Step 2: Occlusion culling

Even though we can remove all polygons outside the vieweing frustum, polygons within often occlude each other.

How do you know what polygons in the viewing frustum are hidden?

- Portals
- Potentially Visible Set

Information Coding / Computer Graphics, ISY, LiTH

Cells and portals method

(often referred to only as “portals”)

Suitable for buildings, with many enclosures.

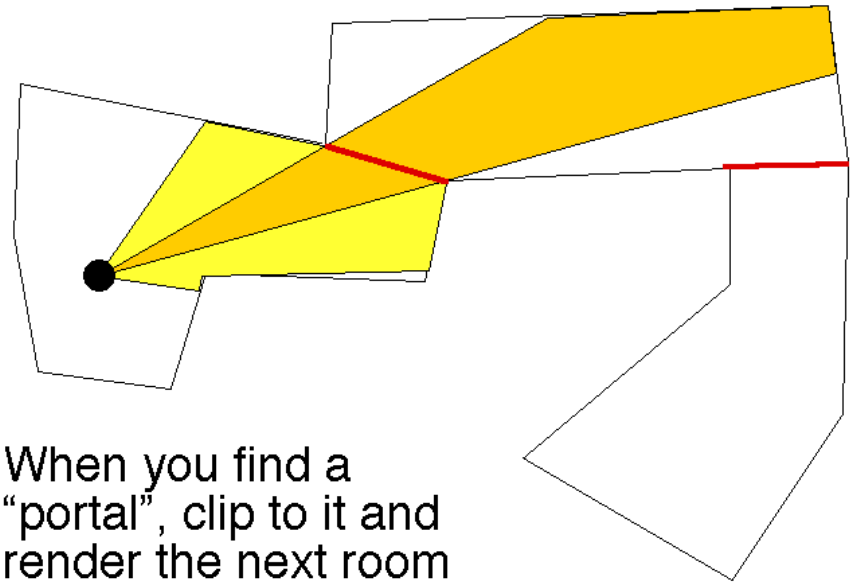
Split the world into smaller parts, check for the “portals” between them. (Dark Forces, Tomb Raider)

Each portal is a branch in an adjacency graph

Straight-forward and simple, but inefficient for outdoor scenes.

Portals

Polygons are grouped into cells, “rooms”

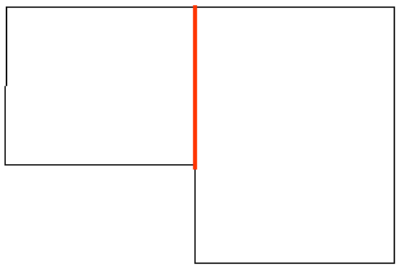


When you find a “portal”, clip to it and render the next room

Information Coding / Computer Graphics, ISY, LiTH

Real-world example: Dark Forces

Level editor reveals portal-based engine



Edit levels by drawing 2D polygons, connect them as portals

Notable limitation in game: Two windows/openings can never be on top of each other!

Potentially visible set (PVS)

A bit list for some part of the world (cell), specifying what polygons may be visible. (Quake)

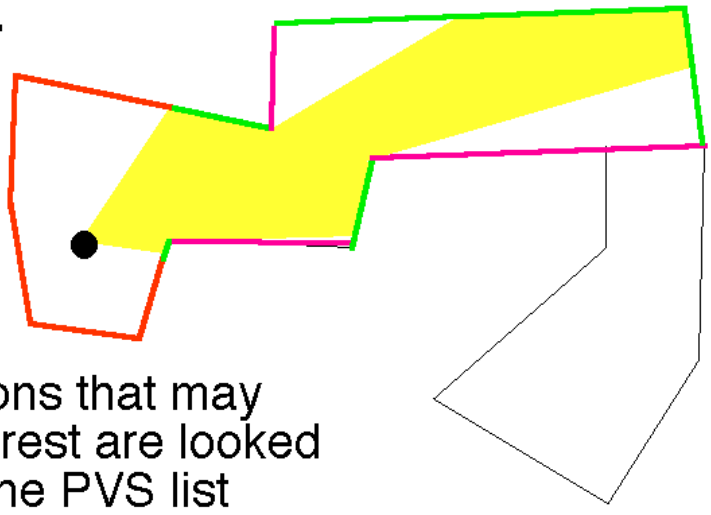
The list is huge, but can be compressed.

Pre-compute the list for a static scene.

Use BSP trees for creating cells automatically.

Potentially Visible Set

More general method, faster for very complex scenes.



All polygons that may be of interest are looked up from the PVS list

Pre-generating the PVS

Done either for a point or for a cell

1) Image-space method

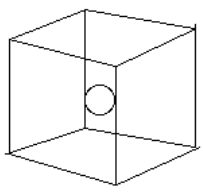
2) Object-space method

Information Coding / Computer Graphics, ISY, LiTH

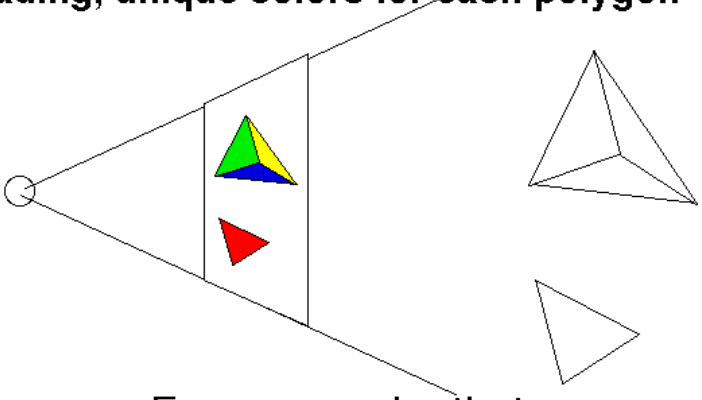
Image-space PVS generation

Render 6 images, all covering 1/6 of direction space

Render with flat shading, unique colors for each polygon



Render to all sides of a cube around the cell



Inspect the resulting images. For every color that appears, the corresponding polygon is added to the PVS

Conclusions about Visibility processing/ High level VSD:

- Frustum culling easy!
- Doesn't have to be perfect - some waste at edges are OK