

Face (3D Object)

In 3D graphics the term *Face* refers to an object described by three [vertizes](#). Essentially it is a triangle in three-dimensional space. As such it has a surface but no volume. The two sides of the face are independently attributed. For example can one side (face) of the triangle have a different [texture](#) than the other (or even none while the other has one). Without a texture or some sort of filling color the face is essentially invisible (similar to an edge or a vertex).

For resource saving reasons often only one side of the face has a texture or color while the other has none. A 3D object that is only shown from the outside will often never show its inside textures (e.g. a solid wall with an actual thickness), so there is no need for them. In other circumstances they might be present (e.g. a wall-like separation that has no thickness). When using a no-clipping mode in 3D games, where you can move freely in the 3D space, allowing you to see world elements from the other side or from their inside, you can observe this very impressively. From the players perspective massive walls just optically vanish, as the face shown to the player has no filling whatsoever.

Faces make up [meshes](#), three-dimensional objects with many interconnected faces. While optically complex surfaces may look like any other shape, meshes are always made up of triangle-shaped faces. For example even a simple square is not modeled by four vertizes giving four edges, but with five edges. The four usually expected edges are present, plus one diagonally placed one, connecting two of the most distant vertizes with one another. This separates the square into two triangles. Similar considerations enter any 3D object/mesh. Everything is made up of triangles.

This also implies, that there are no real gradually changing shapes, like perfectly round circles or balls. All these objects are still approximated up by triangles. So they have a quantization aspect behind them, that makes objects »edgy«. The »edgyness« can be reduced by using more triangles, giving it more detail, up to the point where they are barely noticeable or effectively invisible. But this in turn requires more memory and more processing resources to [render](#) the object as it requires more faces made up of more vertizes. So in reality objects are always a trade-off between looking as good as possible while making the task of rendering them doable with the system resources available.

[[Games Database](#)] [[Game Tech](#)]

From:

<https://www.mobile-infanterie.de/wiki/> - [mwohlauer.d-n-s.name](#) / www.mobile-infanterie.de

Permanent link:

https://www.mobile-infanterie.de/wiki/doku.php?id=en:game_tech:face&rev=1704661041

Last update: **2024-01-07-20-57**

