Chapter 3: Software is Everywhere

Connecting Things

Cisco | Networking Academy®
Mind Wide Open™

Chapter 3 - Sections & Objectives

- 3.1 Programming
  - Explain the value of computer programs.
- 3.2 The Raspberry Pi Single Board Computer (SBC)
  - Use the Raspberry Pi for simple applications.
- 3.3 Building Models of IoT Systems in Packet Tracer
  - Use Packet Tracer to model IoT systems.

3.1 Programming

Cisco | Networking Academy®
Mind Wide Open™

3.1.1 What is Code?

- What is a Program
  - Code is a set of ordered instructions created to accomplish a specific task.
  - A bread recipe can be seen as a program.
  - Computer programs can be written in different programming languages.
- Programs are Everywhere
  - All computers need programs.
  - Operating Systems, firmware, and applications are examples of programs.
- Why Learn Code?
  - Programmers are valued in the job market.
  - Today, programmers may work on firmware, device drivers, mobile applications, web interfaces, data analysis, and more.
  - Programmers can create their own tools.

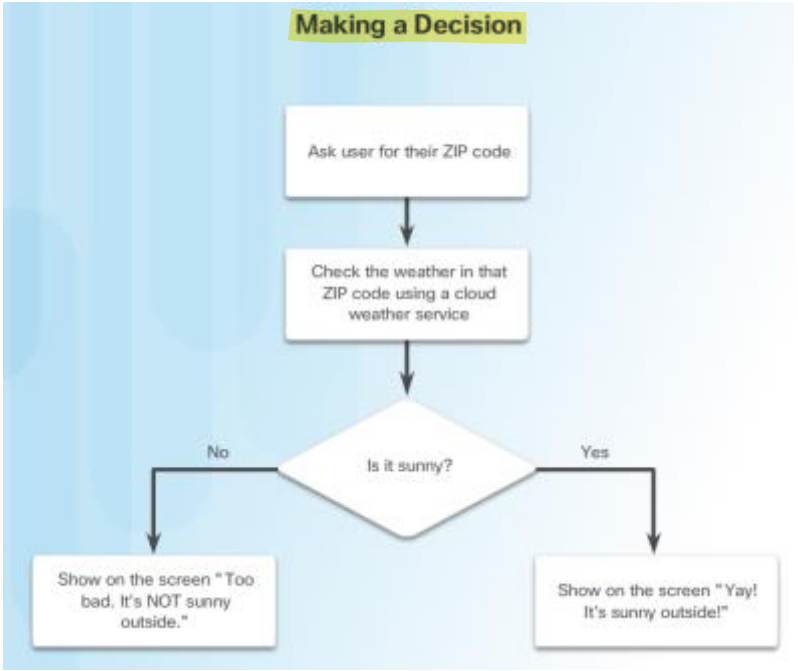
3.1.2 Code Does the Job!

- What Makes Up a Program?
  - Programs allow people impart logic to computers and are made out of logic structures.
  - IF-THEN, FOR Loops, and WHILE Loops are a few logical structures commonly found in programs.
- Interpreted Vs. Compiled
  - Interpreted languages rely on another program to read, parse, and execute the code.
  - Compiled languages rely on a compiler, another program, to turn the human-readable code into a binary executable code.
- Computer Languages
  - There are several different computer languages.
  - Some computer languages are better than others at certain types of tasks.
  - JavaScript, Python, Blockly, C, and Java are examples of computer languages.

```
#include <stdio.h>
int main()
{
  int year;

  printf("Enter a year to check if it is a leap year\n");
  scanf("%d", &year);

  if ( year%400 == 0)
    printf("%d is a leap year.\n", year);
  else if ( year%100 == 0)
    printf("%d is not a leap year.\n", year);
  else if ( year%4 == 0 )
    printf("%d is a leap year.\n", year);
  else
    printf("%d is not a leap year.\n", year);

  return 0;
}
```

3.1.3 Lending Intelligence

- IOT Devices and Data Processing
  - A common IoT application uses sensors to collect data.
  - Data is often not useful until it has been processed. Collected data is often transported and stored in the cloud for processing at a later date.
- IoT Devices Make Decisions
  - Software must be written and uploaded onto IoT devices to allow them to make decisions.
  - Decisions can be as simple as triggering an alarm or as complex as facial recognition.
- Software APIs
  - Application Program Interface (API) is a set of routines and software tools that facilitate one application communicating with another.
  - Different types of APIs exist: operating system APIs, application APIs, website APIs.
  - APIs allow applications to communicate, share data, or ask for specific services from another application.



Lending Intelligence – cont'd

- REST API

```
GET https://www.googleapis.com/calendar/v3/calendars/calendarID
```

  - REST APIs use HTTP based calls between applications to access and manipulate information stored on powerful databases.
  - Web resources used to be identified using a URL. Now resources can be any entity or thing that can be addressed: today's step goal, house temperature setting, glucose setting.
  - A unique Uniform Resource Identifier (URI) can identify an entity. A URI typically begins with a slash (/steps)
  - REST API requests trigger responses in well-defined formats such as XML or JSON

Lending Intelligence – cont'd

- Securing the Code
  - Devices should protect themselves from attacks that impair its function or allow it to be used for unintended purposes without authorization.
  - Devices should protect the private authentication credentials and key material from disclosure to unauthorized parties.
  - Devices should protect the information received, transmitted, or stored locally on the device, from inappropriate disclosure to unauthorized parties.
  - Devices should protect themselves from being used as a vector to attack other devices or hosts on the Internet.

3.2 The Raspberry Pi Single Board Computer (SBC)

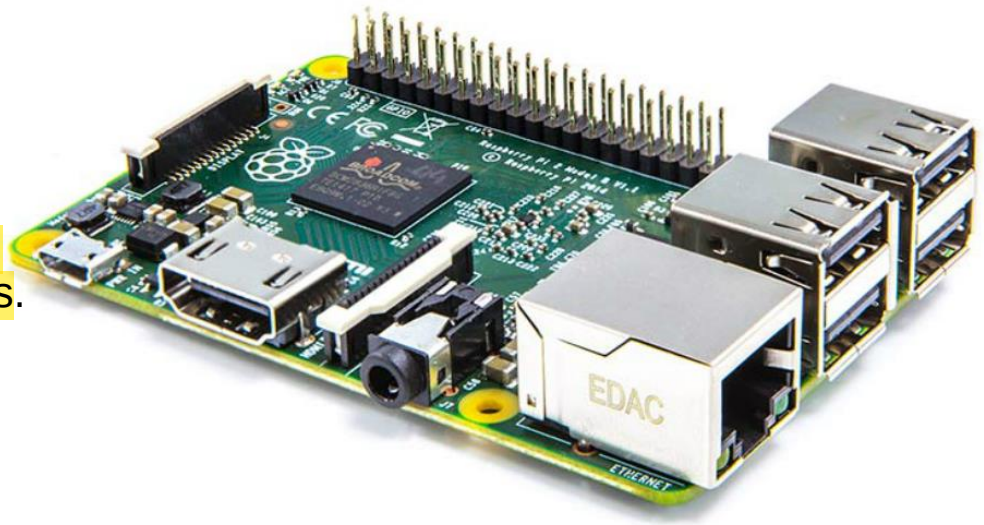
Cisco | Networking Academy®
Mind Wide Open™

3.2.1 Raspberry Pi Hardware

- The Raspberry Pi and its Ports
  - The Pi is a small and inexpensive computer.
  - It has a number of USB ports that can be used to connect various devices including keyboards, mice, external drives and cameras.
  - The Pi includes an 10/100Mbps Ethernet port and 40 GPIO pins, operating at 3.3V.
  - Other Pi ports include an audio out, a micro SD card slot, and a micro USB (used for power) connector.

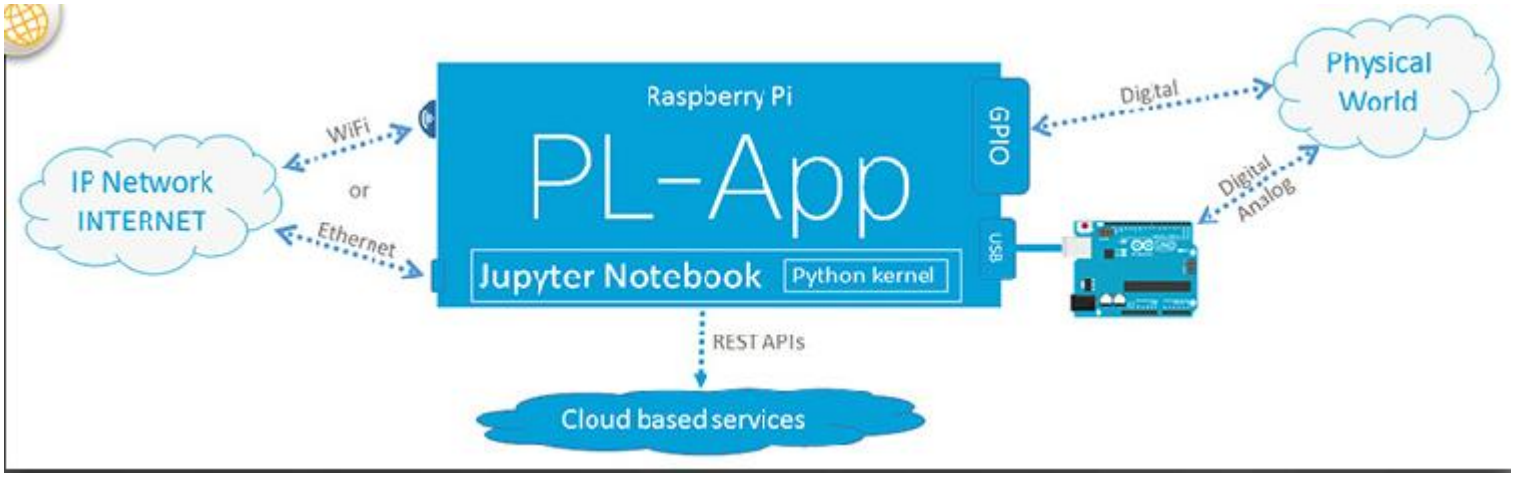
- The Pi3 also adds:
  - 1.2 Ghz 64-bit quad-core ARMv8 CPU
  - 802.11n Wireless LAN
  - Bluetooth 4.1
  - Bluetooth Low Energy (BLF)

- The Pi can run a number of operating systems, including Linux and Windows.



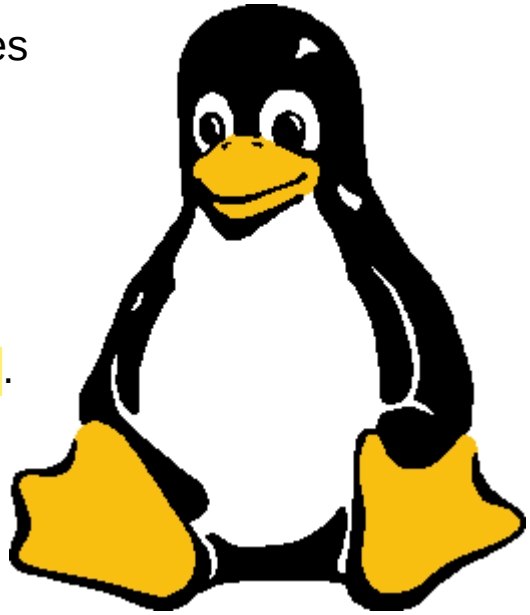
3.2.2 PL-App

- The Raspberry Pi can be accessed locally:
  - 1. Install an operating system image on the micro SD card.
  - 2. Place the card in the micro SD card slot of the RaPi.
  - 3. Connect a USB keyboard.
  - 4. Connect a monitor or TV using the HDMI port.
  - 5. Power the device with a power adapter.
- The Raspberry Pi can be accessed remotely using the PL-App



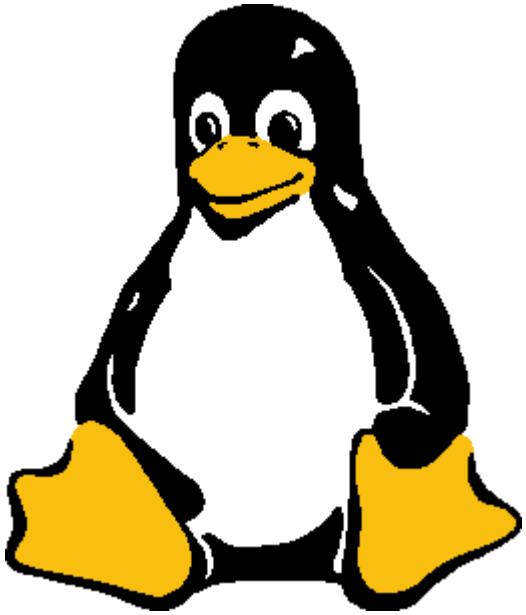
3.2.3 Using the Linux Operating System

- Understanding Linux
  - Linux is open source, fast, reliable and small and requires very little hardware resources to run.
  - Linux is part of several platforms; from wristwatches to supercomputers.
  - Linux distributions include the Linux kernel, plus a number of customized tools and software packages.
  - Debian, Red Hat, Ubuntu and Slackware are just a few examples of Linux distributions.
  - Raspbian is a Linux distribution based on Debian and created specifically for the Raspberry Pi.
- Accessing the Linux Shell
  - The Linux operating system can be divided into kernel and shell.
  - The shell is a command interpreter.
  - The shell is text based and also called CLI (command line interface



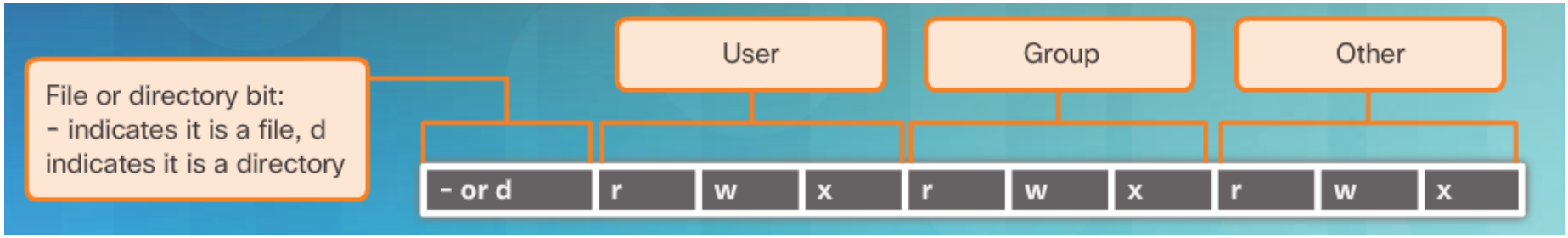
Using the Linux Operating System (Cont.)

- Accessing the CLI
  - The CLI can be accessed directly through a shell in non-graphical systems.
  - Bourne Shell (**sh**), Bash (**bash**), C Shell (**csh**), improved C Shell (**tcsh**), and Z Shell (**zsh**) are popular shells.
  - A terminal emulator application can be used to access the CLI in graphical environments.
  - Popular terminal emulators on Linux are **Terminator**, **eterm**, **xterm**, **console**, and **gnome-terminal**.
- Basic Linux Commands
  - Linux commands are programs created to perform a specific task.
  - To invoke a command via shell, simply type its name.
  - **grep**, **ifconfig**, **iwconfig**, **passwd** and **pwd** are a few basic Linux commands.
  - Commands can be piped together, using the output of one as the input of the other.



Using the Linux Operating System (Cont.)

- Process Managing Commands
  - In Linux, a process is any task or command being executed by the system.
  - PIDs are unique numbers assigned to processes for identification.
  - **ps**, **top** and **kill** are commands used to manage processes.
- File Permissions
  - In Linux, most everything is treated as a file.
  - File Permissions provide a mechanism to define permissions to files.
  - Possible permissions rights are **Read**, **Write**, and **Execute** and can be defined for the user who owns the file, the group, and other system users.
  - The root user can override file permissions.



The Raspberry Pi Single Board Computer (SBC)

Using the Linux Operating System (Cont.)

- Package Managers
  - Maintaining computer programs and their library dependencies manually is not scalable
  - Package managers facilitate the installation, removal, and upgrade of computer programs.
  - Package managers usually include user tools and a remote package repository.
  - The repository hosts software packages and their dependencies.
  - **dpkg** and **rpm** are popular package managers for Debian Linux and Red Hat Linux, respectively.
  - Raspbian includes **dpkg** and **apt** by default.

```
pi@raspberrypi ~ $ sudo apt-get install synaptic
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  aptdaemon aptdaemon-data docbook-xml gir1.2-atk-1.0 gir1.2-freedesktop
  gir1.2-gdkpixbuf-2.0 gir1.2-gtk-3.0 gir1.2-pango-1.0 gir1.2-vte-2.90
  libcairo-perl libglib-perl libgtk2-perl libpango-perl librarian0
  libvte-2.90-9 libvte-2.90-common lsb-release python-apt python-apt-common
  python-aptdaemon python-aptdaemon.gtk3widgets python-chardet python-debian
  python-defer python-gnupginterface python-pkg-resources python-pycurl
  python-software-properties rarian-compat sgml-data
  software-properties-common software-properties-gtk unattended-upgrades
Suggested packages:
  docbook docbook-dsssl docbook-xsl docbook-defguide libfont-freetype-perl
  libgtk2-perl-doc lsb python-apt-dbg python-gtk2 python-vte python-apt-doc
  python-distribute python-distribute-doc libcurl4-gnutls-dev
  python-pycurl-dbg perlsgml w3-recs opensp libxml2-utils dww deborphan
  apt-xapian-index bsd-mailx mail-transport-agent
The following NEW packages will be installed:
  aptdaemon aptdaemon-data docbook-xml gir1.2-atk-1.0 gir1.2-freedesktop
  gir1.2-gdkpixbuf-2.0 gir1.2-gtk-3.0 gir1.2-pango-1.0 gir1.2-vte-2.90
  libcairo-perl libglib-perl libgtk2-perl libpango-perl librarian0
  libvte-2.90-9 libvte-2.90-common lsb-release python-apt python-apt-common
  python-aptdaemon python-aptdaemon.gtk3widgets python-chardet python-debian
  python-defer python-gnupginterface python-pkg-resources python-pycurl
  python-software-properties rarian-compat sgml-data
  software-properties-common software-properties-gtk synaptic
  unattended-upgrades
0 upgraded, 34 newly installed, 0 to remove and 4 not upgraded.
Need to get 8,825 kB of archives.
After this operation, 26.9 MB of additional disk space will be used.
Do you want to continue [Y/n]? Y
Get:1 http://archive.raspberrypi.org/debian/ wheezy/main gir1.2-atk-1.0 armhf 2.
8.0-2rpi2 [61.2 kB]
Get:2 http://archive.raspberrypi.org/debian/ wheezy/main gir1.2-freedesktop armh
f 1.36.0-2rpi2 [20.8 kB]
```