

老男孩 IT 教育

GO 高并发服务端开发

课程介绍

更新时间：2017/3/16

课程版本：v1.0

课程周期：4 个月

适用人群：有一年开发经验

就业薪资：15K+

课程介绍

Go 作为最近几年最火的服务端开发语言，因为其学习门槛低、开发效率高的特性，迅速的占领了各大公司的服务端开发领域。相比其他服务端开发语言，GO 具有如下巨大优势：

1. 部署简单。Go 编译生成的是一个静态可执行文件，除了 glibc 外没有其他外部依赖。这让部署变得异常方便：目标机器上只需要一个基础的系统和必要的管理、监控工具，完全不需要操心应用所需的各种包、库的依赖关系，大大减轻了维护的负担。
2. 并发行好。GO 天生为高并发而生，Goroutine 和 channel 使得编写高并发的服务端软件变得相当容易，很多情况下完全不需要考虑锁机制以及由此带来的各种问题。
3. 性能优异。根据国际相关权威机构测试，GO 的性能只比 C/C++ 减少了 10% 左右。相对其他脚本（python/php），性能具有巨大的优势。

可以说，Go 语言是服务端开发的未来！

附 2016 年 TIOBE 编程语言排行榜:

编程语言排行榜 TOP20 榜单

Oct 2016	Oct 2015	Change	Programming Language	Ratings	Change
1	1		Java	18.799%	-0.74%
2	2		C	9.835%	-6.35%
3	3		C++	5.797%	+0.05%
4	4		C#	4.367%	-0.46%
5	5		Python	3.775%	-0.74%
6	8	▲	JavaScript	2.751%	+0.46%
7	6	▼	PHP	2.741%	+0.18%
8	7	▼	Visual Basic .NET	2.660%	+0.20%
9	9		Perl	2.495%	+0.25%
10	14	▲	Objective-C	2.263%	+0.84%
11	12	▲	Assembly language	2.232%	+0.66%
12	15	▲	Swift	2.004%	+0.73%
13	10	▼	Ruby	2.001%	+0.18%
14	13	▼	Visual Basic	1.987%	+0.47%
15	11	▼	Delphi/Object Pascal	1.875%	+0.24%
16	65	▲	Go	1.809%	+1.67%
17	32	▲	Groovy	1.769%	+1.19%
18	20	▲	R	1.741%	+0.75%
19	17	▼	MATLAB	1.619%	+0.46%
20	18	▼	PL/SQL	1.531%	+0.46%

Go 诞生以来，在开源社区的一些成功项目摘录如下：

1. Docker

到现在，Docker 几乎是 Go 再难找到也难以复制的一个成功案例。Docker 项目在 2014 年 9 月份就拿到了 C 轮 4000 万美元融资，版本迭代速度超快，目前从 GitHub 看到已有 78 个版本，而它仅仅是再 2013 年初才正式开始的一个项目而已。目前，国内 Docker 技术推广也进行的如火如荼，比如 [Docker 中文社区](#)，CSDN 也建立了 [Docker 专区](#)。CSDN CODE 也将在近期与 Docker 中文社区合作，推出 Docker 技术文章翻译活动，届时也请大家多多关注，及时关注与参与。

Docker 团队之所以喜欢用 Go 语言，主要是 Go 具有强大的标准库、全开发环境、跨平台构建的能力。

GitHub 托管地址：<https://github.com/docker/docker>

2. Kubernetes

Kubernetes 是 2014 年夏天 Google 推出的 Kubernetes，基于 Docker，其目的是让用户通过 Kubernetes 集群来进行云端容器集群的管理，而无需用户进行复杂的设置工作。系统会自动选取合适的工作节点来执行具体的容器集群调度处理工作。其核心概念是 Container Pod（容器仓）。

GitHub 托管地址：<https://github.com/GoogleCloudPlatform/kubernetes>

3. Etcd & Fleet

etcd 是由 CoreOS 开发并维护键值存储系统，它使用 Go 语言编写，并通过 Raft 一致性算法处理日志复制以保证强一致性。目前，Google 的容器集群管理系统 Kubernetes、开源 PaaS 平台 Cloud Foundry 和 CoreOS 的 Fleet 都广泛使用了 etcd。详情，可了解《Etcd：用于服务发现的键值存储系统》。Fleet 则是一个分布式的初始化系统。它们之所以选择使用 Go 语言，则是因为 Go 语言对跨平台的良好支持，以及其背后的强大社区。

GitHub 托管地址：<https://github.com/coreos/etcd>

4. Deis

Deis 是一个基于 Docker 和 CoreOS 的开源 PaaS 平台，旨在让部署和管理服务器上的应用变得轻松容易。它可以运行在 AWS、GCE 以及 Openstack 平台下。详情可了解 《Deis v1.0 正式发布！》。

GitHub 托管地址：<https://github.com/deis/deis>

5. Flynn

Flynn 是一个使用 Go 语言编写的开源 PaaS 平台，可自动构建部署任何应用到 Docker 容器集群上运行。Flynn 项目受到 Y Combinator 的支持，目前仍在开发中，被称为是下一代的开源 PaaS 平台。

GitHub 托管地址：<https://github.com/flynn/flynn>

6. Lime

相对上面的几款 Go 语言在云端和服务端之外，Lime 则显得比较特殊。Lime，则是一款用 Go 语言写的桌面编辑器程序，被看做是著名编辑器 Sublime Text 的开源实现。

GitHub 托管地址：<https://github.com/limetext/lime>

7. Revel

Revel 是一个高生产力的 Go 语言 Web 框架。Revel 框架支持热编译，当编辑、保存和刷新源码时，Revel 会自动编译代码和模板；全栈特性，支持路由、参数解析、缓存、测试、国际化等功能。

GitHub 托管地址：<https://github.com/revel/revel>

8. InfluxDB

一个 Go 语言编写的开源分布式的时序、事件和指标数据库，无需外部依赖。其设计目标是实现分布式和水平伸缩扩展。

GitHub 托管地址：<https://github.com/influxdb/influxdb>

9. Syncthing

一款用 Go 语言编写的开源云存储和同步服务工具，用户的数据将由自己完全控制，所有的通信全都加密，每个访问节点都用加密证书验证。该项目被认为是 Dropbox 和 BitTorrent Sync 的开源替代，详情 Syncthing 之所以选择 Go 语言，也是出于跨平台考虑。

GitHub 托管地址: <https://github.com/syncthing/syncthing>

10. Gogs

Gogs 则是一款由国人无闻 (GitHub) 开发的自助 Git 服务项目。Gogs 的目标是打造一个最简单、最快速和最轻松的方式搭建自助 Git 服务。据作者称，之所以选择使用 Go 语言开发，就是 Go 允许 Gogs 可以通过独立的二进制分发，且对跨平台有良好支持。

GitHub 托管地址: <https://github.com/gogits/gogs>

同时 Go 在国内外知名公司得到了广泛的应用，比如 google、facebook、百度、小米、今日头条、360 等公司。

本课程将从基础知识和项目实践两大部分进行讲解，通过理论和实践相结合的方式全面的阐述 Go 在高并发、大流量方面的应用。本课程采用的两个项目全部来自大公司实实在在的线上案例，并不是通过简化的、阉割过的项目。保证学有所用。

为什么要学习 Go

1. 渴望从事高并发、海量用户的服务端开发。
2. 渴望了解 Go 语言机制，提升个人视野以及架构能力。
3. 渴望与时俱进，为将来更好的发展而充电
4. 有一定的脚本开发基础，渴望了解编译型语言的原理与机制。

通过本课程，你能够学到

1. Go 语言理论与应用的最佳实践
2. 高并发服务端开发经验
3. 海量日志系统架构设计与实践
4. 高并发抢购系统架构设计与实践

课程大纲

第一天	第二天	第三天
开课介绍 课程内容介绍 Golang 的发展 开发环境搭建 golang 和其他语言对比 hello world 实例 包的概念 基本类型	字符串详解 变量和函数 流程控制(if/else/for) 函数详解 go 程序编译和 Makefile gofmt 工具简介 godoc 文档生成工具详解 代码演练	数组 结构体 面向对象 指针详解 go 程序调试 代码演练
第四天	第五天	第六天
slice 操作与原理 map 数据类型详解 package 详解 结构体和方法 接口和多态 面向接口编程详解 代码演练	错误处理 反射机制详解 goroutine 和 channel 代码演练	文件操作-创建 文件操作-读取 文件操作-写入 网络编程-tcp 简介 网络编程-http 简介 网络编程-helloworld 网络编程-web 开发 代码演练
第七天	第八天	第九天
channel 回顾 带缓冲区的 channel 线程同步和锁 互斥锁 读写锁 waitgroup 使用 代码演练	容错处理 使用 mysql 数据库 单元测试 go 语言模式 代码演练	json 处理 加解密技术介绍 实用代码片段 redis 使用 代码演练

第十天	第十一天	第十二天
Beego 框架介绍 controller 运行机制 model 逻辑 view 编写 静态文件处理	配置读取 路由设置 错误处理 日志处理 数据输出 代码演练	ORM 使用 CRUD 操作 高级查询 事务处理 测试用户 代码演练
第十三天	第十四天	第十五天
模板语法介绍 模板处理 静态文件处理 模板分页处理 代码演练	秒杀系统之本质 秒杀系统之千万并发支撑 秒杀系统之架构设计 秒杀系统之长连接子系统 秒杀系统之秒杀核心逻辑	秒杀系统之过载保护 秒杀系统之性能优化 秒杀系统总结 海量日志系统简介 海量日志系统架构设计 海量日志系统之收集客户端设计
第十六天		
海量日志系统之 kafka 搭建 海量日志系统之 ES（检索）搭建 海量日志系统之 transfer 模块设计 海量日志系统 web 管理平台 海量日志系统总结		

名师介绍

Tony zhang

曾任职于百度，目前任某国内 top5 知名互联网架构师/技术经理，有着丰富的高并发、大流量实战经验，对于 Nginx、Redis、分布式存储、web 架构有着浓厚的兴趣。

根据公司业务需求修改并优化过多个开源框架源码，开发的补丁插件曾被 Nginx 官方收录，推动部门业务全面从 PHP 迁移到 Go，在 Go 的 Rpc 调用、Rpc 框架、容灾、高可用、自动摘除等基础建设方面积累了丰富的经验。