

D4.C-D5.C - Primo rilascio di produzione dei servizi avanzati, 3D, AR e VR

Autori: Mauro Saccone (Roma TRE).

Revisori: Giovanna Spadafora, Marco Canciani (Roma Tre)

Sommario

Introduzione.....	2
1. Specifiche dei modelli di base.....	3
2. Specifiche dei modelli avanzati.....	4

Introduzione

A seguito delle soluzioni tecniche adottate, i visualizzatori 3D scelti per il progetto (3DHOP e ATON) integrano le funzionalità di AR e VR, pertanto non vi è stato uno sviluppo separato dell'ambiente visualizzatore 3D e dell'ambiente AR/VR ma il tutto è stato risolto con un unico motore 3D. Pertanto il task T4.4 e il T5.2 sono stati condotti congiuntamente e lo stesso è stato fatto per il T4.5 e per il task T5.3. Conseguentemente anche i deliverable previsti in fase di progetto sono stati accorpati: il D4.B con il D5.B e il D4.C con il D5.C. Dal punto di vista dello sviluppo software e dell'integrazione dei visualizzatori 3D tutte le informazioni ed indicazioni sono state inserite nei deliverable D3.B e D3.C.

In questo deliverable sono descritte le specifiche necessarie che devono avere i modelli 3D per essere inseriti all'interno della piattaforma.

La piattaforma D-TECH nasce per gestire copie digitali di beni culturali, modelli tridimensionali che devono essere caricati nel database "copia digitale" come descritto nel deliverable D3.C. La piattaforma prevede una funzione per l'upload di modelli, nelle due versioni, modelli di base e modelli avanzati (fig. 1). I primi vengono visualizzati con il motore 3DHOP, i secondi – che consentono anche l'uso della AR/VR – vengono visualizzati con ATON¹.

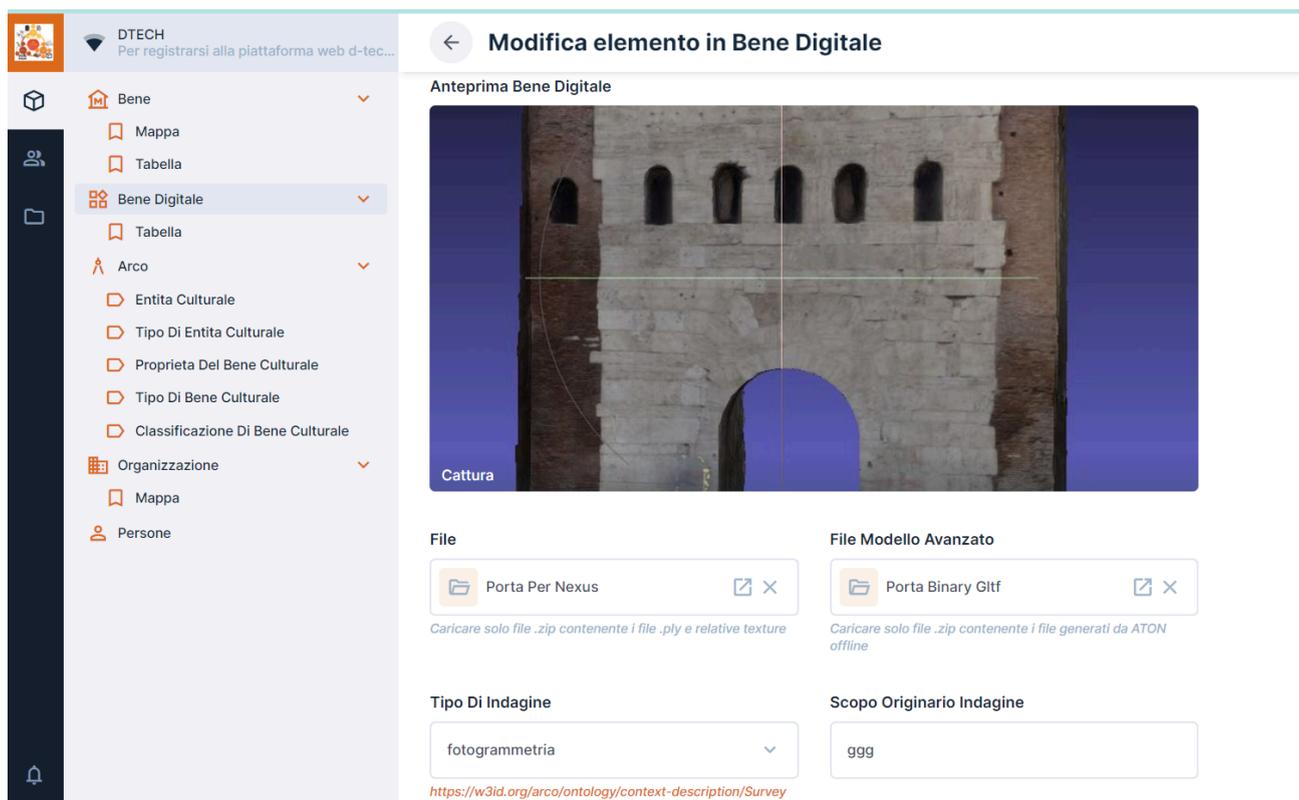


Fig.1- La pagina di caricamento delle copie digitali prevede due funzioni, una per il caricamento di modelli di base e una per i modelli avanzati.

¹ <https://osiris.itabc.cnr.it/aton/>

1. Specifiche dei modelli di base

All'interno della piattaforma possono essere caricate **due tipologie di modelli di base**, in funzione della loro complessità e dimensione in MB:

- I **modelli semplici** (al di sotto di 1 MB) sono modelli 3D standard, possono essere costituiti da mesh o da nuvole di punti, generalmente hanno un numero di triangoli, o di punti, contenuto e possono essere associati a una texture. Per la rappresentazione di questi modelli non viene utilizzata alcuna ottimizzazione durante il rendering. Il modello viene visualizzato "as is" dal motore 3DHOP.
- i **modelli multirisoluzione** (al di sopra di 1 MB) possono essere costituiti da mesh o da nuvole di punti, possono avere un numero di triangoli, o di punti, molto elevato e possono essere associati a una o più texture. Questi modelli multirisoluzione sono renderizzati in modo adattivo, in relazione alle esigenze dell'utente, e ottimizzati in base al punto di vista e alla distanza, bilanciando sempre la qualità del modello con la fluidità del movimento e della visualizzazione.

Per ottimizzare un modello del primo tipo, **modelli semplici**, dotato di una sola texture si consiglia una sua conversione in file PLY, utilizzando il software Open Source Meshlab². Il file dovrà essere salvato con le seguenti caratteristiche:

- binary encoding;
- no vertex normal;
- no face color/normal;
- wedge texture coordinates;

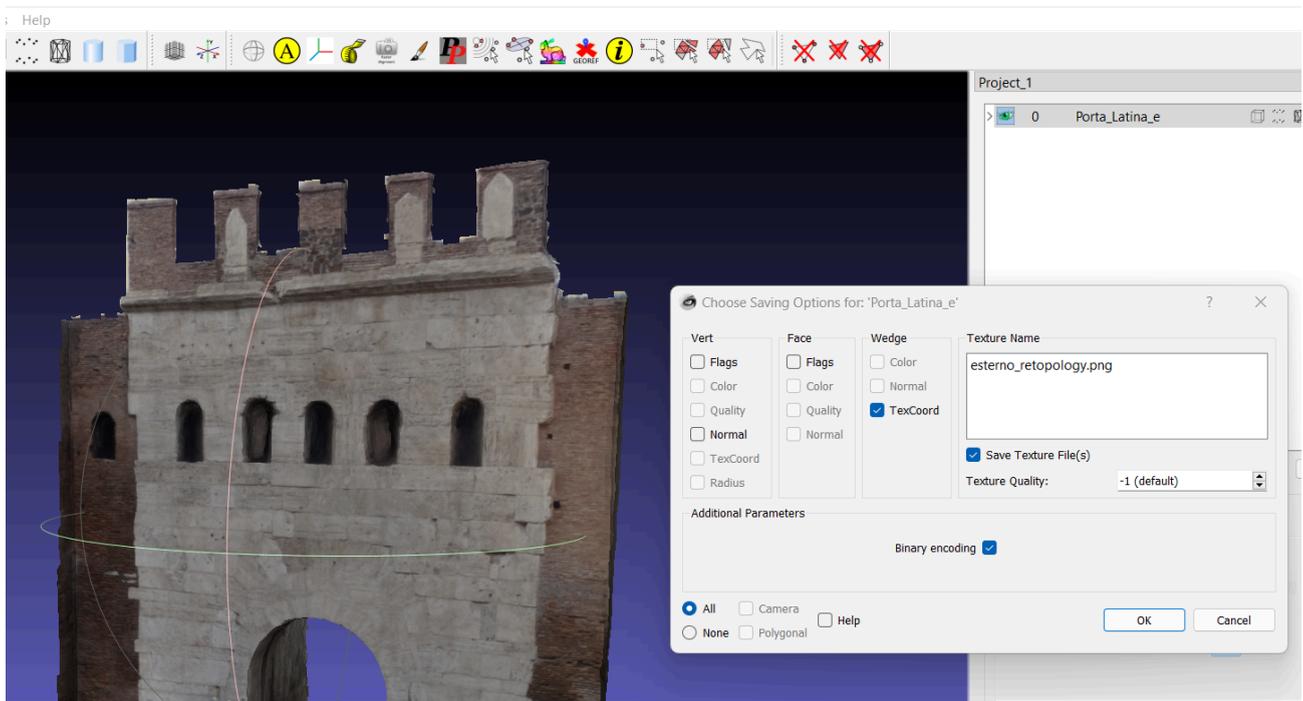


Fig.2- Caratteristiche dei modelli 3D.

² <https://www.meshlab.net/>

Per i modelli del secondo tipo, **modelli multirisoluzione**, la procedura è simile: richiede il passaggio attraverso il software Open Source Meshlab e una conversione in file PLY, ma le caratteristiche di questi modelli, oltre ai parametri già descritti per i modelli semplici, ovvero:

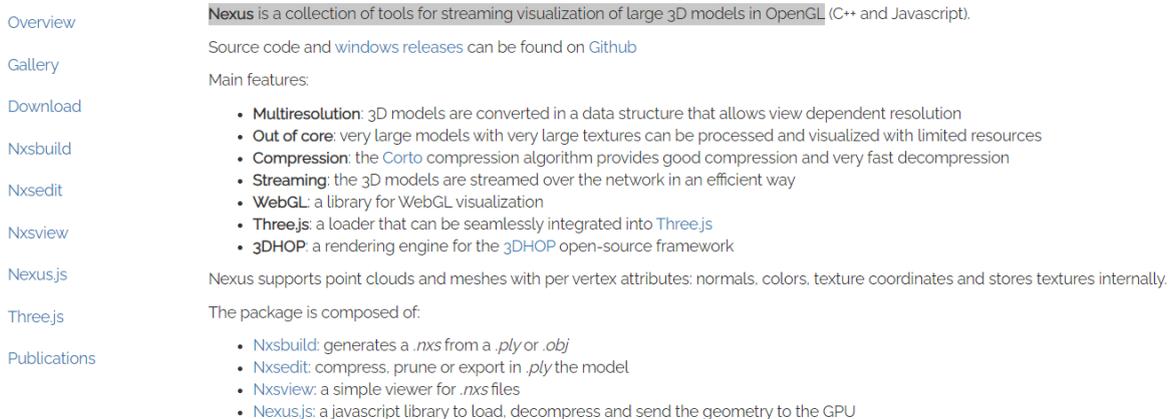
- binary encoding;
- no vertex normal;
- no face color/normal;
- wedge texture coordinates.

devono avere anche:

- texture PNG o JPG;
- e modalità di *Texture mapping* impostata *per-wedge* utilizzando il filtro di Meshlab che consente il passaggio da *per-vertex UV* a *per-wedge UV*;

Alla fine del processo si consiglia di utilizzare un convertitore di modelli 3D, Nexus³, che trasforma il modello e migliora le sue performance per lo streaming. Nexus è un software Open Source che contiene una raccolta di strumenti per la visualizzazione in streaming di modelli 3D di grandi dimensioni in OpenGL, attraverso la quale è possibile trasformare un modello 3D in un modello multi-risoluzione, e comprimere il file per renderlo più accessibile a tutti i dispositivi (fig. 3).

Nexus: adaptive 3D (C++ and Javascript)



Overview

Nexus is a collection of tools for streaming visualization of large 3D models in OpenGL (C++ and Javascript).

Source code and [windows releases](#) can be found on [Github](#)

Gallery

Download

Nxsbuid

Nxsedit

Nxsview

Nexusjs

Three.js

Publications

Main features:

- **Multiresolution:** 3D models are converted in a data structure that allows view dependent resolution
- **Out of core:** very large models with very large textures can be processed and visualized with limited resources
- **Compression:** the *Corto* compression algorithm provides good compression and very fast decompression
- **Streaming:** the 3D models are streamed over the network in an efficient way
- **WebGL:** a library for WebGL visualization
- **Three.js:** a loader that can be seamlessly integrated into *Three.js*
- **3DHOP:** a rendering engine for the *3DHOP* open-source framework

Nexus supports point clouds and meshes with per vertex attributes: normals, colors, texture coordinates and stores textures internally.

The package is composed of:

- *Nxsbuid:* generates a *.nxs* from a *.ply* or *.obj*
- *Nxsedit:* compress, prune or export in *.ply* the model
- *Nxsview:* a simple viewer for *.nxs* files
- *Nexusjs:* a javascript library to load, decompress and send the geometry to the GPU

Fig.3- Caratteristiche principali del software NEXUS.

2. Specifiche dei modelli avanzati

La piattaforma può ospitare anche modelli avanzati, costituiti da mesh o nuvole di punti, modelli che hanno una complessità paragonabile a quelli multirisoluzione precedentemente descritti e che possono essere utilizzati nella piattaforma, usando strumenti come l’annotazione semantica o la visualizzazione in AR/VR. A caratterizzare questi modelli, che saranno visualizzati attraverso il motore 3D ATON, è lo standard utilizzato per la loro definizione, il GL Transmission Format (.glTF e .glb), un formato di file poligonale che

³ <https://vcg.isti.cnr.it/nexus/>

rappresenta la controparte Open Source, e quindi più sostenibile, dei formati proprietari come ad esempio l'FBX. Originariamente creato dal gruppo di lavoro COLLADA, questo formato di file aperto è ora gestito dal gruppo KROHON e supporta modelli statici, animazioni e scene in movimento, esattamente come il formato FBX, ma è quello più utilizzato dagli sviluppatori nelle applicazioni web native. Essendo uno dei formati di file più moderni, supporta i metodi più recenti per l'ombreggiatura e la definizione dei materiali e, a partire dalla versione aggiornata 2.0, supporta materiali di rendering avanzati (PBR - Physically based rendering).

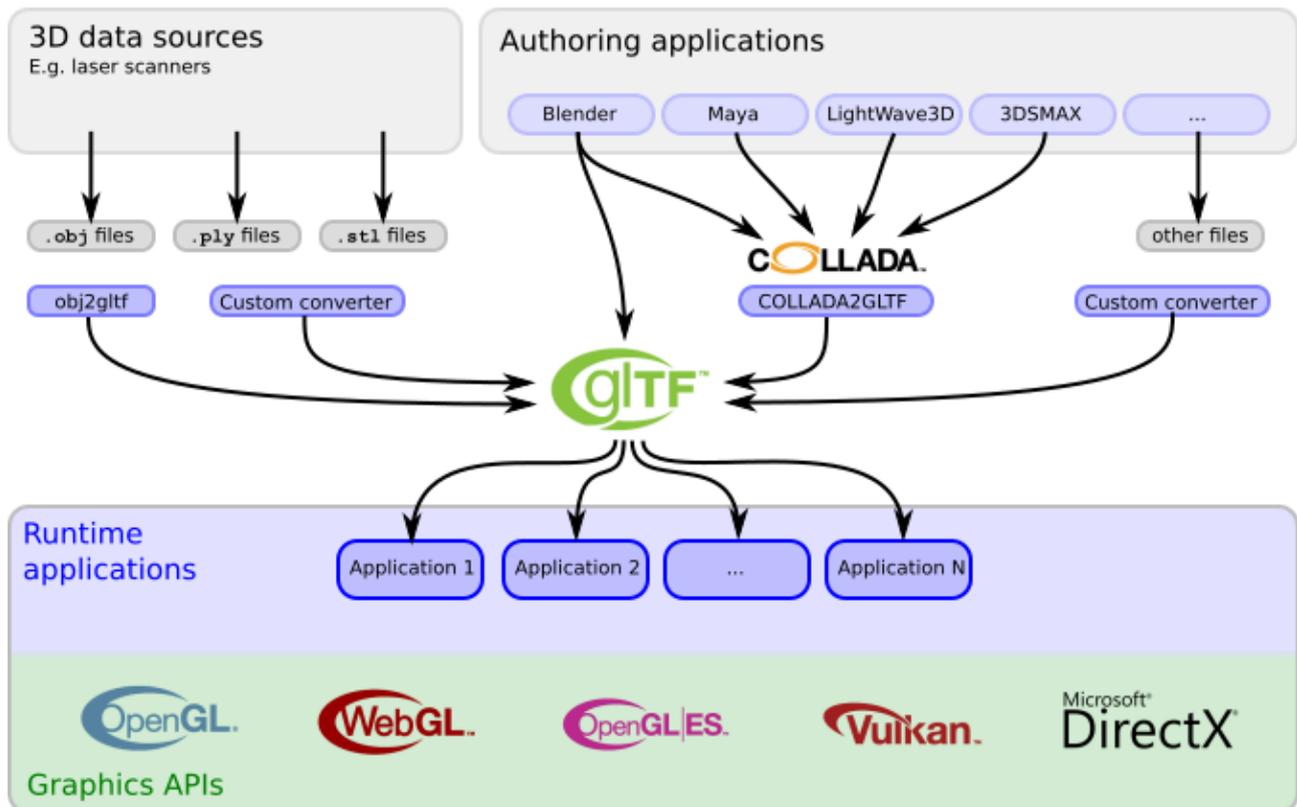


Fig.4- Schema di conversione dei principali formati di modelli 3D in glTF.

I file glTF possono essere realizzati a partire da OBJ o PLY, seguendo semplici regole (fig. 4) ben descritte nel tutorial dedicato al glTF su github⁴, oppure, in alternativa, utilizzando il software open source Blender⁵ che importa mesh e nuvole di punti ed esporta i file nel formato glTF.

⁴ <https://github.com/KhronosGroup/glTF-Tutorials/blob/main/glTFTutorial/README.md>

⁵ <https://www.blender.org/>