Postmortem: Housemarque's Outland

By Ilari Kuittinen,Aki Raula

[*Housemarque's Aki Raula and Ilari Kuittinen delve into the difficulties the team faced when developing its first 2D adventure platformer, Outland, including how a change of core gameplay introduced difficulties mid-production.*]

Housemarque as a company has been around for over 15 years. We started as a PC developer (our first game was a version of the Commodore Amiga's *Super Stardust* on PC), produced two snowboarding games for PC and Xbox, before moving onto the current generation and its downloadable game space.

Our best-known titles include the PSN hits *Super Stardust HD* and *Dead Nation*. *Outland* is our first cross-platform title and we are continuing on that path, while not forgetting our great relationship with Sony (e.g. *Super Stardust Delta* for the PlayStation Vita).

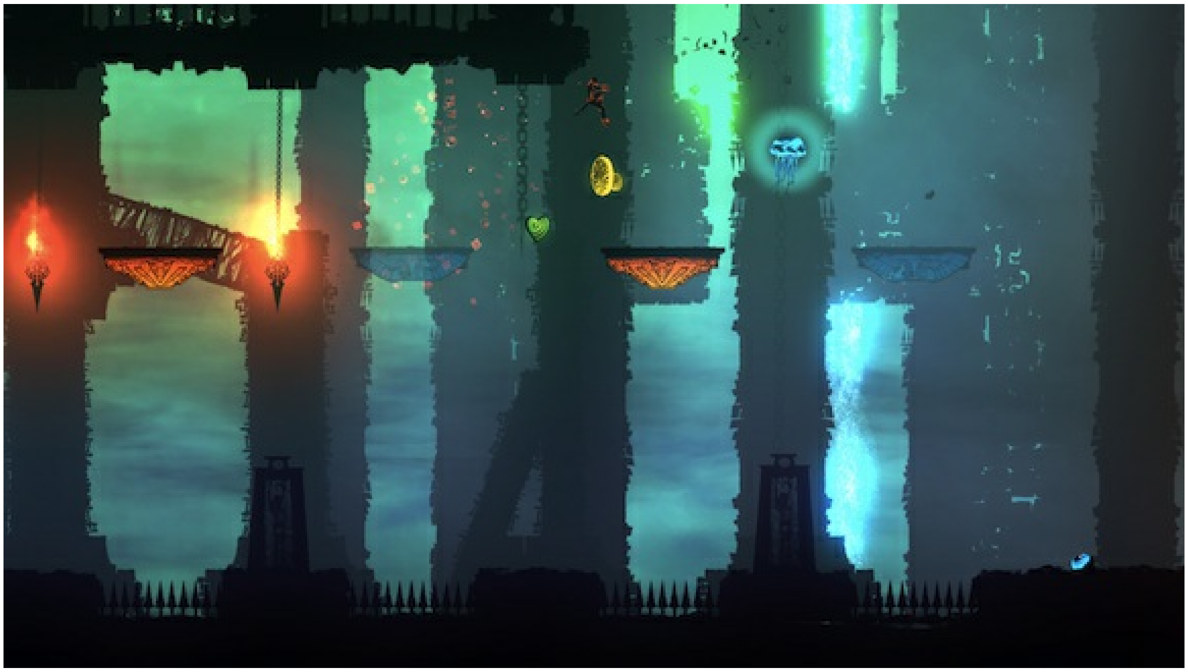
When we originally put together the concept for *Outland*, the aim was to make a 21st century version of the old 8-bit platforming adventure games. Initially the idea was to take titles like *Pitfall II* (Atari 2600 et al.) and *Rick Dangerous* (Commodore Amiga) to a whole new era.

The main character was to be an Indiana Jones-like adventurer, roaming around jungles in search for ancient treasure and truth. Key emphasis was on the physicality of the world, and not yet on the two polar-opposite energies and inhuman agility we ended up with.

For the first six months after signing a contract with Ubisoft, the main character had a pistol but no melee abilities. He was also able to slide slopes and perform longer jumps after gathering speed from those slides. From quite early on we added a special spirit mode where one could do considerably more than in the normal mode: execute heavy melee attacks and run at faster speed, accompanied with appropriate visual effects. This mode was the stepping stone for the final game's spirit-switching mechanic.

Around nine months into preproduction we started to suffer from a lack of direction and vision; originally the concept was more about old school adventure and puzzle solving. It then moved towards "*Sonic* meets Indiana Jones" before settling to agile platforming and *Ikaruga* mechanics.

What added to the lack of vision came partially from restrictions our engine had early on in the project -- we couldn't really do cool physics based gameplay that we had envisioned. As a result, we tried to reinvent the game several times during those first nine months of development.



Even if our journey took longer than originally planned, we ended up with a game that (in our opinion) successfully marries two genres into one. While we lack in some gameplay areas, such as enemies, world structure, and collectibles, we excel in others, such as the tight controls. This is a fact that could not have been achieved without the tremendous character animation by Mr. Tomi Kokki. Ubisoft also made sure we had the bar set high on the (character) animation side, and we think we definitely delivered on that front. This fact can also be seen in many of the reviews that praise the character animation and smoothness of movement.

Overall, we are very taken with and pleased by the reviews and reception the game has gotten. Outland received Editor's Choice nods from IGN and GameSpot, to mention a few. As with many other postmortems, it is easy to imagine that a game with great reviews must have had a carefree development process, but in most cases this isn't true.

What Went Right

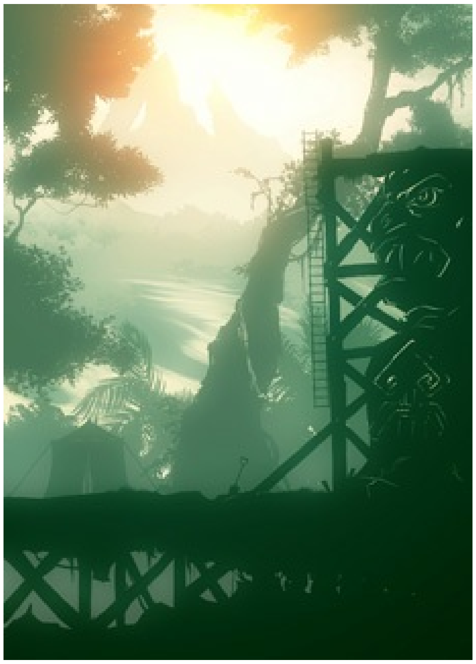
1. Visuals

Aside from the early experiments done during the conception phase, the project had a clear and unique design for the visuals. From early concepts -- six months before we had anything running on-screen -- to the final ones, that visual design stayed consistent.

The key idea from the get-go was for the in-game art to match the concept images 100 percent. Most games have a huge gap between what the concepts look like and what the actual game ends up looking like -- our goal was to make the game look like concept art running in real time.

Visuals ended up our strongest suit, and the most recognizable aspect of the game (along with the polarity concept). We have received a lot of praise from the press, as well as gamers, for our unique looks. Some people draw comparisons to games like *Limbo*, *Braid*, or *Insanely Twisted Shadow Planet*; while we were aware of these games, none of them really acted as inspiration, as our looks were defined before we stumbled upon those masterpieces.

In any case, we all give a big hand to our lead concept artist, Mikko Eerola, for staying true to his vision from the start to the finish, even if the road wasn't always easy.



2. Genre Merger

Working on a platformer that tries to push the genre forward can be tough without a solid breakthrough idea. While we could have had a great game had we realized the original concept for *Outland*, we believe the addition of the polarity switching component truly brought the game onto a whole new level of uniqueness. Maintaining the action-adventure aspect was always one key direction for us -- polar gameplay took that concept and added a twist, while maintaining the overall idea beautifully.

Once we agreed on going towards this direction, it was relatively easy to get the basic gameplay going. After just a few weeks of development we had prototypes running that proved we had made the right choice. From there on it was up to the level designers to do their magic and tap into the wealth of ideas; the concept is such a rich source for gameplay scenarios that we believe we only scratched the surface with *Outland*.

---

3. Character Animation and Controls

When we started the project, we hadn't done humanoid characters in a while. Our skills had gone a bit rusty and we had to relearn how to model and rig realistic humanoids, not to mention how to animate them and bring them to current-gen standards.

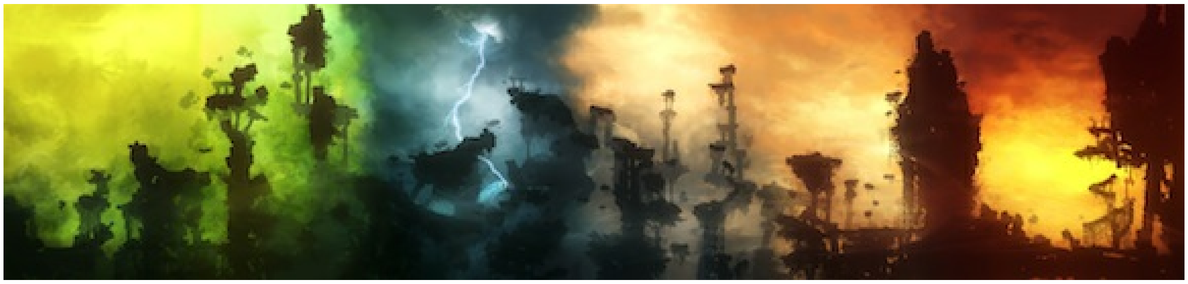
Luckily the studio was working on *Dead Nation* (for PS3/PSN) at the same time and that allowed us to put a strong emphasis on biped animation (*Dead Nation* is a zombie game with dozens of different humanoid characters). Thus, developing biped characters for it automatically helped *Outland* as well.

On top of that, the key character animator on both projects was the same person, which further added to the overall quality of animation; the animation styles in the two projects differ from each other but both still rely on realistically-moving biped characters.

Of course, working with Ubisoft, which is traditionally a very animation-strong company, our initial bar was set high on character movement and responsiveness. Fortunately Ubisoft Montréal animators were able to provide us feedback on the animation and our producers pushed (main) character movement as one of the key development areas for

Outland.

In the end all the hard work really shows in the game. We have received praise from numerous members of media and gamers alike on the high quality of our animation.



The color design for Outland (click for large version)

Apart from animation and actual gameplay, our biggest worry was how the added *Ikaruga* mechanism would work with the controls. But, as we only had to add the polarity switching to our already =0working platforming controls, the problem was smaller than we originally thought. One might say the problem solved itself through its simplicity.

4. Boss Fights

Outland is divided into a tutorial section followed by five gameplay chapters. This division gave us a straightforward goal for building five end-level bosses, or "guardians", as we call them. When we revealed the game at PAX in September 2010, we had only one near-final boss in the game -- all the others were planned to some level of detail but did not yet exist in a playable form.

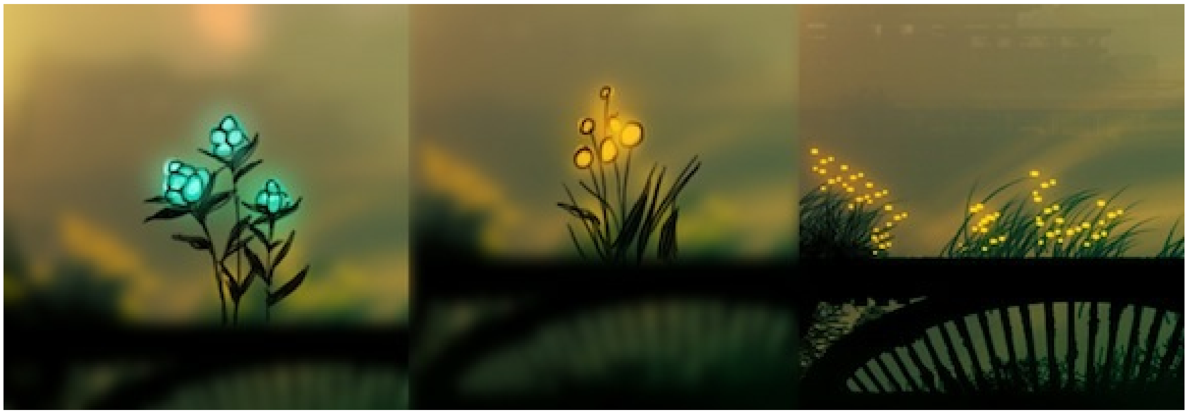
Our design goal with the bosses was to build enemies that were different from the rest of the gameplay and physically larger than the player character, thus adding to the epic scale and atmosphere. Building the first boss probably took the longest, but also came together the easiest. The rest we struggled with, more or less.

The biggest hurdle we had was with the final boss. We were closing in on our beta deadline and still didn't have the final design for the boss. We knew we didn't have the resources to create a complex character encounter with multiple animations and states, and thus we opted for something more metaphysical -- a boss in spirit rather than in flesh. The end result is a mix of platforming and hard core bullet gameplay spiced with fighting (attacking, really) the end boss(es).

Unlike the rest of the gameplay, the actual work on the bosses fell on the programmers. After we hit our alpha stage, we still had a long way to go to have final bosses in the game. Naturally the programmers were really busy with hundreds of other items the game needed, so they really pushed a lot to get the bosses done. Even if we could have done more testing and difficulty balancing on the bosses, we are happy with the end result; *Outland* has the five unique, epic bosses we set out to create.

5. Atmosphere/Audio

A huge part of the atmosphere naturally came from the visual side. But having just beautiful (static) scenery wasn't enough; our artists built fantastic animated plants, clouds, and constructions to be used in the backgrounds. In addition we created dozens of atmosphere particle effects to add to the theme of each level and chapter: leaves, pollen, rain, dripping water, wind, steam, rubble, explosions... the list is long. We used a proprietary tool for creating the particle effects. While the process wasn't the easiest, our tool was flexible enough to accommodate for the mixture of effects we needed for *Outland*.



Unlike nailing the visuals early on in the project, audio (especially music) was one area that needed several iterations. The struggle wasn't really with producing the audio, but rather in agreeing on the style. Early songs were clearly more orchestral in nature, but lacked the high tempo we were planning for *Outland*.

We then tried a more "synth" approach but ended up with dance tunes that really didn't fit the atmospheric graphics. Finally, when talking more in-depth with our composer/sound designer, Ari Pulkkinen, we decided to go with ambient music for building the atmosphere and use uptempo music only during high action scenarios, such as the boss fights. This solution worked well, and Ari was able to compose around 20 songs for the game in a relatively short period of time.

What Went Wrong

1. Maintaining the Game's Vision

After realizing we didn't have all of the technology to create our original vision, the project lost a bit of focus and started drifting back and forth between different ideas. This drifting wasn't helped by the fact we had too many people in our meetings and they often took too long; we rarely accomplished what we were after and at times just wasted time.

We could also have done more research on old classics like *Super Metroid* or the *Castlevania* series. Avoiding them was partially intentional: not to copy directly from those games, and also not to repeat the 8-bit and 16-bit design pitfalls.

In any case, it would have been helpful to analyze the classics to see what they did right and how we could've improved or built on that. Failing to do that meant we ended up driving in circles more than we would have liked.

Both these reasons (technology and game research) ultimately meant we spent too much time on relatively simple things. We over-thought many of the enemies, and the final versions ended up being compromises.

Game structure and progression also suffered from not establishing clear enough targets from the get-go; we started building a more open, exploration-type world, but kept on adjusting it towards the linear based on player feedback, ending up somewhere in the middle.

2. Project Management

Having the lead designer also act as the project lead/producer didn't work well. A relative lack of experience in project management cross-fed to game design, and both areas suffered as a result. We talked about hiring a producer around eight to 10 months before completion, but thought it was already too late to bring someone new in. In hindsight, a producer, even at that point, would have helped.

We also tried managing design and art tasks using Hansoft, but it never caught up. Task lists were assembled several times throughout the project, but art and design team members didn't pay much attention to them. It might have been that the software required too much user input for fluid use, or it might have also been the benefits of using it weren't communicated clearly enough to the team. However, Hansoft was used with success among the programmers.

Poor communication during the first of half development and the lack of vision did considerably lower team morale for months. That partially caused certain people needing to crunch -- for three or four months. Luckily, once we got all the key gameplay pieces together the project started moving forward in much tighter pace. The last six months were what the whole project should have been, apart from the crunch.

3. Big Change in Direction During Production

Fixing a project that was adrift required a strong steering maneuver; bringing in the *Ikaruga* concept was just that. While it didn't deviate far from our original idea of using a special spirit mode, it changed the concept by making the skill non-consumable and removed the option to turn it off.

When the idea came up in a meeting, some 16 months before completion, it was love at first sight. We of course ended up rewriting our engine due to this, and made other smaller changes in the concept, which added to the programmer workload and introduced some delays to the schedule.

We also had to throw away most of the content done at that point and basically start afresh. Player animations were no longer valid for the new gameplay, nor were player physics and movement rules. The change also introduced new requirements from the level tools, further slowing us down.

But as gameplay and project uniqueness increased as a result, the change was for the better. It is possible the game would have never been finished had we continued on the original path.

4. Level Creation

When we changed direction that also meant we had to redesign some of the level building tools. We also had to create new rules for the levels and ended up bouncing back and forth between level requirements and level tools, creating a chicken-and-egg situation.

We used Autodesk Maya for creating our level art and design. All the tools inside Maya were done using either MEL script or Python. This enabled us to do all level-related items inside one program, but it also posed some problems.

We had no real way of dividing the work between level designer and artist, which resulted in one party having to wait for the other to finish their task, before gaining access to level files. In addition, we rewrote our level dressing tool several times, due to lack of proper design before starting the actual creation process -- those with keen eyes may notice this in two different types of texturing throughout the levels.

5. Co-Op Implementation

The original design was for a single player adventure platformer. As we happened to mention the possibility for co-op in a design document, Ubisoft called us on that. Adding co-op was not a simple task as it required us not only to take i into account when designing single player levels, but also to create specific coop-only challenge rooms. As our polarity-changing is the biggest single gameplay feature, figuring out how it works in co-op was big part of the challenge. These tasks by themselves were relatively safe to create, but increased our overall workload and caused shifts in the schedule.

For basic networking technology, we were able to acquire Quazal through Ubisoft. This definitely helped with the networking functionality but didn't remove the need for rewriting gameplay code to accommodate for online. As with the game engine itself, we ended up rewriting the network handling of gameplay objects and enemies; our final implementation ended up being a deterministic system.

As the co-op levels are relative carbon copies of their single player versions, with minor tweaks here and there, the challenge rooms provided a unique aspect to the co-op, while the quickly tacked-on arcade mode turned out to be an experiment that would have benefitted from another two months of development.

6. Gameplay Balancing

Balancing a game that comes together close to the end of the project is hard, and results can be difficult to verify. We did hold multiple sessions with people outside the studio coming over to try the game out. But as these sessions were relatively short, we mostly got feedback on the first chapter and boss. This ultimately meant that we were working on guesswork and hunches when it came to the rest of the game. Some of our levels and chapters are clearly too hard in comparison to the rest of the game; it is also fair to say that the game is on the hard side overall.

Also, having multiple level designers working on their own chapters meant those chapters ended up having their own level of difficulty, something we should have tested more and adjusted for where needed. The second chapter, Underworld, is nearly the hardest one of them all, while the chapter after it, City, is clearly easy by comparison -- a direct result of two different level designers handling the chapters.

Boss fights are another aspect of gameplay that could have used better balancing. While the first boss, the Golem, worked out really well, others we didn't test enough. The second, third, and fourth bosses could have all used a checkpoint halfway through the fight. The final one unnecessarily repeated an arduous 20 second climb each time you died. All of the boss fights should also have restored player health to full at the beginning of the level.

Conclusions

"Project Kingdom" started on the premise of recreating very old platforming classics in a new form, but midway through development, the project transfigured into a more action-driven merger of two genres. The name changed to *Outland* and the character that used to be able to shoot and slide down slopes became a melee warrior who switched between light and dark polarities.

We were first clearly restricted by technology, then by vision, followed by lack of project management experience, and finally by time. Even taking all of that into account, and the fact that we had to push back our deadlines a few times, we were able to create a new kind of a platforming game that hopefully speaks to a wide audience.

There are a lot of areas we would like to improve or redo, but even with its faults, there is hope *Outland* will be remembered as a modern day classic by gamers. Although we had a pretty rough ride at times, we are really proud of what we have accomplished, as *Outland* is our first 2D platformer. Given that we learned a lot during the development process, our hope is to revisit the platformer genre in the future and create another great game you can fall in love with!

Data Box

Developer: Housemarque
Publisher: Ubisoft
Platforms: Xbox Live Arcade, PlayStation 3 PlayStation Network
Release Date: April/June, 2011
Length of development: 22 months
Team Size at the Beginning of the project: 5
Number of People hired: 5
Total Team Size at the End of the project: 12

Return to the full version of this article