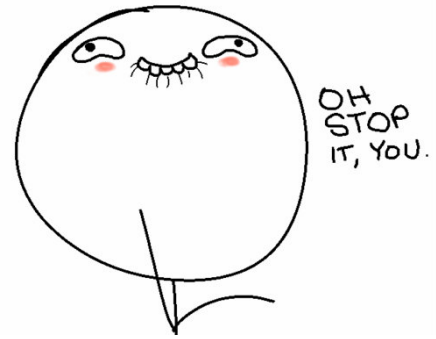
Rendering Countless Blades of Waving Grass

GPU Gems


OH
STOP
IT, YOU.

Why?

Super Mario 64

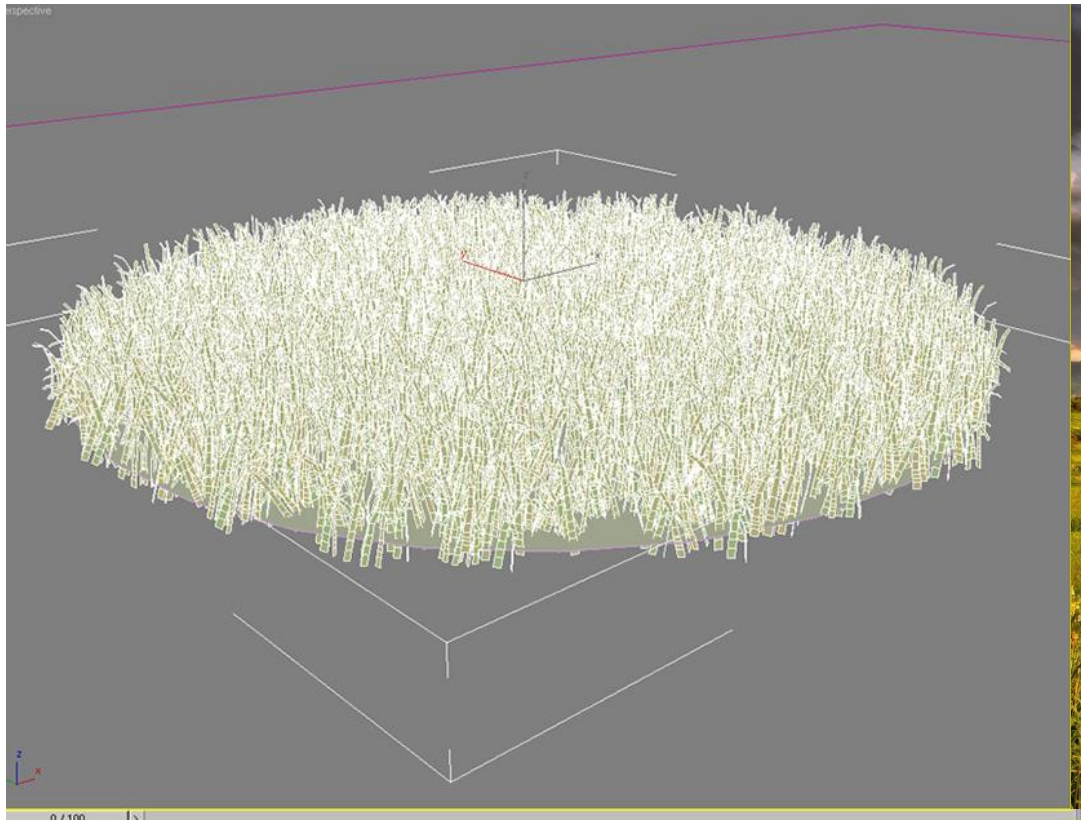
Why?

UE4 “A Boy and His Kite”

Why?

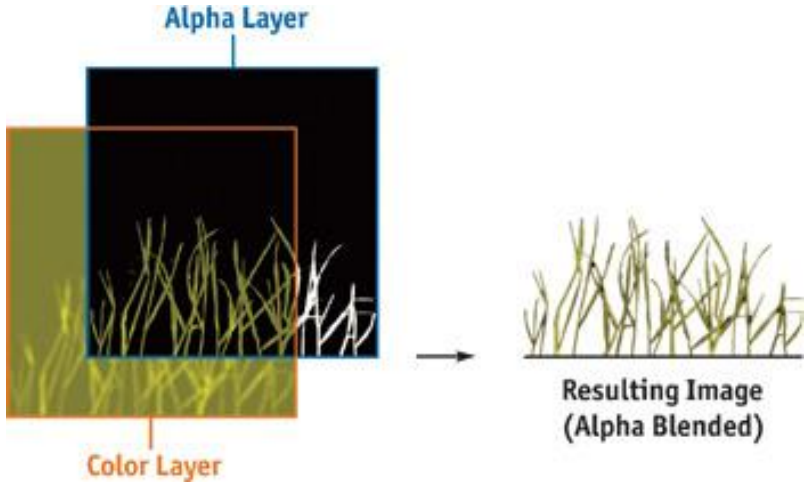
Sigh..

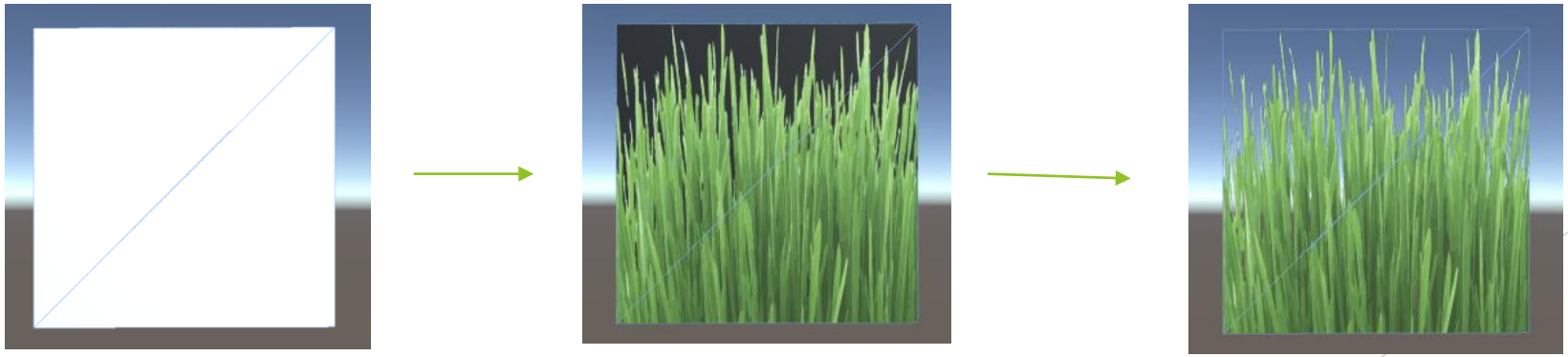
Hardware is currently not fast enough



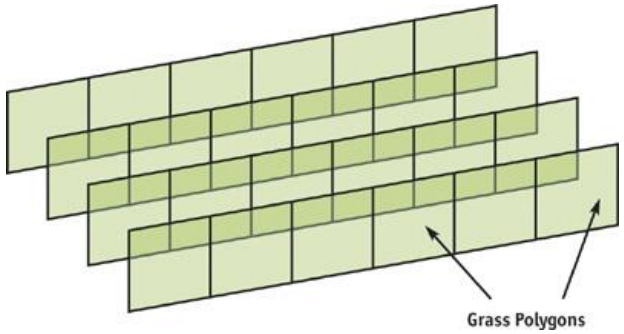
Cheaper Approach

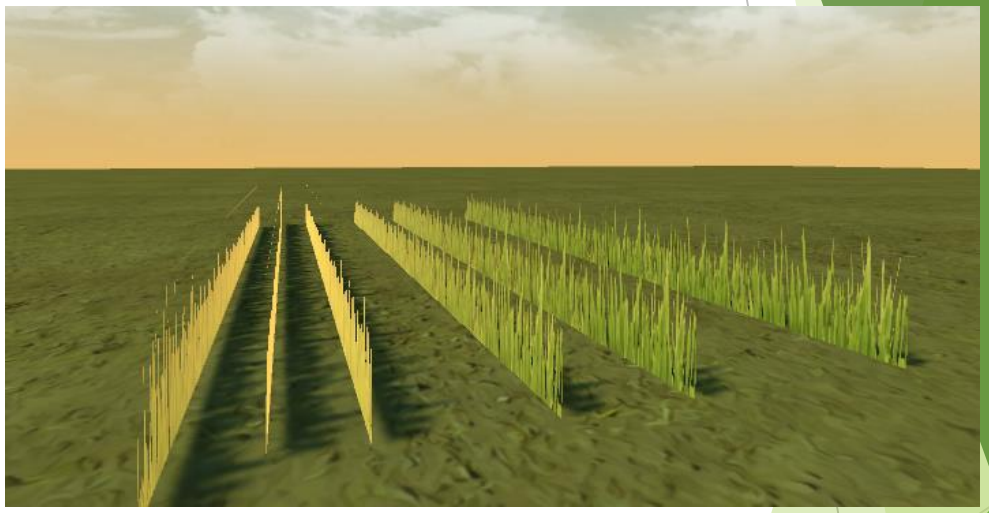
- Grass Texture w/transparency





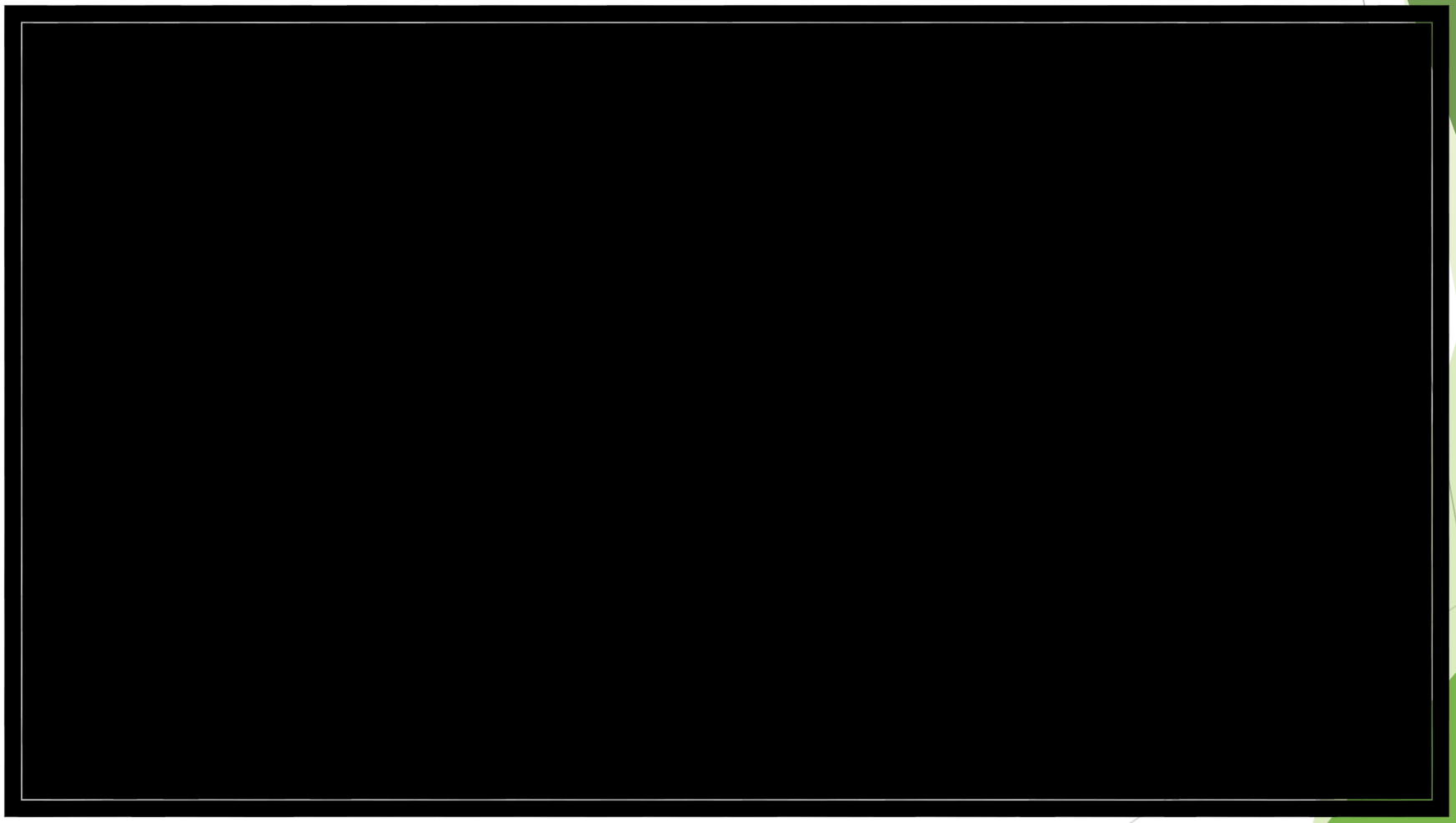
Linear Arrangement



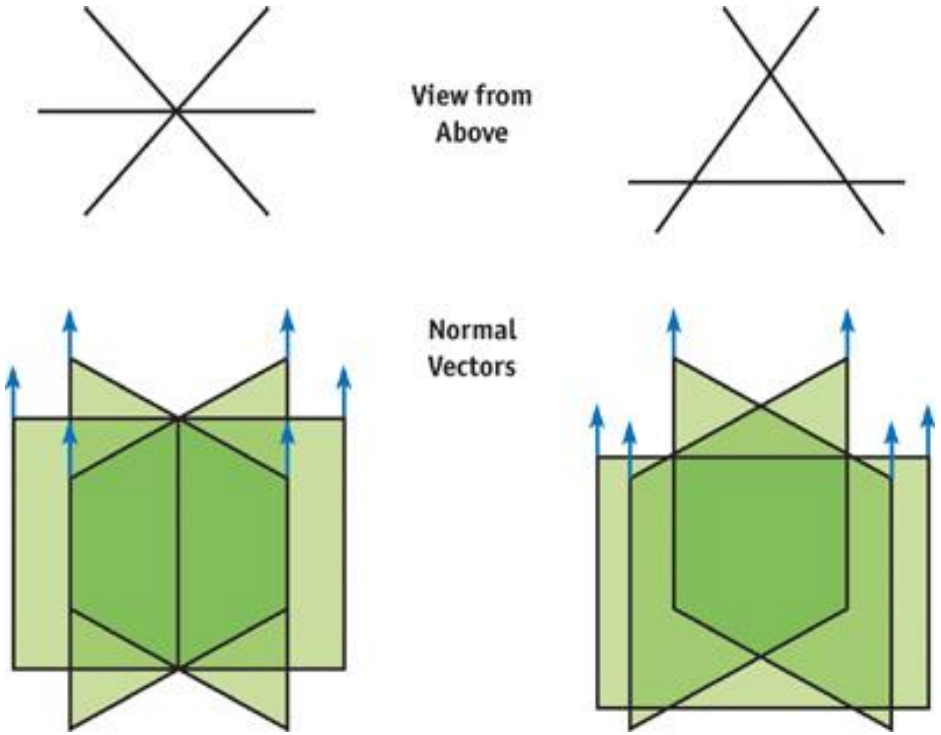


The cake is a lie!

Axial Billboarding



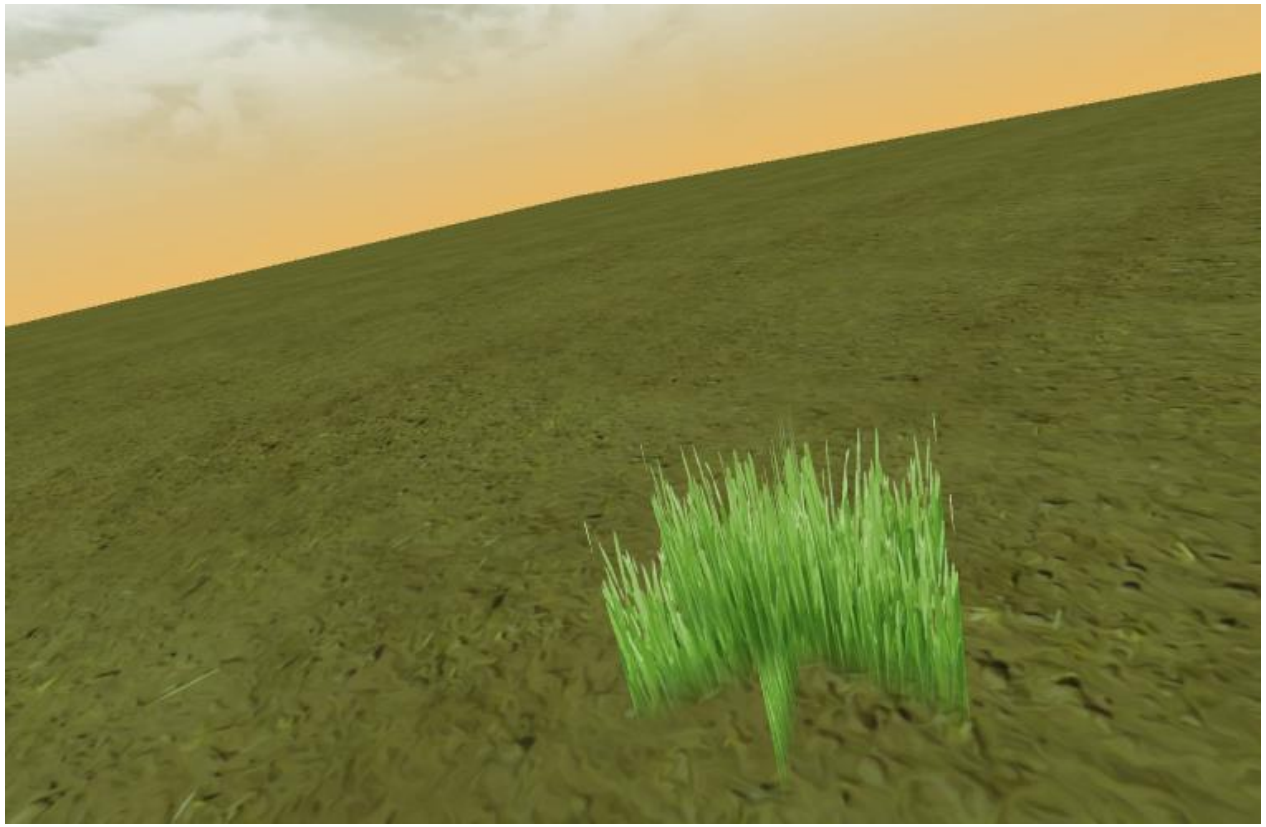
Solution – Grass Object

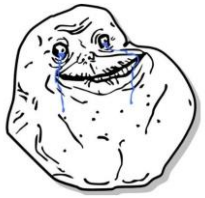


- Disable back-face culling
- Normal vectors should be parallel to the polygon's vertical edges*

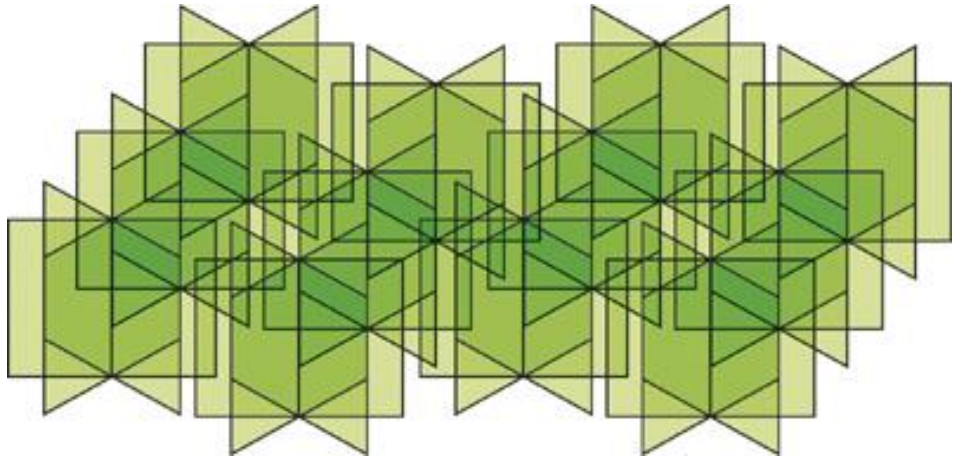
*This guarantees correct lighting for all grass objects situated on slopes, with no differences due to the brightness of the terrain.

Solution – Grass Object

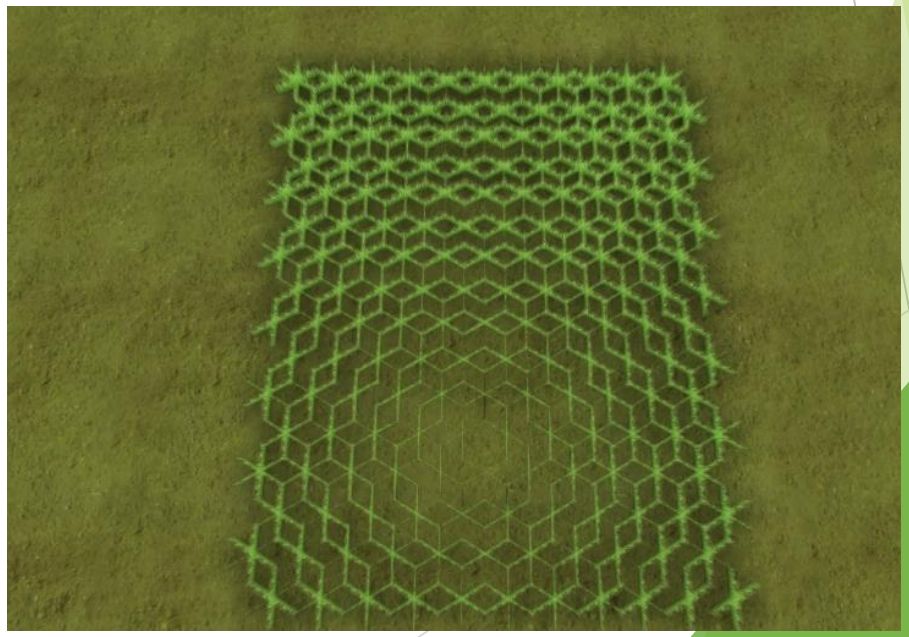




Cluster of Grass Objects

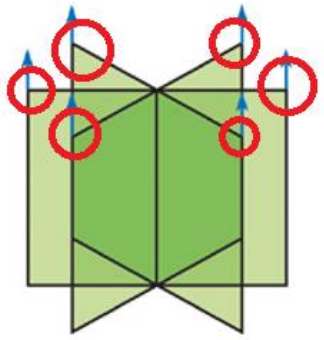


- Enable z-testing
- Sort back-to-front
- Use alpha blending



Animation

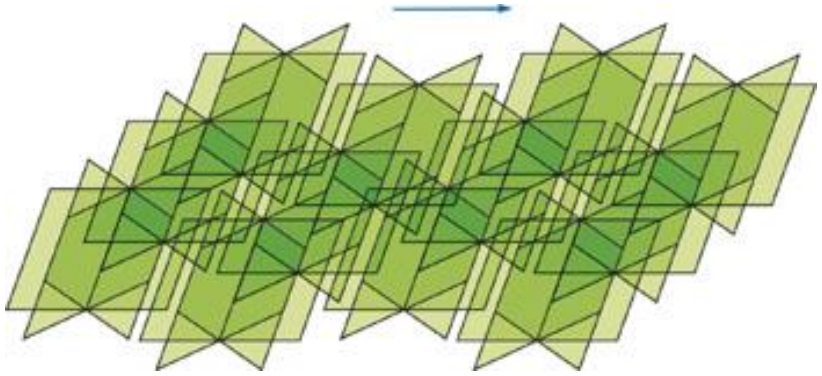
- Trigonometric function e.g. sine and cosine
- Must only animate higher vertices



if (v.TexCoords.y <= 0.1)

- Three Types of Animation
  - Animation per Cluster of Grass Objects
  - Animation per Vertex
  - Animation per Grass Object

Animation per Cluster of Grass Objects - CPU



1. Calculate animation on CPU
2. Set translation vector as vertex constant
3. Execute draw call for cluster
4. Change translation vector constant
5. Render next cluster
6. Repeat 4 – 5

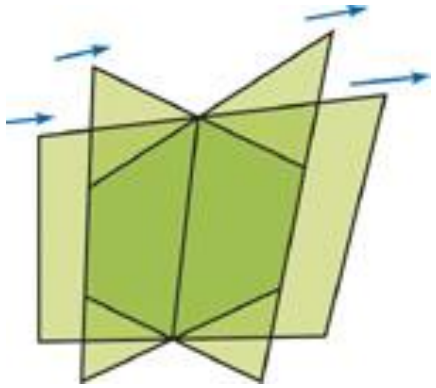
Pros

- Complex animations can be achieved through CPU-based algorithms

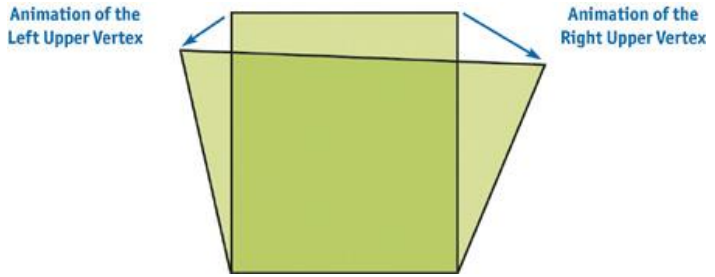
Cons

- Poor performance

Animation per Vertex - GPU



1. Set constants, such as time stamp and wind velocity
2. Execute draw call for complete meadow
3. Calculate animation on vertex shader



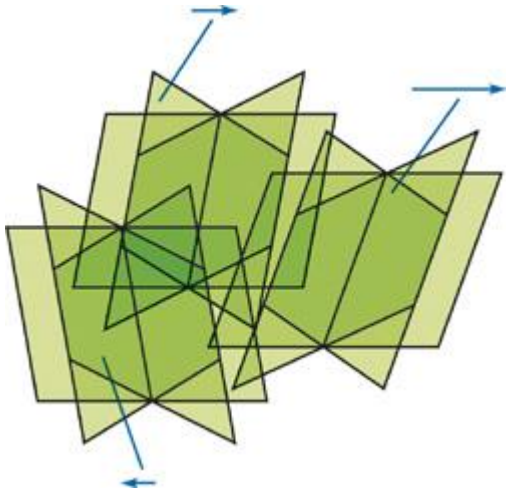
Pros

- Few draw calls to display full meadow
- Clusters become undistinguishable

Cons

- Animation complexity is limited and it may appear homogeneous
- Distortion may appear on textures

Animation per Grass Object



1. Set constants, such as time stamp and wind velocity
2. Execute draw call for complete meadow
   a) For each vertex, send its grass object's center point
3. Calculate animation on vertex shader based on the center position of grass object

Pros

- Few draw calls to display full meadow
- No texture distortions
- More natural look

Cons

- Complexity of animations is still limited
- Additional data is required for each vertex


VERY MUCH
THANK YOU