_science/QuickTime/
4D	MOV (QuickTime File Format)	MOV files - Adobe	https://www.adobe.com/creativecloud/file-types/video/container/mov.html
4D	MOV (QuickTime File Format)	MOV file format description	https://www.fileformat.info/format/quicktime/egff.htm
4D	MOV (QuickTime File Format)	Libray of Congres MOV file description	https://www.loc.gov/marc/bibliographic/bd007m.html
4D	MP3 (MPEG Audio Layer III)	MP3 (MPEG Layer III Audio Encoding) - Library of Congress	https://www.loc.gov/preservation/digital/formats/fdd/fdd000012.shtml

4D	MP3 (MPEG Audio Layer III)	Library of Congress MP3 Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000105.shtml
4D	MP3 (MPEG Audio Layer III)	MP3 - MAGIX	https://www.magix.com/ca/music-editing/audio-formats/mp3/
4D	MP4 (MPEG-4 Part 14)	MP4 - Wikipedia	https://en.wikipedia.org/wiki/MP4_file_format
4D	MP4 (MPEG-4 Part 14)	MP4 files - Adobe	https://www.adobe.com/creativecloud/file-types/video/container/mp4.html
4D	MP4 (MPEG-4 Part 14)	What is MP4? - Cloudflare	https://www.cloudflare.com/learning/video/what-is-mp4/
4D	MP4 (MPEG-4 Part 14)	Libray of Congres MP4 file description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000037.shtml
4D	MP4 (MPEG-4 Part 14)	ISO Base Media File Format - Library of Congress	https://www.loc.gov/preservation/digital/formats/fdd/fdd000079.shtml
4D	MP4 (MPEG-4 Part 14)	MP4 File Format, Version 2 - Library of Congress	https://www.loc.gov/preservation/digital/formats/fdd/fdd000155.shtml
4D	MPEG (Moving Picture Experts Group) Video Codecs	MPEG1 Specifications - ICDIA	http://www.icdia.co.uk/cdprosupport/encoding/pink/mpeg1_specs.htm
4D	MPEG (Moving Picture Experts Group) Video Codecs	MPEG-2 - Wikipedia	https://en.wikipedia.org/wiki/MPEG-2
4D	MPEG (Moving Picture Experts Group) Video Codecs	What is a video codec? - IASA Journal	https://journal.iasa-web.org/index.php/pubs/article/view/51/26
4D	MPEG (Moving Picture Experts Group) Video Codecs	Libray of Congres MPEG file description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000080.shtml
4D	MPEG (Moving Picture Experts Group) Video Codecs	Libray of Congres MPEG File description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000338.shtml

4D	MPEG (Moving Picture Experts Group) Video Codecs	Standards - MPEG.org	https://www.mpeg.org/standards/
4D	MPEG (Moving Picture Experts Group) Video Codecs	The MPEG Immersive Video Standard—Current Status and Future Outlook - ResearchGate	https://www.researchgate.net/publication/363756498_The_MPEG_Immersive_Video_Standard-Current_Status_and_Future_Outlook
3D	OBJ (Wavefront Object)	What is OBJ format and extension? - CAD Exchanger	https://cadexchanger.com/obj/
3D	OBJ (Wavefront Object)	OBJ Converter Online (Free) - Convert 3D	https://convert3d.org/convert/obj
3D	OBJ (Wavefront Object)	Wavefront.obj file - Wikipedia	https://en.wikipedia.org/wiki/Wavefront_.obj_file
3D	OBJ (Wavefront Object)	Library of Congress Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000507.shtml
2D text	ODF	Library of Congress ODF Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000247.shtml
4D	Ogg Vorbis	Library of Congress Ogg Vorbis Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000117.shtml
4D	Ogg Vorbis	Xiph.Org Foundation official Documentation	https://xiph.org/vorbis/
4D	Ogg Vorbis	Ogg/Vorbis in embedded systems	https://www.researchgate.net/publication/277289148_OggVorbis_in_embedded_systems
4D	Ogg Vorbis	OGG Format: An In-Depth Look	https://cloudinary.com/guides/video-formats/ogg-format-an-in-depth-look
2D text	PDF	PDF file structure and creation - Apryse Blog	https://apryse.com/blog/pdf-file-structure-creation-part-2
2D text	PDF	PDF Association	https://pdfa.org/
2D text	PDF	Introduction to PDF Structure Extraction - Acodis Blog	https://www.acodis.io/blog/pdf-structure-extraction
2D text	PDF	Portable Document Format (PDF) Family - Library of Congress	https://www.loc.gov/preservation/digital/formats/fdd/fdd000030.shtml
2D text	PDF	Library of Congress PDF Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000125.shtml
2D text	PDF	PDF file format internal document structure explained - Save Emails as PDF	https://www.save-emails-as-pdf.com/news/pdf-file-format-internal-document-structure-explained/

3D	PLY (Polygon File Format / Stanford Triangle Format)	Maximizing Your 3D Workflow with PLY Format - Cloudinary	https://cloudinary.com/guides/digital-asset-management/maximizing-your-3d-workflow-with-ply-format
3D	PLY (Polygon File Format / Stanford Triangle Format)	Wikipedia PLY Entry	https://en.wikipedia.org/wiki/PLY_(file_format)
3D	PLY (Polygon File Format / Stanford Triangle Format)	PLY - Wikipedia	https://en.wikipedia.org/wiki/PLY_file_format
3D	PLY (Polygon File Format / Stanford Triangle Format)	ImageToSTL PLY Viewer	https://imagetostl.com/view-ply-online
3D	PLY (Polygon File Format / Stanford Triangle Format)	Introduction - Paul Bourke	https://paulbourke.net/dataformats/ply/
3D	PLY (Polygon File Format / Stanford Triangle Format)	Greg Turk's original PLY specification (Stanford)	https://people.sc.fsu.edu/~jburkardt/data/ply/ply.html
3D	PLY (Polygon File Format / Stanford Triangle Format)	Polygon File Format (PLY) Family - Library of Congress	https://www.loc.gov/preservation/digital/formats/fdd/fdd000501.shtml
2D image	PNG	PNGLib – PNG Resources	http://www.libpng.org/pub/png/
2D image	PNG	Research Paper PNG Transparent Images Free Download - Pngtree	https://pngtree.com/so/research-paper
2D image	PNG	Contemporary PNG Studies - Divine Word University	https://search.informit.org/journal/cpngs
2D image	PNG	Compressed image file formats: JPEG, PNG, GIF, XBM, BMP - Library of Congress	https://www.loc.gov/item/99015179/

2D image	PNG	Library of Congress PNG Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000153.shtml
2D image	PNG	W3C PNG Specification	https://www.w3.org/TR/PNG/
2D image	RAW	Guideline #2 - Saving the Original - Office of Research Integrity (HHS)	https://ori.hhs.gov/education/products/RIandImages/guidelines/guideline_2.html
2D image	RAW	What is a DNG file? - Adobe	https://www.adobe.com/creativecloud/file-types/image/raw/dng-file.html
2D image	RAW	JPEG vs. RAW: The Key Differences - Alan Ranger Photography Blog	https://www.alanranger.com/blog-on-photography/jpeg-vs-raw-the-key-differences
2D image	RAW	What are RAW camera image files? - CorelDRAW Blog	https://www.coreldraw.com/en/blog/raw-image/
2D image	RAW	Library of Congress RAW Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000241.shtml
2D image	RAW	Digital Negative (DNG) Specification - PRONOM - The National Archives	https://www.nationalarchives.gov.uk/PRONOM/Format/proFormatSearch.aspx?status=detailReport&id=1225
2D text	RTF	Rich Text Format (RTF) Specification, version 1.6 - SourceForge	https://latex2rtf.sourceforge.net/rtfspec.html
2D text	RTF	Rich Text Format (RTF) Compression Protocol Specification - Microsoft	https://msopenspecs.microsoft.com/files/MS-OXRTFCP/%5BMS-OXRTFCP%5D-101103.pdf
2D text	RTF	Why Build an RTF Parser? - TDGQ Blog	https://tdgq.com.au/structured-editing/why-build-an-rtf-parser/
2D text	RTF	Library of Congress RTF Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000473.shtml
2D text	RTF	Rich Text Format (RTF) - VA Technical Reference Model	https://www.oit.va.gov/Services/TRM/StandardPage.aspx?tid=5248
3D	STEP/STP (Standard for the Exchange of Product model data - ISO 10303)	ISO 10303-21 - Wikipedia	https://en.wikipedia.org/wiki/ISO_10303-21
3D	STEP/STP (Standard for the Exchange of Product model data - ISO 10303)	The 3DViewStation is the leading STEP Viewer and STP Viewer - Kisters	https://www.3dviewstation.com/stp-step-viewer-online.html

3D	STEP/STP (Standard for the Exchange of Product model data - ISO 10303)	Library of Congress Format Description (ISO 10303-21)	https://www.loc.gov/preservation/digital/formats/fdd/fdd000448.shtml
3D	STEP/STP (Standard for the Exchange of Product model data - ISO 10303)	CAD data exchange formats - Slideshare	https://www.slideshare.net/slideshow/cad-data-exchange-format-used-in-industry/265476301
3D	STEP/STP (Standard for the Exchange of Product model data - ISO 10303)	STP file - ZEISS	https://www.zeiss.com/metrology/en/explore/topics/stp-file.html
3D	STL (Stereolithography / Standard Triangle Language)	STL - Wikipedia	https://en.wikipedia.org/wiki/STL_file_format
3D	STL (Stereolithography / Standard Triangle Language)	The StL Format - fabbers.com	https://www.fabbers.com/tech/STL_Format
2D image	TIFF	TIFF - Wikipedia	https://en.wikipedia.org/wiki/TIFF
2D image	TIFF	Library of Congress TIFF Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000022.shtml
2D text	TXT	Writing READMEs for Research Data - Cornell University Library	https://data.research.cornell.edu/data-management/sharing/readme/
2D text	TXT	Text file - Wikipedia	https://en.wikipedia.org/wiki/Text_file
2D text	TXT	Character encoding: Types, UTF-8, Unicode, and more explained - Lokalise Blog	https://lokalise.com/blog/what-is-character-encoding-exploring-unicode-utf-8-ascii-and-more/
2D text	TXT	Describing research data with a README - University of Auckland	https://research-hub.auckland.ac.nz/article/research-data-readme
2D text	TXT	Plain text - Taylor & Francis Online	https://taylorandfrancis.com/knowledge/Engineering_and_technology/Computer_science/Plain_text/

2D text	TXT	How can I create a plain text document? - Digital Education Support at Brookes	https://telsupport.brookes.ac.uk/articles/how-can-i-create-a-plain-text-document/
2D text	TXT	TXT file and how to open, view and edit one - Adobe Acrobat	https://www.adobe.com/uk/acrobat/resources/document-files/text-files/txt.html
2D text	TXT	The simplest way to store information in computer memory is as a single file with a plain text format - The R Book	https://www.stat.auckland.ac.nz/~paul/ItDT/HTML/node38.html
2D text	TXT	The Unicode Standard - Unicode.org	https://www.unicode.org/versions/Unicode16.0.0/
3D	USD (Universal Scene Description)	Adobe's USD Plugins Are Officially Open-Source Now - 80.lv	https://80.lv/articles/adobe-s-usd-plugins-are-officially-open-source-now
3D	USD (Universal Scene Description)	To foster the standardization, development, evolution, and growth of OpenUSD - Alliance for OpenUSD	https://aousd.org/
3D	USD (Universal Scene Description)	Universal Scene Description (USD) in Unreal Engine - Unreal Engine Documentation	https://dev.epicgames.com/documentation/en-us/unreal-engine/universal-scene-description-usd-in-unreal-engine
3D	USD (Universal Scene Description)	USD - Universal Scene Description Format - docs.fileformat.com	https://docs.fileformat.com/3d/usd/
3D	USD (Universal Scene Description)	Wikipedia USD Entry	https://en.wikipedia.org/wiki/Universal_Scene_Description
3D	USD (Universal Scene Description)	Universal Scene Description (USD) - Library of Congress	https://loc.gov/preservation/digital/formats/fdd/fdd000561.shtml
3D	USD (Universal Scene Description)	OpenUSD Official Site	https://openusd.org/
3D	USD (Universal Scene Description)	USD FAQ and Docs (Pixar / OpenUSD)	https://openusd.org/release/usdfaq.html
3D	USD (Universal Scene Description)	Library of Congress Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000561.shtml
3D	USD (Universal Scene Description)	Universal Scene Description in Action - Manning Publications	https://www.manning.com/books/universal-scene-description-in-action-cx
3D	VRML (Virtual Reality Modeling Language)	Wikipedia VRML Entry	https://en.wikipedia.org/wiki/VRML

3D	VRML (Virtual Reality Modeling Language)	Blender Extensions for VRML/X3D	https://extensions.blender.org/add-ons/web3d-x3d-vrml2-format/
3D	VRML (Virtual Reality Modeling Language)	VRML Files - Stratasys Support Center	https://support.stratasys.com/SupportCenter/HTML5UserGuides/Objet260_UG_May_2022/Content/3_Topics_Introducing/VRML_Files_.htm
3D	VRML (Virtual Reality Modeling Language)	The Virtual Reality Modeling Language (VRML) is a language for describing multi-participant interactive simulations - web.cs.wpi.edu	https://web.cs.wpi.edu/~kal/elecdoc/java/vrml.html
3D	VRML (Virtual Reality Modeling Language)	Virtual Reality Modeling Language Family - Library of Congress	https://www.loc.gov/preservation/digital/formats/fdd/fdd000602.shtml?loclr=blogsig
3D	VRML (Virtual Reality Modeling Language)	Application of VRML in Distance Vocational Education - ResearchGate	https://www.researchgate.net/publication/45363398_Application_of_VRML_in_Distance_Vocational_Education
3D	VRML (Virtual Reality Modeling Language)	ISO/IEC 14772-1:1997 Standard (via Web3D Consortium)	https://www.web3d.org/x3d-vrml-most-widely-used-3d-formats
	WAV	Library of Congress WAV Format Description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000001.shtml
4D	WAV (Waveform Audio File Format)	Resource Interchange File Format - Wikipedia	https://en.wikipedia.org/wiki/Resource_Interchange_File_Format
4D	WAV (Waveform Audio File Format)	Understanding WAV Files: The Ultimate Guide to Audio Quality and File Management - Verbit.ai	https://verbit.ai/general/understanding-wav-files-the-ultimate-guide-to-audio-quality-and-file-management/
4D	WAV (Waveform Audio File Format)	Audio Compression on Multimedia Compression Techniques - ResearchGate	https://www.researchgate.net/publication/372186091_Audio_Compression_on_Multimedia_Compression_Techniques
4D	WAV (Waveform Audio File Format)	What is a WAV File? - Venicemusic.co	https://www.venicemusic.co/blog/what-is-a-wav-file

4D	WMV (Windows Media Video)	Windows Media Video - Wikipedia	https://en.wikipedia.org/wiki/Windows_Media_Video
4D	WMV (Windows Media Video)	Library of Congress wmv description format	https://www.digitalpreservation.gov/formats/fdd/fdd000091.shtml
4D	WMV (Windows Media Video)	Libray of Congres WMV file description	https://www.loc.gov/preservation/digital/formats/fdd/fdd000091.shtml
4D	WMV (Windows Media Video)	WMV vs. MP4 - Movavi	https://www.movavi.com/learning-portal/mp4-vs-wmv.html
4D	WMV (Windows Media Video)	WMV Video File Format: Definition, Uses, History - Vodpod	https://www.vodpod.com/video/file-formats/wmv/
3D	X3D (Extensible 3D)	X3D (Extensible 3D) - Wikipedia	https://en.wikipedia.org/wiki/X3D
3D	X3D (Extensible 3D)	Web3D Consortium X3D Resources	https://www.web3d.org/x3d/content/X3dResources.html
3D	X3D (Extensible 3D)	ISO/IEC 19776 Standard via Web3D Consortium	https://www.web3d.org/x3d/specifications/
3D	X3D (Extensible 3D)	X3D Tools and Resources - Web3D.org	https://www.web3d.org/x3d-tools-and-resources

IMPULSE

IMmersive digitisation: uPcycling cULtural heritage towards new reviving StratEgies

UNIWERSYTET JAGIELLOŃSKI W KRAKOWIE

KU LEUVEN

ALMA MATER STUDIORUM UNIVERSITÀ DI BOLOGNA

L-Università ta' Malta

Heritage Malta

FILMUNIVERSITÄT BABELSBERG KONRAD WOLF

HELLENIC REPUBLIC National and Kapodistrian University of Athens

MAGNA ŻMIEN

CLUST-ER CREATE CULTURA E CREATIVITÀ

K8

EXPLODEDVIEW

Funded by the European Union