

第29讲 | 如何选择合适的开发语言？

2018-07-31 蔡能

从0开始学游戏开发

[进入课程 >](#)



讲述：蔡能

时长 08:05 大小 3.71M



有许多编程语言可以用来开发服务器端。一些语言对于网络开发有先天优势，一些语言没有先天优势，但是能完成任务，而有一些语言，则不太适合。今天，我就来具体讲一讲这些语言来开发网络服务的优劣势。

你了解这些编程语言吗？

C/C++ 是最标准的开发语言，速度快，并发性能高，能最大程度利用机器资源完成任务。现在 C/C++ 层面拥有无数网络开发库和 SDK，知名的有 ACE、Boost/Asio、ICE 等等。但是缺点是，开发效率不比别的语言来得高，出错后常常只有熟练的程序员才能定位问题并且作出修复。

Go是 2009 年新出现的语言。Go 天生适合编写网络程序。它也是一种系统级的语言，可以直接编译为执行文件，当然由于封装了大量的语法糖，所以编译出来的文件会更大，它天生支持并发操作，所以很多时候你不需要像 C/C++ 一样手工去处理。缺点是，Go 语言仍然存在许多语法方面的坑，你可以去 <https://studygolang.com/> 学习最新的资料。

Java是公认的编写网络服务的第一大语言。在运行性能和开发效率上，有很好的折中效果。Java 拥有众多的中间件和开发库，调试方便，一般的运维人员也有极为广泛可用的第三方维护工具可以使用。缺点是，Java 的运行效率虽然有了质的飞跃，但因为中间隔了一层虚拟机，所以仍然比不上系统开发语言编写的软件。另外，Java 的发布和部署需要众多的依赖包和库，软件体积庞大也是其重要弊病。

如果深入理解，**Python**、**Ruby**这两种语言的相似程度以及对系统的支持程度，可以用 C 和 C++ 的相似程度来相比。你或许会很疑惑，毕竟 Python 和 Ruby 的语法几乎不一样，Python 需要格式化源代码，而 Ruby 并不需要；Python 更严谨，Ruby 更开放；Python 用户更多，Ruby 用户更少。

不可否认的是，两种语言编写网络程序都非常方便，也非常高效。两种语言都可以在 100 行内编写出一个简单的、完全可以直接使用的网络服务器程序。但是这两种语言的弊病也很明显，那就是速度不够快。

比之 Java，或许运行效率更慢一点，但由于目前机器硬件水平的提升，软件效率不足的缺点一部分已经被硬件所弥补，但是仍然不能否认，Python、Ruby 语言适合 IO 密集型开发，而不适合计算密集型的开发。

Python 的书籍比 Ruby 多好几倍，然而你如果仔细去看的话就会发现，Ruby 的书籍质量明显比 Python 高几个等级，所以如果要看好的脚本语言的书籍，Ruby 相关的书籍是首选，我这里推荐一本 [Programming in Ruby](#)，有兴趣的话可以找来看看。

Node.js从前端语言变成后端语言，让编程界眼前为之一亮。随后的发展大家也有目共睹，Node.js 由于使用 JavaScript 语言语法，所以我们一般采用事件驱动的形式，以及非阻塞的模型来进行网络开发。因为这些特点，它非常适合做分布式的、数据密集型的工作。但是缺点也很明显，Node.js 是单线程，无法很好地使用多核 CPU，这个问题在 Python、Ruby 语言中也很明显。

或许你没有听说过**Erlang**这种语言，这种语言最初是由爱立信开发的。它的初衷是让程序员应对大规模并发、分布式、软件实时并行系统进行编程。最早期的版本在 80 年代就出现了，但是一直到 1998 年才开源。

Erlang 也不是系统语言，它有自己的虚拟机和解释器。当然和 Java 一样，Erlang 也支持将代码编译为 Native Code 来运行。Erlang 的缺点就是类型问题，它并非强类型语言。由于是事件编程，所以导致会在运行时无法匹配类型而出错，不过这些问题可以使用规范的编程方法来规避。

这么多种编程语言，整合起来看，大致可以把他们分为三类。

系统级编程语言，诸如汇编、C、C++。这种编程语言执行效率高，并发量也比较高，作为编写网络服务的第一语言，一台服务器就能支撑许多人。缺点是开发效率不够高，需要几年以上经验的程序员才能搞定。

专门为网络服务器开发的语言，诸如 Go、Erlang。这种语言编写高并发和开发效率都不是问题，有很好的折中效果。缺点就是语言比较新，有许多的坑等着后来的程序员去填，而且语言、语法等系统机制要随着进一步的发展才能稳定下来。

解释型脚本语言，诸如 Python、Ruby。这类语言的开发效率非常高效，在现在的服务器硬件上，也能支撑不少用户，但是唯一的缺点是，运行效率低下。虽然也有解决方案，但仍然不能对抗高性能的系统编程语言和专业网络开发的语言。

如何选择一种合适的语言来编写网络服务？

Web 服务

现在有一种流行的说法叫前后端分离。对于编写 C/S 结构的程序员，听到这种说法应该会比较蒙，客户端和服务端难道不是本来就分离的吗？

很长的一段时间里，在 Web 的世界中，前后端都是混合在一起编写的，比如 PHP 的方式，只有用到 Ajax 交互的时候，才需要用到后端的代码。但是前后端一分离，后台就需要做更多的工作了，当然前端的工作也不会变少。

编写 Web 服务，需要 HTTP 和 HTTPS 的服务体系，那么在这种情况下，使用 nginx、Apache 作为静态页面路由，Java、Tomcat、Python、Ruby 等脚本语言就有了用武之

地。因为页面只需要使用 JSON 交互即可。

所以，编写 Web 服务，我们可以选择 Java、Python、Ruby。但是如果公司财力物力有限，再考虑到招人成本的问题，次选也可以是 Java 语言，第一是写 Java 的人够多，第二是 Java 成熟的类库够多，因此，一旦出问题，有解决经验的人也比较多。

Socket 服务

传统 TCP/IP 和 UDP 服务，或者最近的 WebSocket 等，都需要快速响应和并发操作，在这种情况下，系统级编程语言和网络编程语言就可以派上用场了。

如果公司的项目需要更快更高效，并且财力也允许，那么选择 C、C++、Go、Erlang 等编程语言未尝不是一种选择。当然 Java 也能很好地提供服务，但是从业务上来讲，既然选择了 Socket 服务模式，那么就必然是对并发量有一定的要求，所以选择上述这些语言更合适。

混合模式

这类业务，既有 HTTP/HTTPS 的服务，也有 Socket 服务，那么如何平衡两者之间的语言成本？如何平衡程序员之间技术栈的问题呢？

如果要做一款短期内必须上线的产品，我建议选择成熟的、有大量解决方案的，开发人员不短缺的语言，比如 Java；或者能快速做出原型的语言，比如服务器专有语言 Go。如果是长期发展的产品，并不那么着急成型，那么选择稳定成熟的，人员素质高的语言，比如 Python、Java 等。

至于平衡技术栈的问题，首先要选择网上有众多解决方案的语言，其次是找成熟的语言，比如 Python、Java、Ruby。如果针对某种特殊的产品，比如并发要求特别高的，那么只有选择系统语言或者专门的语言，比如 Go、C++ 等。

看到这里，你是不是觉得 Java 语言是一种万能药，或者是银弹？错了，这个世界上没有银弹。Java 虽然有其独特的优势，但是其被人诟病的地方，也是有不少的。

第一点莫过于速度。就算拥有 JIT 编译，总体速度仍然比不上 C/C++，但是事实上这些因素综合考虑并不算特别大的弊病，因为硬件资源提升后，速度这些问题已经可以“得过且过”了。

那么从语言本身来看，如果说 C/C++ 语言本身的弊病是因为系统平台导致的，那么 Java 语言的弊病就是因为继承自 C++，却没有做更彻底的改革而导致的。

我随便举一个例子，比如说 switch case 判断语句，硬生生地从 C/C++ 处直接继承了下来，因为 C/C++ 只允许使用 int、enum（其实是 int）、char（提升为 int）作为判断类型，而 Java 也是直接将这套规范继承了下来。

再比如，在 Java 里面，异常检查也是一个痛苦的根源，程序员不得不写下无数 try catch 语句以使得将捕获的异常，转变为运行时的异常，然后再将之抛出去，这样一来，使用 Java 编写的 API 将缺少灵活和扩展性。

那如果选择了 Python 或者 Ruby 等脚本语言进行开发，却需要大量高并发的操作该怎么办呢？我们可以选择多进程（不是多线程）编程的方式进行开发，代码尽量简洁、高效，一个进程兼顾一个任务，进程之间的通信方式要尽量高效、简洁，比如可以使用自定义的队列等方式。

小结

学完这一节，你应该对使用各种编程语言来编写网络服务有了一个更深的了解。我主要讲了以下几个内容。

编程语言可以大致分为三类，系统级编程语言、专为网络服务器开发的编程语言和解释型脚本语言。

在编写网络服务的时候，可以根据要编写的是 Web 服务、Socket 服务，还是混合模式，来选择合适的编程语言。

给你留一个小问题吧。

如果让你来使用 C/C++ 粘合 Lua 脚本来编写网络服务器，你会怎么设计这个程序框架？

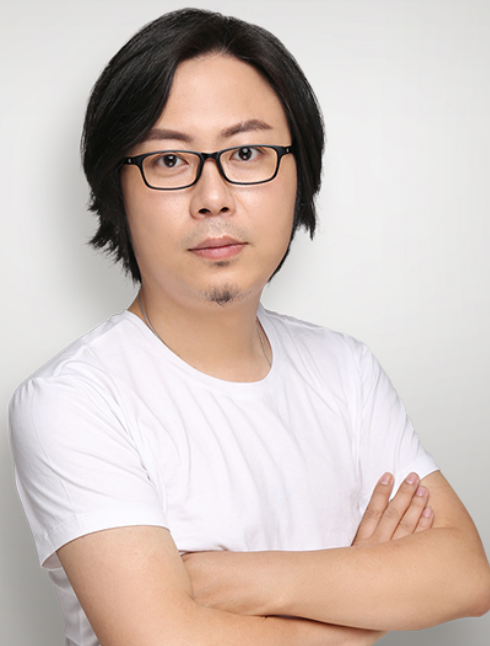
欢迎留言说出你的看法。我在下一节的挑战中等你！

从0开始学游戏开发

你的游戏开发入门第一课

蔡能

原网易游戏引擎架构师
资深游戏底层技术专家



新版升级：点击「 请朋友读」，10位好友免费读，邀请订阅更有**现金**奖励。

© 版权归极客邦科技所有，未经许可不得传播售卖。页面已增加防盗追踪，如有侵权极客邦将依法追究其法律责任。

上一篇 第28讲 | 热点剖析（七）：谈谈微信小游戏的成功点

下一篇 第30讲 | 如何定制合适的开发协议？

精选留言 (2)

写留言



栗芳凯

2018-07-31

2

Java7中，switch的参数可以是String类型了

展开



ithunter

2018-10-27

思考题：用c++写底层如socket，用lua写业务

展开

