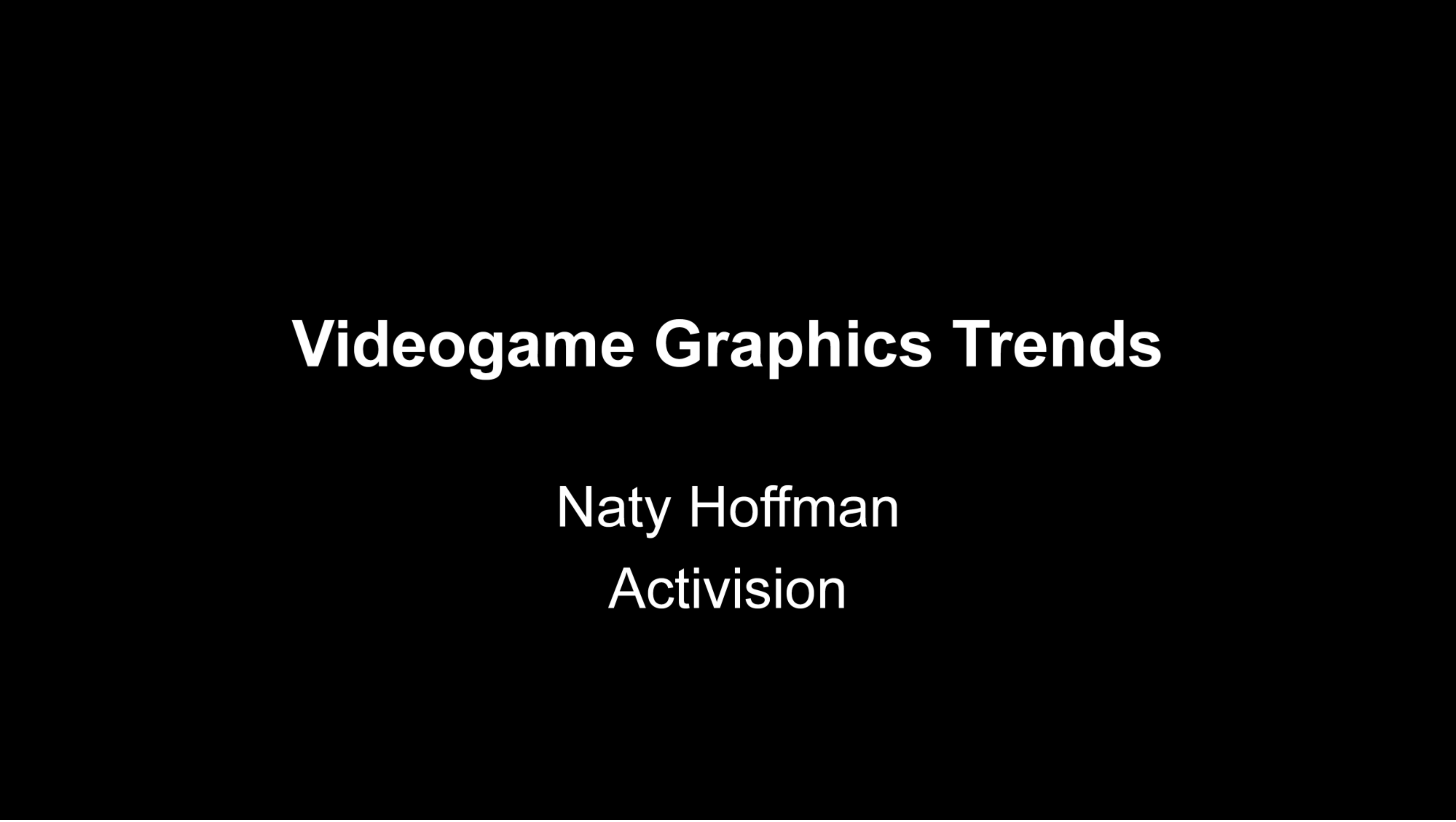


Videogame Graphics Trends

Naty Hoffman

Activision

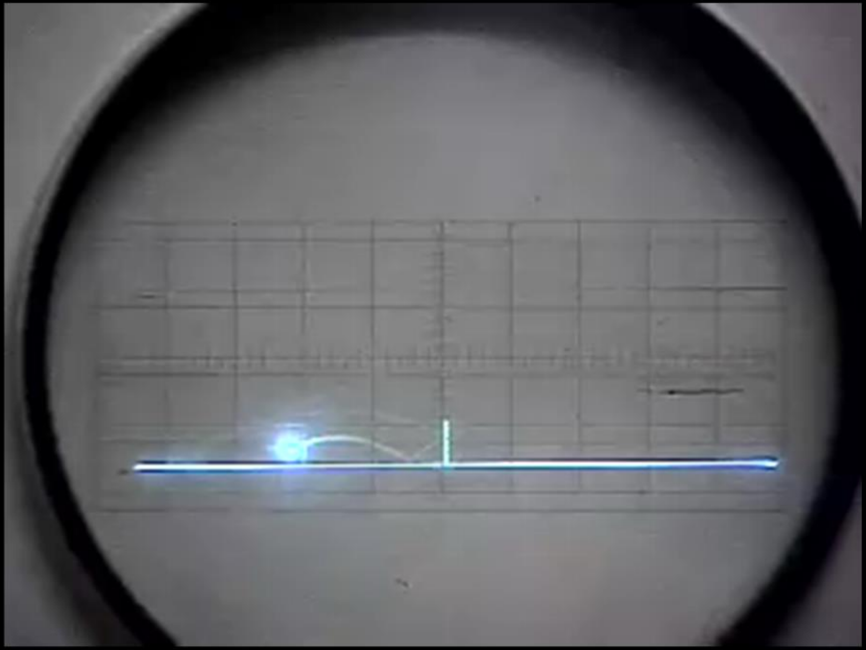
Important Disclaimer

Everything in this talk is my personal opinion, and not in any way the official position of Activision Blizzard

CC Image by User:LBurdak

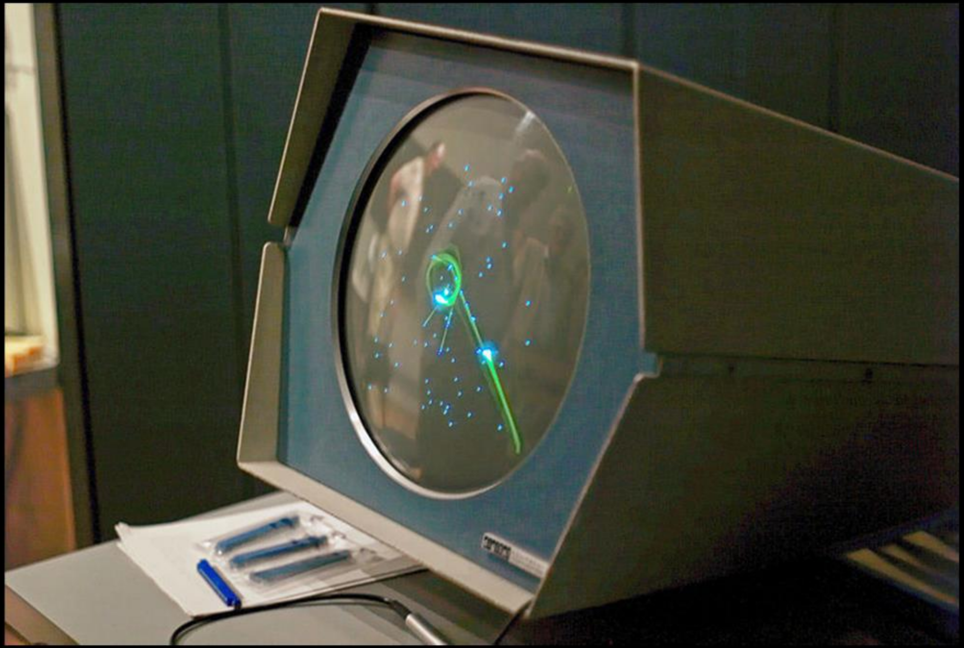
First, some prehistory.

1958 – Tennis for Two



PD Image by William Higinbotham

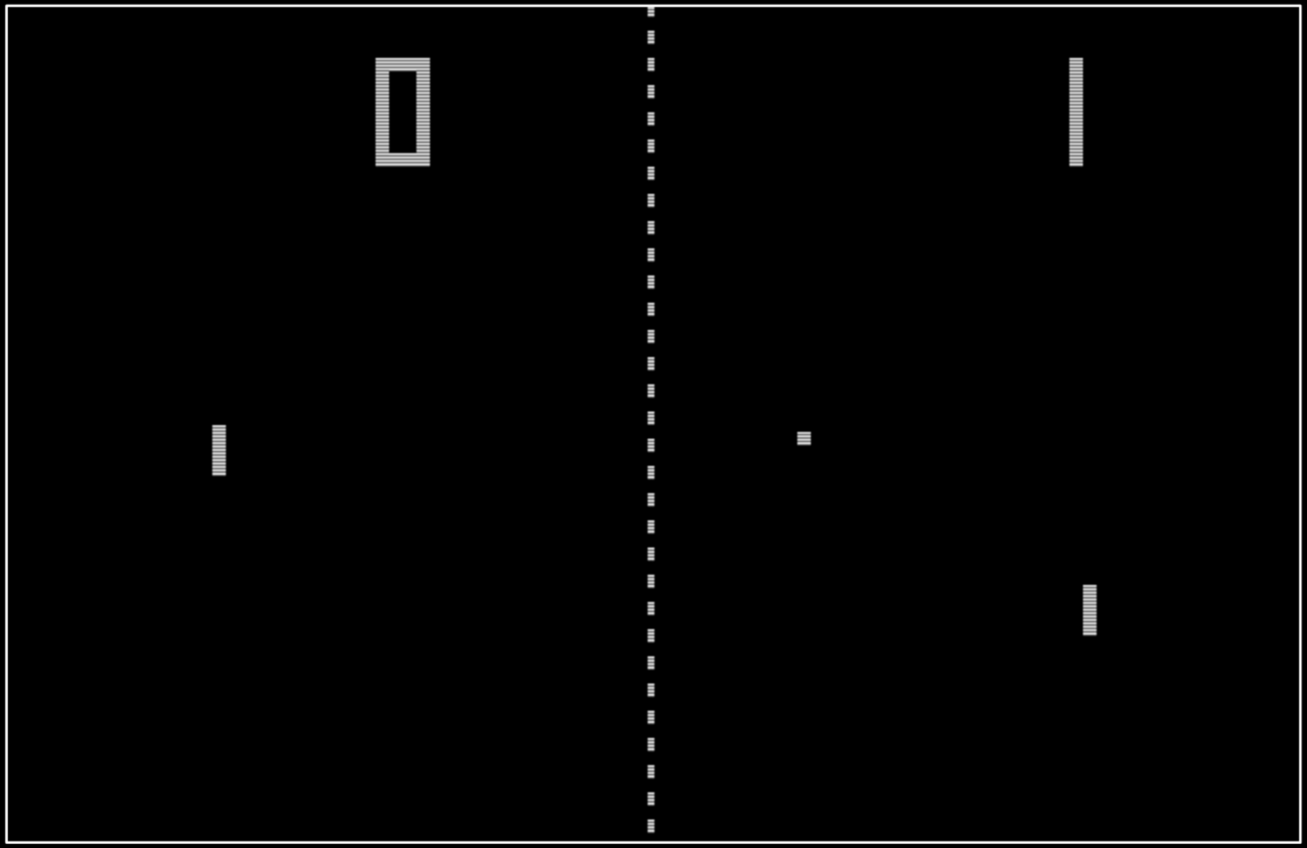
1961 – Spacewar!



CC Image by Joi Ito

“Tennis for Two” wasn’t even a program running on a digital computer – it was analog circuitry hooked up to an oscilloscope. Spacewar! introduced the important innovation of exclamation marks in game titles, and was running on an actual computer (PDP-1) connected to an actual computer display (DEC Precision CRT Display Type 30).

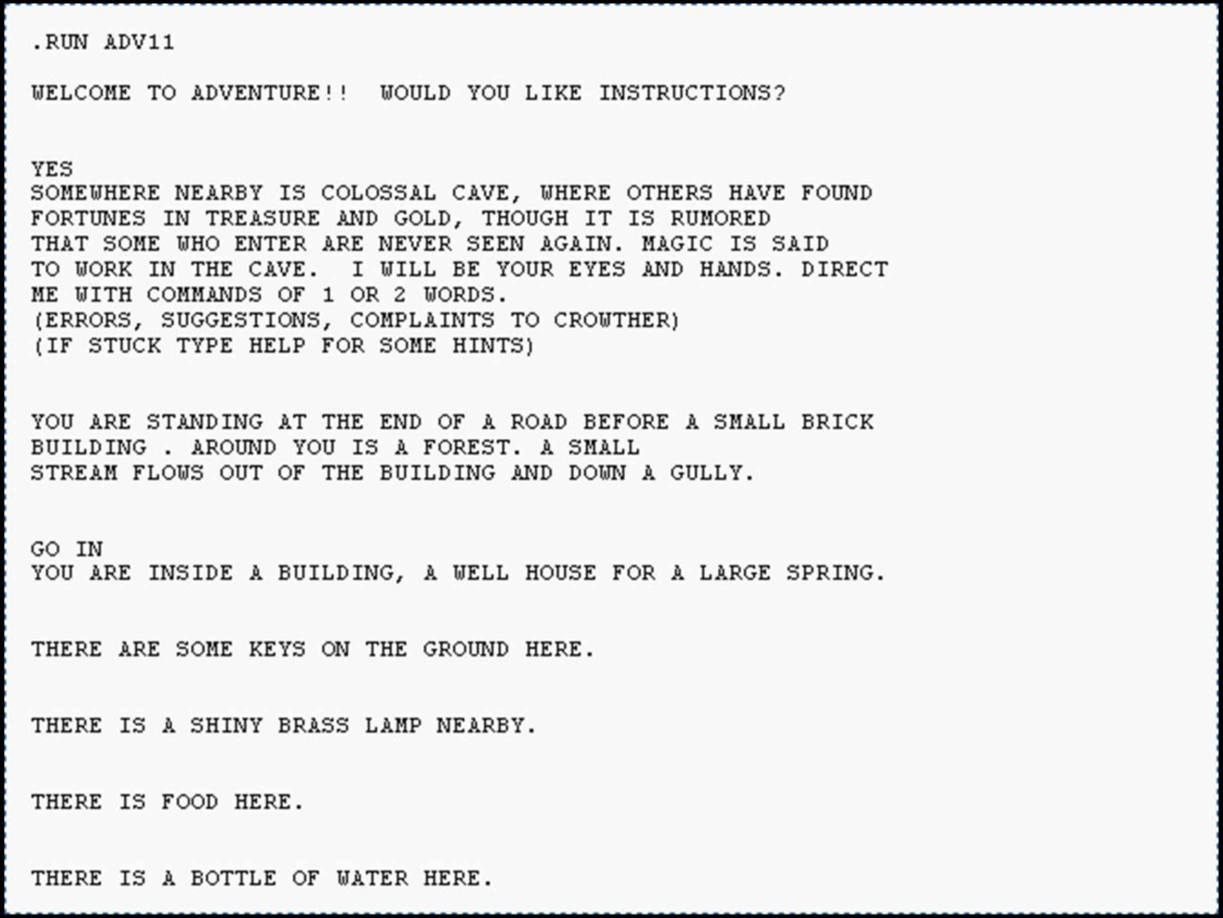
1972 – Pong



PD Image by User:Bumm13

Game graphics during this period (let’s call it the “block era” were simple vector shapes, blocks, or…

1975 – ADVENT



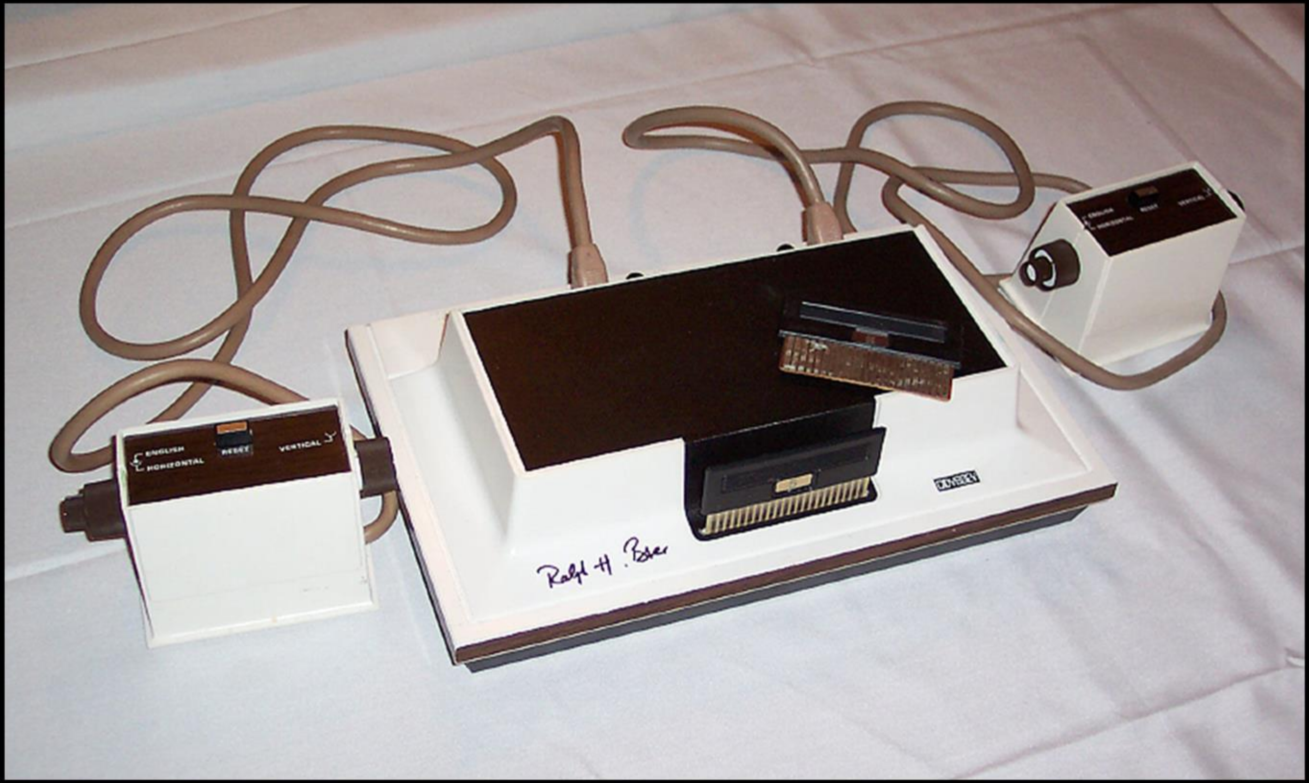
```
.RUN ADV11

WELCOME TO ADVENTURE!!  WOULD YOU LIKE INSTRUCTIONS?


YES
SOMEWHERE NEARBY IS COLOSSAL CAVE, WHERE OTHERS HAVE FOUND
FORTUNES IN TREASURE AND GOLD, THOUGH IT IS RUMORED
THAT SOME WHO ENTER ARE NEVER SEEN AGAIN. MAGIC IS SAID
TO WORK IN THE CAVE.  I WILL BE YOUR EYES AND HANDS. DIRECT
ME WITH COMMANDS OF 1 OR 2 WORDS.
(ERRORS, SUGGESTIONS, COMPLAINTS TO CROWTHER)
(IF STUCK TYPE HELP FOR SOME HINTS)


YOU ARE STANDING AT THE END OF A ROAD BEFORE A SMALL BRICK
BUILDING . AROUND YOU IS A FOREST. A SMALL
STREAM FLOWS OUT OF THE BUILDING AND DOWN A GULLY.


GO IN
YOU ARE INSIDE A BUILDING, A WELL HOUSE FOR A LARGE SPRING.


THERE ARE SOME KEYS ON THE GROUND HERE.


THERE IS A SHINY BRASS LAMP NEARBY.


THERE IS FOOD HERE.


THERE IS A BOTTLE OF WATER HERE.
```

PD Image by User:Prof jerz

…or no graphics at all.

1972 – Magnavox Odyssey

First Console

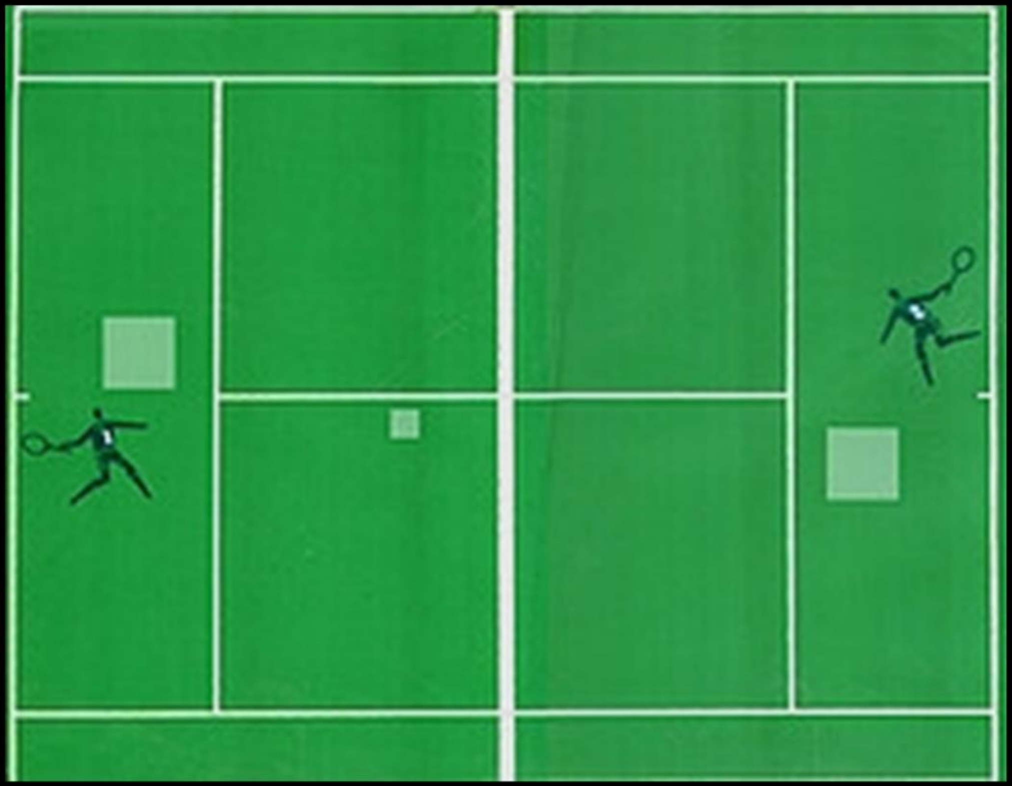


CC Image by Martin Goldberg

The first home game console came out during this period.

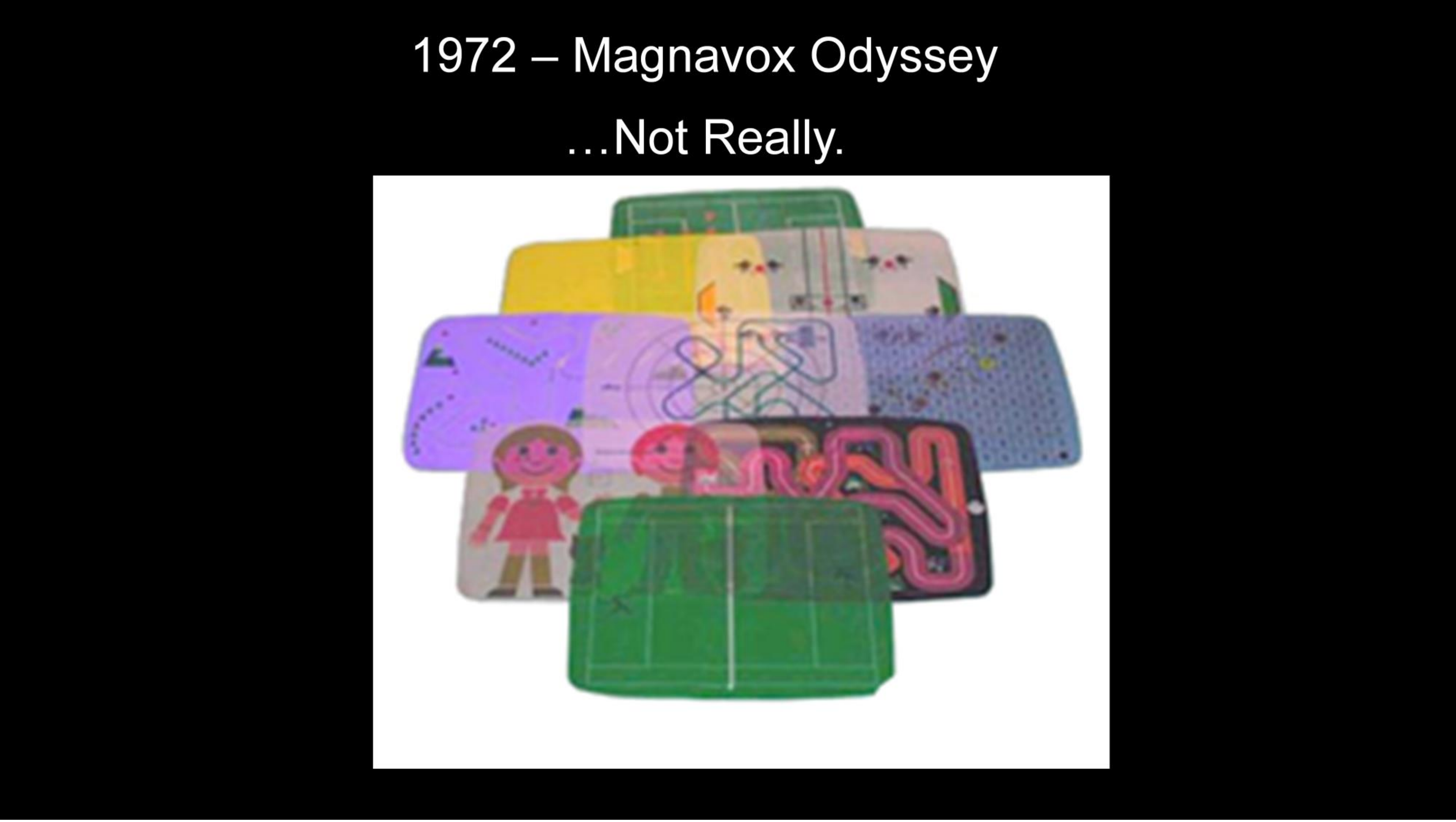
1972 – Magnavox Odyssey

Woohoo! Color! High Resolution!



And had hugely improved graphics! Or did it?

(Magnavox Odyssey Tennis screen image used under fair use – low-res image for scholarly commentary)



1972 – Magnavox Odyssey

…Not Really.

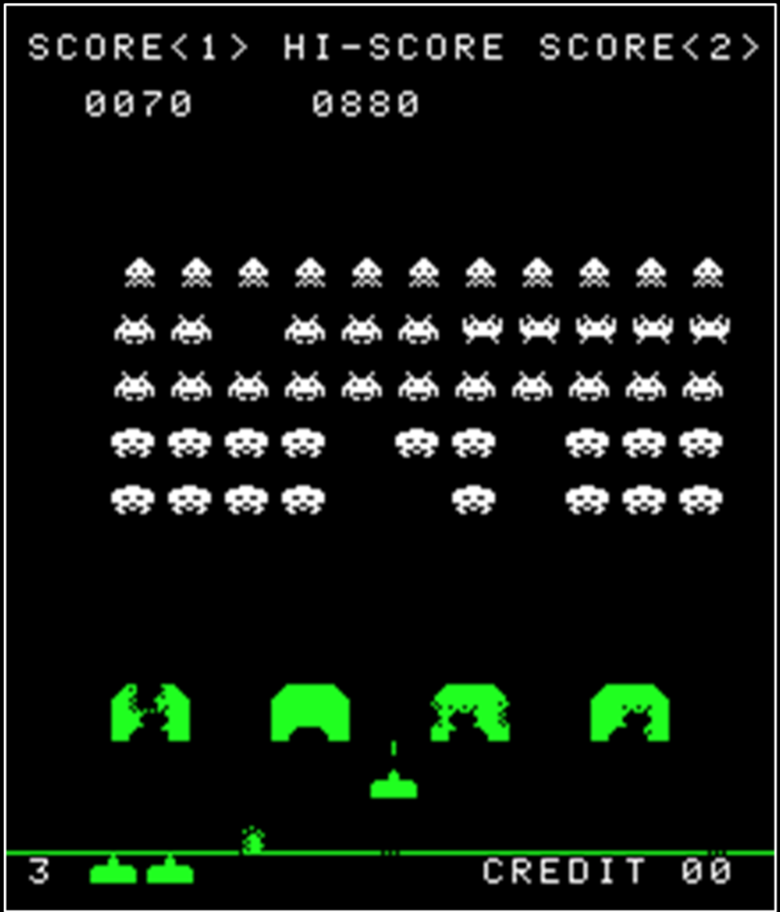
Not really – all the color and detail came from these semi-transparent colored screen overlays. The actual graphics were still just blocks.

(Magnavox Odyssey screen overlay image used under fair use – low-res image for scholarly commentary)

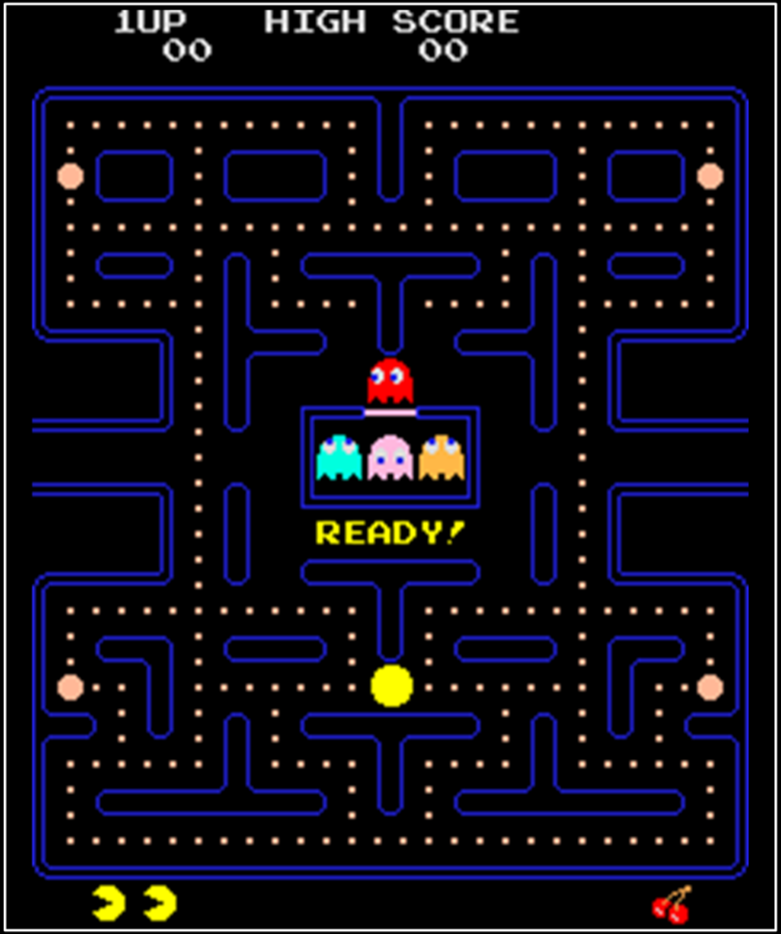
CC Image by Steve Swayne

We’re past prehistory and in ancient history now – moving from the “block era” to the “bitmap era”.

1978 – Space Invaders



1980 – Pac-Man



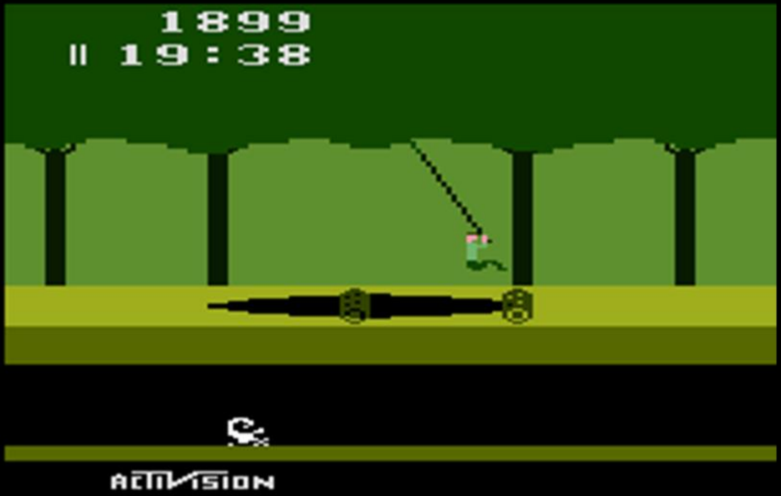
In the late 1970's, game graphics were well beyond moving blocks, using colored bitmaps extensively. This enabled some actual visual creativity for the first time, e.g. painting and animating the bitmaps.

(Space Invaders and Pac-Man screenshots used under fair use – low-res images for scholarly commentary).

1977 – Atari 2600

PD Image by Evan-Amos

1982 – Pitfall!



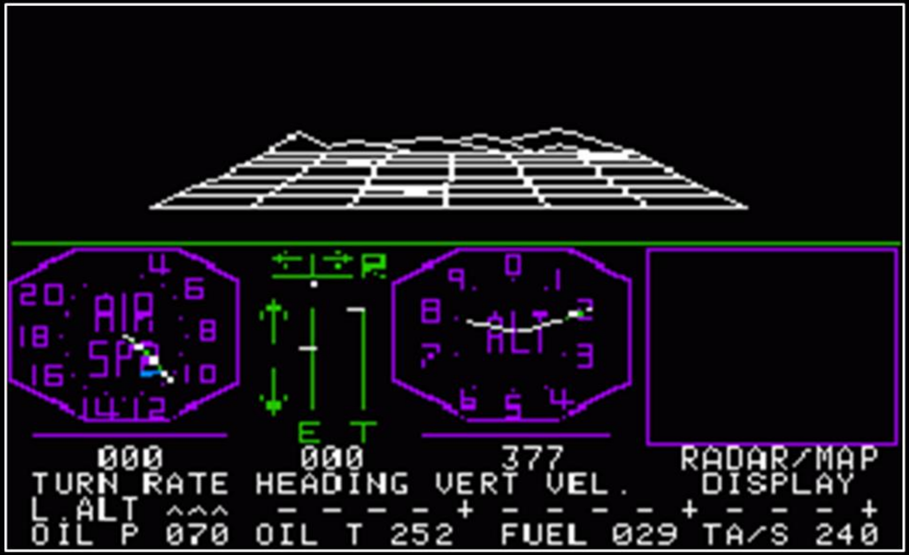
The second wave of home consoles brought bitmap graphics from the arcade to the home. In one sense, the bitmap era never ended – 2D games with hand-painted sprites and backgrounds continued to find audiences, and still do today (e.g., Angry Birds). But at some point, the center of gravity shifted to a new technology…

(Pitfall! screenshot used under fair use – low-res image for scholarly commentary)

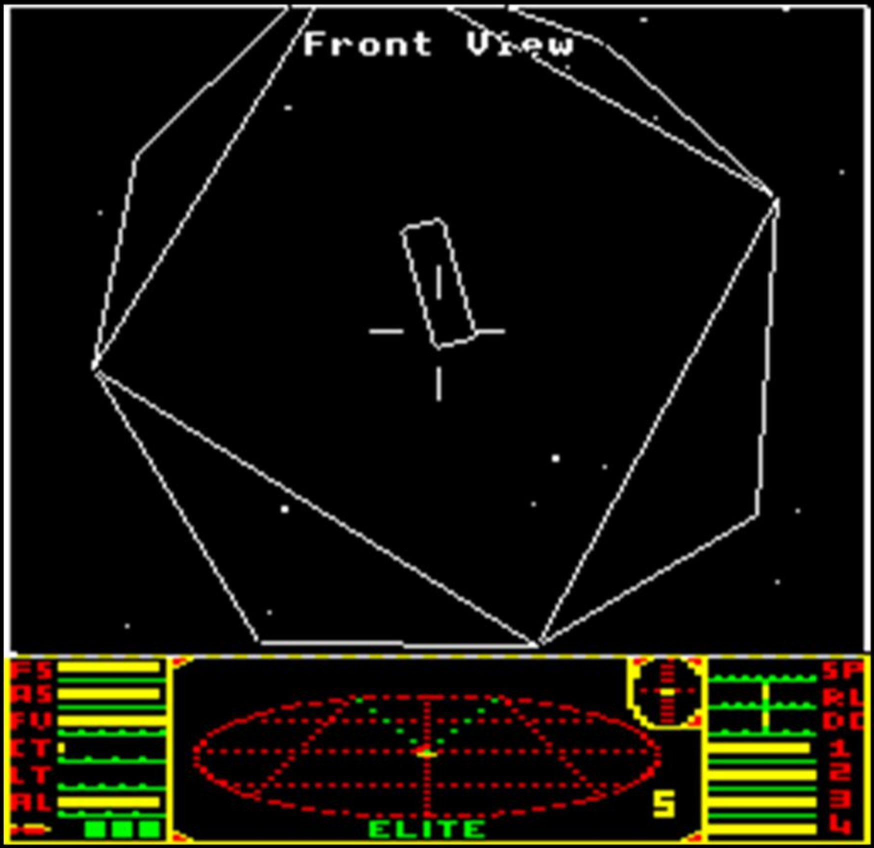
Image courtesy of “Age of Battles”

The “middle ages” of game graphics - early 3D, done without dedicated graphics hardware.

1980 – Flight Simulator


AIR SPD
ALT
E T
000 000 377
TURN RATE HEADING VERT VEL.
RADAR/MAP DISPLAY
L.ALT
OIL P 070 OIL T 252 FUEL 029 TA/S 240

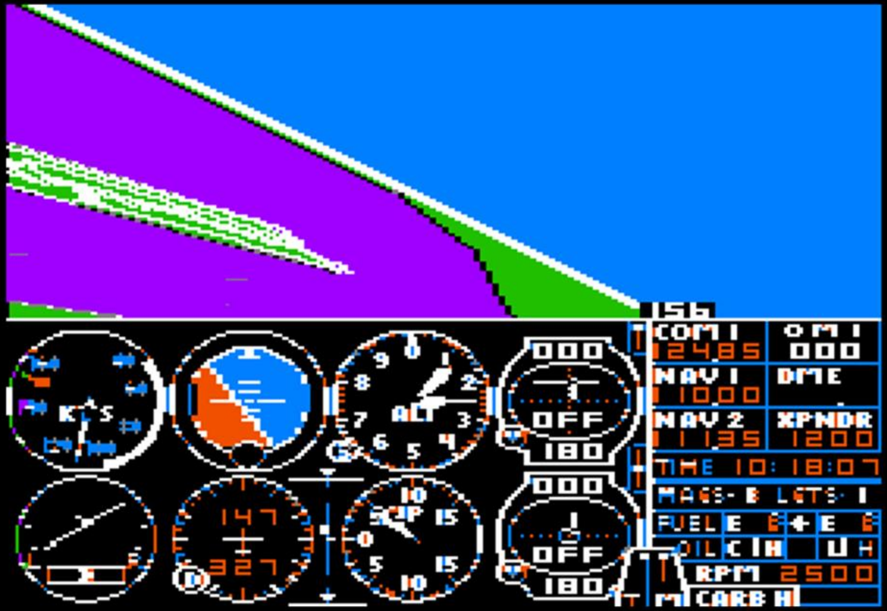
1984 – Elite



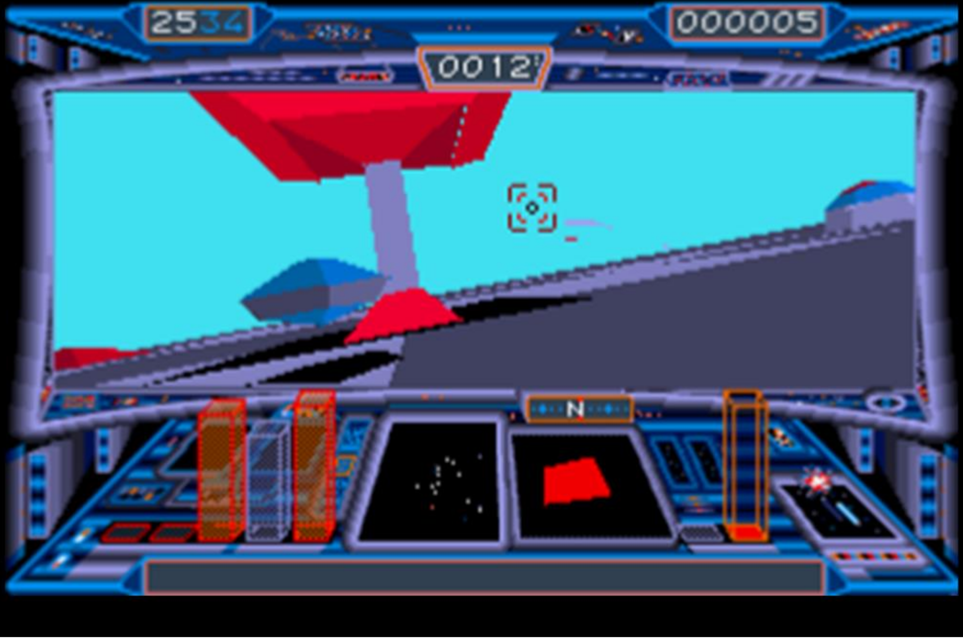
Early 3D games – different presentation modes, each going from limited shapes to more general environments. The first presentation mode used was wireframe – at first for very small heightfields, but eventually for more arbitrary scenes.

(Flight Simulator & Elite screenshots used under fair use – low-res images for scholarly commentary)

1983 – Flight Simulator II



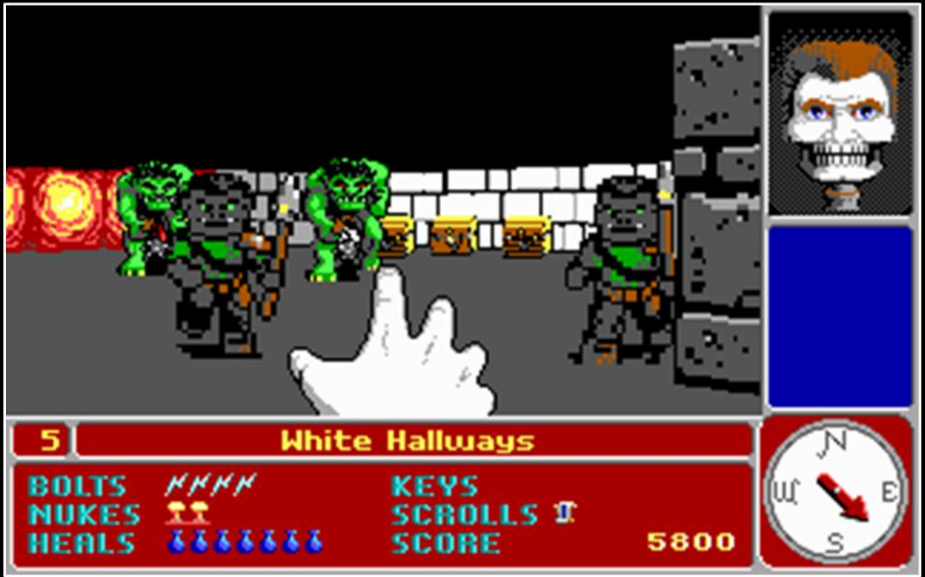
1988 – Starglider 2



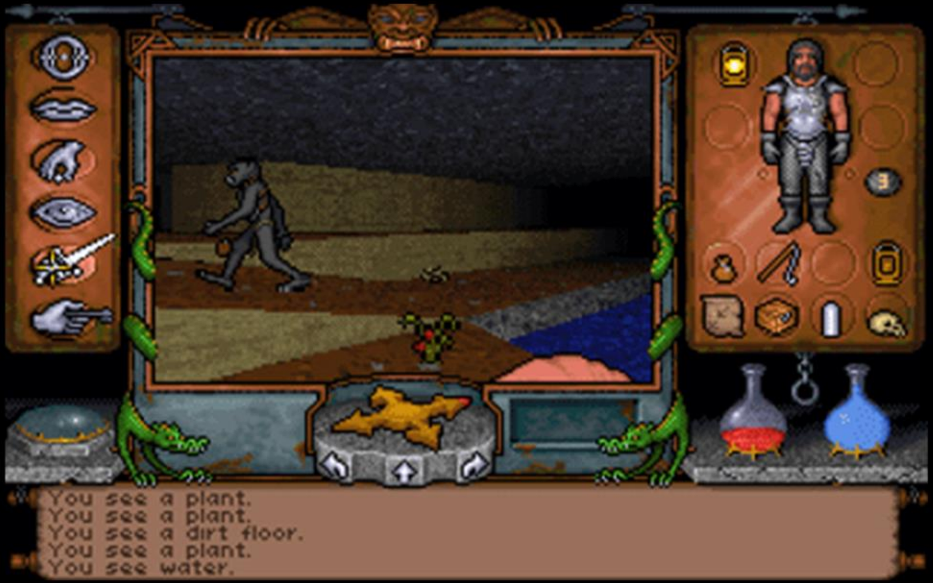
Next came solid colored surfaces, again going from more restricted to more general environments.

(Flight Simulator 2 and Starglider 2 screenshots used under fair use – low-res images for scholarly commentary)

1991 – Catacomb 3-D



1992 – Ultima Underworld

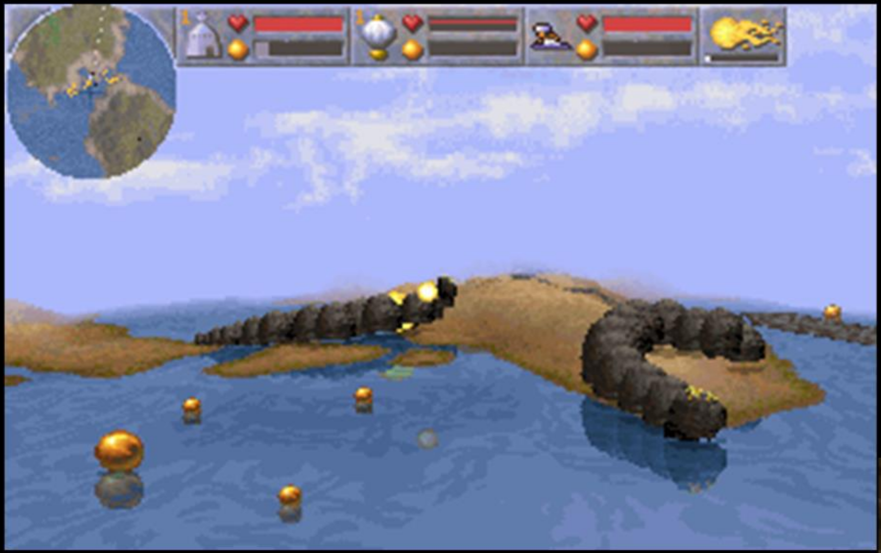


You see a plant.
You see a plant.
You see a dirt floor.
You see a plant.
You see water.

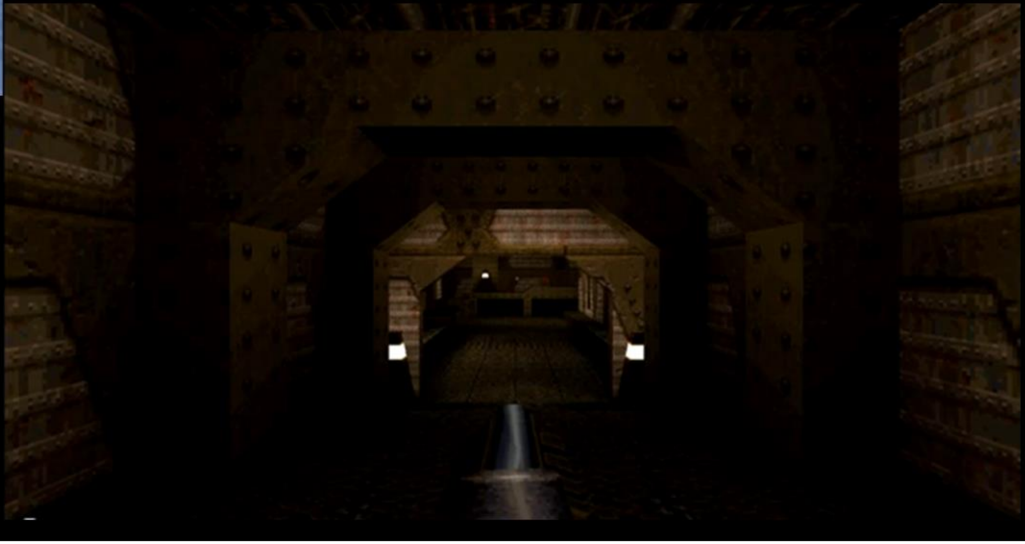
This era culminated with textured surfaces – as before on older systems this could only be done on very restricted environments, but soon the environments became more general…

(Catacomb 3-D and Ultima Underworld screenshots used under fair use – low-res images for scholarly commentary)

1994 – Magic Carpet

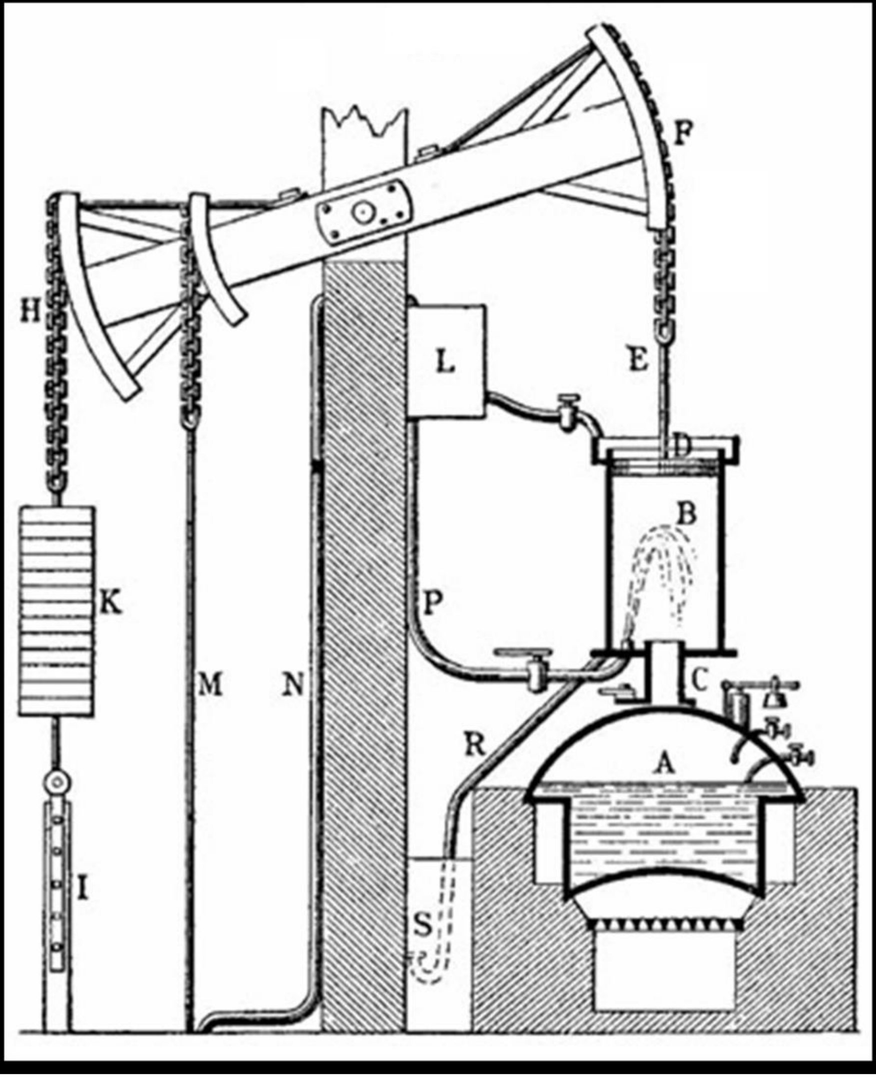


1996 - Quake



…and even more general. Quake is especially notable, since it embodied the transition to the next era.

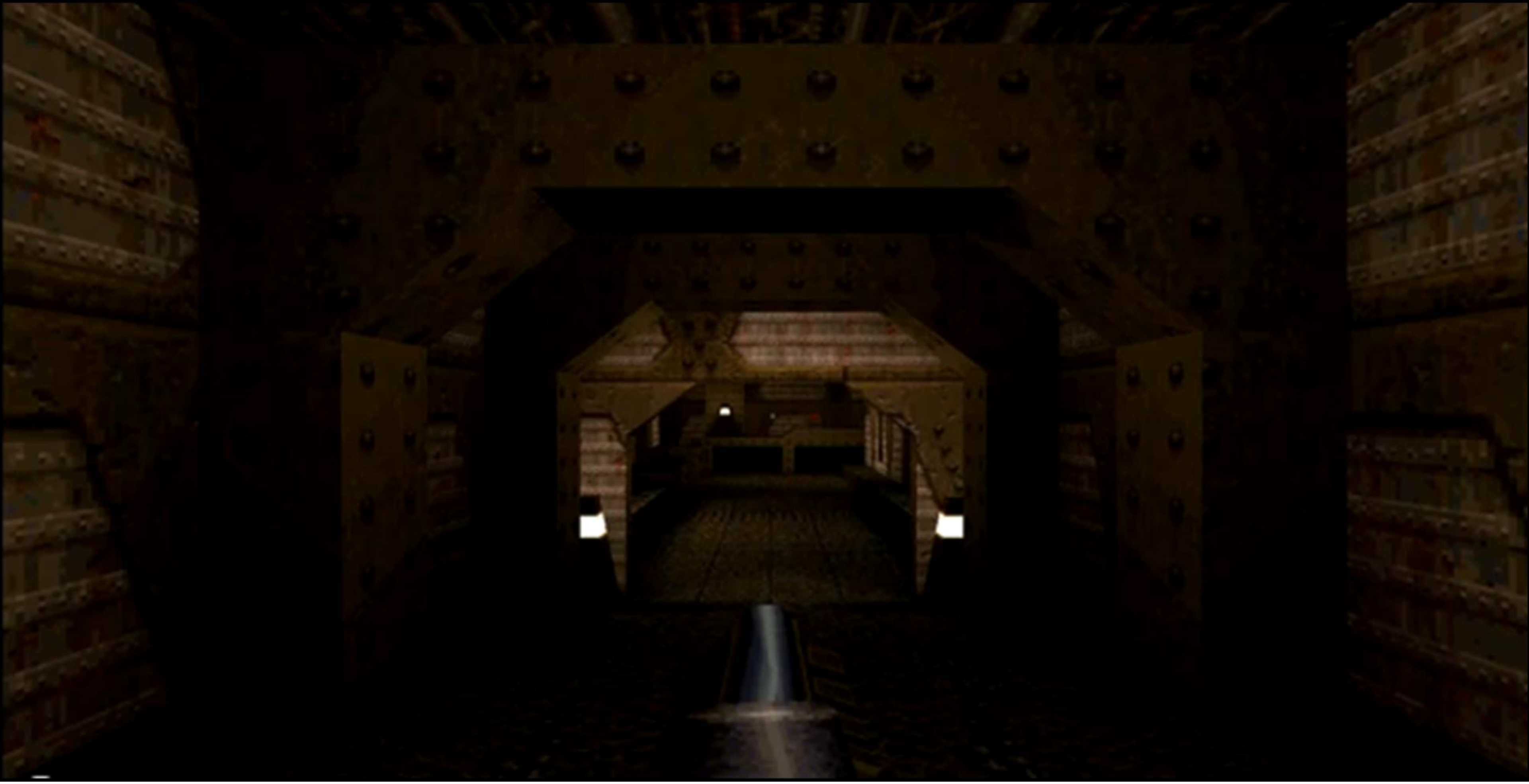
(Magic Carpet and Quake screenshots used under fair use – low-res images for scholarly commentary)



PD Image from *Meyers Konversationslexikon*, 1890

Graphics hardware formed the basis of our industrial revolution, and games graphics would never be the same.

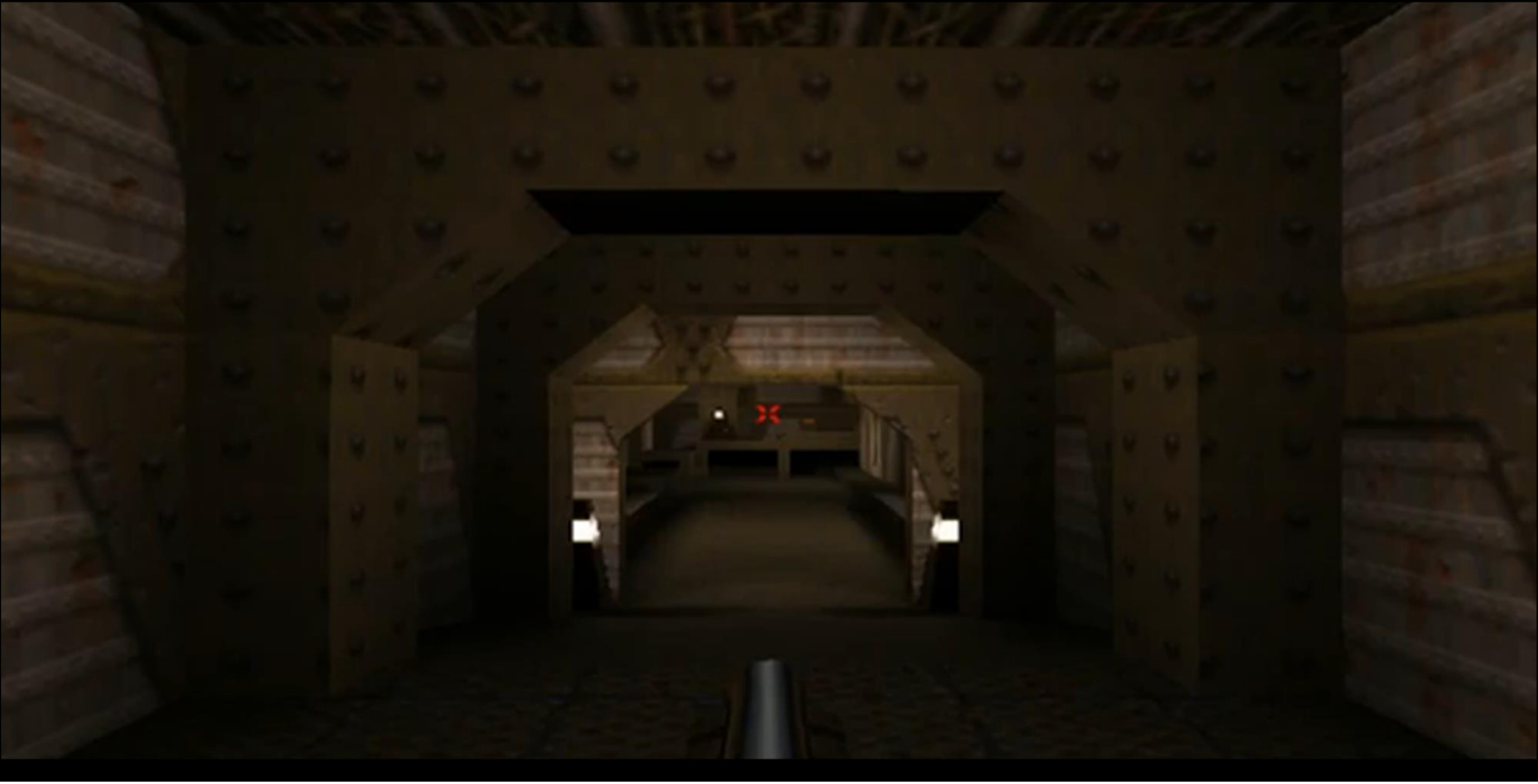
1996 - Quake



I remember when I first saw the difference between Quake…

(Quake screenshot used under fair use – low-res image for scholarly commentary)

1997 – GL Quake



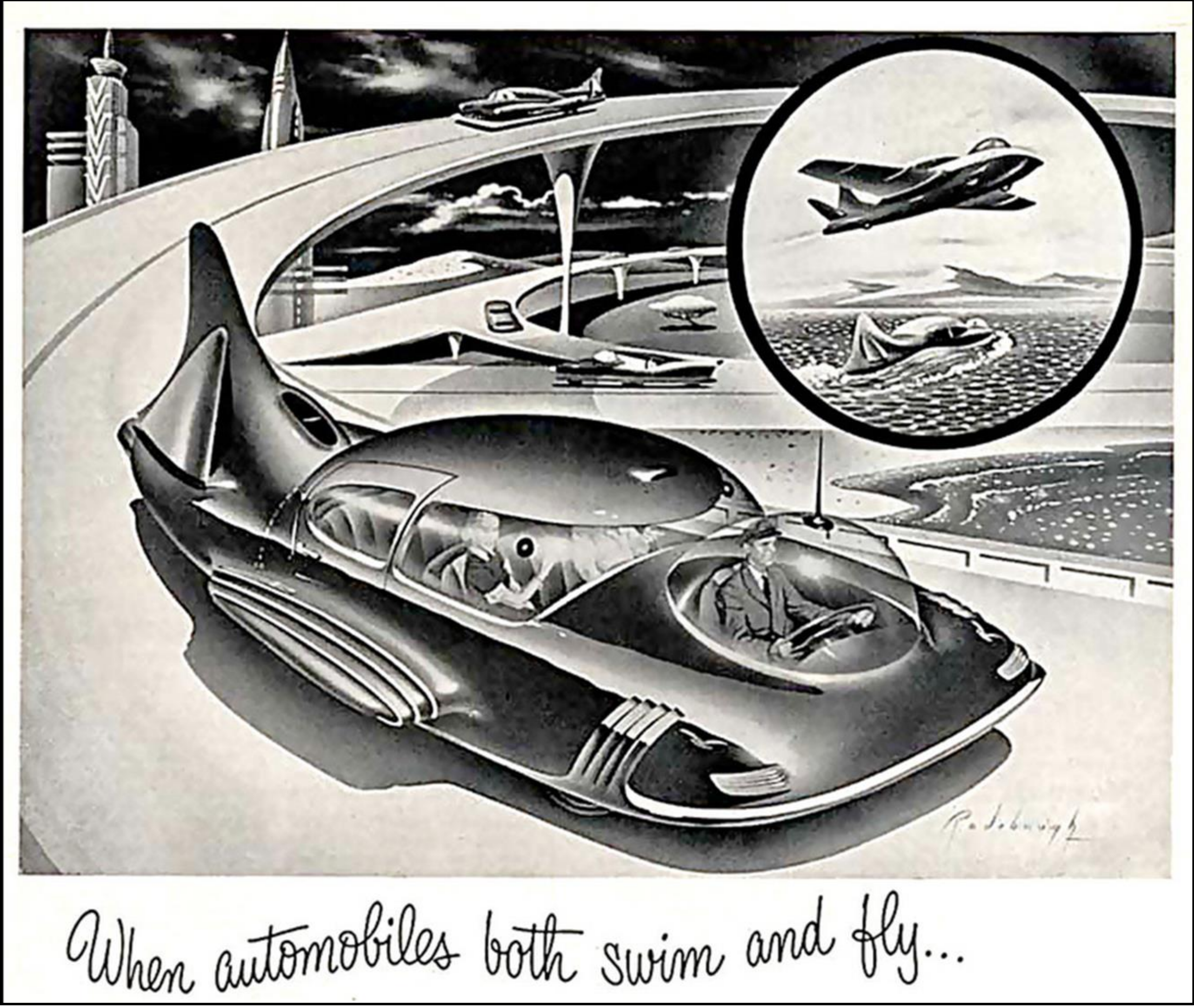
… and GL Quake, I was blown away. These screenshots don't really do it justice – the combination of higher resolution, bilinear filtered textures, antialiased point sprites, etc. was mindboggling.

(GL Quake screenshot used under fair use – low-res image for scholarly commentary)

CC Image by Thom Quine

I’ll skip over the many games that used fixed-function graphics and get right to our current era, which is defined by programmable shaders. From the early DirectX 8 days until today, the primary differences have been quantitative (more performance, more instructions, more and bigger textures). Many of the bigger hardware features (geometry shaders, tessellation) have not been heavily used by games. GPU compute looks like it could be the next big thing, but it hasn’t really taken off yet.

Future Trends in Game Graphics



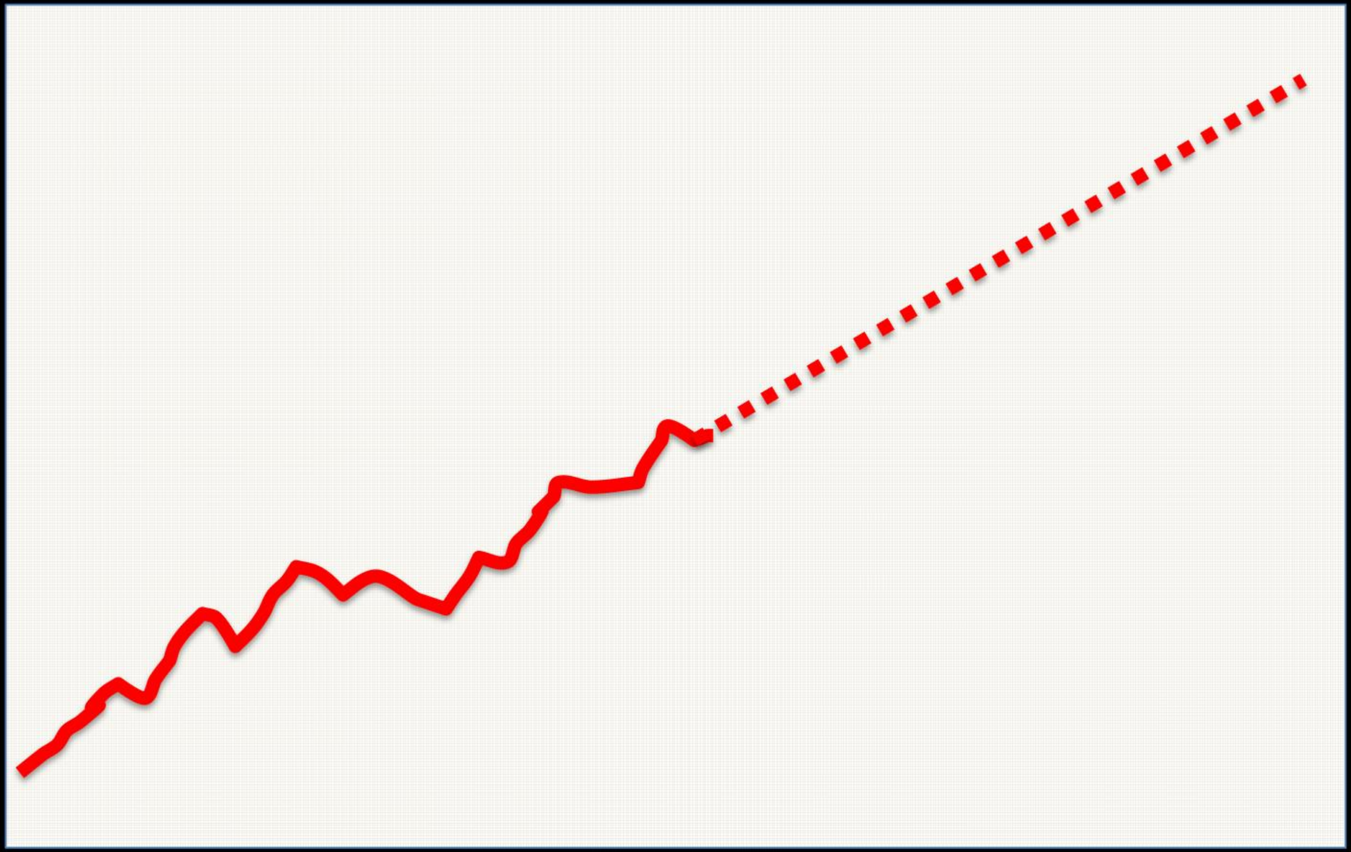
CC Image by flickr:x-ray delta one

For the rest of this talk, I’ll go over a few candidate future trends and talk about whether I think they are “for real” or the game graphics version of the flying car (hey, this one swims too!).

"Prediction is difficult, especially about the future" – Neils Bohr

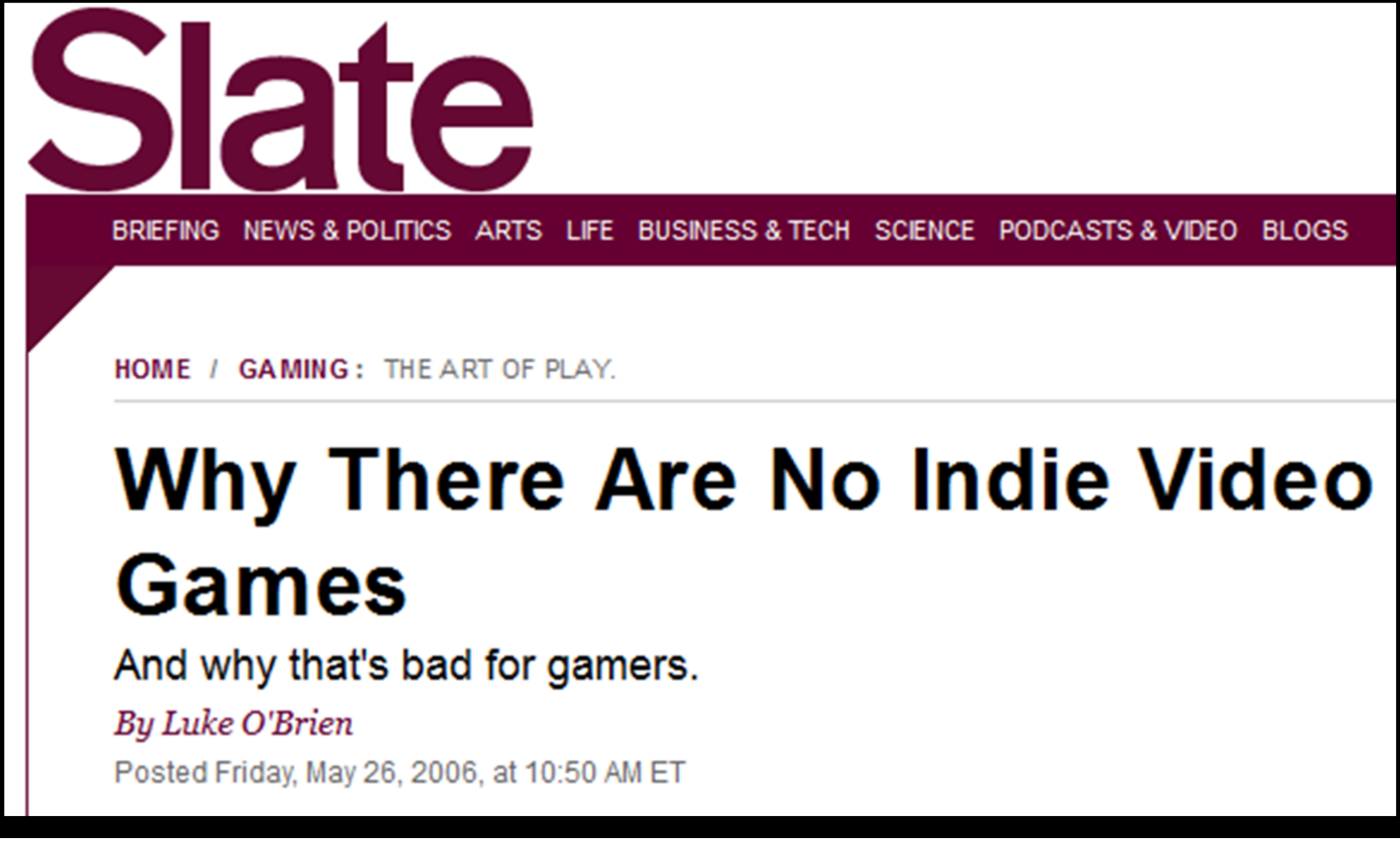
Predicting what's next is famously a hard problem. The "Captain Future" stories (written in the 40's by Edmund Hamilton) took place in 1990. Now I don't know about you, but I sure don't remember 1990 looking anything like that. To hopefully avoid similar errors, I'm going to need all the help I can get.

("Captain Future" cover used under fair use – low-resolution cover image used for commentary about the artwork)



One tool that can sometimes be useful is extrapolation of current trends, though it rapidly loses its utility as you try to look farther ahead, especially in an industry as rapidly changing as videogames.

Game business changes VERY quickly



Slate

BRIEFING NEWS & POLITICS ARTS LIFE BUSINESS & TECH SCIENCE PODCASTS & VIDEO BLOGS

HOME / GAMING: THE ART OF PLAY.

Why There Are No Indie Video Games

And why that's bad for gamers.

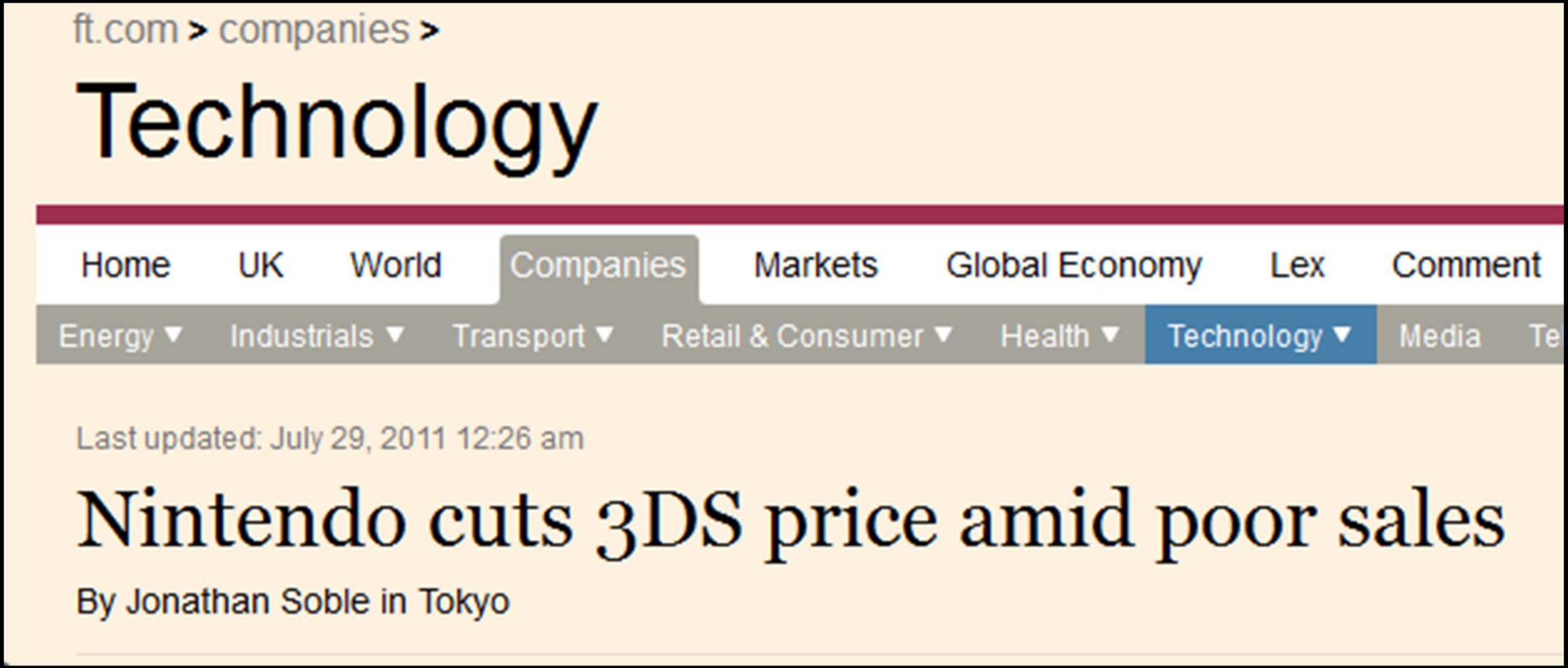
By Luke O'Brien

Posted Friday, May 26, 2006, at 10:50 AM ET

The rapid changes in the game industry – on the business side, not necessarily the technology side, can give you whiplash and prove simple extrapolations wrong. This was written in 2006 – now there is a triving indie game development scene which feels like it has always been around.

(image from "Slate" website used under fair use – partial image for scholarly commentary)

Game business changes VERY quickly



ft.com > companies >

Technology

Last updated: July 29, 2011 12:26 am

Nintendo cuts 3DS price amid poor sales

By Jonathan Soble in Tokyo

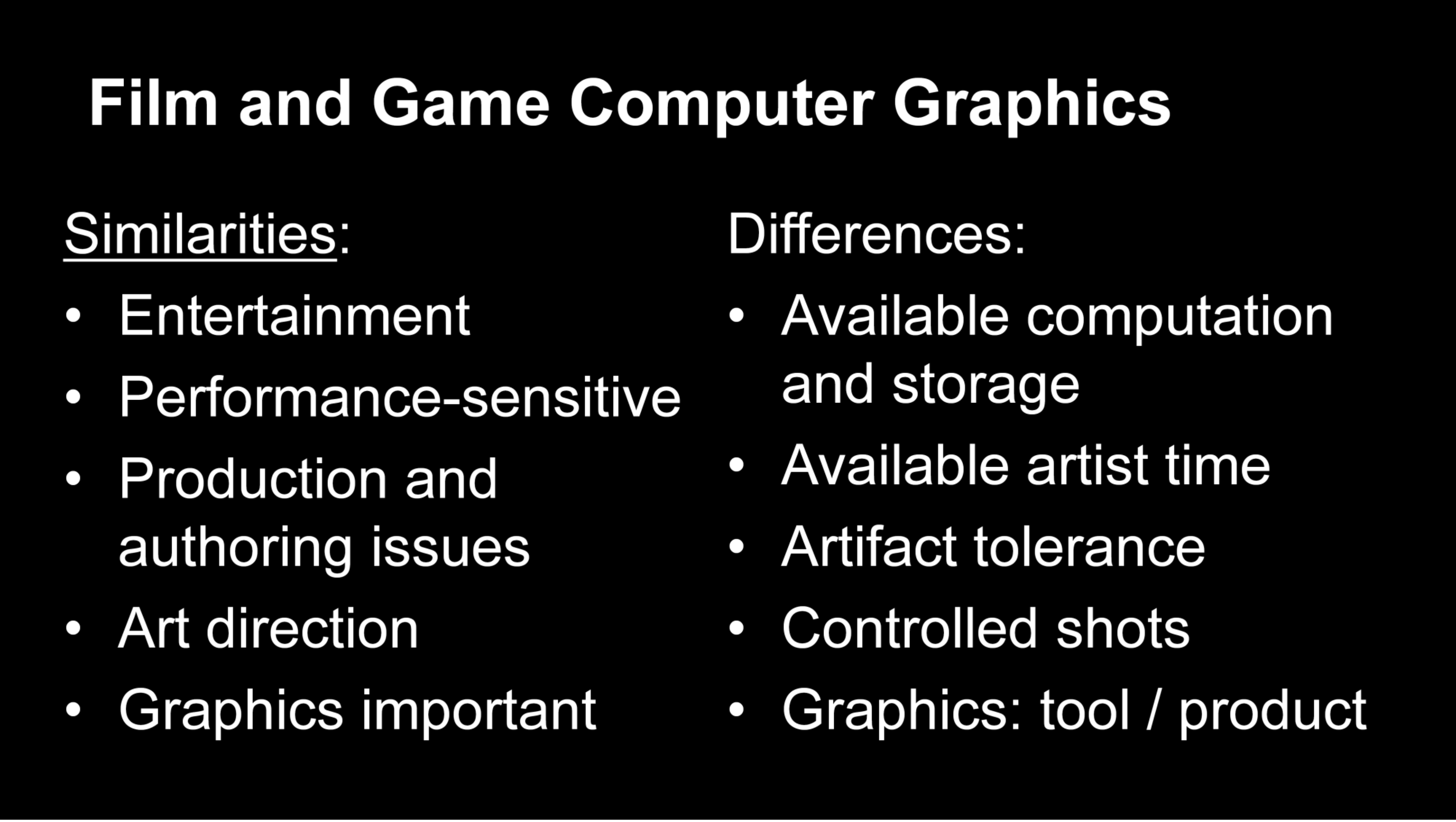
A lot of people were expecting phones and tablets to eat handheld's lunch, but I wasn't expecting it to happen this quickly. I was expecting the 3DS to be the last successful dedicated handheld game system, now it looks like that honor may have to go to the DSi.

(image from "Financial Times" website used under fair use – partial image for scholarly commentary)

Film graphics: an (imperfect) crystal ball for game graphics

US-PD: Detail from "The Crystal Ball" by John William Waterhouse

Another possible tool is to use film production (feature animation and VFX) graphics as a (often distorted) crystal ball for the future of game graphics.



Film and Game Computer Graphics

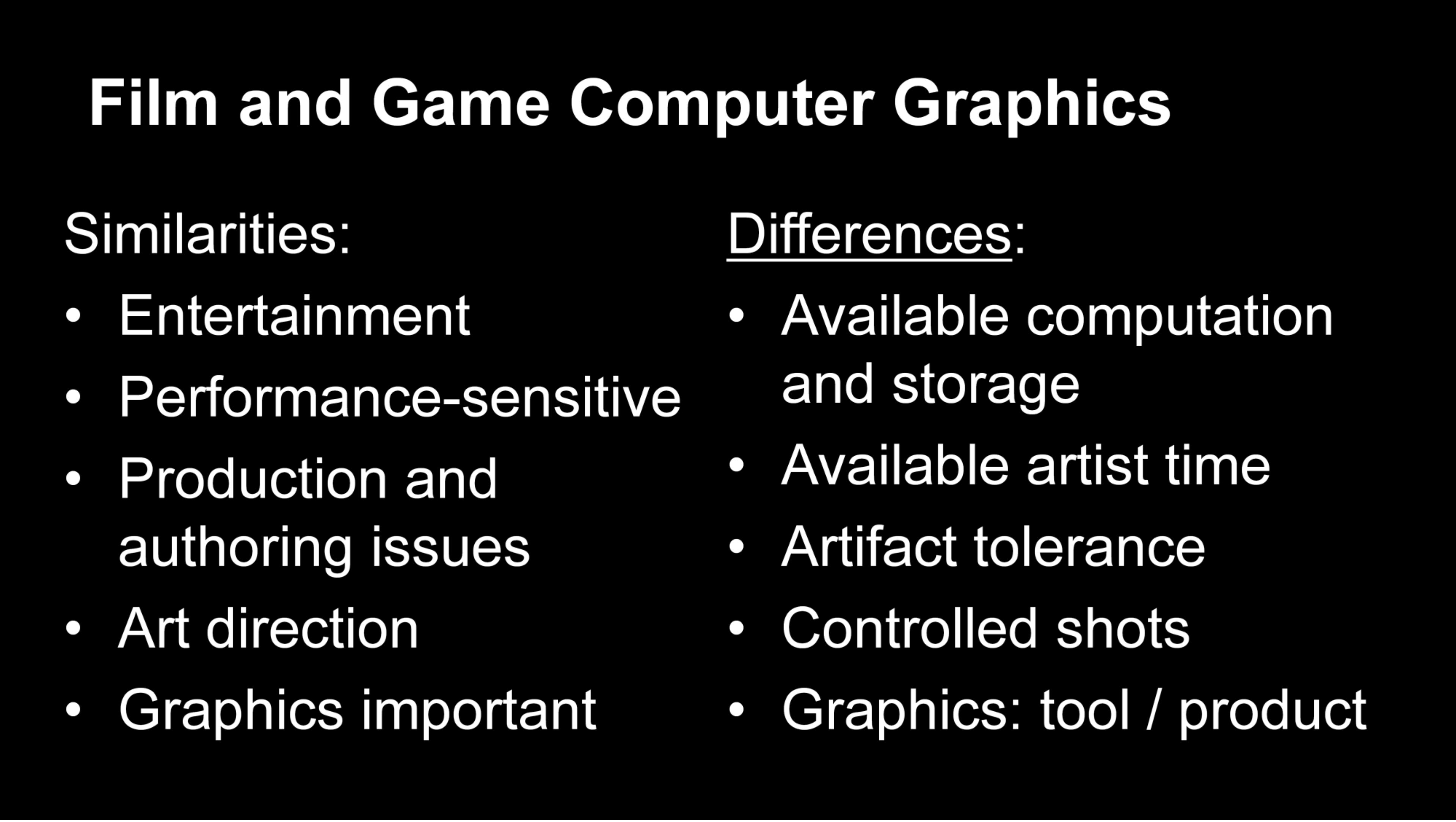
Similarities:

- Entertainment
- Performance-sensitive
- Production and authoring issues
- Art direction
- Graphics important

Differences:

- Available computation and storage
- Available artist time
- Artifact tolerance
- Controlled shots
- Graphics: tool / product

There are many interesting similarities as well as differences between the two. Both are entertainment-focused – the goal is to look compelling and serve a story, not to simulate reality – reality is too boring. Both care about performance, just at different scales. Both have to fit into the complex production pipelines with various authoring tools. And in both cases, the visuals need to be art-directed and the graphics are important for sales.



Film and Game Computer Graphics

Similarities:

- Entertainment
- Performance-sensitive
- Production and authoring issues
- Art direction
- Graphics important

Differences:

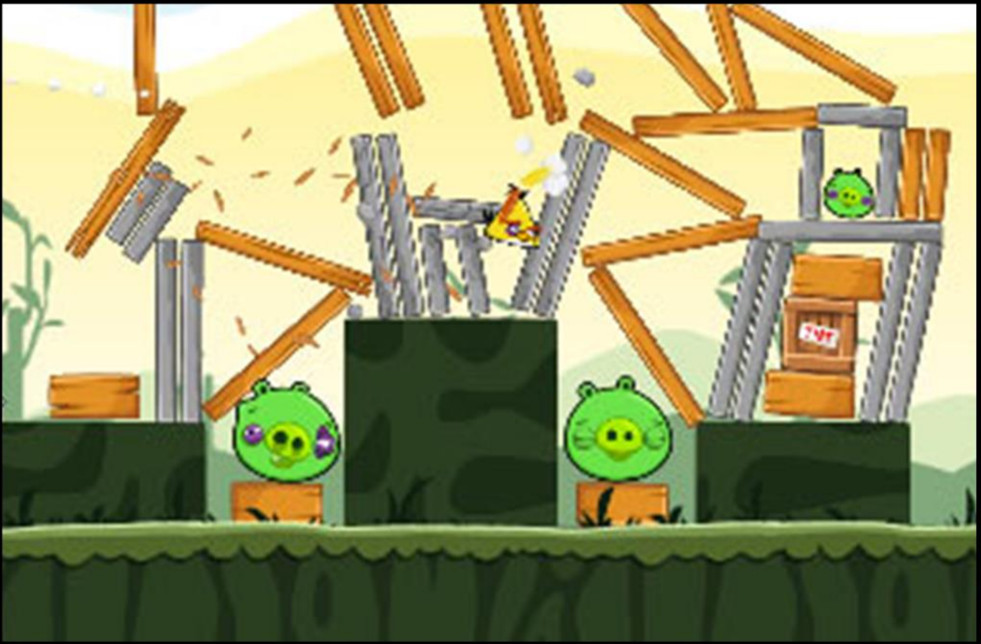
- Available computation and storage
- Available artist time
- Artifact tolerance
- Controlled shots
- Graphics: tool / product

Computation & storage resources clearly differ (hours vs. ms, GBs vs. MBs / frame). Another resource difference: artist time / frame; games have fewer artists and much more screen time. Movies have a "zero-tolerance" policy to visual artifacts, games are a lot more forgiving (by necessity and as a priority – games could have less aliasing at the cost of other visual aspects). Films have controlled shots, games have constant (often unpredictable) camera, lighting & scene changes. The role of animation & rendering tech is profoundly different – in film, tools used to create the final product; in games part of the final product.

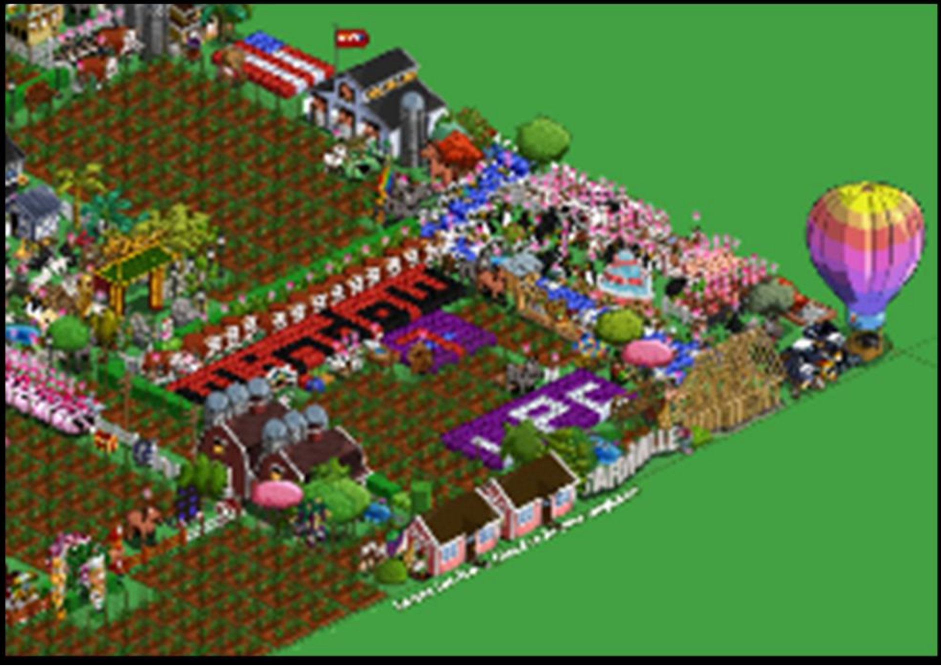
IS there a future for videogame graphics?

?

2009 – Angry Birds



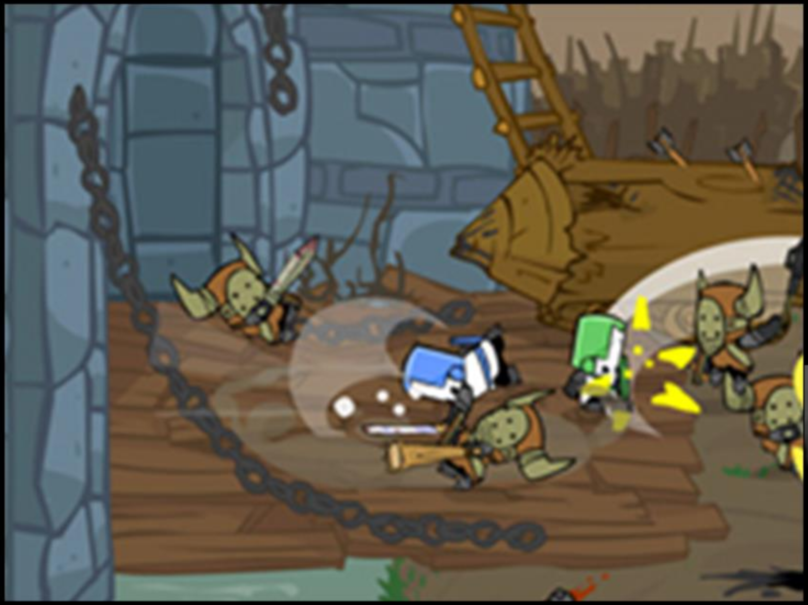
2009 – FarmVille



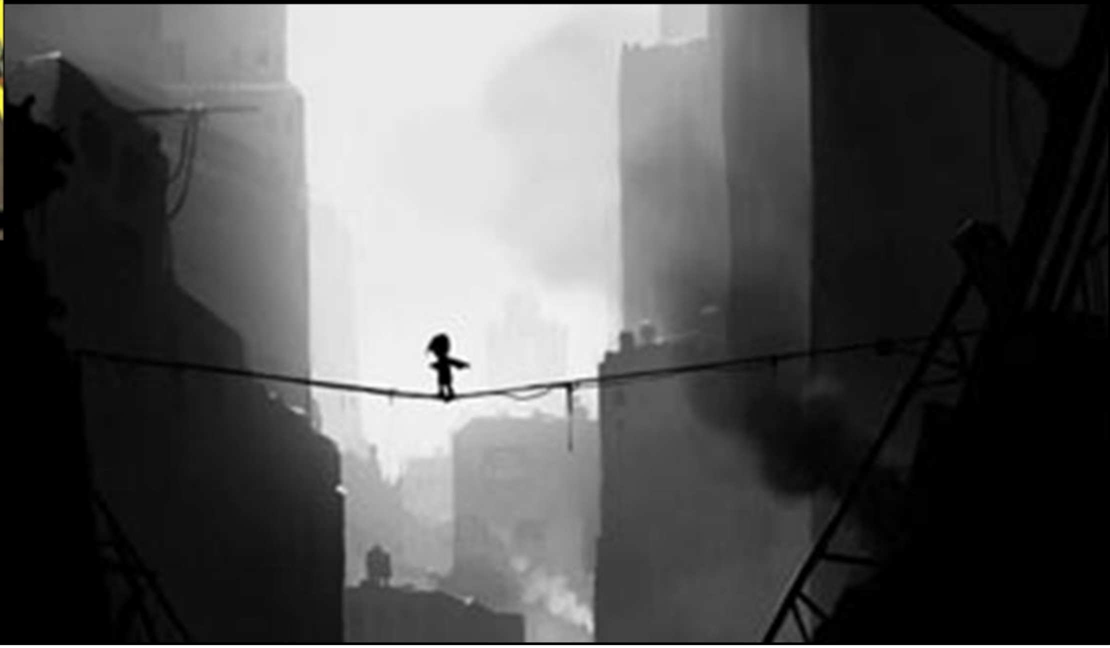
The big new markets are handhelds and social games, and most of those – definitely the most successful ones – use 2D hand-drawn graphics.

(FarmVille and Angry Birds screenshots used under fair use – low-res images for scholarly commentary)

2008 – Castle Crashers

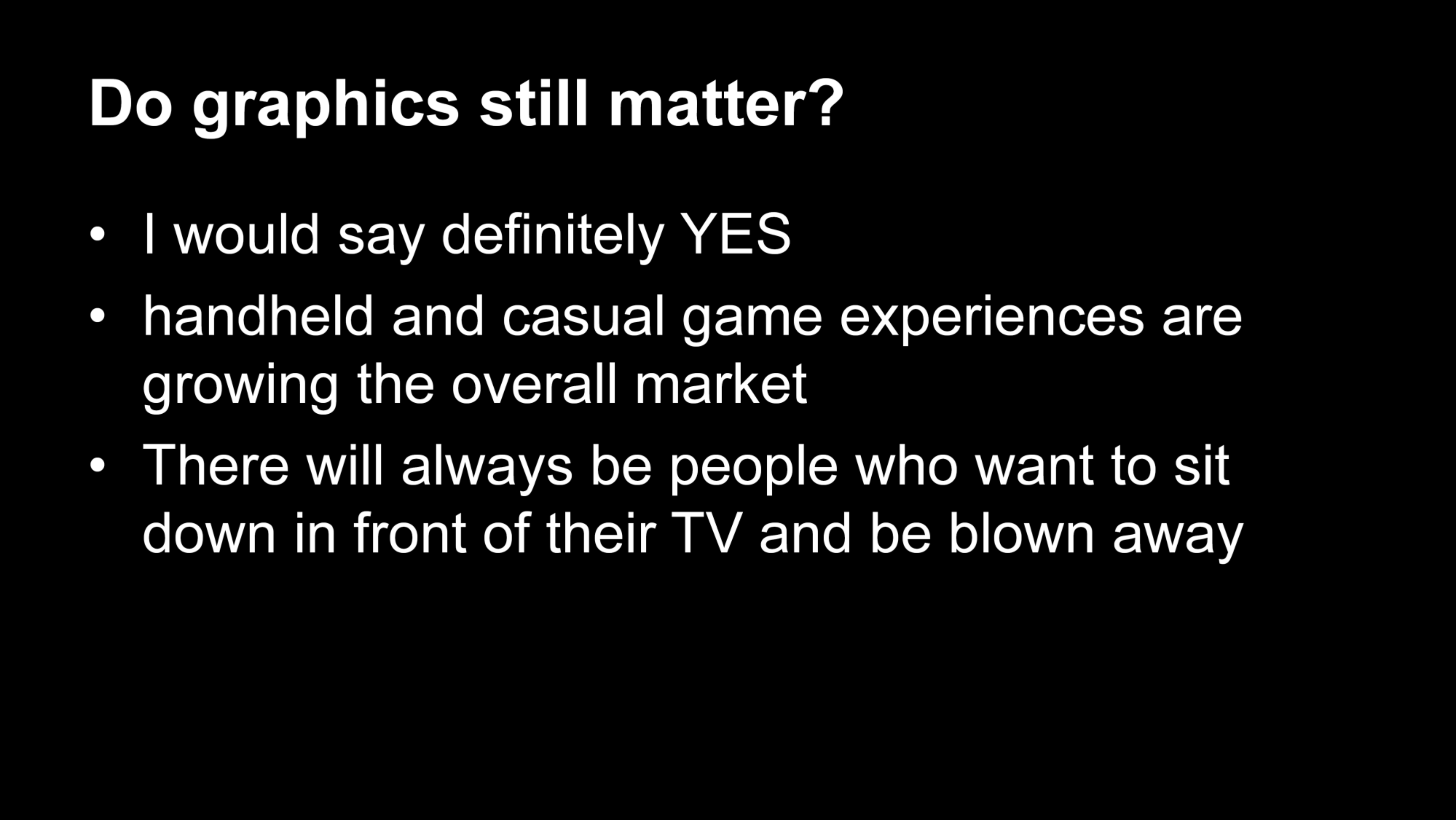


2010 - Limbo



And it’s not just the big commercial successes – a lot of the critically acclaimed indie downloadable games are 2D hand-drawn as well.

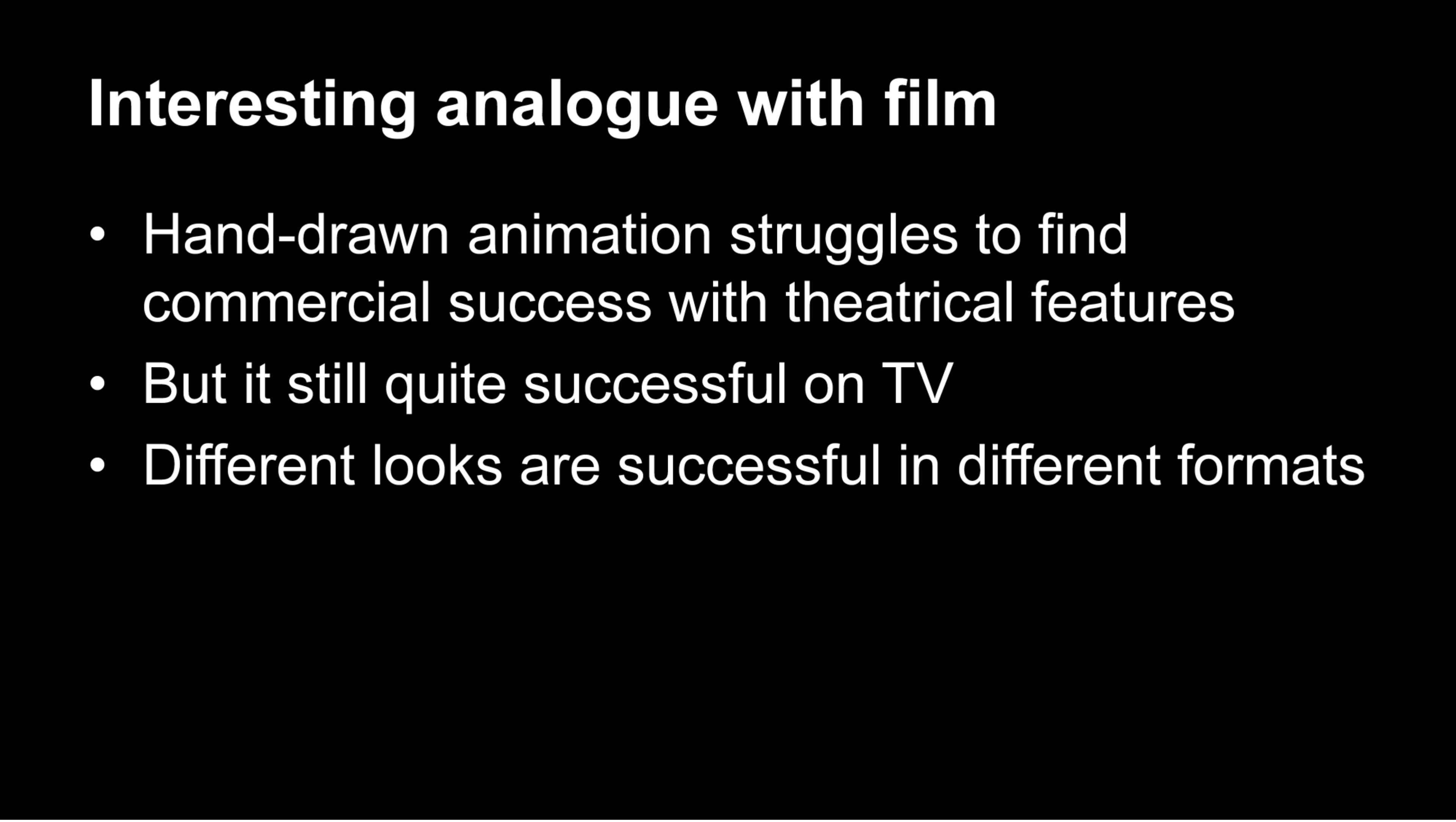
(Castle Crashers and Limbo screenshots used under fair use – low-res images for scholarly commentary)



Do graphics still matter?

- I would say definitely YES
- handheld and casual game experiences are growing the overall market
- There will always be people who want to sit down in front of their TV and be blown away

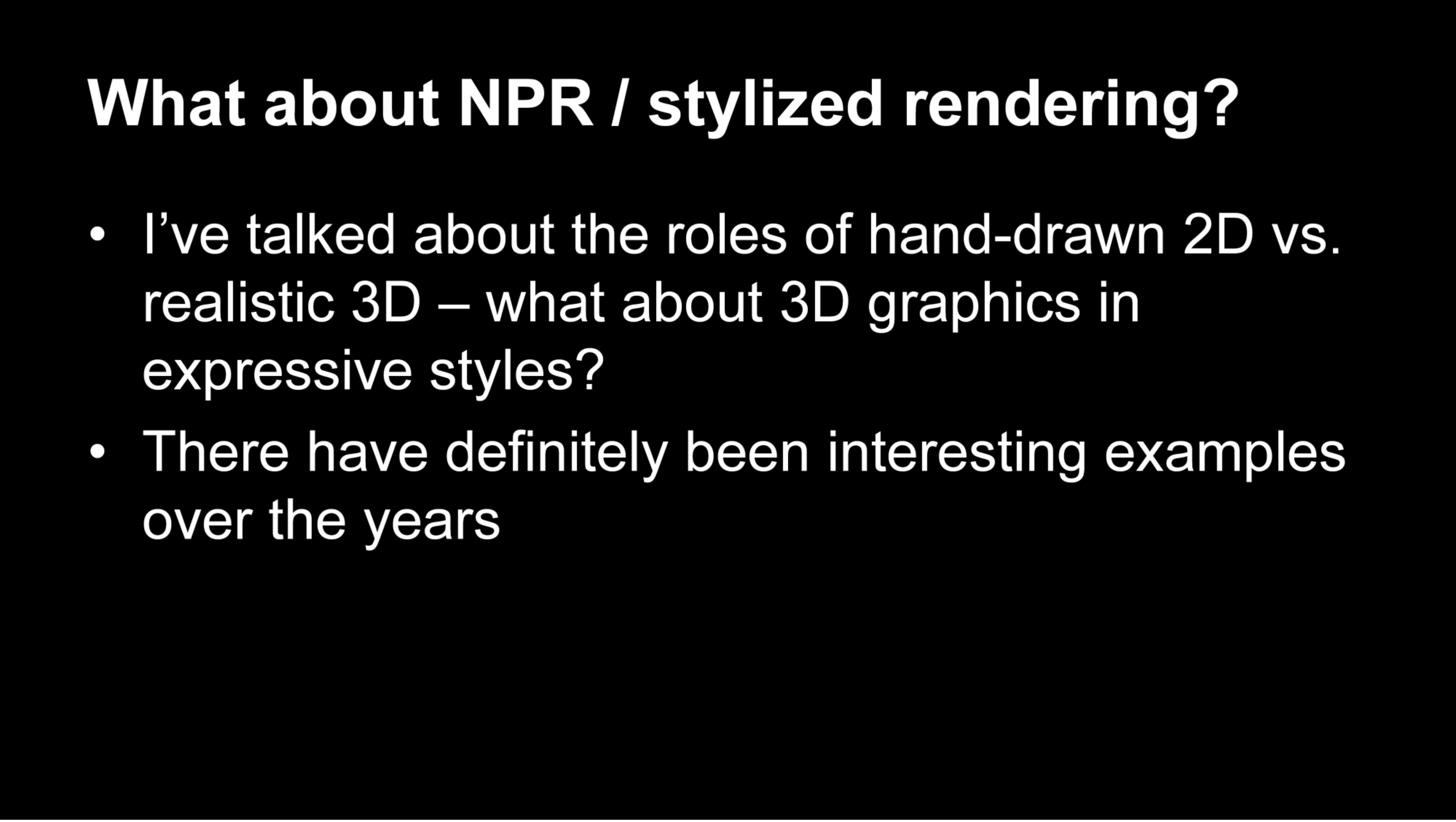
It’s true that a growing percentage of game time is being spent with game experiences (such as handheld, downloadable and browser games) which do not rely on cutting-edge graphics, but a lot of these are new audiences. I strongly believe there will always be a strong market for people sitting down in front of their TV for interactive experiences with amazing graphics.



Interesting analogue with film

- Hand-drawn animation struggles to find commercial success with theatrical features
- But it still quite successful on TV
- Different looks are successful in different formats

“The Princess and The Frog” and “Winnie the Pooh” recently struggled to find audiences, but most successful animated shows on TV are still hand-drawn. There’s a sweet spot in terms of the amount of effort required and audience expectation that I think is similar to the “hand-drawn sprites on small screen” vs. “realistic graphics on big screen” games situation.

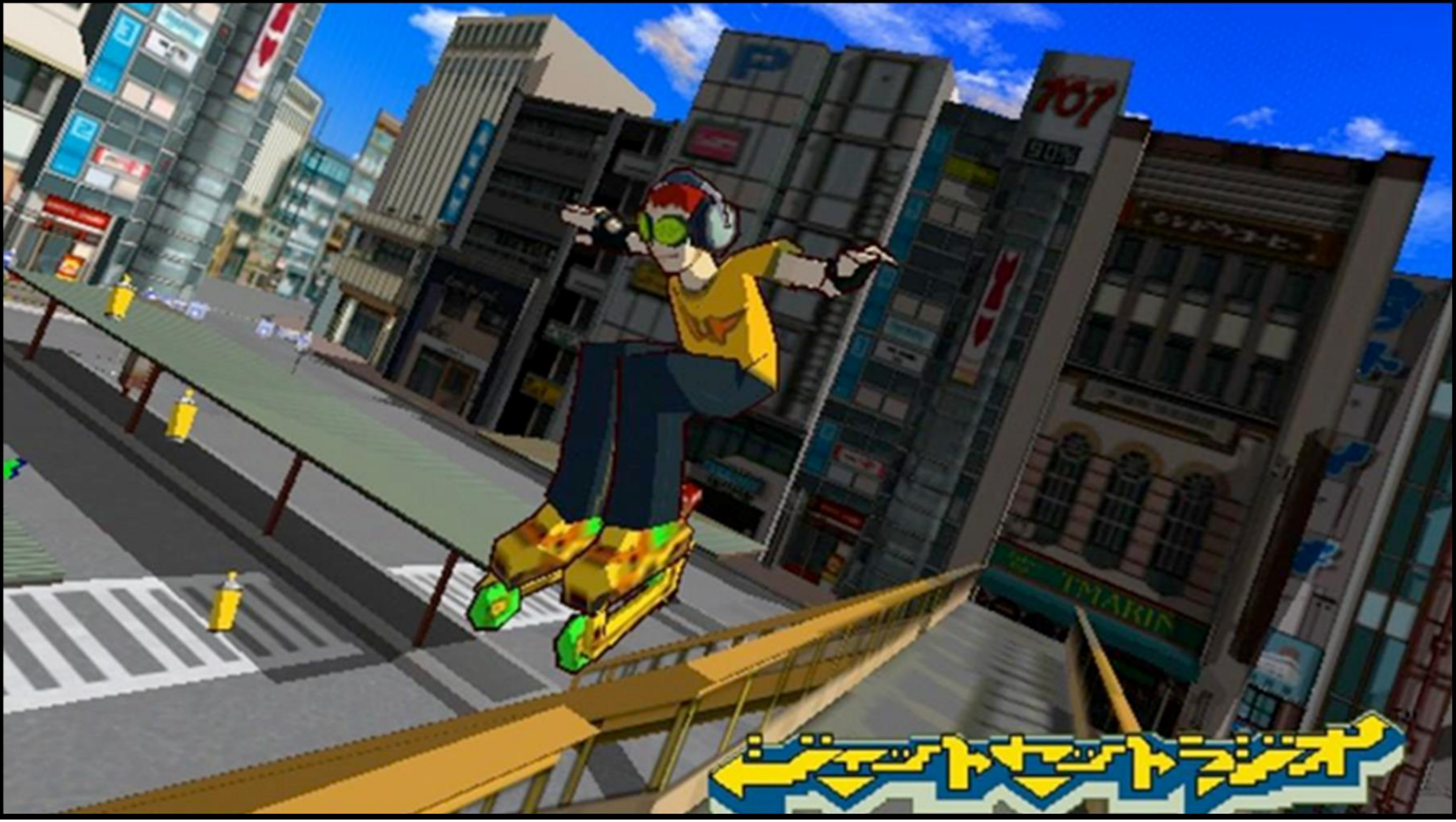


What about NPR / stylized rendering?

- I've talked about the roles of hand-drawn 2D vs. realistic 3D – what about 3D graphics in expressive styles?
- There have definitely been interesting examples over the years

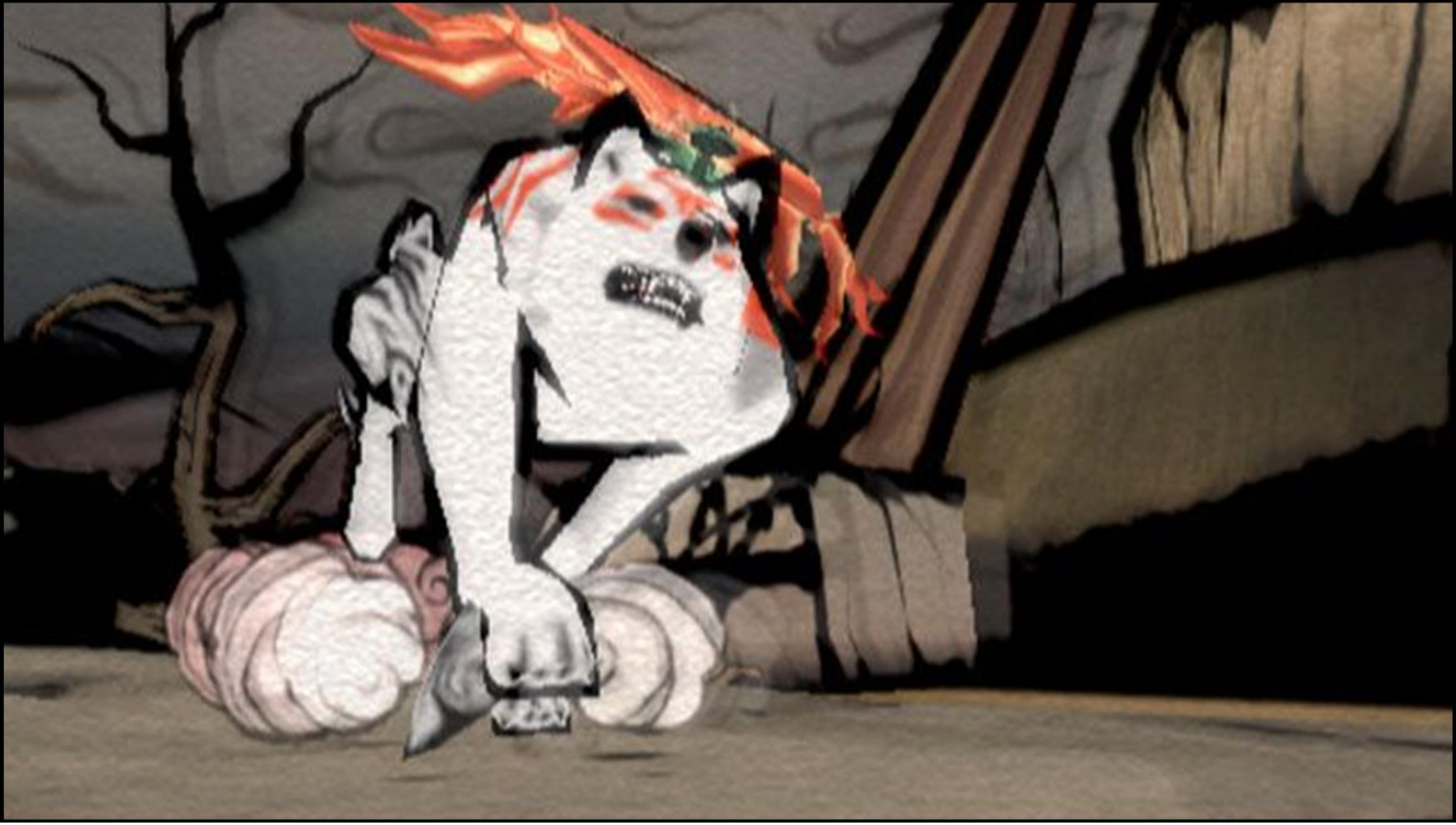
Last year there was a course at SIGGRAPH about this.

2000 – Jet Grind Radio



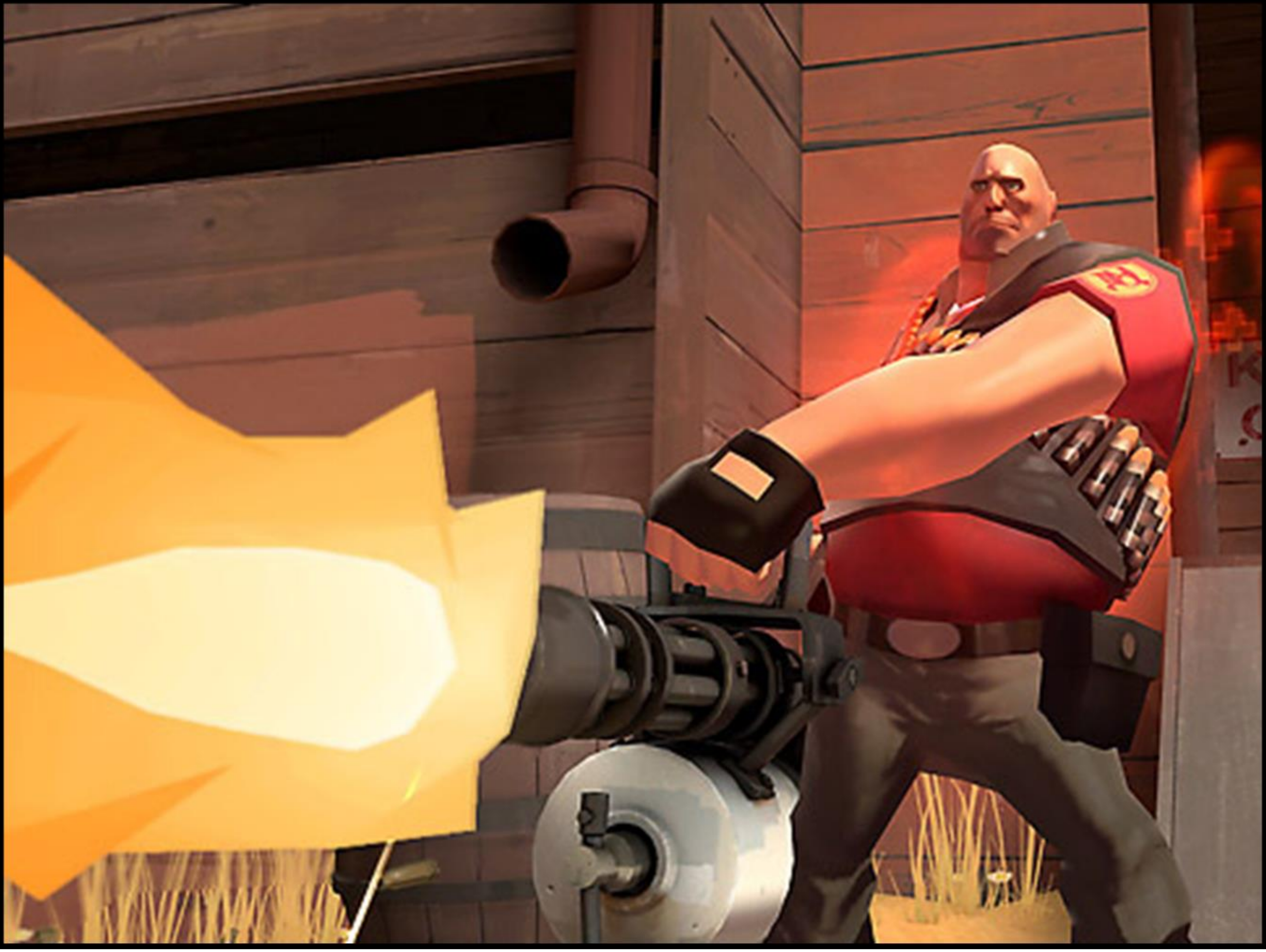
(Jet Grind Radio screenshot used under fair use – partial image for scholarly commentary)

2006 – Okami



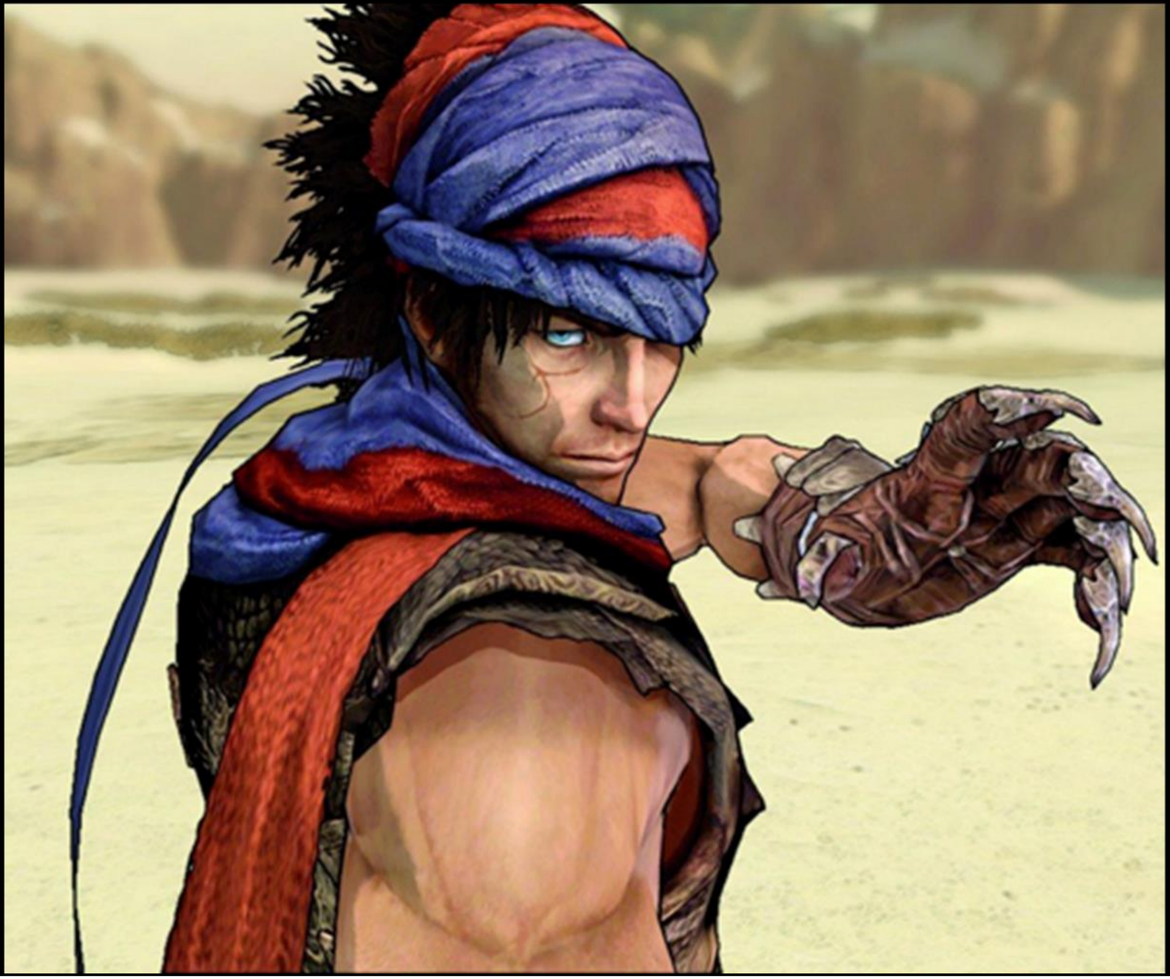
(Okami screenshot used under fair use – partial image for scholarly commentary)

2007 – Team Fortress 2



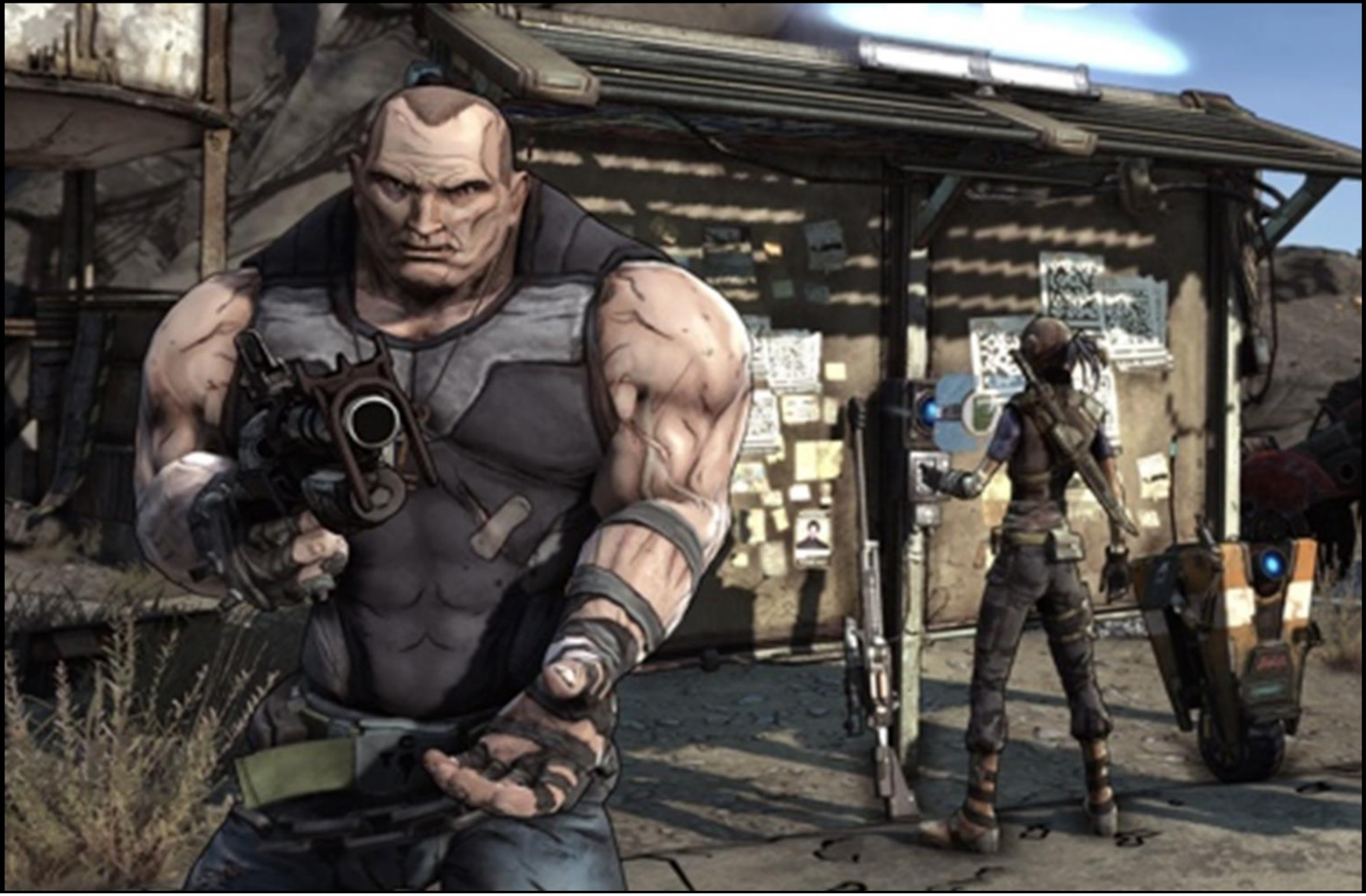
(Team Fortress 2 screenshot used under fair use – partial image for scholarly commentary)

2008 – Prince of Persia



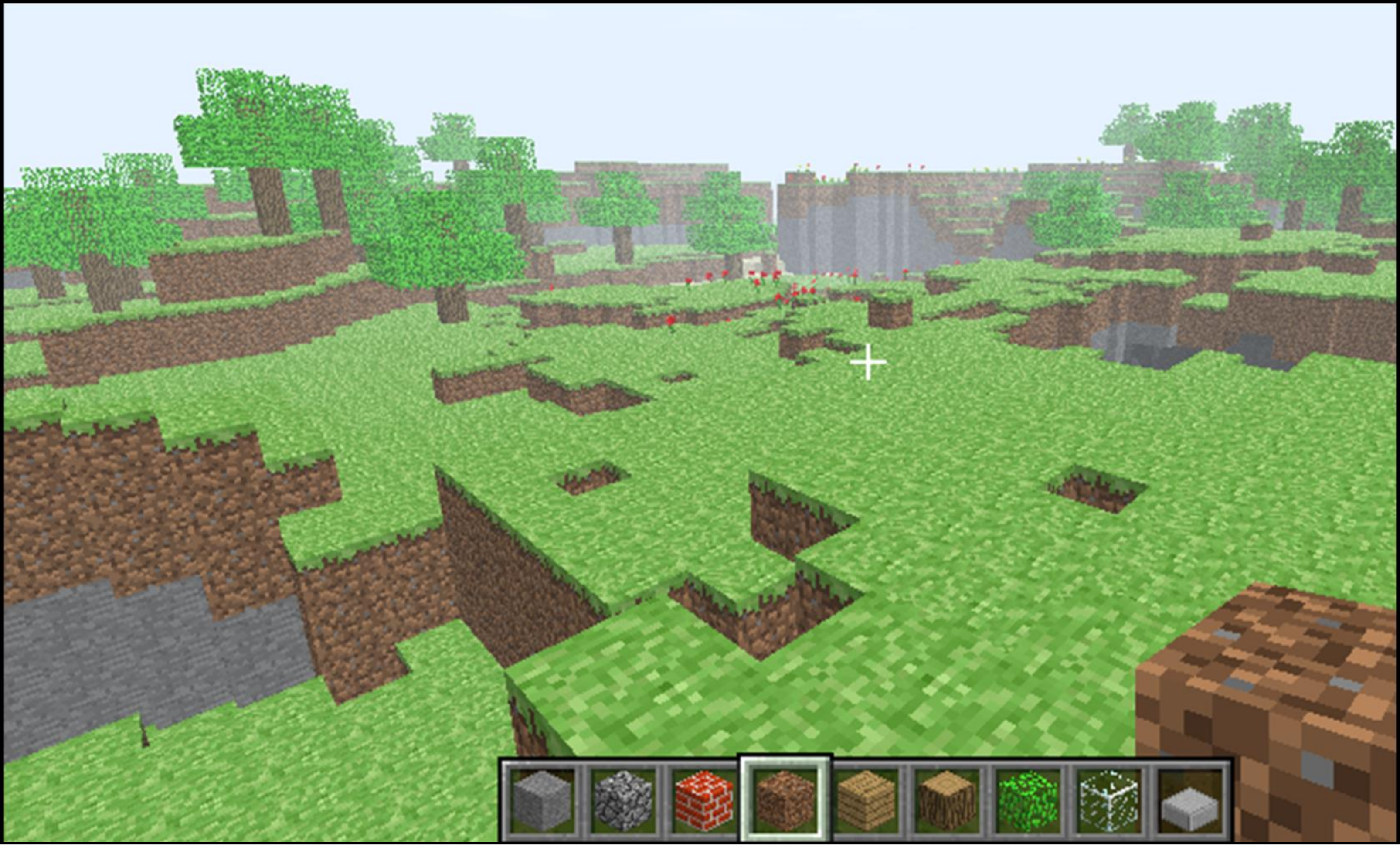
(Prince of Persia screenshot used under fair use – partial image for scholarly commentary)

2009 – Borderlands

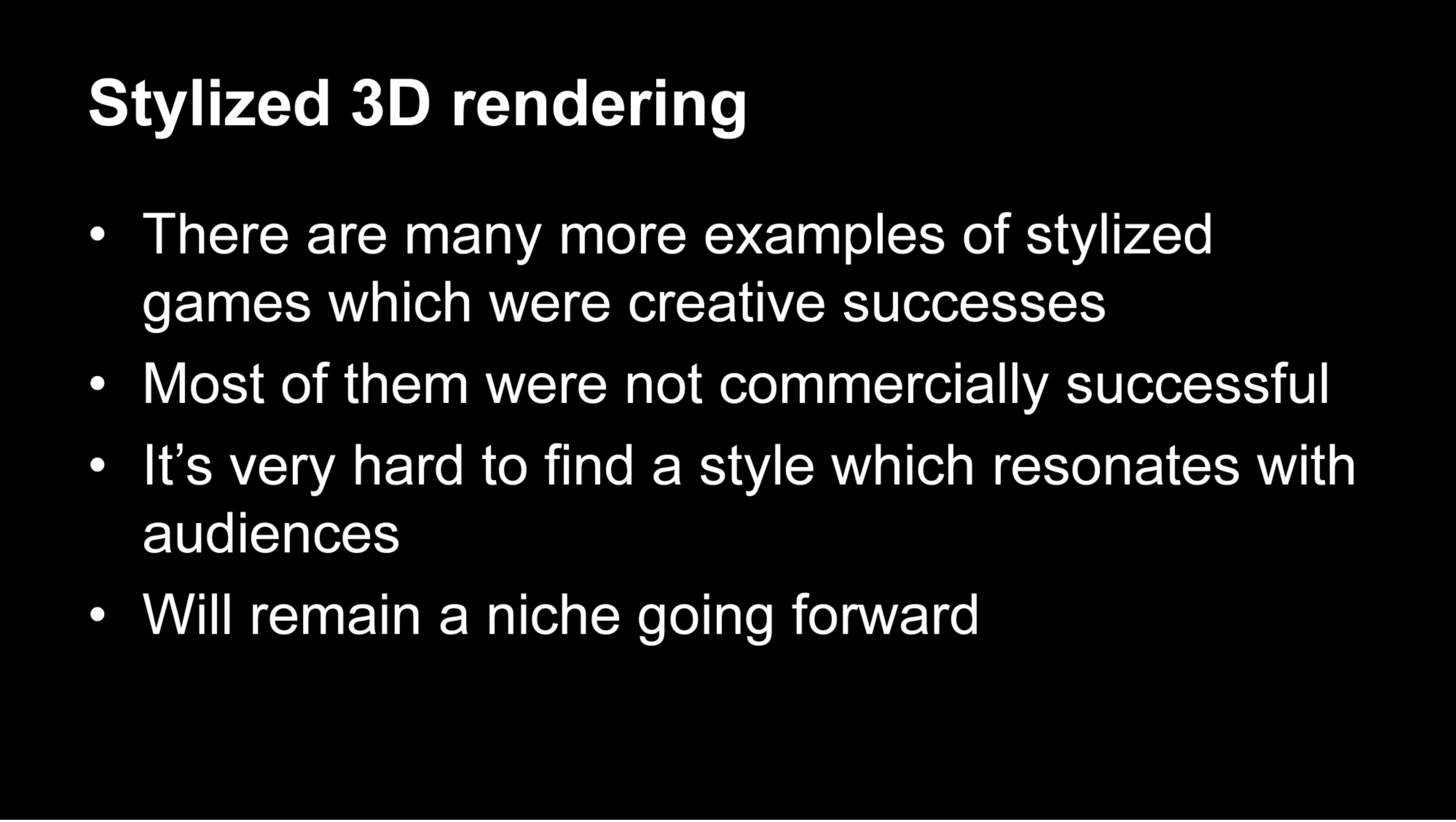


(Borderlands screenshot used under fair use – partial image for scholarly commentary)

2009 – Minecraft



(Minecraft screenshot used under fair use – partial image for scholarly commentary)

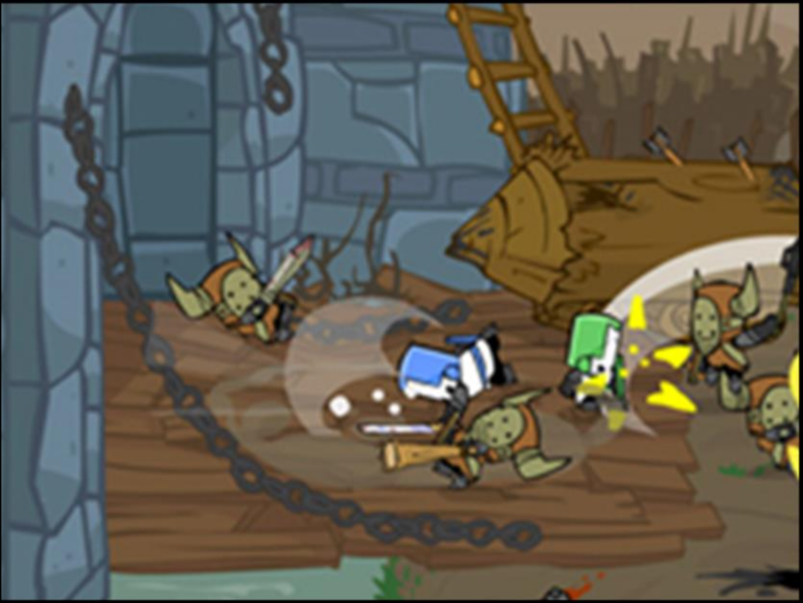


Stylized 3D rendering

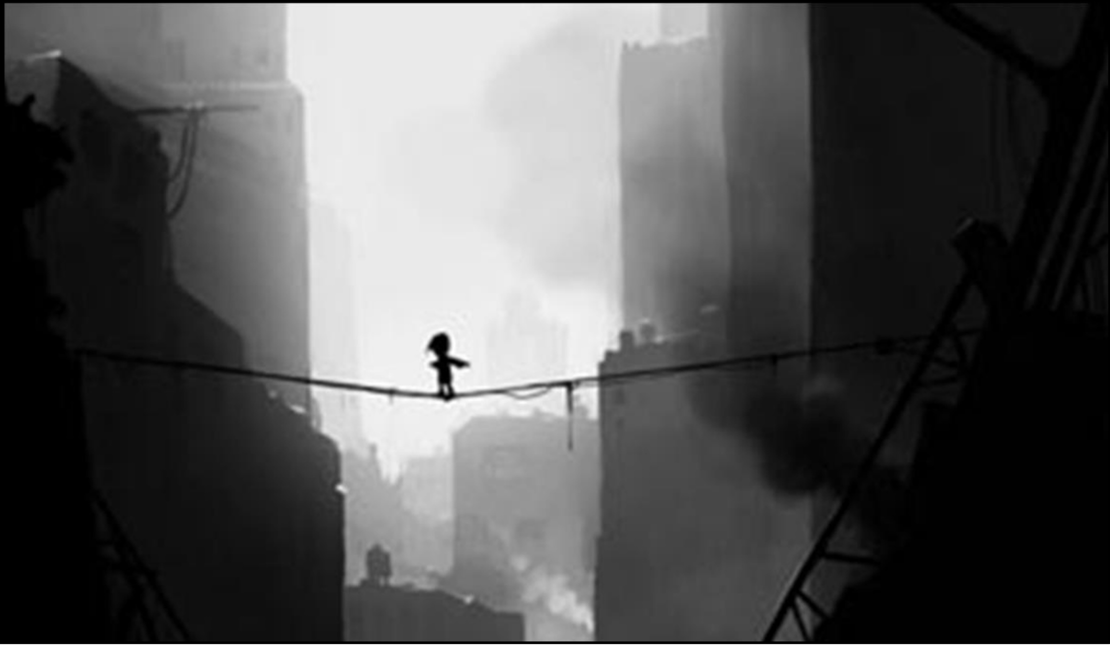
- There are many more examples of stylized games which were creative successes
- Most of them were not commercially successful
- It’s very hard to find a style which resonates with audiences
- Will remain a niche going forward

I think it’s a big risk – the style needs to really fit the kind of experience you are creating, and to resonate with audiences. I think it will be a niche, not mainstream – a handful of games will do amazing and interesting things with this, and some of them will find audiences.

2008 – Castle Crashers

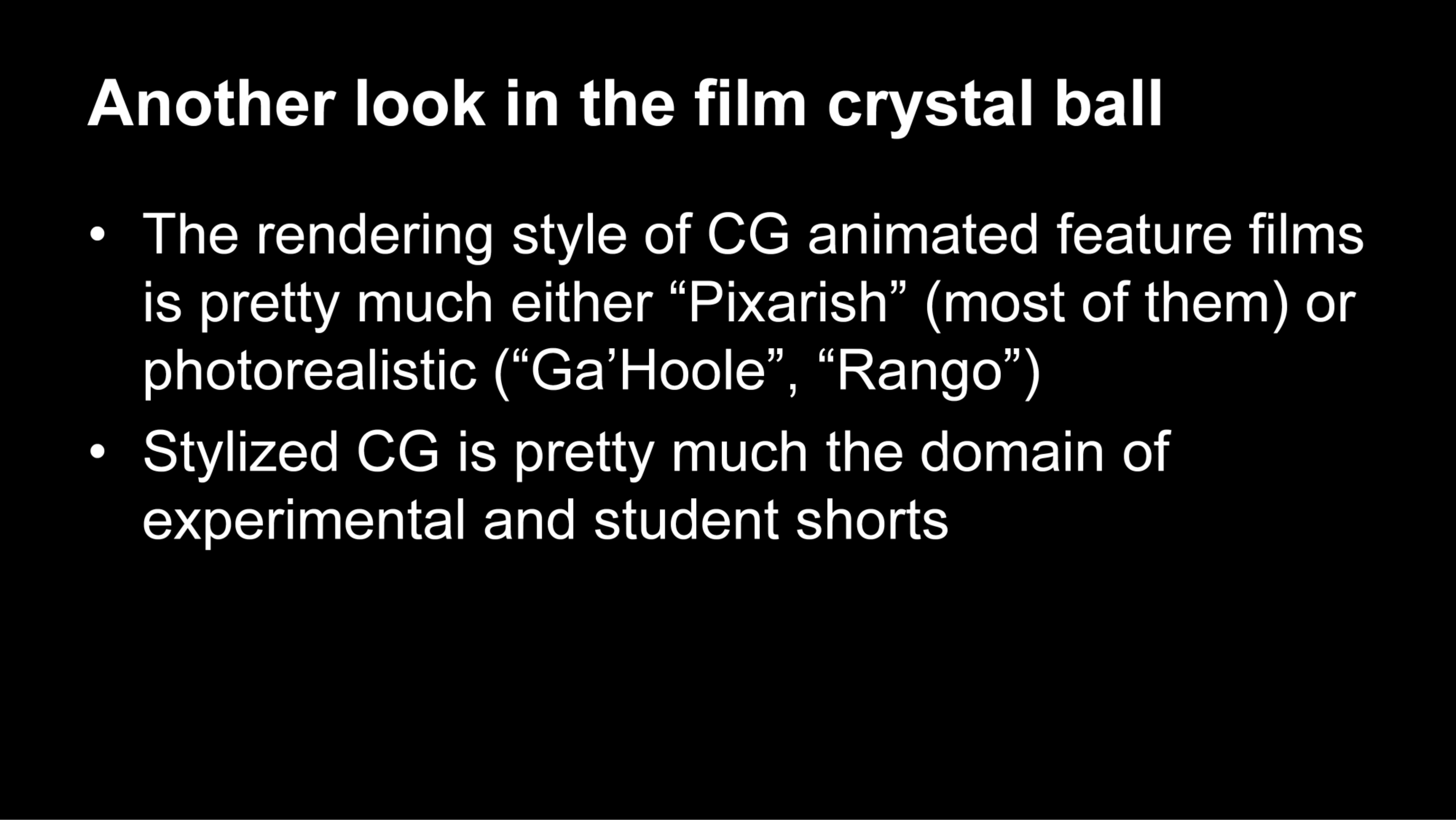


2010 - Limbo



2D games can succeed while pushing much more extreme styles- I think this is just a more natural thing for 2D, where the artists have a lot more control over the final image.

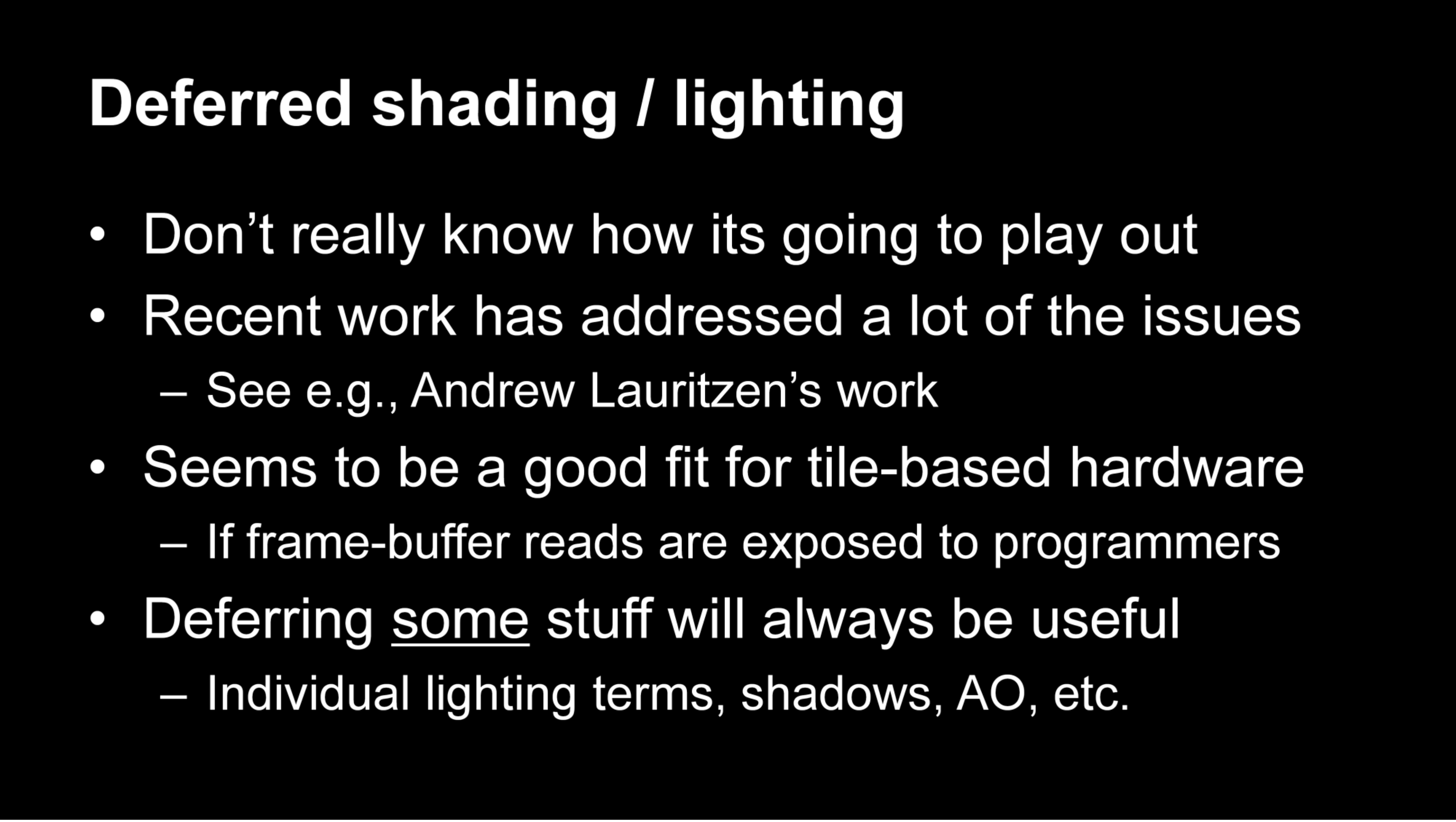
(Castle Crashers and Limbo screenshots used under fair use – low-res images for scholarly commentary)



Another look in the film crystal ball

- The rendering style of CG animated feature films is pretty much either “Pixarish” (most of them) or photorealistic (“Ga’Hoole”, “Rango”)
- Stylized CG is pretty much the domain of experimental and student shorts

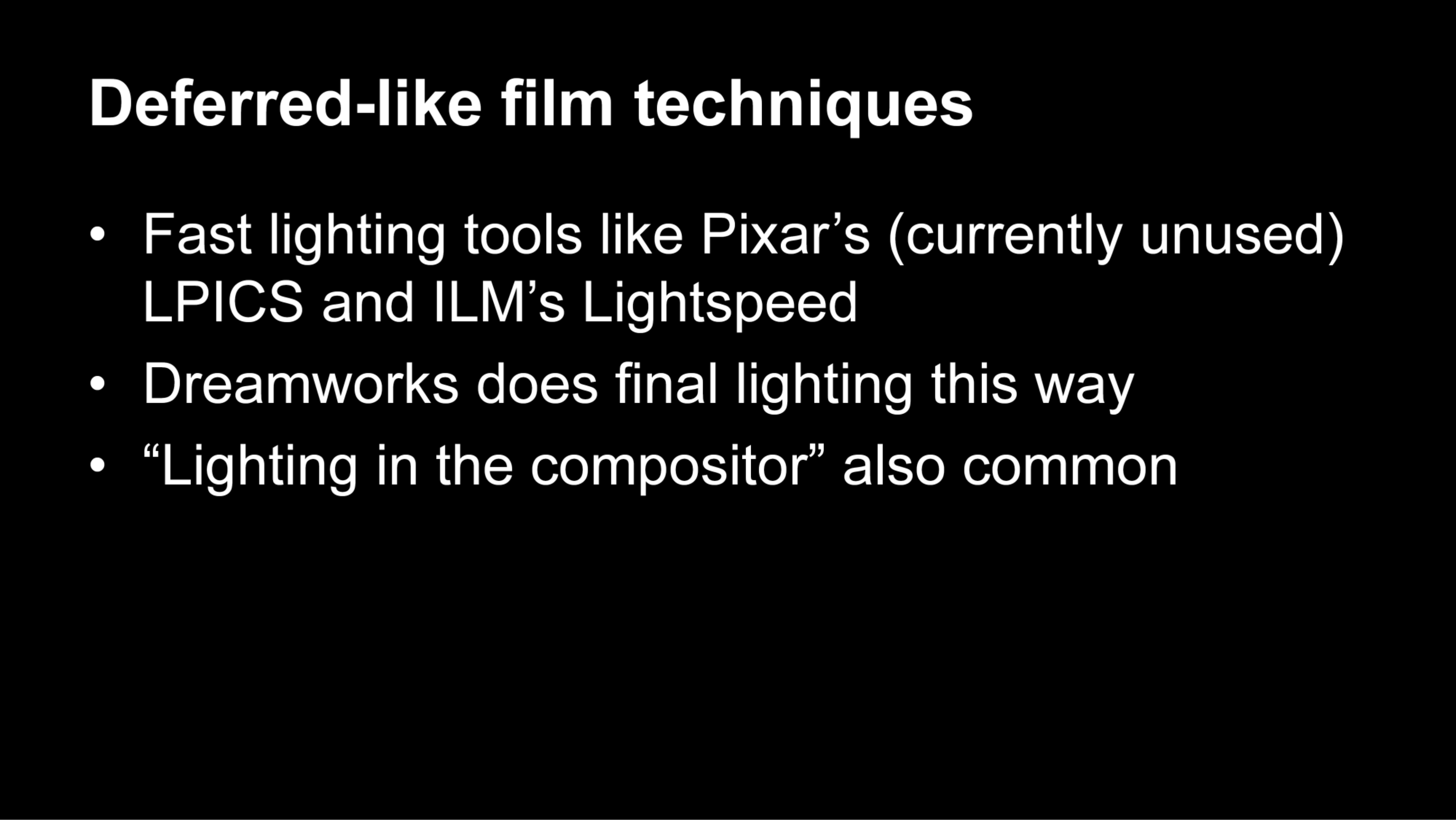
The experiences of the film people corroborates my feeling that this is very hard to get right. Note that I’m talking about rendering style here – CG animated films vary widely in visual styles expressed via character and set modeling, lighting, textures, animation, etc.



Deferred shading / lighting

- Don’t really know how its going to play out
- Recent work has addressed a lot of the issues
  - See e.g., Andrew Lauritzen’s work
- Seems to be a good fit for tile-based hardware
  - If frame-buffer reads are exposed to programmers
- Deferring some stuff will always be useful
  - Individual lighting terms, shadows, AO, etc.

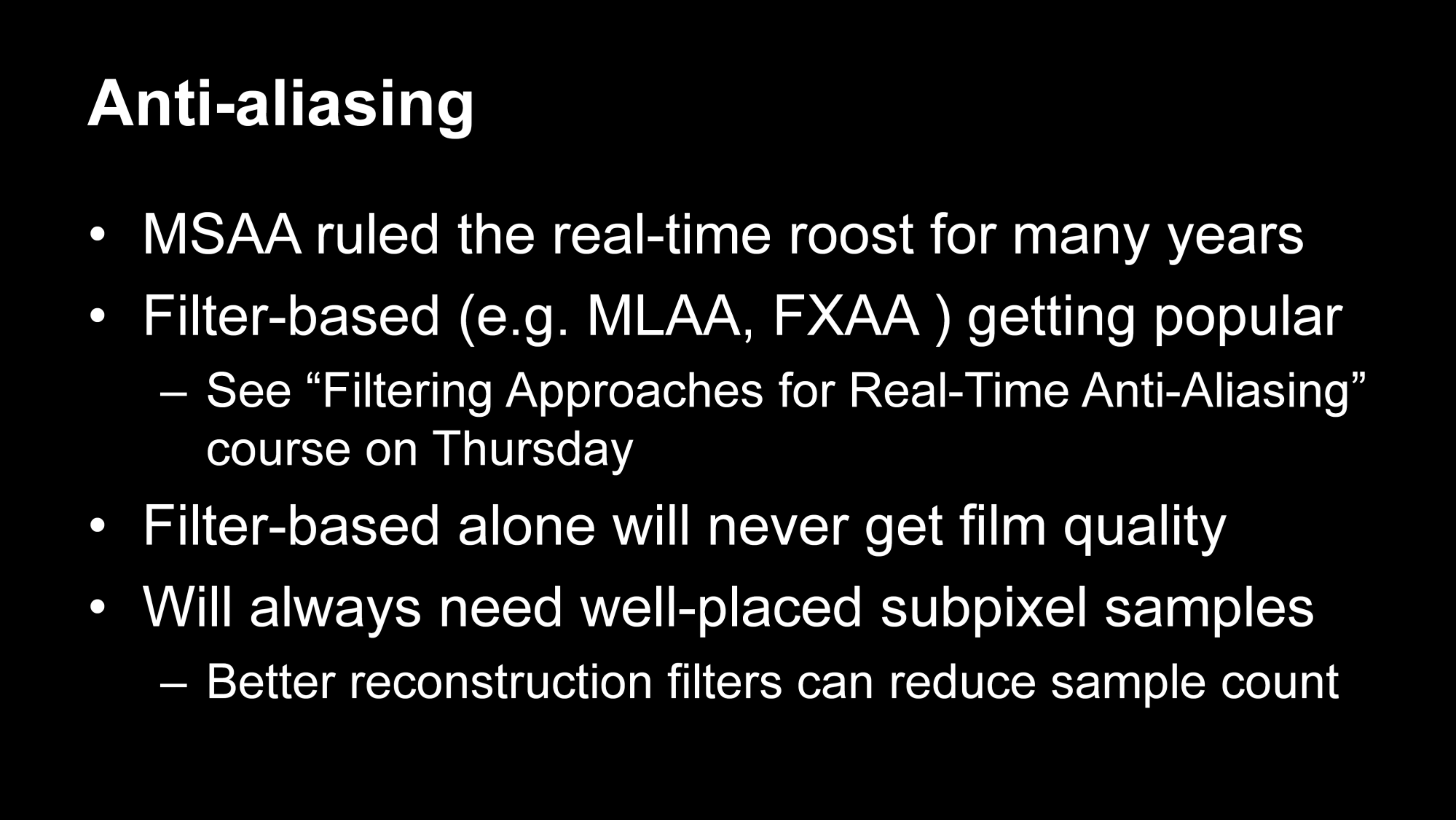
I thought about this one quite a bit, and talked to a lot of people – I’m still not sure how it’s going to play out long term.



Deferred-like film techniques

- Fast lighting tools like Pixar’s (currently unused) LPICS and ILM’s Lightspeed
- Dreamworks does final lighting this way
- “Lighting in the compositor” also common

It’s interesting to also look at rendering techniques used in film production which are similar to deferred shading. Lightspeed is especially interesting since it handles transparency, motion blur, depth of field, antialiasing, etc., by using an indirect G-buffer (not unlike an A-buffer).



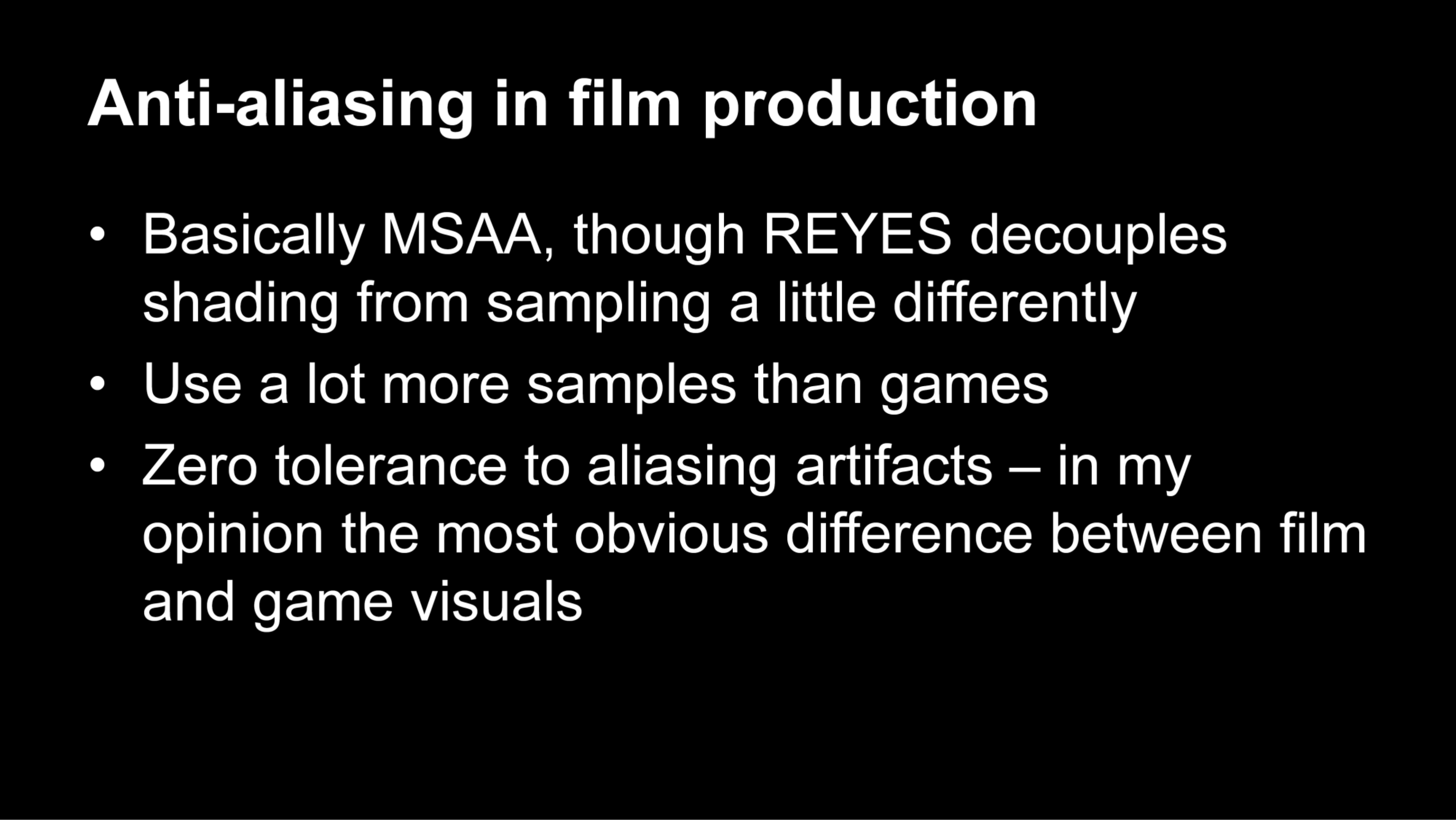
Anti-aliasing

- MSAA ruled the real-time roost for many years
- Filter-based (e.g. MLAA, FXAA) getting popular
  – See “Filtering Approaches for Real-Time Anti-Aliasing” course on Thursday
- Filter-based alone will never get film quality
- Will always need well-placed subpixel samples
  – Better reconstruction filters can reduce sample count

I think the future is a hybrid method, where you have some number of subpixel samples and reconstruct smartly – there was a paper in HPG 2009 called “A Directionally Adaptive Edge Anti-Aliasing Filter”, and there is some more recent research (SRAA at I3D 2011, and SMAA – presented at the Thursday course as well as a technical report by the University of Zaragoza). I think this is still a fertile area of research, and will most likely lead to techniques which will be commonly used by both film and games in future years.

Part of a larger trend

- “Be smart about what you do with your samples”
- Applies to reducing noise, not just anti-aliasing
  - E.g. I3D 2011 paper “A Local Image Reconstruction Algorithm for Stochastic Rendering”



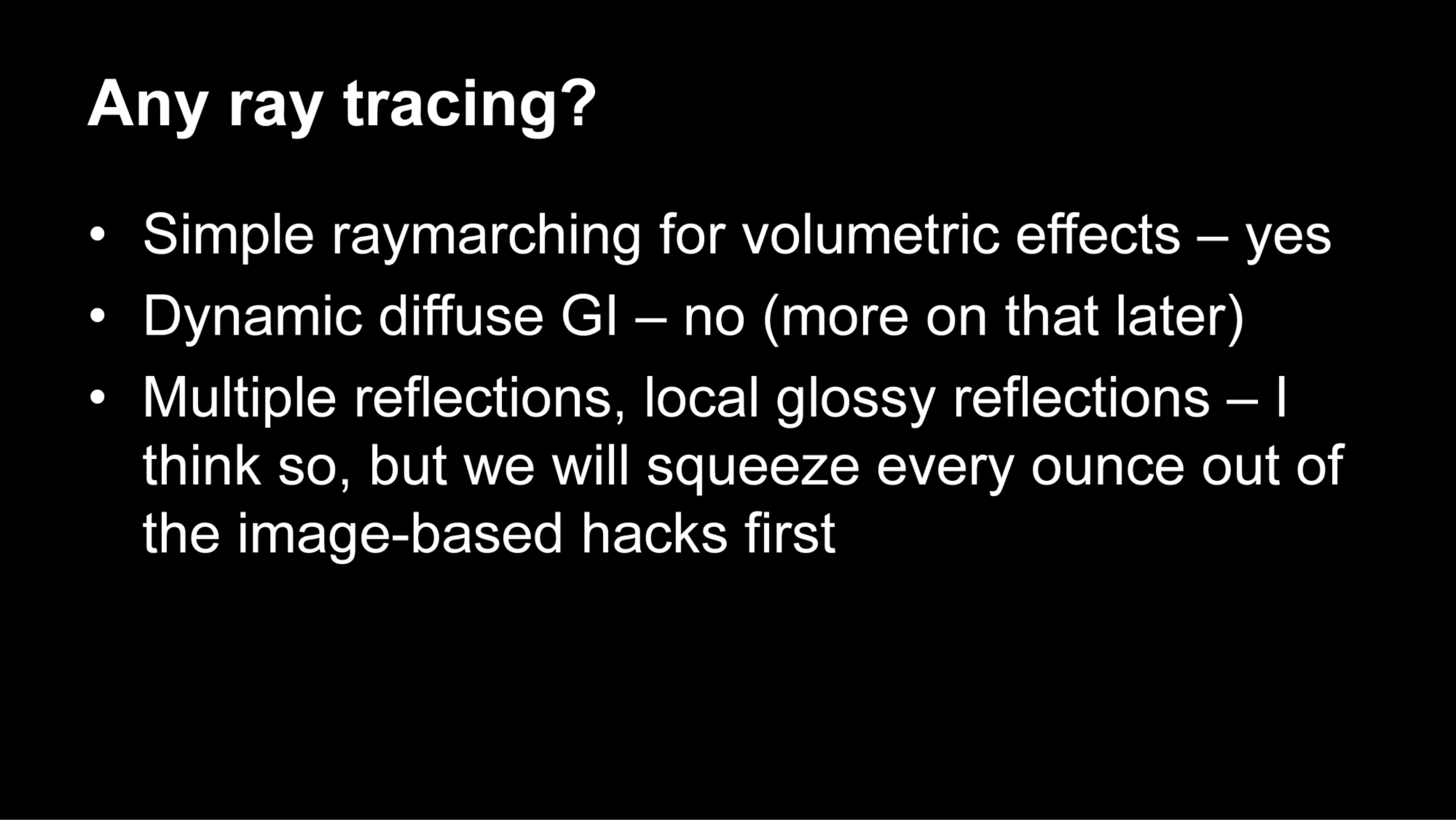
Anti-aliasing in film production

- Basically MSAA, though REYES decouples shading from sampling a little differently
- Use a lot more samples than games
- Zero tolerance to aliasing artifacts – in my opinion the most obvious difference between film and game visuals

We need to get a lot more serious about aliasing in games. It’s true that we have a lot less to work with, but we typically prioritize other things over aliasing. Hopefully with faster platforms we can do a better job. Here I’ve only talked about spatial geometric aliasing – there are other kinds of aliasing, each with different solutions: temporal, shading, etc.

Ray-Tracing vs. Rasterization

- A perennially popular topic.
- Games have gotten a lot of mileage out of rasterization with shadow maps, environment maps, dynamic reflection maps, etc.
- Two questions:
  - Will there be any ray tracing in games at all?
  - Will it completely replace rasterization?



Any ray tracing?

- Simple raymarching for volumetric effects – yes
- Dynamic diffuse GI – no (more on that later)
- Multiple reflections, local glossy reflections – I think so, but we will squeeze every ounce out of the image-based hacks first

Some games are already doing simple raymarching for certain volumetric effects. That will definitely continue and expand. I’ll talk a bit later about dynamic diffuse bounces, but in a nutshell, there are way more efficient ways to get plausible soft GI bounce for games. I do think that eventually, selective use will be made of ray tracing for effects that are very difficult to get otherwise.

Only ray tracing?

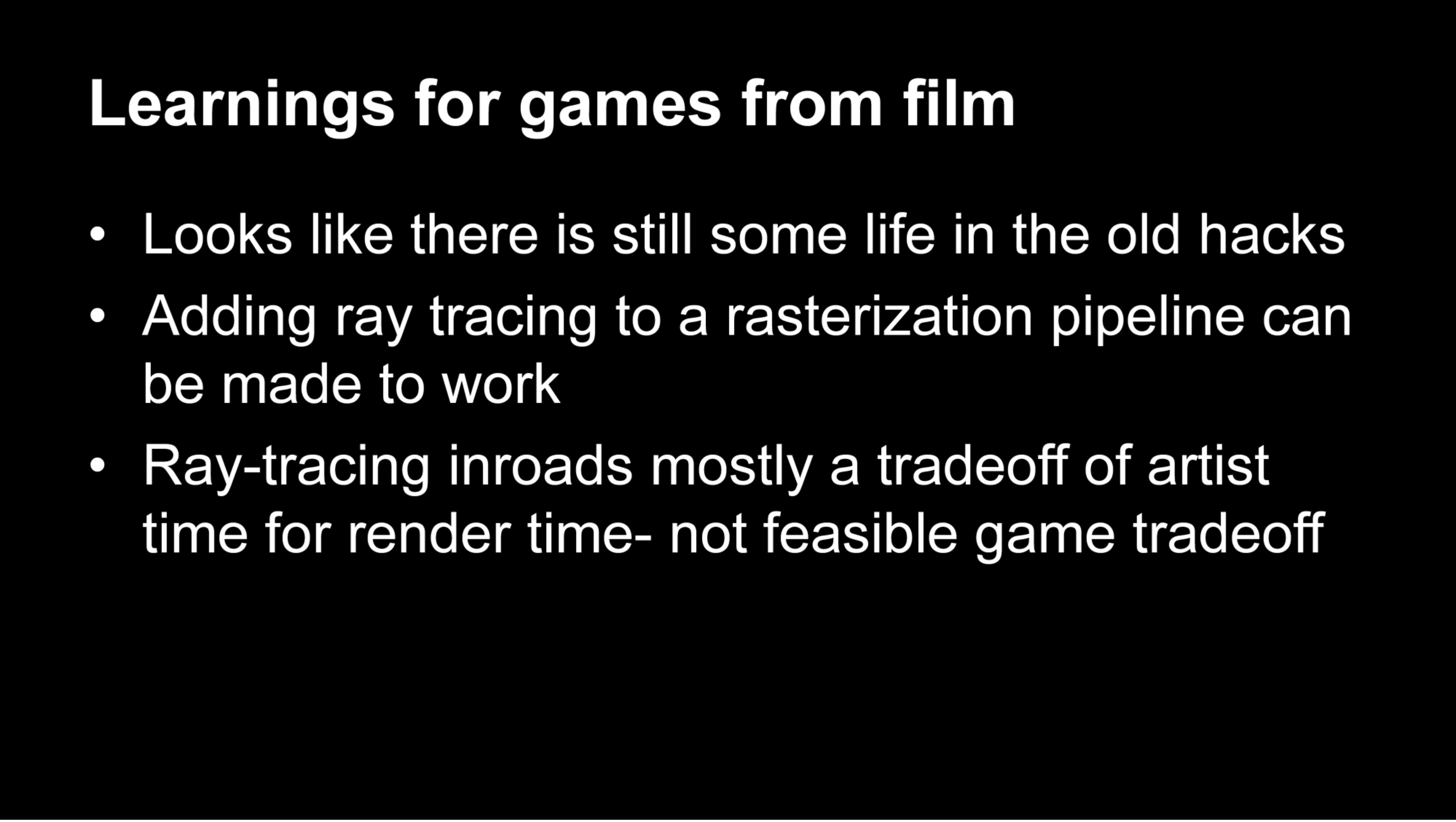
- Rasterization basically a very efficient algorithm/data structure to get primary ray hits
- Power-efficient, fast, highly evolved HW
- Can still use high-level optimizations
  - E.g. “lazy evaluation of complex / procedural geometry” ⇔ “render BVol w. occlusion query & skip”

But

- We want high-quality and robust motion and defocus blur
- Rasterization HW can be generalized, but when these effects are applied heavily no longer clear if it is a win over ray tracing
- Still research to be done in this area – jury is out

What about film?

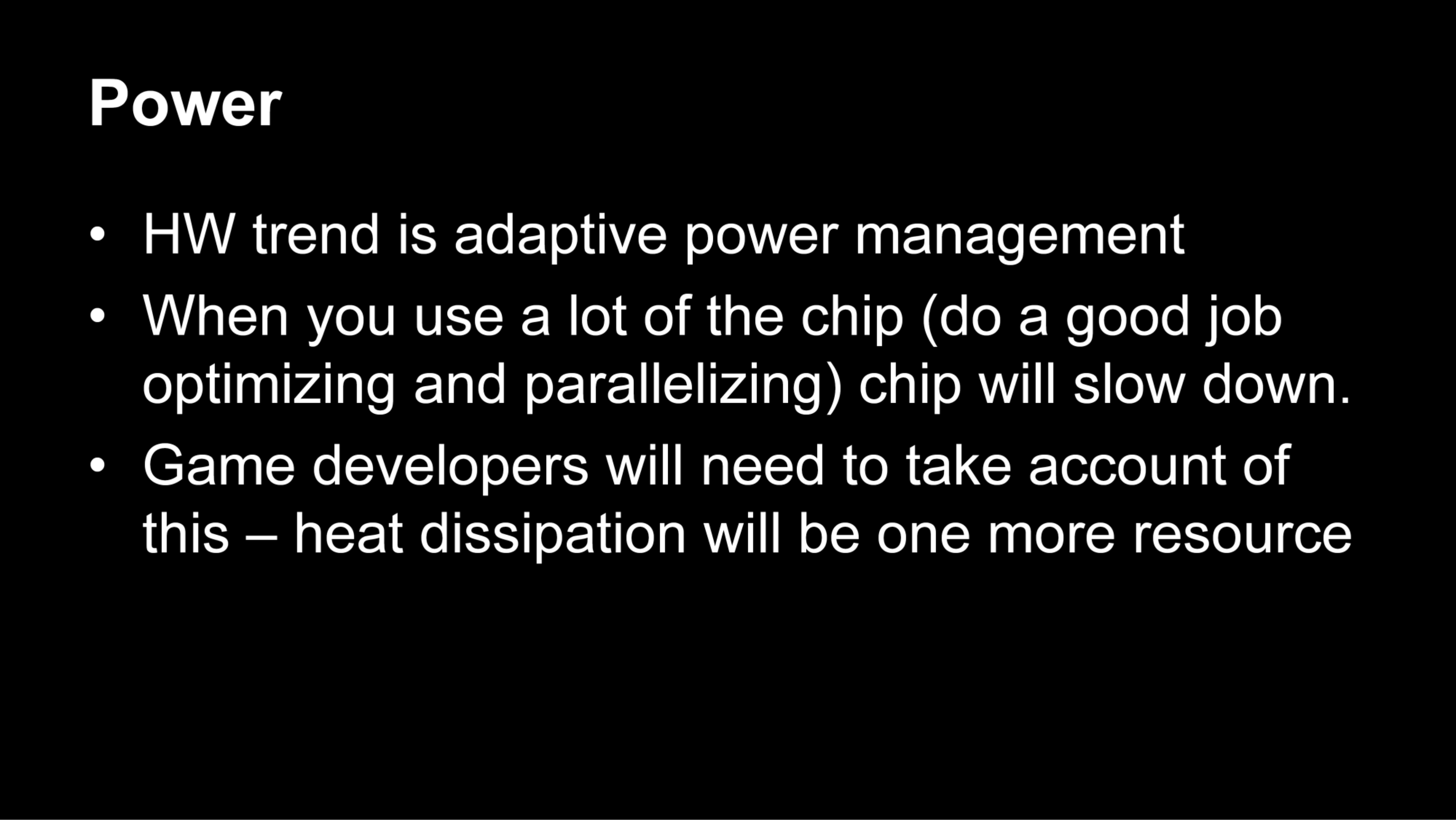
- Film managed without ray tracing for many years, with visuals far beyond current games
  - Using shadow maps, environment maps, etc.
  - With some exceptions like Blue Sky Studios
- Ran against limits and added ray tracing ~2005
- Recently, “pure ray tracing” (Arnold) has been making significant inroads



Learnings for games from film

- Looks like there is still some life in the old hacks
- Adding ray tracing to a rasterization pipeline can be made to work
- Ray-tracing inroads mostly a tradeoff of artist time for render time- not feasible game tradeoff

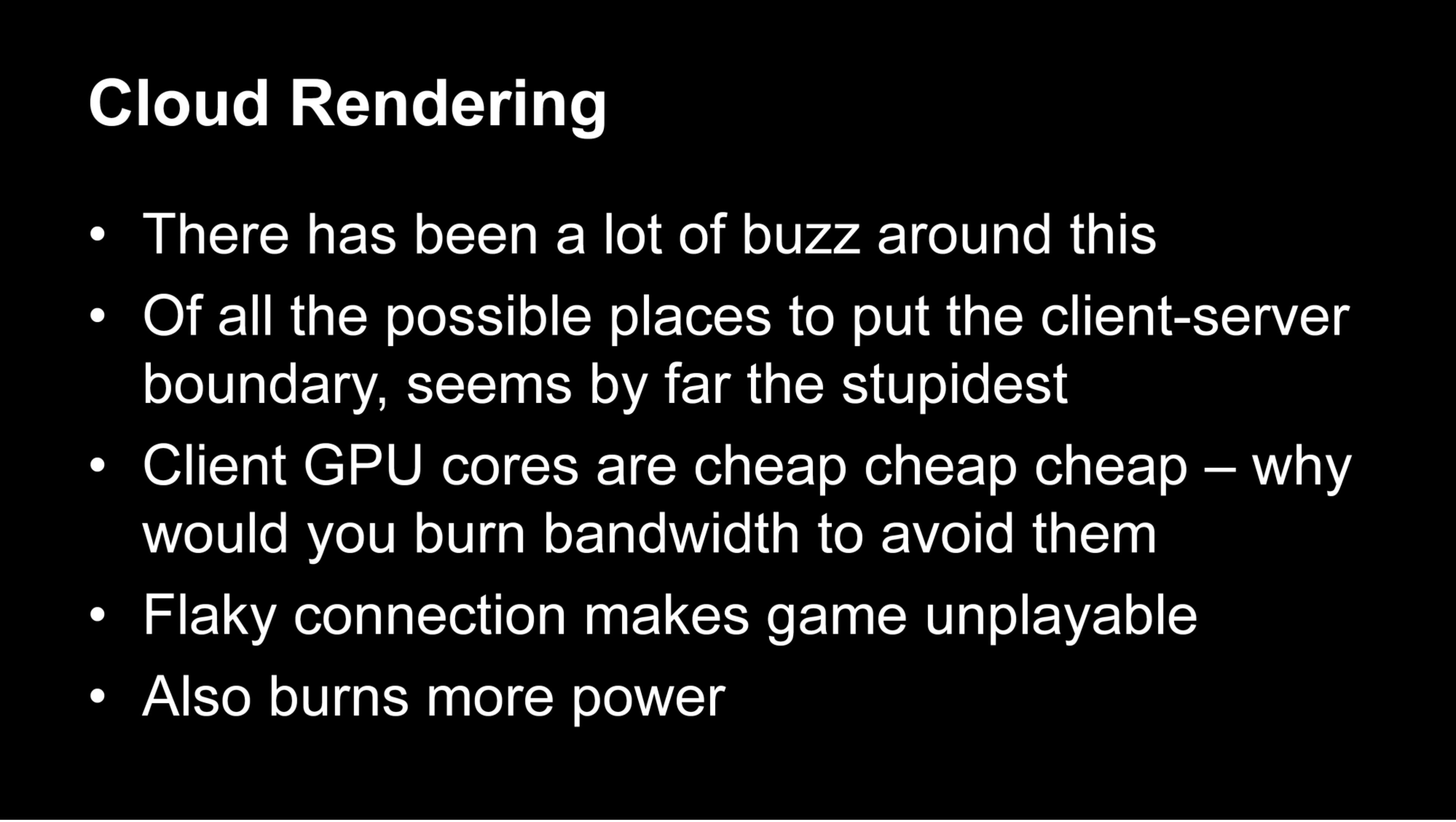
If Pixar could do all their movies until “Cars” without any raytracing, maybe we don’t need it quite yet.



Power

- HW trend is adaptive power management
- When you use a lot of the chip (do a good job optimizing and parallelizing) chip will slow down.
- Game developers will need to take account of this – heat dissipation will be one more resource

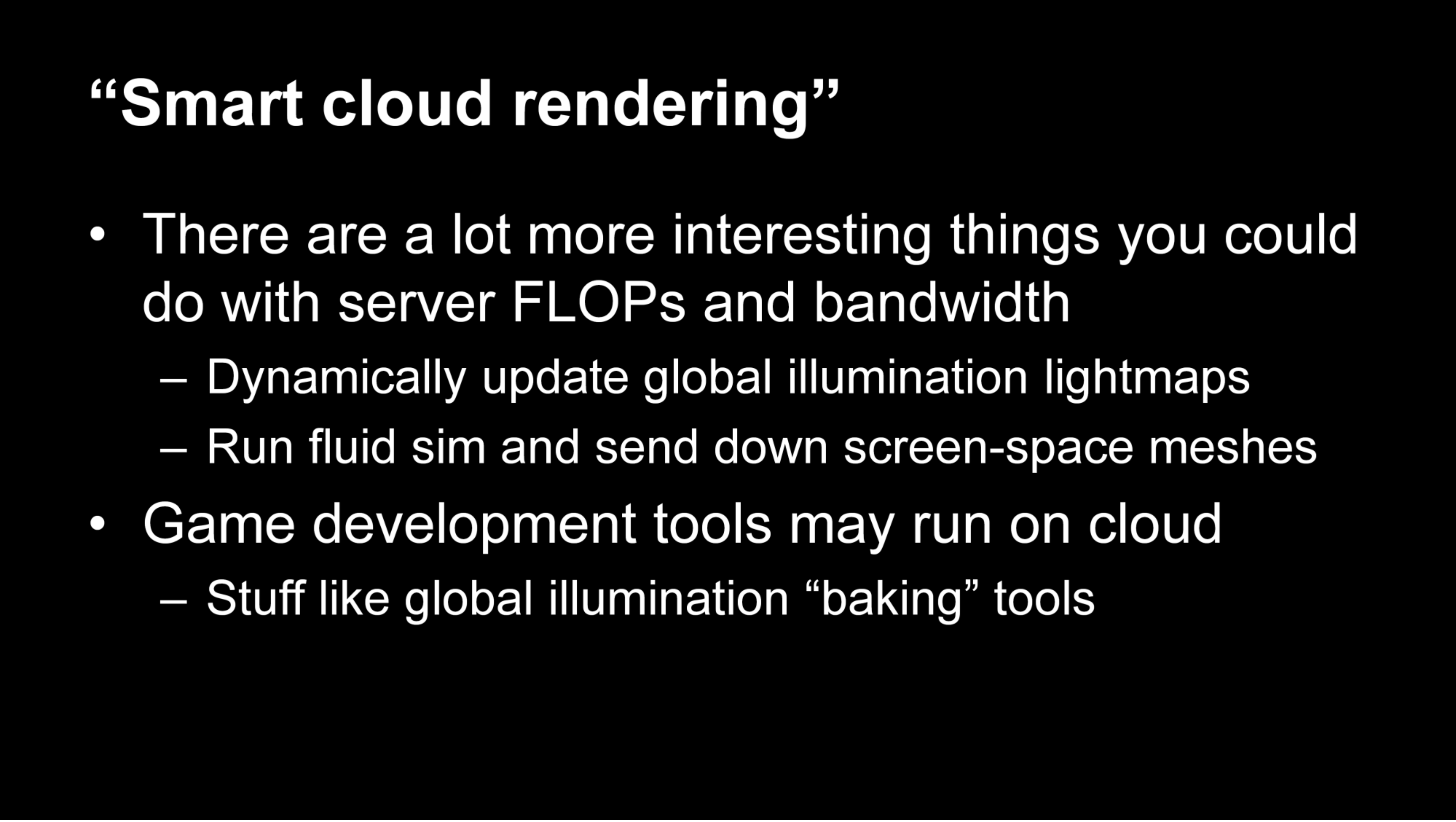
This will be a resource to balance between graphics subsystems and other parts of the game. We are so used to putting all our effort to get something to run at full blast, this will be a painful adjustment.



Cloud Rendering

- There has been a lot of buzz around this
- Of all the possible places to put the client-server boundary, seems by far the stupidest
- Client GPU cores are cheap cheap cheap – why would you burn bandwidth to avoid them
- Flaky connection makes game unplayable
- Also burns more power

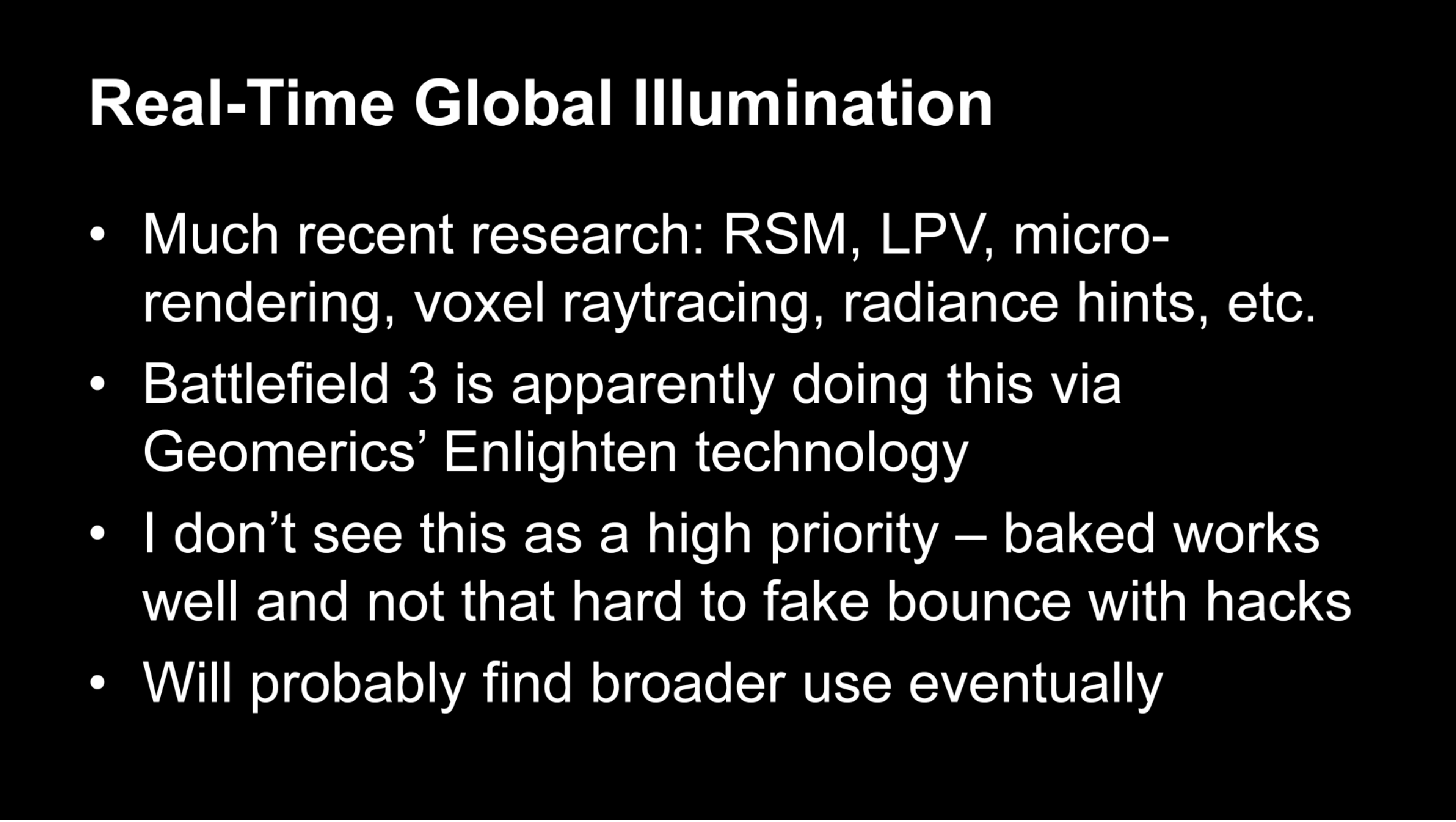
Bandwidth is expensive. Even MMOs, which have tiny packets in comparison, spend a fair amount of engineering effort saving bandwidth. Latency is often an issue as well.



“Smart cloud rendering”

- There are a lot more interesting things you could do with server FLOPs and bandwidth
  - Dynamically update global illumination lightmaps
  - Run fluid sim and send down screen-space meshes
- Game development tools may run on cloud
  - Stuff like global illumination “baking” tools

These two examples are just off the top of my head – I’m sure there are better ones. This is not only a better use of FLOPs and bardwidth, it also has a large robustness advantage; flaky connections will results in less-frequently-updated special effects, not an unplayable game.



Real-Time Global Illumination

- Much recent research: RSM, LPV, micro-rendering, voxel raytracing, radiance hints, etc.
- Battlefield 3 is apparently doing this via Geomerics’ Enlighten technology
- I don’t see this as a high priority – baked works well and not that hard to fake bounce with hacks
- Will probably find broader use eventually

Here I’m mostly referring to diffuse bounce. It’s not that I wouldn’t want it, but I’m not sure it would be worth spending the milliseconds that it would cost, not to mention other issues (like having to construct your scenes a certain way, etc).

Looking at film again

- Diffuse bounce first used around 2003 or so
- Now point-based approach (derived from a real-time technique by Michael Bunnell) popular
- Leads to better looks as well as less work for the lighting artists
- These gains are likely applicable to games also

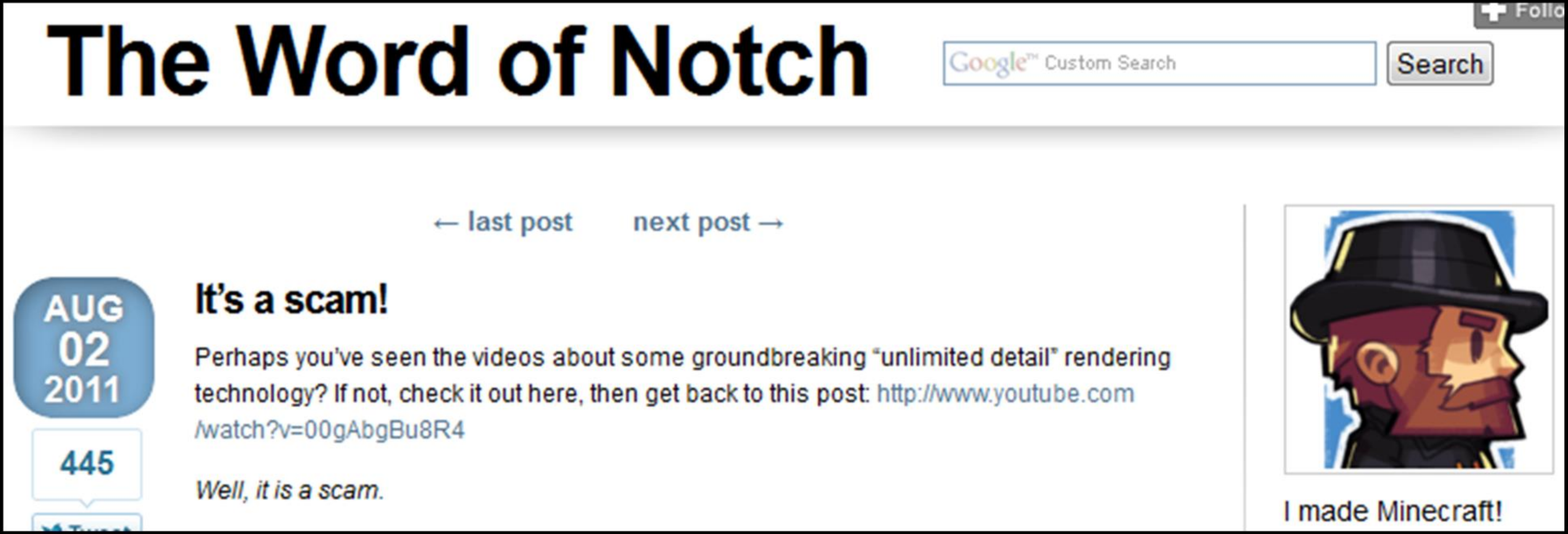
Voxels

- Voxel scene descriptions had some buzz a year or two ago
  - Jon Olick’s demo at SIGGRAPH 2009
  - Gigavoxels (Crassin et. al.)
  - “Efficient Sparse Voxel Octrees” I3D 2010 (Laine & Karras)

A bit more buzz in the last month

(frame from "Infinite Detail" YouTube video used under fair use – low-res image for scholarly commentary)

A bit more buzz in the last month



The Word of Notch

AUG 02 2011

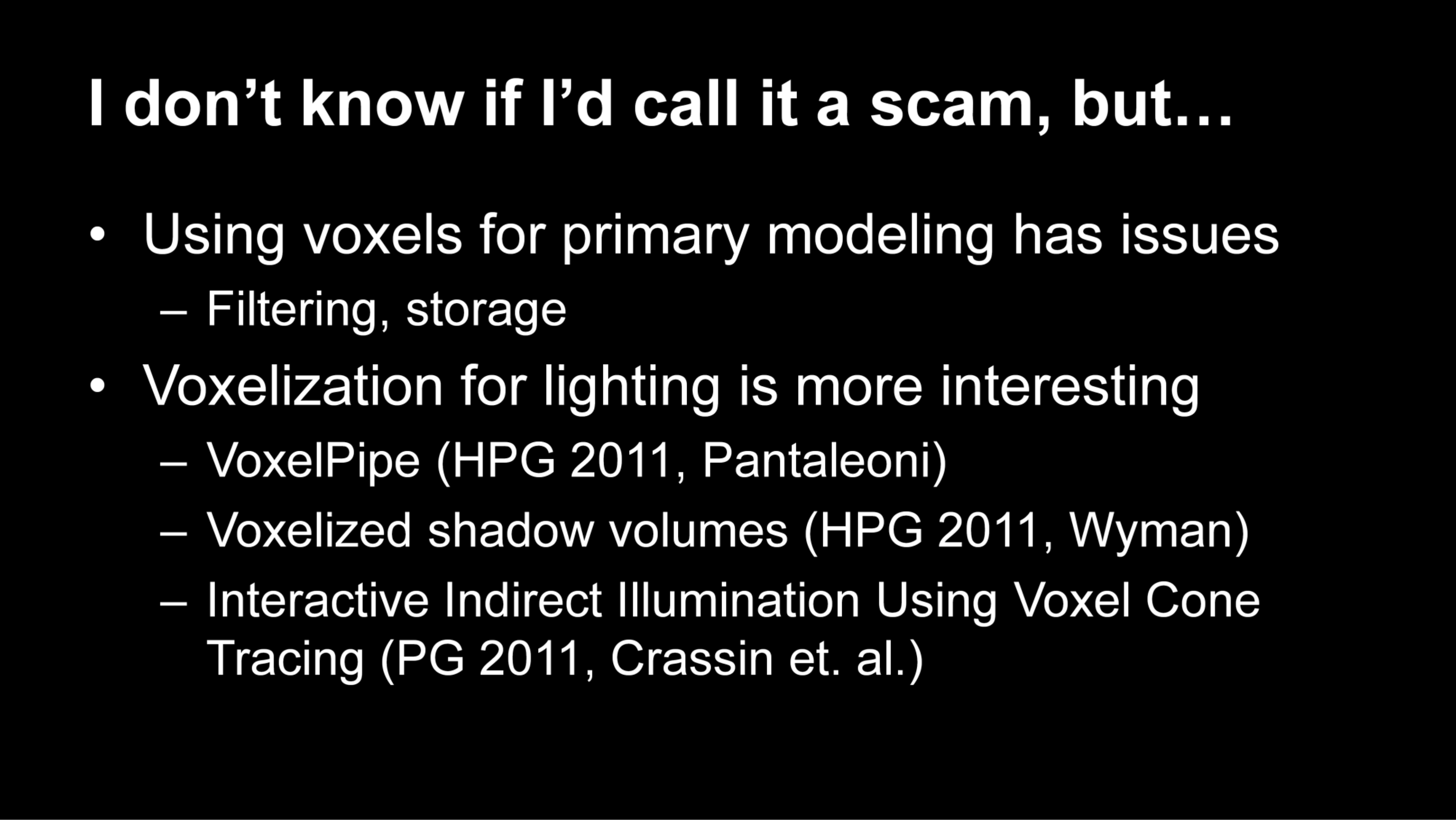
It's a scam!

Perhaps you've seen the videos about some groundbreaking "unlimited detail" rendering technology? If not, check it out here, then get back to this post: http://www.youtube.com/watch?v=00gAbgBu8R4

Well, it is a scam.

I made Minecraft!

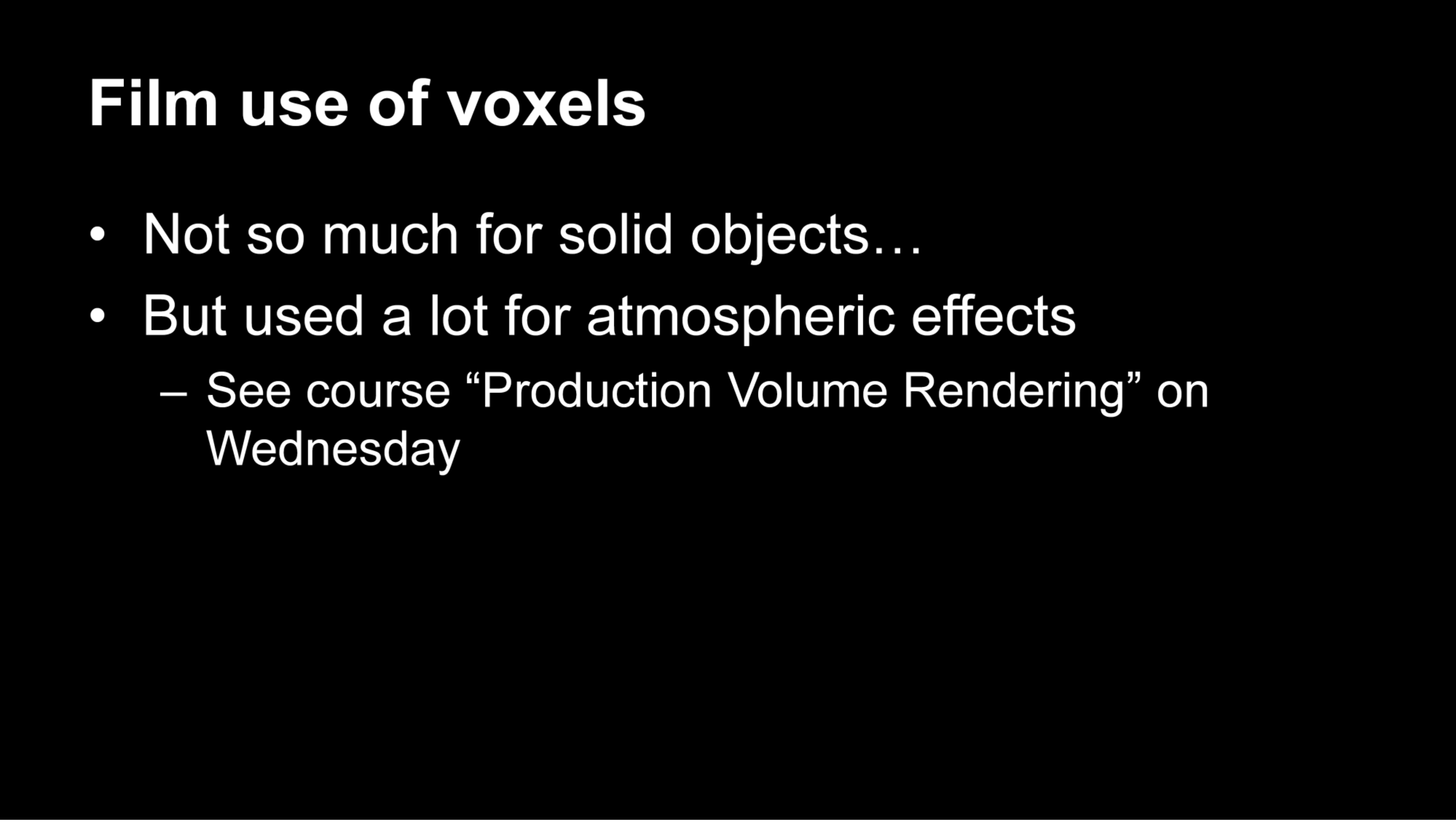
(image from "The Word of Notch" blog used under fair use – partial image for scholarly commentary)



I don’t know if I’d call it a scam, but…

- Using voxels for primary modeling has issues
  - Filtering, storage
- Voxelization for lighting is more interesting
  - VoxelPipe (HPG 2011, Pantaleoni)
  - Voxelized shadow volumes (HPG 2011, Wyman)
  - Interactive Indirect Illumination Using Voxel Cone Tracing (PG 2011, Crassin et. al.)

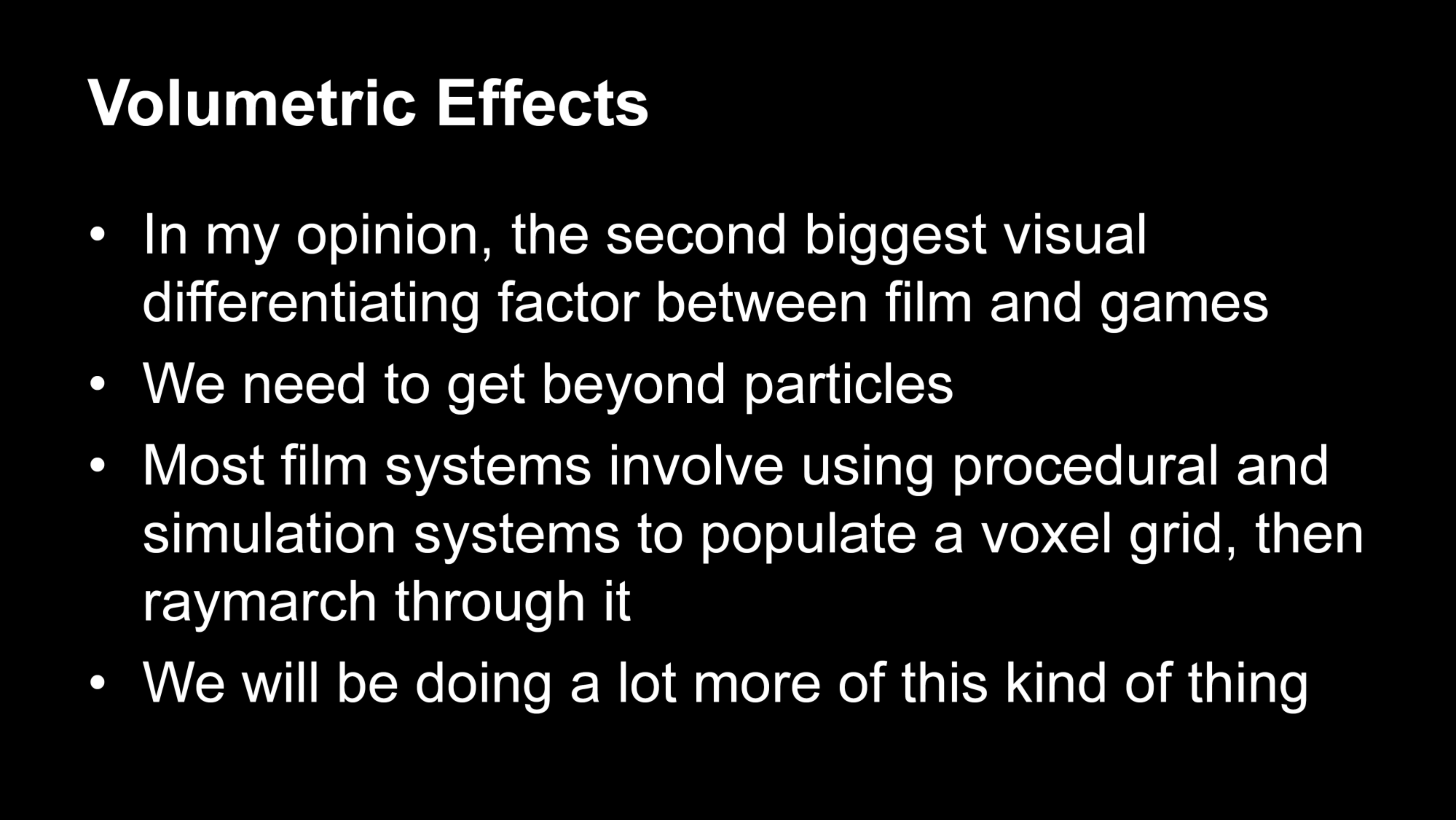
I think voxelization for secondary lighting effects like reflections, bounce shadows, etc. is interesting. This is typically done dynamically and is useful even if it is a pretty coarse representation of the scene.



Film use of voxels

- Not so much for solid objects…
- But used a lot for atmospheric effects
  - See course “Production Volume Rendering” on Wednesday

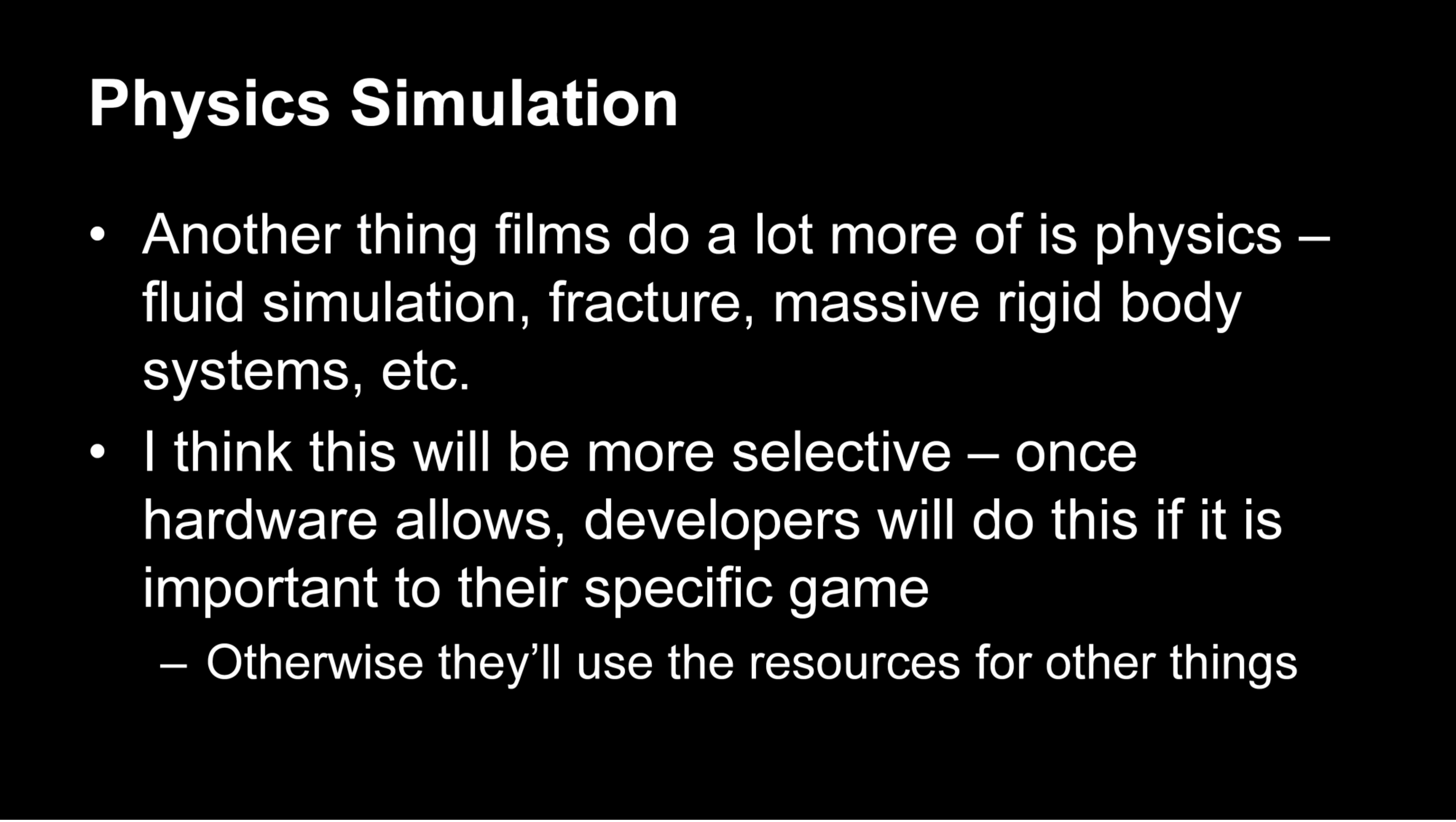
Which leads me to my next topic…



Volumetric Effects

- In my opinion, the second biggest visual differentiating factor between film and games
- We need to get beyond particles
- Most film systems involve using procedural and simulation systems to populate a voxel grid, then raymarch through it
- We will be doing a lot more of this kind of thing

Second after anti-aliasing, in my opinion. Once hardware gets a bit more powerful, we'll be doing more of this kind of stuff – wisps of smoke curling around characters, etc.



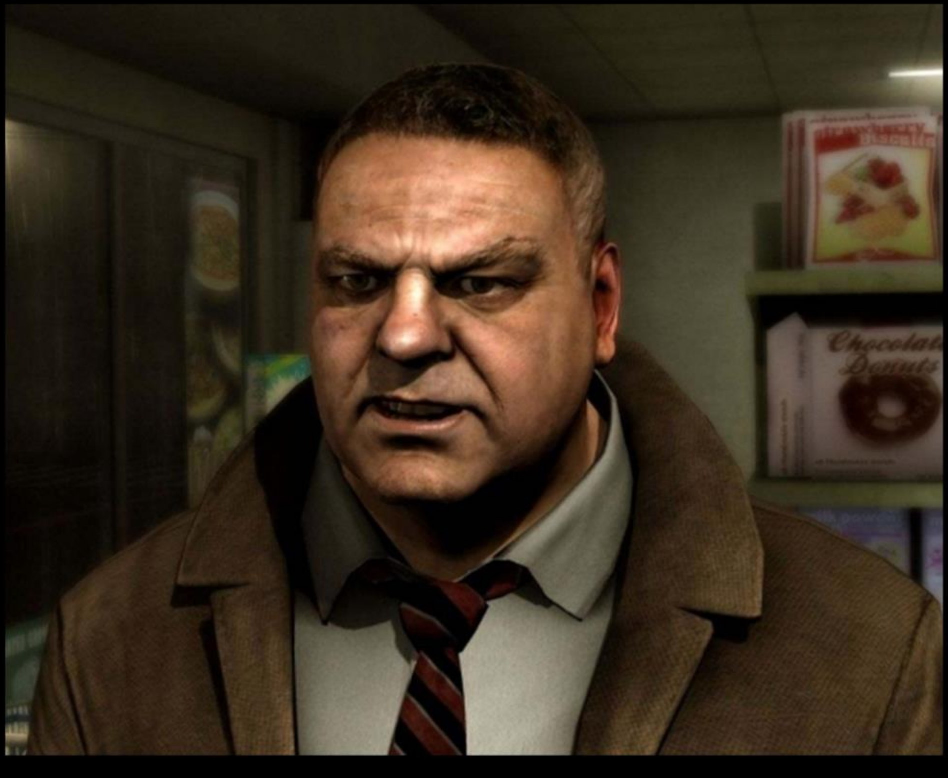
Physics Simulation

- Another thing films do a lot more of is physics – fluid simulation, fracture, massive rigid body systems, etc.
- I think this will be more selective – once hardware allows, developers will do this if it is important to their specific game
  - Otherwise they’ll use the resources for other things

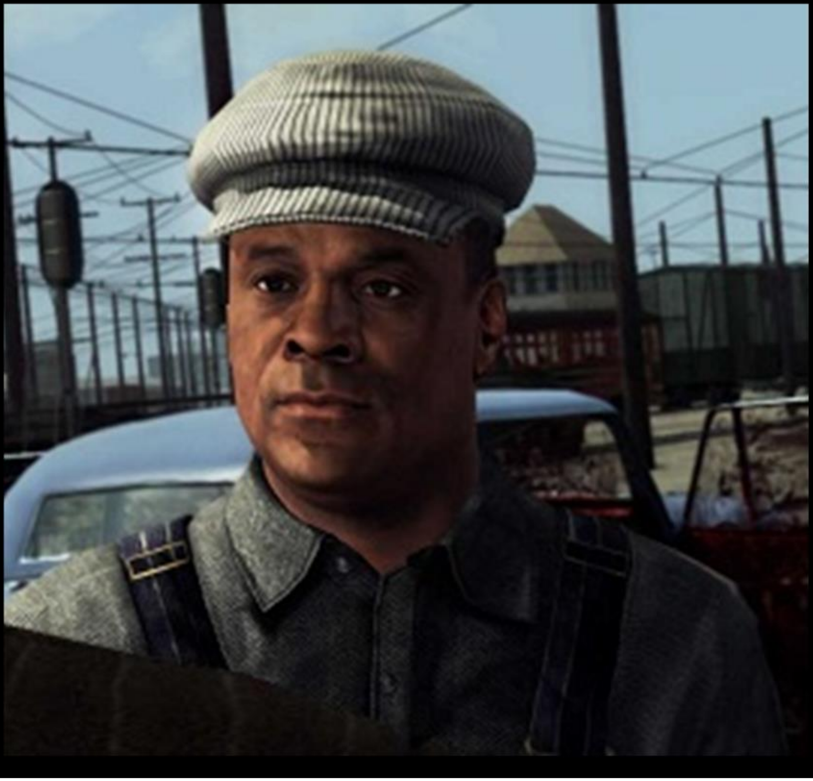
How valuable this is will vary a lot from game to game - for some games, this could actually hurt gameplay, for some it could be irrelevant, for others it could make a huge positive difference. This is unlike atmospheric effects, which would benefit pretty much any game.

Facial Animation

2010 – Heavy Rain

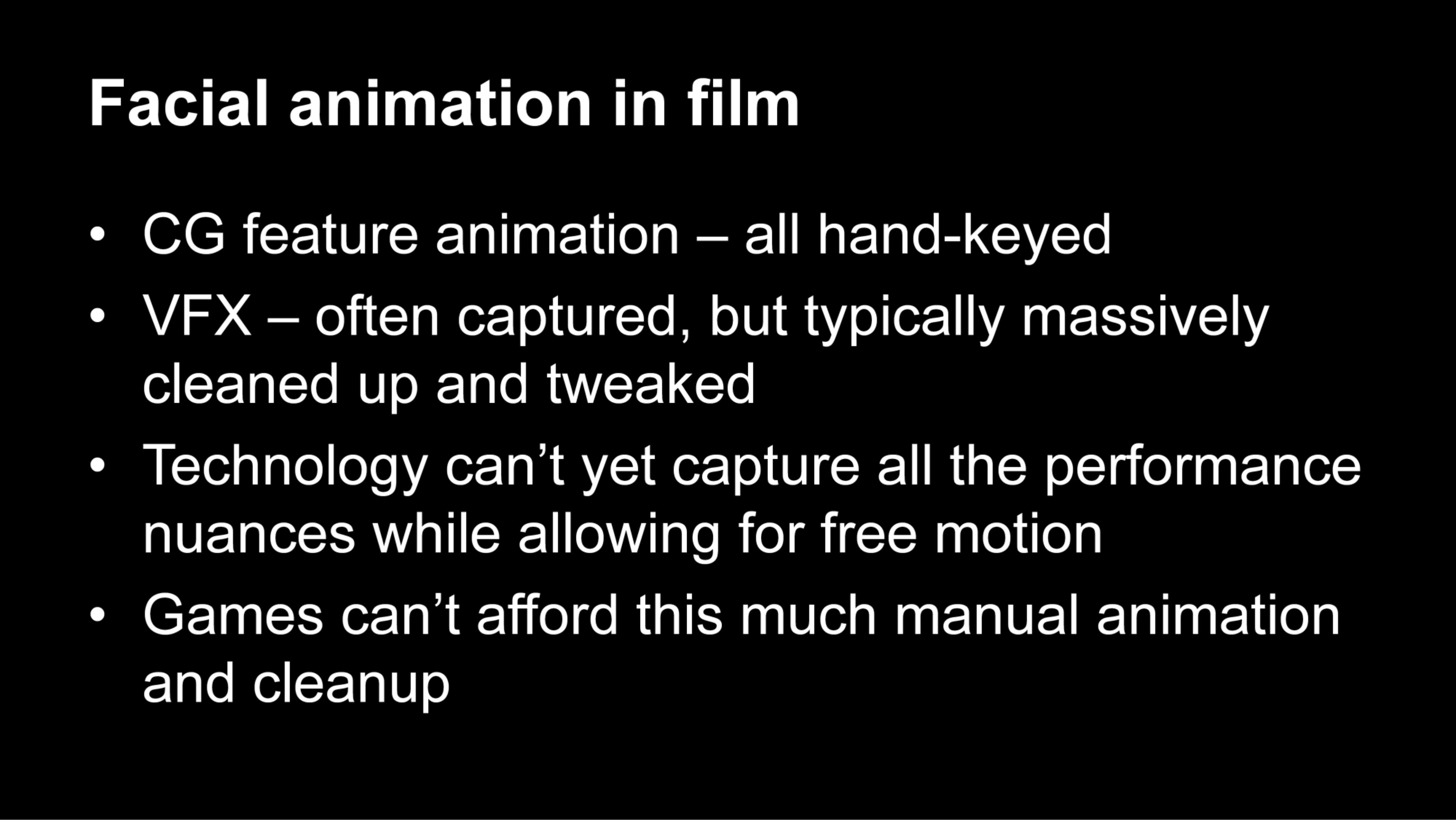


2011 – L.A. Noire



This is a topic of considerable recent interest in the industry – in particular the game L.A. Noire and the MotionScan technology used for the facial performances has captured a lot of attention. The capture technology is flawed in that the characters have to sit still – not conducive to great performances.

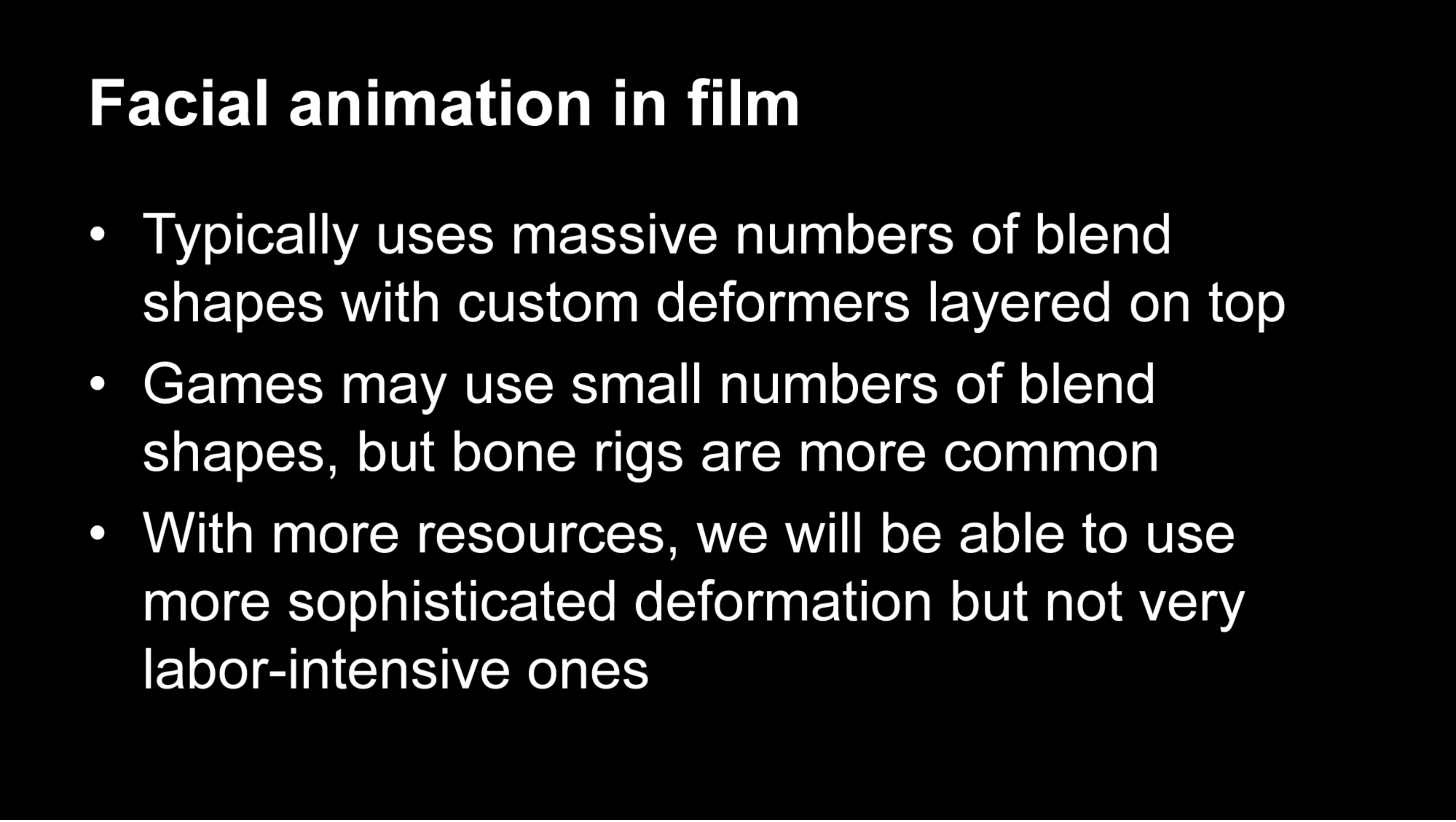
(Heavy Rain and L.A. Noire screenshots used under fair use – partial images for scholarly commentary)



Facial animation in film

- CG feature animation – all hand-keyed
- VFX – often captured, but typically massively cleaned up and tweaked
- Technology can’t yet capture all the performance nuances while allowing for free motion
- Games can’t afford this much manual animation and cleanup

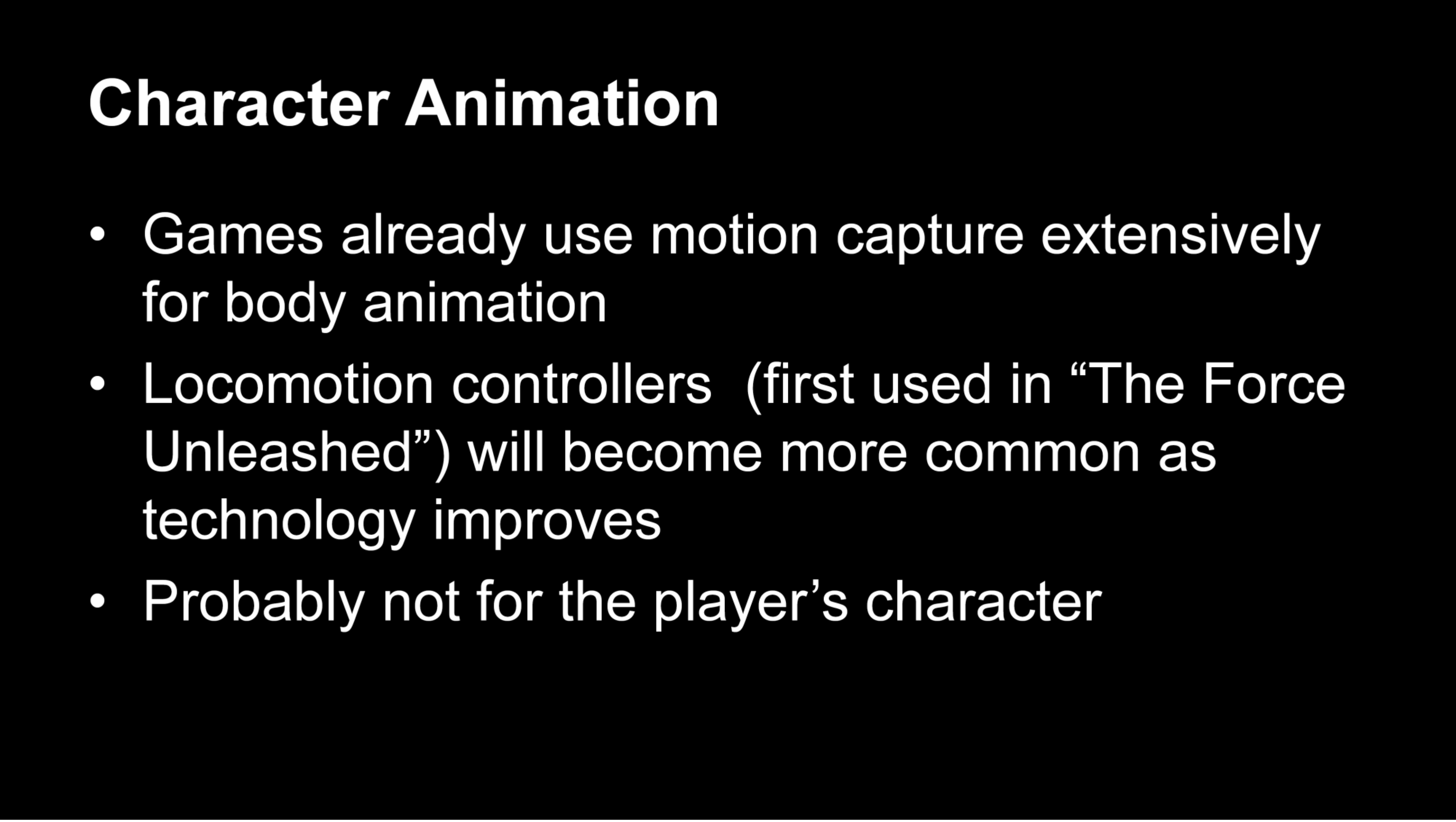
Few CG animated features have relied on facial motion capture and most of these did not do well, either critically or commercially. The current “best practice” in film is highly labor-intensive. Games can’t afford this, since they have significantly less artist hours per screen minute.



Facial animation in film

- Typically uses massive numbers of blend shapes with custom deformers layered on top
- Games may use small numbers of blend shapes, but bone rigs are more common
- With more resources, we will be able to use more sophisticated deformation but not very labor-intensive ones

I think facial deformer rigs in games will evolve, but not necessarily in the same direction as current film ones.



Character Animation

- Games already use motion capture extensively for body animation
- Locomotion controllers (first used in “The Force Unleashed”) will become more common as technology improves
- Probably not for the player’s character

The player’s character needs to be under tight control and to be completely predictable.

Acknowledgements

- Andrew Lauritzen, Pete Shirley, Chris Wyman, Solomon Boulos, Kayvon Fatahalian

Questions?