


GAME DATA

PUBLISHER: Ubisoft
DEVELOPER: Ubisoft Montreal
NUMBER OF DEVELOPERS: 65 at peak (excluding testers)
LENGTH OF DEVELOPMENT: 27 months
DEVELOPMENT HARDWARE: Average machine: Dual AMD Athlon 2000, 1GB RAM, Windows 2000, PlayStation 2, Xbox, and GameCube development kits, PlayStation 2 Performer Analyzer
DEVELOPMENT SOFTWARE: Microsoft Visual .Net 2003, Metrowerks CodeWarrior, PlayStation 2 Tuner, Incredibuild
RELEASE DATE: November 2003
TARGET PLATFORMS: Xbox, PC, GameCube, PlayStation 2, Game Boy Advance
PROJECT SIZE: 4188 files, 1,263,580 lines of code
PS2 BUGS: 11,520
XBOX BUGS: 936
GAMECUBE BUGS: 1,004
PC BUGS: 1,072
TOTAL BUGS: 14,613



YANNIS MALLAT | A producer for Ubisoft, Yannis's primary titles include PRINCE OF PERSIA: THE SANDS OF TIME and RAYMAN ADVANCE. He also worked on production for the movies Little Nicky, The Emperor's New Groove, and Dinosaur.



Ubisoft's PRINCE OF PERSIA: THE SANDS OF TIME

PRINCE OF PERSIA, an original creation by Jordan Mechner, was first released in the U.S. in 1989. The game, which follows the adventures of a young prince's efforts to save a princess, is regarded by many analysts as the first true action/adventure game. The PRINCE OF PERSIA franchise has seen two sequels since its conception: PRINCE OF PERSIA: THE SHADOW AND THE FLAME (1993) and PRINCE OF PERSIA: 3D (1999). By 2001, Ubisoft felt the time had come for the return of the prince.

When PRINCE OF PERSIA was first released in 1989, it got the attention of the game industry. It became an instant classic and laid the foundation for the action/adventure genre. The settings were strong, the storytelling was compelling, and the animations were groundbreaking. The game established new standards for what the public should and would expect from videogames to come.

By May 2001, a number of platformers had been released since the launch of the original PRINCE OF PERSIA. Most of them were inspired by at least some of the elements that made PRINCE OF PERSIA an important achievement. In Spring 2001, Ubisoft announced it had acquired the PRINCE OF PERSIA license and gave the

Montreal team a mandate to start the conceptual phase of the project.

Early on we identified the three core areas that made the original game a success. They are 1) captivating animations and character movements, 2) intense fight sequences, and 3) clever and challenging levels and the gameplay built around them. They were the essence of the brand and, if used with the right formula, the universal ingredients for a stellar action/adventure game. We considered them the heart and soul of the project.

So, there we were, a team of seven, laying down the basis of what would later become PRINCE OF PERSIA: THE SANDS OF TIME. Two game designers worked on defining the main concept, helping to build prototypes in real time with the technical team. One animator created the major moves that essentially brought the prince to life.

We then integrated two engineers into the process. They started the

engine studies and helped the design team conduct gameplay tests. A concept artist was added to the mix to illustrate game design ideas and provide initial art direction (to the extent possible at this stage). He also contributed creative ideas. The final piece of the puzzle was the producer, someone who would also act as a game designer and creative consultant, a role I gladly accepted.

A couple of months later, when we were able to present our first mock ups (AVI files showing how the prince could move and interact with his environment), we asked the original PRINCE OF PERSIA creator Jordan Mechner to look at what we had done. The result of the first presentation was inspiring. He was duly impressed. He hopped on the train and the core team started chugging along full steam, beginning with the pre-production phase and then switching to the production period.



Character production workflow, showing one of the prince's enemies from concept to game.

What Went Wrong

1. Late arrival of the artistic director. While the project effectively began in June 2001 with a fast-track conceptual phase, the art director, Raphael Lacoste, did not join the project until late April 2002. Although it didn't impair the final art direction, the very late arrival of our artistic director did create a huge challenge in time management for the team of artists.

Prior to his arrival, several prototypes had already been made showing the prince's movement set, level design ingredients, and some technological breakthroughs, but nothing very impressive. There was almost no art at all. The game's potential was demonstrated with some very basic level design blocks and monochrome textures.

Raphael's first task was to define the artistic direction and style of the game and to develop all the necessary tools. Light maps were to be added to the engine at the 11th hour of pre-production, along with many other effects (volumetric fog, filter, glow, and so on). The most difficult challenge for the modelers was to keep a steady production pace for the maps while learning about upcoming and unfinished tools. As a matter of fact,

the first final art wasn't available until the E3 2003 demo.

Coming back from the show, the team saw the demo as the standard of quality that should be consistently present throughout the whole game. This seemed impossible, considering just how much we still needed to produce. The demo was approximately 1/30 of the whole game. But the risk management output (including some scope reduction) and the tremendous efforts of our highly motivated team resulted in visual quality that surpassed that of the demo.

2. Fuzzy validation process. From the beginning, we knew that dealing with such a well-known license would present some challenges. We needed a huge pre-production process to help us establish clear goals, which included completing character behavior, macro designs, a compelling storyline, and all tools. A playable proof would then allow us to move forward into production.

That said, we didn't think pre-production would last as long as it actually did. When level production began, we had planned for 10 months; it eventually took more than 14, with a good list of tools and fighting behavior still in pre-

production. Maintaining the right balance between creation and production was hard, and there was no clear distinction between what was approved and what still needed improvement.

The prince's behaviors were often changed, refined, and tweaked, which required major modifications each time. All of this was good for the game's overall quality, but we had already lost precious production time designing, implementing, and rejecting several complete fight systems (in animation and AI). The result was a chain reaction that put other important deliverables in jeopardy. For instance, we started Farah's (the princess) AI development later than expected. We didn't have enough time to really polish the generic AI-supporting level-design scripted events. We had to take care of cooperative gameplay case by case, level by level, situation by situation. All this postponed the start of the real debugging period. We were faced with a mountain of bugs that had to be fixed. But the gold master release date was not going to budge.

3. Complicated enemies. The prince's character was the subject of intense work during pre-production.

With more than 780 animations, he was obviously the most significant—and the largest—component of the game. Unfortunately, this left less time and fewer resources to develop those who would allow him to exploit all his abilities: his enemies.

Enemies represent particular level design ingredients. Being extremely dynamic, they need to complement the main character's combat skills. At the same time, they should also increasingly challenge the players and surprise them with unexpected behaviors in any given situation. We also used specific enemies as tools to teach the player how to fight better—an instrumental aid in the players' learning process.

Due to the late delivery of final maps, all the enemies' behaviors had to be developed and coded on placeholder maps (basically a floor), which did not take into account the geometry of the actual maps. Obviously, in this situation, the enemies' AI came out way too bland, compared to what it should have been. Contextual enemies (such as the Sandbirds, Sandtigers and other mythical creatures) were extremely cost-inefficient to produce. Some of them simply had to be cut, whereas all the bipedal enemies later required a significant debug process.

4 • Lack of strong technical level design. From the beginning, our game was all about level design. Each of the prince's moves drove the micro-gameplay. Much of what the players would enjoy was rooted in level design. Every aspect of the prince's behavior or animation had a match in the geometry of a level. The game was very context-sensitive: you need a wall to make a wall-running maneuver; you need a column to

slide down it.

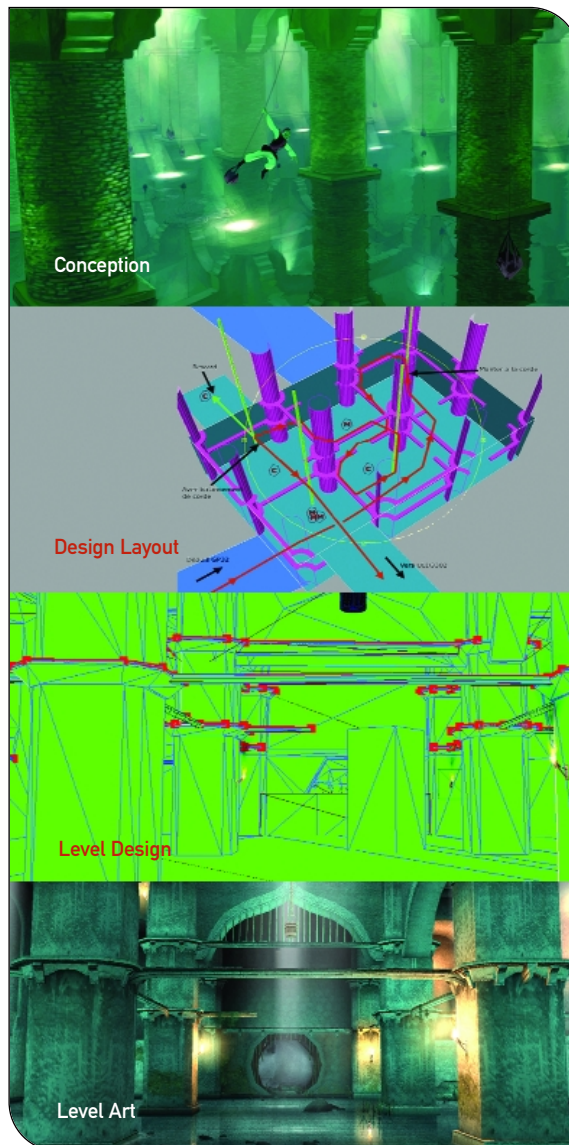
We had to make our technical features behave flawlessly. First of all, the dynamic loading was not ready right from the start of production, so we had nightmares getting everything to fit in memory and adjusting pre-fetch settings. Making all these adjustments was very tricky because we wanted everything loaded in time. To avoid sudden movements or pop-ups, we had to make

everything highly interdependent. On top of that, we had to make sure our rewind feature was always working, since this was how objects/enemies were destroyed—through dynamic loading portals and the like. Combine all these with a bunch of eager QA testers and you get a pretty intimidating bug database. Thankfully, the level designers and the programming team were able to squash all of the bugs.

So, were we starting to see light at the end of the tunnel at this point? Well, not quite. We were forgetting another source of problems. The game wasn't crashing anymore, but the enemies were forgetting their objectives. This led to broken gameplay, where enemies no longer saw the prince or attacked him. Furthermore, since you couldn't beat them, you couldn't complete the level. Even worse, the princess was completely forgetting many of her crucial goals.

So much could have been done in the earlier stages of development to prevent these problems. If only the maps and gameplay had been delivered in advance, a dedicated technical level designer could have foreseen all these issues and fixed them before alpha. Once again, we were not dealing with the problems in a strategic way; we were putting out fires as they occurred. Meanwhile, we were creating a mountain of bugs to deal with later.

5 • Data control. The way the engine was built meant the game data was stored in one master file that contained everything for the developers to review: maps, models, AI, and the rest. Everything was centralized in this file except sound and videos. The situation didn't allow for multiple concurrent data access on the same file, at



Map production workflow for building the Maharajah's palace.

least without written permission.

We soon realized our team had become too large to allow everyone access to the master files at the same time. We inherited a system that was designed with a small team in mind, but it didn't scale well to an army of 45 in crunch mode. Many problems occurred: data was overwritten; the server crashed; files got corrupt. A lot of time was wasted because people had to wait for their turns to enter their changes on the network.

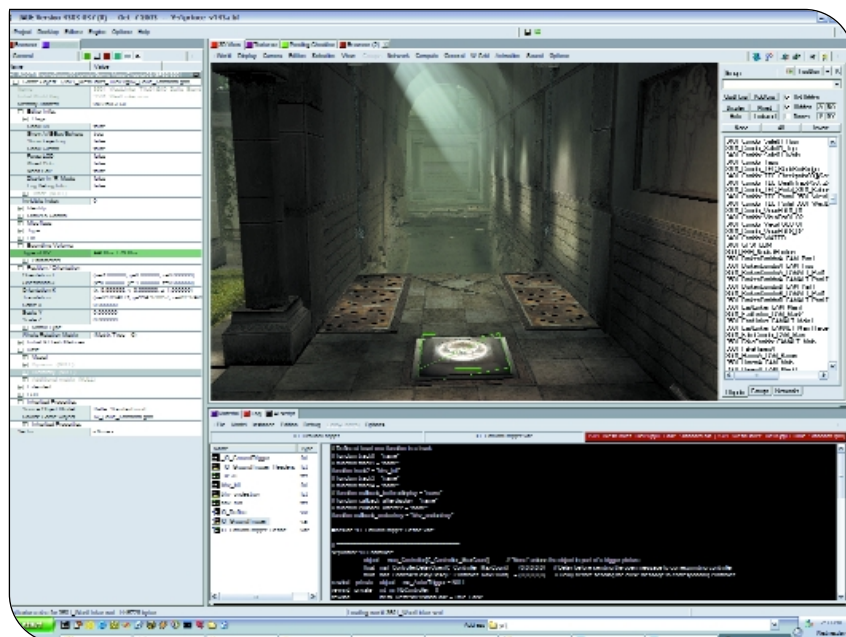
We tried to optimize the data control at the very end of the project, by building a "data monkey" solution that would allow simultaneous access through a server while maintaining a single repository for game data. Unfortunately, the attempt to build such a tool came too late and we never had the chance to alter the system. The risks involved were too serious.

One little thing we did, however, was set up a simple file server to manage the timing of all check-ins. At least the developers could work on something else locally while waiting for project updates, and we could give priority to people trying to make critical changes.

What Went Right

1. The will to achieve. A major element that contributed to the success of the whole project came from the team itself, and we managed to keep the initial motivation and chemistry strong right up to the end. The team was (and still is) a collection of extremely talented people in every field.

The project started well with a very powerful initial deliverable that helped everyone to clearly see what we were aiming for. At the start, the team was composed of less than 10 core people in complete harmony with one another—a tight-knit family. We were able to maintain the most effective form of communication: honesty. Speaking harshly about things that needed to be discussed was not a problem; we shared a common desire: the success of the project. No ego trip threatened the team's interest. Integration of newcomers



Lighting sets the mood in a corridor of the Maharajah's palace, as viewed through Ubisoft's internal playable level editor.

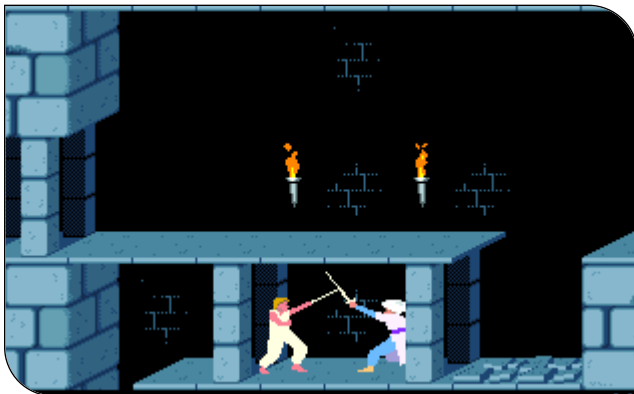
could have disrupted this cohesion, but it didn't, because we didn't add large numbers to the team all at once. Instead, we chose to incorporate newcomers one at a time, easing them into the unit gently.

A succession of morale-boosting events helped maintain the highest level of energy and motivation within the team: Sony decided to show the game at its E3 booth, and our own demo of the game at E3 was well received: people turned out in droves as word spread quickly that this was the game to see. Our high motivation level and confidence in the project allowed us to deal with an incredible amount of pressure (time and quality, for starters), accept some difficult realities (scope reduction and so on), and work extremely hard for a very long period. From the E3 demo preparation (late February) to the very end, we worked on average 16 hours per day, peaking at 20 to 48 consecutive hours sometimes. It's not a good model and we would prefer not to work like this again, but it was essential and the whole team was up for it, with absolutely no complaint.

2. Synchronization between animation and AI. The prince, as he appears in the final game, was our very first success and could not have been achieved without a fantastic duo that was paired up at the very beginning. The lead animator and lead AI for the main character worked very closely together. There was no question of a separate animation production on which we would simply map the AI afterwards. Both were conceived together, created together, and generated and implemented together. The two guys actually placed their desks side by side and worked as if they shared one brain. This is apparent in the way animation and control (AI) work seamlessly together in the final version.

3. Risk management. When tough decisions needed to be made, we made them. We reduced the scope of the game at two crucial times: just before Christmas 2002 and right after E3 2003. Fortunately, these decisions were made early enough in the development process.

The first scope reduction was the



The original PRINCE OF PERSIA titles stretched the boundaries of animation and art design.

hardest to make, because we were still far enough from the gold master date to convince ourselves “everything would be just fine.” Specifically, we were talking about cutting an entire chapter that took place in a slave village featuring exotic gameplay elements. Cutting this specific chapter meant having to tell the story very quickly. We accepted this decision because, in the end, everyone agreed it was the right move; if we had made this decision later, or worse, if we had refused to trim it, we would never have been able to finish the game on time.

When we got back from E3, we faced the bitter reality of chaotic production: most maps were not running at all; some were not even close to completion. Thus, the second scope reduction was logistically easier to make, but still hard on the team: it meant cutting some things that we had spent a lot of time working on, stuff that we were proud of. But, here again, if we had made this decision even a week later, we wouldn’t have met our deadlines.

4. Playable editor and other tools. As I’ve said, this game was all about level design. In PRINCE OF PERSIA: THE SANDS OF TIME, the gameplay was created mainly by the environment. Technically, all the level design distances had to be perfectly adjusted, because the gameplay could not exist with any degree of approximation.

When the prince grabs an edge from a

vertical-wall rebound, his detection zone should be perfectly in synch with the edge (in terms of spacing). This could have been a very strict limitation in level design creativity, but it wasn’t.

The editor was built to let level designers play with a 3D view. This allowed for quick corrections, thanks to a trial-and-error approach. Adjusting a column, adding a rope, or removing any level design ingredients were done on the fly and tested immediately by the level designers. The most interesting and crazy level design sequences were created in a very short amount of time. When the map was on the modeling side, it was also extremely useful to check whether the gameplay was altered by the addition of extra art geometry (such as a light torch on a wall where the prince needs to run). The tool helped us quickly devise interesting gameplay ideas during pre-production, then produce art geometry without wasting time compiling everything for a look at how the map was played.

5. Integrated testing. Finally, we provided development kits to as many testers as possible. At peak time, we had 14 PlayStation 2 development kits for the team, four of which were solely dedicated to QA testers reproducing very rare crash-bugs (with a special debug “strike-team” to take over the machine with a debugger and reverse-engineer strange bugs in retail code).

This started a creative solution to a recurring problem. One day, we realized one of our testers was great at finding A bugs—the rare, nasty ones. She was able to find bugs no one else could. Initially, each developer who was assigned to her bugs got frustrated due to the time commitment in fixing them. Then, we asked her to join the team, equipped her with a development kit, and with her working on the game itself, we got our A bugs curve back to normal. We replicated the model to up to four integrated testers within the team. This dramatically accelerated the pace of finding and fixing bugs, freeing some time for the developers to focus on the fixing side. Eventually, these testers got into the groove of things and spent many long days and nights contributing to our collective masterpiece.

1,001 Nights Later

And there we were, at the end of October 2003. After all the crazy events we had experienced in the previous 36 months, the gold master was finally delivered and the CD-ROMs were pressed. We couldn’t believe it. We had made it.

This team can be very proud of what it achieved. I would gladly work with all of them again in a second (in fact, I am), and we are now ready to welcome newcomers for the next installment of our adventures. 🍷