

Web Version

Unsubscribe

PHASER WORLD

APRIL 2019

ISSUE
145



THIS WEEK...

ENDLESS RUNNER TUTORIAL

ROCKETATE NEXT

PHASER 3 UI COMPONENTS

GO CHICKEN GO!

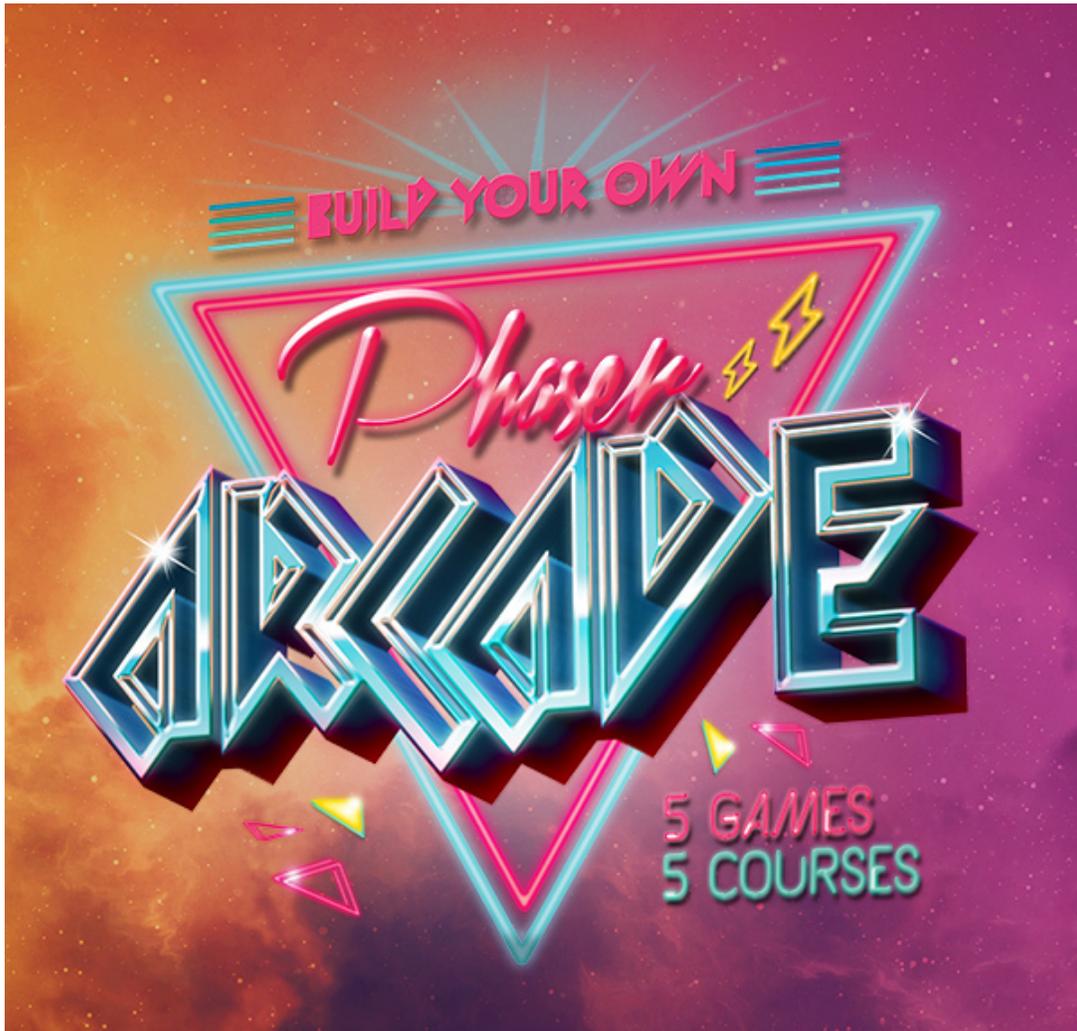
Welcome to Issue 145 of Phaser World

Like many of you, I had a couple of days off as part of the Easter break. The weather here in the UK was glorious, for a change, and it was nice to recharge the brain cells away from the screen for a short while. Now I'm back, it's full steam ahead on the 3.17 release. You can read about the most recent changes in this issues Dev Log.

When adding the games to the site for this issue I had far more fun than I probably should have, playing the stupidly amusing Go Chicken Go! Such a simple concept made all the more fun by the animations and sound effects. Jewels Blitz 3 is almost the opposite end of the spectrum, instead, being polished so far you can see your face in it.

I always find it interesting to see new games created in Phaser and, honestly, it is what motivates me more than anything to carry on :) So please do tell me about your games. It may take a few months for them to appear on the site, but they will eventually! As usual, be sure to [read this newsletter on the web](#) so you don't miss anything.

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 [Build Your Own Phaser 3 Arcade Games Pack](#) is a brand new release and available to buy today. It contains 5 complete training courses that teach you how to remake 5 classic retro games in full ES6 with Phaser 3. Learn how to make Pong, Breakout, Asteroids, Nibbler and Space Invaders with the full tutorial and code per game. If you create a game based on one of these tutorials, please let us know!



The Latest Games



Game of the Week

Jewels Blitz 3

A massively polished match 3 game set deep in Mayan lore. Hundreds of levels, boosts, puzzles and more to be unlocked will keep you playing for days.



Staff Pick

Go Chicken Go!

The last obstacle between 25 chickens and their freedom is a highway and a river. What could possibly go wrong?



Rocketate Next

Use your jetpack, avoid electric cables and shift the boxes around as you rotate the maze to try to get to the exit in this 60 level tricky puzzler.



Jelly Bomb

Can you clear the table of jelly by setting off chain reactions of exploding jelly bears?!



Snake 99

A local multiplayer version of Snake with Domination and Deathmatch modes, inspired by Tetris 99.



What's New?



Phaser 3 UI Components

Scripts and classes that allow you to build your game UI using Adobe Animate and export directly.



Dungeon Dash

An open source dungeon crawler using Phaser 3, TypeScript and public domain assets.



New Tutorial Translations Available

The 'Making your First Phaser 3 Game' Tutorial is now available in 8 new languages including Chinese, Spanish and Polish.



Match 3 Tutorial Part 1

Emanuele Feronato starts a new tutorial series on creating a Match 3 game using ES6.



Endless Runner Tutorial Part 2

In the second part of the series player animation and customizations are added.



Endless Runner Tutorial Part 3

In part three of the series textures and coins are included in the game.



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:

PHASER
SUPPORT PHASER & UNLOCK:

PERKS	MORE PERKS
BACHERS NEWSLETTER A NEW REGULAR EDITORIAL STYLE NEWSLETTER FULL OF INTERESTING AND RELEVANT GAME DEV LINKS	FREE COPY OF INTERPHASE DOWNLOAD A FREE COPY OF THIS PHASER 2 BOOK AND CODE PACK - PLUS GET EARLY ACCESS TO INTERPHASE 3
EXAMPLES ARCHIVE A BACHERS-ONLY FILE ARCHIVE OF PHASER EXAMPLES, GAMES, SNIPPETS AND RESOURCES (COMING DEC. 2019)	PHASER STICKERS PACK PUMP YOUR RIG WITH THESE GREAT VINYL STICKERS! FREE FOR \$30+ AND DISCOUNTED FOR ALL TIERS BELOW
PRIVATE DISCORD CHANNEL YOU'LL GAIN ACCESS TO A BACHERS-ONLY DISCORD CHANNEL, WHERE YOU CAN CHAT WITH FELLOW DEVS	ROADMAP ROUND-TABLE HELP STEER THE FUTURE OF PHASER BY JOINING THE MONTHLY ROUND-TABLE DISCUSSING WHAT'S COMING
ONE-ON-ONE TECHNICAL SUPPORT CHAT DIRECTLY WITH THE DEVELOPER OF PHASER! GET BUGS FIXED OR CODE EXPLAINED (\$10+ BACHERS)	GAME PROMOTION WANT SOME MORE EYEBALLS ON YOUR LATEST GAME? WE CAN HELP WITH SITE AND NEWSLETTER PROMOTIONS
EARLY ACCESS GET EARLY ACCESS TO NEW TUTORIALS, BOOKS, PLUGINS AND VIDEOS. THEN GET A DISCOUNT ON RELEASE	ADVERTISING OPPORTUNITIES FOR HIGH TIER BACHERS - WE CAN RUN BANNER ADS, TWEETS AND MAILING CAMPAIGNS FOR YOUR PRODUCTS
FORUM BADGE SHOW OFF YOUR BACHER STATUS WITH A SPECIAL BADGE ON THE HTML5 GAME DEVS FORUM	HELP SECURE PHASER'S FUTURE THE MOST IMPORTANT REASON OF ALL: YOUR BACKING HELPS SECURE THE FUTURE OF PHASER

Click to see the full list of backer perks

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:

Aqil

Atomic Films

Beda Binder

Daniel Albu

Daniel Peer

Dries Schelfhout

Isaac Gifford

Marian Boda

medecau

Michaela Adams

Oliver Lanz

Stephan Fowler

straker

Tomas

Wilmer Marchan

Also, thank you to **Andrew McOlash** for his donation :)



Dev Log #145

Welcome to Dev Log 145. Work has been progressing at a rapid rate these past few weeks. I've merged a large number of Pull Requests and have been fixing and closing down issues too.

Thanks to a really neat PR from @florianvazelle the Geometry functions have gained a whole bunch of new Intersects tests, such as Circle to Circle, Line to Circle, Triangle to Triangle and many more. Although these existed previously, all they used to return was a boolean. Now thanks to the PR, the new functions will return the point of intersection as well, which is extremely useful in all kinds of situations. Thanks also to @rexrainbow for helping debug these.

Phaser Tutorial Translations

Several issues ago I posted asking for help translating the core Phaser tutorials into languages other than English. I had a good response and am pleased to say that the first 8 translations of the [Making your First Phaser 3 Game](#) tutorial have now gone live onto the site.

We've translations into Bulgarian, Chinese, Czech, Danish, Polish, Portuguese, Slovakian and Spanish uploaded. My thanks to Tomas, Telinc1, Fernando, Huang, Adriano, Juraj, Frederik and Konrad for sending me those translations. I'm still hopeful we'll have more translations available soon, as other people did get in touch last time, they just haven't sent the files yet.

If you would like to translate this tutorial, and are happy editing Markdown files, please email support@phaser.io and I'll send you more details. The more translations we have, the better :)

I'll be looking at the site stats in a few weeks to see which translations are proving to be the most popular, or indeed if they are being used at all. I would hope they are, but the stats will tell!

DOM Elements leave Experimental

DOM Elements were added to Phaser nearly a year ago, back in July 2018. I [wrote about them](#) at the time and published some examples of how they work. Because they hadn't had much testing I kept them under the EXPERIMENTAL flag, which meant they were excluded from the dist builds of Phaser unless specifically enabled by you in the webpack config.

A lot of you still used them though. Which isn't really surprising I guess, what with the on-going rise of IO games and the need for login forms, leaderboards and such-like. I mean, sure, you can do that in Phaser using custom UI components, but really, it's what the browser was created for! So it makes complete sense to use it.

I spent a few days refining how they work, adding in some more options to the constructors and hooking them to the Scale Manager. There is still one lingering bug to do with the Scale Manager auto centering content and initial sizes, but otherwise, I'm happy with how they're progressing, so they'll be available fully in 3.17. My thanks to the team at Quest AI for helping test these.

One of the new features in 3.17 is the ability to load in CSS files to the browser via the normal Phaser Loader. You can also now set the class name of a DOM Element directly, so it's a lot quicker and cleaner to create them than when they were experimental. Here's an example demo:

```
function preload ()
{
    this.load.css('80s', 'assets/loader-tests/80stypography.css');
}

function create ()
{
    var h1 = this.add.dom(450, 100, 'h1', null, 'CHROME');

    h1.setClassName('chrome');

    var h2 = this.add.dom(570, 180, 'h2', null, 'Dreams');

    h2.setClassName('dreams');
    h2.setAngle(-15);

    this.tweens.add({
        targets: [ h1, h2 ],
        y: 500,
        duration: 3000,
        ease: 'Sine.easeInOut',
        loop: -1,
        yoyo: true
    });
}
```

In the above code, you can see the new load.css call being made. This loads in the CSS file, adds it into a style element and appends it to the document head, causing all styles defined in there to be applied immediately.

In the create function I'm creating two DOM Elements. The H1, with the class of 'chrome' and the H2 with the class of 'dreams', and a slight angle for effect. The elements are then tweened up and down just as you would any other Phaser Game Object. It looks like this when running:



You'll notice a 'flash of content' when this example starts up, because the CSS file in turn loads in some web fonts that the demo uses. This is a common problem faced by web developers (the flash of unstyled content), so there are plenty of documented ways to handle it. For the sake of this simple demo, I'm happy to leave that up to you.

You can find more [examples in the Labs](#).

I of the Mask

Another issue I had to address this week revolved around masking. Masks have always been a bit of a mixed bag in Phaser 3. In v2 you could only ever create Geometry Masks, which internally used a Graphics object per mask and created a stencil buffer for each mask as they were processed. However, in v2 they could be stacked up to a huge depth (255 children deep). In v3 I wanted more powerful masks, which meant proper bitmap based masks as well as geometry ones. We implemented this from the very first release, with geometry masks

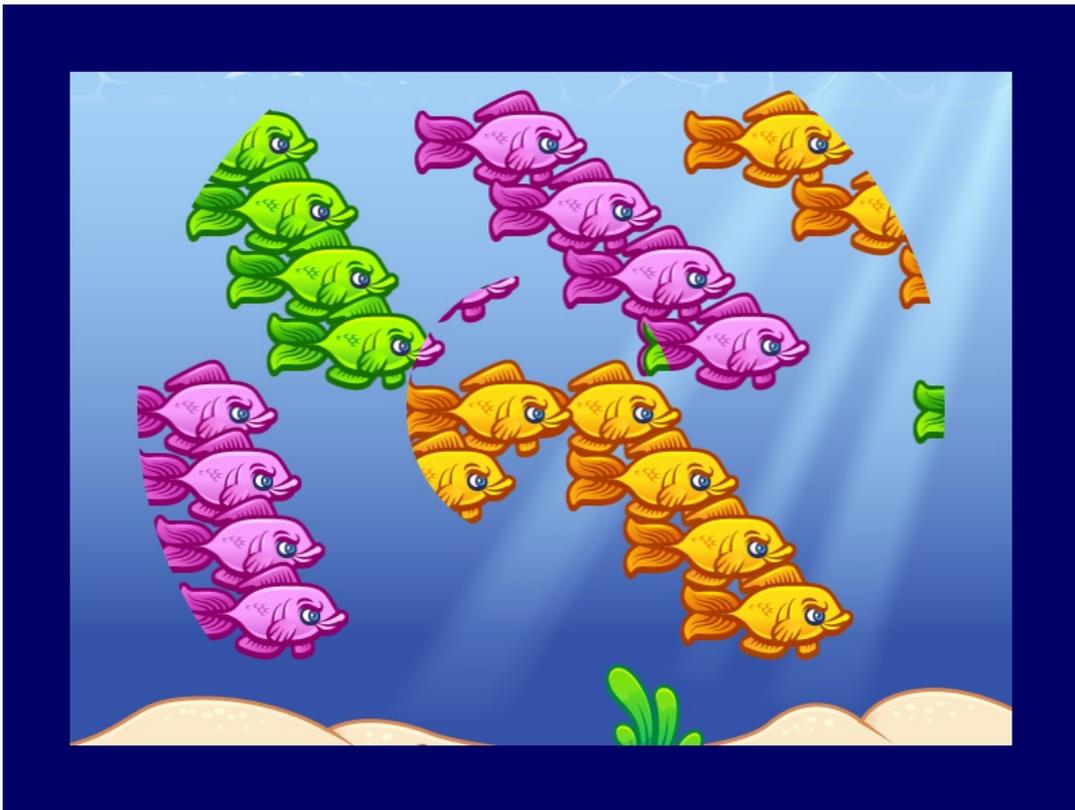
using Graphics objects and a single stencil buffer, or path operations under canvas, and bitmap masks using a frame buffer, making them WebGL only.

Originally though, v3 only had a single-depth display list. None of the Game Objects had children and there was no need to carry masks on down the list. This kept things simple and extremely fast and the masks worked fine under these restrictions. However, there was so much demand for Containers we had to fit them in retrospectively, and as well as introducing a number of issues re: branching, it also meant that masks no longer worked properly on them. You could mask a top-level Container, but any masks a descendant, of any depth, may have, would be ignored. What's more, one of the things I'd always wanted to do, which was to allow a Camera to have a mask, couldn't work either.

With a number of mask-related bugs lingering unresolved in GitHub I started digging into them. As suspected, the Geometry masks stencil buffer was hard coded at only one level deep, and the Bitmap masks frame buffer didn't handle passing control back to another framebuffer, meaning they couldn't be stacked either.

I tackled Geometry Masks first.

I actually took inspiration from the way in which the stencil queue had been handled in Phaser 2. I backported some of that code, merged it with the new structure and after a lot of testing geometry masks were finally working again at multiple depths. Here's an example of a Scene using multiple masks:



It's better to see it running for the full effect. Click the above image to do that. The code is no different from before:

```
var shape1 = this.make.graphics().fillRect(50, 50, 700, 500);
var shape2 = this.make.graphics().fillCircle(400, 300, 300);
var shape3 = this.make.graphics().fillCircle(400, 300, 100);

var geomask1 = shape1.createGeometryMask();
var geomask2 = shape2.createGeometryMask();
var geomask3 = shape3.createGeometryMask();

particles.setMask(geomask2);

emitter2.setMask(geomask3);
```

Here I simply create three different Graphics objects that function as the masks. Create Geometry Masks from them and then apply them to the various Game Objects. In this Scene, there is a single Particle Emitter Manager (called particles in the source) and two Particle Emitters belonging to it (emitter1 and emitter2).

You'll notice that geomask2 is applied to the Particle Emitter Manager itself. This means that any emitter this manager is handling will be masked. This has always worked in Phaser 3. What's new, though, is the way I've also set geomask3 on

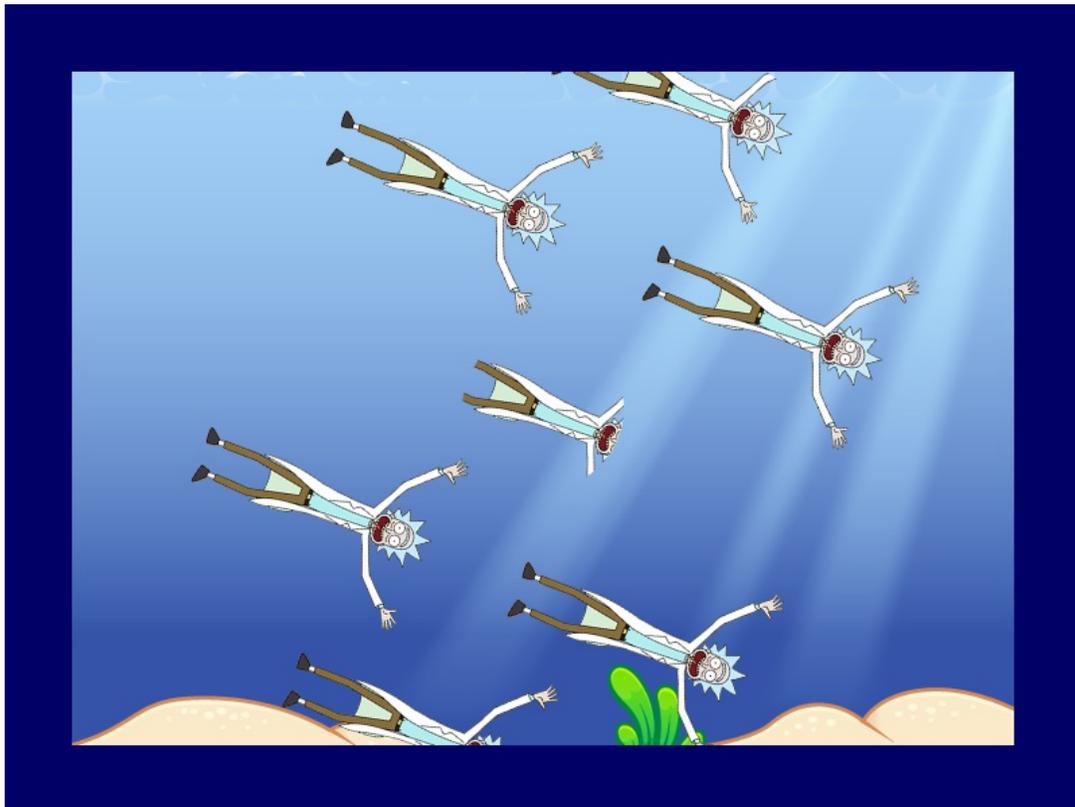
emitter2. Previously, you couldn't have a mask applied to an emitter directly, only to the manager. Now in 3.17, you can have both. If emitter2 wasn't using geomask3 it would use geomask2 instead because that has been applied to its parent, but in this case, it overrides the parent mask.

If you run the demo above you can move the camera with the cursor keys. Notice that the masks are now applied in camera space, so will move with the camera.

Container Masks

Previously, a Container would ignore the fact that any of its children may be masked. You could mask the Container itself of course, just not its children, no matter what type of Game Object they were.

This has been fixed for Geometry Masks in 3.17. You can now mask a child, of any depth, and the parent Container, and it'll use the correct mask accordingly. Again, it's quite hard to see in a static screenshot, but if you click the image below you can see it running:



Camera Masks

The final piece of the puzzle was something I'd wanted to add for a long time: the ability to mask Cameras.

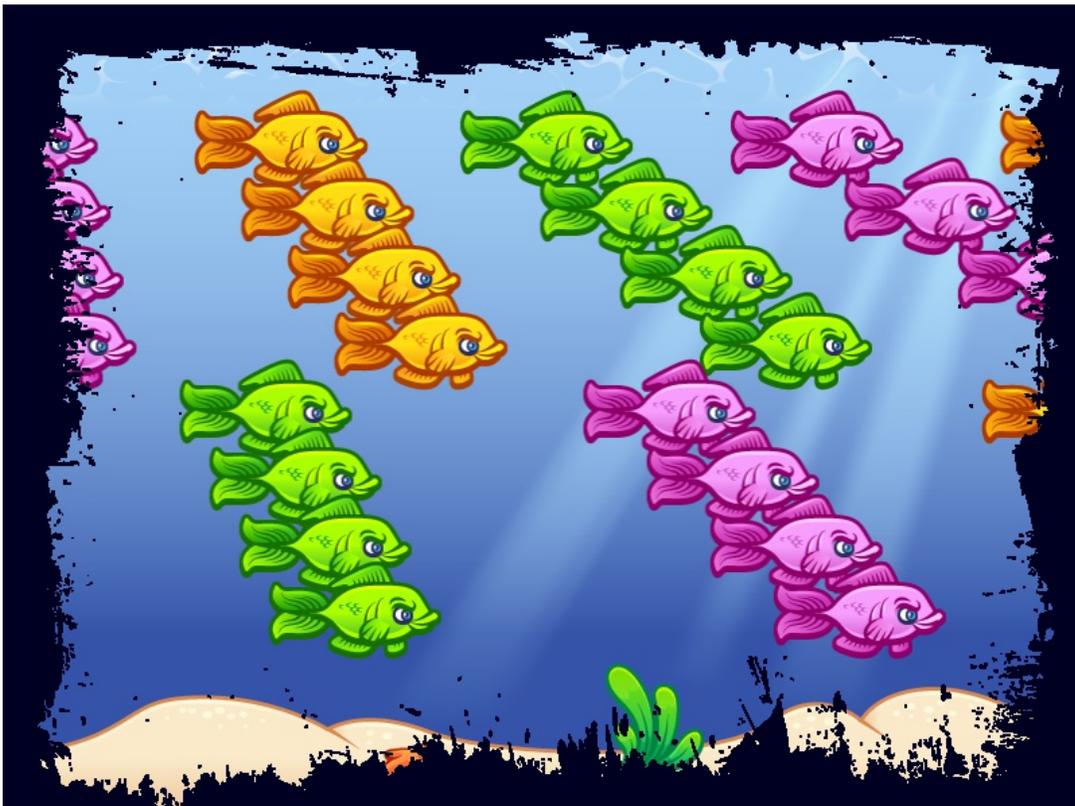
This is now possible in 3.17. I've added a new method to the Camera class called `setMask` and the related `clearMask` method. You can set either a Geometry or Bitmap Mask on a Camera and, as you'd expect, it will impact anything it renders. Here's a small snippet showing the creation of a Bitmap Mask, made from an image loaded in the Loader:

```
var maskImage = this.make.image({
  x: 400,
  y: 300,
  key: 'mask',
  add: false
});

var mask = maskImage.createBitmapMask();

this.cameras.main.setMask(mask, false);
```

When run, it looks like this (as usual, click the image to run the demo):



The mask is applied to the top-level, impacting the images and particles in this Scene. If you were to add another Camera, it would of course not be affected by this mask. Which makes this an interesting way to apply masks selectively.

A special feature of Camera Masks that I added is the ability for them to remain fixed in place. If you use the cursor keys in the above demo you'll notice that the mask moves as the camera moves. This is optional. You can also create a static mask on a camera that ignores the camera scrolling. The following demo shows the difference:



Here you can see the mask remain firmly in place, even though the camera is scrolling around a tilemap. In the first demo try pressing the Q and E keys to zoom in and out of the Scene. With the fixed camera, the mask doesn't zoom. With the flexible one, it does. Also, press Z and X to rotate the Camera. Again, in the first demo, the mask rotates with it, but it won't do that with a fixed mask.

I believe that having these two options will give you a lot of flexibility when it comes to masking your games.

I still need to work on the mask stack and see if the same process can be applied to Bitmap Masks. It's working as intended for Geometry Masks, because I've recoded how the stencil buffers are used, but I've not yet done this for Bitmap Masks. They need stacked framebuffers, which are a whole different kettle of

fish. I'll spend a day on it and see how things go. I would be happy to publish 3.17 with just Geometry Masks working at any depth and extend this to Bitmap Masks in a later release, but if it turns out to be quite easy to solve, I will, of course, do so.

3.17 Release

My focus right now is on getting 3.17 ready for release. Masks were a bit of a diversion for me really. The aim was simply to close down some old issues in GitHub, but as is often the case, the act of doing this uncovered a legacy structural flaw, rather than a code one, which is always harder to fix.

With Easter break over, I'll be working on 3.17 solidly now, so look out for constant commits to GitHub! If you have time to help then I'd appreciate it, either by testing new builds, or reporting issues or seeing if there are any you could help close. Every little bit helps.



PixelJoint collabs are nothing new, they've been going on for years. But [this isometric one](#) is nothing short of stunning!

The Chemical Brothers have a new album out called [No Geography](#) and it's pretty damned good.

I [used the web for a day](#) on Internet Explorer 8 (you brave, brave person)

Phaser Releases

Phaser 3.16.2 released February 11th 2019.

Phaser CE 2.12.0 released 6th February 2019.

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.



©2019 Photon Storm Ltd | Unit 4 Old Fleece Chambers, Lydney, GL15 5RA, UK

[Preferences](#)

[Forward](#)

Powered by [Mad Mimi](#)®
A GoDaddy® company