ti&m

Hack Night Bern
Svelte

Yannik Inniger, Patrizio Brantschen, Sandro Gerber

Bern, 05. Oktober 2023

ti&m

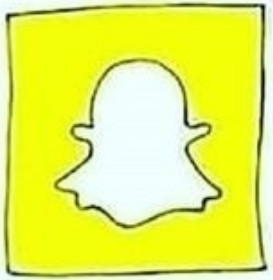
- Yet another JS Framework?
- Comparison to other Frameworks
- Key Concepts
- Hacking

ti&m

WHAT HAPPENS IN ONE MINUTE?

NETFLIX

70,000 Hours of
Netflix watched



3 million videos
watched on Snapchat

Google

Who is Cardi B?

Google Search | I'm Feeling Lucky

Google is asked
2.4 million questions

JS

A new JS framework
appears

History of Svelte

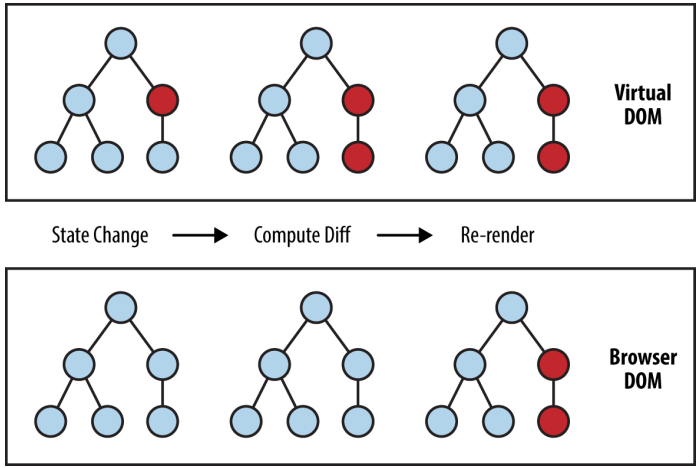
ti&m

Do we really need another Framework?

- One of the most loved framework by developers
- Compiler instead of virtual DOM
- Performance
- Simplicity & built-in functionality

Virtual DOM

```
<html>
  <body>
    <h1>Hello World</h1>
    <div *ngIf=“someState”>
      <some-component></ some-component>
      <other-component></ other-component>
    </div>
  </body>
</html>
```



How does Svelte work?

Svelte Code

Svelte Compiler

Plain HTML/JS/CSS

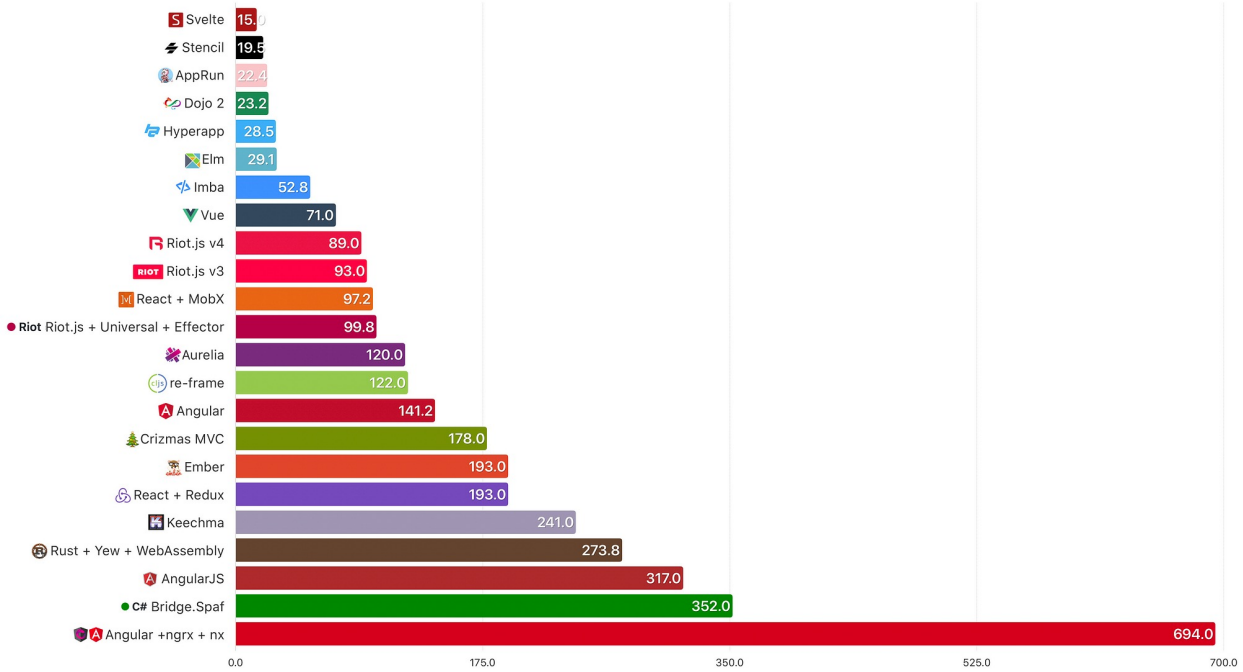
- Yet another JS Framework?
- Comparison to other Frameworks
- Key Concepts
- Hacking

ti&m

When to use Svelte?

- Performance is a concern
- Rapid prototyping and MVP development
- Component Libraries
- Building fast & accessible UIs

Bundle Sizes in KB



ti&m

When to use other Frameworks?

- Reliance on 3rd-party plugins/libraries
- Need for extensive support

Built-in Functionalities

- Rounting
- State managements with Stores
- Animations & Effects
- Accesibillity checkers

- Yet another JS Framework?
- Comparison to other Frameworks
- Key Concepts
- Hacking

ti&m

Your typical Svelte file

```
<script>
  let name = 'Yannik';
</script>

<h1>Hello {name}!</h1>

<style>
  h1 {
    color: red;
  }
</style>
```

Reactivity

```
<script>
    let count = 0;
    $: doubled = count * 2;

    function increment() {
        count += 1;
    }
</script>

<button on:click={increment}>
    Clicked {count}
    {count === 1 ? 'time' : 'times'}
</button>

<p>{count} doubled is {doubled}</p>
```

Components & Props

```svelte
// PackageInfo.svelte

<script>
    export let name;
    export let version;
    export let speed;
    export let website;

    $: href = `https://www.npmjs.com/package/${name}`;
</script>

<div>...</div>
```

```svelte
// App.svelte

<script>
    import PackageInfo from './PackageInfo.svelte';

    const url = 'https://svelte.dev';
    const name = 'svelte';
    const pkg = {
        speed: 'blazing',
        version: 4,
    };
</script>

<PackageInfo website={url} {name} {...pkg} />
```

Logic in Templates

```
<script>
    let count = 0;
</script>

{#if count > 10}
    <p>{count} is greater than 10</p>
{:else if count < 5}
    <p>{count} is less than 5</p>
{:else}
    <p>{count} is between 5 and 10</p>
{/if}
```

```
<script>
    import { getRandomNumber } from './utils.js';

    let promise = getRandomNumber();
</script>

{#await promise}
    <p>...waiting</p>
{:then number}
    <p>The number is {number}</p>
{:catch error}
    <p style="color: red">{error.message}</p>
{/await}
```

Bindings

```
<script>
    let name = 'world';

    const onClick = () => {
        name = ''
    }
</script>

<input bind:value={name} />

<button on:click={onClick}>Reset name</button>

<h1>Hello {name}!</h1>
```

Creating stores

```
import { writable } from 'svelte/store';

export const count = writable(0);
```

Using stores

```
<script>
    import { count } from './stores.js';

    let count_value;

    count.subscribe((value) => {
        count_value = value;
    });

    const increment = () => {
        count.update((n) => n + 1);
    }

    const reset = () => {
        count.set(0);
    }
</script>

<h1>The count is {count_value}</h1>

<button on:click={increment}>+</button>
<button on:click={reset}>reset</button>
```

Yet another JS Framework?

Comparison to other Frameworks

Key Concepts

Hacking