GAMASUTRA
The Art & Business of Making Games

PRINT

What Every Game Developer Needs to Know about Story

By John Sutherland

Increasingly, story is a hot item in games. Partly, this is because the quality bar is rising in this relatively young art form. As games evolve, people want more depth, not just higher polygon counts.

More to the point, game developers want to sell their wares to more people. Selling them to the same ones every time doesn't lead to a lot of growth. It's clear we need to tap into something more universally human.

And story is a universal human experience.

So how do we approach story in games? Well, to answer that, we need to look at what has worked in other story forms, and what is unique to the new story form of games.

Let's start with a statement everybody can agree on: *Games aren't movies.*

But that by itself doesn't get us very far. To figure out what games are, it's helpful to back up to an earlier problem: *Movies aren't plays.*

In the early part of the 20th century, moving pictures were a curiosity, an amusement. They had their addicts right from the beginning, to be sure. But they didn't become a substantial lasting art form until they discovered two related things:

1. They are a form of story, not just a new toy.
2. Their particular form of story differs from all previous forms of story, and has other things in common with all forms of story.

The same is true for games.

The first attempts to make movies into real stories failed. They failed because they were conceived as filmed plays. A camera would be set up about where an audience member would sit in the middle of a theater, and the play would ensue.

It didn't work. Early film makers didn't take into account that the human eye wanders all over the fixed box of the stage during a play, and a camera that does any less will bore the film audience to tears. They also hand discovered the rich tool set of camera angles, close-ups, far shots, and all the language of film we now take for granted. Generally speaking, *they hadn't discovered what this particular story form was good at*.

And frankly, neither have we in games.

Common misperceptions

There are a number of places where we've gone wrong in game stories so far. Most of the problems spring from two basic misunderstandings:

- Story is dialog.
- Story doesn't matter.

Sure, story is partly dialog. And a cake is part frosting. But here's a large fact that I'll elaborate on in just a moment: *Story is CONFLICT*.

The sabotaged dope deal in *Grand Theft Auto: Vice City* is an 'inciting incident' that causes the protagonist's world to be thrown out of order.

Now, the notion that story doesn't matter is worst with the industry old-timers. "Just repeat that 30 seconds of gameplay, and you've got it," I've heard. Or worse: "We've never had to worry about of that story stuff before."

Maybe that's okay for a small audience of addicted gamers, but the new charter for platforms like Xbox 360 is to appeal to a mass audience, not necessarily people who have even played games before. That means that if games are ever to rise to the level of universal cultural experiences, the way movies have, we have to figure out the same story problems movies did in the last century.

A starting place

Okay, to really get this right, we have to talk about story in general. What has always worked, and what will work in *every* form of story, including games?

We could go back to Aristotle's "Poetics," but a more useful reference is from that curmudgeonly guru of Hollywood screenwriters, Robert McKee, author of the book, *Story*, which is based on his many three-day intensive story structure lectures.

Some people have their doubts about McKee, based on his personality, or his emphasis on structure, or simply suspicions of his massive following. But here's something to consider: *He's right.* Deal with it.

Now I'm reasonably sure McKee has never played a video game in his life. He certainly never mentions it in his work. But that's where we come in. If we are to develop games into the fairly advanced story form that movies have become, we need to start by learning everything movies had to learn, and McKee's *Story* is the place where that is best summed up. I'll touch on the main points here, but make the reading of that book your homework assignment. Lots of what I'm going to talk about comes from him.

The real substance of story, as McKee points out, is CONFLICT. Did I already say that? Good. I'm repeating myself on purpose. If you remember nothing else, remember this. *Story is conflict.*

This is no trivial point for game developers. This has huge implications for how we plan our productions cycles, and how story is best presented in the game. For one thing, the conflict is part of the structure, which means it needs to be planned from the beginning of the development process.

So, how does this conflict work through the course of the story? Glad you asked.

How classical stories move

There are a couple of very good reasons for game developers to know about classical story structure:

- It's simple.
- It works.

What I'm going to tell you is not something that will confine your creativity. On the contrary, if you keep this basic structure in your back pocket, it will save you loads of trouble throughout your creative process. And this is not merely a theory from Aristotle. This has been put into practice by story tellers of all kinds for thousands of years. You could say it has been thoroughly tested.

- First, there's a protagonist, a **hero**.
- His or her world is thrown out of order by an **inciting incident**. (Look at the sabotaged dope deal in *Grand Theft Auto: Vice City* for a good example of this.)

- A **gap** opens up between the hero and an orderly life.
- The hero tries the normal, conservative action to overcome the gap. It fails. The world pushes back too hard.
- The hero then has to take a **risk** to overcome the obstacles that are pushing back.
- Then there is a **reversal**. Something new happens, or the hero learns something she didn't know before, and the world is out of whack again. A second gap has opened up.
- The hero has to take a greater risk to overcome the second gap.
- After overcoming the second gap, there is another reversal, opening a third gap.
- The hero has to take the greatest risk of all to overcome this gap and get to that **object of desire**, which is usually an orderly life.

In a three-act classical story, this is what happens. With more acts, you have more obstacles. But three is a minimum, and a good goal for a game. In a comic structure, the last gap is overcome. In a tragic structure, one final gap opens up, and stays there. But games are comic by their nature, so we'll assume it's possible to get to the end.

Character, and why it matters to games

So while we're talking about universal story principles, there are a few important notes we should make about character. There are a lot of misunderstandings about what character is, and what it isn't.

What a character wears, eats, and drives are all important. But those aren't the things that make him who he is. That's *characterization*: the superficial stuff.

Character is what he chooses to do. He's driving by a burning school bus. There are 80 kids trapped inside, and it's going to explode in 37 seconds. Does he risk his life, or get out of the way? That's what defines his character.

Any good story will have pressures on the hero to bring out these choices, and therefore the character. This is called the *principle of antagonism*.

The rest of your cast of character should be designed around the protagonist, because they conflict with her to define her character.

The world of the game should be designed (and often is-this is one thing games already do well) to oppose the player at every turn. As games become more sophisticated, these forces of antagonism will provide more interesting choices, and our characters will become deeper.

The importance of reversals

As you can see in the brief summary of classical story structure, each act is driven by a reversal. When the hero accomplishes the ordeal before him, the reversal is what keeps him going forward instead of resting on his haunches.

A reversal can happen through action (somebody shoots a cop's partner, and that changes everything) or by a revelation (Luke Skywalker finds out that Darth Vader is really his father). The revelation is much more powerful. The important thing is not to cheat. When Danny Glover's character pulls the trigger at the end of *Lethal Weapon 2*, he's causing a reversal through action. He "revokes" the bad guy's diplomatic immunity. But it's a bad reversal, because he could have done the same thing ten minutes into the movie. Don't do that. Your audience can smell cheating.

What a good reversal does is expand the story's universe. Everything you knew was true, but now there's more, so the world is flipped on its head with this new knowledge. The important thing is to make this convincing within the world of the story. If you can pull this off well, twice, and everything else goes well, you're going to have a great game.

Story forms before games

This classical story form is universal, but many of the practical details of storytelling work very differently in plays, in novels, and in movies. You can be sure there will also be differences in games. To find out how games are different, it pays us to look at the ways in which other forms are different from one another. Looking over these, we see a couple of trends.

Types of conflict and types of story

There are different types of conflict in story. There's *internal conflict*, which is what goes on inside your head. There's *interpersonal conflict*, which is between people. And there's *external conflict*, which is conflict with society in general or the physical world. So if you're shooting at an oncoming space ship, it's external conflict; if you're shooting at your ex-husband, it's interpersonal conflict.

Here's why this is important: While it's possible to have any kind of conflict in any form of story, certain story forms to certain forms of conflict more naturally than others.

- Internal conflict happens most naturally in novels.
- Interpersonal conflict happens most naturally in plays, and in soap operas.
- External conflict happens most naturally in movies, *and in games*.

Now all this is theory until you look at another relevant fact: these different levels of conflict are expressed with different levels of visual and auditory storytelling. This directly affects how you handle dialog in your game.

We see movies, but we hear plays. Movies are 80% visual and 20% auditory. Plays are 80% auditory and 20% visual.

Movies with talking heads are boring. But at a play, we listen to the pleasant music of the dialog while our eye is free to wander across the static box of the set.

How does this work in games? Like movies, they are primarily visual. Gamers don't tolerate a lot of dialog, with one exception, which I'll get to in a moment.

So, the evidence is mounting to throw out our original premise, that games aren't movies. In fact, they have a lot in common with movies. They're the closest cousin we have among the story forms. But they're not at all the same.

Empathy, and the big protagonist flip

Here's where games get revolutionary. Through all forms of story, writers have been trying to create heroes that the audience can feel empathy for. We, as viewers, may not always like the protagonists, but we have to be able to relate to them, to feel their pain, to cheer them on.

Now, for the first time, the viewer *is* the protagonist. What does that do to the empathy problem? And what effects does it have on other story elements, such as dialog and pacing?

Well, in a sense, the empathy problem is now solved, but it has been replaced by a more difficult problem: the storyteller has lost direct control of the hero. How do you define character if you're not making the choices? The answer is open-ended: writers and designers need to find new Zen, passive-aggressive ways to think about this, and to create webs of choice that are still channeled into compelling stories. We've done a little of this with branching outcomes, but that's nothing compared to what's possible. Even with a linear story structure, we can create a world in which there are several interesting ways to make the one correct choice.

Pacing stretches, dialog shrinks

One of the biggest consequences of the protagonist flip is that pacing can be drawn out in ways we've never seen in a story before.

Let's say you're playing a trench warfare game set in World War II. You are the hero, stuck in a foxhole with bullets whizzing over your head. In a movie, you would watch a three-minute scene of this, and then get bored. In a game, it's your ass in there, and the bullets are flying at you. If you're pinned down in there for twenty minutes, so be it. The adrenaline is pumping. This is life or death.

But the dialog? The dialog generally shrinks, even from the level of our very visual cousins, the movies. Players who are wrapped up in action don't have a lot of patience for talking.

There is an exception: when dialog acts as a game mechanic, as in RPGs like *Jade Empire* or *Knights of the Old Republic*, players will embrace huge amounts of text. It's part of the flow of the action then. That works.

Looking at a game story that works

How does all this work in practice? Let's look at a little game called *Halo* through the lens of classical story structure.

Good stories have good bones. Even a fairly thin story like *Halo* is structurally sound at the act level. Is the character deep? Not very. He's a witty badass. It's thanks to Cortana that we see even that much of him. But it's got real reversals that drive interest in the game, and here's how they work.

Once upon a time, there was a genetically enhanced super-soldier.

Act I: Halo is a computer that the Covenant want to use as a weapon, so the humans, led by you, the Master Chief, have to get there first and activate its defense functions. ***There's a clear conflict***: You versus the Covenant. And it's spurred by an **inciting incident**: The Covenant have attacked your ship, and you've had to take off in the equivalent of a minivan before the ship explodes. You have no idea what happened to Captain Keyes.

Act II: As you fight the Covenant along side the Marines, you walk into a room wher e a freaked-out Marine shoots you, and you have to kill him. That's a great moment of choice, even in a linear plot.

You also see Covenant corpses that you didn't kill.

The Flood presents an interesting reversal in *Halo* and again in *Halo 2*.

Then you meet The Flood, which try to eat both you and the Covenant. ***Reversal***: The conflict isn't as clear as you thought. It has expanded, and caused your world to change. Some of your friends are now your enemies, and there are new enemies.

Act III: You are "assisted" by a hovering machine Librarian in turning on Halo's defense system, so you can use it against the Flood.

Then you find out from Cortana that you've been tricked: Halo is a machine that wanted to kill the Flood, yes, but also all othe r life forms. Now you have to destroy Halo itself. ***Reversal***: The conflict has expanded again.

The ending

You have to destroy Halo. That's an important point for all stories, and it just got even more important. Aristotle complained about the *Deus Ex Machina*, the machine that swept the hero away so the writer didn't have to think of a way out of the predicament.

Your hero has to cause the ending. He can't just watch it happen. If that's always been true, it's true in spades when the hero is the player.

Do, Show, and Tell

The old adage for writers in every previous form of story has been: *Show, don't tell*.

Now there's a new rule: *Do, don't show*.

We want our players to experience as much of the story first-hand, as the main actor in it. If there's ever an opportunity to create the story through the player's action, make that choice. If there's a part of the story that must be out of the player's control, then show it. Just telling part of the story is always your last choice, even when you're doing exposition. The priority, then, is Do, Show, and Tell, in that order.

Story and the Writer

Here's an obvious statement that people still miss: *Writers are essential to game story.* Now lots of folks will agree with that statement. What they miss is the nature of game story; they fall for the old "story is dialog" confusion, and they wait to bring in the writer after the story structure concrete has set. "Paste on some dialog, buddy, and make it clever!"

No, no, no. Don't do this.

If the writer's job is not just about pasted-on dialog, but also the deep construction of the story, that means the writer needs to be involved with the *beginning* of the project.

Common sense, right? Well, only after you know what story really is.

Story and the Whole Development Team

Here's the other half of the game story riddle: *The writer isn't everything.* If we understand the story-conflict connection, we need to also understand that the principle of antagonism might be planned by the writer, but it's created by the designers, and implemented by the programmers.

The designers will always be the most direct allies of game writers in their craft. It's the designers who are in charge of the forces of antagonism, which are the essential elements of character development and plot advancement.

Of course, it's the art and audio people who control the most perceptible parts of the story's setting. Creating a world that players believe is a critical part of good story. If you had the script to *Casablanca*, but it was cast with bad actors and badly filmed, you would no longer have a great story. You'd have a well-written story that had brilliant structure and dialog, but one that was screwed before it was completely born.

Programmers work with everybody to keep the story alive in what the novelist John Gardner would call "the vivid, continuous dream." Sloppy NIS programming or awkward animations or inconsistent world behavior just breaks the dream. Creating this world so no one notices the code is the hardest job of all.

It's not news that games are a collaborative effort. What I think development teams need to learn, though, is that even the story part is a collaborative effort.

The writer's job is to know all the details that I don't include here. Everyone else on the team, though, needs to know at least the basics I've outlined in this article. The development team has to speak a common language about story. If we all do, we stand an excellent chance of making games that will leave the last generation in the dust.

Return to the full version of this article with comments