

# Vue.js // Berlin

## METEOR + Vue.js



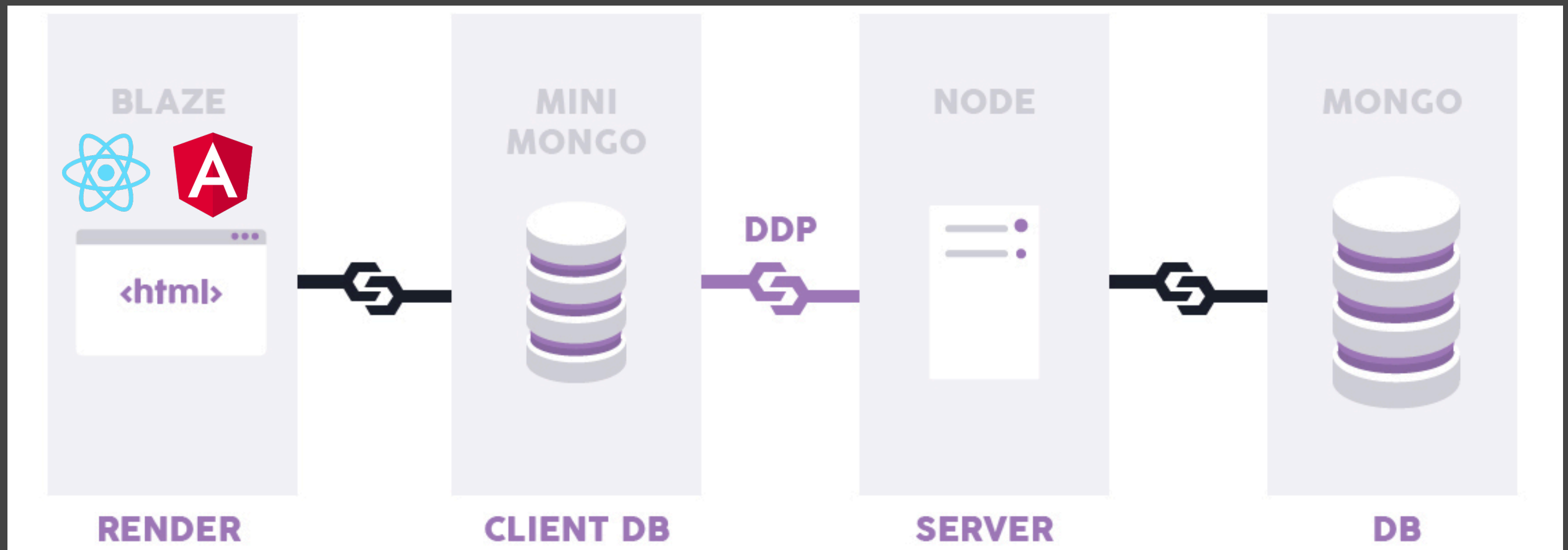
Karel Mácha

Web-Dev @ CleverShuttle 

[karel.macha@karlitos.net](mailto:karel.macha@karlitos.net)

 <https://github.com/karlitos>

# METEOR is ~~a framework~~ a platform



- Automatic real-time updates to the UI
- Database everywhere - **Minimongo**
- Network-latency compensation - **Optimistic UI**
- Built-in Apache Cordova integration
- Meteor packages - **Atmosphere** ( User management, Vue support ...)

# Why meteor ?

1. Real-time Web-Apps ( Chat, Social-media-whatever ...)

2. Easy to start with

- Installation (NIX): `curl https://install.meteor.com/ | sh`
- Create a project: `meteor create myKickAssProject`
- Start: `cd myKickAssProject && meteor`
- => App running at: `http://localhost:3000 /`

# Why not ?

- **Meteor** + MongoDB == BFF ♥
- Relational-data / Graph-like-data ?  $\Rightarrow$  **Apollo** (GraphQL)
- **Meteor** + Blaze | React | Angular == BFF ♥

??? **Meteor** + Vue ???

Enough chit-chat,  
let's see some code!

# akryum:vue-router2

```
/* /client/routes.js */

// we will not directly configure the router instance,
// but // will use a factory instead

// Import the router
import { RouterFactory } from 'meteor/akryum:vue-router2'

// Components
import Home from '/imports/ui/Home.vue'
import Forum from '/imports/ui/Forum.vue'

RouterFactory.configure(factory => {
  // Simple routes
  factory.addRoutes([
    {
      path: '/',
      name: 'home',
      component: Home,
    },
    {
      path: '/forum',
      name: 'forum',
      component: Forum,
    },
  ])
})
```

```
* /client/client.js */

// App layout
import AppLayout from '/imports/ui/AppLayout.vue';

// App start
Meteor.startup(() => {
  // Create the router instance
  const router = routerFactory.create()

  // Start the Vue app
  new Vue({
    router,
    ...AppLayout,
  }).$mount('app');
});
```



# vue-supply

- resources that can automatically be activated and deactivated when used
- injection in the **vuex** store
- components using **supply** = *consumers*
- *supply.grasp()* increments *supply.consumers* by 1
- *supply.release()* decrements *supply.consumers* by 1
- *TestResource.\$on('active', (isActive) => { // Do something })*

```
export default {
  extends: Supply,
  methods: {
    activate () {
      // Subscribe
    },
    deactivate () {
      // Unsubscribe
    },
  },
}
```



- <https://github.com/meteor-vue/vue-meteor>
- <https://github.com/Akryum/meteor-vue2-example>
- <https://forums.meteor.com/c/vue>

**METEOR**