

本文介绍使用NS2进行网络仿真通常需要使用到的语言以及工具（工具都是linux平台下的）。

## （一）Tcl语言

Tcl(最早称为“工具命令语言”“Tool CommandLanguage”)是一种脚本语言。由John Ousterhout创建。Tcl很好学,功能很强大。Tcl经常被用于快速原型开发,脚本编程, GUI和测试等方面。

Tcl本身不提供面向对象的支持。但是语言本身很容易扩展到支持面向对象。许多C语言扩展都提供面向对象能力,包括XOTcl, Incr Tcl 等。另外S NIT扩展本身就是用Tcl写的。使用最广泛的Tcl扩展是TK。 TK提供了各种OS平台下的图形用户界面GUI。连强大的Python语言都不单独提供自己的GUI,而是提供接口适配到TK上。另一个流行的扩展包是Expect。 Expect提供了通过终端自动执行命令的能力,例如(passwd, ftp, telnet等命令驱动的外壳)。

### 1. 特性

- a. 任何东西都是一条命令,包括语法结构(for, if等)。
- b. 任何事物都可以重新定义和重载。
- c. 所有的数据类型都可以看作字符串。
- d. 语法规则相当简单
- e. 提供事件驱动给Socket和文件,基于时间或者用户定义的事件也可以。
- f. 动态的域定义。
- g. 很容易用C, C++,或者Java扩展。
- h. 解释语言,代码能够动态的改变。
- i. 完全的Unicode支持。
- j. 平台无关。Win32, UNIX, Mac 上都可以跑。
- k. 和Windows的GUI紧密集成。
- l. 代码紧凑,易于维护。

Tcl的Procedures 和C的函数差不多。它们有参数,它们返回值。基本定义方法是: proc name argList body。当一个procedure被定义,它就被看做是一个命令,如同Tcl的自带命令一样,通过名字来呼叫,名字后面跟上参数。缺省的, procedure的返回值是它的最后一个命令结果。但也可以通过return命令来返回其他值。Return值可以在procedure的任何地方,一旦执行, procedure就此返回。在procedure中可以通过set创造变量,但是变量只在procedure中有效,而且一旦procedure返回,这些变量就不可访问。如果procedure需要访问主程序中的变量,就需要使用global关键字。

## （二）gawk语言

gawk是一种对文档资料具有很强处理能力的程序语言,简短的几句代码就可以完成对文档的修改、对比、抽取,相对其它语言来说,其对文档资料的处理方面具有很大的优势。gawk是GNU下开发的awk,最初在1986年完成,,此后不断改进和创新,现已包含awk的所有功能。

gawk处理文档的过程是针对档案的每一条记录搜寻指定的模式(Pattern),当记录中有符合指定的模式时,gawk就会在此记录执行被指定的动作(Action)。gawk依照此方式处理输入档案的每一条记录直到输入档案结束。

gawk程序是由很多的Pattern和Action所组成,Action写在大括号{}里面,一个Pattern后面就跟着一个Action,整个gawk程式的结构如下:

```
Pattern (1) {Action (1) }
```

```
Pattern (2) {Action (2) }
```

```
.....
```

### （三）简单动画显示工具NAM

NAM(Network Animator)的功能是根据网络模拟软件或真实环境里的特定格式的Trace 输出文件来运行动画,以观测网络模拟中的Trace和数据分组流向。需要说明的是尽管NAM 工具已经推出了一系列的版本并且性能也趋于稳定,但仍然存在一些bug有待进一步改善。本毕设中就出现需要更正的bug,。(详见后面毕设演示内容)

NAM的使用方法如下:

nam

-g<geometry> 指明NAM窗口的几何位置

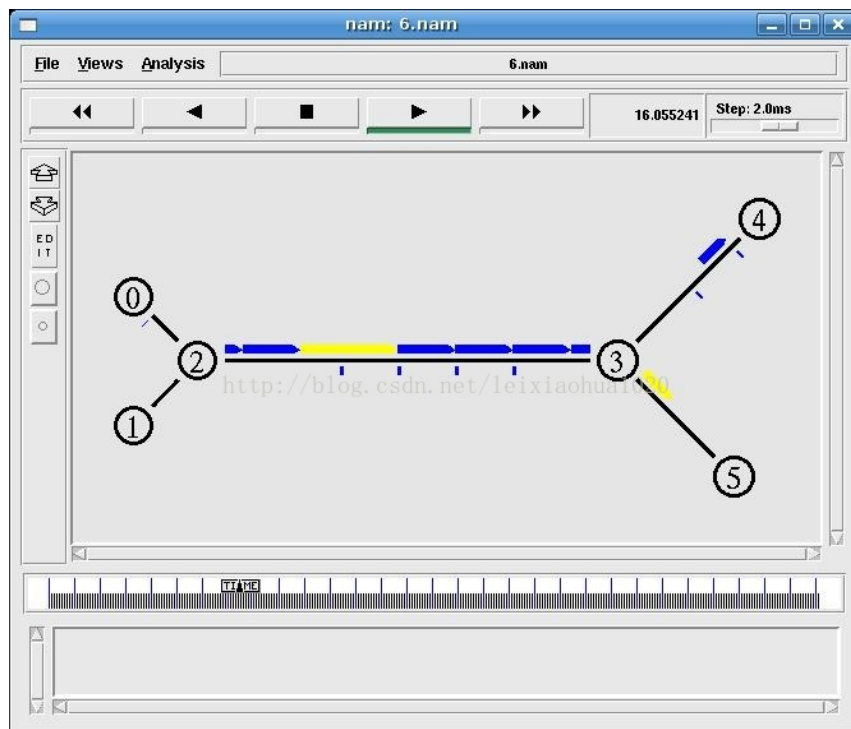
-t<graphInput> 指定NAM使用tkgraph，并为tkgraph确定输入的文件

-i<interval> 指定屏幕刷新率，单位为毫秒（ms），默认为50ms（20帧/秒）

-j<start-up\_time> 指定NAM演示时的仿真时间

-k<initial\_socket\_port\_number> 初始化socket端口号

NAM的播放界面范例如图：



NAM的播放界面范例

## （四）gnuplot 绘图组件

gnuplot是一个命令行的交互式绘图工具 (command-driven interactivefunction plotting program)。用户通过输入命令,可以逐步设置或修改绘图环境,并以图形描述数据或函数,使我们可以借由图形做更进一步的分析。

gnuplot是由Colin Kelly和Thomas Williams于1986年开始开发的科学绘图工具,支持二维和三维图形。它的功能是把数据资料和数学函数转换为容易观察的平面或立体的图形,它有两种工作方式,交互式方式和批处理方式,它可以让使用者很容易地读入外部的数据结果,在屏幕上显示图形,并且可以选择和修改图形的画法,明显地表现出数据的特性。

## （五）Trace 文件

Trace文件是在模拟过程中生成的记录模拟每一个分组调度事件的文件,一次模拟生成的信息基本上全体现在Trace文件中。Trace文件包含了很多有用的可供分析的数据,我们需要从中筛选出我们关注的数据进行分析。

以下通过一个范例对Trace文件的格式做一个说明：

```
s 3.000000000 _0_ RTR--- 0 AODV 48 [0 0 0 0] ----- [0:255 -1:255 30 0]
```

下面按从左往右的顺序对各字符表示的意义进行说明：s代表事件类型,有如下几个类型：

ü s：分组的发送事件；

ü r：分组的接收事件；

ü d：分组的丢弃事件；

ü f：分组的转发事件；

3.000000000代表事件产生的时间；\_0\_代表处理该事件的节点ID；RTR代表Trace种类,有如下几个种类：

ü RTR:路由器Trace；

ü AGT：代理Trace；

ü MAC：MAC层Trace；

---代表分隔符；0代表分组ID；AODV 代表分组类型；48代表分组大小（字节）；[0 0 0 0]代表有关MAC层的详细信息，如0x800：IP分组，0x806：ARP分组；（第一个0表示发送节点在无线信道上发送该分组所期望的时间值，用16进制表示；第二个0表示接收节点的MAC地址；第三个0表示发送节点的MAC地址；第四个0表示MAC层封装的分组类型）；-----代表分隔符；0:255 代表分组发送的源IP地址：节点号.端口号；1:255代表分组发送的目的IP地址：节点号.端口号；30代表分组的TTL值；0代表源节点到目的节点的跳数。

版权声明：本文为博主原创文章，未经博主允许不得转载。 <https://blog.csdn.net/leixiaohua1020/article/details/12233643>

文章标签： ns2 nam gnuplot tcl 网络仿真

个人分类： [计算机网络](#)

此PDF由spygg生成, 请尊重原作者版权!!!

我的邮箱:liushidc@163.com