

# 第二部分

# 异步和性能

单 业 译

---

# 序

多年以来，老板信任我，让我负责面试。如果我们在寻找一个具备 JavaScript 技能的雇员，我的第一个问题就是……好吧，其实就是问问面试者是否需要上厕所或者喝点什么，因为舒适很重要。不过一旦面试者完成了这个液体输入或输出的过程，我就要开始判断他是否了解 JavaScript，还是只了解 jQuery。

不是说 jQuery 哪里不好。jQuery 让你不需要真正了解 JavaScript 就可以做很多事情，这是功能，而不是 bug。但是，如果工作内容需要 JavaScript 性能和维护方面的高级技能，那么应聘的人就应了解如何把像 jQuery 这样的库组合起来。你得能够像这些库一样驾驭 JavaScript 的核心。

如果要整体了解一个人的核心 JavaScript 技能，我最感兴趣的是他们会如何使用闭包（你已经读过这一系列中的《你不知道的 JavaScript（上卷）》了吧？）以及如何充分利用异步，这一点把我们引向了这本书。

首先第一道菜是回调，这是异步编程的面包和黄油（基础）。当然，靠面包和黄油并不足以完成令人非常满意的大餐，接下来的课程就是美味异常的 Promise！

如果还不了解 Promise，现在正是学习的时候。Promise 现在已经是 JavaScript 和 DOM 提供异步返回值的正式方法。所有未来的异步 DOM API 都会使用它们，而且很多已经这么做了，所以做好准备吧！写作本文的时候，多数主流浏览器中已经发布了 Promise，IE 很快也会提供支持。享用 Promise 之后，希望你的胃里还有空间留给下一道美食——生成器。

没有大张旗鼓的宣传，生成器就已经悄悄进入了 Chrome 和 Firefox 的稳定版本，这是因为，坦白地说，它们的复杂性要高于其趣味性。或者说，在看到它们与 Promise 合作之前，我一直都是这么认为的。可之后呢，它们成了提高可读性和可维护性的重要工具。

甜品是……嗯，我不想破坏了惊喜，但是，准备好展望 JavaScript 的未来吧！这本书涵盖的功能会让你对并发和异步有越来越多的控制。

好吧，我不再妨碍你享用这本书了——让精彩继续吧！如果在看此序之前你已经阅读了本书的部分内容，那么请给你自己加 10 个异步分！这 10 分是你应得的。

——Jake Archibald (<http://jakearchibald.com>, @jaffathecake),  
Google Chrome 开发大使