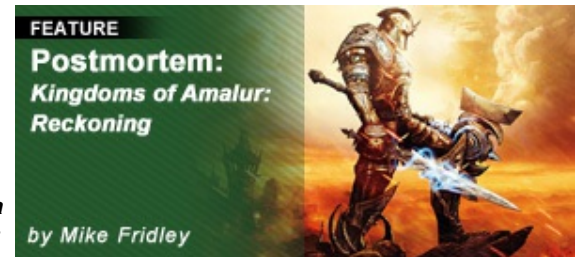


## Postmortem: *Kingdoms of Amalur: Reckoning*

By Mike Fridley

*Kingdoms of Amalur: Reckoning, the single collaboration between Big Huge Games and parent studio 38 Studios, became an inadvertent teachable moment for the games industry when rocky initial sales, mismanagement and no end of poor luck resulted in the complete closure of both companies in May 2012, just three months after the game's release. Financed in part by a loan from the state of Rhode Island, Reckoning is also a fairly unique case of a triple-A game built with the help of alternative funding.*



*In this postmortem, reprinted from the April 2012 issue of [Game Developer magazine](#), former Big Huge Games executive producer Mike Fridley walks through what went right and what went wrong with Kingdoms of Amalur: Reckoning's production leading up to its ill-fated release.*

Over five years ago, Big Huge Games set out to completely change the type of games we make. We switched from making real-time strategy games to role-playing games, and we started making games for consoles in addition to PCs. We made these changes for several reasons, and although profit was one of those reasons, it wasn't the only one. We wanted to do something crazy. We wanted to make a big open-world RPG -- pretty much the craziest project we could think of short of an MMO. But we're all big fans of the genre and thought we could find our niche in it, so we started our quest to convert the studio into an RPG ho use.

At first, our RPG project was named "Crucible" and was being published by THQ. We were making great progress on it, and THQ was happy enough with the progress that they purchased us outright; and we became an internal THQ studio. Around that time we switched some of the key features of the game and renamed the project "Ascendant." We were part of the THQ network of studios for a short period of time right up to the point that THQ started running out of money. Our big, juicy, unproven-in-the-genre studio was a prime target for them to try to sell.

With literally days left on the "close the doors" timer at the studio, THQ sold us to Curt Schilling's 38 Studios, which has R.A. Salvatore as "creator of worlds." It became clear pretty quickly that we would need to change the universe and some of the game features yet again to take advantage of Robert's genius. We changed the project name to "Mercury," which later was given the final shipping name of *Kingdoms of Amalur: Reckoning*.

For those keeping track at home, in five years we were bought and sold twice and changed the name and core features of the project three times. Needless to say, it's been a long, strange trip. The rest of the postmortem will be restricted to the two and a half years we spent working on *Reckoning* rather than the two previous false starts.

## What Went Right

### 1. Combat: RPGs don't have to have boring fights

Shortly after we came out of preproduction, we took a long, hard look at the game we were making and tried to figure out where we were going to be better than the competition. We figured that open-world RPG designs are segmented into four basic quadrants: story, character progression, exploration, and combat. We discovered that it was easy to identify the games leading the industry in story, progression, and exploration, but there was no clear title that does combat well while still meeting the expectations of the player in the other three quadrants. So we decided to go all-in on combat and change our staffing plan to really commit to making combat fun in an open-world RPG.

The game wasn't built solely around combat, but it was definitely built with our flavor of combat in mind. Everything from the minimum size of a dungeon's hallway to the number of enemies we could handle onscreen at a time was governed by the guideline that combat had to remain awesome.

Two of the other things that went right during development were direct results of this focus on awesome combat, usability testing and functional group seating.

### 2. Usability testing -- early and often

We made sure that getting feedback from real players was high on our priority list from the very beginning. Since we couldn't just release work-in-progress builds to the public and take surveys, we did the next best thing and took advantage of EA's usability lab very early in the development process. The lab at EA allowed us to pull in testers from the general public and use them for highly focused testing on systems or content that we were currently developing. For example, if we had the first pass of a crafting system in the game, we could pull in a dozen or so players for a half day and get some players feedback on whether the interface was easy to navigate or whether blacksmithing felt rewarding.

Since EA's lab recorded videos of the wrap-up sessions, we were also able to show our team what the player thought of their part of the game. If the attack chain you were working on felt bad or the quest didn't make any sense to the normal player, the team that worked on those areas of the game got to hear it straight from the consumer's mouth. That kind of direct feedback from the player really helped us fine-tune the combat system, and ultimately, the entire game.

### 3. Functional Groups: Sitting together pays off

As part of our development philosophy, we have cross-departmental teams working closely together. A lot of studios do this, but until this project we didn't really push seating functional groups of people together at BHG.

Some of that may have been because the physical structure of the studio didn't lend itself to more than three people in an office, or it could have been just old-school thinking that never changed until it was forced to change. We did eventually break down the walls (literally) and start sitting larger functional groups together in what we called "pits" around the office. For example, the combat pit has animators and designers all sitting side by side.

This way, an animator working on an attack chain could be sitting just a few feet away from the designer implementing and fiddling with it in-game. They could easily look at each other's work and offer comments or critiques very quickly.

However, functional groups are less about speeding up the feedback process and more about forcing interaction. A lot of developers are lazy about socializing or unaware of what is going on outside their office, but when the people you are directly working with are in your face all day, you start to bond with them. A lot of our functional groups became pretty tight-knit and hung out after hours, really bonding as a group. That translated into more and better communication in their work and really increased the quality of the end product.

---

### 4. Scrum development methodology

Before preproduction started on *Reckoning*, Scrum was starting to gain a lot of momentum in the game development community. I don't think that Scrum is the only way that people should be developing games these days, but after running pure Scrum for the entire development of *Reckoning*, I'm a firm believer in its methods.

I won't go in to the details of Agile development here, but the basic element of Scrum that made it so successful at BHG is the ability for the individual developer to estimate his or her own work.

The old days are gone. You can't expect producers or leads to come up with a huge waterfall of everything they thought would get done over the next three years. In the game development business, it's insane to think you have any insight into what your team will be doing one year from now. You can set major milestones with hard dates, but filling in all the details between those points is an exercise in futility.

With a basic understanding of our time metrics on content development, for example, we were able to do some good old-fashioned waterfall scheduling as well. But those waterfalls were used only to illustrate to the team the pace we'd need to maintain in order to complete the scope of work in the time allotted. For example, we could tell one of our environment artists that he had three months to create all the base pieces of a particular biome before he would start taking time away from the next biome, but we did not plan out any more detail than that. We didn't account for every tree, rock, and scrub in that biome. The actual planning of what would go in to that base set and how long it would take, meanwhile, came from the sprint planning sessions where the artist would come up with his own tasks and time estimates.

Not only did Scrum allow us to plan better, but it also gave the entire team a lot more ownership and visibility over the game. If something came up (and it always does in game development) the team knew that not completing what they had already committed to meant that it was in danger of being cut. That resulted in lots of minicrunches throughout the entire life cycle of the game instead of one humongous death march at the end of development. The team would rather work a little overtime than see something they really wanted in the game get cut or be done poorly. We still had some end-of-development crunching, so Scrum isn't a silver bullet, but it definitely helped.

Scrum allows for that day-to-day accountability that was missing for so long in game development. You understand within 24 hours of a change what is going on. More traditional development methods wouldn't catch those small losses of time for months, which would either



force us into a huge crunch at the end of development or make us cut an entire system or group of content. Also, allowing the entire team to add items to the product backlog was a big win for us.

## 5. EA Partners: A great relationship

Working with a large publisher often can be challenging. Some publishers want to be too involved in the day-to-day development decisions. Other publishers will go to the opposite extreme and remain silent milestone after milestone until your game hits Alpha, at which point they suddenly have issues with things that have been final for months, like the art style or specific gameplay systems. Fortunately, the EAP production crew was neither of those types of publishers. They gave excellent feedback throughout the development cycle and did what you really want from a publisher: They offered excellent support where we needed it most.

During our first meeting with our EAP producers, they showed up with a bunch of PowerPoints outlining all the services we could take advantage of during development -- and take advantage we did. That set the tone for the next couple of years. Any time we had a bump in the road, EAP was there asking what they could do to help. Having options like that available to you when you have an issue is a huge asset. As they are probably largely unsung for their efforts, I want to call out the major production staff players over at EAP that were our go-to guys: David Yee, Ben Smith, Craig Krstolic, and David Luoto. You guys made a lot of big problems much smaller. Thanks.

# What Went Wrong

## 1. Preproduction: Entirely Too Short

Even though we had a lot of production time during the false starts before *Reckoning*, our preproduction time for *Reckoning* itself was entirely too short. At the beginning of *Reckoning*, we were in heavy pitch mode and our goal became to get a publishing deal signed instead of spending the time to figure out the normal outputs you are looking for in the preproduction phase of development. Once we signed a deal with EAP, we needed to get into full production quickly. Or so we thought.

What we should have done was make sure we had defined everything that needed defining. We had a basic scope of content, but we hadn't done much to understand the feature set or the game's major hook. We decided to go all-in on combat fairly soon afterward, but the development budget and schedule had already been set, and so we weren't able to anticipate how many additional animators and designers we would need to bring the combat system to life.

Our content pipelines weren't fully fleshed out, and we only had basic or less than basic functionality of some of the tools we would need to create that content. But given our tight schedule and the mountain of content we had to produce for this game, we jumped in to full production with a lot of questions unanswered. Needless to say, this is not ideal.

We also hadn't really figured out the density of our content (quests, reagents, dungeons, and so on), which had long-term negative repercussions for both design and production. We frankly made "too much game," and we probably wouldn't have (at least not to that extent) if we had more time in preproduction to figure out the density question.

## 2. Tools and Pipelines: Laying down the track when the train is on its way

As I mentioned above, we didn't have a good head start on the development of our tools and pipelines early in development. We knew the basics of what we wanted once we had a feature set figured out for the game, and we knew the type and quantity of content that we were planning on making, but we really didn't have a clue how much tools work we needed to do.

A lot of the systems in the game were still very much in the blueprinting stages where we weren't even sure how they would function in the final game. The dialogue system is one example. We came out of preproduction without giving much thought to the dialogue system other than, "Yeah, we should probably do that." We didn't nail down how we would display dialogue and choices in-game and how designers would enter that data in a tool.



This put a huge amount of pressure on our tools programmers. They had to jump from tool to tool getting functionality to a point where users could actually use it right before they needed it. In a lot of instances, our tools programmers had to roll out a tool before it was fully functional or bug free because of time constraints.

Obviously, this hurts content creation. Devs would submit tool feature requests and not see any movement on them for months (or ever, in most cases) because the tools team had a ton of other issues that were higher priority. It basically meant the majority of our tools were functional but woefully inefficient.

I'm truly amazed that the tools team did as well as they did with such limited time and manpower. For the most part they were able to stay just ahead of the train, and we ended up with a suite of tools that—while still a bit disjointed— work pretty well. Things will only get better as we ramp up into preproduction on our next project.

---

### 3. Demos -- too damn many of them

People who know me are probably expecting me to go completely off the deep end and start bashing marketing and PR right now, but I'm not going to do it. I understand how hard their job is, and how that job is fundamentally motivated by events that are counter to the way developers like to work.

Developers, especially producers, like to be proactive. We make schedules, and we plan dependencies. That's how stuff gets done. Sure, problems pop up that we have to react to, but the goal is to reduce those as much as possible.

Marketing and PR are by necessity much more reactive in their work. If you sat down and tried to formulate the next two years of a marketing plan with the same level of detail that a development schedule has, it would be full of every single possibility that could arise while trying to sell the game. A very small percentage of those opportunities would be sure things. There are some major milestones that can be planned well in advance, such as E3, but you have to remain flexible and opportunistic with a new IP to ensure you follow through on opportunities as they open up. Of course, that means that they'll come to game developers, say "We have this great opportunity, but we'll need a brand-new demo and 30 never-before- seen screenshots by the end of the month," and drive us crazy. Being a brand-new IP, we were aware we couldn't get away with a single demo at E3 and a few dozen screenshots and videos. We knew we had to get our awareness up so people would start paying attention to our game. Marketing decided that the best way to do that was show the press as many different things about the game as possible over a very long period of time.



I'm trying to remember the number of demos we had to create over the development cycle of *Reckoning*, and I honestly end up losing count. Doing a demo for us was a pretty major undertaking, like it is for almost everyone in the business. You're basically taking content and systems that were meant to be first or second pass at a certain point in the schedule and bump it all up to shippable quality long before it's supposed to be shippable quality. This results in a lot of work that is just thrown out because the real content and the real systems end up changing a few weeks or months later. And there is nothing quite as frustrating as working overtime on something that you know is just going to be seen once and then thrown away.

The consumer demo was another hurdle to overcome. There was no way we were going to be able to complete work on the game and create a downloadable demo in parallel. We just didn't have the time. In the end, we had to outsource the demo, and they had to build something with old code and not a lot of time. The result was a buggy experience, but still an experience that a lot of fans enjoyed.

In the future, we'll be sure to plan plenty of time and budget for multiple press demos and work on a better plan to either build the downloadable demo ourselves or better support outsourcers.

### 4. Main quest -- not enough of it fleshed out early enough

In *Reckoning*, a lot of the custom content work we did focused on the main quest. There is a lot of custom content throughout the entire game, but we knew we really wanted to spend more of our time on the main quest, as most players would see the majority of that line. A big chunk of that custom work was cinematics.

Our cinematic team is awesome but very small. Much like most of our teams on the project, they have to produce more content than would normally be expected for a team that size. Not locking down the major beats of the main quest early really hurt the cinematics team.

Going from a storyboard to a finished cinematic takes a long time. Once a cinematic is finished, it is very costly to change. Because we weren't locked down on the major cinematic moments in the game for so long, we ended up having to cut several cinematics that we really wanted to include. The cinematics we have in the game are awesome, and we got all the major beats that we wanted, but we definitely wanted more.

---

### 5. Upper management shuffle

I should probably give a little background on this point before I get into the meat of the issue. When we were purchased by 38 Studios, we retained all of our senior management and development staff. The BHG studio was reporting to 38 Studios corporate, then based in Massachusetts.

In July 2010, about a year into the development of *Reckoning*, five of the most senior studio management team at BHG left the company. This easily could have ended *Reckoning* in a lot of different ways -- the studio could have closed, or the game itself could have become a mess of unrecognizable trash. Luckily, that was not the case. Several people in the BHG studio stepped up to fill the leadership void so we could continue to make the game you'll see at release.

The culture of this studio is unlike anything I've seen anywhere else. I don't want to say we're a family, because that has become cliché. Curt Schilling is fond of using sports metaphor; I'm more fond of military ones. To me, what drives the folks at BHG to do better and go that extra mile is our loyalty to each other. To use a military metaphor, it's like fighting a war, but without all the courage and killing. When you're in the thick of battle, you don't fight a war for the general back at HQ. You fight it because you don't want to let down your buddy next to you in the foxhole. We had other motivations, like wanting our fans to have a great game, but our day-to-day drive came from not wanting to fail each other.

If anything, the senior-management shuffle may have even increased the resolve of the studio to finish this game. It ended up being yet another obstacle that the fates threw at us, and we'd be damned if we weren't going to get past it and make an awesome game.

## Conclusion

I can't possibly hope to cover in this article all the things we did right and wrong on a project this size. It was a huge undertaking to make a game of this scope, and we learned a lot along the way. The studio has definitely leveled up as a whole, and we'll be heading into our next project with a better understanding of our game and with better tools and pipelines to make that game.

In the end, any success that this new IP will enjoy has largely been brought into being through the force of will and talent of its developers. We were understaffed and underfunded, but we simply had too much personal skin in the game to let it fail. The team that finished this game did it through dedication to what we all believed could be the next big single-player RPG franchise. I have never seen a team with so much ownership of a game as this one. Many nights were spent working on some minute detail simply because that developer didn't want to let something that wasn't perfect into their game. That kind of passion is a rare commodity to find in a handful of people -- much less an entire studio -- and I can't wait to see what we can accomplish next.

[Return to the full version of this article](#)

Copyright © 2016 UBM Tech, All rights reserved