08_Graphs

Scene Graphs

- Direct Acyclic Graph
- Adjacency Graph

Direct Acyclic Graph

shaders use this

animation state machine use this

- **Why Important**:
  - Scene graphs are the backbone of game engines like Unity or Unreal, representing the **hierarchical structure** of objects.
  - They handle **transformations**, **instancing**, and **visibility** efficiently.
- **Learn To**:
  - Traverse scene graphs for operations like toggling visibility or applying global transformations.
  - Understand shared nodes (e.g., meshes, materials, lights) and their impact on rendering.

Adjacency Graph

- **Why Important**:
  - Understanding how **vertices, edges, and faces** are connected is key for **mesh traversal** and **editing tools**.
- **Learn To**:
  - Identify neighboring vertices or faces (Adjacency Graph).
  - Optimize for loops like **edge traversal**, **normal calculation**, or **smoothing**.

Example Use Cases:

- Finding connected edges for a bevel tool.
- Building edge loops for mesh simplification.