PHASER WORLD

AUGUST 2018

ISSUE **126**

MAFIA Billiard tricks

THIS WEEK...

MAFIA BILLIARD TRICKS

9-SLICE PLUGIN

HYPERDIVER

GOTHICVANIA CEMETRY PACK

Welcome to Issue 126 of Phaser World

Greetings! I'm back from a wonderful and much needed holiday with another jam-packed issue of Phaser World. Play billiards against the mafia, learn about creating dynamic tilemap layers and catch up with Phaser 3 news.

The summer period is always a little slower when it comes to releases and news as you all take much needed breaks away from your keyboards. As we head in to Autumn this is genuinely my favorite time of the year. It's as if everyone is infected with a new kind of vigor and as the nights draw in, the coding intensifies. We're going to see some awesome things in the coming months :)

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, Discord or Twitter.

Emanuele Feronato is easily the most prolific author of Phaser tutorials on the planet. He has published over 250 of them in the past few years and started writing Phaser 3 tutorials before most others. As the framework has evolved, his tutorials have kept pace. So it's great to see that he's been busy writing a book and it's now available.

It's 155 pages long, with 28 source code examples, taking you through the process of creating a full cross platform game. What's more, buying a copy actively contributes towards funding my work on Phaser too!

HTML5 Cross Platform Game Development Using Phaser 3

Learn to build HTML5 games using Phaser 3 and other free software

Emanuele Feronato

The Latest Games

Game of the Week

Mafia Billiard Tricks
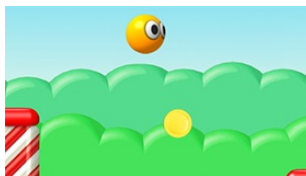
Can you take on the Mafia family in a game of billiards? There are trick shots, timed challenges and full matches to beat them all in.

Staff Pick

Hyperdiver

Take control of your spaceship and start new journey to far stars in this rogue-like sci-fi crawler.
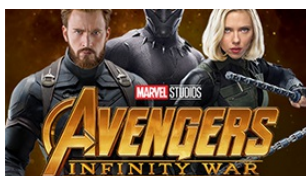

Mondo Hop

A cute side-scrolling platform game. Just bounce to the exit without falling down, collecting coins as you go. Can you complete all 64 levels?

Exoplanet Express

A modern version of Lunar Lander. Navigate the level, burn fuel and be careful with your precious cargo.
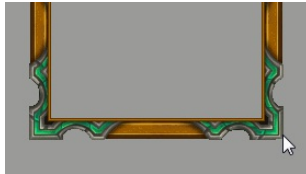

Avengers Infinity War

Take on the role of Captain America or the Black Widow in this gamified playable.
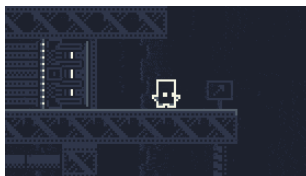
What's New?

Phaser 3 9-Slice Plugin

A powerful and easy-to-use new 9-slice plugin for Phaser 3 with docs and examples.

Gothicvania Cemetry Pack

A complete game art assets pack to build Castlevania style games with sample Phaser source included.

Modular Game Worlds in Phaser 3 Part 2

Part two of one of the most comprehensive, well written and illustrated tutorial on tilemaps in Phaser 3. This time, dynamic worlds.
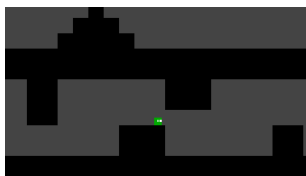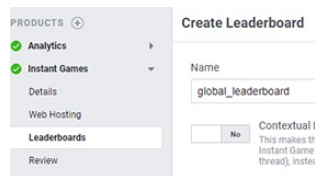
Yeah Bunny Tutorial

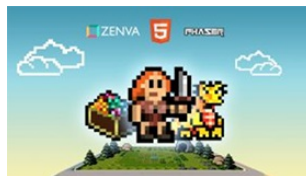Create a platformer inspired by the iOS game Yeah Bunny in Phaser 3 with Arcade Physics.

p4bXB0az0iC
xOjA4OjIxIC
LzE5OTkTkvMDI

Data URI Tutorial

Loading assets as data URI's in Phaser 3

FB Leaderboards Tutorial
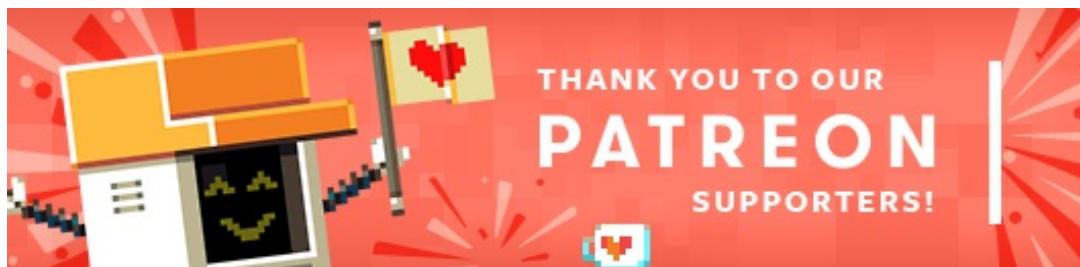
How to integrate Facebook Instant Games API in your Phaser game – managing leaderboards.

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.

Thank you to these awesome Phaser Patrons who joined us recently:

Sébastien Bernery
Jane

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.

Dev Log #126

Welcome to another dev log everyone! It's been a couple of weeks since 125 because I was on vacation and then we had a public-holiday here in the UK. During this, I went to Asylum X, the largest Steampunk festival in the world and had an absolute blast :) But it's time to put the goggles, cogs and gas-powered lights away and return the world of resolutely digital.

Phaser 3.12 Beta 3 Release

On August 23rd I published Phaser 3.12 Beta 3. You can download it from GitHub or grab it from npm under the **beta** tag.

Changes between Beta 3 and 2 are more incremental than revolutionary, as you'd expect this late into the cycle. As usual, the change log covers everything in detail but I'll highlight a few key features in this dev log and discuss plans going forwards.

Please download Beta 3 and try it on your projects. If you're coming from beta 2 you shouldn't have to change anything. If from an earlier version, hopefully very little, but check the log for details.

Release Plans

If you've even glanced at the 3.12 Change Log you'll appreciate how insanely large it is. It has also been over a month since the 3.11 release. This wasn't intentional. I prefer smaller more rapid releases. It's just the work involved in 3.12 spanned a lot of areas and took a while to complete, during which more PRs were merged and issues resolved. However, I'm quite worried about dropping such a huge release in a single build, so I'm going to split things up a bit.

I have moved the Facebook Instant Games Plugin and some other features, like the Scale Manager and DOM Game Objects, under a new experimental build flag. This means they are not included in the dist files, or node package unless you enable them with the conditional build flags. I've also moved the Camera3D system out into its own stand-alone plugin too. The plan is that I'm going to tackle a couple of remaining resolution related bugs and then release 3.12 as-is.

Very shortly after this, I'll release 3.13 which will contain the Facebook Instant Games Plugin. As stated before, this plugin is entirely optional and will provide a rapid and easy way to create Instant Games. It has been wonderful working with Facebook on this plugin and I'm really pleased with how much they like Phaser and are invested in its future. 3.13 will also contain the first usable version of the Scale Manager.

Once 3.13 is out I'll evaluate the state of play and look at 3.14. It will be good to get the GitHub issues total down (although a third of them are feature requests!) and some lingering PRs merged. 3.14 will focus on finishing the Scale Manager features and fixing any bugs which will have surfaced from the 3.12 release. This should take us to mid-September.

At this point, I'm going to recode Containers from scratch. I'm fed up with the issues they are causing and the limitations they are imposing on developers. It's time to get a proper full transform hierarchy in place. It will be one of those things that should have a very little API surface impact but will make a great difference internally and to developers lives.

Beyond that, who knows? There are *loads* of features it would be nice to implement. Proper bone-based animation support. Dynamic fx. Special audio features. Video support. Easier shader management. You name it, it's probably on my wishlist somewhere! Looking back at it, this has been a monumental year. We definitely took some wrong steps along the way, yet you could never say we haven't been working hard to resolve them and improve Phaser with every single iteration since. If you've helped towards this, by contributing a PR, raising an issue or just using it, thank you. And now, onto some new features from 3.12

Beta 3.

Tilemap Render Orders

If you use Tiled to create your tilemaps you may have noticed it has a feature where you can change the 'Render Order' of the map:
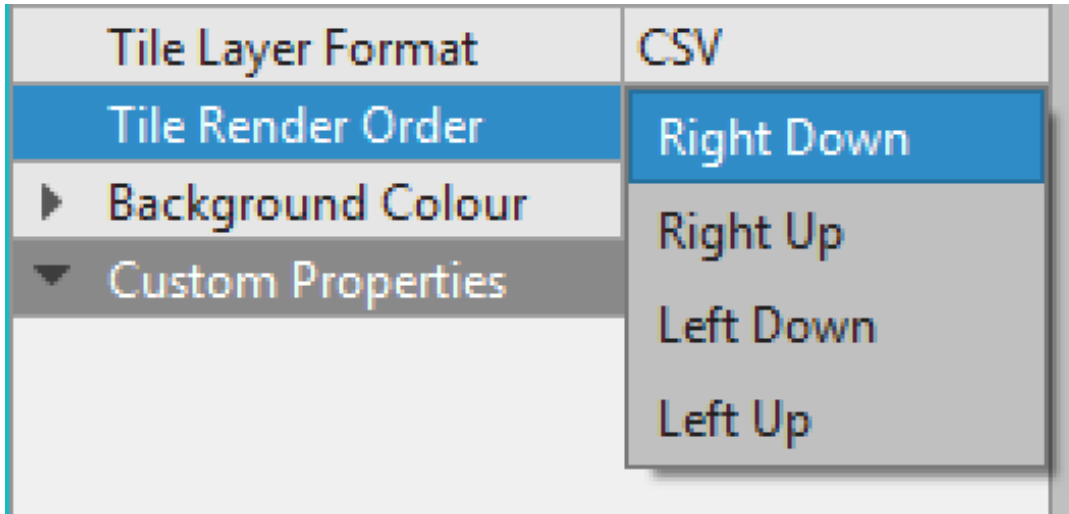
The Render Order controls the direction in which tiles are drawn when rendering a layer. The default is "Right Down", which is the only order that Phaser 2 and 3 have supported until now. When the renderer is drawing the tiles it will start at the top-left and then progress to the right, placing a tile down in each space. When it reaches the end of the row it moves down to the next column and repeats the process. This is what's known as 'Right Down'.

As you can see from the screenshot above there are 3 other orders as well. "Right Up" for example instead starts in the bottom-left of the layer, moves to the right, drawing tiles as it goes, but then moves up to the column above. "Left Down" and "Left Up" are, as you can guess, the same, but draw from the left to the right instead. You may wonder why it even has this option. The reason is that when you use overlapping tile spacing it impacts the way in which tiles render over the top of each other. Here is a small map drawn using the default 'right down' order:

Render Order: right-down
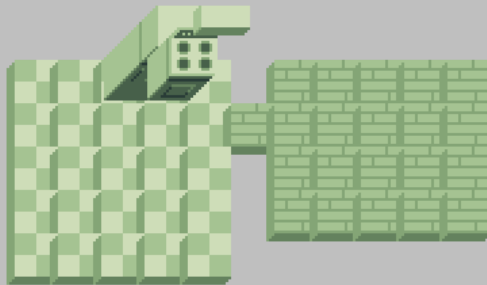
Doesn't look too hot, right? In 3.12 I added the feature for you to be able to specify the render order of a layer, so you can now pick from any of the 4 orders supported by Tiled. Here's the same map, using the correct 'left down' draw order:

Render Order: left-down
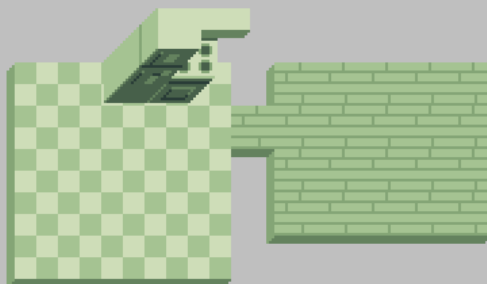
It's important to understand that the render order has no impact on the map data itself. It's still the same tiles, in the same positions, doing the same things. It's just when it comes to rendering the renderer iterates over the tiles in a slightly different way. It's a handy feature to have added and gives you more flexibility when creating your maps.

SVG Resize Options

Support for loading SVG files has been available in Phaser 3 for a while now. Yet, the loader was quite dumb. It worked, in that the SVG files were loaded properly and turned into textures, but that was all it did. Now, thanks to a new feature added to the SVG loader you have more control over this process.

When an SVG file is loaded the loader will check to see if the SVG data has a size defined. If not, it will try and extract one from its viewbox. It then uses these sizes to create a texture of the same size and renders the SVG into it, turning the vector data into a bitmap texture that the Canvas and WebGL renderers in Phaser can use.

The problem is that if the SVG file has a particularly large size, you'll have to scale it down using code within Phaser. This isn't difficult but it does meant it's just wasting GPU texture memory. The reverse is also true. If the SVG dimensions are quite small then you need to scale the texture up via code and the result is likely to look blurred and pixelated, negating the benefit of it being an SVG in the first case.

The SVG Loader now has two new options to help avoid both of these cases. You can either specifiy an exact width and height for the SVG texture, or you can give it a scale:

```javascript
this.load.svg('cartman', 'assets/cartman.svg', {
  width: 416,
  height: 388
});

this.load.svg('fireflower', 'assets/flower.svg', {
  scale: 0.25
});
```

In the above code, you'll see that the fireflower SVG is being scaled down to a quarter of its original size. The size defined in the SVG file exported from Adobe Illustrator was 864 x 864 pixels. This would use a lot of texture memory if it was never going to be displayed at that size in-game. By setting the scale to 0.25 the resulting texture will be only 216 x 216 pixels.

Equally, you can see the Cartman SVG has been given a fixed size, increasing it from the original tiny viewbox size saved in the SVG. The following demo shows the difference:

Click to toggle sizes

Of course, it's possible you could just fix all of your assets in whatever vector tool you use. However, this feature means you could also easily implement @x2 or beyond scaling to a base set of SVG files.

Matter.js Update

I have upgraded the version of Matter.js used in Phaser 3.12 from 0.13.1 to 0.14.2. Most of the changes were documentation related but there is still a handful that will impact Phaser devs, such as fixing the Body scaling for compound bodies or fixing the chamfer radius argument.

You can see the full list on the Matter change log but I also included the most

important ones in the Phaser change log.

A few people have asked why I don't include Matter.js from npm so it could be kept updated more easily. The reason is that Phaser uses a slightly customized version of Matter. Earlier this year a whole bunch of optimization PRs were submitted to the Matter repo which made a huge difference on the collision resolution speeds. After testing them I elected to merge those into our version of Matter. However, they're not in the official master branch yet.

I also made a couple of small tweaks to the bundle to prevent it from including all of the Pixi Renderer files, which would just be pure wasted space in Phaser. And added in support for back-tracing from a Body to its parent Game Object. They are quite small changes but have a big impact on use. So that's why we don't suck it in from npm and cannot do so for the forseeable future.

Phaser 3 Slopes Plugin

Related to both tilemaps and Matter.js, hexus has just released a new version of his Phaser 3 Slopes Plugin. The aim of the original Phaser 2 plugin was to add sloped tile support to Arcade Physics, something v2 didn't provide natively.

The Phaser 3 version of the plugin will aim to provide that eventually as well, but the main focus is on smoothing out Matter.js / Tilemap collisions. When using Matter you often encounter the issue of ghost collisions. There are collisions where the edges of a body strike the edge of a tile body. The following animated gif demonstrates this in action:

```
Ghost Collisions Demo
Grass: Individual Tile Bodies
Stone: A Single Convex Body
```

As you can see, the objects collide where they shouldn't. The plugin solves this problem and using it is a breeze! Just load the plugin like you would any other Phaser 3 plugin, and it'll automatically ensure that Matter Tilemap Layers collide smoothly with other physics bodies. Resulting in this:

Ghost Collisions Demo
Grass: Individual Tile Bodies
Stone: A Single Convex Body

Lovely :) You can grab the plugin from the link above and test it out in your projects. Leave feedback on the GitHub repo.

That's it for this Dev Log. I'll be back next week where we should have a finished 3.12 release as well.

[Streets of Rage 4](#)? Hell yes :)

Here are over [200 professional music loops](#) available for free commercial use.

[Laserface](#) San Francisco - lots of lasers, lots of EDM :)

Phaser Releases

Phaser [3.12.0 Beta 3](#) released August 23rd 2018.
Phaser CE [2.11.0](#) released June 26th 2018.

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.

Preferences   Forward