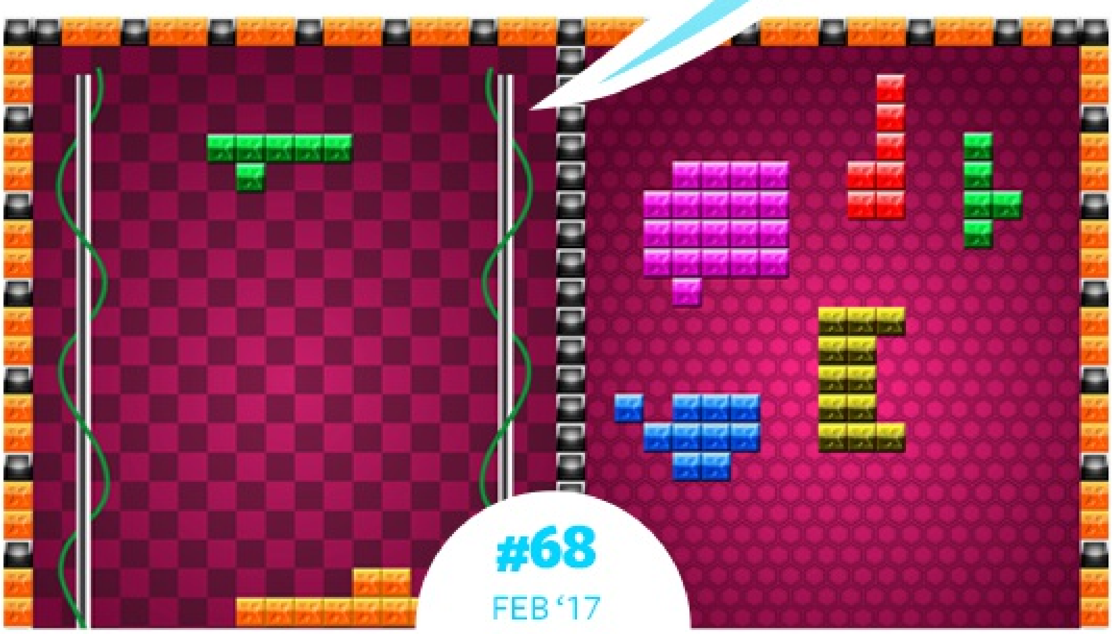
PHASER WORLD



#68

FEB '17

INSIDE

Welcome to Issue 68 of Phaser World

Over the years I've seen Phaser used for some interesting, and down right amazing things. From Phaser games in an Iron Maiden video, to embedded in a robot walking around a trade show. But this week I'm featuring two games that helped the developers with their marriage proposals. Quite fitting for the week in which we had valentines day, and another one to chalk up on my 'awesome' list :)

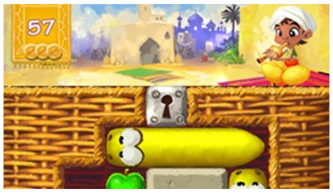
Romance aside we've some great puzzle games this week, a bunch of really neat plugins, and another Phaser 3 dev report. Looking at my calendar I can see we'll hit issue 70 in the first week of March. That feels like quite an achievement, so thank you to everyone who reads, creates things, and tells me about it!

Until the next issue, keep on coding. Drop me a line if you've got any news you'd like featured (you can just reply to this email) or grab me on the Phaser Slack or Discord channels.

Games made with Phaser

Retris

Game of the Week
An interesting and highly polished 'reverse' Tetris puzzle game.

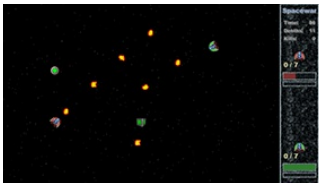


Snake Slider

Staff Pick
A fun, unique, sliding puzzle game. Get the green snake to the exit in as few moves as possible.

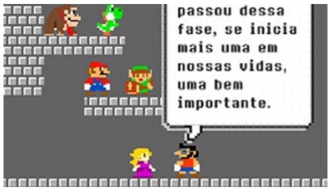
Tahoe Po

See who can eat the most Tofu in the world! But avoid the flying spices!



Spacewar

A modern version of the classic gravity, thrust and blast shooter.


passou dessa fase, se inicia mais uma em nossas vidas, uma bem importante.

Marriage Proposals with Phaser

It's not every day you ask someone to marry you, but here are two proposals delivered with games made in Phaser.

Phaser News & Tutorials

```
game.debug.timerEvents(null, x, y)
game.time.events (4)

show picture1
show picture2
show picture3

game.debug.timerNextEvent(null, x, y)
game.time.events (4)

game.debug.timerElapsed(null, x, y)
game.time.events: 3s
```

Debug Timer Extension

A new plugin to allow you to visually debug timers and timer events.

Blur Launch Jam

Release your game in the Blur Messenger app and you could win prizes and cash.

Phaser Starter Project

A Phaser Starter Project with NPM, Webpack, TypeScript and more!

Grid Physics Plugin

A new plugin to help with grid / tile based movement and physics, with loads of cool features.

Camera Features Tutorial

A tutorial from Envato Tuts all about the new Camera Features in Phaser.

RPG Game Development Course

This brand new video course will teach you everything you need to know about creating an RPG game, now includes the Phaser 101 modules.

Patreon Updates

Thank you and welcome to the following awesome people who joined the Phaser Patron this week: **Chris Andrew** and **Sergio Bonavida Ponce**.

Patreon is a way to donate money towards the Phaser project on a monthly basis. This money is used entirely to fund development costs. You can also make one-off donations via PayPal. Donations receive discounts, forum badges, private technical support, and the eternal gratitude of the Phaser team :)

Development Progress

As usual we're back with another Phaser 3 update. It's a bit slimmer this week (as far as new features are concerned) because Felipe had the week off, and I had a few days off too, but progress was still made, there's a new demo for you to see, and some thoughts on how we're going to handle the input manager.

- There is a global game Registry. This is accessible from any State, and allows you to store common data in a single location, and then save that data out to a file system if needed.

- States can now pass data from one to another easily. The data will be passed in to either the States init or create functions (or both if you've both set). This allows you to do things like have a 'modal pop-up' State, where you pass in the coordinates and other game data to the modal, which it acts upon and then closes down again when done.
- I spent a good amount of time creating the new Color object, and associated functions. Available in the new Phaser.Graphics namespace, it's a way to easily create a unified representation of a color, for use anywhere within Phaser. It can parse most common formats (i.e. hex strings, css values, etc) and creates a single small object that can be used by functions such as 'backgroundColor', text color, and so on.
- Color functions include all kinds of useful features: from HSV color wheels, to hue component extraction.
- On a related note I started work on a new Gradient class, to handle easy creation of gradients and color ramps, with an emphasis on gradients that can change over time (such as a sky effect changing from blue skies to a sunset). Gradient objects will be able to be passed into Graphics objects, Text objects, and anywhere else that makes sense to support them.

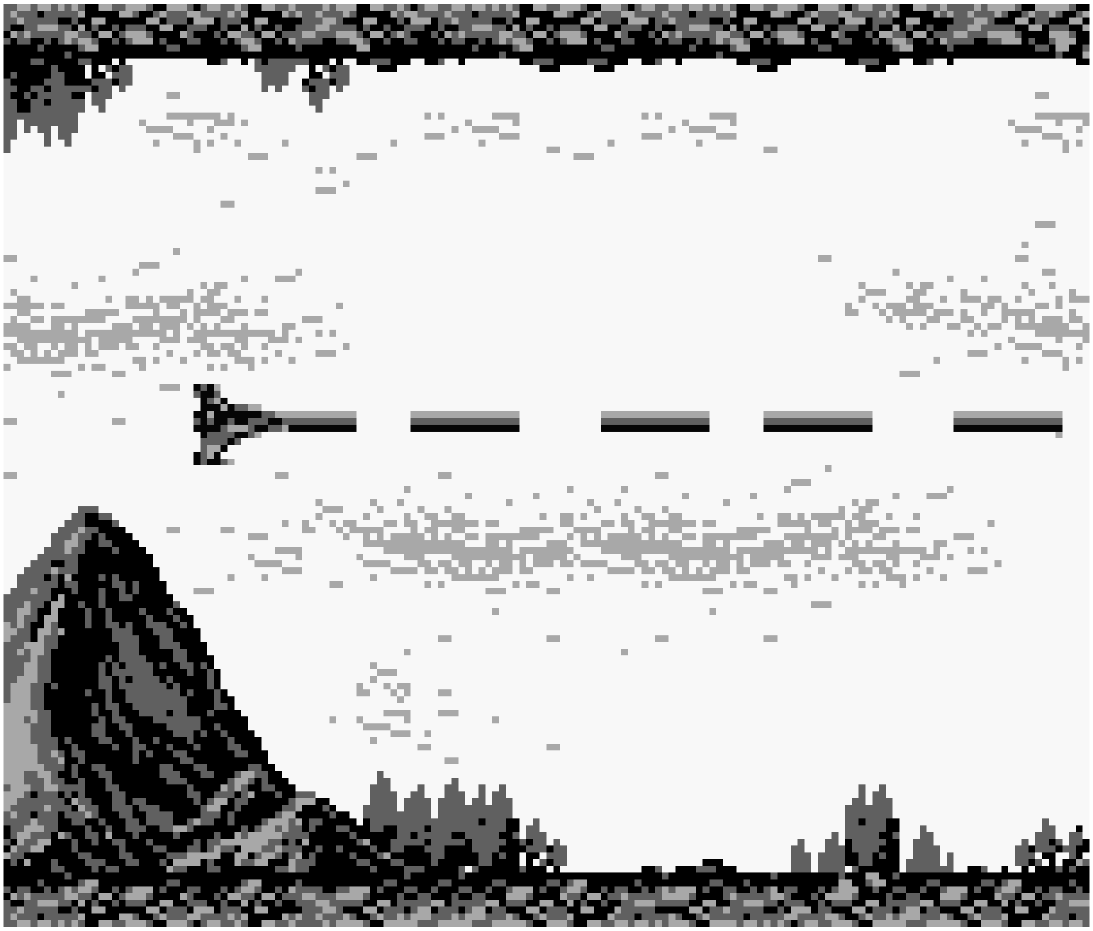
Input Manager Research

We also made a start on dealing with the Input Manager. Interactive objects are always complicated to deal with in a game engine, because they expose so many potential issues off the back of their events. I.e. click handlers adjusting the camera, changing the state of an object, updating physics, etc. Over the years I've seen everything done in the handler, even causes where the entire game was destroyed. In Phaser 2 there are a lot of conditional checks all over the place, to cater for edge cases like this. Part of our research was in to ways to mitigate this, perhaps using an event queue, so we've direct control over when things happen, rather than assuming they could happen at any point.

Then there is the cost of actually *detecting* those input events. To ensure that when you click, or touch with your finger, that the test happens against perfectly accurate coordinates, we need to fully update the transforms of every object on the display list. With deep scenes this is computationally very expensive and not the sort of thing you want to do on a whim.

If it was confined to just click events it might be acceptable, but what if you want to detect mouse over events as well? Perhaps in order to give your game objects some tactile feedback such as displaying an effect while the mouse is over them. Mouse movement polling is high intensity, happens very frequently and produces a lot of data. Again we need to ensure that those input events always happen against the fully updated display list (otherwise you could start getting false positive responses).

We spent so long getting the game and render loop as tight as we could, that we have to be super careful not to negatively impact it again just by making the objects interactive. To that end we've been researching rendering hit masks to a hidden canvas, using dom elements, batched point conversions and all kinds of ideas. We're not there just yet though, so I'll write about this again next week after lots more testing.



Zoom

I'm going to end this update with a little demo. The screen shot above is showcasing a new Phaser 3 feature: the 'zoom' game setting. When passing in your game config you can set the 'zoom' value to any integer. This will upscale your game from the game size given. Used in combination with the new 'pixelArt' boolean it allows you to easily do things like this demo (cursors to move, space to shoot).

What you're seeing is a 160 x 144 sized game with a zoom setting of x4. All graphics are 'borrowed' from the Gameboy title Nemesis (aka Gradius.) - it works especially well for pixel art games, or games where you want to render them to a smaller canvas, but scale them up for display.

Please note that this isn't the new 'Scale Manager'. That is a different beast entirely. This is just a way to let you create games are small resolutions, then 'zoom' them to a larger resolution, from which the Scale Manager could then take over if required.

That's it for this issue. Next week we're both back full time, all guns blazing, so be prepared for more demos and news!

Phaser 3 Mailing List and Developers Guide

If you're interested in helping evolve the shape of Phaser 3, then please join the Phaser 3 Google Group. Discussions this week have included varying render loops. The group is for anyone who wishes to help shape what the Phaser 3 API and feature-set will contain.

The Phaser 3 Developers Guide is available. Essential reading for anyone who'd like to help *build* Phaser 3.

Geek Links

Project ZX is a really interesting looking game due out soon. You fly and shoot through all kinds of Spectrum games. It looks great (although massively IP infringing too), let's hope it gets a release!

A Year in Stardew Valley is a really fantastic piece of writing over on Rock Paper Shotgun. Mixing elements from the game, with the authors real life, it's a captivating read.

If you're as old as me you may remember the famous software of Kai Krause. He was responsible for classics like Kai's Power Tools, Bryce and Goo. This is a look at the innovative UI his software used. Ahhh, the memories!

Phaser Releases

The current version of Phaser CE is 2.7.3 released on January 9th 2017.

Phaser 3.0.0 is in active development in the GitHub v3 folder.

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.