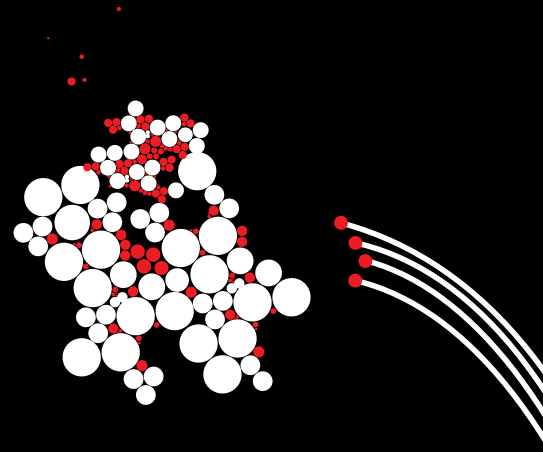
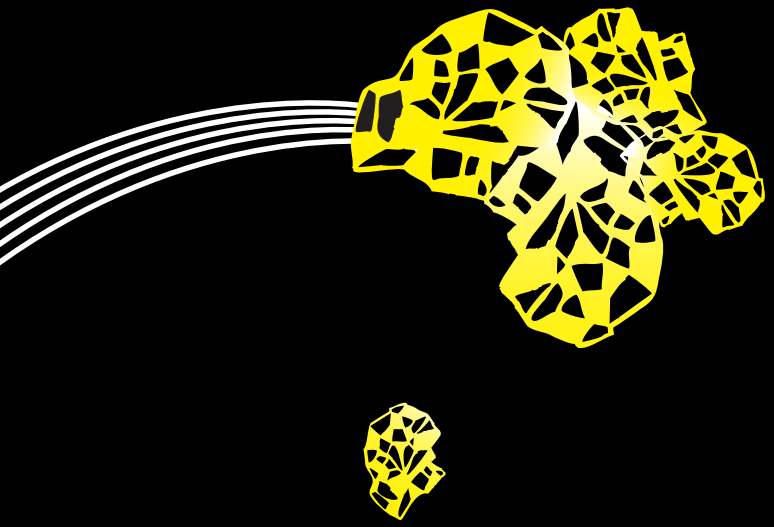
UNIVERSITY OF TWENTE.

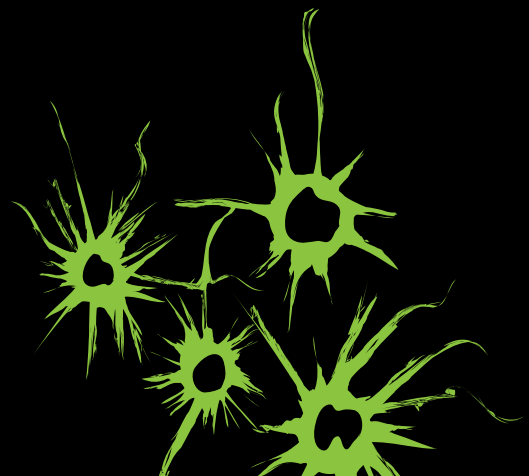


Introducing automated testing to video game development via Behaviour-Driven Development

Michael Mulder, Petra van den Bos



ECMFA 2025



Grand Theft Auto 6 delayed until May 2026

2 May 2025

Share Save

Peter Gillibrand & Tom Richardson
BBC Newsbeat

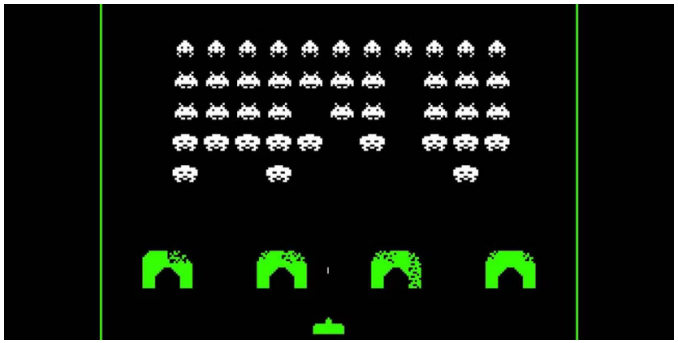
Grand Theft Auto 6 was set to be coming out this Autumn

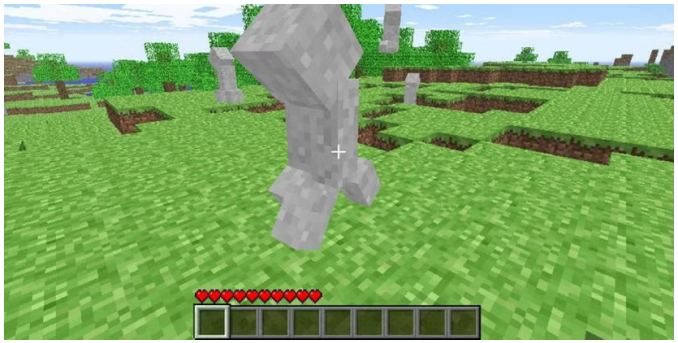
Grand Theft Auto 6 - one of the most eagerly awaited releases of all time - has been delayed until May 2026.

The sequel in developer Rockstar Games' record-breaking video game series was due to be released on consoles later this year.

But it told fans it needed extra time to "deliver at the level of quality you expect and deserve".

Bugs in games





Why care about bugs in games?

- Games are no safety-critical systems, but...
- Lots of money earned in the games-sector
  - $366,000,000,000 revenue in 2023

Current beliefs in game development

Findings from **literature review**:

- Game creation is an art, not a science
- Testing costs money and time
- If testing is applied, then *humans* should test:
  - fun-factor
  - game rule balance
  - user experience
  - emotional appeal
- however: lots of bugs found by players...

Behaviour-Driven Development (BDD)

- Agile development process
- Originates from Test-Driven Development
- Steps in BDD:
  1. Select feature from sprint
  2. Discover examples
  3. Formulate scenarios
  4. Create tests from scenarios
  5. Implement feature and execute tests
- Key aspects of BDD:
  - Communication between stakeholders
  - Natural language
  - Automation into test cases
- Broadly adopted in software engineering

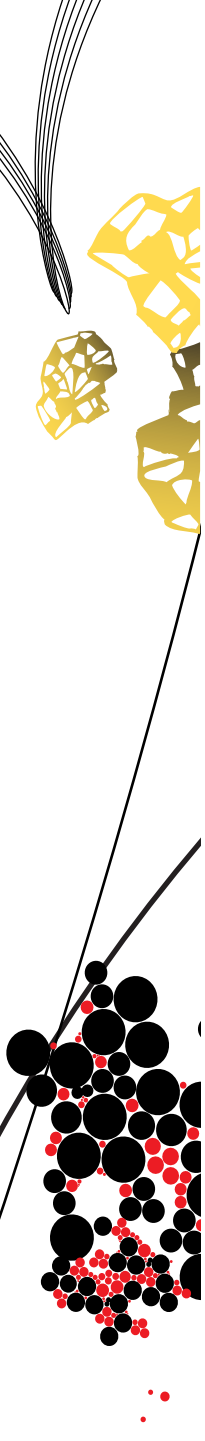
A good match: BDD and game development

- BDD focusses on behaviours instead of units of code
- BDD involves non-technical stakeholders (e.g. game artists)
- BDD could identify the bugs that are now missed

Integrate game development and BDD

Our approach:

1. Behaviour-driven game development **process**
2. BDD test-automation **tool**: UnitySpec
3. Game behaviour **framework**
4. **Case study**



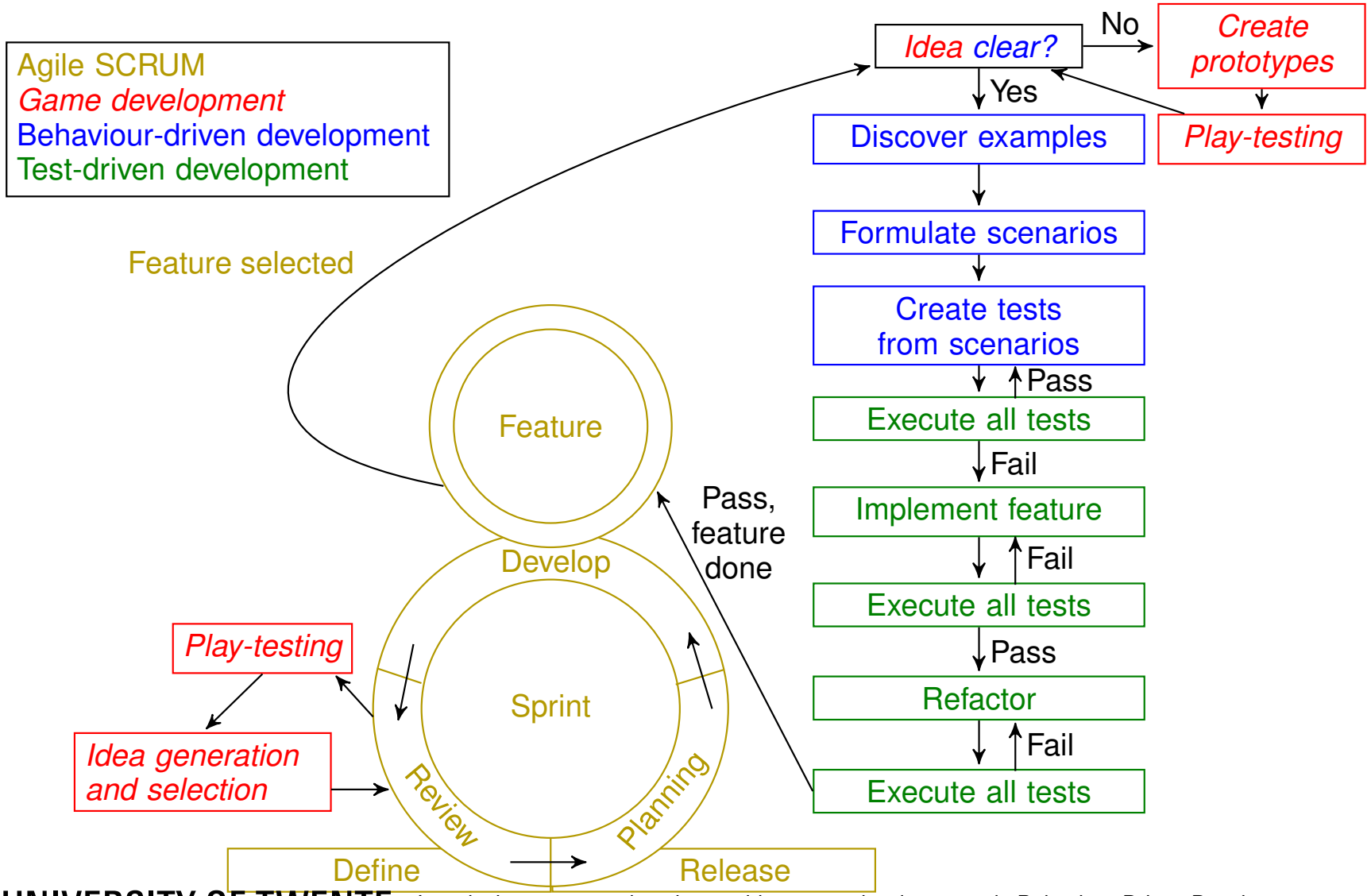
1. Behaviour-driven game development process

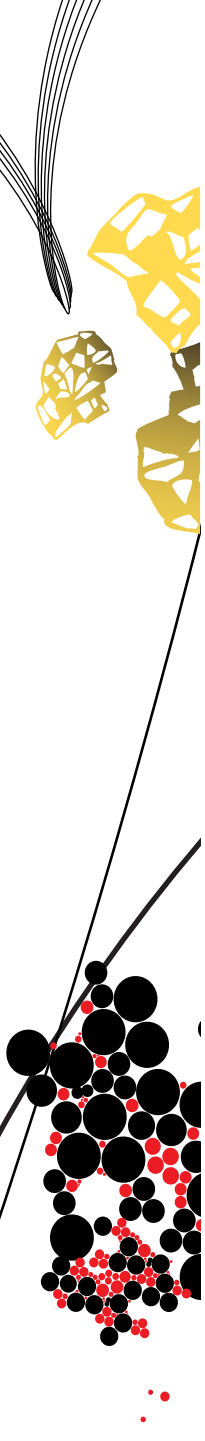
Unique to game development

Findings from literature review:

- **Experimentation**, **idea generation**, and **prototyping**
- Test fun-factor and user experience via **play-testing**
- Lack of attention for functionality

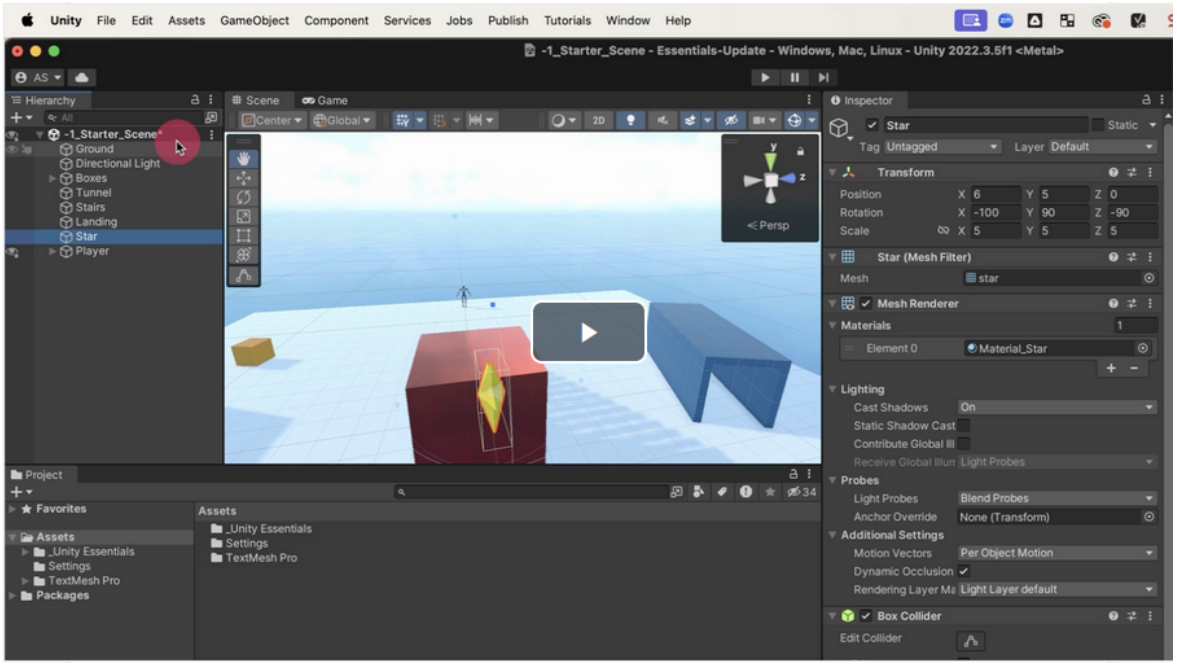
Behaviour-Driven Game development process





2. BDD test-automation tool: UnitySpec

Unity 3D



- Platform to create 3D games
- Physics engine
- Real-time testing
- Scripting in C#
- Documentation, tutorials

SpecFlow

- Write scenarios in Given-When-Then style

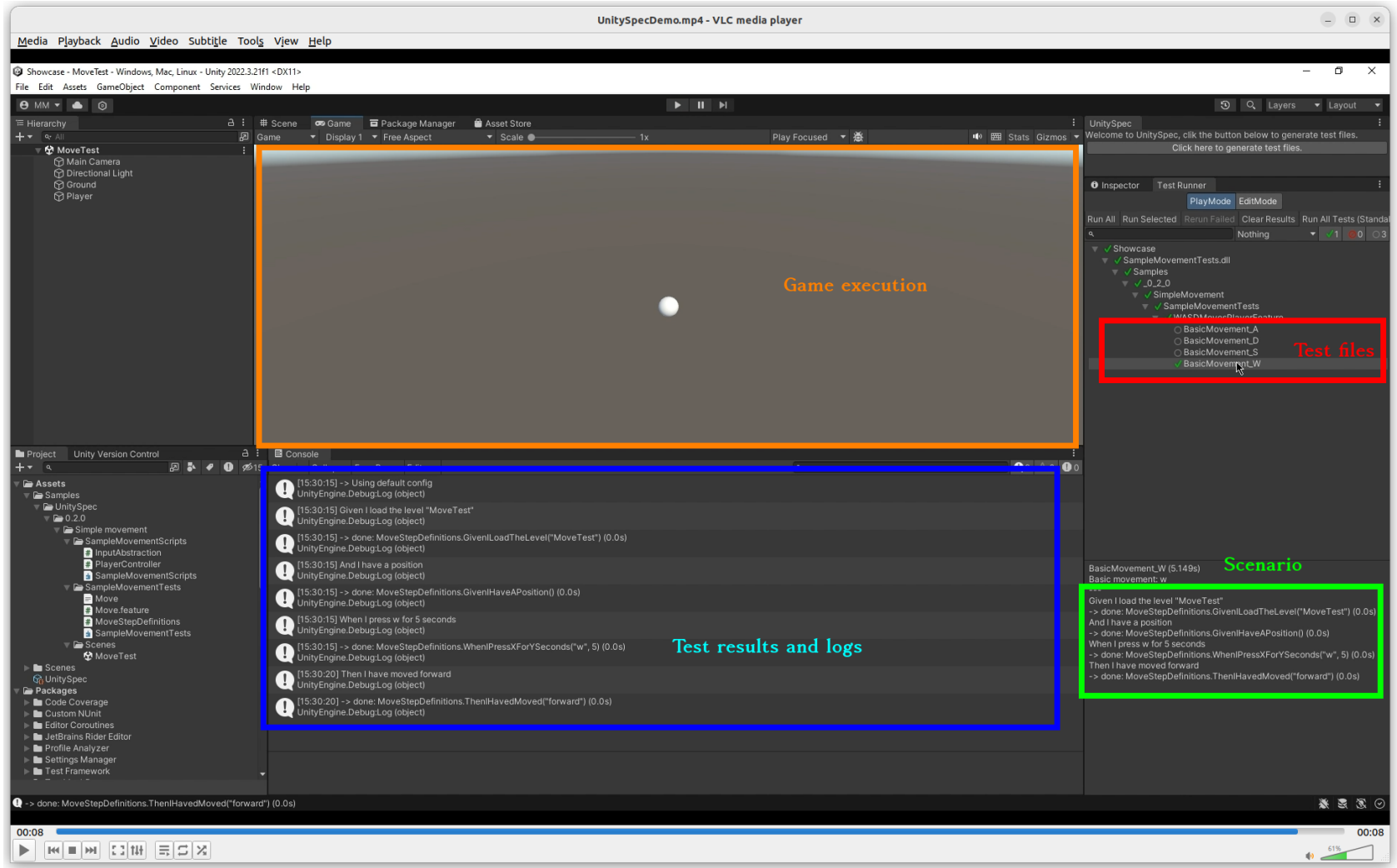
```
Feature: WASD moves player
  As a user I want to be able to move using wasd-keys

Scenario Outline: Basic movement
    Given I load the level "MoveTest"
    And I have a position
    When I press <key> for 1 second
    Then I have moved <direction>

Scenarios:
      | key   | direction |
      | w     | forward   |
      | a     | left      |
      | s     | backward  |
      | d     | right     |
```

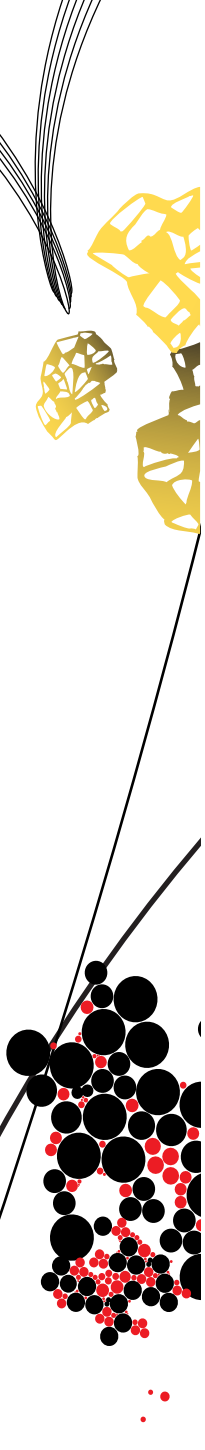
- Implement generated step definition methods

```
[Given(@"I load the level ""(.*)""")]
public void GivenILoadTheLevel(string level) {
  \\ provide implementation here    }
```

- Run scenarios as test cases

UnitySpec


UnitySpecDemo.mp4 - VLC media player
Game execution
Test files
Test results and logs
Scenario



3. Game behaviour framework

Quality of BDD scenarios

- Depends on writer of scenarios
- But also on: right level of abstraction
- Game Behaviour Framework:
  - defines levels for game behaviours

How framework was obtained

1. Identify behaviours in example game projects
2. Extract abstract behaviours
3. Categorize abstract behaviours to find levels

Levels of game behaviours

- Level 1: Basic controls

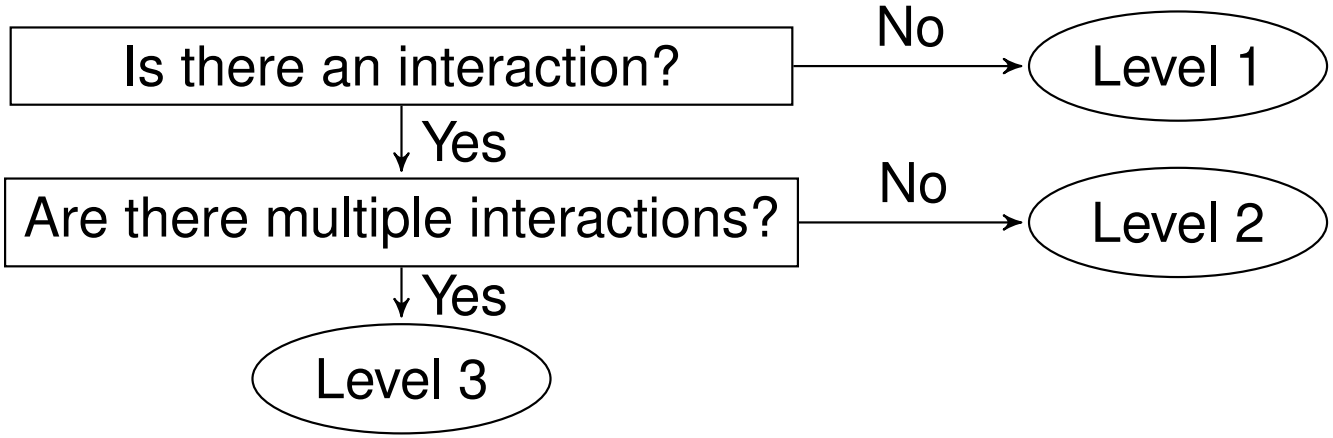
  e.g. player movement, scene load, menu navigation, NPC movement
- Level 2: Singular interactions

  e.g. collision, shooting, location-based trigger
- Level 3: Game elements: with multiple objects during multiple frames

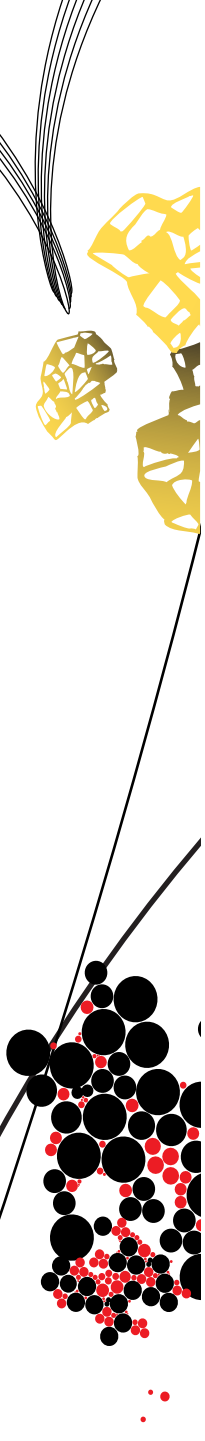
  e.g. full-fletched NPC behaviour, fight, objective completion
- No level ≥ 4

Identify level of game behaviour

1. Check interactions:



2. Inspect list of abstract behaviours
3. Inspect list of concrete behaviours



4. Case study

Case study: Virtual brigade

- Serious game
- Training scenarios: e.g. border control, securing an event
- Game scenarios configurable via Unity 3D
- Game under development

Identified levels of behaviours in case study

- Requirements from roadmap
- Selection of results:

Requirement	**Level**
The visual world for an airport exists	Level 1: Booting / Asset Exists
The player can enter an immigration desk and take place	Level 2: Special Control
People can queue and walk towards the player one by one	Level 3
There is a visually appealing menu structure	Not applicable

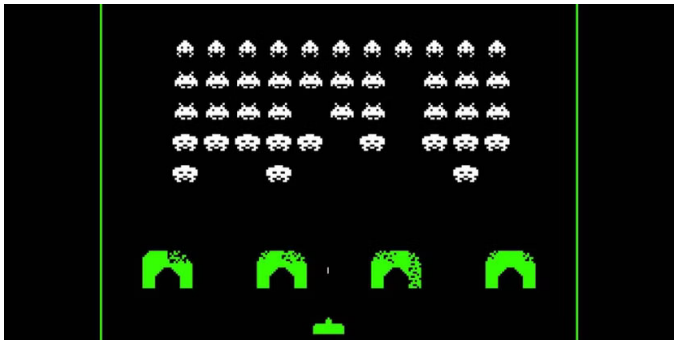
Method for applying process and tool

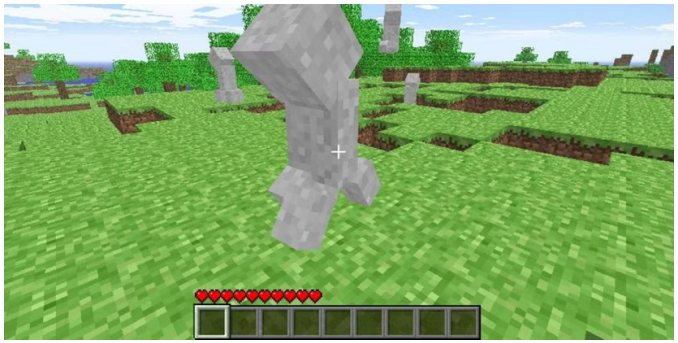
- Identify stakeholders in team
- Select feature
- Explain process and tool to team
- Observe and support team
- Interview stakeholders

Interview results

- Meetings ensured shared understanding
  - Ambiguity resolved into clarity
  - Increased confidence in feature implementation
- UnitySpec used without issues
  - But step definitions not successfully implemented
- Process valued: increase efficiency and reduce errors
  - Especially for new products in large teams
- Learned lessons
  - Invest time in on-boarding
  - Adopt process from start of development

Bugs in games





Conclusions & future work

- Contribution: integration of BDD and game development
  - Behaviour-Driven Game development process
  - BDD test-automation tool: UnitySpec
  - Game behaviour framework
  - Case study

- Future work:
  - Extend game behaviour framework with more examples
  - Apply BDD-game process over longer period
  - Case study with entertainment game

Thank you for listening!

Questions?