

# go语言图标的发展史是什么

🕒 2023-03-10 📁 编程技术 📄 下载本文 🐹 go语言

这篇“#go语言图标的发展史是什么”文章的知识点大部分人都不太理解，所以小编给大家总结了以下内容，内容详细，步骤清晰，具有一定的借鉴价值，希望大家阅读完这篇文章能有所收获，下面我们一起来看看这篇“go语言图标的发展史是什么”文章吧。



go语言图标是囊地鼠（gopher）。gopher是一种生活在加拿大的小动物，go的吉祥物就是这个小动物，它的中文名叫做囊地鼠，它们最大的特点就是挖洞速度特别快；囊地鼠也代表了Go语言的运行速度、开发速度、学习速度（develop）快。

Go语言的logo就是很简洁的GO两个字母。

## Go的吉祥物

Go之所以叫Go，是想表达这门语言的运行速度、开发速度、学习速度（develop）都像gopher一样快。

go语言的标志是一只囊地鼠，有说法说是土拨鼠，但据有心人士（闲得无聊）考据党考据得出go的原型其实是一只囊地鼠（gopher）。

gopher是一种生活在加拿大的小动物，go的吉祥物就是这个小动物，它的中文名叫做囊地鼠，他们最大的特点就是挖洞速度特别快，当然可能不止是挖洞啦。

go语言萌萌的吉祥物在一堆编程语言中画风都不同。

不禁让人怀疑这只地鼠真的是Geek做出的吗？

当然

不是啦！

这只萌萌的吉祥物是由Rob Pike的妻子 Renee French绘制的，Pob Pike是Go的发明者之一，贝尔实验室UNIX小组成员，Plan9主要开发者，他最喜欢做似乎就是：

发明操作系统

发明编辑器

发明语言

....

这位大神虽然很传奇，但他不是我们此次的主角。

我们的主角，golang吉祥物的设计者Renee French是一位知名插画师，她的画风是这样的：

就是她设计出了Golang吉祥物，可爱的Gordon

比心心！

## Go语言的发展

Go 语言起源 2007 年

Go 从 2009 年 9 月 21 日开始正式作为谷歌公司 20% 兼职项目

2009年11月10日Google公司才正式发布Go语言，并以BSD协议完全开源，支持Linux和Mac OS平台，同年11月支持Windows平台。

下面我们以时间轴的方式梳理了在这个迭代过程中，发生了哪些里程碑式的大事件。

若想详细了解Go历史版本更迭请看: <https://golang.google.cn/doc/devel/release.html>

- 2007年9月，雏形设计，Rob Pike（罗伯.派克）正式命名为Go；

- 2008年5月，Google全力支持该项目；

// 2009年11月10日，首次公开发布，Go将代码全部开源，它获得了当年的年度语言；

- 2011年3月16日，Go语言的第一个稳定(stable)版本r56发布。

// 2012年3月28日，Go语言的第一个正式版本Go1发布。

- 2013年4月04日，Go语言的第一个Go 1.1beta1测试版发布。

- 2013年4月08日，Go语言的第二个Go 1.1beta2测试版发布。

- 2013年5月02日，Go语言Go 1.1RC1版发布。

RC=Release Candidate,含义是“发布候选版”，

它不是最终的版本,而是最终版(RTM=Release To Manufacture)之前的最后一个版本。

- 2013年5月07日，Go语言Go 1.1RC2版发布。

- 2013年5月09日，Go语言Go 1.1RC3版发布。

- 2013年5月13日，Go语言Go 1.1正式版发布。
- 2013年9月20日，Go语言Go 1.2RC1版发布。
- 2013年12月1日，Go语言Go 1.2正式版发布。
- 2014年6月18日，Go语言Go 1.3版发布。
- 2014年12月10日，Go语言Go 1.4版发布。

// 2015年8月19日，Go语言Go 1.5版发布，本次更新中移除了“最后残余的C代码”。

go1.5的发布被认为是历史性的。完全移除C语言部分，使用GO编译GO（ps：少量代码使用汇编实现），GO编译GO称之为Go的自举，是一门编程语言走向成熟的表现。另外，他们请来了内存管理方面的权威专家Rick Hudson，对GC进行了重新设计，支持并发GC，解决了一直以来广为诟病的GC时延（STW）问题。并且在此后的版本中，又对GC做了更进一步的优化。到go1.8时，相同业务场景下的GC时延已经可以从go1.1的数秒，控制在1ms以内。GC问题的解决，可以说GO语言在服务端开发方面，几乎抹平了所有的弱点。

- 2016年2月17日，Go语言Go 1.6版发布。
- 2016年8月15日，Go语言Go 1.7版发布。
- 2017年2月17日，Go语言Go 1.8版发布。
- 2017年8月24日，Go语言Go 1.9版发布。
- 2018年2月16日，Go语言Go 1.10版发布。
- 2018年8月24日，Go语言Go 1.11版发布。
- 2019年2月25日，GO语言Go1.12版发布。

ps：在GO语言的版本迭代过程中，语言特性基本上没有太大的变化，基本上维持在GO1.1的基准上，并且官方承诺，新版本对老版本下开发的代码完全兼容。事实上，GO开发团队在新增语言特性上显得非常谨慎，Go语言的版本更迭主要集中在稳定性、编译速度、执行效率以及GC性能等底层性能方面，所以说Go语言使用者不必担心快速的版本更迭带来的兼容性问题

Go语言这两年在语言排行榜(<https://www.tiobe.com/tiobe-index/>)增长曲线如下，总体还是处于上涨趋势的

大量基于GO语言开发的杀手级应用：Docker、Kubernetes、Go-Ethereum、Thraform、Moby、Gogs、Grafana、Etc等。于2009、2016两次夺得TIOBE年度编程语言的称号，2020年世界排名处于第12位

## Go语言的应用场景

编程语言说到底只是一种工具，不选最好的，只选最合适的

## // Go语言适用的场景为：

- 1、服务器编程，以前你如果使用C或者C++做的那些事情，用Go来做很合适，例如处理日志、数据打包、虚拟机处理、文件系统等。
- 2、凭借其出色的并发能力，Go在分布式系统领域，如集群系统、游戏服务器、数据库代理器、中间件等场景同样极具威力
- 3、网络编程，这一块目前应用最广，包括Web应用、API应用、下载应用，而且Go内置的net/http包基本上把我们平常用到的网络功能都实现了。
- 4、数据库操作
- 5、开发云平台，目前国外很多云平台在采用Go开发

## // Go语言不适合的场景为：

- 1、强实时性软件：语音通信、无人驾驶、垃圾回收、自动内存分配等因素都让Go在实时性上力不从心
- 其他

Go 语言被设计成一门应用于搭载 Web 服务器，存储集群或类似用途的巨型中央服务器的系统编程语言。对于高性能分布式系统领域而言，Go 语言无疑比大多数其它语言有着更高的开发效率。它提供了海量并行的支持，这对于游戏服务端的开发而言是再好不过了。Go 语言一个非常好的目标就是实现所谓的复杂事件处理（CEP），这项技术要求海量并行支持，高度的抽象化和高性能。当我们进入到物联网时代，CEP 必然会成为人们关注的焦点。但是 Go 语言同时也是一门可以用于实现一般目标的语言，例如对于文本的处理，前端展现，甚至像使用脚本一样使用它。值得注意的是，因为垃圾回收和自动内存分配的原因，Go 语言不适合用来开发对实时性要求很高的软件。越来越多的谷歌内部的大型分布式应用程序都开始使用 Go 语言来开发，例如谷歌地球的一部分代码就是由 Go 语言完成的。如果你想知道一些其它组织使用Go语言开发的实际应用项目，你可以到 使用 Go 的组织 页面进行查看。出于隐私保护的考虑，许多公司的项目都没有展示在这个页面。在 Chrome 浏览器中内置了一款 Go 语言的编译器用于本地客户端（NaCl），这很可能会被用于在 Chrome OS 中执行 Go 语言开发的应用程序。Go 语言可以在 Intel 或 ARM 处理器上运行，因此它也可以在安卓系统下运行，例如 Nexus 系列的产品。在 Google App Engine 中使用 Go 语言：2011 年 5 月 5 日，官方发布了用于开发运行在 Google App Engine 上的 Web 应用的 Go SDK，在此之前，开发者们只能选择使用 Python 或者 Java。这主要是 David Symonds 和 Nigel Tao 努力的成果。目前最新的稳定版是基于 Go 1.4 的 SDK 1.9.18，于 2015 年 2 月 18 日发布。

以上就是关于“go语言图标的发展史是什么”这篇文章的内容，相信大家都有了一定的了解，希望小编分享的内容对大家有帮助，若想了解更多相关的知识内容，请关注本站行业资讯频道。

导出WORD

## 相关推荐

### Go语言中的基础数据类型如何使用



这篇文章主要介绍了Go语言中的基础数据类型如何使用的相关知识，内容详细易懂，操作简单快捷，具有一定

### Go语言怎么实现聊天小工具



本篇内容介绍了“Go语言怎么实现聊天小工具”的有关知识，在实际案例的操作过程中，不少人都会遇到这样的

### 怎么用Go语言实现UDP版聊天小工具



这篇文章主要介绍“怎么用Go语言实现UDP版聊天小工具”的相关知识，小编通过实际案例向大家展示操作过

### Go语言Sync.Pool不加锁也能实现线程安全的原因是什么



这篇文章主要介绍“Go语言Sync.Pool不加锁也能实现线程安全的原因是什么”的相关知识，小编通过实际案例向

### 基于Go语言怎么实现分金币游戏



今天小编给大家分享一下基于Go语言怎么实现分金币游戏的相关知识点，内容详细，逻辑清晰，相信大部分

### Go语言中怎么实现代码跳转



这篇文章主要讲解了“Go语言中怎么实现代码跳转”，文中的讲解内容简单清晰，易于学习与理解，下面请大家

### Go语言切片是如何扩容的



这篇文章主要讲解了“Go语言切片是如何扩容的”，文中的讲解内容简单清晰，易于学习与理解，下面请大家跟

### Go语言进阶freecache源码分析



这篇文章主要介绍“Go语言进阶freecache源码分析”的相关知识，小编通过实际案例向大家展示操作过程，操作