

02

OPEN ORIENTED

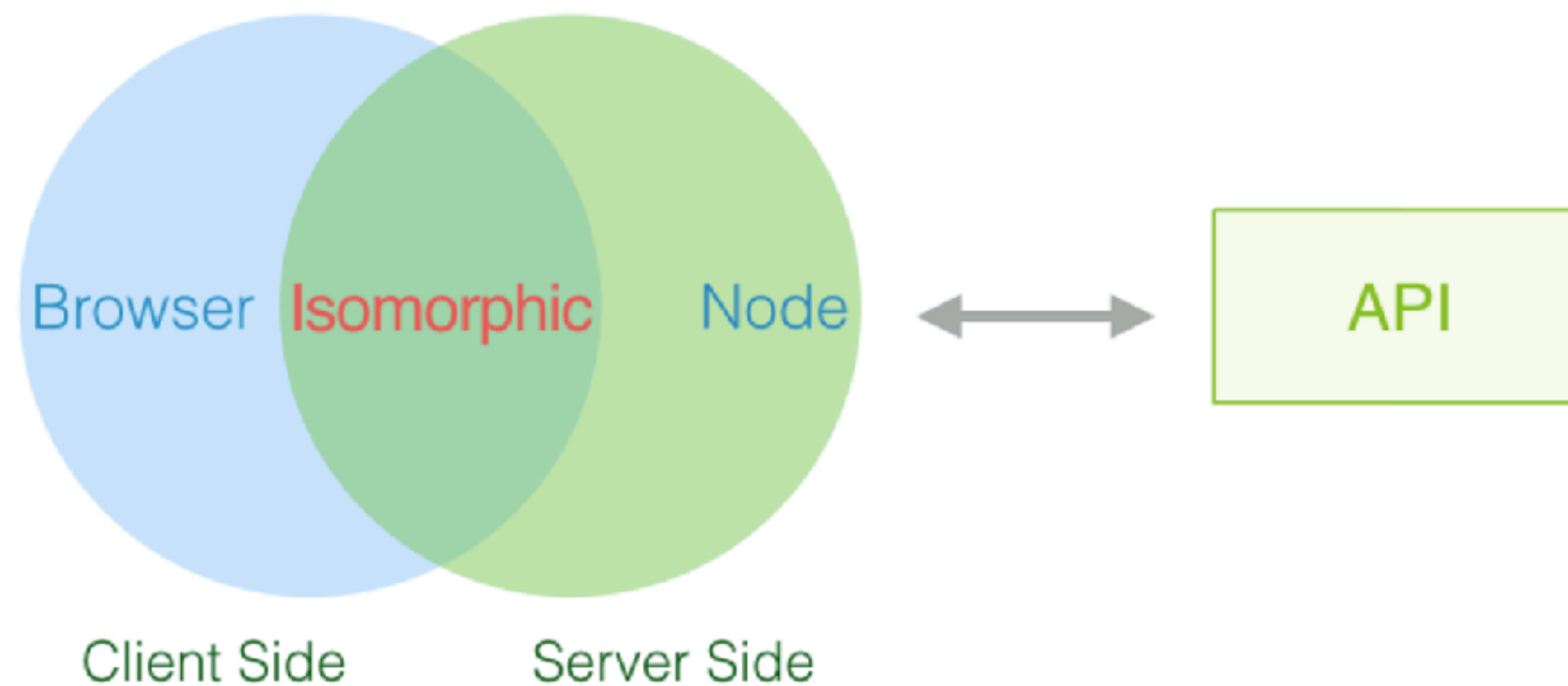
凹凸实验室

Vue.js 服务器端渲染(SSR)

什么是服务器端渲染(SSR)？

服务端渲染（SSR, Server-Side Render）是指通过服务端拉取首屏数据并渲染页面结构，快速响应到客户端，从而降低首屏加载时间、有效降低页面白屏延迟，更好的解决目前前后端分离采用 MV* 框架的 SEO 问题。

Vue.js 服务器端渲染是使用 Vue 构建客户端应用程序大部分代码都可以在服务器和客户端上运行。

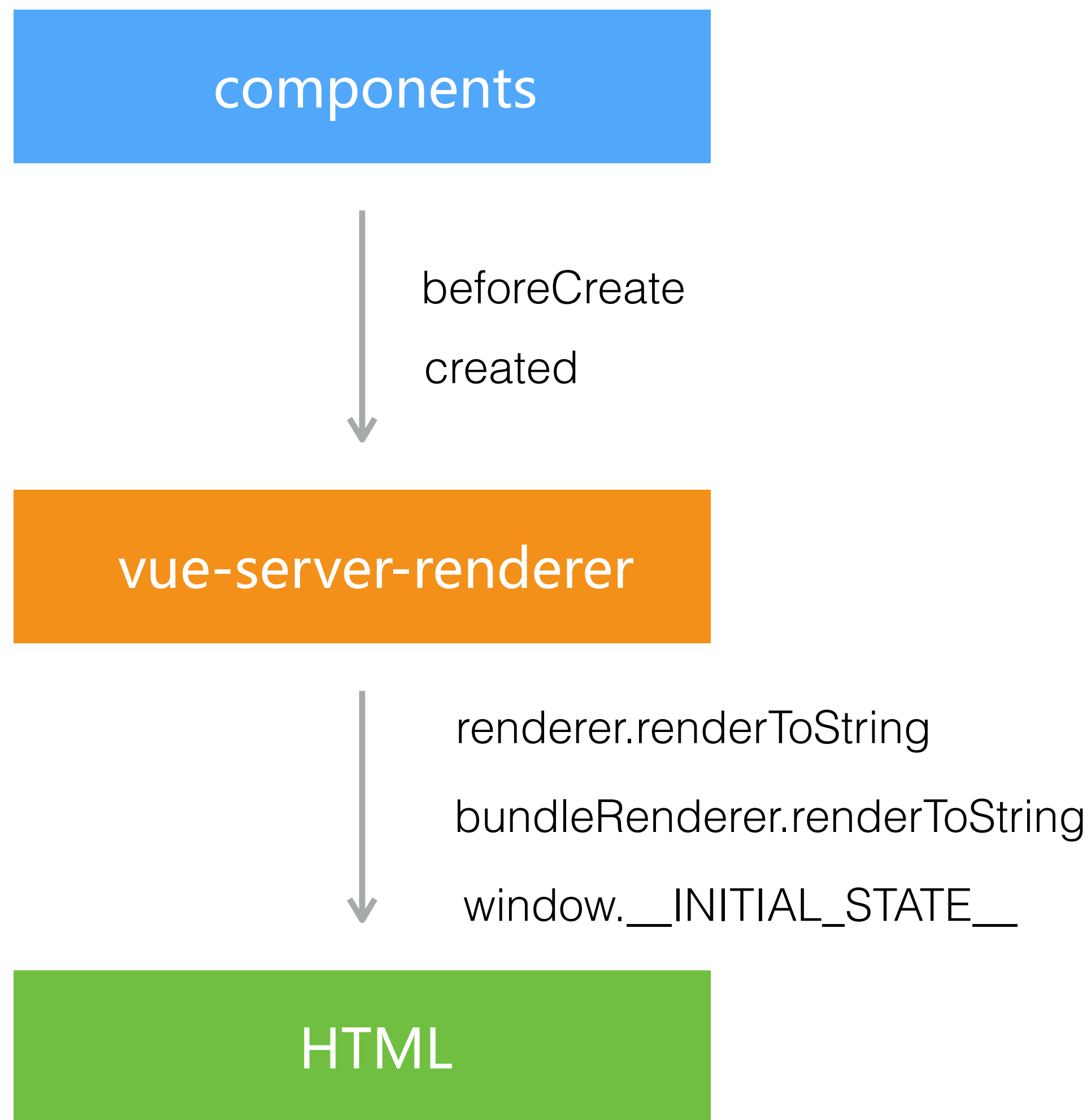


使用服务器端渲染(SSR)好处

与传统 SPA (Single-Page Application - 单页应用程序) 相比, 服务器端渲染 (SSR) 的优势主要在于:

- 更好的SEO, 有利于搜索引擎爬虫抓取直出页面
- 直出保证用户最先看到基本的内容, 减少首屏加载时间
- 共用前端代码, 节省开发时间
- 提高兼容性, vue ie9才支持, 而直出兼容性更好

Vue 如何服务端渲染



具体原理参考：<https://github.com/vuejs/vue/blob/dev/packages/vue-server-renderer/build.js>

Class: Renderer

```
renderer = createRenderer([rendererOptions])
```

```
renderer.renderToString(vm[, context], callback)
```

```
renderer.renderToStream(vm[, context])
```

Class: BundleRenderer

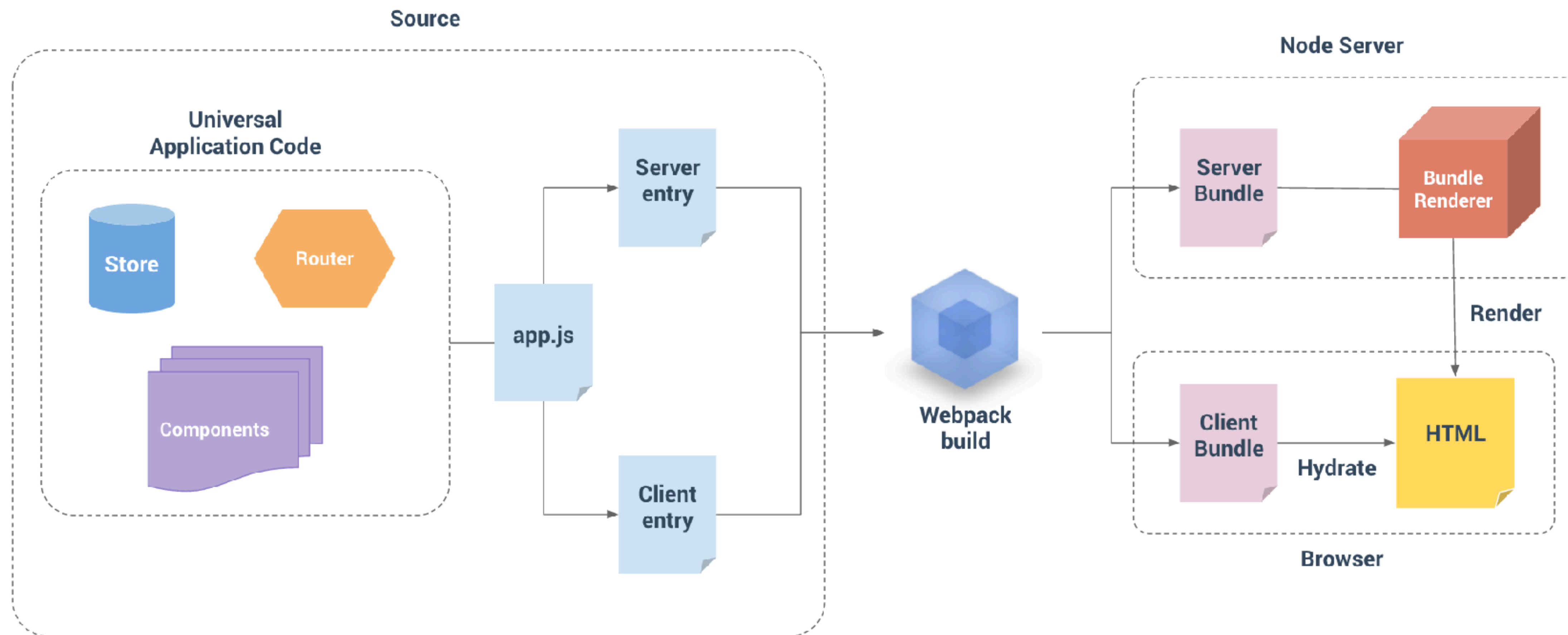
```
bundleRenderer = createBundleRenderer(bundle[, options])
```

```
bundleRenderer.renderToString([context, ]callback)
```

```
bundleRenderer.renderToStream([context])
```

简单例子理解 Vue 直出原理

Vue 服务端渲染架构



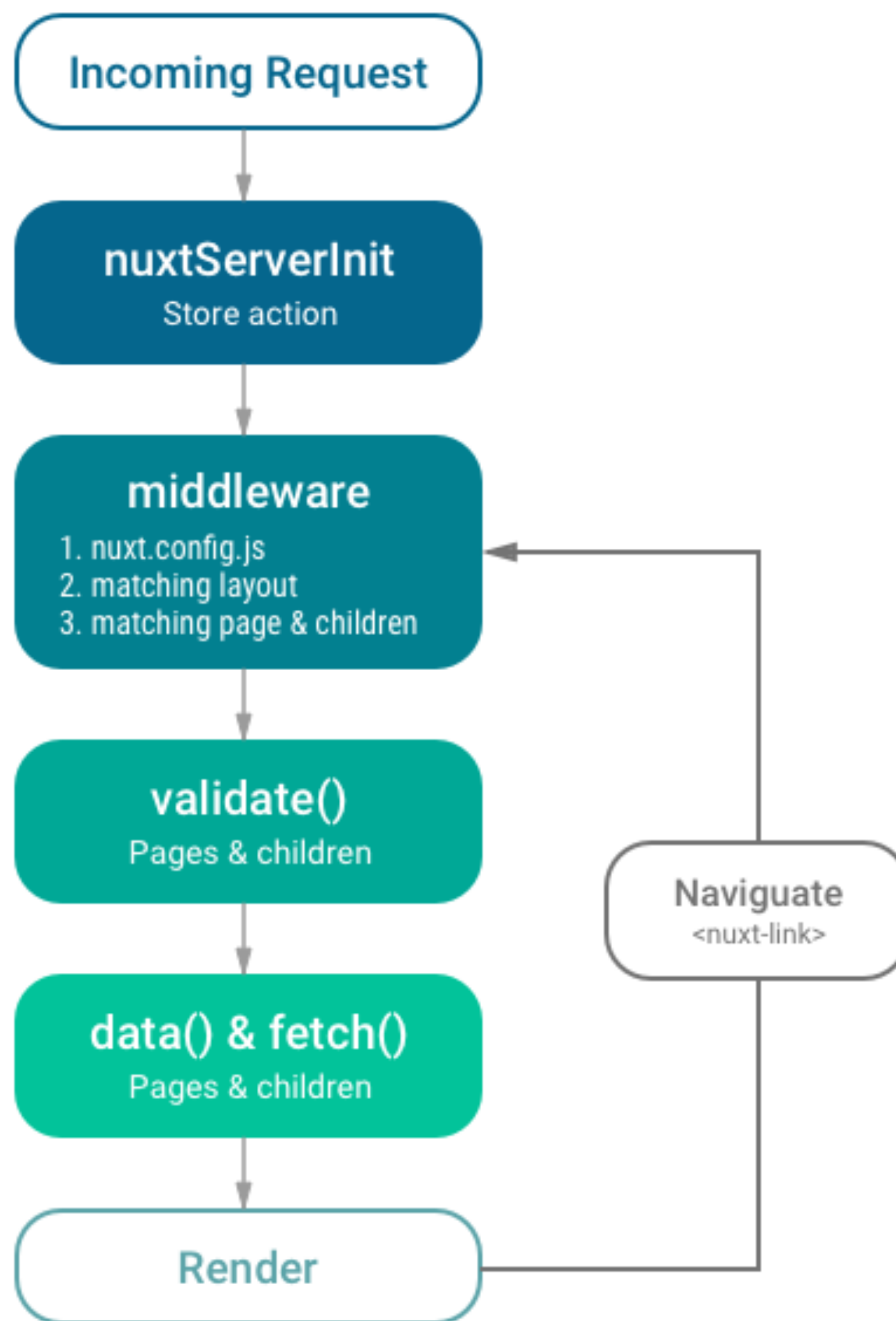
官方示例 vue-hackernews-2.0



Vue.js 通用应用框架

- 自动代码分层
- 服务端渲染
- 强大的路由功能，支持异步数据
- 静态文件服务
- ES6/ES7 语法支持
- 打包和压缩 JS 和 CSS
- HTML头部标签管理
- 本地开发支持热加载
- 集成ESLint
- 支持各种样式预处理器： SASS、LESS、 Stylus等等

Nuxt.js 渲染过程



- [Vue.js 服务器端渲染指南](#)
- [Vue.js 通用应用框架 Nuxt](#)
- [精读前后端渲染之争](#)
- [实测 Vue SSR 的渲染性能：避开 20 倍耗时](#)

