

PHASER WORLD

JANUARY 2018

ISSUE
110



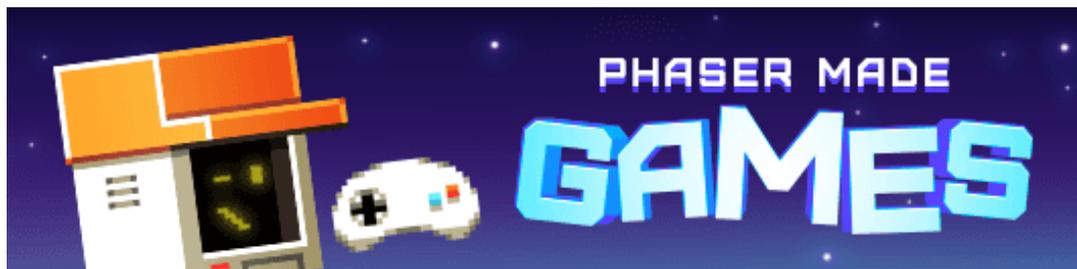
Welcome to Issue 110 of Phaser World

Happy 2018! It seems crazy to think we're in January already and it's even crazier when I look at the schedule and plans for what is coming up this year. It's

going to be a massive year for Phaser and I'm so glad to be back at the coalface, plugging away and getting prepared for some intense work in the coming months.

You lot have been busy too! From GitHub Game jam entries to apps helping you learn Katakana it's been an active few weeks. Awesome to see Phaser used to build the new Minecraft Hour of Code activity too - over 70 million sessions completed so far! Wowzers.

So, until the next issue, keep on coding. Drop me a line if you've got any news you'd like featured by simply replying to this email, messaging me on [slack](#) or sending a tweet to [@phaser_](#).



The Latest Games



Game of the Week

[Minecraft: Hero's Journey](#)

Minecraft is back for the Hour of Code with a brand new activity! Journey through Minecraft with code.



Staff Pick

[Kick Punch](#)

A gloriously 8-bit Gameboy styled beat-em-up. Can you save your wife and dog from the thugs on the street?



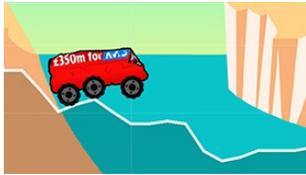
Santa Rush

Control Santa as he rushes around the levels, collecting presents in this single-button action game.



Kana Bento

A drag and drop game for learning Hiragana and Katakana.



Brexit Bus

Take back control of the infamous campaign bus and ride the value of the pound since the EU referendum.



What's New?



Phaser in 2017 and beyond

A look back at the news and numbers of Phaser in 2017 and what the New Year holds.



Ultimate Game Parts Template

All the essentials you could need for creating games in a single template. From message boxes and health bars to screens and MVC controllers.



Phaser CE v2.9.4 Released

The new Phaser CE release includes fixes to the Tween Manager, new method arguments and bug fixes.



MyGameBuilder Hour of Code

Play, fork, build, learn and share your own games with this interactive collaborative game builder.



Warp Shift Tutorial

Recreate this unique puzzle game in Phaser. In part one the first 3 levels are added.



CodeCourse Phaser Tutorial

A new 14 part video tutorial series covering creating a Breakout game using Phaser CE and ES6 from start to finish.



Thank you to our awesome new [Phaser Patrons](#) **Sebastian Bergman**, **Rajasekaran Senthil Kumaran** (who came back to us after Patreon announced their horrendous \$1 pledge tax, then rescinded on it!) and to **Ryan Kubik**, **Nils Nordmark**, **Sergej Suslov**, **Paul McGarry (Rubble Games)** and **tolias**. Thank you also to **Casey** for increasing their pledge and to **Ben Gillbanks** for his donation.

Patreon is a way to contribute towards the Phaser project on a monthly basis. This money is used *entirely* to fund development costs and is the only reason we're able to invest so much time into our work. You can also [donate](#) via PayPal.

Backers get forum badges, discounts on plugins and books, and are entitled to free monthly coding support via Slack or Skype.

Game Asset of the Week



This week we're featuring the [Platformer Game Tileset 9](#). It contains ground tiles, items, objects and more for building a nice looking platformer.



Dev Log #110

Because the whole team was off last week there aren't a great deal of new things to write about. So instead I thought I'd take a moment to share our plans for what work we're taking on next and some related dates.

I also published the [2017 Phaser retrospective](#), which you may wish to read. It covers things like Phaser plugin sales, site traffic, games of the year and so on for 2017. Also, see how many of my 2016 predictions I got right!

Phaser 3 Next Steps

January is release month. This is frightening given the number of tasks left, but

exhilarating because it means we'll finally be at 3.0. I think it's fair to say that I consider 3.0 to be the beginning of the journey, not the end. While there are lots of things that need to be in place for the release (docs and TypeScript defs being the main ones), we're actually only 1 system away from API completion and that should be in place by the end of this week.

This system is the Scale Manager and, honestly, I'm glad it has been left until the end due to recent changes in the browser space relating to this that could never have been foreseen. The brand new Resize Observer API has landed in Chrome and offers some brilliant new ways to handle resize events. My thinking is that the Scale Manager in 3.0 needs to be considerably less intrusive than in V2. I'll be removing a lot of the 'features' that complicated things and taking it right back to basics, with a few key events, some solid browser monitoring and letting you deal with resize events as you see fit.

Felipe will be working on tidying up the renderers this week. There's a lot of duplicate code, shaders that could be moved to webpack included files, batch functions to sort out and so on. It's sadly quite dull work but essential.

Once both are done it leaves us with 3 weeks to finish the docs, create new examples and complete any other internal restructuring we wish to do. There's a whole stack of features I'd like to include but they can wait until 3.0 is released and follow in future updates. It feels tight but it doesn't feel impossible.

Beyond 3.0

It's safe to say that once 3.0 is released our primary focus will be bug fixing. I've allowed for a period of two months to concentrate specifically on addressing issues as they arise. This will include both helping people out in the forum and on slack, creating new examples and fixing issues as they're reported.

Bug fixing is an on-going process anyway, it's not like we'll stop doing it after those two months are up. But it will be important not to start the creation of any major new systems during this time to allow us to keep our focus in the right place.

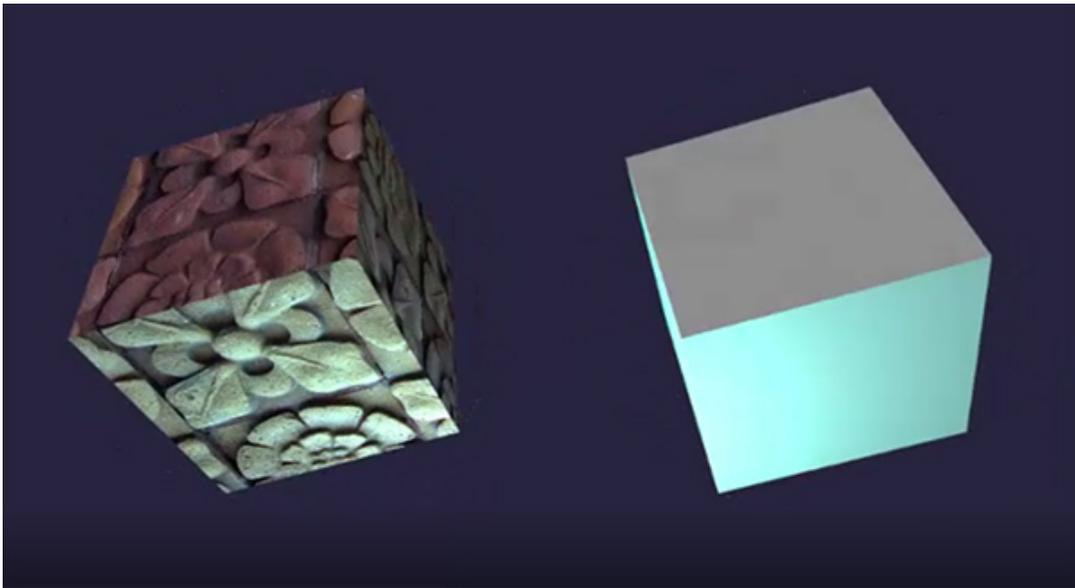
ES6

I fully plan to move the codebase over to ES6 once things have settled down. Due to the way we've built v3 this will not take very long but it was just too much of an undertaking to consider doing prior to release. Making the change should have little to no impact on the public-facing API. By doing it early we should be able to get it moved to ES6 before anyone starts building major plugins for v3

based on the 'old' style.

Skeletal Animation

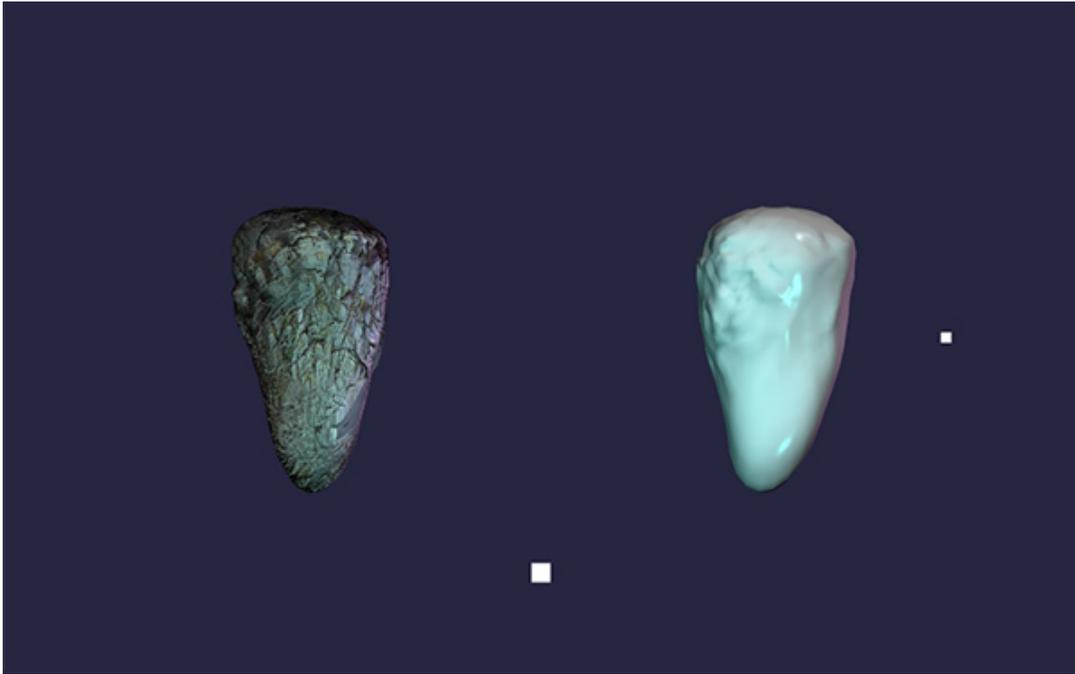
We will be taking a good look at how to make it easy for you to get animations from the likes of DragonBones, Spine, and Spriter into V3. It will require some custom updates to our renderers but it's something we've been considering for a while now, so none of it is unexpected. Personally, I find it frustrating there is no unified 'bone' format. They all do things differently and require bespoke handlers to deal with their idiosyncrasies. Such is life.



Graphics 3D

Those of you who follow the new Phaser twitter account will have seen some sneak peaks of this already. But we've been working on a Graphics 3D component that will let you easily create 3D scenes within your game.

So far we've got 3D primitives in, obj model loading, a camera system, lighting, textures and normal maps all working. We've built this outside of V3 for the time being, as we didn't want to pollute the codebase with it knowing that it wouldn't be part of the 3.0 release.



The aim is to include it as a Graphics 3D game object which you can add to your scene like any other object. It will work much in the same way as the Graphics 2D object does now. You'll interface with methods directly on the object that allows you to create primitives, move the camera, add lights, etc. And it'll all render into its own layer in the scene.

We have quite fixed objectives for this component. You should absolutely be able to use it to create a 3D game or add a 3D element to your 2D games. The problem with 3D is it's a massively deep rabbit hole. Once you start going it never seems to end and before you know it there is megabytes worth of library functions included that barely anyone will need. By keeping the features limited we can ensure it remains useful when it comes to making games, which after all is what Phaser is all about.

Beta 15

During some downtime over the holidays, I went through a number of issues on GitHub and created a Beta 15 milestone. I'm now working towards this and we'll have it released in time for the next issue. Keep an eye on the repo for updates! and if you find any issues in the meantime, please do report them, as they're all being acted upon.

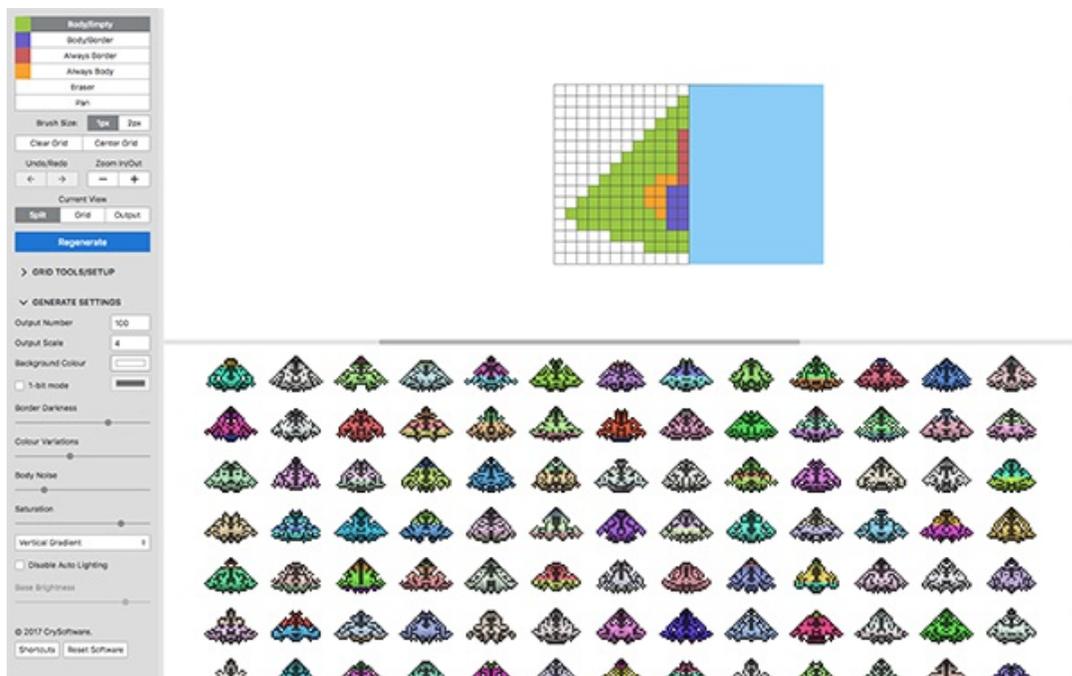
It's going to be a crazy few weeks coming up, make no mistake about it. I foresee lots of late night coding sessions and jsdocs bleeding out of my eyeballs - but it has to happen, we're closer than ever and it feels fantastic!

Phaser 3 Labs

[Phaser 3 Beta 14](#) is out and ready for testing.

Visit the [Phaser 3 Labs](#) to view the API structure in depth, read the FAQ, previous Developer Logs and contribution guides.

You can also join the [Phaser 3 Google Group](#) or post to the [Phaser 3 Forum](#) - we'd love to hear from you!



[CryPixels](#) is a downloadable app for Windows and macOS that is a procedural pixel art generator. It works using a generation grid and from this spawns hundreds of variants. Good fun and potentially useful for your own games!

[NectarJS](#) is a new project that aims to compile JavaScript into native binaries for every platform.

[Tenpence Arcade](#) is a fun podcast about all things retro gaming related. Well worth a listen!

Phaser Releases

Phaser CE [2.9.4](#) released December 20th 2017.

Phaser 3 [Beta 14](#) released December 21st 2017.

Please help [support](#) Phaser development

Have some news you'd like published? Email support@phaser.io or [tweet us](#).

Missed an issue? Check out the [Back Issues](#) page.



©2018 Photon Storm Ltd | Unit 4 Old Fleece Chambers, Lydney, GL15 5RA, UK

[Web Version](#)

[Preferences](#)

[Forward](#)

[Unsubscribe](#)

Powered by [Mad Mimi](#)®

A GoDaddy® company