GAMASUTRA
The Art & Business of Making Games

PRINT

## Postmortem: The Game Design of Surreal's *The Suffering*

By Richard Rouse III

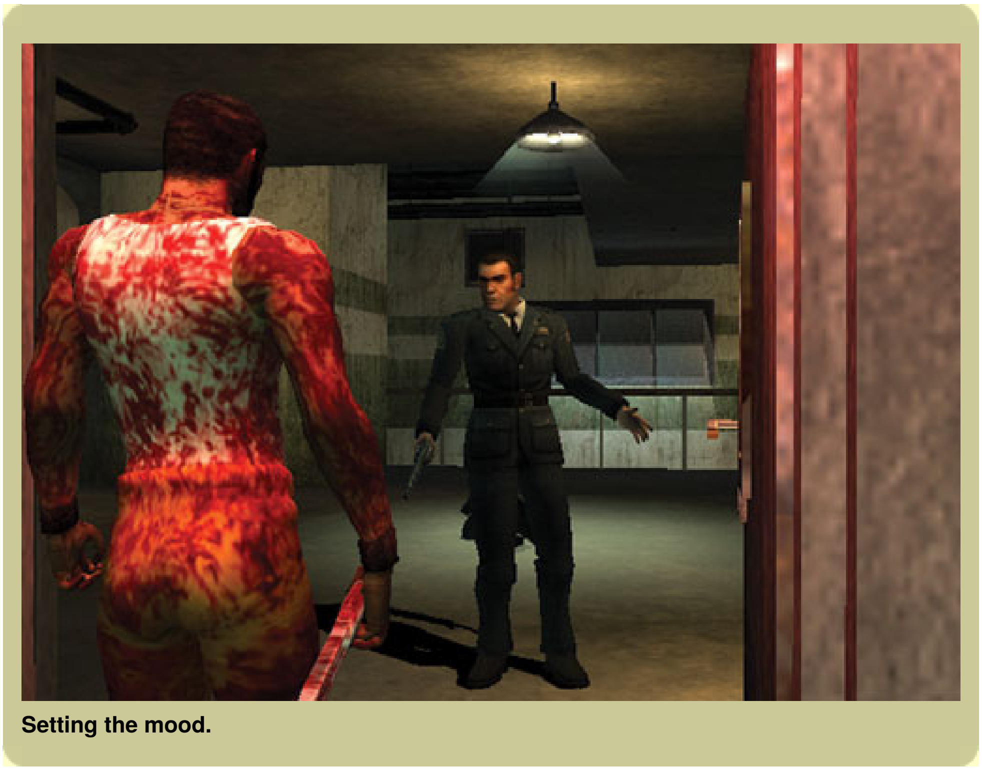
Postmortem:
THE SUFFERING
The trials and tribulations of designing an action-adventure FPS.
By Richard Rouse III
Cover Feature

"A stylized horror shooter. The frenetic gameplay of *Devil May Cry* meets the horror setting of *Resident Evil* and the immersive game-world of *Half-Life*."

So began the two-page pitch document that marked the start of Surreal's development of *The Suffering*. It is odd to read it now, two and a quarter years after it was written, especially given the various twists and turns the game took along its road to completion. But it is even more remarkable how little the game's concept changed from the basics laid out in that initial document.

Like most projects *The Suffering* had many of the classic successes and failures with which long-time readers of *Game Developer* postmortems will be keenly familiar. Though *The Suffering* employs many game mechanics that are well established (it is a shooter after all) a number of design decisions were made early on that we hoped would make the game stand out. Thus it is interesting to look back on the game's development purely from a game design standpoint to see what worked and what did not.

In many ways, *The Suffering* emerged out of the ashes of a game I had spent the prior two years on, an action/RPG Western called *Gunslinger*, a game that ultimately fell prey to the market's aversion to its setting in the old West.



**Setting the mood.**

*The Suffering* was planned from the get-go to be more focused and conservative in what it tried to accomplish than the extremely ambitious *Gunslinger*. I realized that in the end part of what sunk *Gunslinger* was its lofty aspirations, and with *The Suffering* we had a game I knew we could pull off, including its drastically stripped down morality system. From the start, however, I had very concrete design goals for the project that I hoped would make it stand out.

Most important was that *The Suffering* was to be an action horror game, instead of a survival horror game. This meant we were going to focus more on combat and avoid the long cut-scenes, frail central characters, clumsy controls, fixed camera angles, and sparse ammo of many console horror games. Losing those elements we knew we would not be able to pull off the cinematic style of horror employed by *Silent Hill* and *Resident Evil*, and would need to instead focus on establishing a more disturbing and unsettling tone, taking horror novels as our inspiration instead of films.

### Lessons from the Gunslinger

When we first became interested in developing *The Suffering*, we felt that we had learned a lot about third person shooting mechanics through *Gunslinger*. Looking back on it now it seems much of what we had learned was what not to do. The aiming-based shooting combat that ships in *The Suffering* is nothing like the target-lock based shooting that was in the final iteration of *Gunslinger*. Something that carried over more directly was *Gunslinger*'s reputation system, but in a drastically simplified form. In *Gunslinger* our goal was to keep track of the player's "good" or "bad" actions and then have the NPCs in the environment react accordingly. This was an incredibly complex system with multiple groups who might view the player's actions differently incorporated with a unique method of distributing knowledge of the player's actions through a town full of people in a believable fashion. (For those interested in the system, Greg Alt wrote an in-depth article about its architecture for the first *AI Game Programming Wisdom* book.)

We also felt we could be more effective at keeping the player tense and on-edge by immersing them in the game-world as much as possible. In addition to our improved controls and limited cut-scenes, we wanted to immerse the player through player-empowerment. A number of our decisions reflected this: control of the player character would be as smooth and intuitive as possible; the player would be able to interact with the game-world in a believable and consistent way; the player would make their own way through the game-world via multiple paths and numerous optional side-areas; there would be multiple ways to accomplish a given task; players would be able to explore the game's story as much or as little as they want; and players would be allowed to make important choices about how they act in the game world, thus tying into our morality system. Through these important choices, the player is able to determine the main character Torque's guilt or innocence of the crime that landed him in prison. We felt this was our strongest element of player empowerment, allowing players to determine not only Torque's future but also his past, something altogether unique in games.

Finally, to have a disturbing and unsettling tone we knew that creepy monsters alone would not be enough. Therefore we wanted to tie into real-world horrific events. Thus the storyline is suffused with the darkest elements of American history, including prison life and culture, slavery, racism, unethical medical experimentation, mob-mentality executions, and the death penalty. This is fairly serious subject matter for a videogame, particularly an action-adventure, and it amplified the horror of our world tremendously.

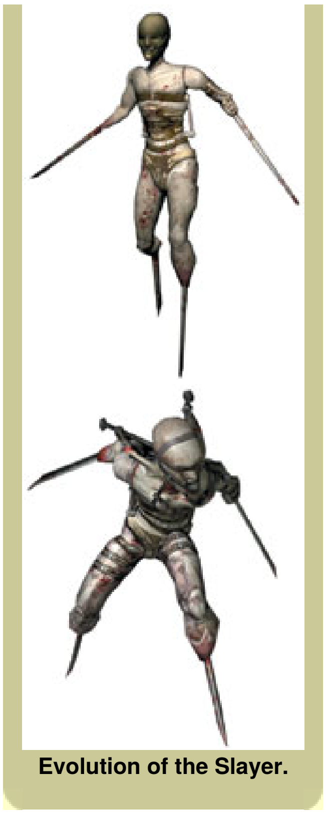
---

### What Went Right

**1. Initial Concept.** As I have discussed, the initial concept of our game changed relatively little over the course of development. Something about "an action horror game set in a prison" was uniquely compelling to our publisher, the press, and gamers alike. Despite containing highly stylized supernatural creatures, the game's very real-world setting was essential to making the game relevant and hooking people. The game's prison setting proved particularly intriguing to gamers and was a rich space for us to explore that had been under-utilized previously.

Shortly after development started, I wrote a fairly detailed back-story for both the game world (Carnate Island) and Torque, and these elements also changed relatively little over the course of development. Though we did not plan on communicating all of this back-story to the player directly, it gave the game tremendous consistency as we built it. As we were given more time to iterate on the project, the back-story documents gave us a strong foundation on which to expand the game without seeming forced.

**2. Focus.** Having established our high-level design goals from the start, we were then extremely frugal about adding features. We knew that in order to properly implement the features the game did need, we would have to omit mechanics that were non-essential. For example, beyond his weapons, health, and flashlight batteries, Torque cannot carry any inventory items, including keys. To some, it was odd that we were making a prison game that didn't include using keys to unlock cells and gates. But in the end we realized that including keys didn't really add much if anything to the core gameplay experience and would have been wasted development time.



**Evolution of the Slayer.**

At the same time, we worked hard to keep the features that enhanced our core gameplay. Our fully playable first person mode evolved out of a more traditional "look around" mode. Over the course of development numerous problems arose and cutting it was suggested numerous times. This feature, however, was a major enhancement to our core gameplay experience, since shooting from the first person perspective is extremely intuitive to players. Indeed, from our gameplay testing we knew this was a very popular feature. Thus we knew that whatever extra time was required to make a fully functional first person mode would be well worth it.

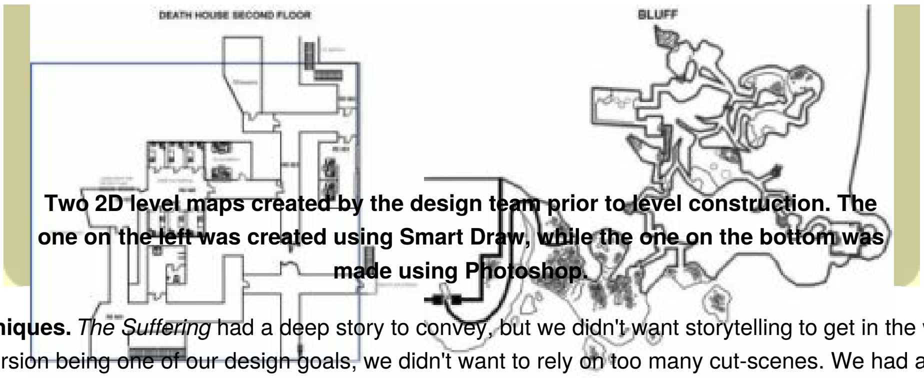
Though we may have been too conservative in a few cases (for example, the game's shooter mechanics would be better off had we included the ability for Torque to crouch), overall our strict policy paid off nicely and allowed us to refine our core features while staying on schedule.

**3. Changing the Control Scheme.** Though I said earlier that we stayed remarkably close to our original concept, there is something that changed significantly from our earliest one-liner: the game stopped being similar to *Devil May Cry* (*DMC*). Indeed, from the very beginning I wasn't much of a fan of the gameplay in *DMC* and preferred shooters that, at that time, were traditionally more popular on the PC. Indeed, *Half-Life* was also mentioned in our concept for exactly that reason. Truth be told, *DMC* was mentioned in the pitch because a number of the publishers we were talking with about *The Suffering* had expressed interest in appealing to the fans of *DMC*.

As a result, from the start our game and level design work had much more in common with *Half-Life* than with *DMC*, except for our controls. When designing control schemes, I feel that you want to give the user something they are familiar with from other games. In general I find relying on other games for inspiration to be problematic, but in the case of controls I think it is crucial. What we had originally implemented was a target-lock system inspired by *Syphon Filter*, the most popular third-person shooter on the PlayStation, and *DMC*, at the time the most popular third-person shooter on the PlayStation 2. With our controls for a console-style shooter but our gameplay from a PC-style shooter, about a year into development we realized we had a dangerous disconnect in our design that made our game tedious instead of fun.

However, by this point *Max Payne*, *Halo*, *Medal of Honor: Frontline*, and *SOCOM* had all been released on the consoles and sold in excess of a million copies each. All were shooting based games in the PC tradition: they eschewed target-lock in favor of double-stick control schemes that simulated the mouse/keyboard experience from the PC. This system had the advantage of forcing players to actually aim at their target while having the disadvantage of being challenging for novice players to pick up. But looking at the sales for these titles, we concluded the installed base of players who were familiar with these controls was now large enough that we could take the risk of turning off a few newbies.

The change was a huge success for the game: it fixed the disconnect in our gameplay and added depth that had been completely missing. There was now very little similarity to *DMC* to be found. Looking at the forums today, I find that some players still have trouble adjusting to the two-stick system, and I believe we have lost some potential players for this reason. However, our significantly deeper game experience has brought in so many players that I know we made the right decision.



**Two 2D level maps created by the design team prior to level construction. The one on the left was created using Smart Draw, while the one on the bottom was made using Photoshop.**

**4. Storytelling Techniques.** *The Suffering* had a deep story to convey, but we didn't want storytelling to get in the way of our core game experience. With immersion being one of our design goals, we didn't want to rely on too many cut-scenes. We had a rule of thumb that cut-scenes were to be used exclusively for pivotal story points or for intensely scary scenes. Furthermore, we wanted to keep Torque's actions fairly neutral during these scenes to avoid negating the player's feeling that they were fully in control of Torque at all times. Thus we needed to use different storytelling techniques.

A lot of story was communicated during gameplay through the various NPCs who function as Torque's guides through the world of Carnate Island. Though the player could kill any human character at any time (thus missing out on the story points they had to convey) being in a horror space allowed us to use supernatural characters who Torque was unable to kill. The player could also hear dialog over radios, PA systems, and telephones, all real-time during gameplay. The player was also able to collect various notes throughout the game in addition to unlocking pages in an archive, both of which revealed more of the back-story to players who were interested. Finally, we used a slow-motion blur effect to convey events from Torque's past and the history of the island. Inspired by some of the imagery from Stanley Kubrick's *The Shining*, this technique was our most innovative and also proved to be fairly frightening.

All of these techniques combined to allow us to tell a story with a minimum of play interruption. Players who wanted to experience the story were able to, while those who would rather stick to playing the game could ignore it. Even with these techniques, the story is kept mysterious enough that players will still be left with numerous unanswered questions. My hope is that players will fill in the blanks with their own imagination, following the tradition of great horror films such as *The Birds*, *The Shining*, *The Blair Witch Project*, and *The Ring*. In horror, the player's imagination is far more disturbing than anything a writer could possibly come up with.

**5. Iteration and Gameplay Testing.** From a design standpoint, one of the most fortunate events of *The Suffering*'s development was getting time to iterate on the game. Midway was quite happy with the game's progress and had seen a strong reaction to it from the press and public alike. Thus they gave us a generous time extension, not because we were behind schedule but because they wanted to make the game as strong as possible. Thus, with our levels all fully built and functional many months before shipping, we were able to do a number of passes on the game. We did a pass on horror elements to make the game more frightening, including adding our real-time environmental flashes that are so key to the final experience. We also did a story pass, not to change the story but to expand on how it was presented to the player. We performed an AI pass to make the creatures much more dynamic and varied in their behaviors. Finally we did a puzzle pass to fix the most egregious problems with the puzzles. The impact of these passes cannot be underestimated. For example, the game's design did not originally plan for the real-time scenes involving Torque's wife and children to be in the game, since we did not have time to build them from an art standpoint. Anyone who has played *The Suffering* knows how crucial those scenes are to the game experience.

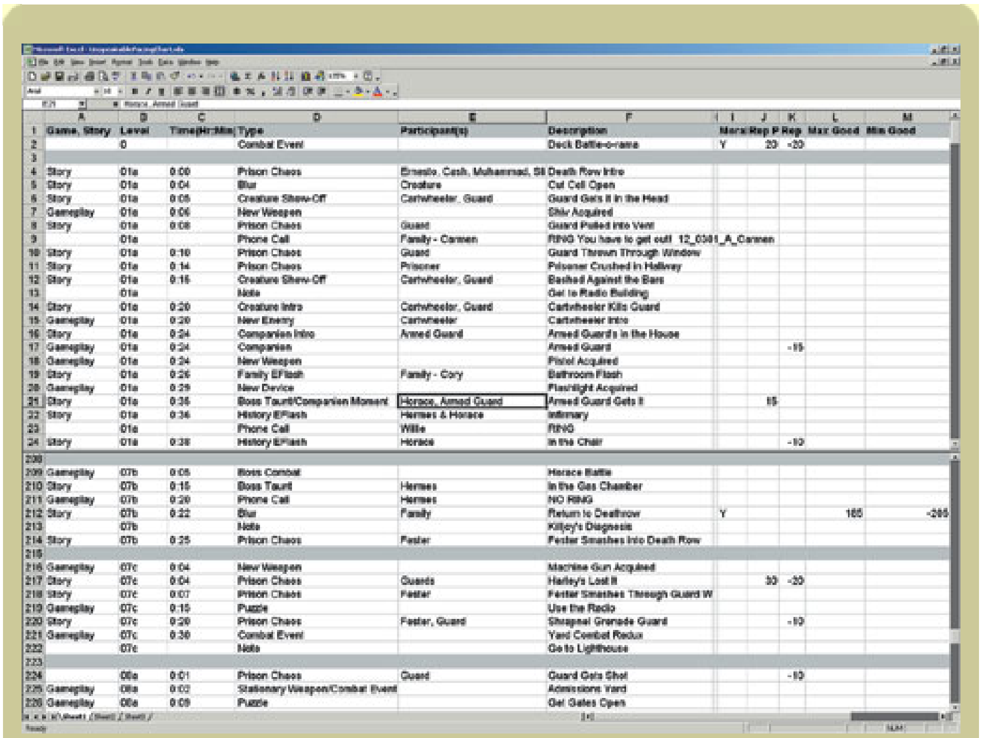
To help us figure out what needed fixing, at numerous points in development we put the game in front of a group of gamers and watched them play and then listened to their feedback. This gameplay testing is distinct from focus testing since these sessions were for development feedback alone, not for marketing use at all. We did this as early as seven months into development, and we were able to fix a lot of major problems early on, including our disjointed control scheme. If anything, the game could have benefited from more gameplay testing, but what we did have time for impacted the game tremendously.

---

### What Went Wrong

**1. Design Changes and Communication Thereof.** Previously I discussed how we shifted away from the console-style, target-lock-based shooting of *DMC* to the aiming-based shooting of PC first person shooters. Without a doubt this was the right decision to make. Unfortunately, though I was never a big fan of emulating *DMC*, we went down that road for the first third of the project and invested a significant amount of time in game mechanics that we threw out. If we had more carefully analyzed what type of game we were making from the start we could have saved six man-months of work. Also, when we finally decided to make the switch, the change was not properly communicated to the whole team, and people frequently asked me, "But I thought we were trying to be like *DMC*?" In a collaborative large-scale project, having a design vision is useless unless it is clearly communicated to the entire team.

**2. Puzzle and Boss Gameplay.** Even given our best efforts and despite being granted time to iterate on the game, our puzzle and boss design was not up to the standards we wanted. Both the puzzles and bosses were flawed because of their unpredictability; they caught players off-guard by involving multiple complex mechanics that players had never used previously. Part of this was due to the fact that we re-did each individual puzzle multiple times, and thus whatever puzzle progression had been planned over the course of the game was no longer applicable. Our bosses were further problematic because we tried to force platformer-style bosses into a PC shooter-style game (indeed, the bosses fit better when the game was more in the vein of *DMC*). This oversight was not recognized until it was too late. It was also a conceptual goal to make the bosses involve non-violent conflict to contrast with the extremely violent nature of the rest of the game. This made designing them and making them fun quite difficult since our non-violent mechanics were significantly limited and not nearly as much fun as our violent mechanics. All of our bosses and almost all of our puzzles were redone one or more times prior to shipping, and though they improved tremendously they were never anyone's favorite part of the game.



|  | A | B | C | D | E | F | I | J | K | L | M |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Game, Story | Level | Time(Hr:Min) | Type | Participant(s) | Description | Mora | Rep P | Rep | Max Good | Min Good |
| 2 |  | 0 |  | Combat Event |  | Deck Battle-o-rama | Y | 20 | -20 |  |  |
| 3 |  |  |  |  |  |  |  |  |  |  |  |
| 4 | Story | 01a | 0:00 | Prison Chaos | Ernesto, Cash, Muhammad, Sli | Death Row Intro |  |  |  |  |  |
| 5 | Story | 01a | 0:04 | Blur | Creature | Cut Cell Open |  |  |  |  |  |
| 6 | Story | 01a | 0:05 | Creature Show-Off | Cartwheeler, Guard | Guard Gets It in the Head |  |  |  |  |  |
| 7 | Gameplay | 01a | 0:06 | New Weapon |  | Shiv Acquired |  |  |  |  |  |
| 8 | Story | 01a | 0:08 | Prison Chaos | Guard | Guard Pulled into Vent |  |  |  |  |  |
| 9 |  | 01a |  | Phone Call | Family - Carmen | RING You have to get out! 12_0301_A_Carmen |  |  |  |  |  |
| 10 | Story | 01a | 0:10 | Prison Chaos | Guard | Guard Thrown Through Window |  |  |  |  |  |
| 11 | Story | 01a | 0:14 | Prison Chaos | Prisoner | Prisoner Crushed in Hallway |  |  |  |  |  |
| 12 | Story | 01a | 0:15 | Creature Show-Off | Cartwheeler, Guard | Bashed Against the Bars |  |  |  |  |  |
| 13 |  | 01a |  | Note |  | Get to Radio Building |  |  |  |  |  |
| 14 | Story | 01a | 0:20 | Creature Intro | Cartwheeler, Guard | Cartwheeler Kills Guard |  |  |  |  |  |
| 15 | Gameplay | 01a | 0:20 | New Enemy | Cartwheeler | Cartwheeler Intro |  |  |  |  |  |
| 16 | Story | 01a | 0:24 | Companion Intro | Armed Guard | Armed Guards in the House |  |  |  |  |  |
| 17 | Gameplay | 01a | 0:24 | Companion |  | Armed Guard |  |  | -15 |  |  |
| 18 | Gameplay | 01a | 0:24 | New Weapon |  | Pistol Acquired |  |  |  |  |  |
| 19 | Story | 01a | 0:26 | Family EFlash | Family - Cory | Bathroom Flash |  |  |  |  |  |
| 20 | Gameplay | 01a | 0:29 | New Device |  | Flashlight Acquired |  |  |  |  |  |
| 21 | Story | 01a | 0:35 | Boss Taunt/Companion Moment | Horace, Armed Guard | Armed Guard Gets It |  | 15 |  |  |  |
| 22 | Story | 01a | 0:36 | History EFlash | Hermes & Horace | Infirmary |  |  |  |  |  |
| 23 |  | 01a |  | Phone Call | Willie | RING |  |  |  |  |  |
| 24 | Story | 01a | 0:38 | History EFlash | Horace | In the Chair |  |  | -10 |  |  |
| 208 |  |  |  |  |  |  |  |  |  |  |  |
| 209 | Gameplay | 07b | 0:05 | Boss Combat |  | Horace Battle |  |  |  |  |  |
| 210 | Story | 07b | 0:15 | Boss Taunt | Hermes | In the Gas Chamber |  |  |  |  |  |
| 211 | Gameplay | 07b | 0:20 | Phone Call | Hermes | NO RING |  |  |  |  |  |
| 212 | Story | 07b | 0:22 | Blur | Family | Return to Deathrow | Y |  |  | 165 | -205 |
| 213 |  | 07b |  | Note |  | Killjoy's Diagnosis |  |  |  |  |  |
| 214 | Story | 07b | 0:25 | Prison Chaos | Fester | Fester Smashes Into Death Row |  |  |  |  |  |
| 215 |  |  |  |  |  |  |  |  |  |  |  |
| 216 | Gameplay | 07c | 0:04 | New Weapon |  | Machine Gun Acquired |  |  |  |  |  |
| 217 | Story | 07c | 0:04 | Prison Chaos | Guards | Harley's Lost It |  | 30 | -20 |  |  |
| 218 | Story | 07c | 0:07 | Prison Chaos | Fester | Fester Smashes Through Guard W |  |  |  |  |  |
| 219 | Gameplay | 07c | 0:15 | Puzzle |  | Use the Radio |  |  |  |  |  |
| 220 | Story | 07c | 0:20 | Prison Chaos | Fester, Guard | Shrapnel Grenade Guard |  |  | -10 |  |  |
| 221 | Gameplay | 07c | 0:30 | Combat Event |  | Yard Combat Redux |  |  |  |  |  |
| 222 |  | 07c |  | Note |  | Go to Lighthouse |  |  |  |  |  |
| 223 |  |  |  |  |  |  |  |  |  |  |  |
| 224 |  | 08a | 0:01 | Prison Chaos | Guard | Guard Gets Shot |  |  | -10 |  |  |
| 225 | Gameplay | 08a | 0:02 | Stationary Weapon/Combat Event |  | Admissions Yard |  |  |  |  |  |
| 226 | Gameplay | 08a | 0:09 | Puzzle |  | Get Gates Open |  |  |  |  |  |

**The pacing chart used by the design team to track the flow of major gameplay and story events, as well as how they interact with the player's reputation.**

**3. Weapon Variety and Distribution.** One of the drawbacks of setting a game in a real-world environment is that you don't have the opportunity for weapon diversity that you do in a science fiction or fantasy game. This is especially problematic for a shooter like *The Suffering*, where constraining yourself to real-world weapons and avoiding those that don't fit the setting (rocket launcher, anyone?) means that you're hurting the gameplay experience for the benefit of the story. In the end, we really could have used one more weapon.

Also, our desire to make the game accessible to a wide audience resulted in a bit more ammunition available in the world than we would have preferred. This had the unfortunate side-effect of most players not using the "Insanity Mode" creature; most simply didn't need it unless they were playing on the harder difficulties. Though we did include multiple difficulty levels, most players will tend to pick the default and stick with it, even if they find the game too easy. Indeed, by the time they figure that out, they're a fair ways into the game and it's hardly fair to expect them to restart their game from scratch at a higher difficulty. It would appear that the answer is a dynamic difficulty adjustment system that automatically makes the game harder or easier based on how well the player is playing the game moment to moment. This is something other games have dabbled in and that we will undoubtedly be using in the future.

**4. AI Design Issues.** Though we tried to make our game more accessible to players by supplying them with plenty of ammo, the primary reason our game is too hard for novice users is because our AI design was not made with our controls and player mechanics in mind. Having a creature that runs right behind you may be cool in concept and is something fairly well supported in a PC shooter using a keyboard/mouse control scheme or a console shooter with target-lock. However, with an aiming-based game using a console controller the player's ability to react to such events is significantly more limited. Indeed, our original target-locking control scheme supported this much better, and having the creatures move in this frantic fashion is another disjointedness that resulted from that transition. Furthermore, this is a classic example of a problem the development team is likely to be completely unaware of: since the developers have been playing the game for so long, they are extremely familiar with the controls and see a problem like this as a challenge instead of an annoyance. Indeed, whenever I see "controls are sluggish" in reviews of the game, I interpret that to be because of this AI-behavior and the controls disconnect.

**5. The Denial Factor.** After you've worked on a game for a year and a half, it's easy to overlook certain glaring flaws with your title. Thankfully our publisher and gameplay testers were able to point out many of these problems and we were able to fix most of them before we shipped. Nevertheless, we spent too much of the project in denial about problems with the game. This was true across all departments, but also in design, where poor puzzles, AI, and mechanics were ignored for too long. Since we put off fixing these issues until so late in the project we often had to use shortcuts and didn't have the time to polish the new solutions. We delivered a "first playable" of the game seven months into development that was supposed to be "shippable quality." Looking back on it now it's hard to imagine what drugs we were taking to make us think it was actually close to good enough. In the future, we plan on being significantly more strict with our quality levels over the course of development and to fix problems before we become accustomed to them.

**Most (but not all) of *The Suffering* development team at Surreal.**

### Heart of Darkness

I see *The Suffering* as somewhat unique among console action-adventure or shooter games in the seriousness of its subject matter and the moral themes it endeavors to explore. This is something that I wanted from day one and I am fairly happy with the result we achieved. To accomplish this, we made a number of key design decisions, from the player's ability to effect the world in a believable way (killing friendly humans), to the storytelling techniques we used, to the morality system that leads to the distinct endings. In the end though, the compatibility of a game about human atrocities over the ages within the "gory shooter" game milieu was perhaps our biggest limitation. As games seek to engage the player with more and more serious subject matter, the game mechanics need to evolve along with them, giving the player a wider expressive range than deciding whether to kill or not to kill. But as they say, Rome wasn't built in a day.

### Game Data

***The Suffering***

**Publisher:** Midway
**Developer:** Surreal Software
**Number of full-time developers:** 25
**Number of part-time developers:** 24 (not including testers or support staff)
**Number of contractors:** Three artists, three external art shops, one musician, 12 voice actors, and four motion capture actors
**Length of Development:** Two years and two months
**Release Date:** March 8, 2004 (U.S. PlayStation 2 and Xbox)
**Platforms:** PlayStation 2, Xbox, and PC
**Development Hardware:** High-end PCs with Windows 2000 or XP
**Development Software Used:** Microsoft Visual Studio .NET, Microsoft SourceSafe, Perforce, ProDG, SN Tuner, PS2 Performance Analyzer, ProView, Photoshop, Maya, Painter, 3DS Max, Sound Forge, CoolEdit Pro, Nuendo, SmartDraw, ACDSee, SourceForge, ProblemTracker, Proprietary Level Editor and Modeler

Return to the full version of this article