# Vue源码解读笔记

[视频](https://coding.imooc.com/learn/list/228.html) （18513179925|a1111111）

【需了解】

* Vue 使用了[Flow](https://flow.org/en/)（libdef的概念）做静态语法类型检查。
* Vue源码基于[Rollup](https://www.rollupjs.com/)构建，相关配置在scripts目录下。（webpack vs rollup vs gulp）
* Vue最终也是发布在了npm上。

【Vue.js源码目录】

|  |
| --- |
| --flow类型检查，有vue很多自定义类型  --src 源码目录  |\_\_compiler 编译器  |\_\_core 核心  |\_\_platforms支持的平台web、weex  |\_\_server 与服务器相关的  |\_\_sfc  |\_\_shared  --scripts脚本目录，对应所有的构建任务  --dist构建目录  --examples示例目录 |

【版本对比】

* Runtime Only

在使用此版本Vue.js时，通常需要借助如webpack的vue-loader工具把.vue文件编译成JavaScript，因为是在编译阶段做的，所以它只包含运行时的Vue.js代码，因此代码体积也会更轻量。

* Runtime + Compiler

在Vue.js2.0中，最终渲染都是通过render函数来实现的。若使用Vue的template属性，需要在客户端编译template为render函数。

显然，编译过程对性能有一定损耗，所以通常推荐使用Runtime-only版本的Vue.js.

[【入口】

platforms/web/entry-runtime-with-compiler.js

src/core/index.js

【Flow学习笔记】

* 安装
* 使用

1. 根目录使用flow init创建.flowconfig文件进行配置，改配置文件告诉flow后台程序进行flow语法检查的根（入口），自此项目级flow已启用。
2. 使用flow status指令启动后台程序对所有flow files进行错误检查。该程序会持续运行以检查新增代码中的错误。

使用flow指令默认执行flow status；

无论执行flow status多少次，任何时间flow status只会有一个后台程序在执行；

使用flow stop停止后台程序。

1. 使用// @flow或/\* @flow \*/将一个js文件标记为flow files。一般情况下，flow后台程序只会检查带有此标识的文件。除非调用flow check --all指令执行全局检查。
2. 编写flow代码，如参数类型校验、返回值类型校验，flow后台程序会自动进行类型检查
3. 检查编码。首次执行flow即启动flow后台程序对文件进行增量监督，再次执行flow指令最新的flow检查结果立即呈现。

* 类型注释

flow具有卓越的类型推断能力，因此，使用类型注释是flow辅助编码的一大亮点。

1. Primitive Types

MDN官网定义JS的基本类型有：Booleans、Strings、Numbers、null、undefined、Symbols。

对于字面量值（ture, “hello”, 3.14, null, undefined）用小写（boolean, string, number）;

对于包装对象（new Boolean(false)...）用大写（Boolean, String, Number）

* **boolean**：function acceptsBoolean(value: boolean) 接受字面量值（true、false、if语句及a===b等条件表达式隐式转换的结果），其他类型只能通过Boolean()转换；
* **number**：42、3.14、NaN、Infinity；
* **string**：接受“”、+拼接的string和number类型，其他类型必须转换（如：String({})、[].toString()、JSON.stringfy({})）；
* **null**（JS中的null）、**void**（JS中的undefined）；
* ?type：用于定义可选参数可能的类型，如：?string、?number，除了为type既定类型，还可以为null、void；
* 对象可选属性：{ propertyName?: string }，除type既定类型还可为void或忽略，不能为null；
* 函数可选参数：同上；
* 函数默认参数：function method(value: string = "default")，同上。
* symbol：Symbol()、Symbol.isConcat；

1. Literal Types字面量类型
2. Mixed Types
3. Any Types
4. Maybe Types