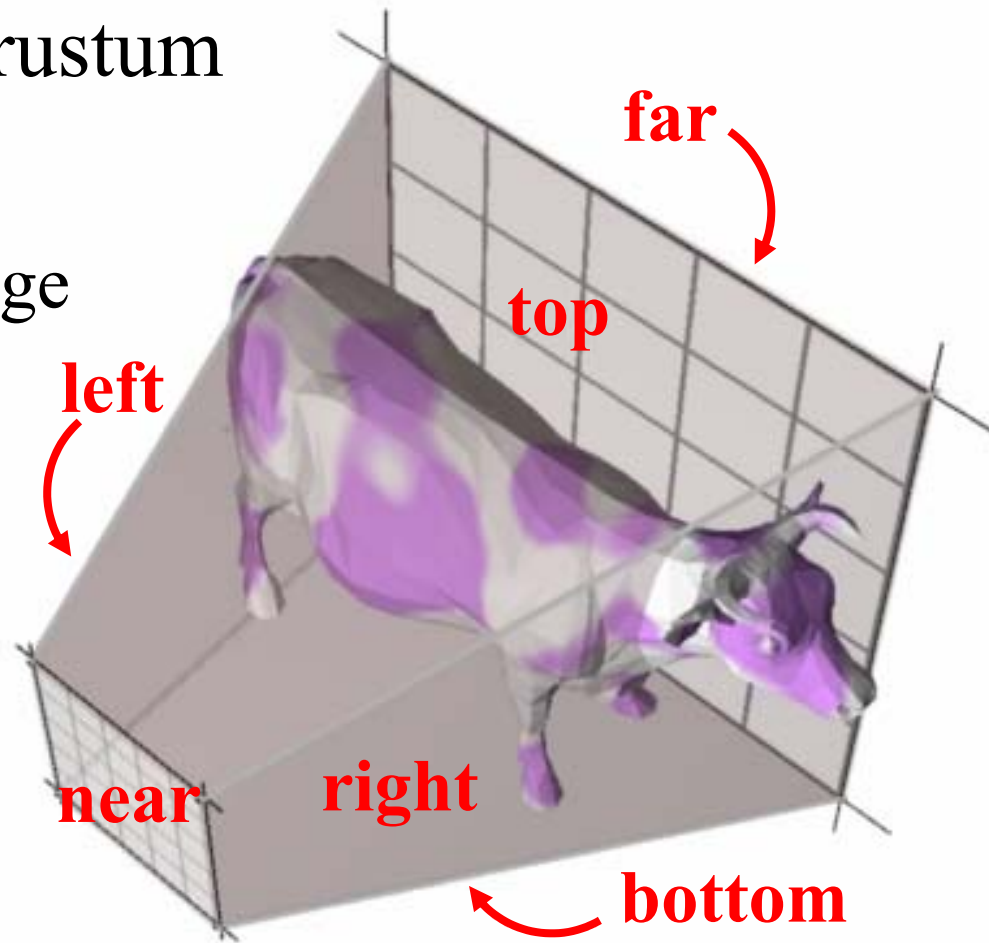


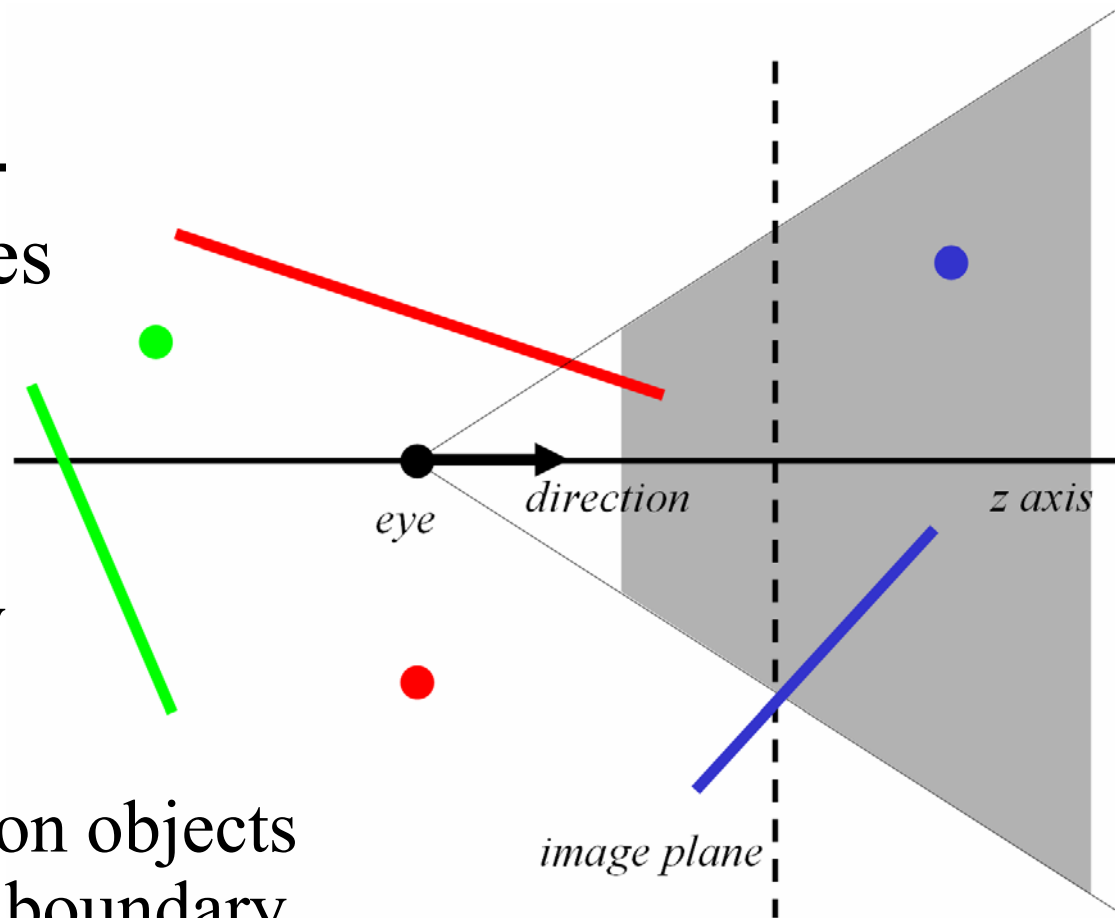
Clipping

- Eliminate portions of objects outside the viewing frustum
- View Frustum
 - boundaries of the image plane projected in 3D
 - a near & far clipping plane
- User may define additional clipping planes



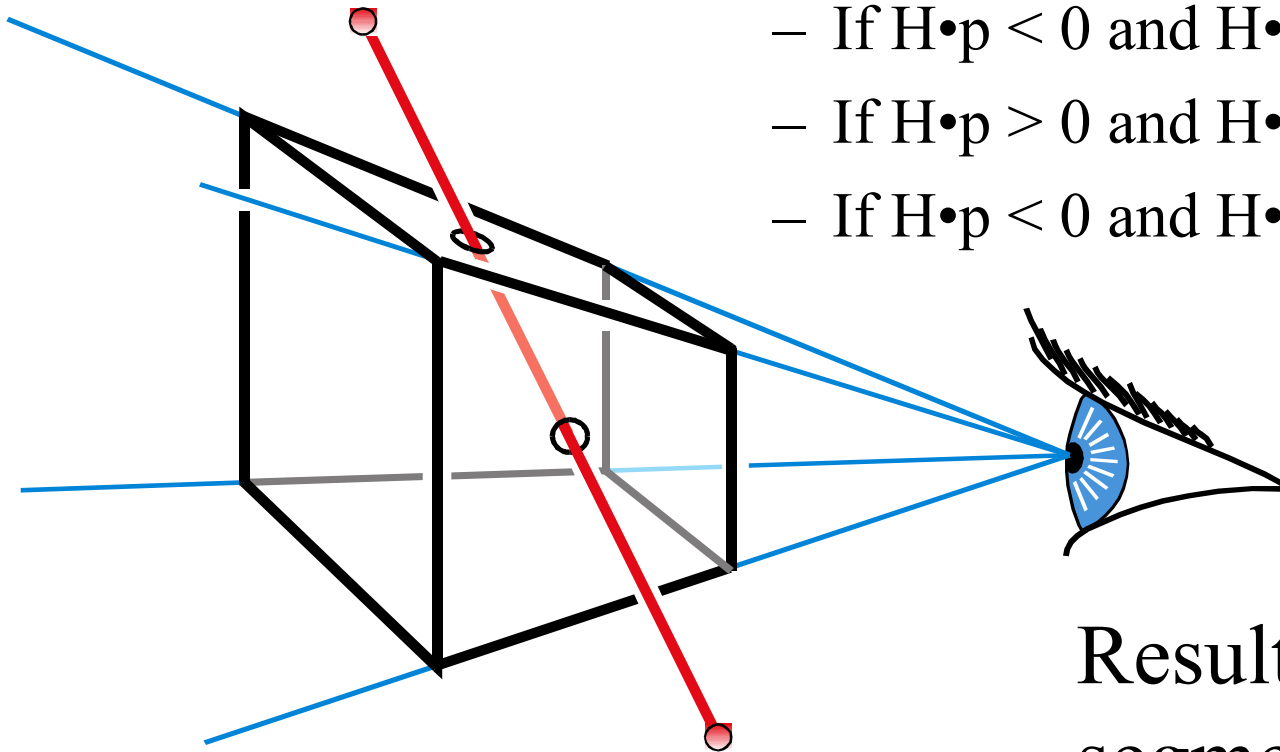
Why clip?

- Avoid degeneracies
 - Don't draw stuff behind the eye
 - Avoid division by 0 and overflow
- Efficiency
 - Don't waste time on objects outside the image boundary
- Other graphics applications (often non-convex)
 - Hidden-surface removal, Shadows, Picking, Binning, CSG (Boolean) operations (2D & 3D)



Clipping against the frustum

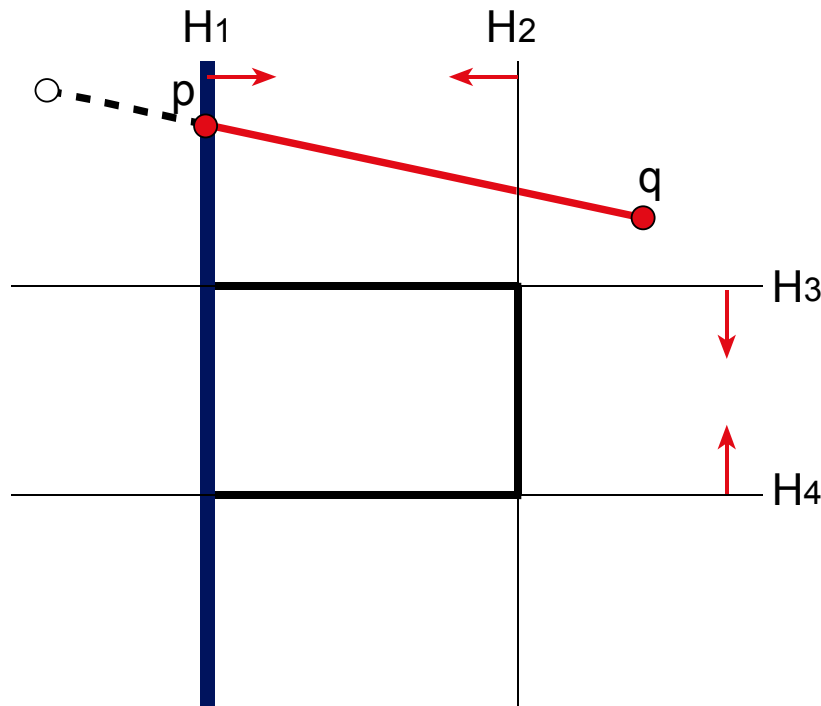
- For each frustum plane H
 - If $H \cdot p > 0$ and $H \cdot q < 0$, clip q to H
 - If $H \cdot p < 0$ and $H \cdot q > 0$, clip p to H
 - If $H \cdot p > 0$ and $H \cdot q > 0$, pass through
 - If $H \cdot p < 0$ and $H \cdot q < 0$, clipped out



Result is a single segment. Why?

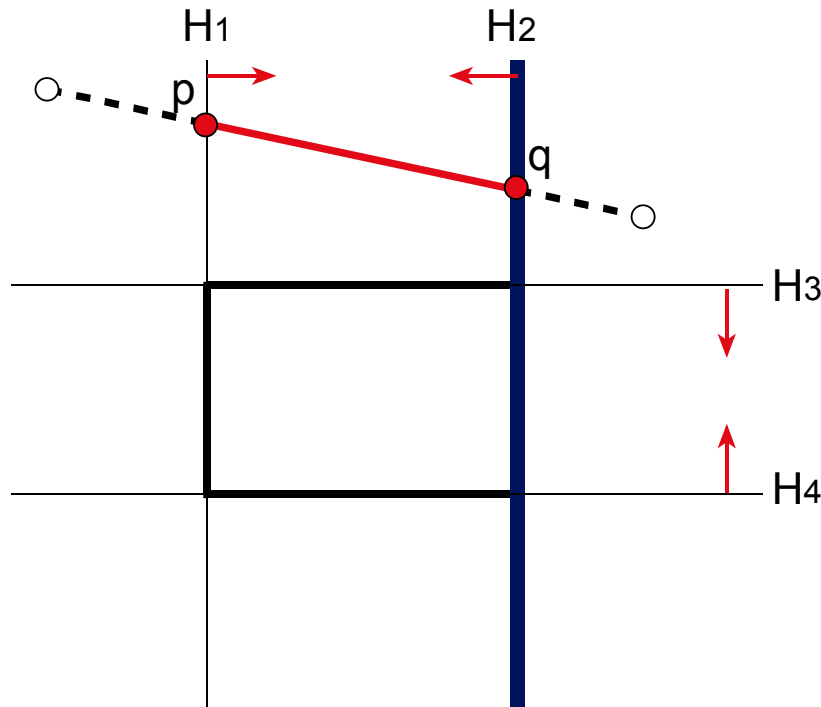
Is this Clipping Efficient?

- For each frustum plane H
 - If $H \cdot p > 0$ and $H \cdot q < 0$, clip q to H
 - If $H \cdot p < 0$ and $H \cdot q > 0$, clip p to H
 - If $H \cdot p > 0$ and $H \cdot q > 0$, pass through
 - If $H \cdot p < 0$ and $H \cdot q < 0$, clipped out



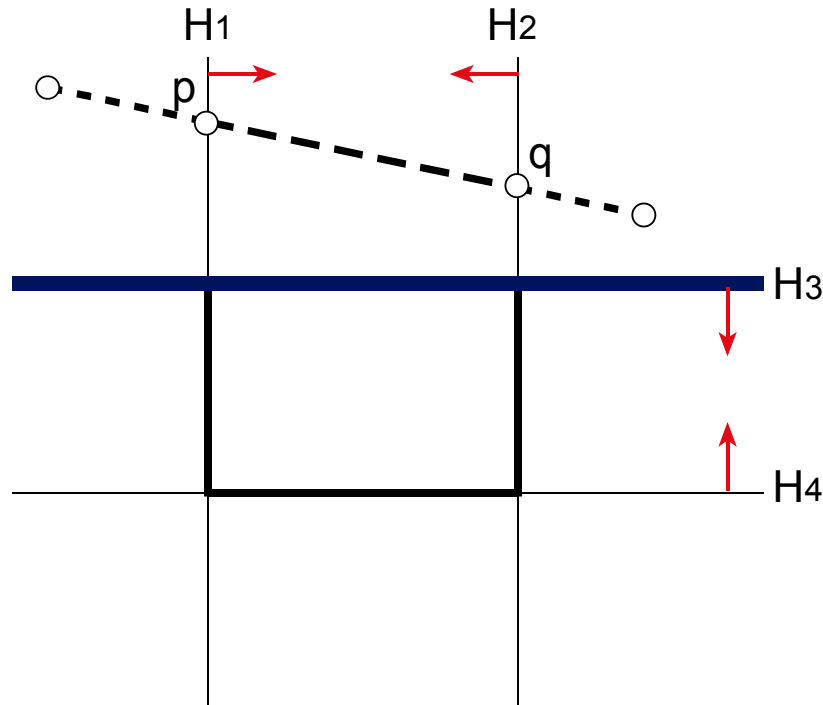
Is this Clipping Efficient?

- For each frustum plane H
 - If $H \cdot p > 0$ and $H \cdot q < 0$, clip q to H
 - If $H \cdot p < 0$ and $H \cdot q > 0$, clip p to H
 - If $H \cdot p > 0$ and $H \cdot q > 0$, pass through
 - If $H \cdot p < 0$ and $H \cdot q < 0$, clipped out



Is this Clipping Efficient?

- For each frustum plane H
 - If $H \cdot p > 0$ and $H \cdot q < 0$, clip q to H
 - If $H \cdot p < 0$ and $H \cdot q > 0$, clip p to H
 - If $H \cdot p > 0$ and $H \cdot q > 0$, pass through
 - If $H \cdot p < 0$ and $H \cdot q < 0$, clipped out



What is the problem?

The computation of the intersections, and any corresponding interpolated values is unnecessary

Can we detect this earlier?