Welcome to Issue 148 of Phaser World

THIS WEEK...

PHASER 3.19.0 RELEASED
BASKETBALL STARS
PHASER 4 ANNOUNCEMENT
MAGIKMON

Welcome to Issue 148 of Phaser World
After a short break, Phaser World is back with a massive double-sized issue. There’s a lot to catch up on, including the public announcement of Phaser 4, the Phaser 3.19 release, stacks of tutorials and awesome new games.

As usual, be sure to read this newsletter on the web so you don’t miss anything - there’s a lot packed in here.

Got a game or tutorial you’d like featured? Simply reply to this email, or message me on Slack, Discord or Twitter. Until the next issue, keep on coding!

The Latest Games

Game of the Week
Basketball Stars
Show off your basketball skills and see if you can score a three-pointer or slam home an impressive dunk! You can play as many different legendary teams and try out both your defensive and offensive skills.

Staff Pick
Pool Club
Pool Club is a smooth solo player pool game. Join our club and let time fly while sinking some balls.
**Chess Grandmaster**
A wide range of checkmate puzzles across two different game modes ensure hours of mental activity for chess fans!

**Glowing Abyss**
Beware of the depths as you swim down into the darkness in order to try and get your eggs back.

**Magikmon**
A combat based RPG spread over a huge map. Imagine Harry Potter mixed with Pokemon!

**Patchwork Harmony**
A charming puzzle platform game in which you have to get the toy bunny to the little girl for her birthday.
What's New?

Phaser 3.19.0 Released
The latest release of Phaser is out! Including the complete Spine plugin, a brand new Tween system, Shader Buffers, Render Texture snapshots and loads, loads more.

GameShell - A hackable portable game console
A portable, modular console you build yourself, capable of running Phaser games directly.

Creating a Phaser 3 Template Part 2
In part two the template is expanded further, with pre-built title, options and credits scenes.
Converting a Flash game to HTML5
A blog post detailing the process of converting a legacy Flash game to HTML5 and the tools available.

Creating a Multiplayer Phaser 3 Game Part 2
In part 2 of the series learn how to add Socket.io and server logic for adding and removing players.

Creating a Multiplayer Phaser 3 Game Part 3
In part 3 learn how to add client side logic for handling player input and collectibles.

Using Gravity in Phaser 3 Tutorial
A mini tutorial all about using gravity and sound effects in Arcade Physics.

Getting Started with Phaser Part 4
Part 4 of the animated video series on getting started with Phaser 3, this time all about moving images.
Getting Started with Phaser Part 5
Part 5 of the animated video series on getting started with Phaser 3, this time sprite sheets and clickable sprites are introduced.

String Avoider Tutorial
Drag the string to the goal as you learn how to create this update of a classic Flash game.

Parallax Scrolling with TileSprites Tutorial
How to create a parallax scrolling effect with Tile Sprites, including pixel art assets to download.

Endless Runner Tutorial Part 4
In the next part of this series Emanuele adds in a scrolling parallax background.
Adding deadly fire to the game and making it more fun to play by reducing the players hitbox.

Making things Jump in Phaser 3 Tutorial
Carrying on from the Gravity tutorial it's time to make things jump!

SameGame Library and Phaser 3 Example
A JavaScript class to handle same-game logic and a Phaser 3 implementation example.

Making a Video Game using Phaser
How a beginner developer took on the challenge of creating his own game from scratch in Phaser and what he learned on the way.

Phaser 3 Game Development Course
A complete Phaser 3 and JavaScript Game Development package. 9 courses, 119 lessons and over 15 hours of video content. Learn to code and create a huge portfolio of cross platform games.
Help support Phaser

Because Phaser is an open source project, we cannot charge for it in the same way as traditional retail software. What’s more, we don’t ever want to. After all, it’s built on, and was born from, open web standards. The core framework will always be free, even if you use it commercially.

You may not realize it, but because of this, we rely 100% on community backing to fund development.

Your support helps secure a constant cycle of updates, fixes, new features and planning for the future. There are other benefits to backing Phaser, too:

I use Patreon to manage the backing and you can support Phaser from $1 per month. The amount you pledge is entirely up to you and can be changed as often as you like.

Please help support Phaser on Patreon
Thank you to these awesome Patrons who recently joined and make continued development of Phaser possible:

Ramon Tavares  
Patrick Griffiths  
Oliver Emberton  
Cameron H  
Oliver Ritter  
Eirik Korsnes  
Joep Bos-Coenraad  
Glenn De Backer  
Gary  
David Brind  
Paolo Ruffini  
Timothy Janssen  
Barry Stump  
Sagar Dabas  
Louis Stadelmann  
Josh Engebretson  
Timothy OBrien  
Gérald Travaillot  
Rain Miao  
Pablo Roman  
Pascal Peuckert  
Andrew Brandt  
Louigi Verona  
Alix Lepinay  
Solar  
Zachery Hysong  
Kheftel  
Aleksandar Micovic  
Cerebral Fix  
Carol Beck

As I’m a bit behind with the newsletters there are lots of new Patrons to thank
who are not on this list, please keep an eye out for your name soon!

Thank you to our new sponsor: Poki.com They've been a supporter of Phaser for a few years, but recently bumped up to being a full-on super-sponsor!

"Poki Games offers you all of the best online games and most popular categories, like .io games, two-player games, and papa's games, stickman games, girl games, as well as games based on your favorite movies or TV shows. In our large collection of fun games, you can also find classic titles, including Mahjong, Bubble Shooter, and Bejeweled, as well as cool games, such as Slither.io, Color Switch, and Happy Wheels. You can play games in 3D, super-fun shooting games, and all of the popular puzzle games! No matter your taste in games, we have great ones for you."

If you're a Phaser developer with high-quality games to license, then get in touch. Details can be found on their web site.

Dev Log #148

Welcome to Dev Log 148. It's been a good while since the last Dev Log and a lot has happened. Phaser 3.18 and, most recently, Phaser 3.19 have been released, so if you're still on an older release now may be the time to upgrade. You'll find
full details about both releases in this Dev Log. I also made an announcement on the Phaser Patreon about how the development of Phaser 4 will begin shortly. Yes, you read that correctly. Read the article about what is planned for Phaser 4, then feel free to chat about it in the Phaser Discord.

Starting with Dev Log 149 I will return to the more usual format of detailing what coding has been taking place and mixing that in with mini-tutorials. For this issue, however, there's a lot of ground to cover. If you follow Phaser closely then you will know about most of this already, but I'm well aware that lots of you like to keep abreast of Phaser just via the newsletters - so this one is especially for you. Please, dig in and see what's been going on these past few months!

**Phaser 4 Announcement**

Hi everyone - I want to share with you some exciting news that paves the way for the future of Phaser in 2019 and beyond. It’s with great pleasure I can announce that thanks to the continued support from the Facebook Gaming and the Instant Games teams, I’m able to start the development work required to build the next iteration of Phaser. This post will outline what’s planned and hopefully answer many of the questions you may have.

Phaser has been working together with Facebook Gaming for some time now. We initially started work together back when we jointly released the Instant Games plugin and associated tutorials for Phaser 3. The working relationship has
been great and has allowed us to come up with the next evolution of Phaser together.

The continued support from Facebook directly allows for something I’ve been wanting to do with Phaser for a while now: Rewrite it in TypeScript.

Those of you who have been around Phaser for long enough will remember it was originally written in TypeScript, way back before TypeScript was even at a 1.0 release. We moved away from it due to an endless stream of bugs with the TypeScript compiler and tools. Quite frankly, we had adopted it far too early and got burnt badly in the process. Fast forward 5 years and the world is a very different place. TypeScript has matured beyond all recognition and the tooling around it now is best-in-class. What’s more, lots of you are now using it for your game development projects, too. Far more than ever before if our TypeScript defs stats are anything to go by.

Let me clear up a few important points now:

1) This is not an API rewrite. Phaser 3 is comprised of hundreds of small modules already, so the work will be focused on porting those to TypeScript. Yes, I will absolutely use this process to improve the API as I go. There’s no way I will make any claims about “the API not changing” because I know it will. The core work, however, isn’t to rewrite or restructure, it’s to port.

2) You will not need to use TypeScript in order to use Phaser. Phaser releases will still be published as ES5 bundles. If you code in ES5 JavaScript nothing about your workflow need change.

3) The TypeScript version of Phaser will be Phaser 4.0. I really didn’t want to have to bump the number like this, because for Phaser the major number is part of the name, not the release version, but I feel it’s an important enough update that I have no choice but to mark it in this way. It will also mark a clean point of separation for users.

4) Even with this continued support from Facebook, I still absolutely need your support via Patreon. All of my projections are based on a combination of resources and support from one source does not cancel out the others. It’s still vitally important to the project that Patreon continues.

Why TypeScript and not ES6?

Although I’ve been discussing this work with Facebook for months now, I hadn’t made a call on if the port should be to ES6 or TypeScript. Facebook had
indicated they were open to either option and personally I had no strong leaning in either direction either. A few things tipped the scales in TypeScript’s favor, though. The first is that I’ve sunk a lot of resources into our current TypeScript definitions generator. While it works, it’s less than ideal and we are forever tweaking the output to make it more accurate for TypeScript developers. This is a problem beset by all projects that use tools to automate the creation of TypeScript defs and I see issues around it across many, many projects outside of Phaser. As more and more developers use TypeScript as their primary language, it will give them a perfect developer experience, while taking nothing away from JavaScript developers.

Recently a few bugs have slipped into the API that would have been solved by using strict types, which TypeScript will bring. Although most bugs are, and always will be, logic rather than type related - catching even just one type bug before it hits production is a win in my books.

Finally, this move will significantly help those new to the API. If they choose to use TypeScript they’ll get full and 100% accurate code insight and context awareness. This is a huge benefit and shouldn’t be under-estimated. It will help me, too, as I work on the API, speeding up my development workflow as much as anyone else’s. I spend hours and hours inside of VS Code each day and I’ll finally be able to benefit from the massive number of tools it contains. Time saved is time gained.

I’m expecting the development of Phaser 4 to begin by the end of August 2019. 3.x issues will still be addressed, and new versions published. As I near completion of 4.0.0 this process will halt. Until then, nothing will change.

I know better than to give any kind of estimation for when Phaser 4 will be published. And honestly, I don’t actually know. I’ve made estimates, of course, but I’m not committing to a date, no matter how vague. As with all versions previously, you will be able to track progress openly via GitHub and my Dev Logs.

I’m really looking forward to this next stage. Phaser 3 had quite a difficult birth, although that was due to scope and expectations, not technology. This work allows me to pull it kicking and screaming into the modern web era and tighten-up ship as we go. It’s not about reinventing for the 4th time, it’s simply about supercharging and streamlining what we already have. And that is genuinely exciting.
Phaser 3.18 Released

After another month of hard work, we're very happy to announce the release of Phaser 3.18. In this release, we took the time to rewrite large chunks of the Input API. This allowed us to fix a number of issues that had arisen, as well as optimizing the internal event flow. Native support for complete Multi-Touch support, Mouse Wheels, and more advanced Pointer Button handling are now available. The whole API is smaller and tidier after the clean-up, which is always a good thing.

We've also added lots of other features and updates, including better Matter physics constraint handling, Arcade Physics improvements, Audio decoding events, Text justification, new Bounds methods and a lot, lot more. As usual, it doesn't end there, though. You'll find hundreds of great new features, updates, and fixes.
Thanks to some hard work updating the Phaser website, you can now run all of the Phaser 3 examples directly from it! Such as this fun multi-touch piano example :)
games, offering a dedicated 2D skeletal animation system and workflow. Our plugin makes integration with Phaser seamless and is fully updated for the Spine 3.7 Runtimes with support for WebGL and Canvas renderers.

This version will properly batch Spine skeletons where possible, potentially saving hundreds of draw calls. The plugin is fully documented and exports both debug and minified files, suitable for ES6 ‘importing’ or ES5 plugin inclusion. The whole plugin is just 68KB in size (min+gz), or a paltry 57KB if you only need the Canvas renderer! That's a really tiny payload for such a massive feature-set. You can find out more about Spine from the Esoteric Software website.

3.19 also introduces a huge overhaul to the Tween system. Tweens now have 100% documentation and we've extended their capabilities significantly. There are lots of new Tween Events to listen for, such as 'COMPLETE' or 'REPEAT' to allow you to trigger actions without needing to create callbacks. Tweens can now tween both 'from' and 'to' a value, with the ability to set a starting value for any tweened property. There are lots of new handy methods and properties, such as `Tween.hasStarted` and a rewrite of the Tween seeking function, so it now allows you to seek to any point in time across a tween.
Finally, we've added in the great new 'StaggerBuilder'. This allows you to easily add staggered offsets to a bunch of tween targets, including all kinds of options such as staggering across a grid layout, stagger directions, starting values and a lot more. Please see the docs and examples for more details.

As usual, it doesn't end there, though :) Another very useful feature is `Shader.setRenderToTexture`. This allows you to redirect a shader to its own framebuffer / WebGL Texture instead of to the display list. This allows you to use the output of the shader as an input for another shader, by mapping a sampler2D uniform to it. It also allows you to save the Shader to the Texture Manager, allowing you to use it as a texture for any other texture based Game Object such as a Sprite. Combined with the new `setSampler2DBuffer` method you can now easily chain shaders together, using them as buffers for other shaders.
One thing I've been seeing asked for a lot on the Phaser Discord is the ability to 'save' a Render Texture to an image. So, I added the new methods `RenderTexture.snapshot` and `snapshotArea`. This allows you to grab whatever is on the Render Texture at that point in time and turn it into an Image. You could then save this image to the Texture Manager, if needed, or just save it out to the filesystem, or transmit it to as web service. Great for things like avatar creators or art packages.

Download Phaser 3.19.0 from the GitHub releases page, or pull it in via the npm package. There are also the pre-built files available on the jsDeliver CDN.

The brand new 3.19 documentation has been updated and published. As before, you can read it online, or check out the repo to have a local copy.

The TypeScript definitions have been improved further still and are now available in the main Phaser repo. So, there's no need to clone another repo when updating Phaser any longer.

As always, please check out the Change Log for comprehensive details about what recent versions contain.
The August Code Bundle is now out for everyone who supports Phaser on Patreon or via PayPal. This month the focus is mostly on Spine, the bone-based IK system from Esoteric Software. I've been working on the Spine plugin for Phaser 3 for a few weeks now and, truth be told, it turned out to be far more complex than expected. Even so, I'm very happy with the end results and you'll
find the latest version of the plugin, along with exclusive examples, in this months bundle, which also contains:

- Sprite Sheet and Atlas Frame Explorers.

I had a need to visually be able to select a frame from an atlas and find out what it's frame name was. Rather than pour through the JSON file, I built a little tool to do
it
and
it
turned
out
to
be
really
quite
handy.
So,
then
I
built
one
for
inspecting
Sprite
Sheets
too
:)

- Spine Examples.
  To
go
with
the
latest
version
of
the
plugin
here
are
9
different
examples
for
you.
They
start
with
the
basics,
such
as
changing animation, skins and slot attachments, and then progress into more complex territory such as bone movement and aiming, vertex effects and color controls. You'll also find an example called 'Commando' where you can run around shooting (trying running while shooting up too!). This is a
handy example of combining Spine with Arcade Physics and keyboard controls - and honestly, it feels ripe for evolving into a full game.

- Phaser 3.19 includes lots of enhancements to the Tween system, including the ability to create powerful new Stagger effects. You'll find two examples
showcasing this. Personally, I really love just clicking through the 'grid stagger' example and seeing what effects it comes up with :)

- I've also included an example showing one method for dragging a sprite along a path. You can tweak the values to achieve different results
but hopefully, it will help some out.

- Last, but by no means least, is Snowmen Attack! Because it's the middle of one of the hottest summers on record here, so what better than a snowball fight? :) Based on the arcade classic Tapper, move the penguin
up and down, throwing snowballs to knock-back the attacking snowmen before they reach, or shoot, your eggs. This is another complete game, full of animation, audio, and special effects. As before, hopefully, all of you enjoy playing it, but perhaps the more beginner level of
you will benefit from the way in which I structure the project and handle scene progression.

If you support Phaser today, you can download the pack instantly, along with all previously released packs too.

A stunning procedurally animated, raymarched SDF creature, created entirely in ShaderToy.
Resonance Arcade is a multi-platform gaming podcast hosted by 3 gamers from the north of England.

The cost of JavaScript in 2019 - a fascinating and detailed blog post from the V8 Internals team about performance tuning your web pages (it's worth adding that when it comes to games, the same rules don't always apply)

---

Phaser Releases


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.