

Optimization: Polygon Budget

In realtime 3d applications, most of the graphics on screen are defined by a series of triangles connected together to form solid-looking objects.

One of the main problems with models transferred from architectural programs is the lack of optimization, especially when it comes to internal geometry and curved surfaces. This extra detail is essential for the model's original use. Unfortunately, a single building can come in into max as tens of millions of polygons, which is extremely excessive for a realtime scene, and most of that is for detail the user won't ever see.

Modern video games, on the other hand, are modeled so efficiently that they can display entire cities complete with vehicles and characters without the user ever seeing more than 2 million triangles on screen at any given time. The ideal Vizard scene should follow most of the same conventions.

Checking the number of triangles in 3ds Max

1. Open the model and press 7 on the keyboard
2. The upper left corner of the screen should now show the number of polygons.
3. This is different from the number of triangles (polys are made of triangles).
4. Go to View -> Viewport Configuration -> Statistics
5. Turn off Polygon Count, Turn on Triangle Count, and switch the mode from Total to Total + Selection
6. This will show the triangle counts for the scene and any object you select. Press OK

Checking the number of triangles in Vizard

1. Load the model in Vizard and press F4 twice.
2. If the polycount is over 2 million, and the FPS is under 60, you'll need to remove some objects or reduce their detail (eg. With the ProOptimize modifier)

Notes

- If a triangle doesn't contribute to the silhouette of an object, you usually don't need it – the extra detail should be created through texture instead. To avoid overdetailing models, pick a set distance such as 2 or 4 meters as a reference, then only model details that will be noticeable from that far away.
- Get rid of internal and hidden geometry whenever possible. Any geometry that is never seen by the user still has to be processed, even if it's behind another object. These objects should be removed. The extra detail should instead be used on things the user can see.
- Use transparent textures for plants and chain link fences. There is a strong tendency for plant props in architecture programs to model out every single leaf or blade of grass. In some cases, millions of triangles are used for a small number of plants that contribute very little to the overall scene. Rather than getting rid of the plants altogether, remake them using transparent polygons that use transparency to show clusters of leaves. Ideally, study the way plant props are handled in modern video games.
- Within the area accessible to the user, geometric detail should be distributed fairly evenly. This is a bit more complex than the way textures are distributed since some surfaces need more triangles to describe the surface than others
- Most geometric detail should be focused around the area that the user can get close to. Background areas that never get any closer than 50 meters do not need nearly as much geometric detail as props in the active area.
- In many cases round objects from architecture programs, such as rails and pipes, come in with an excessive amount of tessellation, sometimes with as many as 2000 sides. For realtime applications, it is much more common to give cylinders somewhere between 8 and 16 sides. The extra detail can probably be better used elsewhere.