

An Introduction to Agile Game Development with Scrum, XP and Lean
An IGDA Webinar by Clinton Keith
September 29, 2009

Introduction by Joshua Caulfield, IGDA Executive Director

Joshua welcomed everyone to the first Webinar of the IGDA. The Webinar series is a new way in which the IGDA is adding value to it's membership. He is hoping to have a new speaker for the series every 2-3 weeks. If you are interested in becoming a speaker or if you need to contact Joshua, his e-mail is: joshua@igda.org

The slideshow for this Webinar were available to attendees at [Clinton Keith's Website](http://clintonkeith.com/): <http://clintonkeith.com/>. However, many of the slides can be found in the slides from his GDC Keynote speech found [here](http://www.agilegamedevelopment.com/2009/03/gdc-slides.html): <http://www.agilegamedevelopment.com/2009/03/gdc-slides.html>

What is Agile and Why Use It

1. Agile is an approach for developing products using short iterations and using the results of each iteration to adjust the project plan
 - Each iteration is like a short project in itself
 - Uses "inspect and adapt" practices to adjust the goals and measure progress
2. Agile Manifesto
 - Value the left over the right
 - People and Communication over Process and Tools
 - Working Game over Design Documentation--*A working game is more honest and tells us more than the design document possibly can.*
 - Customer Collaboration over Contract Negotiations--*Get publishers on your side! Keep the customers in the loop and part of the ongoing process.*
 - Responding to Change over Following a Plan--*Be free to change and improve on things and to take advantage of cool things as they present themselves. Scrum should be fluid.*
 - [Website for Agile Manifesto](http://www.agilemanifesto.org/): <http://www.agilemanifesto.org/>

Why Adopt Agile?

1. Reduce wasted effort and crunch times.
2. Find the Fun First!--*You can't tell from a design document which elements are fun or really not fun. Play as you go.*
3. Reduce Waste
 - Discover what works and what's not working before mass producing an item--make assets in the present and use them before mass producing them. If something doesn't work, you will only be wasting

one, not 12 or more. If something else works better, you'll be able to take advantage of that discovery without a lot of extra effort.

- As much as possible, don't postpone value, debugging and optimization until the end. Issues such as debugging become a huge file of fixes at the end of a project if not dealt with in the present time. Keeping the work and the problems transparent is part of Scrum.

4. Let value, risk, cost and knowledge drive the order of what we work on and not a preordained schedule.

Overview of Scrum

1. Sprint

- 2-4 weeks long
- Daily Meetings with Scrum team to discuss what's working and what is not.
- The deadline is crucial, known by all members of the team and must always be met. Always deliver. The deadline is sacred.
- Result of Sprints is an incrementally improved game. The publisher can see and modify the game at the end of sprints, which reduces waste. You also have deliverables for the publisher every 2-4 weeks which keeps him working with you as a member of the team.

2. Jump

- The goal of this Sprint--*What needs to be accomplished in all departments in the next 2-4 weeks.*

3. Sprint Backlog

- A list of the tasks that need to be completed in order to reach the goal of the sprint.

4. Sprint Team

- Programmer
- Artist
- QA Tester
- Designer
- Animator
- Scrum Master--Not an authority role. Someone facilitating the team and helping them to overcome impediments.
- Product Owner--Lead Design, Publisher, Producer. Dream Holder of the project, who goes between the developer and the publisher.

5. Sacred Deadline

- The deadline is functional. If, for instance, something has caused an impediment, the team may put in a mini-crunch in the 2-4 week sprint which saves a much bigger crunch later on. Also, if something is not working or is much a bigger task than anticipated, it can be removed from the sprint backlog for a later sprint. This is what keeps Scrum fluid and transparent and that is what is at the heart of Scrum.

- You must have a potentially demo-able/playable game at the end of each sprint.
- There is a reciprocal commitment in each sprint: The team commits to delivering some amount of functionality and the publisher commits to leaving priorities alone during the sprint.

6. The Task Board

- User Stories--*The task is written in a single sentence with the end user in mind. What does the end user want or need? This is what the team commits to provide.*
- Tasks/Sprint Backlog
- Burndown Chart--*Found in the Sprint Backlog section of the Task Board. This is where impediments go. It allows the team to react to things which are not working. A postmortem at the end of every sprint works on the impediments that were dealt with during the sprint. Did the solution work? Did we learn something new for going forward or was it a fix only for that particular project? This is also where you decide if you need to increase hours to complete a task before the deadline, so it works in real time.*
- Completed Tasks

7. Scrum of Scrums

- The leadership team
- A representative from each team meets weekly to discuss impediments and issues. The Scrum of Scrums helps keep the project flowing smoothly.
- "A project doesn't become a year late overnight....it loses hours a day" - Fred Brooks

XP (Extreme Programming)

1. Includes test driver development (TDD) and Pair Programming Practices.
 - Code is cheaper to fix if buggy IF you catch it *right away*, while everyone remembers what they did and why.
2. Pair Programming is continuous peer review and brainstorming.
 - 1 PC, 1 Screen
 - Mentoring--a newer programmer works alongside a more experienced one. Newbie learns and might show experienced a few new tricks.
 - Consistent Standards
 - Resource Sharing--Every piece of programming has 2 programmers who know the code.

Traditional Vs. Agile Planning

1. The problem with traditional planning is that you miss the boat when you arrive at the end result only to find out that it's not right in some way and now you have to crunch to fix it!
2. "Plans are Nothing. Planning is Everything"--Dwight Eisenhower

A Project is a Series of Releases

1. 3 month releases, generally
 - Concept/Green Light
 - Pre-Production
 - Production
 - Alpha and Beta Releases

Lean

1. Stages
 - Iterate
 - Incremental improvements
 - There is a difference in how you look at your project during pre-production and production.
2. Pre-Production
 - Questions: What is fun? How will we build it?
 - Goal: Correctness
 - State of Mind: Collaboration
 - Approach: Iterative and Incremental
3. Production
 - Questions: Build 10 Hours of it!
 - Goal: Efficiency
 - State of Mind: Flow
 - Approach: Incremental
4. Lean game production is a translation of lean manufacturing principles and practices to video game asset production.
 - In game production: Concept->Low Res Geometry->High Res Geometry->Audio Layout->Gameplay Tuning
5. Uses the "Deming Cycle": Plan, Do, Check, Act.
6. 7 Lean Principles
 - Eliminate Waste
 - Amplify Learning
 - Decide as late as possible--Minimize changes by introducing things in real time instead of doing many builds before introducing them and finding out something doesn't work or isn't needed.
 - Deliver as fast as possible
 - Empower the Team
 - Build Integrity In--Balance Disciplines
 - See the Whole
7. Clinton Kieth mentioned a whitepaper on the subject, but I don't have a link to that. Sorry folks.

The Myths

1. Scrum with prevent problems or solve them for you
 - Reality: Scrum is transparent. It will reveal problems that need solving.

2. Scrum can achieve impossible goals.
3. Scrum doesn't require competent leadership.

Adopting Agile

1. Have a beachhead team.
 - A team designated to use Scrum and pave the way for the rest of the team later.
2. Try it out "by the book".
3. Iterate alongside your current way of working.
4. Use a coach to help you out!

Clinton went on to answer questions for a half hour after the seminar. He is leading a Scrum Certification Course at this years IGDA Leadership Forum November 10-13, 2009 at the San Francisco Airport Marriott. For more information on this event and to sign up, please go to <http://www.igda.org/leadership/>.