I3S: Support in open frameworks

Tamrat Belayneh, Snr. Software Engineer

2021 ESRI
DEVELOPER SUMMIT

Agenda

- I3S: Evolution and Standardization efforts
- I3S: Support in open visualization framework: loaders.gl & deck.gl
- I3S: 3DTiles ⇔ I3S Conversion
- I3S: Support of I3S 1.7 in Cesium.JS
- I3S: PCSL to LAS conversion using pdal.io

I3S | Evolution

- First OGC community standard adopted for storage and transmission of large, heterogeneous 3D geospatial data sets

- Adopted as Indexed 3D Scene Layer (I3S) Community Standard
  - https://www.ogc.org/standards/i3s
  - https://github.com/Esri/i3s-spec

- Designed to be Cloud, Web and Mobile friendly based on modern web standards and best practices

- Point Cloud Scene Layer (PCSL) adopted by OGC as I3S 1.1 Community Standard in Feb. 2020

I3S | Supports Multiple Profiles and Layer Types

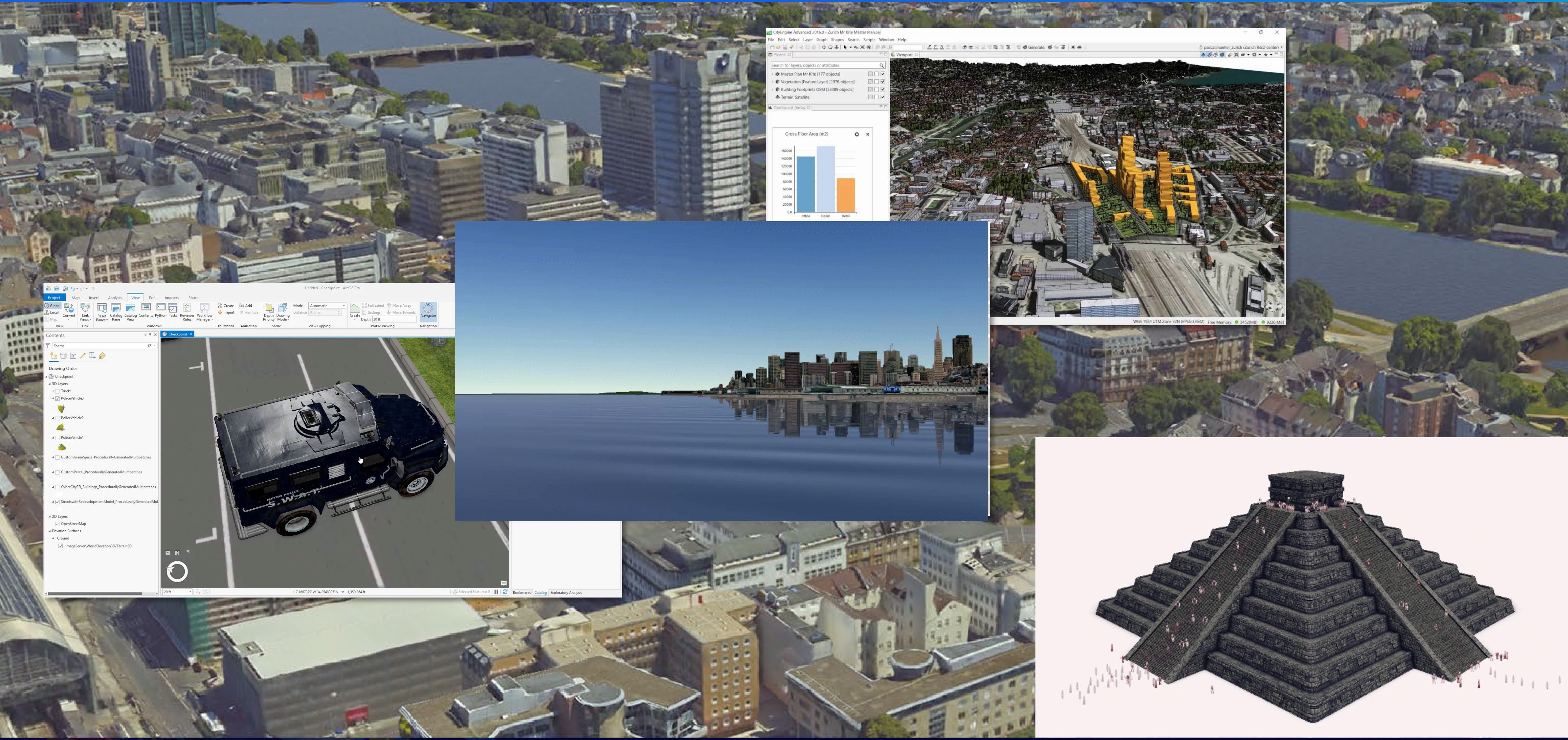
- I3S defines a concept of Profile and Layer types

Layer Type	Profile	Attributes
3D Object	Mesh Pyramids	Yes
Integrated Mesh	Mesh Pyramids	Triangle attributes
Point	Point	Yes
Point Cloud	PointCloud	Yes
Building	Building	Yes

OGC Approved Standard

Newly Released

I3S | Supports Enhanced Visual Aesthetics



I3S | I3S 1.7 under OGC adoption consideration

https://github.com/Esri/i3s-spec

Version I3S version 1.7 brings increased Performance for 3D Object and Integrated Mesh Layers Via:

- Enhanced visual aesthetics by supporting advanced materials
- Paged Node access pattern – by significantly reducing client-server traffic for Node content
- More compact Geometry – supports using Draco compression for the binary geometry content
- More optimal Selection Strategy - standardizes on Oriented Bounding Boxes (OBBs) for node selection

Easy to use converter released as part of the Spec update

I3S | Supported by various content providers

Bentley
CONTEXT CAPTURE
nearmap
PIX4D
Skyline
VRICON
Agisoft
nFrames
EMPOWERING PHOTOGRAMMETRY
melown
TECHNOLOGIES
aerometrex.
Unfolded
Vis.gl
易智瑞
GeoScene

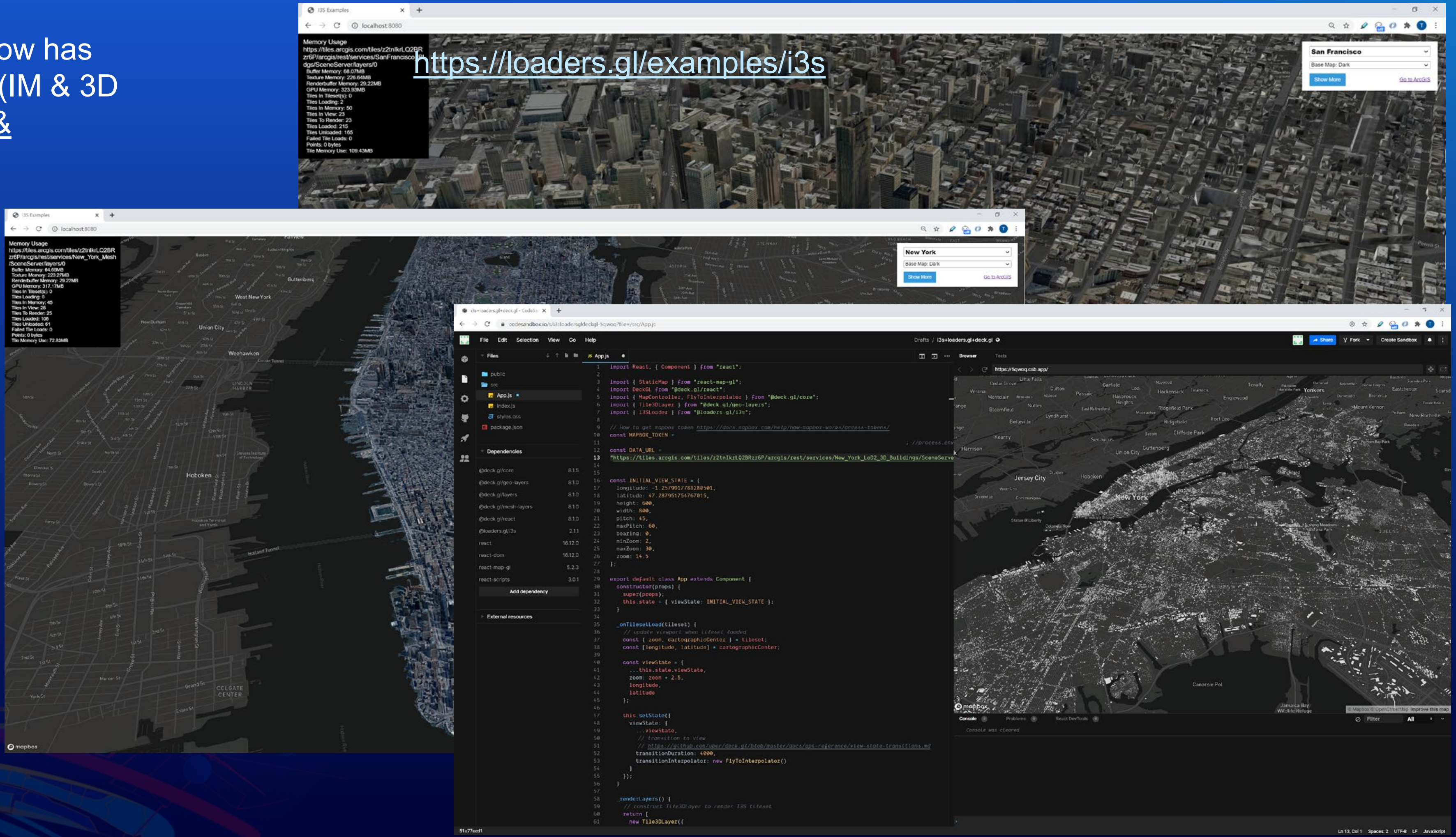
I3S | Support from more open source solutions

The Vis.gl open framework now has support for I3S consumption (IM & 3D Object layers) using deck.gl & loaders.gl modules.

Vis.gl

Frameworks for WebGL-powered large-scale data visualization

https://loaders.gl/examples/i3s



I3S | 3DTiles ⇔ I3S Conversion

- A loaders.gl based converter between I3S and 3D tiles tilesets
- Supports 3DTiles Batched3DModel ⇔ I3S IntegratedMesh Scene Layer
- 3DTiles Batched3DModel + Hierarchy extension ⇔ I3S 3D Object Scene Layer
- Ability to batch convert a 3D Tiles content to a Scene Layer Package (SLPK) and Indexed 3D Scene Layer REST (i3sREST)
- Allows conversion directly from cesium ion - via its REST api as well as a file-based 3D Tiles content

I3S | 3DTiles ⇔ I3S Conversion cont'd

- Ability to batch convert an I3S 1.7 in Scene Layer Package (SLPK) and Indexed 3D Scene Layer REST (i3sREST) format to 3D Tiles
- Support for: 3DTiles version 1.0, embedded glTF version 2.0
- Coordinates: supports WGS84 as the only supported 3D Tiles CRS
- Handles correct gravity models by converting between orthometric (i3s preferred height model) to ellipsoidal (cesium preferred)

I3S | 3DTiles ⇔ I3S Conversion cont'd

- ***The converter is delivered in 3 options and runs in both Linux and Windows***

1. ***As a CLI module:***
   - *node converter.min.js --input-type 3dtiles --tileset ../Frankfurt-3d-tiles/cesiumJpg/tileset.json --name Frankfurt_completed_bundle --output data --max-depth 6 --slpk --egm /home/user/pgm/egm2008-5/geoids/egm2008-5.pgm*
   - *node converter.min.js --input-type 3dtiles --tileset https://assets.cesium.com/29328/tileset.json --name CairoLayer --egm /home/user/pgm/egm2008-5/geoids/egm2008-5.pgm –slpk*
   - *node converter.min.js --input-type i3s --tileset https://tiles.arcgis.com/tiles/z2tnIkrLQ2BRzr6P/arcgis/rest/services/Rancho_Mesh_mesh_v17_1/SceneServer/layers/0 --name Rancho --max-depth 4 --egm /home/user/pgm/egm2008-5/geoids/egm2008-5.pgm*

Requirements:

1. NodeJs
2. location of Earth Gravity Model *.pgm file to convert heights from ellipsoidal to gravity-related format. A model file can be downloaded from GeographicLib https://geographiclib.sourceforge.io/html/geoid.html
3. External archiver (for slpk mode):
   1. Linux/Ubuntu: apt install zip
   2. Windows: 7-Zip. Default 7-zip location is "C:\Program Files\7-Zip\7z.exe" but there is option "--7zExe" that can be used for setup "7z.exe" location manually.

I3S | 3DTiles ⇔ I3S Conversion cont'd

2. As an npm package using npx:

```
npx converter --input-type <I3S | 3DTILES> --tileset <tileset> --name <tileset name> [--output <output folder>] [--no-draco] [--max-depth 4] [--slpk] [--7zExe <path/to/7z.exe>] [--token <ION token>] [--egm <path/to/*.pgm>]
```

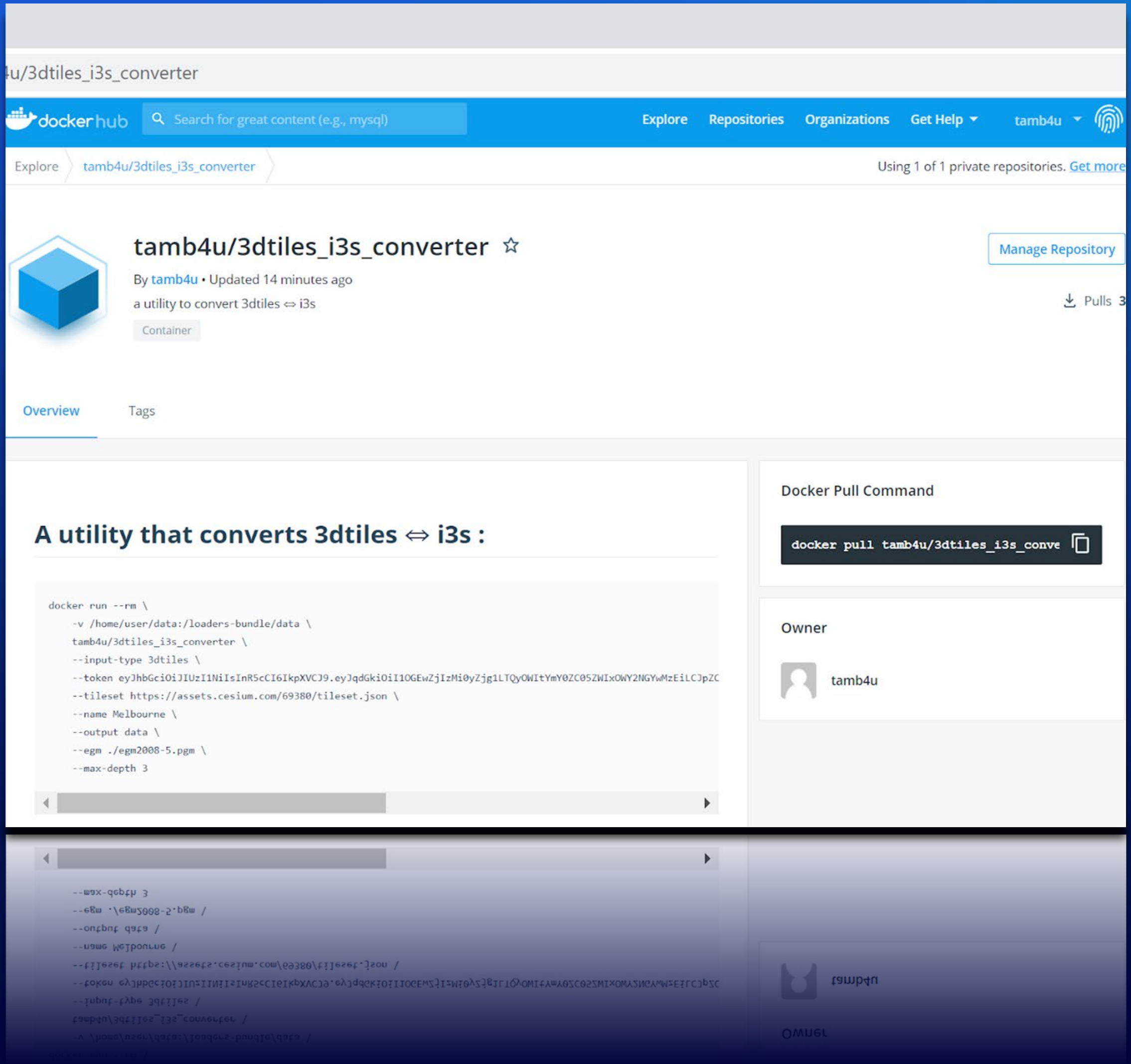
Requirements:

1. NodeJs
2. location of Earth Gravity Model *.pgm file to convert heights from ellipsoidal to gravity-related format. A model file can be downloaded from GeographicLib https://geographiclib.sourceforge.io/html/geoid.html
3. External archiver (for slpk mode):
    1. Linux/Ubuntu: apt install zip
    2. Windows: 7-Zip. Default 7-zip location is "C:\Program Files\7-Zip\7z.exe" but there is option "--7zExe" that can be used for setup "7z.exe" location manually.

I3S | 3DTiles ⇔ I3S Conversion cont'd

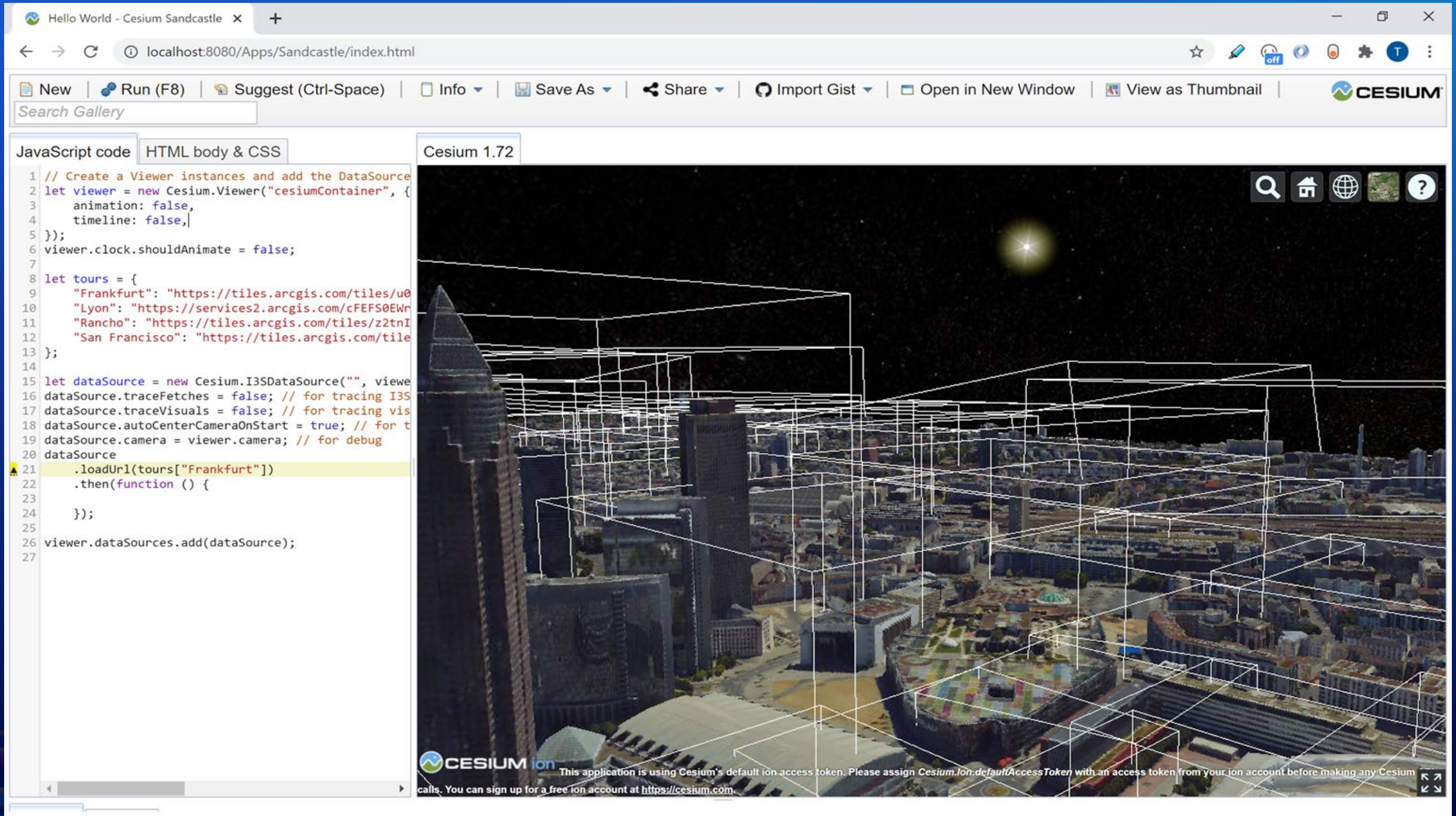
- 3. As a docker image:

```
docker run --rm \
    -v /home/user/data:/loaders-bundle/data \
    tamb4u/3dtiles_i3s_converter \
    --input-type 3dtiles \
    --token eyJhbGciOiJIUzI1NiIsInR5cCI6I...
    --tileset https://assets.cesium.com/69380/tileset.json \
    --name Melbourne \
    --output data \
    --egm ./egm2008-5.pgm \
      --max-depth 3
```

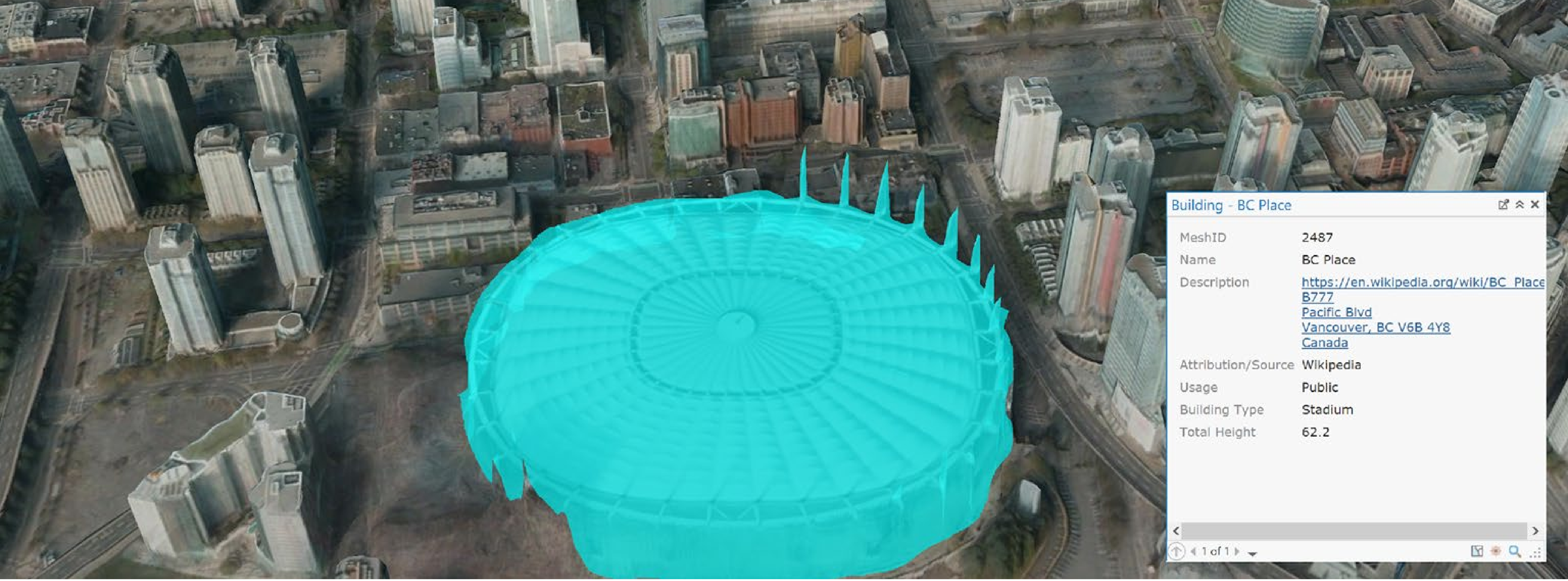


I3S | Support of I3S 1.7 in Cesium.JS

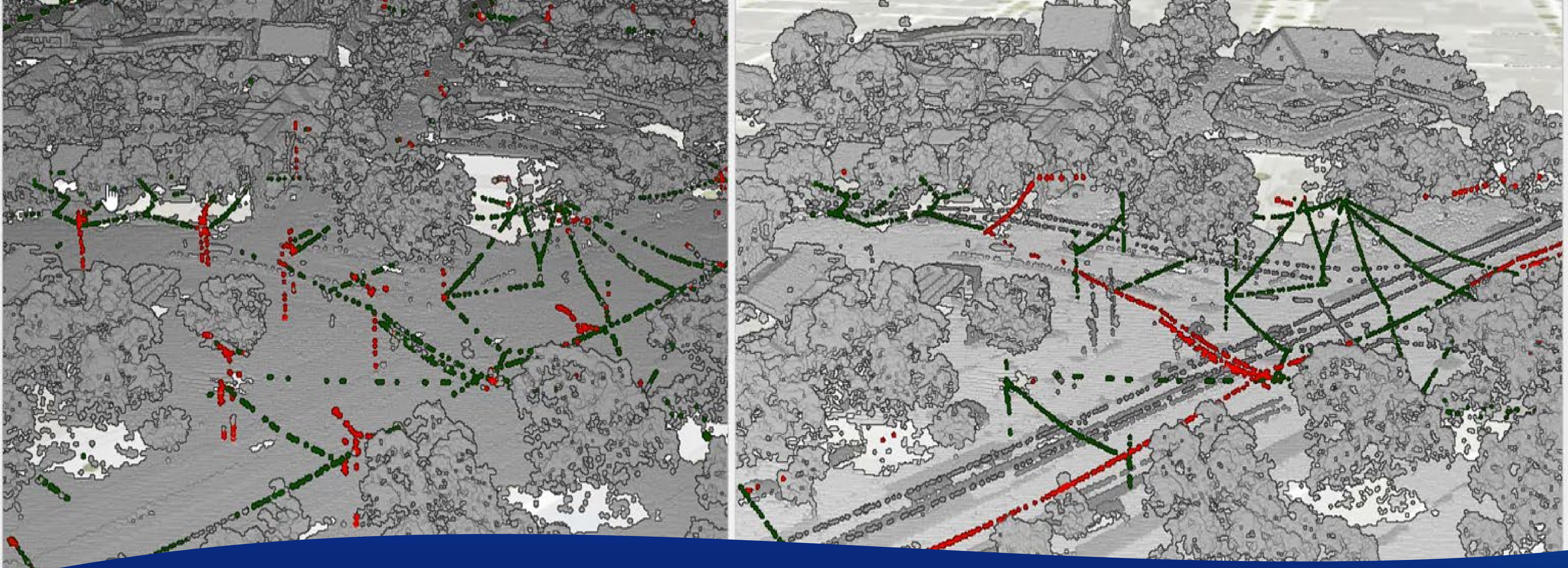
A module for reading and viewing I3S leveraging CesiumJS framework. Supporting IntegratedMesh and 3D Object Scene Layers in I3S 1.7 format



I3S | AI and Deep Learning



Use Artificial Intelligence (AI) and Deep Learning (DL) to segment Point Cloud and Mesh data



I3S: AI and Deep Learning ...

I3S - well suited for AI and Deep Learning manipulation:

- Automatic mesh segmentation of an Integrated Mesh
- Point cloud classification
- We are working on providing tooling to create, train and predict (classify/segment) point cloud and mesh data using various neural networks
- In March 2020, the arcgis.learn module added PointCNN, to efficiently classify and segment points from a point cloud dataset

Demo: I3S in Open frameworks

esri

THE SCIENCE OF WHERE®

Please provide your feedback for this session by clicking on the session survey link directly below the video.