

Baldur's Gate II: The Anatomy of a Sequel

By Ray Muzyka

□

Introduction

Just over two years ago *Baldur's Gate* was released. It was the culmination of nearly 90 man-years of work by a number of inexperienced, but very talented and creative individuals at BioWare. BioWare was a Canadian game developer, with only a single title (*Shattered Steel*) to its credit prior to *Baldur's Gate*. Published by Black Isle Studios (the internal RPG division of Interplay Productions), *Baldur's Gate* was the next in a line of famed RPGs that included the venerable *Bard's Tale* and the highly respected *Fallout*, both developed by Black Isle. *Baldur's Gate* beat the odds and was both a critical and a commercial success (it collected nearly all of the industry's PC RPG of the year awards and a few "Game of Year" awards, and it has since sold about 1.5 million copies worldwide).

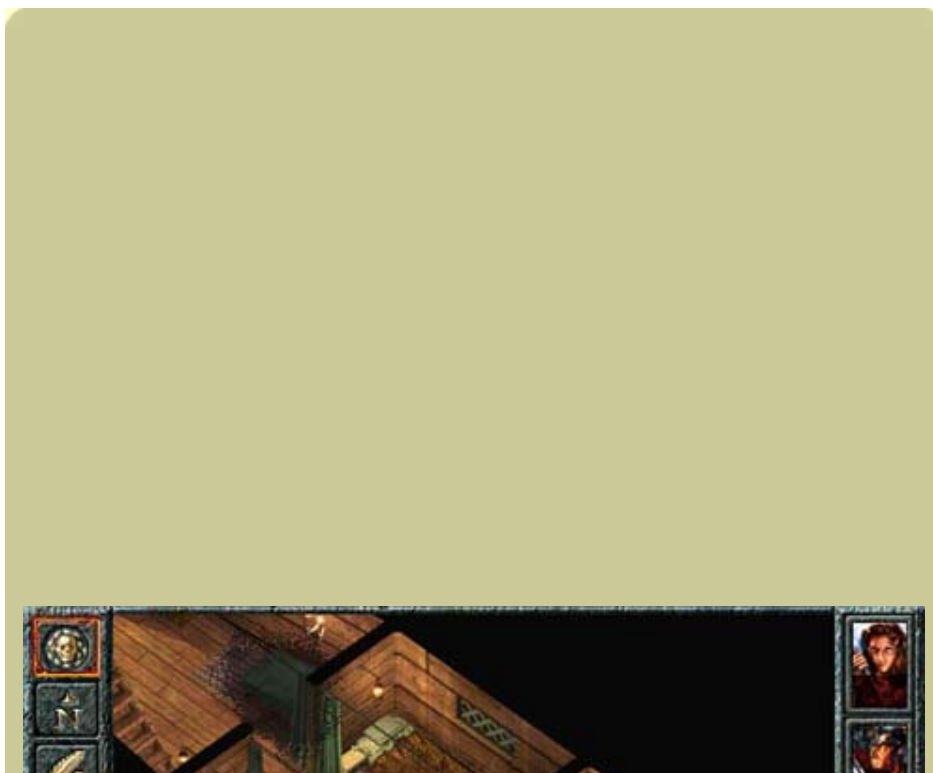
After the resounding success of *Baldur's Gate*, BioWare began the development of *Baldur's Gate II* and set out to prove the magic of *Baldur's Gate* could not only be repeated, but that a great game could be made even better.

The Challenge

Building an excellent sequel is not nearly as easy as it may sound. In making *BG2* we knew everyone would be looking very carefully at the result. Facing comparisons with multiple great games using our BioWare Infinity Engine like *Baldur's Gate*, *Icewind Dale* and *Planescape: Torment* (the two latter games both developed by our publisher's Black Isle Studios after they licensed the BioWare Infinity Engine for this purpose), our work was cut out for us.

In developing a sequel, you must start with the right philosophy: the goal must be to make the game better, and not just to make the same game over again. You also need a mechanism to quantify your previous mistakes and learn from them. If you don't make a point of figuring out what you did wrong last time, you're not likely to fix it the second time around.

At BioWare we have learned to do thorough post-project reviews to analyze both the strong and weak development areas of our projects. The best way to start a sequel is to review and improve upon the processes used in the original. In the case of the original *Baldur's Gate*, we felt we didn't have adequate time to reach our design goals; we were simultaneously developing the BioWare Infinity Engine while creating the content in *Baldur's Gate*. This led to extreme pressure to have simple areas and game design. With *Baldur's Gate II*, we resolved to allow the designers adequate time to allow the game to reach its full potential. We had learned to review our previous projects, learn from our mistakes, and apply these solutions to all new and ongoing projects.





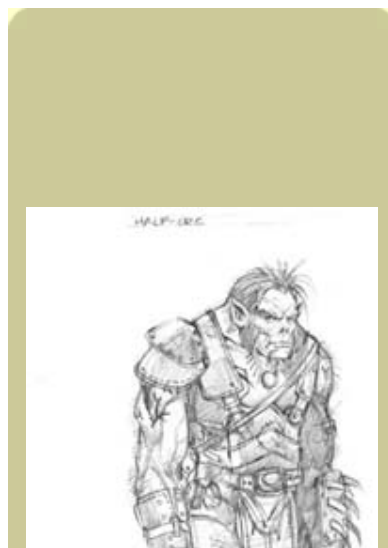
The success of the original *Baldur's Gate* set the bar for the sequel quite high.

Make a Feature List

Part of any design phase should be creating a feature list. Thanks to the AD&D license attached to *Baldur's Gate II*, there were thousands of possible features we could add to the game. This being the case, our challenge was to determine which features to add. We used two routes - the first was to make an internal list (generated by BioWare and our publisher Black Isle/Interplay) of what was feasible and reasonable considering the engine, and the second was to ask the fans what they wanted to see. Fortunately in the case of *BG2*, a number of fans on the newsgroups already had done much of the work for us, and compiled a list of what they wanted to see in *BG2*. This list gave us a sense of what our hard core fans were expecting, and helped point us in the proper direction. The major feature list that we eventually came up with looked like this.

1. Higher resolution (800x600 and up).
2. 3D support for 3D graphics cards.
3. Non-pausing dialogue in multiplayer.
4. Drop off panels in the interface.
5. Multiple new character kits (subclasses) for all classes.
6. Faster character movement.
7. Dual wielding of weapons.
8. Improved (more detailed and more frames) character animation.
9. Inclusion of all of the 'famous' AD&D monsters, including the most famous of all, the dragon.
10. Spells up to 9th level.
11. Streamlined Journal, annotatable Map.
12. Deathmatch mode.
13. Character interaction on par with *Final Fantasy*.
14. Character romances.
15. Definite evil and good paths to allow for alignment-based roleplaying.

We added several features as the game went on, including a new race (the half-orc) and three new classes (sorcerer, monk and barbarian) plus a myriad of character kits. Very few features had to be cut or weren't implemented in a fashion that worked as well as we hoped they would.





The half-orc was one of several new features added as the game went on.

One thing we did not do was to rank the game features into simple classes such as essential, important, less important, etc. When it came to making feature decisions we opted to keep as many as we could but we didn't have an agreed upon list or mechanism to resolve the decisions. Fortunately we were using a mature engine that we had developed so adding most features was relatively easy. However we certainly can't claim that all of our decisions were enlightened.

Deathmatch was a feature that should have been cut early on, but persisted until close to the end of the project. It then became obvious that the ship date would have to move back in order to accommodate deathmatch. Considering that multi-player code was some of the most fragile in the engine, and deathmatch wasn't being very well received by QA, we reluctantly decided to cut it.

Non-pausing dialogue was the most problematic feature. Early in the project it was cut due to time constraints. In early 2000 we decided to add the feature back in, as the amount of dialogue in the game was making multi-player very frustrating. Looking back, this was probably the wrong decision. Most of the dialogue had already been written under the assumption that the game paused in dialogue mode. We had to create a hybrid system where plot critical dialogue would still pause. Our changes to the multi-player code also created several instabilities that led to some very late nights for the programmers.

The lessons we learned included the need to make a feature list, ranking features in order of importance and as well noting which features could be cut if needed. We also learned to not reverse decisions lightly - reversing a development decision should be considered only if it is absolutely essential and then only after being carefully considered.

The Design Guidelines

One thing we definitely didn't want to do with *Baldur's Gate II* was make some of the same design mistakes that we had followed with the original game. Since some of our team members were brand new, and since many of our (the authors') memories seemed rather porous, we decided to make up a set of 'guidelines'. While each department had its own set of guidelines, the level design guideline was by far the largest, as it was the area with the most room for improvement in *Baldur's Gate II*. Below is a truncated version of the guideline used by the game designers:

Basic Design Rules:

1. The player must always feel as if it is HIS actions that are making him succeed. He should feel that through his smart decisions and actions that he has solved a puzzle or battle.
2. The player must feel as if he is having an effect on the environment. His actions are making a VERY visible difference with how things are running in the game world. His actions have consequences.
3. When designing, a good and evil path must be considered. Several plots should be marked as changing according to the player's alignment.

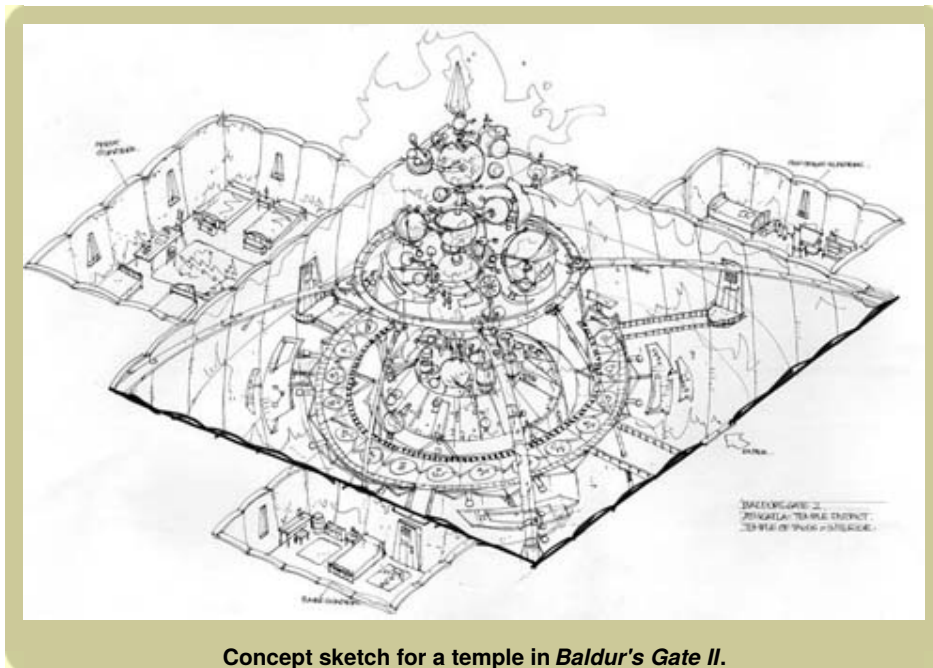
Story Design:

1. The story should always make the player the focus. The player is integral to the plot, and all events should revolve around him.
2. It is important that the player is kept informed about the progress of the villain. This can be done through cutscenes during chapter transitions, or through integrating him/her into the main plot from time to time.
3. It is important that there be a twist in the story (or even more than one). This is where a revelation is made to the player that makes him reevaluate what's going on with the story. All of the twists should involve the main player. Twists that the player figures out on his own are also better.
4. It is good to keep the ending of the story open ended, especially if a sequel or expansion packs are being planned.

Environment Design:

1. The game world should be divided into chapters. Each chapter should be of equal size and exploration potential. Each of these chapters should have a rather obvious goal, but one that the player can achieve in any fashion that he wants.
2. Certain areas should be marked as core areas. These areas are usually towns or similar places that the player will be returning to often. Core areas should change as the environment changes. As the player performs actions in other areas, there should be changes to reflect this in the core areas.

3. The player must always feel that he or she is exploring interesting areas. This means that areas always need to have a unique feel to the art.
4. It is not a good idea to have the player moving between areas often. This becomes annoying. Plots should be kept within the confines of a single area.
5. It's good to show things to the player that he cannot use or places that he cannot go. Later on, these objects or places will become enabled.



Game Systems Design:

1. A well thought out reward system must be created. The player should be rewarded OFTEN during the course of the game. These rewards can come in the form of XP, items, story rewards, new spells, new monsters, new art, romances, etc.
2. It is important that the player is able to personalize his character. This means that he should feel that the character he is playing is his own.
3. It is important that the world reflect the ways in which the player has personalized his character.

Writing Guidelines:

1. No modern day profanity. This excludes lesser profanity, i.e. damn, hell, bitch, bastard.
2. Each of the dialogue nodes (dialogue piece) spoken by an NPC should be limited to two lines. Only in VERY RARE circumstances are more than two used.
3. All character responses should be one line when they appear in the game. There should be no reason for them to be longer than this.
4. Try not to use accents in dialogue. For certain characters (Elminster, sailor types) it is all right, but for the most part it should be avoided.
5. When using player choices, try to keep the visible number to about three. Two or four are all right, but only when really necessary.
6. When an NPC talks directly to the main player, this should be noted for scripting purposes. Other dialogue should be included for when someone other than the main player talks to this character.
7. Random dialogue should be avoided, or at least used sparingly. Commoners should have only a few random dialogue lines, but there should be several different commoners to talk with.

There are a few important points to be made regarding these guidelines. First, they were a work in progress, and the version you see here is not the version that we used at the beginning of the development process. Second, we considered them as a set of guidelines, not the absolute law. If a situation dictated the guidelines not be followed, and it made sense to do so, the designers were given the latitude needed to follow their creative goals. Sometimes this worked and at other times it didn't.





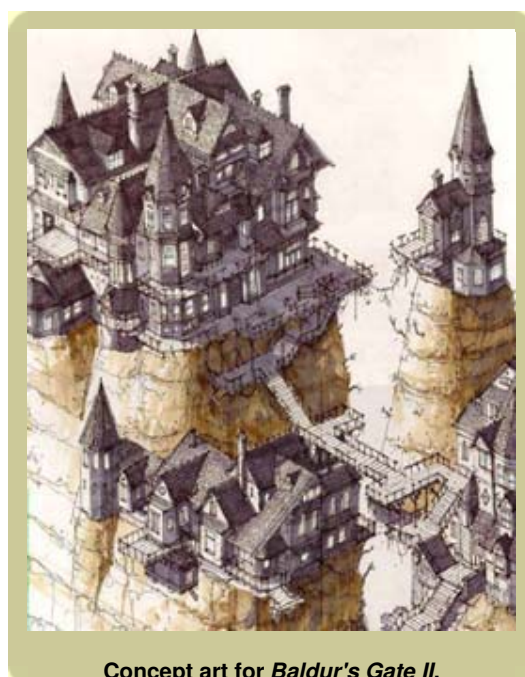
In retrospect, it would have been very helpful to have this finished set of guidelines at the start of the project, rather than at the end. A number of decisions that were made very early on in the project were not followed and could not be undone. Most noteworthy was Chapter 2 of the game – it included a story segment that was similar to those in other chapters, but in Chapter 2 the player could also access all of the class-specific subquests. This led to Chapter 2 potentially dwarfing all other chapters in length because the players could spend 60 hours doing subquests. We needed to put the subquests at a point where all players could access them equally, but end result was that it bloated an early section of the game. In the end there was nothing we could do to fix the chapter disparity so we simply worked around it.

Another major problem area related to the programming constraints that had been established early on in the project not being followed in all cases by the other departments (design and art). For example, we had audio file maximum sizes established, and maximum sizes (both in area and number of frames of animation) set for spell effects, as well as a maximum area size of six by six 640x480 areas and a maximum number of characters per area. In some cases these guidelines were not followed which lead to framerate slowdowns at some points in the game. This lead to some frantic optimization efforts needed to get the game playing faster near the end of the development cycle when little time was available neither to identify the problem areas nor to fix them.

The lesson we learned here was to establish development guidelines, follow them, but also continually work on refining the guidelines based on the progress of the game.

Improve the Pipeline

One of the most important elements of any game development process is the art and content pipeline. The pipeline is the means by which artists and designers put their content into the game. Essentially the pipeline for *Baldur's Gate II* remained the same as it was in the original *Baldur's Gate*. In *BG1* the pipeline started off looking rather nebulous, but had solidified into a concrete operation by game's end. With *Tales of the Sword Coast* (the *BG* expansion) we had another 4 months to refine the entire content creation process.



Concept art for *Baldur's Gate II*.

There are four basic divisions in the *Baldur's Gate* pipeline: programming features, movies, in game animations and game levels. The largest and most complex of these is the game level pipeline. Going into *BG2* we had an 8-stage process that we followed when creating levels for

the game. The process for creating a game level was:

1. Designers map out an area and write up a description.
2. Concept artist draws an isometric concept of the level.
3. Models are created for the level.
4. Models are placed within the level and then textured.
5. The level is dressed with smaller objects (barrels and chairs). Lighting is done for the level, and then any final tweaks are completed.
6. The art piece is given to the designers so that the clipping, luminosity, height and search map can all be done.
7. Creatures, items, traps and triggers are all added to the level.
8. The scripting for the level is completed.

By the end of the project we had found several weaknesses in the overall procedure. We found that we needed a better way to document the changes that were made to a game level during development. We had tried to keep our word documents as up to date as possible, but with the amount of people involved, and the enormity of the game, it was nearly impossible to keep these documents completely accurate. Some elements of the large team worked independently from each other - designers sometimes didn't interface adequately with artists resulting in missing elements in the game areas and different naming conventions between art and design, potentially a huge problem when you consider that *BG2* had hundreds of areas and thousands of creatures and pieces of individual art. Improvement of the integration between different disciplines (programming, art, design, quality assurance, audio, etc.) is a now goal for all of our projects. For example in *Neverwinter Nights* we have a database (called the Event editor) where we keep track of all changes to a game level so that developers from various areas can all simultaneously be aware of the specific status of game content.

Another oversight in the *Baldur's Gate II* level design process included the lack of a specific early testing stage (effectively the ninth stage of area development). Early testing of a game level would have allowed us to make changes and tweaks while the level was being developed, when it was still relatively easy to modify, rather than doing it in the final QA pass. This would have streamlined the final testing process. Instead we didn't start testing until large sections of the game were fully content complete. While *Baldur's Gate II* was in development we added an in house QA department and to BioWare in order do more early testing. We can now run game levels through this department as soon as we have a working version of the level and fine tune it earlier, rather than later. Much QA support also was provided by our publisher Black Isle/Interplay, in that some QA testers visited BioWare for the last few months of the project and additional QA testing occurred down at Black Isle/Interplay.

Interestingly, we did such an excellent job at automating the level development process that there was little time to review a game level as it went through the stages. A designer might submit a level description and receive it, art complete, a month later ready for scripting, but missing some key features (almost always a door). We would then have to determine whether the omission was important enough to have the art piece redone, or whether we could simply tweak the design of the level to fit the finished art. Again, this relates back to the issue of integration of the disciplines, something we will perpetually work on with our large scale RPG projects.



***Baldur's Gate II* inventory screen.**

During the development of *Baldur's Gate II* we added Line Producers to assist the three Producers in maintaining team communication and task tracking. By its end, *Baldur's Gate II* had a Line Producer/Designer assigned to making builds of the game and managing *BG2*'s gigantic resources and another Line Producer responsible for the thousands of bugs on the bug list. We added a third Line Producer near the very end of the project to work on compatibility issues and to help with answering technical questions on the bginfo@bioware.com support email.

We learned to make sure all elements of the team are talking to each other and working as a group, rather than as a bunch of individuals!

Manage the Mid-Project

During the development of any game, no matter how cool, there comes a time where people have been working on it for a long time and they start to tire of it. This mid-project doldrums period has to be managed very carefully, with attention paid to the individuals involved. In the case of *BG2*, our situation was complicated by the fact we had a shiny new project in the form of *Neverwinter Nights* (also with Black Isle/Interplay as publisher) in production just across the hall, and another cool new project in the form of our Star Wars RPG (with LucasArts as the publisher) also recently underway.

At BioWare we like to have monthly events for the entire company - like going to a movie or having a barbecue to give people a break by taking them out of the office. It's our way of pointing out we're still in this business for the joy of it - we're making games, and we should be having fun!

For the *Baldur's Gate II* team specifically we spent a lot of time talking to people throughout the project, especially at the mid-project low-point, in order to make sure we were providing enough support for the people who were slaving away on the game. We also shifted some people around - one of the major benefits of being a larger developer is that we have multiple projects arranged in a matrix organization, and at any given time there are likely a few people that wouldn't mind switching games. Also certain people are "starters" while others are "finishers" - it's important to understand each individual and tailor their tasks to their working style.

The lesson we learned was to beware the mid-project; morale can take a precipitous drop before it again climbs when people see the light at the end of the tunnel.

Start the Wrap-Up

In a project as content rich as *Baldur's Gate II*, we didn't really have to worry about cutting content. While we shipped with nearly all the features we originally planned, we did start cutting quests and characters well before the final testing phase. We still ended up with over 200 hundred hours of gameplay.

In retrospect we should have started this process many months earlier. One of the dangers of development is that game developers have a tendency to always add content if they are given time. They don't naturally spend time limiting and polishing content; instead, more time means more stuff. It's wise to use that prioritized feature list to hone the work (of course ours was informal, which made it a little difficult).

We learned to look at our target date and adjust our content development accordingly. In many ways, quality is more important than quantity. Even though *Baldur's Gate II* was bigger than *Baldur's Gate*, the actual content was much better quality - we just didn't realize how much more we had made in *BG2* until it was too late!

Test Test Test

Because of its immense size, *Baldur's Gate II* was a tester's nightmare - this was compounded by the fact that we didn't do enough testing as areas were being developed. *Baldur's Gate II* contains roughly 290 distinct quests - some of these are very small (20 minutes long) while others are quite large (a couple hours in length). Each quest needed to be tested both in single player and multiplayer modes.



The main lobby of the Hall of Wonders

During testing we adopted a very sound task and bug tracking method introduced to us by Feargus Urquhart, the Director of Black Isle Studios and Chris Parker and Doug Avery, our Black Isle producers (all of whom helped the project in many different ways). We put a number of white-boards in the halls of the testing and design area and listed all of the quests on the boards. We then put an 'X' next to each quest. We broke the designers and QA teams into paired subgroups - each pair (one tester and one designer) had the responsibility of thoroughly checking and fixing each quest. After they were certain the quest was bulletproof, its 'X' was removed. It took about 2 weeks to clear the board (on the first pass).

In addition to the subquest testing, we had another BioWare QA team (consisting not only of a couple people from QA, but also some junior programmers and some designers) work through the game in multiplayer mode. This was in addition to an Interplay Multiplayer QA team working onsite at BioWare and the nearly 30 QA people working down at Interplay. The experience with *Baldur's Gate II* reinforced the point that role-playing games really need significant QA commitment to be successful.

In the end we found and crushed over 15,000 bugs in *Baldur's Gate II*. Thanks to the hard work of everyone involved in QA of *Baldur's Gate II* we were able to ship a giant game with no significant bugs.

The Lesson: test early! You often don't have the time at the end to test adequately.

The Final Crunch

In the final days of working on *BG2* there was a strangely serene feeling in the office. We didn't experience the headlong panic that is sometimes prevalent while finishing a game, but we certainly did experience considerable stress as we built 21 final candidates in 3 days. After a few long nights with the whole team playing the game over and over again, we reached a point where we built a good final candidate. Then it was send to the duplicators!

We learned to save some energy during the final crunch - because you need it at the very end.

Post Support

The final area to mention regarding *Baldur's Gate II* is the topic of post game support. We've recently come to the realization that we believe in personally providing support to purchasers of our games. Though we had always provided this support, it was only after the conclusion of *BG2* that we formalized it as one of the goals of BioWare. A group of Line Producers, managed by their Producers, have recently been given the formal duty of providing very quick (same day or as close to it as possible) technical support to fans - their goal is to make sure people who purchase our games can play them.

We could rely on standard routes of customer support (and we do have this as well of course in the form of CS from our publishers Black Isle/Interplay and LucasArts) but we also want to make the time to directly interface with the purchasers of our games. After all, who better to work through technical issues than the people who made the game? For the first weekend after release we forgot to assign anyone the duty of watching the message boards and support (bginfo@bioware.com) email so the joint CEO's of BioWare did it themselves. It was very entertaining when people disputed whether it was actually us on the boards or answering their emails - they couldn't believe that we would bother to answer them in person (we did).

However, one of the most important things we've learned in our years in the industry is how important it is to support the fans that buy our games. This means first shipping a bug-free product, and second being completely available to help people that are having trouble with our games - on message boards, via contact emails, and anywhere else we can think of. What's the point of making games if you can't make sure people can play them?

Summary: What worked Well:

- Stable engine technology
- Team dedication to the project
- Veterans returning to improve on a system they created, ensuring familiarity with the development pipeline and engine
- Good project discipline
- Quality Assurance (QA) in the endgame

Summary: What could have been better:

- Fragmentation of team communication
- Content bloat (Game too big)
- Lack of early Quality Assurance
- Late asset delivery - audio and sound
- Poor coordination of localization (translation)
- Multi-player - non pausing dialog, non-protagonist characters

Conclusion

In conclusion we'd like to thank all of the people that worked on *Baldur's Gate II*, both on the development team at BioWare and at our publisher Black Isle Studios/Interplay. Like any big game *BG2* had its ups and downs, but in the end we are all very proud of the game we

made. We hope this retrospective provides you the reader some insight into our development methods and gives you some tangible ideas that you can apply to your own productions. In the end, it's all about the game - if you've put forth an honest effort you will always be satisfied with the result.

Author's note: Special thanks to the entire Baldur's Gate II team for working so damn hard, being so good to work with, and for creating a great sequel. Thanks as well to everyone else at BioWare and our publishers Black Isle/Interplay and LucasArts for their outstanding efforts on all of our other projects currently in development - Neverwinter Nights, Baldur's Gate II expansion pack, Star Wars RPG, and MDK2: Armageddon.

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

Copyright © 2016 UBM Tech, All rights reserved