

Table of Contents

开篇语]	1.1
IC 偏恋 Tcl	1.2

last modified by , 04/17/2018 01:49:30

IC 极客

技术服务支持:support@icxhub.com

产品购洽联系:sale@icxhub.com

Copyright © 2018 [xHub](#).
All rights reserved.

IC 偏恋Tcl

1. Tcl 的诞生历史

Tcl, 发音做"tickle", 是Tool Command Language的首字母缩写。

Tcl/Tk 的发明人 John Ousterhout 教授在八十年代初, 是伯克利大学的教授。在其教学过程中, 他发现在集成电路 CAD 设计中, 很多时间是花在编程建立测试环境上。并且, 环境一旦发生了变化, 就要重新修改代码以适应。这种费力而又低效的方法, 迫使 Ousterhout 教授力图寻找一种新的编程语言, 它即要有好的代码可重用性, 又要简单易学, 这样就促成了 Tcl (Tool Command Language) 语言的产生。

Ousterhout 教授被 SUN 于 1994 年召致麾下, 加入 SUNLab, 领导一个小组从事将 Tcl 移植到所有其它平台的工作, 如 Windows 和 Macintosh。同时为 Tcl 增加了 Safe-Tcl 安全模块, 并为浏览器的开发了 Tcl plug-in, 以及支持 Java bytecode 的编译器, 大字符集, 新的 IO 接口, 与 Java 的计算平台相连等。Tcl还有自己的浏览器。在面向对象程序设计占主导地位的今天, 又开发了支持面向对象的 incr Tcl。为了鼓励各厂商开发第三方的程序, Tcl 的源代码可免费下载。所有这些努力使 Tcl 成为一个适应当代信息产业潮流的, 支持多平台的, 优秀的开发语言。

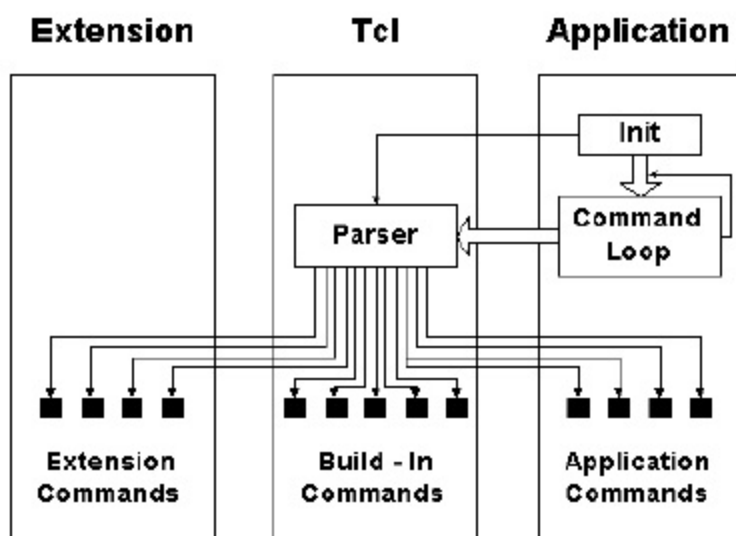
Tcl 是用 C 语言开发的。它现在可运行在 Unix, Windows 和 Macintosh 等各种平台上。

Ousterhout 教授于 1998 年初, 离开了 SUN, 自立 Scriptics 公司, 继续 Tcl/Tk 的研究和开发工作。

Tcl 最初的构想的是希望把编程按照基于组件的方法 (component approach), 即与其为单个的应用程序编写成百上千行的程序代码, 不如寻找一个种方法将程序分割成一个个小的, 具备一定“完整”功能的, 可重复使用的组件。这些小的组件小到可以基本满足一些独立的应用程序的需求, 其它部分可由这些小的组件功能基础上生成。不同的组件有不同的功能, 用于不同的目的。并可为其它的应用程序所利用。当然, 这种语言还要有良好的扩展性, 以使用户为其增添新的功能模块。最后, 需要用一种强的, 灵活的“胶水”把这些组件“粘合”在一起, 使各个组件之间可互相“通信”, 协同工作。程序设计有如拼图游戏一样, 这种设计思想与后来的 Java 不谋而合。终于在 1988 年的春天, 这种强大灵活的胶水 - Tcl 语言被发明出来了。

按照 Ousterhout 教授的定义, Tcl 是一种可嵌入的命令脚本化语言 (Command Script Language)。“可嵌入”是指把很多应用有效, 无缝地集成在一起。“命令”是指每一条 Tcl 语句都可以理解成命令加参数的形式:

Tk (Tool Kit) 是基于 Tcl 的图形程序开发工具箱, 是 Tcl 的重要扩展部分。Tk 隐含许多 C/C++ 程序员需要了解的程序设计细节, 可快速地开发基于图形界面 Windows 的程序。据称, 用 Tcl/Tk 开发一个简单的 GUI 应用程序只需几个小时, 比用 C/C++ 要提高效率十倍。需要指明的是这里所说的“窗口”是指 Tcl 定义的窗口, 与 X-Windows 与 MS Windows 的定义有所不同, 但它可完美地运行在以上两个系统上。



1.2.1 - Structure

从图中我们可以了解到有个概念很重要，你输入到Tcl的所有交互的指令或者其他接口都被认为是文本，按一定的顺序进行parser。

EDA软件对Tcl的支持现在都已经到8.6了，由于其扩展性好，调试简单，C接口友好，使用者学习成本低，这些都是商用软件的价值所在，最重要的是开源，免费。在IC设计流程中，有些文件格式是基于Tcl的，比如SDC。

Date	Event	
January 1990	Tcl announced beyond Berkeley (Winter USENIX).	
June 1990	Expect announced (Summer USENIX).	
January 1991	First announcement of Tk (Winter USENIX).	
June 1993	First Tcl/Tk conference (Berkeley). [table] geometry manager (forerunner of [grid]), [incr Tcl], TclDP and Groupkit, announced there.	
August 1997	Tcl 8.0 introduced a bytecode compiler.	
April 1999	Tcl 8.1 introduces full Unicode support and advanced regular expressions.	
August 1999	Tcl 8.2 introduces Tcl Extension Architecture (TEA)	
August 2000	Tcl Core Team formed, moving Tcl to a more community-oriented development model.	
September 2002	Ninth Tcl/Tk conference (Vancouver). Announcement of starkit packaging	system. Tcl 8.4.0 released.
December	Tcl 8.5 added new datatypes, a new extension repository, bignums,	

2007	lambdas.
December 2012	Tcl 8.6 added built-in dynamic object system, TclOO, and stackless evaluation.

我们现在最新使用的也是Tcl8.6, 最近一次发布时Jul 27, 2016。

2. Tcl 诞生近30年为什么在EDA 和CAD 方向依然未被取代？

有些本身看起来并没有优势, 脱离开应用场景甚至说是一门可以被淘汰的语言, 那么我们来看看它没有被淘汰的原因有哪些。Tcl 依然做为IC 设计中地主力语言存在着, 为什么IC 偏偏恋上了Tcl, 今天我就语言的特点谈谈我对这个话题的看法。

- 免费
- C 的结合精密
- 开发部署周期短
- TK 图形界面
- 调试性语言
- 整合性语言
- 入门简单, 转移关注度
- 扩展性强(package 的优势)

跨平台 网络功能

3. IC 工程师为什么要熟悉Tcl 以及其他编程语言

- Lisp
 - Scheme
 - SKILL
- Perl
- Python
- Shell

从JD 谈起, 如何自查tcl 水平和提高Tcl 编程能力。本文的第二个重点。开发若干问题, 由浅入深提到Tcl 的概念做为测试的级别划分。

4. Tcl 家族: Tcl/Tk/Expect

普及Tk 和Expect, 本文的第三个重点, 既然放在最后, 关注度没有前两个高。

5. 好的Tcl 程序的标准

为xhub 的结构铺路

6. Tcl 优势的几个实例鉴赏

几个Tk 程序的例子

7. 参考内容

- <https://blog.csdn.net/larryliuqing/article/details/20902181>
- <http://scc.qibebt.cas.cn/docs/linux/script/TclTkall.pdf>

Copyright © 2018 xHub.
All rights reserved.