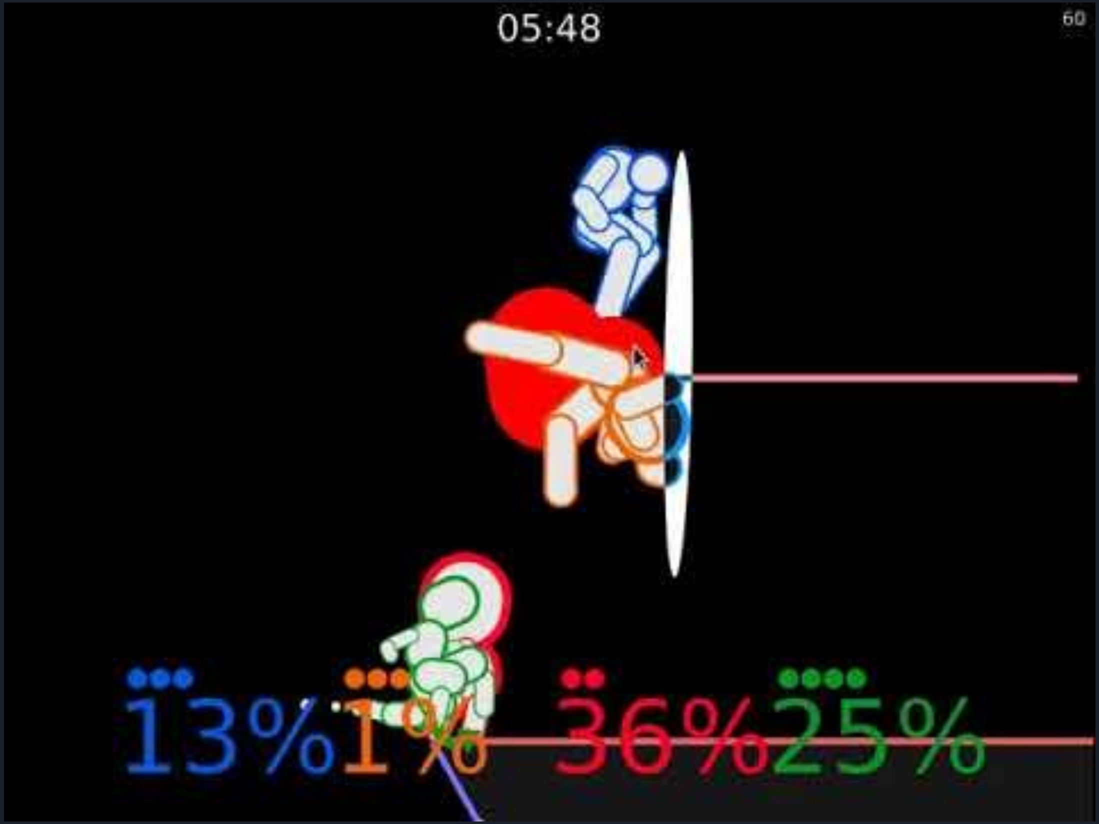
Building a framedata website in rust

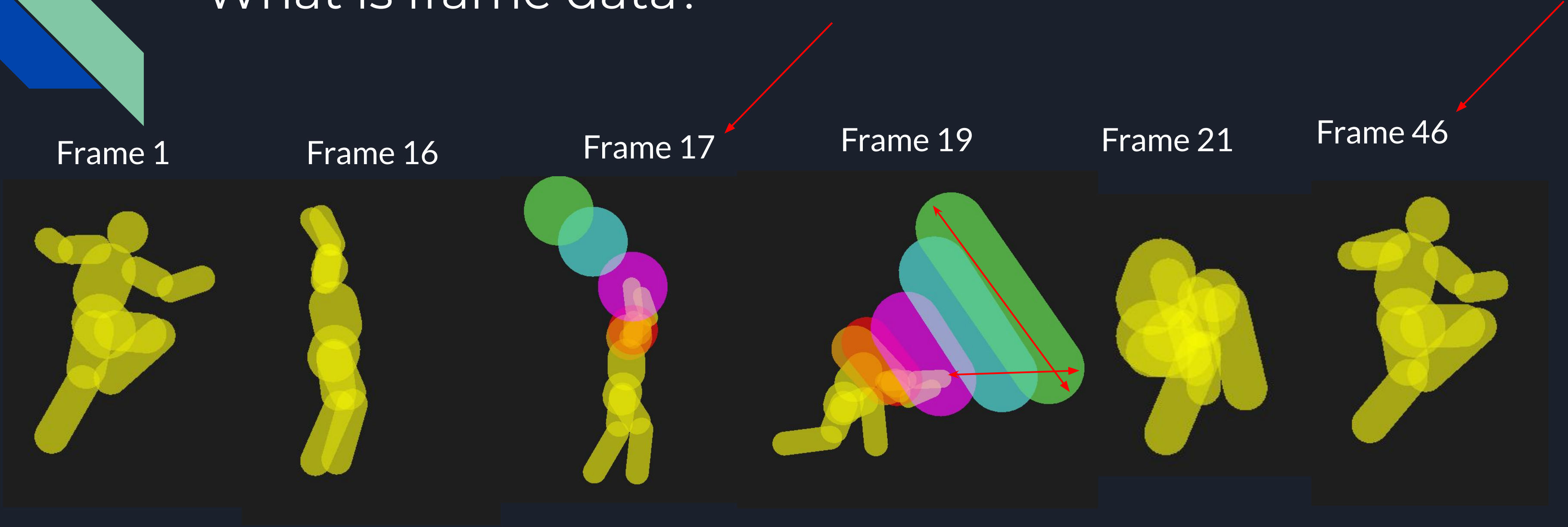
- https://twitter.com/thisIsRukai
- https://github.com/rukai
- A recording of this talk

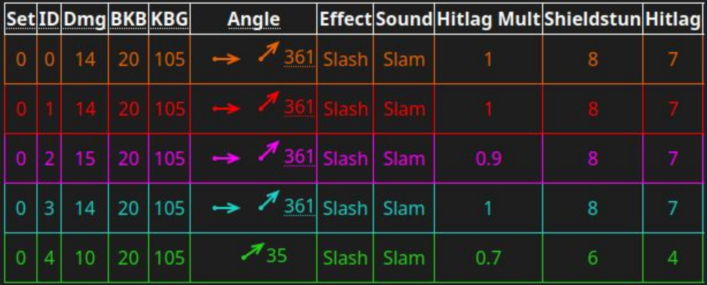
My starting point

- pf_sandbox - a smash-like game engine
- brawllib_rs - reads character data from smash's game files
- Combine them both to get a game with content!



What is frame data?

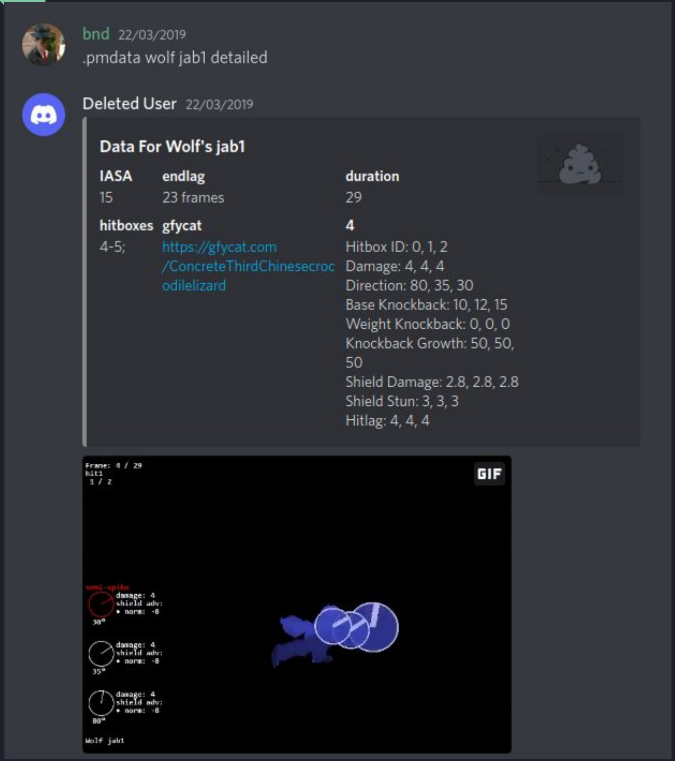




Set	ID	Dmg	BKB	KBG	Angle	Effect	Sound	Hitlag Mult	Shieldstun	Hitlag
0	0	14	20	105	→ ↗ 361	Slash	Slam	1	8	7
0	1	14	20	105	→ ↗ 361	Slash	Slam	1	8	7
0	2	15	20	105	→ ↗ 361	Slash	Slam	0.9	8	7
0	3	14	20	105	→ ↗ 361	Slash	Slam	1	8	7
0	4	10	20	105	↗ 35	Slash	Slam	0.7	6	4

Unsatisfied with existing resources

A discord bot
(semi-automated)



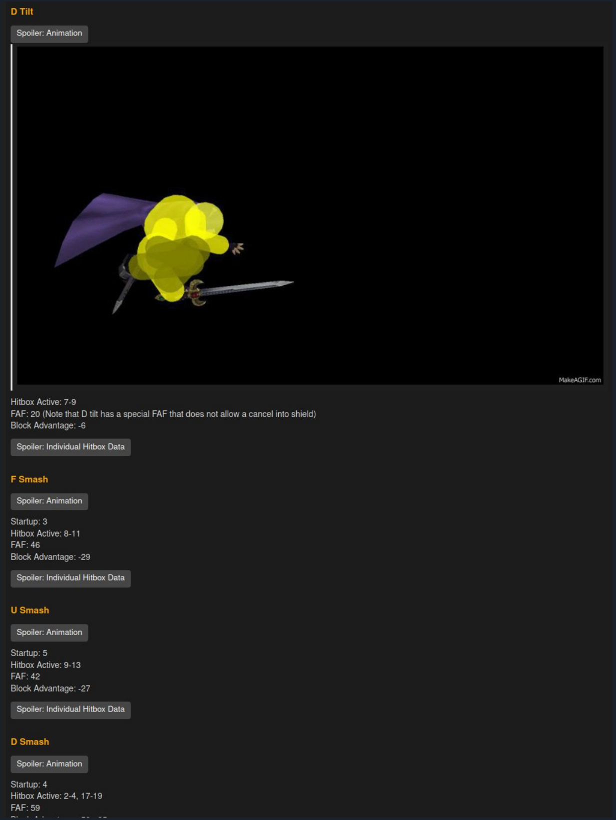
bnd 22/03/2019
.pmdata wolf jab1 detailed

Deleted User 22/03/2019

Data For Wolf's jab1

IASA	**endlag**	**duration**
15	23 frames	29
hitboxes	**gfycat**	**4**
4-5;	https://gfycat.com/ConcreteThirdChinesecrocodilelizard	Hitbox ID: 0, 1, 2 Damage: 4, 4, 4 Direction: 80, 35, 30 Base Knockback: 10, 12, 15 Weight Knockback: 0, 0, 0 Knockback Growth: 50, 50, 50 Shield Damage: 2.8, 2.8, 2.8 Shield Stun: 3, 3, 3 Hitlag: 4, 4, 4

forum threads
(data/graphics manually compiled)



D Tilt

Spoiler: Animation

Hitbox Active: 7-9
FAF: 20 (Note that D tilt has a special FAF that does not allow a cancel into shield)
Block Advantage: -6

Spoiler: Individual Hitbox Data

F Smash

Spoiler: Animation

Startup: 3
Hitbox Active: 8-11
FAF: 46
Block Advantage: -29

Spoiler: Individual Hitbox Data

U Smash

Spoiler: Animation

Startup: 5
Hitbox Active: 9-13
FAF: 42
Block Advantage: -27

Spoiler: Individual Hitbox Data

D Smash

Spoiler: Animation

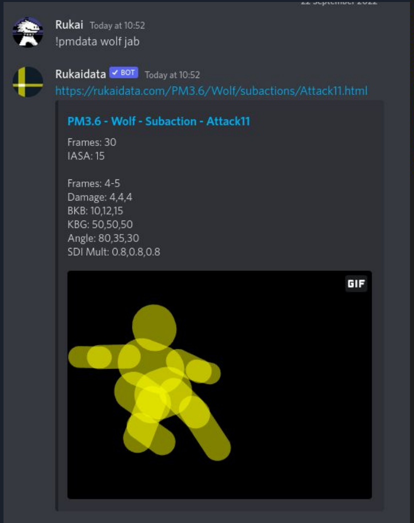
Startup: 4
Hitbox Active: 2-4, 17-19
FAF: 59

I'm going to use brawllib_rs to make something better

By the magic of retrospection I can show you what it will look like:

- 3d renderer
- Step through frame by frame
- More detail
- Automation
- Optional discord bot shows preview with gifs

Discord bot



Rukai Today at 10:52
!pmdata wolf jab

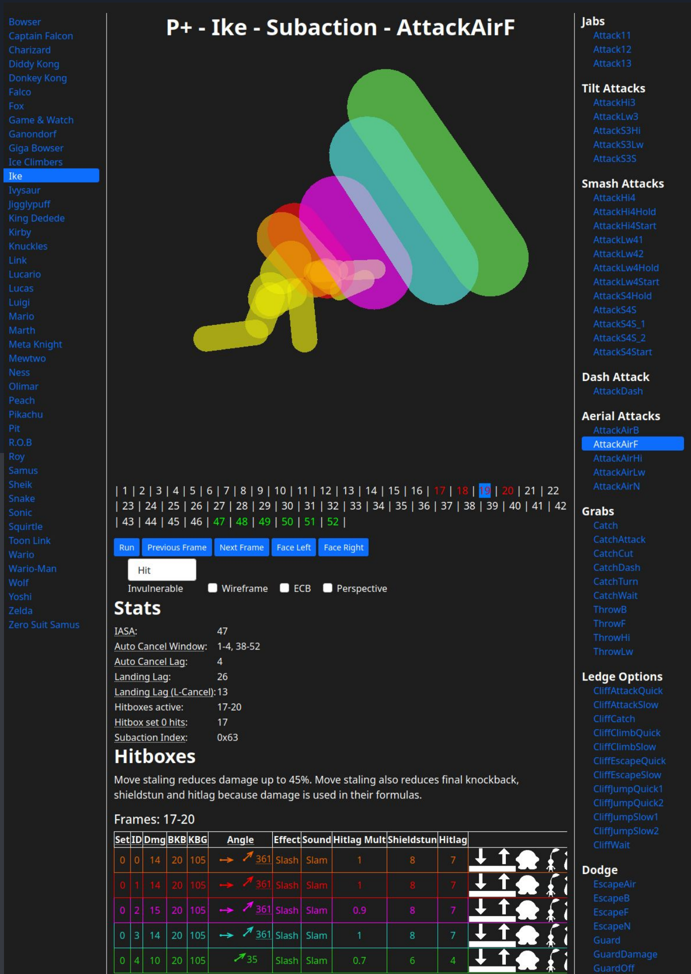
Rukaidata ✓ BOT Today at 10:52
https://rukaidata.com/PM3.6/Wolf/subactions/Attack11.html

PM3.6 - Wolf - Subaction - Attack11

Frames: 30
IASA: 15

Frames: 4-5
Damage: 4,4,4
BKB: 10,12,15
KBG: 50,50,50
Angle: 80,35,30
SDI Mult: 0.8,0.8,0.8

website



Bowser
Captain Falcon
Charizard
Diddy Kong
Donkey Kong
Falco
Fox
Game & Watch
Ganondorf
Giga Bowser
Ice Climbers
Ike
Ivysaur
Jigglypuff
King Dedede
Kirby
Knuckles
Link
Lucario
Lucas
Luigi
Mario
Marth
Meta Knight
Mewtwo
Ness
Olimar
Peach
Pikachu
Pit
R.O.B
Roy
Samus
Sheik
Snake
Sonic
Squirtle
Toon Link
Wario
Wario-Man
Wolf
Yoshi
Zelda
Zero Suit Samus

P+ - Ike - Subaction - AttackAirF

| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 | 21 | 22
| 23 | 24 | 25 | 26 | 27 | 28 | 29 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 | 38 | 39 | 40 | 41 | 42
| 43 | 44 | 45 | 46 | 47 | 48 | 49 | 50 | 51 | 52 |

Stats

IASA:	47
Auto Cancel Window:	1-4, 38-52
Auto Cancel Lag:	4
Landing Lag:	26
Landing Lag (L-Cancel):	13
Hitboxes active:	17-20
Hitbox set 0 hits:	17
Subaction Index:	0x63

Hitboxes

Move staling reduces damage up to 45%. Move staling also reduces final knockback, shieldstun and hitlag because damage is used in their formulas.

Frames: 17-20

Set	ID	Dmg	BKB	KBG	Angle	Effect	Sound	Hitlag Mult	Shieldstun	Hitlag
0	0	14	20	105	→ ↗ 361	Slash	Slam	1	8	7
0	1	14	20	105	→ ↗ 361	Slash	Slam	1	8	7
0	2	15	20	105	→ ↗ 361	Slash	Slam	0.9	8	7
0	3	14	20	105	→ ↗ 361	Slash	Slam	1	8	7
0	4	10	20	105	↗ 35	Slash	Slam	0.7	6	4

Jabs
Attack11
Attack12
Attack13

Tilt Attacks
AttackHi3
AttackLw3
AttackS3Hi
AttackS3Lw
AttackS3S

Smash Attacks
AttackHi4
AttackHi4Hold
AttackHi4Start
AttackLw41
AttackLw42
AttackLw4Hold
AttackLw4Start
AttackS4Hold
AttackS4S
AttackS4S_1
AttackS4S_2
AttackS4Start

Dash Attack
AttackDash

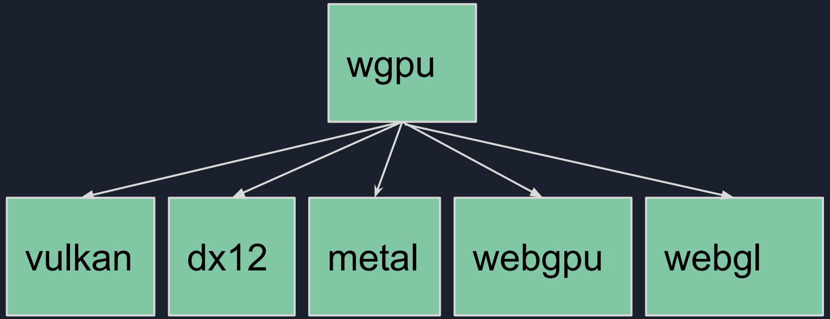
Aerial Attacks
AttackAirB
AttackAirF
AttackAirHi
AttackAirLw
AttackAirN

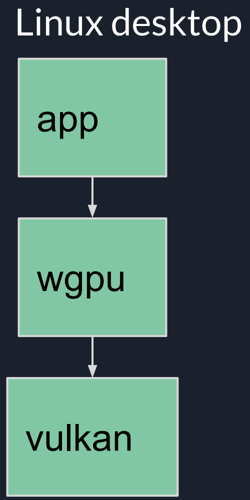
Grabs
Catch
CatchAttack
CatchCut
CatchDash
CatchTurn
CatchWait
ThrowB
ThrowF
ThrowHi
ThrowLw

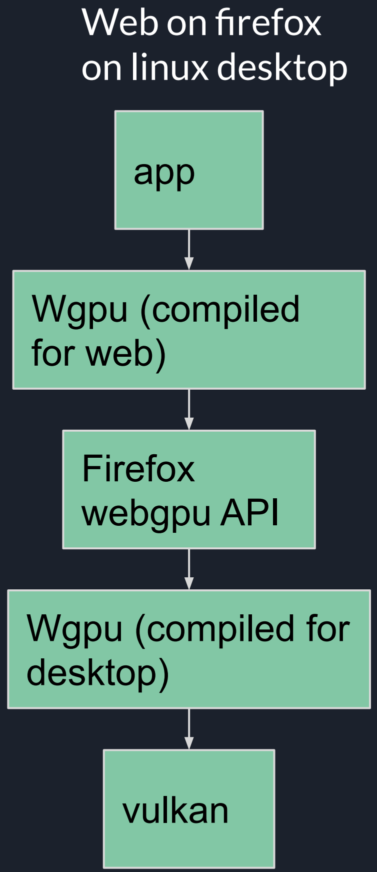
Ledge Options
CliffAttackQuick
CliffAttackSlow
CliffCatch
CliffClimbQuick
CliffClimbSlow
CliffEscapeQuick
CliffEscapeSlow
CliffJumpQuick1
CliffJumpQuick2
CliffJumpSlow1
CliffJumpSlow2
CliffWait

Dodge
EscapeAir
EscapeB
EscapeF
EscapeN
Guard
GuardDamage
GuardOff

I need a graphics library...



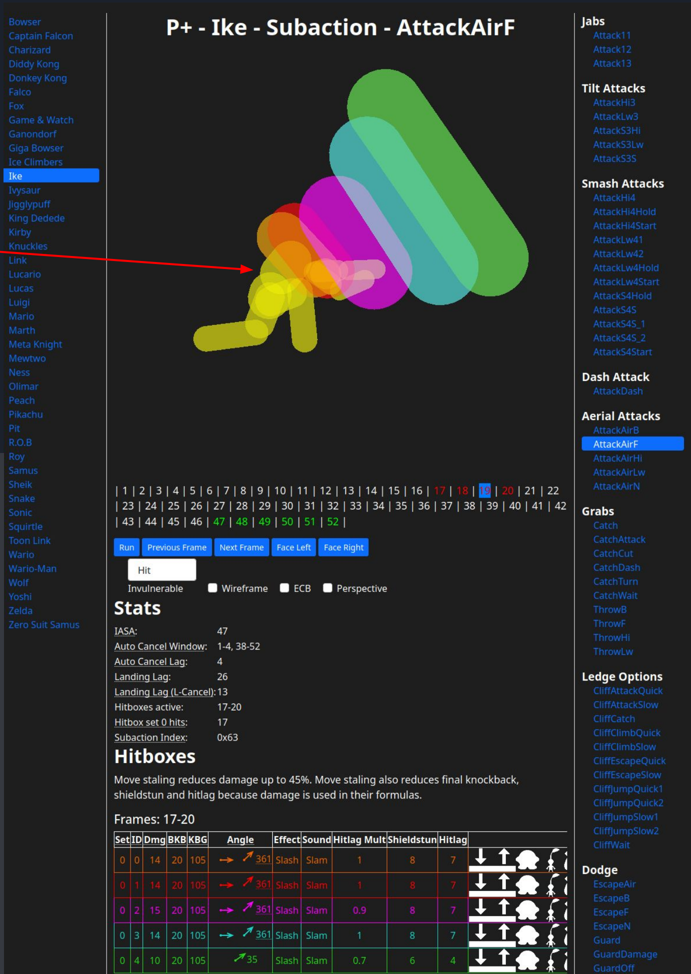




The renderer

The renderer running in a wasm app

website



P+ - Ike - Subaction - AttackAirF

Stats

IASA:	47
Auto Cancel Window:	1-4, 38-52
Auto Cancel Lag:	4
Landing Lag:	26
Landing Lag (L-Cancel):	13
Hitboxes active:	17-20
Hitbox set 0 hits:	17
Subaction Index:	0x63

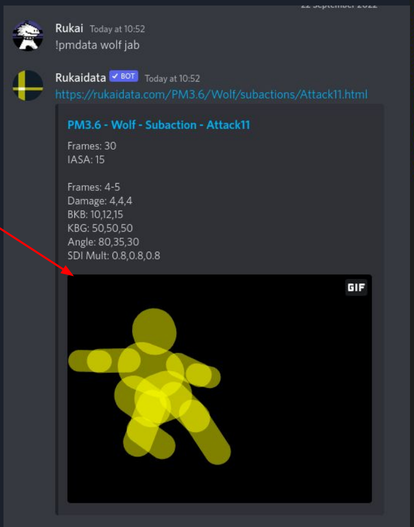
Hitboxes

Move staling reduces damage up to 45%. Move staling also reduces final knockback, shieldstun and hitlag because damage is used in their formulas.

Frames: 17-20

Set	ID	Dmg	BKB	KBG	Angle	Effect	Sound	Hitlag Mult	Shieldstun	Hitlag
0	0	14	20	105	→ ↗ 361	Slash	Slam	1	8	7
0	1	14	20	105	→ ↗ 361	Slash	Slam	1	8	7
0	2	15	20	105	→ ↗ 361	Slash	Slam	0.9	8	7
0	3	14	20	105	→ ↗ 361	Slash	Slam	1	8	7
0	4	10	20	105	↗ 35	Slash	Slam	0.7	6	4

Discord bot



Rukai Today at 10:52
!pmdata wolf jab

Rukaidata BOT Today at 10:52
https://rukaidata.com/PM3.6/Wolf/subactions/Attack11.html

PM3.6 - Wolf - Subaction - Attack11

Frames: 30
IASA: 15

Frames: 4-5
Damage: 4,4,4
BKB: 10,12,15
KBG: 50,50,50
Angle: 80,35,30
SDI Mult: 0.8,0.8,0.8

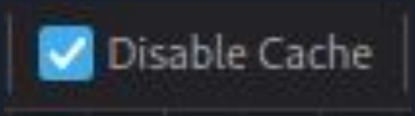
Prerendered gif included in the pages embed metadata:
<meta name="twitter:image" content="/P+/Mario/subactions/AttackHi3.gif">

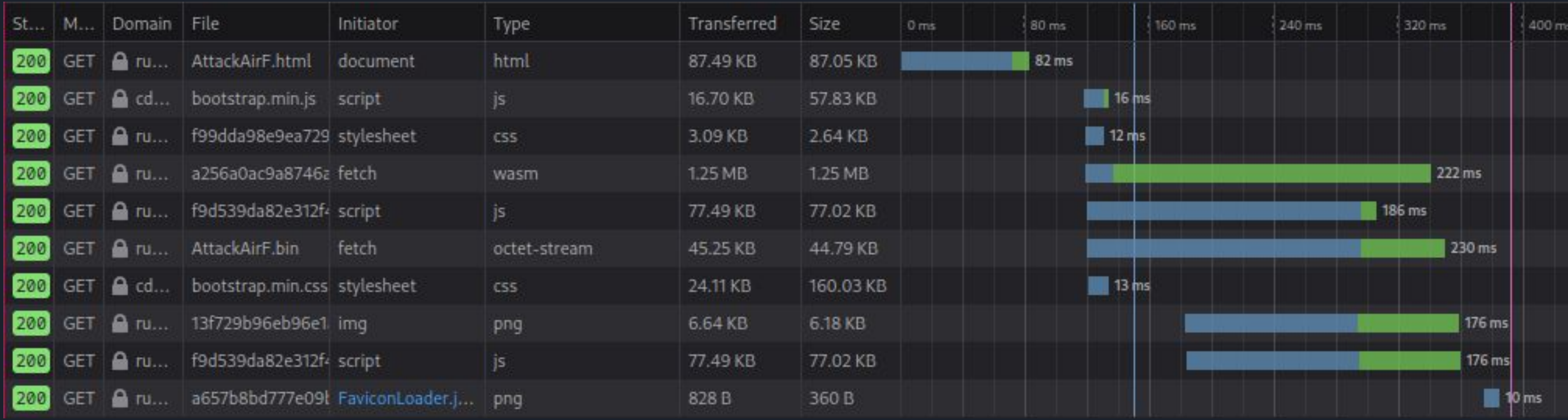
Generated by the same renderer but compiled natively instead.

I need a webserver...

- Originally built with rocket
- Halfway through I swapped to a static site
- Storing the generated website takes some serious space
  - Average storage per action (page + gif) = 400KB
  - Average number of actions per character = 500
  - Average number of characters per game = 40
  - Number of games = currently 4
  - So… 400 * 500 * 40 * 4 = ~30GB
- So generate and store the website entirely within AWS to avoid uploading 30GB

I think it was worth it

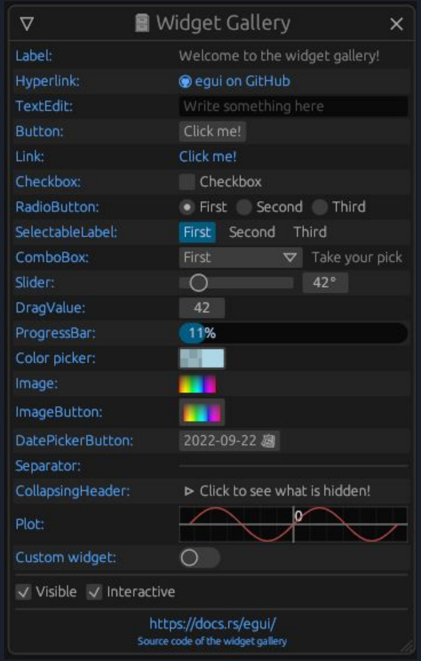

[✓] Disable Cache



St…	M…	Domain	File	Initiator	Type	Transferred	Size	0 ms 80 ms 160 ms 240 ms 320 ms 400 m
200	GET	ru…	AttackAirF.html	document	html	87.49 KB	87.05 KB	82 ms
200	GET	cd…	bootstrap.min.js	script	js	16.70 KB	57.83 KB	16 ms
200	GET	ru…	f99dda98e9ea729	stylesheet	css	3.09 KB	2.64 KB	12 ms
200	GET	ru…	a256a0ac9a8746a	fetch	wasm	1.25 MB	1.25 MB	222 ms
200	GET	ru…	f9d539da82e312f4	script	js	77.49 KB	77.02 KB	186 ms
200	GET	ru…	AttackAirF.bin	fetch	octet-stream	45.25 KB	44.79 KB	230 ms
200	GET	cd…	bootstrap.min.css	stylesheet	css	24.11 KB	160.03 KB	13 ms
200	GET	ru…	13f729b96eb96e1	img	png	6.64 KB	6.18 KB	176 ms
200	GET	ru…	f9d539da82e312f4	script	js	77.49 KB	77.02 KB	176 ms
200	GET	ru…	a657b8bd777e09b	FaviconLoader.j…	png	828 B	360 B	10 ms

I need a UI...

egui



Widget Gallery

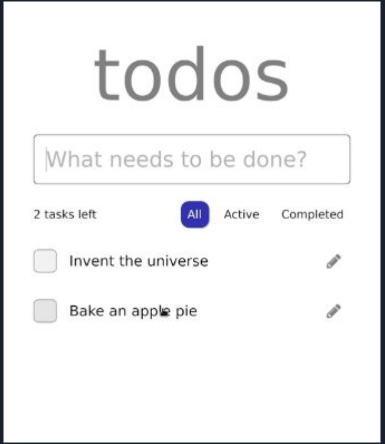
Label:	Welcome to the widget gallery!
Hyperlink:	egui on GitHub
TextEdit:	Write something here
Button:	Click me!
Link:	Click me!
Checkbox:	Checkbox
RadioButton:	First Second Third
SelectableLabel:	First Second Third
ComboBox:	First Take your pick
Slider:	42°
DragValue:	42
ProgressBar:	11%
Color picker:	
Image:	
ImageButton:	
DatePickerButton:	2022-09-22
Separator:	
CollapsingHeader:	▶ Click to see what is hidden!
Plot:	
Custom widget:	

✓ Visible ✓ Interactive

https://docs.rs/egui/

Source code of the widget gallery

iced



todos

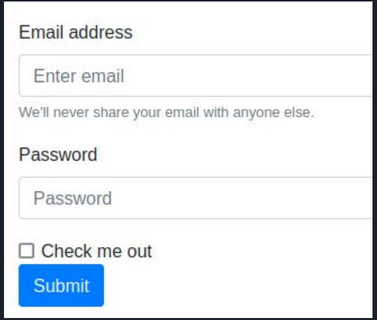
What needs to be done?

2 tasks left All Active Completed

Invent the universe

Bake an apple pie

HTML (bootstrap styled)



Email address

Enter email

We'll never share your email with anyone else.

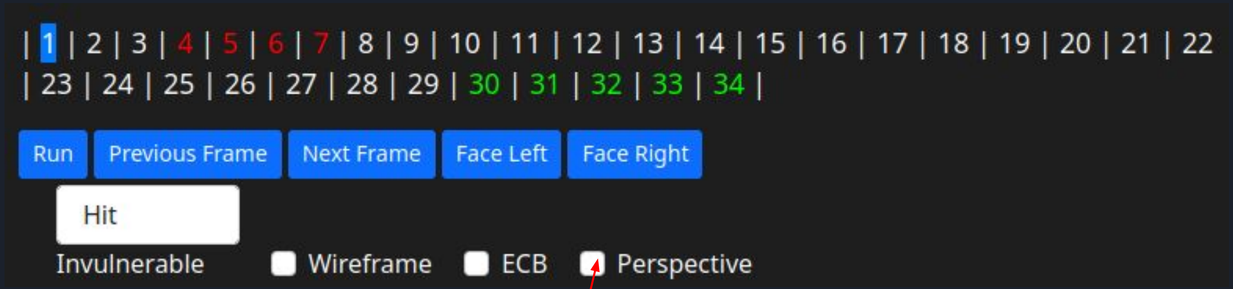
Password

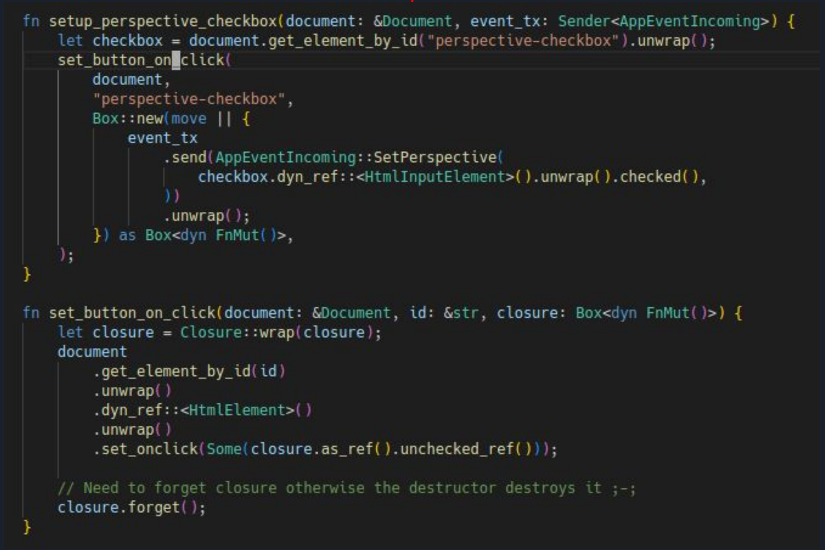
Password

Check me out

Submit

The UI





```rust
fn setup_perspective_checkbox(document: &Document, event_tx: Sender<AppEventIncoming>) {
    let checkbox = document.get_element_by_id("perspective-checkbox").unwrap();
    set_button_on_click(
        document,
        "perspective-checkbox",
        Box::new(move || {
            event_tx
                .send(AppEventIncoming::SetPerspective(
                    checkbox.dyn_ref::<HtmlInputElement>().unwrap().checked(),
                ))
                .unwrap();
        }) as Box<dyn FnMut()>,
    );
}

fn set_button_on_click(document: &Document, id: &str, closure: Box<dyn FnMut()>) {
    let closure = Closure::wrap(closure);
    document
        .get_element_by_id(id)
        .unwrap()
        .dyn_ref::<HtmlElement>()
        .unwrap()
        .set_onclick(Some(closure.as_ref().unchecked_ref()));

    // Need to forget closure otherwise the destructor destroys it ;-;
    closure.forget();
}
```

A quick demo

https://rukaidata.com/P+/Marth/subactions/AttackAirF.html

Hacks

- Wgpu is running over webgl instead of webgpu
- Using a fork of winit that uses the unstable ResizeObserver