Console Development in 15 minutes

- all you need to know to look like a Pro

Overview

- Different hardware platforms
  - PC, XBox, PS3
  - Each have its own unique properties
- Consoles are weaker that PC
  - But they have controllers!
- Dev Kits used for development:
  - Debug hardware specific problems

Memory

- Whatever you do, think of memory
- Avoid leaks (**everyone** fails that)
- Know your memory management system
- Avoid memory fragmentation
- Never use global scope new
- Sometimes useful to un-inline functions
- Careful with templates
- Always use same allocator to delete stuff
- Stack size is limited too
- Use memory analysis tools available

Performance

- Always think of the performance
- Understand usage of the code that you write
- Avoid extra nested loops
- Avoid extra pointer dereferencing
- Avoid cache misses
- Sometimes useful to handcraft/unroll loops
- Sometimes useful to inline functions
- Use performance analysis tools available

Teamwork

- You must know and follow your processes
- Simple code > smart code
- Keep relevant people in loop
- Code reviews are mandatory
- No build breaks
- No major functionality breaks

TRC Specifics

- TRC stands for Technical Requirements Checklist:
  - Failing to meet TRC cost a lot of money;
  - It is better to pass first party TRC check at once;
- Be aware of TRC violations:
  - due to unresponsiveness;
  - due to controller unplugs;
  - to screen space allocation;
  - due to network connectivity loss;
- Can use game restart as last resort

Package Development Specifics

- Pay attention to your interface
- Design clean interfaces, use PIMPL
- Maintain ABI
- Try to use PODs in your interfaces

Screen Development Specifics

- EA Projects are using Flash-like engine for UI, its called APT:
- It have some constraints:
  - Class usage is limited and could lead to ZOMBIES;
  - Zombie is a dead class that corrupt UI behavior;
  - Once occurred zombies will lead to complete game unusability;
- There are 3 stage pipeline: make SFW -> build BIG -> deploy on console
- For each platform there is a separate BIG
- Remember to rebuild enums / screens

General Coding Specifics

- Try to reuse code,EA has most of tools ready
- Know and use EASTL:
  - It is quite good comments and documentation
- No globals, minimize statics
- No magic numbers
- Defensive programming, assert a lot
- Use includes, namespaces correctly