ALPEN-ADRIA UNIVERSITÄT KLAGENFURT | WIEN GRAZ

Phaser Tutorial for Beginners

gameJAM POWER'D BY ANEXIA

Andreas Leibetseder

Overview

- General
  - Phaser Introduction
  - Quick Demonstration
  - Environment Setup
- Basics
  - Simple Projects and Structuring
  - Game Objects
  - Special Functions
  - Assets: Loading and Manipulation
  - Physics
- Further Aspects, Techniques and Examples
  - Asset Packs
  - Game Scaling
  - Example Games

1 General

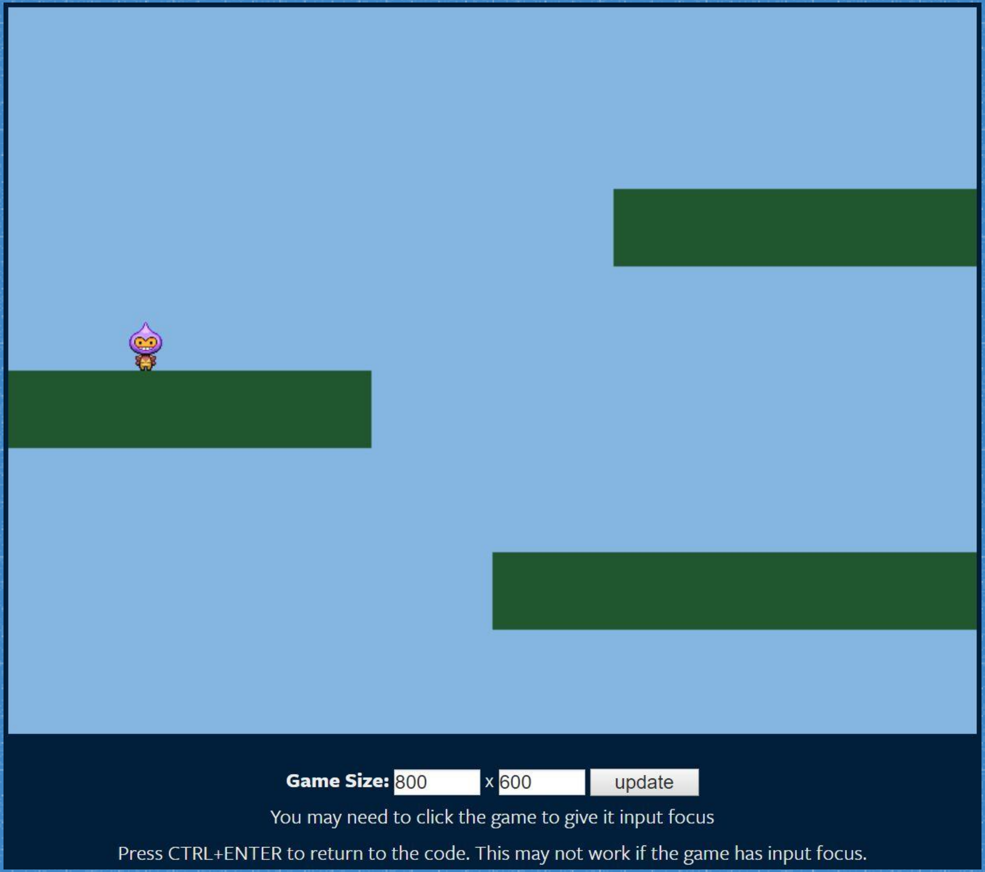
Phaser Introduction and
Environment Setup.

What is Phaser?

"A fast, free and fun open source framework for Canvas and WebGL powered browser games."

- Simple
  - HTML5 Game Engine
  - JavaScript-based
- Versatile
  - Create: Any Text Editor
  - Deploy: Common Web Server
  - Access: Capable Browser

Quick Demonstration


Game Size: 800 x 600 update

You may need to click the game to give it input focus

Press CTRL+ENTER to return to the code. This may not work if the game has input focus.

https://phaser.io/sandbox/edit/3

Getting Started: Prerequisites

Web Server

XAMPP, Apache2, MAMP, Mongoose Binary, Z-WAMP, …

IDE/Editor

Atom, Brackets, Sublime Text, IntelliJ IDEA, …

Tutorial Environment: XAMPP + Atom

1. Download & install XAMPP
   https://www.apachefriends.org
2. Download & install Atom
   https://atom.io
3. Clone github repo into htdocs
   ```
   git clone https://github.com/amplejoe/PhaserTutorial.git
   ```
   (e.g. on Windows C:\xampp\htdocs)
4. [Optional] Atom Phaser auto completion
   follow guide: https://tinyurl.com/mj3z6de
5. Launch tutorial overview page
   http://localhost/PhaserTutorial

Useful Links

- Learning Tutorials
  https://phaser.io/learn
- Categorized Examples
  https://phaser.io/examples
- Sandbox for online coding
  https://phaser.io/sandbox
- API (2.6.2)
  https://phaser.io/docs/2.6.2
  Alternative lookup (2.4.7 only):
  http://phaserchains.boniatillo.com
- Free spline/path editor
  https://phaser.io/waveforms

2 Basics

Create simple games
using State Transitions,
Physics, custom Assets,
Sounds and Animations.

Tutorial 00 - Hello Phaser

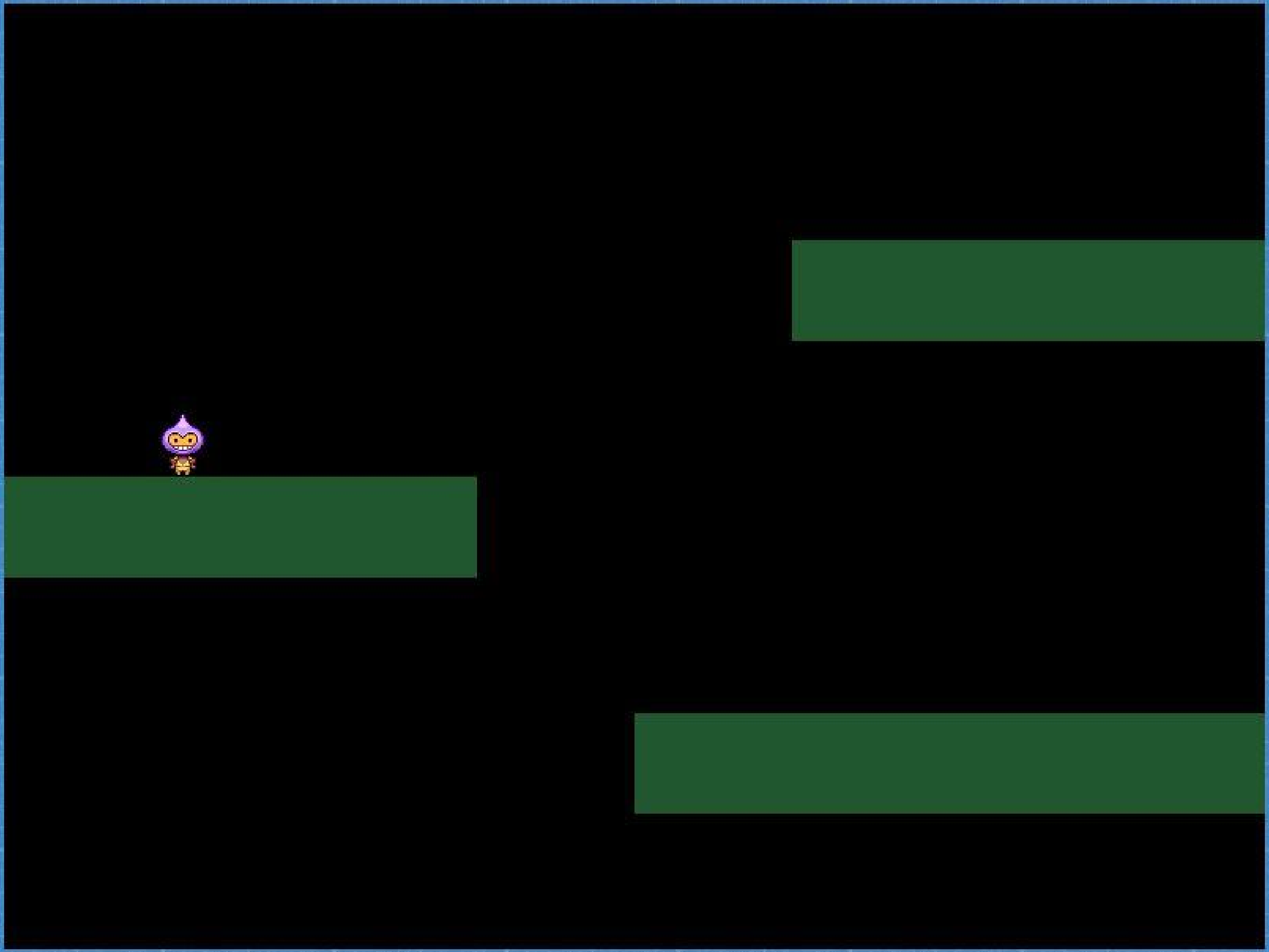
00_Hello_Phaser/index.html

```html
<!doctype html>
<html>
    <head>
        <meta charset="UTF-8" /> <title>Hello Phaser!</title>
        <script src="../resources/lib/phaser_2.6.2/phaser.min.js"></script>
    </head>
    <body>
        <script type="text/javascript" src="js/app.js"></script>
    </body>
</html>
```

00_Hello_Phaser/js/app.js

```javascript
window.onload = function() {
    let game = new Phaser.Game(800, 600, Phaser.AUTO, '', { preload: preload, create: create });
    function preload () {
        game.load.image('logo', 'assets/sprites/phaser.png');
    }
    function create () {
        let logo = game.add.sprite(game.world.centerX, game.world.centerY, 'logo');
        logo.anchor.setTo(0.5, 0.5);
    }
};
```

Tutorial 01 - Basic Game



Tutorial 02 - Game Structure

- Phaser States
  - Split game into smaller chunks

    no limit to # states, but **exactly 1 active at any time**!
  - Typical Structuring - e.g.:

    by screen: preloading, title, game, …

    by game behavior: level, items shop, boss fight, …
  - Why?

    keeps code understandable

    simplifies collaboration
- States for Tutorial
  - Boot: phaser settings, preloads loading bar
  - Load: loads all assets, displays loading bar
  - Title: game title screen
  - Game: actual game

Tutorial 03 - Game Objects

- Objects with different properties:

  `Phaser.Image` light-weight graphics, no physics/animation
  `Phaser.Sprite` nearly everything visual, physics/animation attachable
  `Phaser.Graphics` draw primitives like Rectangles, Circles or Polygons
  `Phaser.Text` displayed Text, can only display pre-loaded fonts
  `Phaser.Sound` audio with controls like volume, loop, …
  `Phaser.Button` special sprite handling pointer events
  `Phaser.Tween` alter obj. properties over time: motion, scale, alpha …
  `Phaser.Group` object grouping, useful for batch transformations/z-order/…
  …
- Object Creation (see Tutorial)
  - Simple

    `GameObjectCreator` create object (`Phaser.Game.make`)
    `GameObjectFactory` create and add to it the world (`Phaser.Game.add`)
  - Advanced

    Extend Phaser objects (`Object.create`)

Special Phaser Functions (tinyurl.com/n3zvnys)

Preparation

`preload` (called first)
load assets here

`loadUpdate` (repeatedly called ≙ `update`)
update e.g. custom progress bar

`loadRender`(called after `loadUpdate`)
render specific code (usually not needed)

`create`(called after `preload` finished)
setup code: create sprites/particles/etc.

`update`(called every frame, after Preparation)
game logic: move objects/handle collisions

`render` (called after WebGL/canvas render)
use for post-render effects/debug overlays

`resize` (called if game in RESIZE scale mode)
used to change scale, 2 params: width, height

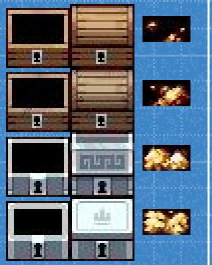
`shutdown` (called if state is shutdown)
for code between switching states

Game

- Reserved functions Phaser calls at specific times
- Should **NOT** be overwritten
- Use them where needed: states, objects, …
- **NO** need creating empty stumps!
- Most common: `preload`, `create`, `update`

Tutorial 04 - Assets: Sprites

- 2D image components of a scene
- Different ways of importing Sprites
  - Plain single image
  - Sprite sheets
    - without map file: fixed dimension sub-sprites
    - with map file: XML, JSON Hash/Array, …
    - Why? → tinyurl.com/krp4wlu



© tinyurl.com/lj8dka4

- Tools for Atlas / Sprite Sheet Creation (see extras/tools folder)
  - Leshy SpriteSheet Tool (free) tinyurl.com/z9z5psk
  - ShoeBox (free) renderhjs.net/shoebox
  - Sprite Sheet Packer (free) tinyurl.com/ohbdt48
  - TexturePacker (1mo. trial) tinyurl.com/obqszql
  - Phaser Editor (40$) tinyurl.com/k2unwe6

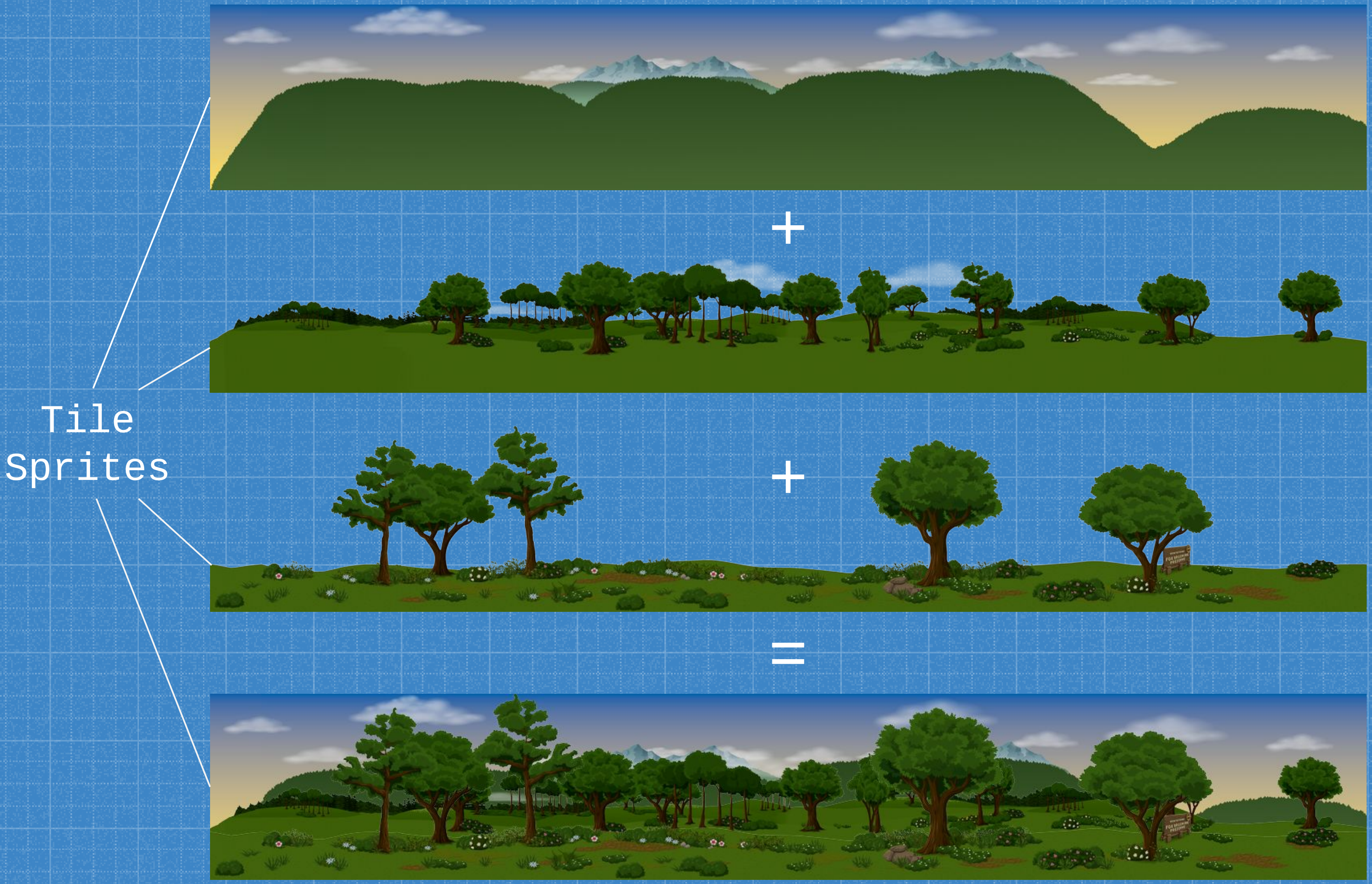
Tutorial 05 - Assets: Tilemaps

- Specific arrangement of sprite sheet tiles on one or more layers
- Described via JSON / CSV file
- Useful for designing levels
- Tilemap Editor
  Tiled (free)
  http://www.mapeditor.org

Tutorial 06 - Assets: Tilesprites

- Sprites with repeating texture
- Texture scrollable and scalable, independently of Tilesprite
- Textures wrap automatically
- Purpose
  Seamless texture game backdrops
- Similar to plain sprites - can be transformed, animated, tinted, …

Tutorial 06 - Parallax Scrolling



Tutorial 07 - Assets: Text

- Textures, rendered from local hidden canvas object
  - Standard System Fonts already loaded
  - Custom Fonts need to be preloaded!

- Different ways of adding Text
  - System Fonts (standard)
  - Downloaded Fonts (loaded via CSS)
  - Web Fonts (loading script, req. internet)
  - Bitmap Fonts (spritesheets and XML map)

Tutorial 08 - Assets: Audio

- Playability depends on browser
  - Safest bets: mp3, ogg, wav, m4a
  - Wait for decoding on encoded files
- Importing Audio
  - Single Files
  - Audio Sprites: single files containing multiple tunes/sfx, sections defined by markers (manually set or JSON Hashes conforming to tinyurl.com/nhsh3lt)
- Playback functions
  - `play`, `stop`, `pause`, `loop`, `fading`, …
  - Plugins: ProTracker, .ym support (Atari)

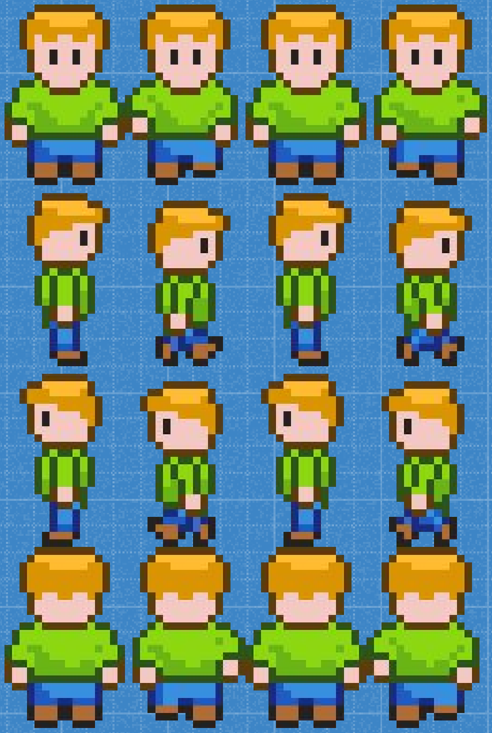
Input

- Support: Mouse, Keyboard, Touch and Gamepad
- Mouse
  Drag tinyurl.com/mne2zoo, Drop Lock tinyurl.com/kpwg44q
  Click tinyurl.com/mml73h3
  Z-Ordering tinyurl.com/lgeage4
  Virtual Buttons tinyurl.com/nyfj8tw
- Keyboard
  Cursor keys tinyurl.com/ltfzsnp
  single keys tinyurl.com/kkja8rn, debug tinyurl.com/kgv3sch
  Enabled across states: `Phaser.game.input.resetLocked = true;`
- Touch
  Tap tinyurl.com/lvlh8ys
- Gamepad (needs browser support)
  tinyurl.com/kqnlyqa, tinyurl.com/kwsgkgr, tinyurl.com/nyx5fmt
- Plugins
  Skinnable Virtual Joystick (16$) tinyurl.com/lj9mnyx

Tutorial 09 - Animation

- Created using sprite sheets
  - Single contained frame sequences form continuous animation

- Several possibilities
  - Single animation sheet: use every frame
  - Multiple animation sheet: define animations in code
  - Dynamically create (simple) animation in code



tinyurl.com/kyen5ez

Tutorial 10 - Tweens

- Object property manipulations
  - Time-based: e.g. change alpha within 2s
  - Easing Functions
    http://phaser.io/docs/2.6.2/Phaser.Easing.html
  - Set up and start movement/effects without needing to handle them in `update`:
    ```
    game.add.tween(object).to (properties, duration, ease,
        autoStart, delay, repeat, yoyo)
    ```
  - Chainable tinyurl.com/lequnkp
- More advanced tween effects
  - Atari Intro tinyurl.com/kgo5cqo
  - Sine Wave tinyurl.com/l9d72zp

Tutorial 11 - Physics

- Build-in Engines for simulating object dynamics like gravity, collisions, …
- Different Systems (tinyurl.com/l8cedo3)
  - Arcade: simple bounding boxes (AABB), fast
  - P2: polygon body shapes, HW intensive
  - Ninja: AABB + simple shapes: triangles, circles etc., good compromise
  - Plugin: Box2D (32$) tinyurl.com/jpsrnwt
- Enable physics
  `game.physics.startSystem(Phaser.Physics.[ARCADE|P2JS|NINJA])`
- Enable sprite bodies (none by default)
  `game.physics.[arcade|p2|ninja].enable`
- **Several** Systems can be active, yet only **one** can be enabled for an object’s body

Tutorial 12 - Camera

- View into the game world
  - Properties: position, size
  - Only renders objects **in its view**, unless: `object.autoCull = false;`
- Freely movable
  - `Phaser.Camera.[x|y]`
- Stick objects to camera
  - Score, HUD, …
- Follow objects
  - Modes: Platformer, Top-Down, … (tinyurl.com/l6srfm8)
  - Momentum: smooth following (tinyurl.com/n5ut5le)
  - Dead zone: disable following within area (tinyurl.com/ket7waf)
- Camera Effects
  - flash, shake, fade, …

3 Further Aspects, Techniques and Examples

Asset Organization,
Game Presentation and
Example Games.

Asset Packs - Description

- Reference all assets in **ONE** file
  - Greatly facilitates workflow
  - JSON format
  - Specify type, key, URL & attributes
  - Further structuring: level1, level2, …
  - Single line in-game loading
    `game.load.pack(key, file_url, json_data, callbackContext);`
    (can either use `file_url` OR `json_data` object)
- Asset Packer Tools
  - Grunt Task (Free)
    tinyurl.com/kwf6blm
  - Phaser Editor (30$) - Asset Pack File
    tinyurl.com/k949f5y

Asset Packs - Example

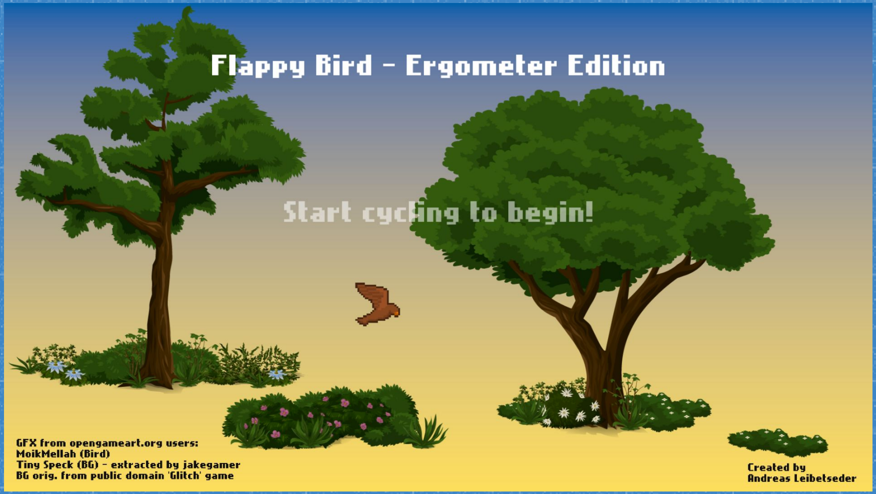
```
{
"level1": [
    {
        "type": "image",
        "key": "image_1",
        "url": "assets/pics/image_01.png",
        "overwrite": false
    },
    {
        "type": "audio",
        "key": "audio_1",
        "url": "assets/pics/audio_01.mp3",
        "autoDecode": true
    },
    ...
  ],
  "level2": [
      {
          "type": "spritesheet",
          "key": "character",
          "url": "assets/pics/character_32x32.png",
          "frameWidth": 32,
          "frameHeight": 32,
          "frameMax": 24,
          "margin": 0,
          "spacing": 0
      }
      ...
    ],
    "meta": {
    "generated": "1401380327373",
    "app": "Phaser Asset Packer",
    "url": "http://phaser.io",
    "version": "1.0",
    "copyright": "Photon Storm Ltd. 2014"
    }
}
```

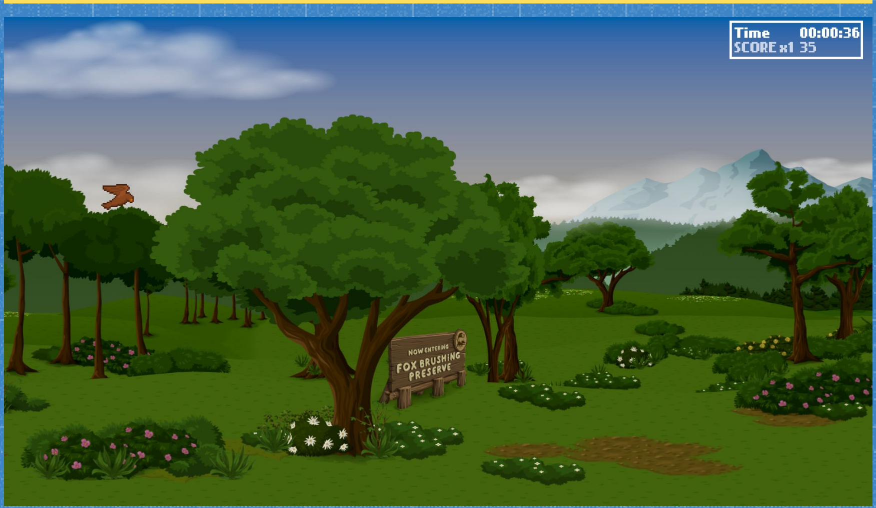
cf. https://phaser.io/examples/v2/loader/asset-pack

Tutorial 13 - Game Scaling

- Spend time on making your game presentable!
- Use `Phaser.ScaleManager` to adjust resolution/aspect/layout to your needs
- Adjustments could also be tied to JavaScript’s window resize event (`window.onresize`)

Example Games - Flappy Cycling

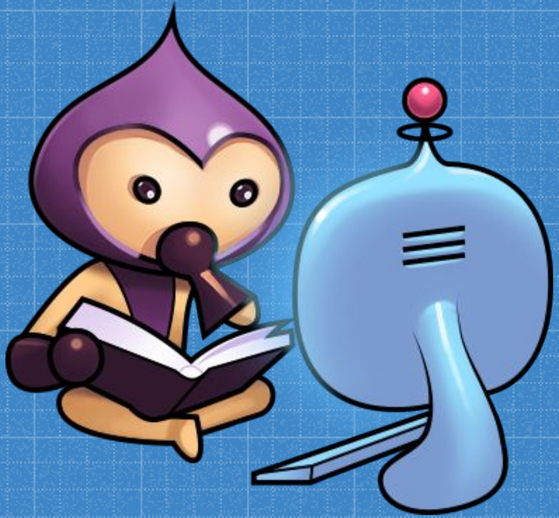

Flappy Bird - Ergometer Edition
Start cycling to begin!
GFX from opengameart.org users:
MoikMellah (Bird)
Tiny Speck (BG) - extracted by jakegamer
BG orig. from public domain 'Glitch' game
Created by
Andreas Leibetseder



https://github.com/amplejoe/FlappyCycling

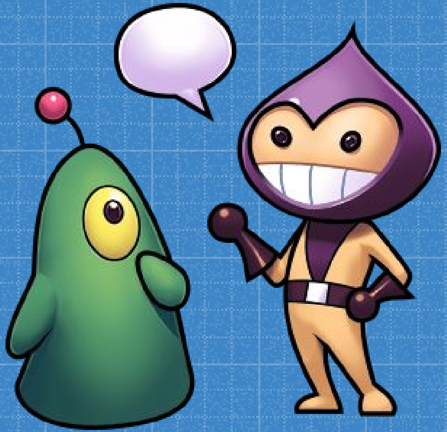
Example Games - Don’t Bug Me!

http://tantriccycle.com/dontbugme



Happy Jamming!

ANY QUESTIONS?



Contact:

aleibets@itec.aau.at