Scene Services and the Open Indexed 3D Scene Format

Thorsten Reitz (Esri R&D Center Zurich)
Chris Andrews (Esri Inc.)
Agenda
An in-depth view at the Scene Services REST API and the Indexed 3D Scene Format

1. Web 3D as part of the ArcGIS Platform
2. Indexed 3D Scenes (i3s) & Scene Services REST API
3. i3s Persistence options
4. The i3s developer’s toolkit
5. Partner implementations
Web 3D
A new Capability across the ArcGIS platform
A New Way of Working in 3D

3D across the platform
How to create Scene Services?

1. Geodatabase
2. *.obj, CityGML
3. Other Formats
4. 3rd Party Software
5. ArcGIS Online/Portal
6. ArcGIS PRO
7. ArcGIS Online/Portal
8. ArcGIS Server
9. ArcGIS Server
10. ArcGIS Online/On Premise Portal
Detailed Publishing Workflow: ArcGIS Pro + Server

Feature Layer → Publish → Scene Layer

Feature Service → Publish → Scene Service

Geodatabase → Publish → CouchDB/FS/Package
Publishing from ArcGIS Pro
Scene Services in Web Scenes
Demonstration
Publishing & Using Scene Services
Indexed 3D Scenes
A new open format for multiscale geospatial 3D data
Requirements for a 3D GI visualization format

1. **Web friendly:** JSON + Typed Arrays
2. **Mobile friendly:** Works good with varying bandwidth
3. **Extensible:** Support different types of content
4. **Declarative:** Reduce required implicit knowledge
5. **Efficient:** Use spatial indexing for quick delivery
6. **Scalable:** Provide Level of Detail Support
7. **Protected:** Ensure that content is protected from theft
8. **Open:** Full Specification publicly accessible
Content Profiles
Support different geometry types through subsetting i3s

- Individual Features
  - Points
  - Lines
  - Polygons
  - Multipatches
- Integrated Meshes
- Analytics
- Pointclouds
- Symbols
The Key to Scalability: Indexing

- Adapt Index type to data
  - R-Tree
  - Quadtree, Octtree
  - Standard Tiling Schemes

- Load-Balanced
  - Near-Constant Data Volume per Node
### Level of Detail

- **Feature Switching LoD**
  - *Sparse Tree* with Feature-based Replacement
  - Use for:
    - Authored LoD (CityGML)

- **Node Switching LoD**
  - *Full Representation* Pyramid with Node-based Replacement
  - Use for:
    - Homogeneous contextual data (Integrated Meshes)

---

**Capacity-driven texture & geometry reduction**
Scene Services Resources & REST API

- Service
  - Layer
    - Symbols
    - Node
      - Features
      - Geometry
      - Texture
      - Shared Resource

1 Node Index Document

1 Feature Data
1..* Geometry Files
0..* Texture Files
1 Shared Resource

- SceneServer
  - /SceneServer
- /SceneServer/layers
- /SceneServer/layers/0/symbols/0
- ...layers/0/nodes/0
- ...layers/0/nodes/0/features/0
- ...layers/0/nodes/0/geometry/0
- ...layers/0/nodes/0/textures/1_2
- ...layers/0/nodes/0/shared

→ SceneServiceInfo.json
→ 3dSceneLayer.json
→ FeatureData.json
→ NodeIndexDocument.json
→ FeatureData.json
→ Geometry Typed Array
→ Image with Texture LoD
→ sharedResource.json
"Streaming"-friendly resource structure

**Node 3-3-4-1, 10 MB capacity**

- **Feature Data**
  - Doc. 0
  - Doc. 1
  - Doc. n

- **Geometry Data**
  - TypedArray 0
  - TypedArray 1
  - TypedArray n

**Shared Resources**

- **Texture Set**
  - 0_0
  - 1_0
  - n_0
  - 0_1
  - 1_1
  - n_1

**Bundle 0, <1MB**
- Feature Data
- Geometry Data

**Bundle 1, <1MB**
- Feature Data
- Geometry Data

**Bundle n, <1MB**
- Feature Data
- Geometry Data

- Contains a subset (1..*) of all features belonging to the node.
- Set of 1..* Textures for LOD 0 (Full resolution).
- Texture LOD 1 (Half resolution).

**EXAMPLE Size:**
- 40KB
- 140KB
- 250KB
- 70KB
Declarative: Geometry buffer metadata

```
"defaultGeometrySchema": { // geometry resource layout for nodes that declare the use of defaultGeometrySchema in the node index.
  "header": [ // header fields that precede the vertex data
    {
      "property": "vertexCount", // vertex count
      "type": "UInt32"
    }
  ],
  "topology": "PerAttributeArray", // one of ["PerAttributeArray", "InterleavedArray", "Indexed"]. When "Indexed", the indices must also be declared in the geometry.
  "vertexAttributeOrder": ["position", "normal", "uv0", "region"], // provides the order of the keys in vertexAttributes.
  "vertexAttributes": [ // the vertex attributes must appear in the order that they are declared here.
    "position": { // the name of the vertex attribute; here: vertex positions
      "valueType": "Float32", // the element type, either UInt8, UInt16, UInt32, Int16, Int32, Int64 or *Float32*, Float64
      "valuesPerElement": 3 // number of (Float32) values need to make a valid element (here a xyz position)
    },
    "normal": { // the name of the vertex attribute; here: vertex normals
      "valueType": "Float32", // the element type, either UInt8, UInt16, UInt32, Int16, Int32, Int64 or *Float32*, Float64
      "valuesPerElement": 3 // number of (Float32) values need to make a valid element (here a xyz position)
    }
  ]
}
```
i3s persistence options
Storing and serving i3s resources
File System Folder Layout

Direct mapping of the REST API

- All resources reside in the file system as individual files.
- These files are organised in folders in the following schema:

/3dSceneLayer.json
/nodes/root/3dNodeIndexDocument.json
/nodes/root/features/0.json ... n.json
/nodes/root/geometries/0.bin ... n.bin
/nodes/root/shared/SharedResource.json
/nodes/root/textures/0_0.bin ... n_m.bin
CouchDB – JSON Object Storage

Apply liberally with IndexedDB or other KV-Stores
i3p – the i3s package format
Best for single-file exchange and portability of i3s stores
The i3s toolkit
Basic tools to build i3s-based services and apps
Scene Service & i3s Specifications
License: Creative Commons – Attribution – No Derivatives 3.0

- Scene Service REST API Specification
- Indexed 3D Scene Format Specification
- Validation Rules
- Documentation
- Examples (8x)
I3s Rulesets
Formal Definition of the specification

- Modularity
- Structure
  - Type
  - Pattern
  - Cardinality
  - Nullable
- Domain Restrictions
- Dependencies
  - "Apply a rule only if this set of conditions is met"
Validators
Structure + content validation for JSON documents

- Standalone web application
- Node.js application
- Embedded into grunt build process

- Single Resource Validator
  - *Schema validation* of a single file
- Store Validator
  - *Consistency validation* of a full or partial store
Node.js Utilities

Helpful tools to create the specification and work with i3s content

- **Working with services:**
  - harvest.js: Download a published Scene Service from its REST API and store on disc
  - packi3p.js: Create an i3p file from a cache on disc
  - unpacki3p.js: Unpack an i3p file to disc

- **Working with i3s rulesets:**
  - schemagen.js: Generate a ruleset from an example
  - multischemamerge.js: Perform an additive ruleset merge on a folder of ruleset files
  - difftool.js: Find differences between two rulesets
  - docgen.js: HTML documentation from ruleset
Partner Implementations
How to become part of ArcGIS Web Scenes
Vision
An Open Specification for the Geospatial Community

• „The“ visualisation format for any kind of Geospatial 3D Data
  - Scalable
  - Web friendly
  - Declarative
  - Full feature data
  - Visualisation friendly

• i3s satisfies a basic infrastructure need of the GIS community

• Most valuable if implemented everywhere

• Lower possible hurdles to implementation
Roadmap for working with Partners
Current & planned process

• Phase 1
  - Rapid spec evolution with small set of partners
  - Performance Optimization
  - Clean Up
• Phase 2
  - Slower spec evolution with full public participation
  - New Format Features

• Collaboration through Github
  - „caniuse“ of features of each known implementation
Get Involved with Scene Services!
Implement Encoders, Services, Clients

- **Specification License**: Creative Commons-Attribution-No Derivatives 3.0
- **Toolkit License**: Apache
- **Specification and Toolkit available upon request for early adopters**
  - Ask us at
    - treitz@esri.com
    - candrews@esri.com
Thank you...
Any Questions? Ideas? Comments?

Esri.com/RateMyDevSummitSession

Scene

Esri International Developer Summit

Session Evaluation
Find a Session
Enter session name

Scene

Scene Services and the Open Indexed 3D Scene Format [Santa Rosa] (Wednesday)
How does the 3D Scene Viewer Work? [Mesquite GH] (Thursday 01:00 PM)
Understanding our world.