Verified Gaming

Joe Kiniry
KindSoftware Research Group
Systems Research Group
CASL: Complex & Adaptive Systems Laboratory
University College Dublin
Dublin, Ireland
for the most part, we will not be the ones solving the Grand Challenges

interest in computer science is low

any mention of mathematics frightens off most students

how do we get new generations of students interested in our (sub)field?
Verified Gaming

- use computer gaming as a “hook”
- introduce complex topics through secret ninja formal methods
  - see Thursday’s talk in the main symposium for details about our ninja techniques
- use running systems as specifications
- provide examples of the use of verification in the gaming industry
Verification in Software Engineering

- the teaching context for this work
- 1st—3rd year software engineering project modules at UCD in Ireland
- individual vs. group work
- summative and formative feedback
- concepts, tools, and technologies covered include everything from domain analysis through reasoning about implementations
Past Projects

- The Guinness Screensaver
  - your display sleeps, a pint is pulled for you
- The Computer Simulator
  - make the abstract concrete by simulating subsystems of their own design
- Flow
  - aesthetic game with minimal I/O and GUI
- 1D cellular automaton simulator
This Year’s Project:
C=64 Game “Thrust”

- the video game “Thrust”
- classic but not well-known C=64 game

motivations for project choice
- students do not understand or appreciate the resources they have at their disposal

system decomposition
- simple I/O, persistence, GUI, discrete event simulation, physics, domain analysis
Verification via Process and Tools

- a formal methods-rich process, with no formality
- (concept) analysis
- (formal, contract-based) design
- (refinement as a cut-and-paste) implementation
- (scenario- and contract-driven) validation testing
- verification (via static checking)
metrics is the simplest motivator

- textual I/O and the logging-based interface
- formative and summative feedback to students via static checkers
- manual system and automated unit testing
- reflections on validation and verification practices of students
Current State

- the world's first verified game?
- a “gold standard” case study in a verification-centric process in JML+Java
- future project ideas
  - reuse of past C=64 projects
  - popular board games
  - classic console games (e.g., Space Invaders)
- pedagogical resources
  - The UCD CSI Trac—http://csi-trac.ucd.ie/
Course Corrections

- individual vs. team projects
- tutorials on each static checker
- more/better lab support for students
- regular interim evaluations
- integrating automated grading
- cover fewer concepts and tools (we now cover about a dozen core topics of SE)
- produce working solution throughout the term from which students can work