

Web Version

Unsubscribe

PHASER WORLD

DECEMBER 2018

ISSUE
134



THIS WEEK...

SNAIL BOB 8 - ISLAND STORY

QUEST.AI

FANTASY WARS

SOKOBAN TUTORIAL SERIES

Welcome to Issue 134 of Phaser World

In true Christmas tradition, this is a double-sized issue! So if your mail client truncates it, be sure to [read it on the web](#) so you don't miss anything.

It feels strange to think that in a few days we'll enter 2019. This year seems to have flown by at warp speed. It's been quite the year for Phaser and it never ceases to amaze me how vibrant you and the rest of the community truly are. A large number of you seem to have managed the transition to Phaser 3 well and we've now hundreds of incredible tutorials written about it and games created with it. I'll publish my annual Phaser retrospective in the next issue and look back on what happened, what worked, what didn't and what's coming next.

For now, please enjoy this final 2018 dose of Phaser news and, if you celebrate it, have a fantastic Christmas. Things will be quieter across GitHub and the forum as I take some time off, back in the Phaser office again January 4th 2019.

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](#).



Fresh batch of Phaser Sticker Packs arrived

Regular readers will know that I've been mailing out literally hundreds of Phaser sticker packs during December. I've loved seeing the photos you sent as the stickers arrived and got plastered over laptops, fridges, and workspaces.

Due to their unexpected popularity, I've had another batch printed, so there's now

plenty of them in stock. What's more, the next 100 orders will get a **Phaser pin-badge** and a new **pixel-art sticker** included in the packs as well, for no extra charge.

Please note that no sticker packs will be posted between December 21st 2018 and January 3rd 2019. I will resume mailing packs again on January 4th.

[Click here to order a sticker pack.](#)



The Latest Games



Game of the Week

[Snail Bob 8](#)

All Bob wants to do is catch some fish, but instead he has to puzzle his way through 30 increasingly fiendish levels in this superb puzzle platform game.



Staff Pick

[ROCKet Racing](#)

The newest extreme sport for rich people with too much time and money. Try to race your rocket-powered-rock down the hill faster than the opponents!



Fantasy Wars

A turn-based strategy game where six fantasy races fight it out to the death in a small crowded world.



Ho ho Hotride

Santa grabs some skis and must weave his way to the village in this voxel styled Christmas game.

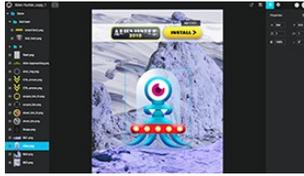


The Adventure of Bleekofla

You play an intergalactic scout, trying to recover your ship, in this charming pixel based platformer.



What's New?



Quest.AI

Build complete Phaser games, interactions and product pages directly from PSD files, using a comprehensive visual front end to apply effects and publish to multiple platforms.



Phaser Backer Examples December 2018

The Phaser Backer Examples are now available including a Scrollbars component, Scene Builder and Swipe Gestures.



Facebook Instant Games Ads Tutorial

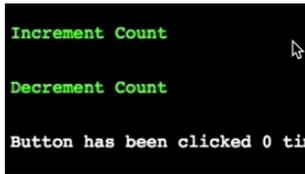
Learn how to create and display ads in your Facebook Instant Games using the Phaser 3 Plugin.



DesignCode Phaser 3 Course

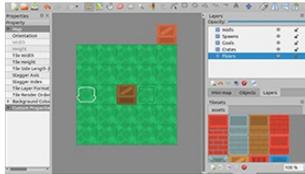
DesignCode have a brand new step-by-step course available about creating a fun game in Phaser 3.





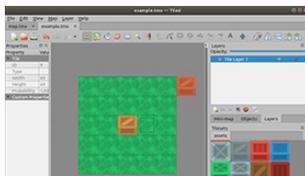
Buttons in Phaser 3

A tutorial on creating your own Text Button Game Object in Phaser 3.



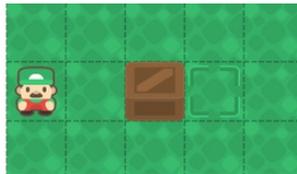
Sokoban Phaser 3 Tutorial

A comprehensive tutorial series on creating the classic game Sokoban in Phaser 3.



Sokoban Phaser 3 Tutorial Part 2

In part 2 of this comprehensive tutorial series it covers creating a fully interactive level.



Sokoban Phaser 3 Tutorial Part 3

In part 3 of this comprehensive tutorial series it covers handling input and game logic.



Sokoban Phaser 3 Tutorial Part 4

In part 4 of this comprehensive tutorial series it covers creating a game overlay in React.



Sokoban Phaser 3 Tutorial Part 5

In the final part of this comprehensive tutorial series learn how to publish your game and earn money from ads.



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.

Back the Phaser project

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. It's part of our manifesto that the core framework will always be free, even if you use it commercially, as many of you do.

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

Those funds allow Phaser to improve, and when it improves, everyone involved benefits. 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

- BACHERS NEWSLETTER**
A NEW REGULAR EDITORIAL STYLE NEWSLETTER FULL OF INTERESTING AND RELEVANT GAME DEV LINKS
- EXAMPLES ARCHIVE**
A BACHERS-ONLY FILE ARCHIVE OF PHASER EXAMPLES, GAMES, SHIPPETS AND RESOURCES (COMING DEC. 2019)
- PRIVATE DISCORD CHANNEL**
YOU'LL GAIN ACCESS TO A BACHERS-ONLY DISCORD CHANNEL, WHERE YOU CAN CHAT WITH FELLOW DEVS
- ONE-ON-ONE TECHNICAL SUPPORT**
CHAT DIRECTLY WITH THE DEVELOPER OF PHASER! GET BUGS FIXED OR CODE EXPLAINED (\$10+ BACHERS)
- EARLY ACCESS**
GET EARLY ACCESS TO NEW TUTORIALS, BOOKS, PLUGINS AND VIDEOS. THEN GET A DISCOUNT ON RELEASE
- FORUM BADGE**
SHOW OFF YOUR BACHER STATUS WITH A SPECIAL BADGE ON THE HTML5 GAME DEVS FORUM

MORE PERKS

- FREE COPY OF INTERPHASE**
DOWNLOAD A FREE COPY OF THIS PHASER 2 BOOK AND CODE PACK - PLUS GET EARLY ACCESS TO INTERPHASE 3
- PHASER STICKERS PACK**
PUMP YOUR RIG WITH THESE GREAT VINYL STICKERS! FREE FOR \$30+ AND DISCOUNTED FOR ALL TIERS BELOW
- ROADMAP ROUND-TABLE**
HELP STEER THE FUTURE OF PHASER BY JOINING THE MONTHLY ROUND-TABLE DISCUSSING WHAT'S COMING
- GAME PROMOTION**
WANT SOME MORE EYEBALLS ON YOUR LATEST GAME? WE CAN HELP WITH SITE AND NEWSLETTER PROMOTIONS
- ADVERTISING OPPORTUNITIES**
FOR HIGH TIER BACHERS - WE CAN RUN BANNER ADS, TWEETS AND MAILING CAMPAIGNS FOR YOUR PRODUCTS
- HELP SECURE PHASER'S FUTURE**
THE MOST IMPORTANT REASON OF ALL: YOUR BACHING 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.

Patreon renews monthly, just like Netflix. You can, of course, cancel at any point. Tears will be shed on this end, but that's not your concern.

All Amounts Matter

Traditionally, the Patreon platform promotes the concept of "the more you pledge, the more rewards you get". Their whole system is geared around this. I felt, though, that for something like open-source support it was time to level the playing field. This is why all rewards are available to everyone, regardless of how much they pledge. There are a couple of exceptions, which I outline on the backers page, yet to me, this felt like the fairest approach to take.

If you're able to be a higher tier backer, then I cannot stress enough how much I thank you for that. Please don't feel that your contribution is undermined in any way. At the end of the day, the money is being used to keep development of Phaser going, and that will never falter. This move is about making everyone feel equally rewarded, regardless of financial ability. Equally, if it's at all possible to

slightly increase your pledge at some point, there are now lots of more minor-increase tiers you can explore.

Backer Perks

- Backers
Newsletter
(issue
#1
due
January
2019)
- Exclusive
Backers
Examples
(first
batch
out
now)
- Private
Discord
Channel
- One-
on-
One
Technical
Support
(time
restrictions
apply,
see
page)
- Free
copy
of
Interphase
1
- Early
Access
to
new
Tutorials
- Early
Access
to

- new
Books
and
Plugins
- Forum
Badge
- Phaser
Laptop
Stickers
- Roadmap
Round
Table
- Promotion
and
Advertising
for
your
games
or
product
- Special
Offers

[Please help support Phaser on Patreon](#)

Full details about the perks can be found on the [Phaser backers page](#).



Thank you to these awesome [Phaser Patrons](#) who joined us recently:

Andrzej Mazur
Ayman Maas
Bobonthenet
Cetian
Chris B Paynter
Cyrus Elder
Erlend Sogge Heggen

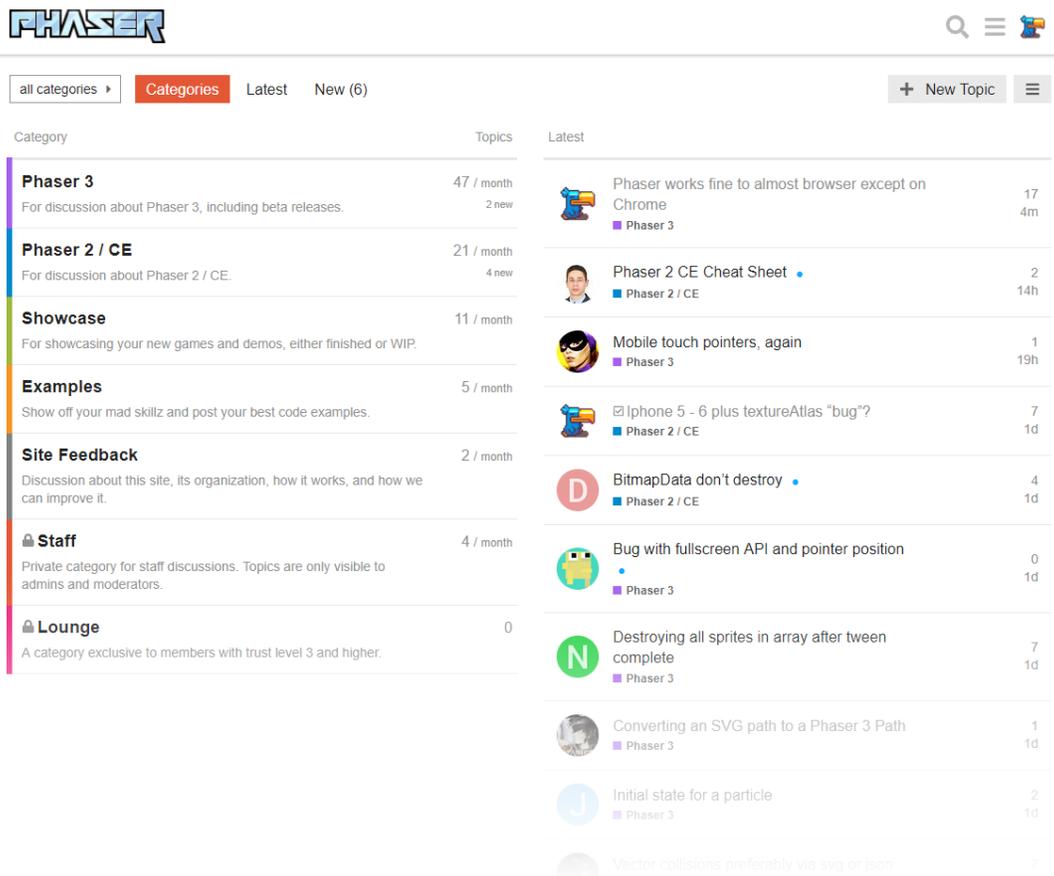
Esteban Izaguirre Peñin
Francesco Raimondo
Francois-Cedric Weber
Gokce nur Oguz
Hans Gerber
Howard Schwid
Jairo Molina Jr.
Jakub Joras
Javier Corra
Jogos
Jordan Roher
Kevin Ryan
KingCosmic
Konrad Fedorczyk
Manuel
Marcus Krupa
Michael West Hadley
Pablo Cotoia
Peter Achberger
Peter Ginsberg
Pierre Poupin
Renaud Collet
Reza Akbar
Rob Lynch
Robert Greenberg
Sargis Sargsyan
Steve Barr
Tatsuro Ido
Thomas Lynge
Tilde White
Vincent Rose
Yalon Keret

Also, thank you to **Usikre Soren** and **Pham Quan** for the donations.



Dev Log #134

Welcome to the final Dev Log of 2018. I'm not going to get all retrospective on you. Not this issue, at least. Even so, it's been *quite* the year! I wanted to cover what I've been working on for the past couple of weeks and what's left to be done before I can release Phaser 3.16. First of all though, let's talk about the forum.



The screenshot shows the Phaser forum interface. At the top left is the Phaser logo. To the right are search and menu icons. Below the logo is a navigation bar with 'all categories' (dropdown), 'Categories' (highlighted), 'Latest', and 'New (6)'. On the right of the navigation bar is a '+ New Topic' button and a menu icon. The main content area is divided into two columns: 'Category' and 'Latest'.

Category	Topics
Phaser 3 For discussion about Phaser 3, including beta releases.	47 / month 2 new
Phaser 2 / CE For discussion about Phaser 2 / CE.	21 / month 4 new
Showcase For showcasing your new games and demos, either finished or WIP.	11 / month
Examples Show off your mad skillz and post your best code examples.	5 / month
Site Feedback Discussion about this site, its organization, how it works, and how we can improve it.	2 / month
Staff Private category for staff discussions. Topics are only visible to admins and moderators.	4 / month
Lounge A category exclusive to members with trust level 3 and higher.	0

Latest	Topics
Phaser works fine to almost browser except on Chrome ■ Phaser 3	17 4m
Phaser 2 CE Cheat Sheet ■ Phaser 2 / CE	2 14h
Mobile touch pointers, again ■ Phaser 3	1 19h
iPhone 5 - 6 plus textureAtlas "bug"? ■ Phaser 2 / CE	7 1d
BitmapData don't destroy ■ Phaser 2 / CE	4 1d
Bug with fullscreen API and pointer position ■ Phaser 3	0 1d
Destroying all sprites in array after tween complete ■ Phaser 3	7 1d
Converting an SVG path to a Phaser 3 Path ■ Phaser 3	1 1d
Initial state for a particle ■ Phaser 3	2 1d
Vector collisions preferably via svg or json	7

New Phaser Forum

Since Phaser first began, we have maintained a board over at the HTML5 Game Devs Forum. This has been great for many years but we felt it was time to move to our own home. I had looked at Discourse before. As a forum app, it seemed light-years ahead of what Invision, the software at HTML5GD, could do. Sadly, it was just too expensive for us to consider. So all those power features seemed out of reach.

Which is why, when Discourse offered Phaser a free managed installation, as part of their initiative to help open source projects, we jumped at the chance! :)

I had the option to move over all of the posts from the HTML5GD forum to the

new one. Which would have given us a rich vein of archived material to draw on. However, it was impossible to determine which users had registered on HTML5GD just for Phaser, rather than one of the other projects hosted there, like Babylon.js. I also felt that, as a user, I wouldn't be happy to find my account details moved from one site to another without my permission. With those two facts in mind, I took the decision to start the new forum afresh.

So, [please join the new Phaser forum](#). It's still a bit of a blank slate, albeit a much more powerful one than before! It has proper integration with GitHub, Patreon and other services. A really neat notification system, genuinely useful email summaries and a lot more. I will keep the HTML5GD Phaser boards open, but they will eventually be moved to be read-only.



The Copper Bones Key

In the last Dev Log, I talked about how the work on the Scale Manager required me to recode parts of the input system to deal with the new scale modes. As a continuation of this work I also revisited the way the Keyboard Events are handled in Phaser 3.

The specificity of the Keyboard events has been changed to allow you more control over event handling. Previously, the Keyboard Plugin would emit the global ``keydown_CODE`` event first (where CODE was a keycode string, like ``keydown_A``), then it would emit the global ``keydown`` event. In previous versions, ``Key`` objects, created via ``this.input.keyboard.addKey()``, didn't emit

events.

The `Key` class now extends `EventEmitter` and emits two new events directly: `down` and `up`. This means you can listen for an event from a `Key` you've created, i.e.: `yourKey.on('up', handler)`.

The order has also now changed. If it exists, the `Key` object will dispatch its `down` event first. Then the `Keyboard Plugin` will dispatch `keydown_CODE` and finally the least specific of them all, `keydown` will be dispatched.

You also now have the ability to cancel this at any stage either on a local or global level. All events handlers are sent an event object which you can call `event.stopImmediatePropagation()` on. This will immediately stop any further listeners from being invoked in the current `Scene`. Therefore, if you call `stopImmediatePropagation()` in the `Key.on` handler, then the `Keyboard Plugin` will not emit either the `keydown_CODE` or `keydown` global events. You can also call `stopImmediatePropagation()` during the `keydown_CODE` handler, to stop it reaching the global `keydown` handler. As `keydown` is last, calling it there has no effect.

There is also the `stopPropagation()` function. This works in the same way as `stopImmediatePropagation` but instead of being local, it works across all of the `Scenes` in your game. For example, if you had 3 active `Scenes` (A, B and C, with A at the top of the `Scene` list), all listening for the same key, calling `stopPropagation()` in `Scene A` would stop the event from reaching any handlers in `Scenes B` or `C`. Remember that events flow down the `Scene` list from top to bottom. So, the top-most rendering `Scene` in the `Scene` list has priority over any `Scene` below it.

All the above also works for `keyup` events.

New in 3.16 is the ability to receive a global `keydown` or `keyup` event from any key on the keyboard. Previously, it would only emit the event if it came from one of the keys listed in the `KeyCodes` file. Now, those global events will fire for any key, regardless of location.

Keyboard Captures

Key capturing is the way in which you stop a keyboard DOM event from activating anything else in the browser by calling `preventDefault` on it. For example, in tall web pages, pressing the `SPACE BAR` causes the page to scroll down. Obviously, if this is also the fire or jump button in your game, you don't want this to happen. So the key needs to be 'captured' to prevent it. Equally, you may wish to also capture the arrow keys, for similar reasons. Key capturing is

done on a global level. If you set-up the capture of a key in one Scene, it will be captured globally across the whole game.

In 3.16 you now do this using the new `KeyboardPlugin.addCapture`` method. This takes keycodes as its argument. You can either pass in a single key code (i.e. 32 for the Space Bar), an array of key codes, or a comma-delimited string - in which case the string is parsed and each code it can work out is captured.

To remove a capture you can use the `KeyboardPlugin.removeCapture`` method, which takes the same style arguments as adding captures. To clear all captures call `KeyboardPlugin.clearCaptures``. Again, remember that these actions are global.

You can also temporarily enable and disable capturing using `KeyboardPlugin.enableGlobalCapture`` and `KeyboardPlugin.disableGlobalCapture``. This means if you set-up a bunch of key captures, but then need to disable them all for a while (perhaps you swap focus to a DOM text field), you can call `disableGlobalCapture`` to do this, and when finished in the DOM you can enable captures again with `enableGlobalCapture``, without having to clear and re-create them all.

Default captures can be defined in the Game Config in the `input.keyboard.captures`` object. The captures are actually stored in the `KeyboardManager`` class. The `KeyboardPlugin`` is just a proxy to methods in the Keyboard Manager, but is how you should interface with it.

There are a bunch of other smaller changes to the Key classes to make them easier to use. Please check out the full Change Log for details if you're curious, or wait for the 3.16 release.



BlendModes.ERASE

I added this feature a while ago, but haven't discussed it in a Dev Log yet. It's a new blend mode called ERASE which, as its name implies, allows you to erase something from the screen. This is mostly useful when combined with Render Textures or Canvas Textures. It allows you to literally 'eat away' at a texture without using any kind of expensive mask, and it works the same in WebGL as it does in Canvas.

Click around (or 'paint') in the following example to see the effect in action:



Render Textures, specifically, have a new 'erase' method that allows you to do this quickly and easily:

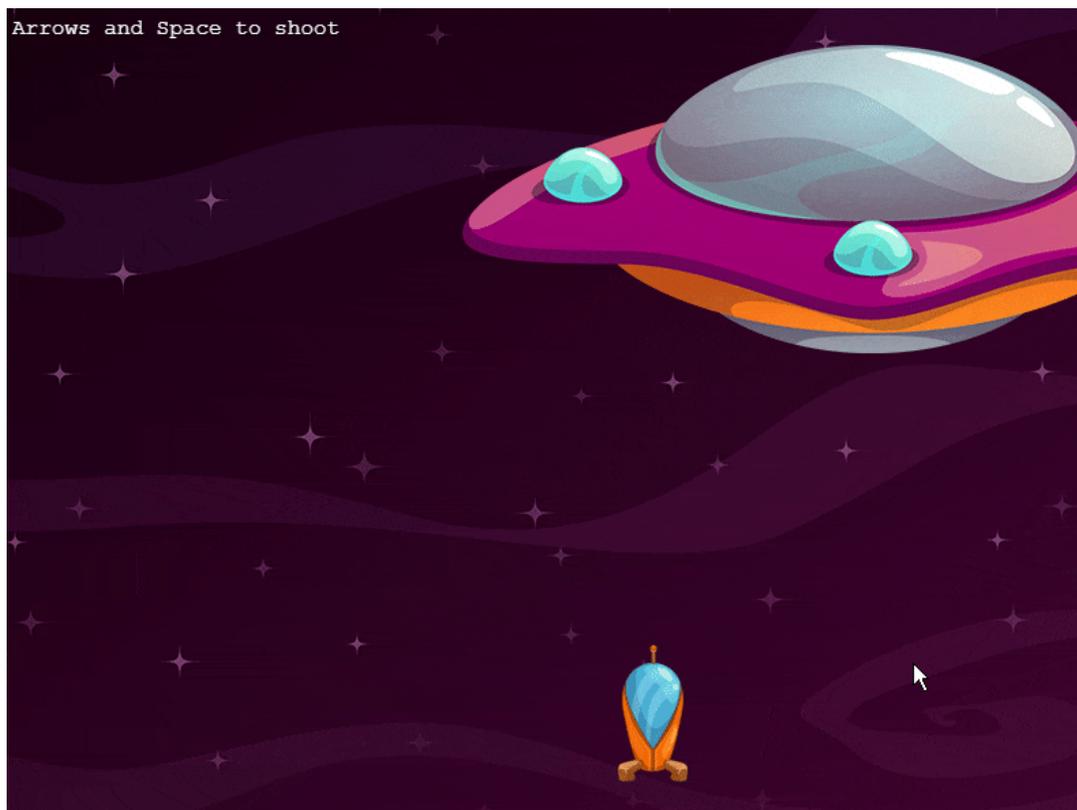
```
var brush = this.make.image({ key: 'brush' }, false);  
this.input.on('pointermove', function (pointer) {  
    if (pointer.isDown)  
    {  
        rt.erase(brush, pointer.x, pointer.y);  
    }  
}, this);
```

It can accept a texture based Game Object, like the Image in the code above, or any of the Shape objects (like a circle or polygon), or indeed anything else a Render Texture can draw to itself. It just uses an erase blend mode internally, which causes that object to eat-away at whatever was on the Render Texture previously.

This is particularly useful for certain types of special effects. For example, you could have a Render Texture filled with a solid color and then wipe it away using a brush and tween sequence to create really nifty reveal transitions. Or it could

form a part of a game element, such as blowing away parts of a texture based object.

Following a similar thread I've also made a number of updates to the Canvas Texture class. First, is the addition of the 'drawFrame' method. This allows you to draw a texture frame, from any loaded atlas or sprite sheet, directly to the Canvas Texture. To go with this, I also added 'getPixels', which lets you grab a region of the Canvas Texture and return all of the pixels within it. Combined with the new 'setPixel' method it means you can now quickly modify the contents of the texture using direct methods. Here's an animated gif showing a player literally blowing pixels away out of a ufo:



This is now trivial to achieve and can be put to great use in a matter of ways. The above gif is from a demo that will be part of the Phaser Backers Examples package in February. In December I released the first batch of [Backers Examples](#), with another pack due on January 4th. Eventually, these examples will be released publically, too. For now, though, they're a perk for those who go above and beyond and genuinely help Phaser to survive.

Getting Animated

In 3.16 I also made a number of subtle, but powerful, changes to the way

animations are handled.

When creating an animation it will no longer raise a console warning if the animation key is already in use. Instead, it will return the animation belonging to that key. A brand new animation will only be created if the key isn't already in use. When this happens, the ``add`` event is emitted by the Animation Manager. If no event is emitted, the animation already existed. You can also use the new `AnimationManager.exists` method to check to see if an animation is already present using the given key.

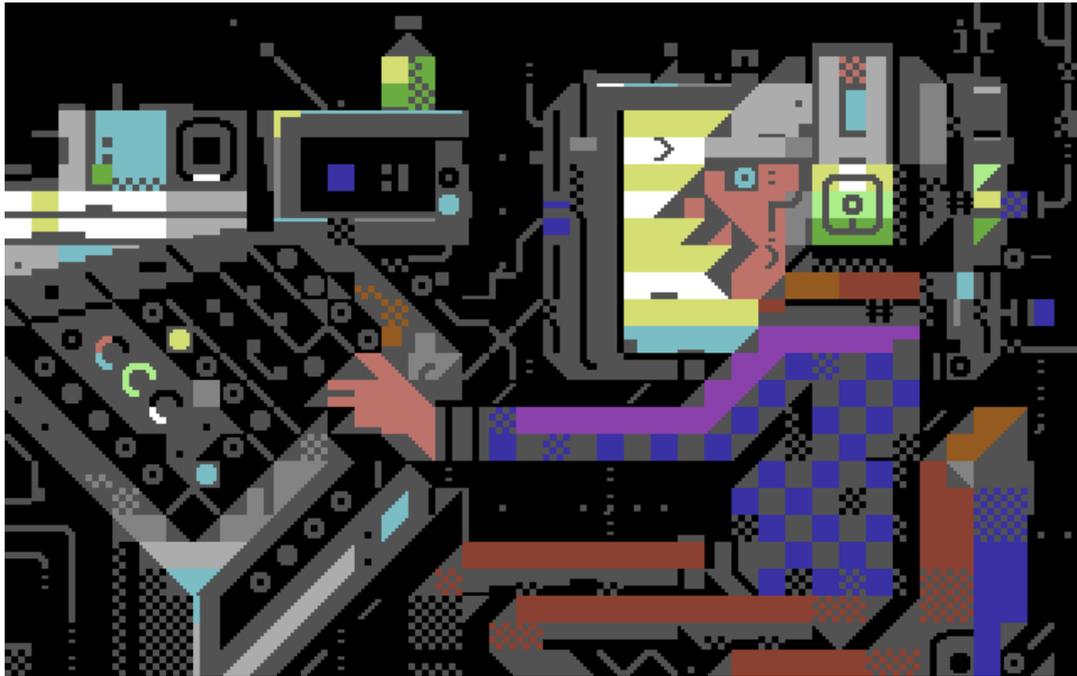
More significantly, the Animation class now extends the Event Emitter and dispatches events itself. This allows you to listen for events from a specific Animation, rather than via a Game Object. This is handy, for example, if you had an explosion animation that you wanted to trigger a sound effect when it started. You can set this up on the animation directly, rather than a Sprite playing it, which makes more logical sense.

Animations will emit the following events: `'start'`, when it is played by any Game Object, either forwards or in reverse. `'restart'` when any Game Object restarts it and `'complete'` when it finishes playing on any Game Object.

Not to be outdone, Game Objects now emit some new events, too. They will emit events that carry the name of the animation within them. For example, if you had an animation called `'explode'` then a Game Object will now emit `'animationstart-explode'` when it starts playing, or `'animationcomplete-explode'` when it finishes. This allows you to listen for animation events on a Game Object level, rather than global one, right down to a specific animation instance.

Finally the last new feature is called `'chain'`. This allows you to line-up another animation to start playing as soon as the current one stops, no matter how it stops (either by reaching its natural end, or directly by having `stop` called on it). You can chain a new animation at any point, including before the current one starts playing, during it, or when it ends (via its ``animationcomplete`` callback). Chained animations are specific to a Game Object, meaning different Game Objects can have different chained animations without impacting the global animation they're playing.

All in all, I feel like animations are a lot more flexible now, at least in the way in which you can deal with their events. You can choose to handle things on a global, local or instance level, whichever your coding style prefers.



3.16 in the New Year

I'm a bit frustrated that I didn't get to release 3.16 this month. It's extremely close to being finished. This week I completed the new Size component, which is very much inspired by the Aspect Ratio Fitter from Unity. It allows you to define a width and height, lock the aspect ratio between them and then scale it based on a variety of modes - such as 'fit', 'envelope', 'width takes priority' and more. In short, it's really useful for handling the scaling of objects, which is why it's a class in its own right. It's also now the core of the new Scale Manager and has allowed me to remove a whole bunch of duplicate code and properties.

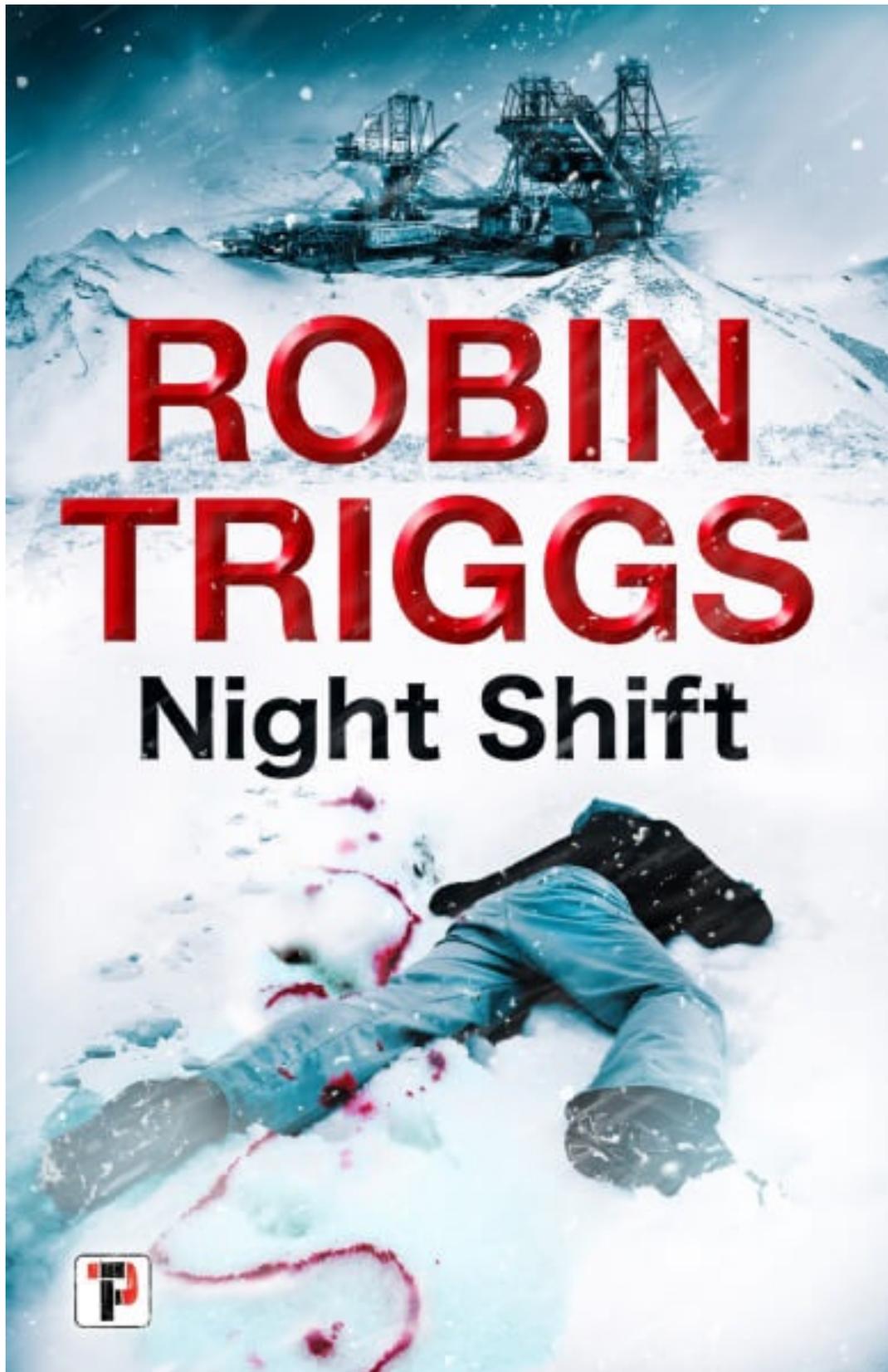
Finishing the Scale Manager is my absolute priority when I return in January and then 3.16 will be released. I'll do my best to make sure that future updates are faster, although that should be a lot easier because future updates aren't likely to be as far reaching and involved as the Scale Manager was! When you try inserting an octopus into the tank, it's hard not to get its tentacles everywhere, and that's very much what it feels like the Scale Manager was. Still, I'm very happy with where I am, and don't believe there is all that much work left to be done before I can release.

Beyond that, there's plenty more to get on with. The GitHub issues list has backed-up a bit, so could do with a clear down. The Examples repo is being worked on too, and is having a make-over. You can track the changes in its new branch, but suffice to say, when we merge it there will no longer be any examples that don't run. They'll all have proper names and be in more sensible folders where required. This is a piece of work I've got someone helping with as it just really needed doing, so I diverted some funds to pay for this as I knew I didn't

have time myself. When complete we can release the examples onto the Phaser web site and labs can return to being what its name implies: the Phaser labs, specifically for experiments and tests.

That just leaves me to wish everyone a very Merry Christmas and Happy New Year. I won't be around that much over the next 10 days as I'll be playing with the kids and enjoying some time to revitalize my grey cells. Personally, I'll be digging in to Just Cause 4 (I absolutely love that series), and Belen convinced me to try Assassins Creed: Odyssey, too :) A little gaming, lots of family time and back in the office before you know it, raring to finish off 3.16. Take care everyone!





"Antarctica. A mining base at the edge of the world. Anders Nordvelt, last-minute replacement as head of security, has no time to integrate himself into the crew before an act of sabotage threatens the project. He must untangle a complex web of relationships from his position as prime suspect. Then a body is found in the

ice." - This is the premise of [Night Shift](#), published by Flame Tree Press and written by Robin Triggs. I mention it because he's my brother-in-law and it's a really cracking read :)

"Kick, punch, it's all in the mind!" [PSone.css](#) is a Playstation 1 style CSS Framework.

[How a phone glitch sparked a teenage riot](#). This is a fantastic article about kids circumventing the 1980s Swedish phone system and the long-running implications of that.

Phaser Releases

Phaser 3.15.1 released October 16th 2018.

Phaser CE 2.11.1 released October 2nd 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.



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

[Preferences](#)

[Forward](#)

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