# Expect-lite Update

## Linux Symposium 2014

A quick look four years later

by

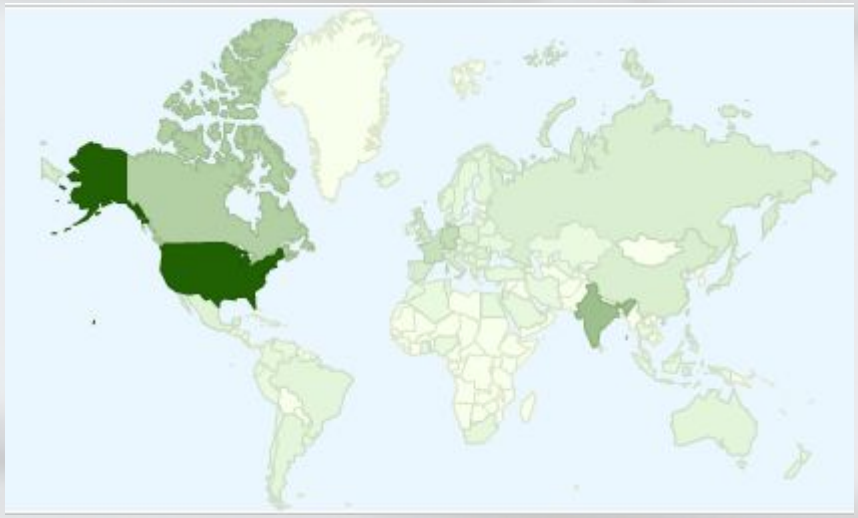
Craig Miller

# Quick Introduction

## What is expect-lite

- Expect-lite is an open-source automation software
  - Designed for non-programmers
  - Directly maps terminal session to automated script
  - Full Featured, Mature (9 years), with hundreds of users world-wide
- Portable: Runs on Windows, Linux, MacOS X
- World Wide Use (2014)

| Rating | Country | Num of Downloads |
|---|---|---|
| 1 | United States | 3441 |
| 2 | Canada | 843 |
| 3 | India | 714 |
| 4 | China | 591 |
| 5 | Germany | 487 |
| 6 | France | 354 |
| 7 | Spain | 251 |
| 8 | Austria | 220 |
| 9 | United Kingdom | 212 |
| 10 | Russia | 191 |
| 11 | Italy | 150 |
| 12 | Japan | 130 |



# Quick Introduction

## What is expect-lite

- A script written in Don Libe's Expect
- Very portable
  - Runs everywhere Expect does
- Keep it Simple

  > to send

  < to expect
- > is wait for prompt and send text
- Plays well with Bash

# Quick Introduction

## Example 1

- Checking an IP Address

  >ifconfig

  <192\.168\.1\.1

- Or a range using regex \d+ (means one or more digits)

  >ifconfig

  <192\.168\.1\.\d+

# Introduction
## Variables & Constants

- All variables begin with $
  - assignment$IP=192.168.1.1
  - use >telnet $IP
- Constants are parameters passed on the command line at invocation
  - Are immutable override existing variable values
  - Allows changing script behaviour
- Setting a Constant on the command line
  - example2.elt IP=172.30.1.1

# Introduction

## Other commands

| Char | Action |
|---|---|
| >> | send string to remote host, without waiting for prompt |
| << | literal string MUST be received |
| ~< | fuzzy expect, expect an approximate number (decimal or hex) |
| -< | NOT expect; string/regex must NOT be received from the remote host |
| # | used to indicate comment lines, and have no effect |
| ; | are also used to indicate comment lines, but are printed to stdout (for logging) |
| ;; | similar to above, but no extra newlines are printed (useful for printing script help) |
| @ | changes the expect timeout to num of seconds |
| $var= | static variable assignment at script invocation |
| +$var= | dynamic variable assignment |
| =$var | math functions, perform bitwise and arithmetic operations: << >> & \| ^ * / % + - |

# Introduction

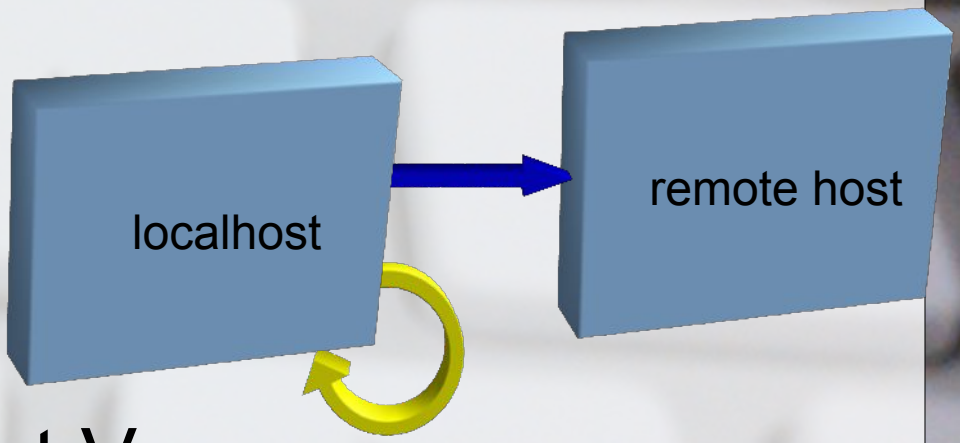
## Other commands (cont)

| Char | Action |
|---|---|
| | c-style if/then/else in the format ?cond? |
| ? | action::else_action |
| % | label - used for jumping to labels |
| ~fname | includes a expect-lite script file, fname |
| ! | embedded expect (native tcl) |
| [ $i < $n<br>>ls<br>] | while loop, comparison at top of loop |
| [ $i=$L<br>>$i<br>] | for each loop, iterates thru list, setting value $i |
| =$str | string math, remove, search/replace |

# Quick Introduction

## Automatic Remote Login

- Designed for server farm
- Modes of operation
  - Local or none
  - Loopback
  - Remote Host
    - telnet, ssh, ssh with keys
- Controlled by Environment Vars



# New Features

- Color (3.7)
  - *EXP_INFO
- IDE (4.0)
  - *TIMESTAMP > commands
- Library Release (4.1)
  - use expect-lite as a TCL library
- Code blocks (4.2)
  - While Loop
  - Multi-line if statements
  - IPv6 ready

# New Features (cont)

- User Defined help (4.3)
  - Include file improvements - simple regression
  - *NOINTERACT
- Native Logging (4.4)
  - *LOG *NOLOG
- Code blocks - else statement (4.5)
- Foreach Loops with code blocks (4.6)
  - String Math: search/replace, concat, remove
- Fuzzy Expect (4.7)
  - Expect an approximate value e.g. 11 is about 10

# Integrated Debugger (IDE)

- Breakpoint = *INTERACT
- See Variables with <esc>v
- Copy/Paste any expect-lite lines into IDE
- <esc>h for IDE help

- Instant-Interact, creates breakpoint on the fly with ^\
  - All IDE functions available

# Code Blocks

- Old expect-lite was line oriented
- Code blocks start with [ and end with ]
- Provides structure to script, enabling:
  - While loops
  - Foreach loops
  - Multi-line if statements (and else statements)

# Code Blocks

## Example 2

- While Loop

  $i=1

  [ $i < 10

    >ping -c 1 192.168.1.$i

    <64 bytes from

    +$i

  ]

-

# Code Blocks

## Example 3

- Foreach Loop

```
$host_list=1 5 15 200
[ $host=$host_list
  ;  pinging host: $host
  >ping -c 1 192.168.1.$host
  <64 bytes from
]
```

-

# Other Expects

- Literal expect
  - <<This is a pipe |
- NOT expect
  - -<ERROR
- Fuzzy expect
  - # set the fuzziness
  - ~=2
  - >uptime
  - ~<load averages: (3)

# Fuzzy Expect

## Example 4

- Checking ping time

  $host=8.8.8.8

  ~=25

  ; Check ping time to host: $host

  >ping -c 1 $host

  <64 bytes from $host

  ~<time=(50) ms

- Run to another host using a constant
  - check_ping.elt host=sf.net

# Conclusion

- Easy to Use, Very Portable, Executables
- Advanced Features:
  - Variables, Constants, Conditionals, Regex Evaluation, Multiple Sessions, Code Blocks (while loops, foreach loops), TCL Library Integration (STA), NOT expect, fuzzy expect, IPv6 support
- Easy to Debug with IDE
  - Set Breakpoints, copy/paste script into running script
- Blog with helpful tips
  - http://expect-lite.blogspot.ca/
- Automation for the rest of us!

# Thank You

- Presentation & Examples at:

http://expect-lite.sf.net/expect-lite_ols.html

# Additional Info