

# Web Front-end Framework - Angular

信息学院 2016 级计算机科学与技术专业  
631607040104 刘强 指导老师：米波

**【Abstract】** Angular is a platform that makes it easy to build applications with the web. Angular combines declarative templates, dependency injection, end to end tooling, and integrated best practices to solve development challenges. Angular empowers developers to build applications that live on the web, mobile, or the desktop.

**【Key Words】** Angular; Vue; Front-end framework; Web

## 1. Introduction

The front end is the front-end part of the website, which runs on the PC side, the mobile side and the like to display the web page browsed by the user. With the development of Internet technology, HTML5, CSS3, front-end framework applications, cross-platform responsive web design can adapt to a variety of screen resolutions, perfect dynamic design, giving users a very high user experience.

Angular is a platform and framework for building client applications in HTML and TypeScript. Angular is written in TypeScript. It implements core and optional functionality as a set of TypeScript libraries that you import into your apps.

## 2. Front-end back-end separation

To understand the concept of separation between front and rear, we must first understand what is the separation between the front and the back.

In the traditional way, servlet, springmvc, struts, tomcat will run Java code in the background, then generate all the html code in the background, and then transfer the html code to the browser through the http protocol.

The problem:

(1) When running in the background, if you need to connect a lot of databases to find out the required data, then the process of preparing html will be slower, and the generated html is relatively large, and the user may feel slower.

(2) The degree of joint coupling developed by the front-end and back-end is also very high, and sometimes it is necessary to wait for each other, so that the development efficiency of both parties is reduced.

So what is the concept of separation between front and back? The content of an html page can be thought of as simply containing the data part and not including the data part. So first prepare a html that doesn't contain data and pass it to the browser. This speed itself will be very fast, because there is no database operation part that takes the most time. Then through Ajax technology, just get "pure data" from the server, and then display the pure data

on html.

The benefits:

(1) Even if the back-end database takes a lot of time, the user experience is better than the previous one, because the user will see some of the pages first, and then see the data after a while, waiting for the experience to be circled in a blank page.

(2) The back end only provides data, so the development coupling degree of the front and back ends is reduced a lot, and the overall development efficiency can be greatly improved.

### 3. My understanding

I have used Angular and Vue to develop websites. The feeling for me is that Vue is like a young man, very energetic, and then paired with ElementUI. To make his youthful vitality play, Vue makes many troubles very simple. And Angular is more like a very mature middle-aged person, and everything is very stable. Angular has been two years since its release in 2016. With the constant iteration, its advantages over other frameworks have become more and more important. I've been working on some projects with Angular for a while, and I'm talking about Angular from my personal journey:

- (1) Typescript.
- (2) A complete set of frameworks.
- (3) Angular's advantages.

#### 3.1 *Typescript*

Angular introduces a strongly typed language, TypeScript, which makes it very easy for Java programmers to turn front-end. Java programmers can develop front-ends with Java's fully object-oriented thinking. Angular's perfect support for TypeScript, with VSCode the sharp weapon itself is a great improvement for development efficiency or project quality. As we all know, although TypeScript has been proposed a few years before Angular's release, it has not been popular due to the lack of some framework/tools. The later version of Angular is obviously driving the TypeScript. I think many developers are just like me, learning Typescript from Angular. But from the developer's point of view, it may just want to learn a new framework. As a result, when I first started looking at the documentation, I found that I had to learn a new "language" (even if it is very similar to JavaScript), which definitely made the developer the threshold has increased. The official documentation is very well written, the official website (<https://angular.io/>), and the Chinese official website (<https://angular.cn/>), which allows developers with poor English to learn very quickly.

#### 3.2 *A complete set of frameworks*

Angular is a complete web front-end development framework, which contains a lot of components, unlike Vue, only the logical framework of the front-end View layer, you need to install the axios plug-in even for asynchronous requests. Angular itself integrates classes for asynchronous requests and can be used directly.

Learning Angular, to understand the basic data binding, you also need to learn to use Rx.js to handle a variety of asynchronous events, use modules to modularize your application, use service to encapsulate business logic to improve reusability and dependency injection.

Wait. You need to understand these concepts when learning Angular, and follow some of these conventions in development. From an engineering point of view, Angular's basic features such as component, module, and service are very good practices. When building large applications, these features of Angular can make your application very maintainable and expandable. Another benefit of these conventions is that if the project has newcomers to join, Angular's "stipulations" can allow a person who has no prior experience with Angular to write quality code.

Many lightweight business scenarios may not require an overly huge framework like Angular. In contrast, Vue/React, as a View's Library, can be easily integrated into projects. The concept of a progressive framework proposed by Vue is also easier for many people to accept and easier to use. So use Angular instead of the size and difficulty of the project.

### *3.1 Angular's advantages*

Some large projects take into account stability and scalability, and will consider Angular at the beginning of the project. Developers will also benefit from knowing and being proficient in using Angular. I think a lot of moderate/heavy developers who have used Angular will find it worthwhile to use Angular and will love this framework.

But as a framework, Angular also takes time to understand and learn. It may not be as fast as React/Vue. As a result, Angular may also inadvertently label the "Applicable to large projects." This also limits the promotion of Angular to a certain extent. But what can't be changed is that Angular is a framework worth learning to use.

## **4. Conclusion**

Angular is a very worthwhile web front-end framework that not only makes programmers logically clear, improving the efficiency of development, but also provides a complete set of plug-ins that allow you to quickly build applications.

## **Bibliography (References)**