

Let's have a View on Vue.js

Norbert Frank

- Developer, Consultant, IT-Architect @ Lucom GmbH
- Vue-Enthusiast (and Web and JS in general)
- @norbertdfrank



Seven reasons to learn Vue.js in 2019

Taken from: <https://dev.to/rossta/seven-reasons-learn-vuejs-in-2019-2n9o>

1. No build step required
2. A world-class command line interface
3. Friendly learning curve
4. A solid ecosystem
5. The Vue instance
6. Vue 3 lands this year
7. An amazing community

Bonus: It's just plain fun

Vue.js - The Progressive Framework

JQuery-Alternative

No Build-Step

Simple SPA

Single-File-Components, Webpack

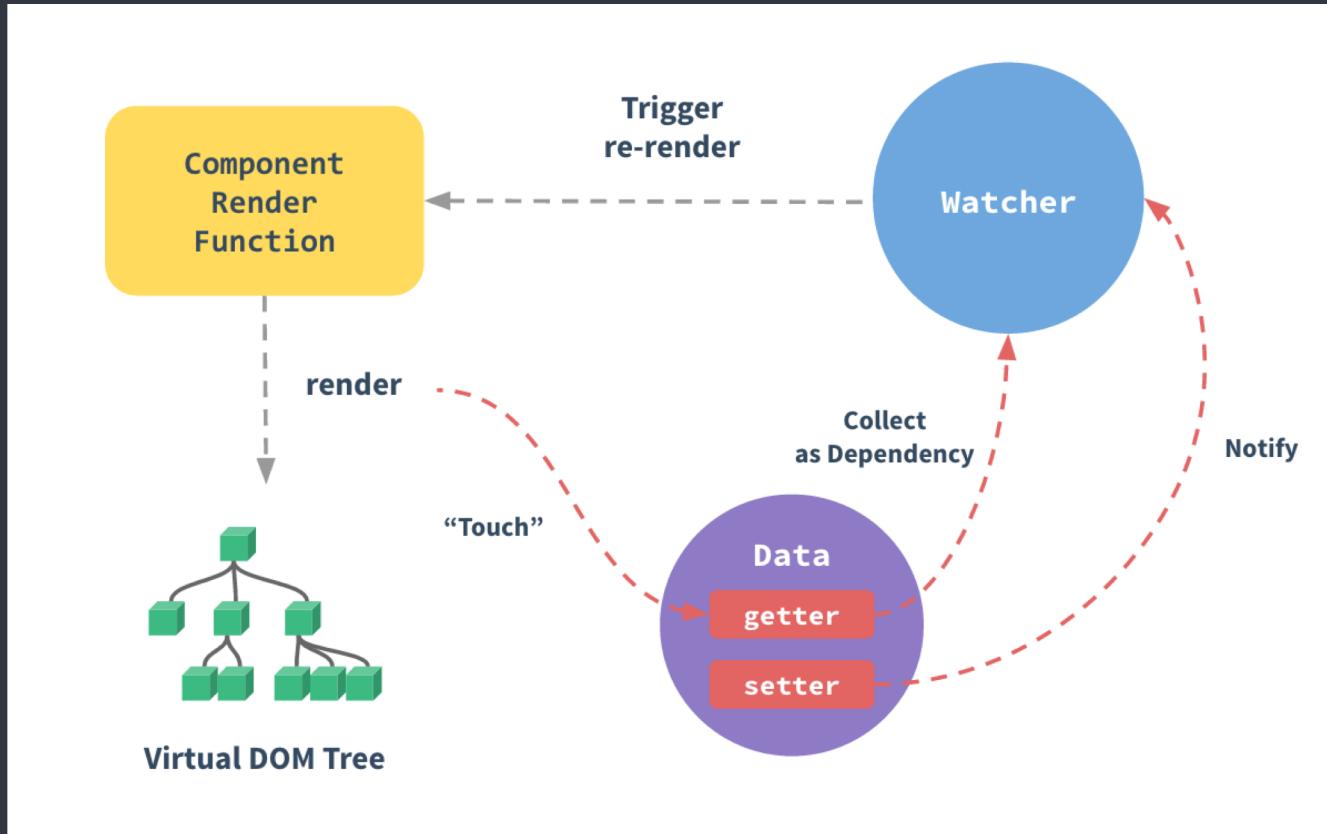
Complex App

Routing, State Management, Lazy Loading,
SSR, complex component hierarchy

Complexity

Components & Reactivity

The Building Blocks



Source: <https://vuejs.org/v2/guide/reactivity.html>

„Components are reusable Vue instances with a name“

<https://vuejs.org/v2/guide/components.html>

Components

- Are Vue instances
- Have a template or a render function
- Are part of the component tree
- Are configured via an object:
 - Data-props, computed props, watcher functions
 - Lifecycle callbacks, methods
- Can publish events
- Can integrate content from parent components via slots

Render functions & JSX

- Templates are compiled internally to render functions
- Render functions can be used instead of templates
- Optionally JSX can be used with render functions

State Management with Vuex

Vuex

- Vuex is the official State Management Library for Vue.js
- It is installed as a plugin but maintained by the same team as Vue itself
- There are other state management libs available but I would only recommend them in special cases
- When using central state management, not only is the state organized differently but also the logic for changing the state!
- The documentation is excellent

Vuex

- Vuex is based on the reactivity system of Vue.js (no immutability!)
- Single state tree but optionally multiple modules
- Changing the state:
 - Mutations: synchronous
 - Actions: asynchronous → Actions commit themselves mutations
- Plugin-Tip: Vuex-Pathify: <https://davestewart.github.io/vuex-pathify/#/>
- News: Actions and mutations might be combined in the next major release!

Routing

Routing

Vue-Router is the official Routing-Library for Vue.js

- Nested routes
- Flexible route matching (query, wildcards, params)
- Multiple routing outlets
- Navigation Guards
- Props and meta data
- View transition effects

There are many other routing libraries for Vue.js as well.

Tip: sync route data with Vuex: <https://github.com/vuejs/vuex-router-sync>

Further Concepts worth mentioning

Mixins

Mixins are a flexible way to manage reusable logic.

- Mixins are imported into components and merged with the component definition
- A component can import multiple Mixins
- Lifecycle-Callbacks in Mixins and the component itself are both called (first from the Mixin)
- Mixins can be defined globally and apply then to all(!) components – use with caution!

Plugins

„Plugins are Vue-Functions, that get Vue as a parameter and do something with it.“

Many possibilities:

- Define global components
- Add logic to the Vue prototype
- Global Mixins
- Directives, Filters, Transistions ...
- Plugins have an `install`-function, which gets Vue and optional further parameters to register the plugin functionality

Scoped Slots

Scoped Slots are a powerful feature but can be hard to understand at first.

A generic component can bind properties and callbacks to a slot. The parent component can get access to the slot scope and use it inside the slot content definition.

GenericComponent.vue

```
...  
<slot :data="item" :id="currentId"></slot>  
...
```

1. Assign properties to the scope

App.vue

```
<generic-component>  
  <template slot-scope="genericCScope">  
    <p>{{genericCScope.data}}</p>  
  </template>  
</generic-component>
```

3. Access the scope

2. Assign a name
To the slot

Extend

Extend a base component with the `extend` feature

```
import BasicComponent from './BasicComponent.vue';

export default {
  extends: BasicComponent,
  // ...
}
```

Alternative to Mixins and Higher Order Components (HoC).

Hint: the HoC-pattern can be used in Vue.js but is not so common as in React.

Transitions & Animation

Vue.js has powerful features to apply transition effects when items are added or removed or routes are changed.

- Automatically apply CSS classes
- JavaScript Events

Vue.js does not contain animations itself but all the logic to write them very easily or integrate 3rd party animation libs like velocity.js

Roadmap

<https://github.com/vuejs/vue/projects/6>

Vue 3

We can expect Vue 3 to be:

- Faster
- Smaller
- Have more maintainable source code (in TypeScript)
- More native-friendly
- Easier to use

<https://medium.com/the-vue-point/plans-for-the-next-iteration-of-vue-js-777ffea6fabf>

Becoming a Vue Ninja

Tips & Ressources

Sample-App

<https://github.com/NoFrank/vue-life-planner>

Contains sample implementations of multiple concepts and can be used
for custom experiments.

Special Tip

You can learn a lot about the concepts by looking at the source code of popular libraries, especially UI-component-libraries:

- Vuetify: <https://github.com/vuetifyjs/vuetify>
- Bootstrap-Vue: <https://github.com/bootstrap-vue/bootstrap-vue>

Tips & Resources

- **Typescript in Vue** <https://vuejs.org/v2/guide/typescript.html>
- **Integrate RxJS with vue-rx** <https://github.com/vuejs/vue-rx>
- **Portal-Vue** <https://github.com/LinusBorg/portal-vue>
- **Progressive Web Apps** <https://github.com/vuejs-templates/pwa>
- **Server Side Rendering** <https://vuejs.org/v2/guide/ssr.html>
- **Vue Enterprise Boilerplate** <https://github.com/chrisvfritz/vue-enterprise-boilerplate>
- **Vue-Test-Utils** <https://vue-test-utils.vuejs.org/>
- **NativeScript** <https://nativescript-vue.org/>
- **And of course Nuxt** <https://nuxtjs.org/>

Tips & Resources

Worth checking out:

- Gridsome ("Gatsby for Vue") <https://gridsome.org/>

Information about Vue:

- Styleguide: <https://vuejs.org/v2/style-guide/>
- Cookbook: <https://vuejs.org/v2/cookbook/>
- Podcast “Views on Vue”: <https://devchat.tv/views-on-vue/>
- Articles on alligator.io <https://alligator.io/vuejs/>
- Vue-Newsletter <https://news.vuejs.org/>
- Twitter: #vuejs und @vuejs

Thanks – I hope you enjoyed it!

@norbertdfrank