

Java：一个帝国的诞生

Original 刘欣 码农翻身 2016-05-17



来自专辑

用故事讲解技术，尽在码农翻身

1 C语言帝国的统治

现在是公元1995年，C语言帝国已经统治了我们20多年，实在是太久了。

1972年，随着C语言的诞生和Unix的问世，帝国迅速建立统治，从北美到欧洲，从欧洲到亚洲，无数程序员臣服在他的脚下。

帝国给我们提供了极好的福利：贴近硬件，运行极快，效率极高。

使用这些福利，程序员们用C开发了很多系统级软件，操作系统，编译器，数据库，网络系统.....

但是帝国也给我们安上了两个沉重的枷锁：指针和内存管理

虽然指针无比强大，能直接操作内存，但是帝国却没有给我们工具去做越界的检查，导致很多新手程序员轻易犯错。

至于内存管理，帝国更完全是放任的态度：你自己分配的空间，自己去释放！

更要命的是这些问题在编译期发现不了，在运行时才会突然暴露，常常让我们手忙脚乱，昏天黑地去调试。

我们的大量时间和宝贵的精力都被浪费在小心翼翼的处理指针和内存分配上。

每个程序员都被这两个东西搞的焦头烂额！

帝国宣称的可移植性骗了我们，他宣称我们在一个机器上写的程序，只要在另外一个机器上编译就可以了，实际上不是这样。他要求我们尽量用标准的C函数库。其次，如果遇到了一些针对特定平台的调用，需要对每个平台都得写一份！有一点点小错误，都会导致编译失败。

1982年，帝国又推出了一门新的语言C++，添加了面向对象的功能，兼容C，有静态类型检查，性能也很好。

但是这门新的语言实在是太复杂了，复杂到比我聪明的多的人都没有办法完全掌握这门语言，它的很多特性复杂的让人吃惊。

C++在图形领域和游戏上取得了一些成功，但是我一直学不好它。

2 反抗

我决定反抗这个庞大的帝国，我偷偷的带领着一帮志同道合的兄弟离开了，我们要新建一块清新自由的领地。

为了吸引更多的程序员加入我们，我们要建立一个新的语言，这个语言应该有这样的特性：

语法有点像C，这样大家容易接受

没有C语言那样的指针

再也不要考虑内存管理了，实在受不了了

真正的可移植性，编写一次，到处运行

面向对象

类型安全

还有，我们要提供一套高质量的类库，随语言发行。

我想把这个语言命名为C++- ，即C++减减， 因为我想在C++的基础上改进， 把它简化。

后来发现不行， 设计理念差别太大。

干脆重启炉灶。

我看到门口的一棵橡树， 就把这个语言叫做Oak。

但是后来发布的时候， 发现Oak已经被别人用了， 我们讨论很久， 最终决定把这门新的语言叫做 Java。

为了实现跨平台， 我们在操作系统和应用程序之间增加了一个抽象层： Java 虚拟机

用Java写的程序都跑在虚拟机上， 除非个别情况， 都不用看到操作系统。

3 一鸣惊人

为了吸引更多的人加入我们的新领地， 我们决定搞一个演示， 向大家展示Java 的能力。

出世未久的Java其实还远不完善。 搞点什么好呢？

我们把眼光盯上了刚刚兴起的互联网， 1995年的网页简单而粗糙， 缺乏互动性。 于是我们在浏览器上弄了个小插件， 把java 运行环境放了上去。

然后在上面开发了一个图形界面的程序(Applet)， 让它看起来美轮美奂， 震撼人心。

每一个看到他的程序员都会发出“Wow”的惊叹！ 为之倾倒。

Java 活了！

通过Applet， 无数的程序员看到了Java这门语言， 了解了这门语言特性以后， 很多无法忍受C帝国暴政的程序员， 很快加入了我们， 我们的领地开始迅速扩大。

连C语言帝国里的一些商业巨头也纷纷来和我们合作， 其中就包括Oracle， 微软这样的巨头， 微软的头领Bill Gates还说： 这是迄今为止设计的最好的语言！

但是Bill Gates非常的不地道， 买了我们的Java 许可以后， 虽然在自家的浏览器上也支持Applet, 但是他们却偷偷的试图修改Java， 想把Java绑死在自家的操作系统上赚钱，

Java会变的不可移植。

这是我们难于忍受的， 我们和微软发起了一场旷日持久的游击战争， 逼着微软退出了Java领域， 开发了自己的.NET，这是后话。

4 开拓疆土

从1995年到1997年，我们依靠 Java 不断的攻城略地， 开拓疆土，我们王国的子民不断增加， 达到了几十万之众， 已经是一个不可忽视的力量了。

但是大家发现， Java除了Applet, 以及一些小程序之外， 似乎干不了别的事情。

C帝国的人还不断的嘲笑我们慢， 像个玩具。

到了1998年， 经过密谋， 我们Java 王国决定派出三只军队向外扩展：

Java 2 标准版(J2SE)： 去占领桌面

Java 2 移动版(J2ME)： 去占领手机

Java 2 企业版(J2EE)： 去占领服务器

其中的两只大军很快败下阵来。

J2SE 的首领发现， 开发桌面应用的程序员根本接受不了Java， 虽然我们有做的很优雅的Swing 可以开发界面， 但是开发出的界面非常难看， 和原生的桌面差距很大。 尤其是为了运行程序还得安装一个虚拟机， 大家都受不了。

J2ME也是， 一直不受待见， 当然更重要的原因是乔布斯还没有重新发明手机， 移动互联网还没有启动。

失之东隅，收之桑榆， J2EE赶上了好时候， 互联网大发展， 大家忽然发现， Java简直是为写服务器端程序所发明的！

强大， 健壮， 安全， 简单， 跨平台！

在J2EE规范的指导下， 特别适合团队开发复杂的大型项目。

我们授权BEA公司第一个使用J2EE许可证， 推出了Weblogic， 凭借其集群功能， 第一次展示了复杂应用的可扩展性和高可用性。

这个后来被称为中间件的东西把程序员从事务管理，安全管理，权限管理等方面解放出来，让他们专注于业务开发。这立刻捕获了大量程序员的心。

很快Java 王国的子民就达到数百万之众。

榜样的力量是无穷的，很快其他商业巨头也纷纷入场，尤其是IBM，在Java 上疯狂投入，不仅开发了自己的应用服务器 Websphere，还推出了Eclipse这个极具魅力的开源开发平台。

当然IBM利用java 获得了非常可观的效益，软件+硬件+服务 三驾马车滚滚向前，把IBM推向了一个新的高峰。

5 帝国的诞生

大家也没有想到，除了商业巨头以外，程序员们也会对Java王国这么热爱，他们基于Java 开发了巨多的平台，系统，工具，例如：

构建工具: Ant, Maven, Jekins

应用服务器: Tomcat, Jetty, Jboss, Websphere, weblogic

Web开发: Struts, Spring, Hibernate, myBatis

开发工具: Eclipse, Netbean, IntelliJ IDEA, Jbuilder

。。。。等等等等。。。。

并且绝大部分都是开源的！

微软眼睁睁的看着服务器端的市场被Java 王国占据，岂能善罢甘休？他们赶紧推出.NET来对抗，但我们已经不在乎了，因为他的系统是封闭的，所有的软件都是自家的：

开发工具是Visual Studio, 应用服务器是IIS, 数据库是SQL Server，只要你用.NET，基本上就会绑定微软。

另外他们的系统只能运行在Windows服务器上，这个服务器在高端市场的占有率实在是太低了。

2005年底，一个新的王国突然崛起，他们号称开发效率比java 快5-10倍，由此吸引了大批程序员前往加盟。

这个新的王国叫做Ruby on Rails, 它结合了PHP体系的优点（快速开发）和Java体系的优点（程序规整）， 特别适合快速的开发简单的Web网站。

虽然发展很快， 但没有对Java 王国产生实质性的威胁， 使用Ruby on Rails搭建大型商业系统的还很少。

除了Ruby on Rails ， 还有PHP， Python ， 都适合快速开发不太复杂的Web系统。 但是关键的， 复杂的商业系统开发还是Java 王国的统治之下。 所以我们和他们相安无事。

2006年， 一只叫Hadoop的军队让Java王国入侵了大数据领域， 由于使用Java 语言， 绝大多数程序员在理解了Map/Reduce， 分布式文件系统在Hadoop中的实现以后， 很快就能编写处理海量数据的程序， Java 王国的领地得到了极大的扩展。

2008年， 一个名叫Android 的系统横空出世， 并且随着移动互联网的爆发迅速普及， 运行在Android之上的正是Java ！

Java 王国在Google的支持下， 以一种意想不到的方式占领了手机端， 完成了当年J2ME 壮志未酬的事业 ！

到今年为止， 全世界估计有1000万程序员加入了Java王国， 它领土之广泛， 实力之强大， 是其他语言无法比拟的。

Java 占据了大部分的服务器端开发， 尤其是关键的复杂的系统， 绝大多数的手机端， 以及大部分的大数据领域。

一个伟大的帝国诞生了。

这个帝国能生存多久？ 谁会摧毁这个庞大的帝国呢？

我不知道， 你呢？

热门文章：

我是一个线程

我是一个Java class

Javascript: 一个屌丝的逆袭

小王的架构师之路

程序员在工作中必备的能力

TCP/IP 之 大明王朝的邮差

CPU 阿甘

IE为什么把Chrome和火狐打伤了

Node.js :我只需要一个店小二

假如我是计算机系老师

假如时光倒流，我会这么学Java

学会编程，而不是学会Java

15年编程生涯，资深架构师总结的7条经验