



# Testing Horizon Zero Dawn

Ana Barbuta  
QA Manager  
Guerrilla Games



# Topics

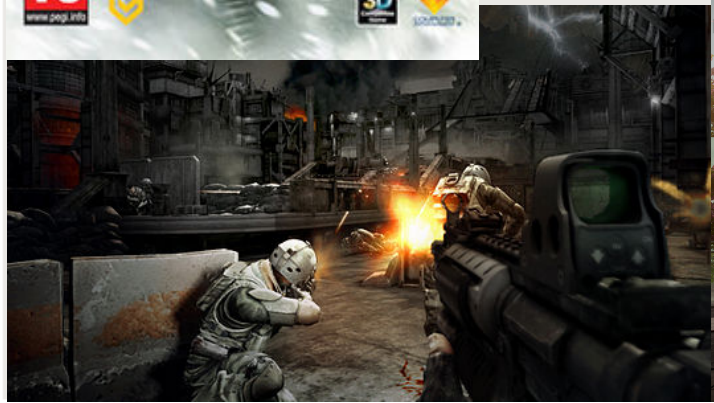
- Intro
- Game testing vs Software testing
- Testing *Horizon Zero Dawn*
  - Approach
  - Tools
  - Exploratory testing



## Who we are – Guerrilla Games

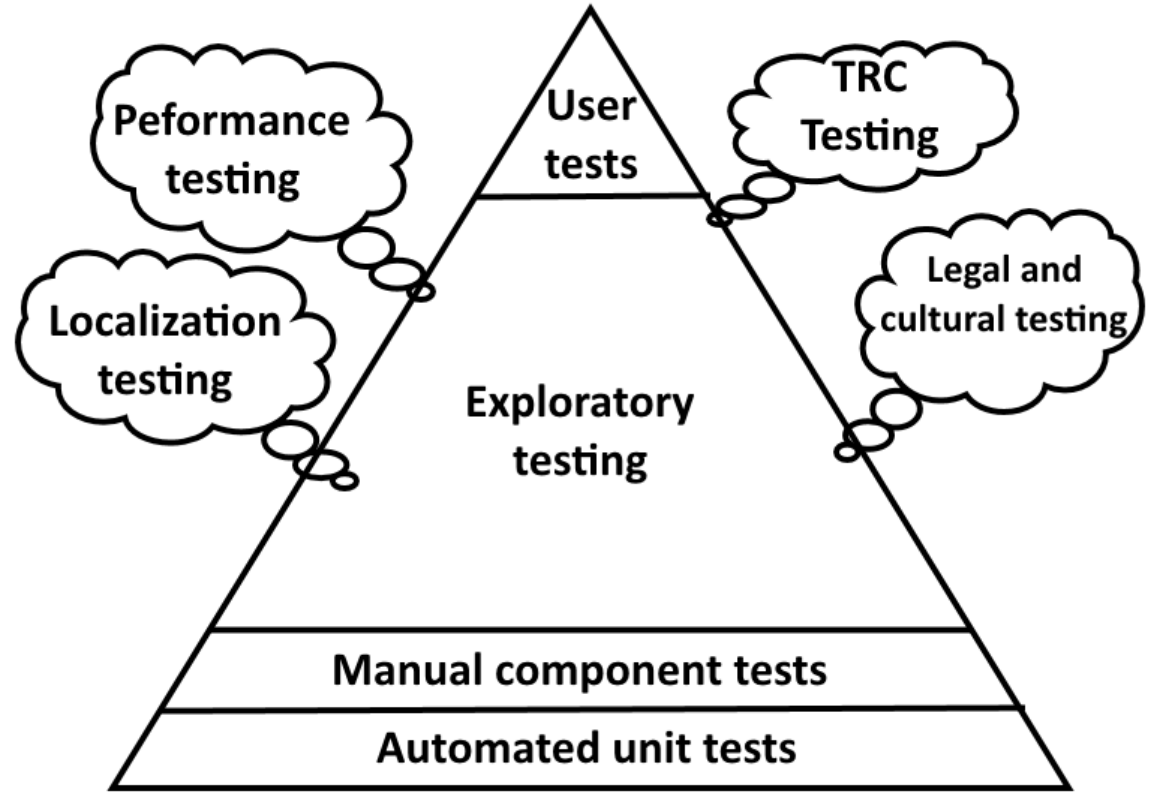
- Founded in 2000
- Acquired by Sony in 2005
- Until HZD, we developed first person shooter games for PlayStation platforms
- We developed our own engine, called Decima
- Mainly worked on the Killzone franchise



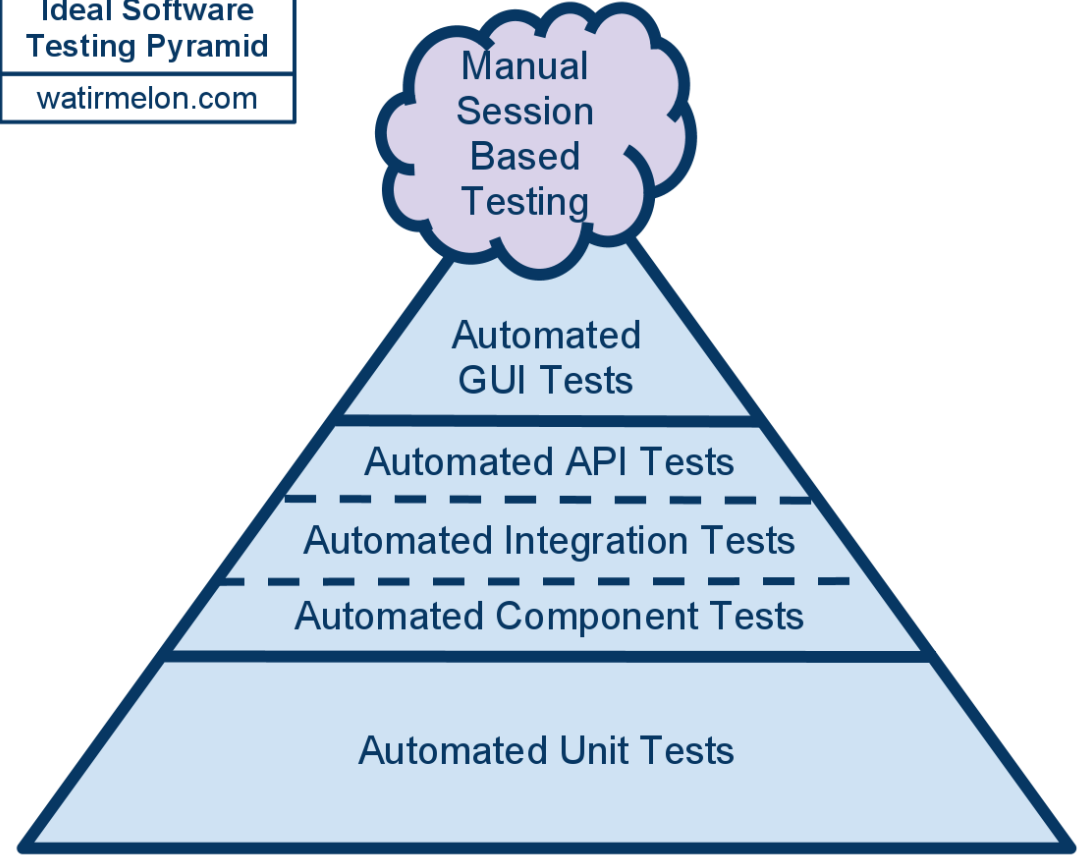





# Game testing versus software testing



Ideal Software Testing Pyramid  
watirmelon.com





# Game testing versus software testing

- Games are very INTERACTIVE
- Dozens, sometimes hundreds of components
- Bugs usually appear at intersections
- Focus on integration/system tests
- Acceptance testing takes the form of playtests and usability tests





# Components in a game

- Code (engine and features)
- Content (missions, encounters, scripts)
- Animated models
- Static assets and game spaces
- AI, animation and physics systems
- Lighting and post-processing
- Dialogue, sound effects, music
- Visual effects and shaders
- (Online components)





# Testing Horizon Zero Dawn

The challenge, and how we solved it





# The project

- Our first open world game
- Our first action-RPG
- Non-linear gameplay
- Engine had to be adapted
- New pipelines
- New development processes
- New tools







## Some numbers

- 45 hours of play time (average)
- 57 quests
- 492 unique combat encounters
- 44 unique enemies
- 10 hours of cinematics
- Over 480 000 words of dialogue
- Over 180 000 hours of testing
- Over 2.6 million units sold in the first 2 weeks





## The challenge

- Much larger scope
- Almost infinite number of possible player paths
- New and untried test strategy required
- Limited test team size
  - Both internal and external
- We wanted to test smarter





# What we did





## What we did

- Changed the QA team structure





## What we did

- Changed the QA team structure
- Used testing tools and debug tools





## What we did

- Changed the QA team structure
- Used testing tools and debug tools
- Used telemetry to inform our testing







## What we did

- Changed the QA team structure
- Used testing tools and debug tools
- Used telemetry to inform our testing
- Changed our test strategy to leverage exploratory testing





# Team Structure

- Internal QA team = 8
- Embedded specialists
- Ownership of their area
- Mostly autonomous
- Close collaboration with devs
- Supplement with Sony QA





## Why this worked

- QA gained deep knowledge of their areas
- Created good connections with the development teams
- Ownership = fully invested
- Embedded QA was in the best position to decide test strategy





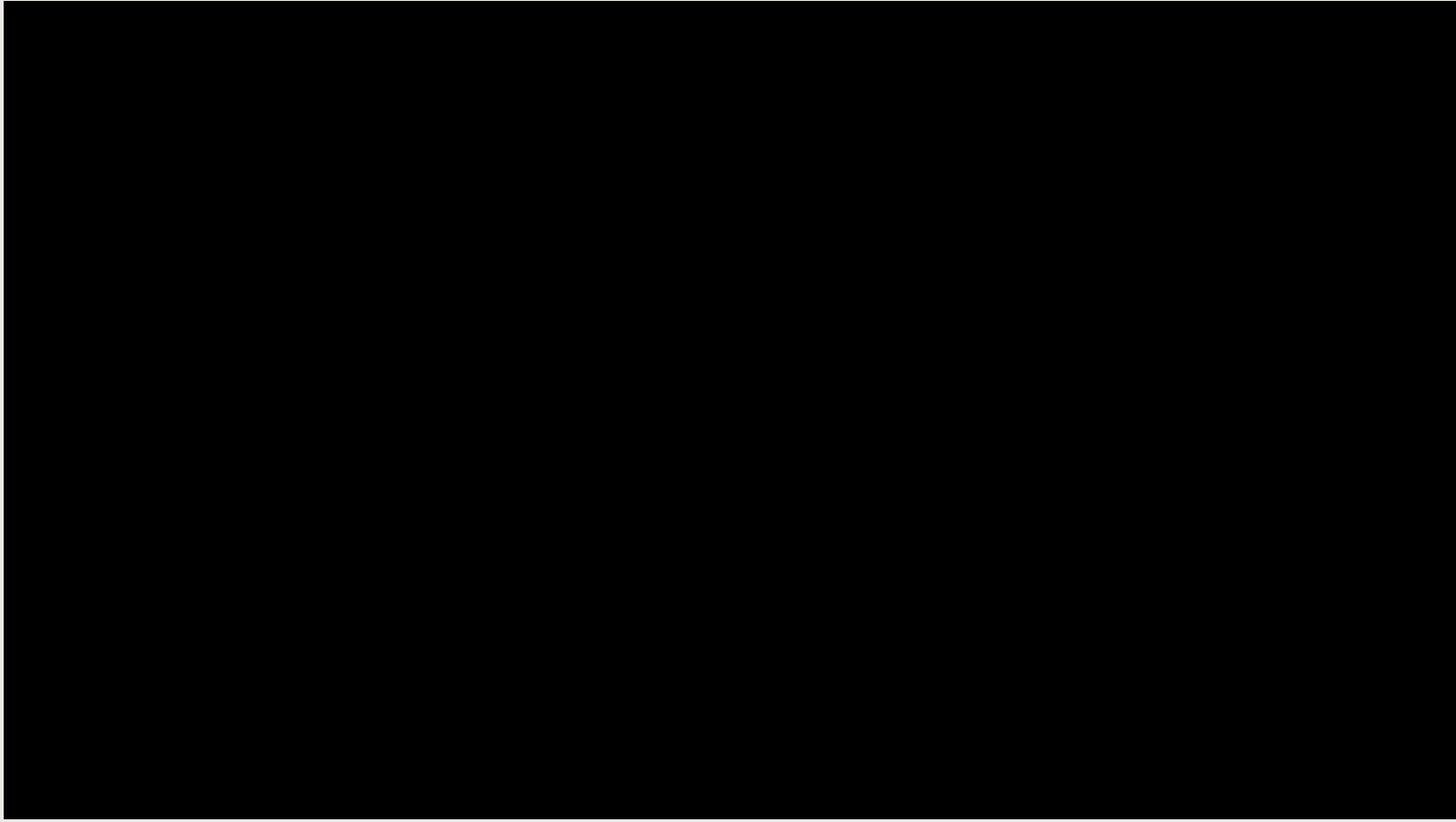
# Tools

- Debug views allow us to see what the game is doing
- Debug views help developers
- Debug tools allow us to change what the game is doing
- Debug tools help testers





# Examples of debug views and tools





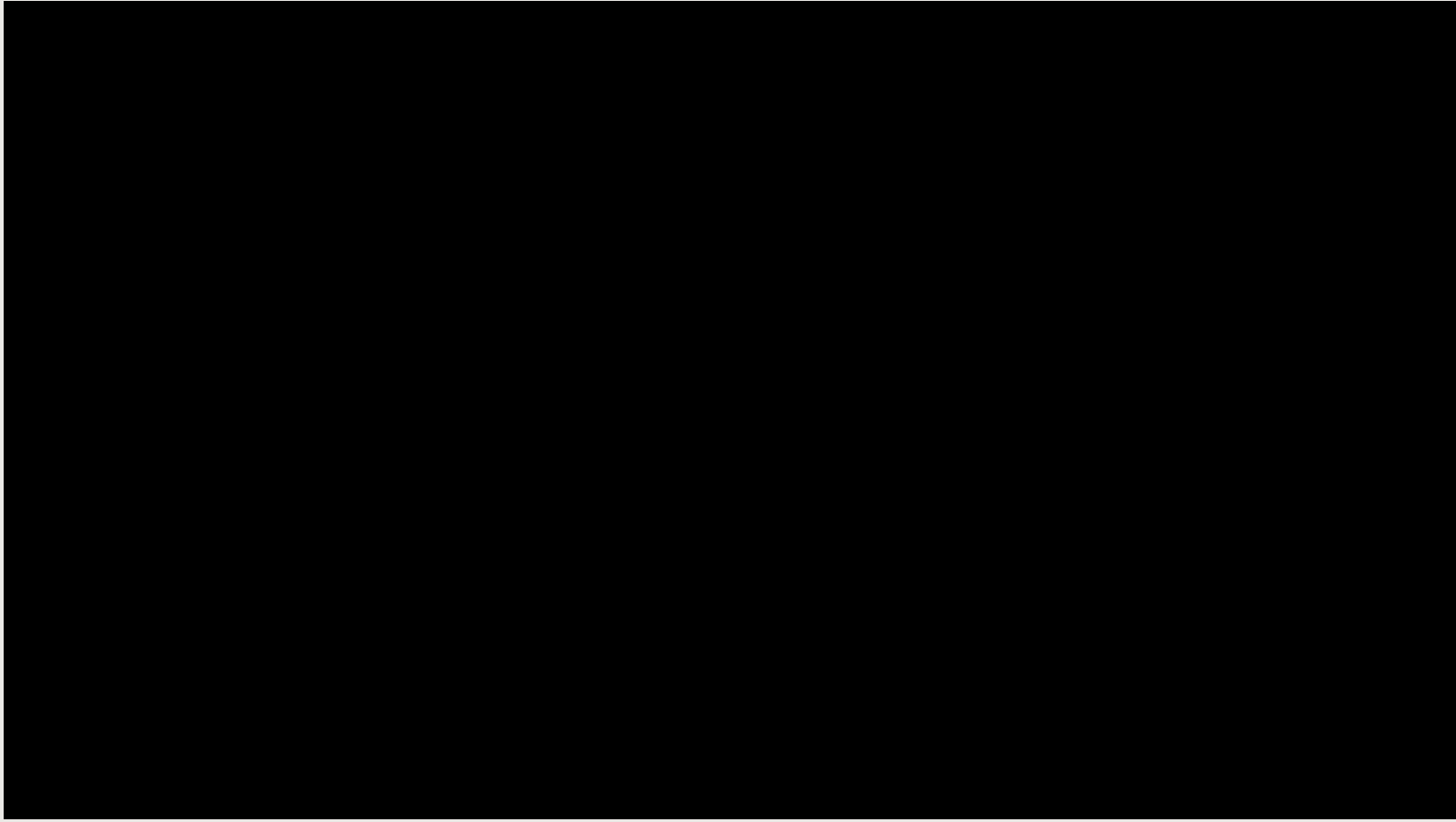
# Automation and telemetry

- Huge challenge for test automation:
  - Non-linear game
  - Open-world
  - No pre-existing automation framework
- Automated bots roaming the world
- We used telemetry to inform our testing





# Examples of automation and telemetry use





# Test Strategy

- We needed a major shift in our testing strategy
- Focus on exploratory testing
- Guided by the testers' experience and knowledge
- Risk-based
- Focused on things that changed

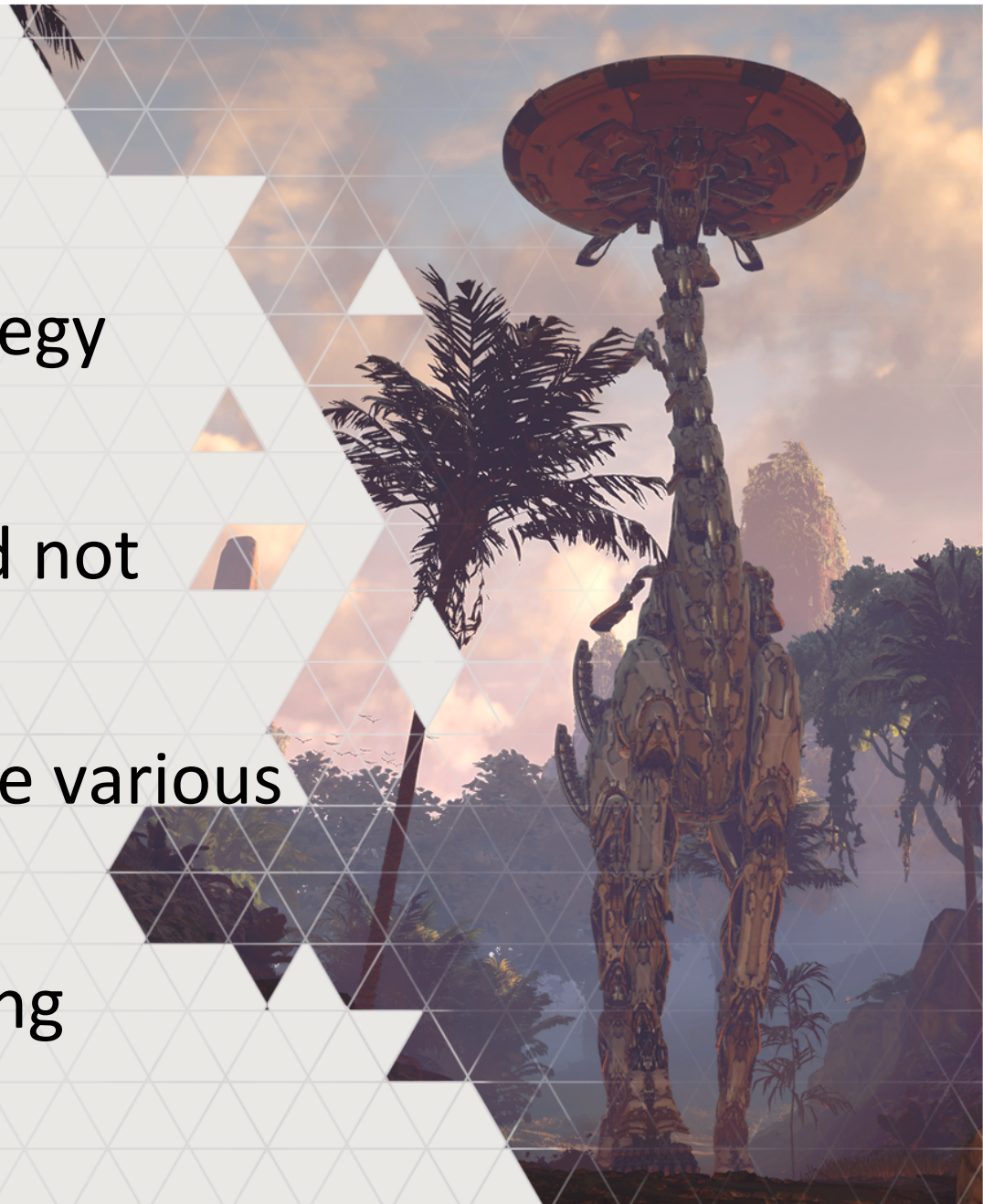






# Test Strategy

- Major revisions of our test strategy throughout the project
- Persona-based playthroughs did not work as we expected
- We adapted our approach to the various project stages
- Charter-based exploratory testing





# Why exploratory testing?

- We could not test everything
  - Need to focus
- Avoids tunnel vision in testers
- Leverages specialist testers
- The method most likely to find bugs
- Very flexible
- Does not need extensive test cases





## Example exploratory charters

- Attempting to fast travel during fights
- Backtracking during quests
- Using interactive objects while wielding a weapon
- Reloading the game during a cutscene
- Trying to kill quest characters





## Example bugs found

- The game crashes if a quest is deactivated on the map screen while player health is 0
- If the player fast-travels while throwing a rock, their weapons will be disabled
- The player can break a stealth tutorial by killing the robot with rocks
- The moon orbits in the wrong direction

