

Postmortem: Stoic Studio's The Banner Saga 2

 gamasutra.com/view/news/274238/Postmortem_Stoic_Studios_The_Banner_Saga_2.php

John Watson is the co-owner and technical director of [Stoic Studio](#). He has been programming since he was 6. He came to Stoic after doing some work on the Hubble at NASA, and serving as lead combat programmer on Bioware's The Old Republic.

INTRODUCTION

[Banner Saga 2](#) is the second part of a planned trilogy. After the successful launch of *Banner Saga 1*, we had a mighty tailwind and our sails were full.

However, we soon hit some challenges which slowed us down and distracted us for some time.

After what amounted to two years of brutal crunch to get the first game launched we took a much-needed break. The three of us variously rested, traveled and caught up on the parts of our lives that had been suppressed or neglected.

Then, a series of technical projects, including localization and porting, consumed most of John Watson's time for the next year. Alex Thomas, the writer of the first game, decided to set off on his own to work on his own project. We brought Drew McGee on to write the next game, and he and Arnie Jorgensen began preproduction and planning. We had some false starts, but a year later, with E3 2015 looming on the horizon, we started production in earnest. We brought Matthew Rhoades onto the team for technical design work and that E3 showing was our first fully fleshed out vertical slice of the game.

WHAT WENT WELL

1) Everyone Worked Hard

With such a small team, this is incredibly important. There were four of us working full time on the game, with important collaborations occurring at various times throughout. Composer Austin Wintory is involved from day one in blocking out the story arc with us. Igor Artyomenko, a fantastic artist hailing from Kazakhstan, came in to help Arnie with the art load. KPow Audio are involved in planning. Powerhouse Animation needs to get their workload in the pipeline early, because they could easily become a bottleneck otherwise.

The four full-timers spent every day collaborating on Slack, with impromptu video chat meetings throughout the day, and a daily kickoff video meeting each morning at 9 a.m. We tracked our tasks and milestones carefully, and everyone did their best to maintain forward momentum and prevent blocking anybody else.

Each person certainly has their expertise and role to fill, but no role is rigidly partitioned from everyone else. The team's maturity allowed us to have candid conversations and discussions without worry about hurting people's feelings or causing ill will.



3) High Morale

Seeing the game come together was very exciting for everyone, and most of the time the mood was upbeat and optimistic. We were all confident that we were making something good, and that we would be able to pull it off. The biggest question in development was about how long it would take to complete. Having a confident, optimistic outlook toward the game as a whole helped us weather the difficult periods of heavy workload and unfortunate crunch.



4) Flexibility

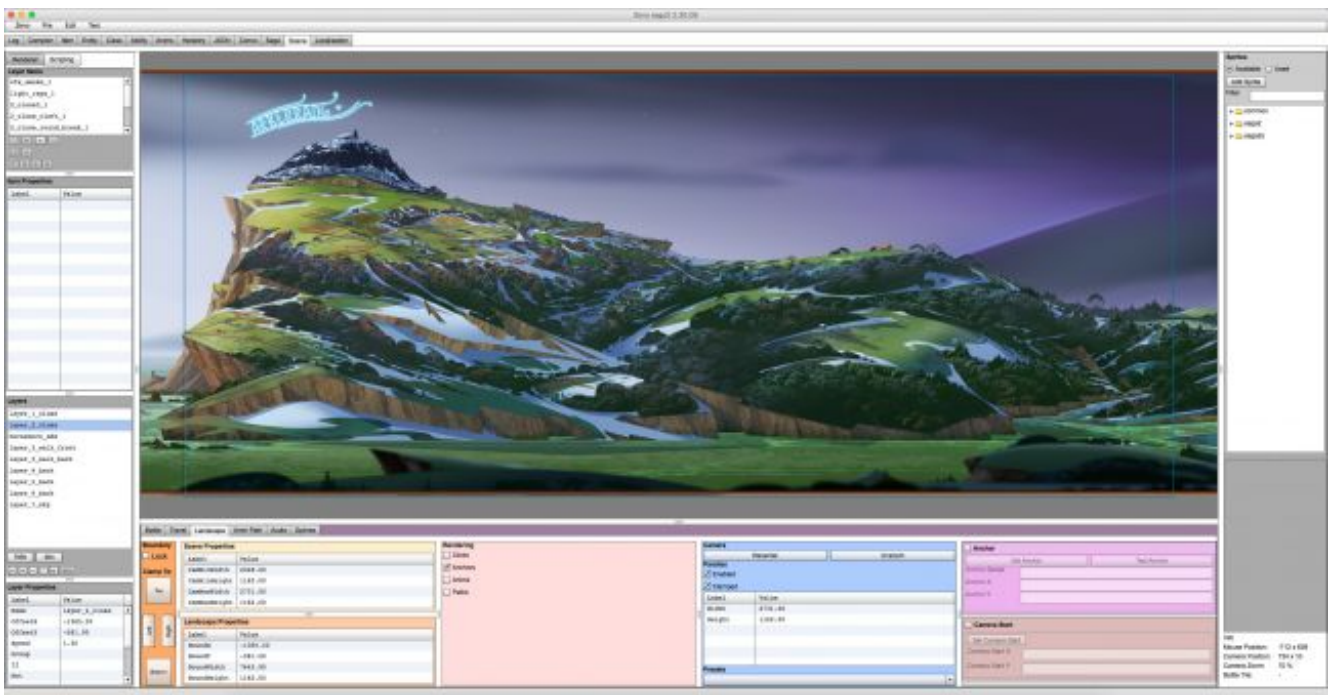
As a team, we were able to switch gears quickly and make major changes to the game where needed. Something not coming together well, or taking longer than expected? Seriously consider cutting it. Chapter flow not smooth? Re-arrange, split, and re-combine chapter boundaries. Game mechanic not fun enough? Redesign it.

Some examples:

We originally made horseborn units rectangular 1x2 tile units, thinking it would add some refreshing variety to the battlefield. However, after much effort, they were still a little wonky and not really adding anything but confusion. They got cut back to squares after all that work.

Chapter 10 turned into a mega-chapter with the highly complex events of Lundar being only the midpoint. Chapter 10 got split, chapter 12 got cut, and chapter 14 got re-arranged.

Our original design for higher-ranked units involved a common mechanic of resetting stats to their minimum values and allowing the player to slowly re-build them up. However, once in the game, we really didn't like the way it played. We went over about four different iterations before we came up with the Talent system, which we all really feel good about.



5) Pre-Existing Tech

We built *Banner Saga 2* with the same engine we've been using all along. This allowed us to hit the ground running with production, and re-use all the tools with which we are familiar. We built quite a few new features onto the engine. If you slice off the year we spent working on localization, porting, and slow pre-production, we really made the entire game in less than a year using our existing tech.

The Banner Saga Engine has a nice mature pipeline allowing us to quickly make changes and see them immediately in-game without needing a new build at all. Our automated build system operates in the cloud and serves out new builds whenever the code changes, with a turnaround time of less than five minutes. The builds are served to the content developers directly through the Steam client on a private branch, which means no manual intervention is needed to get new builds or make sure you have the latest build. This all streamlines things quite a bit for everyone.

We had already gone through an entire iteration of music and sound implementation, and that experience

informed both Austin Wintory's and Kpow's efforts on *Banner Saga 2*. Powerhouse animation had created an immense number of animations for the first game, so when we gave them another sizeable workload, they were able to ramp up immediately.



WHAT WENT POORLY

1) Tools

Since John's time was almost entirely consumed in creating new features and supporting localization and porting efforts, the content development tools didn't advance as much as they could have. Many of the irritants and inconveniences of the tools from the first game persisted throughout development.

"Many of the irritants and inconveniences of the tools from the first game persisted throughout development."

The game content is almost entirely data-driven, but some of the systems, notably the ability system, has no tool to assist in its data creation. So combat abilities, some of the most complex behaviors in the game, have to be created by editing JSON in a text editor. There are other subsystems that likewise have no tool, and require manual JSON intervention.

The game dialog and story was written in Inkle Writer, a free tool by our friends at Inkle who are responsible for the fantastic game *80 Days*. Our heavy usage of Inkle Writer, however, has exceed its intended limits and causes problems on a regular basis. As much as Inkle Writer has helped us get the game off the ground, it's time for *Banner Saga 3* to upgrade.

The game engine itself is based on Adobe AIR, which is a fairly capable piece of technology. AIR provides a compiler and runtime for ActionScript 3 bytecode, and a library of application interfaces for input, file handling, networking, and simple graphics. It also provides a full 3D interface for doing GPU accelerated rendering. Furthermore, it compiles natively to Windows, Mac OS X, iOS, and Android, and allows you to link with as much custom native code as you like. The unfortunate side of this is that Adobe puts very little priority on supporting developers, and as a result the AIR technology is difficult to use, the tools are perpetually buggy, and it appears

to be slowly dying through neglect. The kicker is that it's a total black box, and no source code license is available, so when something goes wrong, you may well just be screwed.

The difficulties with Adobe AIR manifest primarily on the Android platform, which is a nightmare scenario already due to the extensive device and OS fragmentation. However, it extends also the the console porting world. Since Adobe AIR isn't being actively pushed forward onto new platforms, it means that we needed to take a different approach with porting. Fortunately, based on our years of experience with Autodesk ScaleForm, we knew that we could wrap the game entirely in ScaleForm. The downside was that wrapping an entire game of this size is not a typical use case for ScaleForm. It's generally used in Triple-A studios to provide cool GUIs and HUDs in games. We definitely pushed ScaleForm to its limits and beyond, and had to work closely with Autodesk throughout to address issues. More on this later.



2) Multitasking & Bottlenecks

A downside to having a small team is that each member is responsible for a wide variety of tasks. When you are multitasking, it is very likely that the neglected or down-priority tasks are blocking other people from getting their work done effectively or on time. As with most games, each of the elements and systems depend on most of the others, inevitably creating circular dependencies and feedback loops. While desirable from a gameplay standpoint, these interlocking dependencies can create production challenges. We're hoping to be able to hire a full time producer for *Banner Saga 3* to help us keep on top of these issues.

"When a small team is multitasking, it is very likely that the neglected or down-priority tasks are blocking other people from getting their work done effectively or on time."

One of the things that consumed a heap of John's time was console porting. At the beginning of the process, we contacted Autodesk and got several recommendations for developers with ScaleForm experience. We then investigated them and ended up hiring 3 different developers to do a 2-week prototype. When one of the 3 ended up making much more progress in that time, we felt very clever and hired them to do the rest of the porting.

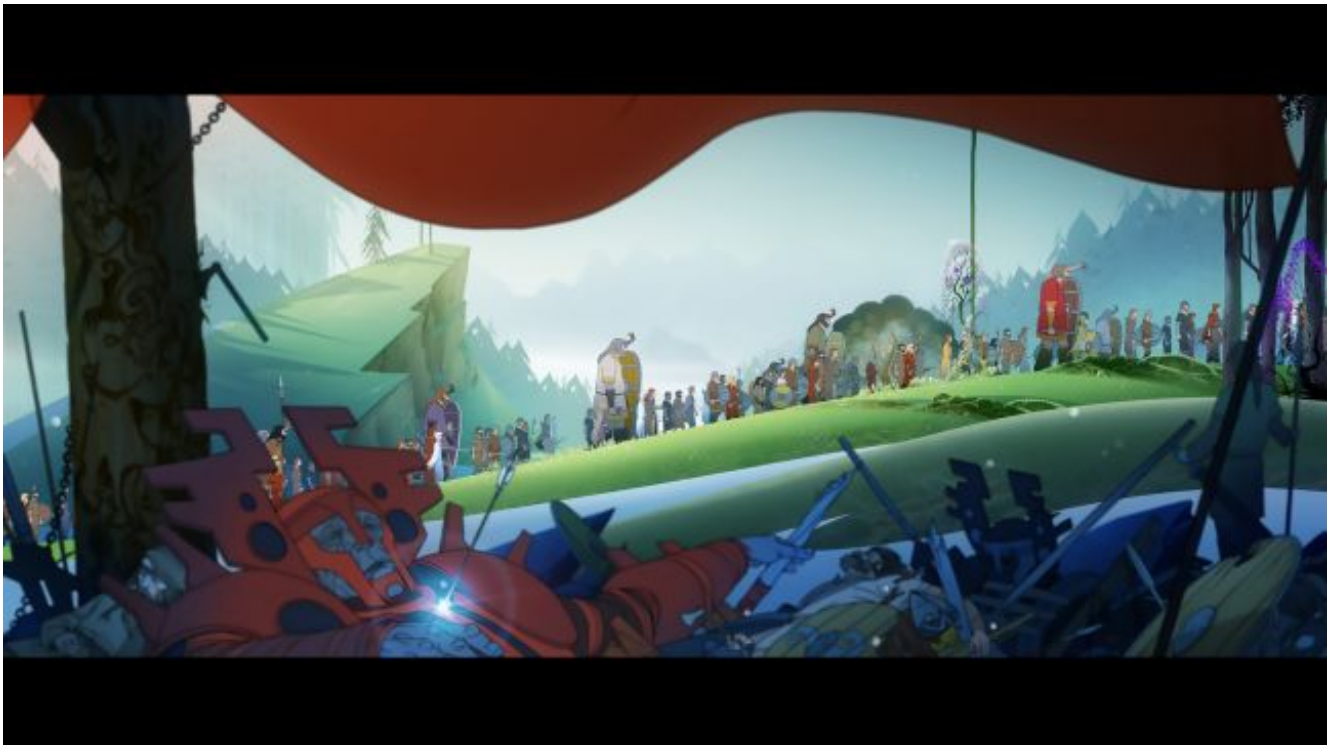
This particular console porting house was not able to do gameplay enhancements like implementing the gamepad controller interfaces. We were also very interested in keeping that in-house, as we really wanted to ensure that the gamepad played well. We did work with a consultant to point us in the right direction and give us a basic controller vocabulary for the game, but after that point, John took over implementing the gamepad controls for the entire game. It was a painstaking process, and lengthy, but we ended up happy with the way it turned out.

In the meantime, the console porting company missed milestone after milestone. The bulk of their team left the company, leaving the entire weight of the project on a single capable developer. The company hired several short term programmers, each of which jumped ship shortly after coming on board. After 6 months of turnover and churn, the porting company finally threw in the towel and abandoned their contract. In our eternal optimism, we had made the mistake of paying them for more milestones that they had actually delivered, hoping to keep the ball rolling. The end result is that the porting efforts were now a year behind schedule and we were out what amounted to about one year's salary. The porting company was bankrupt and broke, so there was no feasible way of recovering the lost money. An expensive lesson was learned.

As soon as the first major milestone was missed, we should have laid down an ultimatum and pulled the plug. However, optimism always made us think we might be farther along than we were. The existing relationship had taken time to establish, and we feared having to go back to square one and find a new porting partner. In hindsight however, we ended up having to do that anyway, too many months and dollars later.

Once the console porting efforts went down the drain, we scrambled to find someone else to resume the work. We were hoping to hit a date with Sony and sim-ship with Vita. Although PS4 had made some progress (it was runnable, albeit not fully playable), the Vita port had made zero progress. The Vita port being the most risky due to performance and resource constraints, we decided to back burner it and get a team working on PS4 and Xbox One immediately. That team ended up being Shiny Shoe, who have done a fantastic job ever since.

During all of this work and drama, John's time was effectively consumed, preventing him from making substantial progress on *Banner Saga 2*'s numerous new features, and preventing the kind of internal tool improvements that we need.



3) Communication

Our team works completely remotely. Not a single member is in the same room with another. On top of that, we

are spread out across the entire globe, from Australia, every time zone of North America, Europe, and reaching all the way to Russia. Despite our best efforts at creating a routine and maintaining constant communication, it often breaks down. The biggest problem with remoting is not being sure exactly when someone will get back to you with answers to questions or other requirements. Without diligent status updating, it's impossible to tell if someone is at their desk or not. We'd like to come up with a way to more closely replicate the feeling of being in a physical studio together, without doing something invasive like leaving the video chat room up at all times.

"The biggest problem with remoting is not being sure exactly when someone will get back to you with answers to questions or other requirements."

One month after the launch of *Banner Saga 2*, the entire team got together in Austin Texas for a day of retrospective. We hired a producer to run the retrospective, and the 5 of us went through the day long exercise.

The producer led us through a variety of activities, designed primarily to capture all of our knowledge and feelings about the way the project had gone. What parts made us happy, or unhappy? What potential solutions could we start thinking about?

This was a very useful day, in which we were all able to get on the same page. There were even some surprising results. We're going to use this exercise as an important input when we start planning out *Banner Saga 3* in earnest, and hopefully bringing the producer into the team for this will make his ramp-up happen much more quickly and smoothly.



4) Crunch

For all of our efforts to avoid it, we ended up once again falling into months of crunch. While I do believe that nothing great can be made without sacrifice, I am in opposition to the idea of jettisoning all the other important parts of your life for long periods of time. We kept pushing toward milestones, and the milestones kept slipping. After the initial development period, we had more and more difficulty in identifying parts of the game we could cut. It seemed like we had cut it as far as we could, and everything else just had to make it in by pure force.

For us, there are two main pressures that create the motivation to crunch. The first is timing. We have a desire to get the next part of the trilogy out to the players while the idea is fresh in their minds. We don't want to make

people wait too long to play the next part. We probably put too much weight on this, but it's hard to shake the feeling.

The other pressure is money. Making a game like *Banner Saga* takes a considerable amount of money. Four full time team members for a year or two, a dozen other people helping out in various capacities, extensive animation work, marketing and promotion, and an original orchestral score recording all add up quickly. So during development we are always looking to the future, projecting our horizon. If it looks like we are going to run out of money in, say, 18 months, then we absolutely must finish the game in advance of that. It would be tragic for us to get *almost there* and then run out of money to pay the team.



5) Professional Development

One of the side effects of our demanding schedule was that professional development took a back seat. Everyone on the team is spending all their waking hours on a particular game. This means that we aren't learning new technologies, trying out new ideas, and finding different ways to express ourselves. Everyone who does creative work knows how important it is to always branch out and push your boundaries. Doing technical work requires always learning new tools and techniques. Socializing with other developers, and going to events like GDC for exposure to new ideas is critical.

During our retrospective we came up with several great ideas for improving this area. There were the obvious low hanging fruits like carving out a week for GDC come hell or high water. Too often it seems like GDC comes at a time when the pressures of the project either prevent everyone from attending, or even if attended, the attendees spend most of their time dealing with the project. For instance, John went to GDC this past year for the week, but ended up spending the majority of his time in the AirB&B with a laptop, working. We want to foster a culture that understands GDC is a 'sacred' week that should be prioritized.

Another idea was a weekly game time, where the four or five of us, although spread remotely, take an afternoon to play some games together. It's quite common for us as game developers to spend all of our time developing and too little time playing. Establishing a culture of regular playtime together will mitigate this, as well as improve team communication as a whole.

Yet another idea was to establish regular game jams, where the team sets aside the main project for a few days, and completes development of a small game together. This exercise really allows you to try out new ideas, stretch your wings, take some risks, and learn some new things. We did something like this prior to the

retrospective. We found ourselves sitting around doing not much of anything after launch. We had two weeks left before our company-wide vacation started. So we brainstormed on how we might take the considerable assets at our disposal and refactor them into something fun. We spent that time creating a battle-focused mode of play called 'Survival Mode' which is fun and challenging, and is being beta-tested at this time on Steam.



CLOSING THOUGHTS

Going into production of *Banner Saga 3*, we have many important lessons under our belts. We all feel that we are perfectly positioned to continue. Our momentum is considerable, and our familiarity with the tools is at an all time high.

The biggest challenges will be to continue to improve remote team communication, and to properly project manage and plan. We must scope the next project to fit the time we have available. If we can slow down a bit and reduce the amount of crunching to a minimum, we can hopefully improve our quality of life across the board.

Overall, the development of *Banner Saga 2* was a wonderful experience and a dream come true for all of us. We are fortunate to be working independently at a time when it is feasible to do so, and on a project that has been so well received by our audience.