Modeling Guidelines

Pick the right tool for the job

In general, subdivision surfaces have many advantages but sometimes you actually need a polygon mesh.

Subdivision surfaces

- **Pros**
    - Perfectly smooth at any resolution.
    - Can have sharp or semi-sharp features with creases and corners.
    - Can be very lightweight in memory.
    - Less displacement cracks (watertight)
- **Cons**
    - Different modeling techniques/skills
    - Need to tag sharp edges and vertices
    - Some topological constraints (quads or tris, no non-manifold geometry)

Polygon meshes

- **Pros**
    - Great for heavy geometry like 3D scanned mesh
    - Best for geometry that will be fractured / simmed.
    - Good for models with only sharp edges that don't need displacement
    - No microploygon generation if not displaced.
- **Cons**
    - Heavy models use a lot of memory, however small they may be in the image.
    - Shadow terminator may be visible on low resolution smoothed objects
    - Displacement cracks may be visible

NURBs

- **Pros**
    - Perfectly smooth at any resolution.
    - Lightweight in memory
- **Cons**
    - Quad-only modeling
    - Not supported in XPU
    - Not very popular nowadays

Suggested reading:

- Geometry
- Subdivision Surfaces
- Polygons

Subdivision modeling

- Always model the simplest control cage possible.
- Always view your model with smoothing or render it using PxrVisualizer.
    - You can even model in a live PxrVisualizer render !
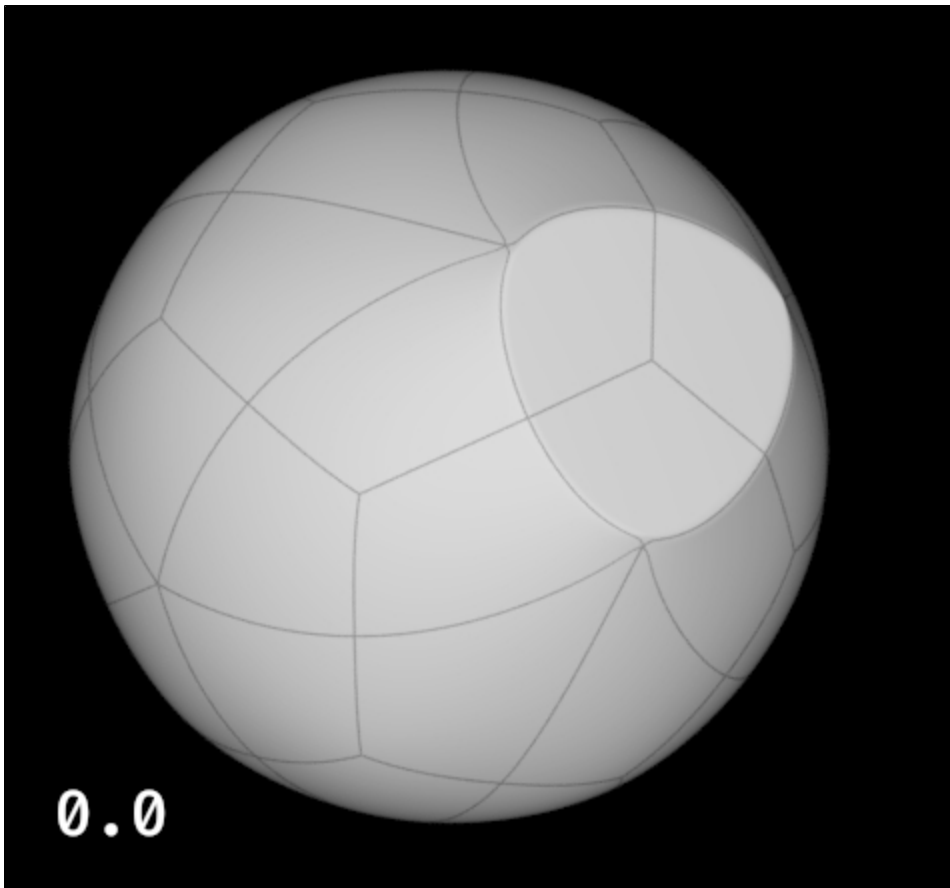- Use creases to make sharp or semi-sharp edges without adding extra edge loops.
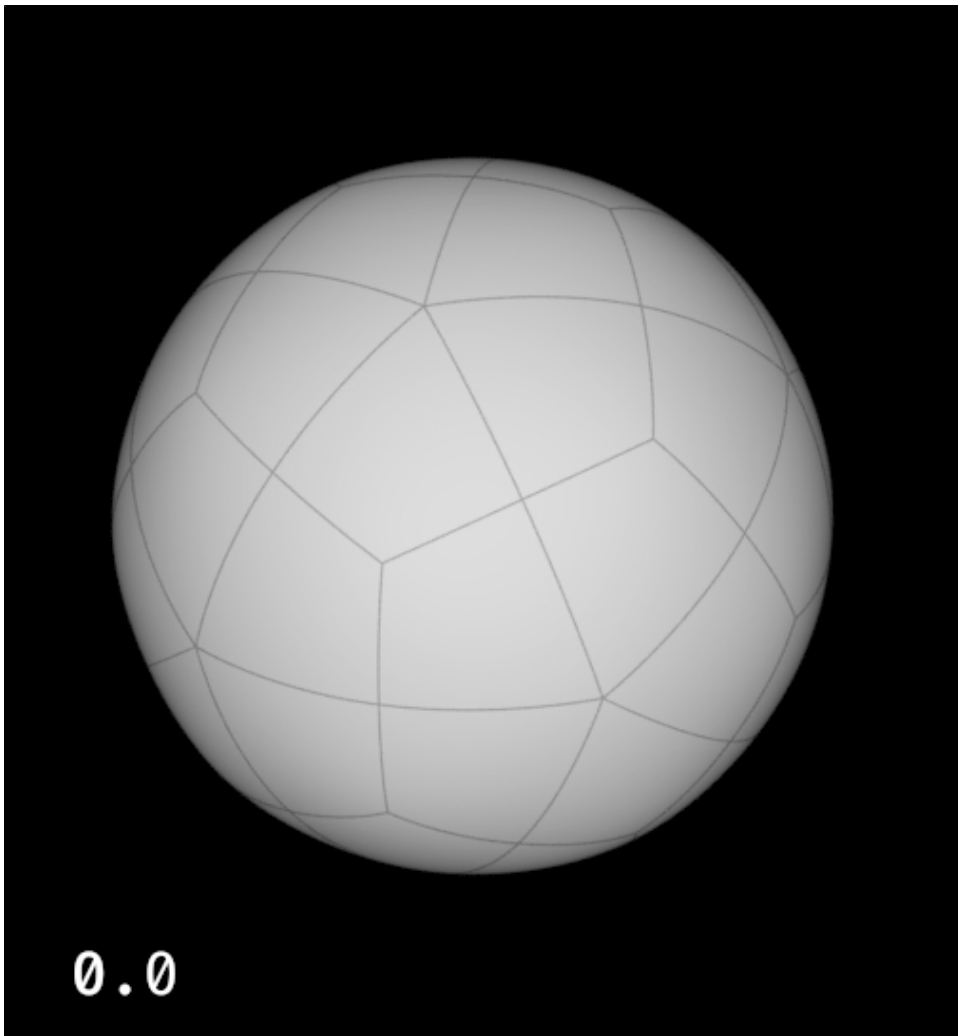
Subdiv Crease demo

We selected all the edges of an isocahedron and animated their crease value from 0.0 to 5.0.

All edges can have different crease values and still keep smooth transitions accross the final surface.

- Corners work like creases for vertices.

Subdiv Corner demo

In this animation, 3 edges of a triangular face have a crease value of 5.0.

We increase the triangle's vertices corner values from 0 to 5.0.

Note the very smooth transitions around the tightened areas.

- There are 3 type of subdivision surfaces.
    - Catmull-Clark is the most flexible and popular one.
    - Loop is a triangle-based subdivision surface.
        - No creases or corners
        - useful for cloth simulations

○ Bilinear is like a catmull-clark with infinitely hard creases on all edges

catmull-clark
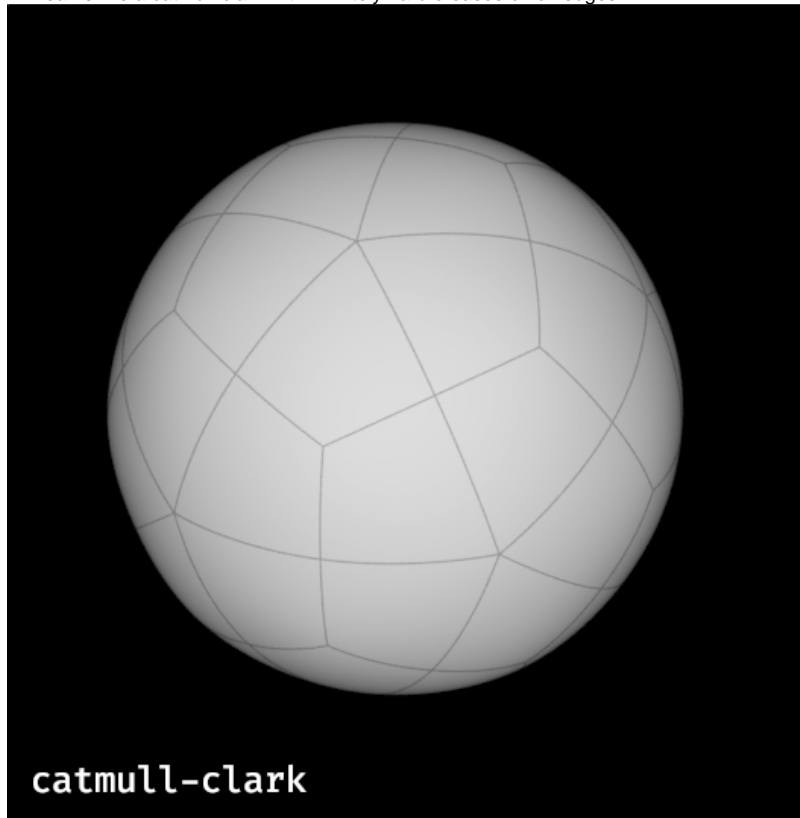
Tesselation

If you are coming from another renderer where you have to set subdivision levels, forget about it: **tesselation of surfaces is automatic** in RenderMan (we call it **dicing**).

The main control is the **micropolygon length**.

- By default, it is expressed in pixels and a micropolygonlength of 1 (the default) means that every micropolygon will be approximately 1 pixel wide.
- If you don't want the tesselation to change dynamically, you can set it in object or world space or define a dicing camera.

⊘ We recommend disabling "raster-oriented dicing", unless you see some artefacts.
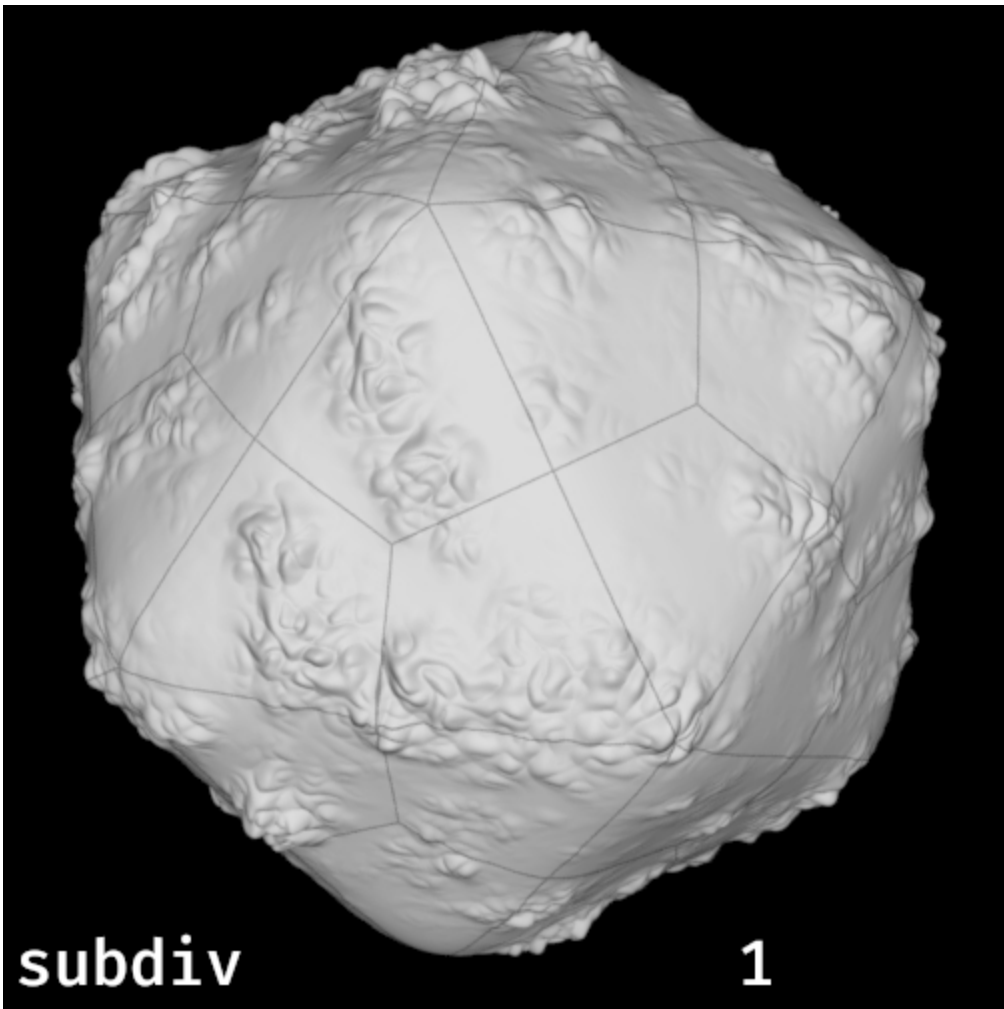
subdiv 1

Subdiv vs. Polygon Dicing demo

We compare a creased displaced catmull-clark subdivision surface with an equivalent polygon mesh.

Increasing the micropolygon length is equivalent to making the object's size smaller in the image.

- When the object is large, it is finely tesselated into milions of micropolygons, showing fine displacement detail.
- When the object becomes smaller than a pixel, only the base mesh remains.

Micropolygon Length works a bit like an "automatic level of detail" and preserves memory when an object is so small in screen that it doesn't need an fully detailed representation.

> ⓘ Notice that the subdiv degrades to a much lower number of faces, preserving memory and speeding up raytracing.

This is what we call **Data Amplification**, when a compact geometric representation can be augmented to create more detail on demand.

Model detailing

Often, it is better to rely on displacement and bump mapping to add details.

- Displacement allows you to add larger geometric features that impact your object's outline and shadowing.
  - Displacement impacts time to first pixel (micropolygon generation) but doesn't slow down shading.
- Bump or normal maps will add small scale details
  - Bump mapping add small cost during shading (recomputed at each ray hit).
  - Note that our bump mapping nodes will NOT compute bump if the incoming ray is too "blurry", which means an absence of bump will go un-noticed.

If you have modeled a highly detailed object in ZBrush, you should consider:

- retopologizing to create a lightweight model
- extracting a displacement map from the high-resolution mesh.

> ✓ Our displacement nodes have a "model displacement" input to connect the extracted displacement map.

Memory usage and performance

Polygon meshes

- Shortest **time to first pixel** (TTFP) if not displaced.
- When displaced, the mesh will be diced and micropolygons generated.
- Un-displaced heavy meshes use the same amount of RAM, whether they are full screen or 2 pixels high.

Subdivision surfaces

- Slightly longer TTFP but dicing is multi-threaded and efficient.
- Use a bit more memory than meshes for the same number of vertices.
- A sparsely modeled subdiv will only contain the base mesh faces when 2 pixels high.
  - This adaptivity lowers memory usage and speed up raytracing.