
BUNGiE®

Halo Reach Effects Tech

Chris Tchou

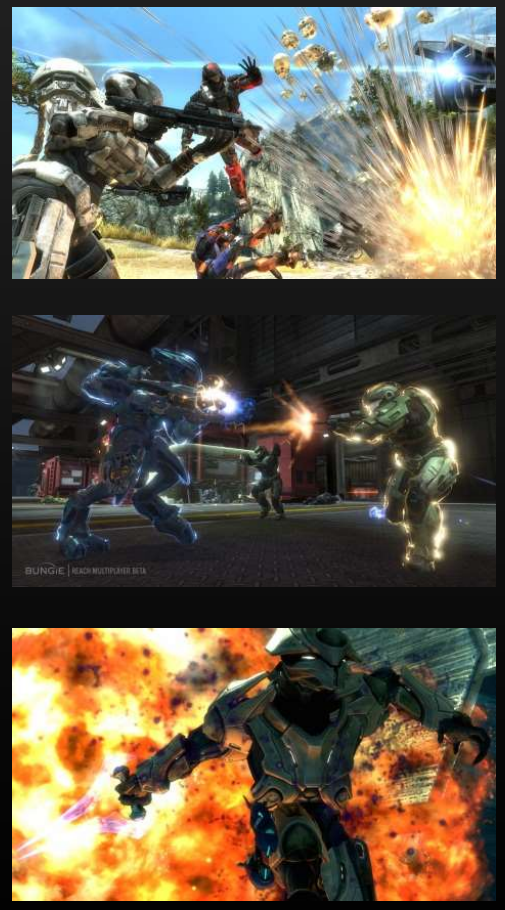
BUNGiE

Effects Directions

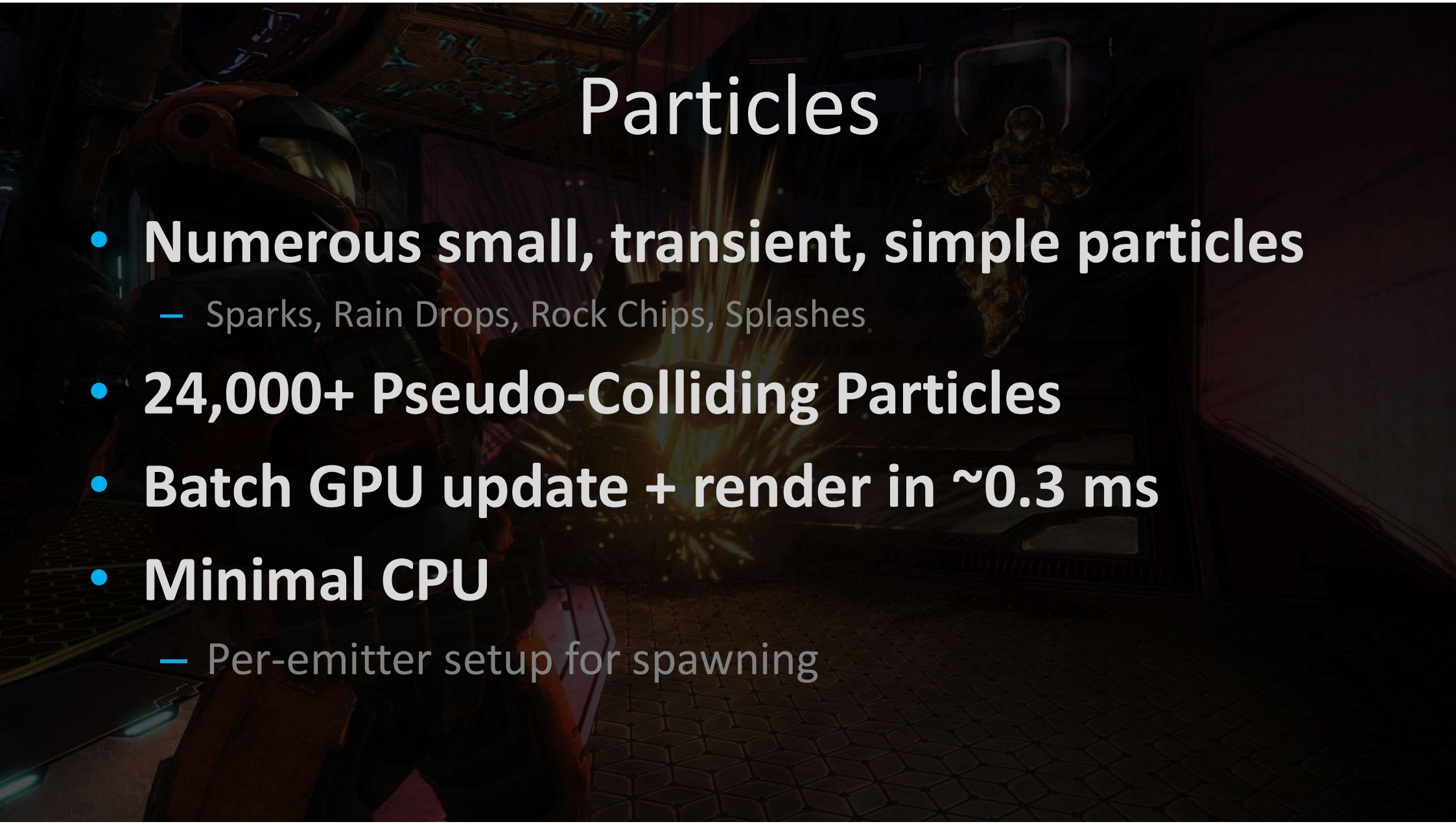
- Atmospheric
- Dense/Visceral
- Pretty
- Easy-To-Use
- Fast

Covered Techniques

- **Cheap Colliding Particles**
- **‘Shields’ & Depth Effects**
- **Low Resolution Transparents**




Particles!



Particles

- **Numerous small, transient, simple particles**
  - Sparks, Rain Drops, Rock Chips, Splashes
- **24,000+ Pseudo-Colliding Particles**
- **Batch GPU update + render in ~0.3 ms**
- **Minimal CPU**
  - Per-emitter setup for spawning

Dynamic Particle State

- **GPU data (28 bytes/particle)**

position.x	position.y	position.z	age

  - 32f
  - sign(age) signifies ‘at rest’ vs. ‘moving’ particles

velocity.x	velocity.y	velocity.z	Δ age

  - 16f

Dynamic Particle State

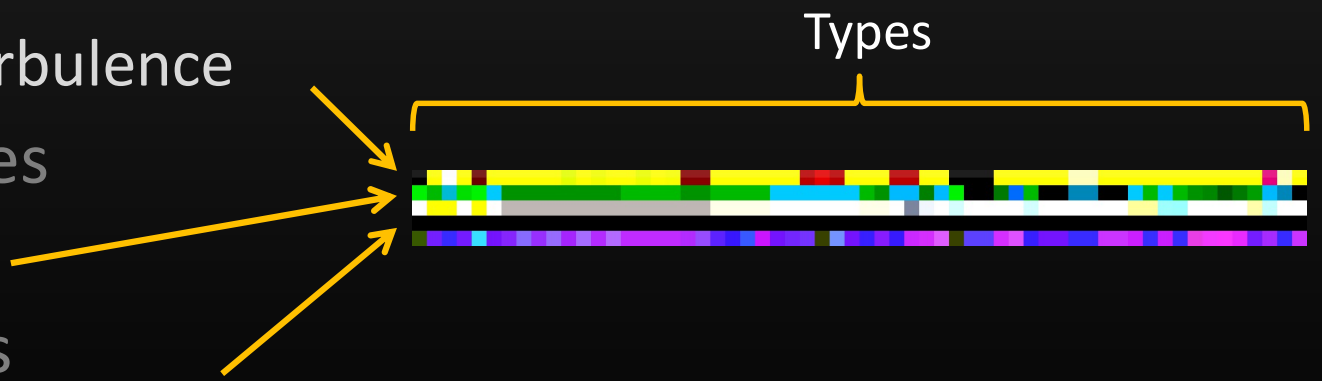
- **GPU data (28 bytes/particle)**

up.x	up.y	brightness	type

  - 8-bit integer
  - 2D rotation + scale (as up vector)
  - brightness encodes simple illumination
  - type points to static particle properties

Particle Types

- **Static type data stored in a library texture**
  - Physics properties
    - drag, gravity, turbulence
  - Collision properties
    - elasticity
  - Render properties
    - VS: size, orientation (velocity or screen facing), motion blur
    - PS: color tint, texture index

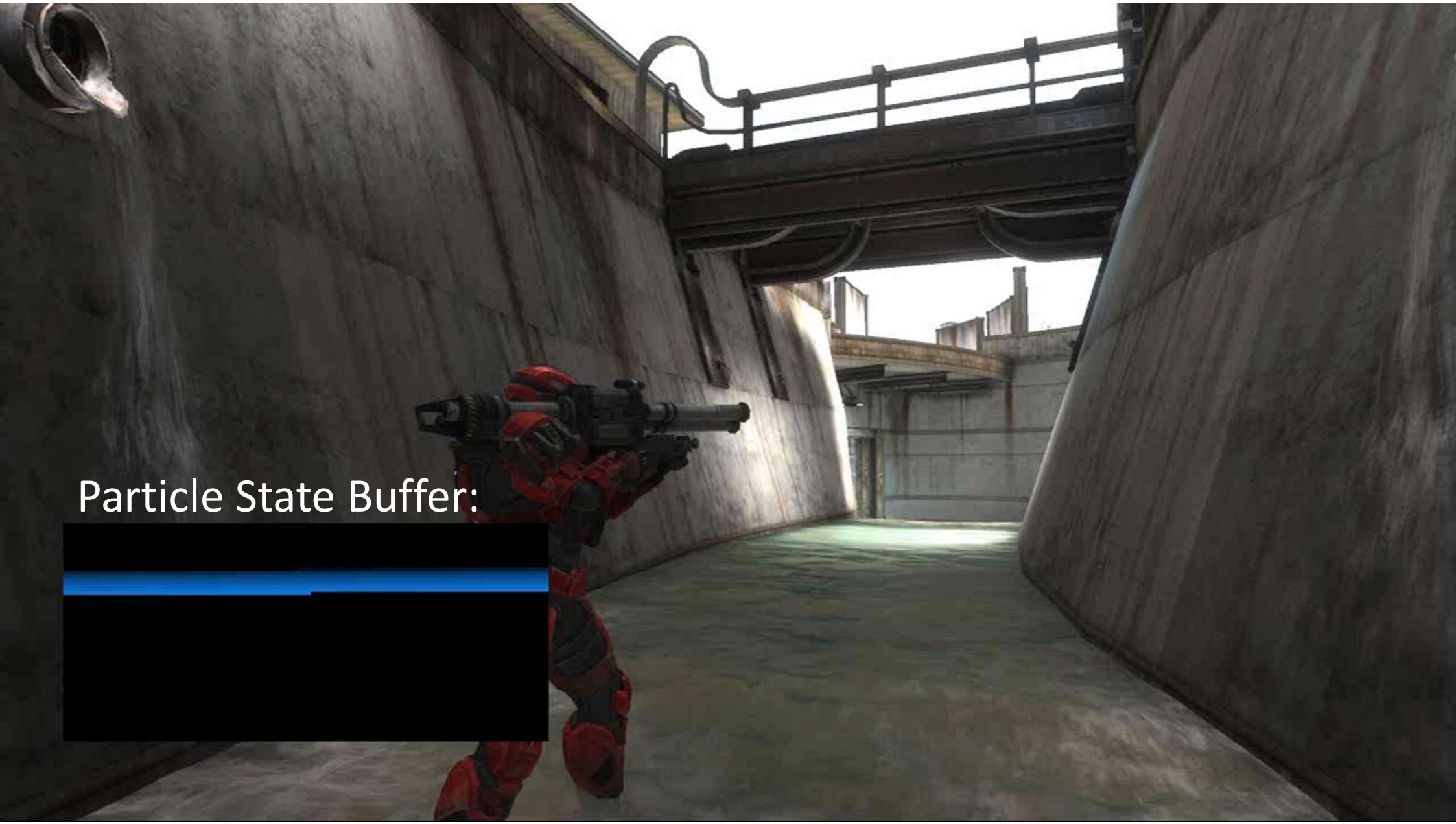


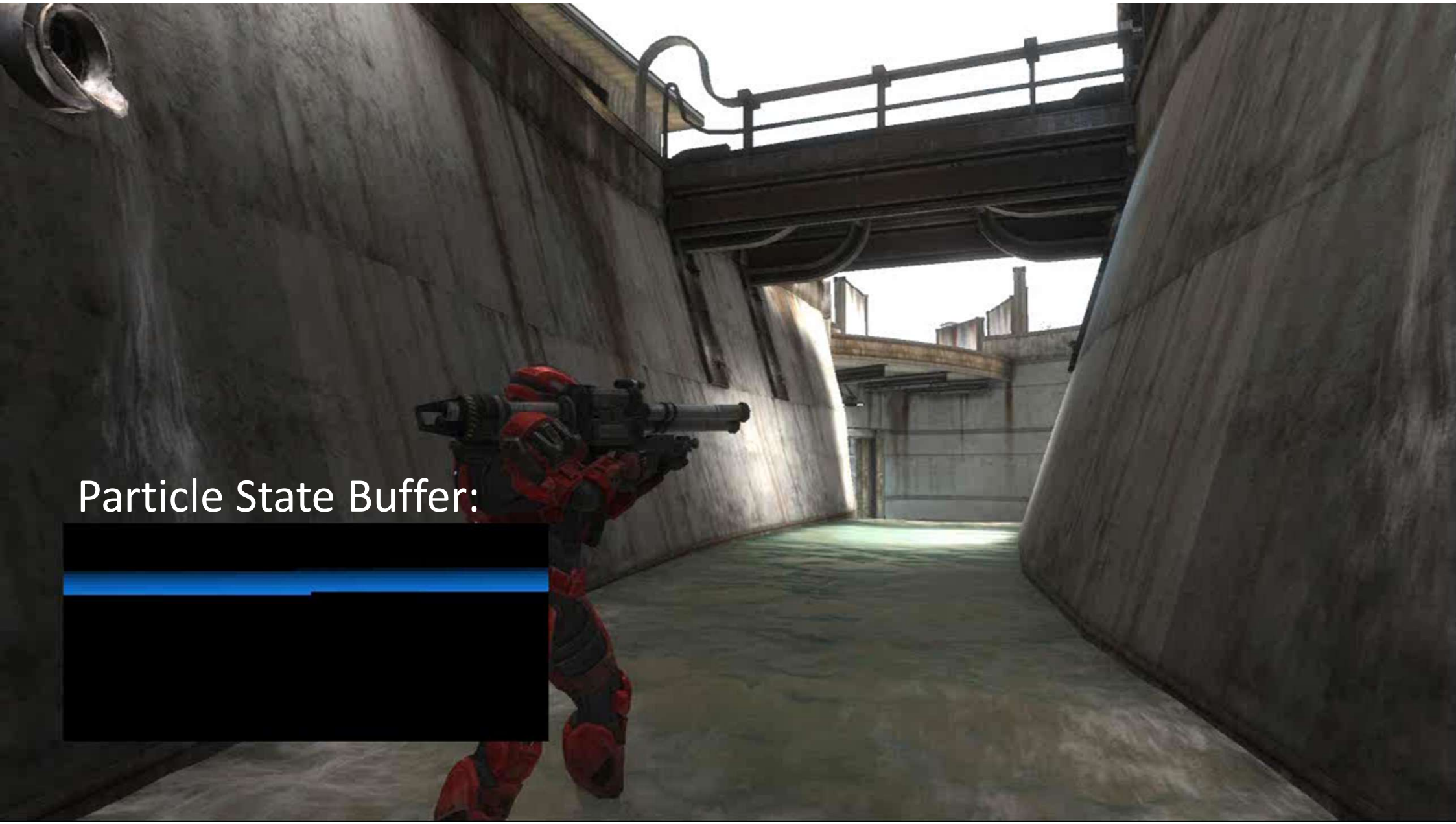
Particle Render

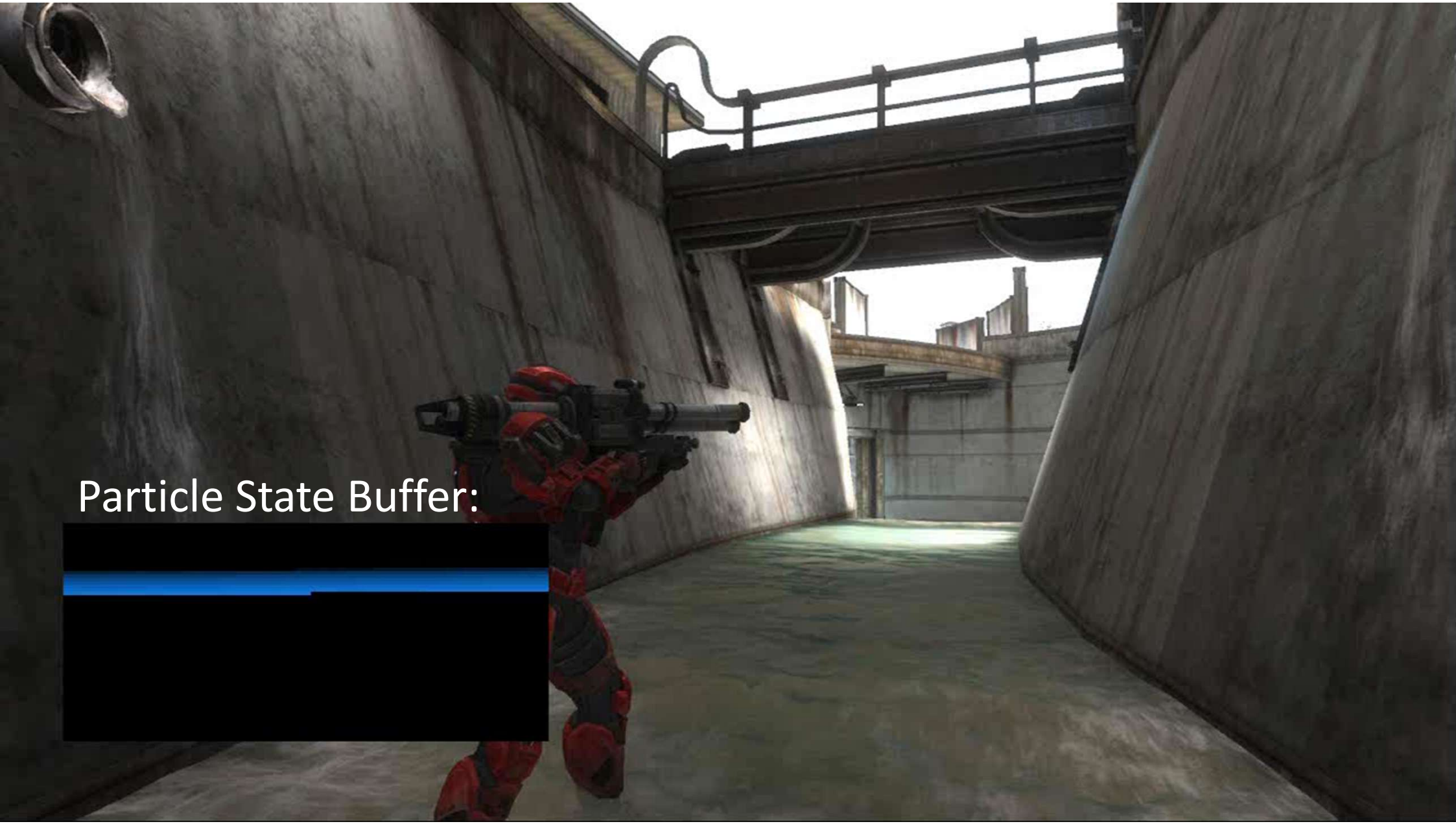
- **Single GPU draw call to render all particles**
- **Read particle state + type texture**
- **Create particle card geometry in VS**
- **Apply render properties in PS**

Particle Spawning

- **For each emitter:**
  - CPU determines spawn count
  - CPU issues GPU draw to the dynamic state buffer
- **GPU generates new particle state**
  - Sample randomly from a texture, and apply artist transform
  - Default textures have random values
  - Custom textures can generate complex spawn shapes

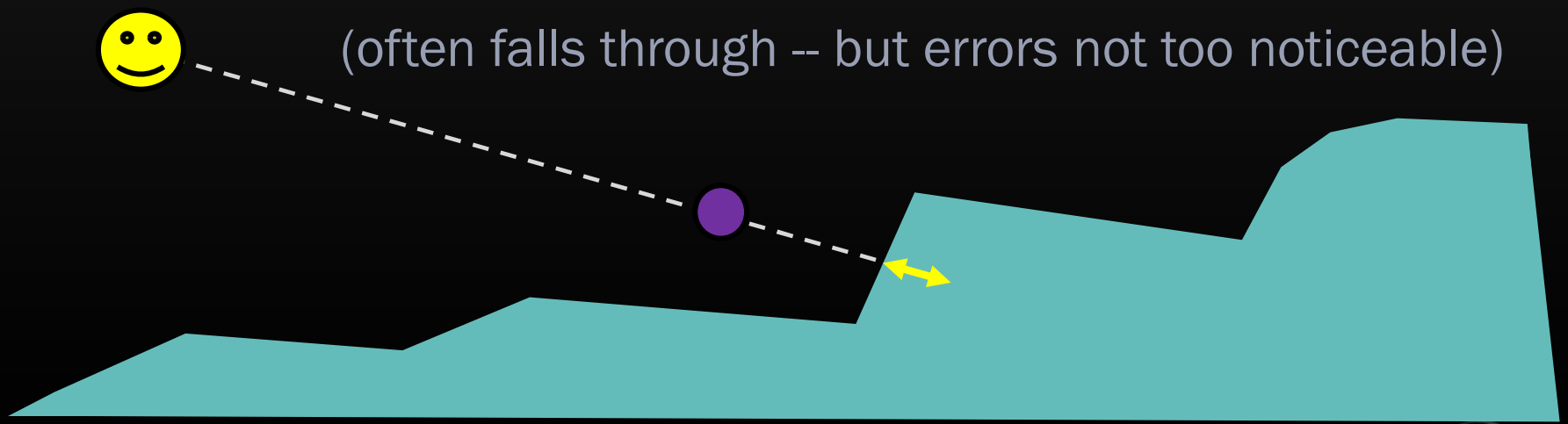

Particle State Buffer:


Particle State Buffer:


Particle State Buffer:

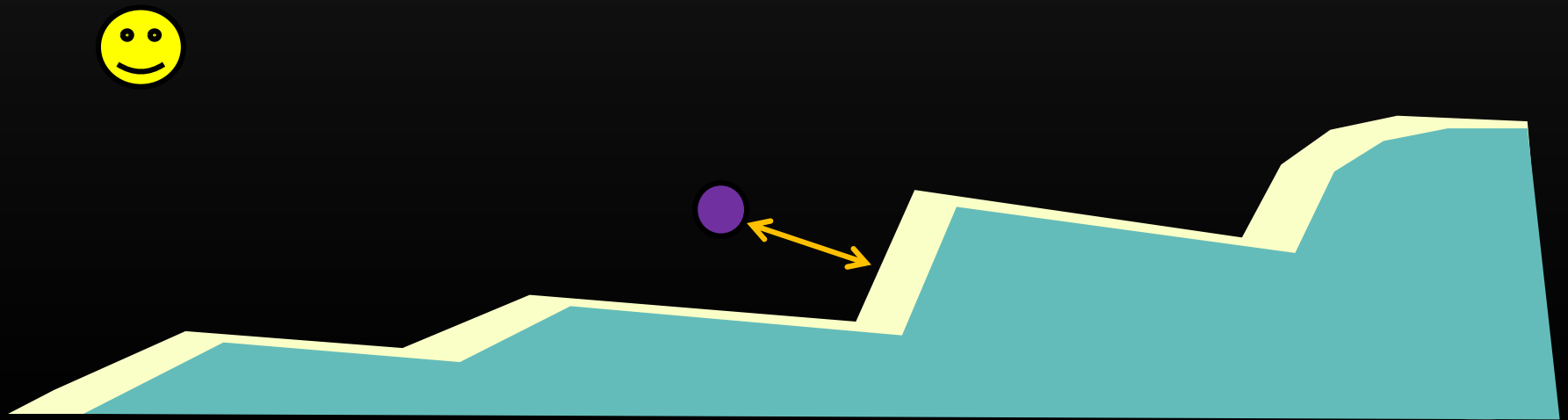
Particle Collision

- **Sample depth buffer for local collision**
  - Particles collide in a range behind the depth buffer
  - Only bounces on-screen, against visible surfaces

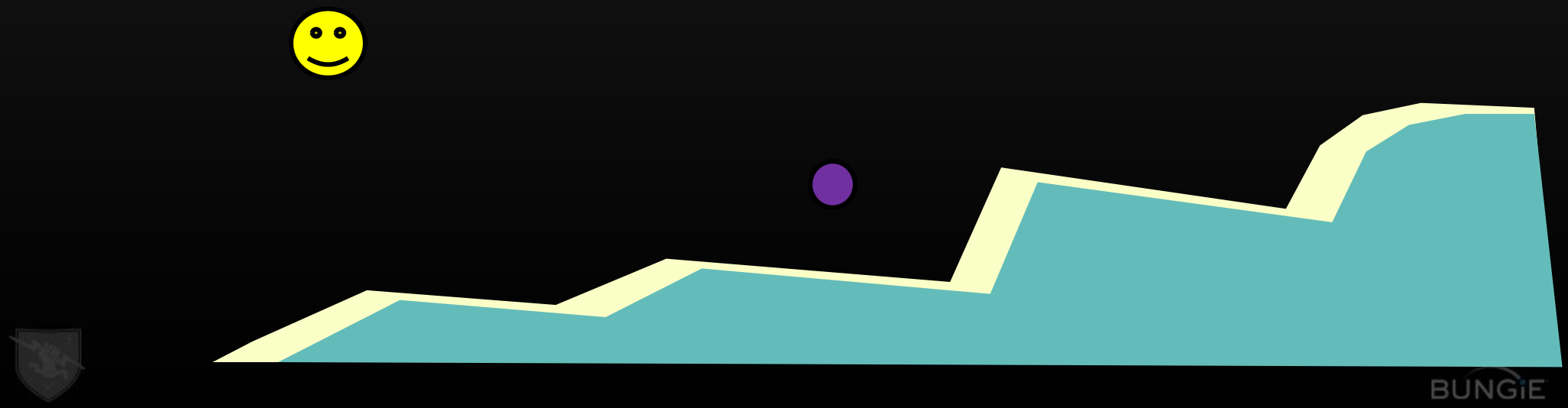


Particle Collision

- **‘At rest’ particles don’t move**
  - If not close to depth sample, set to ‘moving’
  - Allows them to fall if the surface moves

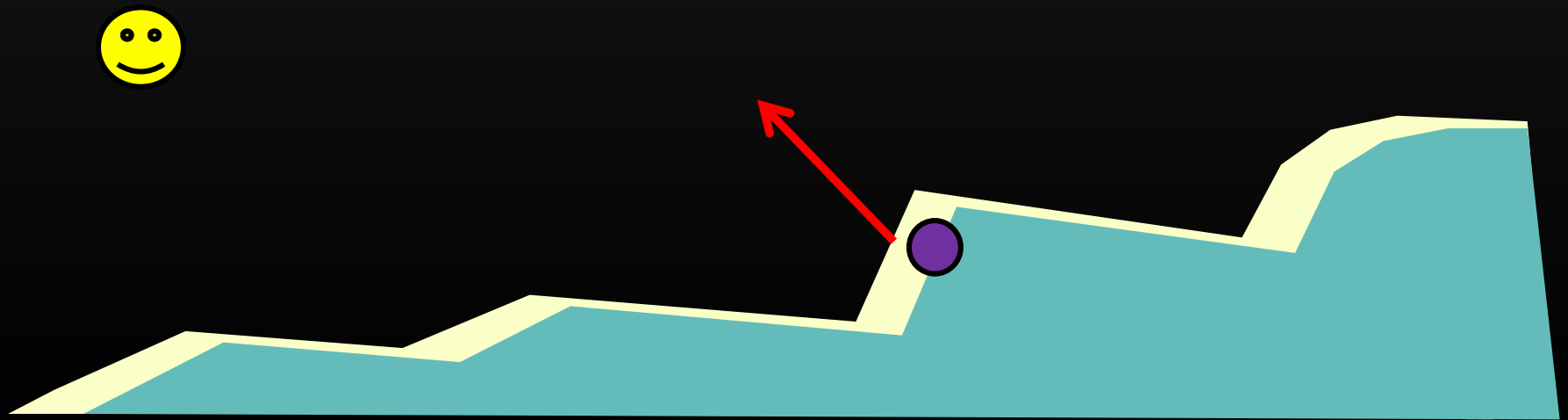


Particle Collision



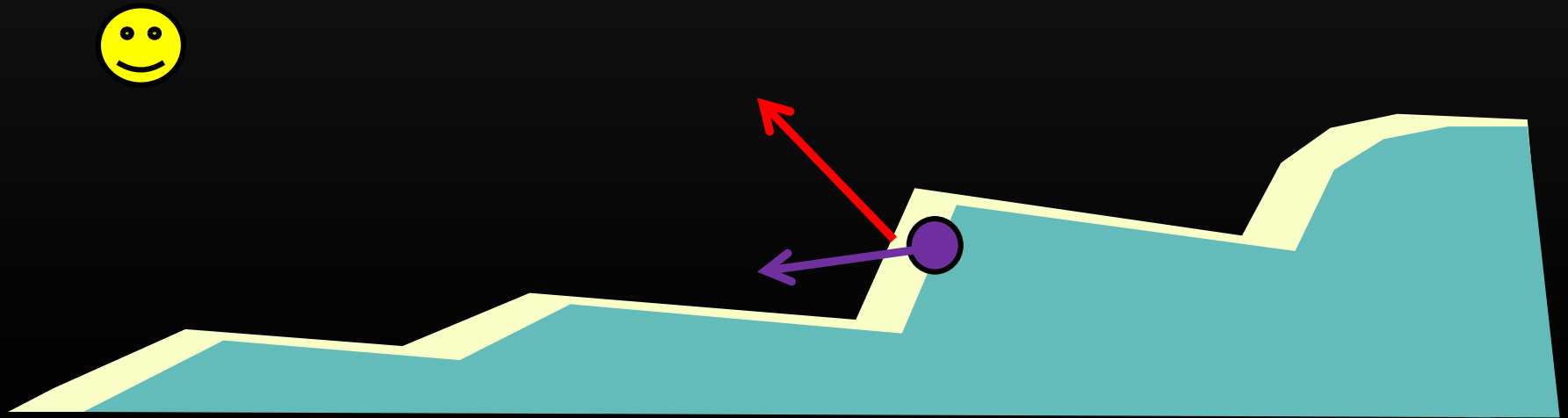
Particle Collision

- **Sample normal buffer**



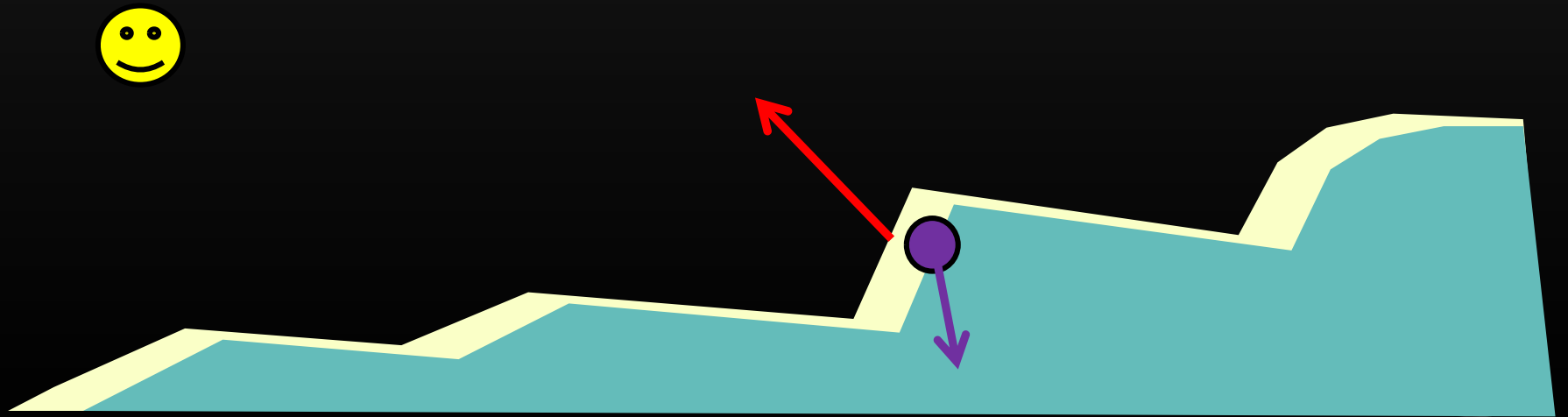
Particle Collision

- **Compare normal + particle velocity**
- **Outgoing particles are destroyed**
  - Stops particles from bleeding through visible surfaces



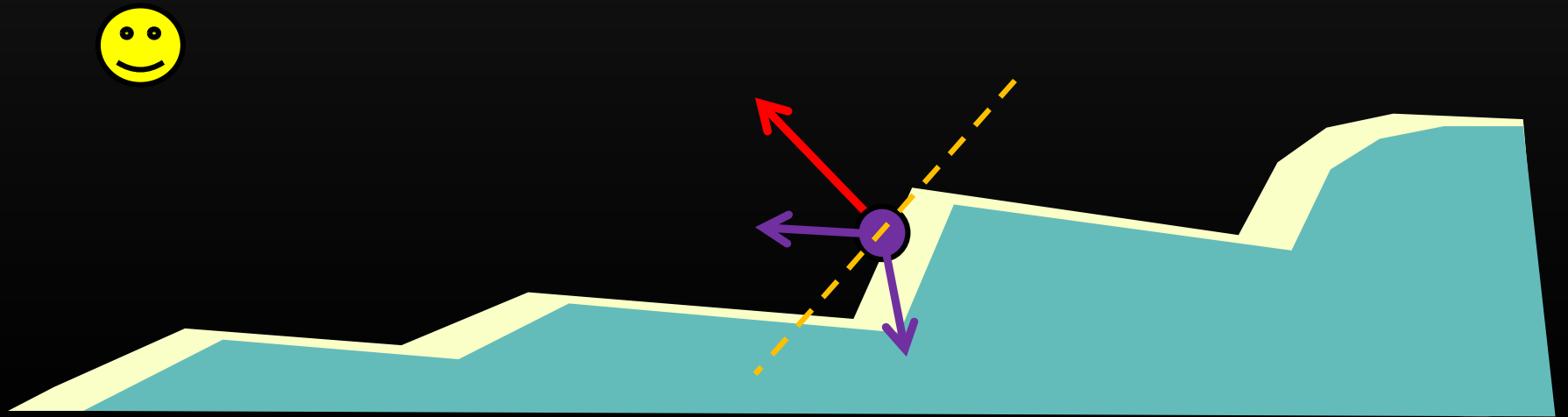
Particle Collision

- **Incoming particles are bounced:**
  - Ensure particle is not embedded:
  - Re-project position onto surface



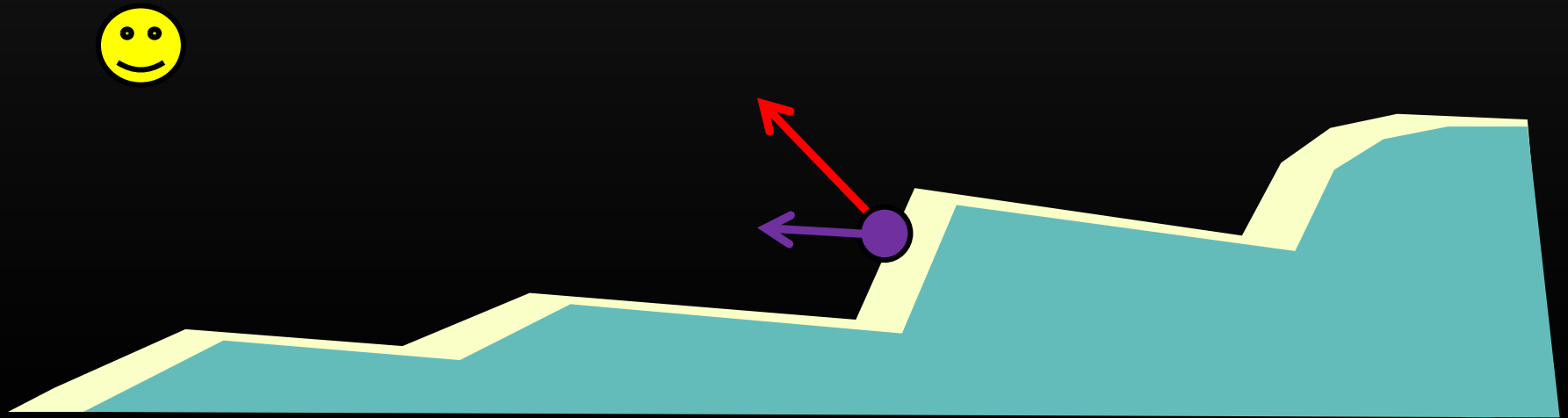
Particle Collision

- **Incoming particles are bounced:**
  - Reflect velocity around normal
  - Reduce velocity by elasticity



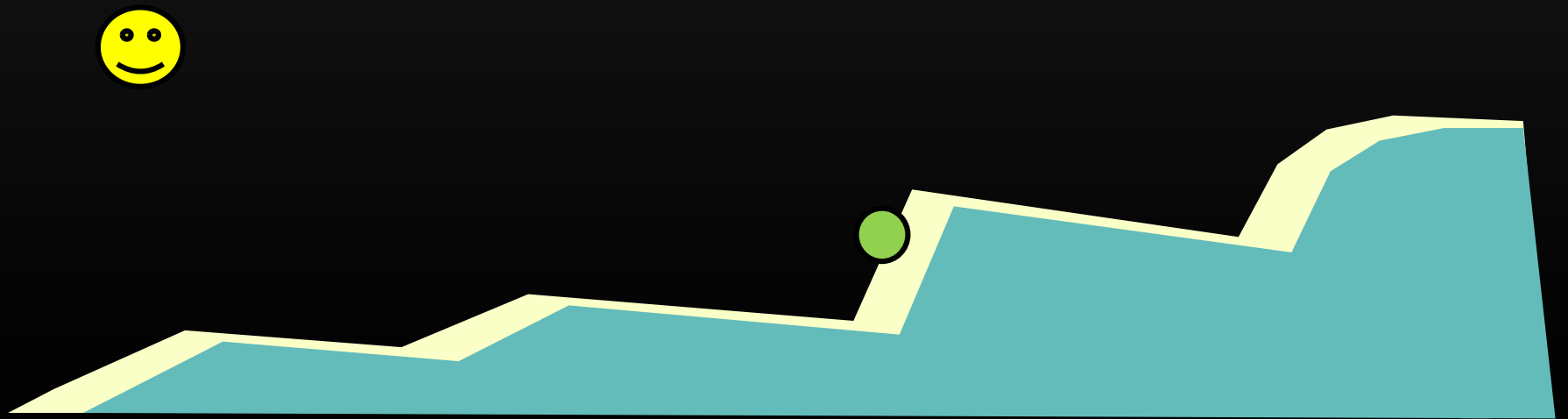
Particle Collision

- **Incoming particles are bounced:**
  - Switch particle type (optional)
  - If velocity is below a threshold, set ‘at rest’



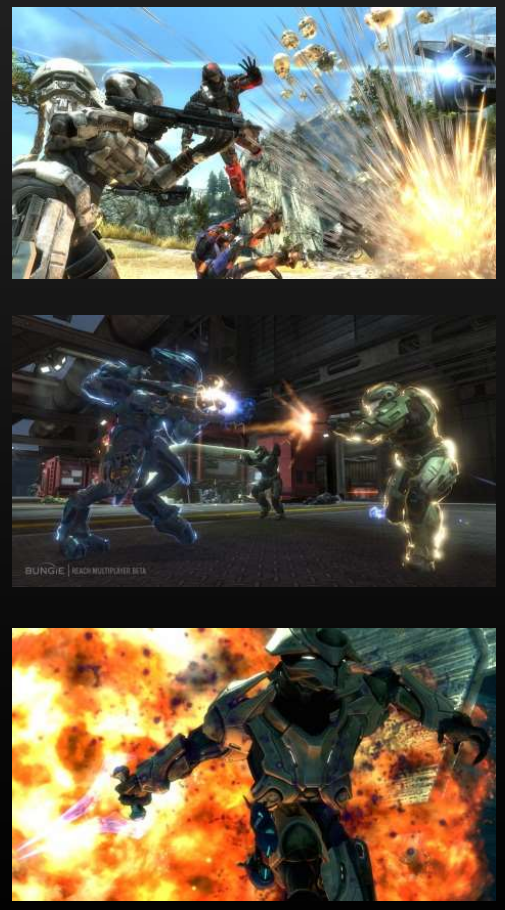
Particle Collision

- **Incoming particles are bounced:**
  - Switch particle type (optional)
  - If velocity is below a threshold, set ‘at rest’

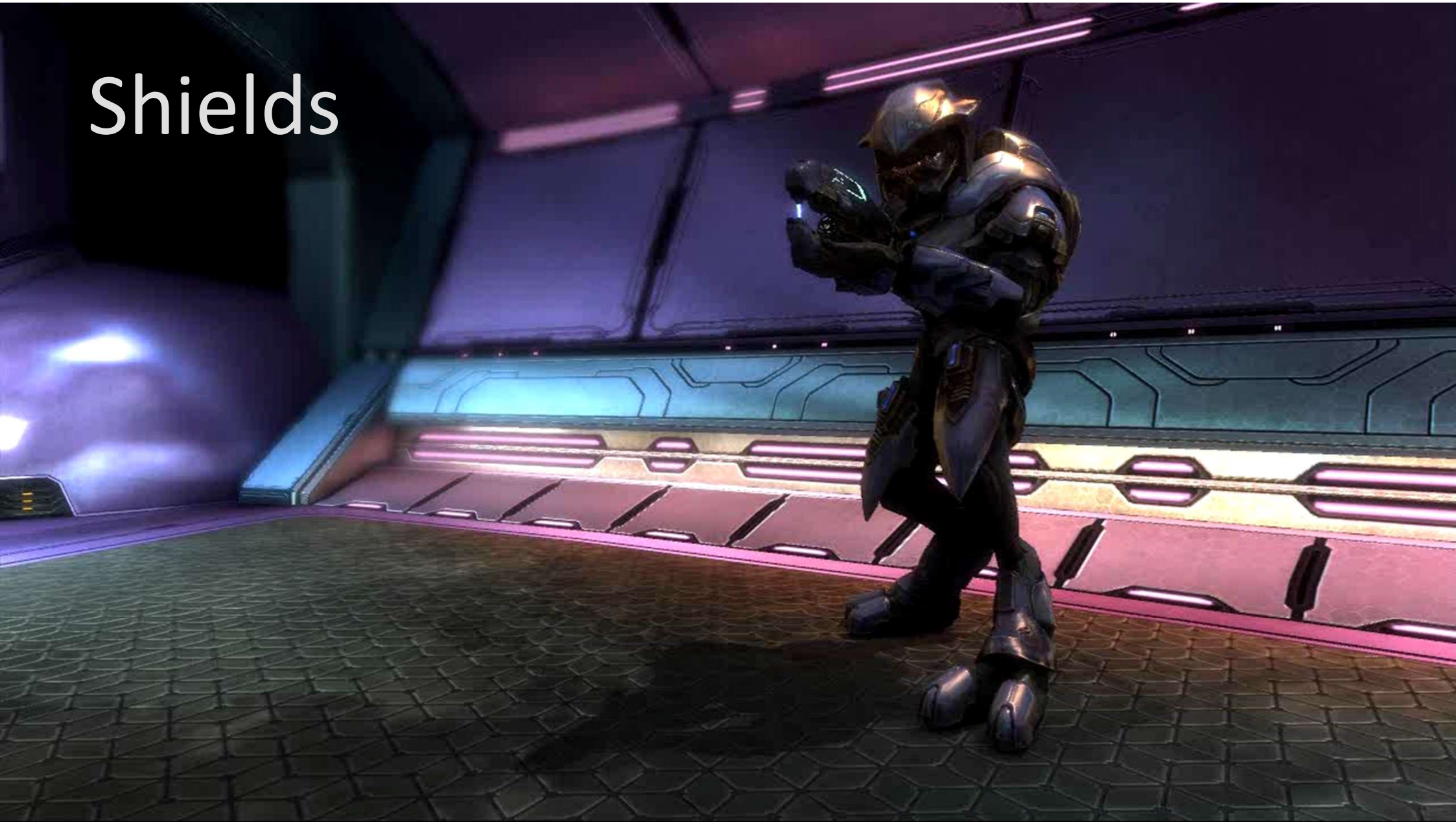


Covered Techniques

- **Cheap Colliding Particles**
- **‘Shields’ & Depth Effects**
- **Low Resolution Transparents**



Shields




Halo 3

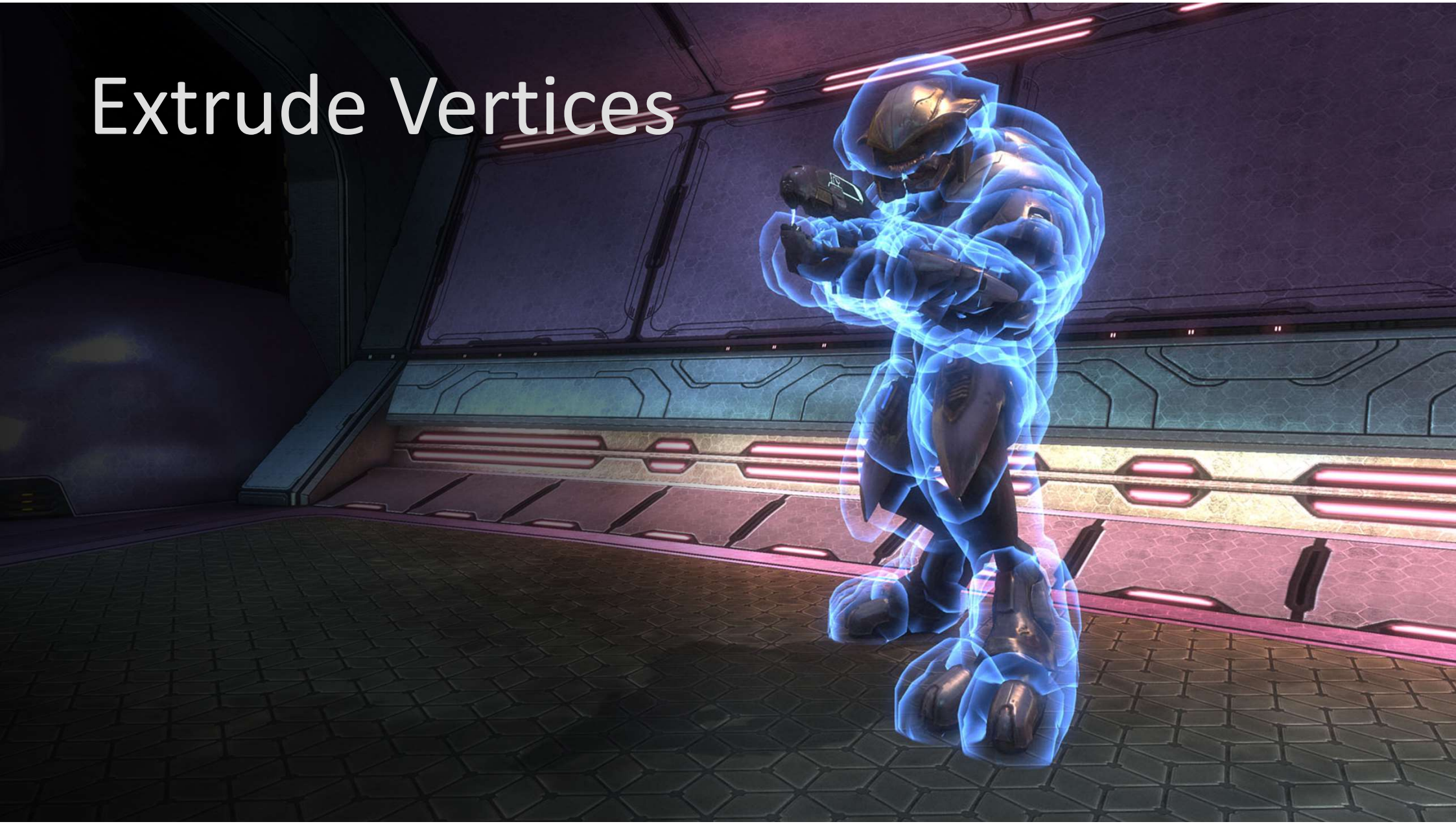

Consistent Look
BUNGiE | REACH MULTIPLAYER BETA


Don’t Hide Characters
BUNGiE

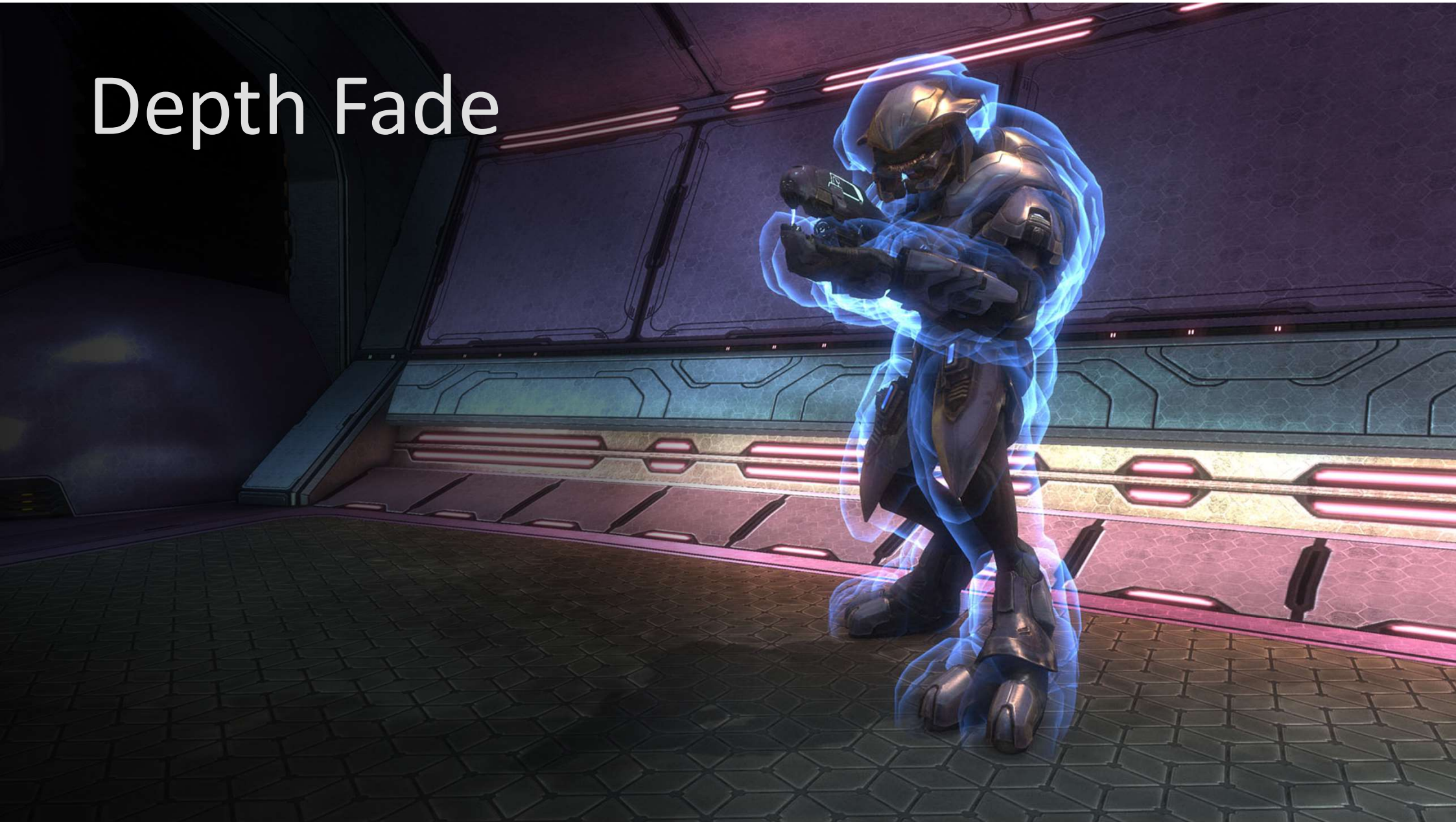

High Readability
BUNGIE

Minimal Art Overhead

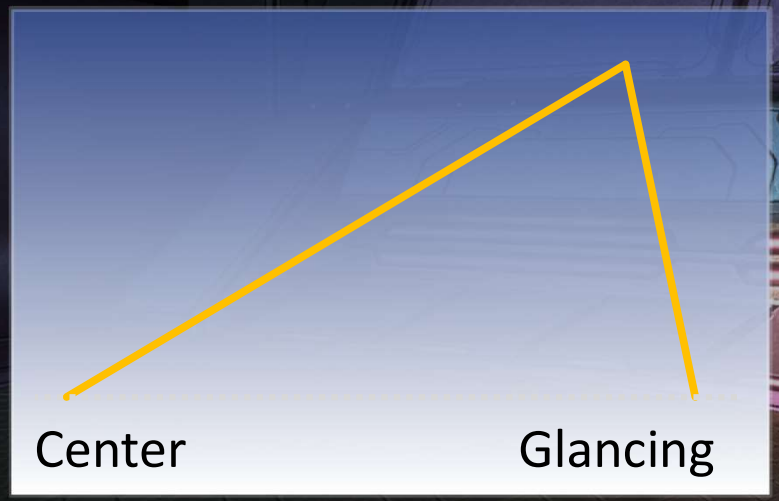
Extrude Vertices



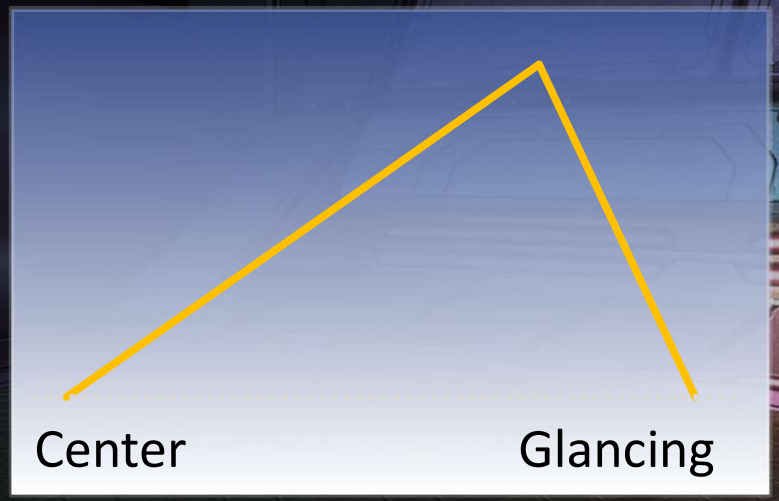
Depth Fade



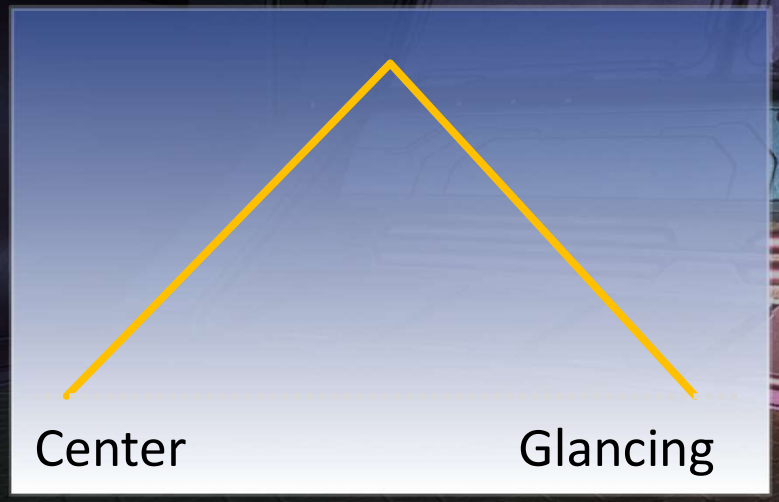
Custom Edge Fade



Custom Edge Fade



Custom Edge Fade



Custom Edge Fade

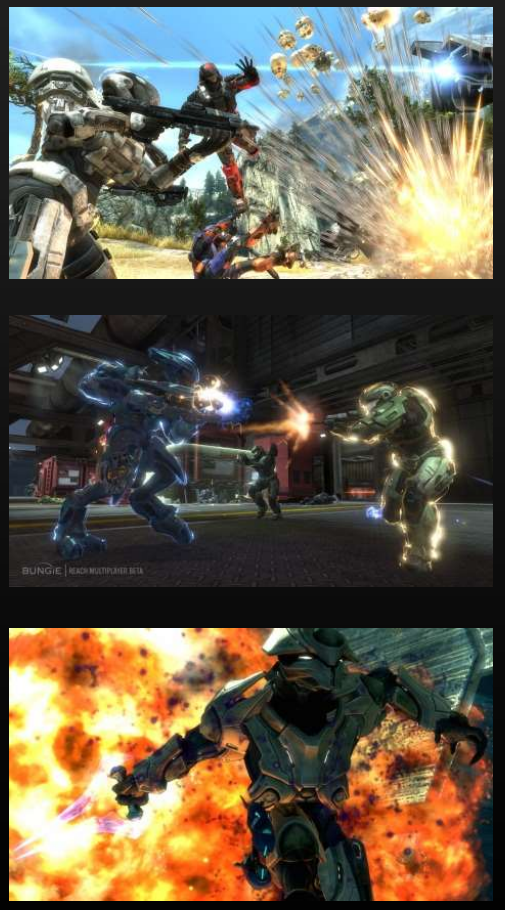
Depth Fade Plasma


Shield Doors
UNSC


Area Boundaries

Covered Techniques

- **Cheap Colliding Particles**
- **‘Shields’ & Depth Effects**
- **Low Resolution Transparents**



Overdraw Costs

- **Too many layers?**
  - Limited by fill rate & pixel shader
- **Render Fewer Pixels!**
  - Low resolution
  - Alias buffer as 4x MSAA
  - But want smooth upsampling…

BUNGiE

Reach Low Res Transparents

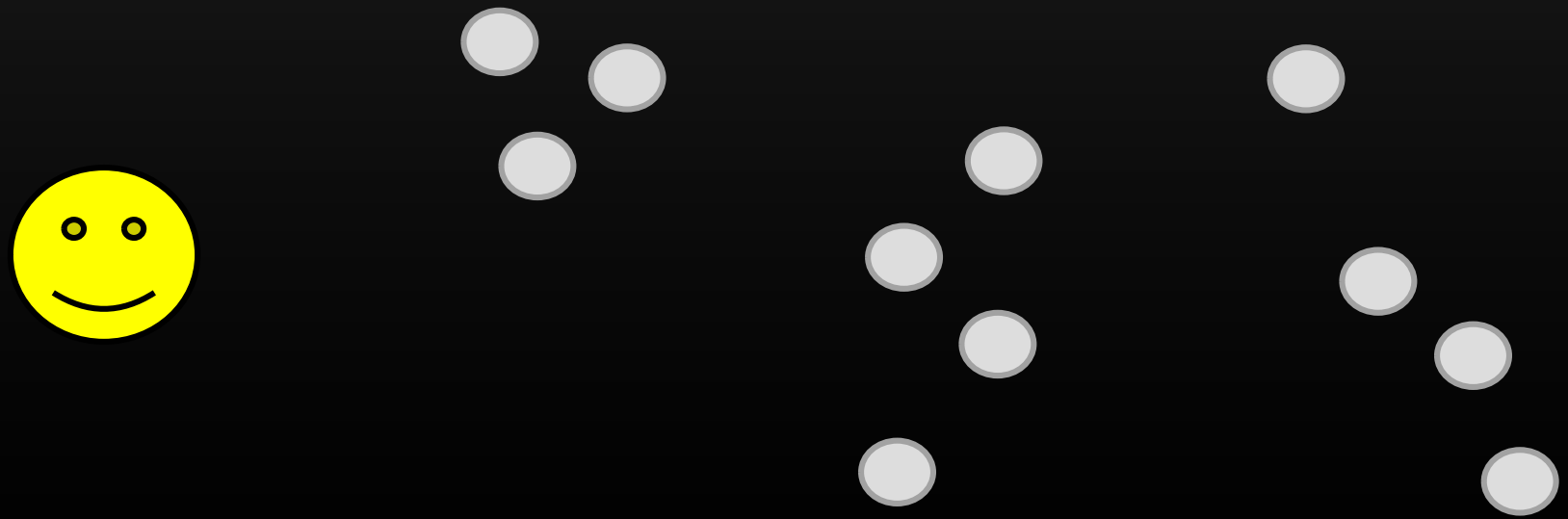
- **Render to separate buffer**
  - 1/2 resolution, 4x MSAA, use existing depth buffer
- **Inline with normal high-res transparency**
- **Low -> High transition: resolve and composite**
  - Smooth bilinear upsampling ☺
  - Transitions are expensive! ☹

Optimizing Transitions

- **Use low-res only when you expect savings**
  - Automatic switch for flagged transparents
- **Stencil touched areas to reduce composite fill**
- **Limit transitions by bucketing during sort**
  - N buckets means at most N-1 transitions
  - N = 3 worked well

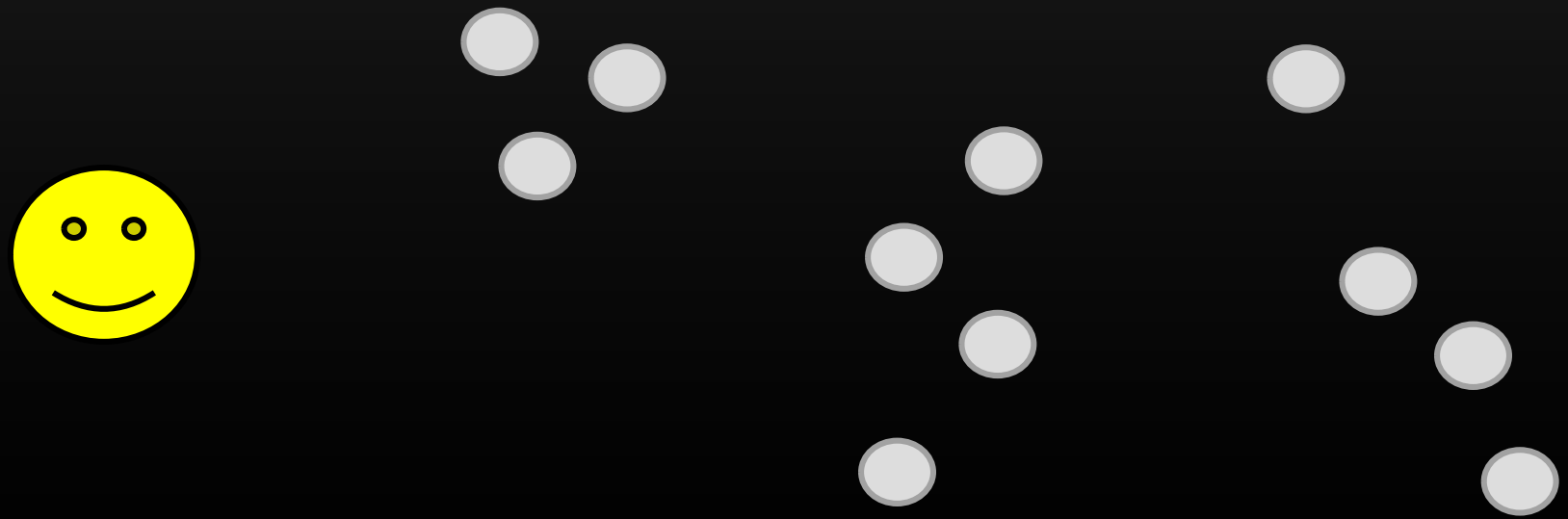
Well-Behaved Bucketing

- **How to group these to minimize sort pops?**



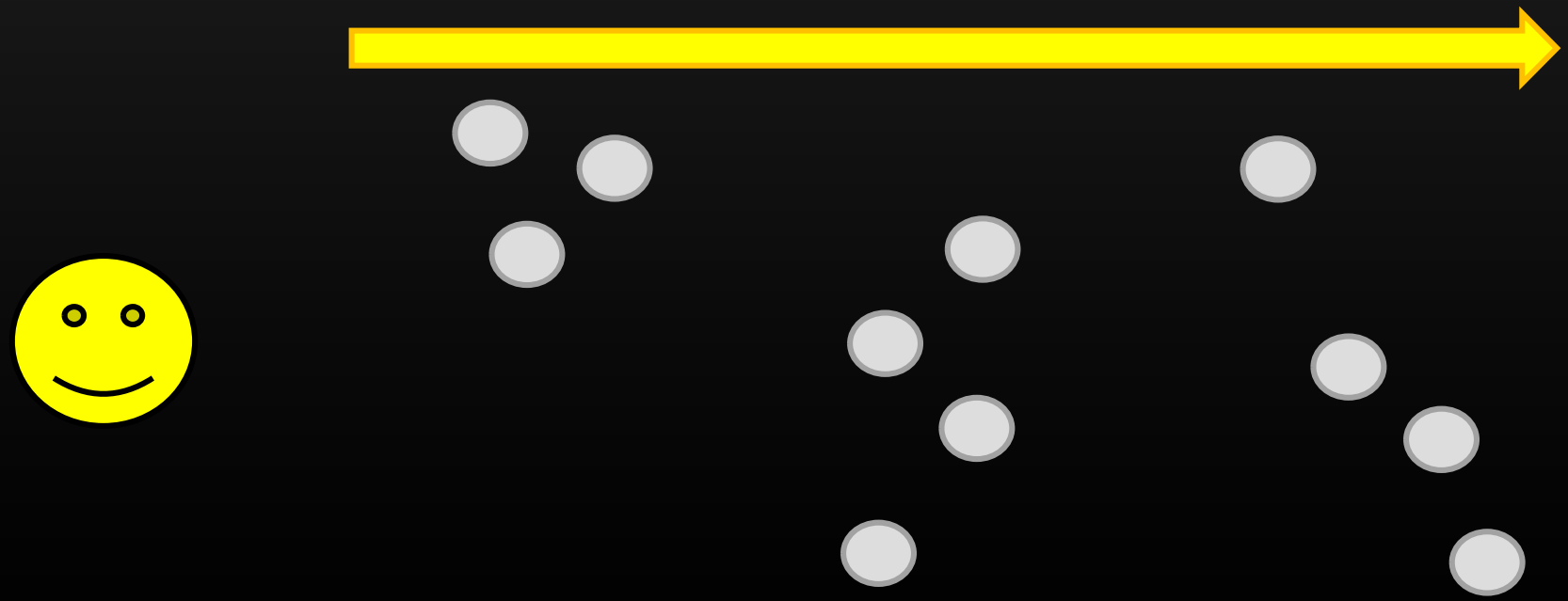
Best Behaved Bucketing

- **For N buckets, find (N-1) largest gaps**



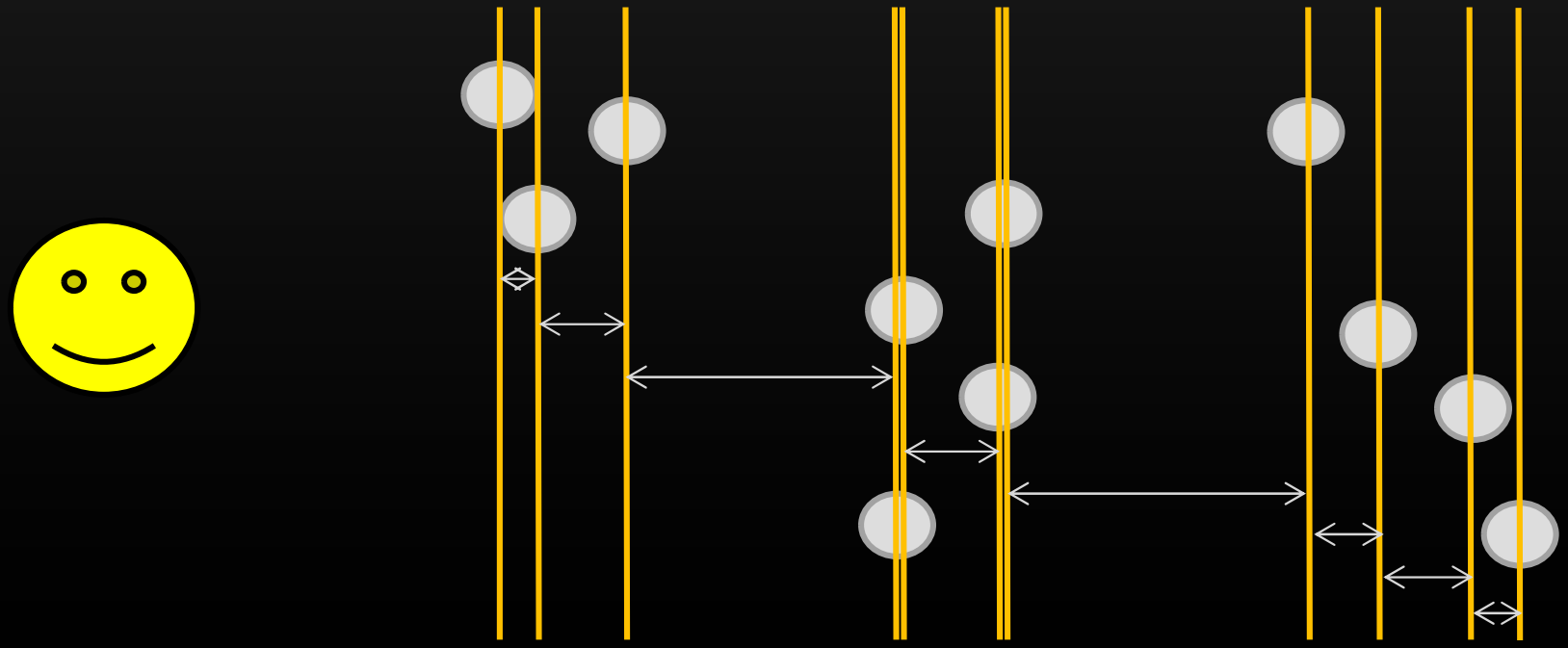
Best Behaved Bucketing

- **Sort by Depth**



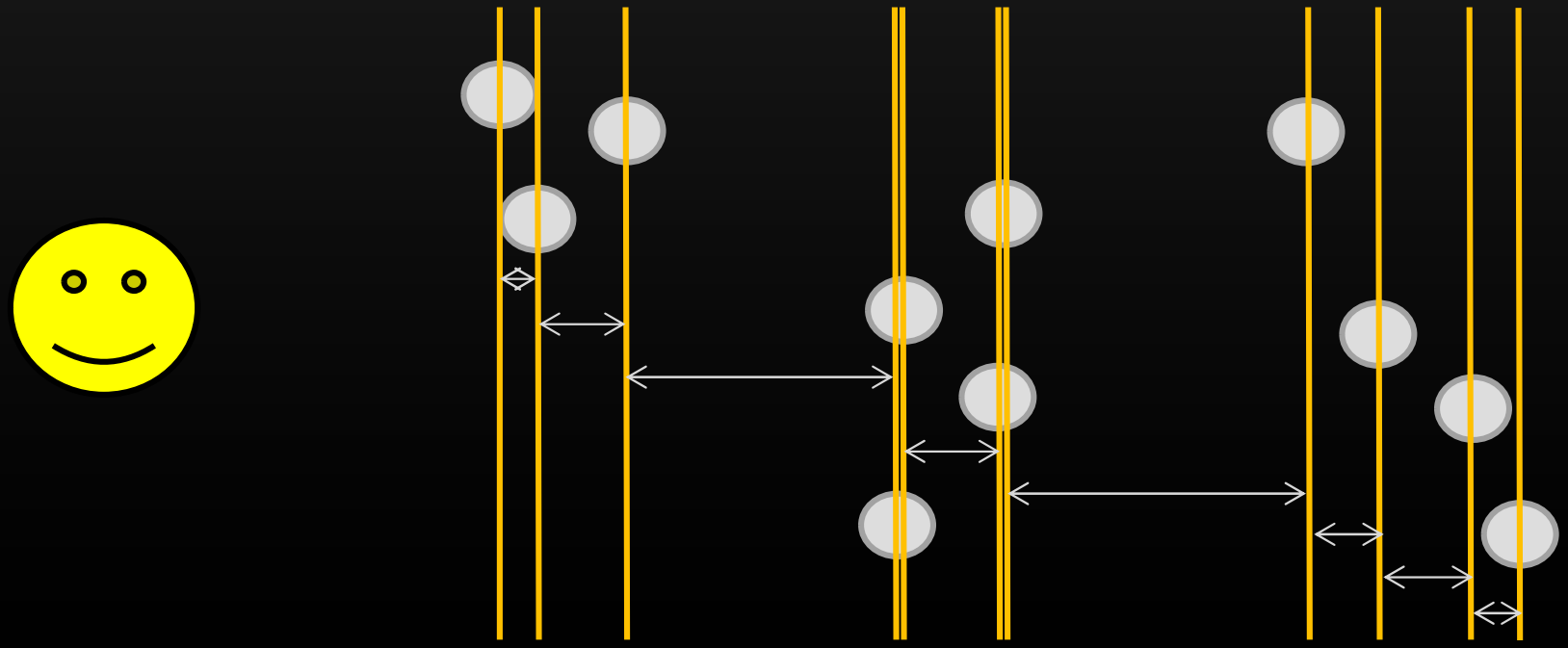
Best Behaved Bucketing

- **Distance between neighbor elements**
  - Use heap to grab (N-1) largest



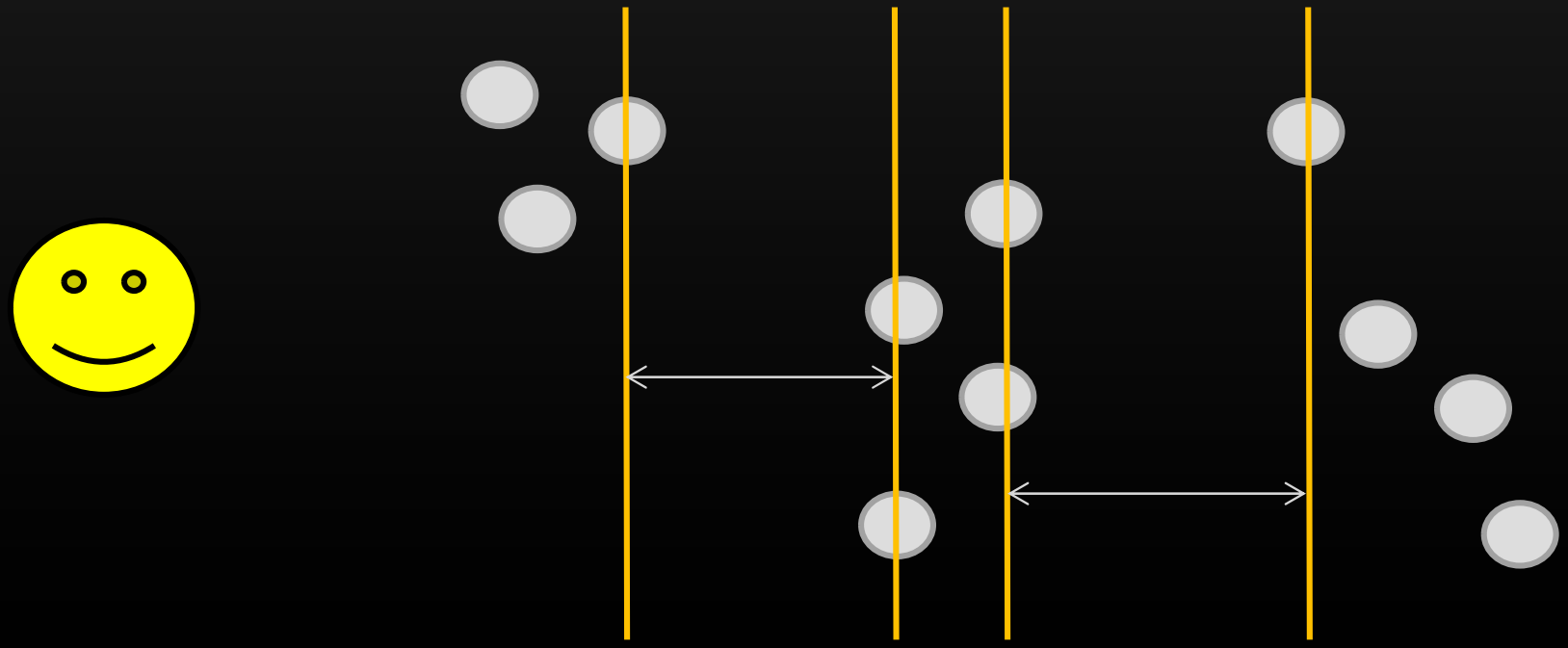
Best Behaved Bucketing

- **Distance between neighbor elements**
  - Use heap to grab (N-1) largest



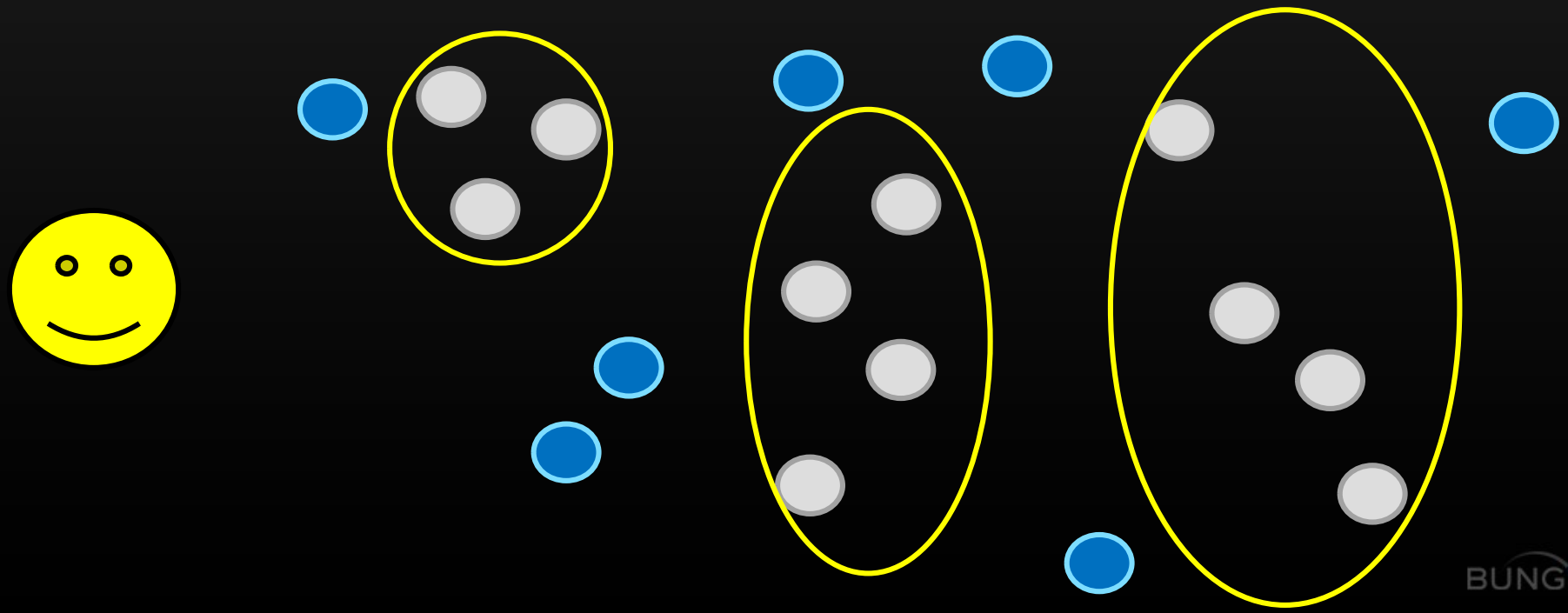
Best Behaved Bucketing

- **Distance between neighbor elements**
  - Use heap to grab (N-1) largest



Best Behaved Bucketing

- **Divide at these gaps**
  - Render each group together, mixed in with high-rez content




SW
W
NW
25m

Covered Techniques

- **Cheap Colliding Particles**
- **‘Shields’ & Depth Effects**
- **Low Resolution Transparents**

BUNGiE

Summary

- **Understand the desires of your content guys**
  - Big explosions, smoke, stuff flying everywhere!
- **Figure out where they are limited and fix!**
- **You can often sacrifice correctness for speed**
  - And minimize artifacts with work arounds
- **Depth buffer is incredibly useful**


E
SE
S
SW
4
LARZ
Lars
02
6:11
Firefight
01
6
3859
25m
Circle

We’re hiring!

careers@bungie.com http://bungie.net/careers

Chris Tchou
ctchou@bungie.com

BUNGiE