

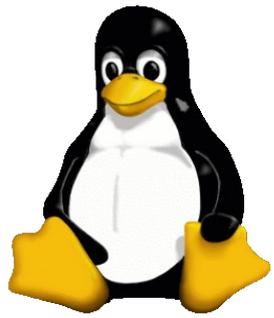
STL! / unix / usr / group

Simple Redirection

Welcome

Stan Reichardt

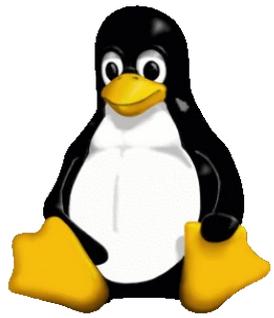
stan2006@sluug.org



STL! / unix / usr / group

Agenda

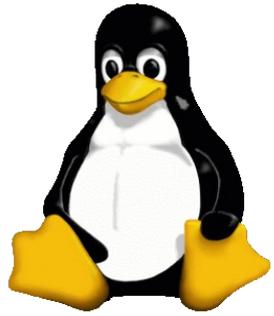
- Who
- What
- Where
- When
- Why
- How



STL!/unix/usr/group

Definitions

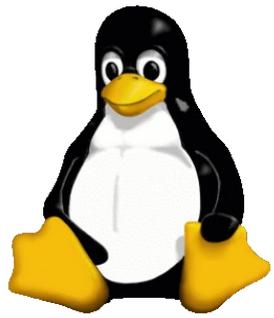
- **shell** - A command interpreter that allows a user to run executable code. Shells also store and configure additional information about a user's executable paths, environment variables, and usability options.
- **Command Line Interface (CLI)** - A means of communication between a program and its user, based solely on textual input and output. Commands are input with the help of a keyboard or similar device and are interpreted and executed by the program. Results are output as text or graphics to the **terminal**.
- **Graphical User Interface (GUI)** - A visual object that enhances interaction between a computer and a user. A GUI displays windows, icons, menus, labeled buttons, and a virtual desktop. A GUI sends commands to the system via these icons and menus rather than typed commands.



STL! / unix / usr / group

More Definitions

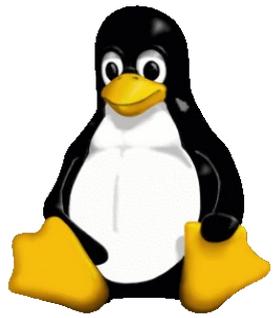
- **Standard Input/Output (stdio)** -
 - **Standard In (stdin)** - User input is usually read from the keyboard
 - **Standard Out (stdout)** - The output of scripts is usually to the *terminal (screen)*
 - **Standard Error (stderr)** - A special type of output used for error messages. Usually sent to the *terminal (screen)*



STL! / unix / usr / group

More Definitions Continued

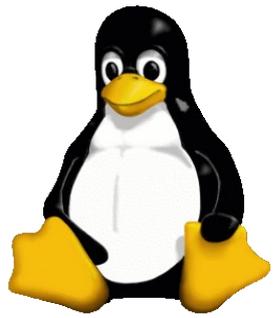
- **file** - A piece of data stored as a single addressable object on a file system. Files can be executable, text or binary-based, and can also represent computer system hardware to be controlled by the operating system.
- **filter** - a program performs operations on input and writes the result to the standard output keyboard input -
- **pipe** - A programming device that lets you use the output of one process as the input of another. You must use a **vertical bar** (|) between two command invocations to indicate piping.
- **redirection** - Accepting input from a source other than the standard input or sending output to a destination other than the standard output.



STL! / unix / usr / group

Lots of Concepts

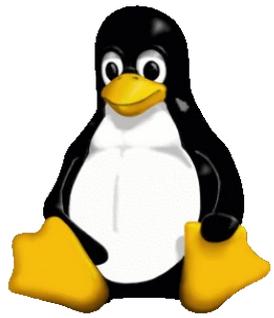
- Not all in individual man pages or info documentation
- Loosely mentioned in passing
- Covered in guides and books
- Simple yet profound
- Simple building blocks



STL! / unix / usr / group

Other Concepts of Interest

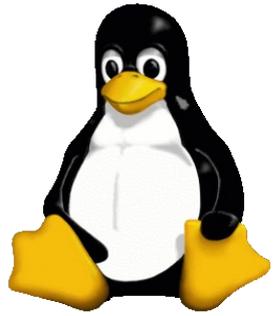
- shell
- shell scripting
- Inodes
- terminal
- processes (**ps**, **top**, **kill**)
- background processing (**fg**, **bg**, **jobs**)



STL! / unix / usr / group

Standard Input/Output (stdio)

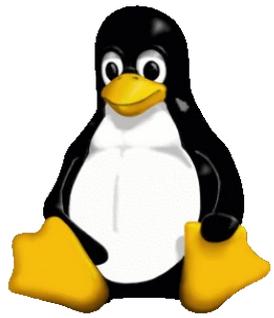
- Input and output of data is treated as a stream
- Data is treated as a stream
 - Data is stored in files (**input**).
 - Data is read from files (**output**).
- Devices are treated as *files* that provide a stream of data
 - Devices read data (**input**).
 - Devices write data (**output**).



STL! / unix / usr / group

Three Parts of stdio

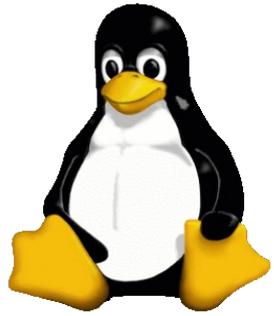
- Standard Input/Output (**stdio**)
 - Standard In (**stdin**) -
 - Standard Out (**stdout**) -
 - Standard Error (**stderr**) -



STL! / unix / usr / group

Designating Parts of stdio

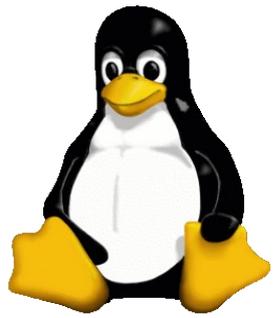
- Stdio is assigned by numbers
 - 0=**stdin** (*usually the keyboard*)
 - 1=**stdout** (*the normal output on your screen*)
 - 2=**stderr** (*the error messages you get on the screen*)
- NOTE:
 - the 0 for **stdin** is usually not needed
 - the 1 for **stdout** is usually not needed



STL! / unix / usr / group

Representing Parts of stdio

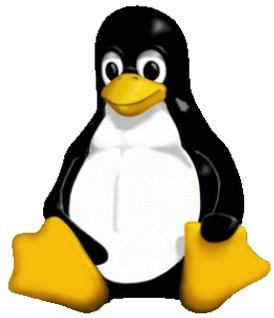
- stdio representation
 - 0=**stdin** - represented as a < (*left arrow*)
 - 1=**stdout** - represented as a > (*right arrow*)
 - 2=**stderr** - represented as 2>



STL! / unix / usr / group

Redirection Operators:

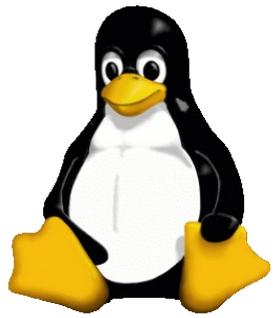
- `<`,
- `<<`,
- `>`,
- `>>`,
- `2>`,
- `2>>`, ???
- `>&`, ???
- `&>` ???



STL! / unix / usr / group

Redirector Functions

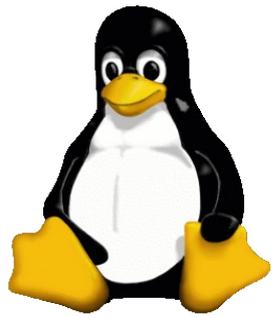
- > file - Redirects stdout to specified file
- 2> file - Redirects stderr to specified file
- >> file - Redirects stdout to specified file, appending output to the file if the file already exists
- 2>> file - Redirects stderr to specified file, appending output to the file if the file already exists
- &> file - Redirects stdout and stderr to the specified file
- < file - Redirects stdin to the specified file
- << text - Reads stdin until a line matching text is found, at which point end of file is posted
- cmd1 | cmd2 - Takes the standard input of cmd2 from the standard output of cmd1 (*also known as the pipe redirector*)



STL! / unix / usr / group

Using Redirection

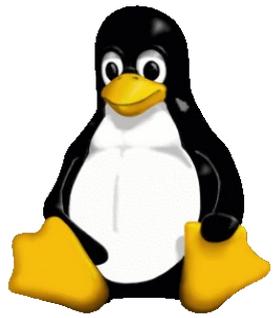
- example> \$ cat /etc/passwd 0< keyboard.txt
- example> \$ cat /etc/passwd 1> screen.txt
- example> \$ cat /etc/passwd 2> errors.txt



STL! / unix / usr / group

More Using Redirection

- example> \$ cat /etc/passwd 0< keyboard.txt
- example> \$ cat /etc/passwd 0<< keyboard.txt (HERE doc)
- example> \$ cat /etc/passwd 1> screen.txt
- example> \$ cat /etc/passwd 1>> screen.txt (append)
- example> \$ cat /etc/passwd 2> errors.txt



STL! / unix / usr / group

Related Commands of Interest

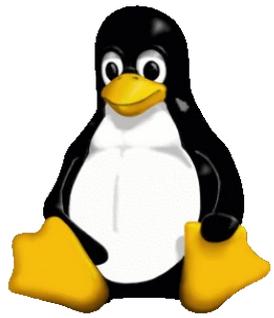
- **bash** (1) - GNU **B**ourne-**A**gain **S**hell
- **history** [builtins] (1) - bash built-in commands
- **ps** (1) - report a snapshot of the current processes
- **top** (1) - display Linux tasks
- **kill** (1) - terminate a process
- **pipe** (2) - create pipe
- **fork** (2) - create a child process
- **ln** (1) - make links between files
- **tee** (1) - read from **stdin** and write to **stdout** and **files**
- **script** (1) - make typescript of *terminal* session
- **replay** (1) - play back typescripts, using timing info



STL! / unix / usr / group

Standard Reference Commands:

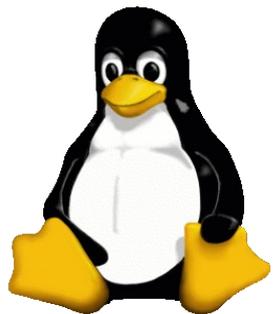
- man **bash**
- info/pinfo **bash**
- man/info/pinfo **stdio**
- man/info/pinfo **stdin, stdout, stderr**



`STL! / unix / usr / group`

Book References:

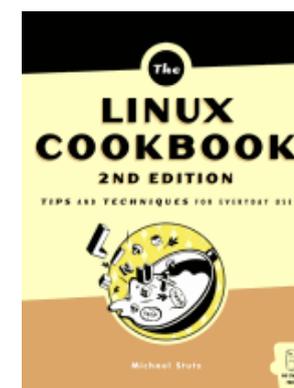
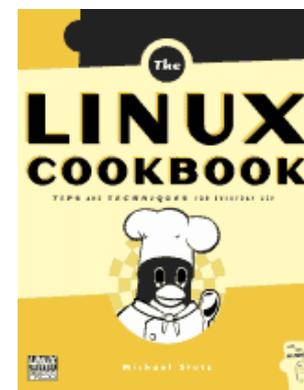
- The Linux Cookbook
- LINUX: Rute User's Tutorial and Exposition

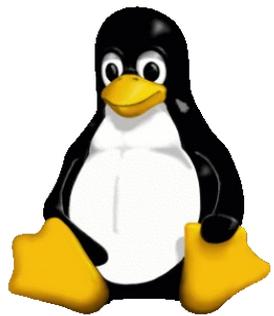


STL!/unix/usr/group

Book References (Continued):

- The Linux Cookbook
 - by Michael Stutz, No Starch Press, Aug 2001, \$29.95
 - ISBN: 1-886411-48-4
 - STLC: 005.432/S937L
 - <http://dsl.org/cookbook>
 - Debian & emacs yet excellent
 - Covers PostScript in detail



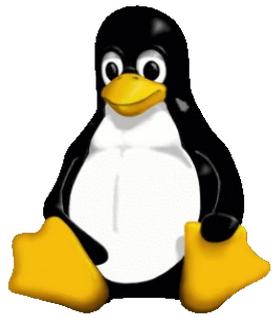


STL!/unix/usr/group

Book References (Continued):

- LINUX: Rute User's Tutorial and Exposition
 - By Paul Sheer, Prentice Hall, Oct2001, \$39.99
 - Debian & Red Hat, 650pp paperback+cd
 - ISBN: 0-13-033351-4
 - STLC: 005.4469/S541L
 - Don't worry about sec 2.1
 - Excellent Appendixes
 - Down loadable HTML
 - <http://www.icon.co.za/~psheer/rute-home.html>

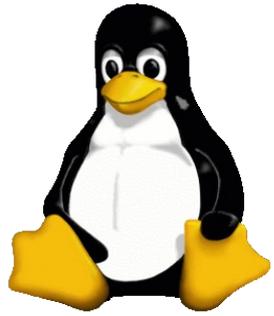




STL! / unix / usr / group

Internet References:

- Inspiration:
 - *Work the Shell - Get Started with Redirection* By Dave Taylor Dec'05
 - <http://www.linuxjournal.com/article/8563/>
- Others:
 - <http://www.tldp.org/LDP/Linux-Dictionary/html/index.html>
 - http://www.tldp.org/LDP/intro-linux/html/chap_05.html
 - http://www.brunolinux.com/02-The_Terminal/Redirections.html
 - <http://www.lowfatlinux.com/linux-redirection.html>
 - http://www.oreilly.com/catalog/debian/chapter/book/ch13_02.html
 - <http://freeengineer.org/learnUNIXin10minutes.html>



STL! / unix / usr / group

Simple Redirection

Questions?

Stan Reichardt

stan2006@sluug.org