

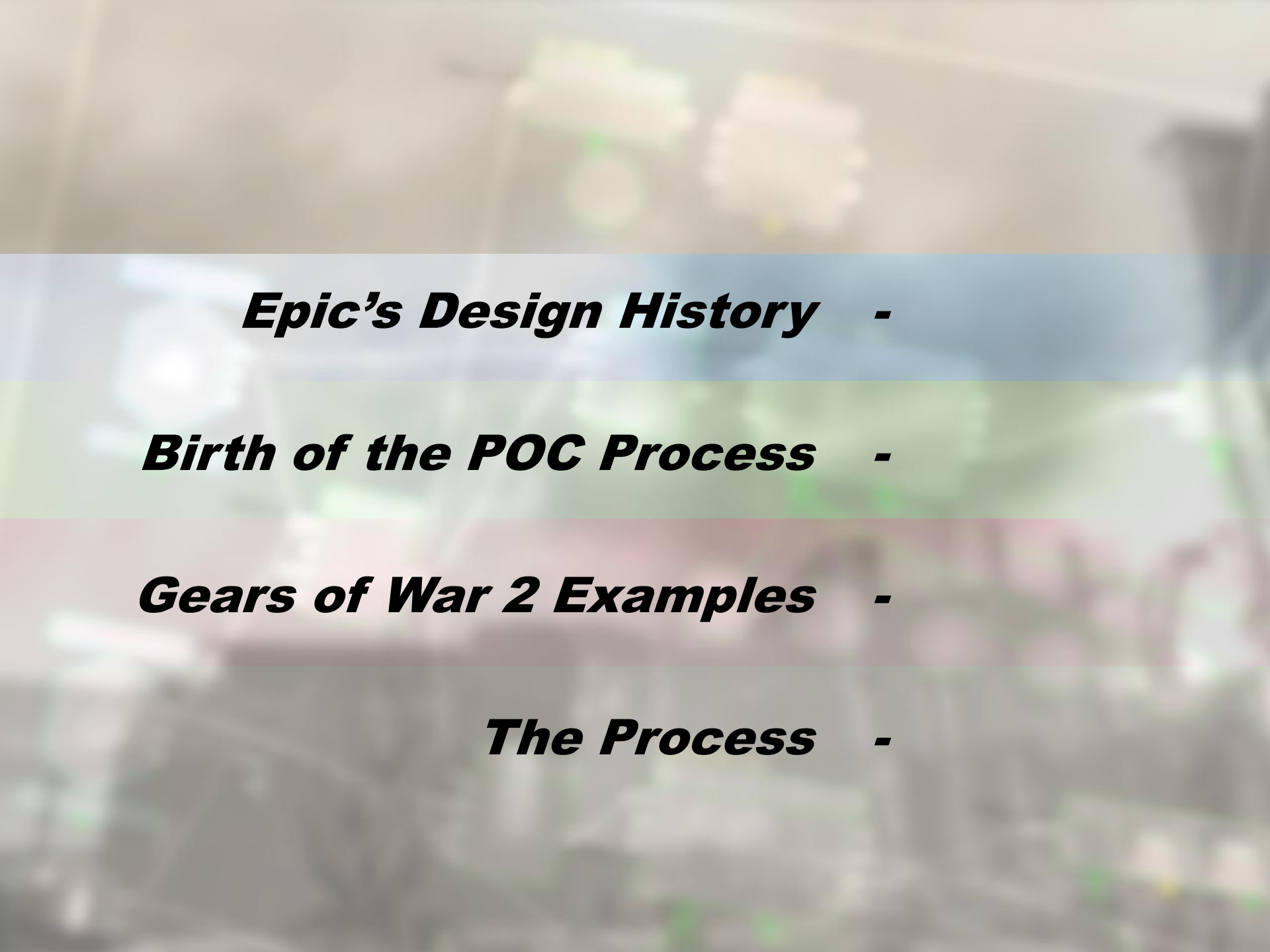


# Prototype Based Design

**A better, faster way to  
design your games**

***Lee Perry***

Senior Gameplay Designer  
Epic Games



***Epic's Design History -***

***Birth of the POC Process -***

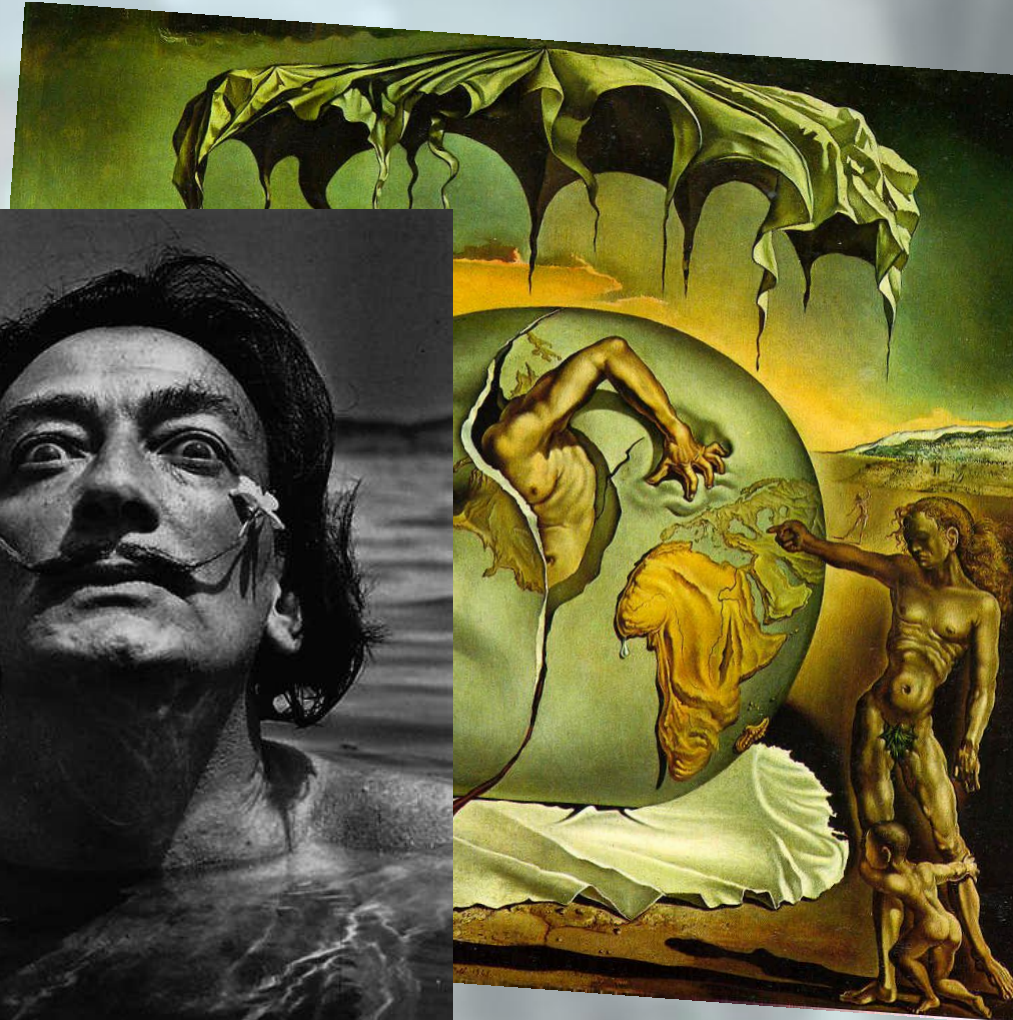
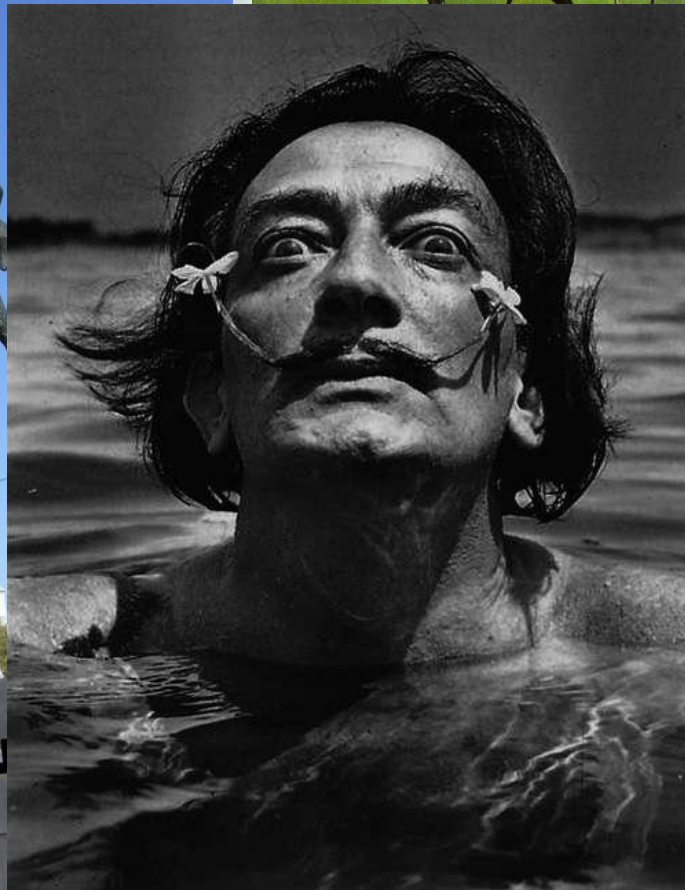
***Gears of War 2 Examples -***

***The Process -***

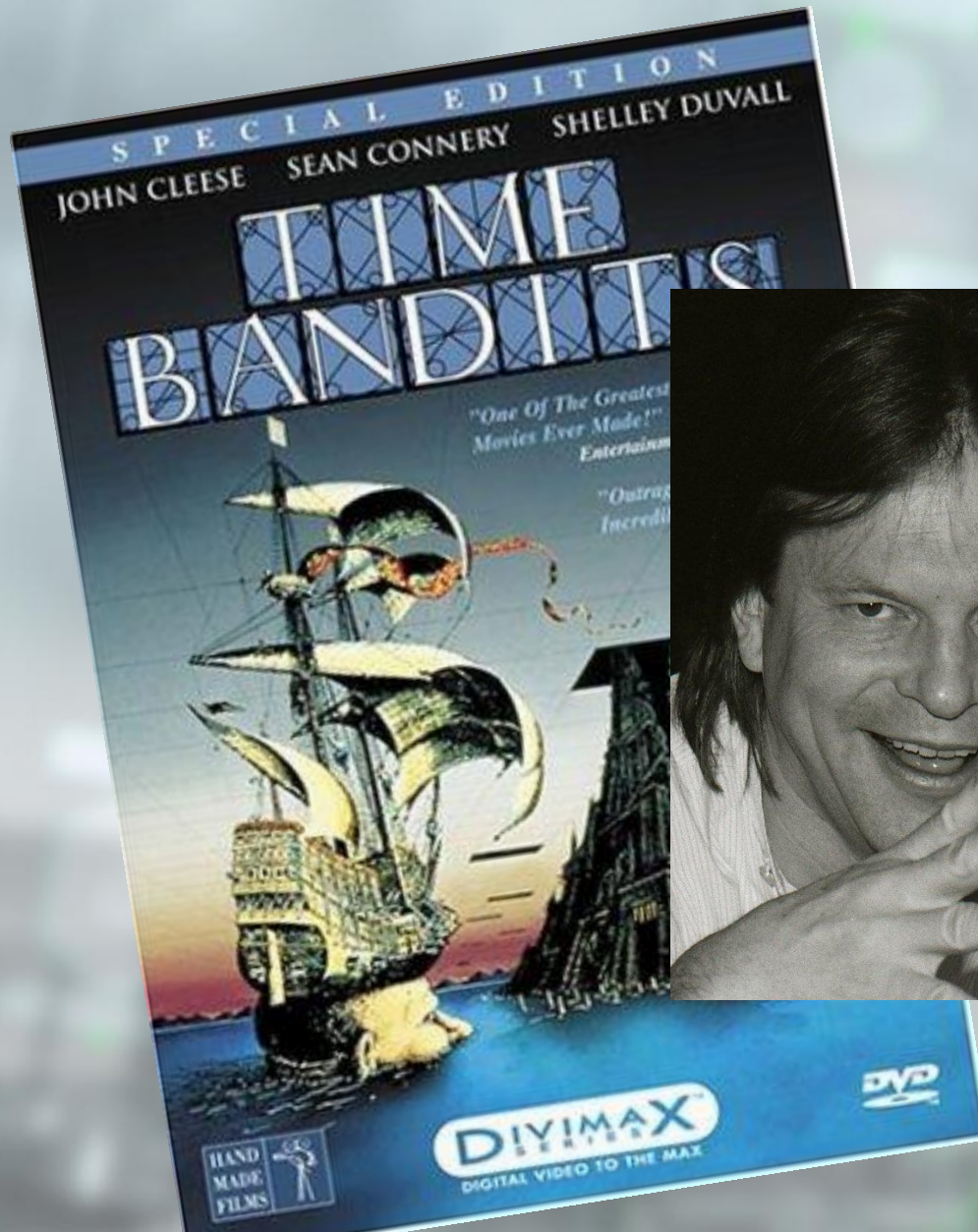
The background is a heavily blurred photograph of a city street. A car is visible in the upper left, and various buildings and street elements are scattered throughout the scene, all out of focus.

# **Epic's Design History**

# Where we came from:



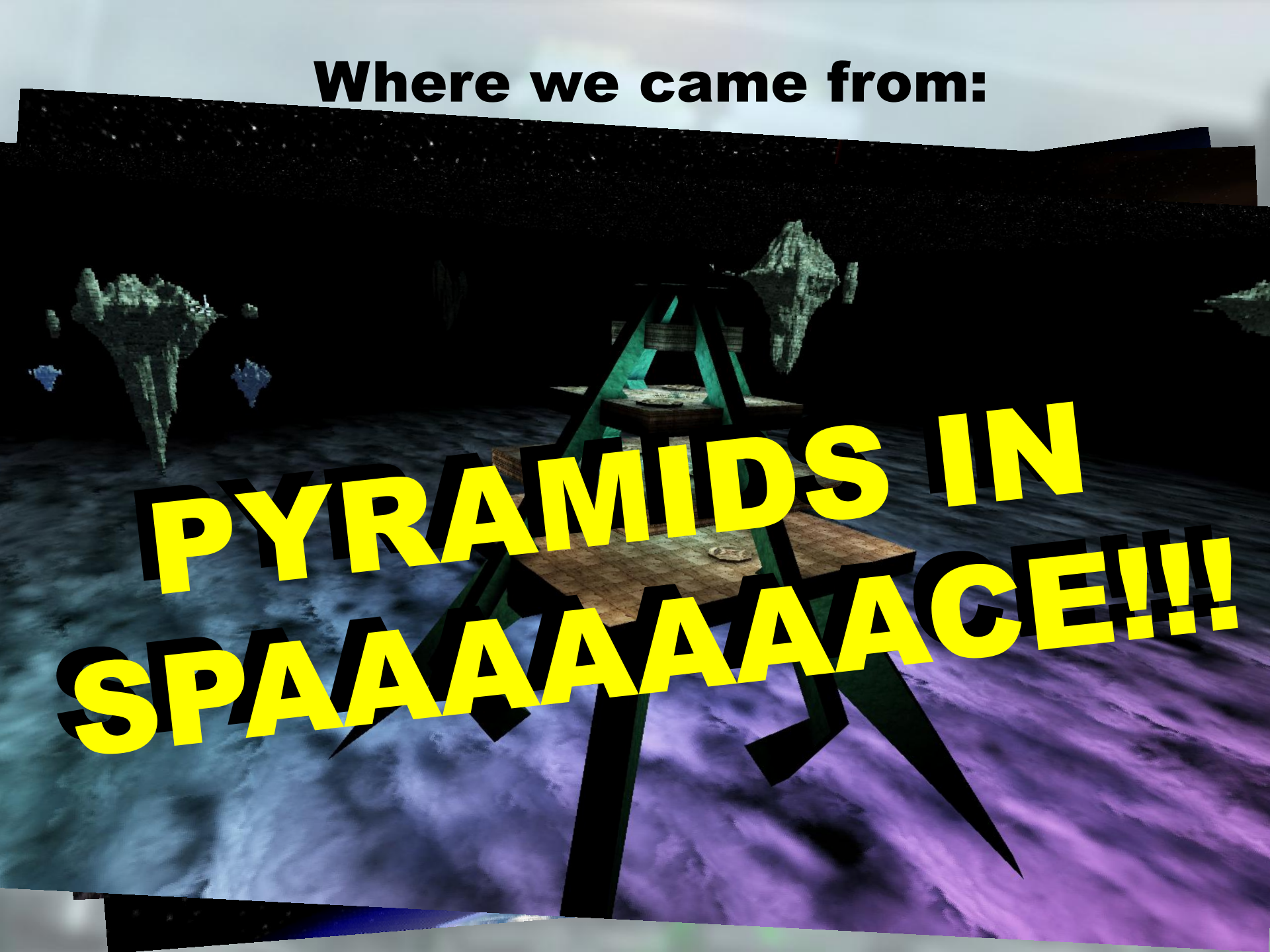
# Where we came from:



**Where we came from:**



**Where we came from:**



**PYRAMIDS IN  
SPACE!!!**

**Where we came from:**



# **The “New Deal”**

- Microsoft wanted our next game
- Lack of coherent design history
- Now we needed a change
- We began production anyway

# **Early Gears of War:**

- Assets and documents coming online
- No shortage of ideas
- Collision course with randomness
- Problem: coordinating creativity with docs

# Why weren't design docs working?

- Too much detail? People won't read it
- Too little detail? Ambiguity, preconceptions
- Reliance on the reader
- The Nay-sayer personality types
- Limited coding resources
- Iteration time
- Designers become food critics

# **An unlikely avenue: Kismet**

Player pulled trigger



Spawn particles / play sound



Cause damage



Spawn blood effects

# **An unlikely avenue: Kismet**

- Created an overweight creature
- Showed it to people
- Shorted out most debate
- Went into production...

# **An unlikely avenue: Kismet**

The “Boomer”



# **The Process: traditional design**

- Design it in your head
- Document it
- Endless debate
- Team prototyping
- Pray it comes out as intended
- Review it for production

A better way...

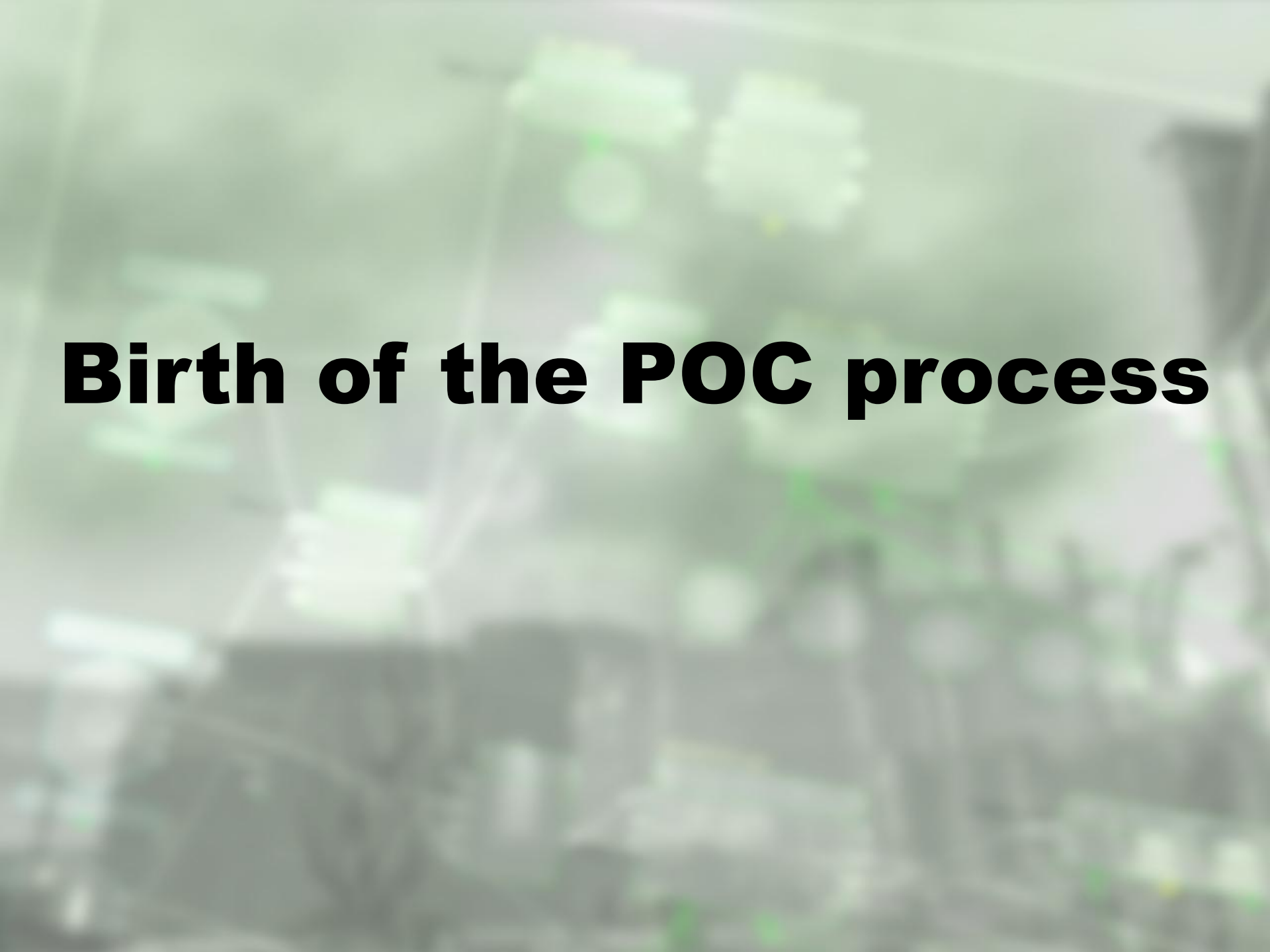
# The Process: traditional design

- Design it in your head
- ~~Document it~~
- ~~Endless debate~~
- ~~Team prototyping~~
- ~~Pray it comes out as intended~~
- Review it for production

# The Process:

## Prototyping based design

- Design it in your head
- ***Designer prototypes it themselves***
- Review it for production
- Document “findings”



# **Birth of the POC process**

**POC (PEE – OH – SEE) : Noun.**

**1) Proof Of Concept. 2) Method  
to avoid total design gridlock.**

**As in: “OMFG I don’t want to argue about this  
anymore, I’m just going to POC the thing!”**

**Or**

**“AW SNAP! Check out this POC of this guy with a  
tank for a head! I call him Riflemouth!”**

**One Example**

**“Unexpected Benefits”**

# Two scenarios

Dialogue and camera system experiments



*\* all videos shown are actual POC levels from Gears production in all their rough glory, not retouched for presentation.*

# Two scenarios

What I envisioned...

- Frames the speaker, shows reactions
- Camera pivots in front of the player's face, with speaker always in background
- Allows the player to continue character relative movement while speaking
- Trigger volumes define interaction space
- Abortable, but cleverly so

# Two scenarios

## Scenario 1 – document that explains:

Camera positioning, location of pivot point, field of view, and explain tracking speeds.

Transition times between lines and with new speakers. How scripting triggers the encounter and manages dialogue lines and camera targets

Walking speed, character relative controls. Scenarios limiting combat interactions, selectively disabling controls, handling edge cases for non present participants.

How to abort the interaction, button mapping and timing. Structuring dialogue for the “I get it, shut up” branching dialogue lines. Explaining the need for “dialogue summary” lines for aborted sequences.

Facial animation implications. “Listening” animations, head tracking, when head tracking is applicable. Discuss the need for higher quality lip syncing, eye animations, facial morphing and increased cinematic quality to the character assets themselves. Are those loaded all the time? When do those transitions happen?

... is it even cool in the first place?!

# Two scenarios

## Scenario 2 – POC level:



# Two scenarios

## Scenario 2 – POC level:

Aborted



# Two scenarios

## Scenario 2 – POC level:

Unexpected benefits



# **Two scenarios**

## **Scenario 2 – POC level:**



# **Two scenarios**

## **Scenario 2 – POC results:**

We discovered issues, and plenty of them

... but it was a very minor investment in time  
(8-10 hours)

Side benefits, and saved boatloads of  
developer effort!



# **Gears of War 2**

**A tough proposition**

# **Gears 2 Development**

**Areas we used POC maps:**

- Creatures
- Weapons
- Gameplay systems
- Level one-offs

**Examples - lets see'em!**

# Creatures

## Bloodmount (1)



# Creatures

## Bloodmount (2)



# Creatures

Bloodmount

Result : shipped



# Creatures

## Cover Worm



# Creatures

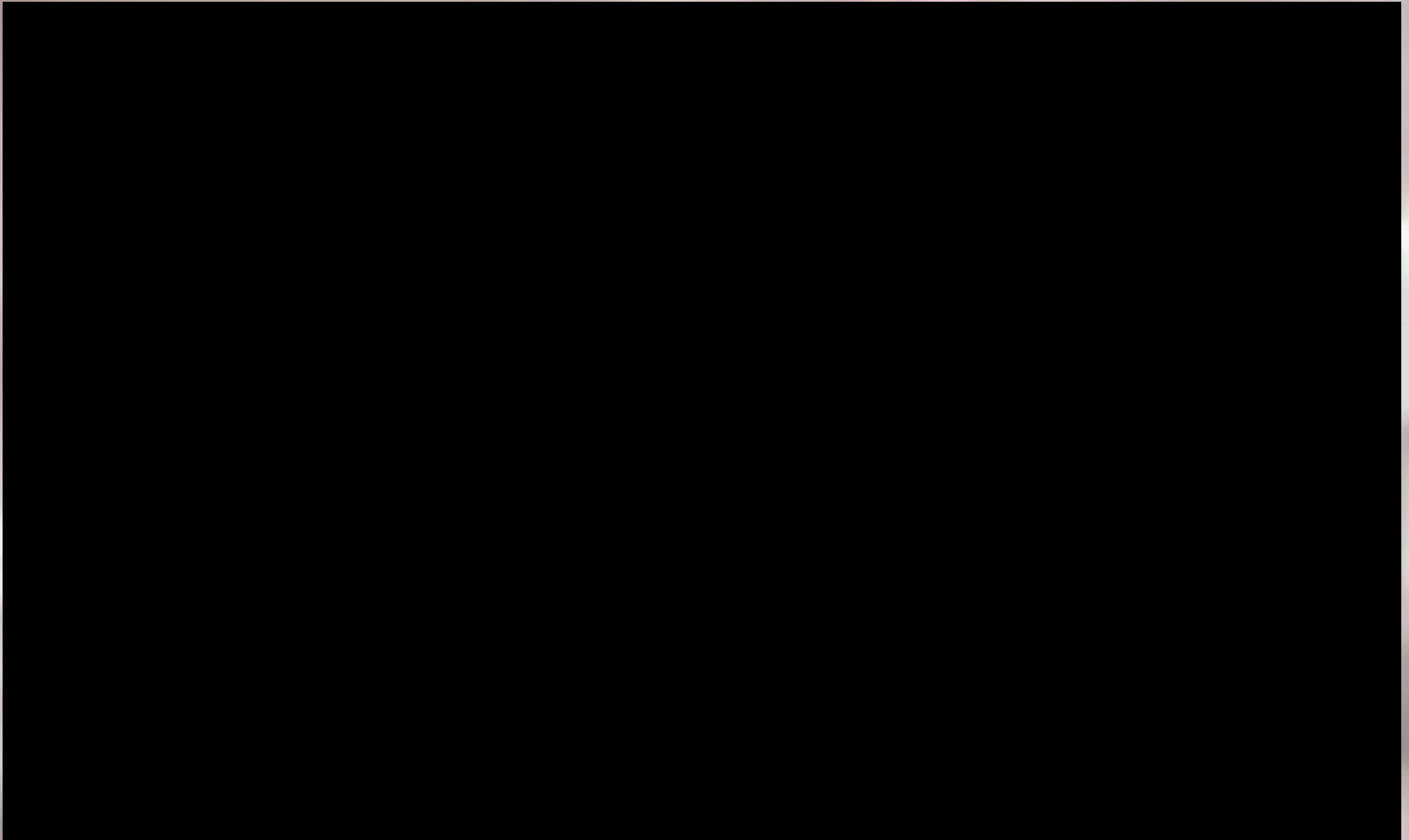
Cover Worm

Result : shipped



# Creatures

## Ground Reavers



# Creatures

Ground Reavers

Result : reduced scope



# Creatures

## Shield Boomer



# Creatures

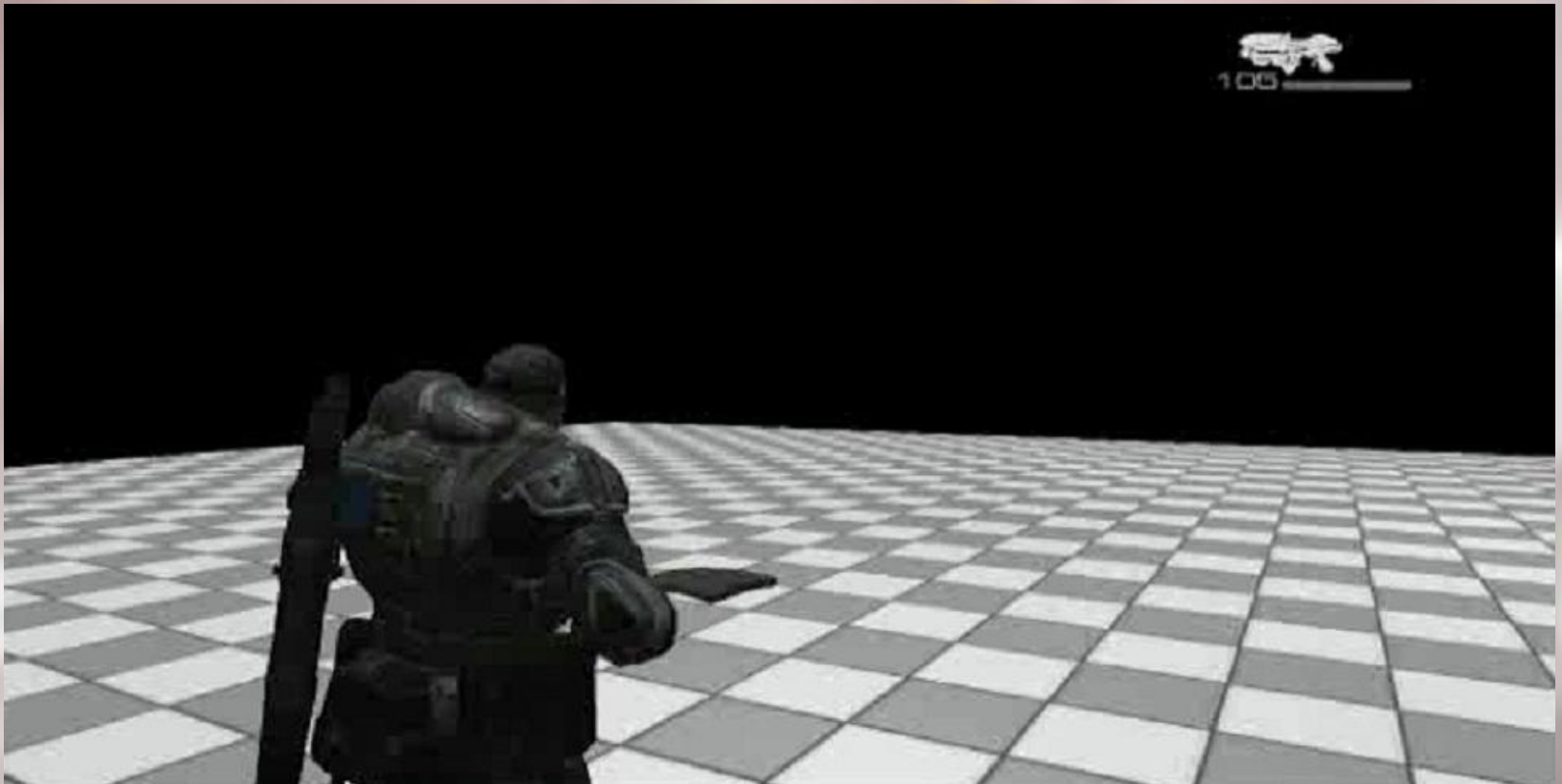
Shield Boomer

Result : shipped



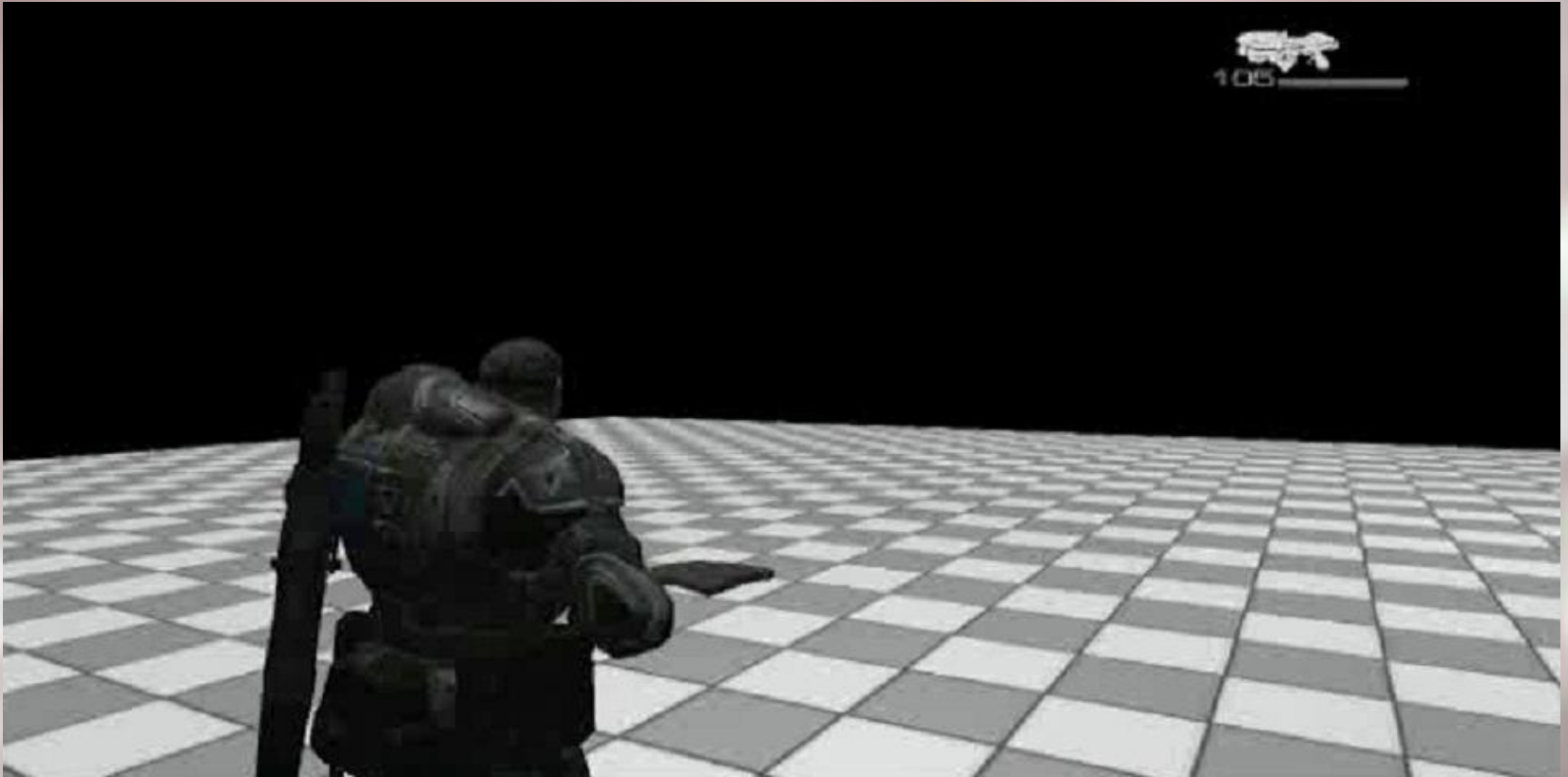
# Weapons

Player Shield (Button toggle)



# Weapons

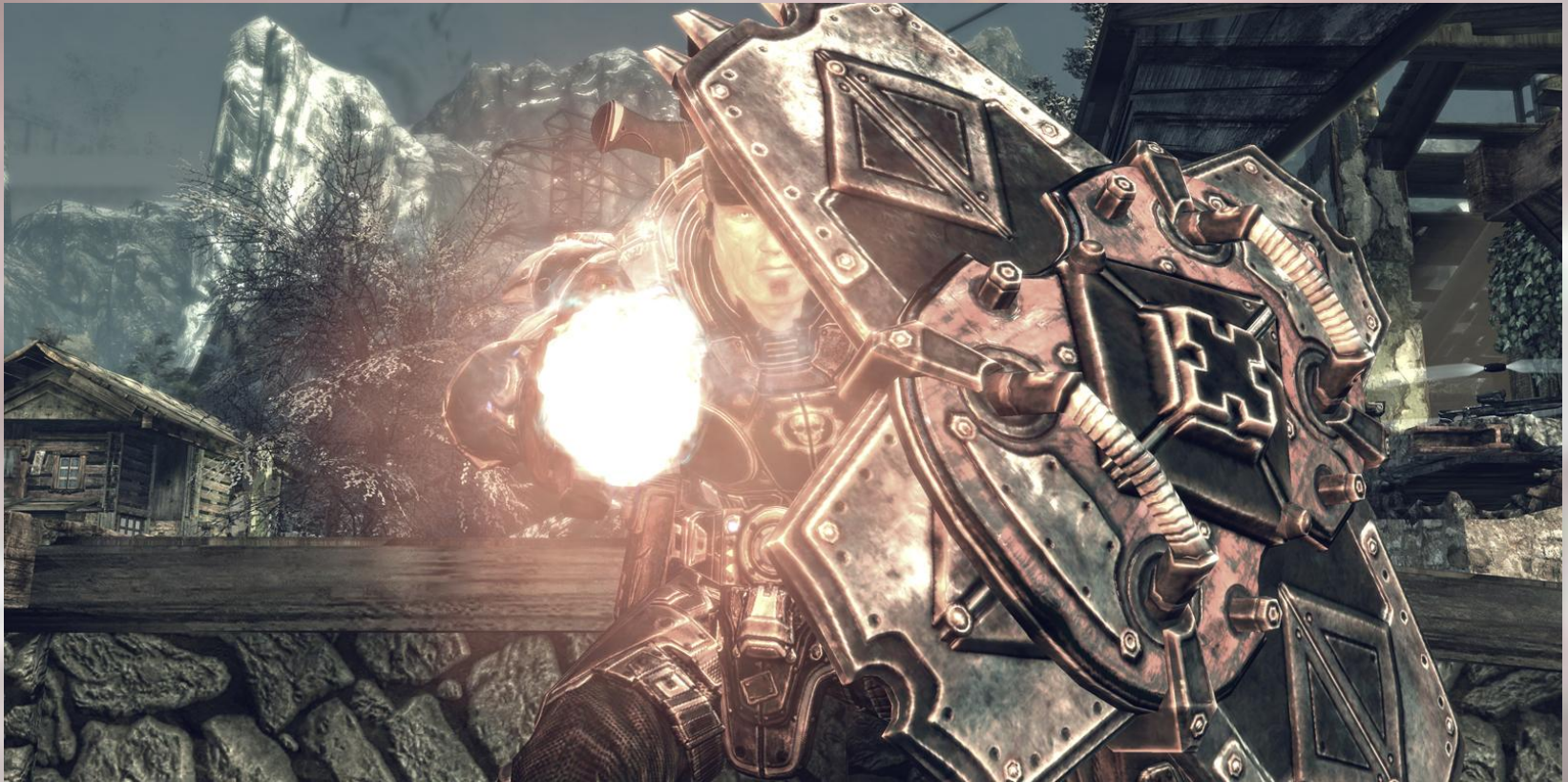
Player Shield (Aimed)



# Weapons

Player Shield (Aimed)

Result : shipped



# Weapons

Scorcher



# Weapons

Scorcher

Result : lost to Flamethrower



# Gameplay Systems

## Flippable Cover



# Gameplay Systems

Flippable Cover

Result : shipped as is



# Gameplay Systems

## Perching



# Gameplay Systems

Perching

Result : trashed



# Level Scenarios

## Reaver Ride



# Level Scenarios

Reaver Ride

Result : shipped



# Level Scenarios

## Riftworm Heart



# Level Scenarios

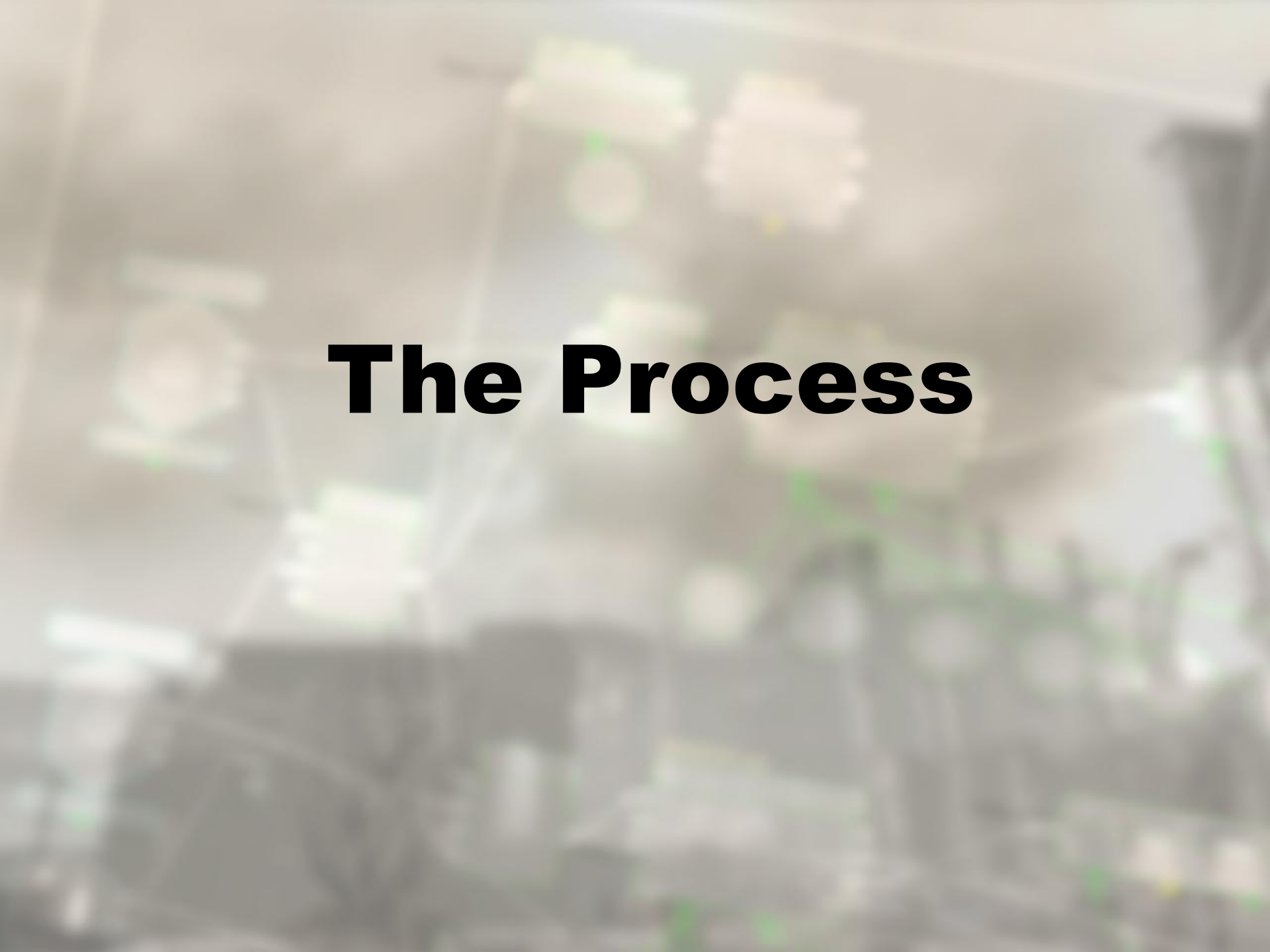
Riftworm Heart

Result : shipped



# **Voila! Gears of War 2**

- Turned it around in 24 months
- One gameplay programmer for first 6 months
- Reasonable amounts of iteration time
- Minimal churn (even with only 40% success)
- Critical success... Christmas was saved!



# **The Process**

# Making a POC

- It needs to be created by one “designer”
- It needs to come online quickly
- It needs to use existing assets
- It needs iteration
- It needs feedback
- It needs more feedback
- ***Seriously***, don't skip the feedback

# **Tools needed for a POC**

- A few basics (the building blocks for “cheating”)
  - Trigger events
  - Move objects
  - Attach objects and triggers
  - Play sounds and visual effects
  - Treat cameras like any object
  - Dummy fire weapons
  - Trap controller inputs
- Accessible to designers and artists
- Quick to preview and iterate with

# **Designers needed for a POC**

Need to:

- Be creative, obviously
- Cross disciplined
- Be able to improvise and scavenge
- Value presentation & salesmanship
- Be an internal ambassador for your game...
- Be optimistic and open minded

# Cons and Pitfalls

- Ultimately, it's still just a design doc
- Coders can feel like they're on clean-up
- Prototypes are ugly, internal only
- With too much freedom, Space Pyramids
- Don't be afraid to throw out ideas
- Tools and training needed
- Even with easy tools, some designers might still not get it

# Where Now?

- Rethink what really needs to go into your Proof of Concept (POC) maps
- Rethink “design”, break dependencies
- Find these people, give them an outlet
- Get some tools, or arrange code help
- Find some contentious test cases
- Use assets from unexpected places
- Find a way to take more risks, let it happen!

**Lee Perry**

**Senior Gameplay Designer**

**Epic Games**

**South Hall, Booth BS200**

**UnrealTechnology.com**

**EpicGames.com**

**...end of line**

# **Lee Perry**

## **Senior Gameplay Designer**

### **Epic Games**

### **South Hall, Booth BS200**

Copyright © 2010, Epic Games, Inc. All Rights Reserved.  
Epic, Epic Games, the Epic Games logo, Gears of War, Unreal Tournament,  
Unreal Engine, Unreal MCP and the Circle-U logo,  
are trademarks or registered trademarks of Epic Games, Inc.  
in the United States of America and elsewhere.  
Other brands or product names are the trademarks of their respective owners.

**...end of line**