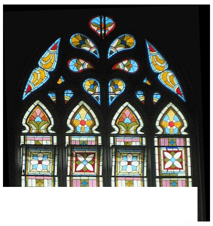
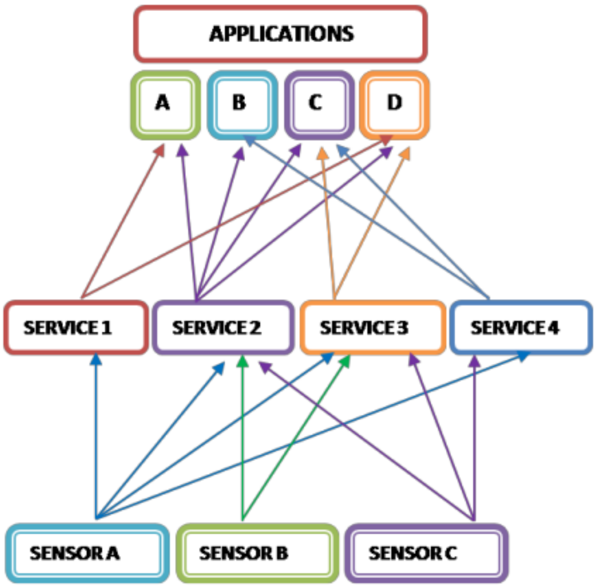
1. What is IoT?





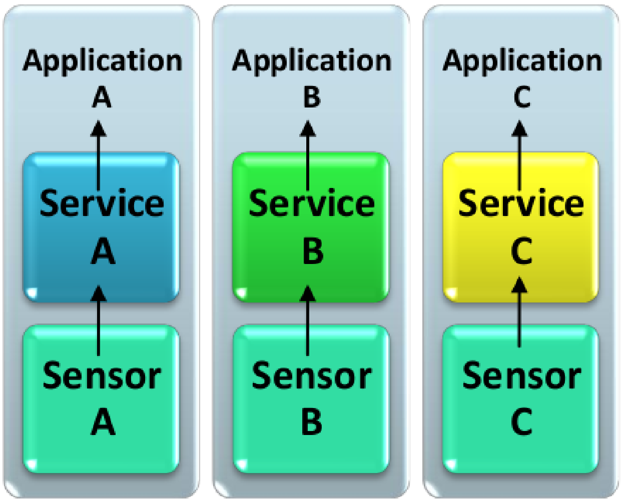


Figure 6
Non-interoperable IoT

Theory of Internet of Things or “I o t”

Introduction:

There is a difference between the figures above. So far as I can tell that is why the “Internet of Things” is a separate construct.

The ‘Sensor’ – ‘Server’ – ‘Application’ chain requires intelligent (often microprocessor driven) control(s). Note that there is NO requirement (in definitions) for feedback looping.

The current definitions suggest that the field is too big with too many options/ paths.

A basic IoT could be;

A device responding to a mobile phone command with the device maybe providing feedback (or forward) re the sensed situation. That involves at least;

>
> 1. **sensors** (example movement infra-red sensor for passing cats)
>
> 2. **response** software (maybe Raspberry Pi Zero W based)
>
> 3. response **hardware** - wi-fi and/or other remote comms. Maybe just the Pi Zero W or something. More often a chain of device components.
>
> 4. mobile phone alerting **signal**. Ring, display, await response.
>
> 5. human **response** options (eg touch screen to electrocute the cat)
>
> 6. **communicate response** to the IoT device (a chain of hard / soft ware tied to a house mostly)
>
> 7. action the **remote command local** to the IoT (240v to metal grid cat is on?)
>
> 8. Waste disposal (cat?) and Climate Change implications.
>

Maybe

Here is a simple variation. Q – Is this an IoT? Maybe...

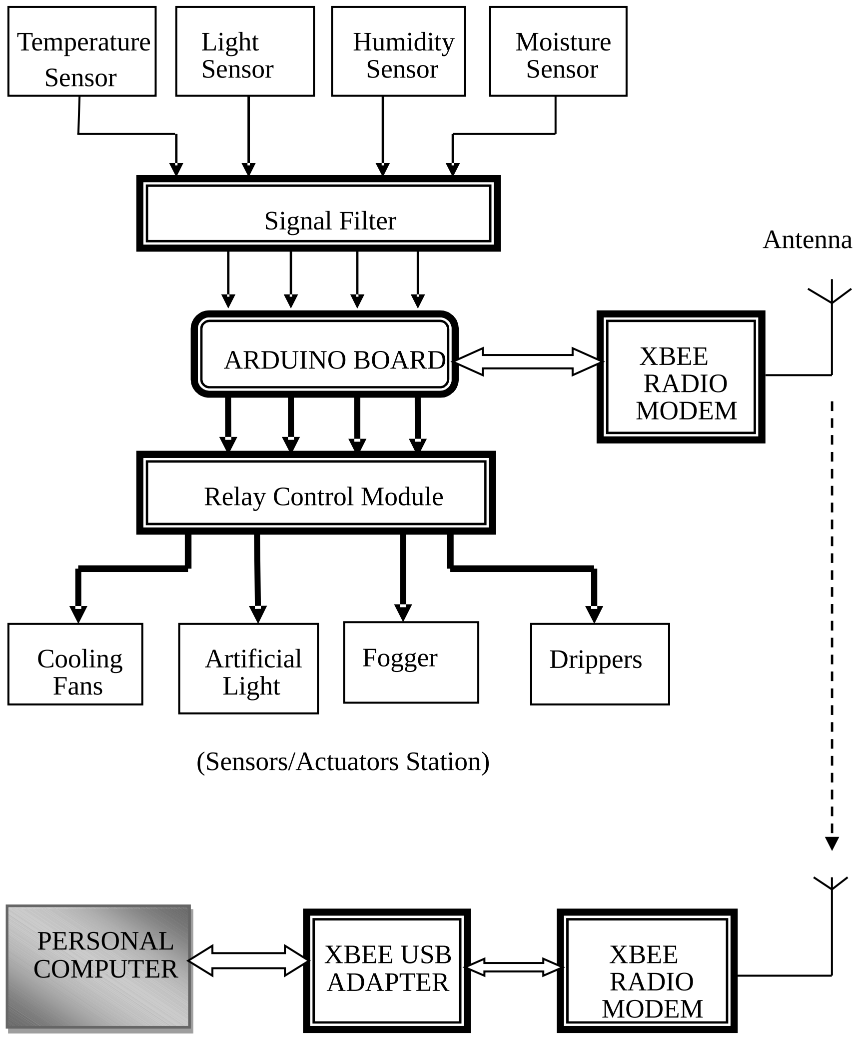


Figure 1. Block Diagram of Automated Greenhouse Control System (Remote Monitoring Station)

2.4 Software Design

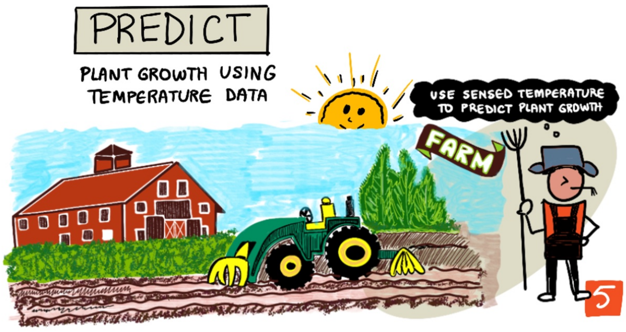
ref: An Automated Greenhouse Control System by E.Enokela and R.Otholabe, 2015

Horticultural Example of the Internet of Things.

[*Acknowledgement – these diagrams are sourced from a Microsoft tutorial on IoT. You can find it with Google if you must*]

The MS tute tries to demonstrate an upmarket use for their **IoT in the cloud** software. This example below (one of 24 tutes) is taken from a very common tool used in horticulture. The idea is that one can **predict the ripening time** of fruit and vegetable crops by sensing the temperature and then adding 'useful' temperature up to get to a point where a crop should be ready to harvest.

In real life this IS USED by the wine industry to guestimate grape harvest times.

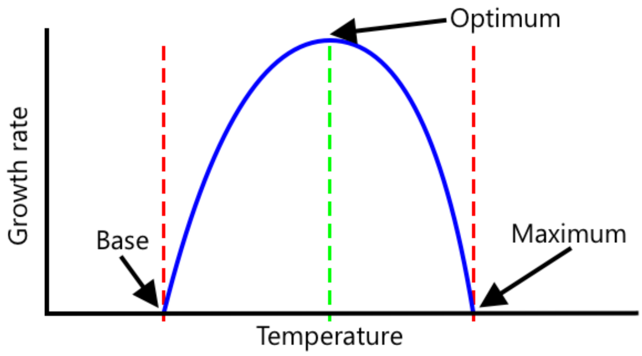


Source:Microsoft image The diagrams below shows the MS idea visually (© MS). 'IoT_MS_Azure_project.doc' in Downloads in Ryzen.

Here is the equation to work out the "Growth Degree Days" you can see that a T-max is used where more heat than actually inhibits growth.

Vines in Australia slow down above about 32 °C. It is not "STOP"!.

Below about 10°C most plants stop abruptly. That is "T base" here. Using the maximum and minimum and BOM standard weather station readings works for vines. Then all you do is calc for each day of the growing season.



$$GDD = \frac{T_{max} + T_{min}}{2} - T_{base}$$

$$GDD = \frac{16 + 12}{2} - 10 = 4$$

$$GDD = \frac{25 + 12}{2} - 10 = 8.5$$

So for this day we get a total of 5 Degree-Days.

Then you add them for the fruit growing – ripening season to get a total of about 1,000 dd for vines in Australia.

It works for planning **where to grow** a variety. Some varieties need to accumulate more heat dd than others.

This model can, and is, used by Australian horticulture to plan harvest labour and so on.

A simple **Internet of Things** system of sensors, data transmission and human interface is used.

Note the sensor data can be from multiple sensors and transmitted from multiple transmission system to multiple interfaces (and it is in large viticuture companies because the various company wineries need to “know” when, where and what quality of grapes are coming through.

In Australia we use sensors within the grape sampling auger to “pre-feed” grape sugar / acid to the QA lab which can then accept or reject the picked bin of grapes.

2. Basic Options

Use of Microprocessors

For businesses, Raspberry Pi, Arduino, etc are not a replacement of heavy IT infrastructure, however, they are efficient for these things;

a) Setting up a low-cost server for handling light websites
b) Testing IP connectivity and computer components
c) Developing software concepts with a multiple-language support
d) Building internal dashboards to visualize data, and more.

“With an in-built quadcore processor, Raspberry Pi can serve as the “Internet Gateway” for IoT devices. Powered by a cloud network, Pi acts as a web server for uploading and transiting sensor data on IoT platforms. Custom code, an operating system, a Python library, and a WWW network are all it takes to use Pi Computer as a web server to the IoT.”

===== References =======

Microprocessors – various brands with sub-sets.
    Arduino and copies
    Raspberry Pi – Pi1, 2, 3, 4 and now Raspberry Pi 5
    Texas Instruments, Bosch, etc – sensors
    A-Tiny’s
    See Jaycar, Sturt St Aztronics, Altronics, Element-14,.   Core-Electronics – Newcastle. Online.

===========ooo===========

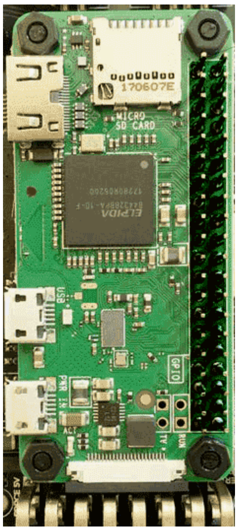
Wireless

Usually Wi-Fi is used locally for the communication carrier. Others, such as ZigBee use other radio frequencies (amateur bands?) and then some IoT uses long-wave freqencies which result in slow but distant communication (water tanks).

My current device is a **Raspberry Pi Zero W** model. It has a RISC chip design with enough capability to run a ‘Micro-Python’ language subset. That then allows it to more easily be linked to other larger parts of the system. Learning Python only once rather than assembly for an I/O chip and then upward to C++, Rust, Go, etc makes sense. Small size allows for embedding in devices and low power with wake-up software facilitates low power drain.

BUT

Main problem is that developing and initial use requires peripherals – screen, keyboard, power source, voltage step-up/down, data storage, etc. It is very messy!



Pi Zero W for IoT

3. Playing with IoT – options

ISSUE – Getting the message out.

In practice the computer that receives a message needs to “communicate” it. That mostly boils down to a output buffer that converts the low power microprocessor signals into some useful signal.

That requires a power source (often common with the IoT device), connections (plugs and wires), an indicator LED up to coloured display, often a High voltage relay, feedback of status to the microprocessor. Messy!

ISSUE – Lack of common language.

Now there are perhaps ten main languages in use. MBASIC, Python (and ten+ variants), JSON, HTML, Perl, C and C++, Arduino assembly, RISKs, Rust, FORTRAN, and so on. Cross-compiling / communicating is an issue. Messy!

Now there are a number of ‘radio’ systems in use as the transport layer. These include Bluetooth LE, Light (laser and flashing), Near field and RFID systems, Wi-Fi developed by CSIRO, Zigbee low power, low data local, 4G and 5G mobile phone systems, low power long range eg LoRaWAN. Also Ethernet v6 cables and 240V systems like ETSA uses.

ISSUE – Boxes.

There is no standard for the containment of IoT devices and connections. Could there be? Messy! With power variation 12 to 5 to 3.3volts. What next of my box of stuff will be obsolete?
Or “Why Bother?” - just use an old laptop! Has it all.

ISSUE – Speed

The development speed of the microprocessors in particular is an issue with users like us. I have docs on various Arduino systems, Various Raspbery Pi systems - There is a ‘crazy’ variation in Python / uPython instruction sets (I think). Atiny boards, ***. Hard for users to use due to slight but important variations in code. Best example this week is Thonny – micropython to R Pi interface. Ver 3.xx to V 4.00 Very Messy!

***

ISSUE – Artificial Intelligence

“AI” rapid development is resulting in obsolescence in hacker control of projects (ie Silicon Chip type projects). The number of types of maker projects is declining because of AI built in to the components. Think of audio amp on a chip. Why bother to solder anything up. Loss of knowledge. Loss of need to think. Not Messy!

ISSUE – Hidden communication.

Example, facial recognition AND transmission is not visible. Cf IoT things to people ratio. In 1999 0.08 : 1 now > 2.00 : 1. Example is Bunnings. Who do they sell/give the data to? Not messy AND not visible. Another example is the inkjet printer sensors.

PLUSSES

Software – lots of choice
Developments systems – Arduino, Thonny, etc – all free.
Documentation – masses of very good stuff. My favourite is Core Electronics in Newcastle. Also the “Read the Docs” series are excellent.
Cost - $5 to $200 for a working IoT
Robots – potential use in aged care.
Home automation – voice control. “Siri”
Wearable technology – “nice” ornaments that flash
Medical – not only person to home modem for person data transfer but also online consultation (+- robot)
Medical – In body IoT devices. Eg Creatinine level and kidney failure
Use of RFID devices fabricated on paper. BUPA surveilence.
Farming – long been used in horticulture eg dried fruit sorting.
Context aware automation

ISSUE – Privacy

“What can you say?” (that wont be recorded?)

ISSUE – Network Architecture

There was a problem with lack of nodes with the IPv4 system. IPv6 is touted to fix that. IoT devices don’t move much data so less problem. There is a potential problem with CHAOS (in the Systems Analysis maths concept).

ISSUE – Social IoT

Systems designed to be reactive and adaptive to circumstance. Control by AI maybe. Example road traffic control systems. Sensors on / off as a car goes along. Un-Messy?

ISSUE – “Server” variations

Arduino MQTT client – server model
cf https/docs/arduino/tutorials/uno-wifi-rev2/mqtt/??
PicoMite – MMBASIC system with Silicon Chip support
Bluetooth (and bluedot py for Android systems)
FTP on mobiles
Python3 http server to local host 8000
Jaycar #XC4508 radio nRF24 is a 2.4Ghz system
Queensland developers use long wave radio for water troughs and tanks.

4. Philosophy?

I note that LINUX people had the software that we use sorted out and available as BASHish commands circa 1985. Same protocol being used for the Lunar rovers in part (cf NASA) !!

REFERENCE bits – no order at all!

see "IoT_MS_Azure_project.doc" in Downloads in Ryzen. Microsoft text
For businesses, Raspberry Pi is not a replacement of heavy IT infrastructure, however, Pi is efficient for-

a) Setting up a low-cost server for handling light websites
b) Testing IP connectivity and computer components
c) Developing software with a multiple-language support
d) Building internal dashboards to visualize data, and more.

How can IoT Applications use Raspberry Pi?
With an in-built quadcore processor, Raspberry Pi can serve as the “Internet Gateway” for IoT devices. When powered by a local or cloud network, Pi or similar may act as a web server for uploading and transiting sensor data on IoT platforms.

Custom code, an operating system, a Python library, and a local or cloud network are all it takes to use Pi or similar computer as a web server.

An IoT project follows a simple design, implementation, and modification route, making it highly suitable for IoT applications.

Example: IoT implementation with raspberry Pi
Deploying such a small setup can enable individual or businesses to build IoT applications such as-
1) Home automation
2) Automated traffic control
3) Corporate surveillance and security systems
4) Medical or heart monitoring devices
5) Miniature scale industry, and other pilot projects.

====================
ref: https://artificialintelligence.oodles.io › blogs › Iot-Applications-With-Raspberry-Pi - A blueprint of IoT implementation using Raspberry Pi is demonstrated.

https://magpi.raspberrypi.com › articles › 10-amazing-raspberry-pi-iot-projects

https://github.com/microsoft/IoT-For-Beginners

see the file Nov9th and 7th
============ooo===========
https://artificialintelligence.oodles.io › blogs › Iot-Applications-With-Raspberry-Pi. Tracking to contactless thermal scanning and weather tracking to surveillance and automated irrigation. This blog post discusses the many benefits that Raspberry Pi brings to the table in IoT app development.
see https://artificialintelligence.oodles.io › blogs › iot-applications-with-raspberry-pi4 Next-gen IoT Applications With Raspberry Pi - AI Oodles
1111[1]
Https://Www.Raspberrypi.Com › News › Learn-The-Internet-Of-Things-With-Iot-For-Beginners-And-Raspberry-Pi. Learn the Internet of Things with "IoT for Beginners" and Raspberry Pi

https://magpi.raspberrypi.com › articles › 10-amazing-raspberry-pi-iot-projects

https://www.raspberrypi.com › news › getting-started-with-iot. It works with Ubidots out of the box and has a number of tutorials that will help you learn common IoT tools like using Node-RED on the Pi (including a PIR counter)
https://github.com/microsoft/IoT-For-Beginners. sfile Nov9th and 7th., 2023

1 LINUX Internet of Things Feb, 14, 2024

****====   ====*****

>>==================***************=======================

Microsoft tutorial on IoT – 25 units

Demonstrate Pi Zero W, glider software, other Rasp Pi’s, “Raspberry Pi Pico Getting Started” printout.

Note about Thonny 4.0 versus ver 3.* , Thonny BASH script, NASA software catalog.json FYI

*** Canvas.html ***

IoT is like an iceberg. Take care!