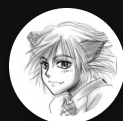
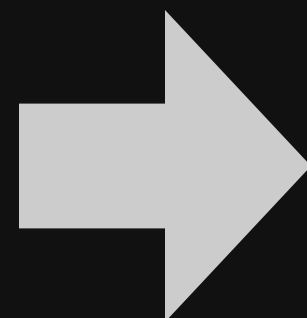


Waaaaaasm?

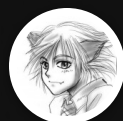
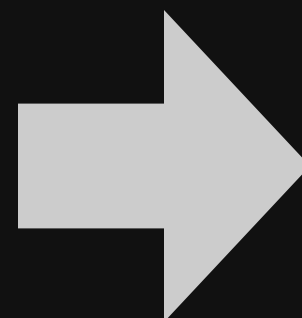
Валерий Кузьмин, СКБ Контур

FrontTalks, сентябрь 2017





Kontur.Recognition











WEBASSEMBLY



Как?

Бизнес-логика на JS

OMG, тут все тормозит!

Бизнес-логика на JS



Как?

Бизнес-логика на JS

Шустрый код на C

Бизнес-логика на JS



WebAssembly - это...

- Развиваемая W3C спецификация **платформы**
- Работающая рядом с JS "виртуальная машина"
- Текстовый и **бинарный** формат
- Цель компиляции для C/C++/Rust

"asm.js на стероидах"



WebAssembly - это не...

- Не замена Javascript
- Не язык сам по себе
- Не цель компиляции для TypeScript
- Не цель компиляции для Java, C# *

* Вообще неправда, уже есть компиляторы



Применение WASM

Обработка видео

Математика, переборы,
сортировки

Рендеринг PDF

Криптография

Работа с
изображениями

Браузер в браузере

Machine Learning

Использование готовых игровых
движков

Работа с архивами

Базы данных на клиенте

Сериализация и
десериализация xml, json

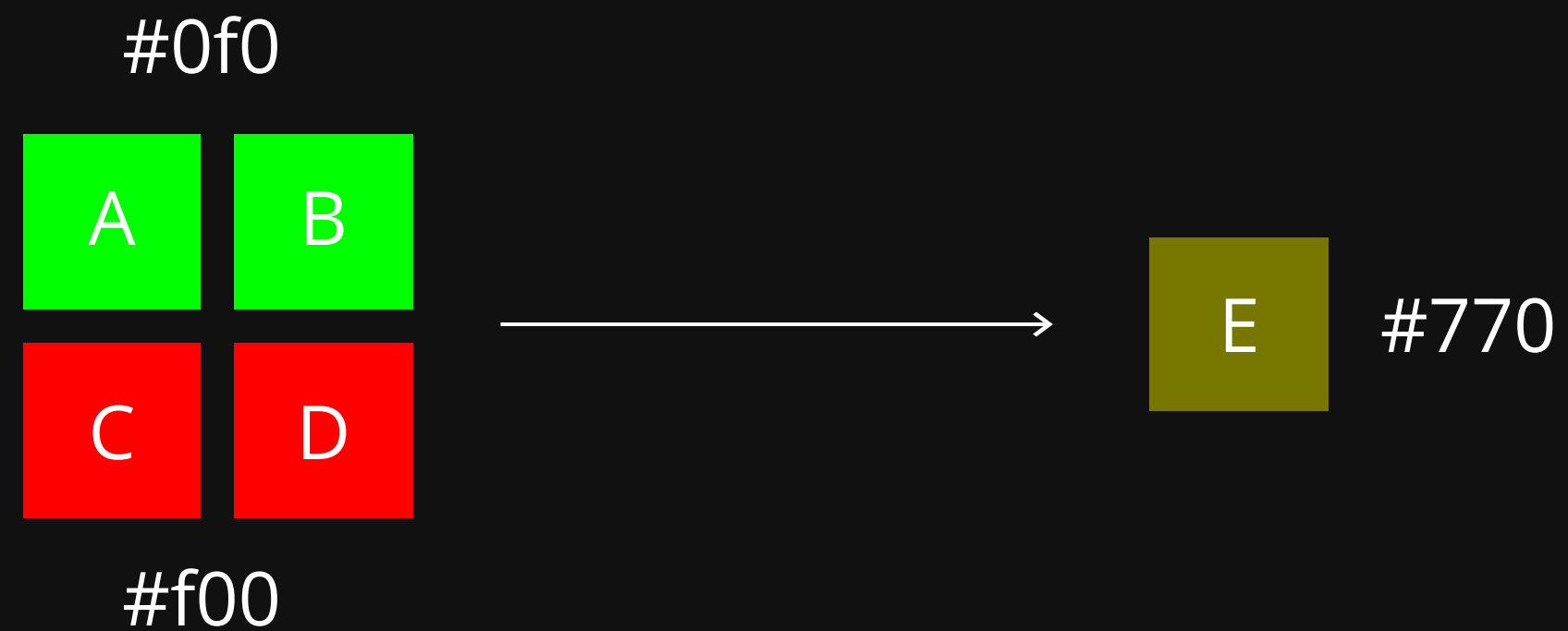
Запуск других языков



Downsampling JPEG

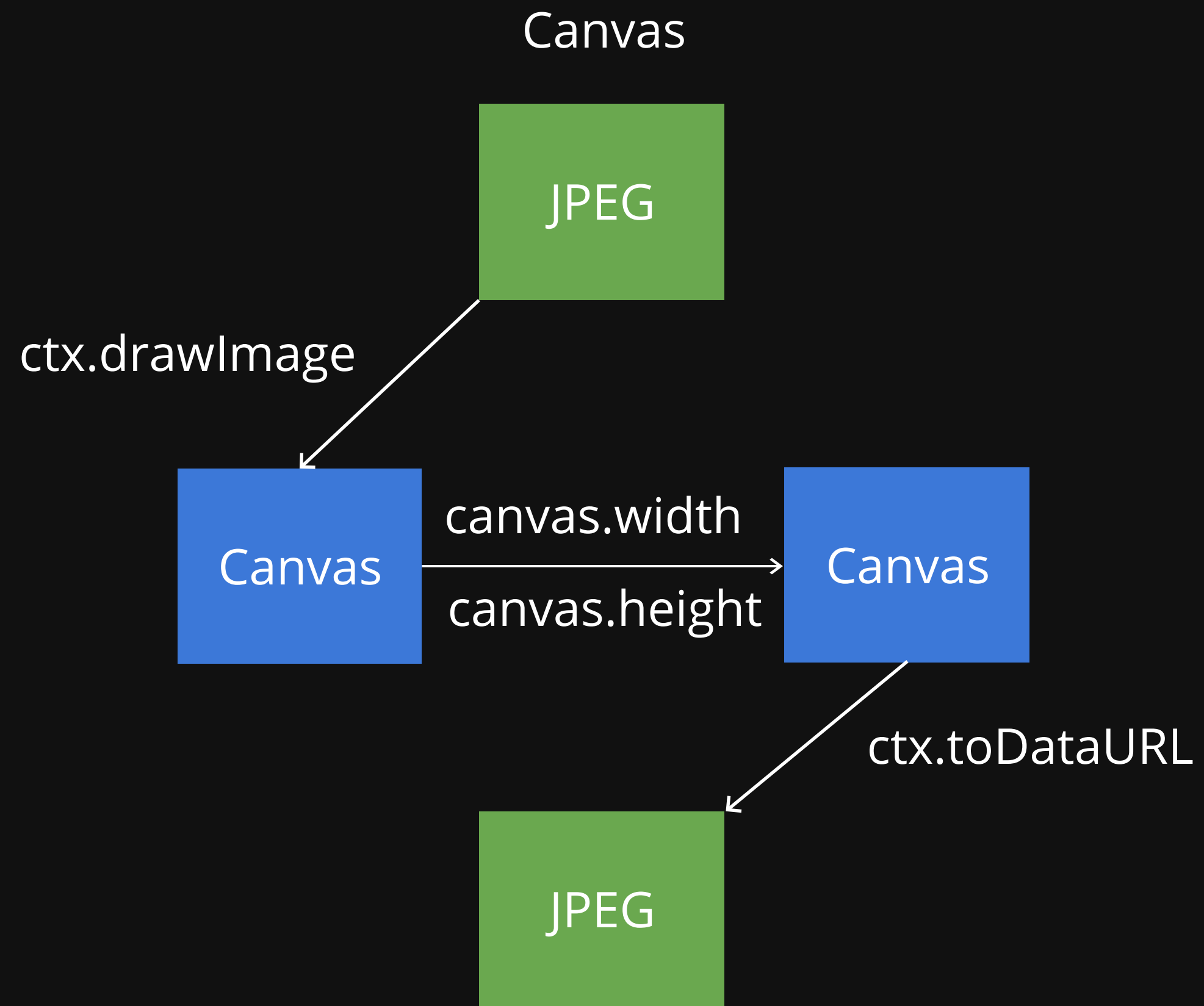


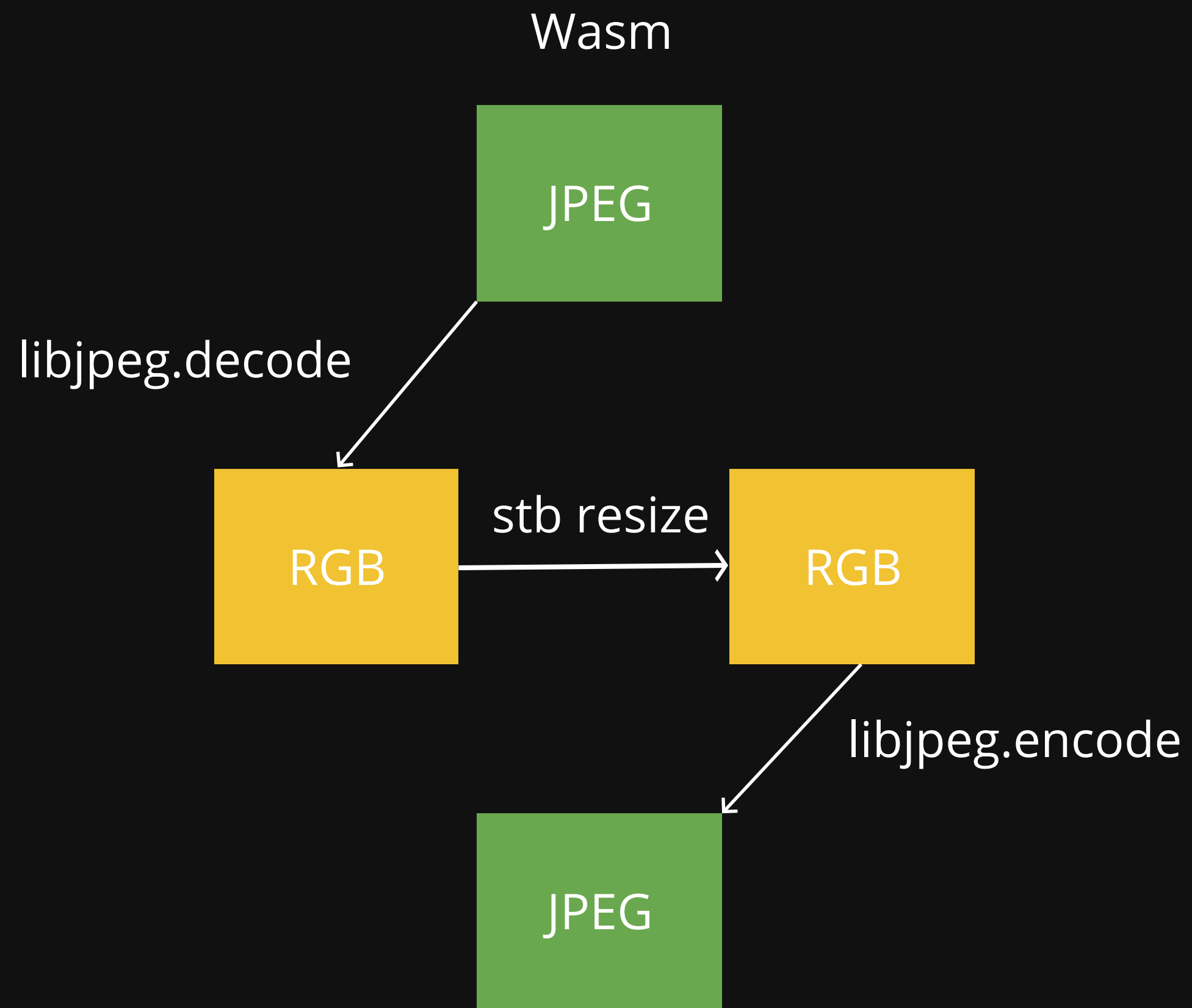
На пальцах



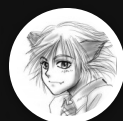
$$E = (A + B + C + D) / 4$$



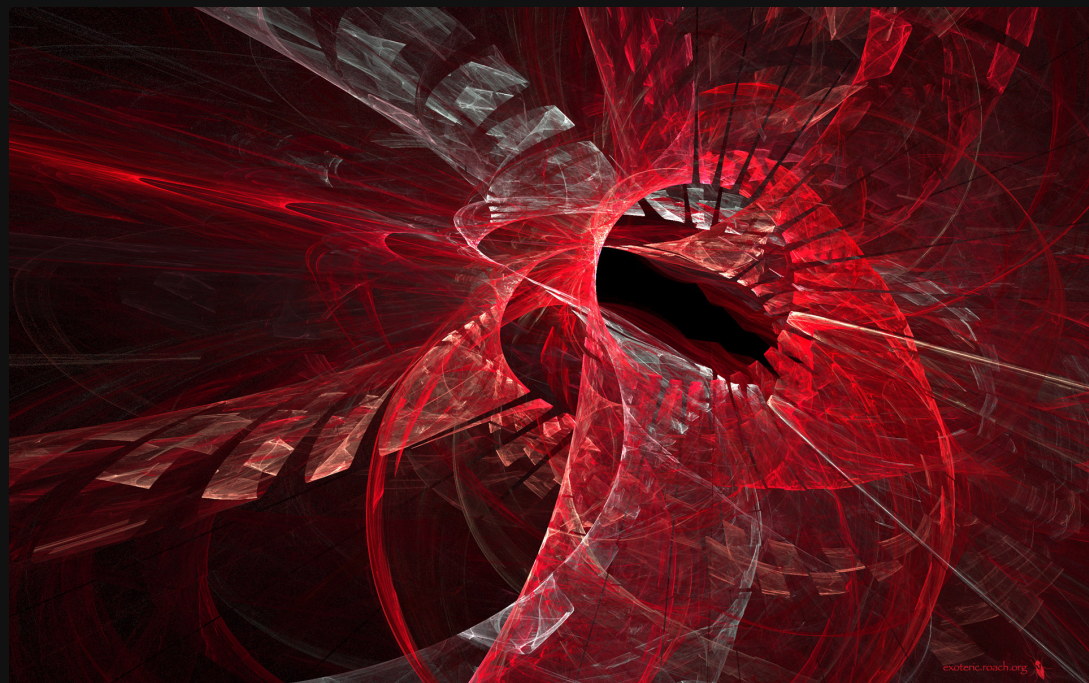




Демо (downsample)



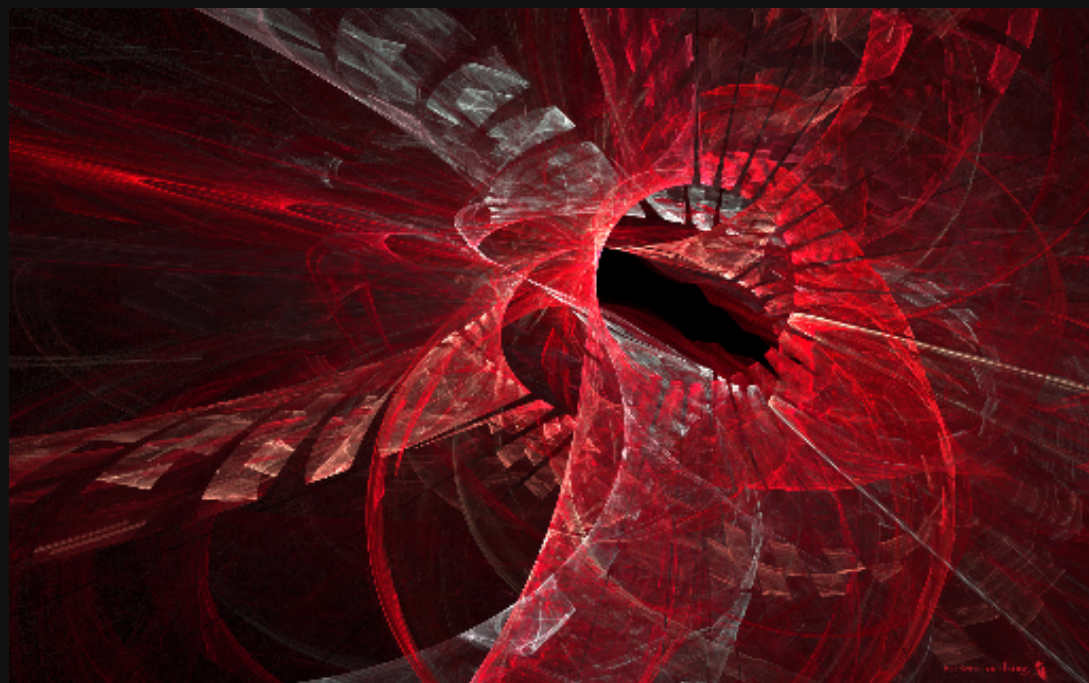
Original



Chrome - canvas + sq=high



FF, IE - canvas

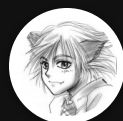


Wasm - laczos3

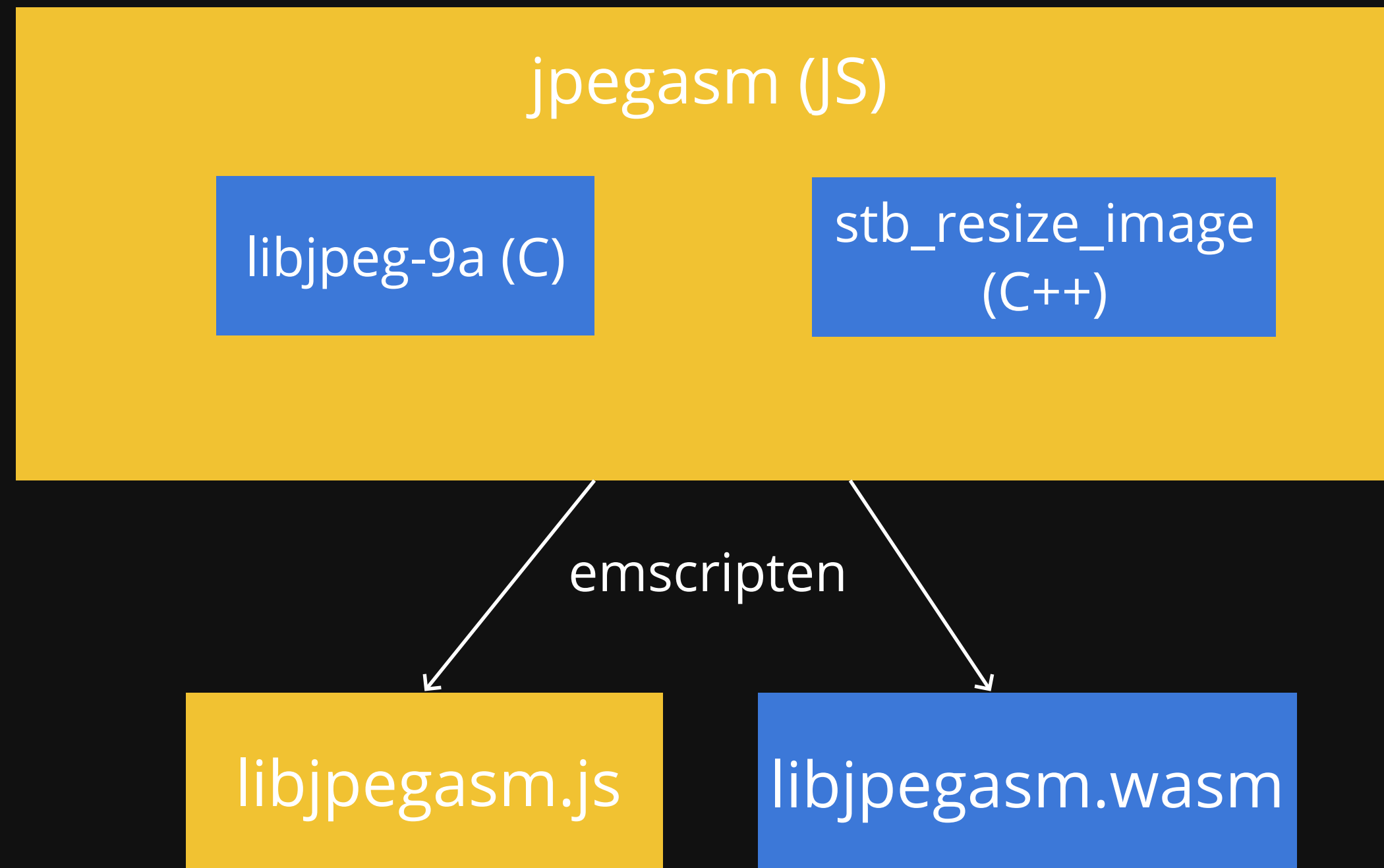


А что внутри?

(и как это повторить у себя в проекте)



Модули



С из JS (cwrap)

```
int encode_jpeg(  
    unsigned char* rgb_buffer,  
    unsigned int rgb_width,  
    unsigned int rgb_height,  
    ...);
```

```
var encode_jpeg = Module.cwrap('encode_jpeg',  
    'number', ['number', 'number', 'number',  
    ...]);
```



C++ из JS (embind)

```
int stbir_resize_uint8(  
    const unsigned char *input_pixels ,  
    int input_w , int input_h , ...)
```

```
EMSCRIPTEN_BINDINGS() {  
    function("stbir_resize_uint8_generic",  
        optional_override(...) {  
            ...магия с маршалингом...  
        }  
    ));  
}
```



JS из C (EM_ASM)

```
#include <emscripten.h>

...
float coef = stbir__filter_info_table[filter]
    .kernel((float)(out_last_pixel + 1)
    + 0.5f - out_center_of_in, scale_ratio);

EM_ASM_({console.log('coef', $0)}, coef)

...
```

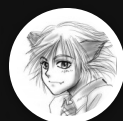


Резюме

- собираем emscripten
- cwrap для C, embind для C++
- optional_override для сложных типов
- EM_ASM в обратную сторону
- Забудьте про GC, .delete() руками



Вместо заключения

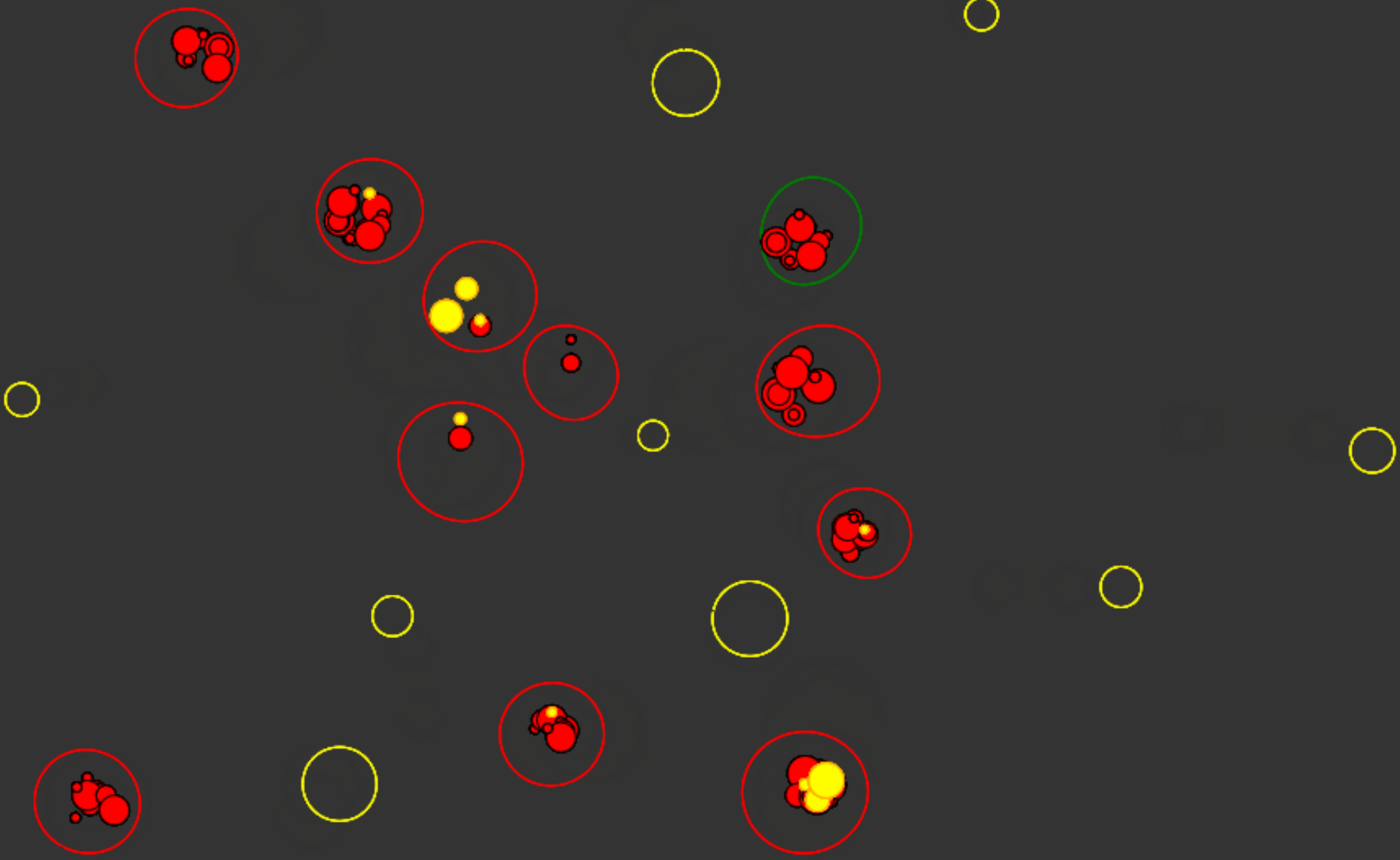


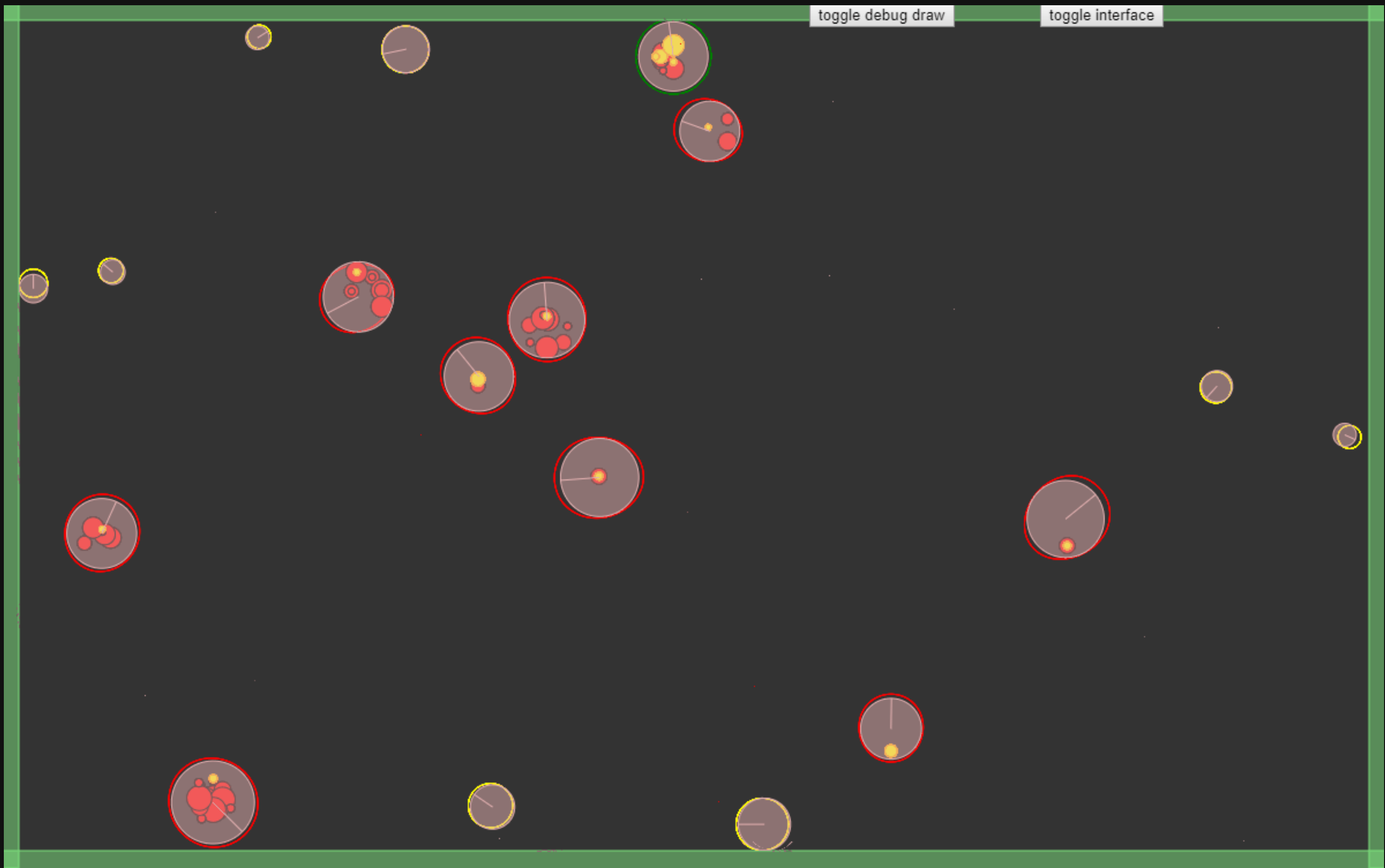
Демо (игра)



toggle debug draw

toggle interface





Полезности

- Спецификация `wasm`
- `emscripten`
- демка
- исходники демки
- `pico`



Спасибо за внимание!
Вопросы?



malcoriel



malcoriel@gmail.com

<http://bit.ly/wasm-ft>



WebAssembly - OTHER

Global

58.25%

WebAssembly or "wasm" is a new portable, size- and load-time-efficient format suitable for compilation to the web.

Current aligned Usage relative Date relative Show all

IE	Edge *	Firefox	Chrome	Safari	Opera	iOS Safari *	Opera Mini *	Android Browser *	Chrome for Android
			49						
			59					4.4	
	14	54	60			10.2		4.4.4	
11	³ 15	55	61	10.1	47	10.3	all	56	61
	16	56	62	11	48	11			
		57	63	TP	49				
		58	64						

Notes Known issues (0) Resources (7) Feedback

MS Edge status: **Preview Release**

³ Can be enabled via the Experimental JavaScript Features flag

