"Software Architectures and the Creative Processes in Game Development"

Written by Alf Inge Wang and Njål Nordmark
Presented by Markus Bøyum Johansen and William Gjerberg Tresselt

Introduction

- Game dev changes rapidly
- Before:
  - Small teams
  - Simple software architecture
- Now:
  - Requires a multitude of different professions
  - Involvement of so different professions poses challenges for the development
  - Complicated software architecture
- Present what characterizes game dev

Research goal, Questions and Methods

- Goal: "examine how software architecture is used and how creative processes are managed from the POV of a game developer in the context of video game development"
- Research questions:
  1. what role does the software architecture play in game development
  2. how do game developers manage changes to the software architecture
  3. how are creative development processes managed and supported
  4. how has game development evolved the last couple of years
- Mix of qualitative and quantitative data

Related work

- Kanode and Haddad identified challenges in game dev
- Main emphasis on requirement engineering, coding tools and techniques
- More plagued by scoping challenges, delayed schedules, budgeting errors etc than traditional software development
- Game dev industry has adopted some agile methods

Results RQ1

ID	Statement	Agree	Neutral	Disagree	N/A
Q1	Design of software architecture is an important part of our game development process	69%	15%	8%	8%
Q2	The main goal of our software architecture is performance	54%	15%	23%	8%
Q3	Our game concept heavily influences the software architecture	69%	8%	15%	8%
Q4	The creative team is included in the design of the software architecture	69%	15%	8%	8%
Q5	Our existing software suite provides features aimed at helping the creative team do their job	92%	8%	0%	0%
Q6	Our existing software architecture dictates the future game concepts we can develop	15%	47%	38%	0%

Results RQ2

ID	Statement	Agree	Neutral	Disagree	N/A
Q7	The creative team has to adopt their ideas to the existing game engine	31%	46%	23%	0%
Q8	During development, the creative team can demand changes to the software architecture	69%	31%	0%	0%
Q9	The technical team implements all features requested by the creative team	69%	15%	8%	8%
Q10	It is easy to add new gameplay elements after the core of our game engine has been completed	70%	15%	0%	15%
Q11	During development, the creative team has to use the tools and features already available	47%	15%	38%	0%

Technical team	Management	Creative team
10%	40%	50%

Results RQ3

ID	Statement	Agree	Neutral	Disagree	N/A
Q13	Our game engine supports dynamic loading of new content	92%	8%	0%	0%
Q14	Our game engine has a scripting system the creative team can use to try out and implement new ideas	70%	15%	15%	0%
Q15	The creative team is included in our development feedback loop (e.g., scrum meetings)	86%	8%	0%	8%
Q16	Our game engine allows rapid prototyping of new levels, scenarios, and NPC's/behavior	86%	8%	0%	8%

Results RQ4

ID	Statement	Agree	Neutral	Disagree	N/A
Q17	Today our company uses more 3rd-party modules than 3 years ago	46%	15%	8%	31%
Q18	It is easier to develop games today than it was 5 years ago	77%	8%	15%	0%
Q19	Middleware is more important to our company today than 3 years ago	55%	15%	15%	15%
Q20	Game development is more like ordinary software development today than 5 years ago	38%	24%	38%	0%

Conclusion

- RQ1: software architecture in game dev
  - Performance, availability, security, and modifiability
  - Game concept influences choice of game engine
  - Creative team
  - Future game concepts
- RQ2: changes to software architecture
  - Change-request decisions
  - Implementing features with scripting and emergence
- RQ3: supporting the creative process
  - Creative processes enabled through support of GUI tools, scripting, dynamic and loading of elements
- RQ4: changes over time
  - Tools simplifies process, but player expectations increase technical complexity
  - As conventional software development?