



Scene Graph Adapter

An efficient Architecture to Improve Interoperability
between 3D Formats and 3D Applications Engines



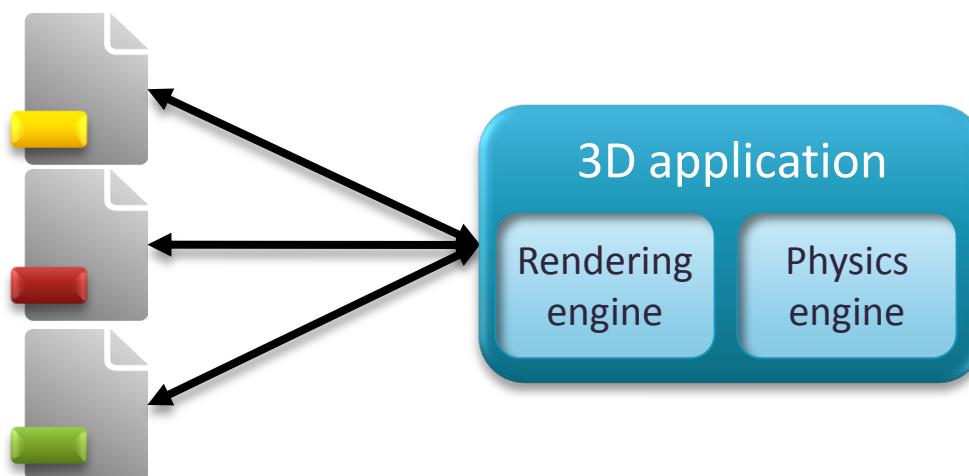
Rozenn Bouville Berthelot - Jérôme Royan (*Orange Rennes, France*)
Thierry Duval - Bruno Arnaldi (*IRISA Rennes, France*)

Scene Graph Adapter - Purpose



Scene Graph Adapter - Purpose

- Goal :
 - mix several 3D formats in a single 3D application
- Requirements :
 - preventive against **functionality** loss
 - extensible to most existing or coming **3D formats**
 - compliant with most existing or coming **3D application engine**



Issue of 3D format mixing

- Mix several 3D formats in a single 3D application, why ?

1. many, many 3D formats

3D modeler formats



3D renderer formats



consortium formats



Issue of 3D format mixing

- Mix several 3D formats in a single 3D application,
why ?

1. many, many 3D formats
2. reuse 3D models



Issue of 3D format mixing

- Mix several 3D formats in a single 3D application, why ?

Now

3D
application
1

3D
application
2

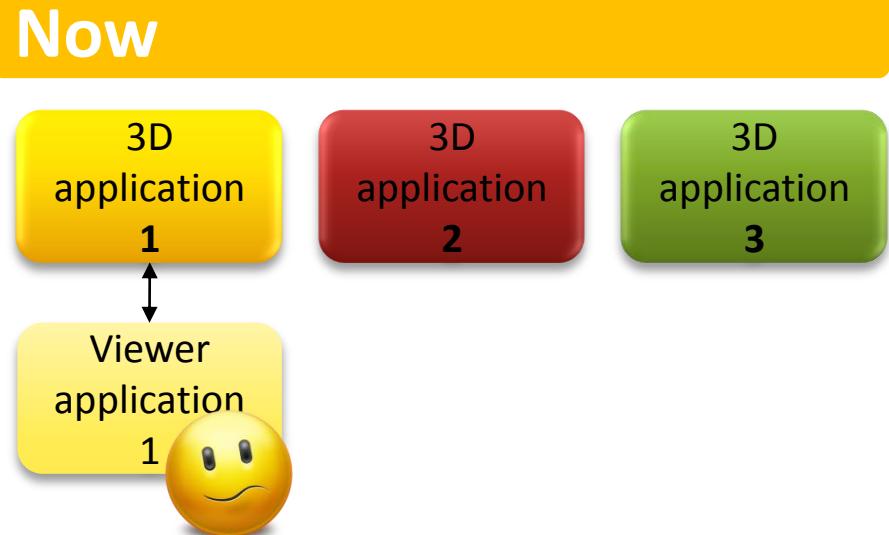
3D
application
3

- many, many 3D formats
- reuse 3D models
- increase user access to 3D environments**

Issue of 3D format mixing

- Mix several 3D formats in a single 3D application, why ?

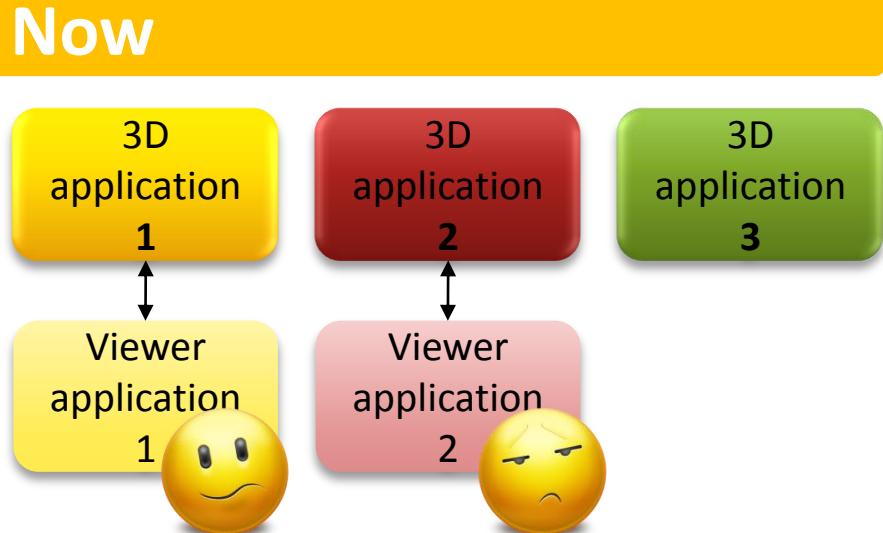
- many, many 3D formats
- reuse 3D models
- increase user access to 3D environments**



Issue of 3D format mixing

- Mix several 3D formats in a single 3D application, why ?

- many, many 3D formats
- reuse 3D models
- increase user access to 3D environments**

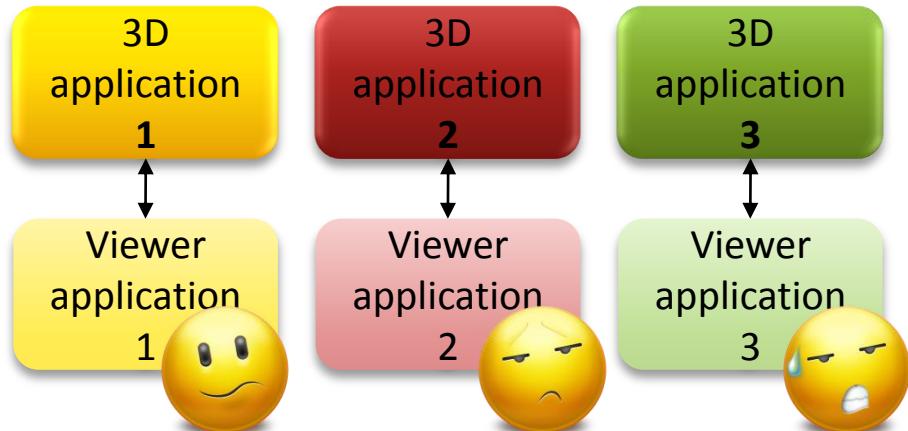


Issue of 3D format mixing

- Mix several 3D formats in a single 3D application, why ?

- many, many 3D formats
- reuse 3D models
- increase user access to 3D environments**

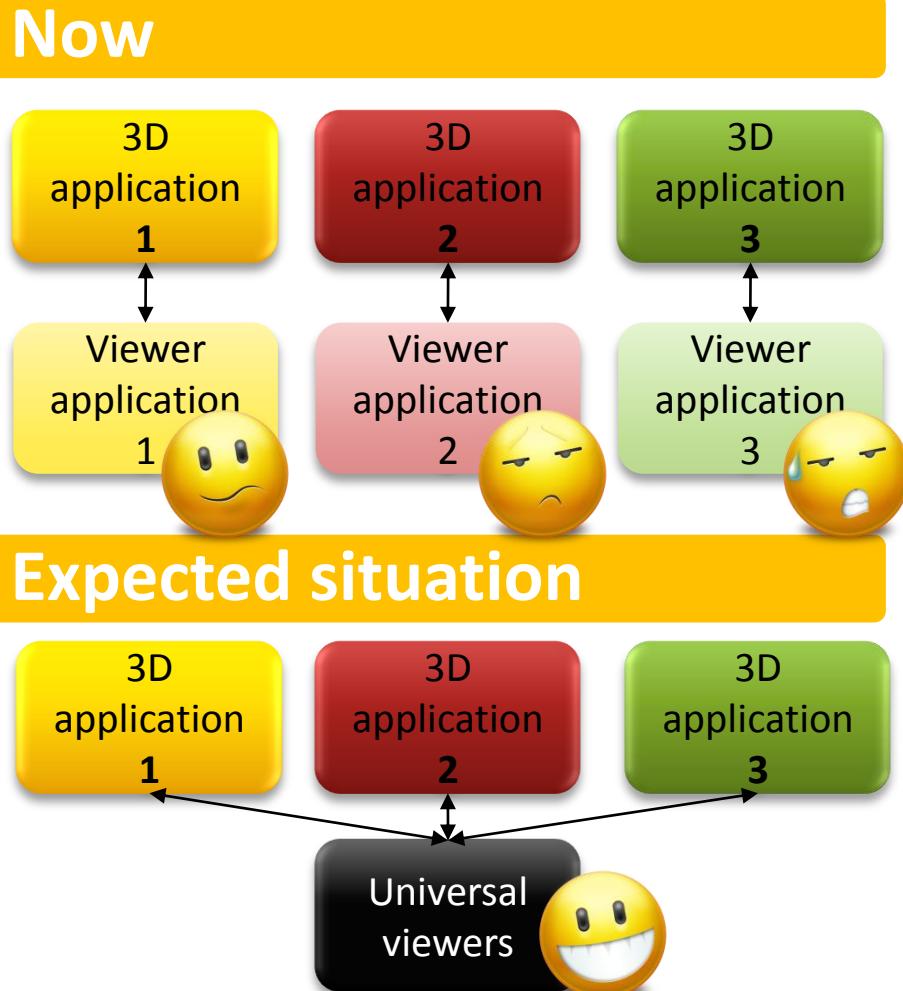
Now



Issue of 3D format mixing

- Mix several 3D formats in a single 3D application, why ?

- many, many 3D formats
- reuse 3D models
- increase user access to 3D environments**



Issue of 3D format mixing

- Mix several 3D formats in a single 3D application,
why ?

Now

3D
application
1

3D
application
2

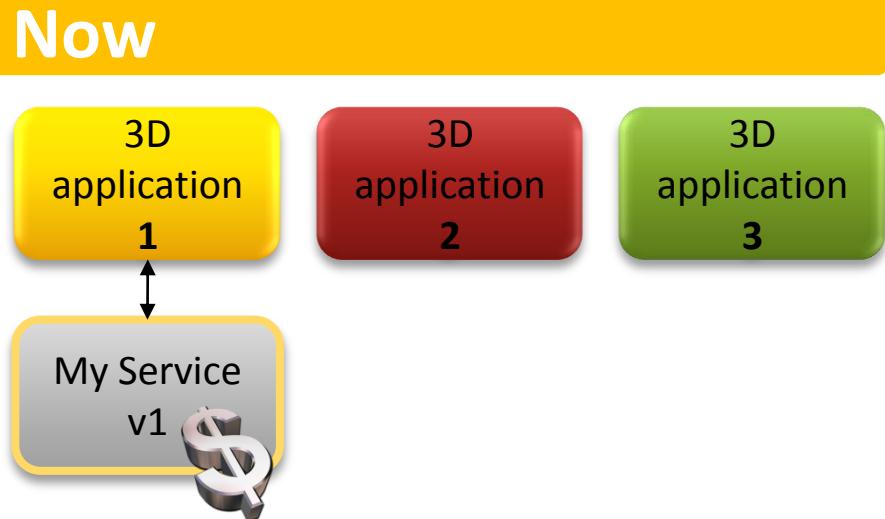
3D
application
3

- many, many 3D formats
- reuse 3D models
- increase user access to 3D environments
- increase industry investment in 3D environment**

Issue of 3D format mixing

- Mix several 3D formats in a single 3D application, why ?

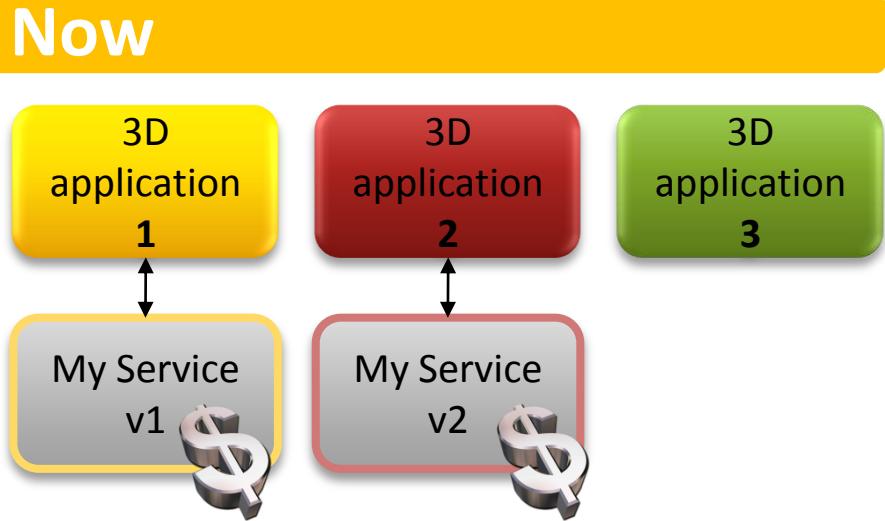
- many, many 3D formats
- reuse 3D models
- increase user access to 3D environments
- increase industry investment in 3D environment**



Issue of 3D format mixing

- Mix several 3D formats in a single 3D application, why ?

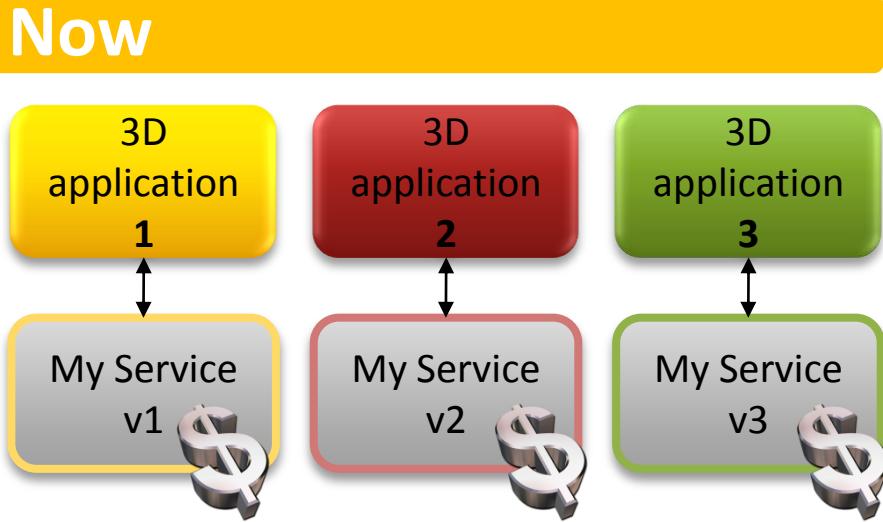
- many, many 3D formats
- reuse 3D models
- increase user access to 3D environments
- increase industry investment in 3D environment**



Issue of 3D format mixing

- Mix several 3D formats in a single 3D application, why ?

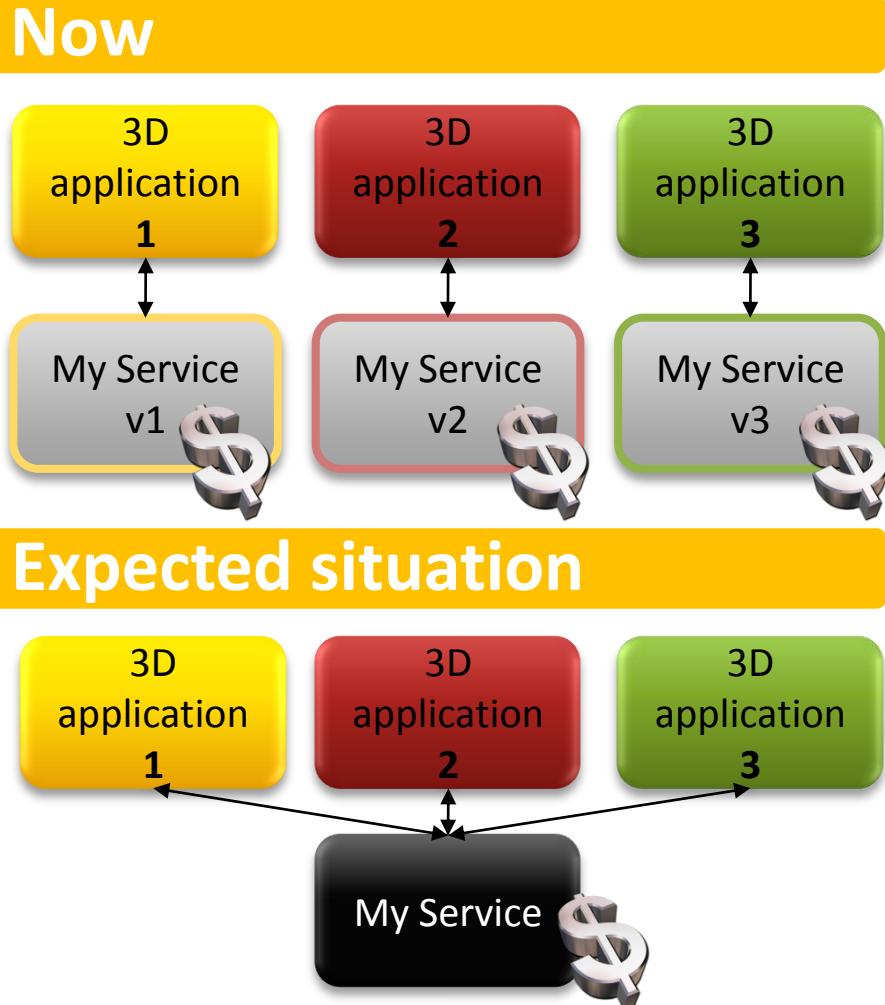
- many, many 3D formats
- reuse 3D models
- increase user access to 3D environments
- increase industry investment in 3D environment**



Issue of 3D format mixing

- Mix several 3D formats in a single 3D application, why ?

- many, many 3D formats
- reuse 3D models
- increase user access to 3D environments
- increase industry investment in 3D environment**



Issue of 3D format mixing

- Mix several 3D formats in a single 3D application, why ?

1. many, many 3D formats
2. reuse 3D models
3. increase user access to 3D environments
4. increase industry investment in 3D environment

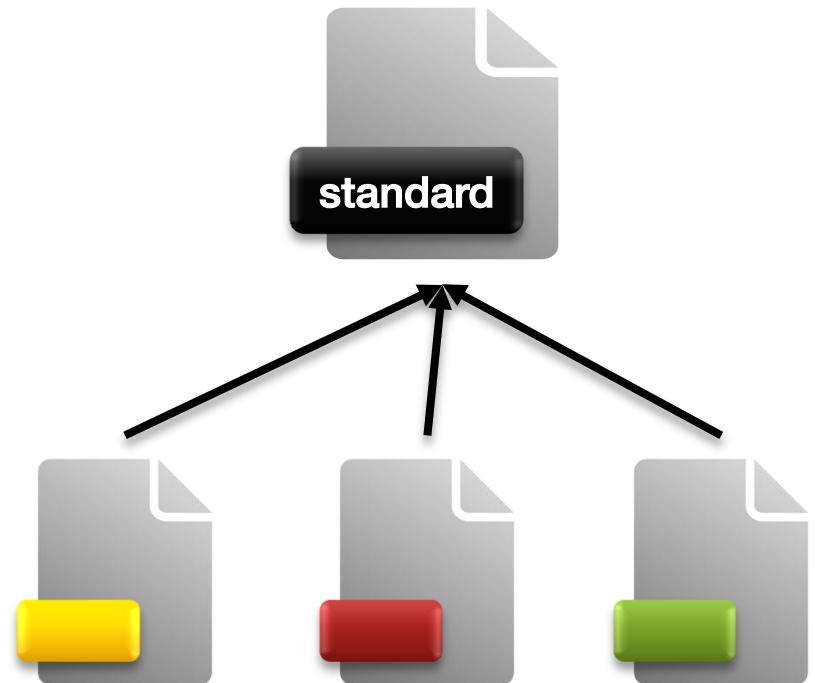
- Solution :
allowing interoperability between 3D formats and 3D application engines

Interoperability solutions

- 3 types of solution as proposed in [Haslhofer and Klas 2010] :

1. model agreement

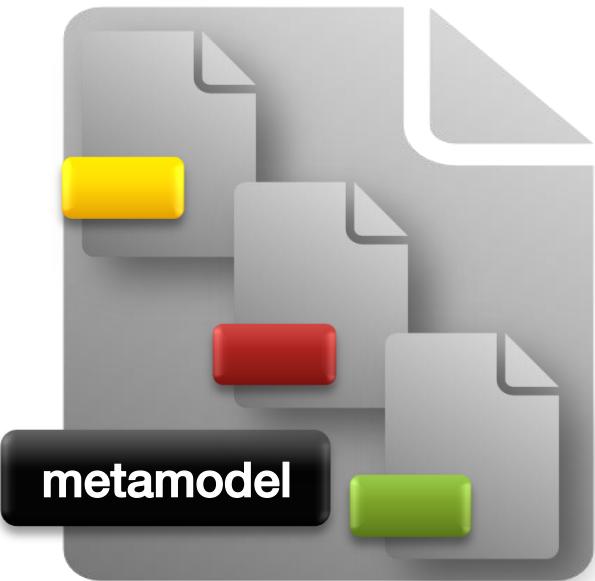
- WebGL [Khronos Group 2011]
- X3DOM [Berh et al. 2010]
- Collada
- MPEG4
- OGC
- ...



Interoperability solutions

- 3 types of solution as proposed in [Haslhofer and Klas 2010] :

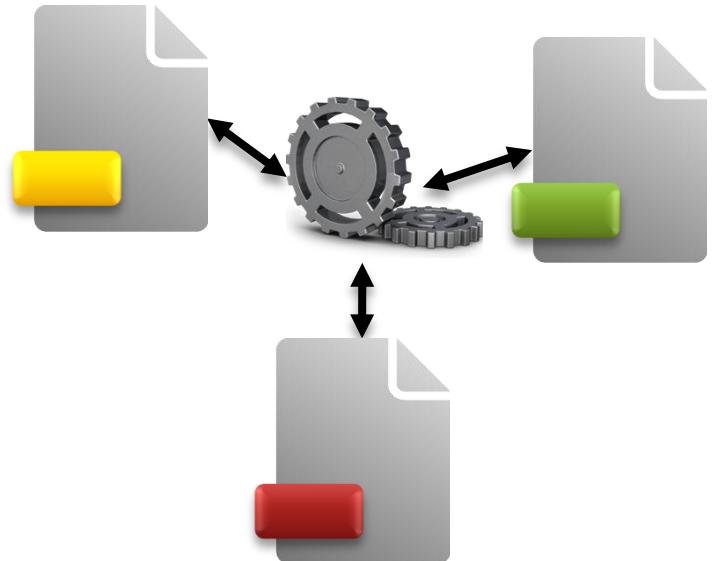
1. model agreement
2. **metamodel agreement**
 - no existing solution



Interoperability solutions

- 3 types of solution as proposed in [Haslhofer and Klas 2010] :

1. model agreement
2. metamodel agreement
3. **model reconciliation**
 - MPEG-V Part 2 [Gelissen 2008]
 - vWRAP [Bell et al. 2010]
 - PLUG [Hu and Jiang 2008]

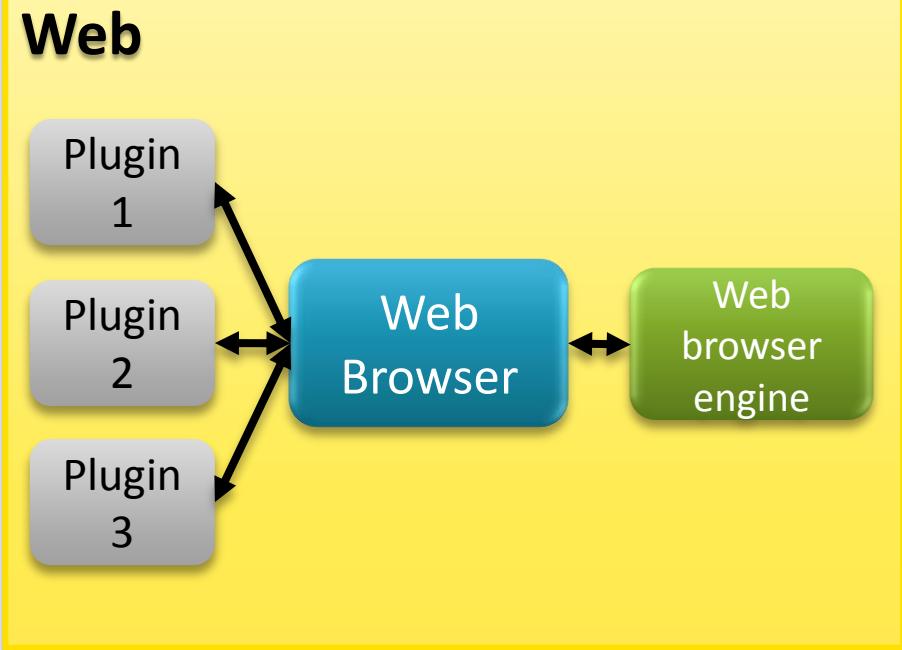


Interoperability solutions summary

- None of these solutions allows **reusing** directly existing 3D contents
- Most of them require **transcoding**
- All of them imply format **functionality loss**

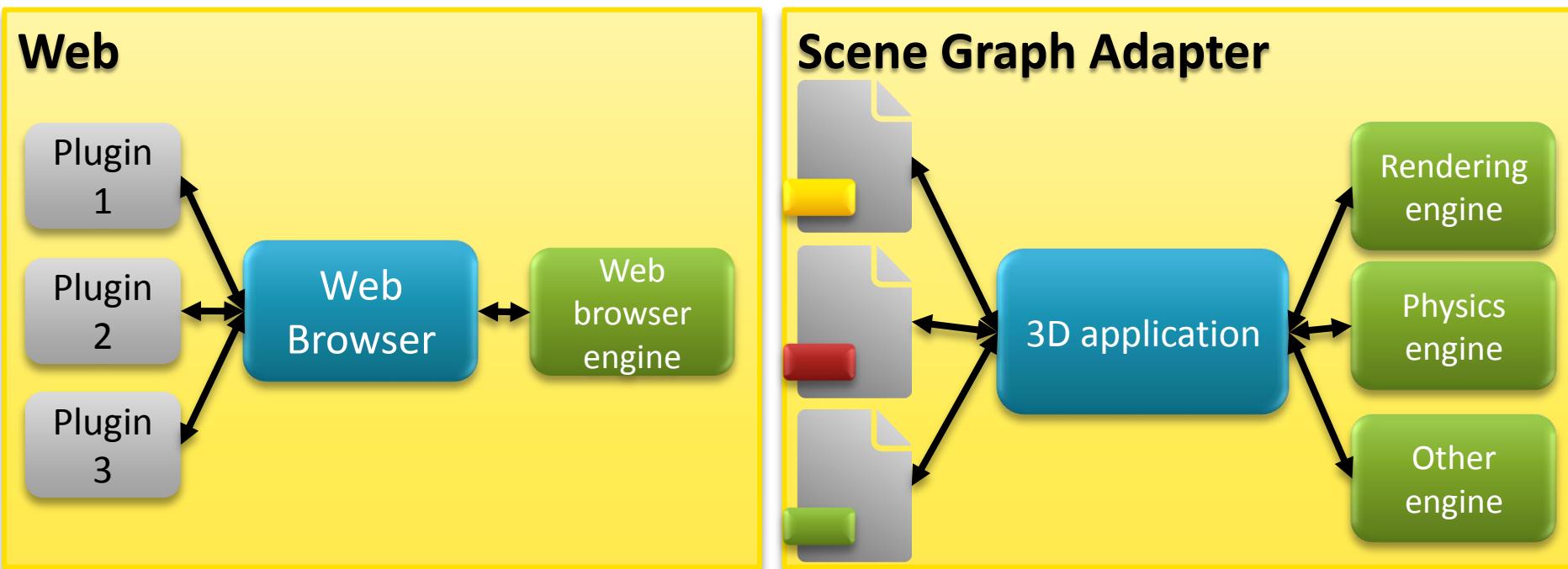
Scene Graph Adapter - Overview

- A model reconciliation **interoperability** solution
- Compliant with most **3D formats**
- Compliant with most **rendering components**



Scene Graph Adapter - Overview

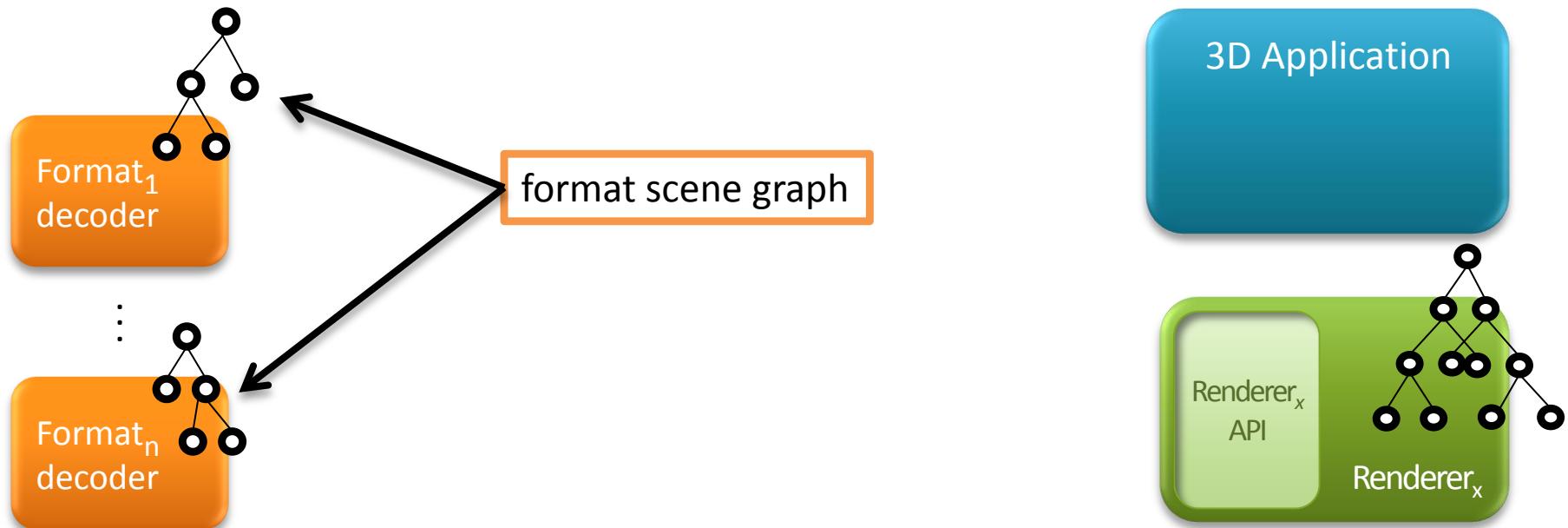
- A model reconciliation **interoperability** solution
- Compliant with most **3D formats**
- Compliant with most **3D application engine**



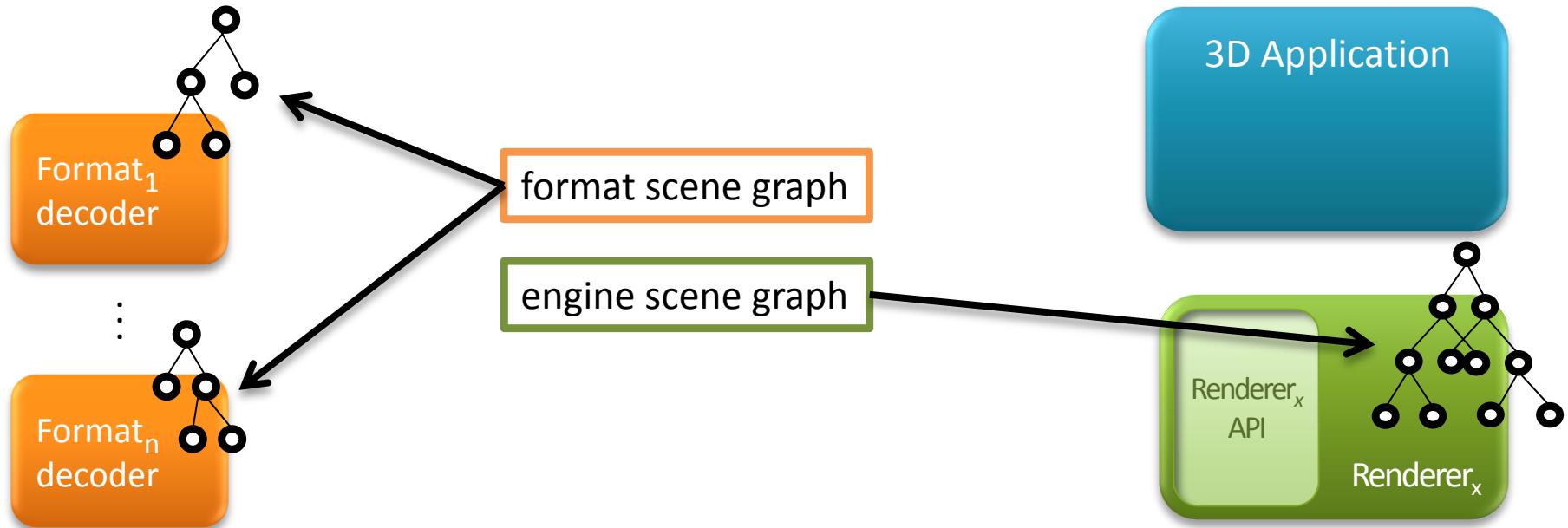
Scene Graph Adapter - Architecture



Scene Graph Adapter - Architecture

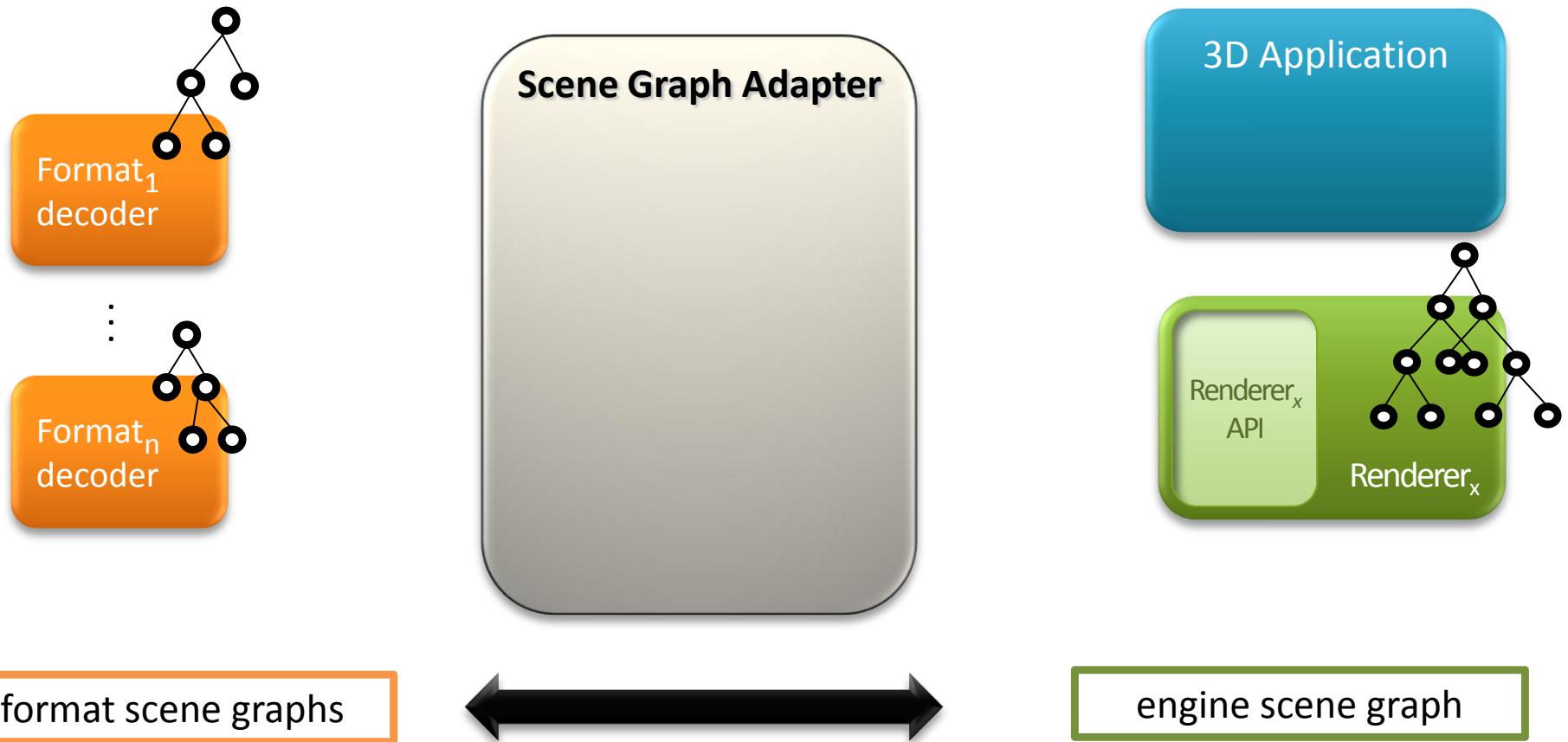


Scene Graph Adapter - Architecture

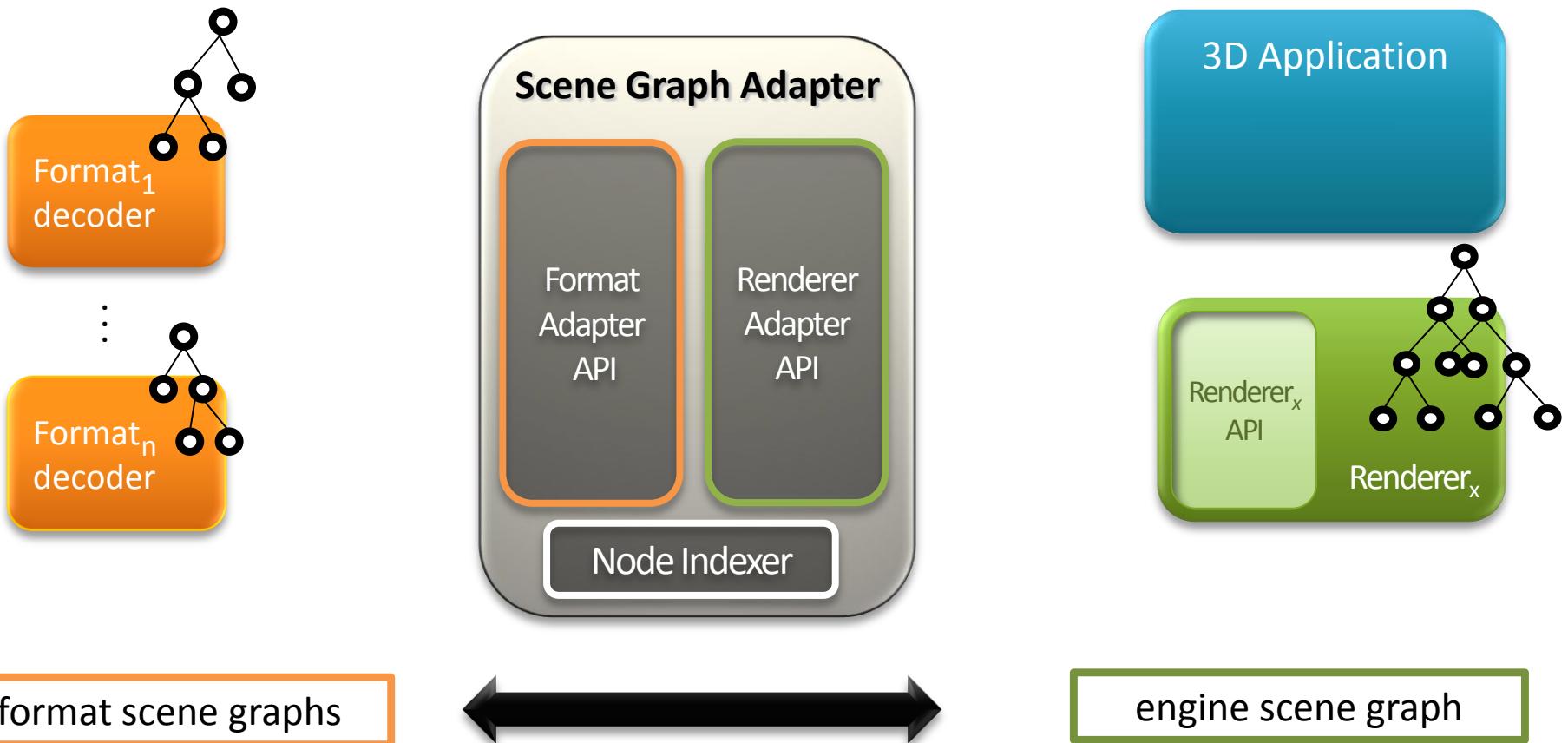


- Similarities between format decoder and renderer :
 - scene-graph-based structure
 - same model and data organization in scene graph

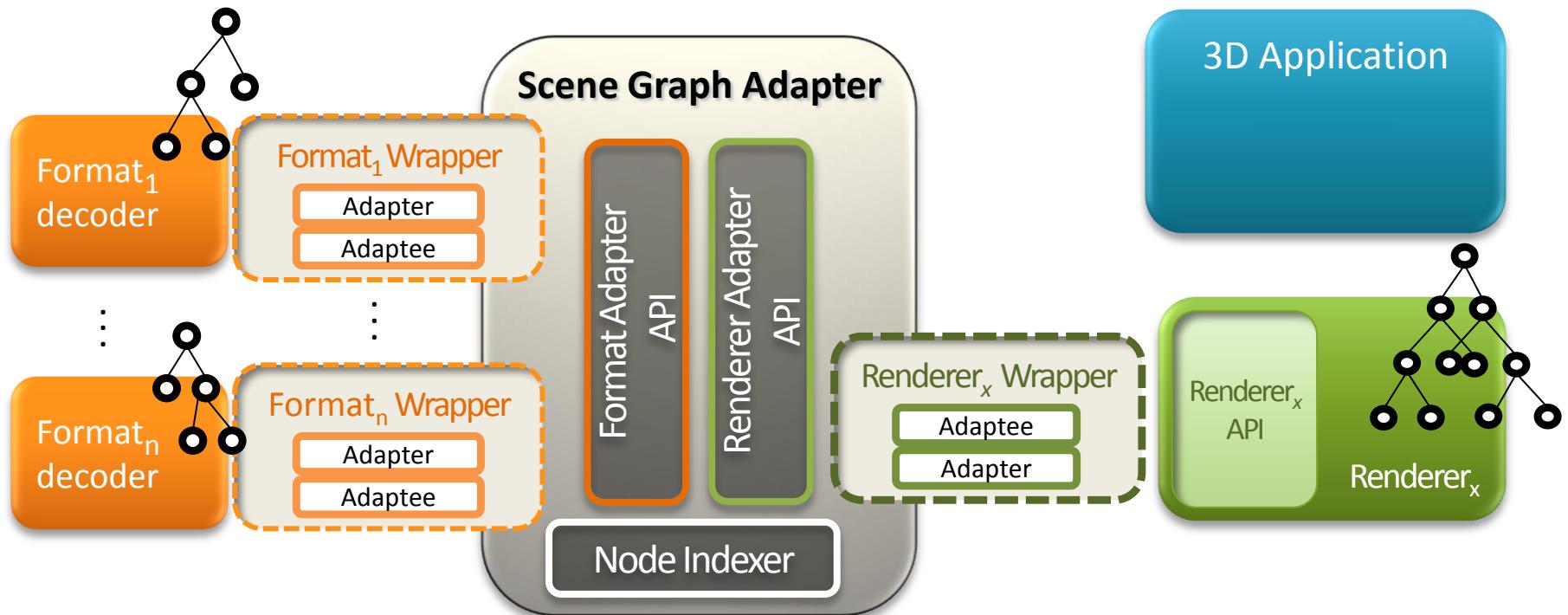
Scene Graph Adapter - Architecture



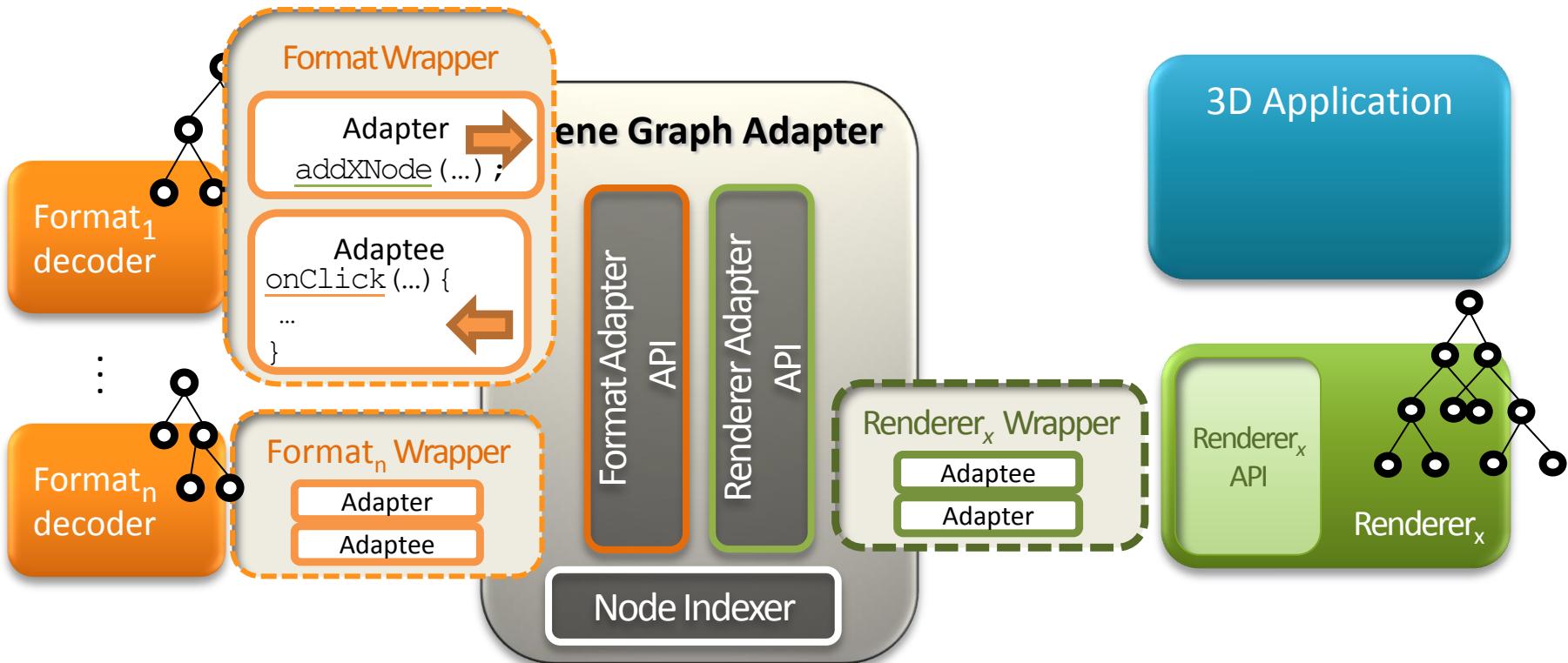
Scene Graph Adapter - Architecture



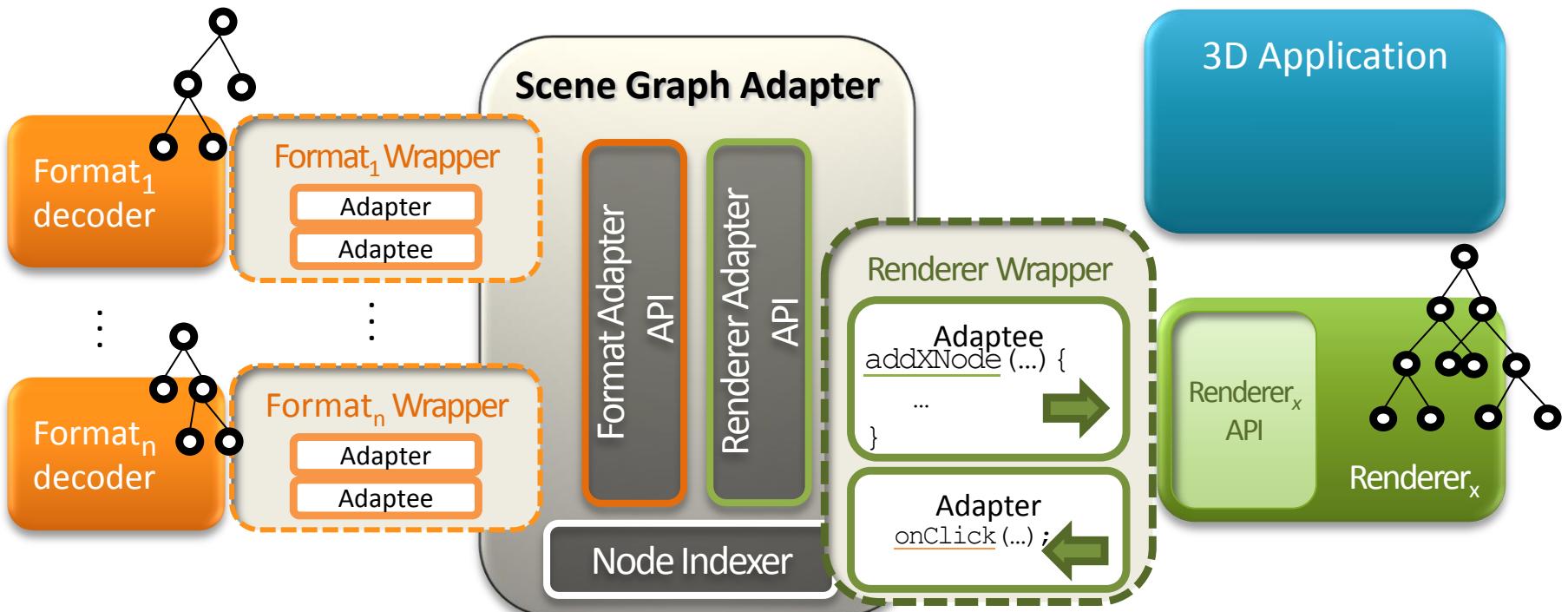
Scene Graph Adapter - Architecture



Scene Graph Adapter - Architecture

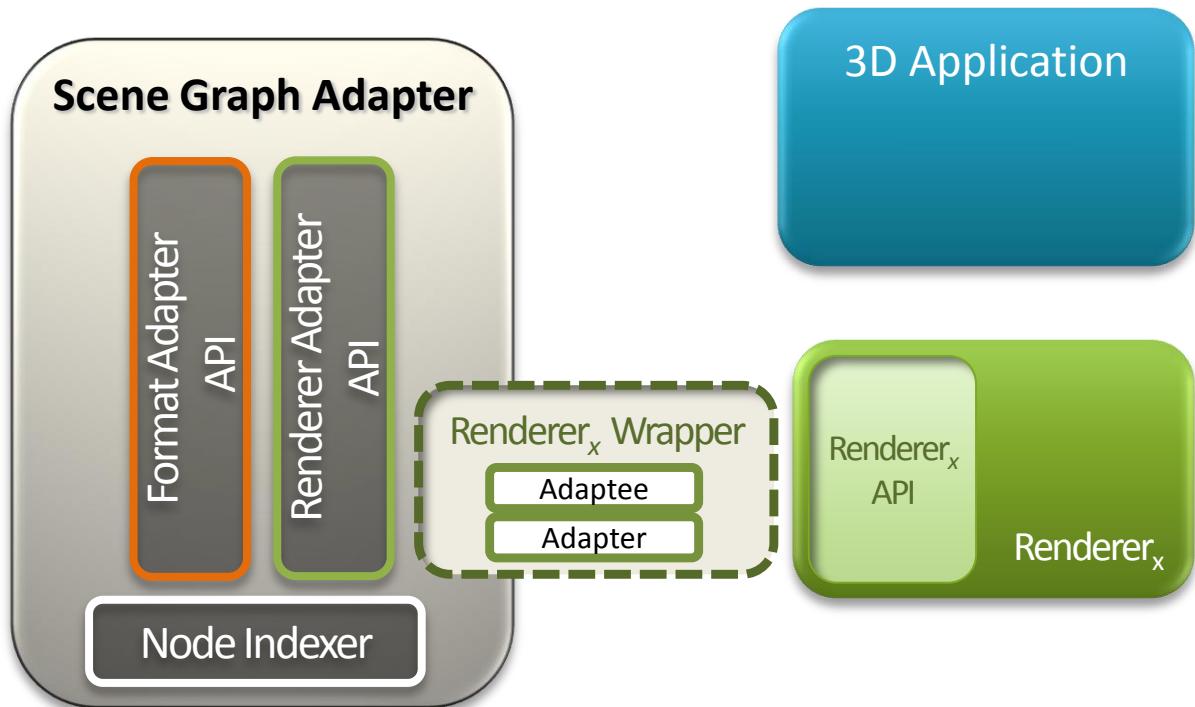


Scene Graph Adapter - Architecture



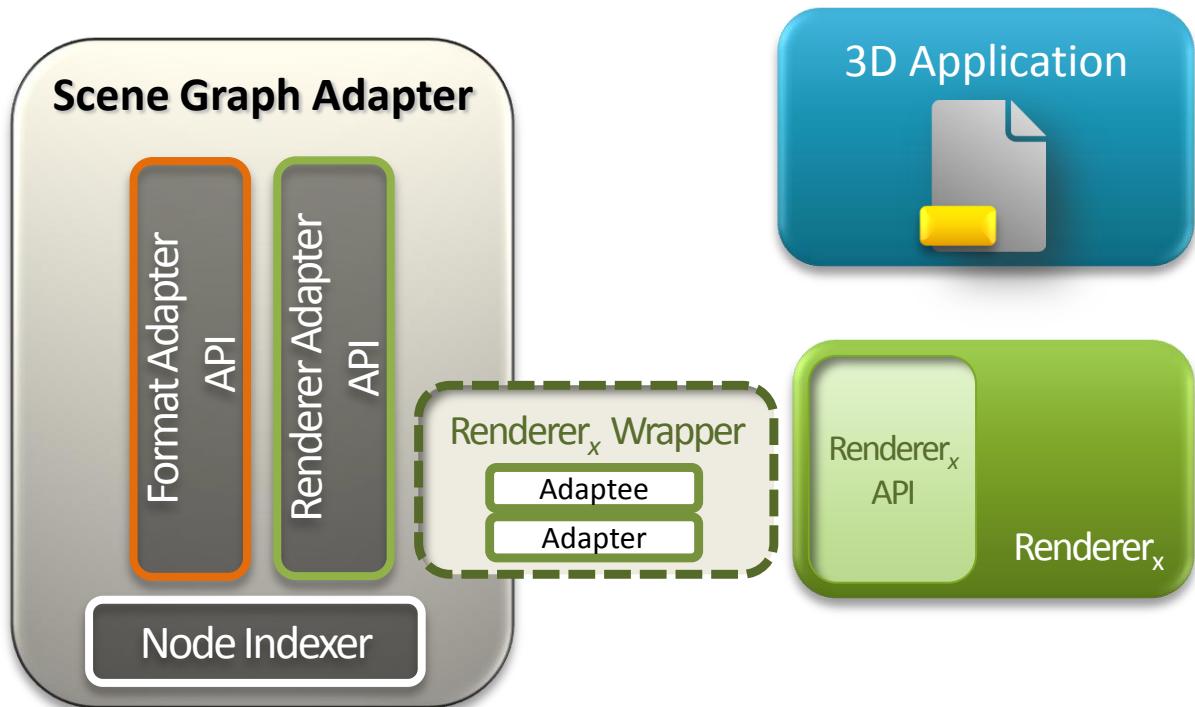
Functioning 1

- Loading a file :



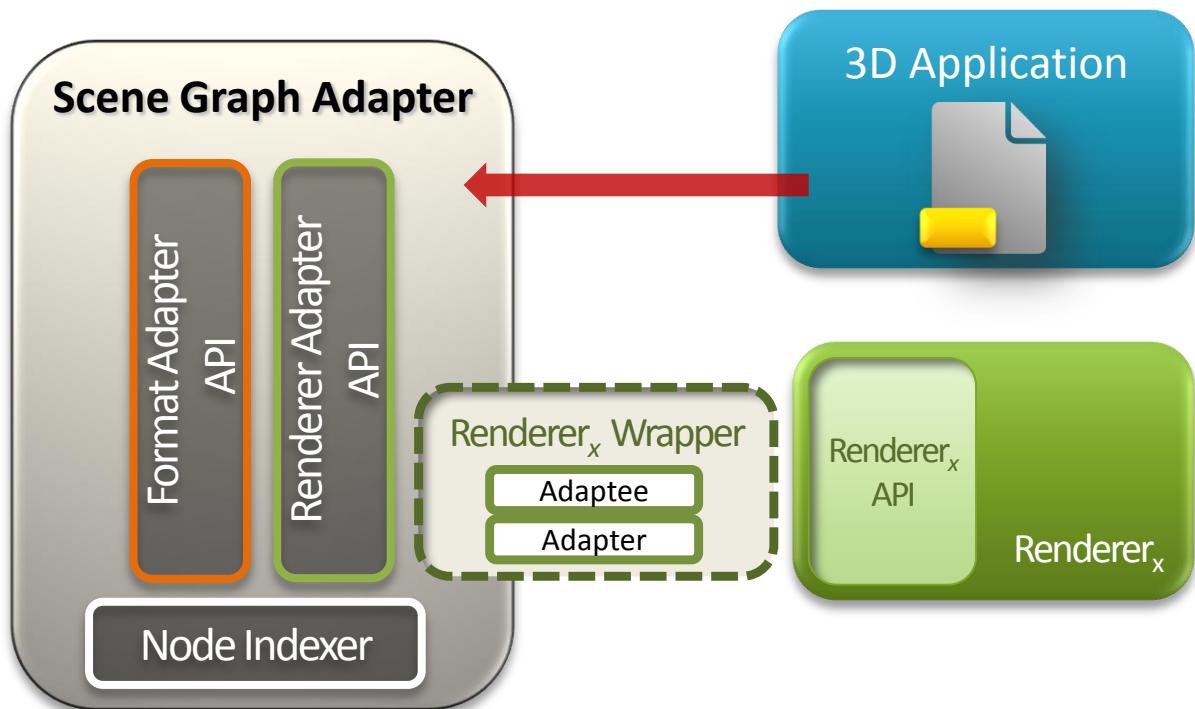
Functioning 1

- Loading a file :



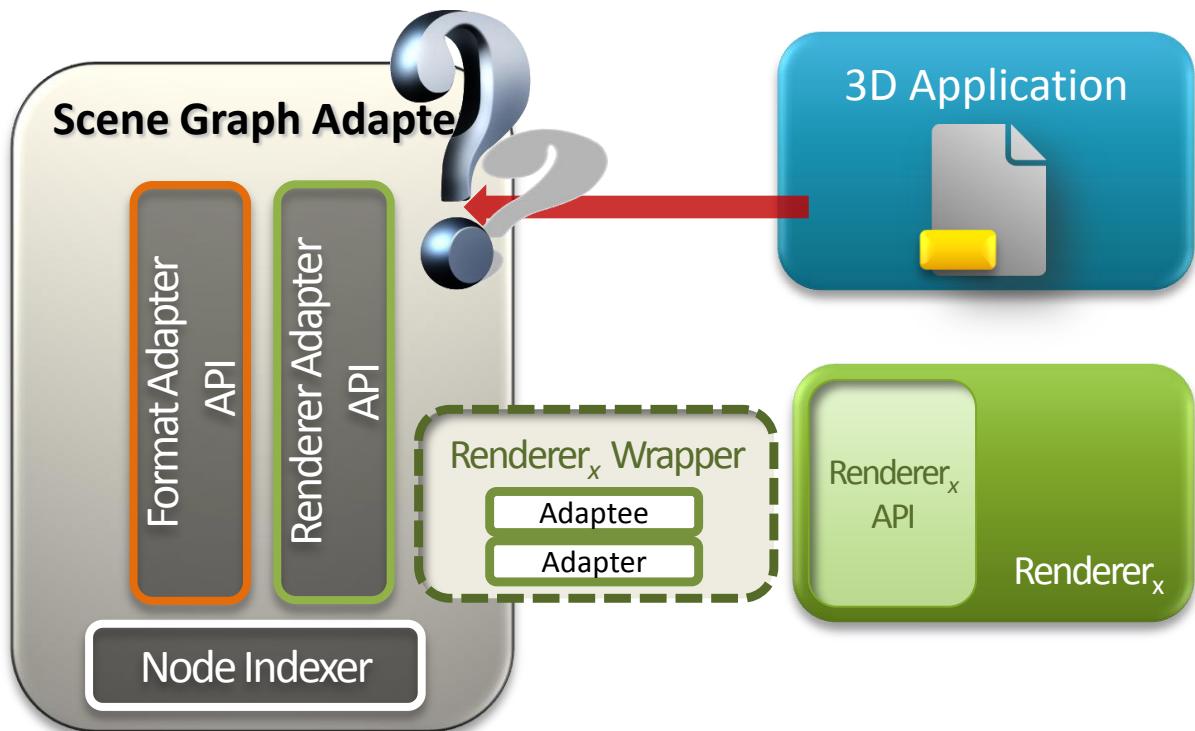
Functioning 1

- Loading a file :



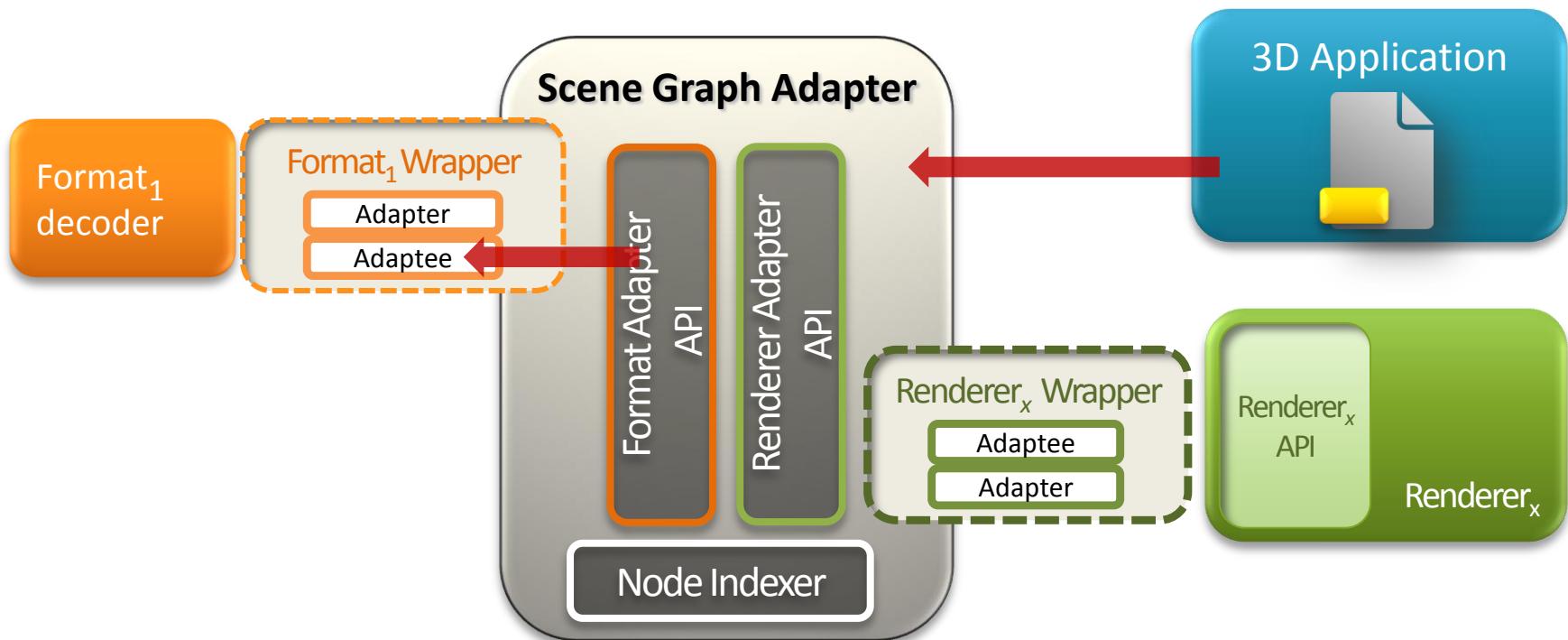
Functioning 1

- Loading a file :



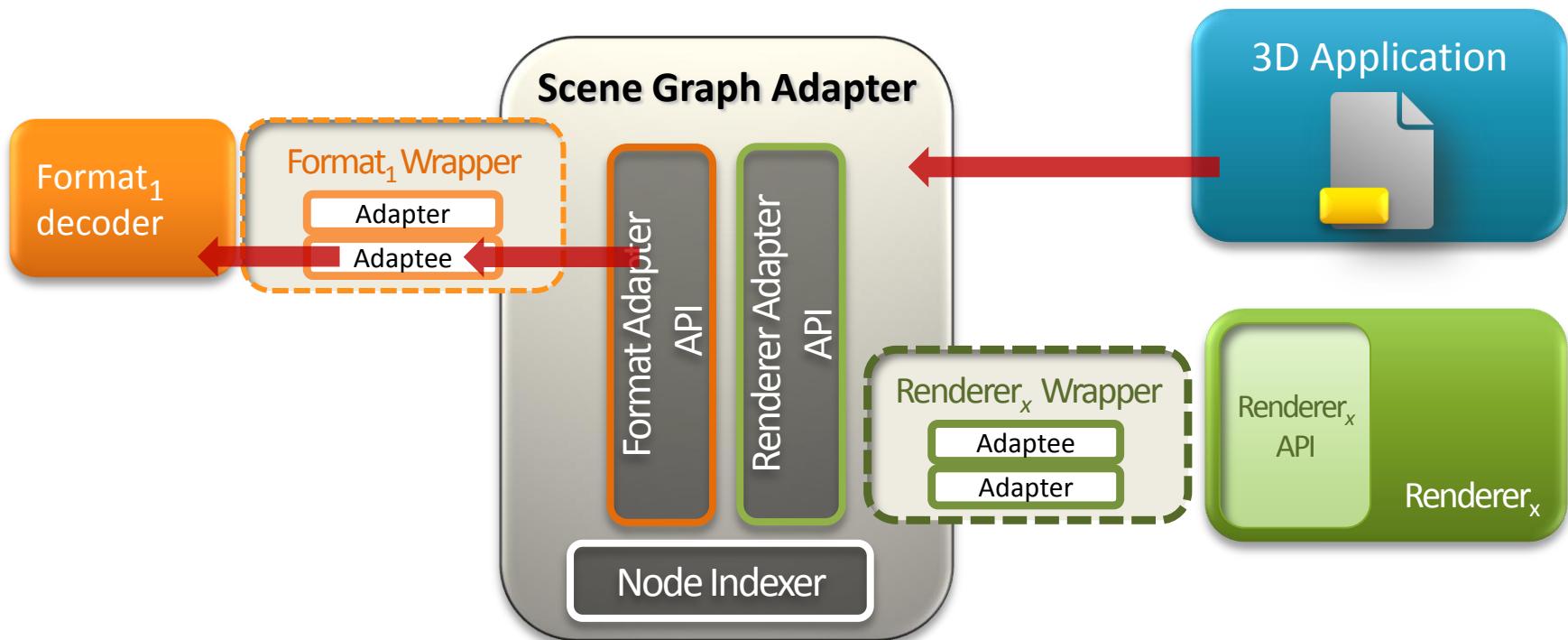
Functioning 1

■ Loading a file :



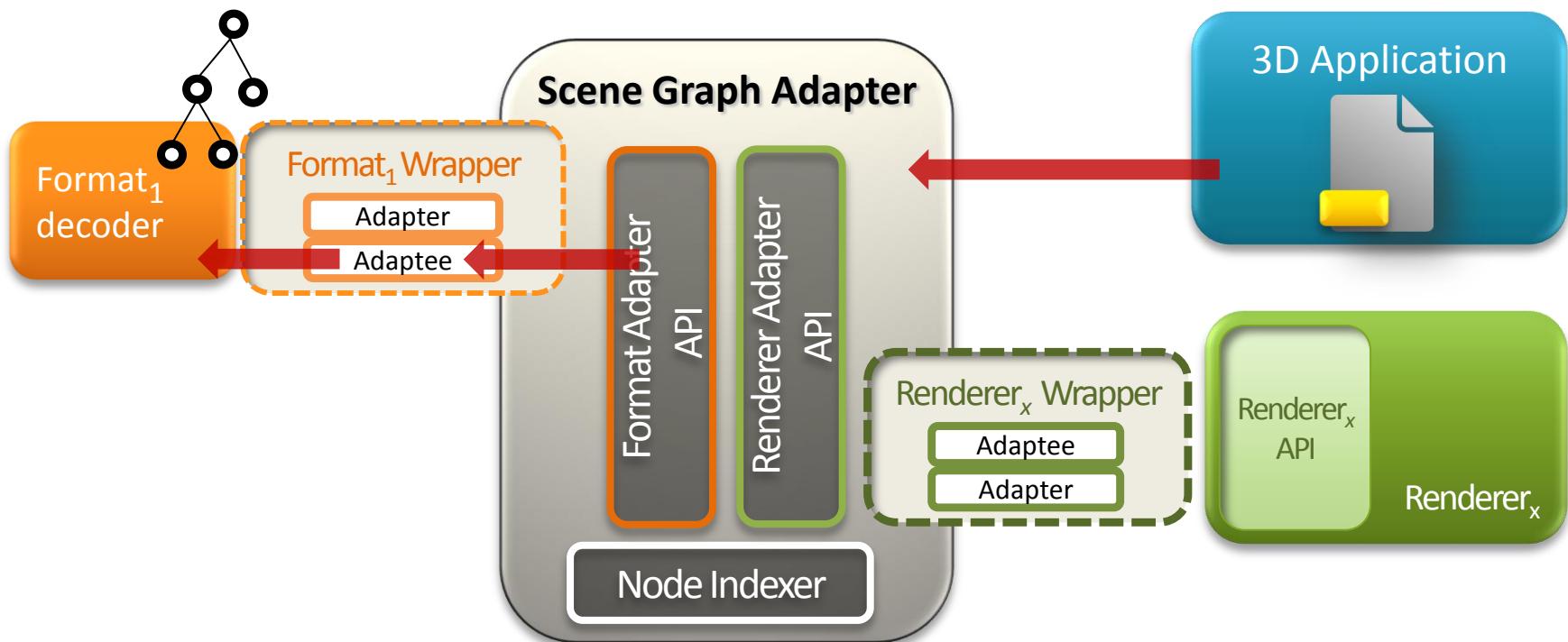
Functioning 1

■ Loading a file :



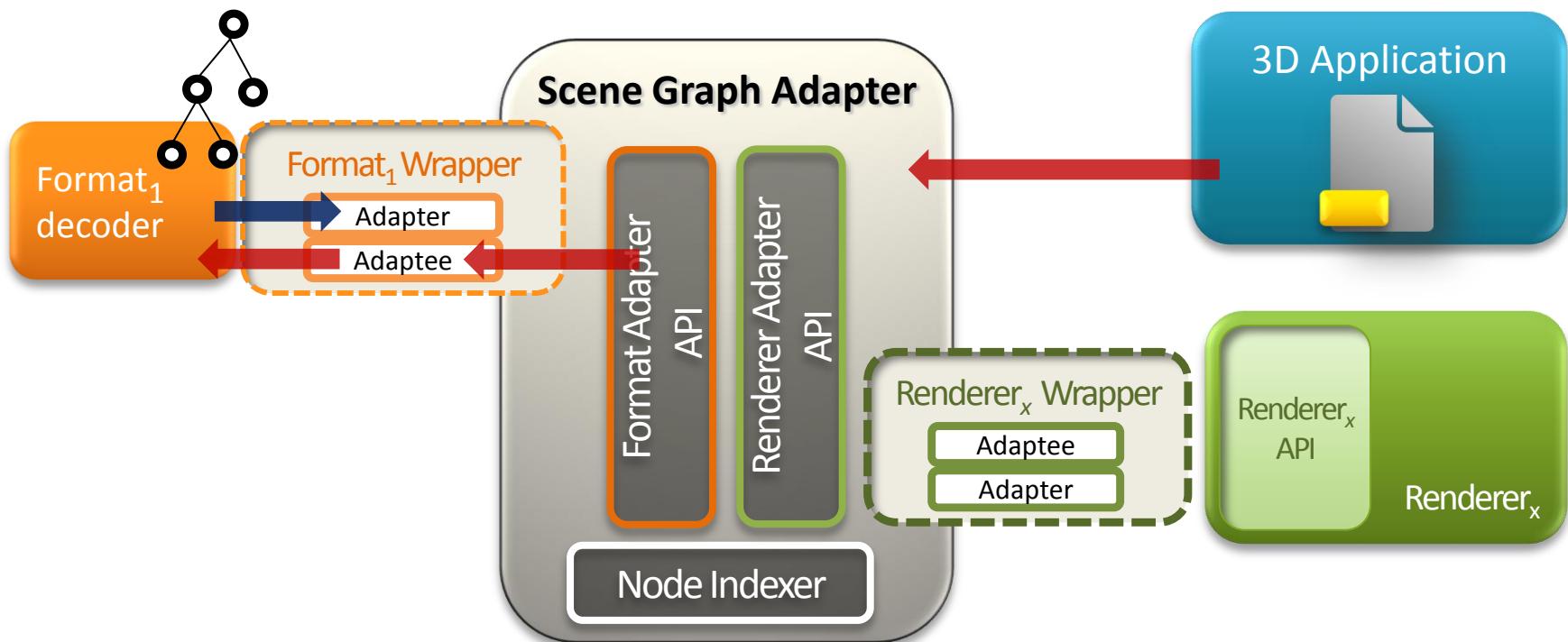
Functioning 1

■ Loading a file :



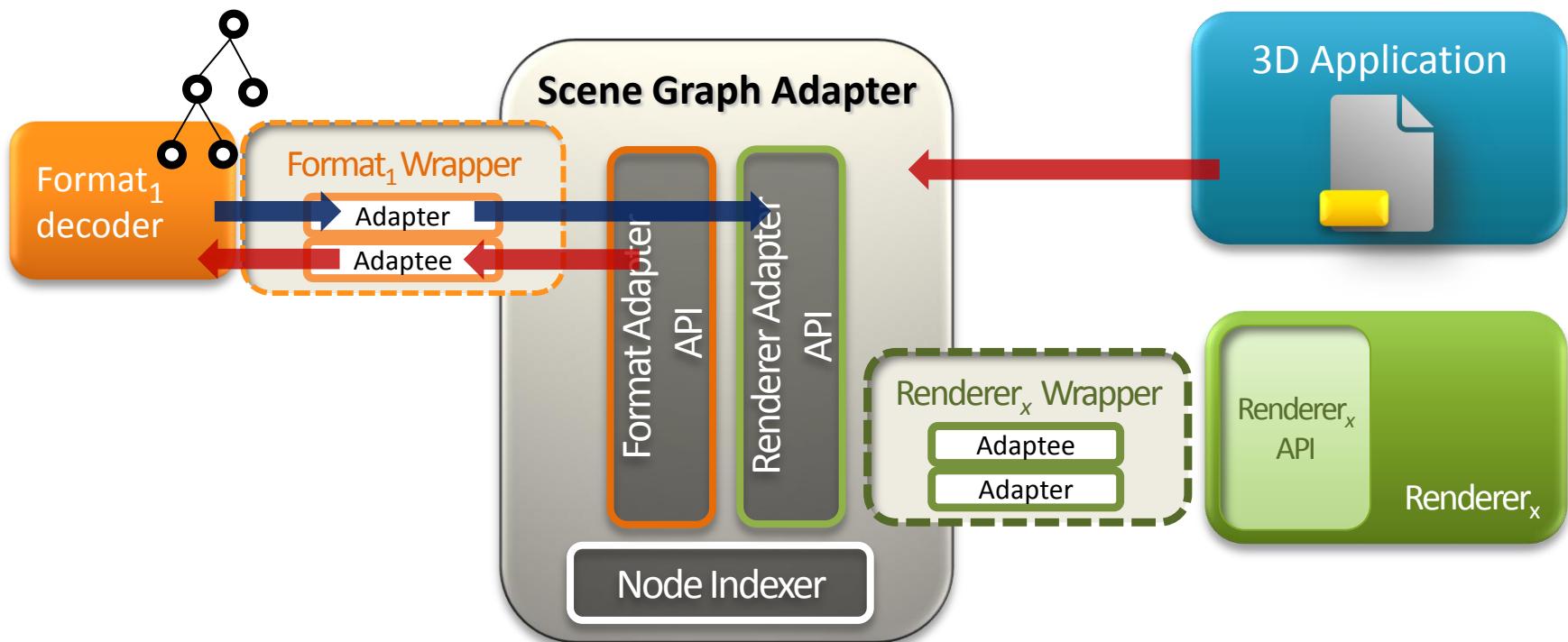
Functioning 1

■ Loading a file :



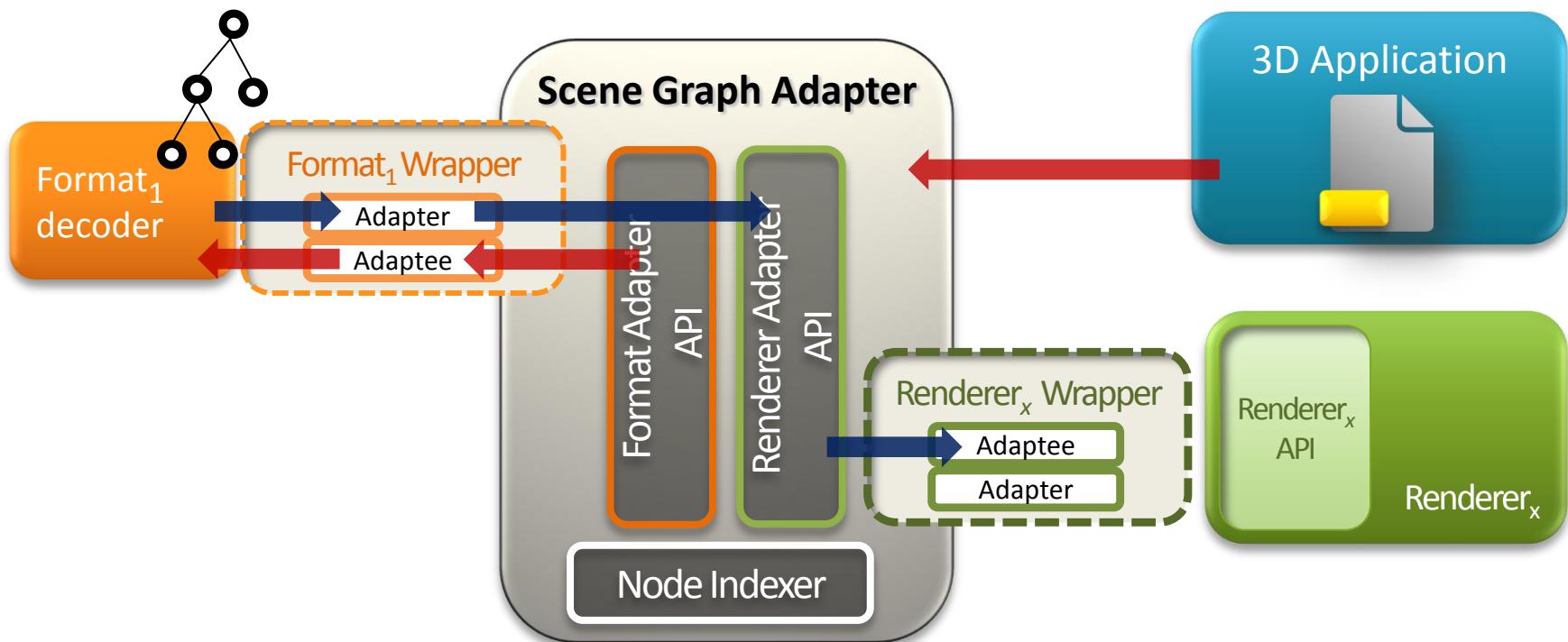
Functioning 1

■ Loading a file :



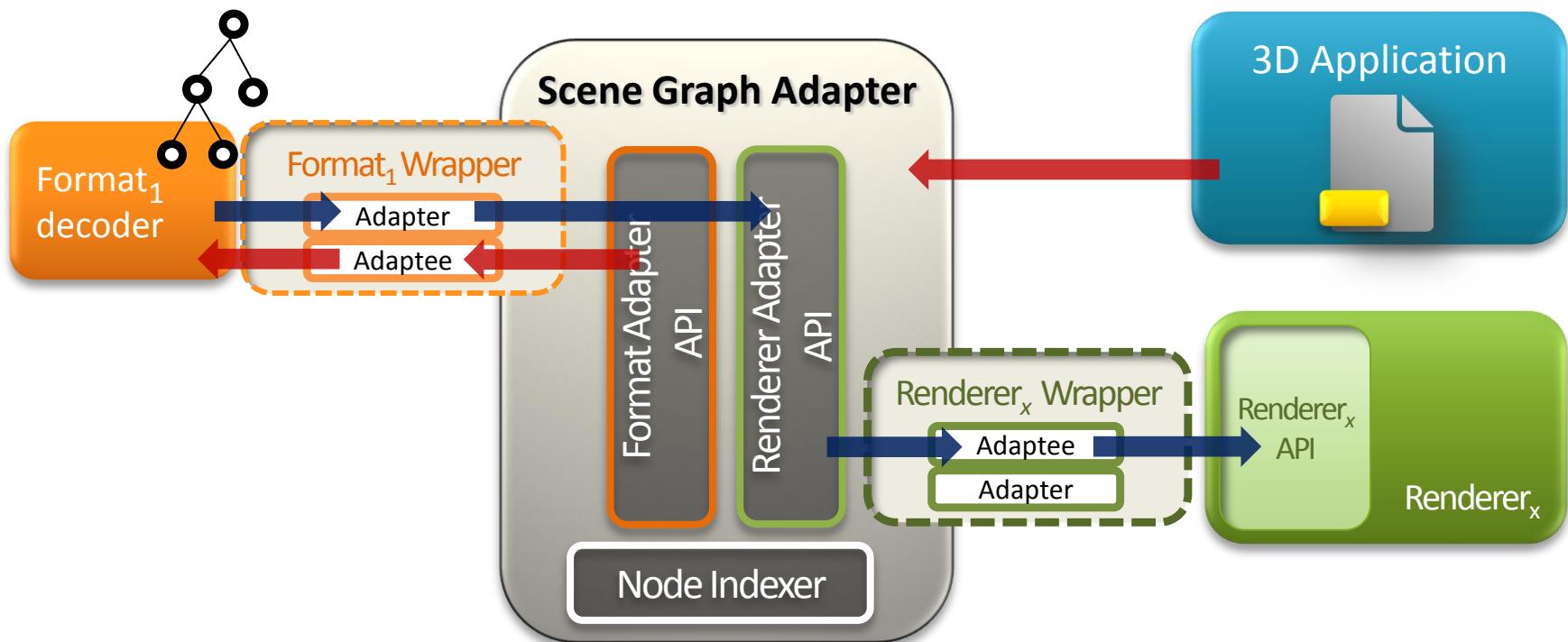
Functioning 1

■ Loading a file :



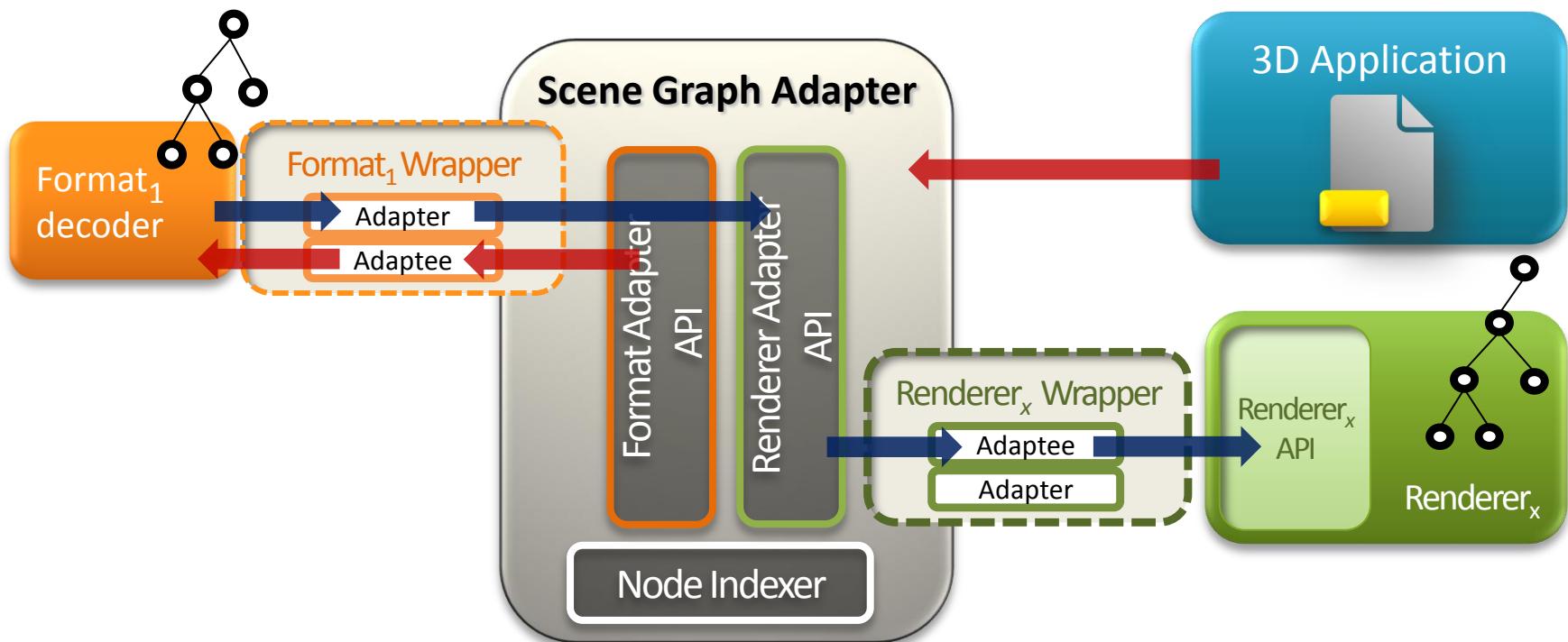
Functioning 1

■ Loading a file :



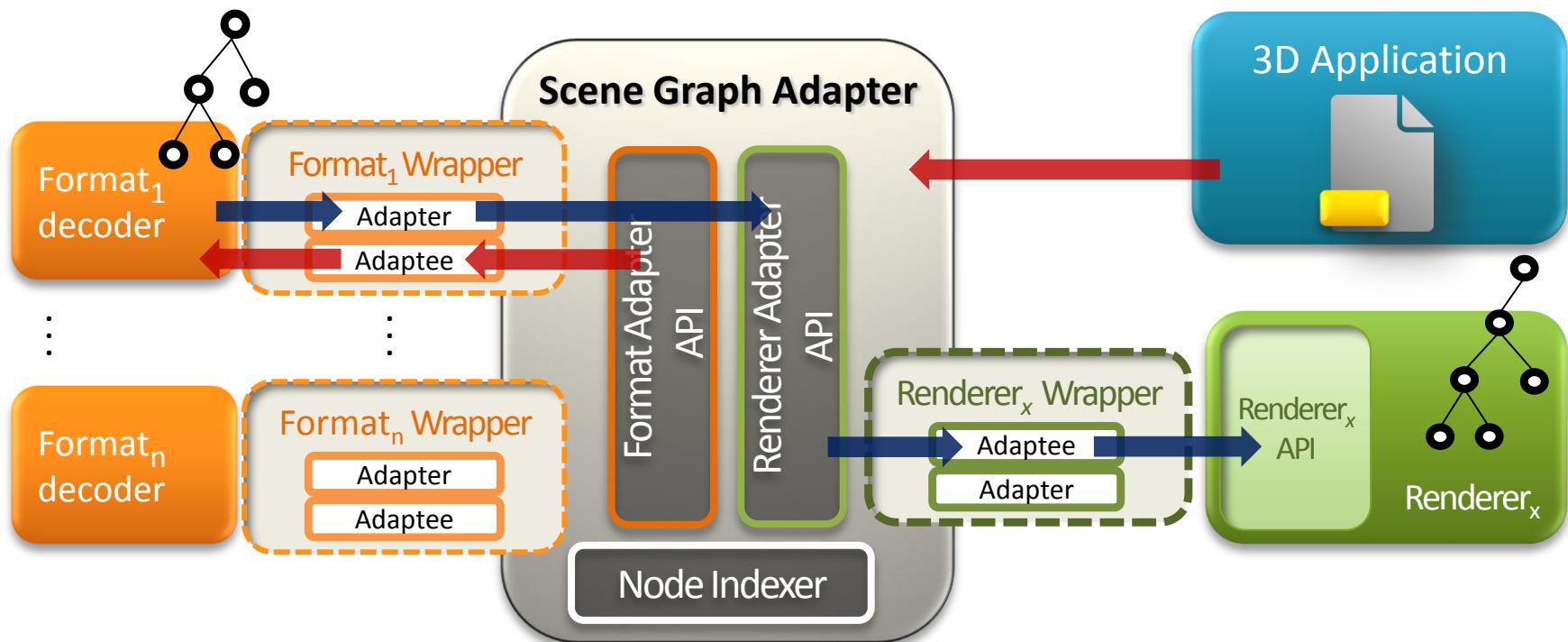
Functioning 1

■ Loading a file :



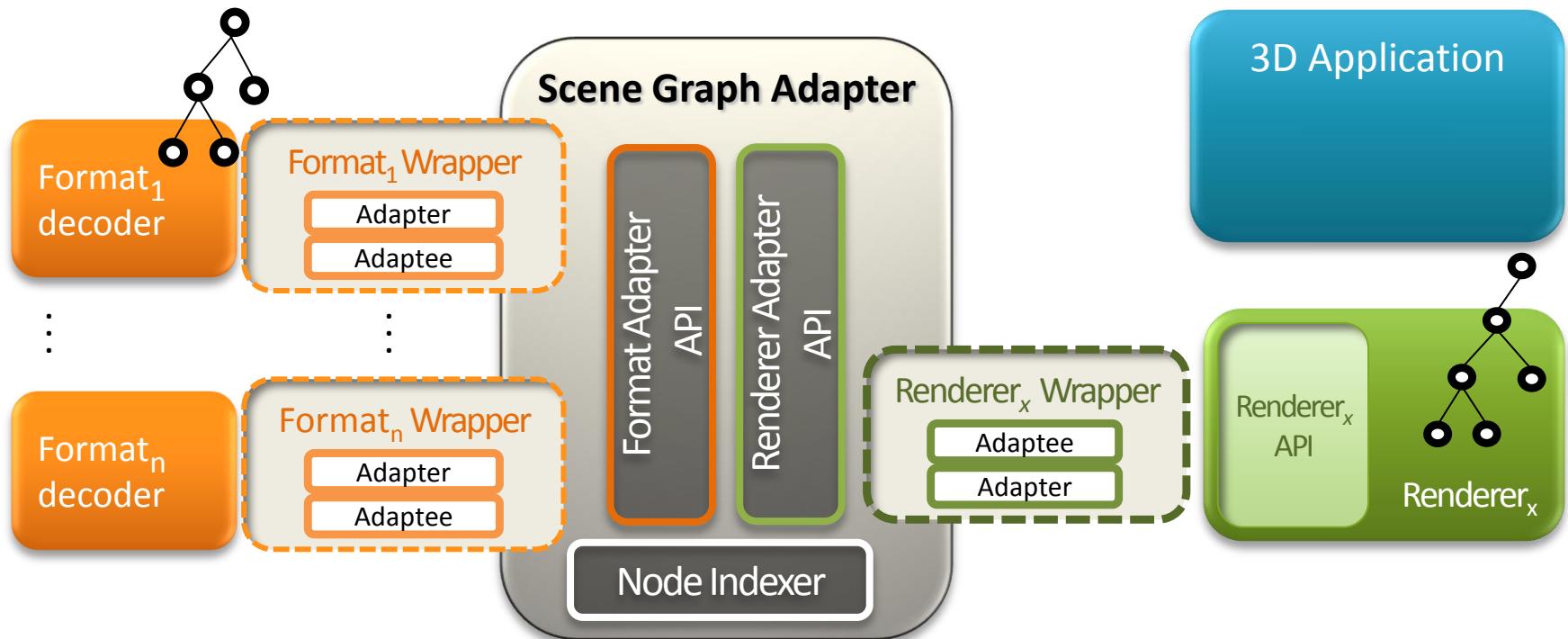
Functioning 1

■ Loading a file :



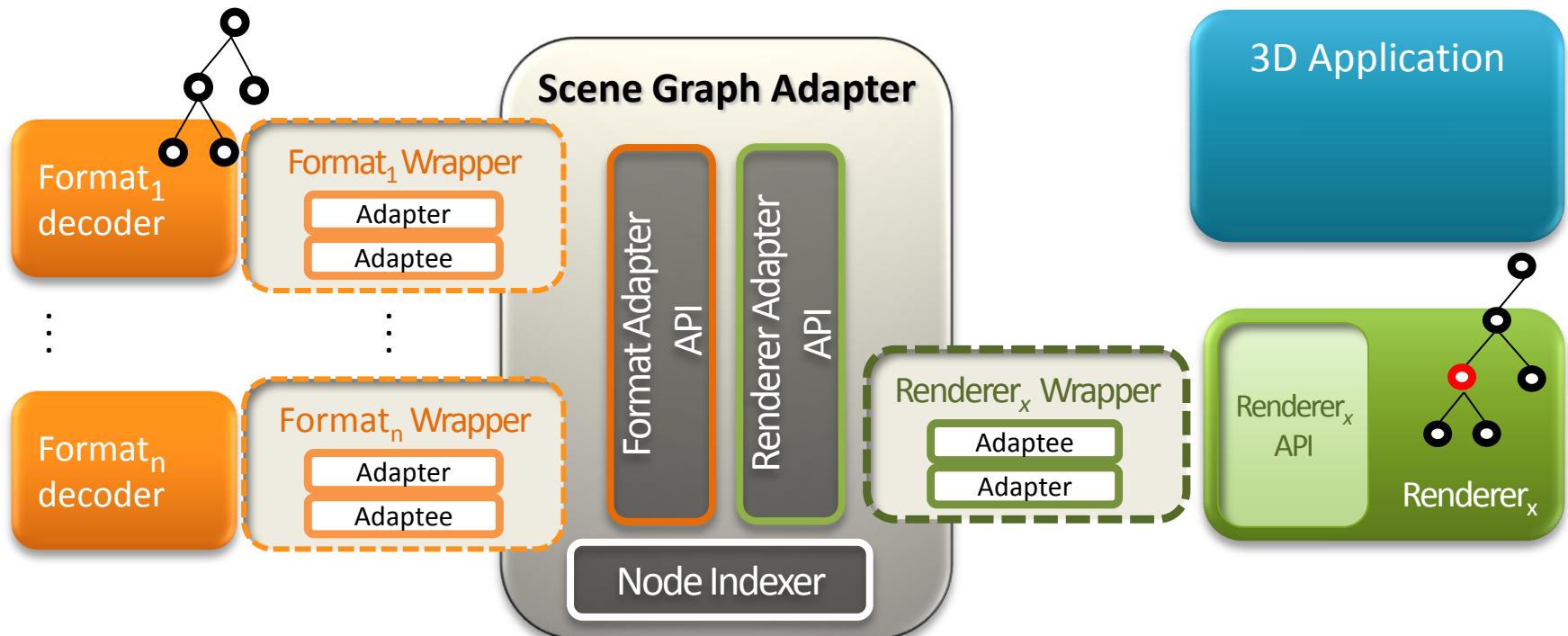
Functioning 2

Object picking :



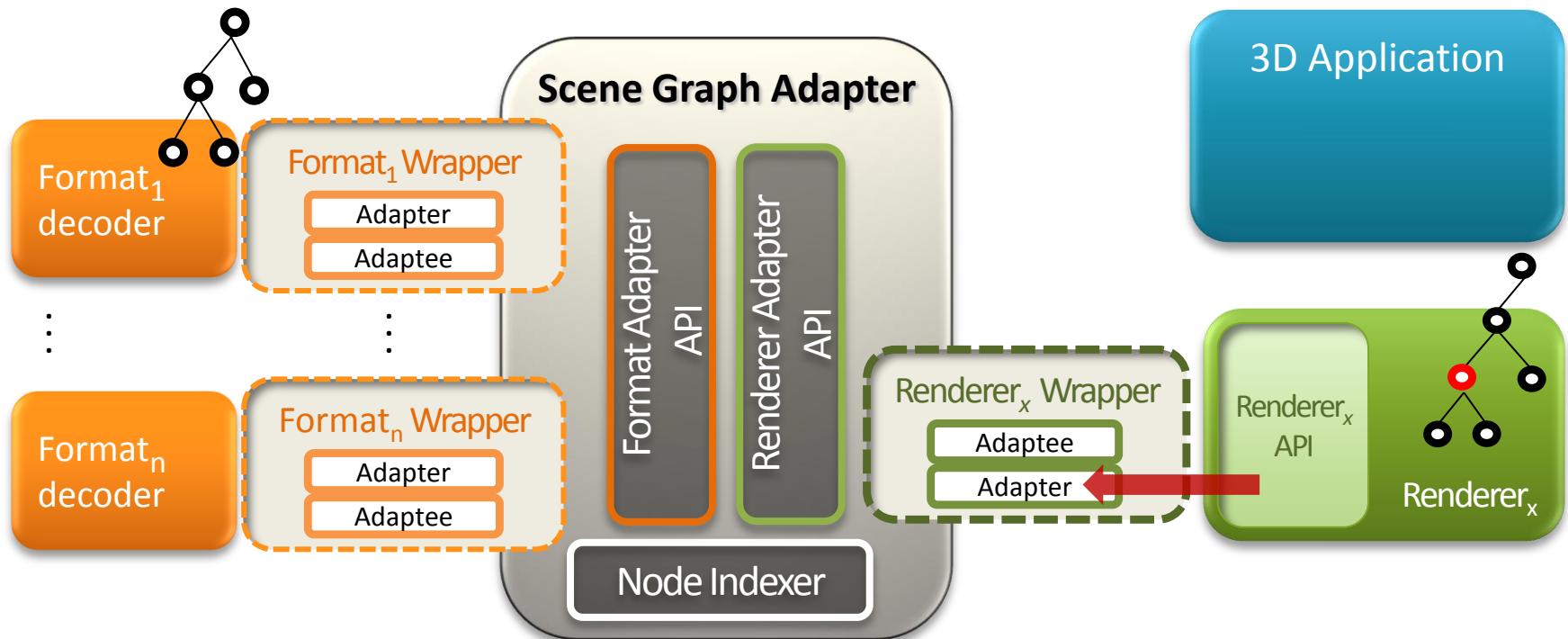
Functioning 2

Object picking :



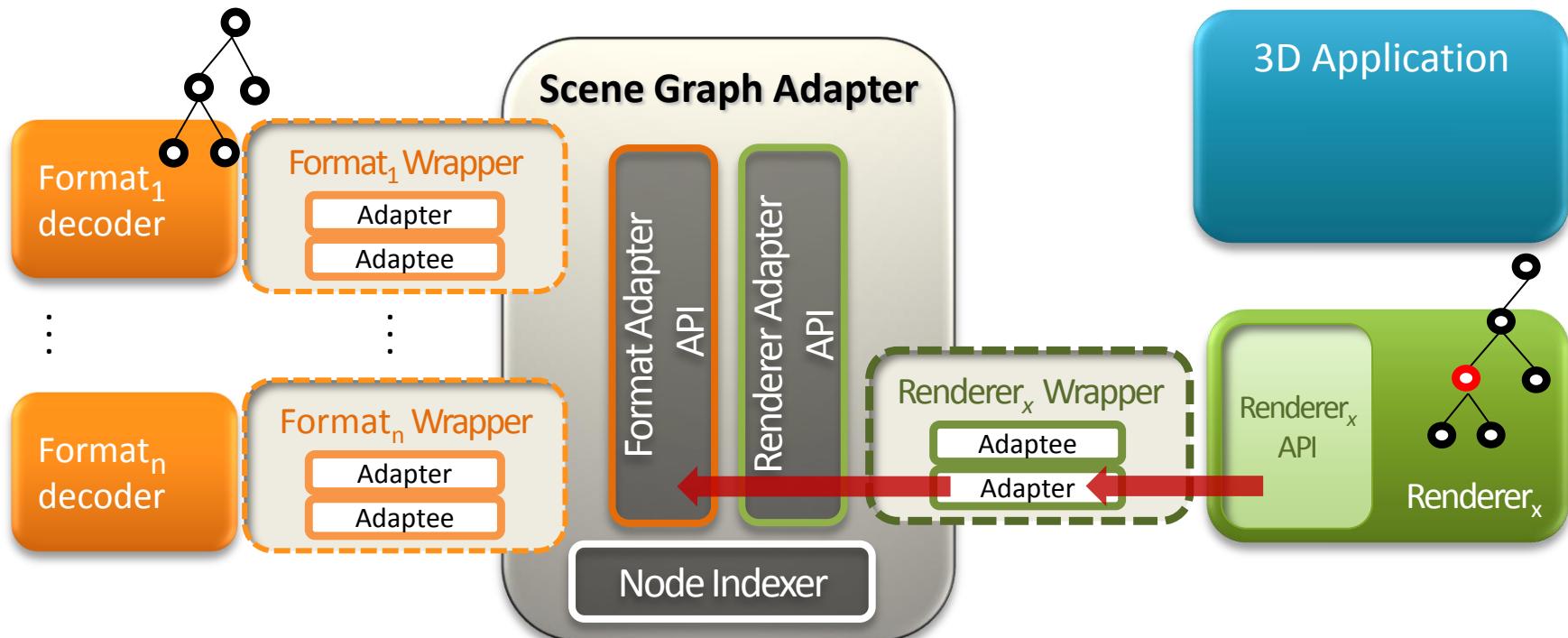
Functioning 2

Object picking :



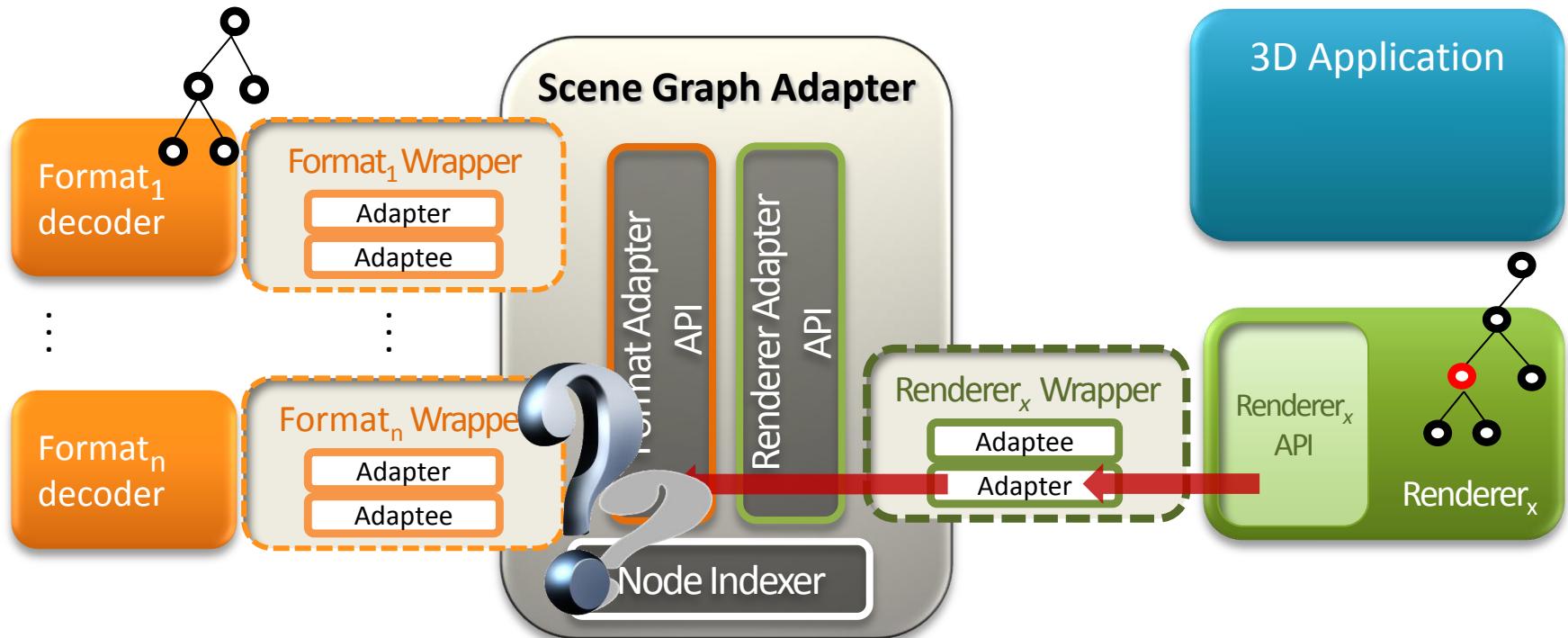
Functioning 2

Object picking :



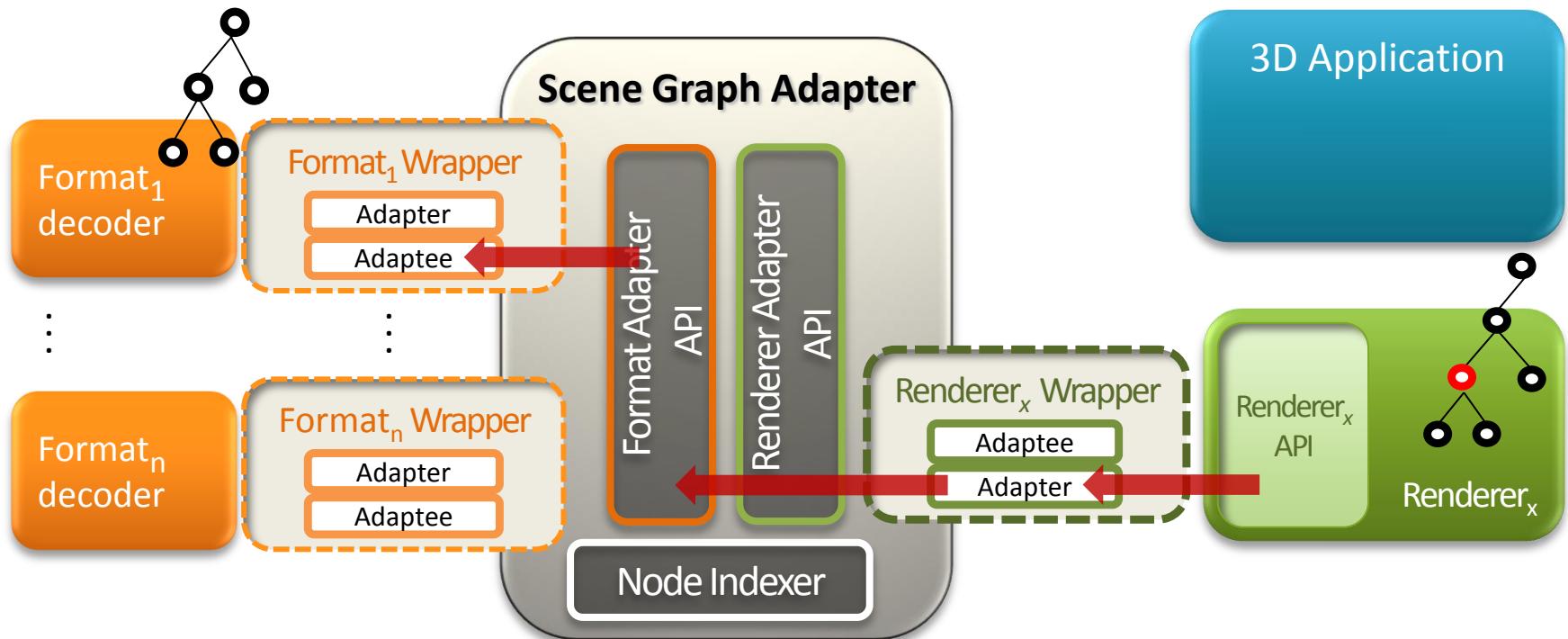
Functioning 2

Object picking :



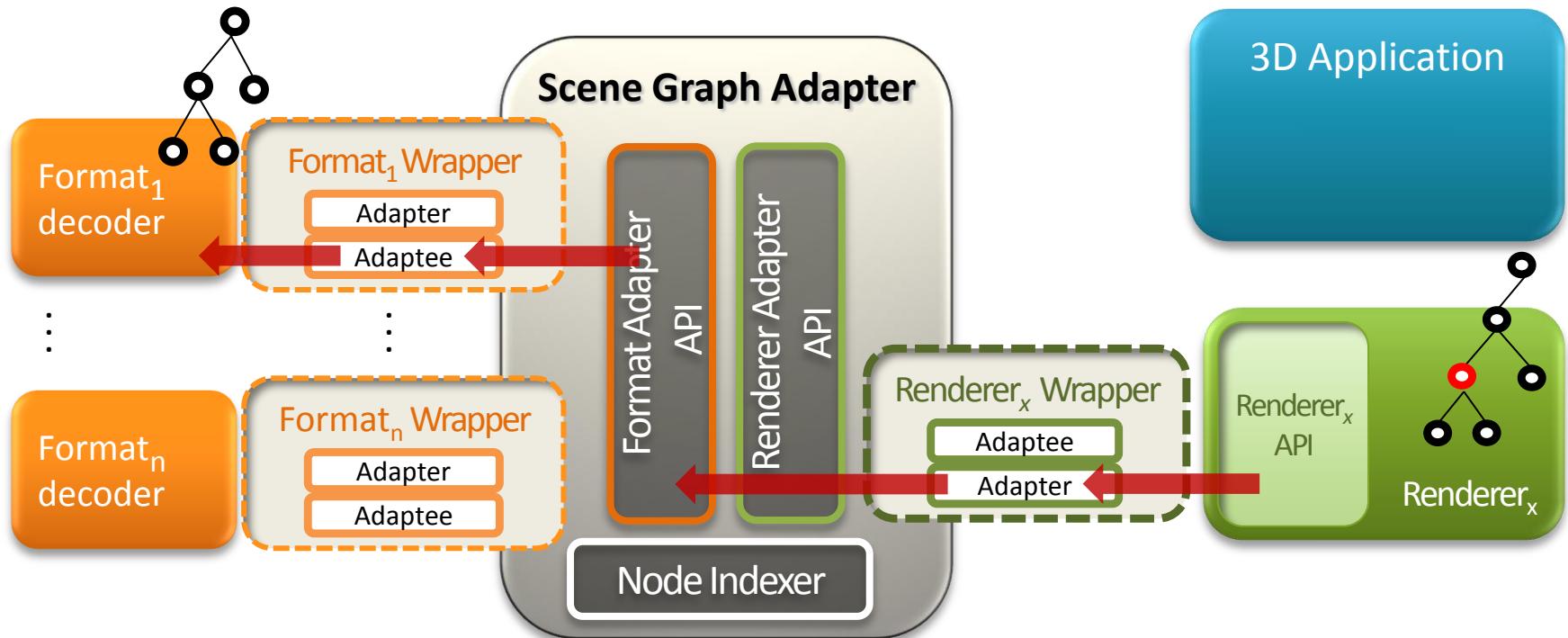
Functioning 2

Object picking :



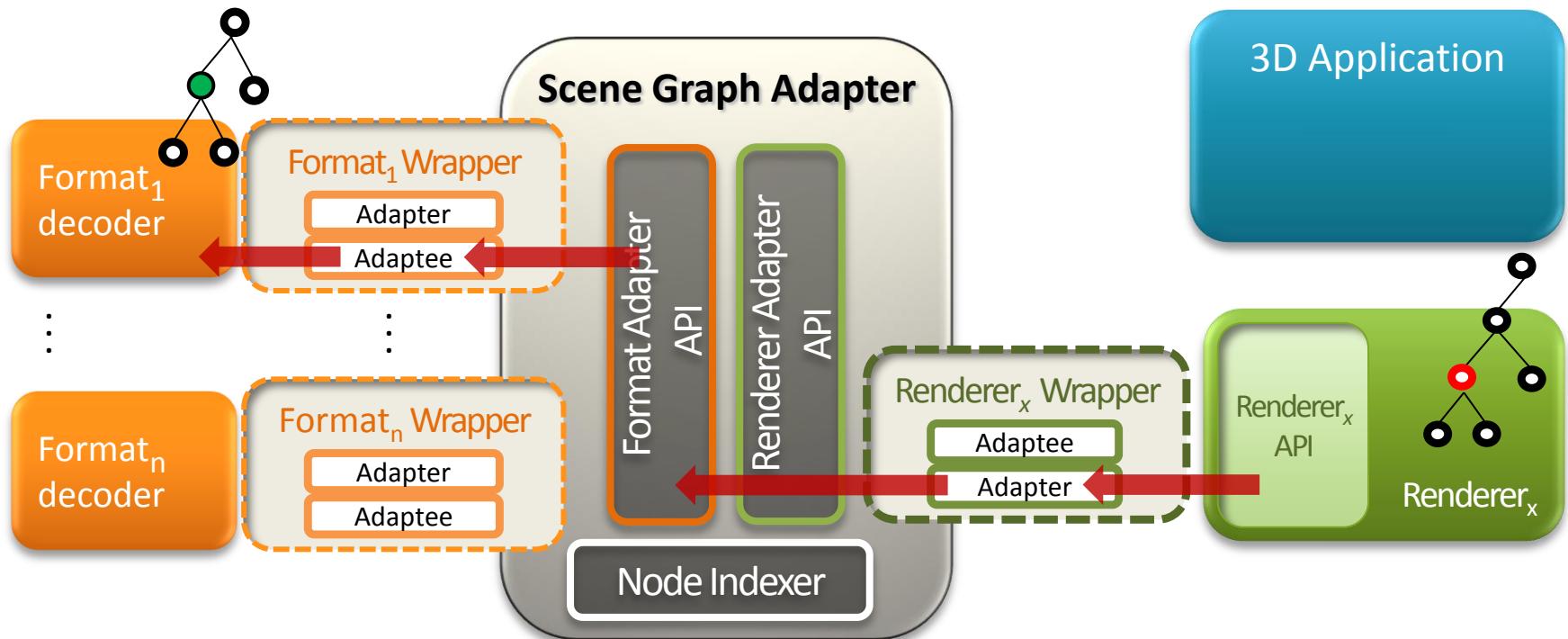
Functioning 2

Object picking :



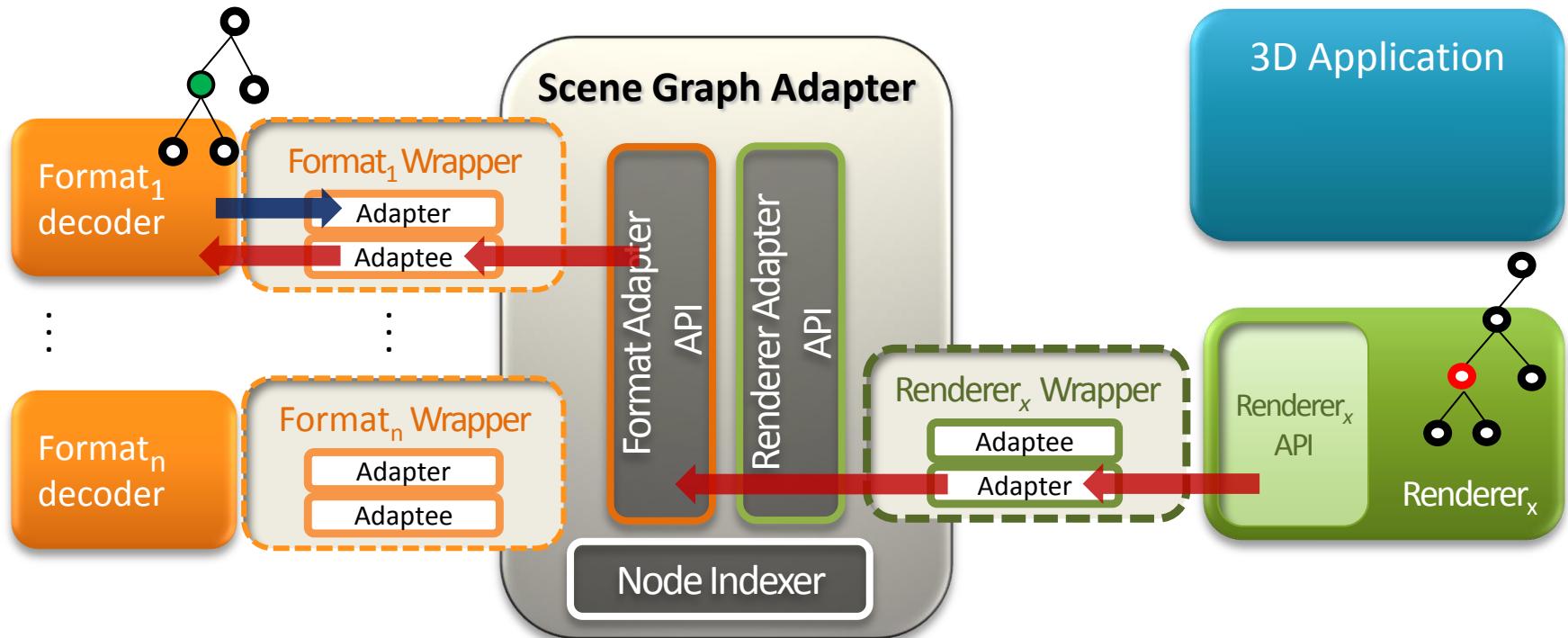
Functioning 2

Object picking :



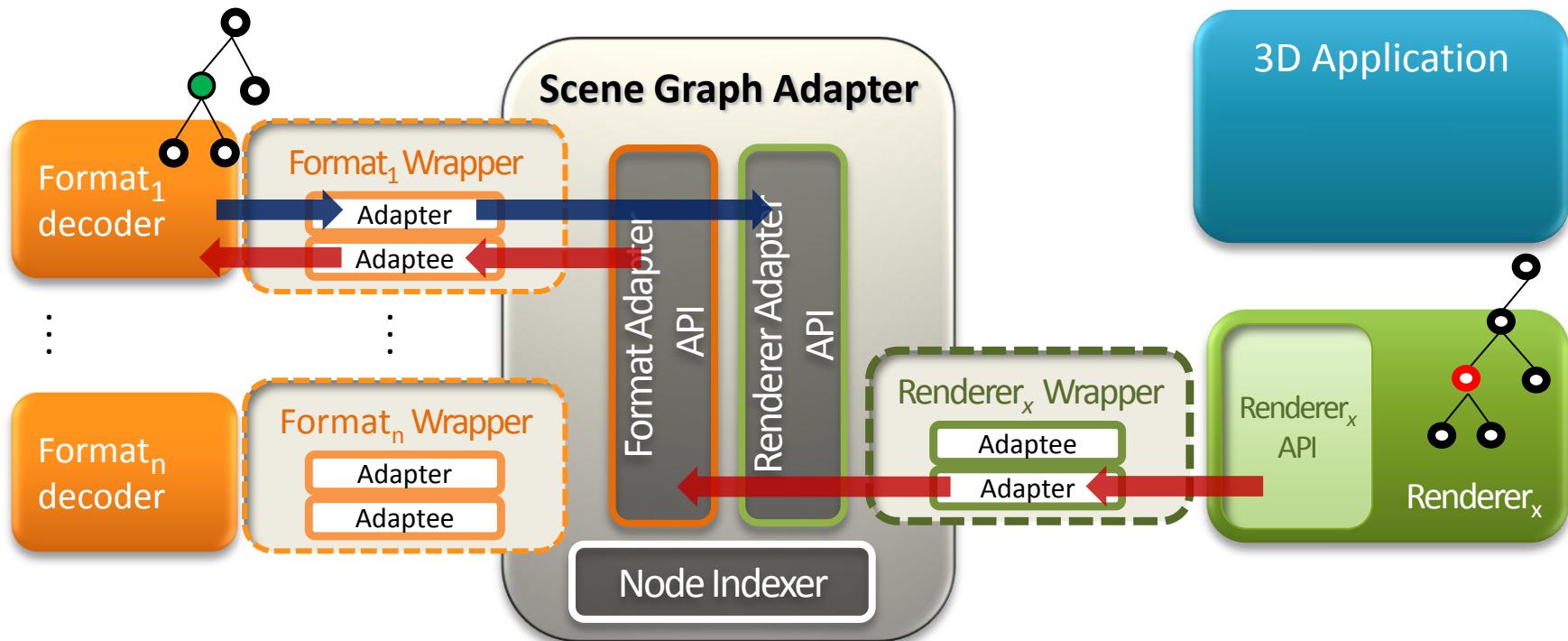
Functioning 2

Object picking :



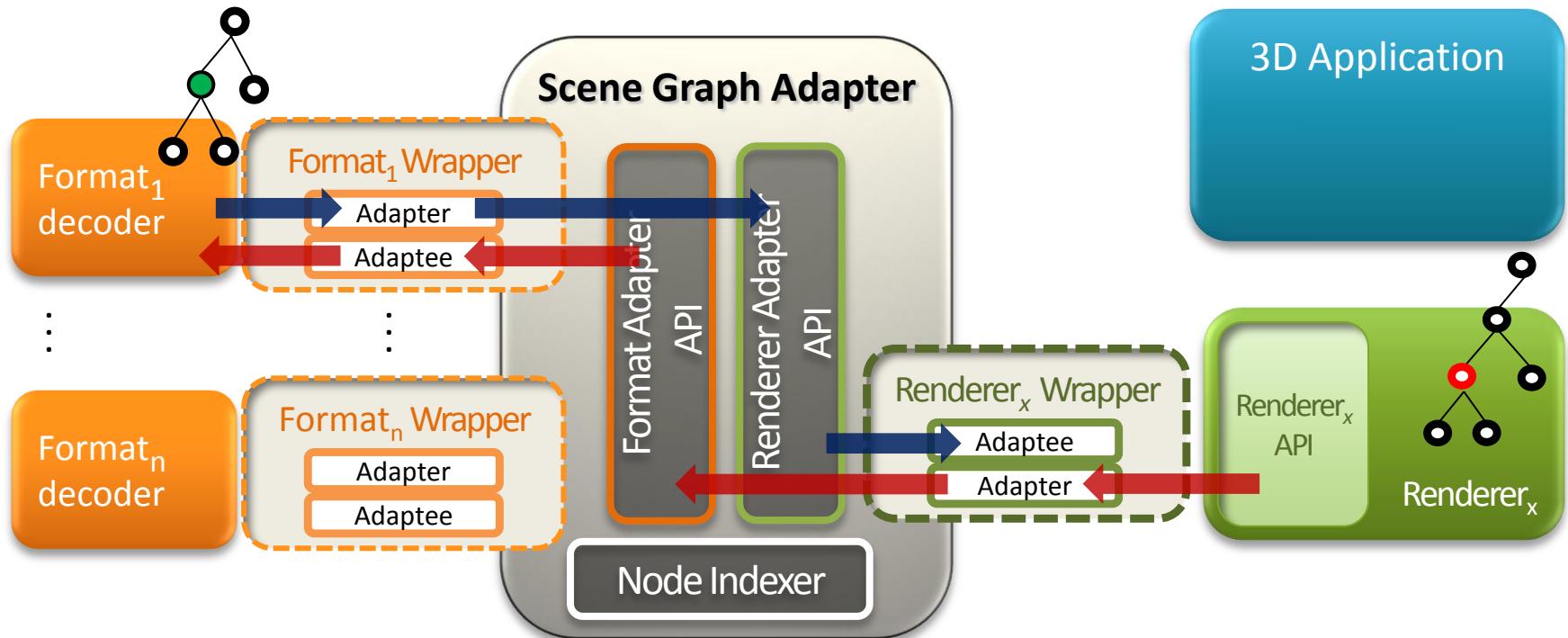
Functioning 2

Object picking :



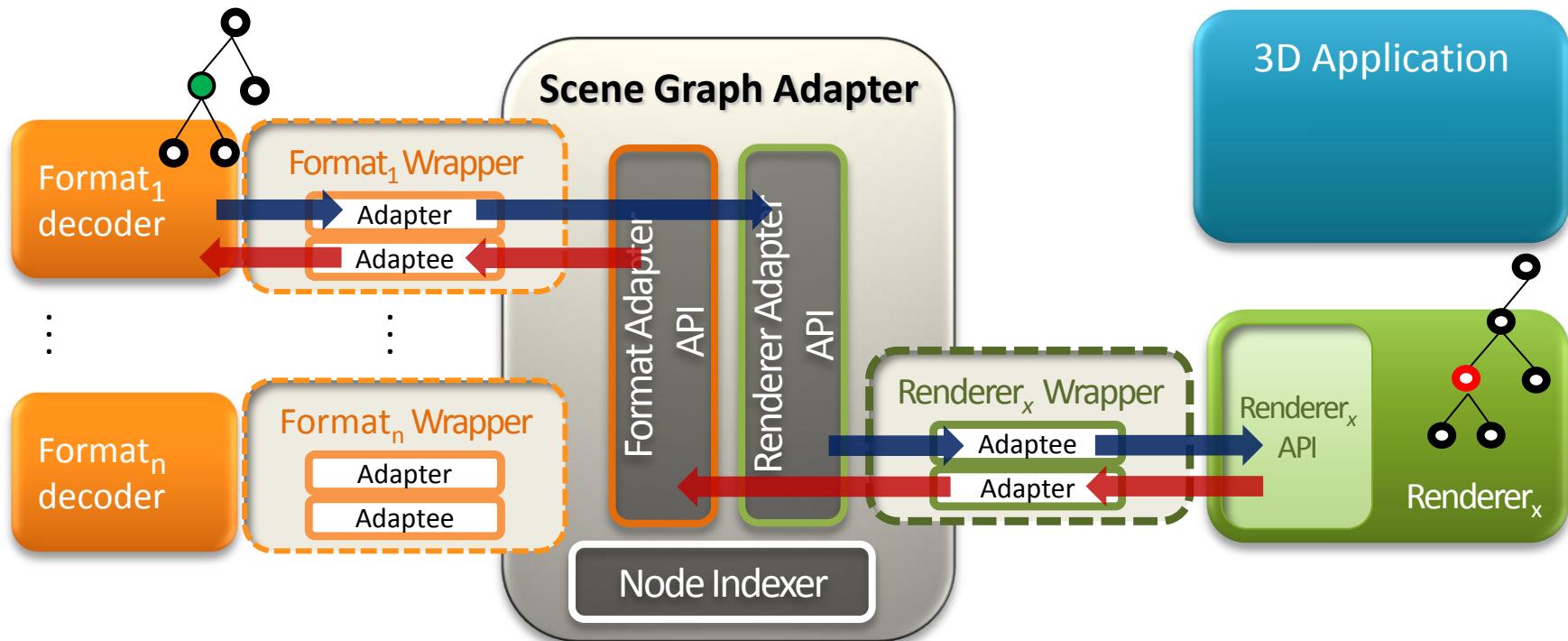
Functioning 2

Object picking :



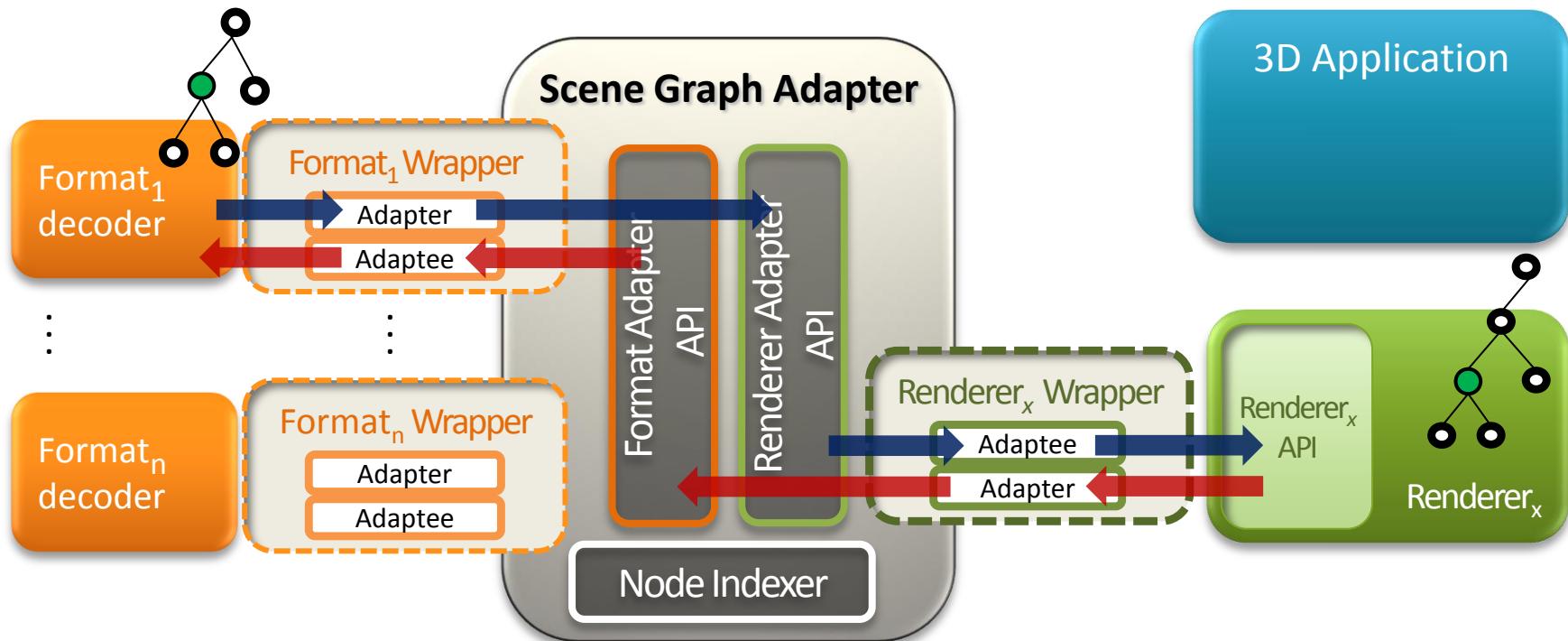
Functioning 2

Object picking :

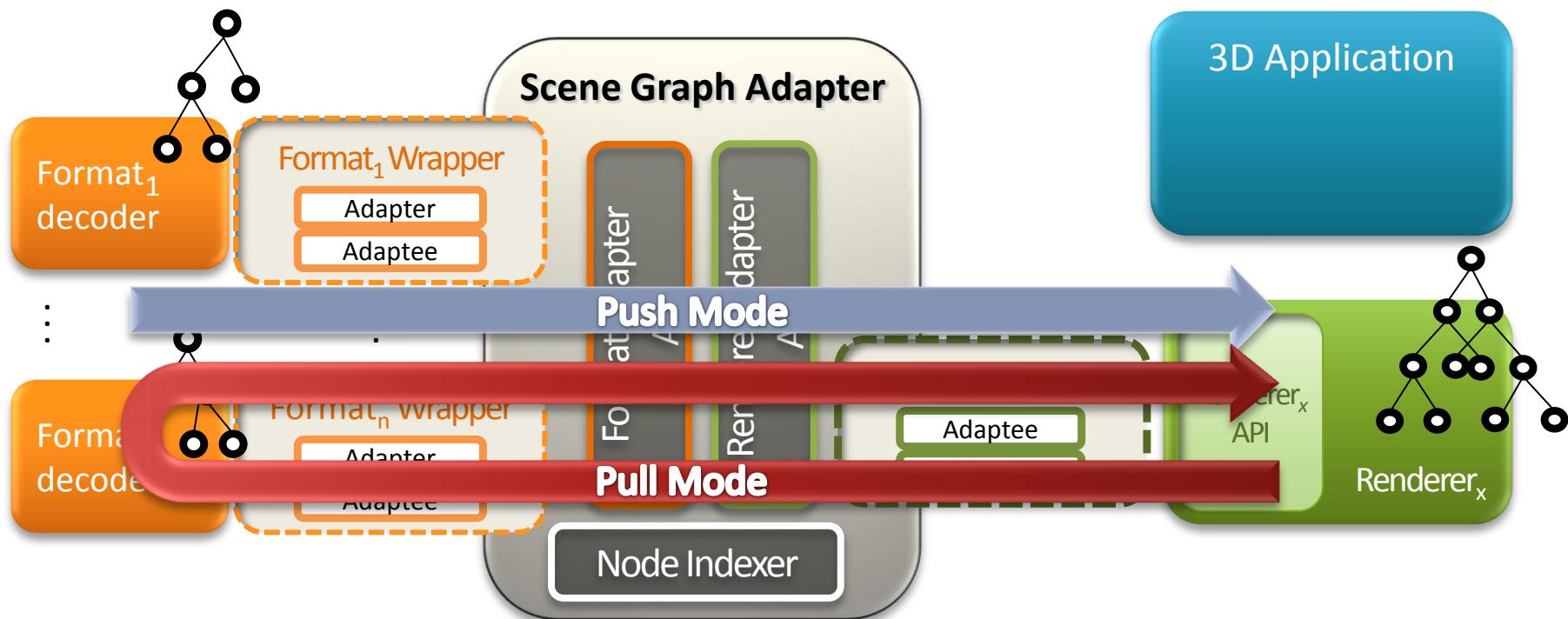


Functioning 2

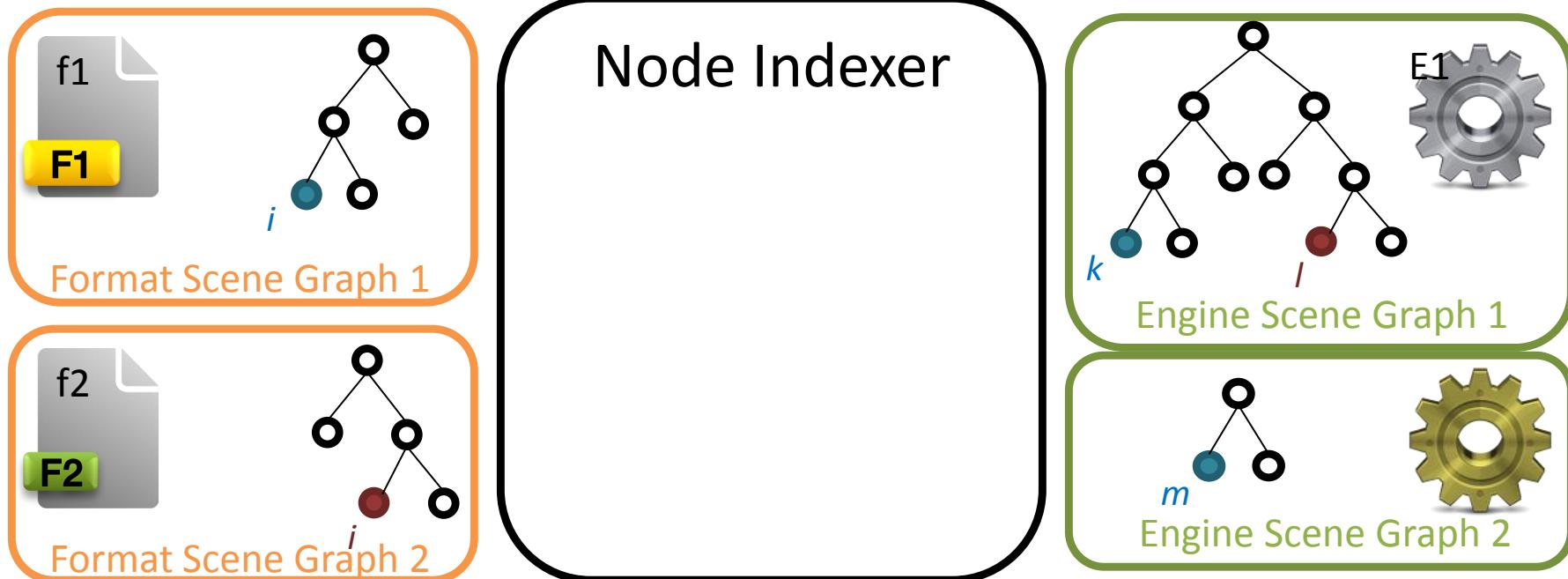
Object picking :



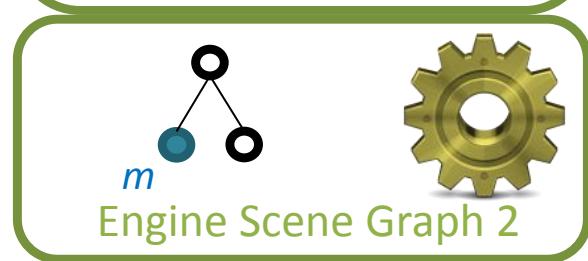
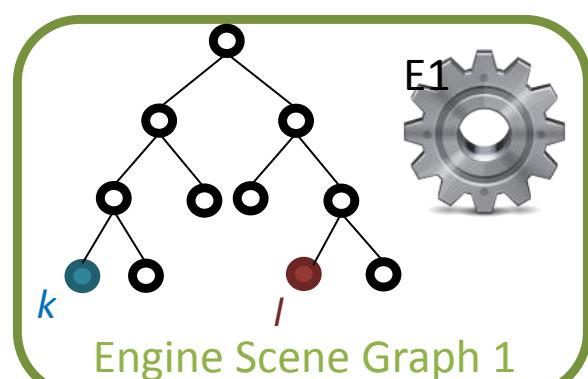
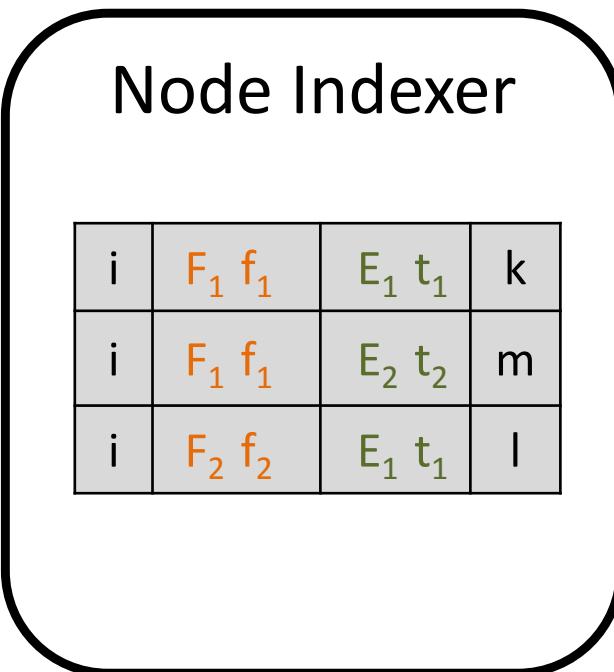
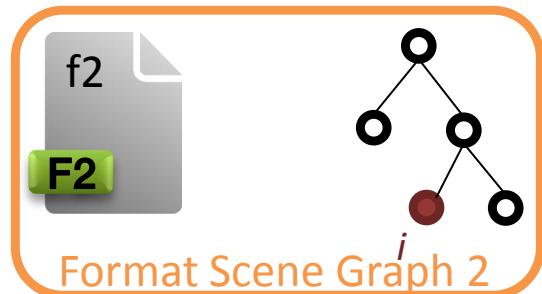
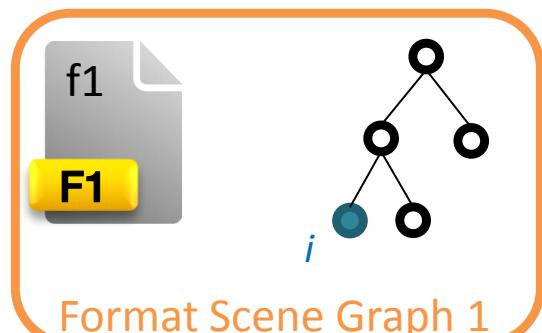
Functioning summary



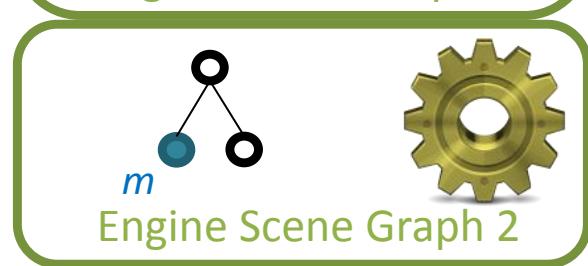
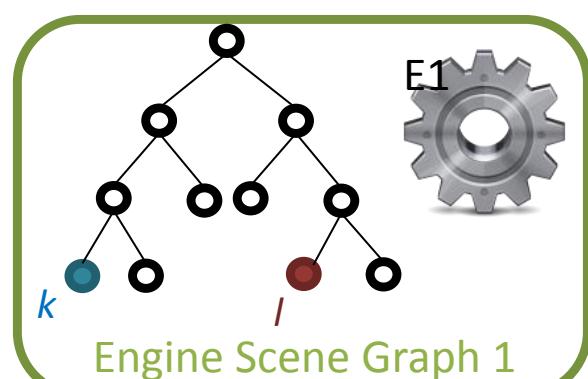
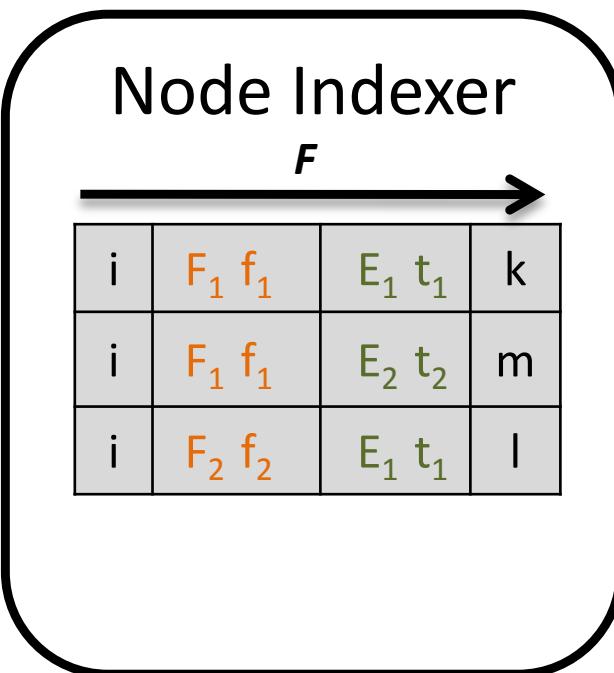
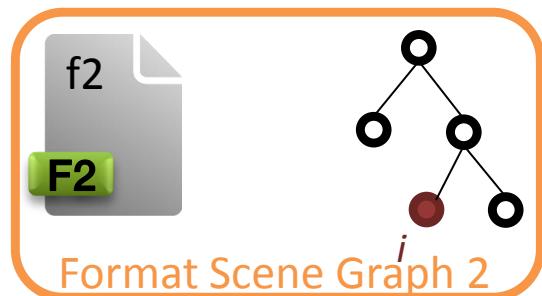
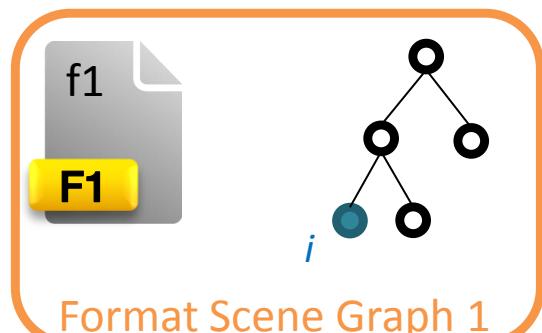
Scene Graph Adapter - Node Indexer



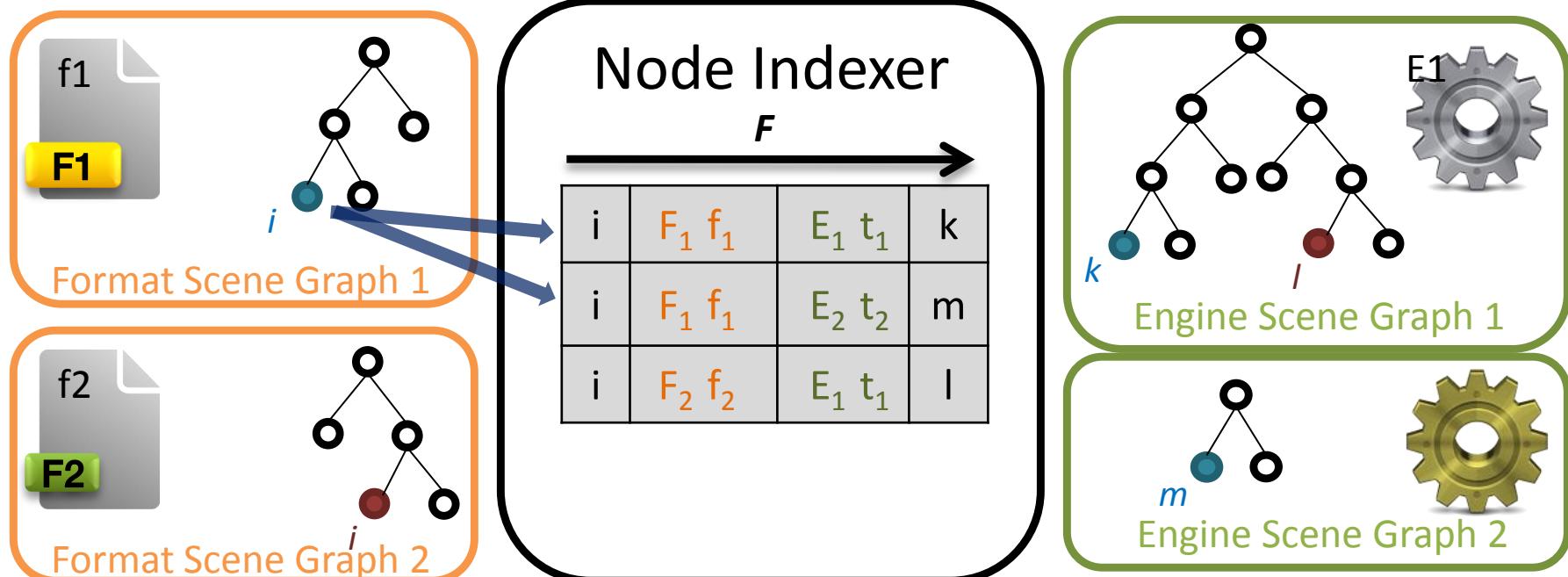
Scene Graph Adapter - Node Indexer



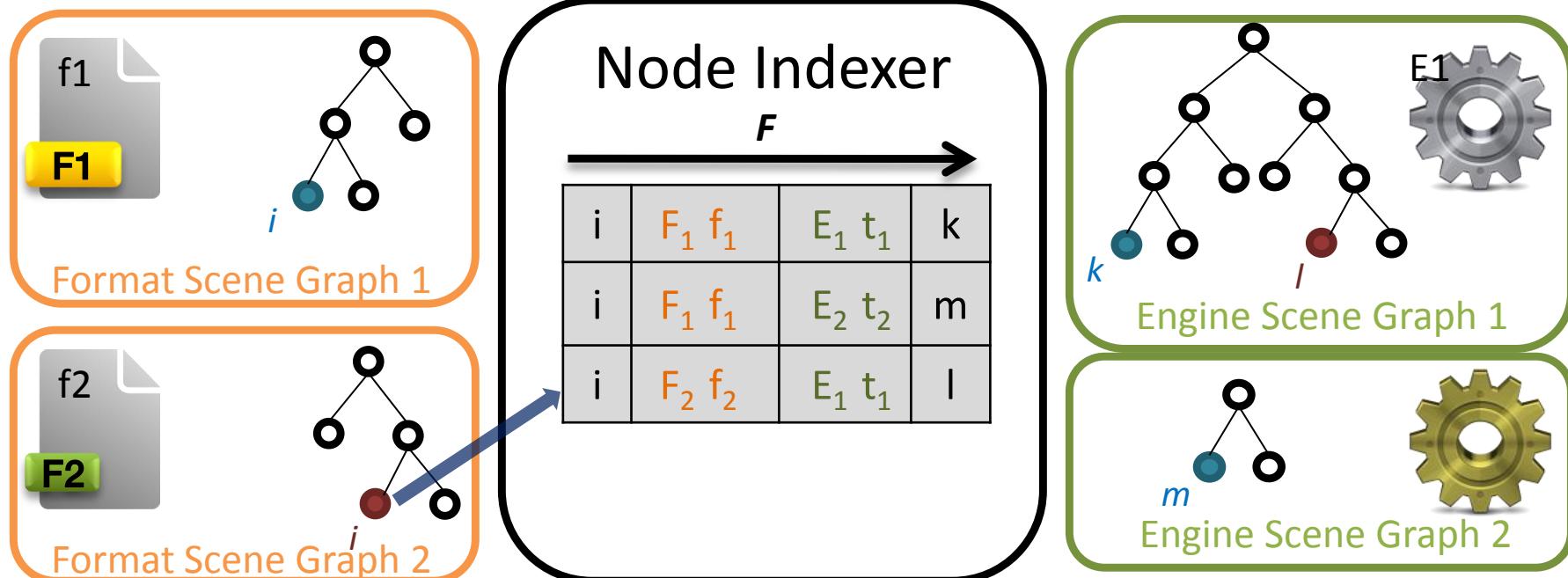
Scene Graph Adapter - Node Indexer



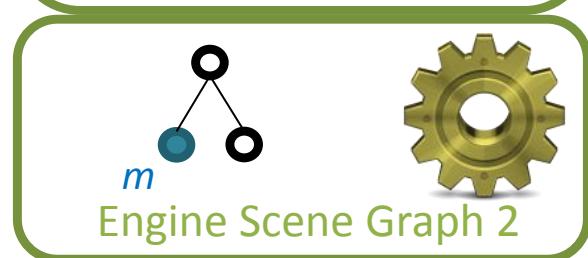
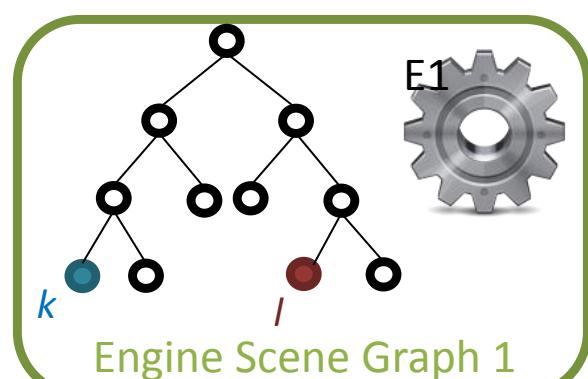
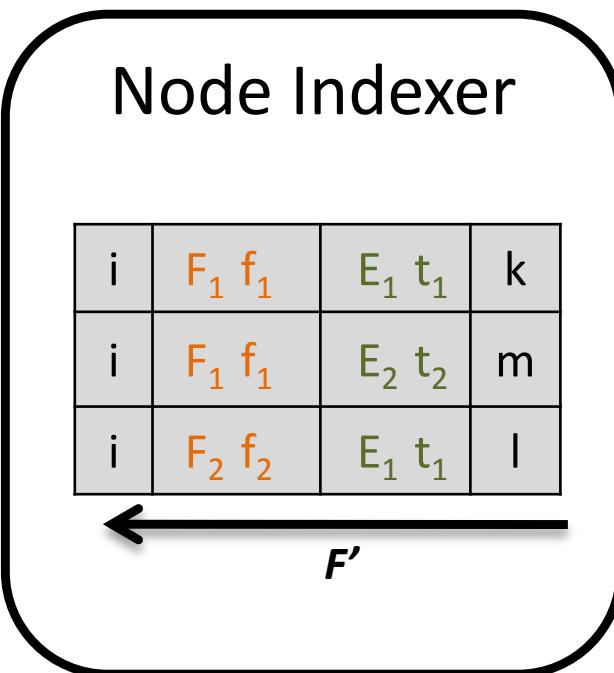
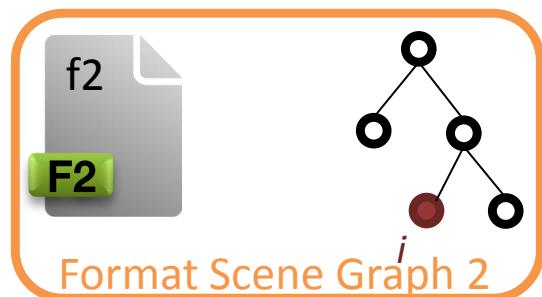
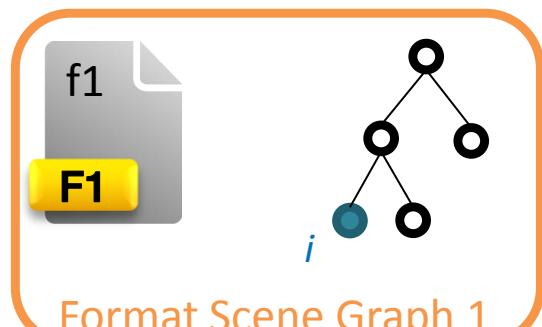
Scene Graph Adapter - Node Indexer



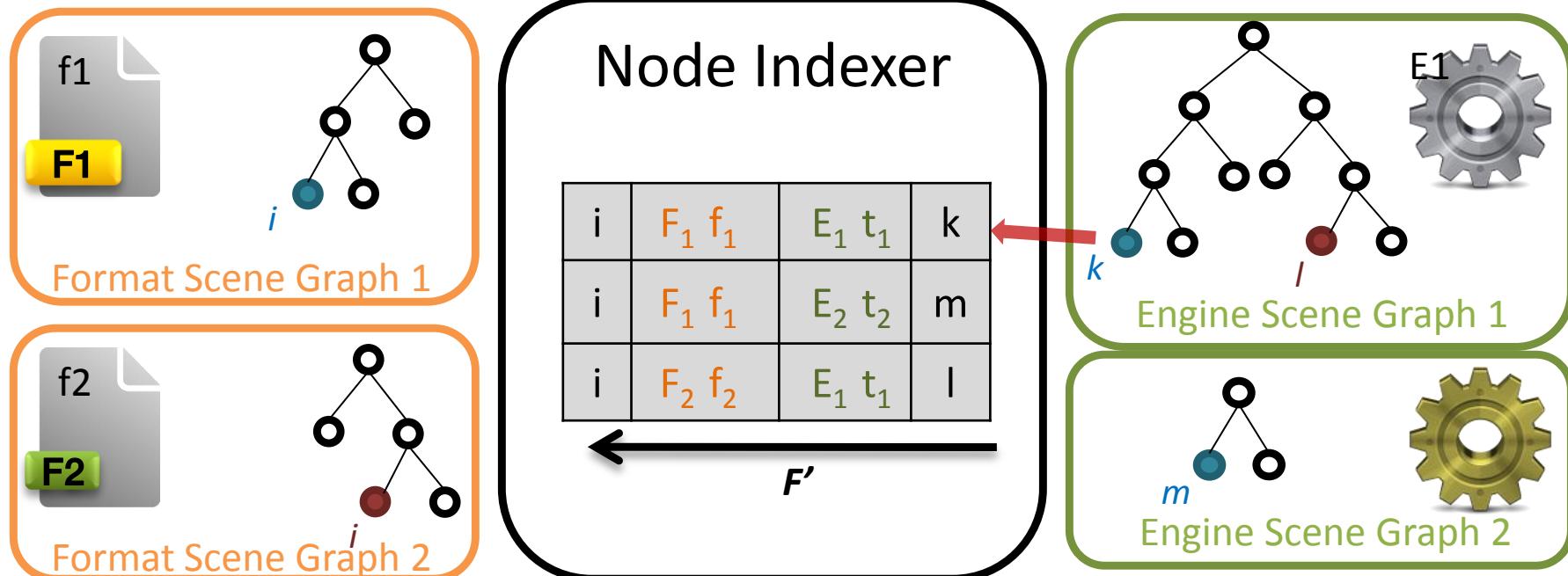
Scene Graph Adapter - Node Indexer



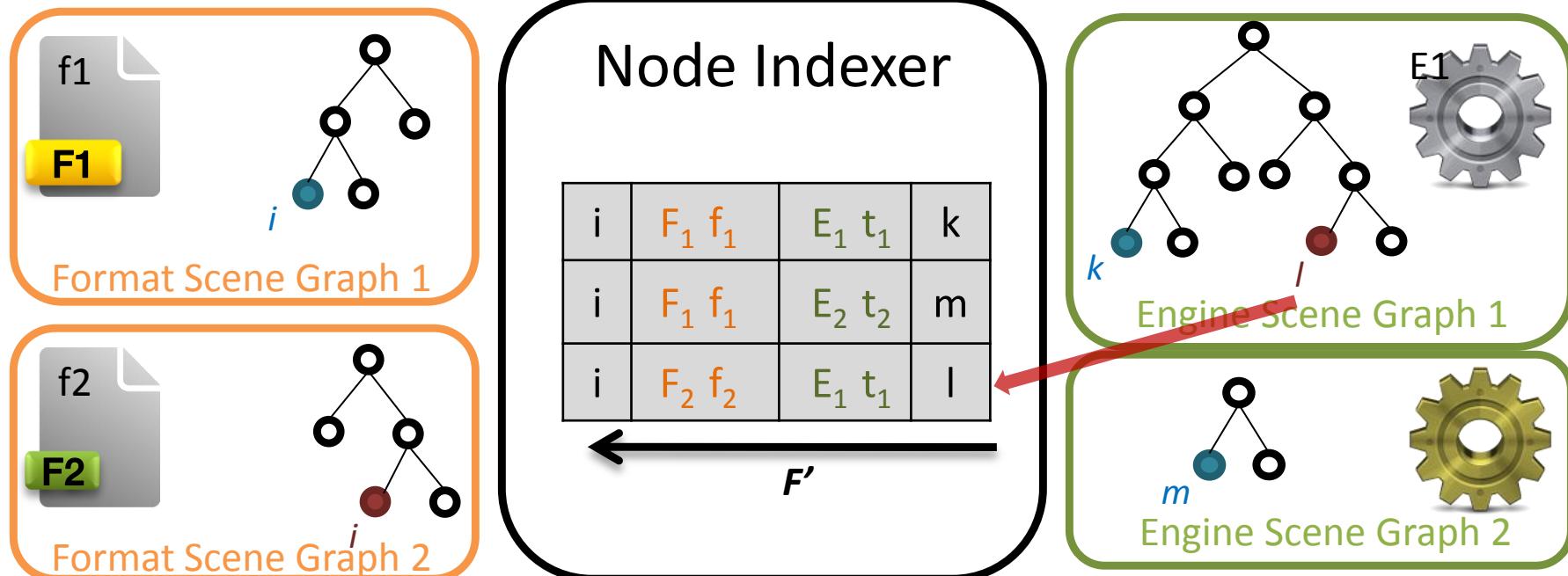
Scene Graph Adapter - Node Indexer



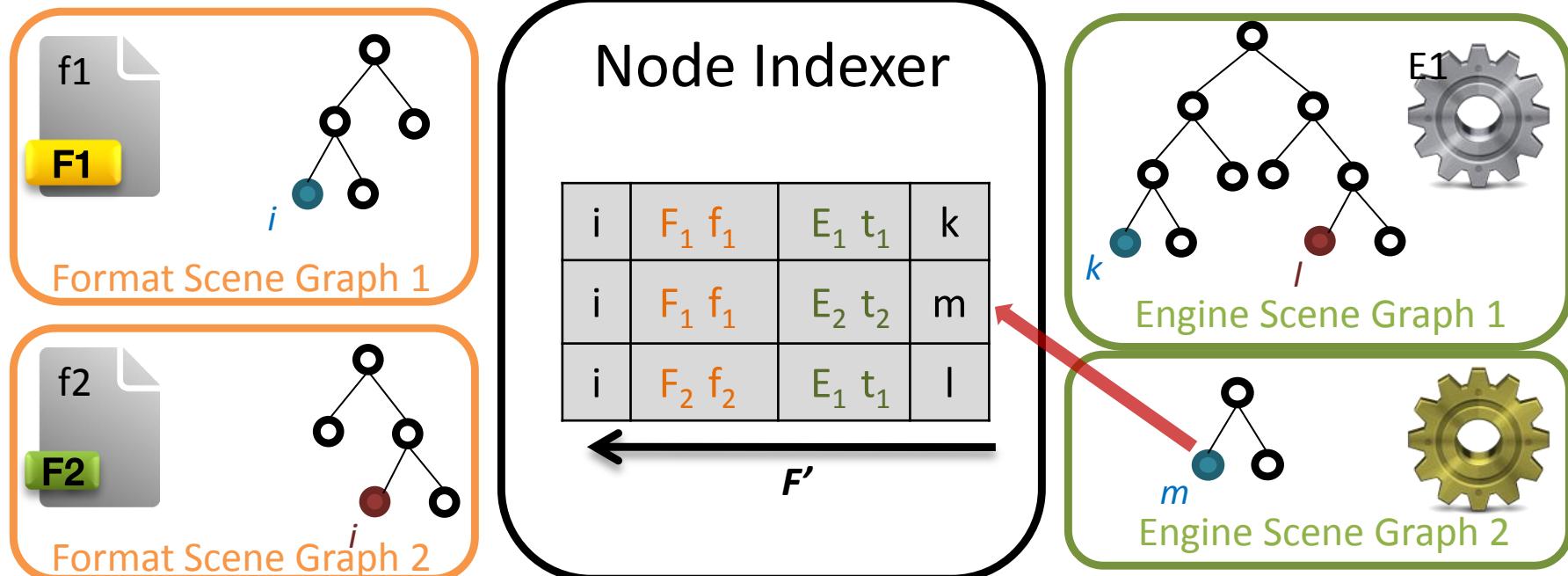
Scene Graph Adapter - Node Indexer



Scene Graph Adapter - Node Indexer



Scene Graph Adapter - Node Indexer



APIs details

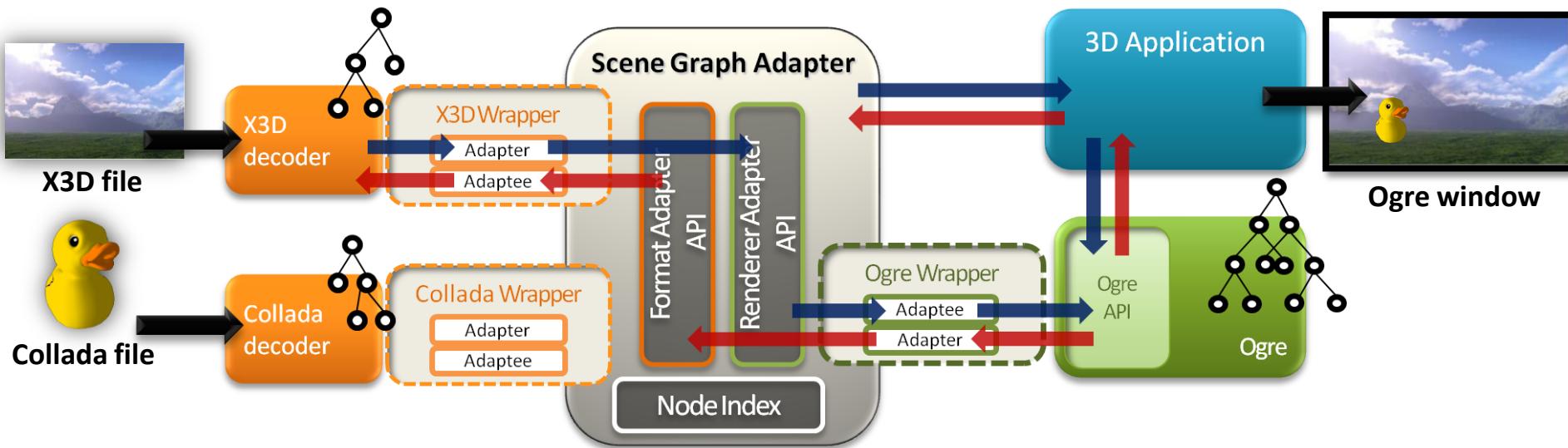
■ Format Adapter API :

- scene graph loading:
 - loadFile, loadNode, ...
- frame events:
 - setTime, setViewpoint, ...
- user events:
 - onClick, onDrag, ...
- ...

■ Renderer Adapter API :

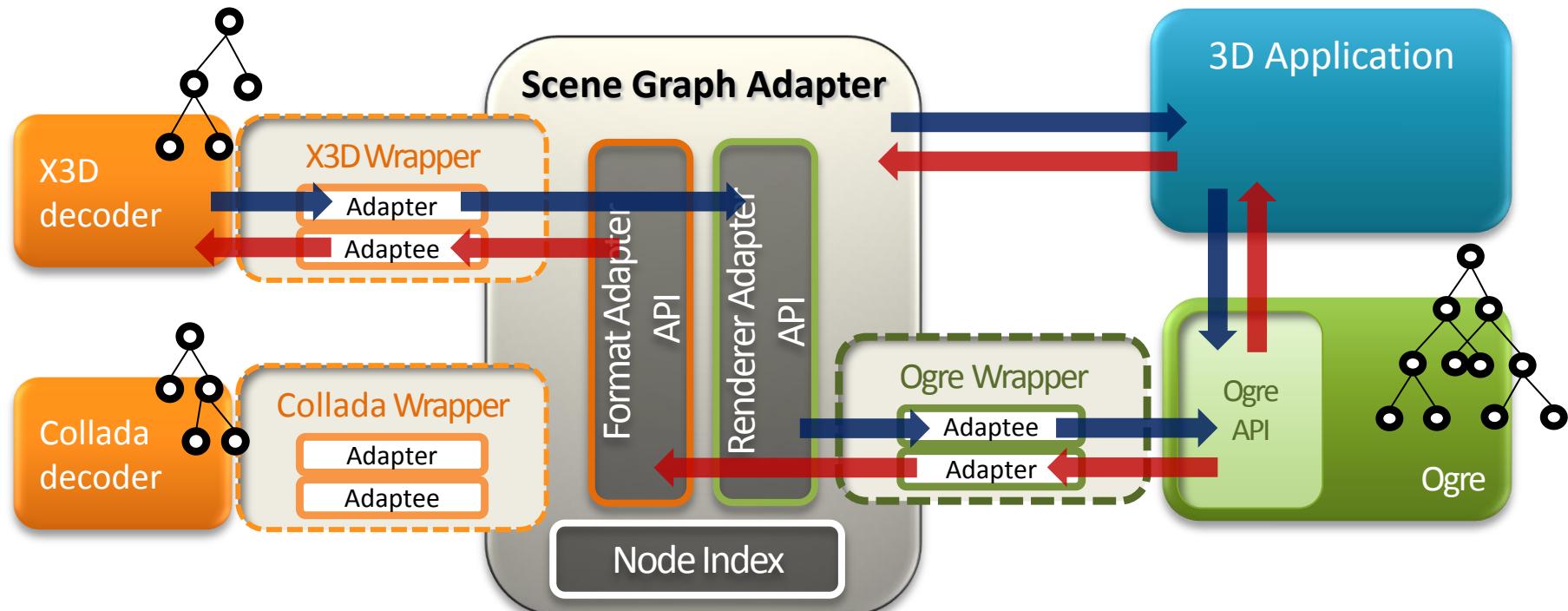
- environment settings:
 - setViewpoint, setBackground, setFog, ...
- camera settings:
 - createCamera, setCamera, ...
- scene graph settings :
 - createNode, deleteNode, ...
- geometry settings :
 - createSphere, createPolygonMesh
- transform settings:
 - createTransformNode, ...
- material settings :
 - createMaterial, setMaterial, ...
- ...

Implementation example

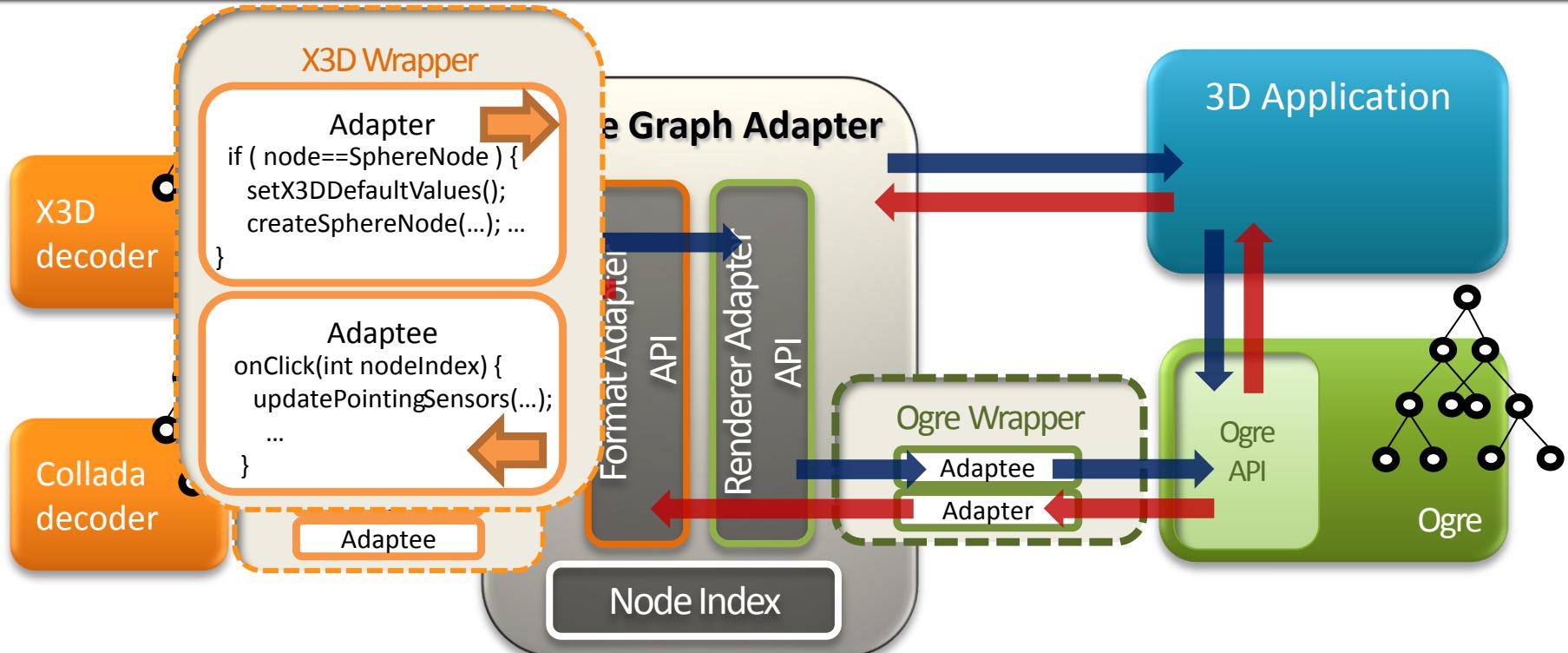


- Rendering engine :
 - Ogre
- Input formats :
 - X3D
 - Collada

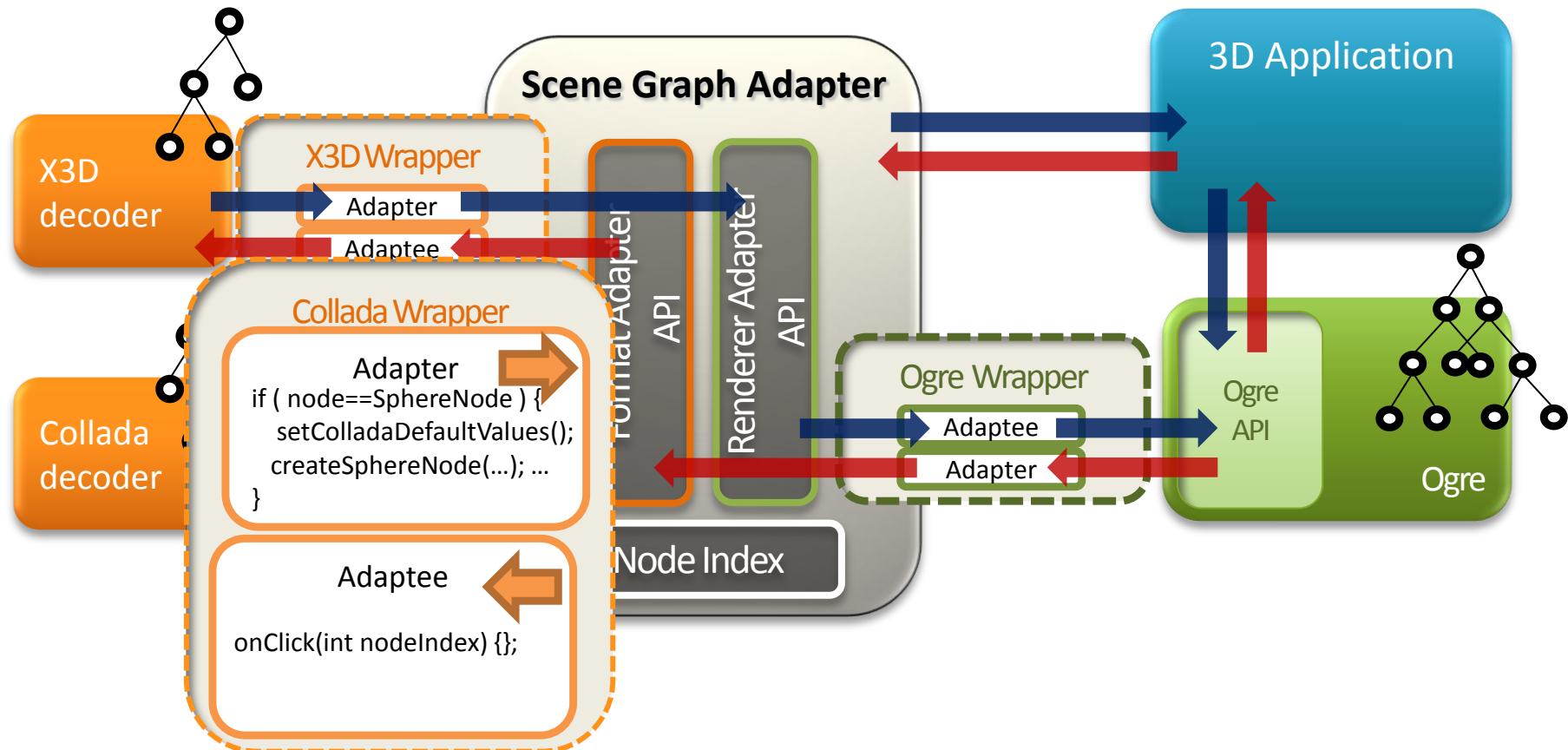
Implementation details



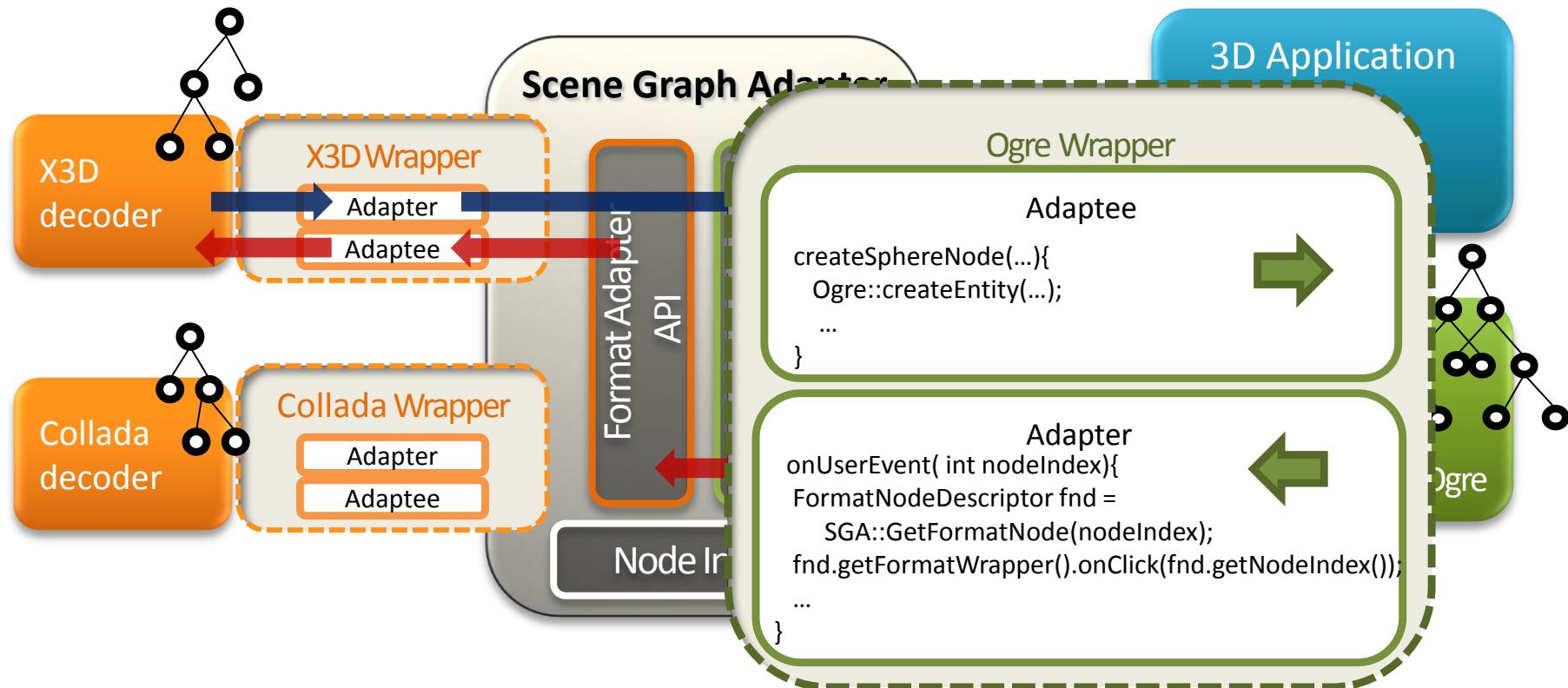
Implementation details



Implementation details



Implementation details



Scene Graph Adapter - Benefits

- Works with **every scene-graph-based 3D formats**
- Works with **every scene-graph-based rendering engine**
- **Extensible solution**
- **Reusable components: cumulative effort**

Future work

- Use other kinds of engines
 - physics engine, network engine, ...
- Investigate a WebGL-based rendering
- Enable file inlines without formats constraints
 - `<inline url="duck.dae" />`
- Handle interaction between input files
 - use an X3D interpolator to animate a Collada model

Thanks for your attention.

Questions ?

Contact : rozennb@gmail.com

Rozenn Bouville Berthelot - Jérôme Royan (*Orange Rennes, France*)

Thierry Duval - Bruno Arnaldi (*IRISA Rennes, France*)