



# Is it time to migrate to Vue.js 3.x?

**FOSDEM 2023**

Bruxelles, February 5th, 2023

# WHO AM I



## Denny Biasioli

Full Stack Developer  
(JavaScript, Python, Go)

[denny.biasioli@gmail.com](mailto:denny.biasioli@gmail.com)

Front End Developer UX/ UI  
Fingerprint Supervision Ltd

[www.dennybiasioli.com](http://www.dennybiasioli.com)

@dennybiasioli

Italy, Savigliano (CN)



Vue.js Version	Release date
2.6	February 4, 2019
3.0	September 18, 2020
3.1	June 7, 2021
3.2	August 5, 2021
3.x as new default	February 7, 2022
2.7 (maintenance mode)	July 1, 2022
2.x End of Life	End of 2023

# OPTIONS API (VUE 2, VUE 3)

```
export default {  
  data() {  
    return { count: 0 }  
  },  
  methods: {  
    increment() {  
      this.count++  
    },  
  },  
}
```

```
<template>  
  <button @click="increment">Count is: {{ count }}</button>  
</template>
```

SEPTEMBER 18, 2020

## VUE 3.0 OFFICIALLY ANNOUNCED

- Composition API
- `<script setup>` (experimental)
- no IE11 support

<https://blog.vuejs.org/posts/vue-3-one-piece.html>



# COMPOSITION API IN `setup()` (VUE 3)

```
<script>
import { ref } from 'vue'

export default {
  setup() {
    const count = ref(0)

    function increment() {
      count.value++
    }

    return { count, increment }
  },
}
```

```
<template>
  <button @click="increment">Count is: {{ count }}</button>
</template>
```

# COMPOSITION API IN `setup()`

(Vue 2 + `@vue/composition-api`)

```
import VueCompositionAPI from '@vue/composition-api'
```

```
Vue.use(VueCompositionAPI)
```

```
- import { ref, ... } from 'vue'
```

```
+ import { ref, ... } from '@vue/composition-api'
```



# COMPOSITION API IN `setup()`

(Vue 2 + `@vue/composition-api`)

```
import VueCompositionAPI from '@vue/composition-api'  
  
Vue.use(VueCompositionAPI)
```

```
- import { ref, ... } from 'vue'  
+ import { ref, ... } from '@vue/composition-api'
```

## BROKEN TESTS

# COMPOSITION API IN `setup()`

(Vue 2 + `@vue/composition-api`)

```
import VueCompositionAPI from '@vue/composition-api'  
  
Vue.use(VueCompositionAPI)
```

```
- import { ref, ... } from 'vue'  
+ import { ref, ... } from '@vue/composition-api'
```

## BROKEN TESTS

```
Vue.use(VueCompositionAPI)  
// or  
localVue.use(VueCompositionAPI)
```

in tests files

# COMPOSITION API + SCRIPT SETUP (VUE 3)

```
<script setup>
import { ref } from 'vue'

const count = ref(0)

function increment() {
  count.value++
}
</script>
```

```
<template>
  <button @click="increment">Count is: {{ count }}</button>
</template>
```

JULY 1, 2022

# VUE 2.7 "NARUTO" RELEASED

- Composition API
- `<script setup>` (partial)

<https://blog.vuejs.org/posts/vue-2-7-naruto.html>



# VUE 2.7 UPGRADE GUIDE

## 1. Upgrade local `@vue/cli-xxx` dependencies

```
npx @vue/cli@latest upgrade
```

Upgrade to the latest version in your major version range (if applicable)

- `~4.5.18` for v4
- `~5.0.6` for v5

# VUE 2.7 UPGRADE GUIDE

## 2. Upgrade vue to ^2.7

```
npm install -S vue@~2.7
```

You can also remove `vue-template-compiler` from the dependencies, it is no longer needed in 2.7.

If you are using `@vue/test-utils`, you may need to keep it in the dependencies for now, but this requirement will also be lifted in a new release of test utils.

Always keep `vue` and `vue-template-compiler` in sync.

```
npm install -D vue-template-compiler@~2.7
```

# VUE 2.7 UPGRADE GUIDE

## 3. Check your package manager lockfile for version requirements

```
npm ls vue-loader # should be at least ^15.10.0  
npm ls vue-demi # should be at least ^0.13.1
```

If not, you will need to remove `node_modules` and the lockfile and perform a fresh install to ensure they are bumped to the latest version.



# VUE 2.7 UPGRADE GUIDE

## 4. Remove `@vue/composition-api` and update imports to `vue` instead

```
npm uninstall @vue/composition-api
```

```
- import VueCompositionAPI from '@vue/composition-api'  
- Vue.use(VueCompositionAPI)
```

```
- import { ref, ... } from '@vue/composition-api'  
+ import { ref, ... } from 'vue'
```

More info about Vue 2.7 upgrade

<https://blog.vuejs.org/posts/vue-2-7-naruto.html>

# USING `<script setup>`

```
- <script>
+ <script setup>
  import { ref } from 'vue'

  export default {
    setup() {
      const count = ref(0)

      function increment() {
        count.value++
      }

      return { count, increment }
    },
  }
</script>
```



# Broken vue-router usage!

```
npm i -S vue-router@^3.6.5
```

```
+ import { useRoute, useRouter } from 'vue-router/composables'
```

```
+ const route = useRoute()
```

```
+ const router = useRouter()
```

```
- this.$route
```

```
+ route
```

```
- this.$router
```

```
+ router
```

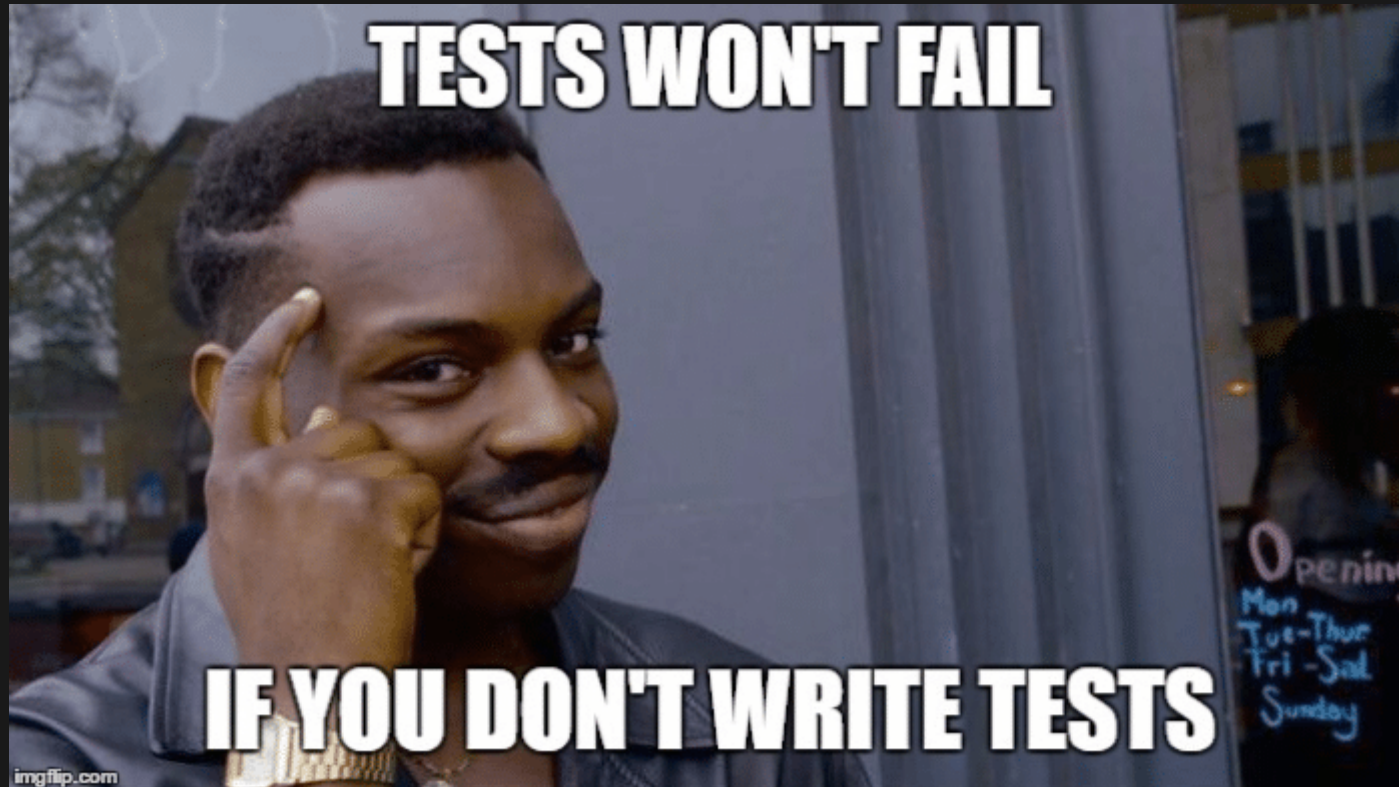
# Same for vuex \$store usage?

## Import the store and navigate through it

or migrate to pinia

```
import store from '@store';  
  
store.state.propertyName  
store.state.moduleName.propertyName  
store.getters.getterName  
store.getters[ 'moduleName/getterName' ]  
store.commit( /* */ )  
store.dispatch( /* */ )
```

# WHAT ABOUT TESTS?



# Broken tests!

```
console.error
```

```
[Vue warn]: Property or method "count" is not defined  
on the instance but referenced during render.
```

```
Make sure that this property is reactive,  
either in the data option, or for class-based components,  
by initializing the property.
```

```
See: https://v2.vuejs.org/v2/guide/reactivity.html#Declaring-R
```

```
console.error
```

```
[Vue warn]: Property or method "increment" is not defined  
on the instance but referenced during render.
```

```
Make sure that this property is reactive,  
either in the data option, or for class-based components,  
by initializing the property.
```

```
See: https://v2.vuejs.org/v2/guide/reactivity.html#Declaring-R
```



# Broken tests!

```
console.error
```

```
[Vue warn]: Property or method "count" is not defined  
on the instance but referenced during render.
```

```
Make sure that this property is reactive,  
either in the data option, or for class-based components,  
by initializing the property.
```

```
See: https://v2.vuejs.org/v2/guide/reactivity.html#Declaring-R
```

```
console.error
```

```
[Vue warn]: Property or method "increment" is not defined  
on the instance but referenced during render.
```

```
Make sure that this property is reactive,  
either in the data option, or for class-based components,  
by initializing the property.
```

```
See: https://v2.vuejs.org/v2/guide/reactivity.html#Declaring-R
```

@vue/vue2-jest 27.x  
not working with Vue 2.7

**SEPTEMBER 15, 2022**

@vue/vue2-jest 29.1 released  
with support for `<script setup>`  
in Vue 2.7...

**SEPTEMBER 15, 2022**

@vue/vue2-jest 29.1 released  
with support for `<script setup>`  
in Vue 2.7...  
...with Jest 29

# USING JEST DIRECTLY

- update jest.config.js

from node\_modules

@vue/cli-plugin-unit-jest/presets/default/jest-preset.js

- update package.json test script

```
- "test:unit": "vue-cli-service test:unit",  
+ "test:unit": "jest",
```

- run tests without vue-cli-service

```
npm run test:unit  
# or  
npx jest
```

# UPGRADING JEST TO 29.X

```
npm uninstall @vue/cli-plugin-unit-jest
npm install -D \
  jest-environment-jsdom \
  jest-serializer-vue \
  jest-transform-stub \
  jest-watch-typeahead \
  @vue/vue2-jest@^29 \
  babel-jest@^29 \
  jest@^29
```

and fix deprecation warnings

```
- testURL: 'http://localhost/',
+ testEnvironmentOptions: {
+   url: 'http://localhost/',
+ },
```

Components using `<script setup>`  
are closed by default

```
const spyIncrement = jest.spyOn(wrapper.vm, 'increment')
```

Won't work anymore

```
defineExpose({ increment }) # not working
```

## Components using `<script setup>` are closed by default

```
const spyIncrement = jest.spyOn(wrapper.vm, 'increment')
```

Won't work anymore

```
defineExpose({ increment }) # not working
```

## Temporary (not suggested) workaround

```
// FIXME: temporary workaround  
- const spyIncrement = jest.spyOn(wrapper.vm, 'increment')  
+ const spyIncrement = jest.spyOn(  
+   wrapper.vm._setupState, 'increment')
```

# COMPONENT TESTING

*Test what a component does,  
not how it does it.*

```
- // FIXME: temporary workaround
- const spyIncrement = jest.spyOn(
-   wrapper.vm._setupState, 'increment')
- wrapper.find('button').trigger('click')
- expect(spyIncrement).toHaveBeenCalled()
+ expect(wrapper.find('p').text()).toBe('Count is: 0')
+ await wrapper.find('button').trigger('click')
+ expect(wrapper.find('p').text()).toBe('Count is: 1')
```

<https://vuejs.org/guide/scaling-up/testing.html#component-testing>



# MIGRATION TO VUE 3



<https://v3-migration.vuejs.org/>

# MIGRATION TO VUE 3

<https://v3-migration.vuejs.org/>

- New Features
- Breaking Changes
- Recommendations
- Migration Build

# MIGRATION TO VUE 3

## BREAKING CHANGES

*While it looks like a lot has changed, a lot of what you know and love about Vue is still the same; but we wanted to be as thorough as possible and provide detailed explanations and examples for every documented change.*

<https://v3-migration.vuejs.org/breaking-changes/>

# NEW FRAMEWORK-LEVEL RECOMMENDATIONS

- New versions of Router, Devtools & test utils w/ Vue 3 support
- Build Toolchain: Vue CLI -> Vite
- State Management: Vuex -> Pinia
- IDE Support: Vetur -> Volar
- ...

<https://v3-migration.vuejs.org/recommendations.html>

# EASY MIGRATIONS

- Devtools
- IDE Support: Vetur -> Volar

## MANDATORY MIGRATIONS

- Vue Router
- Vuex
- Test utils
- Third party libraries  
(Vuetify, Quasar, ElementUI, etc...)

# AVOIDABLE MIGRATIONS

- Vue CLI -> Vite
- Jest -> Vitest

## MIGRATION BUILD: @vue/compat

- build of Vue 3
- configurable Vue 2 compatible behavior
- runs in Vue 2 mode by default
- usage of changed/deprecated features will emit runtime warnings

<https://v3-migration.vuejs.org/migration-build.html>



## KNOWN LIMITATIONS

- dependencies that rely on Vue 2 internal APIs or undocumented behavior.
- usage of private properties on VNodes (Vuetify, Quasar, ElementUI, ...)
- IE11 support dropped
- Server-side rendering

# UPGRADE WORKFLOW

## 1. upgrade tooling if applicable

- `vue-cli`:

```
npx @vue/cli@latest upgrade
```

- custom webpack setup:  
upgrade `vue-loader` to `^16`

## 2. update vue in package.json

```
"dependencies": {  
-   "vue": "~2.7",  
+   "vue": "^3.2.39",  
+   "@vue/compat": "^3.2.39"  
    ...  
},  
"devDependencies": {  
-   "vue-template-compiler": "~2.7"  
}
```

```
npm i -S vue@^3.2 @vue/compat@^3.2 --force  
npm uninstall vue-template-compiler --force
```

## 3.1. alias vue to @vue/compat

```
// vue.config.js
module.exports = {
  // ...
  chainWebpack: (config) => {
    config.resolve.alias.set('vue', '@vue/compat')
    // ...
  }
}
```

Webpack or Vite config on  
<https://v3-migration.vuejs.org/migration-build.html>

## 3.2. enable compat mode via Vue compiler options

```
config.module
  .rule('vue')
  .use('vue-loader')
  .tap((options) => {
    return {
      ...options,
      compilerOptions: {
        compatConfig: {
          MODE: 2
        }
      }
    }
  })
})
```

Webpack or Vite config on

<https://v3-migration.vuejs.org/migration-build.html>

## 4. if using TypeScript, update vue's typing

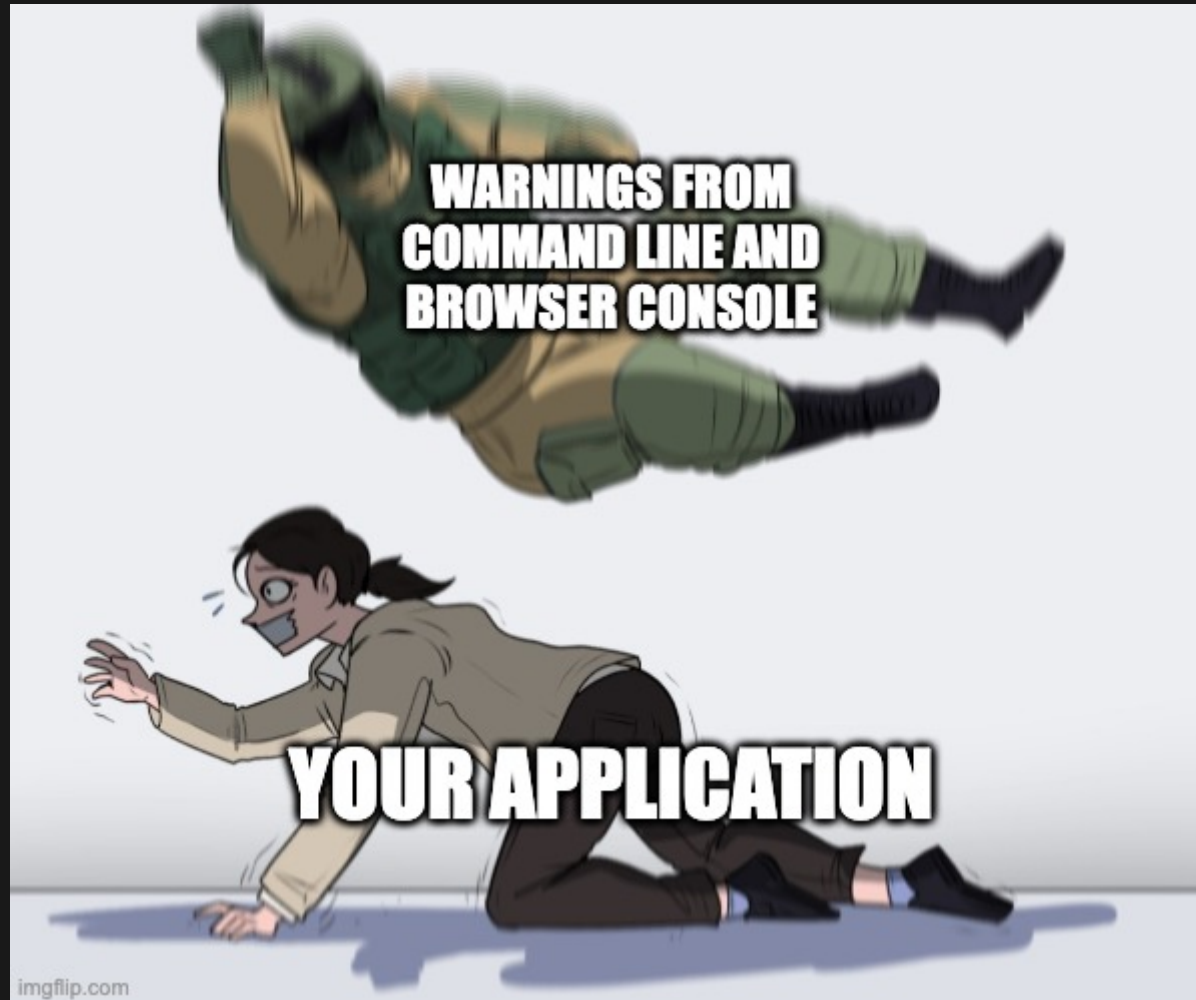
```
declare module 'vue' {  
  import { CompatVue } from '@vue/runtime-dom'  
  const Vue: CompatVue  
  export default Vue  
  export * from '@vue/runtime-dom'  
  const { configureCompat } = Vue  
  export { configureCompat }  
}
```

## 5. fix compile-time errors

When compiler warnings are gone, you can set the compiler to Vue 3 mode

```
compilerOptions: {  
  compatConfig: {  
    MODE: 3  
  }  
}
```

## 6. Run the app





## 6. Run the app

- the app should be able to run if no limitations mentioned above
- focus on fixing your own source code warnings first

## 7. Update <transition> class names

This feature does not have a runtime warning.

```
Search for `.*-enter` and `.*-leave` CSS class names
```

```
- .*-enter  
+ .*-enter-from  
  
- .*-leave  
+ .*-leave-to
```

## 8. Update app entry to use new global mounting API

```
- import Vue from 'vue'  
+ import { createApp } from 'vue'  
  import App from './App.vue'  
  import router from './router'  
  import store from './store'  
  
- Vue.config.productionTip = false
```

```
- new Vue({  
+ const app = createApp({  
  router,  
  store,  
-  render: h => h(App)  
+  ...App,  
- }).$mount('#app')  
+ })  
  
+ app.mount('#app')
```

## 9. Upgrade vuex to v4

```
npm i -S vuex@^4 --force
```

<https://vuex.vuejs.org/guide/migrating-to-4-0-from-3-x.html>

## 9. Upgrade vuex to v4

```
// src/store/index.js
- import Vue from 'vue'
- import Vuex from 'vuex'
+ import { createStore } from 'vuex'

- Vue.use(Vuex)
```

```
- export default new Vuex.Store({
+ export default createStore({
-   state: {
-     count: 0,
+   state() {
+     return {
+       count: 0,
+     };
+   },
  // ...
```

## 9. Upgrade vuex to v4

```
// src/main.js
import store from './store'

const app = createApp({
  router,
-  store,
  ...App,
})

+ app.use(store)
```

## 9. Upgrade vuex to v4

### Composition API usage

```
import { useStore } from 'vuex'  
  
const store = useStore()
```

## 10. Upgrade vue-router to v4

```
npm i -S vue-router@^4 --force
```

<https://router.vuejs.org/guide/migration/index.html>



# 10. Upgrade vue-router to v4

```
// src/router/index.js
- import Vue from 'vue'
- import VueRouter from 'vue-router'
+ import { createRouter, createWebHistory } from 'vue-router'

- Vue.use(Vuex)
```

```
const routes = [
  // ...
]

- const default new VueRouter({
+ export default createRouter({
-   mode: 'history',
-   base: process.env.BASE_URL,
+   history: createWebHistory(process.env.BASE_URL),
  routes
})
```

## 10. Upgrade vue-router to v4

```
// src/main.js
import router from './router'

const app = createApp({
-   router,
  ...App,
})

+ app.use(router)
```

# 10. Upgrade vue-router to v4

## Composition API usage

```
import { useRouter, useRoute } from 'vue-router'  
  
const router = useRouter()  
const route = useRoute()
```

11. Pick off individual warnings (if any)

Always refer to the doc.

<https://v3-migration.vuejs.org/>

12. Remove the migration build and switch to Vue 3 when all warnings are fixed.

Note you may not be able to do so if you still have dependencies that rely on Vue 2 behavior.

```
npm uninstall @vue/compat --force
```



# BROKEN TESTS!

## Upgrade testing packages

```
npm uninstall @vue/vue2-jest --force  
npm i -D @vue/vue3-jest@^29 --force  
npm i -D @vue/test-utils@^2 --force
```

```
// jest.config.js  
transform: {  
-   '^.+\\.vue$': '@vue/vue2-jest',  
+   '^.+\\.vue$': '@vue/vue3-jest',  
}
```

<https://test-utils.vuejs.org/migration/>

# VUE TEST UTILS V2 CHANGES

- `propsData` is now `props`
- no more `createLocalVue`
- `mocks` and `stubs` are now in `global`
- `findAll().at()` removed  
use `findAll()[ ]`
- ...

<https://test-utils.vuejs.org/migration/>





**MY FRESHLY  
MIGRATED PROJECT**

**ENVIRONMENT UPDATES, BREAKING CHANGES,  
BROKEN TESTS, UNSUPPORTED DEPENDENCIES,  
COMPILE/RUNTIME WARNINGS/ERRORS,**

imgflip.com

# RECAP

- Vue 2.6 + @vue/composition-api
  - Composition API (`setup ( )` only)
- Vue 2.6 → 2.7
  - Composition API
  - `<script setup>`
- Vue 2.x → 3.x
  - long term support for your codebase

# THANK YOU!

- [blog.vuejs.org](https://blog.vuejs.org) | [vuejs.org](https://vuejs.org)
- [v3-migration.vuejs.org](https://v3-migration.vuejs.org)
- [vuex.vuejs.org](https://vuex.vuejs.org) | [router.vuejs.org](https://router.vuejs.org)
- [test-utils.vuejs.org](https://test-utils.vuejs.org)
- [github.com/dennybiasiolti/vue-migration-2-to-3](https://github.com/dennybiasiolti/vue-migration-2-to-3)  
`migrate-to-vue3` branch

@dennybiasiolti



[www.dennybiasiolti.com](https://www.dennybiasiolti.com)