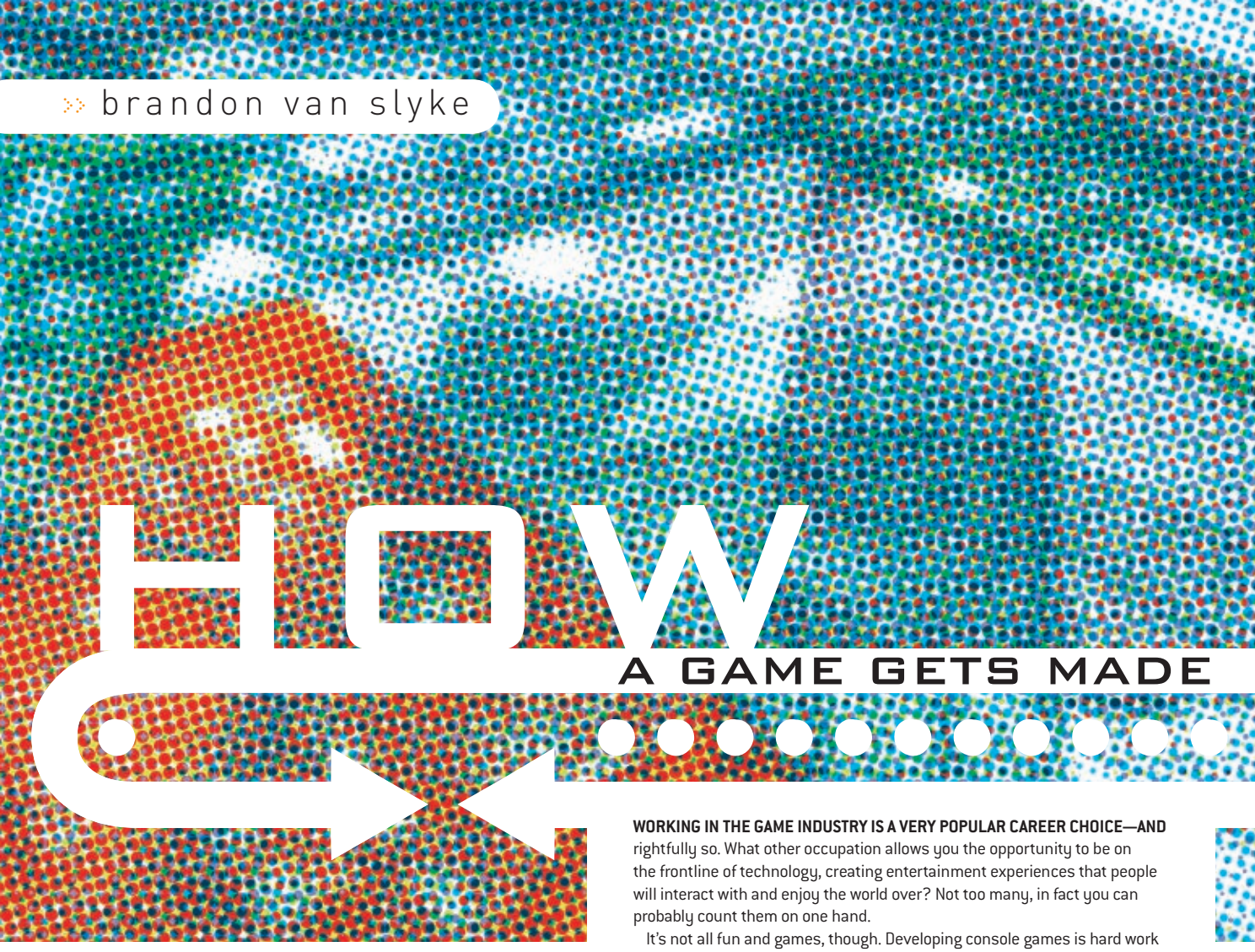
brandon van slyke



HOW A GAME GETS MADE

A game's journey from concept to store shelves

WORKING IN THE GAME INDUSTRY IS A VERY POPULAR CAREER CHOICE—AND rightfully so. What other occupation allows you the opportunity to be on the frontline of technology, creating entertainment experiences that people will interact with and enjoy the world over? Not too many, in fact you can probably count them on one hand.

It's not all fun and games, though. Developing console games is hard work and requires serious persistence and a ton of dedication. The process takes blood, sweat, and a metric ton of Mountain Dew, and is reliant on the skills and expertise of countless professionals whose own passion for games drive them to create fun and rewarding experiences.

How exactly does a game go from the cavernous depths of a game designer's imagination to the shiny cellophane covered package you find lining store shelves? If you try and imagine it like an assembly line with each station in the chain representing where a core piece of the development process gets integrated, game development is similar in that certain steps must be accomplished before work can begin on the next. From high concept to retail shelf, let's find out how a video game gets made.

PREPRODUCTION

Preproduction encompasses the planning stages of development and is the time when ideas are expanded upon, designs get fleshed out, prototypes are built, and decisions are made that will affect the project throughout development. It should come as no surprise that preproduction is far and away the most creative phase of a game's lifecycle. A lot of the work done in this initial period is thrown away, but it's necessary waste needed to determine what direction the game will ultimately take. It's during this time that the team really hunkers down and decides on the core elements of the game.

We often read about games that were rushed into production. It's a common occurrence in this industry, caused by a variety of factors, including

BRANDON VAN SLYKE *is a professional video game designer with experience designing commercial console games, mobile games, and casual downloadable web games. He is currently employed as a game designer at Vicarious Visions (an Activision studio) in New York. Email him at* **bvanslyke@gdmag.com.**

PHOTO BY JANE M. SAWYER

the need to release a game the same day as the movie it's based on. Another reason a game might be rushed into production is that the publisher has a limited amount of time before its rights to a certain license expire, or to hit a big holiday sales period. Despite these kinds of circumstances, it's generally understood that all games have some kind of preproduction phase, in one form or another.

During preproduction, the project's team size is very small and is primarily made up of each discipline's individual leads. The number of people working on a video game project changes during the course of the game's timeline, based on need. The kinds of things that happen during preproduction don't require a full team of programmers, artists, audio engineers, tools creators, and so forth. Successful game companies try to always have a number of projects going at different stages of development, which allows them to reshuffle their employees appropriately. As one project ends, the production phase of the next one is just starting.

> **"Preproduction: Finding the right context in which to frame your idea is a key part of moving forward with its implementation."**

For the first few months of a new project a game designer's time is spent creating flow charts to demonstrate in-game pacing, working on exhaustively detailed rule sets, fleshing out the game's narrative, and comprehensively documenting all the objects, characters, levels, enemies, and NPCs that need to be created for the game. With the majority of their time being spent working on documentation, designers must also find time to meet up with other members of the team who are busy with their own tasks.

Artists, for instance, spend most of their preproduction time fleshing out character designs, creating concept art for levels, and storyboarding scripted in-game sequences. Because they're ultimately in charge of the look and feel of the game, it's important that they also interact regularly with the game designers and writers to make sure that important design considerations are being factored into their work.

Programmers will often use their time during preproduction to create tools needed for development, while also documenting all the technical specifications for the project, which will cover both the tools they are creating as well as any issues and problems they foresee with implementation.

WHERE DOES THE IDEA COME FROM?

Let's take a step back and look at where an idea for a game comes from. The initial spark can originate from just about anywhere and can be influenced by pretty much anything: classic novels, a leisurely stroll through the park, or a favorite movie on the 37th viewing. More important than the actual idea, though, is how well it lends itself to interactivity. Finding the right context in which to frame your idea is a key part of moving forward with its implementation.

It's a common expression in the game industry that good ideas for video games are a dime a dozen, and in the case of today's larger console titles, popular franchises, recognizable intellectual property (IP) and big name licenses will more often than not win out due to previously established market awareness. This is why you tend to see far more sequels to successful titles and a lot of games based on hit movies and television shows; they simply provide lower risk to the publisher.

Where does this leave new ideas? Historically, original IP has come from one of two places: an independent studio that is able to self-fund the project or a publisher's most successful teams who have spent years proving themselves to their risk-adverse owners.

With that out of the way, the question then becomes, who decides what game gets made? In the case of an independent studio, that honor goes to the owner. If it's a mod group, the entire team may be a part of the decision. It's really based on the dynamic of the group.

On the other hand, when the studio is under the umbrella of a large publisher, the studio has the ability to put dibs on a popular IP or pitch its own unique idea. At this point, factoring in the state of the market and the track record of the studio, the publisher decides on whether an idea gets a green light.

Conceptualization. The core purpose of preproduction is to allow the development team to plan out every detail of the project and outline the production schedule based on the time estimates provided by each department. When the title is a sequel to an existing franchise, the length of preproduction is shortened. The emphasis then becomes figuring out what features will be added and evaluating existing data to see what is already available to work with. It's at this point that new artistic directions are experimented with while determining which path to take.

Planning it all out. When it comes to planning a massive project like a AAA console game, producers can be a very valuable resource to have. It's their job during the preproduction process to outline a production schedule based on time estimates provided by members of the development team. If the game is a sequel to an already existing franchise, the emphasis is on figuring out what additional features are going to be added or changed and whether existing data will be reused or if all assets will be created from scratch.

To properly plan, a producer starts by creating an in-depth schedule using a program like Microsoft Excel or Microsoft Project. It's here that they break down each individual task and identify the key points in the project, called milestones. Milestones are fixed dates in which the developer must deliver agreed-upon work to the approval of the publisher. There are actually a number of ways producers can go about scheduling a project. One of the more popular methods, and one you will find is being adopted by a lot of professional studios, is called agile development (see the sidebar "What is Agile Development").

Prototyping. Once initial planning has been completed, the team begins working on a prototype. A prototype is a rough mock-up of the game that can be played to see if the design mechanics work together when they're actually in motion. Most prototypes use placeholder art, require minimum assets, and are put together in very short periods of time. More often than not, the work done during this phase is tossed out when

actual production begins because it's important not to leave any artifacts in the code that may introduce problems later on down the line.

Because prototyping takes place in the preproduction process and is just a test bed, it is often done in tandem with the initial design phase to quickly try out new mechanics and ideas to see if they'll work in relation to the game as a whole. Once the prototype has been completed, it's often demoed to the publisher to gain their confidence and to get the actual production of the game green lit.

Once the prototype has been completed and the project has been green lit, the team is able to move on to the next stage of the development process: production. This is when things really begin to heat up, when teams grow, and communication between everyone involved becomes increasingly important.

WHAT IS AGILE DEVELOPMENT?

Agile development is one of the most popular development methodologies used in the game industry, if not *the* most popular. Agile development is characterized by modularity and a frequent review of the state of the project, hence giving the team "agility" to easily and quickly change direction if something isn't working. At the same time, it provides product owners with the information they need and the ability to see progress as it's being made.

How it works is that the individual tasks needed to complete any given feature are broken down and assigned to different members of the team who then estimate the amount of time it will take to complete them. Throughout the creation of the game, the team works in "sprints," which usually last anywhere from two to four weeks. During sprints, producers are able to gauge the amount of time a certain feature is taking to implement and can cut or reduce scope as needed.

Agile in and of itself is a fairly complex system and entire books and articles have been written about the benefits it provides. It's far too large a topic to discuss in this article. If you're interested I highly recommend doing a bit of research on it as you may be introduced to it when you get your first job in the industry.

PRODUCTION

When an idea for a game has finally gotten the green light, it's time to begin developing it into a full release title. But where do you begin?

Just as construction companies don't start erecting a building by randomly pounding nails into a board, game developers don't begin working on a game without a solid foundation or plan. All those prior months of preproduction now show their worth as the team begins to build the game to spec based on the design document.

That being said, when making video games, no matter how preplanned and thought-out the preproduction phase seemed to be, things can and will change. Making a game is a largely iterative process with new ideas often implemented on the fly. It's obviously not a free for all, but most studios encourage some form of structured experimentation as long as it can be demonstrated to improve the game.

Production methodologies. One of the most important aspects of production is keeping track of everything. Being able to maintain visibility among not only the development team, but also the production staff helps keep everyone on the same page and makes it easier to quickly rectify any issues as they crop up. However, issues and dependencies can be difficult to track within a large team. This is where production methodologies come into play. A production methodology is simply the process a development team uses to keep track of, and divide up all the work that needs to be done.

Building the game. Up to this point we've discussed where the idea comes from and how planning and prototyping the concept are necessary steps to getting the game off the ground. Now it's time to take a look at how the game is built.

To keep production rolling smoothly and to minimize potential blockers, dependencies must be identified early on. Recognizing which tasks need to be completed before work can commence on another allows developers to properly stagger the workload and production pipeline accordingly. Identifying dependencies also gives the design, art, and engineering teams an opportunity to work independently of one another, effectively maximizing everyone's efficiency.

To kick things off artists begin by creating and texturing character models that the player will interact with as well as the unique props that populate the game world. These objects include items such as cars, buildings, and oh-so-popular wooden crates. When a character model is complete, it's then passed off to animators, who rig the skeletons and generate all the different animations players see in-game. Because art and asset creation is the most time-consuming part of the process, artists usually must get a head start on all of this while the design and engineering teams lay out the core functionality and data that will drive everything behind the scenes.

At this point, level designers are busy creating rough passes of each level in the game. This includes tasks like determining spawn points for enemies, deciding where scripted events will occur, plotting out AI pathing nodes, and demonstrating cool ways that geometry can be incorporated into gameplay. Once they have all of that fleshed out, the level designers begin populating their grey-boxed levels with the custom props that have been recently created by the art team.

On the data and engineering side of things, the team is busy putting everything together. It's here that system designers start hooking up all the animation and FX data so that characters and entities show up in-game. If need be, the designers will sit down with the animators to tweak animation timing to ensure that it matches the specifications outlined during preproduction. It's also a systems designer's responsibility to tune gameplay and balance the values that define everything from the number of hit points the player's avatar starts off with to the amount of damage a point blank area-of-effect radial attack delivers.

Over in the land of audio, sound designers are kept busy sampling sound effects for different event triggers in-game and acquiring and directing any voiceover talent that is needed for character dialogue or narration. As might be expected, they're also in charge of composing the game's musical tracks and score. Audio is often an overlooked discipline that should not be taken for granted. Their contribution to the game is very important and their work will not only bring the game to life and help set the mood, but can also be used to aid the design.

Don't break the build! If you have a team of 70 people all working on the same data, how is it that nothing gets deleted, overwritten, and ultimately destroyed? I'll admit it can be a fairly chaotic process. However, there is a solution.

Development teams utilize version-control software to help ensure that any changes made to the current build, like a variable in code or a value in data, doesn't break what's already

there. Version-control software is great because it allows developers to overwrite, redo, and add changes as they see fit. If there is a conflict, it can be resolved by interactively merging each team member's version of the file. The whole system provides developers with constant updates and an impressive amount of visibility, effectively guaranteeing that no work is accidentally lost.

From part to whole. At various times throughout the project, the team should get together to review and critique the most recent build. It's at these points that the people working in the trenches can take a break and assess the game as a whole, allowing them to see what their teammates have been up to the past few weeks.

Not only does everyone get to have their work constructively critiqued, but they can also point out and discuss their contributions and progress, which might not have any visible effects on the game. For example, if someone made a new effects-rendering system or wrote new documentation outlining a narrative for a cut scene, their contribution might not be readily apparent—but it is important, and these meetings allow everyone on the team the chance to acknowledge those kinds of contributions. More than anything, though, these meetings are meant to maintain open communication among the team and enlighten everyone on the current status of the project.

When does testing start? Quality assurance (QA) testers are usually integrated toward the middle of production, although leads are on the project from the beginning. During the beginning of production, they're often busy getting the bug tracking software set up and testing each build to make sure all known issues are recorded and assigned to the proper person on the development team.

At about two-thirds of the way through production, additional staff will be brought on to fully test the game. It's at this point that the team should have the majority of features implemented. Now it's time to refine everything and squash any and all bugs that come your way.

> **"Prototyping: The work done during this phase is tossed out when actual production begins because it's important to not leave any artifacts in the code which may introduce problems later on down the line."**

Home stretch. Once the game reaches alpha (a near complete phase), things will seem to be shaping up nicely. The staff will report bugs at a record pace, and the developers will be resolving them just as fast. The only item left is to market the game and get it on store shelves!

POSTPRODUCTION

Congratulations, the game is complete. That must mean all the work is done, right? Well it is—for the development team. It's now time to find out who steps in and what happens after the game is finally made and on its way to store shelves.

PR and marketing. While the developers have been toiling away designing boss battles and drawing up finite state machines, the marketing staff on the publisher's end has been busy figuring out how they will position the title once it's complete.

Marketing sometimes has significant influence on how well a game is understood by both critics and fans, and their work is often the first thing to make an impression on a potential customer. For instance, they coordinate exclusive reveals with industry magazines and popular video game web sites, as well as manage the deployment of television ads and tie-ins with other consumer products.

Publishing and distribution. Along with funding the development of the game, the publisher also handles regional localization, creation of the game's manual, and orchestrates manufacturing of the final packaging. When the final build of the game is completed (that version is called the gold master disc) and sent off to be printed, the publisher has usually lined up a distributor to handle getting the game onto store shelves.

Once the game has been printed and orders have been made, the game is ready to be shipped to your local retailer, where it's hopefully enjoyed by everyone who's decided to check it out.

WAIT, THERE'S MORE?

Even when a new game is finally released, it sometimes isn't finished. With the rising number of high-speed internet enabled consoles and the ever-growing adoption of digital distribution, developers are now regularly being tasked with creating additional downloadable content to help extend the life of their titles. Downloadable content can include anything from additional characters and costumes, to new level maps, and even interviews with the developers. A lot of the people who are hired during the course of the project are likely to be retained to create these additional features.

If the work gets outsourced, it's important that the studio developing the downloadable content has the ability to communicate with members of the original development team. This is especially true as issues arise and they learn to work within the constraints of the data. A wiki is often created as a way of chronicling important need-to-know information and is usually made accessible in situations like these.

Earlier I mentioned the fact that team sizes fluctuate—what happens if the company doesn't need you any more when it's working on the downloadable content? What happens when your role on the project is finally complete? Throughout the project, and especially near the end, tasks and roles come to an end and it's time to start work on the next one. It's at this time that managers and team leads reshuffle staff and place them either on the studio's next game, move them laterally to another project they have in development, or in more unfortunate cases lay them off.

And there you have it! That's more or less how a modern day console game gets made. Now that you understand a little more about the process of bringing a game from concept to completion, it's important to reiterate that teamwork, communication, and proper planning are essential to creating a game in today's competitive environment. Now what are you waiting for? Get out there and make some games!